SpaceStat TUTORIAL

A Workbook for Using SpaceStat in the Analysis of Spatial Data

by

Luc Anselin

University of Illinois, Urbana-Champaign
Urbana, IL 61801
luc@spacestat.com

Copyright © 1992 by Luc Anselin. All Rights Reserved.

The original version of this Tutorial was distributed as Technical Report S-92-1 of the National Center for Geographic Information and Analysis, University of California, Santa Barbara, CA.
INTRODUCTION

Empirical work in many scientific fields is based on data for which the location of the observations is an important attribute. This type of data may be referred to as spatial data. The observations consist of a single cross-section of spatial units, or of a time series of cross-sections (often referred to as pooled cross section and time series data, or panel data). Some examples of the range of empirical studies where spatial data are encountered are: the analysis of patterns of tree growth in forestry; the study of plant diversity in ecology; the analysis of geological strata; the study of urban crime by census tract; the analysis of ancient settlements; the study of crop yields; the analysis of social networks; the quantitative study of international relations; the analysis of regional labor markets; the study of regional mortality differentials; and the analysis of the spread of diseases (see Anselin, 1993, for a more extensive illustration).

The distinctive characteristic of the statistical analysis of spatial data is that the spatial pattern of locations and values, the spatial association between them, and the systematic variation of phenomena by location become the major foci of inquiry. In addition to being of interest in and of itself (from a geographer’s perspective), the spatial pattern in the data causes a number of measurement problems, referred to as spatial effects, such as spatial dependence and spatial heterogeneity, which affect the validity of traditional statistical methods. This has given rise to an increasingly sophisticated body of specialized techniques, developed in the fields of spatial statistics and spatial econometrics (for recent overviews, see, e.g., Cressie, 1991; and Anselin, 1990, 1992).

In spite of the methodological advances in the field, and of solid indications that spatial effects matter, most empirical work that uses spatial data still fails to take its distinctive characteristics into account. This is illustrated by findings in the reviews of recent journal articles in Anselin and Griffith (1988), and Anselin and Hudak (1992). Similarly, the technologies of remote sensing and geographic information systems (GIS), while powerful means to collect, organize and display spatial data, currently still fall far short from allowing a sophisticated statistical analysis of spatial pattern and spatial association (see Anselin and Getis, 1992). Also, none of the well known commercial statistical and econometric software contain even the most rudimentary tests for spatial effects, let alone estimation routines for spatial models.

This dearth of specialized software is often cited as one of the main reasons for the lack of acceptance of spatial data analysis techniques by empirical analysts (e.g., in Haining, 1990). While the situation has much improved in the past few years, particularly in the form of macros written for existing statistical packages (see Anselin and Hudak, 1992, for a review), up until now, there was no comprehensive software package that covered a reasonable range of familiar techniques in spatial statistics and spatial econometrics.
I developed **SpaceStat** to provide access to these techniques to as wide an audience as possible and in a user-friendly manner as possible. **SpaceStat** is NOT a comprehensive statistical package in the traditional sense. I intentionally omitted many tests and methods commonly available in commercial software. My objective was to focus on those techniques that would take considerable investment in terms of programming, instead of replicating what is already available. The main purpose of **SpaceStat** is to be an aid in research and in teaching undergraduate and graduate courses in “Statistics for Geographers.” The availability of software by means of which the spatial techniques can be readily illustrated allows the instructor to introduce the “spatial” aspects of the data up front. In my own experience in teaching such courses, I have found that it is crucial for geography students (and others dealing with spatial data) to acquire an intuitive understanding of the special features of spatial data early on in their careers, especially now that easy access to powerful GIS has made “spatial analysis” such a common interest. No more excuses ...

The lineage of **SpaceStat** goes back to Fortran routines written as part of my doctoral dissertation almost 15 years ago. Some of these routines were earlier made available in technical reports (Anselin, 1985 and 1986). The main development occurred over the past five years, originally as part of an effort (funded by the National Science Foundation) to improve the dissemination of spatial statistical techniques. This development was carried out in various versions of the GAUSS mathematical software and some earlier and partial versions were distributed to a limited degree (Anselin, 1989). During the past two years, a major effort was undertaken to make the software more user-friendly, to develop a consistent user interface, make the code faster and largely (I hope) bug-free, and to create a limited interface with GIS software.

**Acknowledgments**

Commercial software referred to in this tutorial is copyrighted: GAUSS is copyrighted by Aptech Systems, Inc., IDRISI is copyrighted by Ron Eastman and Clark University School of Geography, and ARC/INFO is a trademark of Environmental Systems Research Institute.

The development of various aspects of **SpaceStat** was supported in part by a number of grants from the National Science Foundation: the early work by grants SES 83-09008 and SES 86-00465, and the bulk of the more recent developments by grants SES 87-21875 and SES 89-21385. The C code for the GIS interface was written by Rustin Dodson, supported by the National Center for Geographic Information and Analysis (NCGIA) under NSF grant SES 88-10917. My writing of the manual/tutorial and the various other supporting materials was supported by NCGIA as well. Without this considerable financial help and the release time it made available, **SpaceStat** would never have been completed.
Over the years, I received many helpful, supportive and constructive comments (and also some less helpful ones) from a large number of people. First and foremost among these is Sheri Hudak, who suffered through innumerable revisions and changes (including a major change in the user interface), but persisted in using SpaceStat for her thesis, becoming an invaluable guinea pig in the process. Other graduate students who made many helpful suggestions are Rusty Dodson and Serge Rey. Raymond Florax, John O'Loughlin, Ayse Can and Waldo Tobler were users of early versions who pointed out problems and offered many suggestions. Three cohorts of students were subjected to versions of the software in the classroom and in workshops, which provided a very effective testing ground for the user interface. It also made me aware of the tremendous difference between writing software for one's own use and writing it for a wider audience. Arthur Getis and Dan Griffith commented on earlier drafts of the tutorial. Emily, Emma and Lucie suffered patiently.

**Standard Disclaimer**

No warranties, expressed or implied are made by the author that the computer program, documentation or tutorial are free of error. The software is not warranted for correctness, accuracy, or fitness for a task. Users rely on the results solely at their own risk; no responsibility is assumed in connection therewith.

**References**


Anselin, Luc (1993). *Spatial data analysis with GIS: an introduction to application in the social sciences*. National Center for Geographic Information and Analysis, University of California, Santa Barbara.


PART I

PRELIMINARIES
CHAPTER 1

GETTING STARTED

1.1 Installing SpaceStat

1.1.1 Hardware and Software Requirements

*SpaceStat* is written and compiled in GAUSS, a matrix language and statistical system from Aptech Systems, Inc. *SpaceStat* may be run either as a GAUSS program, or as a freestanding program, by means of the GAUSS runtime module. This module is included with the version of *SpaceStat* distributed by NCGIA. Running *SpaceStat* in GAUSS is the preferred approach, but using the runtime module may be your only alternative if you don't have GAUSS installed on your machine.

The current version of *SpaceStat* is written in GAUSS 2.2 and runs exclusively on IBM PS/2 or compatible machines equipped with a 386 cpu and accompanying mathematical co-processor (387). This includes 386SX (with 387SX) and 486 machines. *SpaceStat* will not run if the mathematical coprocessor is not installed. It will also not run on PC simulators available for some unix workstations, unless the simulator is for a 386/387 (most simulators to date are for a 286 cpu). *SpaceStat* may be executed as a DOS program in WINDOWS, although this will be considerably slower than in DOS itself.

The memory requirements for using *SpaceStat* depend on the size of the data set you wish to analyze. The compiled program is only about 270K, but its workspace will occupy as much RAM as is made available in GAUSS, or as is present on your system (when you use the runtime module). At least 4 megabytes of RAM is recommended in order to implement all features of the program for small data sets (up to 250 observations). For larger data sets much more RAM will be needed, since GAUSS stores all matrices in double precision. A good rule of thumb to determine needed RAM is to take three times the square of the number of observations, and to multiply the result by 8. This gives you the number of bytes needed to store spatial weights matrices in the computation of tests for spatial dependence and in the estimation of spatial regression models. For example, for a data set with 500 observations, this would yield $3 \times 250,000 \times 8 = 6,000,000$ or approximately 6 megabytes that must be available as workspace (in addition to the memory required by GAUSS itself and by the operating system).

Note that GAUSS also supports virtual memory. When you run *SpaceStat* with GAUSS in this configuration, the limitations in RAM are no longer binding, since memory is swapped to disk. In this way, you will be able to analyze much bigger data sets, although execution will be considerably slower.
1.1.2 The Install Batch Program

You must use the INSTALL.BAT batch program on the SpaceStat distribution disk to decompress the SpaceStat files and copy them onto your machine. You must first have created a directory for SpaceStat. The installation program assumes C:\SPACE as the default, but you can specify any directory name.

To run the installation program, first insert the distribution disk in one of the floppy disk drives on your machine. Next, change the default directory to the name of the drive in which you inserted the diskette, i.e., either A:\ or B:\. At the prompt, type:

\>install normal

and press the Return or Enter key. This will install SpaceStat on the default directory C:\SPACE. Alternatively, you may explicitly specify the name of the directory in which you want the SpaceStat files to reside. For example, if you wanted SpaceStat to be in the directory E:\ANALYSIS, you would type:

\>install E:\ANALYSIS

followed by the Return key.

If the installation program cannot detect the directory you specified (or the default directory), you will be prompted to abort installation and to create the directory. When no errors are detected, the installation program will decompress all files on the distribution disk, create subdirectories for the example data sets and copy all files onto your machine.

1.1.3 Files Needed

In order to execute SpaceStat, you must have the following program files: SPACE.BAT, SPACE.G32 and RAS2WM.EXE. In addition, if you don’t have GAUSS installed on your machine, you will need the GAUSS runtime files GSRUN386.EXE, GSRUN1P.OVR, GSRUN2P.OVR, and STARTRUN.G32, and the functions EIGRS.GXE, EIGRG.GXE, INDEXCAT.REX and UNIQINDEX.REX.

The above files must always be in the same directory. This can be your current directory or a directory included in the PATH command in the AUTOEXEC.BAT file. The latter is the preferred approach and will allow you to analyze data sets that are in any directory.

SpaceStat does not use graphics, but requires that the device driver ANSI.SYS is installed in your CONFIG.SYS file. Otherwise, the extended ascii characters used in the SpaceStat menu screens will appear strange and your machine may lock up as SpaceStat executes. If this happens to you, insert the following line in the CONFIG.SYS file:

DEVICE=ANSI.SYS
and make sure to include the ANSI.SYS driver in your root directory (or specify the full path-name in the DEVICE line). If you are unsure about how to alter the CONFIG.SYS file, refer to your DOS manual.

1.2 Running SpaceStat

1.2.1 Running SpaceStat in GAUSS

The most stable way to take advantage of all the features of SpaceStat is to execute the program as a GAUSS function, from within GAUSS. This avoids some of the problems that occur when using the runtime module.

After you start GAUSS and have specified the workspace requirements (in GAUSS), you execute SpaceStat by typing the following at the GAUSS prompt:

```plaintext
run space
```

followed by a Return. Since neither GAUSS nor SpaceStat are case sensitive, you may also type the above commands in upper case.

Next, a title screen appears, which lists the version number and revision date (this is important information when you report problems). By pressing any key you get to a welcome screen that lists the main menu items at the top. This top row remains shown in all of the following submenus of SpaceStat as well.

1.2.2 Running SpaceStat with the GAUSS Runtime Module

The small batch program SPACE.BAT invokes the GAUSS runtime module and executes SpaceStat. You start it from the DOS prompt by typing:

```plaintext
space
```

followed by a Return (this is not case sensitive). This will clear the screen and a welcome message will appear, as above in 1.2.1.

If you prefer, you can also invoke the runtime module explicitly to execute SpaceStat. In that case, you would type the following at the DOS prompt:

```plaintext
gsrun386 space
```

followed by a Return. The program proceeds in the same fashion as described above.

1.2.3 Getting Out

From any menu in SpaceStat you can exit the program by pressing the ALT-Q key combination. **Never** use the ESC key to try to abort the program!
1.2.4 Common Problems

*SpaceStat* has been extensively tested and is now fairly stable. Of course, no-one can ever claim that a program is entirely bug-free and reports of problems are greatly appreciated. *SpaceStat* is designed to detect many common errors. If such an error is encountered (e.g., file name incorrect, variable name incorrect, variable name not specified, etc.), you will hear a beep, followed by a brief error message. If you press *Return* at this point, you will end up in the latest submenu from which you entered commands.

A few problems may be due to hardware limitations. For example, if you attempt to analyze a data set whose memory requirements exceed what you have available, *SpaceStat* will crash. In such instances, you will see the following GAUSS error message: *Insufficient Workspace Memory*. The only practical solution to this problem is to add more RAM to your machine.

A number of problems you may encounter when running *SpaceStat* may be due to the limitations of the runtime module that is included with the program. As much as possible I have discussed them with the relevant material in this tutorial.

1.3 Prerequisites for the Tutorial

I assume that you are somewhat familiar with statistical reasoning and with techniques for the analysis of spatial data. Only a limited amount of background information will be covered here. This tutorial stresses the use of *SpaceStat* as a software product, but does not intend to provide you with a comprehensive introduction to spatial data analysis. You may find the illustrations in Anselin (1992 and 1993) useful as accompanying materials to this tutorial. These papers go through various steps in the analysis of spatial data and provide example data sets. Each chapter in the tutorial will also list a small number of pertinent references.

There are now several texts in which you can find extensive discussions of spatial data analysis, spatial statistics and spatial econometrics. For example, at an introductory level, you could refer to the materials in Griffith (1987) or Odland (1988). At an intermediate level, Upton and Fingleton (1985) or Haining (1990) may be more appropriate. You may find a rigorous and more advanced discussion of many of the topics covered in this tutorial in Cliff and Ord (1971, 1981), Anselin (1988), Griffith (1988), and Cressie (1991).

On average, I expect it may take you 20 to 30 minutes to complete each example in the tutorial. However, this is only a rough estimate and if you are not very familiar with spatial statistics (and/or with computers) it may take you longer, especially if you need to refer to technical background materials.
1.4 Example Data Sets

1.4.1 Data Sets Included

Throughout this tutorial, I will be referring to three data sets to illustrate the various techniques. The main examples will be based on data for 49 neighborhoods in Columbus, Ohio. This data set contains information on crime (combined residential burglaries and residential thefts per thousand households), income per capita (in $1,000s), housing value (in $1,000s), and neighborhood centroids. There is also an indicator variable that distinguishes between neighborhoods east and west of the major north-south transportation axis. This data set is the same as the one used for the examples in Chapter 12 of my *Spatial Econometrics* book (Anselin, 1988). The various data files are included in the \COLUMBUS directory.

Another example that will be used throughout is from Anselin (1992). It consists of 7 by 7 square raster grids with 10 arc-minute spacing from the *Global Change Database* (NOAA, 1990). The 49 observations are roughly situated around the border between the Central African Republic, Sudan and Zaire. The four variables included in the data set are a greenness vegetation index, temperature, elevation and precipitation. The relevant data files are included in the \AFRICA directory.

Finally, I will also refer to the well-known data set for Irish counties used in Cliff and Ord (1981, pp. 207-208). The relevant files for this are included in the \EIRE directory.

1.4.2 Your Data

Of course, you are welcome to use your own data sets to follow along in this tutorial, and I would encourage you to do so. Keep in mind that in order for your data set to be applicable in the examples outlined below, you will need the following information:

- at least 20 observations for contiguous spatial units (e.g., counties, census tracts, states)
- one dependent variable and a few explanatory variables (say, 2 or 3)
- X and Y coordinates of the centroids of the spatial units (or for any other meaningful point, such as the county seat for a county)
- a map with the spatial units, or a digitized boundary file.

The current version of **SpaceStat** includes a limited interface with both the ARC/INFO and IDRISI geographic information systems. However, for the examples in this tutorial, I will only use the latter. If you have access to this GIS, you will be able to extract information on the spatial arrangement of your observations. You will also have limited facilities for the
graphical display of the results of your spatial data analyses. Of course, this implies that you previously constructed a rasterized vector boundary file in the format of an IDRISI image.

1.5 Organization of the Tutorial

The tutorial is organized in six parts. In the remainder of this first part, I give a brief overview of the functionalities of SpaceStat. In part II, I outline the ways in which data sets are created and manipulated in the program. Part III deals with ways to incorporate information on the spatial arrangement of the observations. It includes a treatment of the creation of spatial weights matrices from an ascii file and from information stored in a GIS. It also covers how such weights matrices can be row-standardized, powered to a higher order and analyzed in terms of their characteristics. In Part IV, I discuss simple descriptive statistics and the exploratory analysis of spatial association. Among others, I treat join count statistics for spatial autocorrelation, Moran’s I, Geary’s c, QAP and observation-specific indices (the G statistics). Part V deals with spatial regression analysis and includes the estimation of a linear regression model with diagnostics for spatial effects, and the estimation of models with spatial dependence. In Part VI, I outline several special forms of spatial regression models, including a trend surface model, spatial ANOVA, spatial regime regression and the expansion method. An Appendix contains detailed information on the menu structure of SpaceStat and various file formats, as well as figures and data tables referred to in the text.

1.6 Notational Conventions

All output from the SpaceStat program will be listed in Courier type, while the rest of the tutorial is in the ITC Bookman type. Italics are used for your response to a prompt in SpaceStat. All file names will be capitalized.

References

CHAPTER 2

OVERVIEW OF SPACESTAT

2.1 Getting Started

As mentioned in the previous chapter, you start SpaceStat by typing space (or SPACE) from the DOS prompt, or run space from the GAUSS prompt. Next, a title screen appears, which lists the version number and revision date (this is important information when you report problems). By pressing any key you get to a welcome screen that lists the main menu items at the top. This top row remains in any of the following submenus of SpaceStat as well. You get out of SpaceStat at any point by pressing the ALT-Q key combination.

2.2 Menu Structure

SpaceStat uses a very simple menu interface. The main menu and options are listed at the top of each menu screen. The other menus are listed immediately below this line. Each of the first four items of the main menu corresponds to a module in SpaceStat. There are four such modules:

- Data: the creation and manipulation of data sets
- Tools: the manipulation of spatial weights
- Explore: descriptive measures of spatial association
- Regress: spatial regression analysis

In the Data and Tools modules, the menu structure for the commands contains two layers. In the Explore and Regress modules, this menu structure consists of three layers. Each of these menus contains a list of numbered items. To each item corresponds another menu or a specific action. You move around in the menus by means of the cursor direction keys (the arrow keys near the bottom right hand side of your keyboard) or by typing in the sequence number of the submenu or action. I will return to this with specific examples for each item covered in this tutorial.

The complete menu structure is included as Appendix A. To execute a command, you enter the letter corresponding to the module in the main menu, followed by two or three numbers. Alternatively, you may move the cursor to the desired item and press Return.1 In the remainder of this tutorial, I will refer to SpaceStat command sequences as a letter followed

---

1. In the remainder of the tutorial, the Enter or Return key will be referred to as Return. The italics indicate that this is an action you will have to perform.
by two numbers. The letter corresponds to the module on the main menu, while each number corresponds to an item on a submenu. For example, to invoke the Ascii to Data Set conversion command on the Input menu of the Data module, the command sequence is D-1-1.

2.3 Hot Keys

In the main menu, the first letter for each module is highlighted, to indicate that these are "hot keys." In order to reach the menu for any module, you simply type the letter that corresponds to it. For example, to start the Data module, you would type d or D (hot keys are not case sensitive). The full list of hot keys is given in Table 2.1. In this table, note three keys in addition to those that correspond to the four main modules in Spacestat: the F1 and F2 function keys (at the top or left hand side of your keyboard) and the combination ALT-Q (press down the ALT key and the Q key simultaneously). The most important of these is ALT-Q, which lets you exit SpaceStat from any menu or submenu. The F2 function key allows you to run DOS programs from within SpaceStat, and the F1 key lets you set various options for the program.

<table>
<thead>
<tr>
<th>Key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D</td>
<td>Data module</td>
</tr>
<tr>
<td>T</td>
<td>Tools module</td>
</tr>
<tr>
<td>E</td>
<td>Explore module</td>
</tr>
<tr>
<td>R</td>
<td>Regress module</td>
</tr>
<tr>
<td>ALT-Q</td>
<td>Quit SpaceStat</td>
</tr>
<tr>
<td>F1</td>
<td>Options</td>
</tr>
<tr>
<td>F2</td>
<td>DOS</td>
</tr>
</tbody>
</table>

2.4 Running DOS Programs

By pressing the F2 function key, you get to the DOS system prompt. This allows you to carry out simple functions such as copying or deleting files, and lets you run small programs, such as some text editors. You can return to SpaceStat by typing exit.

By means of this feature you are able to run software such as the IDRISI GIS from within SpaceStat, provided that you have sufficient memory on your system (4 megabytes of RAM or more). This provides an easy way to graphically display some of the results of your analyses (see the discussion of the GIS interface in the next chapter).

You may encounter problems if the memory requirements of the DOS programs you want to execute are too big. In those instances, you will get the familiar out of memory DOS error message. The only way to run these bigger programs would be to first get back to SpaceStat by typing exit and subsequently to quit the program by typing ALT-Q.
2.5 Getting Out of SpaceStat

As mentioned above, you can exit SpaceStat from any menu by pressing the $\text{ALT-}Q$ key combination. However, in some situations you may inadvertently have started a command that you do not want to carry out. In those instances, you cannot immediately exit by means of the $\text{ALT-}Q$ key combination.

All commands in SpaceStat request responses to a few prompts. The required responses are typically file names or variable names. In almost all cases, you can get back to the previous menu by simply pressing the Return key in response to a prompt. This will give you an error message, but another Return will get you back to the menu. In the few cases where Return itself is a valid response, you may have to proceed to the next query and press Return again. This will almost always work. Once you are back in a menu, $\text{ALT-}Q$ will get you out of SpaceStat.

One response to avoid is to press the $\text{ESC}$ key at a prompt for a filename or variable name. While this is harmless in most cases, except for generating a series of \ on your screen, in some instances it can hang up the program. If this happens to you, there is no other way out than to type $\text{CTRL-C}$ (or $\text{BREAK}$) to interrupt SpaceStat (in the worst case, you may have to reboot).

2.6 SpaceStat Options

If you press the $F1$ function key, you exit from the current menu and get a new screen, with a list of options as in Table 2.2. All options are given with their current defaults. You can change any of these settings by typing the number that corresponds to the option, or by moving the cursor down (or up) to the line of the option and pressing the Return key. For three of the options (1, Interactive Operation; 3 Long Output; and 5, Idrisi Interface) this is all you have to do. These options have so-called toggle switches that move back and forth between two settings (YES and NO) whenever they are invoked. For the other options, you will next be prompted for further information, such as a file name or value for an option.

<table>
<thead>
<tr>
<th></th>
<th>Table 2.2 SpaceStat Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Interactive Operation:</td>
</tr>
<tr>
<td>2</td>
<td>Output to a File:</td>
</tr>
<tr>
<td>3</td>
<td>Long Output:</td>
</tr>
<tr>
<td>4</td>
<td>Indicator Variable:</td>
</tr>
<tr>
<td>5</td>
<td>Idrisi Interface:</td>
</tr>
<tr>
<td>6</td>
<td>Convergence Criterion:</td>
</tr>
<tr>
<td>7</td>
<td>Number of Permutations:</td>
</tr>
<tr>
<td>8</td>
<td>Maximum Iterations:</td>
</tr>
<tr>
<td>9</td>
<td>Random Number Seed:</td>
</tr>
</tbody>
</table>
You get out of the options menu and back to the menu you were in before by pressing the \textit{ESC} key from the options menu. The current settings of the options will be used until you change them again. Note that you cannot permanently change them or save them for future use: SpaceStat always starts up with the same set of defaults.

Below follows a brief description of what each option achieves.

\subsection*{2.6.1 Interactive Operation}

The default is \textbf{YES}, which means that \textbf{SpaceStat} waits after listing a screen of results until you press the \textit{Return} key to proceed to the next one. This is the most natural way to analyze the output of \textbf{SpaceStat}. However, in some instances you may want to run a large number of analyses (say, overnight). If you change the option setting to \textbf{NO} (by typing 1 or moving the cursor to it and pressing \textit{Return}), all results will be listed without a pause. This setting is only useful if you have the second option (Output to a File) set to \textbf{YES}.

\subsection*{2.6.2 Output to a File}

The default is \textbf{NO}, which means that \textbf{SpaceStat} will not automatically provide "hard copy" of your output, but only list the results to the screen. This is useful when you are experimenting with the program or when you carry out exploratory analysis. If you change the setting to \textbf{YES}, you will be prompted for the file name of the output file. This file will contain all screen listings in an ascii format. You can manipulate this file by means of a text editor or word processor, e.g., to produce tables with your results. If you press the \textit{Return} key in response to the file name prompt, the default output file is \textit{SPOUT.DOC}. The file name of the output file will be listed next to the \textbf{YES} on the second line of the \textbf{Options} menu.

If you want to change the output file name to another one after you already switched the setting to \textbf{YES}, you need to proceed in two steps. First, you type 2 to switch the setting to \textbf{NO}. Next, you type 2 again to switch it back to \textbf{YES} and enter the new file name in response to the prompt.

\subsection*{2.6.3 Long Output}

The default is \textbf{NO}, which means that some results are not automatically provided. In order to have those results listed to the screen or to the output file, you need to change the setting to \textbf{YES}. The \textbf{Long Output} option is only relevant in the Explore and \textbf{Regress} modules and affects items such as the listing of all iterations in the numerical optimization, the listing of regression residuals, etc. You change the option by typing 3, or by moving the cursor down to the third line and pressing the \textit{Return} key.
2.6.4 Indicator Variable

The default setting is NO, which means that observations will be referred to by their sequence number in the data set. Often, you may want to use the values of a particular variable to do this. For example, you may want to refer to census tract observations by means of their tract number. This feature is particularly useful when you interface SpaceStat with a GIS. In order to change the setting, you need to type 4 (or move the cursor down to the fourth line and press the Return key) and enter the name for the indicator variable in response to the prompt. If you don’t specify a variable name, but press Return instead, the default variable name of OBS will be used. If the variable specified in this option is not present in your data set, an error message will be generated and your analysis will not be completed. If the option setting is YES, the indicator variable will be listed next to it.

Note that only numeric integer indicators may be used.

2.6.5 Idriesi Interface

The default setting is NO, which means that no particular adjustments are made to provide output in a format useful for IDRISI. If you change the setting to YES, some of the SpaceStat output will be written to files with a file extension of .VAL. These .VAL files only contain two columns of values. If the observations in the files correspond to polygons in an IDRISI image whose indicators match the values in the first column of the .VAL file, then the values of the second column (e.g., regression residuals) can easily be converted to a new IDRISI image. You can change this option by typing 5.

2.6.6 Convergence Criterion

The default setting is 0.000001, which is the absolute difference between values in subsequent iterations in the nonlinear optimization at which convergence is assumed to be reached. For most applications, the default setting is more than precise enough. If you wish to change it, type 6 and enter the new value in response to the prompt.

2.6.7 Number of Permutations

The default setting is 99, which is the number of resampled data sets that will be created whenever a random permutation approach is used. The default setting is sufficient for initial analyses, but is too small if you desire a high degree of precision. In those instances, you should use values such as 999 or 9999. Note that the computation time needed for the permutation approach will be a direct function of how high this option is set. You can change the setting by typing 7 and entering the new value at the prompt.
2.6.8 Maximum Iterations

The default setting is 100. This option is only used in a few instances where nonlinear optimization may not always converge. In those instances, you can use this option to "force" convergence. You change the setting by typing 8 and entering the new value in response to the prompt.

2.6.9 Random Number Seed

The default setting is 397204094, which is the value used by the GAUSS system. You will only want to set the seed in this fashion if you want to exactly replicate results obtained by means of a random permutation approach. If you want to use the CPU clock to set the seed instead, you type 8 and respond -1 to the prompt for a new seed. The value of 0 (zero) will be listed next to the option. Of course, you may also enter an alternative value for the seed.
CHAPTER 3

INPUT AND OUTPUT

3.1 GAUSS Data Sets and GAUSS Matrices

SpaceStat treats all data sets and matrices internally in the binary format used by the GAUSS system. This greatly speeds up input and output operations, but precludes you from entering and editing data sets or matrices directly.

All variables analyzed by means of SpaceStat commands must be included in a GAUSS data set. Consequently, the first step in any analysis will be to convert your data into this GAUSS data set format (see Part II). Similarly, all matrices used by SpaceStat must be in the GAUSS matrix format (see Part III).

The filenames of the data set and/or matrices used by SpaceStat commands are entered interactively, in response to a prompt, or are contained in a so-called Problem File (see below).

3.2 Ascii Input Files

For several commands contained in the Data module, information must be moved from an ascii format into the GAUSS binary format. In those instances, the input file must be a standard ascii file. At this point, SpaceStat is not yet able to read files in various non-ascii formats, such as those used by word processing, spreadsheet or data base software. If your data is produced by such software, you must first convert it to an ascii format (e.g., in a so-called .PRN file created with many spreadsheet programs). Alternatively, you may use one of a number of file conversion programs available on the market to convert your data from a non-ascii format into the GAUSS format directly.

There are five common types of ascii input files used by SpaceStat. Each of them has to conform to a specific structure or errors will be generated in the conversion of data from ascii to GAUSS format. The respective file structure is briefly outlined below.

3.2.1 Ascii Data Files

An ascii data file is the input file used to create a GAUSS data set. It is organized as a matrix, with the rows corresponding to observations and the columns to the variables. The actual observations are preceded by a header with the following items:

- first item: number of observations
- second item: number of variables
- third item: list of variable names
3.2.2 Ascii Matrix Files

An ascii matrix file is the input file used to create a generic GAUSS matrix. Note that a generic matrix is treated differently from the spatial weights matrices used in many types of spatial data analyses. A generic matrix can have any dimension, whereas spatial weights matrices must conform to a more restrictive structure (see Part III). A generic matrix input file is organized row by column (e.g., the elements of the first row first, then the elements of the second row, etc.). The data themselves are preceded by a brief header with the number of rows. For further details on creating a GAUSS matrix from this input file, see Chapter 5.

3.2.3 Contiguity Files

Contiguity files contain the information on the spatial arrangement of the observations in a spatial data set. For each observation, the number of neighboring observations is recorded as well as their identifiers. The latter are either simple sequence numbers or other numeric values (e.g., census tract numbers) that match the values taken by an indicator variable. For each observation, the contiguity is recorded as follows:

- observation identifier
- number of contiguities
- identifiers of the contiguities.

This follows the so-called GAL (geographic algorithm library) format for contiguity files. The contiguity information is preceded by a header which differs slightly depending on the type of identifier used for the observation. When simple sequence numbers are used (e.g., corresponding to the row and column of the spatial weights matrix), the header only contains the number of observations. In contrast, when the values for an indicator variable are used to uniquely identify each observation the header is as follows:

- first item: number of observations
- second item: name of the GAUSS data set that contains the values for the indicator variable for each observation
- third item: variable name for the row indicator
- fourth item: variable name for the column indicator

Note that you must specify an indicator variable for both row and column of the weights matrix. This variable can be the same, but it does not have to be. This allows for the greatest flexibility in treating information on contiguity. If the indicator variables listed are not included in the data set specified, an error message will be generated. For further details on the creation of spatial weights matrices, see Part III.
3.2.4 **Spatial Weights Files**

Spatial weights or general contiguity files contain information on the strength of the potential interaction between two observations, corresponding to the non-zero elements of a spatial weights matrix. Each such element is represented by the following items:

- identifier for row observation
- identifier for column observation
- value of the spatial weight

Examples of spatial weights that may be created in this fashion are the length of common boundary (e.g., as computed in a GIS) and the distance between two observations. As in the case of simple contiguity, the weights information is preceded by a header file that differs between the use of sequence numbers and the use of indicator variables as identifiers. In the former case, the header simply gives the number of observations. When the values for an indicator variable are used to uniquely identify each observation, the header is as follows:

- first item: number of observations
- second item: name of the GAUSS data set that contains the values for the indicator variable for each observation
- third item: variable name for the row indicator
- fourth item: variable name for the column indicator

For further details on the creation of spatial weights matrices, see Part III.

3.2.5 **Auxiliary Input Files**

A number of SpaceStat commands allow you to perform operations on a subset of a data set or a spatial weights matrix. This subset is determined by selecting those observations or row/column pairs whose values for an indicator variable match the values contained in a list. The indicator variable must be contained in an existing GAUSS data set. In addition, the variable name and the list of values must have been previously written to an ascii file. The structure of such an auxiliary input file is:

- first item: indicator variable
- second item: list of values for observations to be selected (or deleted)

3.3 **Raster Image Files**

SpaceStat includes a limited direct interface to a number of raster-based GIS. This facility allows the information on contiguity to be constructed directly from an image for a rasterized vector data set, i.e., a data set where the polygons that correspond to the units of observation have been converted into a raster format. The current version of SpaceStat allows you to con-
struct a contiguity matrix from the binary raster files for IDRISI and OSU-MAP, as well as for a generic raster format (provided that you know the format). Further details are provided in Part III.

3.4 Standard Output Files

There are two types of output files that are created by SpaceStat. The first type are data sets or matrices that result from manipulations of existing data in the Data and Tools modules. The second type are ascii files that contain the results of a spatial data analysis from the Explore and Regress modules. These ascii files contain the same information as is listed on the screen, in the same format. The standard output is only written to a file if the Output to a File option (option 2) is set to YES. The output is appended to the file specified in that option (old output files with the same name are thus never over-written).

3.5 Report Files

3.5.1 Generic Report Files

When the Long Output option is set to YES, some additional output is written to a so-called report file. This report file contains the output in a simpler format, that is easier for importing into other software, such a spreadsheet packages or graphics software. A report file typically contains a small header with variable names (e.g., for the predicted value or residual), including one for the indicator variable that uniquely identified each observation. This indicator variable is set by means of option 4. The report file is a simple list of values of a small number of statistics for each observation. The name of the report file is specified in response to a prompt, or is contained in a problem file (see below). For further details, see Parts IV-VI.

The name you specify for the report file should not include a file extension. A file extension of .DOC is automatically added by SpaceStat. If you include an extension, SpaceStat will crash.

3.5.2 Report Files for IDRISI

When the Idrisi Interface option is set to YES, a special type of report file is created, that is in a format suitable for immediate display in IDRISI. Such files are characterized by a .VAL file extension. They contain no headers and consist of only two columns: the first column corresponds to the values of a polygon identifier used in the IDRISI image, while the second column contains the actual values for the statistic or data item reported. These report files can be generated for a number of data analyses (details are given in Parts IV-VI). In addition,
any variable contained in a GAUSS data set can be listed as a .VAL file for immediate display in IDRISI.

3.6 Problem File

A **Problem File** is a file that contains all the necessary information to carry out an analysis by means of **SpaceStat**. For example, it contains the variable names, file names for the data set, etc. A Problem File can be created interactively, by entering the information for each item in response to **SpaceStat** prompts. Alternatively, you can also use a text editor to create a Problem File. However, it is extremely important that you adhere to a specific format. The full details of this format are included as Appendix B.

3.7 Spatial Weights Files

**SpaceStat** stores spatial weights files as GAUSS matrix files, in order to speed up computations. At first, these matrices are square. However, once **SpaceStat** performs an operation on the weights matrices, such as row-standardization, or computes some of its characteristics, such as the eigenvalues and matrix traces used in the tests and estimation methods, two additional columns are added. The first column contains up to 20 characteristics of the weights matrix. The second column contains its eigenvalues. As a result, when you check the dimensions of a weights matrix, the row and column dimensions will typically not be the same, but the latter will exceed the former by 2. The detailed format of the two additional columns is given in Appendix C.
PART II

DATA SETS
CHAPTER 4

CREATING A DATA SET

4.1 Basic Principles

For all analyses in SpaceStat the program assumes that your data have been converted into the binary (i.e., not ascii) format used to store data sets in the GAUSS System. The Data module in SpaceStat provides you with a number of functions to create and manipulate such data sets.

The first step in creating a SpaceStat data set is to organize your data in a tabular form in an ascii file, with the rows corresponding to observations and the columns to variables. At the top of this file should be the following three items (in this exact order):
- number of observations
- number of variables
- variable names

The variable names should have no more than 8 characters. Note that in some SpaceStat operations a prefix is added to variable names, which may result in the last 3 characters being truncated (since the overall variable name length limit of 8 characters remains). It is thus good practice (if possible) to use variable names with only 5 characters.\(^1\) It is not necessary that all values for a single observation (i.e., for all variables) fit on the same line in the ascii file, as long as the sequence of values corresponds to the sequence of variable names listed at the top of the file. In SpaceStat, only numerical values are allowed in a data set. If you have variables with character data (e.g., county names) in your ascii file, they should be edited out before you can proceed and create the data set for use by SpaceStat.

4.2 Command: D\(-\)1\(-\)1

You create the data set by means of the SpaceStat Data - Input - Ascii to Data Set command (D\(-\)1\(-\)1). You start this command by first typing \(D\) (for the Data module), followed by \(1\) (for the Input menu in the Data module) and another \(1\) (for the Ascii to Data Set action in the menu). You will be prompted for the name for the ascii file with the data and for the name of the data set you wish to create. The input ascii file must conform to the format outlined in the previous section. The output of the command is a data set in the GAUSS format. Such data sets are actually represented by two files, one with a .DAT extension and one with

\(^{1}\) You can always change the variable name in a GAUSS data set by means of the SpaceStat Data - Var Create - Relabel Variables command (D\(-\)4\(-\)1).
CREATING A DATA SET

a .DHT extension. Neither of these extensions should ever be included in a response to a prompt for a data set name.

4.3 Example

4.3.1 Files Needed

To complete this example, you will need the file COL.ASC. This file is included in the \COLUMBUS directory. If this file is not yet present in your current or work directory, you should now copy it. If you are in SpaceStat, you should use the F2 key to move to DOS and type exit after you finished copying the file.

4.3.2 Input File Contents

You will now create a data set in the GAUSS format for the Columbus crime data. Before you start entering commands in SpaceStat, take a look at the contents of the file COL.ASC. In Table 4.1, the first few lines of the file are listed. The full contents of this file are given in Appendix D.

<table>
<thead>
<tr>
<th>Table 4.1 Columbus Crime Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>49 7</td>
</tr>
<tr>
<td>NEIG  CRIME  INCOME  HOUSING  X  Y  EW</td>
</tr>
<tr>
<td>1   18.802  21.232  44.567  35.62  42.38  0</td>
</tr>
<tr>
<td>2   32.388  4.477   33.200  36.50  40.52  0</td>
</tr>
<tr>
<td>3   38.426  11.337  37.125  36.71  38.71  0</td>
</tr>
</tbody>
</table>

Note the items at the top of the file (the first line in Table 4.1):

49 for the number of observations (the number of neighborhoods)

7 for the number of variables

These items are followed by the variable names NEIG (for the neighborhood sequence number), CRIME, INCOME, HOUSING (for housing value), X, Y, and EW (the east-west indicator variable).

4.3.3 Creating the Data Set

Once you are in SpaceStat, type $d$ (or $D$) to get to the menu for the Data module. The first item in this menu is Input. Type 1 (or press Return, since the cursor is on the first line) to get the menu for Input. The first item on this menu is Ascii to data set. Again, type 1 (or press Return) to start this command.
Next, you get two prompts for file names and should enter respectively the filenames `col.asc` (for the ascii file) and `col` (or any other name you choose to give to the new data set), each followed by a Return, as shown:

Creating a data set from an Ascii file

Enter the name of the Ascii input file
File name is: col.asc

Enter the data set filename (do not include .DAT or .DHT)
File name is: col

SpaceStat will read the ascii file, convert it to a data set in the GAUSS format, clear the screen and give you the following message:

The GAUSS dataset col has been created
It contains 49 observations on 7 variables
The variables contained in the data set are:
NEIG CRIME INCOME HOUSING X Y EW

You get back to the Data Input menu by pressing the Return key.

You have now created your first SpaceStat data set. You can find it as the files COL.DAT and COL.DHT on the current directory. By the way, if you chose not to carry out the data set creation steps at this point, you may find the same data set on the \COLUMBUS directory.

4.4 Exercise

For further practice, you can now also convert your own data to the GAUSS data set format, or create a data set for the African or Cliff-Ord Irish example data. The African data are in an ascii file AFRICA.ASC on the \AFRICA directory, while the Irish data are contained in the ascii file EIRE.ASC in the \EIRE directory. If you decide to work with your own data, you will first have to type in the variable names and values with a text editor or word processor (make sure to save the file in plain ascii or DOS format), or export them from a spreadsheet or data base package. Make sure not to forget to put the number of observations, number of variables and variable names at the top of the file.

You can always check the contents of a data set by means of the first command on the Data List menu, Summary Data Set: type D-8-1 and enter the name of the data set at the prompt.
CHAPTER 5

CREATING A GAUSS MATRIX

5.1 Basic Principles

Whenever matrices are used by SpaceStat, the program assumes that they have been converted into the binary GAUSS format. The Data module in SpaceStat provides you with a number of functions to create and manipulate such matrices. In this chapter, I will illustrate the creation of a generic matrix, which does not have to be used as a spatial weights matrix. Spatial weights matrices are treated differently, since their special structure allows for a more efficient approach. This is covered in the chapters of Part III.

The first step in creating a GAUSS matrix with SpaceStat is to organize your data row by row in an ascii file. The values for a row may extend over several lines, but the order of the rows must correspond to that in the matrix. At the top of this file should be the following item:

- number of rows in the matrix

The number of columns should not be specified: it is derived internally by the program. In SpaceStat, only numerical values are allowed in a matrix.

5.2 Command: D-1-2

You create the matrix set by means of the SpaceStat Data - Input - Ascii to Matrix command (D-1-2). You start this command by first typing D (for the Data module), followed by 1 (for the Input menu in the Data module) and 2 (for the Ascii to Matrix action in the menu). You will be prompted for the name for the ascii input file and for the name of the matrix file you wish to create. The input ascii file must conform to the format outlined above. The output of the command is a matrix in the GAUSS format. Such a matrix can be recognized by the .FMT file extension. It is always stored in double precision (8 bytes per element). The file extension should never be included in a response to a prompt for a matrix file name.

5.3 Example

5.3.1 Files Needed

To complete this example, you will need the file COLW1.ASC. This file is included in the \COLUMBUS directory. If this file is not yet present in your current or work directory, you should now copy it. If you are in SpaceStat, you should use the F2 key to move to DOS and type exit after you finished copying the file.
5.3.2 Input File Contents

You will now create a matrix in the GAUSS format which corresponds to the first order contiguities for the Columbus neighborhoods. Note that this is actually the least efficient way to create a weights matrix, but it is a good example of the construction of a generic matrix. Before you start entering commands in SpaceStat, take a look at the contents of the file COLW1.ASC. In Table 5.1, the first few lines of the file are listed.

Table 5.1 Columbus Contiguity Matrix (partial)

|     | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|     | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|     | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|     | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|     | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|     | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Note the number of rows (49) at the top of the file. Each pair of the following lines in the file correspond to the contiguities for a neighborhood: the 1 signifies the presence of a contiguity, the 0 an absence (see Part III for a more extensive treatment).

5.3.3 Creating the GAUSS Matrix

Once you are in SpaceStat, type d (or D) to get to the menu for the Data module. The first item in this menu is Input. Type 1 (or press Return, since the cursor is on the first line) to get the menu for Input. The second item on this menu is Ascii to matrix. Type 2 (or move the cursor down to the second line and press Return) to start this command.

Next, you get two prompts for file names and should enter respectively the filenames colw1.asc (for the ascii file) and col_1 (or any other name you choose to give to the new matrix file), each followed by a Return, as shown:

Creating a matrix file from an Ascii file

Enter the name of the Ascii input file
File name is: colw1.asc

Enter the matrix filename (do NOT include .FMT)
File name is: col_1

SpaceStat will read the ascii file, convert it to a matrix in the GAUSS format, clear the screen and give you the following message:

The ASCII file colw1.asc has been created as a GAUSS matrix col_1.fmt
The matrix has 49 rows, and 49 columns

You get back to the Data Input menu by pressing the Return key.

You have now created a matrix in the GAUSS format. You can find it as the file COL_1.FMT on the current directory, and also on the \COLUMBUS directory.

5.4 Exercise

For further practice, you can now convert any other table of numbers into a GAUSS matrix format. You may want to construct a few matrices to experiment with some of the matrix operations covered later in Part II. Make sure to have the number of rows as the first item in the ascii input file.

You can always check the dimensions of a GAUSS matrix by means of the fifth command on the Data List menu, Summary Matrix: type D-8-5 and enter the name of the matrix file at the prompt.
CHAPTER 6

LISTING THE CONTENTS OF DATA SETS AND MATRICES

6.1 Introduction

Once you have created a data set or matrix in the GAUSS format, you can list its contents by means of the commands in the List menu of the Data module. These listings only allow a limited degree of formatting: the same width and number of decimals is listed for all values in a data set or matrix. The listing is written to the ascii file specified in the Output File option (option 2) when this option is set to YES. If it is set to NO, the listing will only be shown on the screen. In addition, when the Idrisi Interface option is set to YES, a file with file extension .VAL is created for each variable listed. This file is in the correct format to be displayed by means of the IDRISI GIS.

The listing commands are invoked by the \texttt{D-8} command sequence, followed by the option number from the List menu.

6.2 Listing Summaries

A quick way to check the contents of a GAUSS data set or the dimensions of a GAUSS matrix is to use the \texttt{D-8-1} and \texttt{D-8-5} command sequences in SpaceStat. The first command yields a list of the number of observations, number of variables, and the variable names in the data set you specify in response to the prompt. The second command yields the row and column dimensions of the matrix file you entered in response to the prompt. These listings are the same as those provided at the end of the creation of a GAUSS data set or matrix, illustrated respectively in Chapters 4 and 5.

6.3 Listing the Contents of a Data Set

6.3.1 Listing the Full Data Set: \texttt{D-8-2}

You can list the full contents of a GAUSS data set by means of the \texttt{D-8-2} command sequence. You are prompted for the name of the data set and asked to specify the format for the listing. This format consists of two elements: the number of spaces reserved for each value (width) and the precision (number of decimals). The default of 12 spaces with 6 values after the decimal is chosen by typing 0 in response to the prompt. The listing is presented one screen at a time, with each screen showing 20 observations for as many variables as can be included, given the specified format. Once all the observations for the first set of variables
is shown, the listing continues with the next set. At the end, you return to the List menu by pressing the Return key.

6.3.2 **Listing Selected Variables: D–8–3**

You can select a number of variables to be listed instead of the full data set. You invoke this function by means of the D–8–3 command sequence. After you enter this sequence, you are prompted for the name of the data set. Next, you are given a list of the variables in the data set and asked to enter the names for the variables you want to list. You end the process of entering variable names by pressing the Return key in response to the prompt. Only those variables for which the name you typed corresponds exactly with a variable name in the data set will be listed. After you entered the variable names, you are prompted for the desired format of the listing, in exactly the same manner as outlined in the previous section. The listing follows the same conventions as for the full data set.

6.3.3 **Listing a Range of Observations: D–8–4**

You can also elect to list all variables for only a range of observations. You carry this out by means of the D–8–4 command sequence. After you enter the name of the data set in response to the prompt, you specify a range by first choosing an indicator variable that takes unique consecutive integer values for each observation in the data set. Next, you specify the lower and upper values for the range. These values must be positive integer numbers or SpaceStat will generate an error message. Note that you can always create an indicator variable with the desired properties by means of the D–4–6 command sequence (Data - Var Create - Create Observation Numbers). This is illustrated in more detail in Chapter 8.

Once you specified the data set, indicator variable and range, you are prompted for the format of the listing, as before (see 6.3.2). The listing follows the same convention as for the full data set (see 6.3.2).

6.3.4 **Idrisi Interface**

When you set the Idrisi Interface option to YES, SpaceStat creates special output files that conform to the format needed for simple display in IDRISI. These output files are characterized by a .VAL file extension. One file is created for each variable, except for the first one in the list of variable names. This first variable is assumed to contain the values that correspond to the polygon identifiers in an IDRISI image file. Note that if the first variable listed for a data set is not the proper indicator variable, you must use the List Selected Variables command (D–8–4) and specify the correct indicator variable as the first one selected.
The filenames for the output files match the variable names in the data set. Each file contains two columns of values: the first one corresponds to the indicator variable, the second one to the variable given in the file name. If a file with the same name already exists, it will be over-written. This contrasts with the convention used for standard output files, which are never over-written.

The listing with the Idrisi Interface option set to YES operates in the same fashion as the standard operation outlined in sections 6.3.1-6.3.3.

6.4 Listing the Contents of a Matrix: D-8-6

You can list the full contents of any GAUSS matrix by means of the D-8-6 command sequence. All the elements of the matrix are listed, row by row. For spatial weights matrices, the listing includes the two extra columns that SpaceStat adds to them (see Part III). If you don’t wish to list the contents of these additional columns, you should first convert the weights matrix back to the standard matrix format by means of the Tools - Weight Trans - Convert Weights Format to Matrix Format (T-2-4) command sequence (for details, see Part III). However, more efficient ways to list the contents of spatial weights matrices are given in the next section (6.5).

After you enter the command sequence, you are prompted for the file name of the matrix and for the format to be used in the listing. The conventions are the same as for listing data sets.

6.5 Listing the Contents of a Spatial Weights Matrix

The most efficient way to list the contents of a spatial weights matrix is to take advantage of its sparseness (i.e., the large number of zeros in the matrix). SpaceStat allows two different formats for sparse weights matrices (see Part III) and these are also used in the listing. When the weights matrix only contains information on contiguity, in the form of a 0-1 binary contiguity matrix, the listing follows the format suggested in the geographic algorithms library (GAL). For a general spatial weights matrix, the value of the weight is listed for each nonzero element, together with its row and column number.

6.5.1 Listing a Simple Contiguity Matrix: D-8-7

The contents of a simple contiguity matrix are listed by means of the command sequence D-8-7. The only prompt is for the file name of the matrix. After you enter this name, the screen clears and a header appears which echoes the file name. Note that this header is not written to the output file (when the Output File option is set to YES). The listing of contiguities
is preceded by the number of observations, in conformance with the GAL format (see also 3.2.3 and Part III). For each observation, the following three items are listed:

- observation sequence number
- how many contiguous observations
- sequence numbers of contiguous observations

The listing is given one screen at a time. At the end, you get back to the List menu by pressing the Return key.

### 6.5.2 Listing a General Spatial Weights Matrix: D–8–8

The contents of a general spatial weights matrix are listed by means of the command sequence D–8–8. The only prompt is for the file name of the matrix. After you enter this name, the screen clears and a header appears which echoes the file name. Note that this header is not written to the output file (when the Output File option is set to YES). The listing of contiguities is preceded by the number of observations, following the convention used for all weights matrices in SpaceStat (see also 3.2.4 and Part III). For nonzero element in the weights matrix, the following three items are listed:

- sequence number for the row
- sequence number for the column
- value of the weight

The weights are listed with eight decimals of precision. The listing is given one screen at a time. At the end, you get back to the List menu by pressing the Return key.

### 6.6 Example

To illustrate the various types of listings, you will use three files: COL.DAT, COL.DHT and COL_1.FMT. If you went through the examples in the previous chapters, you will already have created these files. They are also contained in the \COLUMBUS directory. Make sure that these files are on your current working directory.

You start by typing d (or D) to get the menu for the Data module, followed by 8 (or move the cursor to the bottom of the menu and press Return) for the List menu. Now, set the option for the Output File and for the Idrisi Interface. Press the F1 key, followed by 2. In response to the prompt, enter collist.doc (or any other filename), as shown:

```
Enter name for output file (or Return for default): collist.doc
```

Also, type 5 to switch the Idrisi Interface toggle to YES. Finally, press the ESC key to get back to the List menu.

Next, type 3 (or move the cursor down to the third line and press Return) to list selected variables. In response to the prompts, enter col as the name for the data set and
NEIG, CRIME and INCOME as the selected variables (of course, you can select any other subset of the variables), as shown:

Listing of selected variables in a data set

Enter the data set filename (do not include .DAT or .DHT), or press Return for directory listing: col

Choose the variable(s) from the following list
(each variable name should correspond exactly to one in the list):

NEIG  CRIME  INCOME  HOUSING  X  Y  EW

Enter the variable name, or press Return to stop

Variable name: NEIG
Variable name: CRIME
Variable name: INCOME
Variable name:

Press the Return key at the prompt following your entry of the INCOME variable to end the queries. Next, you are asked to specify the format for the listing. Enter 6 and 3 for respectively the width and precision, as shown:

Enter the format for the listing
Format is w.d with w as width d as number of decimals (default 12.6)
Enter the width (or 0 as default): ? 6
Enter the number of decimals: ? 3

After this last entry, the screen will clear and you will see the first 20 observations on the variables NEIG, CRIME and INCOME. The first few lines of this listing are shown in Table 6.1. When you press Return, you see the values for the next 20 observations, and so on, until all 49 have been listed. One more Return gets you back to the List menu. Now, type F2 to escape to the DOS prompt and check the contents of your working directory (use the DOS dir command). You will find a file named COLLIST.DOC, as well as two files with a file extension of .VAL, CRIME.VAL and INCOME.VAL. These files can be used to produce a choropleth map in the IDRISI GIS, provided that you earlier created a corresponding image. Such an image for the Columbus neighborhood data is contained in the \COLUMBUS directory as the files COLUMBUS.DOC and COLUMBUS.IMG. The CRIME.VAL and INCOME.VAL files are included in this directory as well. The first few lines of the CRIME.VAL file are listed in Table 6.2. Note the absence of headers to indicate the variable names.
In order to list only a limited range of observations for the variables in data set COL, type 4 in the List menu. Enter col and neig in response to the prompts, as shown:

Listing of selected observations in a data set

Enter the data set filename (do not include .DAT or .DHT), or press Return for directory listing: col

Enter the name for the variable that holds the observation number

Choose the variable(s) from the following list

(Each variable name should correspond exactly to one in the list):
NEIG   CRIME   INCOME   HOUSING    X     Y    EW

Enter the variable name, or press Return to stop

Variable name: NEIG

Now, type 10 and 20 respectively (or any other range between 1 and 49) in response to the prompt for the lowest and highest value in the range of observations, as shown:

Enter the range for the observations to be listed

(Integer values only, enter 0 to return to menu)

From the lowest value (inclusive): ? 10
To highest value (inclusive): ? 20

Next, enter the format information as in the previous example. The screen will clear and the values for all variables in the data set, for observations 10 through 20 will be listed. If you had left the Output File option to YES, this list will also be appended to the COLLIST.DOC
file. In addition, if you had the Idrisi Interface option set to YES, six files with a file extension of .VAL will have been created, containing observations 10 through 20 for each of the variables in the data set (except for NEIG, which is used as the polygon identifier). You get back to the List menu by pressing the Return key.

From the List menu, now type 6 to produce a complete listing of the contents of the GAUSS matrix COL_1. Enter col_1 in response to the prompt for the file name, 1 for the width and 0 for the number of decimals, as shown:

Listing of complete matrix

Enter the matrix filename (do not include .FMT),
or press Return for directory listing: col_1

Enter the format for the listing
Format is w.d with w as width d as number of decimals (default 12.6)
Enter the width (or 0 as default): ? 1
Enter the number of decimals: ? 0

After the last entry, the screen will clear and the matrix elements will be listed row by row, a screen full at a time. The first few lines of this listing, corresponding to the first three observations are illustrated in Table 6.3. Press Return to see each screen as well as at the end of the listing to get back to the List menu.

Table 6.3 Listing of COL_1.FMT File

<table>
<thead>
<tr>
<th>The contents of matrix col_1 with 49 rows and 49 columns:</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 1 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>1 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>0 1 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1</td>
</tr>
<tr>
<td>0 0 0 0 0 0 0 0 0</td>
</tr>
</tbody>
</table>

For a listing of the same matrix in sparse format, you now can use commands 7 and 8 on the List menu. Proceed in the same fashion as before and respond col_1 to the prompt for the file name. This is the only prompt you will get. The first few lines of the listing, corresponding to the first three observations are given in Table 6.4 for simple contiguity and in Table 6.5 for general weights. Note that the brief statements at the beginning of the listing that appear on your screen (and are illustrated in the tables) are not written to the output file (when the Output File option is set to YES). This will allow you to use the output file to re-create the weights matrix if needed, using respectively the D-1-3 (Sparse Ascii to Binary
Contiguity) and D-1-5 (Sparse Ascii to Spatial Weights) command sequences illustrated in Part III.

Table 6.4 Listing of COL_1.FMT File as Simple Contiguity

Binary contiguity in sparse form (GAL format) for matrix col_1

49
1  3
2  5  6
2  4
1  3  6  7
3  6
2  4  7  37  38  39

Table 6.5 Listing of COL_1.FMT File as General Spatial Weights

General spatial weights in sparse form for matrix col_1

49
1  2  1.00000000
1  5  1.00000000
1  6  1.00000000
2  1  1.00000000
2  3  1.00000000
2  6  1.00000000
2  7  1.00000000
3  2  1.00000000
3  4  1.00000000
3  7  1.00000000
3  37 1.00000000
3  38 1.00000000
3  39 1.00000000

6.7 Exercise

As an exercise, you can now produce listings for the data sets and weights matrices included on the \AFRICA and \EIRE directories. For the first, the data set is AFRICA (.DAT and .DHT) and the first order contiguity matrix is AFRICA_1.FMT. For the second, the data set is EIRE (.DAT and .DHT) and the first order contiguity matrix is EIRE_1.FMT.
7.1 Introduction

Once you have created a data set or matrix in the GAUSS format, the Data module of SpaceStat allows you to manipulate these files to a limited extent. Some simple data base functions are included to allow you some flexibility without having to resort to actual GAUSS commands. In this chapter, I focus on commands to add and remove variables or observations from a data set or matrix file. These commands are included in the Merge/Select menu of the Data module.

7.2 Merging Data Sets

7.2.1 Principle

Data sets in SpaceStat are essentially stored as matrices for which the rows correspond to the observations and the columns to the variables. In order to combine two existing data sets, they must share one common dimension: either the number of observations or the number of variables must be the same for the two. Both data sets must be present on the current working directory, or SpaceStat will generate an error message.

7.2.2 Merge by Observation: D–3–1

In the first instance of a merger of two data sets (same number of observations), the variables from the second data set are simply added as additional columns to the first one. Only those variables that are not already in the first data set are added (i.e., when the variable names are the same, the first copy is kept). The program makes sure that the number of observations is the same between the two data sets, and generates an error message when this is not the case. However, it is not able to check whether the observations are in the same order. In SpaceStat, the combination of the variables in two data sets is called Merge by Observation (add vars). It is the first item on the Merge/Select menu of the Data module and is invoked by the command sequence D–3–1.

After you enter the command sequence, you are prompted for the name of the first, second and new data sets. The latter may be the same as the first data set, although this is not recommended (unless you know exactly what is in each file). In the current version of SpaceStat, the size of the merged data set that can be created is limited by the memory available on your system. If you attempt to create a data set that is larger than the available
workspace, **SpaceStat** will crash with a GAUSS **Insufficient Workspace Memory** error message.

7.2.3 **Merge by Variable: D–3–2**

In the second instance of a merger of two data sets (same number of variables), the observations from the second data set are simply appended as additional rows at the bottom of the first one. The program makes sure that the variable names are identical between the two data sets and generates an error message when this is not the case. In **SpaceStat**, the combination of the observations in two data sets is called **Merge by Variable (add obs)**. It is the second item on the **Merge/Select** menu of the **Data** module and is invoked by the command sequence **D–3–2**.

After entering the command sequence, you are prompted for the name of the first, second and new data sets. The latter may be the same as the first data set, although this is not recommended (unless you know exactly what is in each file). In the current version of **SpaceStat**, the size of the merged data set that can be created is limited by the memory available on your system. If you attempt to create a data set that is larger than the available workspace, **SpaceStat** will crash with a GAUSS **Insufficient Workspace Memory** error message.

7.3 **Selecting and Deleting Variables**

7.3.1 **Principle**

You may select or delete a subset of variables from an existing GAUSS data set to form a new one, or to replace the existing one. If you choose the latter, you will lose the variables you did not select, since **SpaceStat** does not make a backup of a data set. For the most efficient use of this feature, you should use the function that requires the least number of variables to be specified: when you wish to select more than half the variables in the data set, you should use the delete command, and conversely, when you wish to delete more than half the variables, should use the select command.

Variables are selected or deleted for all observations in the data set. If you need to perform this operation for specific observations, you must also use the commands outlined in 7.4.

The data set for which you select or delete variables must be present on the current working directory, or **SpaceStat** will generate an error message. This will also be the case when you make a typing error while entering the name of the data set in response to a prompt. When this happens, you get back to the **Merge/Select** menu by pressing the **Return** key one or two times.
7.3.2  **Selecting Variables: D–3–3**

You select a subset of variables of a GAUSS data set by means of the third command in the **Merge/Select** menu of the **Data** module: **Select Variables from Data Set**. You invoke this function with the **D–3–3** command sequence. After you enter this sequence, you are prompted for the name of the existing data set and the name for the new data set that will contain only the selected variables. This second data set may be the same as the first one. However, in this instance, you will lose the variables in the first data set that were not selected.

After you specify the data sets, you are given a list of the variables in the first data set and are prompted to specify the ones to be selected. You may edit the variable name before you type the **Return** key, but you cannot change your selection afterwards. Also, only those variables for which the name you type exactly matches the variable name in the data set will be selected. It is therefore safest to always specify a different data set name for the second data set. Otherwise, if you make a slight mistake in the name of a variable you wish to select, it will be erased from the data set (since it will not have been selected).

You type **Return** in response to the last prompt when you are finished entering variable names. Next, the screen will clear and a list of the selected variables is given. Make sure to check whether this is indeed what you intended. You get back to the **Merge/Select** menu by pressing the **Return** key.

7.3.3  **Deleting Variables: D–3–4**

You delete a subset of variables of a GAUSS data set in the same way as outlined in the previous section. This is achieved by means of the fourth command in the **Merge/Select** menu of the **Data** module: **Delete Variables from Data Set**. You invoke this function with the **D–3–4** command sequence. After you enter this sequence, you are prompted for the name of the existing data set and the name for the new data set that will contain the variables that you did not delete. This second data set may be the same as the first one. However, in this instance, you will lose the variables in the first data set that were deleted.

After you specify the data sets, you are given a list of the variables in the first data set and are prompted to specify the ones to be deleted. You may edit the variable name before you press the **Return** key, but you cannot change your selection afterwards. Also, only those variables for which the name you type exactly matches the variable name in the data set will be deleted.

You type **Return** in response to the last prompt when you are finished entering variable names. Next, the screen will clear and a list of the variables in the new data set is given. Make sure to check whether this is indeed what you intended. You get back to the **Merge/Select** menu by pressing the **Return** key.
7.4 Selecting and Deleting Observations

7.4.1 Principle

In SpaceStat, you may select or delete a subset of observations from a GAUSS data set, or a matching subset of rows/columns from a spatial weights matrix. In order to do so, you need an auxiliary file in ascii format that contains the name of an indicator variable and a list of values for that variable. The indicator variable must be included in a GAUSS data set and must take a unique value for each observation. Those observations for which the value of the indicator variable matches a value in the list will be selected or deleted. Similarly, those rows/columns of a weights matrix will be selected or deleted for which the sequence numbers correspond to observations in a data set with values for the indicator variable that match the values in the list.

The data set or matrix and the auxiliary file must be present on the current working directory, or SpaceStat will generate an error message. This will also be the case when you make a typing error when entering the name of the data set or the auxiliary file in response to a prompt. When this happens, you get back to the Merge/Select menu by pressing the Return key one or two times.

Note that this feature lends itself well to an integration between a GIS and SpaceStat. In a GIS such as ARC/INFO, you can designate a subset of observations by means of a windowing or lasso function. This easily generates a list of polygon identifiers for the selected observations. This list, with the indicator variable as the first item, can be used as the auxiliary file to perform select/delete functions on data sets and weights matrices in SpaceStat.

7.4.2 Selecting Observations: D–3–5

You select a subset of observations from a GAUSS data set by means of the fifth command on the Merge/Select menu of the Data module. You invoke this function with the D–3–5 command sequence. After you enter this sequence, you are prompted for three file names:

- the file name for the existing data set
- the file name for the new data set
- the file name for the auxiliary input file

You should not include the file extension for the GAUSS data set, but you must specify the full filename for the auxiliary file.

After these prompts, the screen clears and you get a brief listing with the name of the new data set, its number of observations, number of variables, and the names of the variables in the new data set. You get back to the Merge/Select menu by pressing the Return key.
7.4.3 Selecting Rows/Columns from a Weights Matrix: D–3–7

You select a subset of rows/columns from a spatial weights matrix in the GAUSS format (or, from any GAUSS matrix) by means of the seventh command on the Merge/Select menu of the Data module. You invoke this function with the $D\text{-}3\text{-}7$ command sequence. After you enter this sequence, you are prompted for four file names:

- the file name for the existing weights matrix
- the file name for the new weights matrix
- the file name for the data set that contains the indicator variable
- the file name for the auxiliary input file

You should not include the file extension for the matrix files and the GAUSS data set, but you must specify the full filename for the auxiliary file.

After these prompts, the screen clears and you get a brief listing with the name of the new matrix and its dimensions. You get back to the Merge/Select menu by pressing the Return key.

Note that when the number of observations in the data set with the indicator variable does not match the dimensions of the spatial weights matrix, there may be unforeseen results. When the number of observations in the data set is larger than the dimensions of the matrix, an error message will be generated. However, the reverse case is not detected by SpaceStat.

7.4.4 Deleting Observations: D–3–6

You delete observations from an existing GAUSS data set with the sixth function in the Merge/Select menu of the Data module. You invoke this function with the $D\text{-}3\text{-}6$ command sequence. It operates exactly like the selection of observations (7.4.2), except that the observations with matching values for the indicator variable are deleted from the data set (instead of being selected). For details, see 7.4.2 above.

7.4.5 Deleting Rows/Columns from a Weights Matrix: D–3–8

You delete rows/columns from an existing spatial weights matrix with the eighth function in the Merge/Select menu of the Data module. You invoke this function with the $D\text{-}3\text{-}8$ command sequence. It operates exactly like the selection of rows/columns (7.4.3), except that the rows/columns with matching values for the indicator variable are deleted from the weights matrix (instead of being selected). For details, see 7.4.3 above.
7.5 Example

Before the features of the **Merge/Select** menu can be illustrated, you must create a scratch data set and a scratch GAUSS matrix. For example, you can simply copy the COL data set (both .DAT and .DHT files) to a file named COLWORK by means of the DOS *copy* command (first type F2 to get to the DOS prompt). Similarly, you can copy the COL_1.FMT file to a file named COLWORK.FMT. To avoid contaminating your original data, make sure to keep a clean copy of the Columbus data and contiguity weights. You also will need an ascii file to determine the observations to be selected or deleted. The file COLSEL.ASC on the \COLUMBUS directory contains the variable NEIG as an indicator variable and a list of observations from 1 to 20. You can copy this file into your working directory, or, alternatively create a similar auxiliary file with a text editor.

You start the example by typing `d` (or `D`) to move to the **Data** module, followed by 3 (or move the cursor down to the third line and press *Return*) for the **Merge/Select** menu. Next, type 3 to start the **Select Variables from Data Set** command. Enter `colwork` in response to the prompt for the first data set, `colwork2` for the new data set and select the variables **NEIG**, **CRIME**, and **INCOME**, as shown below:

```
Selecting variables from a data set

Enter the data set filename (do not include .DAT or .DHT),
or press Return for directory listing: colwork

New data set for the selected variables

Enter the data set filename (do not include .DAT or .DHT),
or press Return for directory listing: colwork2

Choose the variable(s) from the following list
(each variable name should correspond exactly to one in the list):

NEIG  CRIME  INCOME  HOUSING  X  Y  EW

Enter the variable name, or press Return to stop

Variable name: NEIG
Variable name: CRIME
Variable name: INCOME

Finally, press the *Return* key in response to the last prompt to end the queries for variable names. The screen will clear and a brief message will appear to confirm that the new data set has been created, as shown:

```
A new dataset has been created
```
The dataset colwork2
with 49 observations on 3 variables:
NEIG  CRIME  INCOME

Press the Return key to get back to the Merge/Select menu. Next, type 6 for the Delete Observations from Data Set command and answer colwork2 for the data set, colwork3 for the new data set, and colsel.asc for the ascii input file, as shown:

Deleting observations from a data set
An Ascii file with indicator variable and values must be used

Enter the data set filename (do not include .DAT or .DHT),
or press Return for directory listing: colwork2

New data set with observations deleted
Enter the data set filename (do not include .DAT or .DHT),
or press Return for directory listing: colwork3

File name with chosen values for indicator variable

Enter the name of the Ascii input file
File name is: colsel.asc

After this last prompt, the screen clears and a brief message appears, indicating that the new data set has been created, as in:

A new dataset has been created
The dataset colwork3
with 29 observations on 3 variables:
NEIG  CRIME  INCOME

Press the Return key to get back to the Merge/Select menu.

In spatial data analysis, it is always important to keep a one-to-one correspondence between the dimension of the data set (i.e., the number of observations) and the dimension of the associated spatial weights matrix. For example, after you reduce the number of observations in data set COLWORK3 to 29, you should also extract a 29 by 29 submatrix from the original weights matrix COLWORK.FMT. To achieve this, type 8 to start the Delete Rows/Cols from a Weights Matrix command. In response to the prompts, enter the filename colwork for the old and the new matrix, colwork for the data set with the indicator variable, and colsel.asc for the auxiliary file, as shown:

Deleting rows/columns from a spatial weights matrix
An Ascii file with indicator variable and values must be used

Enter the matrix filename (do not include .FMT),
or press Return for directory listing: colwork
New weights matrix file with rows/cols deleted

Enter the matrix filename (do not include .FMT),
or press Return for directory listing: colwork

Data set with indicator variable

Enter the data set filename (do not include .DAT or .DHT),
or press Return for directory listing: colwork

File name with chosen values for indicator variable

Enter the name of the Ascii input file
File name is: colsel.asc

After this last prompt, the screen clears and a brief message appears, indicating that the new matrix has been created, as in:

A new weight matrix has been created
The matrix file colwork
with 29 rows and 29 columns

Press the Return key to get back to the Merge/Select menu.

The final message differs between generic matrices and spatial weights matrices. When the latter are used in a command of the Tools, Explore or Regress modules, two additional columns are added to the actual weights, containing information on the dimension, row-standardization, etc. When a submatrix is created using the D-3-7 or D-3-8 command sequences, the information in the additional columns is updated (e.g., the submatrix is row-standardized for the new values) and appended to the weights. Consequently, the dimensions reported in the final message will give 2 more columns than rows (e.g., in the previous example, the dimensions would be 29 rows and 31 columns).

Note that in the example, you have specified the same file name for the old and new weights matrix. As a result, the original file has been erased. In practice, you should always be very careful to keep the weights matrix for the full data set intact, in order to easily extract submatrices for various spatial subsets.

To illustrate the merge feature, you can now concatenate the two data sets COLWORK2 and COLWORK3 into a new data set COLWORK4. Both existing data sets have the same variables, so that you can append the observations from COLWORK3 to those in COLWORK2 by means of the D-3-2 command sequence. From the Merge/Select menu, type 2 (or move the cursor down to the second line and press Return). Next, enter the three filenames just mentioned in response to the prompts, as shown:
Merging two data sets by variable (add observations)

First Data Set:

Enter the data set filename (do not include .DAT or .DHT),
or press Return for directory listing: colwork2

Second Data Set:

Enter the data set filename (do not include .DAT or .DHT),
or press Return for directory listing: colwork3

Merged Data Set:

Enter the data set filename (do not include .DAT or .DHT),
or press Return for directory listing: colwork4

After this last prompt, the screen clears and a brief message appears, summarizing the characteristics of each data set and of the merged data set, as in:

Dataset colwork2 has 3 variables and 49 observations
Dataset colwork3 has 3 variables and 29 observations
The merged dataset has been created as colwork4
It contains 78 observations on 3 variables:
NEIG CRIME INCOME

Press the Return key to get back to the Merge/Select menu.

7.6 Exercise

To further practice these commands, you can experiment with other subsets of the Columbus data, or alternatively, use copies of the Africa or Eire data sets and weights matrices. You can always check the contents of the new data sets and weights by means of the commands in the Data List menu outlined in the previous chapter.
CHAPTER 8

CREATING AND TRANSFORMING VARIABLES IN A DATA SET

8.1 Introduction

Once you have constructed a data set in the GAUSS format, the Data module of SpaceStat allows you to create certain types of new variables and to transform existing variables. These functions are included to allow you some flexibility without having to resort to actual GAUSS commands. In this chapter, I focus on commands included in the Var Create and Var Transform menus of the Data module.

8.2 Creating New Variables

8.2.1 Principle

The standard way to add variables to a data set in SpaceStat consists of three steps. First, the observations for the new variables must be included in an ascii file. Next, this ascii file must be converted to the GAUSS format by means of the D-1-1 command sequence (Data-Input-Ascii to Data Set). Finally, the newly created data set must be merged with an existing one to add the new variables, by means of the D-3-1 command sequence (Data-Merge/Select-Merge by Observation).

For a limited number of variable types, this rather involved process can be avoided by means of the commands included in the Var Create menu of the Data module. These commands allow for the creation of a variable either directly, or based upon information incorporated in an existing variable in the data set. There are eight such commands in the Var Create menu:  
- Relabel Variables \((D-4-1)\): changing the name of one or more variables in the data set;  
- Recode Variables \((D-4-2)\): changing the values taken by a variable in the data set for all observations that are in given range;  
- Create Dummy Variables (Categories) \((D-4-3)\): create a series of dummy variables based on the values taken by a categorical variable in the data set  
- Create Dummy Variables (Range) \((D-4-4)\): create a dummy variable that takes on a value of 1 for all observations for an existing variable that are within a given range;  
- Create Constant \((D-4-5)\): create a new variable with all observations equal to a given value;  
- Create Observation Numbers \((D-4-6)\): create a new variable that contains the sequence number for all observations;
- **Create Uniform Random Variable** (*D*-4-7): create a new variable that follows the uniform distribution for a given range;

- **Create Normal Random Variable** (*D*-4-8): create a new variable that follows a normal distribution for a given mean and standard deviation.

Details on invoking these commands are given below.

### 8.2.2 Relabel Variables: *D*-4-1

You can change the name of a variable that is already contained in a GAUSS data set by means of the first command of the **Var Create** menu in the **Data** module. You invoke this command with the *D*-4-1 command sequence.

You are prompted for the name of the existing data set and presented with a list of the names of the variables contained in it. Next, you are asked to enter the name for each variable you wish to relabel, followed by the new name, or by **Return** if you accept the default. The default is to keep the name as is. In other words, by pressing the **Return** key in response to the query for the new variable, nothing is changed. This is a way to avoid unintentional changes. You stop the query for variable names by typing **Return** in response to the prompt for the existing variable.

Next, you will see a brief message, summarizing the new make-up of the data set, including the number of observations, number of variables, and a list of variable names. You should carefully check the list of new variable names to make sure that all your intended changes have been made. Only those variables in the data set whose names match exactly what you typed in will have been altered. You get back to the **Var Create** menu by pressing the **Return** key.

### 8.2.3 Recode Variables: *D*-4-2

You can change the values taken on by a variable that is already contained in a GAUSS data set by means of the second command of the **Var Create** menu in the **Data** module. You invoke this command with the *D*-4-2 command sequence.

You are prompted for the name of the existing data set and presented with a list of the names of the variables contained in it. Next, you are asked to enter the name for each variable you wish to recode, followed by the new name, or by **Return** if you accept the default. The default is to keep the name as is. In other words, by pressing the **Return** key in response to the query for the new variable, the original values for the variable will be changed. This is not recommended, unless you are fairly sure of what you are doing. You stop the query for variable names by typing **Return** in response to the prompt for the existing variable.
Next, you are asked to specify a lower bound (inclusive) and an upper bound (exclusive) to which the recoding will apply, and the new value that should be substituted. All observations that take on values within the specified range will be changed to the new value, for all the variables entered in response to the prompt. If you wish to change a single value to a new magnitude, you must specify the same value for the lower and upper bound. Again, this is done for all variables specified. This is particularly useful to change observations from zero to a very small value (say 0.00001), e.g., before performing a log transformation.

After entering this information, the screen clears and you will see a brief message, summarizing the new make-up of the data set, including the number of observations, number of variables, and a list of variable names. You can easily check the new values for the recoded variables by means of one of the Data List commands, such as D-8-3, List Selected Variables. You get back to the Var Create menu by pressing the Return key.

### 8.2.4 Dummy Variables: D-4-3 and D-4-4

There are two ways to create dummy variables in SpaceStat. In the first approach, invoked by the D-4-3 command sequence, there must be a categorical indicator variable present in the data set. Based on the value taken by this variable, one dummy variable is created for each category, i.e., with values of 1 corresponding to the category and 0 otherwise. The last 1 or 2 characters of the variable name for the dummy variable correspond to the category, and the same prefix is used for all. The maximum number of categories allowed is 20. When the maximum value for the categorical indicator variable exceeds this limit, an error message is generated. The only way to create dummy variables for more than 20 categories is to divide the operation into subsets and to zero-out the other categories by means of the Recode Variables function (D-4-2). When the categorical indicator variable is binary (i.e., takes values of 0 and 1), only one additional dummy is created, equal to its complement, and with _1 as the last two characters of the variable name.

In the second approach, the dummy variable takes a value of 1 for observations on an indicator variable that fall within a given range, and is 0 otherwise. This is invoked by the D-4-4 command sequence. In this instance, only one dummy variable is created, the name of which must be specified explicitly. Similar to the convention used for the Recode Variables command, you can change observations with a specific value to a value of 1 by setting the lower and upper bound of the value range to the same magnitude.

In both commands, you are first prompted for the name of the existing data set, and presented with a list of the names of the variables contained in it. For the categorical dummies, you are asked for the name of the indicator variable. You are next queried for the prefix that the dummy variable name should take (the default for this is CAT_). This prefix will be followed
by the index number that corresponds to the category in the indicator variable. For the range dummies, you are queried for the name of the existing variable (with the range information) and the name for the dummy variable. If you respond the same name to both queries (this is the default), the original information will be erased and only the dummy variable will remain. This is not recommended, unless it is intended. Next, you are asked to specify values for the lower (inclusive) and upper (exclusive) bounds of the range.

After entering this information, the screen clears and you will see a brief message, summarizing the new make-up of the data set including the number of observations, number of variables and a list of variable names. You can easily check the values for the dummy variables by means of one of the Data List commands, such as D-8-3, List Selected Variables. You get back to the Var Create menu by pressing the Return key.

8.2.5 Constant: D-4-5

You can create a variable equal to a constant value by means of the D-4-5 command sequence. Such a variable can subsequently be used in various algebraic operations, as outlined in Chapter 9. Note that you never need to specify a constant term to perform a regression analysis, since this is done internally in SpaceStat (see Part V).

After invoking the command, you are prompted for the name of the data set to which the constant will be added, asked to specify a name for the variable, and queried for the constant value. After entering this information, the screen clears and you will see a brief message, summarizing the new make-up of the data set including the number of observations, number of variables and a list of variable names. You can easily check the value for the constant by means of one of the Data List commands, such as D-8-3, List Selected Variables. You get back to the Var Create menu by pressing the Return key.

8.2.6 Observation Number: D-4-6

You can create a variable that contains the sequence numbers for the observations in a data set by means of the D-4-6 command sequence. The new variable takes on values from 1 to N, where N is the dimension of the data set. It is often useful to have such a variable when you take various subsets of a data set. This variable will make it easier for you to keep track of the various subsets. It is also necessary to create a variable with the observation sequence numbers (and to name it OBS) when you want to take advantage of the Indicator Variable option. If you set this option to yes, with the default of OBS, you must create this variable by means of the D-4-6 command sequence, unless it was included at the time of the creation of the data set. Otherwise, an error message will be generated whenever the Indicator Option is relevant.
After invoking the command, you are prompted for the name of the data set to which the new variable will be added, and asked to specify its name. After entering this information, the screen clears and you will see a brief message, summarizing the new make-up of the data set including the number of observations, number of variables and a list of variable names. You can easily check the value for the indicator variable by means of one of the Data List commands, such as \texttt{D-8-3}, \texttt{List Selected Variables}. You get back to the \texttt{Var Create} menu by pressing the \texttt{Return} key.

8.2.7 \textbf{Random Variates: D-4-7 and D-4-8}

There are two functions included in \texttt{SpaceStat} to generate random variates: one for uniform random variates (\texttt{D-4-7}) and one for normal random variates (\texttt{D-4-8}). They are based on the random number generators included in \texttt{GAUSS}. The random seed for these generators is the one specified in the \texttt{Random Number Seed} option (Option 9) of \texttt{SpaceStat}. Unless you want to always get the same results, you should change this option to the use of the system clock to set the seed, by specifying -1 in response to the prompt for Option 9 (see Chapter 2).

The random variates created with these commands can be used in a number of spatial transformations to simulate data sets with various forms of spatial dependence, as outlined in Part III.

After invoking the command, you are prompted for the name of the data set to which the new variable will be added, and asked to specify its name. For the uniform variate, you are next queried for the range of the distribution, while for the normal variate, you are queried for the mean and standard deviation. After entering this information, the screen clears and you will see a brief message, summarizing the new make-up of the data set including the number of observations, number of variables and a list of variable names. You can easily check the value for the random variable by means of one of the \texttt{Data List} commands, such as \texttt{D-8-3}, \texttt{List Selected Variables}. You get back to the \texttt{Var Create} menu by pressing the \texttt{Return} key.

8.3 \textbf{Transforming Existing Variables}

\texttt{SpaceStat} includes an easy way to transform variables that are already present in a \texttt{GAUSS} data set. This is carried out by means of the commands in the \texttt{Var Transform} menu of the \texttt{Data} module. All the commands are interactive and based on \texttt{GAUSS} functions. They allow you to carry out the transformation without having to resort to actual \texttt{GAUSS} programming. The transformations are invoked by means of the \texttt{D-5} command sequence, followed by the number corresponding to the specific function, as shown below. Eight transformations are included:
- Ln: D-5-1
  \ln(x)

- Exp: D-5-2
  \exp^x

- Integer Power: D-5-3
  \(x^a\) (including negative values for a)

- Inverse: D-5-4
  \(1/x\) (also obtained by using -1 for the integer power transformation)

- Square Root: D-5-5
  \(\sqrt{x}\)

- Absolute Value: D-5-6
  \(|x|\)

- Deviation from Mean: D-5-7
  \(x-\mu\) (where \(\mu\) is the mean of \(x\))

- Standardize: D-5-8
  \((x-\mu)/\sigma\) (where \(\mu\) is the mean and \(\sigma\) the standard deviation of \(x\))

All transformations operate in the same fashion. After invoking the command, you are prompted for the name of the existing data set and presented with a list of the names of the variables contained in it. Next, you are asked to enter the name for each variable you wish to transform, followed by the new name, or by Return if you accept the default. The default is to keep the name as is. In other words, by pressing the Return key in response to the query for the new variable, the original values for the variable will be changed. This is not recommended, unless you are fairly sure of what you are doing. You stop the query for variable names by typing Return in response to the prompt for the existing variable. This is the extent of your input for all but the inverse power transformation. For the latter, you must also specify the value for the exponent. This must be an integer value.

SpaceStat generates error messages when an invalid transformation is attempted, such as taking the logarithm or the square root of a negative number, or dividing by zero. If an error message is encountered for one of the variables, none of the transformations is carried out. You get back to the Var Transform menu by pressing the Return key. You will have to start all over again.

If no errors are encountered, the screen clears and you will see a brief message, summarizing the new make-up of the data set, including the number of observations, number of variables and a list of variable names. You can easily check the value for the transformed
variables by means of one of the Data List commands, such as D-8-3, List Selected Variables. You get back to the Var Transform menu by pressing the Return key.

8.4 Example

Since you will be creating several additional variables, you should use the same scratch data set as for the examples of the Merge/Select menu. Use the data set COLWORK you created before, or, if you have not already done so, copy the COL data set (both .DAT and .DHT files) to a file named COLWORK by means of the DOS copy command (first type F2 to get to the DOS prompt). To avoid contaminating your original data, make sure to keep a clean copy of the Columbus data.

You start the example by typing d (or D) to move to the Data module, followed by 4 (or move the cursor down to the fourth line and press Return) for the Var Create menu. Next, type 8 to start the Create Normal Random Variable command. As before, enter colwork in response to the prompt for the data set. Next, enter normal (or any other variable name you choose) for the new variable, and 20 and 5 respectively for its mean and standard deviation, as shown below:

Create a normal random variable

Enter the data set filename (do not include .DAT or .DHT),
or press Return for directory listing: colwork

Enter a name for the new variable: normal
Enter the mean for the normal random variate: ? 20
Enter the standard deviation: ? 5

If you enter a name for the normal variate that is already taken by a variable in the data set, SpaceStat will generate an error message. Otherwise, after the last prompt, the screen will clear and a brief message will appear with the data set name, number of observations, number of variables and the list of variable names, as shown:

The new variable has been added to the dataset
The dataset colwork
with 49 observations on 8 variables:

NEIG  CRIME  INCOME  HOUSING  X  Y  EW  NORMAL

Press the Return key to get back to the Var Create menu.

Next, type d (or D) to get back to the main Data menu, followed by 5 (or move the cursor down to the fifth line and press Return) to start the Var Transform menu. Type 1 (or Return) to choose the natural logarithm transformation and respond colwork to the prompt for the data set, as before. Next, you are shown a list of the variables in the data set and asked to specify the name for the existing variable to be transformed and for the new variable.
Enter *normal* for the former (or whichever name to specified for the normal random variate) and *ln_norm* (or any other name you choose) for the new variable, as shown:

Choose the variable(s) from the following list
(each variable name should correspond exactly to one in the list):

NEIG CRIME INCOME HOUSING X Y EW
NORMAL

Enter the name for each existing variable first, or press Return to stop
Then enter the name for the new variable, or press Return for the default
Default is same name as existing variable

Existing variable: normal  New Variable (Return for default): ln_norm
Existing variable:

Press Return to end the queries. Note that if you type the same name for both existing and new variable, the original observations will be erased. Next, the screen will clear and a brief message will appear with the data set name, number of observations, number of variables and the list of variable names, as shown:

The new variables have been added to the dataset
The dataset colwork
with 49 observations on 9 variables:

NEIG CRIME INCOME HOUSING X Y EW NORMAL
LN_NORM

Press the Return key to get back to the Var Transform menu.

As a final illustration, type 8 (or move the cursor down to the last line and press Return) to invoke the Standardize command. Again, enter *colwork* for the data set. You will see a list of the variables in the data set, which now includes NORMAL and LN_NORMAL (or the variable names you specified in the two earlier steps). Enter these two names in response to the query for the existing variable and choose *snormal* and *sln_norm* (or any other variable name), as shown:

Choose the variable(s) from the following list
(each variable name should correspond exactly to one in the list):

NEIG CRIME INCOME HOUSING X Y EW
NORMAL LN_NORM

Enter the name for each existing variable first, or press Return to stop
Then enter the name for the new variable, or press Return for the default
Default is same name as existing variable

Existing variable: normal  New Variable (Return for default): snormal
Existing variable: ln_norm  New Variable (Return for default): sln_norm
Existing variable:
Again, you press *Return* to end the queries. The screen will clear and a brief message will appear with the data set name, number of observations, number of variables and the list of variable names, as shown:

```
The new variables have been added to the dataset
The dataset colwork
with 49 observations on 11 variables:
    NEIG  CRIME  INCOME  HOUSING  X  Y  EW  NORMAL
    LN_NORM  SNORMAL  SLN_NORM
```
Press the *Return* key to get back to the **Var Transform** menu.

### 8.5 Exercise

To further practice these commands, you can experiment with the full range of variable creation and transformation commands, either for the Columbus data set, or for another data set. You can always check the contents of the new data sets by means of the commands in the **Data List** menu outlined in Chapter 6.
CHAPTER 9

ALGEBRA

9.1 Introduction

The Var Algebra and Mat Algebra menus of the Data module in SpaceStat allow you to carry out some simple algebraic operations on two variables in an existing GAUSS data set, or on two existing GAUSS matrices. These functions are included to allow you some flexibility without having to resort to actual GAUSS commands.

9.2 Simple Algebraic Operations on Two Variables

The first four commands in the Var Algebra menu of the Data module allow you to perform simple addition, subtraction, multiplication and division of two variables that are contained in an existing GAUSS data set. Both variables must be in the same data set. The result of the operation can either be stored as a new variable, or may replace an existing variable (whose original value is lost). The same operation can be carried out for as many variable pairs as you wish. In combination with the commands in the Var Create menu, these algebraic operations allow you to construct a wide range of variable transformations. For example, to rescale a variable, you could first create a constant equal to the scaling factor (using the D-4-5 command sequence) and then multiply the selected variable by this constant. Similarly, you can create a random variable consisting of a deterministic part (e.g., the sum of several variables, or a weighted sum of variables) and a random part (generated by means of the D-4-7 or D-4-8 command sequence), by adding the two together.

The specific command sequences to invoke the algebraic operations are as follows:
- addition: D-6-1
- subtraction: D-6-2
- multiplication: D-6-3
- division: D-6-4

The four commands all operate in the same fashion. You are first prompted for the name of the data set containing the variables. Next, you are given a list of the variable names in the selected data set and asked to specify the name for the first, second and new variable. You don’t have to specify the latter explicitly: if you press Return in response to the query, the new variable takes on the name of the first variable. As a result, the observations on the first variable will be replaced by the new values, which is not recommended, unless it is intended. You end the series of prompts by pressing the Return key in response to the query for an additional first variable. If no errors are encountered, the screen will clear and a brief
message will appear, indicating that the new variables have been added to the data set, and summarizing the contents of the data set in the usual fashion (number of observations, number of variables, and a list of variable names). You get back to the Var Algebra menu by pressing Return.

Only those variable pairs for which the names you type exactly match a variable name in the data set will yield a result. You can check the extent to which this is the case in the summary list of variable names in the data set. When you attempt to carry out division by zero, you will receive an error message. No transformations will have been carried out. In such a case, you must start over again, after pressing Return to get back to the Var Algebra menu.

9.3 Linear Combination of Variables (D-6-5).

The fifth command of the Var Algebra menu allows you to compute a weighted sum or linear combination of any number of variables in an existing data set. For example, a new variable z may be created as a weighted sum of three variables $x_1$, $x_2$ and $x_3$ as:

$$z = \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_3$$

where the $\alpha_1$, $\alpha_2$, and $\alpha_3$ are coefficients you must specify. Clearly, if you use the estimates of a regression analysis for the coefficients $\alpha$, you can compute predicted values by means of this command. Also, by adding the predicted value thus obtained to a random error term, generated by means of the D-4-8 command sequence, you can simulate values for the dependent variable in a regression model.

You invoke the linear combination by means of the D-6-5 command sequence. You are first prompted for the name of the data set that contains the variables, followed by a query for the name of the new variable. The latter must be different from any name already in the data set, or an error message will be generated. Next, you are given a list of the variables in the data set, and asked to specify the variable name and the coefficient value for each element in the weighted sum. You stop the prompts by pressing Return in response to the query for the name of the existing variable. If no errors are encountered, the screen will clear and a short summary message is listed, with an update on the contents of the data set (number of observations, number of variables, and a list of variable names). You get back to the Var Algebra menu by pressing the Return key.

Note that you can also perform a rescaling of variables by means of this command by specifying only a single variable name in response to the prompts.
9.4 Special Variables used in Spatial Regression Analysis

9.4.1 Principle

The Regress module of SpaceStat includes a number of special regression specifications, referred to as spatial regressions. They are outlined in detail in Part VI of the tutorial. Spatial regression models incorporate certain locational aspects of the data in an explicit fashion. These locational aspects are called spatial drift, when they are continuous with location, and spatial regimes, when they correspond to discrete spatial subsets of the observations. Examples of the former are trend surface and spatial expansion models, while the latter are incorporated in spatial ANOVA and in regression with structural change, or spatial regimes.

The special variables needed to implement a trend surface, spatial expansion or spatial structural change model are all computed internally in the Regress module. However, for the purposes of exploratory analysis, or to visualize certain results, it may be useful to create such variables explicitly. This is accomplished by means of commands six through eight in the Var Algebra menu.

9.4.2 Expansion Variables: D–6–6

In the spatial expansion method, a new set of variables (referred to as expanded variables) is constructed by combining each original variable in the model (referred to as the initial variables) with a polynomial in a set of auxiliary variables (referred to as the expansion variables). Expansion variables may be any set of variables that explain a drift in the parameters of the model. In spatial data analysis, the coordinates of the location for each observation are often used for this purpose (hence, this method was originally referred to as the spatial expansion method).

Formally, if the initial variable were x, and the expansion variables were z₁ and z₂, then a linear expansion would necessitate the product of x with z₁ and of x with z₂ as additional (expanded) variables. Similarly, a quadratic expansion would need the product of x with z₁², of x with z₂² and of x with z₁ times z₂ as expanded variables. In the current version of SpaceStat, only linear and quadratic polynomials are implemented. Technical details on the expansion method are given in Part VI of the tutorial.

The creation of expanded variables is invoked by means of the D–6–6 command sequence. You are first prompted for the name of the data set that contains both initial and expansion variables. Next, you are asked to specify the names for the initial variables (after a list of all variable names in the data set is shown). This query is in the usual fashion and is terminated by pressing the Return key. It is followed by a prompt for the variable names of the expansion variables. Again, you end the query by pressing the Return key. The final prompt is for the order of the expansion, either 1 for a linear expansion, or 2 for a quadratic
expansion. After you finish entering this information, the screen is cleared and a summary of the contents of the updated data set is given. This includes the number of observations, number of variables, and a list of the variable names. The expanded variables can be recognized by a distinct set of prefixes A_, AA_, B_, AB _, BB_, etc. that precede the name of each initial variable. Each of the letters in the prefix corresponds with an expansion variable you specified in response to the prompts: A stands for the variable given at the first prompt, B for the one at the second prompt, etc. The prefix either consists of one letter (for a first order expansion), or is made up of two letters (for a quadratic expansion). Of course, these variable names may always be altered subsequently by means of the Data-Var Create-Relabel Variables (D-4-1) command sequence. You get back to the Var Algebra menu when you press the Return key after the data set summary.

9.4.3 Trend Surface Variables: D-6-7

In a trend surface regression, the explanatory variables consist of a polynomial in the coordinates of the observations, as outlined in more detail in Part VI. Such variables can be created explicitly in a SpaceStat data set by means of the D-6-7 command sequence. In fact, a polynomial of any order in two variables may be created in this fashion. The new variables consist of all terms in the polynomial of an order less than or equal to the order specified in response to the prompts. Note that trend surface variables may also be constructed "the hard way" by means of a combination of the Integer Power command in the Var Transform menu (D-5-3) and the Multiply command in the Var Algebra menu (D-6-3). The explicit Trend Surface command differs from this procedure only in that it creates all new variables in one operation.

You invoke this function by means of the D-6-7 command sequence. You are prompted for the name of the data set that contains the coordinates. Next, you are given a list of all the variables contained in this data set and asked to enter the names for the two coordinates. You must specify two variable names, or an error message is generated. The last query is for the order of the trend surface polynomial. After you enter a value, the screen is cleared and a summary of the contents of the updated data set is presented, consisting of the number of observations, the number of variables, and a list of the variable names. The terms of the trend surface are given variable names consisting of XX_ and YY_, corresponding to the first and second coordinate, followed by the power to which they are raised. A variable name such as XX_2YY_1 thus corresponds to the product of the second power of the first coordinate with the first power of the second coordinate. As before, these variable names may always be altered subsequently by means of the Data-Var Create-Relabel Variables (D-4-1)
command sequence. You get back to the Var Algebra menu when you press the Return key after the data set summary.

9.4.4 Spatial Regimes: D-6-8

In regression models with spatial structural change, the coefficients take on a different value in each spatial subset of the observations (for further details, see Part VI). In order to implement this model, each explanatory variable must be transformed to as many new variables as there are regimes. The new variables are zero for all observations that do not fall in the regime to which they correspond. In other words, if there are two regimes, two new variables will be created, one with nonzero values in the first subset of observations, the other with nonzero values in the second subset of observations. This is easily accomplished by multiplying the variable in question with a set of dummy variables that each correspond to one of the regimes. In SpaceStat, this can be done by first creating this set of dummy variables, using the Var Create-Create Dummy Variables (Categories) command (D-4-3), followed by a Var Algebra-Multiply operation (D-6-3). The Var Algebra-Regimes command combines these two steps into a single procedure, but is otherwise equivalent.

You invoke this operation by means of the D-6-8 command sequence. You are first asked to specify the data set that contains the variables needed for the procedure. Next you are given a list of the variables in the data set and prompted to enter the names of the ones that need to be converted into regimes. This is followed by a query for the name of the categorical variable that defines the regimes. The conventions used for this variable are the same as those outlined for the Var Create-Create Dummy Variables command in Chapter 8. After you enter the name for this variable, the screen is cleared and a summary of the contents of the updated data set is presented, consisting of the number of observations, the number of variables, and a list of the variable names. The variables for the regimes are distinguished by their last two characters. These consist of an underscore followed by the sequence number for the regime, such as _1 and _2 in the case of two regimes. The _1 always refers to the category with the lowest value, typically 1 for a categorical variable that takes on a set of consecutive integer values, or zero for a binary categorical variable. As before, these variable names may always be altered subsequently by means of the Data-Var Create-Relabel Variables (D-4-1) command sequence. You get back to the Var Algebra menu when you press the Return key after the data set summary.

9.5 Principal Components of Variables (D-6-9)

When a large number of explanatory variables are used in a regression analysis, there is often a problem with multicollinearity. The multicollinearity results from the presence of
a linear relationship (or correlation) between some of all of the explanatory variables. This is addressed in more detail in Part V. For trend surface models and specifications that follow the spatial expansion approach, a high degree of multicollinearity is unavoidable, due to the way in which the explanatory variables are constructed (as powers and cross products). It is sometimes suggested to replace these variables by their principal components, which are uncorrelated (orthogonal) by construction. The principal components of a set of variables, organized as the columns of a matrix X, are obtained from the eigenvectors of the matrix XX. The elements of each eigenvector yield weights that are used to construct new variables as weighted sums of the original ones. There are as many so-called principal components as original variables.

In SpaceStat, you can carry out the computation of principal components for a set of variables in both the Algebra menu of the Data module and in the Describe menu the Explore module (see Part IV). The two are equivalent in terms of computations, but in the Explore module the components are not added to an existing data set. Also, more statistical detail is given in the Explore module, while no information on the eigenvalues or eigenvectors is provided in the Data module. In the Data module, you invoke this operation by means of the D-6-9 command sequence. You are prompted for the name of the data set that contains the variables. Next, you are given a list of the variables in the selected data set and asked to specify the ones for which you wish to compute the principal components. You finish entering variables by typing Return in response to the last prompt. The screen is cleared and a summary of the contents of the updated data set is presented, consisting of the number of observations, the number of variables, and a list of the variable names. The variables for the principal components are given as PCOMP_1, PCOMP_2, etc., up to the number of original variables specified. Note that if a name of a variable that you entered does not exactly correspond to a name in the data set, then this variable will not be included in the computation of principal components. You can check for this by making sure that the highest sequence number of the PCOMP_ variables corresponds to the number of variables you entered.

As before, you may alter the variable names for the components by means of the Data-Var Create-Relabel Variables (D-4-1) command sequence. You get back to the Var Algebra menu when you press the Return key after the data set summary.

9.6 Element by Element Operations on Two Matrices

The first four commands in the Mat Algebra menu of the Data module allow you to carry out some simple element by element operations on two existing GAUSS matrices. These operations are applied to the individual matching elements in two matrices, instead of to the matrix as a whole. In order for these operations to work, the two matrices must have the
same dimensions. If they don’t, SpaceStat will generate an error message. An error message is also generated when one of the elements in the second matrix would lead to a division by zero.

With $a_{ij}$ as the element in row $i$ and column $j$ of the first matrix (A), and $b_{ij}$ as the corresponding element in the second matrix (B), the four commands included are:

- **element by element addition:** \( D-7-1 \)
  \[ a_{ij} + b_{ij} \]

- **element by element subtraction:** \( D-7-2 \)
  \[ a_{ij} - b_{ij} \]

- **element by element multiplication:** \( D-7-3 \)
  \[ a_{ij} \cdot b_{ij} \]

- **element by element division:** \( D-7-4 \)
  \[ a_{ij} / b_{ij} \]

The element by element operations are performed on the matrix as it is. No special treatment is provided for spatial weights matrices. In other words, if the weights matrices contain the two extra columns with information on dimension, row-standardization, etc., the latter will be included in the element by element operations. This is often not the intention and should be avoided by first transforming the weights matrix back to a standard matrix format, by means of the \( T-2-4 \) command sequence in the **Tools** menu (see Part III).

The four element by element matrix commands operate in the same fashion. You are prompted for the name of the first matrix, the name of the second matrix and the name for the new matrix (you may also first press Return to see a list of GAUSS matrices in the current directory). If no errors are encountered, the screen is cleared and you see a brief message with the dimensions of the new matrix. Pressing the Return key gets you back to the **Mat Algebra** menu.

In order to see the actual contents of the new matrices, created with any of the element by element operations, you should use the \( D-8-6 \) command sequence (**Data-List-List Matrix**) discussed in Chapter 6.

### 9.7 Common Matrix Operations

The last four commands in the **Mat Algebra** menu allow you to perform several common matrix operations, without having to resort to programming in GAUSS. Included are the following functions:

- **matrix multiplication:** \( D-7-5 \)
- **matrix inverse:** \( D-7-6 \)
- **determinant of a matrix:** \( D-7-7 \)
- trace of a matrix: \[ D-7-8 \]

The last three operations only apply to square matrices, and an error message will be generated if the matrix you specify does not conform to this. Therefore, if you wish to apply the commands to spatial weights matrices, you should first convert them to the standard matrix format by means of the \[ T-2-4 \] command sequence in the \textbf{Tools} menu (see Part III). An error message is also generated if you attempt to compute the inverse of a singular matrix. The results of the determinant and trace functions are written to the output file specified in the \textbf{Output} File option (if it is set to YES), but otherwise only screen output is given.

The commands all operate in the same fashion. You are queried for the name of the matrix file, or for the name of the first and second matrix (for matrix multiplication). After you enter the file name, the computations are carried out and you see a brief message indicating the result. You move back to the \textbf{Mat Algebra} menu by pressing the \textit{Return} key.

In order to see the actual contents of the new matrices, created with the matrix multiplication or matrix inverse commands, you should use the \[ D-8-6 \] command sequence (\textbf{Data-List-List Matrix}) discussed in Chapter 6.

\subsection*{9.8 Example}

Since you will be creating several additional variables, you should use the data set COLWORK you created before as a scratch data set. If you did not carry out the illustrations in the previous chapters, copy the COL data set (both .DAT and .DHT files) to a file named COLWORK by means of the DOS \textit{copy} command (first type \textit{F2} to get to the DOS prompt). To avoid contaminating your original data, make sure to keep a clean copy of the Columbus data. In the illustration below, the listings will be for the case where you created all the variables used in the example of the previous chapter. If you did not, or if you tried additional examples, your listings may slightly differ from the ones given below.

For the example, you will be creating the variables for a quadratic trend surface in the coordinates X and Y, to use these to construct spatial expanded variables for CRIME. You will also create two new CRIME variables, corresponding to the two spatial regimes expressed by the dummy variable EW. You start the example by typing \textit{d} (or \textit{D}) to move to the \textbf{Data} module, followed by \textit{6} (or move the cursor down to the sixth line and press \textit{Return}) for the \textbf{Var Algebra} menu. Next, type \textit{7} to start the \textbf{Trend Surface} command. As before, enter \textit{colwork} in response to the prompt for the data set. Next, enter \textit{x} and \textit{y} for the coordinates, and \textit{2} for the order of the polynomial, as shown below:

\begin{verbatim}
Trend surface variables
Enter the data set filename (do not include .DAT or .DHT),
\end{verbatim}
Enter the names for the coordinates (2 variables only)
Choose the variable(s) from the following list
(each variable name should correspond exactly to one in the list):
   NEIG  CRIME  INCOME  HOUSING  X  Y  EW
   NORMAL  LN_NORM  SNORMAL  SLN_NORM

Enter the variable name, or press Return to stop
Variable name: x
Variable name: y

Enter the order of the polynomial: ? 2

After this last entry, the screen will clear and a brief message will appear with the data set name, number of observations, number of variables and the list of variable names, as shown:

The new variables have been added to the dataset
The dataset colwork
with 49 observations on 14 variables:
   NEIG  CRIME  INCOME  HOUSING  X  Y  EW  NORMAL
   LN_NORM  SNORMAL  SLN_NORM  XX_2  YY_2  XX1_YY1

Note the special variable names used for the polynomial terms in the trend surface. Press the Return key to get back to the Var Algebra menu.

Next, type 6 (or move the cursor up one line and press Return) to start the Expansion command. Again, enter colwork for the data set. Next, answer crime in response to the prompt for the initial variable, as shown:

Spatial Expansion variables

Enter the data set filename (do not include .DAT or .DHT), or press Return for directory listing: colwork

Enter the names for the initial variables
Choose the variable(s) from the following list
(each variable name should correspond exactly to one in the list):
   NEIG  CRIME  INCOME  HOUSING  X  Y  EW
   NORMAL  LN_NORM  SNORMAL  SLN_NORM  XX_2  YY_2  XX1_YY1

Enter the variable name, or press Return to stop
Variable name: crime

Press Return and continue with xx_2, yy_2 and xx_1yy_1 in response to the query for the expansion variables, as follows:
Enter the names for the expansion variables

Choose the variable(s) from the following list
(each variable name should correspond exactly to one in the list):

<table>
<thead>
<tr>
<th>NEIG</th>
<th>CRIME</th>
<th>INCOME</th>
<th>HOUSING</th>
<th>X</th>
<th>Y</th>
<th>EW</th>
</tr>
</thead>
<tbody>
<tr>
<td>NORMAL</td>
<td>LN_NORM</td>
<td>SNORMAL</td>
<td>SLN_NORM</td>
<td>XX_2</td>
<td>YY_2</td>
<td>XX1_YY1</td>
</tr>
</tbody>
</table>

Enter the variable name, or press Return to stop

Variable name: xx_2
Variable name: yy_2
Variable name: xx_1yy_1

Again, press Return to end the prompts and proceed by choosing 1 (for a linear expansion) as the order of the expansion, as shown:

Enter the order of the expansion, 1 for linear, 2 for quadratic
Expansion order is: ? 1

After this last entry, the screen will clear and a brief message will appear with the data set name, number of observations, number of variables and the list of variable names, as shown:

The new variables have been added to the dataset
The dataset colwork
with 49 observations on 17 variables:

<table>
<thead>
<tr>
<th>NEIG</th>
<th>CRIME</th>
<th>INCOME</th>
<th>HOUSING</th>
<th>X</th>
<th>Y</th>
<th>EW</th>
<th>NORMAL</th>
<th>LN_NORM</th>
<th>SNORMAL</th>
<th>SLN_NORM</th>
<th>XX_2</th>
<th>YY_2</th>
<th>XX1_YY1</th>
<th>A_CRIME</th>
<th>B_CRIME</th>
<th>C_CRIME</th>
</tr>
</thead>
</table>

Note the special variable names for the expanded variable CRIME. The A_CRIME variable corresponds to the product of CRIME with \(x^2\), the B_CRIME to the product with \(y^2\) and the C_CRIME to the product with \(xy\). Press the Return key to get back to the Var Algebra menu.

Next, type 8 (or move the cursor down two lines and press Return) to start the Regimes command. Again, enter colwork for the data set. Next, answer crime in response to the prompt for the variable names, as shown:

Regime (structural change) variables
Enter the data set filename (do not include .DAT or .DHT),
or press Return for directory listing: colwork

Enter the names for the initial variables
Choose the variable(s) from the following list
(each variable name should correspond exactly to one in the list):

<table>
<thead>
<tr>
<th>NEIG</th>
<th>CRIME</th>
<th>INCOME</th>
<th>HOUSING</th>
<th>X</th>
<th>Y</th>
<th>EW</th>
<th>NORMAL</th>
<th>LN_NORM</th>
<th>SNORMAL</th>
<th>SLN_NORM</th>
<th>XX_2</th>
<th>YY_2</th>
<th>XX1_YY1</th>
<th>A_CRIME</th>
<th>B_CRIME</th>
<th>C_CRIME</th>
</tr>
</thead>
</table>

Enter the variable name, or press Return to stop
Variable name: crime
Variable name:

Press Return and continue with ew in response to the query for the categorical variable, as follows:

Select the categorical variable to be used to create regimes

Enter the variable name, or press Return to stop
Variable name: ew

After this last entry, the screen will clear and a brief message will appear with the data set name, number of observations, number of variables and the list of variable names, as shown:

The new variables have been added to the dataset
The dataset colwork
with 49 observations on 19 variables:
    NEIG   CRIME  INCOME  HOUSING     X     Y     EW   NORMAL
    LN_NORM  SNORMAL  SLN_NORM   XX_2   YY_2  XX_1YY_1  A_CRIME  B_CRIME
    C_CRIME  CRIME_1  CRIME_2

Note the special variable names for the new regime variables, CRIME_1 corresponding to zero values of EW, and CRIME_2 to values of one for EW. Press the Return key to get back to the Var Algebra menu.

9.9 Exercise

To further practice the commands of the Var Algebra and Mat Algebra menus, you can experiment with the full range of commands, either for the Columbus data set, or for another data set. You can use the various contiguity matrices in the \COLUMBUS or other directories to practice manipulating matrices, or read in matrices from ascii files, using the D-1-2 (Data-Input-Ascii to Matrix) command sequence. You can always check the contents of the new data sets and matrices by means of the commands in the Data List menu outlined in Chapter 6.
PART III

SPATIAL WEIGHTS MATRICES
CHAPTER 10

CREATING A SPATIAL WEIGHTS MATRIX FROM A FILE

10.1 Introduction

One of the major distinguishing characteristics of spatial data analysis (as opposed to a-spatial analysis) is that the spatial arrangement of the observations is taken into account. This is formally expressed in a spatial weights matrix, $W$, with elements $w_{ij}$, where the $ij$ index corresponds to each observation pair. The nonzero elements of the weights matrix reflect the potential spatial interaction between two observations. This may be expressed in different ways, such as simple contiguity (having a common border), distance contiguity (having centroids within a critical distance band), or in function of inverse distance or squared inverse distance. Elements that are zero indicate a lack of spatial interaction between two observations (by convention, the diagonal elements of the weights matrix are set to zero).

**SpaceStat** allows for many different ways of creating spatial weights matrices, including interfacing with a Geographical Information System (GIS) to obtain information on the spatial arrangements of observations. In this chapter, I outline the construction of a weights matrix from information contained in an ascii file.

10.2 Ascii Input File

The direct conversion of a full matrix from an ascii format into a GAUSS format is discussed in Chapter 5. However, this is not a very efficient approach to create spatial weights matrices, since all elements of the matrix must be contained in the ascii file, including the zeros. Two sparse formats that may be used to avoid listing all non-interacting observation pairs are outlined in Chapter 3. One of these pertains to simple contiguity. The corresponding ascii input file contains the number of observations, and for each observation it lists its identifier, the number of contiguous observations and their identifiers (this is referred to as the GAL format). The second format for the sparse input file must be used for general spatial weights, although it is applicable to simple (binary) contiguity as well. The input file contains the number of observations, and for each nonzero weights element it lists the identifier for the row and column, followed by the value of the weight. Details on these formats are given in Chapter 3. In both approaches, the identifier (for each observation) may either be the sequence number of the observation, or the unique value taken by an indicator variable that is included in an existing GAUSS data set. In all, there are thus four options, contained in the **Input** menu of the **Data** module:
- Sparse Ascii to Binary Contiguity (Row-Col Numbers): D-1-3
- Sparse Ascii to Binary Contiguity (Indicator Variables): D-1-4
- Sparse Ascii to Spatial Weights (Row-Col Numbers): D-1-5
- Sparse Ascii to Spatial Weights (Indicator Variable): D-1-6

In order to use this approach, you must previously have recorded the structure of contiguity between observations, either from visually inspecting a map, such as Figure 10.1 for the Columbus neighborhoods, or by running a function (or macro) within a GIS to create the file.

10.3 Creating a Weights Matrix

You invoke the creation of a spatial weights matrix from an ascii input file by means of the \texttt{D-1} (\texttt{Data-Input}) command sequence, followed by the appropriate option number, 3, 4, 5 or 6. You will be prompted for the name of the ascii input file and for the name you wish to give to the weights matrix. The latter is stored in the GAUSS matrix format, which, similar to the data set format, is binary. This makes weights matrices unreadable by standard text editors or word processors. Matrices in the GAUSS format are identified by an \texttt{.FMT} extension. Note that you should never include this extension in your response to a prompt for a matrix file name.

After you enter this information, press the \texttt{Return} key and the screen will clear. The input file is read and a preliminary weights matrix is constructed, exemplified by a brief message. Next, you are asked whether or not you wish to carry out a symmetry check. In most applications, the original weights matrix information should lead to a symmetric matrix, and it is thus good practice to request a symmetry check. The check will list potential row-column pairs where there may be a problem with lack of symmetry. No weights matrix is saved to disk when problems with symmetry are encountered. If you still wish to create the weights matrix from the same ascii file, you have to start over and skip the symmetry test on the second try. At the end, a brief message will appear, listing the file name and the dimensions of the new matrix. You get back to the \texttt{Input} menu by pressing \texttt{Return}.

10.4 Example

In this example, I will only illustrate the creation of a simple contiguity matrix, since the other cases operate in essentially the same way. To carry out the example, you will need a file with the contiguity information for the Columbus neighborhoods. This is included in the \texttt{\COLUMBUS} directory as \texttt{COLSPA.ASC}. You may also want to use the file \texttt{COLBAD.ASC}. The latter contains an error in order to illustrate the symmetry check included in \texttt{SpaceStat}.

The contents of the file \texttt{COLSPA.ASC} are shown in Appendix E. The first few lines of the file are listed in Table 10.1 below. You can visually check the information for neigh-
neighborhood contiguity on the map in Figure 10.1. Note that the first item in the file gives the number of observations (49). The second line indicates that neighborhood 1 has 3 contiguous neighborhoods. They are listed on the third line as 2, 5, and 6. The same set of information is given for the second neighborhood next (neighborhood 2 has 4 contiguous neighborhoods), etc.

Table 10.1 Columbus Neighborhood Contiguity

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>6</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>7</td>
<td>37</td>
<td>38</td>
<td>39</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>37</td>
<td>39</td>
<td>40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>5</td>
<td>7</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Type \texttt{d} (or \texttt{D}) to get to the menu for the \textbf{Data} module. The first item in this menu is \textbf{Input}. Type \texttt{1} (or press \textit{Return}, since the cursor is on the first line) to get the menu for \textbf{Input}. The third item on this menu is \textbf{Sparse Ascii to Binary Contiguity (Row-Col numbers)}. Type \texttt{3} (or move the cursor down to the third line and press \textit{Return}) to start this command.

Next, you get two prompts for filenames. You should enter the filenames \texttt{colspa.asc} (for the ascii file) and \texttt{colw1} (or any other name you choose to give to the new matrix file), each followed by a \textit{Return}, as shown below:

\begin{verbatim}
Binary contiguity from sparse matrix info

Enter the name of the Ascii input file
File name is: colspa.asc

Enter the matrix filename (do NOT include .FMT)
File name is: colw1
\end{verbatim}

After you enter the second file name, the screen clears and you get the message:

Building the weights matrix ...

This is followed by a query for a symmetry check:

\begin{verbatim}
Check on the symmetry of the weights matrix?
(type \texttt{n} for no check, or \textit{Return} for check):
\end{verbatim}
In most instances, it is good practice to make sure that the contiguity matrix is symmetric. Of course, there are always exceptions where you may want to have an asymmetric matrix, such as when you model directional flow data.

A problem with lack of symmetry may result when there are typographical errors in the ascii input file. A common error is that a contiguity between two spatial units is only recorded once. In terms of the weights matrix, this means that a non-zero row-column combination does not have a matching column-row non-zero element.

You check for symmetry by pressing the Return key. For the Columbus example, the data are symmetric, and you will see the following message:

```
Matrix is symmetric in its contiguities

The weights matrix has been created and saved as
The matrix file colw1
with 49 rows and 49 columns
```

Pressing the Return key once more gets you back to the Data Input menu.

You have now created a first order contiguity matrix for the Columbus neighborhoods as the file COLW1.FMT. You can also find this file on the \COLUMBUS directory.

### 10.5 Common Problems

What happens when the matrix you tried to create is not symmetric? To get an idea, use your text editor to change the second and third lines in the COLSPA.ASC file from

```
1  3
2  5  6
```

to:

```
1  2
2  5
```

In other words, you have eliminated the contiguity between neighborhood 1 and neighborhood 6. However, on line 13 of the file (also the last line in Table 10.1), neighborhood 1 is still listed as contiguous to neighborhood 6.

If you want to save the original file, first copy COLSPA.ASC to COLBAD.ASC before editing the file (go to DOS by means of function key F2 and use the copy command; return to SpaceStat by typing exit).

---

1. Note that you also have to adjust the number of contiguities on the second line from 3 down to 2. If you do not, you may get unpredictable results. Only in some instances is SpaceStat able to capture errors, i.e., when the values in the file result in an attempt to read beyond the end of the file. However, in many other instances, this error does not occur and a seemingly correct weights matrix is created. Unless you carry out the symmetry check (which will catch most of these input errors), you may end up with a nonsensical weights matrix.
Now repeat the steps above, i.e., type D-1-3 (or 3 if you are still in the Data Input menu), but now enter colbad.asc and colbad in response to the respective prompts. If you had entered colw1 again as the filename for the weights matrix, the correct file will be overwritten with the wrong one, unless you request the symmetry check! Press the Return key in response to the symmetry check request. You will hear a beep and get the following message:

WARNING: Non-symmetric weights matrix
Check the input file for possible problems with:
Cell 1
Missing matching cells 6
Cell 6
Missing match in cells 1

In most instances, the warning will give you enough information to track down problems in the input file. When your input file fails the symmetry check, no weights matrix is created. For example, if you now type F2 to get to DOS and carry out a dir command, you will not find a COLBAD.FMT file on your current directory (unless you created it previously). Type exit to get back to SpaceStat.

10.6 Exercise

You can now also create a first order contiguity matrix for your own data (by visually inspecting a map) or for the Cliff-Ord Irish data (by inspecting the map on p. 207 of Cliff and Ord, 1981). As with data sets, you will have to record this information in an ascii file. Make sure to list the number of observations as the first element in this file. Don’t be discouraged when you don’t pass the symmetry check the first time: this is very common and is the main reason for using a GIS to record the spatial arrangements of observations.

As indicated in Chapter 6, SpaceStat also has a feature that allows you to convert the weights matrix you constructed back to a sparse ascii file, e.g., if you want to make sure that the correct information was recorded. You do this by means of the Data List command 7 List Binary Contiguity in Sparse Form: enter D-8-7 and give the name for the weights matrix at the prompt. The results will be listed on the screen and also written to the file specified as the output file (option 2 in the Options menu). You can check the output file by going to DOS (press F2) and using your text editor (you get back into SpaceStat by typing exit).

---

2. Make sure that you only press Return in response to the symmetry request and do not enter anything else. For any other key (e.g., y or Y), the check will be skipped and a weights matrix created.
CHAPTER 11

CREATING A CONTIGUITY MATRIX FOR A REGULAR LATTICE

11.1 Introduction

When you work with data that are arranged on a regular square or rectangular lattice (or grid), the contiguity structure can be derived directly, without having to resort to a visual inspection of a map. In a regular grid, contiguity can be defined in three ways: having a common border (rook criterion), having a common corner (bishop criterion), and having either a border or corner in common (queen criterion). Typically, this results in four contiguous grids according to the rook or bishop criteria, and eight contiguous grids following the queen criterion. For grid cells that are at the border of the lattice, some of the contiguities are lost.

SpaceStat allows you to create a binary contiguity matrix for a regular lattice using any of the three criteria. The three criteria are options in the Tools Raster Wts menu. The only information needed are the dimensions of the lattice. The contiguity matrix is saved in the GAUSS matrix format (characterized by a FMT file extension).

11.2 Creating a Contiguity Matrix

The contiguity matrix for a regular lattice is created by means of the commands in the Raster Wts menu of the Tools module. You reach the Tools menu by typing t (or T) from any menu in SpaceStat. Next, type 3 (or move the cursor down to the third line and press Return) for the Raster Wts menu. You now have three options, corresponding to the three contiguity criteria:

- the Rook criterion: T-3-1
- the Bishop criterion: T-3-2
- the Queen criterion: T-3-3.

After you choose one of options by typing the corresponding command sequence, you will be prompted for three items:

- the file name for the weights matrix to be computed
- the vertical dimension of the lattice
- the horizontal dimension of the lattice

The dimensions of the lattice are the number of rows and columns of grid cells. This is NOT the dimension of the resulting weights matrix. The latter is the product of the vertical and horizontal dimensions. For example, a square lattice with 7 rows and 7 columns consists of 49 cells. Consequently, the weights matrix will be of dimension 49 by 49.1
In the construction of the weights matrix, the cells are numbered by starting in the upper left corner and moving left to right. In other words, the top row on the 7 by 7 lattice would contain observations 1 to 7, the second row observations 8 to 14, etc.

With the information on the dimensions, SpaceStat constructs the weights matrix and saves it in the GAUSS matrix format. The screen is cleared, and a brief message is listed, summarizing the dimensions of the new matrix. You get back to the Raster Wts menu by pressing the Return key.

11.3 Example

In this example, you will create a simple contiguity matrix for the African data set. As you may recall, this data set is for a 7 by 7 regular square lattice containing 49 grid cells. You will use the Rook criterion to define contiguity.

Start by typing t (or T) from any menu to reach the Tools menu. Next, type 3, followed by 1 (or Return) for the Rook criterion. In response to the prompts, enter afrook (followed by a Return) as the file name for the contiguity matrix and 7 (followed by a Return) for both the vertical and horizontal dimensions, as shown:

```
Binary contiguity matrix for regular lattice: rook case
File for contiguity matrix
Enter the matrix filename (do NOT include .FMT)
File name is: afrook
Vertical dimension of lattice (number of rows):    : ? 7
Horizontal dimension of lattice (number of columns): ? 7
```

After this, the screen will clear and the following message appears:

```
Binary contiguity matrix afrook of dimension  49
       corresponding to a regular lattice
with    7 rows and    7 columns
using the rook criterion
```

You will find that the file AFROOK.FMT has been created on your current directory.

11.4 Exercise

You can now also create a contiguity matrix for the Bishop and Queen criteria, following the same steps as above. You can check the contents of these matrices by means of the Data List command 7 List Binary Contiguity in Sparse Form that was outlined in Chapter 6. Enter D–8–7 and give the name for the weights matrix at the prompt.

1. The size of the weights matrix that can be created depends on the amount of RAM installed on your machine. In the GAUSS matrix format all matrices are stored in double precision, i.e., taking 8 bytes per matrix element.
CHAPTER 12

CREATING A SPATIAL WEIGHTS MATRIX BASED ON DISTANCE

12.1 Introduction

Contiguity between spatial units can also be defined in function of the distance that separates them. In practice, this distance must be computed between two points that are uniquely associated with the spatial units, such as their centroids or other meaningful points. Two units are then considered to be contiguous if these points are less than a specified critical distance apart.

In SpaceStat, two types of spatial weights matrices can be constructed from information on the distance between observations. The first yields a simple contiguity matrix by using a critical distance cut-off point, while the other type uses any integer power of the inverse distance between two observations as the weights. In addition, the distances themselves may be used as well, although this is not very useful for most of the techniques discussed in this tutorial. The creation of these spatial weights is included in the Distance Wts menu of the Tools module.

The process of constructing the weights matrices consists of two steps. In the first, you must compute a distance matrix from X and Y coordinates. Both euclidean and great circle distance functions are included. The coordinates must be contained in an existing data set that conforms to the GAUSS data set format. The distance matrix is saved in the GAUSS matrix format (characterized by a .FMT file extension). In the second step, the information in the distance matrix is used to create a spatial weights matrix, which is also saved in the GAUSS matrix format. The last command in the Distance Wts menu allows you to compute a few summary characteristics of a distance matrix.

12.2 Creating a Distance Matrix Based on Euclidean Distance: T-4-3

You start the computation of distances between observations in a data set by means of the T-4-3 command sequence (Tools-Distance Wts-Create Distance Matrix). Following this, you will be prompted for three items:
- the filename of the data set that contains the coordinates
- a filename for the distance matrix you are to compute
- the variable names that correspond to the X and Y coordinates (these do not have to be labeled as X and Y).

After entering this information, the screen will clear and the distance computation begins, indicated by a message. In some instances, the distance matrix for the coordinates you specified will contain off-diagonal values that are less than 1 (by convention, the diagonal elements are
set to zero). Clearly, this will be a function of the scale in which the coordinates are expressed. Distances less than 1 may create problems in the computation of inverse distance weights. Therefore, if this happens, a warning message will be listed and you will be asked if you wish to have the distance matrix re-scaled such that the smallest distance equals 1. In most instances, this is a good idea and you should respond affirmatively by typing y (or Y). Following your response, a brief message is listed, confirming that the distance matrix has been saved. You get back to the Distance Wts menu by pressing the Return key.

12.3 Creating a Distance Matrix Based on Great Circle Distance: T-4-4

When the spatial scale at which you carry out the data analysis is global in nature, the straight line euclidean distance is no longer an appropriate measure. Instead, distance between two points should be based on the concept of great circle distance or arc distance on a sphere. The Create Arc Distance Matrix command on the Distance Wts menu provides a way to compute such great circle distance from information on latitude (X) and longitude (Y).

The coordinates must be stored as a decimal number in an existing GAUSS data set. Therefore, if you start from information on latitude and longitude in degrees, you must first convert it to decimal format. You can implement this in SpaceStat by means of the Linear Combination function of the Data module (command sequence D-6-5). First, you must incorporate the information on degrees, minutes and second as three separate variables in a GAUSS data set for both latitude and longitude. Next, you invoke the D-6-5 command sequence and compute the decimal expression for each coordinate as a linear combination of the form:

\[ \text{decimal} = \text{degrees} \times 1 + \text{minutes} \times 0.01666667 + \text{seconds} \times 0.00027778 \]

The resulting variables are in the proper format to be used as input for the latitude (X) and longitude (Y) coordinates in the arc distance computation. The arc distance \(d_{ij}\) between two locations \(i\) and \(j\) is calculated as follows:

\[ d_{ij} = 3959.0 \times \text{arc cos} \left( \text{cos} |Y_i - Y_j| \times \sin X_i \times \sin X_j + \cos X_i \times \cos X_j \right) \]

where the X and Y are first transformed to radians, as:

\[ X = (90 - \text{LAT} + \pi) / 180 \]
\[ Y = \text{LON} + \pi / 180. \]

You invoke the computation of great circle distances by means of the T-4-4 command sequence (Tools-Distance Wts-Create Arc Distance Matrix). The prompts are the same as for the euclidean distance computation:

- the filename of the data set that contains the coordinates
- a filename for the distance matrix you are to compute
- the variable names that correspond to the latitude and longitude coordinates.
The rest of the messages are the same as well: in the unlikely case that any of the arc distances are less than 1, you will be queried for a rescaling. The procedure ends with a brief message that the distance matrix has been saved. You get back to the Distance Wts menu by pressing the Return key.

12.4 Summary Characteristics of a Distance Matrix: T-4-5

A few summary characteristics of a distance matrix can be computed by invoking the T-4-5 command sequence (Tools-Distance Wts-Characteristics of Distance Matrix). The only prompt is for the name of the distance matrix. After you enter this, the screen is cleared and the following items are displayed:

- name of the matrix
- dimension
- average distance
- distance range (maximum less minimum)
- minimum distance
- first quartile
- median distance
- third quartile
- maximum distance.

These characteristics are computed from the non-zero upper-triangular elements of the distance matrix. They are only displayed to the screen (i.e., they are not written to the output file). They are also provided as part of the weights creation commands (see below).

SpaceStat verifies whether the file name you specified is in the correct GAUSS format. If it is not, or if the file is not present on the current directory, an error message will be generated. However, SpaceStat is not able to determine whether the matrix you specified is indeed a distance matrix. As long as the matrix is square, or corresponds to the format of a weights matrix, the characteristics will be computed.

You get back to the Distance Wts menu by pressing the Return key.

12.5 Creating a Simple Contiguity Matrix from Distances: T-4-1

If you have earlier created a distance matrix, you can proceed to the computation of binary contiguity weights directly by means of the T-4-1 command sequence (Tools-Distance Wts-Distance to Binary Weights). Next, you are prompted for two file names:

- the filename for the existing distance matrix (stored in GAUSS matrix format)
- the filename prefix for the contiguity matrices that are to be created
The filename prefix for the contiguity matrices should not have more than 5 characters (if it does, up to the last three may be truncated). The use of a prefix allows you to create several contiguity matrices in one operation when different distance bands are derived from the same distance matrix. Next, the screen is cleared and the summary characteristics of the distance matrix are listed. These are the same ones as provided by the **Characteristics of Distance Matrix** command \((T-4-5)\).

The list of characteristics is followed by queries for an upper bound (inclusive) and a lower bound (non-inclusive) for the distance bands. Those bounds will be used to define contiguity. If you enter *Return* in response to the first prompt for the upper bound, a full matrix of ones will be created. Such a matrix is NOT acceptable as a contiguity matrix, but it may be useful in other matrix manipulations, e.g., for commands in the **Mat Algebra** menu of the **Data** module. In all other cases, you will continue to be prompted for an upper and lower bound, until you press the *Return* key in response to the upper bound query. At that point, the contiguity matrices are created by converting all matrix elements with distances within the specified bounds to one (and setting all other elements to zero). The weights are saved in the GAUSS matrix format (with a .FMT file extension).

To indicate successful completion of this process, the screen is cleared again and a brief message appears, listing the distance band(s) and corresponding file(s). Each of the matrix filenames is of the form prefix\_i, where prefix is the one specified in response to the prompt and i is the sequence number of the upper and lower bound interval for the distance band. When only one set of bounds is specified (and when a full matrix is created), the file name for the contiguity matrix consists of the prefix only.

You get back to the **Distance Wts** menu by pressing the *Return* key.

### 12.6 Creating Inverse Distance Spatial Weights: \(T-4-2\)

The \(T-4-2\) command sequence (Tools-Distance Wts-Inverse Distance Weights) allows you to use the information in a distance matrix to create general spatial weights that are integer powers of the inverse distance. For example, this can be used to construct measures of potential interaction between two observations as squared inverse distance weights, in accordance with the gravity model of spatial interaction. After invoking the command, you are prompted for two file names:

- the filename for the existing distance matrix (stored in GAUSS matrix format)
- the filename prefix for the weights matrices that are to be created

The filename prefix for the weights matrices should not have more than 5 characters (if it does, up to the last three may be truncated). The use of a prefix allows you to create several weights matrices in one operation for different integer powers computed from the same
distance matrix. All powers pertain to the same distance band. However, to construct distance weights for powers of different distance bands, you must carry out a separate command.

After you enter the file names, the screen is cleared and the summary characteristics of the distance matrix are listed. These are the same ones as provided by the Characteristics of Distance Matrix command (T-4-5).

The list of characteristics is followed by a query for an upper bound (inclusive) and a lower bound (non-inclusive) for the distance bands. Those bounds should be used when the potential interaction reflected in the inverse distance weights is assumed to fall to zero beyond a certain critical distance. In contrast, when continuous change of interaction with distance is assumed, a full matrix of inverse distance weights (or, inverse distance powers) must be created. This is carried out by entering Return in response to the prompt for the upper bound.

Next, you are asked to specify the integer powers to which the inverse distance should be raised. If you press Return in response to the first prompt, a default of 1 (or, simple inverse distance) is implemented. In contrast, if you entered another power, the queries will continue until you press Return.

At that point, the contiguity matrices are created by converting all matrix elements with distances within the specified bounds to their inverse (and setting all other elements to zero), raised to the specified power(s). The weights are saved in the GAUSS matrix format (with a .FMT file extension).

To indicate successful completion of this process, the screen is cleared again and a brief message appears, listing the powers used, distance bands and corresponding files. Each of the matrix filenames is of the form prefix_i, where prefix is the one specified in response to the prompt and i is the sequence number of the power you entered in response to the queries. As in the simple contiguity case, when only one matrix is created, its file name corresponds to the prefix.

You get back to the Distance Wts menu by pressing the Return key.

12.7 Example

In a previous example, you created the data set for the Columbus crime data, COL. This data set, which is also in the \COLUMBUS directory, contains the variables X and Y for the centroid coordinates of the neighborhoods. You will now use these coordinates to construct three contiguity matrices, for increasing distance bands of 0 to 5, 0 to 10 and 0 to 15. Later, these will be needed for the computation of the Getis-Ord measures of spatial association in Part IV.
First, you need to create a distance matrix based on the X and Y coordinates. Type `t` (or `T`) from any menu to reach the Tools menu. Next type 4 for Distance Wts, followed by 3 to start the Create Distance Matrix command. The first prompt is for the filename of the data set. Enter `col` in response to this. Alternatively, if you press Return at this prompt, a list of data sets in the current directory is given, followed by another prompt for a file name, as illustrated below.

```
Create euclidean distance matrix
Data set that contains the coordinate variables

Enter the data set filename (do not include .DAT or .DHT),
or press Return for directory listing: Return
Files in default directory:
    COL

Enter file name: col
```

If you had created several data sets in the current directory, their file names would be listed in addition to COL. The next set of queries is for the filename of the distance matrix and the coordinate variables. Enter `coldis` for the former and `x` and `y` for the latter, as shown below:

```
File name for distance matrix
Enter the matrix filename (do NOT include .FMT)
File name is: coldis

Choose the coordinate variables from the following list
    NEIG  CRIME  INCOME  HOUSING  X  Y  EW

Variable for x-coordinates: x
Variable for y-coordinates: y
```

Next, the following message appears:

```
Computing distance matrix ...
```

For very large matrices, this message may be present for some time. However, for the Columbus data, it quickly disappears, the screen is cleared and you get the warning for the scale problem mentioned above. Type `y` (or `Y`) in response to the prompt and the distance computation is completed, as shown:

```
WARNING:
Smallest distance is < 1, rescale coordinates? (Y or N): y

Distance matrix rescaled such that the smallest distance = 1

Distance matrix saved as coldis
```
Pressing the *Return* key gets you back to the **Tools Distance Wts** menu. Note that the distance matrix is also present as the file COLDIS.FMT in the \COLUMBUS directory.

In the second step of this process, you will use the just computed distance matrix to create binary contiguity matrices for three distance bands. From the **Distance Wts** menu, type *1* for the **Distance to Binary Weights** command. The first two prompts are for the file name of the distance matrix and the prefix for the weights matrix. Type *coldis* for the former and *cold* for the latter (each followed by *Return*), as shown below:

**Binary weights from a distance matrix**

Enter the matrix filename (do not include .FMT), or press Return for directory listing: **coldis**

Enter the matrix filename (do NOT include .FMT)

File name is: **cold**

Next, the screen is cleared and selected characteristics of the distance matrix are listed. For the distance matrix coldis, these appear as:

**Characteristics of distance matrix coldis**

- **Dimension:** 49
- **Average distance between points:** 13.5984
- **Distance range:** 35.3976
- **Minimum distance between points:** 1.00000
- **Quartiles**
  - **First:** 8.01943
  - **Median:** 12.8458
  - **Third:** 18.3170
- **Maximum distance between points:** 36.3976

This is followed by a series of queries for the upper and lower bounds of the distance bands. Note that the upper bound is inclusive (less than or equal to), while the lower bound is not (strictly greater than). You type *Return* at the upper bound prompt to stop the queries. If you press *Return* at the first prompt, a matrix is created that consists of all ones, except for the diagonal elements. This is not very meaningful as a contiguity matrix and should not be used for this purpose. However, for other purposes, you may want to construct such a matrix and this is the way to achieve it.

For the three distance band matrices you want to construct, you respond to the prompts as shown below:

Enter the upper and lower distance to define a distance band

or press Return to stop
(press Return at first prompt for full matrix)

Upper <= :  5       Lower >  :  0
Upper <= :  10      Lower >  :  0
Upper <= :  15      Lower >  :  0
Upper <= : Return

Next, the screen is cleared and the following message appears:

Weights matrices created and saved for distance bands
Band 1 lower bound: 0.00000 upper bound: 5.00000 saved as: cold_1
Band 2 lower bound: 0.00000 upper bound: 10.0000 saved as: cold_2
Band 3 lower bound: 0.00000 upper bound: 15.0000 saved as: cold_3

Pressing the Return key gets you back to the Distance Wts menu.

You have now created three contiguity matrices, COLD_1, COLD_2 and COLD_3. These
are also present on the \COLUMBUS directory.

12.8 Exercise

You may now also create inverse distance spatial weights for the Columbus data, for
different combinations of distance bands and integer powers. Also, if you have coordinates
for the spatial units in your own data set, you can now proceed to create a set of distance
based contiguity matrices.

Alternatively, you could practice with the Irish data. However, since this data set does
not include centroids, I have included an ascii file with them as EIREXY.ASC in the \EIRE
directory. Note that the file follows the format needed to create a data set, i.e., the first entry
is 26 (the number of observations), next is 2 (the number of variables), followed by X and Y
as variable names.

Before you can create the distance and contiguity matrices for the the Irish example,
you must add the X and Y coordinates to your data set. You accomplish this in two steps.
First, you convert the ascii file to a GAUSS data set, say EIREXY, using the Data - Input -
Ascii to Data Set command sequence (D-1-1) illustrated before. Next, you need to merge
this data set with the existing one. For this, you use the Data - Merge/Select - Merge by
Observation (Add Vars) command sequence (D-3-1). Enter the name of the existing Irish
data set in response to the first prompt (i.e., type eire), the file name for the coordinates in
response to the second (i.e., type eirexy), and the first filename again in response to the third
prompt (i.e., type eire). The merged data set will have the same name as the original one,
but will now also contain the X and Y as variables. You proceed from here to create the distance
and contiguity matrices as outlined above.
CHAPTER 13

CREATING A SPATIAL WEIGHTS MATRIX FROM A GIS

13.1 Introduction

By far the most efficient, and for large spatial data sets the only feasible strategy to construct a spatial weights matrix is to use a digitized boundary file for the spatial units. This boundary file is typically contained within a Geographic Information System (GIS). SpaceStat provides a number of ways to interact with GIS boundary files. These are contained in the GIS Interface menu of the Data module. The first two options in this menu pertain to the ARC/INFO vector-based system, the others to raster-based systems (IDRISI, OSU-MAP and a generic raster format).

In order to exploit the ARC/INFO interface, you must first execute a series of AML (Arc Macro Language) commands from within ARC/INFO.1 These commands generate an ascii file that can be read by SpaceStat and from which SpaceStat constructs a spatial weights matrix. This is an example of what in the GIS literature is referred to as "loose coupling" of the GIS and a spatial analysis module (in this case SpaceStat), since it necessitates a two-step process.2

The interface with the raster format GIS boundary files is much more in the spirit of "close coupling." SpaceStat is able to scan the contents of the raster file in the specified format to create a weights matrix directly. This weights matrix is in the GAUSS matrix format (characterized by an .FMT file extension). The only requirement to execute the GIS interface is that the relevant binary files be present on the current directory. For the Generic Raster Format Binary Contiguity option you must also know the exact format in which the raster data are stored. Of course, this implies that you previously have digitized a base map and converted the vector format to a raster format, or acquired the digital information in another manner.

The conversion from a raster format to the spatial weights is carried out by a free standing executable program RAS2SWM.EXE. This program can be run by itself or is called from within SpaceStat in the Gis Interface menu.3

---

1. Examples of such AML programs are listed in Anselin, Hudak and Dodson (1992).

2. For a more extensive discussion of the linkage between GIS and spatial data analysis, see Anselin and Getis (1992).

3. For technical details on the RAS2SWM program, see Anselin, Hudak and Dodson (1992). The RAS2SWM program was written in C by Rustin F. Dodson, NCGIA, University of California, Santa Barbara.
The interface with ARC/INFO allows you to create both a general weights matrix (based on the length of the common boundary between spatial units) and a simple contiguity matrix. In the current version of SpaceStat, only the latter can be created from the raster GIS systems.

13.2 Simple Contiguity Matrix from ARC/INFO: D-2-1

The AML routines referred to in the introduction to this chapter create an output file in ascii format that contains the information on common boundary length between polygons. Since the AML routines are written for a unix workstation environment, you will have to convert the files from the unix to the DOS format. The routines outlined in Anselin, Hudak and Dodson (1992) include various options to deal with special cases, such as nodes with multiple arcs (high node valency). The output file of the routines consists of three columns, the first one for the polygon ID, the second for the neighbor’s polygon ID, and the third for the weight. The first item in the output file is the number of observations. These files can thus be manipulated in the same way as for the standard Data-Input-Sparse Ascii to Spatial Weights command (D-1-5). However, the D-2-1 command includes two additional features: polygon IDs that equal to zero are ignored, and the general weights are converted to simple contiguity (i.e., all nonzero weights are set equal to one).

After you invoke the ARC/INFO interface with the D-2-1 command sequence, you are prompted for the following items:
- the name of the ascii input file
- the file name for the new contiguity matrix
- the file name for the GAUSS data set that contains the polygon identifier
- the name of the polygon identifier

Also, after the file is read in and the matrix is constructed, you are asked if you wish to carry out a symmetry check. As in the other cases where you create a contiguity matrix from an ascii input file, this is highly recommended. When a lack of symmetry is detected, some indication is given of the possible cause of the problem, similar to what is presented in Chapter 10. After the final weights are created and saved, the screen is cleared and a brief message appears indicating the dimensions of the new matrix. You get back to the GIS Interface menu by pressing the Return key.

13.3 Boundary Length Weights Matrix from ARC/INFO: D-2-2

This second form of an interface with the ARC/INFO GIS operates in exactly the same manner as the previous one, with one exception. Since the goal is to create a general weights matrix, the weights are not converted to 0-1 values, but kept with their original magnitude. All inputs and outputs are the same as in section 13.2.
13.4 Simple Contiguity from IDRISI Raster Files: D–2–3

You start the interface with an IDRISI raster file by invoking the D–2–3 command sequence. After this, you are prompted for the name you wish to give to the new contiguity matrix. Next, you see a brief message indicating that the interface with IDRISI has started. This is followed by a query for the name of the IDRISI image file. You should not include any file extensions (such as .FMT for the weights matrix or .DOC and .IMG for the IDRISI image) in the file names.

After you enter this information, you are shown a brief summary of the structure of the image, followed by an account of the scanning process. At the end, the screen is cleared and a message is displayed which gives the file name for the new matrix and its dimensions. You get back to the GIS Interface menu by pressing the Return key.

When you run SpaceStat by means of some versions of the GAUSS Runtime module, the program is not able to locate the IDRISI file. This error is trapped by SpaceStat and you can recover by running the program from within DOS. You are given precise instructions on how to execute the commands. This is illustrated in more detail in section 13.8 below. This problem does not occur when you run SpaceStat as a program using the full implementation of GAUSS.

The IDRISI interface command creates two files. One is the weights matrix, in the GAUSS matrix format. The other is an ascii file, FULL____SCM that is a byproduct of the RAS2SWM program. Typically, you will not further need this file and could delete it. In fact, if you create weights matrices for several IDRISI images, the FULL____SCM file will be overwritten every time the program is executed.

13.5 Simple Contiguity from OSU-MAP Raster Files: D–2–4

The interface with the OSU-MAP raster GIS operates in the same manner as the one with IDRISI. You invoke it by means of the D–2–4 command sequence and are asked for the name of the new contiguity matrix. Next, you see a message indicating that the interface is established and are prompted by the name of the OSU-MAP file. The rest of the procedure is identical as for the IDRISI interface (the same problem occurs when the interface is run with the GAUSS Runtime module).

13.6 Simple Contiguity from Generic Raster Files: D–2–5

The third form of an interface with a raster GIS allows almost any raster file to be read by SpaceStat, provided that you are able to input all needed parameters. You invoke this interface by means of the D–2–5 command sequence. You are prompted for the name
of the new contiguity matrix. Next, you see a message indicating that the interface is established and are queried for the name of the generic raster file. This is followed by a few prompts to determine the format of the raster file:

- the number of rows in the raster image
- the number of columns in the raster image
- whether the file is ascii (type a) or binary (type b)
- for a binary file, the number of bytes per cell in the raster image (1, 2, or 4)
- the number of header bytes that should be skipped

This information allows the RAS2SWM program to scan the raster image and to pass the necessary information to SpaceStat for the creation of a contiguity matrix in the GAUSS format. After you enter these items, you are shown a brief summary of the structure of the image, followed by an account of the scanning process. At the end, the screen is cleared and a message is displayed which gives the file name for the new matrix and its dimensions. You get back to the GIS Interface menu by pressing the Return key.

13.7 Example

A rasterized vector boundary file for the Columbus neighborhoods, in the format used by IDRISI, is contained in the \COLUMBUS directory in the files COLUMB.DOC and COLUMB.IMG. You need to have both these files on your current directory. The image was constructed by rasterizing a vector boundary file in ARC/INFO and converting this to the IDRISI format. You should keep in mind that certain errors are created in this process, since the rasterization is only an approximation to the boundaries of the spatial units.

To start the interface with this image, type d (or D) from any menu in SpaceStat to reach the Data module. Next type 2 (or move the cursor down to the second line and press Return) to get to the Data GIS Interface menu. The IDRISI option is the third one: type 3 (or move the cursor down to the third line and press Return) to start the command.

The first prompt is for the name of the weights matrix to be created. Enter colwidri in response to the query, as shown:

```
Binary contiguity from Idrisi format raster file
Enter the filename for the output file: colwidri
```

Next, the screen is cleared and you see the following message:

```
Interfacing with Idrisi raster file ...
```

In fact, this message indicates that control has been passed to the RAS2SWM program. The next query for the name of the IDRISI image. Enter columb, as shown:

```
Enter the name of the Idrisi image (no extension): columb
```

The program now has all the information needed to carry out the conversion between the IDRISI format and the GAUSS matrix format used by SpaceStat. You are given a brief
CREATING A SPATIAL WEIGHTS MATRIX FROM A GIS

summary of the structure of the image, followed by an account of the progress of the scanning process, as illustrated below:

The raster has 300 rows, 300 columns, 1 bytes/cell (0=ascii)

FIRST PASS: Scanning the image for unique polygons ...

Rows remaining to be processed:
300
250
200
150
100
50

The raster has 49 polygons. Polygon ID's:
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
23 24 25 26 27 28 29 3
0 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49

SECOND PASS: Checking for adjacencies ...

Rows remaining to be processed:
300
250
200
150
100
50

At the end of this process, the screen is cleared and the following message displayed:
The weights matrix has been created and saved as
The matrix file colwidri
with 49 rows and 49 columns
Pressing the Return key gets you back to the Data GIS Interface menu.
You will note two files on your current directory: the file COLWIDRI.FMT and the file
FULL___SCM. Both are also included in the \COLUMBUS directory.

13.8 Common Problem

A problem encountered with the implementation of SpaceStat under the current version
of the GAUSS Runtime Module (but not when you run SpaceStat from the GAUSS System
itself) is that the program is not able to find the RAS2SWM.EXE file. If this happens to you,

---

4. This problem will be eliminated in future releases of SpaceStat, by avoiding this limitation of the GAUSS Runtime module.
you will hear a beep and the following error message will be displayed right after the Interfacing with Idrisi raster file ... message:

   ERROR: executable ras2swm file cannot be accessed

This error is trapped by SpaceStat and does not terminate the command. However, the queries for your input are slightly different in that you will in effect be running the RAS2SWM program from the DOS prompt, but with an easy return to SpaceStat. If you press the Return key after the error message, the screen will clear and you get the following instructions:

  Moving to DOS ...

Please type ras2swm i at the DOS prompt
and type exit when the program is completed
(this will return you back to SpaceStat)

Next you will see the familiar DOS System header and > prompt. At the prompt, enter the command:

  >ras2swm i
followed by Return.

It is very important that you include the command line option i (which refers to the IDRISI option of the RAS2SWM program). After this, you will get the usual prompts and messages as illustrated above. The program terminates by getting you back to the DOS prompt >. Now type the following command:

  >exit
followed by Return. This will get you back to SpaceStat and to the summary message about the completion of the weights matrix construction.

The same problem occurs for the OSU-MAP and generic raster interfaces as well. The respective commands that you should enter from the DOS prompt are:

  >ras2swm os
for OSU-MAP, and

  >ras2swm gs
for the generic format.

13.9 Exercise

If you have a digitized boundary file for your own data set in the format used by IDRISI, or OSU-MAP, or if you know the exact specifications of the format, you can now proceed to create a contiguity matrix from it. Alternatively, you can use the IDRISI image files EIRE.DOC and EIRE.IMG on the \EIRE directory to create a contiguity matrix for the Irish data.

You also may want to compare this contiguity matrix to the one created from the ascii file in Chapter 10. The easiest way to achieve this (except for using the file comparison features
of your operating system) is to create an ascii file that gives the contiguities between neighborhoods in sparse format. As mentioned before, you accomplish this by means of the Data - List - List Binary Contiguity in Sparse Form command sequence (D-8-7). For example, when comparing the resulting files for the weights COLW1 (or Appendix E) and COLWIDRI, you will note that there are a few differences between the two files, due to the approximation in the rasterizing process and to the additional judgement used when constructing the contiguity information in Appendix E.5

You are likely to reach similar conclusions about the Irish data or for your own data set. The implication of this "error propagation" is that the results of your spatial data analyses may differ, depending on how your weights matrix was constructed. You may want to carefully assess the extent to which this may affect your conclusions, by means of a sensitivity analysis.

References


Anselin, Luc, Sheri Hudak and Rustin F. Dodson (1992). Spatial data analysis and GIS: interfacing GIS and econometric software, National Center for Geographic Information and Analysis, University of California, Santa Barbara.

5. This contiguity information was not derived from Figure 4.1 directly, but from a much more detailed base map. Hence, there are some apparent contradictions between the neighborhood outlines in Figure 4.1 and what is contained in Appendix E.
CHAPTER 14

CHARACTERISTICS OF A SPATIAL WEIGHTS MATRIX

14.1 Introduction

The structure of potential interaction implied by the elements of a spatial weights matrix can be characterized by a few summary measures. The Weight Chars menu of the Tools module in SpaceStat includes five commands to compute various features associated with this structure. Two of these (Roots and Dominant Root) are based on the eigenvalues of the weights matrix, which are also very important for the maximum likelihood estimation of spatial regression models treated in Part V. Another command (Connectivity) yields a number of measures of connectivity. The last two commands in the Weight Chars menu compute sums of elements of the weights matrices and matrix traces that play an important role in the calculation of tests for spatial autocorrelation, treated in Part IV. These commands are invoked by entering the T-5 command sequence, followed by the number that corresponds to the respective function.

As soon as one of the characteristics is computed, it is included as an element of two column vectors that are appended to and stored with the weights matrix. The eigenvalues are stored in the second column, while the various summary measures of connectivity are stored in the first column. Since these values are stored with the weights matrix, they no longer need to be recomputed when the same matrix is used in tests or regression models. This considerably speeds up computation, without a loss of precision. The detailed format of the additional columns of the weights matrix is given in Appendix C.

14.2 Eigenvalues of the Weights Matrix: T-5-1 and T-5-3.

Two commands in the Weight Chars menu deal with the eigenvalues of the weights matrix. The first, Roots, yields the complete list of eigenvalues and highlights the largest and smallest value, $\lambda_{\text{max}}$ and $\lambda_{\text{min}}$. These eigenvalues are important in the computation of maximum likelihood estimates of spatial autoregressive models. The proper estimate for the autoregressive coefficient must be included in the interval $1/\lambda_{\text{min}}$ to $1/\lambda_{\text{max}}$, as outlined in more detail in Part V. For a row-standardized weights matrix (see Chapter 16) the largest eigenvalue is always +1 (see Ord, 1975, for a derivation), but the smallest eigenvalue is not -1 (see also Anselin and Hudak, 1992, for implementation details). For a symmetric and unstandardized weights matrix, the largest and smallest eigenvalues do not fall within a set range. For example, for the Columbus first order contiguity weights contained in the file COL_1, the largest eigenvalue is 5.91, while the smallest one is -3.10. This would constrain the estimates for the spatial autore-
gressive parameter to the range -0.32 to 0.17, which is much narrower than the usually presumed
range of -1 to +1.

The second command, Dominant Root, gives only the largest eigenvalue. This root
is often considered to be the best summary indicator of the structure of connectivity embodied
in the contiguity matrix (see Griffith, 1988, for an extensive discussion and additional references).
For a symmetric and binary (0-1) contiguity matrix, the largest root for a planar graph of the
same dimension is computed as well, following the suggestion in Boots and Royle (1991). This
planar equivalent is found as $0.5 \times [3 + \sqrt{(8N-15)}]$ , where N is the dimension of the con-
tiguity matrix. The ratio of the largest root to this planar equivalent is computed as well. This
ratio gives an indication of how the connectivity in the weights matrix approximates the structure
of the most connected planar graph of the same dimension. This index allows the structure
of matrices of different dimensions to be readily compared (for further details, see Boots and
Royle, 1991). For the Columbus neighborhoods, the largest root is 5.91 (as shown above) with
a planar equivalent of 11.21, and corresponding ratio of 0.53.

Both commands operate in the same fashion. After you enter the command sequence,
respectively T-5-1 for Roots and T-5-3 for Dominant Root, you are queried for the file name
of the weights matrix. After you enter this file name, the screen clears and the results of the
computations are listed. These results are also written to the file specified in the Output File
option. You get back to the Weights Chars menu by pressing the Return key.

14.3 Connectivity of a Weights Matrix: T-5-2

The second command in the Weights Chars menu computes eight or nine characteristics
of the connectivity structure embodied in a spatial weights matrix. These characteristics are:
- symmetry or asymmetry of matrix, i.e., whether or not it has been row-standardized
  (if not row-standardized, the matrix is assumed to be symmetric, row standardization
  is treated in Chapter 16)
- dimension of the matrix
- number of nonzero links, i.e., the number of elements in the matrix W that are not zero
- the percentage of nonzero weights as a percent of the N \times N-1 off-diagonal elements
  of the matrix
- the average weight: this will always be 1 for a symmetric binary contiguity matrix
- the average number of links: the average number of non-zero elements in a row of
  the weights matrix
- the most connected observations: the observation sequence numbers for those spatial
  units with the highest number of contiguous observations (i.e., the highest number of
  non-zero elements in the corresponding row of the weights matrix)
- the least connected observations: the observation sequence numbers for those spatial units of the ones with at least one contiguity that have the lowest number of contiguous observations (i.e., the lowest number of non-zero elements in the corresponding row of the weights matrix)
- unconnected observations: the observation sequence numbers of those spatial units that have no contiguous observations

The computation of the characteristics is invoked by the T-5-2 command sequence. As for the other commands in the Weights Chars menu, there is only a single prompt, for the file name of the weights matrix. After you enter this file name, the screen clears and the results of the computations are listed. These results are also written to the file specified in the Output File option. You get back to the Weights Chars menu by pressing the Return key.

14.4 Sums of Matrix Elements: T-5-4

The fourth command in the Weights Chars menu computes three sums of the elements of the weights matrices that are used in the various tests for spatial autocorrelation in Part IV and Part V. In the literature, these sums are referred to as $S_0$, $S_1$, and $S_2$ (for details, see Cliff and Ord, 1981, p. 21). They are defined as:

\[
S_0 = \sum_i \sum_j w_{ij} \\
S_1 = 0.5 \times \sum_i \sum_j (w_{ij} + w_{ji})^2 \\
S_2 = \sum_i (w_{ir} + w_{ri})^2
\]

with $w_{ij}$ as the element in the i-th row and j-th column of the weights matrix $W$, $w_{ir}$ as the row sum for the i-th row ($\sum_j w_{ij}$), and $w_{ri}$ as the column sum in the i-th column ($\sum_j w_{ji}$). For the first order contiguity matrix of the Columbus neighborhoods, the values of these sums are respectively 232 for $S_0$, 464 for $S_1$, and 5136 for $S_2$. For a row-standardized weights matrix, the value of $S_0$ will always equal the dimension of the matrix, N.

The computation of these sums is invoked by the T-5-4 command sequence. As for the other commands in the Weights Chars menu, there is only a single prompt, for the file name of the weights matrix. After you enter this file name, the screen clears and the results of the computations are listed. These results are also written to the file specified in the Output File option. You get back to the Weights Chars menu by pressing the Return key.

14.5 Traces of Matrix Products: T-5-5

The last command in the Weights Chars menu yields the results of the computation of matrix traces that are used in the calculation of the Lagrange Multiplier tests for spatial autocorrelation in regression models (see Part V). The respective expressions are:
- trace(WW)
- trace(W^2)
- trace(W'W + W^2)

For symmetric contiguity matrices, the first two results are the same. However, for row-standardized matrices, they are not the same, in contrast to what is sometimes asserted in the literature. For the Columbus first order contiguity matrix, the trace of both WW and W^2 equals 232. Note that this is identical to S_0. This identity does not hold for asymmetric row-standardized weights.

The computation of the matrix traces is invoked by the T-5-5 command sequence. As for the other commands in the Weights Chars menu, there is only a single prompt, for the file name of the weights matrix. After you enter this file name, the screen clears and the results of the computations are listed. These results are also written to the file specified in the Output File option. You get back to the Weights Chars menu by pressing the Return key.

14.6 Example

In this example, you will compute the connectivity characteristics of the first order contiguity matrix for the Columbus neighborhoods. This is the file COL_1 or COLW1 you created earlier. These files are also included on the \COLUMBUS directory.

Start by typing t (or T) from any menu to reach the Tools menu. Next, type 5 (for Weights Chars), followed by 2 for the Connectivity command. Enter col_1 or colw1 in response to the prompt for the matrix file name, as shown:

```
Connectivity of a spatial weights matrix
```

```
Enter the matrix filename (do not include .FMT),
or press Return for directory listing: col_1
```

Following this, the screen will clear and the following results are displayed:

```
Connectivity characteristics of weights matrix col_1
Weights matrix is symmetric
Dimension: 49
# nonzero links: 232
% nonzero weights: 9.86395
Average weight: 1.00000
Average # links: 4.73469
Most connected observation(s):
  17
Least connected observation(s):
  5  8  15  45  47  48  49
```
If you have set the **Output File** option to YES and specified a file name, the results will be written to that file as well. You get back to the **Weights Chars** menu by pressing the *Return* key.

### 14.7 Exercise

To further familiarize yourself with the typical results for the various weights characteristics, you can now also assess the connectivity structure and other measures for the various spatial weights matrices included in the \COLUMBUS, \EIRE or \AFRICA directories, or for the weights matrices you may have constructed for your own data.

### References


CREATING HIGHER ORDER CONTIGUITY MATRICES

CHAPTER 15

CREATING HIGHER ORDER CONTIGUITY MATRICES

15.1 Introduction

Higher order contiguity between spatial units is defined in a recursive fashion, as first order contiguity to units that are contiguous of the next lower order. For example, a spatial unit would be second order contiguous to a given observation if it is first order contiguous to another unit that itself is first order contiguous to the observation. To illustrate this concept, refer to Figure 10.1 with the Columbus neighborhoods. For the neighborhoods at the top of the figure, you see that neighborhood 7 would be considered second order contiguous to neighborhood 5, since it is first order contiguous to neighborhood 6, and not already first order contiguous. In contrast, neighborhood 1 would not be second order contiguous to 5 (even though it is first order contiguous to 6), since it is already first order contiguous.

Higher order contiguity is used to compute spatial correlograms for the tests against spatial autocorrelation covered in Part IV. The contiguity matrices are obtained by powering a first order contiguity matrix and correcting for circularity (i.e., a unit cannot be designated as contiguous for more than one order). In SpaceStat, this is accomplished by the second command in the Weight Trans menu of the Tools module.

A first order contiguity matrix must have been previously created and must be present as a file in the GAUSS matrix format (i.e., with file extension .FMT). Moreover, this matrix must be symmetric and only consist of 0 and 1 as elements. If the matrix you specify has earlier been row-standardized (see Chapter 16), it will no longer be 0-1. SpaceStat is able to trap this and generates an error message. However, if this matrix has not previously been manipulated in SpaceStat, there is no way to check whether or not the matrix is proper. This may result in the creation of nonsensical higher order contiguities for improper input matrices. Note that you can always check the contents of a matrix by means of the D-8-B (List Spatial Weights in Sparse Form) command sequence.

15.2 Creating Higher Order Contiguity Matrices: T-2-2

The construction of higher order contiguity matrices is invoked by the T-2-2 command sequence. Following this, you will be prompted for three items:

- the filename for the first order contiguity matrix (this matrix must be in the GAUSS matrix format)
- a file prefix of no more than 5 characters for the higher order weights matrices
- the highest order of contiguity
For small data sets, you should not specify too high an order of contiguity. In fact, for any size data set you will reach a point where the higher order contiguity will start resulting in unconnected spatial units (i.e., spatial units for which the corresponding row in the contiguity matrix consists of zeros only). This is inappropriate for the computation of many spatial statistics and will generally result in an error or warning message. If the higher order contiguity matrix would result in a zero matrix (e.g., for a high order of contiguity and a small data set), no matrix will actually be saved.

After you enter this information, the screen will clear and a series of messages will appear, indicating that the matrices are being powered and that a check for circularity is carried out. At the end of this operation, a short note is listed to confirm that the matrices have been saved. The new matrices get a filename of the form prefix_i, where prefix is the name specified in response to the second prompt and i is the order of contiguity.

You get back to the Weight Trans menu by pressing the Return key.

15.3 Example

Your point of departure will be the first order contiguity matrix COLW1 created earlier. This matrix is also present as file COLW1.FMT in the \COLUMBUS directory. If you have not used this matrix for any other operations (e.g., row-standardization), it should be in the proper format and contain only 0 and 1.

To start, type t (or T) from any menu to get to the Tools module, followed by 2 for Weight Trans and another 2 for the Higher Order Contiguity command. Next, enter the filename colw1 and the prefix colw in response to the prompts, as shown:

Higher order contiguity

File name for the first order contiguity matrix

Enter the matrix filename (do not include .FMT),
or press Return for directory listing: colw1

Enter the file name prefix for the higher order contiguity weights
max 5 characters (do NOT include the .FMT extension)
File name is: colw

This is followed by a query for the highest order of contiguity. Enter 3 (or any other number you want to experiment with, but the example will be for 3), as shown:

Enter the highest order of contiguity: 3

Next, the screen is cleared and a number of messages appear, keeping you informed of the progress in the computations and alerting you to the presence of zero rows (or zero matrices). For the Columbus example, the output looks as follows:
Powering the contiguity matrices ...

Order 2 saved as colw_2
Order 3 saved as colw_3

Checking for circularity ...
Contiguity matrix for contiguity order 3 is OK

New contiguity matrices have been saved

If you press Return after this, you get back to the Tools Weight Trans menu. You may check your current directory (F2 to get to DOS and dir), to make sure that the files COLW_2.FMT and COLW_3.FMT are present. These files are also included on the \COLUMBUS directory. Note that the size of COLW_2 and COLW_3 (20016K) is not the same as for COLW1 (19232K), even though all three matrices pertain to the same data set, since information has been appended on the matrix dimension, symmetry and possible presence of rows consisting of zeroes. You can assess the characteristics of the higher order contiguity weights by means of the Tools-Weight Char-Roots (T-5-1) and Tools-Weight Char-Connectivity (T-5-2) command sequences covered in the previous chapter. For example, for the second order contiguity, the number of nonzero links increases to 410, relative to 232 for the first order contiguity. Its largest root also increases, to 10.51 (compared to 5.91 for first order contiguity).

15.4 Exercise

You now can also create higher order contiguity matrices from the first order contiguity matrix you set up earlier, either for the African or Irish data, or for your own data set, in the same manner as outlined above. If you want to get a more precise idea of how these operations alter the connectivity structure of the spatial units, you can carry out the various commands of the Weight Char menu, or list the contents of the matrix by means of one of the List commands in the Data module.
ROW-STANDARDIZING A SPATIAL WEIGHTS MATRIX

16.1 Introduction

For the computation of many spatial statistics, and particularly for the estimation of spatial regression models, the spatial weights matrix should be row-standardized to yield a meaningful interpretation of the results. The row standardization consists of dividing each element in a row by the corresponding row sum. Each element in the new matrix thus becomes:

\[ \frac{w_{ij}}{\sum_j w_{ij}} \]

Although SpaceStat computes statistics and estimates models correctly for both unstandardized and standardized weights, row-standardizing the matrix is generally considered to be a good practice.

You achieve this by means of the Row Standardization command in the Weight Trans menu of the Tools module (T-2-1). Clearly, in order to row-standardize a weights matrix, it must have been previously created and stored in the GAUSS matrix format (with file extension .FMT). Once you carry out the row-standardization, SpaceStat will keep track of this characteristic of the weights matrix.

Of course, you may always convert a row-standardized weights matrix directly from an ascii file into the GAUSS matrix format, by means of one of the Data Input commands. For example, this would be useful if you computed the standardization in another program. A drawback of this approach is that SpaceStat cannot assess that the matrix is row-standardized. This may result in misleading messages (e.g., referring to the matrix as not being row-standardized) but should not affect the computations. If you want to get the correct messages at all times, you should explicitly run the T-2-1 row-standardization command sequence for each weights matrix used in your analyses.

16.2 Row-Standardizing a Spatial Weights Matrix: T-2-1

The row-standardization is invoked by the T-2-1 command sequence. You are prompted for two file names:
- the file name for the existing weights matrix (should be stored in the GAUSS format, with a .FMT file extension)
- the file name for the row-standardized weights matrix

The two file names do not have to be different. However, when they are the same, the original unstandardized weights matrix will be replaced by the row-standardized one. Consequently, the original file will be lost.
After entering this information, a brief message is listed to indicate that the row-standardization has been carried out. You get back to the **Weight Trans** menu by pressing the *Return* key.

### 16.3 Example

In order to compute a correlogram for Moran's I measure of spatial autocorrelation, using the Columbus data (in Part IV), you will need to row-standardize the three contiguity matrices created earlier. For this, you will need the files COLW1.FMT, COLW_2.FMT and COLW_3.FMT. If you did not construct these files as part of the previous examples, you may find them in the \COLUMBUS directory.

Type `t` (or `T`) to get to the **Tools** module, followed by 2 for the **Weight Trans** menu and 1 for the **Row Standardization** command. Next, enter the two filenames, respectively `colw1` for the existing matrix and `colws_1` for the row-standardized one (each followed by `Return`), as shown:

```
Row standardization of a spatial weights matrix

Enter the matrix filename (do not include .FMT),
or press Return for directory listing: colw1

File for row standardized weights
Enter the matrix filename (do NOT include .FMT)
File name is: colws_1

The following message announces the completion of the row-standardization:
Weights matrix colw1 is row-standardized as colws_1
Pressing the Return key gets you back to the **Tools Weight Trans** menu.
```

You will notice the new file COLWS_1.FMT on your current directory (use F2 to get to DOS, followed by `dir`; type `exit` to get back to **SpaceStat**). Since the row-standardization characteristic has been added to the file as an additional item, the file size is now 20016K (compare to the 19232K of COLW1.FMT).

You can proceed in an identical manner to create row-standardized weights for COLW_2 and COLW_3. In the following sections, these will be referred to as COLWS_2 and COLWS_3. The three row-standardized weights matrices for the Columbus data are also included in the \COLUMBUS directory.

The characteristics of the contiguity matrices change substantially after row-standardization. You can check this by means of any of the commands in the **Weight Chars** menu of the **Tools** module. For example, the average weight of the matrix COLWS_1 is now 0.21 (compared to the 1.0 for the simple contiguity) and its largest and smallest root are respectively
+1.0 and -0.65. This implies that the estimate for an autoregressive coefficient in a model based on this spatial weights matrix will have to be contained within the interval from -1.54 to +1.0.

16.4 Exercise

You can now also create row-standardized matrices that correspond to the other spatial weights matrices that you constructed earlier and assess the effect on the structure of connectivity by means of the commands in the **Weight Chars** menu.
CHAPTER 17

TRANSFORMING THE ELEMENTS OF A SPATIAL WEIGHTS MATRIX

17.1 Introduction

In addition to row-standardization and the creation of higher order contiguity matrices, outlined in the previous two chapters, the Weight Trans menu of the Tools module contains a few other functions to transform the elements of a spatial weights matrix. Two of these commands allow you to change the format of the spatial weights matrix, either from general weights to binary contiguity, or from a spatial weights format (using the structure listed in Appendix C) to a standard GAUSS matrix format. The other commands perform element by element matrix operations on the weights matrix. These operations differ from the functions included in the Mat Algebra menu of the Data module in that they explicitly take into account the special format used for weights matrices in SpaceStat.

17.2 Changing the Format of Spatial Weights Matrices

17.2.1 From General Weights to Contiguity Weights: \texttt{T-2-3}

This function converts the nonzero elements of a general spatial weights matrix to 1 and saves the new matrix in the spatial weights format used by SpaceStat. The new matrix is not row-standardized, even if the original general weights matrix was, and the row-standardization flag is set accordingly. The function is the third item on the Weight Trans menu, invoked by the command sequence \texttt{T-2-3}. You are prompted for the name of the existing general weights matrix and for the name of the new matrix. This file name may be the same as for the existing matrix, but the latter will then be overwritten. After these two prompts, the screen is cleared and a brief message appears, indicating that the new weights matrix has been created and saved. You get back to the Weight Trans menu by pressing the Return key.

17.2.2 From Spatial Weights Format to Standard Matrix Format: \texttt{T-2-4}

This function strips the first two columns with weights matrix characteristics from the spatial weights file and saves the resulting square matrix as a GAUSS matrix file. The elements of the weights matrix are unaltered. This operation is useful if you don't wish to include the two columns with weights characteristics in a matrix listing (Data-List-List Matrix, \texttt{D-8-6}) or in any of the matrix operations in the Mat Algebra menu of the Data module. The function is the fourth item on the Weight Trans menu, invoked by the command sequence \texttt{T-2-4}. You are prompted for the name of the existing general weights matrix and for the name of the
new matrix. This file name may be the same as for the existing matrix, but the latter will then be overwritten. After these two prompts, the screen is cleared and a brief message appears, indicating that the new weights matrix has been created and saved. You get back to the Weight Trans menu by pressing the Return key.

17.3 Element by Element Matrix Operations on the Spatial Weights Matrix

The Weight Trans menu of the Tools module contains three commands to transform the elements of an existing spatial weights matrix:

- element by element power: $T-2-5$
  $$w_{ij}^p$$

- element by element inverse: $T-2-6$
  $$1/w_{ij}$$

- boundary shares over distance weights: $T-2-7$
  $$w_{ij}^p/d_{ij}^p$$

The first function raises every nonzero element $wij$ of a weights matrix to a specified integer power $p$. It is invoked by the command sequence $T-2-5$. You are prompted for the name of the existing weights matrix, the value for the integer power (entering 1 keeps the matrix as is), and for the file name for the new matrix.

The second function takes the inverse of every nonzero element $wij$ of a weights matrix. This may thus be carried out for weights matrices that contain zero as well as nonzero weights. The operation is invoked by the command sequence $T-2-6$. You are prompted for the name of the existing weights matrix, and for the file name for the new matrix.

The third function divides every element of an existing weights matrix, $wij$, assumed to represent boundary shares, by the corresponding element in a distance matrix, $dij$. Both the elements of the boundary shares and the distance matrix may be raised to an integer power as well. However, in contrast with the previous command, no zero elements may be present in the distance matrix (the boundary shares matrix may have zero elements). If this is the case, an error message is generated and no transformation will be carried out. This function is invoked by the command sequence $T-2-7$. You are prompted for the name of the existing boundary shares weights matrix and the value for the integer power to which it should be raised (entering 1 keeps the matrix as is). Next, you are queried for the same information with respect to the distance matrix (filename and integer power). Finally, you are asked to specify the file name for the new matrix.

All three functions operate in the same fashion. If you specify the same filename for the new weights as that of any of the input matrices, the latter will be overwritten. After you specify the necessary input information, the screen is cleared and a brief message appears,
indicating that the new weights were created and saved. You get back to the **Weight Trans** menu by pressing the *Return* key.

### 17.4 Example

To illustrate the weight transformations, you will create a new weights matrix, consisting of the squared inverse distances between Columbus neighborhoods, but only for contiguous neighborhoods. You can accomplish this by means of the **Boundary Shares over Distance Weights** command (T-2-7), if you specify the unstandardized first order contiguity matrix as the first matrix and the distance matrix as the second. The distance matrix will be raised to the second power, but the contiguity matrix may be kept as is. In order to carry this out, you will need the first order contiguity matrix COL_1 and the inter-neighborhood distance matrix COLDIS. Both of these are included on the \COLUMBUS directory.

You start the example by typing `t` (or `T`) from any menu to reach the **Tools** module. Next, type 2 (for **Weights Trans**), followed by 7 for the **Boundary Shares over Distance Weights** command. Enter `col_1` and `coldis` in response to the prompts for the matrix file names, 1 for the power of the first and 2 for the power of the second matrix, as shown:

```
Boundary shares over distance weights

File name for boundary shares
Enter the matrix filename (do not include .FMT),
or press Return for directory listing: col_1
Power for boundary shares (1 for no power): 1

File name for distance weights
Enter the matrix filename (do not include .FMT),
or press Return for directory listing: coldis
Power for boundary shares (1 for no power): 2
```

Next, type `coldis2` (or any other name you choose) as the name for the new file:

```
File for new weights matrix
Enter the matrix filename (do NOT include .FMT),
File name is: coldis2
```

After you enter this information, the screen will clear and the following message will appear:

```
New weights matrix saved as coldis2
Pressing the Return key gets you back to the **Weight Trans** menu.
```
If you now wish to see the contents of the newly created weights, you can invoke the
Data-List-List Spatial Weights in Sparse Form (D-8-8) command sequence. After you enter
coldis2 in response to the prompt for the file name, the contents of the new weights matrix
will be listed. If you have the Output File option set YES, these contents will also be written
to the specified file. Table 17.1 shows the first few lines of this output for the COLDIS2 file.
Note that the squared inverse distance is only computed for the neighborhoods that are first
order contiguous (e.g., as shown in Table 10.1).

17.5 Exercise

You can now continue to experiment with the various weight transformation commands
and apply them to the other spatial weights matrices that you constructed earlier.
18.1 Introduction

The first menu of the Tools module, Space Trans, contains a number of commands that allow you to create variables that incorporate spatial effects, either in the form of a spatial average, or based on a spatial process. In addition, both a spatial autoregressive and a spatial moving average filter are included. The particular form of spatial dependence that is used in the respective transformations and filters is based on the specification of a spatial process model, treated in more detail in Part V.

The Space Trans menu also includes a command that allows you to resample non-contiguous spatial units from a dataset, based on the information contained in a spatial weights matrix.

18.2 Spatial Averages

18.2.1 Spatial Lag: $T^{-1}1$

A spatially lagged variable or spatial lag is a weighted average of the values in locations neighboring each observation. More precisely, if an observation on a variable $x$ at location $i$ is represented by $x_i$, then its spatial lag is $\Sigma w_{ij}x_j$, i.e., the sum of the product of each observation in the data set with its corresponding weight from the $i$-th row of the spatial weights matrix. In matrix notation, the spatial lags for all observations on the variable $x$ may be expressed as the matrix product $Wx$ (with $W$ as the $N$ by $N$ weights matrix). The weights matrix may be unstandardized or row-standardized, although the latter is preferred, since it implies a form of spatial smoothing. When a non-standardized matrix is used, SpaceStat generates a warning message, although the computations are carried out as usual.

In SpaceStat, the computation of a spatial lag is invoked by the $T^{-1}1$ command sequence. You are prompted for the name of the data set that contains the variables to be spatially lagged, and for the file name of the spatial weights matrix. After this, you are given a list of the variables in the selected data set. Next, you will be asked to enter the name for the variables to be lagged, and to specify a variable name for the spatial lag. If you press Return, a default prefix of $W_-$ will be added to the name of the existing variable. This query continues until you press Return in response to the name of the existing variable. The same spatial weights matrix is used in the transformations for all the variables. The screen clears and a brief message appears, indicating that the new variables have been added to the data.
set, and summarizing its contents (number of observations, number of variables, and a list of variable names). You get back to the **Space Trans** menu by pressing the *Return* key.

For the Columbus data set listed in Appendix D, and using the row-standardized first order contiguity matrix, based on the information in Appendix E, the neighbors of the first observation are 2, 5 and 6, taking on values for CRIME of respectively 32.388, 15.726 and 30.627. A spatial lag for crime would thus be $78.741/3 = 26.247$, or, one third of the sum of the values observed in the locations neighboring observation 1.

### 18.2.2 Window Average: T–1–2

By convention, the diagonal elements of the spatial weights matrix $W$ are set to zero. As a result, the value for each observation itself is not included in the computation of a spatial lag. This may be achieved by means of the **Window Average** command in the **Space Trans** menu. The value for each observation, $x_i$, is incorporated in the computation of a spatially smoothed variable as:

\[
(x_i + \sum w_{ij}x_j) / (1 + \sum w_{ij})
\]

Such an average is only meaningful if the weights matrix is used in unstandardized form (in contrast to the recommended use for the computation of the spatial lag). In essence, a "window" is centered on each location, consisting of the location itself and those observations included in the corresponding row of the spatial weights matrix. All elements in the window receive an equal weight in the computation of the weighted average.

You invoke the **Window Average** function by means of the *T–1–2* command sequence. You are prompted for the name of the data set that contains the variables to be averaged, and for the file name of the spatial weights matrix. This matrix should be in unstandardized form. If a row-standardized matrix is used, the computation will proceed, but a warning message will be generated. In this instance, the resulting window average will equal $(x_i + \sum w_{ij}x_j)/2$.

After entering the weights matrix, you are given a list of the variables in the selected data set. Next, you will be asked to enter the name for the variables to be averaged, and to specify a variable name for the window average. If you press *Return*, a default prefix of *W_* will be added to the name of the existing variable. This query continues until you press *Return* in response to the name of the existing variable. The same spatial weights matrix is used in the transformations for all the variables. The screen clears and a brief message appears, indicating that the new variables have been added to the data set, and summarizing its contents (number of observations, number of variables, and a list of variable names). You get back to the **Space Trans** menu by pressing the *Return* key.
For the CRIME variable in the Columbus data set, a window average for the first observation would consist of the average of observations 1, 2, 5 and 6, i.e., \((18.802 + 78.741)/4\), or 24.38575.

18.3 Spatial Filters

18.3.1 Spatial Autoregressive Filter: T–1–3

A spatial autoregressive filter may be used to eliminate spatial dependence in a variable. The spatial dependence is assumed to follow a spatial autoregressive process. In matrix notation, the filter is:

\[ I - \rho W \]

with I as an N by N identity matrix, \( \rho \) as the spatial autoregressive coefficient and W as the N by N spatial weights matrix. The spatial autoregressive coefficient must be specified by you. This may be any acceptable value, but should ideally be based on the results of the estimation of a spatial process model by means of the techniques outlined in Part V.

A variable that is processed through a spatial autoregressive filter is transformed in the following manner:

\[(I - \rho W)x = x - \rho Wx\]

The filtered variable is thus the difference between the original observations and the scaled spatial lag (with the autoregressive coefficient as the scaling factor).

A Spatial Autoregressive Filter is invoked by the T–1–3 command sequence. You are asked to specify the data set with the variables to be filtered. Next, you are prompted for the file name of the spatial weights matrix, and for the value to be used for the coefficient in the spatial autoregressive filter.

After entering this information, you are given a list of the variables in the selected data set. Next, you will be asked to enter the name for the variables to be filtered, and to specify a variable name for the new variable. If you press Return, a default prefix of W_ will be added to the name of the existing variable. This query continues until you press Return in response to the name of the existing variable. The same spatial weights matrix is used in the spatial filters for all the variables. The screen clears and a brief message appears, indicating that the new variables have been added to the data set, and summarizing its contents (number of observations, number of variables, and a list of variable names). You get back to the Space Trans menu by pressing the Return key.
18.3.2 Spatial Moving Average Filter: T-1-4

A spatial moving average filter may be used to eliminate spatial dependence in a variable. The spatial dependence is assumed to follow a spatial moving average process. In matrix notation, the filter is:

\[(I + \rho W)^{-1}\]

i.e., a matrix inverse, with I as an N by N identity matrix, \(\rho\) as the spatial moving average coefficient, and W as the N by N spatial weights matrix. The spatial moving average coefficient must be specified by you. This may be any acceptable value, but it cannot induce singularity in the filter. If this is the case, an error message will be generated and no filter will be carried out. A variable that is processed through a spatial moving filter is transformed in the following manner:

\[(I + \rho W)^{-1}x\]

The filtered variable is thus equal to the product of the vector of the original observations and a matrix inverse.

A Spatial Moving Average Filter is invoked by the T-1-4 command sequence. You are asked to specify the data set with the variables to be filtered. Next, you are prompted for the file name of the spatial weights matrix, and for the value to be used for the coefficient in the spatial moving average filter.

After entering this information, you are given a list of the variables in the selected data set. Next, you will be asked to enter the name for the variables to be filtered, and to specify a variable name for the new variable. If you press Return, a default prefix of W_ will be added to the name of the existing variable. This query continues until you press Return in response to the name of the existing variable. The same spatial weights matrix is used in the spatial filters for all the variables. The screen clears and a brief message appears, indicating that the new variables have been added to the data set, and summarizing its contents (number of observations, number of variables, and a list of variable names). You get back to the Space Trans menu by pressing the Return key.

18.4 Creating Spatially Dependent Variables

18.4.1 Spatial Autoregressive Transformation: T-1-5

By means of the Spatial Autoregressive Transform command in the Space Trans menu, you may construct variables that show a preset degree of spatial dependence, in accordance
with a spatial autoregressive process. In this manner, you may create simulated data sets with various degrees of embedded spatial dependence. The spatial autoregressive transformation is of the form:

$$(I - \rho W)^{-1}$$

i.e., a matrix inverse, with $I$ as an $N \times N$ identity matrix, $\rho$ as the spatial autoregressive coefficient, and $W$ as the $N \times N$ spatial weights matrix. The spatial autoregressive coefficient must be specified by you. This may be any acceptable value, but should ideally be based on the results of the estimation of a spatial process model by means of the techniques outlined in Part V. If the coefficient you specify yields a singular transformation matrix, SpaceStat will generate an error message and no computations will be carried out.

A variable that is processed through a spatial autoregressive transformation is transformed in the following manner:

$$(I - \rho W)^{-1}x$$

The transformed variable is thus equal to the product of the vector of the original observations and a matrix inverse.

A Spatial Autoregressive Transform is invoked by the T-1-5 command sequence. You are asked to specify the data set with the variables to be transformed. Next, you are prompted for the file name of the spatial weights matrix, and for the value to be used for the coefficient in the spatial autoregressive transformation.

After entering this information, you are given a list of the variables in the selected data set. Next, you will be asked to enter the name for the variables to be transformed, and to specify a variable name for the new variable. If you press Return, a default prefix of $W_-$ will be added to the name of the existing variable. This query continues until you press Return in response to the name of the existing variable. The same spatial weights matrix is used in the spatial transformations for all the variables. The screen clears and a brief message appears, indicating that the new variables have been added to the data set, and summarizing its contents (number of observations, number of variables, and a list of variable names). You get back to the Space Trans menu by pressing the Return key.

18.4.2 Spatial Moving Average Transformation: T-1-6

By means of the Spatial Moving Average Transform command in the Space Trans menu, you may construct variables that show a preset degree of spatial dependence, in accordance with a spatial moving average process. In this manner, you may create simulated data
sets with various degrees of embedded spatial dependence. The spatial moving average transformation is of the form:

\[ I + \rho W \]

with \( I \) as an \( N \times N \) identity matrix, \( \rho \) as the spatial moving average coefficient, and \( W \) as the \( N \times N \) spatial weights matrix. The spatial moving average coefficient must be specified by you. This may be any acceptable value.

A variable that is processed through a spatial moving average transformation is transformed in the following manner:

\[(I + \rho W)x = x + \rho Wx\]

The transformed variable is thus the sum of the original observations and the scaled spatial lag (with the moving average coefficient as the scaling factor).

A **Spatial Moving Average Transform** is invoked by the \textit{T-1-6} command sequence. You are asked to specify the data set with the variables to be transformed. Next, you are prompted for the file name of the spatial weights matrix, and for the value to be used for the coefficient in the spatial moving average transformation.

After entering this information, you are given a list of the variables in the selected data set. Next, you will be asked to enter the name for the variables to be transformed, and to specify a variable name for the new variable. If you press \textit{Return}, a default prefix of \textit{W} will be added to the name of the existing variable. This query continues until you press \textit{Return} in response to the name of the existing variable. The same spatial weights matrix is used in the spatial transformations for all the variables. The screen clears and a brief message appears, indicating that the new variables have been added to the data set, and summarizing its contents (number of observations, number of variables, and a list of variable names). You get back to the **Space Trans** menu by pressing the \textit{Return} key.

### 18.5 Non-Contiguous Random Sample: **T-1-7**

**SpaceStat** does not include facilities to carry out traditional forms of spatial sampling. Nevertheless, the last command in the **Space Trans** menu allows you to create a subsample from a dataset, such that the observations selected are noncontiguous. The contiguity criterion is based on the information in a spatial weights matrix. This matrix may be row-standardized or unstandardized, since only the existence or absence of a contiguity is taken into account.

The resampling is useful when you wish to use traditional (non-spatial) statistical techniques to estimate spatial regression models and avoid the complexities caused by simultaneous
spatial dependence (see Part V). For example, if you wish to estimate a mixed regressive spatial autoregressive model based on this resampled set of observations, the standard ordinary least squares (OLS) estimator may be used. In contrast, when all observations are included, OLS is not an appropriate estimator, as pointed out in the discussion in Part V. Such re-coding or re-sampling of observations is an easy way to implement the so-called conditional spatial process models, although it is by no means the only approach to achieve this. Clearly, the resampling procedure will result in a loss of information, and it should only be implemented when a large number of observations is available.

The non-contiguous sample is created by means of a process of random sampling without replacement. First, all observations that have no contiguities (i.e., that are unconnected) are eliminated from the list of potential choices. In most applications, a spatial lag will be included in the models that are implemented with the new dataset, so that an unconnected observation would be useless (the spatial lag would be zero). After this first step, observations are selected at random, and their neighbors are eliminated from the list of potential choices. The random selection is determined by the random number generator contained in GAUSS. The seed for this generator is set by the Random Number Seed option. If you don’t want to always create the same subset, you should clear the default seed used by SpaceStat, as outlined in Chapter 2. The sampling process is continued until the list of observations is exhausted. This procedure does not yield a random sample in the strict sense. Nevertheless, it provides a meaningful subset of the observations that is unlikely to suffer from problems of spatial dependence, while allowing spatial dependence to be modeled (in the form of a spatial lag).

The observation sequence numbers of the selected observations are written to an ascii file. The first item in this file is the variable name OBS. This file may consequently be used to select a subset from a data set by means of the Data-Merge/Select-Select Observations from Data Set (D-3-5) command sequence. However, in order for this to work, a variable with the name OBS must be contained in the target data set. If this is not the case, such a variable must first be created by means of the Data-Var Create-Create Observation Numbers (D-4-6) command sequence. Alternatively, you may edit the ascii file and replace OBS by an indicator variable that is contained in the data set.

You invoke the Non-Contiguous Random Sample by means of the T-1-7 command sequence. First, you are prompted for the file name of the spatial weights matrix. Next, you are asked to specify a file name for the ascii file to which the observation numbers will be written. If the file specified already exists, the selected observation numbers will be appended. Note that a file with multiple sets of observations will NOT work as an input to the Merge/Select commands. After this, the screen is cleared and a brief message is given, showing how
many observations were selected and listing their sequence numbers. You get back to the **Space Trans** menu by pressing the *Return* key.

### 18.6 Example

As an illustration of the various spatial transformations, you will now create a new variable for the Columbus data set that has embedded spatial dependence, and next take out the spatial dependence by means of a spatial filter. In order to carry this out, you should use the COLWORK data set created before. If you don’t have this data set, make a copy of the COL.DAT and COL.DHT files that are contained in the \COLUMBUS directory. You will also need the row-standardized contiguity matrix COLWS_1, created as part of the example in Chapter 16. This file is also included in the \COLUMBUS directory.

You start the example by typing `t` (or `T`) from any menu to get into the **Tools** module, followed by `1` for the **Space Trans** menu and `5` for the **Spatial Autoregressive Transform**. Next, enter `colwork` for the data set, `colws_1` for the spatial weights matrix, and `0.5` for the autoregressive coefficient, as shown:

```plaintext
Spatial autoregressive transform

Enter the data set filename (do not include .DAT or .DHT),
or press Return for directory listing: colwork

Spatial weights file

Enter the matrix filename (do not include .FMT),
or press Return for directory listing: colws_1

Enter the coefficient for the spatial AR transformation: ? 0.5
```

After this, you are given a list of the variables in the data set, and queried to specify variable names for the existing and the new variables, as shown:

```plaintext
Choose the variable(s) from the following list
(each variable name should correspond exactly to one in the list):

   NEIG    CRIME    INCOME    HOUSING    X    Y    EW

Enter the name for each existing variable first, or press Return to stop
Then enter the name for the new variable, or press Return for the default
Default is W_ prefix

Existing variable: crime       New Variable (Return for default): ar_crime
Existing variable:

Press *Return* to end the queries. The screen is cleared and the following message appears:

The new variables have been added to the dataset
The dataset colwork
with 49 observations on 8 variables:

NEIG  CRIME  INCOME  HOUSING  X   Y   EW  AR_CRIME

Press the Return key to get back to the Space Trans menu.

Next, type 3 from the Space Trans menu to invoke the Spatial Autoregressive Filter command. Enter the same information as for the transformation in response to the queries, with the exception of the variable names. Now, take ar_crime as the existing variable and sf_crime as the new variable. After the command is carried out, you may compare the variables crime, ar_crime and sf_crime by means of the Data-List-List Selected Observations (D-8-4) command sequence. For the first observation in the data set, the value for CRIME should be 18.802, for AR_CRIME should be 47.545776, and for SF_CRIME again 18.802.

Finally, you may try the non-contiguous sampling command for the columbus data as well. Type 7 from the Space Trans menu, and respond colws_1 for the spatial weights file and colran.asc (or any other name you choose) for the ascii file, as shown:

Noncontiguous random sample

Enter the matrix filename (do not include .FMT),
or press Return for directory listing: colws_1

Enter the name of the Ascii output file: colran.asc

After this, the screen will clear and the following message will appear:

16 randomly selected non-contiguous observations
based on weights matrix colws_1
written to the Ascii file colran.asc

selected observations are:

2  5  8  12  16  19  22  24  26  32  38  40  43  45  48  49

Note that if you did not use the default random number seed of SpaceStat, the sequence numbers of the selected observations may differ. Press Return to get back to the Space Trans menu. Now you may check the contents of the COLRAN.ASC file by means of the DOS type command. Press the F2 key to get to the DOS prompt and enter type colran.asc. The file contents will be:

OBS

2  5  8  12  16  19  22  24  26  32  38  40  43  45  48  49

You get back to SpaceStat by typing exit at the DOS prompt.

If you wish to use the COLRAN.ASC file to select a subset of the COLWORK data set, by means of the D-3-5 command sequence (Data-Merge/Select-Select Observations from Data Set), you must first create a variable OBS for that data set (unless it is already present from one of the earlier examples). You may also use this file to select a submatrix from the
contiguity matrix COLWS_1, in the usual fashion. If you check the contents of the new matrix (e.g., by means of a Data-List-List Matrix, d-l-l command sequence), you will see that it consists of only zeroes, as intended (no contiguities).

18.7 Exercise

You can now continue to experiment with the various spatial transformations for other variables in the Columbus, Irish or African data sets, or in other data sets you may have created. You can also create non-contiguous subsets based on the contiguity matrices for these data sets.
PART IV

EXPLORATORY SPATIAL DATA ANALYSIS
CHAPTER 19

PROBLEM FILES IN THE EXPLORE MODULE

19.1 Introduction

All analyses in the Explore module (and also in the Regress module) need a Problem File as input. This file contains information on the data set, spatial weights or other matrices and variables used in the statistical analysis. A Problem File conforms to a specific format, which is given in Appendix B. You may create a Problem File by means of a text editor, as long as the resulting file conforms to the format in Appendix B. Alternatively, you may also create a Problem File interactively, either in the process of carrying out a particular analysis, or for later use.

The query for the Problem File, for its interactive creation during the analysis, or for its interactive creation for later use follows the choice of the type of statistical analysis in a submenu of the Explore module. Each such analysis is chosen by a three item command sequence, beginning with E (for the Explore module), and consisting of the sequence number for the menu item and the command. After you choose a specific command, say E-1-1, for Explore-Describe-Descriptive Stats, the screen clears and you are given a list of three options:  

1 Batch
2 Interactive
3 Make Problem File

You choose the appropriate option by typing its sequence number, or by moving the cursor to the sequence number and pressing Return. For the first and third options, you must specify the name of the Problem File. This file must exist for option 1 and will be created for option 3, overwriting any existing file with the same name. For the second option, you may specify the name of a Problem File. This file will be saved upon completion of its interactive specification for later use in other analysis. In this option, you can only create one Problem File, while in the third option, you may create as many as you wish. If you don’t want the problem information saved to a Problem File when using the Interactive option, you press Return at the prompt for the file name.

The same Problem File can be used for all statistical analyses in SpaceStat, with one exception: for the QAP-Matrix and Partial Matrix Association command (E-6-1) only GAUSS matrices are allowed as inputs. All other commands require the data set name, spatial weights, and variable names in the same sequence.

This flexibility allows you to create a Problem File in one submenu of SpaceStat and to subsequently use it for a wide range of other commands. There are only some minor mod-
ifications needed when the Problem File that was originally created in a command of the Explore module must be used for the spatial regression specifications in the Regress module. If such changes are not made to the Problem File, the default of a generic regression is assumed in the Regress module (see Part V for further details on regression specifications). However, a Problem File created in any of the commands of the Regress module can be used unchanged for all commands in the Explore module.

19.2 Problem File Structure in the Explore Module

19.2.1 Analyses that Use Variable Names

As mentioned in the introduction to this chapter, variable names must be specified in the Problem File for all but one command in the Explore module. In addition, the name of the GAUSS data set that contains the variables must be given as well. These are the only two mandatory items for the non-spatial analyses in the Describe menu of the Explore module. All other analyses also need a file name for at least one spatial weights matrix.

Optionally, the file name for a Report File may be specified. Such a file is not used in all commands, but only for a few that generate an index or value for each observation in the data set (E-1-3, E-1-5, E-5-3, and E-5-4). This information is also written to the screen and to the standard output file when the Long Output option is set to YES. The distinctive characteristic of a Report File is that headers and all other extraneous information are eliminated, resulting in a format that is more suitable for import into mapping or GIS software packages (e.g., to visualize the results).

For the join count statistics (E-2) and the QAP measures of spatial association (E-6-2, E-6-3 and E-6-4), a structural change indicator variable may be specified as well (see Chapters 21 and 24 for further details).

When the Interactive and Make Problem File options of the specification of the Problem File are invoked in the Explore module, the sequence of queries is as follows:

- Data set name: required
- Report file: optional
- Spatial weights: required for all spatial analyses
- Variables: required
- Structural change indicator: only in E-2, E-6-2, E-6-3 and E-6-4, optional

After you enter this information in the Interactive option, the screen clears and you see the messages:

```
Reading in data ...
Starting analysis ...
```
This indicates that the Problem File was specified correctly and that the analysis proceeds properly.

In the Make Problem File option, the sequence of prompts is concluded by a query for another problem specification. If you wish to add another problem to the Problem File, you must type \textit{Y} (or \textit{y}), otherwise the current Problem File is saved with the file name you specified.

19.2.2 \textit{Matrix and Partial Matrix Association}

In the Matrix and Partial Matrix Association command (\textit{E-6-1}), only the file names of the matrices must be specified. The Problem File may not contain a data set name or variable names. As a result, a Problem File created for this command cannot be used as input to any of the other commands.

When the Interactive and Make Problem File options of the specification of the Problem File are invoked for this command, there is only one prompt:

- Matrices for matrix comparison: required

After you enter the file names of two or three (for partial matrix comparison) GAUSS matrices, the program proceeds in the same manner as outlined in 19.2.1.

19.3 Errors

In SpaceStat, there are two types of error checks on the items contained in a Problem File, or entered in response to the queries of the Interactive and Make Problem File options. The first type of check happens during the interactive entry of items. When you fail to enter a required item in response to a prompt, e.g. a file name for the data set, you will get an error message. Depending on the type of error, you will get a second chance or be returned to the menu from where you entered the command. Note that there is no check on the correctness of the items entered. For example, the mistaken entry of a variable name that is not included in the data set is not trapped in the current version of SpaceStat.

The second type of error check happens when the Problem File is interpreted and the various items are read from their respective files. A check is made for all required items. When they are not included in the Problem File, an error message is given (e.g., \textit{spatial weights must be specified for this analysis}). SpaceStat also makes sure that files with the names given for the data set and the spatial weights are actually present on the current directory. If they are not, an error message is generated. SpaceStat proceeds to the next problem contained in the Problem File, when several problems are included, or returns you to the menu from which you entered the command. In the Explore module, only those variables are analyzed whose names match the variable names for the specified data set. No error message is generated.
when some variable names do not match, only when not a single match can be found. The
analysis is carried out for the proper variables.

A final type of error may occur when the file you specified as Problem File does not
follow the format specified in Appendix B. Depending on the circumstance, this may be
trapped, and an error message is given (e.g., attempt to read beyond end of problem file).
You are subsequently put back in the menu from which you entered the command. However,
in most instances, this will result in a program crash.

19.4 Example

To illustrate these concepts, an example follows for the construction of a Problem File
to be used for the computation of Moran’s I in Chapter 22. In order to accomplish this, you
invoke the Explore module (type e or E from any menu), type 3 (or move the cursor down
to the third line and press the Return key) to get to the Moran menu, followed by any of
the command numbers (since you will not actually carry out the analysis at this point, it does
not matter which one you enter). This gets you to the menu with as options the Batch, Interactive
or Make Problem File commands. You should choose option 3 and enter coll.btc
(followed by Return) in response to the prompt for the name of the problem file, as shown:

    3 MAKE PROBLEM FILE Enter problem file name (for saving): coll.btc

Your screen will clear and a series of queries appear. The first two pertain to the file
names for the data set and for the report file. Enter col for the former (followed by a Return).
The report file is used so that specific results can be written to it, in order to facilitate importing
these results into other software packages at a later time (e.g., spreadsheet or graphics packages).
You will not need it for this example, but since you will be using the same problem file for
a number of analyses (COL1.BTC), I suggest you enter the file name rep at this point (followed
by Return), as shown:

    INTERACTIVE PROBLEM FILE CREATION

    Problem no.  1

    Data set for problem

    Enter the data set filename (do not include .DAT or .DHT),
or press Return for directory listing: col

    Report file for problem

    Name for report file (no extension) or Return for no file: rep
The next screen deals with the specification of the spatial weights. A list is given of all matrices in the GAUSS matrix format that are on the current directory. Note that some of these may not be proper spatial weights, but SpaceStat cannot determine that at this point. You specify the weights to be used in this problem by entering their filenames at the prompt, or pressing Return when no more need to be included. When no spatial weights are needed in the analysis (this is the case for only a few commands in SpaceStat), you can press Return at the first prompt. If you carried out all previous examples for the Columbus data (and only those), the items listed will correspond to those given below. Enter the file names colws_1, colws_2 and colws_3 in response to the prompts, as shown:

```
Spatial weights for problem

Enter the filenames for the spatial weights from the following list or press Return to stop

Files in default directory:
    COLW1   COLDIS   COLD_1   COLD_2   COLD_3   COLW_2   COLW_3
    COLWS_1  COLWS_2  COLWS_3

Weight file: colws_1
Weight file: colws_2
Weight file: colws_3
Weight file: Return
```

The next screen deals with the variables for the problem. Similar to the format used for the weights, you see a list of the variables that are contained in the data set you specified earlier. You enter the variable names in response to the prompt, or press Return to stop. If you will be using the same problem file later to carry out regression analyses, you should always enter the variable name for the future dependent variable as the first one (in the Explore module, the order of the variables does not matter). In your example, this means that you should enter crime, income, and housing before typing Return in response to the Variable name prompt, as shown:

```
Variables for problem

Choose the variable(s) from the following list (each variable name should correspond exactly to one in the list):
    NEIG      CRIME     INCOME    HOUSING          X          Y        EW

Enter the variable name, or press Return to stop
Variable name: crime
Variable name: income
Variable name: housing
```
Variable name: Return

Make another problem?

Enter Y to continue or press Return to stop

Press the Return key to end the creation of the Problem File. You will be back in the Explore Moran menu.

If you check the contents of the current directory (use the F2 key to get to the DOS prompt), you will notice the file COL1.BTC. This file is also included on the \EXAMPLES directory. Its contents are as shown in Table 19.1. The first value, 1, indicates that only one problem was created. The next line shows 15 flags for various items used by SpaceStat. In this example, the first 0 indicates that the Problem File was created for a command in the Explore module; the 1 that follows shows the choice of a Report file; the 3 stands for three weights matrices; the next 1 for a constant term (ignored in the Explore module, but needed in the commands of the Regress module); the next 1 for one dependent variable (CRIME, but its nature as dependent variable is ignored in the Explore module); and the 2 stands for two explanatory variables. On the next line, you find several file and variable names. First comes the data set (COL), next the file name for the report file (REP), next the three weights files (COLWS_1, COLWS_2, COLWS_3), followed by the three variables (CRIME, INCOME, HOUSING).

<table>
<thead>
<tr>
<th>Table 19.1 Problem file COL1.BTC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>0 1 3 1 1 0 2 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>COL REP COLWS_1 COLWS_2 COLWS_3 CRIME INCOME HOUSING</td>
</tr>
</tbody>
</table>

19.5 Exercise

You can now create Problem Files for analyses using the Irish or African data sets, or other data sets you may have created. In the remainder of Part IV, I will assume that you know how to create a Problem File for the Explore module and will not repeat the various prompts for each command.
CHAPTER 20

BASIC DESCRIPTIVE STATISTICS

20.1 Introduction

The **Describe** menu of the **Explore** module contains a number of basic non-spatial descriptive statistics, as well as multivariate measures of spatial association. The range of statistics included is limited by design, since most commercial statistical packages are well equipped in this respect, and I saw no need to replicate these facilities. For the non-spatial descriptive statistics (**E-1-1**, **E-1-2**, and **E-1-3**), only a dataset and variable names must be specified in the **Problem File**. For the two other commands, at least one spatial weights matrix must be given as well. All commands are invoked by entering their command sequence, followed by the name of the **Problem File** in batch mode (option 1), or by entering the requested information in interactive mode (option 2).

20.2 Methodology

20.2.1 Descriptive Statistics: **E-1-1**

The **Descriptive Stats** command yields four types of results for each variable specified in the **Problem File**. A brief outline of the methodological basis is given next.

**Moments**

Five moments of the distribution are computed for each variable, as follows:

- mean: \( \mu = (1/N) \sum x_i \)
- variance: \( \sigma^2 = (1/N) \sum (x_i - \mu)^2 \)
- standard deviation: \( \sigma = \sqrt{\sigma^2} \)
- skewness: \( b_1 = (1/N) \sum (x_i - \mu)^3 / (\sigma^2)^{3/2} \)
- kurtosis: \( b_2 = (1/N) \sum (x_i - \mu)^4 / (\sigma^2)^2 \)

where \( x_i \) is the i-th observation of the variable \( x \), and \( N \) is the number of observations. Note that the variance (and standard deviation) is computed as the consistent estimate and not as the unbiased estimate. The latter can be easily obtained by re-scaling the result given by **SpaceStat** by a factor \( N/N-1 \). When a variable has a variance of zero (i.e., it is a constant), a warning is given and the variance and standard deviation are set to 1.

**Range**

The minimum and maximum value observed for each variable are given, as well as the range (maximum less minimum).
Quartiles
The values corresponding to the first, second (median) and third quartile are listed, and the interquartile range is computed (third quartile less first quartile). A comparison of mean and median, and of range with interquartile range provides good insight into the symmetry (or lack of symmetry) of the distribution of each variable.

Test for normality
A test for normality is computed for each variable, based on the skewness and kurtosis. This is an asymptotic Wald test, distributed as $\chi^2$ with 2 degrees of freedom. Although many tests exist to assess normality, this particular one is included here since it is based on the same principles as the one used in the analysis of regression residuals in the Regress module. Its formal expression is:

$$W = N\frac{b_1^2}{6} + \left(b_2 - 3\right)^2/24$$

where $b_1$ and $b_2$ are the moments given above.

20.2.2 Correlation: E-1–2
A matrix of Pearson product moment correlation coefficients is computed, with each coefficient as:

$$r = \frac{1}{N}\sum(x_i - \mu_x)(y_i - \mu_y)/\sigma_x \cdot \sigma_y$$

where $\mu_x$, $\mu_y$, are the means of $x$ and $y$, and $\sigma_x$ and $\sigma_y$ are the standard deviations. When a variable has a variance of zero (e.g., it is a constant), its standard deviation is set to 1 in subsequent computations, and a warning is given.

20.2.3 Principal Components: E-1–3
The matrix of correlation coefficients for a number of variables can be used to compute an associated set of principal components. The k-th principal component is found as $X\gamma_k$, where $X$ is a $N$ by $K$ matrix of observations on the variables and $\gamma_k$ is a $K$ by 1 eigenvector corresponding to the k-th eigenvalue $\lambda_k$ of the correlation coefficient matrix. With each eigenvalue is associated a "variance proportion," $\lambda_k/\Sigma\lambda_k$, which is a measure of the fraction of overall covariation between the variables that is "explained" by the corresponding principal component.
20.2.4 \textit{Multivariate Spatial Correlation: E-1-4}

The multivariate measure of spatial correlation computed in \textbf{SpaceStat} follows the approach suggested by Wartenberg (1985). First, all variables are standardized:

\[ z_k = (x_k - \mu_k) / \sigma_k \]

where the subscript \( k \) refers to the vector of observations on the \( k \)-th variable, \( \mu_k \) is the mean for variable \( k \), and \( \sigma_k \) is its standard deviation. Also, the spatial weights matrix is converted to a stochastic matrix, i.e., a matrix for which all elements sum to one. The resulting matrix \( W' \) is always symmetric, with as elements

\[ w'_{ij} = w_{ij} / \sum_i \sum_j w_{ij} \]

where \( w_{ij} \) are the elements in the unstandardized weights matrix.

A matrix of coefficients of spatial association is constructed as:

\[ M = Z W' Z \]

where \( Z \) is a matrix with the values for the standardized variables as columns. The association represented in this matrix is similar in form to a bivariate Moran coefficient between variables \( k \) and \( l \) (see also Chapter 24):

\[ m_{kl} = z_k' W' z_l \]

The diagonal elements differ from the traditional Moran’s I spatial autocorrelation coefficients (see Chapter 22) by a scale factor:

\[ m_{kk} = z_k' W' z_k \]

The interpretation of the multivariate measures of spatial association is not without difficulty. They should be considered for exploratory purposes only, since a traditional significance test on all coefficients would create horrendous problems of multiple comparisons (i.e., the significance levels would have to be very small in order to obtain meaningful results). For further details, see Wartenberg (1985).
20.2.5  Spatial Principal Components: E–1–5

The matrix of multivariate spatial correlation coefficients can be used to derive principal components in the same way as outlined in 20.2.3. Wartenberg (1985) suggests the use of these spatial principal components (as well as the eigenvalues and associated eigenvectors) to map patterns of spatial association. One problem often encountered in practice is that the matrix of multivariate spatial association is not necessarily positive definite. As a consequence, some of its eigenvalues may be negative. Such negative eigenvalues cannot be given a meaningful interpretation in the context of computing principal components. If this is the case, SpaceStat will report the results for the Spatial Principal Components command with a warning.

20.3  Example Problem File

To illustrate the various descriptive statistics, you can construct the Problem File COL2.BTC listed in Table 20.1. This Problem File is also included on the \EXAMPLES directory. You give the data set as col, the report file as rep, spatial weights as colw1 (or any other weights matrix for the Columbus data set), and variables as crime, income and housing. For details on the creation of a Problem File, see Chapter 19.

Table 20.1 Problem file COL2.BTC

<p>| | | | | | | | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

   COL   REP   COLW1  CRIME  INCOME  HOUSING

20.4  Program Output

20.4.1  Descriptive Statistics

The results for the moments, range, quartiles and test for normality are listed in Table 20.2. A cursory glance at these results reveals non-normality for the variables INCOME and HOUSING, whose distribution is rather skewed (skewness > 0, mean > median, range >> inter-quartile range/2). In contrast, the variable CRIME conforms well to the normal distribution. This will be important in the interpretation of the tests for spatial autocorrelation covered in Chapters 22 and 23.
### 20.4.2 Correlation

The results for the correlation matrix are given in Table 20.3. *Spaestat* lists these results row by row, together with the name for the variable that corresponds to the row. For example, the row for CRIME in Table 20.3 indicates a negative correlation of -0.70 between CRIME and INCOME and a negative correlation of -0.57 between CRIME and HOUSING.

#### Table 20.2 Descriptive Statistics

<table>
<thead>
<tr>
<th>DATA SET: COL</th>
<th>MOMENTS</th>
<th>VARIOUS</th>
<th>ST.DEV.</th>
<th>SKEWNESS</th>
<th>KURTOSIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>VARIABLE</td>
<td>MEAN</td>
<td>VARIANCE</td>
<td>ST.DEV.</td>
<td>SKEWNESS</td>
<td>KURTOSIS</td>
</tr>
<tr>
<td>CRIME</td>
<td>35.12884</td>
<td>274.2476</td>
<td>16.56042</td>
<td>0.03422164</td>
<td>2.225951</td>
</tr>
<tr>
<td>INCOME</td>
<td>14.37494</td>
<td>31.86467</td>
<td>5.64488</td>
<td>0.9260998</td>
<td>3.771052</td>
</tr>
<tr>
<td>HOUSING</td>
<td>38.43622</td>
<td>334.0366</td>
<td>18.27667</td>
<td>1.337858</td>
<td>4.312181</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DATA SET: COL</th>
<th>RANGE</th>
<th>VARIABLE</th>
<th>MIN</th>
<th>MAX</th>
<th>RANGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRIME</td>
<td>0.178</td>
<td>68.892</td>
<td>68.714</td>
<td></td>
<td></td>
</tr>
<tr>
<td>INCOME</td>
<td>4.477</td>
<td>31.07</td>
<td>26.593</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HOUSING</td>
<td>17.9</td>
<td>96.4</td>
<td>78.5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DATA SET: COL</th>
<th>QUARTILES</th>
<th>VARIABLE</th>
<th>Q1</th>
<th>MEDIAN</th>
<th>Q3</th>
<th>Q RANGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRIME</td>
<td>19.146</td>
<td>34.001</td>
<td>50.732</td>
<td>31.586</td>
<td></td>
<td></td>
</tr>
<tr>
<td>INCOME</td>
<td>9.918</td>
<td>13.38</td>
<td>18.477</td>
<td>8.559</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HOUSING</td>
<td>23.6</td>
<td>33.5</td>
<td>44.333</td>
<td>20.733</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DATA SET: COL</th>
<th>WALD TEST FOR NORMALITY</th>
<th>VARIABLE</th>
<th>TEST</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRIME</td>
<td>1.232834</td>
<td>0.53987541</td>
<td></td>
<td></td>
</tr>
<tr>
<td>INCOME</td>
<td>8.218044</td>
<td>0.01642383</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HOUSING</td>
<td>18.13261</td>
<td>0.00011549</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
20.4.3 Principal Components

The **Spacestat** output for the **Principal Components** command is listed in Table 20.4. The standard output consists of two parts. First, as shown at the top of the table, the eigenvalues of the correlation matrix and associated variance proportion are listed. In the example, the first eigenvalue accounts for roughly 73% of the total variance. The second part of the standard output consists of the eigenvectors associated with each eigenvalue. The elements of these eigenvectors are given in the order of the variables with which they correspond. For example, the eigenvector for the first eigenvalue yields a coefficient of -0.61 for CRIME, 0.58 for INCOME and 0.54 for HOUSING. These coefficients are used in the computation of the principal components for the three variables.

When you set the **Long Output** option to YES, the full set of values for the principal components are listed on the screen and written to the **Report File**. The first three lines of this file are illustrated in Table 20.5. Note that the first column in this file gives the values for the variable NEIG, which was used as **Indicator Variable** in the options. If this option is set to NO, a variable OBS will be used instead (note that the variable OBS is not necessarily part of the data set, but is created only for the purpose of this listing). The principal components are given the variable names PCOMP_i, where i is the sequence number, similar to the convention used in the **Data** module (**Data-Var Algebra-Principal Components**).

### Table 20.3 Correlation

<table>
<thead>
<tr>
<th></th>
<th>CRIME</th>
<th>INCOME</th>
<th>HOUSING</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRIME</td>
<td>1.000000</td>
<td>-0.695589</td>
<td>-0.574487</td>
</tr>
<tr>
<td>INCOME</td>
<td>-0.695589</td>
<td>1.000000</td>
<td>0.499879</td>
</tr>
<tr>
<td>HOUSING</td>
<td>-0.574487</td>
<td>0.499879</td>
<td>1.000000</td>
</tr>
</tbody>
</table>
20.4.4 Multivariate Spatial Correlation

The SpaceStat output for the Multivariate Spatial Correlation command is listed in Table 20.6, using the first order contiguity weights COLW1. The same format is used as for the Pearson product moment correlation matrix. The diagonal elements indicate that CRIME has the strongest pattern of spatial autocorrelation (0.52 vs. 0.41 for INCOME and 0.22 for HOUSING). The elements of the first row confirm the pattern of negative association found earlier, but now between CRIME and a spatial lag of both INCOME (-0.44) and HOUSING (-0.24).
20.4.5 Spatial Principal Components

The spatial principal components are listed in the same fashion as their non-spatial counterparts. This is illustrated in Table 20.7. Now, the first eigenvalue accounts for more than 90% of the total variance. The coefficients of the associated eigenvector are -0.69 for CRIME, 0.62 for INCOME and 0.37 for HOUSING.

In the same way as for the non-spatial principal components, a Report File is created when you set the Long Output option to YES. The full set of values for the principal components are listed on the screen and written to this file, using the same conventions as in 20.4.3.

Table 20.6 Multivariate Spatial Correlation

<table>
<thead>
<tr>
<th>DATA SET:</th>
<th>COL</th>
<th>WEIGHTS:</th>
<th>COLW1</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRIME</td>
<td>0.520636</td>
<td>-0.443865</td>
<td>-0.241421</td>
</tr>
<tr>
<td>INCOME</td>
<td>-0.443865</td>
<td>0.413448</td>
<td>0.226092</td>
</tr>
<tr>
<td>HOUSING</td>
<td>-0.241421</td>
<td>0.226092</td>
<td>0.219772</td>
</tr>
</tbody>
</table>

20.5 IDRISI Interface

When you set the Idrisi Interface option to YES in the computation of both the traditional and the spatial principal components, additional output files are created. One file is created for each principal component, with a file name of PCOMP_i.VAL, where i is the sequence number of the principal component. These files are in a format suitable for display by the IDRISI GIS. The first column in each file consists of the observation number, or the matching value of the Indicator Variable if this option is set to YES. The second column consists of the values for the principal component. No headers or other information are included in the VAL file.

Note that this file is created irrespective of the setting of the Long Output option. For example, if you have the latter set to NO, you will not see the values of the principal components on the screen, nor will they be written to the Report File. However, the .VAL files will be created.

When you specify more than one weights matrix in the computation of the spatial principal components, the .VAL files are generated for the last weights matrix only. Since the file
name does not reflect the weights used in the computations, you should avoid specifying more than one weights matrix when you want to create an interface with IDRISI.

20.6 Exercise

It should be fairly straightforward for you to experiment with descriptive statistics for other variables in the Columbus data set or in the other data sets furnished.

References

21.1 Introduction

The simplest measures of spatial autocorrelation are the join count statistics for binary variables. Such variables take on only the values of 1 and 0. Areal units with observations 1 are often referred to as colored Black, while units with observations of 0 are referred to as colored White. Join counts are counts of the number of times a join, i.e., a contiguity, corresponds to a similar or dissimilar value in the neighboring units. There are three such join counts: BB (Black-Black), or the number of times a Black unit is contiguous to another Black unit; BW (Black-White), or the number of times a Black unit is bordered by a White unit; and WW (White-White), or the number of times that both neighbors are White units. Since the total number of possible joins equals one half the sum of all nonzero elements of the weights matrix (i.e., \( \sum \sum w_{ij} / 2 \)), only two of the three join counts need to be calculated. The third can be obtained as the difference between one half \( S_0 \) and the sum of the other two. The computation of join counts is carried out in SpaceStat with the commands in the Join Count menu of the Explore module:

- BB Join Counts: E-2-1
- BW Join Counts: E-2-2
- WW Join Counts: E-2-3

These commands are invoked by entering the appropriate command sequence, followed by the name of the Problem File in batch mode (option 1), or by entering the requested information in interactive mode (option 2). A Problem File for join count computation must contain the name of a dataset, at least one spatial weights matrix, and the name of at least one binary variable. Optionally, a categorical indicator variable for structural change may be included as well. When this is the case, partial measures of spatial association are computed. These are treated in more detail in Chapter 24.

21.2 Methodology

Join counts are only meaningful for symmetric unstandardized contiguity matrices. In SpaceStat, this is always ensured, even if you enter a row-standardized weights matrix in the Problem File. The latter is converted to a symmetric binary matrix by setting all nonzero weights equal to 1. With \( x_i = 1 \) for Black and \( x_i = 0 \) for White, and \( w_{ij} \) as the i-jth element in the spatial weights matrix \( W \), the join count statistics are:

\[
BB = (1/2) \sum \sum w_{ij} x_i x_j
\]

In SpaceStat, the join count statistics are computed as a special case of the QAP measures of association. Technical details about the inference and output of QAP statistics are discussed in Chapter 24. There are two bases for inference, one using the normal approximation, the other using a permutation approach. In the permutation approach, a number of randomly resampled data sets are created for which the join count statistic is computed. The value obtained for the actual data set is then compared to the empirical distribution for the resampled data sets. The number of resampled data sets to be used in the permutation approach is determined by the Number of Permutations option. The default is 99, which is sufficient for a quick and dirty analysis, but is insufficient if precise insight is needed.

21.3 Example Problem File

To illustrate the computation of join counts, you can construct the Problem File COL3.BTC listed in Table 21.1. This Problem File is also included on the \EXAMPLES directory. You give the data set as col, the spatial weights as colws_1 (or any other weights matrix for the Columbus data set), and the variable as hicrime. For details on the creation of a Problem File, see Chapter 19.

In order to carry out the join count tests, you must first construct the variable hicrime as a dummy variable that takes on a value of 1 for crime rates higher than 35 (the mean). You accomplish this by means of the Data-Var Create-Create Dummy Variables (Range) command (D-4-4). You enter col as the name of the data set, crime for the existing variable, hicrime for the new variable, 35 for the lower bound and 80 for the upper bound. The spatial pattern of the HICRIME variable in the Columbus neighborhoods is illustrated in Appendix F (dark neighborhoods for HICRIME=1).

The \EXAMPLES directory also contains the Problem File COL3A.BTC. This file is identical to COL3.BTC, except that the variable EW is specified as a structural change indicator. This is needed for the computation of measures of partial spatial association, illustrated in Chapter 24 for the general case.

\[ BW = (1/2) \sum \sum w_{ij} (x_i, x_j)^2 \]
\[ WW = (1/2) \sum \sum w_{ij} (1 - x_i) (1 - x_j) \]


<table>
<thead>
<tr>
<th>Table 21.1 Problem file COL3.BTC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 0 0 1 1 1 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>COL COLWS_1 HICRIME</td>
</tr>
</tbody>
</table>
21.4  Program Output

The output for the BB join count computation for the variable HICRIME is listed in Table 21.2. This output follows the standard format for QAP (see Chapter 24). The value listed under GAMMA is the join count statistic, CORR1 and CORR2 are pseudo correlation coefficients, E[GAMMA] is the expected value under the null hypothesis (using either the normal approximation or the random permutations). For the normal approximation, a standardized z-variate (Z-GAMMA) and its associated probability (PROB) are reported. In contrast, for the permutation approach, no z-value is computed, since the E[GAMMA] and SD[GAMMA] are derived from the empirical distribution of \( \Gamma \) statistics for the resampled data sets. The probability measure (PROB) in the permutation case gives the fraction of resampled \( \Gamma \) values that are equal to or larger than the observed \( \Gamma \) (including the latter), in the usual fashion (see Chapter 24 for details). This should not be interpreted as a probability in the traditional sense, but is a so-called pseudo-significance level.

The BB coefficient for HICRIME is 54, which confirms the visual impression of spatial clustering of high crime values (HICRIME=1), found in Appendix F. This strong indication of clustering is according to both the normal approximation and the permutation approach. Note how the empirical mean obtained for the resampled data sets (27.35) is very similar to the theoretical mean under the normal assumption (27.22).

When you set the Long Output option to YES, a listing of \( \Gamma \) coefficients computed for all resampled data sets is produced as well. The values in this listing will vary with the random number seed that is set in the options. In Table 21.3, the values for 99 resampled data sets are reported, using the default random seed.

The results for the BW and WW join counts for HICRIME are listed in Table 21.4 and Table 21.5. The significant BB association of neighboring high crime areas is supported by a highly significant negative BW statistic. The BW value of 28 is significantly below its expected value of 59.18 (normal) or 59.21 (permutation). This indicates a less than usual (i.e., under the null hypothesis of no association) number of BW joins, which is consistent with the higher than usual number of BB joins found above. The value of 34 found for the WW joins is not significant (it is only slightly above the expected value of 29.59 for the normal and 29.43 for the permutation approach). Again, this supports the strong indication of spatial clustering for HICRIME=1.

Finally, note that the sum of the three counts, 54+28+34 equals 116, which is one half the number of nonzero weights (232) in the first order contiguity matrix for Columbus neighborhoods (see the example in Section 14.6).
### Table 21.2 BB Join Counts for HICRIME

**BB JOIN COUNTS FOR BINARY VARIABLES**

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>GAMMA</th>
<th>CORR1</th>
<th>CORR2</th>
<th>E[GAMMA]</th>
<th>Z-GAMMA</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>HICRIME</td>
<td>54</td>
<td>0.180172</td>
<td>0.301794</td>
<td>27.2245</td>
<td>6.330708</td>
<td>0.000000</td>
</tr>
</tbody>
</table>

Using the normal approximation

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>GAMMA</th>
<th>CORR1</th>
<th>CORR2</th>
<th>E[GAMMA]</th>
<th>Z-GAMMA</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>HICRIME</td>
<td>54</td>
<td>0.180172</td>
<td>0.301794</td>
<td>27.3535</td>
<td>4.30014</td>
<td>0.010000</td>
</tr>
</tbody>
</table>

Using 99 random permutations

### Table 21.3 Resampled BB Join Counts for HICRIME

**LIST OF GAMMA COEFFICIENTS FOR 99 RANDOM PERMUTATIONS**

| 52.000000 | 62.000000 | 58.000000 | 60.000000 | 56.000000 | 58.000000 |
| 34.000000 | 54.000000 | 66.000000 | 56.000000 | 74.000000 | 56.000000 |
| 64.000000 | 44.000000 | 36.000000 | 62.000000 | 50.000000 | 64.000000 |
| 64.000000 | 36.000000 | 56.000000 | 58.000000 | 52.000000 | 70.000000 |
| 52.000000 | 44.000000 | 56.000000 | 76.000000 | 50.000000 | 54.000000 |
| 50.000000 | 48.000000 | 56.000000 | 48.000000 | 54.000000 | 48.000000 |
| 52.000000 | 50.000000 | 54.000000 | 42.000000 | 50.000000 | 58.000000 |
| 52.000000 | 38.000000 | 48.000000 | 54.000000 | 48.000000 | 58.000000 |
| 52.000000 | 52.000000 | 46.000000 | 54.000000 | 58.000000 | 62.000000 |
| 44.000000 | 54.000000 | 48.000000 | 48.000000 | 60.000000 | 58.000000 |
| 52.000000 | 70.000000 | 48.000000 | 46.000000 | 58.000000 | 64.000000 |
| 58.000000 | 46.000000 | 40.000000 | 58.000000 | 58.000000 | 50.000000 |
| 56.000000 | 42.000000 | 58.000000 | 62.000000 | 48.000000 | 60.000000 |
| 52.000000 | 70.000000 | 66.000000 | 50.000000 | 56.000000 | 58.000000 |
| 72.000000 | 62.000000 | 60.000000 | 58.000000 | 78.000000 | 58.000000 |
| 60.000000 | 40.000000 | 54.000000 | 48.000000 | 52.000000 | 48.000000 |
| 46.000000 | 58.000000 | 68.000000 |
21.5 Exercise

Since the binary variable HICRIME was constructed rather arbitrarily, you may now compute join count statistics for a different dummy variable based on CRIME, or for other variables in the Columbus data set, after first converting them to a dummy variable. Of course, you can easily do the same for variables in the other data sets furnished. After you work through Chapter 24, you may want to return to this chapter and repeat the computation of join counts, but now using the Problem File COL3A.BTC, to analyze partial association as well.

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>GAMMA</th>
<th>CORR1</th>
<th>CORR2</th>
<th>E[GAMMA]</th>
<th>Z-GAMMA</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>HICRIME</td>
<td>28</td>
<td>-0.177896</td>
<td>0.106134</td>
<td>59.1837</td>
<td>-6.088366</td>
<td>0.000000</td>
</tr>
</tbody>
</table>

Using the normal approximation

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>GAMMA</th>
<th>CORR1</th>
<th>CORR2</th>
<th>E[GAMMA]</th>
<th>Z-GAMMA</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>HICRIME</td>
<td>34</td>
<td>0.028838</td>
<td>0.182259</td>
<td>29.5918</td>
<td>1.014095</td>
<td>0.155269</td>
</tr>
</tbody>
</table>

Using 99 random permutations

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>GAMMA</th>
<th>CORR1</th>
<th>CORR2</th>
<th>E[GAMMA]</th>
<th>Z-GAMMA</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>HICRIME</td>
<td>34</td>
<td>0.028838</td>
<td>0.182259</td>
<td>29.4343</td>
<td>4.22633</td>
<td>0.170000</td>
</tr>
</tbody>
</table>

References

MORAN’S I AND GEARY’S C TESTS FOR SPATIAL AUTOCORRELATION

22.1 Introduction

Moran’s I (Moran, 1948) and Geary’s c (Geary, 1954) are probably the best known measures to test for spatial autocorrelation. They are fairly simple to compute, but not always so straightforward to interpret. In SpaceStat, Moran’s I is computed by means of the commands in the Moran menu of the Explore module (E-3), while Geary’s c is computed by means of the commands in the Geary menu (E-4). Both menus follow the same structure, allowing for three different methods to carry out inference (normal, randomization, permutation), and including two different ways to present the results, either organized by weights matrix (with all variables listed for the same weights matrix), or organized by variable, in the form of a spatial correlogram.

The commands are invoked by entering the appropriate command sequence, followed by the name of the Problem File in batch mode (option 1), or by entering the requested information in interactive mode (option 2). A Problem File for the computation of Moran’s I or Geary’s c must contain the name of a dataset, at least one spatial weights matrix, and the name of at least one variable. The weights matrix may be either unstandardized or row-standardized.

22.2 Methodology

22.2.1 Moran’s I

Formally, Moran’s I is:

\[ I = \frac{N}{S_0} \sum_i \sum_j w_{ij} (x_i - \mu) (x_j - \mu) / \Sigma_i (x_i - \mu)^2 \]

where \( N \) is the number of observations, \( w_{ij} \) is the element in the spatial weights matrix corresponding to the observation pair \( i, j \), \( x_i \) and \( x_j \) are observations for locations \( i \) and \( j \) (with mean \( \mu \)), and \( S_0 \) is a scaling constant,

\[ S_0 = \Sigma_i \Sigma_j w_{ij} \]

i.e., the sum of all weights. For a row-standardized spatial weights matrix, which is the preferred way to implement this test, the normalizing factor \( S_0 \) equals \( N \) (since each row sums to 1), and the statistic simplifies to a ratio of a spatial cross product to a variance:
Moran’s I is similar but not equivalent to a correlation coefficient and is not centered around 0. In fact, the theoretical mean of Moran’s I is -1/N-1. In other words, the expected value is negative and is only a function of the sample size (N). Note however, that this mean will tend to zero as the sample size increases. The theoretical variance of Moran’s I depends on the stochastic assumptions that are made (see 22.2.3). A Moran’s I coefficient larger than its expected value indicates positive spatial autocorrelation, and a Moran’s I less than its expected value indicates negative spatial autocorrelation.

22.2.2 Geary’s c

Formally, Geary’s c is:

\[ c = \frac{(N-1)}{2S_0} \left( \sum w_{ij} (x_i - x_j)^2 / \sum (x_i - \mu)^2 \right) \]

in the same notation as above. The theoretical expected value for Geary’s c is 1. The theoretical variance of Geary’s c depends on the stochastic assumptions that are made (see 22.2.3). A value of Geary’s c of less than 1 indicates positive spatial autocorrelation, while a value larger than 1 points to negative spatial autocorrelation.

22.2.3 Inference

Instead of using the I or c statistics by themselves, inference is typically based on a standardized z-value. This is computed by subtracting the theoretical mean and dividing the result by the theoretical standard deviation. For example, for Moran’s I, this would give:

\[ z = \frac{(I - E[I])}{SD[I]} \]

where E[I] is the theoretical mean and SD[I] is the theoretical standard deviation.

For different assumptions about the data and the nature of spatial autocorrelation, the theoretical expressions for E[I], E[c], SD[I] and SD[c] will vary. Consequently, the value for \( z_i \) and \( z_c \) will vary as well, as will the interpretation of significance (or lack thereof). For a technical discussion and detailed expressions for the moments under the various assumptions, see Cliff and Ord (1973, 1981).

The most common approach is to assume that the variable in question follows a normal distribution. Based on asymptotic considerations (i.e., by assuming that the sample may become infinitely large) the z-value, using the proper measures for mean and standard deviation, follows a standard normal distribution (i.e., a normal distribution with mean 0 and variance 1). Sig-
nificance of the statistic can then be judged by comparing the computed z-value to its probability in a standard normal table.

A second, often used approach is to assume that each value observed could equally likely have occurred at all locations. In other words, the location of the values and their spatial arrangement is assumed to be irrelevant. This is referred to as the randomization assumption. Based on this assumption, different theoretical standard deviations for Moran's I and Geary's c are obtained, which yield different z-values. Again, these z-values follow a standard normal distribution (asymptotically) so that their significance can be judged by means of a standard normal table.

A final approach is similar to the randomization assumption in that again, each value is taken to be equally likely to be observed at any location. However, rather than using a theoretical mean and standard deviation, a reference distribution for I or c is generated empirically, from which the moments (mean and standard deviation) are computed. In practice, this is carried out by randomly reshuffling the observed values over all locations (i.e., by permuting the values) and by re-computing the I or c statistic for each new sample. This is referred to as the permutation approach. The mean and standard deviation for I or c are then simply the computed moments for the reference distribution for all permutations. The permutation approach takes considerably longer to compute than the two other alternatives.

**SpaceStat** includes all three options in both the Moran and the Geary menu of the Explore module. Each option is invoked as a separate command. Furthermore, there are two ways to present the statistic, depending on the focus of your interest. If you are mostly interested in evidence on spatial autocorrelation for several variables, using the same weights matrix (e.g., typically when you have more variables than weights matrices), you should use the standard option, i.e., the commands Normal (E-3-1 or E-4-1), Randomization (E-3-2 or E-4-2) or Permutation (E-3-3 or E-4-3). If, on the other hand, you are interested in the difference in spatial autocorrelation for a variable over different weights matrices, you should use the corresponding Correlogram commands (respectively, E-3-4 or E-4-4, E-3-5 or E-4-5, and E-3-6 or E-4-6).

### 22.2.4 Interpretation

A positive and significant z-value for Moran's I (as shown by a low probability reported by **SpaceStat**) indicates positive spatial autocorrelation. In other words, similar values, either high values or low values, are more spatially clustered than could be caused purely by chance. The same is evidenced by a negative and significant z-value for Geary's c.

In contrast, a negative and significant z-value for Moran's I, and a positive and significant z-value for Geary's c indicate negative spatial autocorrelation, i.e., the opposite of clustering.
This concept of negative spatial autocorrelation is somewhat harder to grasp. It reflects a lack of clustering, more so than would be the case in a random pattern. Perfect negative spatial autocorrelation is represented by a checkerboard pattern.

The results for both Moran’s I and Geary’s c are to a large extent determined by the choice of the spatial weights matrix. It is important to keep this in mind when interpreting the results. In general, it is a good practice to check the sensitivity of your conclusions to the choice of the weight matrix.

### 22.3 Example Problem File

To illustrate these concepts, you will construct a spatial correlogram, using Moran’s I and Geary’s c for the variables CRIME, INCOME and HOUSING in the Columbus example. A choropleth map for CRIME is given as Appendix G. In Chapter 19, you created a Problem File for this analysis as file COL1.BTC. For the sake of completeness its contents are listed here again in Table 22.1. The Problem File COL1.BTC is also included on the \EXAMPLES directory.

#### Table 22.1 Problem file COL1.BTC

<table>
<thead>
<tr>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>0</td>
</tr>
</tbody>
</table>

You create the spatial correlogram using Moran’s I, for each of the three assumptions by means of the command sequences E-3-4, E-3-5 and E-3-6, respectively for the normal, randomization and permutation approach. Enter col1.btc in response to the prompt for the Problem File in the Batch option. For Geary’s c, the corresponding command sequences are E-4-4, E-4-5 and E-4-6.

### 22.4 Program Output

#### 22.4.1 Moran’s I

The results of the Moran’s I spatial correlogram, as they appear on your screen and are written to the output file you specified in the Output File option are listed in Table 22.2 for the normal, Table 22.3 for the randomization, and Table 22.4 for the permutation approach. These results appear one screen at the time, and you need to press the Return key to get the next screen. I have only listed the results for the first variable, CRIME. The results for the other variables are similar in appearance and listing them would be too repetitive.
Table 22.2 Spatial Correlogram for Crime, Using Moran’s I - Normal Assumption

<table>
<thead>
<tr>
<th>WEIGHT</th>
<th>I</th>
<th>MEAN</th>
<th>ST.DEV.</th>
<th>Z-VALUE</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLWS_1</td>
<td>0.5109472</td>
<td>-0.021</td>
<td>0.093701</td>
<td>5.675307</td>
<td>0.000000</td>
</tr>
<tr>
<td>COLWS_2</td>
<td>0.1684821</td>
<td>-0.021</td>
<td>0.072083</td>
<td>2.626342</td>
<td>0.004315</td>
</tr>
<tr>
<td>COLWS_3</td>
<td>-0.1389299</td>
<td>-0.021</td>
<td>0.065526</td>
<td>-1.802275</td>
<td>0.035751</td>
</tr>
</tbody>
</table>

The first part of Table 22.2 shows a summary of the characteristics of the spatial weights matrices (this summary is not repeated for the other tables, but is generated each time a Moran or Geary command is invoked). By checking this list, you can discover problems in terms of unstandardized weights, or weights that contain zero rows (i.e., unconnected observations). As expected, all three weights used in the analysis are row-standardized and have no zero-rows.

The second part of Table 22.2 contains a small header which lists the assumption used to compute the moments for Moran’s I (normal approximation), the data set used (COL) and the variable for which the statistics are computed (CRIME). A similar heading is found in the other tables as well.

Table 22.3 Spatial Correlogram for Crime, Using Moran’s I - Randomization Assumption

<table>
<thead>
<tr>
<th>WEIGHT</th>
<th>I</th>
<th>MEAN</th>
<th>ST.DEV.</th>
<th>Z-VALUE</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLWS_1</td>
<td>0.5109472</td>
<td>-0.021</td>
<td>0.094386</td>
<td>5.634091</td>
<td>0.000000</td>
</tr>
<tr>
<td>COLWS_2</td>
<td>0.1684821</td>
<td>-0.021</td>
<td>0.072604</td>
<td>2.607513</td>
<td>0.004560</td>
</tr>
<tr>
<td>COLWS_3</td>
<td>-0.1389299</td>
<td>-0.021</td>
<td>0.065990</td>
<td>-1.789612</td>
<td>0.036758</td>
</tr>
</tbody>
</table>

1. If you set the Interactive Operation option to NO in the Options menu, you will not have to press Return after each screen. However, there is no longer a pause between screens, so that it becomes very difficult to see the results. This is only practical if you want to run a very large number of analyses and primarily want to study them in the ascii output file.
The header is followed for each weights matrix (COLSW_1, COLSW_2 and COLSW_3) by the value for I, its theoretical mean and standard deviation, the associated z-value and the probability for this z-value according to a standard normal distribution. For example, Moran’s I of 0.51 for CRIME with a first order (row-standardized) contiguity yields a z-value of 5.68, which is highly significant. This indicates strong positive spatial autocorrelation, confirming the visual impression of spatial clustering given by the map in Appendix G. Note that the expected value E[I] is the same for all three weights matrices, i.e., -0.021 = -1/48 (49 observations for the Columbus neighborhoods). However, the theoretical standard deviation is different for each weights matrix, i.e., 0.094 for first, 0.072 for second, and 0.066 for third order contiguity. If you check the results for the other variables, you will see that this theoretical standard deviation is only a function of the weights matrix. For example, the same value of 0.072 is found for Moran’s I for all variables computed with COLWS_1.

The correlogram as a whole shows significant and strong spatial autocorrelation for the first and second orders of contiguity, but much less significance (a p-value of 0.04) for the third order (which also yields negative spatial autocorrelation). This pattern of decreasing autocorrelation with increasing orders of contiguity is typical of many spatial autoregressive processes.

The results in Table 22.3 and Table 22.4 are very similar. Note that under the randomization assumption, the expected value is still -0.021, but this is not the case for the permutation approach. In the latter, the expected value is the mean for the empirical distribution derived from the resampled data (e.g., -0.048 for COLWS_1 in Table 22.4). Note also how the theoretical standard deviation is different from the one in Table 22.2. In terms of significance, the pattern shown in Table 22.3 is virtually identical to that in Table 22.2. In Table 22.4 there are slight differences. However, remember that the pseudo significance level reported by SpaceStat is determined in part by the number of resampled data sets that were used. Since only 99 permutations were generated, the highest degree of significance that could be obtained is 0.01 (i.e., none of the resampled Moran’s I are larger than or equal to the observed Moran’s I).
22.4.2  Geary’s c

The results for Geary’s c are given in Table 22.5 for the normal, Table 22.6 for the randomization, and Table 22.7 for the permutation approach. The tables follow the same format as the ones for Moran’s I. Note that for Geary’s c the permutation approach takes considerably longer to compute than the other approaches. The pattern of significance is consistent with that found for Moran’s I, except for the third order contiguity. Geary’s c does not show a significant autocorrelation, irrespective of the assumption used. Also, note that the signs for the z-values are the opposite of those found for Moran’s I. For example, Geary’s c of 0.53 for first order contiguity is less than its theoretical expectation of 1 (or empirical mean of 2.03), which yields a negative z-value.

Table 22.5 Spatial Correlogram for Crime, Using Geary’s c - Normal Assumption

<table>
<thead>
<tr>
<th>DATA SET: COL WS</th>
<th>VARIABLE: CRIME</th>
<th>WEIGHT</th>
<th>C</th>
<th>MEAN</th>
<th>ST.DEV.</th>
<th>Z-VALUE</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLWS_1</td>
<td></td>
<td>0.5298745</td>
<td>1.000</td>
<td>0.101348</td>
<td>-4.638736</td>
<td>0.000002</td>
<td></td>
</tr>
<tr>
<td>COLWS_2</td>
<td></td>
<td>0.8112884</td>
<td>1.000</td>
<td>0.087445</td>
<td>-2.158052</td>
<td>0.015462</td>
<td></td>
</tr>
<tr>
<td>COLWS_3</td>
<td></td>
<td>1.130277</td>
<td>1.000</td>
<td>0.091190</td>
<td>1.428636</td>
<td>0.076554</td>
<td></td>
</tr>
</tbody>
</table>

Table 22.6 Spatial Correlogram for Crime, Using Geary’s c - Randomization Assumption

<table>
<thead>
<tr>
<th>DATA SET: COL WS</th>
<th>VARIABLE: CRIME</th>
<th>WEIGHT</th>
<th>C</th>
<th>MEAN</th>
<th>ST.DEV.</th>
<th>Z-VALUE</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLWS_1</td>
<td></td>
<td>0.5298745</td>
<td>1.000</td>
<td>0.098756</td>
<td>-4.760466</td>
<td>0.000001</td>
<td></td>
</tr>
<tr>
<td>COLWS_2</td>
<td></td>
<td>0.8112884</td>
<td>1.000</td>
<td>0.082422</td>
<td>-2.289586</td>
<td>0.011023</td>
<td></td>
</tr>
<tr>
<td>COLWS_3</td>
<td></td>
<td>1.130277</td>
<td>1.000</td>
<td>0.083154</td>
<td>1.566703</td>
<td>0.058592</td>
<td></td>
</tr>
</tbody>
</table>

22.5  Exercise

To further practice the computation of spatial correlograms, you can repeat the analysis for the African or Irish data, or for your own data.

References


Table 22.7 Spatial Correlogram for Crime, Using Geary’s c - Permutation Assumption

<table>
<thead>
<tr>
<th>DATA SET: COL</th>
<th>VARIABLE: CRIME</th>
<th>WEIGHT</th>
<th>I</th>
<th>MEAN</th>
<th>ST.DEV.</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLWS_1</td>
<td></td>
<td>0.5298745</td>
<td>0.010000</td>
<td>2.028</td>
<td>1.461809</td>
<td></td>
</tr>
<tr>
<td>COLWS_2</td>
<td></td>
<td>0.8112884</td>
<td>0.020000</td>
<td>1.983</td>
<td>1.429492</td>
<td></td>
</tr>
<tr>
<td>COLWS_3</td>
<td></td>
<td>1.130277</td>
<td>0.950000</td>
<td>2.010</td>
<td>1.447345</td>
<td></td>
</tr>
</tbody>
</table>
CHAPTER 23

G STATISTICS FOR OBSERVATION-SPECIFIC SPATIAL ASSOCIATION

23.1 Introduction

A slightly different approach to measuring spatial association was recently suggested by Getis and Ord (1992), based on so-called distance statistics. These statistics are computed by defining a set of neighbors for each location as those observations that fall within a critical distance (d) from the location. For each different distance a separate weights matrix is constructed, W(d).1

The G statistics developed by Getis and Ord can only be computed for positive variables. In addition to providing an alternative measure of overall spatial association (G), observation-specific measures of spatial association are introduced as well. These measures (the Gi and Gi* statistics) indicate the extent to which a location is surrounded by a cluster of high or low values for the variable under consideration. The G statistics are computed by means of the commands in the G-Stats menu of the Explore module. The first two commands pertain to the G measure of overall spatial association, either presented by weights matrix (E-5-1), or in the form of a spatial correlogram (E-5-2). The third command yields the Gi statistics (E-5-3) and the last one the Gi* statistics (E-5-4).

These statistics are invoked by entering the appropriate command sequence, followed by the name of the Problem File in batch mode (option 1), or by entering the requested information in interactive mode (option 2). A Problem File for the computation of G statistics must contain the name of a dataset, at least one spatial weights matrix, and the name of at least one variable. The weights matrices specified in the Problem File must be symmetric and should preferably only contain 0-1 values. If a weights matrix is not 0-1, it will be converted to that format by SpaceStat. In contrast, if the matrix is asymmetric, the G statistics cannot be computed and an error message is generated.

23.2 Methodology

23.2.1 G Statistic

Formally, the G statistic, for a chosen critical distance d, G(d), is defined as:

\[ G(d) = \frac{\sum_{i} \sum_{j} w_{ij}(d)x_{i}x_{j}}{\sum_{i} \sum_{j} x_{i}x_{j}} \]

1. This may be done by means of the Tools - Distance Wts - Distance to Binary Weights (T-4-1) command sequence.
where $x_i$ is the value observed at location $i$, and $w_{ij}(d)$ stands for an element of the symmetric (unstandardized) spatial weights matrix for distance $d$. The numerator of this statistic is similar to that of Moran’s $I$, but its denominator is different. Its significance is assessed by means of a standardized $z$-value, obtained in the usual fashion (see Chapter 22). The mean and variance for the $G(d)$ statistic can be derived under a randomization assumption and the $z$-value can be shown to tend to a standard normal variate in the limit (see Getis and Ord, 1992, for the detailed derivations).

### 23.2.2 $G_i$ and $G'_i$ Statistic

For each observation $i$, the $G_i$ and $G'_i$ statistics indicate the extent to which that location is surrounded by high values or low values for the variable under consideration, for a given distance band $d$. Formally, the $G_i$ and $G'_i$ statistics are defined as:

$$G_i = \frac{\sum w_{ij}(d) x_j}{\sum x_j}$$

where the $w_{ij}(d)$ are the elements of the contiguity matrix for distance $d$. The $G_i$ and $G'_i$ measures differ with respect to the number of observations that are included in the computation of the denominator, $\sum x_j$. For the $G_i$ statistic, $j + i$, while for the $G'_i$ statistic $j = i$ is included in the sum. In other words, the $G'_i$ measure provides a measure of spatial clustering that includes the observation under consideration, while the $G_i$ measure does not.

Inference about the significance of the $G_i$ and $G'_i$ statistics is based on a standardized $z$-value, which is computed by subtracting the theoretical mean (respectively, $E(G_i)$, or $E(G'_i)$) and dividing by the theoretical standard deviation. So far, only results for a normal approximation have been developed in the literature. You can find further technical details in the Getis and Ord (1992) paper.

### 23.2.3 Interpretation

A positive and significant $z$-value for a $G_i$, $G'_i$ or $G''_i$ statistic indicates spatial clustering of high values, whereas a negative and significant $z$-value indicates spatial clustering of low values. Note that this interpretation is different from that of the more traditional measures of spatial autocorrelation, where spatial clustering of like values - either high or low - are both indicated by positive autocorrelation. The identification of locations (or clusters of locations) with high $G_i$ or $G'_i$ statistics may help discovering problems with the spatial scale of your observational units, may assist in filtering out the spatial dependence, or may provide a clue to the existence of outliers. These statistics allow for the decomposition of a global measure of spatial association into its contributing factors, by location. They are thus particularly suitable
to detect potential non-stationarities in a spatial data set, e.g., when the spatial clustering is concentrated in one subregion of the data only. A global measure of spatial association, such as Moran’s I, Geary’s c, or the G(d) statistic, will fail to detect such a pattern.

23.3 Example Problem File

To illustrate these concepts, you will now compute the $G$, $G_i$ and $G_i^*$ statistics for the variable CRIME in the Columbus data set, using one of the distance-based contiguity matrices you created in Chapter 12 (e.g., COLD_1). The COL data set and the weights matrices are included in the \COLUMBUS directory. The Problem File to carry out these analyses is COL4.BTC, which is included in the \EXAMPLES directory and also listed in Table 23.1 (see Chapter 19 for details on constructing a Problem File).

You compute the $G$ statistic by means of the $E-5-1$ or $E-5-2$ command sequences (since there is only one weights matrix, the two are equivalent), the $G_i$ statistic by means of the $E-5-3$ command sequence, and the $G_i^*$ statistic by means of the $E-5-4$ command sequence.

23.4 Program Output

23.4.1 $G$ Statistic

The results for the $G$ statistic for CRIME are listed in Table 23.2. As is the case for Moran’s I, the first screen (the top of Table 23.2) summarizes the characteristics of the weights matrix that was specified in the Problem File, in terms of its row-standardization and the presence of zero rows. Since COLD_1 is a simple contiguity matrix constructed for a distance cutoff of 5, its lack of row standardization is indicated. The second screen (the bottom half of Table 23.2), lists the data set (COL), and for each variable and weight, the $G$ value, its theoretical mean and standard deviation, the corresponding z-value and significance level. For the CRIME variable, the strong indication of spatial clustering given by Moran’s I, Geary’s c, and the visual impression given by the map in Appendix G are confirmed here as well. The $G$ statistic of 0.165 yields a z-value of 6.63, which is highly significant.
23.4.2 $G_i$ and $G^*_i$ Statistics

The results for the $G_i$ and $G^*_i$ statistics are listed in Table 23.3 and Table 23.4. The statistics are preceded by a message about the characteristics of the weights matrices, which is not repeated in the tables given here, since it is the same as in Table 23.2. The main results consist of the $G_i$ or $G^*_i$ statistics and associated z-values for the observations with the 10 highest (most positive) and the 10 lowest (most negative) measures. The table header indicates the data set (COL), weights (COLD_1) and variable (CRIME) for which these measures were computed. The locations are identified by means of their sequence numbers in the data set in the first column. This column is labeled with the variable name specified in the Indicator Variable option, if the latter was set to YES. In Table 23.3, this is the variable NEIG. When the Indicator Variable option is set to NO, the label is OBS, as in Table 23.4. This is irrespective of the presence of a variable with this name in the data set.²

Table 23.2 G Statistic for CRIME

<table>
<thead>
<tr>
<th>TEST FOR SPATIAL ASSOCIATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUMMARY OF WEIGHTS MATRICES</td>
</tr>
<tr>
<td>Weights matrix COLD_1 is not row standardized</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>G-STATISTIC FOR SPATIAL ASSOCIATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA SET: COL</td>
</tr>
<tr>
<td>VARIABLE CRIME WEIGHT G MEAN ST.DEV. Z-VALUE PROB</td>
</tr>
<tr>
<td>CRIME COLD_1 0.1653341 0.111 0.008141 6.625949 0.000000</td>
</tr>
</tbody>
</table>

Table 23.3 Gi Statistics for CRIME

<table>
<thead>
<tr>
<th>G-1 STATISTICS FOR SPATIAL ASSOCIATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA SET: COL WEIGHTS: COLD_1 VARIABLE: CRIME</td>
</tr>
<tr>
<td>LARGEST (MOST POSITIVE Z) VALUES</td>
</tr>
<tr>
<td>NEIG G-1 Z PROB</td>
</tr>
<tr>
<td>35 0.287489 3.6817 0.0001</td>
</tr>
<tr>
<td>42 0.286348 3.6369 0.0001</td>
</tr>
<tr>
<td>36 0.306275 3.4824 0.0002</td>
</tr>
<tr>
<td>37 0.274613 3.2289 0.0006</td>
</tr>
<tr>
<td>32 0.248923 3.2122 0.0007</td>
</tr>
<tr>
<td>38 0.272178 3.1622 0.0008</td>
</tr>
<tr>
<td>31 0.31986 3.1226 0.0009</td>
</tr>
<tr>
<td>9 0.221664 3.0891 0.0010</td>
</tr>
<tr>
<td>34 0.189675 2.8862 0.0019</td>
</tr>
<tr>
<td>33 0.2404 2.8771 0.0020</td>
</tr>
</tbody>
</table>

² You can always create a variable that explicitly contains the observation numbers. To accomplish this, you need to use the Data - Var Create - Create Observation Numbers command sequence (D-4-6).
The results for $G_i$ and $G_i^*$ are very similar. For example, for $G_i$, the three highest $z$-values (in Table 23.3) are 3.68 for neighborhood 35, 3.64 for neighborhood 42, and 3.48 for neighborhood 36, compared to the three highest $z$-values for $G_i^*$ (in Table 23.4) of 3.66 for neighborhoods 42 and 36, and 3.62 for neighborhood 35. These are all highly significant, indicating strong clusters of high crime values focused on these locations. If you refer to the map of the Columbus neighborhoods in Chapter 10 (Figure 10.1), you will note that these neighborhoods are immediately to the north of the Downtown area (typically defined by the intersection of High Street and Broad Street, which is on the boundary between neighborhoods 34 and 35 in Figure 10.1), and are often characterized as high crime areas. In contrast, the locations with the lowest (most negative) $z$-values are neighborhoods 46 ($G_i$ $z$-value of -2.39 and $G_i^*$ $z$-value of -2.67) and 49 ($G_i$ $z$-value of -2.18 and $G_i^*$ $z$-value of -2.44), both located in the western suburbs of the city.

If you set the Output File option to YES, these results are also written to the file you specified.

The results of $G_i$ or $G_i^*$ for all locations can be obtained by setting the Long Output option in the Options menu to YES. The 10 highest and 10 lowest values are then followed by a list of results for $G_i$ or $G_i^*$ and their associated $z$-values for all observations, both unsorted and sorted by magnitude. This long set of values is not written to the regular output file, but to the Report File instead. If you run the $G_i$ or $G_i^*$ commands with the Problem File COL4.BTC and set Long Output to YES, these results will be contained in the file REP.DOC.

Having the long list of $G_i$ or $G_i^*$ statistics and $z$-values in a separate file allows you to import these results into a graphics or mapping package, possibly after some minor editing (e.g., the removal of the header and additon of variable or format specifications). You can
then associate symbols to the most and least significant values and visualize the results as aids in exploratory spatial data analysis. This is illustrated for the Columbus CRIME data in Appendix H, where the most extreme $G_i$ statistics for the CRIME variable are represented by triangles. Triangles with the top up indicate positive z-values, while triangles with the top down indicate negative ones. Only three sizes of triangles are given, corresponding respectively to significance levels of $p<0.01$, $0.01<p<0.05$ and $0.05<p<0.10$. Other recent illustrations of the use of the $G_i$ statistics in exploratory spatial data analysis (by interfacing SpaceStat with a GIS) can be found in Anselin (1992) and Anselin and O’Loughlin (1992).

23.5 IDRISI Interface

When the Idrisi Interface option is set to YES, irrespective of the setting of the Output to a File or Long Output options, a number of additional output files are created. For the last weights matrix specified in the Problem File, three separate files are created for each variable, with a file extension .VAL. For example, for the CRIME variable the corresponding file names would be: CRIME_G.VAL, CRIME_Z.VAL and CRIME_P.VAL. Each of these files is in a format that can be readily converted into a new IDRISI image, provided that an image already exists with a polygon indicator that matches the indicator variable set in the SpaceStat options. If no indicator variable was set, the simple sequence number of observations will be used. The files contain two columns, one for the values of the indicator variable, the other for the matching statistic: the $G_i$ or $G_i^*$ statistic in the file with the _G added to the variable name as file name; the corresponding z-value in the file with the _Z added; and the associated probability in the file with the _P added in the file name.

The special output files are only created for the last weights matrix listed in the Problem File. The file name of the output contains the variable name, but it cannot differentiate between weights matrices. It is therefore up to you to keep track of which weights were used to compute the statistics. In other words, if you want to take advantage of the Idrisi Interface feature, you should not specify multiple spatial weights in the problem file, since the resulting output file may not be for the weights you intended.

23.6 Exercise

You can now proceed and compute the $G$, $G_i$ and $G_i^*$ statistics for the other distance weights created for the Columbus data set, COLD_2 and COLD_3, or for distance weights for the African or Irish data.

---

3. The maps in Appendices F and G were generated by means of the ARC/INFO GIS.
References


CHAPTER 24

QAP MEASURES OF ASSOCIATION

24.1 Introduction

Several measures of spatial association, such as the join count statistics, Moran's I and Geary's c have been shown to be special cases of a general cross-product statistic (also referred to as QAP, or Quadratic Analysis Paradigm). Such an index is often termed a gamma index ($\Gamma$), and indicates the association between two matrices of similarity (or dissimilarity) for a set of objects. The gamma index is an example of combinatorial data analysis, as outlined in Hubert (1985, 1987), and applied to measures of spatial autocorrelation in Hubert et al. (1981, 1985), among others.

The $\Gamma$ index is based on two (or more) matrices that show the similarity between items. SpaceStat has commands to compute the $\Gamma$ statistic in the QAP menu of the Explore module.

Both a generic matrix comparison (E-6-1) as well as measures specialized to indicate spatial association are included. For the latter, three options are available:

- QAP - Moran: E-6-2
- QAP - Geary: E-6-3
- QAP - Sokal: E-6-4

The QAP-Moran and QAP-Geary measures are equivalent to the statistics described in Chapter 22. However, they do not yield identical numerical results and differ from the traditional measures by a scaling factor. The procedures included in the QAP menu are the same as the ones outlined in Anselin (1986).

The commands are invoked by entering the appropriate command sequence, followed by the name of the Problem File in batch mode (option 1), or by entering the requested information in interactive mode (option 2). Two different types of Problem Files are used in the QAP menu. For the Matrix and Partial Matrix Association command, the Problem File must contain the names of at least two matrices, but no data set, nor variable names. For the QAP measures of spatial association, the Problem File follows the same format as for the other tests, i.e., it must contain the name of a dataset, at least one spatial weights matrix, and the name of at least one variable. The weights matrix may be unstandardized or row-standardized.
24.2 Methodology

24.2.1 QAP Measures of Matrix Association

A $\Gamma$ index of matrix association shows the degree of correspondence between two matrices that contain a measure of similarity between objects. Each element $i,j$ in such a similarity matrix reflects the similarity between the object that corresponds to the row $(i)$ and the object that corresponds to the column $(j)$. For example, the similarity may be formalized such that each element in the matrix, say $a_{ij}$, is the product or difference between two values, e.g., $a_{ij} = x_i x_j$ or $a_{ij} = x_i - x_j$.

Formally, the gamma index of matrix association consists of the sum of all the cross products between the corresponding elements of each matrix of similarity:

$$\Gamma = \sum_i \sum_j a_{ij} b_{ij}$$

where $a_{ij}$ and $b_{ij}$ are two measures of similarity between objects $i$ and $j$, contained in square matrices $A$ and $B$, of dimension $N$ by $N$.

24.2.2 QAP Measures of Partial Matrix Association

The association between two matrices $A$ and $B$ may be due in part to the similarity between objects that is reflected in a third matrix, $T$. The gamma cross-product statistic has been extended to include such a concept of partial association. Formally, this is carried out by considering the association between a matrix $C$ consisting of the products of $a_{ij}$ and $b_{ij}$, and a third matrix $T$, with elements $t_{ij}$:

$$\Gamma_{AB(T)} = \sum_i \sum_j (a_{ij} b_{ij}) t_{ij}$$

The extent to which $\Gamma_{AB(T)}$ is greater than its expectation indicates the degree to which some of the association between $A$ and $B$ may be attributed to $T$. On the other hand, the degree to which $\Gamma_{AB(T)}$ is less than its expectation would show how $T$ may have a suppression effect on the association between $A$ and $B$ (see Hubert, 1985, for technical details).

If the matrix $T$ consists only of values between 0 and 1, the association between $A$ and $B$ may be decomposed into two parts, one that can be attributed to $T$, the other consisting of what is left after $T$ is taken into account. In the statistical literature, these parts are referred to as the differential index and the partial index:

$$\Gamma_{AB} = \Gamma_{AB(T1)} + \Gamma_{AB(T2)}$$
where \( \{T_1\}_{ij} + \{T_2\}_{ij} = 1 \), or, alternatively,

\[
\{T_1\}_{ij} = t_{ij}
\]

\[
\{T_2\}_{ij} = 1 - t_{ij}
\]

Even when the values in T are not between 0 and 1, they can be easily converted to this range by means of a simple rescaling (e.g., by dividing all elements by the maximum in the matrix). However, in order for the logic of differential and partial association to work, the association between A and B must be positive (i.e., it must measure the same direction of similarity). In **SpaceStat**, a check for this condition is automatically carried out and an error message is generated when it is not present. Also, **SpaceStat** automatically carries out the rescaling of the matrix T when needed.

### 24.2.3 QAP Measures of Spatial Association

Measures of spatial association can be expressed in the form of a gamma index by making one of the matrices of similarity correspond to a contiguity or distance matrix, W. Indeed, each element \( w_{ij} \) in such a matrix indicates the locational similarity between the object corresponding to the row (i) and the object corresponding to the column (j). In **SpaceStat**, the weights matrix is always considered to be the second matrix in the computation of the \( \Gamma \) index. The first matrix, say A, with elements \( a_{ij} \), then expresses value association. Depending on the form of the latter, three types of spatial association may be encompassed:

- **a Moran-like measure:**
  \[ a_{ij} = x_i \cdot x_j \]

- **a Geary-like measure:**
  \[ a_{ij} = (x_i - x_j)^2 \]

- **a Sokal-like measure:**
  \[ a_{ij} = |x_i - x_j| \]

In **SpaceStat**, these three measures are computed for the variables in standardized form:

\[ z_i = (x_i - \mu) / \sigma \]

where \( \mu \) is the mean of \( x \) and \( \sigma \) is its standard deviation.

The gamma index for QAP-Moran is then:

---

1. See Royaltey et al. (1975) and Sokal (1979) for further details on the original idea behind this statistic.
the gamma index for QAP-Geary:

$$\Gamma = \Sigma_i \Sigma_j (z_i z_j)w_{ij}$$

and the gamma index for QAP-Sokal:

$$\Gamma = \Sigma_i \Sigma_j (z_i - z_j)^2 w_{ij}$$

The relation between the gamma indices and the traditional measures of spatial autocorrelation can be easiest seen for the Moran statistic. The QAP-Moran index is $S_0$ times the traditional Moran’s I, where $S_0$ is, as before, the sum of all elements of the weights matrix, $\Sigma_i \Sigma_j w_{ij}$.

The QAP indices of spatial association may be computed for an unstandardized or row-standardized weights matrix. However, in order to ensure the validity of the inference for the general product moment statistic, all weights matrices are made symmetric by means of the usual transformation, $(W+W')/2$, where $W'$ is the transpose of the weights matrix.

### 24.2.4 Partial Spatial Association

An indication of partial spatial association may be computed for the three QAP spatial association indices by specifying an indicator variable for structural change in the Problem File (or in response to the interactive queries). Such an indicator variable takes on only integer values, with each value corresponding to a category or regime. SpaceStat transforms the indicator variable into a third matrix of similarity in order to compute the differential and partial $\Gamma$ indices outlined above. Formally, this matrix is obtained as the outcome of the following logical expression:

$$T = t \cdot eqt'$$

where `eq` is an element by element logical equal operator, $t$ is the vector with observations on the indicator variable, $t'$ is its transpose, and $T$ the third matrix of similarity. The result of this expression is a block diagonal matrix where each block corresponds to a particular category or regime. In spatial data analysis, such regimes are often subregions of the data set. The index of partial spatial association provides insight into the extent to which the spatial association may be due to the regional differentiation (or, spatial heterogeneity) reflected in subregions of the data. Partial spatial association is computed in the same manner for the join count statistics discussed in Chapter 21.
24.2.5 Inference and Interpretation

The gamma index in and of itself is not very meaningful, since it is scale dependent. In order to facilitate its interpretation, a number of pseudo correlation coefficients have been proposed that are simple transformations of the raw $\Gamma$ index. These transformations do not yield correlation coefficients in a strict sense, but they result in absolute values that are less or equal to 1, which sometimes facilitates their interpretation as an intuitive measure of association. For example, it can be shown that with the proper standardization, the $\Gamma$ index for a particular form of A and B matrices reduces to familiar measures of rank correlation (for technical details, see Hubert, 1985 and 1987).

In SpaceStat, two alternative transformations are implemented, denoted as CORR1 and CORR2 in the program output. The first one corresponds to:

$$
\Gamma_1 = \sum_i \sum_j \left\{ \frac{[a_{ij} - \mu(a_{ij})]}{\sigma(a_{ij})} \right\} \left\{ \frac{[b_{ij} - \mu(b_{ij})]}{\sigma(b_{ij})} \right\} / N(N-1)
$$

where $\mu(a_{ij})$ and $\mu(b_{ij})$ are the means for the elements of respectively the A and B matrix, and $\sigma(a_{ij})$ and $\sigma(b_{ij})$ are the respective standard deviations.

The second pseudo-correlation coefficient corresponds to:

$$
\Gamma_2 = \sum_i \sum_j a_{ij} b_{ij} / [\left( \sum_i \sum_j a_{ij}^2 \right) \left( \sum_i \sum_j b_{ij}^2 \right)]^{1/2}
$$

This type of coefficient is referred to in the literature as Daniel’s generalized correlation coefficient (for technical details, see Hubert, 1985 and 1987).

Inference with respect to the $\Gamma$ index is based on two approaches. In the first, a normal approximation is used to convert the raw $\Gamma$ index to a standard z-value, in the usual fashion:

$$
z(\Gamma) = \{ \{ \Gamma - E[\Gamma] \} \} / \{ V[\Gamma] \}^{1/2}
$$

The moments $E[\Gamma]$ and $V[\Gamma]$ for the $\Gamma$ statistic are derived in Mielke (1979). However, the normal distribution for the z value is only an approximation, and a higher order approximation based on a third moment can be shown to be more precise (Costanzo et al., 1983; Anselin, 1986).

The second approach is the preferred one. It is based on the permutation strategy outlined in Chapter 22. The mean and standard deviation for $\Gamma$ are computed from the empirical distribution of $\Gamma$ indices generated by the permuted samples. A pseudo significance level is obtained as $T+1/M+1$, where $T$ is the number of resampled $\Gamma$ indices that are equal to or larger than the observed one, and $M$ is the number of replications (99 is the default). A low value of this pseudo significance implies that $\Gamma$ is extreme with respect to its reference dis-
tribution, and thus that the null hypothesis of no association should be rejected. Alternatively, the value for $\Gamma$ may be compared to the mean and standard deviation in the reference distribution and the two-sigma rule used to assess significance: the null hypothesis is rejected if $\Gamma$ differs from its empirical mean by more than 2 standard deviations.

The $\Gamma$ index and associated $z$-value may be interpreted in two ways. As a one-sided test, an extreme $\Gamma$ is one that is much larger than its expected value, i.e., a positive $z$-value. For the matrix association and QAP Moran measures this implies a positive association, while for QAP-Geary and QAP-Sokal it implies a negative association (negative spatial autocorrelation). As a two-sided test, both very large and very small values of the $\Gamma$ index are taken as an indication to reject the null hypothesis of no association. Note that for QAP-Geary and QAP-Sokal, a negative $z$-value yields the more intuitive interpretation of positive spatial association.

### 24.3 Example Problem File

To illustrate these concepts, you will need two different Problem Files, one for the generic matrix comparison, the other for the spatial QAP measures.

#### 24.3.1 Generic Matrix Comparison

In Table 24.1, three 5 by 5 matrices are listed that you can use as inputs into the Matrix and Partial Matrix Association command. These matrices must first be converted into the GAUSS format by means of the Data-Input-Ascii to Matrix command sequence. They are included on the \EXAMPLES directory as the files MAT1.FMT, MAT2.FMT and MAT3.FMT.

The purpose of this example is to first compare the measures of similarity between five objects that are expressed in the matrices MAT1 and MAT2, and subsequently to assess the extent to which this may be attributed to the similarity in matrix MAT3. Note that the latter may be easily constructed by means of the logical operation $t .eq t'$ from a column vector $t$ with as observations 0, 0, 1, 1, 1.

The Problem File to carry out these matrix comparisons, COL5.BTC, is listed in Table 24.2. The file COL5.BTC is also included on the \EXAMPLES directory. Note that no data set or variable names are given in this Problem File. Refer to Chapter 19 for details on the creation of a Problem File. You invoke the matrix comparison by means of the $E-6-1$ command sequence, followed by $col5.btc$ for the Batch option.

#### 24.3.2 QAP Measures of Spatial Association

An example Problem File for the computation of QAP measures of spatial association, COL6.BTC, is listed in Table 24.3. The file COL6.BTC is also included on the \EXAMPLES directory. The Problem File specifies COL as the data set, COLWS_1 as the spatial weights
file (these weights are the row standardized first order contiguity for the Columbus neighborhoods), and CRIME as the variable of interest. The variable EW is an indicator for structural change, taking on a value of 1 for neighborhoods in eastern part of the city. This variable will be used to compute a measure of partial spatial association. Note that its presence is indicated by a 1 for the ninth flag in the Problem File.

You invoke the QAP-Moran, QAP-Geary or QAP-Sokal functions by means of the command sequences E-6-2, E-6-3, or E-6-4. Enter col6.btc in response to the prompt for the Problem File in the Batch option.

Table 24.1 Input Matrices for Generic Matrix Comparison

<table>
<thead>
<tr>
<th>MAT1</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
</tr>
<tr>
<td>0    21  29  22  31</td>
</tr>
<tr>
<td>25   0    24  20  33</td>
</tr>
<tr>
<td>35   24   0    20  36</td>
</tr>
<tr>
<td>23   19   19   0    21</td>
</tr>
<tr>
<td>32   33   34   19   0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MAT2</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
</tr>
<tr>
<td>0    0    1    1    1</td>
</tr>
<tr>
<td>0    0    0    1    1</td>
</tr>
<tr>
<td>1    0    0    0    1</td>
</tr>
<tr>
<td>1    1    0    0    0</td>
</tr>
<tr>
<td>1    1    1    0    0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MAT3</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
</tr>
<tr>
<td>1    1    0    0    0</td>
</tr>
<tr>
<td>1    1    0    0    0</td>
</tr>
<tr>
<td>0    0    1    1    1</td>
</tr>
<tr>
<td>0    0    1    1    1</td>
</tr>
<tr>
<td>0    0    1    1    1</td>
</tr>
</tbody>
</table>

Table 24.2 Problem file COL5.BTC

| 1 |
| 0  0  3  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0 |

<table>
<thead>
<tr>
<th>MAT1</th>
<th>MAT2</th>
<th>MAT3</th>
</tr>
</thead>
</table>

24.4 Program Output

24.4.1 Matrix and Partial Matrix Association

The output for the QAP index of matrix association is given in Table 24.4. The header contains the names of the first and second matrix, and indicates which approach is taken to carry out inference (normal approximation or permutation). Next follow the \( \Gamma \) index (GAMMA), and the two pseudo correlation coefficients (CORR1 and CORR2). Under the normal approximation, the expected value of \( \Gamma \) (E[GAMMA]), its corresponding z-value (Z-GAMMA) and associated probability are given. For the permutation approach, the empirical mean of \( \Gamma \) (E[GAMMA]), its standard deviation (SD[GAMMA]) and associated pseudo significance level are listed, but no z-value. However, the latter can be easily computed as \( (\Gamma - E[GAMMA])/SD[GAMMA] \). The number of random permutations, 99, is the default set in the Number of Permutations option.

In the example, \( \Gamma \) is 347, substantially higher than both its theoretical mean of 312 and its empirical mean of 311.97. The corresponding correlation coefficients are 0.597 and 0.840. The \( \Gamma \) index is highly significant, as illustrated by a z-value of 2.39 under the normal approximation and a pseudo-significance level of 0.01 under the randomization assumption (i.e., the observed \( \Gamma \) was higher than any of the 99 resampled values).

If you set the Long Output option to YES, the values of the \( \Gamma \) indices computed for each of the resampled data sets will be given as well.

### Table 24.4 Matrix Association

<table>
<thead>
<tr>
<th>GAMMA</th>
<th>CORR1</th>
<th>CORR2</th>
<th>E[GAMMA]</th>
<th>Z-GAMMA</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>347</td>
<td>0.597023</td>
<td>0.839547</td>
<td>312</td>
<td>2.394790</td>
<td>0.008315</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GAMMA</th>
<th>CORR1</th>
<th>CORR2</th>
<th>E[GAMMA]</th>
<th>SD[GAMMA]</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>347</td>
<td>0.597023</td>
<td>0.839547</td>
<td>311.97</td>
<td>13.3957</td>
<td>0.010000</td>
</tr>
</tbody>
</table>

The output for the partial matrix association is listed in Table 24.5. Two sets of results are given, one using the third matrix and one using its complement. The two \( \Gamma \) indices add
up to the original measure, as required (see 24.2.2): \(70 + 277 = 347\). The pseudo correlation coefficients are the same in both instances, but CORR1 changes sign. The first index is below its expected value of 138.8, but not sufficient to indicate a significant association (z-value is 1.34 with \(p=0.09\)). In other words, the association between MAT1 and MAT2 cannot be attributed to the structure of similarity expressed in MAT3.

### Table 24.5 Partial Matrix Association

<table>
<thead>
<tr>
<th>QAP MATRIX ASSOCIATION</th>
<th>PARTIAL ASSOCIATION BASED ON MATRIX</th>
<th>MAT3</th>
</tr>
</thead>
<tbody>
<tr>
<td>First matrix: MAT1</td>
<td>Second matrix: MAT2</td>
<td></td>
</tr>
<tr>
<td>Using the normal approximation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GAMMA</td>
<td>CORR1</td>
<td>CORR2</td>
</tr>
<tr>
<td>70</td>
<td>-0.471567</td>
<td>0.242042</td>
</tr>
</tbody>
</table>

| First matrix: MAT1    | Second matrix: MAT2                |      |
| Using 99 random permutations |
| GAMMA | CORR1 | CORR2 | E[GAMMA] | SD[GAMMA] | PROB      |
| 70    | -0.471567 | 0.242042 | 148.303 | 57.7418 | 0.200000 |

<table>
<thead>
<tr>
<th>QAP MATRIX ASSOCIATION</th>
<th>PARTIAL ASSOCIATION BASED ON COMPLEMENT OF MATRIX</th>
<th>MAT3</th>
</tr>
</thead>
<tbody>
<tr>
<td>First matrix: MAT1</td>
<td>Second matrix: MAT2</td>
<td></td>
</tr>
<tr>
<td>Using the normal approximation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GAMMA</td>
<td>CORR1</td>
<td>CORR2</td>
</tr>
<tr>
<td>277</td>
<td>0.471567</td>
<td>0.782037</td>
</tr>
</tbody>
</table>

| First matrix: MAT1    | Second matrix: MAT2                |      |
| Using 99 random permutations |
| GAMMA | CORR1 | CORR2 | E[GAMMA] | SD[GAMMA] | PROB      |
| 277   | 0.471567 | 0.782037 | 198.697 | 57.7418 | 0.200000 |

#### 24.4.2 QAP-Moran

The output for the **QAP-Moran** measure of spatial association is listed in Table 24.6. The top half of the table gives the standard results, following the same format as for the generic matrix comparison. The only slight difference in the header is the explicit listing of the data set, weights matrix and variable name. The other items are the same. Note, that in contrast
to the output for the traditional Moran coefficient (in Chapter 22), no indication is given about the row-standardization of the weights matrix.

The results are given for both the normal and the permutation approach. Also, if the **Long Output** option is set to YES, the values of $\Gamma$ computed for each of the resampled data sets will be listed as well.

The $\Gamma$ index of 25.04 is 49 ($S_0$) times the coefficient for Moran's I of CRIME (0.51) found in Chapter 22. Note that the $z$-value of 5.63, which is highly significant, differs from the one for Moran's I, since different values are used for the theoretical mean and standard deviation (these are only approximations). A similar, strong indication of positive spatial association is given for the permutation approach (pseudo-significance of 0.01).

An indication of partial spatial association is given in the bottom half of the table. Using the indicator variable EW to construct a third matrix of similarity, the resulting $\Gamma$ coefficient is 19.07, which is significantly above its expected value of 12.69, yielding a $z$-value of 2.68 (significant at $p<0.01$). This suggests that the measure of spatial association shown by QAP-Moran can be explained by the existence of a spatial structural differentiation between the eastern and western neighborhoods in the Columbus urban area. In other words, the indication of spatial association in the CRIME variable may be due to the spatial differentiation of the landscape, i.e., to spatial heterogeneity. Such an indication may be used as a guide to select variables to be incorporated into a regression specification (see Part V). The results for the complement of EW are not listed in the table, but they are generated in a manner similar to that described for generic matrix comparison.

### 24.4.3 QAP-Geary

The output for the **QAP-Geary** measure of spatial association is illustrated in Table 24.7. The results of partial spatial association are not listed since they follow the same format as for QAP-Moran and are consistent with the indications given in Table 24.6.

The result for $\Gamma$ of 53.01 is significantly below its expected value of 100.04, yielding a $z$-value of -4.76. Note that the CORR1 pseudo-correlation coefficient picks up the negative sign (-0.167), but the CORR2 coefficient (0.098) does not. This negative value indicates positive spatial association, as for the Geary's $c$ statistic.

### 24.4.4 QAP-Sokal

The output for the **QAP-Sokal** measure of spatial association is illustrated in Table 24.8. Again, the results of partial spatial association are not listed since they follow the same format as for QAP-Moran and are consistent with the indications given in Table 24.6.
The result for $\Gamma$ of 37.08 is significantly below its expected value of 57.09, yielding a z-value of -5.99. Note that, similar to the results for QAP-Geary, the CORR1 pseudo-correlation
coefficient picks up the negative sign (-0.153), but the CORR2 coefficient (0.157) does not. This negative value indicates positive spatial association, as for the Geary's c statistic.

Table 24.8 QAP-Sokal Measure of Spatial Association

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>GAMMA</th>
<th>CORR1</th>
<th>CORR2</th>
<th>E[GAMMA]</th>
<th>Z-GAMMA</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRIME</td>
<td>37.075</td>
<td>-0.153108</td>
<td>0.156769</td>
<td>57.094</td>
<td>-5.987159</td>
<td>0.000000</td>
</tr>
</tbody>
</table>

Using 99 random permutations

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>GAMMA</th>
<th>CORR1</th>
<th>CORR2</th>
<th>E[GAMMA]</th>
<th>SD[GAMMA]</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRIME</td>
<td>37.075</td>
<td>-0.153108</td>
<td>0.156769</td>
<td>57.0084</td>
<td>3.3295</td>
<td>0.010000</td>
</tr>
</tbody>
</table>

24.5 Exercise

You can now extend the analysis of QAP measures of spatial association to the other variables in the Columbus data set, for the first order or higher order contiguity matrices. Alternatively, you may carry out the analysis for the African or Irish data, or for your own data.

References


PART V

SPATIAL REGRESSION ANALYSIS
CHAPTER 25

PROBLEM FILES IN THE REGRESS MODULE

25.1 Introduction

As is the case for the Explore module, all analyses in the Regress module need a Problem File as input. This file contains information on the data set, spatial weights and variables used in the statistical analysis. A Problem File conforms to a specific format, which is given in Appendix B. You may create a Problem File by means of a text editor, as long as the resulting file conforms to the format in Appendix B. Alternatively, you may also create a Problem File interactively, either in the process of carrying out a particular analysis, or for later use.

In the Regress module, the query for the Problem File operates for the most part in the same fashion as for the Explore module. The main features are illustrated in detail in Chapter 19 and will not be repeated here. There are however a few additional items that may be specified for the spatial regression models, which are not provided as prompts in the Explore module. These items pertain to special types of variables needed for the estimation of the models or in diagnostic tests.

The models in the Regress models are organized along two dimensions. The first dimension pertains to the general estimation methodology, and is reflected in the menu structure. There are four general classes of models, which each correspond to a menu in the Regress module:
- the classic regression model ($R-1$)
- the model with spatial error dependence ($R-2$)
- the model with heteroskedastic errors ($R-3$)
- the model with a spatially lagged dependent variable ($R-4$)

The second dimension pertains to the form of the model specification. There are five distinct forms incorporated in SpaceStat:
- Generic Regression (1)
- Trend Surface (2)
- Spatial Regimes (3)
- Spatial Expansion (4)
- ANOVA (5)

The model specification is determined by the first flag in the Problem File. It is the fourth query in the Interactive option. You must specify the sequence number of the model (as given in parentheses above). The special models 2 through 5 are discussed in more detail in Part VI.
The general structure as well as the specific items that must be entered in the **Problem File** for each of the two dimensions are outlined below.

### 25.2 Generic Structure of Problem Files in the Regress Module

The following items form the generic structure of a **Problem File** to carry out regression analysis. They are listed in the same sequence as how the prompts appear in the Interactive option.

- **Data set**: required
- **Report file**: optional
- If specified, the report file will be used for a complete listing of observed values, predicted values and residuals when the **Long Output** option is set to YES.
- **Spatial weights**: required for models with a spatial lag, optional for others
- When no spatial weights are specified, no diagnostics for spatial effects are carried out. However, all the other diagnostics are computed in the usual way.
- **Type of problem**: required
- The sequence number of the problem type must be specified: 1 is for the generic regression model, 2-5 are for the special spatial regression models (see Part VI).
- **Constant term**: optional
- The inclusion of a constant term is optional. However, it is the default (press **Return**) and is highly recommended. You should **never** include a constant term explicitly as one of the explanatory variables in the **Problem** File.
- **Variables**: required
- For the generic regression specification and in most other models, at least two variables must be specified. The first one is used as the dependent variable, the other as explanatory variables. In some models (i.e., models with a spatially lagged dependent variable) it is allowed to not specify any explanatory variables.
- **Heteroskedastic variables**: optional

  The default is to use the squares of the explanatory variables in the specification of the error variance used to test for additive heteroskedasticity and to estimate the generic heteroskedastic model. For some special models, another specification is used as the default (e.g., for the spatial expansion model and for spatial ANOVA, see Part VI). If you wish to override the default, you must specify the variables to be used in the heteroskedastic function explicitly. Note that the observations on these variables are always squared in the heteroskedastic function. In other words, if you wish a truly linear relation between the error variance and the variables in question, you must first transform them into their square roots (e.g., by means of the **Data-Var Transform-Square Root** command,
Heteroskedastic variables are ignored in the OLS-Robust estimation of the classical regression model (R-1-2) and in the instrumental variables estimation of the spatial autoregressive model (R-4-2 and R-4-3).

### 25.3 Distinctive Problem File Structure by Estimation Method

#### 25.3.1 Spatially Lagged Dependent Variable

For the spatial autoregressive model (R-4) and the model with spatially autocorrelated errors (R-2), you may specify the spatial lag Wy explicitly in the Problem File. For example, you may want to do this if you computed this lag earlier by means of the Tools-Space Trans-Spatial Lag command (T-1-1). However, the default is that SpaceStat computes the lag internally, based on the weights matrix and dependent variable specified in the Problem File. The default avoids potential problems with using a lag variable that was computed with an incompatible weights matrix.

In the Interactive option, the prompt for the spatially lagged dependent variable follows immediately after the specification of the other variables in the model. The variable name of the dependent variable is listed and you are asked to type in a variable name for the lag, or to press Return for the default.

#### 25.3.2 Spatially Lagged Explanatory Variables

For the model with spatially autocorrelated errors (R-2), you may also specify the spatially lagged explanatory variables WX explicitly in the Problem File. For example, you may want to do this if you computed these lag terms earlier by means of the Tools-Space Trans-Spatial Lag command (T-1-1). However, the default is that SpaceStat computes the lags internally, based on the weights matrix and explanatory variables specified in the Problem File.

In the Interactive option, the prompt for the spatially lagged explanatory variables follows the one for a spatially lagged dependent variable. It is organized in the same fashion: press Return for the default.

#### 25.3.3 Instrumental Variables

For the spatial autoregressive model that is estimated by means of instrumental variables or bootstrap methods (R-4-2 or R-4-3), you must specify the variable names for the instruments. If you fail to do so, an error message is generated. In the Interactive option, the prompt for the instruments is the very last one.
25.4 Distinctive Problem File Structure for Spatial Regression Models

25.4.1 Trend Surface

For the trend surface model, you must select the model specification as 2. You can only specify two explanatory variables. The first one corresponds to the X coordinates and the second one to the Y coordinates. These variables do not have to be labeled X and Y.

You must also set the order for the trend surface polynomial. This may be any positive integer. In the Interactive option, the prompt for the trend surface order follows immediately after the selection of the dependent and explanatory variables.

Note that you can also create the explanatory variables for a trend surface regression explicitly, by means of the Data-Var Algebra-Trend Surface command (D-6-7). You may include such variables as explanatory variables in the Problem File in the standard way. However, if you choose this approach, you must carry out estimation of the model as a generic regression (option 1) and not as a trend surface regression. In all other respects, the two approaches are equivalent.

25.4.2 Spatial Regimes

For the spatial regimes model, you must select the model specification as 3. You must also specify an indicator variable to define the regimes. This indicator variable may only take on integer values: one distinct value for each regime. SpaceStat converts the categorical indicator variable internally into the proper dummy variables. You can only specify one indicator variable to define the regimes. If you fail to specify this variable, an error message is generated.

In the Interactive option, the prompt for the structural change variable follows the query for dependent and explanatory variables (or comes after the prompt for the lagged dependent or lagged explanatory variables, when those are needed).

Instead of using the internal computation, you can also create the explanatory variables for a spatial regime regression explicitly, by means of the Data-Var Algebra-Regimes command (D-6-8). You may include such variables as explanatory variables in the Problem File in the standard way. However, if you choose this approach, you must carry out estimation of the model as a generic regression (option 1) and not as a regime regression. Also, several tests for the significance of the regimes (structural instability) will only be carried out if the regime specification is use.

25.4.3 Spatial Expansion

For the spatial expansion model, you must select the model specification as 4. You must also specify the expansion variables and the order for the expansion polynomial. The
latter may be either 1, for a linear expansion, or 2, for a quadratic expansion. If you enter any other values, a linear expansion is assumed.

In the Interactive option, the prompt for the expansion variables follows the queries for the dependent and explanatory variables (or comes after the prompt for the lagged dependent or lagged explanatory variables, when those are needed). The selection of expansion variables is immediately followed by a prompt for the order of the expansion.

Note that you can also create the explanatory variables for a spatial expansion regression explicitly, by means of the Data-Var Algebra-Expansion command (D-6-5). You may include such variables as explanatory variables in the Problem File in the standard way. However, if you choose this approach, you must carry out estimation of the model as a generic regression (option 1) and not as a spatial expansion regression. Also, several tests for the significance of the expansion (parameter drift) will only be carried out if the spatial expansion specification is use.

25.4.4 Spatial ANOVA

For the spatial ANOVA model, you must select the model specification as 5. There is no prompt for a constant term, since one is always used. The explanatory variables you specify for this model must be categorical indicator variables that take on a finite number of integer values. Each integer value corresponds to a regime or control. The indicator variables are converted into dummy variables internally. If you choose to specify the dummy variables explicitly, e.g., after you have computed them by means of the Data-Var Create-Create Dummy Variables (Categories) command (D-4-3), you must use the generic regression option (option 1). Otherwise, your model will fail due to perfect multicollinearity.

25.5 Example

To illustrate these concepts, an example follows for the construction of a Problem File for a generic regression model. The series of prompts is identical to the one illustrated in Chapter 19, except for the items listed here (the duplicate items will not be repeated).

After you start the procedure from the Classic Model-OLS menu in the Regress module (R-1-1), choose the third option (Make Problem File) and enter col7.btc as the Problem File. The screen will clear and the series of prompts will start, just as in Chapter 19. After entering the data set as col, the report file as rep, and the spatial weights as COLWS_1, COLWS_2 and COLWS_3, you are prompted for the type of model specification, as shown:

Model Specification for problem

Options:
1.  Generic Regression
2.  Trend Surface
3.  Spatial Regimes
4.  Spatial Expansion
5.  ANOVA

Enter option number: ? 1 (Return)

Enter 1 to select the generic regression specification. This is followed by a query for the use of a constant term. Press the Return key for the default of a constant term (highly recommended), as shown:

Constant term

Press Return for default (constant term) or N for no constant

This is followed by the prompts for the dependent and explanatory variables, in the same format as in Chapter 19. Enter crime for the dependent variable (the first variable name specified) and income and housing for the explanatory variables. Next, the screen clears and you are asked to specify the heteroskedastic variables. Type Return to select the default, as shown:

Heteroskedastic variables for problem

Enter variable names for additive heteroskedasticity or Return for default

Choose the variable(s) from the following list
(each variable name should correspond exactly to one in the list):
NEIG CRIME INCOME HOUSING X Y EW

Enter the variable name, or press Return to stop

Variable name: Return

This concludes the definition of the Problem File for a generic regression. The file COL7.BTC is included on the EXAMPLES directory. The contents of this file are shown in Table 25.1. Note that, apart from the 1 as the first flag (instead of 0), this Problem File is identical to the file COL1.BTC created in Chapter 19.

Table 25.1 Problem file COL7.BTC

<p>| | | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>COL</th>
<th>REP</th>
<th>COLWS_1</th>
<th>COLWS_2</th>
<th>COLWS_3</th>
<th>CRIME</th>
<th>INCOME</th>
<th>HOUSING</th>
</tr>
</thead>
<tbody>
<tr>
<td>COL7.BTC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
25.6 Exercise

You can now create Problem Files for analyses using the Irish or African data sets, or other data sets you may have created. In the remainder of Parts V and VI, I will assume that you know how to create a Problem File for the Regress module and will not repeat the various prompts for each command.
CHAPTER 26

STANDARD REGRESSION WITH DIAGNOSTICS FOR SPATIAL EFFECTS

26.1 Introduction

The standard approach to linear regression analysis is included in the Classic Model menu of the Regress module. There are two commands, one for the usual ordinary least squares regression, the other for a robust approach:

- **OLS** (R-1-1)

- **OLS - Robust** (R-1-2)

The main difference between the two commands is in the inference about the significance of the coefficients. The coefficients themselves are identical (and so are predicted values and residuals), but their estimated standard deviations and t-values are not. Also, most of the diagnostics are not computed in the robust option.

The commands are invoked by entering the appropriate command sequence, followed by the name of the Problem File in batch mode (option 1), or by entering the requested information in interactive mode (option 2). A Problem File for the estimation of a linear regression model must contain the name of a dataset, the type of model specification, a dependent variable and at least one explanatory variable. Spatial weights matrices may be specified, either in row-standardized or unstandardized form, but the former is recommended. If no spatial weights matrices are specified, no diagnostics for spatial effects will be computed. Variables to be used in an additive heteroskedastic specification may be included as well. For details on the creation of Problem Files in the Regress module, see Chapter 25.

26.2 Methodology

26.2.1 The Standard Regression Model

The general purpose of linear regression analysis is to find a (linear) relationship between a dependent variable and a set of explanatory variables. Formally, this relation is expressed as:

\[ y = X\beta + \varepsilon \]

where \( y \) is the dependent variable (in vector form, with \( N \) rows), \( X \) is a matrix with observations on \( K \) explanatory variables (with \( N \) rows and \( K \) columns), \( \beta \) is a vector with \( K \) regression coefficients (i.e., of dimension \( K \) by 1), and \( \varepsilon \) is a random error term (in vector form, with \( N \) rows). I will follow the usual notational convention and represent the unknown (population)
regression coefficients and unobservable random error by Greek symbols (respectively $\beta$ and $\varepsilon$), to contrast them with the estimated coefficient $b$, and residuals $e$.\(^1\)

There are typically two objectives one tries to achieve when carrying out regression analysis. One is to find a good match (or fit) between predicted values $Xb$ (sum of the values of the explanatory variables, each multiplied with their regression coefficient) and observed values of the dependent variable $y$. The other objective is to discover which of the explanatory variables contribute significantly to the linear relationship.

*Ordinary Least Squares (OLS)*

The method of ordinary least squares (OLS) estimation accomplishes both stated objectives in an optimal fashion according to a number of criteria, and is therefore referred to as a Best Linear Unbiased Estimator (BLUE).\(^2\) The OLS estimates for $\beta$ are found by minimizing the sum of squared prediction errors (hence, least squares).\(^3\)

In order to obtain the BLUE property and to be able to make statistical inferences about the population regression coefficients $\beta$ by means of your estimates $b$, you need to make certain assumptions about the random part of the regression equation (the random error $\varepsilon$). Two of these assumptions are crucial to obtain the unbiasedness and efficiency of the OLS estimates (the $U$ and $E$ part in BLUE):

- the random error has mean zero (there is no systematic misspecification or bias in the population regression equation):
  \[ E[\varepsilon_i] = 0 \quad \text{for all } i \]

- the random error terms are uncorrelated and have a constant variance (homoskedastic):
  \[ E[\varepsilon_i \varepsilon_j] = 0 \quad \text{for all } i + j \]
  \[ E[\varepsilon_i^2] = \sigma^2 \quad \text{for all } i \]

A third assumption is needed in order to carry out hypothesis tests and to assess significance of the regression coefficients:

- the random error term follows a normal distribution:
  \[ \varepsilon \sim N(0, \sigma^2) \]

---

1. The residuals are the difference between the observed value for the dependent variable, $y$, and the predicted value obtained with the OLS regression coefficients, $b$: $e = y - Xb$.
2. This is not the place to go into an extensive discussion of regression methods. My treatment will therefore be brief and may seem to lack rigor to specialists in the field. This is on purpose, and I refer you to any text on linear modeling, multivariate statistics or econometrics for a more technical discussion, such as Montgomery and Peck (1982), Johnston (1984), Judge et al. (1985), and Greene (1990).
3. In matrix notation, the OLS estimate $b = (X'X)^{-1}X'y$. 
In multivariate form, i.e., for all N observations jointly, these properties are expressed as:

- $E[\epsilon] = 0$
- $E[\epsilon\epsilon'] = \sigma^2 I$
- $\epsilon \sim N(0, \sigma^2 I)$

where $\epsilon$ is as before, a N by 1 vector of random error terms, $0$ is a N by 1 vector of zeros, $\sigma^2$ is the population error variance, and $I$ is an identity matrix of dimension N by N.

These assumptions introduce an additional parameter to be estimated (in addition to the regression coefficients $\beta$), i.e., the error variance $\sigma^2$, whose estimate I will denote by $s^2$. 4 SpaceStat reports both an unbiased and a maximum likelihood estimate for $\sigma^2$, as well as their square root (the standard deviation for the error term).

**Measures of Fit**

The extent to which the predicted values match the observed values for the dependent variable is measured by the $R^2$. This measure of fit is based on the decomposition of the total sum of squares, SST (the sum of squares for the dependent variable) into an explained sum of squares (the sum of squares of the predicted values), referred to as the regression sum of squares, SSR, and a residual sum of squares (the sum of squared residuals), RSS.5

$$SST = SSR + RSS$$

The $R^2$ is then defined as:

$$R^2 = 1 - \frac{RSS}{SST}$$

When the regression includes a constant term, this decomposition is equivalent to a decomposition of the total variance into an explained and residual variance (when all sums of squares are computed for the deviations from the mean).6

In general, the model with the highest $R^2$ is considered to have the best fit. However, the simple $R^2$ measure is not always a good indicator of how well the regression "explains" the observed values, since it increases with every additional explanatory variable that is included in the model specification. In order to provide a better guide that compensates for "over-fitting" the data, an adjusted $R^2$ ($R_a^2$) is computed as:

$$R_a^2 = R^2 - \frac{(1 - R^2)(K - 1)}{(N - K)}$$

---

4. There are two estimates for $s^2$. One is typically furnished with OLS regression results and is unbiased: $s^2 = \frac{e'e}{N-K}$ or, the sum of squared residuals divided by the degrees of freedom in the regression (the difference between the number of observations and the number of regression coefficients). The other estimate is not unbiased, but is the basis of many tests that use the maximum likelihood framework for estimation (see Chapters 27 and 29 on the spatial lag and spatial error models). It consists of the sum of squared residuals divided by the number of observations, $s^2_{ML} = \frac{e'e}{N}$. I will refer to this estimator as the ML estimate of the error variance.

5. The residual sum of squares is included in the SpaceStat regression output.

6. When the regression does not include a constant term, the residuals no longer have a mean of zero. As a result, the mean of the dependent variable $y$, and the predicted values, $Xb$, are no longer the same. This must be taken into account in the computation of $R^2$. SpaceStat makes the correct adjustment for this case.
This adjusted $R^2$ does not necessarily increase when additional variables are added to the model. When it does not, this means that the additional variable(s) do not contribute sufficiently to the model fit to warrant the loss of degrees of freedom from their inclusion.

Both the standard $R^2$ and the adjusted $R^2$ are reported by SpaceStat.

An alternative set of measures of fit computed in SpaceStat is based on the maximum likelihood (ML) approach to estimation. For the standard regression model, these are not really needed, since the OLS estimates are equivalent to the maximum likelihood estimates. However, if you wish to compare the fit of the standard regression to that of one of the spatial regression models, you can no longer rely on the $R^2$ and must use the ML based measures.7

Maximum likelihood estimation is based on the concept of a joint density or distribution function for the observed data, $y$, which is referred to as the likelihood function.8 This likelihood function is expressed in function of the explanatory variables $X$ and a set of parameters: $\beta$ and $\sigma^2$. The ML estimates for the parameters are those values that obtain the highest probability or joint likelihood. In order to implement ML estimation, you need to assume a form for the joint distribution. Typically, this will be the normal distribution. The logarithm of the likelihood that is obtained for the OLS estimates of the regression coefficients is reported by SpaceStat as an alternative to the $R^2$ measure of fit.9 The model with the highest log likelihood is the one that achieves the best fit. As is the case with the $R^2$, the log likelihood always increases when additional variables are included in the model, and thus may not be a good indicator of how well the model "explains" the data.

In order to correct the log likelihood for overfitting, a number of so-called information criteria (IC) have been proposed. SpaceStat includes two of these, the Akaike Information Criterion (AIC) and the Schwartz Criterion (SC).10 In general terms, an information criterion is of the form:

$$IC = -2L + f(K,N)$$

where $L$ is the maximized log likelihood and $f(K,N)$ is a function of the number of variables ($K$) and the number of observations ($N$), i.e., of the degrees of freedom in the model.11 The overall principle behind the IC is that the assessment of fit is penalized by a function of the degrees of freedom. The best model is the one with the lowest value for an information criterion.

---

7. For a more detailed technical discussion of model validation in spatial regression models, see Anselin (1988a).
8. Strictly speaking, the density function is developed for the error terms, $e$, since they are intrinsically random, and thus have a distribution.
9. For the assumption of a normal error term, the logarithm for the likelihood is estimated as:

$$L = -(N/2) \ln 2\pi - (N/2) \ln s^2 - 0.5 e' e / s^2$$

with $e$ as the OLS residuals ($e = y - Xb$) and $s^2$ as the ML estimate for the error variance, $s^2_{ML} = e' e / N$. Note that some regression packages do not include the simple scaling factor $(N/2) \ln 2\pi$ in the log likelihood they report, but SpaceStat does.
10. For a technical discussion of the nature of information criteria, see Akaike (1981), and also Anselin (1988a).
11. For the Akaike Information Criterion (AIC), this function corresponds to $f(K,N) = 2K$, and for the Schwartz Criterion (SC) it is $f(K,N) = K \ln(N)$. 
Note that each correction factor corresponds to a different objective function, and that the best model according to one is not necessarily best according to others.

**Hypothesis Tests**

When your real interest is in statistical inference, you are not so much concerned with the particular values of the regression coefficient estimates \( b \), but with the way in which you can use them to draw conclusions about the unknown population parameters \( \beta \). In order to get to this point, you need to make an assumption about the distribution of the random error term \( \varepsilon \), as argued above. You also need an estimate for the variance of the regression coefficients \( \beta \). The variance for an individual regression coefficient is the corresponding diagonal element in the covariance matrix for all coefficients. The latter is estimated as \( s^2(X'X)^{-1} \), where \( s^2 \) is the unbiased estimator for the error variance, \( e'e/(N-K) \), in the notation used above.\(^{12}\)

You typically are interested in finding out whether the population coefficient is different from zero, or, in other words, whether the associated variable contributes to the regression equation. Formally, this is a hypothesis test on the null hypothesis that the population regression coefficient in question (say \( \beta_h \)) is zero:

\[
H_0: \; \beta_h = 0
\]

If you are able to reject this null hypothesis, you may conclude that the population coefficient must be non-zero, within the limits given to you by the chosen significance level (or Type I error).\(^{13}\)

The significance of individual regression coefficients can be tested by means of a t-test. Under the assumption of normal error terms, the statistic

\[
t_h = \frac{b_h}{SE(b_h)}
\]

i.e., the regression estimate divided by its standard deviation (the square root of its variance), follows a Student t distribution with \( N-K \) degrees of freedom. **SpaceStat** reports the standard deviation, t-statistic and associated probability for each regression coefficient. If the probability of the t-statistic is below a chosen critical level (chosen by you, that is), you may conclude that the null hypothesis of \( \beta_h = 0 \) is unlikely, and thus that the population coefficient is likely to be non-zero (or significant).

You may also be interested in the significance of the regression specification as a whole, or, in other words, whether the slope coefficients in the population regression (i.e., all but the constant term) are jointly non-zero. This results in a set of \( K-1 \) null hypotheses of the form given above that need to be satisfied together. A test for these joint hypotheses is the so-called F-test, which is similar to the approach taken in analysis of variance (ANOVA). The

---

\(^{12}\) **SpaceStat** provides the complete covariance matrix as part of the regression output when the **Long Output** option is set to YES in the **Options** menu.

\(^{13}\) The concept of hypothesis testing is treated extensively in most statistics and econometrics texts, and I will not dwell on it here.
F-test statistic can be expressed in terms of the constrained and unconstrained residual sum of squares (RSS, i.e., sum of squared residuals):

\[
F = \frac{(RSS_C - RSS_U)/(K-1)}{RSS_U/(N-K)}
\]

The constrained residual sum of squares (RSS_C) is found from a regression in which the null hypothesis holds, i.e., a regression without slope coefficients and only a constant term. The RSS_C is thus nothing but the sum of squares of the dependent variable (in deviations from its mean). The unconstrained sum of squares (RSS_U) is the residual sum of squares in the regression, as reported by SpaceStat. The F test statistic is distributed as a F variate with K-1,N-K degrees of freedom. The F statistic and its associated probability level are reported in the standard regression output of SpaceStat. Note that only in unusual circumstances this statistic will not be significant. Typically this will be the case for very poor model specifications, as indicated by a low R^2 or even negative R^2. 14

Robust Inference

In many instances in spatial data analysis, the assumptions of homoskedastic and uncorrelated error terms may not be very realistic. Heteroskedasticity in particular is often recognized to be a problem for cross-sectional data. A wide range of diagnostics for the presence of heteroskedasticity have been suggested (see 26.2.2). Upon finding evidence of this form of misspecification, you may estimate a model that explicitly takes it into account. Such models are discussed in Chapter 30. Alternatively, you may implement a robust inference based on the OLS estimates. Indeed, the OLS estimates remain unbiased, even in the presence of heteroskedasticity, although their variance takes on a more complex form:

\[
V[b] = \sigma^2 (X'X)^{-1}X'\Omega X(X'X)^{-1}
\]

where \(\Omega\) is the error variance matrix, scaled such that its trace equals N. An important result due to White (1980) shows that while \(\Omega\) itself is not estimable unless further assumptions are made (as in the models of Chapter 30), the expression \(\sigma^2 X'\Omega X\) may be consistently estimated by \(X'SX\), where S is a diagonal matrix of squared regression residuals.

A consistent estimate for the variance of the OLS regression coefficients in the presence of heteroskedasticity can thus be computed as:

\[
V[b] = (X'X)^{-1}X'SX(X'X)^{-1}
\]

14. In the standard regression model, there is a direct relation between the R^2 and the F test.
where \( S \) is as defined above. This estimator has been shown to perform rather poorly in finite sample situations, and an improved version was suggested by MacKinnon and White (1985). This so-called \emph{adjusted White variance} consists of dividing the squared residuals in the diagonal matrix \( S \) by the correction factor \( 1 - k_{ii} \), where \( k_{ii} \) is the \( i \)-th diagonal element of the idempotent matrix \( X(X'X)^{-1}X' \). \textit{SpaceStat} gives the standard deviations and asymptotic \( z \)-values computed on the basis of the adjusted White variance in the output of the \textbf{OLS-Robust} command.

An alternative approach to estimating a robust covariance matrix for the OLS estimates in the presence of heteroskedasticity may be based on the Jackknife (Efron, 1982). The Jackknife is a resampling approach where each observation in turn is dropped from the data set. The empirical distribution of the OLS estimates obtained for all \( M \) replications provides the basis for a Jackknife estimate of variance:

\[
V[b] = \frac{1}{M} \sum_{i=1}^{M} \{ b(i) - \frac{1}{M} \sum_{j=1}^{M} b(j) \} \{ b(i) - \frac{1}{M} \sum_{j=1}^{M} b(j) \}',
\]

where \( M \) is the number of replications and \( b(i) \) is the OLS estimate obtained in a dataset from which the \( i \)-th observation has been dropped.\(^{15}\) The Jackknife results are also given in the output for the \textbf{OLS-Robust} command.

\[\text{2.2.2 Specification Diagnostics}\]

Obviously, the assumptions of normal, homoskedastic and uncorrelated error terms that lead to the BLUE characteristic of OLS estimates are not necessarily satisfied by the real models you use with real data. An important part of good econometric practice consists of checking for the extent to which these assumptions may be violated. When dealing with spatial data, you must give special attention to the possibility that the errors or the variables in the model show spatial dependence. To date, tests for spatial dependence in the regression specification are absent in all commercial statistical and econometric software packages.\(^{16}\) Therefore, one of the goals of \textit{SpaceStat} is to provide you with a range of diagnostics to detect spatial dependence, in addition to many of the standard diagnostics that are included in most commercial software.

\(^{15}\) MacKinnon and White (1985) have shown that this variance matrix may be computed directly as:

\[
V[b] = \frac{1}{N-1} \left[ \frac{X'X}{N} - \left( \frac{1}{N} \right) (Xe^*e^*X) \right] (X'X)^{-1}
\]

where \( \Omega^* \) is a diagonal matrix with as elements the adjusted squared residuals, \( [e_i/(1-k_{ii})]^2 \), \( k_{ii} \) is as before, and \( e^* \) is a vector with the square root of the diagonal elements of \( \Omega^* \). See also Anselin (1990a), for a further discussion of robust approaches in spatial regression analysis.

\(^{16}\) See Anselin and Hudak (1992) for a discussion of how these diagnostics may be implemented with standard software.
Multicollinearity

A problem that you may often encounter in empirical work is the high correlation between the observations for the explanatory variables included in the regression specification. In principle, these explanatory variables should be uncorrelated. If there is an exact linear relationship between some of the variables (i.e., a correlation of 1), OLS estimation will break down. This situation is referred to as perfect multicollinearity.\textsuperscript{17} Often however, the extent of the multicollinearity is not ”perfect,” but appears in the form of a strong linear relation (i.e., correlation) between the explanatory variables. As a consequence, OLS estimation will not break down, but the estimates will have very large estimated variances. Therefore, very few coefficients will be found to be significant, even though the regression as a whole may seem to achieve a reasonable fit. This combination of high $R^2$ with very low $t$ statistics is often a good indicator that something is wrong in terms of multicollinearity. Another telltale sign is when the estimates vary considerably as a result of adding or dropping a single observation.

There are no tests in the strict sense for multicollinearity. However, there are a number of diagnostics that may point to a potential problem. One of these, included in the \texttt{SpaceStat} regression output, is the so-called condition number, popularized by the work of Belsley et al. (1980).\textsuperscript{18} As a rule of thumb, values of the condition number larger than 20 or 30 are considered to be suspect. A total lack of multicollinearity yields a condition number of 1.

Non-Normal Errors

Most hypothesis tests and a large number of regression diagnostics are based on the assumption of a normal error distribution. It is hard to assess the extent to which this may be violated, since the errors themselves (i.e., the random error term in the population regression model) cannot be observed. Instead, tests for non-normal errors must be computed from the regression residuals. The \texttt{SpaceStat} regression output includes the results of a test suggested by Kiefer and Salmon (1983). This is an asymptotic test, so it may not be very reliable for small data sets. The statistic follows a $\chi^2$ distribution with 2 degrees of freedom.\textsuperscript{19} \texttt{SpaceStat} reports the statistic as well as its associated probability. A low probability indicates a rejection of the null hypothesis of a normal error. If this is the case, the tests for heteroskedasticity and spatial dependence should be interpreted with caution, since they are based on the normal

\textsuperscript{17} Perfect multicollinearity results in the singularity of the matrix $XX$. As a consequence, its inverse $(XX)^{-1}$ does not exist, which precludes the OLS estimates from being computed. \texttt{SpaceStat} traps this situation and issues an error message when perfect multicollinearity is detected.

\textsuperscript{18} The condition number is the square root of the ratio of the largest to the smallest eigenvalue of the matrix $XX$, after standardization.

\textsuperscript{19} The Kiefer-Salmon test is only one of many tests for normality. It is based on both skewness and kurtosis of the residuals. Its formal expression is:

\[ W = N \left( \frac{b12}{6} + \frac{b23}{24} \right) \]

where $N$ is the number of observations and $b1$ and $b2$ are related to the third and fourth moments. See Kiefer and Salmon (1983) for further details.
assumption. In many cases, a simple transformation of the dependent variable, such as a logarithm (e.g., by means of the command `D-5-1`), may induce normality.

**Heteroskedasticity**

Heteroskedasticity is the situation where the random regression error does not have a constant variance over all observations (i.e., is not homoskedastic). As a consequence, the indication of precision given by assuming a constant error variance in OLS will be misleading. While the OLS estimates are still unbiased, they will no longer be most efficient. More importantly, inference based on the usual t and F statistics will be misleading, and the R² measure of goodness-of-fit will be wrong. In spatial data analysis, you will frequently encounter this problem, especially when using data for irregular spatial units (with different area), when there are systematic regional differences in the relationships you model (i.e., spatial regimes), or when there is a continuous spatial drift in the parameters of the model (i.e., spatial expansion). The presence of any of these spatial effects would make a standard regression model that ignores them misspecified. Hence, an indication of heteroskedasticity may point to the need for a more explicit incorporation of spatial effects, in the form of spatial regimes or spatial expansion of the parameters.

There are many tests against heteroskedasticity available in the literature, and SpaceStat only includes a few. All tests start from the null hypothesis of homoskedasticity:

\[ H_0: E[\varepsilon_i^2] = \sigma^2 \]

The alternative hypothesis is that each observation’s error term has a different variance, \( \sigma_i^2 \). In many instances, this is too general to be very useful. The degree of specificity with which the alternative hypothesis for heteroskedasticity can be expressed will depend on your knowledge of the factors that may cause it. Note that if you are fairly confident that heteroskedasticity will be present and if you have a good idea of what may cause it (e.g., from theoretical considerations), you should not estimate a standard regression model. Instead, you should explicitly include the heteroskedastic error in the specification of the model, by means of one of the Regress-Heterosked Error commands (see Chapter 30). Alternatively, you may implement an estimator that is robust to the presence of heteroskedasticity, such as in the OLS-Robust command described above.

A common approach towards specifying the alternative hypothesis is to relate the variability in the error variance to a number of variables, via a functional form that includes a few parameters (say P parameters), as in:

---

20. For extensive overviews, see any recent econometrics text, such as Judge et al (1985) or Greene (1990).
where \( \sigma_i^2 \) is a simple scale factor, \( f \) is a functional form, \( \alpha_0 \) the \( \alpha_p \) are parameters and the \( z_{pi} \) are \( P \) variables for observation \( i \). Commonly used functional forms are the linear one (for so-called additive heteroskedasticity) and the exponential one (for so-called multiplicative heteroskedasticity). Only additive heteroskedasticity is implemented in \textit{SpaceStat}.

The \( z \)-variables included in the heteroskedastic specification may be any relevant variable. Often, area of the spatial unit, or any other variable that relates to its size (total population, total income) are good choices. In \textit{SpaceStat}, the default specification pertains to the model with random coefficient variation. This is the situation where the regression coefficients are not fixed (as in the standard model), but they vary randomly around a mean, with a given variance. In this case, the \( z \) variables in the heteroskedastic specification become the squares of the explanatory variables in the regression model (i.e., the \( x_{ih}^2 \) and the corresponding coefficients (i.e., the \( \alpha_p \)) are the coefficient variance. If the default specification is not chosen, the \( z \) variables must be specified explicitly in the \textit{Problem File}.

Three tests against heteroskedasticity are implemented in \textit{SpaceStat}, but only two of those are reported at any time. The first reported test is either the Lagrange Multiplier test developed by Breusch and Pagan (1979), or its studentized version suggested by Koenker (1981) and Koenker and Bassett (1982). Which of the two is reported, depends on the outcome of the normality test. When the errors are non-normal, the Breusch-Pagan (BP) test has been shown to achieve poor power in small samples. Hence, when the Kiefer-Salmon normality test fails (for an \( \alpha \) level of 0.01), \textit{SpaceStat} does not report the results for the BP test, but uses the Koenker-Bassett (KB) test instead. Both tests are asymptotic and achieve a \( \chi^2 \) distribution with \( P \) degrees of freedom (where \( P \) is the number of \( z \) variables in the heteroskedastic specification).\(^{21}\)

Both BP and KB tests require that you specify the variables to be used in the heteroskedastic specification. A different approach is needed for the situation where there is very little prior information about the form of the heteroskedasticity. In such instances, a test developed by White (1980) is more appropriate, since it has power against any unspecified form of heteroskedasticity. This test is also asymptotic and achieves a \( \chi^2 \) distribution.\(^{22}\) When there

\[ \sigma_i^2 = \sigma^2 f(\alpha_0 + \Sigma_{pi} z_{pi} \alpha_p) \]

---

\(^{21}\) The BP test is equivalent to one half the explained sum of squares in a regression of \( (e_i^2/S_{ML} - 1) \) on a constant and the \( z \) variables. The KB test is a studentized version of this, in that the \( S_{ML} \) is replaced by a more robust estimate of the fourth moment. For details, see the references given in the main body of the text.

\(^{22}\) The White test consists of \( N \) times the \( R^2 \) in an auxiliary regression of the squared OLS residuals on all cross products between the explanatory variables. In some instances, the squares or cross-products of variables are already included as explanatory variables in the original regression. For example, this will be the case for a trend surface specification. In that situation, only the unique cross products should be included to avoid perfect multicollinearity. \textit{SpaceStat} uses a fuzzy comparison procedure to avoid selecting the same variable twice. The number of variables included in the auxiliary regression determines the degrees of freedom for the test. For more details, see White (1980).
are sufficient degrees of freedom to carry out a meaningful regression (i.e., leaving at least 15
degrees of freedom), \textit{SpaceStat} reports the results of the White statistic in addition to either
BP or KB.

For the three tests, \textit{SpaceStat} gives the value of the test statistic, its degrees of freedom
and the associated probability level. One issue to keep in mind in situations where both het-
eroskedasticity and spatial dependence may be present is that the tests against heteroskedasticity
have been shown to be very sensitive to the presence of spatial dependence.\textsuperscript{23} In other words,
while the tests may indicate heteroskedasticity, this may not be the problem, but instead spatial
dependence may be present. As I will outline in the next section, the reverse holds as well.

\textit{Spatial Autocorrelation}

Spatial autocorrelation, or more generally, spatial dependence, is the situation where
the dependent variable or error term at each location is correlated with observations on the
dependent variable or values for the error term at other locations. The general case is formally:

\[ E[y_i y_j] \neq 0 \]

or

\[ E[\epsilon_i \epsilon_j] \neq 0 \]

for neighboring locations i and j. This specification is too general to allow for the estimation
of potentially \( N \times (N-1) \) interactions from \( N \) observations. Therefore, the form of the spatial
dependence is given structure by means of a spatial weights matrix (\( W \)), which reduces the
number of unknown parameters to one, i.e., the coefficient of spatial association in a spatial
autoregressive or spatial moving average process.

The consequences of ignoring spatial autocorrelation in a regression model, when it
is in fact present, depend on the form for the alternative hypothesis. As in all tests for mis-
specification, the null hypothesis reflects the absence of misspecification, or, in this case, the
standard regression model with homoskedastic and uncorrelated errors. There are two important
alternative models. In one, the ignored spatial autocorrelation pertains to the dependent variable,
y. I will refer to this case as the \textbf{spatial lag} case, or, as substantive spatial dependence (sub-
stantive since it pertains to the dependent variable \( y \)). This alternative is formalized in a mixed
regressive, spatial autoregressive model:

\[ y = \rho Wy + X\beta + \epsilon \]

\textsuperscript{22} For extensive simulation results, see Anselin (1990b) and Anselin and Griffith (1988).
where $Wy$ is a spatially lagged dependent variable, and $\rho$ is the spatial autoregressive coefficient. The null hypothesis of no autocorrelation corresponds to:

$$H_0: \rho = 0$$

If this form of spatial autocorrelation is ignored, the OLS estimates will be biased and all inference based on the standard regression model (i.e., the model without the $Wy$ term) will be incorrect. In a sense, this is similar to the consequences of omitting a significant explanatory variable in the regression model.

The second form of spatial autocorrelation in a regression model pertains to the error term. I will refer to this case as the **spatial error** case, or, to spatial dependence as a nuisance (a nuisance since it only pertains to the errors). Formally, this dependence is expressed by means of a spatial process for the error terms, either of an autoregressive or a moving average form. Such an autoregressive process can be expressed as:

$$y = X\beta + \varepsilon$$

$$\varepsilon = \lambda W\varepsilon + \xi$$

with $W\varepsilon$ as a spatially lagged error term, $\lambda$ as the autoregressive coefficient and $\xi$ as a well-behaved (i.e., homoskedastic and uncorrelated) error term. A spatial moving average process in the error term $\varepsilon$ takes the form:

$$\varepsilon = \lambda W\xi + \bar{\xi}$$

where now the spatial lag pertains to the errors $\xi$ and not to the original $\varepsilon$. In both cases, the null hypothesis is of the form:

$$H_0: \lambda = 0$$

or, the error term is uncorrelated. It turns out that the tests for the presence of either form of spatial error dependence are the same. The consequences of ignoring spatial error dependence are the same as for heteroskedasticity: the OLS estimator remains unbiased, but is no longer efficient, since it ignores the correlation between error terms. As a result, inference based on $t$ and $F$ statistics will be misleading and indications of fit based on $R^2$ will be incorrect.

**SpaceStat** contains four tests for spatial dependence, three of which pertain to the spatial error case and one to the spatial lag case. The first test is an extension of Moran’s I to measure spatial autocorrelation in regression residuals. This test is formally equivalent to the one outlined...
in Chapter 22. As detailed there, inference is based on a standardized $z$-value that follows a normal distribution (asymptotically). The expressions for the theoretical mean and standard deviation for the statistic that are needed to obtain this standardized $z$-value are more complex in the case of regression residuals, but otherwise the interpretation of the statistic is the same as for the general case. **SpaceStat** reports the value for the $I$ statistic, the corresponding $z$-value and associated probability (according to a standard normal distribution).

Even though Moran’s $I$ for regression residuals is by far the most familiar among the four procedures included in **SpaceStat**, it is actually a fairly unreliable test. As shown in a large number of Monte Carlo simulation experiments in Anselin and Rey (1991), this statistic picks up a range of misspecification errors, such as non-normality and heteroskedasticity, as well as spatial lag dependence. Moreover, it does not provide any guidance in terms of which of the substantive or error dependence is the most likely alternative.

The second test listed in the **SpaceStat** regression output is a Lagrange Multiplier test, originally suggested by Burridge (1980). It is an asymptotic test, which follows a $\chi^2$ distribution with one degree of freedom. The test is the same for an alternative hypothesis of spatial autoregressive and that of spatial moving average errors. **SpaceStat** reports the statistic, its degrees of freedom (always one) and the corresponding probability.

The third test is a recently developed specification robust procedure by Kelejian and Robinson (1992). In contrast to the Moran’s $I$ and Lagrange Multiplier tests, this test does not require normality for the error terms. It also is applicable to both linear and nonlinear regressions and requires less information about the exact form of the spatial weights matrix. **SpaceStat** reports the value for the statistic, its degrees of freedom (always one) and the corresponding probability.

Formally Moran’s $I$ test is $NS_0/\sum_i \sum_j w_{ij} e_i e_j / \sum_i e_i^2$, with $N$ as the number of observations, $S_0$ as the sum of all spatial weights, and $e_i$ as the regression residuals. For a detailed derivation of the moments and other technical details, see Cliff and Ord (1972, 1981).

25. The formal expression for this test is:

$$LM_{LM} = (e'Ve/s^2)^2/\text{tr}[WW']$$

where $\text{tr}$ stands for the matrix trace operator, $e$ is a vector of OLS residuals, $s^2 = e'e/N$ is the ML estimate for error variance and $W$ is the spatial weights matrix. For further details, see Burridge (1980) and Anselin (1988b).

26. The Kelejian-Robinson statistic is obtained from an auxiliary regression of cross products of residuals and cross products of the explanatory variables (collected in a matrix $Z$ with $P$ columns). The cross products are for all pairs of observations for which a nonzero correlation is postulated (but each pair is only entered once), for a total of $h N^2$ pairs. Using $\gamma$ for the coefficient vector in this auxiliary regression, and $\alpha$ for the resulting residual vector, the Kelejian-Robinson statistic is:

$$KR = (\gamma'Z\gamma)/(\alpha'\alpha/h)$$

The statistic is a large sample test and follows a $\chi^2$ distribution with $P$ degrees of freedom. For technical details, see Kelejian and Robinson (1992).
The final test is a Lagrange Multiplier diagnostic for a spatial lag, suggested in Anselin (1988b). This test is only valid under the assumption of normality and is asymptotic in nature. As is its counterpart for spatial errors, the LM lag test is distributed as a $\chi^2$ variate with one degree of freedom. SpaceStat reports the statistic, its degrees of freedom (always one) and the associated probability.

As evidenced in a large number of Monte Carlo simulation experiments in Anselin and Rey (1991), the joint use of the LM_ERR and LM_LAG statistics provides the best guidance with respect to the alternative model, as long as the assumption of normality is satisfied. Each test has the highest power against the case for which it is designed, even though it also has power against the other alternative. In other words, when both tests have high values (indicating significant spatial dependence), the one with the highest value (or lowest probability) will tend to indicate the correct alternative.

26.3 Example Problem File

As an example, you will carry out a regression of the CRIME variable in the Columbus data set on a constant term and the explanatory variables INCOME and HOUSING. Make sure that the data set (the files COL.DAT and COL.DHT) is on your current directory. In addition, the problem file COL7.BTC and the weights matrices COLWS_1.FMT, COLWS_2.FMT, and COLWS_3.FMT should be on the current directory as well. The Problem File COL7.BTC, created in Chapter 25 contains all the information to carry out this example. It is contained on the \EXAMPLES directory and also listed below, as Table 26.1.

You start the regression analysis by means of the R-1-1 command sequence. In order to carry out robust regression, you may enter the command sequence R-1-2 instead. Enter col7.btc as the name for the problem file (followed by Return).

Two messages indicate that the program has started:

Reading in data ...
Starting analysis ...

Next follow the results, one screen at a time. These results will also be written to the file you specified in the Output File option.

---

27. The formal expression for this test is:

$$LMLAG = \frac{(e'Wy / s^2)^2}{(WXb)'MWXb / s^2 + \text{tr}[W'W+W^2]}$$

where $\text{tr}$ stands for the matrix trace operator, $M=I-X(X'X)^{-1}X'$, $y$ is a $N$ by 1 vector of observations on the dependent variable, $e$ is a vector of OLS residuals, $W$ is the spatial weights matrix, $s^2$ the ML estimate of error variance and $b$ is a $K$ by 1 vector with the OLS coefficient estimates. For further details, see Anselin (1988b).
26.4 Program Output

26.4.1 Standard Output

The first screen of results contains the estimates, their standard errors, t-statistics and associated probabilities, as shown in Table 26.3. All coefficients are highly significant and have the expected signs. The header for this output includes information on the data set (COL), dependent variable (CRIME), observations (49), variables (3) and degrees of freedom (46), as well as several indications for goodness-of-fit: $R^2$ (0.55), adjusted $R^2$ (0.53), maximized log likelihood (LIK, 187.4), Akaike Information Criterion (AIC, 380.8), and Schwartz Criterion (SC, 386.4). Next follow the residual sum of squares (RSS, 6014.9), and the F test on the joint significance of slope coefficients. The latter takes on a value of 28.39, which, for an F distribution with 2,46 degrees of freedom is highly significant. This is to be expected, since both slope coefficients (INCOME and HOUSING) are highly significant by themselves at probability levels of less than 0.01. The last row of the header lists the unbiased and maximum likelihood estimates for the error variance as well as the respective standard deviations. You can easily verify the various relations between these statistics. For example, $\text{SIG-SQ}$ times the DF (130.8x46) yields the RSS.

Next follow three screens with regression diagnostics. You move from screen to screen by pressing the Return key. The first set of diagnostics pertains to the multicollinearity condition number and the Kiefer-Salmon test for normality of the residuals, as shown in Table 26.4. The condition number of 6.54 is well below the acceptable limit of 20, so that no problems are expected from this source. In addition, the Kiefer-Salmon test takes on a low value of 1.84, which, for a $\chi^2$ variate with 2 degrees of freedom is far from significant (p=0.40). Consequently, you can safely interpret the results of the various misspecification tests that depend on the normality assumption, such as the various Lagrange Multiplier tests.

The next screen deals with heteroskedasticity, as illustrated in Table 26.5. The second line indicates that the tests were carried out for the default case of random coefficient variation. If you had specified heteroskedastic variables in the problem file (or in response to the interactive queries of the interactive mode), their names would be listed instead. Two test results are
reported. Since no problems were revealed with respect to a lack of normality, the Breusch-Pagan statistic is given. If the normality had been rejected (with p < 0.01), the Koenker-Bassett test would have been used instead. Since there were 2 explanatory variables (not counting the constant term), whose squares will be used in the test, the BP statistic has 2 degrees of freedom. Its value of 7.90 is fairly significant (p < 0.02), indicating a potential problem. This is confirmed by the robust White statistics, which at 19.95, for 5 degrees of freedom (the original variables, their squares and cross product) is significant at p << 0.01.

Table 26.3 Ordinary Least Squares Regression Estimates

<table>
<thead>
<tr>
<th>ORDINARY LEAST SQUARES ESTIMATION</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA SET</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEPENDENT VARIABLE</td>
<td>CRIME</td>
<td>OBS 49</td>
<td></td>
<td>VARS 3</td>
<td></td>
</tr>
<tr>
<td>R2</td>
<td>0.5524</td>
<td></td>
<td>R2 adj</td>
<td>0.5329</td>
<td></td>
</tr>
<tr>
<td>LIK</td>
<td>-187.377</td>
<td></td>
<td>AIC</td>
<td>380.754</td>
<td></td>
</tr>
<tr>
<td>RSS</td>
<td>6014.86</td>
<td></td>
<td>F-test</td>
<td>28.3856</td>
<td>Prob</td>
</tr>
<tr>
<td>SIG-SQ</td>
<td>130.758</td>
<td></td>
<td>(11.4349)</td>
<td>SIG-SQ(ML)</td>
<td></td>
</tr>
<tr>
<td>VARIABLE</td>
<td>COEFF</td>
<td>S.D.</td>
<td>t-value</td>
<td>Prob</td>
<td></td>
</tr>
<tr>
<td>CONSTANT</td>
<td>68.6189</td>
<td>4.73547</td>
<td>14.490397</td>
<td>0.000000</td>
<td></td>
</tr>
<tr>
<td>INCOME</td>
<td>-1.5973</td>
<td>0.33413</td>
<td>-4.780490</td>
<td>0.000012</td>
<td></td>
</tr>
<tr>
<td>HOUSING</td>
<td>-0.273931</td>
<td>0.103198</td>
<td>-2.654414</td>
<td>0.010868</td>
<td></td>
</tr>
</tbody>
</table>

Table 26.4 Diagnostics for Multicollinearity and Normality

<table>
<thead>
<tr>
<th>REGRESSION DIAGNOSTICS</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MULTICOLLINEARITY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CONDITION NUMBER</td>
<td>6.541828</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TEST ON NORMALITY OF ERRORS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TEST</td>
<td>DF</td>
<td>VALUE</td>
<td>PROB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kiefer-Salmon</td>
<td>2</td>
<td>1.836070</td>
<td>0.399303</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 26.5 Tests against Heteroskedasticity

<table>
<thead>
<tr>
<th>DIAGNOSTICS FOR HETEROSKEDASTICITY</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RANDOM COEFFICIENTS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TEST</td>
<td>DF</td>
<td>VALUE</td>
<td>PROB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Breusch-Pagan test</td>
<td>2</td>
<td>7.900764</td>
<td>0.019247</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPECIFICATION ROBUST TEST</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TEST</td>
<td>DF</td>
<td>VALUE</td>
<td>PROB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>White</td>
<td>5</td>
<td>19.946087</td>
<td>0.001279</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 26.6 Tests for Spatial Dependence

<table>
<thead>
<tr>
<th>DIAGNOSTICS FOR SPATIAL DEPENDENCE</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>FOR WEIGHTS MATRIX COLWS_1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TEST</td>
<td>MI/DF</td>
<td>VALUE</td>
<td>PROB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Moran's I (error)</td>
<td>0.235636</td>
<td>2.954559</td>
<td>0.003131</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lagrange Multiplier (error)</td>
<td>1</td>
<td>5.723012</td>
<td>0.016744</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kelejian-Robinson (error)</td>
<td>3</td>
<td>11.550185</td>
<td>0.009094</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lagrange Multiplier (lag)</td>
<td>1</td>
<td>9.363396</td>
<td>0.002214</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The third screen of diagnostics pertains to spatial dependence. Three sets of results are listed, one for each weights matrix, of which the first only is shown in Table 26.6. The header gives the name for the weights matrix used and lists whether or not it was row-standardized. All tests are highly significant, illustrating the difficulty to conclude which is the proper alternative: the spatial error model or the spatial lag model. Moran’s I of 0.24 yields a standardized z-value of 2.95, which is very significant (p=0.003). The robust Kelejian-Robinson test, distributed as $\chi^2$ with 3 degrees of freedom, is at 11.55 also highly significant (p=0.009). The LMERR statistic is slightly less so, at 5.72 for a $\chi^2$ variate with 1 degree of freedom (p=0.017). When compared to the LMLAG statistic of 9.36, which corresponds to a probability of 0.002, it becomes clear that the spatial error tests are picking up the omitted spatial lag. At any rate, there is strong evidence of the presence of spatial dependence. It remains to be seen however, to what extent this result may be influenced by potential heteroskedasticity, or vice versa, whether the indication given by the BP and White tests is really due to spatial dependence. I will return to this model validation issue in the chapters on estimating spatial lag and spatial error models.

26.4.2 Long Output

If you had left the Long Output option to its default value of NO, the results shown in Table 26.3 to Table 26.6 are the extent of the output provided for the OLS regression estimates. If you now press the Return key, you will be back in the Regress Classic menu. However, if you had the Long Output option set to YES, there are two additional items listed.

The first item, immediately following the screen with the estimates and measures of fit, lists the complete covariance matrix for the coefficient estimates, as shown in Table 26.7. You can use this matrix to carry out hypothesis tests on linear and non-linear combination of coefficients. The matrix is organized such that the variable corresponding to each row is listed above the row.

Table 26.7 OLS Coefficient Variance Matrix

<table>
<thead>
<tr>
<th>COEFFICIENT VARIANCE MATRIX</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONSTANT</td>
</tr>
<tr>
<td>INCOME</td>
</tr>
<tr>
<td>HOUSING</td>
</tr>
</tbody>
</table>
The second item generated by the Long Output option is listed on your screen, but it does not appear in the Output File specified in the options. Instead it is included in the Report File listed in the Problem File, provided that you have the Output to a File option set to YES. As you can see from Table 26.1, the filename for this Report File in the problem file COL7.BTC is REP, which will create a file with filename REP.DOC on your current directory. As illustrated in Table 26.8, this second set of results consists of the observation sequence number, observed value for CRIME, predicted value and residual. If you have set the Indicator Variable option to YES and specified a variable other than OBS, the proper indicator variable and its corresponding values will be listed instead of the simple sequence numbers.

No other information besides a variable header and the actual values are contained in the report file, which facilitates its use as an input (or import) file into a graphics or mapping package. This may give a more intuitive and visual insight into any spatial patterns that may be present. For example, in Appendix I, a mapping of the regression residuals is given, using an ARC/INFO coverage for the Columbus neighborhoods.

26.4.3 Robust Inference

When you run the OLS-Robust command instead of the generic OLS estimation, the output is slightly different. As pointed out earlier, the estimates are the same, but the inference differs. The output for this command is listed in Table 26.9. There are two parts to the table: one pertains to the standard errors computed for the adjusted White variance, the other to the Jackknife estimates. The estimates and measures of fit are identical to the ones given in Table 26.3. The main difference with the standard OLS results is that the coefficient for HOUSING is no longer significant. For the adjusted White variance, its standard deviation is 0.18, which leads to an asymptotic z-value of -1.54, while for the Jackknife variance, the standard deviation is 0.20, with a z-value of -1.38.

If you set the Long Output option to YES, the full covariance matrix is listed as well, as illustrated in Table 26.10 for the adjusted White variance and in Table 26.11 for the Jackknife variance.

26.4.4 IDRISI Interface

When you set the IDRISI Interface option to YES, and irrespective of the setting of the Output to a File (Option 2) or Long Output (Option 3) options, the predicted values and residuals are not written to the report file specified in the Problem File. Instead, two special files are created, with as file name the name of the dependent variable, followed by _YP.VAL for the predicted values and _E.VAL for the residuals. For example, in the regression for the crime
data, these two files would be CRIME_YP.VAL and CRIME_E.VAL. These files are in the format that facilitates converting the data into an IDRISI image for display.

Table 26.8 OLS Predicted Values and Residuals

<table>
<thead>
<tr>
<th>OBS</th>
<th>CRIME</th>
<th>PREDICTED</th>
<th>RESIDUAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>18.802</td>
<td>22.4966</td>
<td>-3.69462</td>
</tr>
<tr>
<td>2</td>
<td>32.388</td>
<td>52.3732</td>
<td>-19.9852</td>
</tr>
<tr>
<td>3</td>
<td>38.426</td>
<td>40.3405</td>
<td>-1.91453</td>
</tr>
<tr>
<td>4</td>
<td>0.178</td>
<td>34.596</td>
<td>-34.418</td>
</tr>
<tr>
<td>5</td>
<td>15.726</td>
<td>15.3795</td>
<td>0.346501</td>
</tr>
<tr>
<td>6</td>
<td>30.627</td>
<td>35.9142</td>
<td>-5.2872</td>
</tr>
<tr>
<td>7</td>
<td>50.732</td>
<td>44.2839</td>
<td>6.44805</td>
</tr>
<tr>
<td>8</td>
<td>26.067</td>
<td>35.1402</td>
<td>-9.07316</td>
</tr>
<tr>
<td>9</td>
<td>48.585</td>
<td>47.9179</td>
<td>0.66708</td>
</tr>
<tr>
<td>10</td>
<td>34.001</td>
<td>20.4918</td>
<td>13.5092</td>
</tr>
<tr>
<td>11</td>
<td>36.869</td>
<td>41.5319</td>
<td>-4.66285</td>
</tr>
<tr>
<td>12</td>
<td>20.049</td>
<td>21.7523</td>
<td>-1.70334</td>
</tr>
<tr>
<td>13</td>
<td>19.146</td>
<td>27.3233</td>
<td>-8.17731</td>
</tr>
<tr>
<td>14</td>
<td>18.905</td>
<td>21.615</td>
<td>-2.71003</td>
</tr>
<tr>
<td>15</td>
<td>27.823</td>
<td>26.7079</td>
<td>1.11512</td>
</tr>
<tr>
<td>16</td>
<td>16.241</td>
<td>3.99646</td>
<td>12.2445</td>
</tr>
<tr>
<td>17</td>
<td>0.224</td>
<td>-3.27093</td>
<td>3.49493</td>
</tr>
<tr>
<td>18</td>
<td>30.516</td>
<td>26.1199</td>
<td>4.3961</td>
</tr>
<tr>
<td>19</td>
<td>33.705</td>
<td>41.5748</td>
<td>-7.86983</td>
</tr>
<tr>
<td>20</td>
<td>40.97</td>
<td>50.1439</td>
<td>-9.17386</td>
</tr>
<tr>
<td>21</td>
<td>52.794</td>
<td>41.9918</td>
<td>10.8022</td>
</tr>
<tr>
<td>22</td>
<td>41.968</td>
<td>46.312</td>
<td>-4.3403</td>
</tr>
<tr>
<td>23</td>
<td>39.175</td>
<td>40.7549</td>
<td>-1.57987</td>
</tr>
<tr>
<td>24</td>
<td>53.711</td>
<td>44.6594</td>
<td>9.05163</td>
</tr>
<tr>
<td>25</td>
<td>25.962</td>
<td>32.3503</td>
<td>-6.3883</td>
</tr>
<tr>
<td>26</td>
<td>22.541</td>
<td>28.7892</td>
<td>-6.2482</td>
</tr>
<tr>
<td>27</td>
<td>26.645</td>
<td>42.4086</td>
<td>-15.7636</td>
</tr>
<tr>
<td>28</td>
<td>29.028</td>
<td>38.444</td>
<td>-9.41604</td>
</tr>
<tr>
<td>29</td>
<td>36.664</td>
<td>40.2069</td>
<td>-3.54291</td>
</tr>
<tr>
<td>30</td>
<td>42.445</td>
<td>29.5763</td>
<td>12.8687</td>
</tr>
<tr>
<td>31</td>
<td>56.92</td>
<td>49.8111</td>
<td>7.10888</td>
</tr>
<tr>
<td>32</td>
<td>61.299</td>
<td>50.2007</td>
<td>11.0983</td>
</tr>
<tr>
<td>33</td>
<td>60.75</td>
<td>45.8499</td>
<td>14.9001</td>
</tr>
<tr>
<td>34</td>
<td>68.892</td>
<td>40.2433</td>
<td>28.6487</td>
</tr>
<tr>
<td>35</td>
<td>38.298</td>
<td>31.3065</td>
<td>6.99149</td>
</tr>
<tr>
<td>36</td>
<td>54.839</td>
<td>51.2895</td>
<td>3.54949</td>
</tr>
<tr>
<td>37</td>
<td>56.706</td>
<td>47.1179</td>
<td>9.58808</td>
</tr>
<tr>
<td>38</td>
<td>62.275</td>
<td>51.2954</td>
<td>10.9796</td>
</tr>
<tr>
<td>39</td>
<td>46.716</td>
<td>41.9433</td>
<td>4.77272</td>
</tr>
<tr>
<td>40</td>
<td>57.066</td>
<td>40.9533</td>
<td>16.1127</td>
</tr>
<tr>
<td>41</td>
<td>54.522</td>
<td>41.6791</td>
<td>12.8429</td>
</tr>
<tr>
<td>42</td>
<td>43.962</td>
<td>31.1225</td>
<td>12.8395</td>
</tr>
<tr>
<td>43</td>
<td>40.074</td>
<td>46.1278</td>
<td>-6.05381</td>
</tr>
<tr>
<td>44</td>
<td>23.974</td>
<td>36.949</td>
<td>-12.975</td>
</tr>
<tr>
<td>45</td>
<td>17.677</td>
<td>32.8495</td>
<td>-15.1725</td>
</tr>
<tr>
<td>46</td>
<td>14.306</td>
<td>28.7433</td>
<td>-14.4373</td>
</tr>
<tr>
<td>48</td>
<td>16.531</td>
<td>18.5037</td>
<td>-1.9727</td>
</tr>
<tr>
<td>49</td>
<td>16.492</td>
<td>15.1476</td>
<td>1.34437</td>
</tr>
</tbody>
</table>
26.5 Exercise

With the Problem Files you created earlier, you can now also carry out ordinary least squares regression. You may want to set the Long Output option to YES to produce a map for the residuals (by importing the report file into a graphics or mapping package).
References

CHAPTER 27

MAXIMUM LIKELIHOOD ESTIMATION OF THE SPATIAL LAG MODEL

27.1 Introduction

The estimation of a model with a spatially lagged dependent variable may be carried out according to two different principles. One is based on an underlying normal distribution and is an application of Maximum Likelihood (ML) estimation. The other is more robust, and utilizes Instrumental Variables in the estimation. The ML estimation is invoked by the first command in the Space Lag Model menu of the Regress module:

- SAR-ML (R-4-1)

Instrumental variables estimation of the spatial lag model is outlined in Chapter 28.

The SAR-ML command is started by entering the appropriate command sequence, followed by the name of the Problem File in batch mode (option 1), or by entering the requested information in interactive mode (option 2). A Problem File for the estimation of a spatial lag model must contain the name of a dataset, the file name for a spatial weights matrix, the type of model specification, a dependent variable and at least one explanatory variable. If no spatial weights matrices are specified, an error message will be generated. When more than one spatial weights matrix is included, only the first one is used to estimate the model, while the others are used in the diagnostics for spatial dependence. Variables to be used in an additive heteroskedastic specification may be included as well. For details on the creation of Problem Files in the Regress module, see Chapter 25.

27.2 Methodology

27.2.1 The Spatial Lag Model

The spatial lag model or mixed regressive spatial autoregressive model includes a spatially lagged dependent variable, Wy, as one of the explanatory variables:

\[ y = \rho Wy + X\beta + \epsilon \]

where \( y \) is a N by 1 vector of observations on the dependent variable, \( Wy \) is a N by 1 vector of spatial lags for the dependent variable, \( \rho \) is the spatial autoregressive coefficient, \( X \) is a N by K matrix of observations on the (exogenous) explanatory variables with associated a K by 1 vector of regression coefficients \( \beta \), and \( \epsilon \) is a N by 1 vector of normally distributed random error terms, with means 0 and constant (homoskedastic) variances \( \sigma^2 \).
The presence of the spatial lag is similar to the inclusion of endogenous variables on the RHS in systems of simultaneous equations. This model is therefore often referred to as the simultaneous spatial autoregressive model. If the autoregressive parameter were known, the model would simplify to a standard regression of "filtered" dependent variables $y - \rho Wy$ on the explanatory variables $X$, as in:\footnote{If you know (or assume) the value for the spatial autoregressive coefficient, you can construct a spatially filtered (or screened) dependent variable by means of the \texttt{Tools - Space Trans - Spatial Autoregressive Filter} command sequence (T-1-3). For a more elaborate discussion of the filtering or screening approach to dealing with spatial dependence, see Getis (1990).}

$$y - \beta Wy = X\beta + \epsilon$$

However, typically the $\rho$ coefficients are unknown and must be estimated jointly with the regression coefficients. The main consequence of the inclusion of Wy on the RHS of the specification is that OLS no longer achieves consistency. This is similar to what happens in systems of simultaneous equations. Instead of OLS, estimation must be based on an instrumental variables approach (IV or 2SLS), as outlined in the next chapter, or on an explicit maximization of the likelihood function, as discussed here.

You can interpret the mixed regressive spatial autoregressive specification in two different ways. In one, you consider the inclusion of Wy in addition to other explanatory variables as a way to assess the degree of spatial dependence, while controlling for the effect of these other variables. Hence, the main interest is in the spatial effect. Alternatively, the inclusion of Wy allows you to assess the significance of the other (non-spatial) variables, after the spatial dependence is controlled for. This would be the spirit in which you would implement spatial filtering or screening.

\textit{Maximum Likelihood Estimation}

Maximum likelihood estimation of the spatial lag model is based on the assumption of normal error terms. Given this assumption, a likelihood function can be derived that is a nonlinear function of the parameters and must be maximized.\footnote{The resulting likelihood function is of the form:}

\[ L = \prod_i \ln(1 - \rho \omega_i) - N/2 \ln(2\pi) - N/2 \ln(\sigma^2) - (y - \rho Wy - X\beta)'(y - \rho Wy - X\beta) / 2\sigma^2 \]

with $\omega_i$ as the eigenvalues of the weights matrix and the rest of the notation as in the main text. For full details on the derivation and implementation considerations, see Ord (1975), Anselin (1980), Anselin (1988a), and Anselin and Hudak (1992).
gressive coefficient $\rho$.

A simple search over values for $\rho$ will quickly yield the ML estimate. The other parameters can then be found from a least squares regression of $y - \rho Wy$ on $X$. *SpaceStat* implements a bisection search over values of $\rho$ in the interval $1/\omega_{\text{min}}$ to $1/\omega_{\text{max}}$, where $\omega_{\text{min}}$ and $\omega_{\text{max}}$ are respectively the smallest and the largest eigenvalues of the weights matrix.

This interval defines which values for $\rho$ yield a stable specification for the autoregressive model. Values outside this interval are not acceptable.

**Measures of Fit**

The traditional $R^2$ measure of fit, based on the decomposition of total sum of squares into explained and residual sum of squares, is not applicable to the spatial lag model. Instead, a number of so-called pseudo $R^2$ measures can be computed. *SpaceStat* includes two such measures. One is a simple ratio of the variance of the predicted values over the variance of the observed values for the dependent variable. This is reported as $R^2$ in the *SpaceStat* output. The second measure is the squared correlation between the predicted and observed values, listed as Sq. Corr. in the *SpaceStat* output.

The proper measures for goodness-of-fit for the spatial lag model are based on the likelihood function. These include the value of the maximized log likelihood, the Akaike Information Criterion (AIC) and Schwartz Criterion (SC), introduced in the previous chapter. The likelihood-based measures are directly comparable with those achieved for the standard regression model, while the pseudo-$R^2$ is not. The model with the highest log likelihood, or with the lowest AIC or SC is best.

**Hypothesis Tests**

All statistical inference for estimates obtained with the maximum likelihood approach is based on asymptotic considerations, i.e., for sample sizes that become infinitely large in the limit. This may not be very reliable for small data sets. Specifically, the asymptotic standard errors (or, the asymptotic variance matrix) will tend to give an overly optimistic indication of the precision of the estimates. In other words, the asymptotic standard errors for the estimates will tend to be smaller than they should be for the actual sample size used, which will result in a stronger indication of significance than may be merited. Several small sample corrections to asymptotic results have been suggested in the literature, but none of them applies directly to the spatial models. What this means in practice is that you should be cautious in deciding on the significance of a coefficient when this significance is only marginal. However, when

---

3. The concentrated likelihood function takes on the form:

$$L_c = -(N/2) \ln [(\epsilon_0 - \rho \epsilon_1)(\epsilon_0 - \rho \epsilon_1)/N] + \sum \ln (1 - \rho \omega_i)$$

with $\epsilon_0$ and $\epsilon_1$ the residuals in an OLS regression of $y$ on $X$ and $Wy$ on $X$ respectively. For further details, see Anselin (1988a), and Anselin and Hudak (1992).

4. See Anselin and Hudak (1992) for technical details on the implementation of this bisection search.

In the standard regression model, this variance ratio is equivalent to the $R^2$, but in the spatial lag model, it is not.
a coefficient is highly significant (e.g., with a p value < 0.001) or not at all significant (e.g., with a p value > 0.20), the fact that you are using an asymptotic approximation is unlikely to matter very much.

The analytical expressions for the asymptotic standard errors for the estimates in the spatial lag model are complex. In contrast to what is the case for many "canned" nonlinear optimization routines, where these asymptotic standard errors are often found from a numerical approximation, **SpaceStat** uses the correct analytical expressions. The standard errors are obtained by substituting the ML estimates for the parameters into this expression.\(^6\)

Given the asymptotic nature of the standard errors, the significance of the individual model coefficients is based on a standard normal distribution (the asymptotic result) and not on the Student t distribution (as is the case for OLS). The so-called asymptotic t-test (which is actually a z-value since the limiting distribution is the standard normal) is obtained as the ratio of the estimate to its asymptotic standard error. **SpaceStat** lists the estimates, their asymptotic standard errors, associated z-value and corresponding probability.

### 27.2.2 Specification Diagnostics

Similar to the situation in the standard regression model, the results of the spatial lag model are subject to a number of assumptions. However, since estimation is based on the maximum likelihood approach, a much narrower range of specification diagnostics is available. Such tests are either Lagrange Multiplier tests or Likelihood Ratio tests. Their are all asymptotic and may lead to inconsistent conclusions in finite samples.\(^7\) **SpaceStat** includes Lagrange Multiplier tests against heteroskedasticity, a Likelihood Ratio test on the spatial autoregressive coefficient \(\rho\), and a Lagrange Multiplier test on remaining spatial error autocorrelation.

**Heteroskedasticity**

Two statistics for heteroskedasticity are reported in the **SpaceStat** output for the spatial lag model. One is the Breusch-Pagan test, based on the residuals from the ML estimation, but otherwise identical to the formulation used for the standard regression model. Strictly speaking, this test is incorrect, since it ignores the spatial dependence in the model, but for all practical purposes it does not seem to make much difference. The test statistic and its degrees of freedom are reported, as well as the corresponding probability according to a \(\chi^2\) distribution. The proper Lagrange Multiplier test for heteroskedasticity in a spatial lag model includes some adjustments to the Breusch-Pagan framework.\(^8\) It is only slightly different from the standard BP result.

---

6. The complete expression for the asymptotic variance is given in Ord (1975), Anselin (1980), among others.
7. See Anselin (1988a, Chapter 6) for an extensive discussion of the Wald, Lagrange Multiplier and Likelihood Ratio tests in spatial models.
8. For technical details, see Anselin (1988b) and also Chapter 6 of Anselin (1988a).
**SpaceStat** reports this spatial BP statistic as well, with its degrees of freedom and corresponding probability according to a $\chi^2$ distribution.

As before, for the standard model, the tests against heteroskedasticity either use a set of variables specified by you, or take the random coefficients model as the default.

**Spatial Lag Dependence**

**SpaceStat** includes a Likelihood Ratio (LR) test on the spatial autoregressive coefficient $\rho$. This LR test corresponds to twice the difference between the log likelihood in the spatial lag model and the log likelihood in a standard regression model with the same set of explanatory variables (i.e., with $\rho$ set to zero). It is distributed as a $\chi^2$ variate with one degree of freedom. The statistic, its degrees of freedom (always 1) and corresponding probability are given. Note that you can also compute this test in a straightforward way from the outputs for the two models, but its magnitude is listed explicitly for ease of interpretation.

Even though the LR, Wald (asymptotic t-test) and LM tests (for a spatial lag in the standard regression model) are all asymptotically equivalent, they will tend to yield different results in finite samples. In most instances, the ordering of statistics in terms of their magnitude is as:

$$W \geq LR \geq LM$$

This order implies that you are more likely to consider the autoregressive coefficient to be significant from the results of a Wald test than based on a LM test. In some cases, the three tests may give conflicting indications, e.g. a significant Wald test (say at $p=0.01$), marginally significant LR test (say at $p=0.05$) and insignificant LM test (say at $p=0.15$). In such instances you should exert extreme caution when interpreting the results of the model.

You may check the relative order of statistics by taking the square of the z-value listed by **SpaceStat** for the autoregressive coefficient, the value for the LR test and the value for the LM test on a spatial lag in the standard regression model (of course, all need to be computed for the same spatial weights matrix). Typically, your results will satisfy the ordering given above. When they do not, you should consider it as an indication of a potential specification error. Examples of such specification problems are non-normal error terms, a nonlinear relationship between dependent and explanatory variables, and a poor choice of the variables included in the model and/or of the spatial weights matrix.

**Spatial Error Dependence**

If the spatial lag model you specified is indeed the correct one, then no spatial dependence should remain in the residuals. The Lagrange Multiplier test for spatial error autocorrelation in the spatial lag model, suggested by Anselin (1988b), is a diagnostic for this. The
statistic is asymptotically distributed as $\chi^2$ with one degree of freedom. Its value, degrees of freedom (always one) and associated probability are reported by SpaceStat, for each weights matrix specified in the Problem File.

A significant result for this test indicates one of two things. If the weights matrix for the test is the same as the weights matrix in the spatial lag model, the latter must be misspecified. The significant statistic shows that not all spatial dependence has been eliminated (or new, spurious patterns of spatial dependence have been created) which casts a serious doubt on the appropriateness of the spatial weights specification in the model. Instead of the current spatial lag model, you could try a higher order spatial autoregressive model, a different weights matrix, or a completely different model specification (e.g., a spatial error model). On the other hand, if the weights matrix for the test is not the same as the one in the spatial lag, a significant statistic may point to the appropriateness of a mixed autoregressive spatial moving average model, i.e., a model with a spatial lag as well as a spatial moving average process in the error terms.\(^9\)

27.3 Example Problem File

As an example, you will continue the analysis of the regression model that relates the CRIME variable in the Columbus data set to a constant term and the explanatory variables INCOME and HOUSING. Make sure that the data set (the files COL.DAT and COL.DHT) is on your current directory. In addition, the problem file COL7.BTC and the weights matrices COLWS_1.FMT, COLWS_2.FMT, and COLWS_3.FMT should be on the current directory as well. The Problem File COL7.BTC, created in Chapter 25 contains all the information to carry out this example. It is contained on the \EXAMPLES directory and also listed below, as Table 27.1.

You start the spatial lag model by means of the R-4-1 command sequence. Enter col7.btc as the name for the Problem File (followed by Return).

If you have the Long Output option set to NO and you did not earlier estimate a spatial lag or spatial error model using the weights matrix COLWS_1, you will get the following series of messages:

```
Reading in data ...
Computing eigenvalues of the weights matrix ...
Starting analysis ...
Bisection search for rho ...
```

9. The LM statistics takes on the form:

$$LM_{stat} = \frac{(e'W'e)/{tr(W'W)}}{(\text{var}(\rho))}$$

where $W$ is the weights matrix, $e$ are the residuals in the ML estimation, var($\rho$) the estimated asymptotic variance for the spatial autoregressive coefficient. For technical details, see Anselin (1988b) and also Anselin (1988a, Chapter 8).

10. The estimation of such a model is not yet implemented in SpaceStat.
Computing asymptotic variance matrix ...

These messages keep you up to date on the progress of the program. When your data set is small, as is the case for the Columbus example, these messages come up fairly quickly, with hardly a delay between them. However, when you carry out the estimation for moderately sized data sets, there can be quite some time between them, especially for the eigenvalue computation and the computation of the asymptotic variance matrix. When the eigenvalues for the spatial weights matrix are stored in the weights file (i.e., they were computed as part of an earlier command), the second message is skipped.

The messages are followed by the results, one screen at a time. These results will also be written to the file you specified in the Output File option.

<table>
<thead>
<tr>
<th>Table 27.1 Problem file COL7.BTC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>1 1 3 1 1 0 2 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>COL REP COLWS_1 COLWS_2 COLWS_3 CRIME INCOME HOUSING</td>
</tr>
</tbody>
</table>

### 27.4 Program Output

#### 27.4.1 Standard Output

The first screen of the standard output lists the estimates, their asymptotic standard deviations, associated z-values and corresponding probability levels, as illustrated in Table 27.2. Note that all coefficients are highly significant, including the $\rho$ parameter associated with the spatially lagged dependent variable W.CRIME. This confirms the pronounced pattern of spatial clustering for CRIME found in previous sections (e.g., using Moran’s I) and illustrated in the map included as Appendix G. Compared to the OLS estimates, the coefficient for HOUSING has largely remained the same (-0.266 vs. -0.274 for OLS), but both INCOME (-1.032 vs -1.597) and the constant term (45.08 vs 68.62) decreased substantially in absolute value. A change in the regression coefficients is to be expected whenever the spatial lag is highly significant, since the estimates in the OLS model are then biased (due to the omission of a significant explanatory variable, the Wy). If, in contrast, the spatial lag were not significant, there should not be much change in the coefficients between OLS and spatial lag results.

At the top of the table of results is some general information on the model. This is the same as for the standard regression model, except that the spatial weights matrix (COLWS_1) is now listed explicitly as well. Following are several measures of goodness-of-fit. As pointed out above, the pseudo-$R^2$ and squared correlation between observed and predicted values are only of limited value, and should not be used to make comparisons with the OLS results. In contrast, the likelihood based measures (LIK, AIC and SC) can be used to compare the fit of
the two models. It turns out that for the Columbus example, the fit considerably improves when the spatial lag is added to the model, as indicated by an increase in the log likelihood (from -187.4 for OLS to -182.4) and a decrease in AIC and SC (respectively from 380.8 for OLS to 372.8 and from 386.4 for OLS to 380.3). The improved fit was to be expected, since the spatial lag coefficient turned out to be highly significant.

Table 27.2 Maximum Likelihood Estimates in Spatial Lag Model

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>COEFF</th>
<th>S.D.</th>
<th>z-value</th>
<th>Prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_CRIME</td>
<td>0.43102</td>
<td>0.117681</td>
<td>3.662603</td>
<td>0.000250</td>
</tr>
<tr>
<td>CONSTANT</td>
<td>45.0794</td>
<td>7.17736</td>
<td>6.280777</td>
<td>0.000000</td>
</tr>
<tr>
<td>INCOME</td>
<td>-1.03161</td>
<td>0.305143</td>
<td>-3.380757</td>
<td>0.000723</td>
</tr>
<tr>
<td>HOUSING</td>
<td>-0.265927</td>
<td>0.0884987</td>
<td>-3.004871</td>
<td>0.002657</td>
</tr>
</tbody>
</table>

The estimates are followed by a screen of diagnostics, as illustrated in Table 27.3. First are the two tests for heteroskedasticity. As is the case for the OLS regression output, the variables used in the heteroskedastic tests are listed only if you specified them explicitly. In the default case, the random coefficients model is used, as in the example in Table 27.3. It is evident that there still may be quite a bit of ignored heteroskedasticity in the model, which merits further attention. The second item in the table pertains to the LR test on the spatial autoregressive coefficient. The file name for the weights matrix is listed, as well as whether or not it has been row-standardized. The LR statistic of 9.97 corresponds to twice the difference between the log likelihood of the OLS and spatial lag models (2x4.987) and is highly significant. Note that the order of the Wald, LR and LM tests on the spatial lag is as expected:

\[13.41 > 9.97 > 9.36\]

The value for the Wald test (13.41) is the square of the asymptotic t-test on \( \rho \) (3.66 in Table 27.2) and the value for the LM test (9.36) is from Table 26.6 in Chapter 26.

The final item in the list of diagnostics displays the results of the Lagrange Multiplier tests on spatial error dependence. For the three weights matrices specified in the problem file (COLSW_1, COLSW_2 and COLSW_3) a brief note shows whether or not the matrix was row-standardized (below STAND in the table) and whether or not there are zero rows present (below

---

11. The estimation of a spatial lag model with heteroskedastic errors is not implemented in the current release of SpaceStat.
ZERO in the table), followed by the degree of freedom (1), the value for the statistic and associated probability. Clearly, none of the tests is significant, which confirms that the spatial dependence has been adequately dealt with by incorporating the spatial lag term.

Table 27.3 Specification Diagnostics for the Spatial Lag Model

<table>
<thead>
<tr>
<th>REGRESSION DIAGNOSTICS</th>
<th>DIAGNOSTICS FOR HETEROSEDASTICITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>TEST</td>
<td>DF</td>
</tr>
<tr>
<td>Breusch-Pagan test</td>
<td>2</td>
</tr>
<tr>
<td>Spatial B-P test</td>
<td>2</td>
</tr>
</tbody>
</table>

Diagnostics for spatial dependence:

<table>
<thead>
<tr>
<th>SPATIAL LAG DEPENDENCE FOR WEIGHTS MATRIX COLWS_1 (row-standardized weights)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TEST</td>
</tr>
<tr>
<td>Likelihood Ratio Test</td>
</tr>
</tbody>
</table>

Lagrange multiplier test on spatial error dependence:

<table>
<thead>
<tr>
<th>WEIGHT</th>
<th>STAND</th>
<th>ZERO</th>
<th>DF</th>
<th>VALUE</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLWS_1</td>
<td>yes</td>
<td>no</td>
<td>1</td>
<td>0.319548</td>
<td>0.571880</td>
</tr>
<tr>
<td>COLWS_2</td>
<td>yes</td>
<td>no</td>
<td>1</td>
<td>0.068918</td>
<td>0.792919</td>
</tr>
<tr>
<td>COLWS_3</td>
<td>yes</td>
<td>no</td>
<td>1</td>
<td>0.648143</td>
<td>0.420777</td>
</tr>
</tbody>
</table>

27.4.2 Long Output

If you have set the Long Output option to YES, the bisection search message will not appear, since you will see the intermediate results for each iteration of the search, as in Table 27.4. Note that the precision of the search and its stopping point are determined by the value for the Convergence Criterion option (6 in the Options menu). The iterations start with a lower and upper bound, which is determined in function of the value of the partial derivative of the concentrated log likelihood for $\rho=0$. You are also given the limits for the acceptable range of parameter values. These are the $1/\omega_{\min}$ and $1/\omega_{\max}$ discussed before. For a row-standardized weights matrix, the upper limit will always be +1, but the lower limit will typically not be -1. For each iteration, the new value for $\rho$, the corresponding partial derivative for the concentrated log likelihood and the associated trace of an auxiliary matrix are listed. The latter two are useful for identifying potential problems, but a discussion of their interpretation is beyond the scope of this tutorial.

In the long output case, the first screen of estimates is followed by the asymptotic variance matrix for the coefficients, as listed in Table 27.5. The matrix is organized in the same fashion as for the OLS estimates, except that a row for the estimated error variance is included.
as well. Note that this matrix is not block diagonal between the error variance and the other coefficients of the model, in contrast to what is the case in the standard regression model. The square roots of the elements on the diagonal yield the figures for the standard errors given in Table 27.2.

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Rho</th>
<th>DLik</th>
<th>Trace</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.499950</td>
<td>-4.69461</td>
<td>7.45193</td>
</tr>
<tr>
<td>2</td>
<td>0.249975</td>
<td>10.5250</td>
<td>3.03268</td>
</tr>
<tr>
<td>3</td>
<td>0.374962</td>
<td>3.54457</td>
<td>4.96888</td>
</tr>
<tr>
<td>4</td>
<td>0.437456</td>
<td>-0.422917</td>
<td>6.12149</td>
</tr>
<tr>
<td>5</td>
<td>0.406209</td>
<td>1.59937</td>
<td>5.52611</td>
</tr>
<tr>
<td>6</td>
<td>0.421833</td>
<td>0.597776</td>
<td>5.81868</td>
</tr>
<tr>
<td>7</td>
<td>0.429645</td>
<td>0.0898071</td>
<td>5.96876</td>
</tr>
<tr>
<td>8</td>
<td>0.433550</td>
<td>-0.165962</td>
<td>6.04479</td>
</tr>
<tr>
<td>9</td>
<td>0.431597</td>
<td>-0.0379289</td>
<td>6.00669</td>
</tr>
<tr>
<td>10</td>
<td>0.430621</td>
<td>0.0259762</td>
<td>5.98770</td>
</tr>
<tr>
<td>11</td>
<td>0.431109</td>
<td>-0.00596708</td>
<td>5.99719</td>
</tr>
<tr>
<td>12</td>
<td>0.430865</td>
<td>0.0100069</td>
<td>5.99244</td>
</tr>
<tr>
<td>13</td>
<td>0.430987</td>
<td>0.00202048</td>
<td>5.99482</td>
</tr>
<tr>
<td>14</td>
<td>0.431048</td>
<td>-0.00197315</td>
<td>5.99600</td>
</tr>
<tr>
<td>15</td>
<td>0.431018</td>
<td>2.37015E-05</td>
<td>5.99541</td>
</tr>
<tr>
<td>16</td>
<td>0.431033</td>
<td>-0.000974176</td>
<td>5.99571</td>
</tr>
<tr>
<td>17</td>
<td>0.431025</td>
<td>-0.000475505</td>
<td>5.99556</td>
</tr>
<tr>
<td>18</td>
<td>0.431021</td>
<td>-0.000225901</td>
<td>5.99548</td>
</tr>
<tr>
<td>19</td>
<td>0.431020</td>
<td>-0.000101100</td>
<td>5.99545</td>
</tr>
</tbody>
</table>

A Report File will also be created with as filename the one you specified in the problem file (REP), with the extension .DOC added to it. This file will contain an indicator variable

---

12. This "correlation" between the variance of the estimates for $\sigma^2$ and the other coefficients is due to the simultaneity between the spatial lag Wy and the error term. See Anselin (1988a, Chapter 6) for further details.
(or observation sequence number if none was specified in the *Indicator Variable* option), the dependent variable, predicted values and residuals for the model.

### 27.4.3 IDRISI Interface

If the *Idrisi Interface* option is set to YES, and irrespective of the settings for the Output to a File (Option 2) or Long Output (Option 3) options, then two special files are created, with as file names the name of the dependent variable with extensions _YP. VAL and _E.V AL. These files contain the values for the indicator variable in the first column (or the observation sequence number, if no *Indicator Variable* was specified) and respectively the predicted value and residual for the spatial lag model. In the Columbus example, these files would be named CRIME_YP. VAL and CRIME_E. VAL. Note that this is the same filename as used for the OLS regression. Files with identical file names will be generated for all types of regression models. It is up to you to keep track of them. You may therefore want to change the file name for these files if you intend to keep several of them on the same directory (if you don’t change the names, only the latest result will be kept, the rest will be overwritten).

Table 27.5 Asymptotic Variance Matrix for ML Estimates in Spatial Lag Model

<table>
<thead>
<tr>
<th>COEFFICIENT VARIANCE MATRIX</th>
<th>W_CRIME</th>
<th>CONSTANT</th>
<th>INCOME</th>
<th>HOUSING</th>
<th>SIGMA-SQ</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_CRIME</td>
<td>0.0138489</td>
<td>-0.697576</td>
<td>0.0126625</td>
<td>0.000866450</td>
<td>-0.323630</td>
</tr>
<tr>
<td>CONSTANT</td>
<td>-0.697576</td>
<td>51.5145</td>
<td>-1.32603</td>
<td>-0.161638</td>
<td>16.3014</td>
</tr>
<tr>
<td>INCOME</td>
<td>0.0126625</td>
<td>-1.32603</td>
<td>0.0931123</td>
<td>-0.0117960</td>
<td>-0.295905</td>
</tr>
<tr>
<td>HOUSING</td>
<td>0.000866450</td>
<td>-0.161638</td>
<td>-0.0117960</td>
<td>0.00783201</td>
<td>-0.0202478</td>
</tr>
<tr>
<td>SIGMA-SQ</td>
<td>-0.323630</td>
<td>16.3014</td>
<td>-0.295905</td>
<td>-0.0202478</td>
<td>379.776</td>
</tr>
</tbody>
</table>

### 27.5 Exercise

You may now use the same commands with the problem file you constructed for the least squares regression exercise to estimate a spatial lag model with your own data, or for the African vegetation index or Cliff-Ord Irish data. Pay particular attention to the extent of the change in parameter values for the model. You can compare the fit of the spatial lag model to that of the least squares solution by means of the likelihood measures. You can also check whether the inequality between Wald, LR and LM statistics is satisfied in your model.
References


CHAPTER 28

INSTRUMENTAL VARIABLES ESTIMATION OF THE SPATIAL LAG MODEL

28.1 Introduction

The estimation of a model with a spatially lagged dependent variable may also be carried out by means of instrumental variables (IV) methods. This is a robust alternative to maximum likelihood estimation, since the assumption of normally distributed error terms is not needed. SpaceStat contains two commands based on this principle in the Space Lag Model menu of the Regress module:

- **SAR-IV** (R-4-2)
- **SAR-BOOT** (R-4-3)

The first is a straightforward application of instrumental variables estimation or two-stage-least-squares (2SLS) to the spatial lag model. The second approach uses the IV estimator to construct a bootstrap procedure. Since these are robust methods, no specification diagnostics are provided for either approach.

The SAR-ML and SAR-BOOT commands are started by entering the appropriate command sequence, followed by the name of the Problem File in batch mode (option 1), or by entering the requested information in interactive mode (option 2). A Problem File for the IV estimation of a spatial lag model must contain the name of a dataset, the file name for a spatial weights matrix, the type of model specification, a dependent variable, at least one explanatory variable, and a sufficient number of instrumental variables. If no spatial weights matrices or no instrumental variables are specified, an error message will be generated. When more than one spatial weights matrix is included, only the first one is used to estimate the model, while the others are ignored. Variables to be used in an additive heteroskedastic specification are ignored. For details on the creation of Problem Files in the Regress module, see Chapter 25.

28.2 Methodology

As shown in the previous chapter, the spatial lag model or mixed regressive spatial autoregressive model includes a spatially lagged dependent variable, \( W_y \), as one of the explanatory variables:

\[
Y = \rho W_y + X\beta + \epsilon
\]
where $y$ is a $N$ by 1 vector of observations on the dependent variable, $Wy$ is a $N$ by 1 vector of spatial lags for the dependent variable, $\rho$ is the spatial autoregressive coefficient, $X$ is a $N$ by $K$ matrix of observations on the (exogenous) explanatory variables with associated a $K$ by 1 vector of regression coefficients $\beta$, and $\epsilon$ is a $N$ by 1 vector of random error terms, with means 0 and constant (homoskedastic) variances $\sigma^2$. Note that normality is no longer required. For ease of notation, I will represent the spatial lag $Wy$ and the "exogenous" explanatory variables $X$ jointly in a matrix $Z$, of dimension $N$ by $K+1$.

The presence of the spatial lag is similar to the inclusion of endogenous variables on the RHS in systems of simultaneous equations. The instrumental variables estimator (IV) or two-stage-least-squares estimator (2SLS) exploits this feature by constructing a proper instrument for the spatial lag. The resulting estimate is consistent, but not necessarily very efficient. It may be used as the basis for a bootstrap procedure.

### 28.2.1 Instrumental Variables Estimation

The principle of instrumental variables estimation is based on the existence of a set of instruments, $Q$, that are strongly correlated with the original variables, $Z$, but asymptotically uncorrelated with the error term. Once these instruments are identified, they are used to construct a proxy for the endogenous variables, which consists of their predicted values in a regression on the instruments and the exogenous variables. This proxy is then used in a standard least squares regression. Formally, this process of two-stage-least-squares yields the estimate:

$$
\theta_{IV} = (ZQ)(Q'Q)^{-1}(Q'Z)(Q'Q)^{-1}Q'y
$$

where $\theta$ is a $K+1$ by 1 vector with the estimate for $\rho$ as the first element, followed by the estimates for $\beta$, $Q$ is a $N$ by $P$ matrix of instruments (including the exogenous variables $X$), and the other notation is as before. It can be shown that this estimate is consistent and asymptotically efficient. However, its properties in finite samples depend in a crucial way on the choice of the instruments and are not always tractable.\(^1\)

One potential problem with IV estimation is that the estimate for $\rho$ does not necessarily fall in the acceptable range and values larger than 1 in absolute value may be obtained. Typically, this points to potential problems with the specification of the model.

#### Choice of Instruments

In the standard simultaneous equations framework, the instruments are the "excluded" exogenous variables. In the spatial lag model there is no simple equivalent of this notion and

\(^1\) See Bowden and Turkington (1984) for an extensive review of the instrumental variables method, and Anselin (1980, 1984, 1988) for a detailed treatment of its application to spatial lag models.
a number of suggestions have been formulated (see Anselin, 1988, Chapter 7, for an overview). Recently, Kelejian and Robinson (1992) have shown that a series of spatially lagged exogenous variables, for first order and higher order contiguity matrices are the proper set. In practice, this series may be truncated and only the first order spatially lagged explanatory variables may be included. Formally, this results in a matrix Q containing X and WX, where the constant term and other variables that would cause perfect multicollinearity are excluded from WX.

In SpaceStat, you must specify the instruments explicitly. If you choose the spatially lagged exogenous variables, you must first have created these by means of the Tools-Space Trans-Spatial Lag (T−1−1) command sequence. You should never include the exogenous variables (X) explicitly as instruments, since this will cause perfect multicollinearity (they will be included twice in the Q matrix).

**Measures of Fit**

The traditional $R^2$ measure of fit is not applicable to the results of IV estimation. Instead, SpaceStat includes two pseudo $R^2$ measures, the same as in the ML estimation of the spatial lag model. One is a simple ratio of the variance of the predicted values over the variance of the observed values for the dependent variable. This is reported as R2 in the SpaceStat output. The second measure is the squared correlation between the predicted and observed values, listed as Sq. Corr. in the SpaceStat output. However, neither of these measures are comparable with the results for ML estimation.

**Hypothesis Tests**

All statistical inference for estimates obtained with the instrumental variables approach is based on asymptotic considerations, i.e., for sample sizes that become infinitely large in the limit. This may not be very reliable for small data sets.\(^2\)

Given the asymptotic nature of the standard errors, the significance of the individual model coefficients is based on a standard normal distribution (the asymptotic result) and not on the Student t distribution (as is the case for OLS). The so-called asymptotic t-test (which is actually a z-value since the limiting distribution is the standard normal) is obtained as the ratio of the estimate to its asymptotic standard error. SpaceStat lists the estimates, their asymptotic standard errors, associated z-value and corresponding probability.

### 28.2.2 Bootstrap Estimation

The bootstrap is a robust estimator which exploits the randomness present in artificially created resampled data sets as the basis for statistical inference. This leads to alternative param-
eter estimates, measures of bias and variance, and the construction of pseudo significance levels and confidence intervals. In regression analysis, there are two main approaches that lead to a bootstrap estimate: one based on residuals, the other on observation points in multidimensional space. As shown in Anselin (1988, 1990), only the former is acceptable for spatial lag models.

The bootstrap in the spatial lag model is based on a procedure suggested for simultaneous equations by Freedman and Peters (1984a, 1984b). In a first step, an instrumental variable estimation is carried out, which provides an estimate for the error vector ε in the form of the residuals:

\[ e = y - \rho Wy - X\beta \]

with ρ and β replaced by their IV estimates.

In the second step, pseudo error terms are generated by random resampling (with replacement) from the vector e. As shown in Anselin (1988, 1990), a vector of pseudo observations on the dependent variable may be computed for each set e, of N such resampled residuals, as:

\[ y_r = (I - \rho W)^{-1} (X\beta + e_r) \]

with X as the fixed exogenous variables, and ρ and β replaced by their IV estimates. An estimate for ρ and β in the resampled data set is obtained by means of instrumental variables, using Wy as the spatial lag. This procedure is repeated a large number of times, say R, to generate an empirical frequency distribution for the r and b estimates. The bootstrap estimate is then taken to be the mean of this empirical frequency distribution.

Measures of fit may be computed in the same fashion as for the IV estimates. However, they are only of limited value and are not comparable between models.

The number of bootstrap replications is set by means of the Number of Permutations option (Option 7). The default of 99, which is appropriate in most tests for spatial autocorrelation, is much too small for bootstrap estimation and should be changed.

---

3. For extensive overviews, see Efron (1982), Efron and Tibshirani (1986), and Léger et al. (1992).
4. The variance for the bootstrap estimates is found as:

\[ \text{var}[	heta_r] = \frac{1}{(R - 1)} \sum_{k=1}^{R} (\theta_r - \theta_k)(\theta_r - \theta_k)' \]

where \( \theta_r \) is the estimate for each replication r, and R is the total number of replications. See Anselin (1988, 1990) for details.
28.3 Example Problem File

As an example, you will continue the analysis of the regression model that relates the CRIME variable in the Columbus data set to a constant term and the explanatory variables INCOME and HOUSING. Make sure that the data set (the files COL.DAT and COL.DHT) is on your current directory.

In order to carry out the IV and bootstrap estimation, you will need a new Problem File that contains instrumental variables. The Problem File COL8.BTC, listed in Table 28.1 and contained on the \EXAMPLES directory includes the variables W_INCOME and W_HOUSING as instruments. Note the 2 as the fourth from last element in the series of flags, which indicates that two instruments will be used. The variables W_INCOME and W_HOUSING must be present in your data set COL. You can create them by means of the Tools-Space Trans-Spatial Lag (T-1-1) command sequence with the spatial weights matrix COLWS_1.

You start the instrumental variables and bootstrap estimation of the spatial lag model respectively by means of the R-4-2 or R-4-3 command sequence. Enter col8.btc as the name for the Problem File (followed by Return). Before you run the bootstrap estimation, make sure to set the Number of Permutations option to a large number (say 1000 or 10000).

28.4 Program Output

28.4.1 Standard Output for IV Estimation

The only screen of the standard output lists the estimates, their asymptotic standard deviations, associated z-values and corresponding probability levels, as illustrated in Table 28.2. Note that all coefficients are highly significant, including the \( \rho \) parameter associated with the spatially lagged dependent variable W_CRIME. Compared to its ML counterpart, the estimate obtained here (0.444) is only slightly different (compare to 0.431). The main distinction between the two sets of results (compare Table 28.2 to Table 27.2) is in the estimated standard deviations. For example, for \( \rho \), the SD is 0.181 for IV versus 0.118 for ML, resulting in a less extreme indication of significance (p=0.01 vs. p=0.0003).

At the top of the table of results is some general information on the model. This is the same as for the ML estimation of the spatial lag model, except that the variable names for the instruments are listed as well. Following are two measures of goodness-of-fit, the pseudo-
R² and squared correlation between observed and predicted values. These are only of limited value, and should not be used to make comparisons with the OLS results. The last item listed is an estimate for residual variance, SIG-SQ \((e'e/N)\) and its square root. Again, this estimate should be interpreted with caution.

### Table 28.2 Instrumental Variables Estimates in Spatial Lag Model

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>COEFF</th>
<th>S.D.</th>
<th>z-value</th>
<th>Prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_CRIME</td>
<td>0.444197</td>
<td>0.181258</td>
<td>2.450636</td>
<td>0.014260</td>
</tr>
<tr>
<td>CONSTANT</td>
<td>44.3597</td>
<td>10.692</td>
<td>4.148860</td>
<td>0.000033</td>
</tr>
<tr>
<td>INCOME</td>
<td>-1.01432</td>
<td>0.371318</td>
<td>-2.731672</td>
<td>0.006301</td>
</tr>
<tr>
<td>HOUSING</td>
<td>-0.265682</td>
<td>0.0881213</td>
<td>-3.014962</td>
<td>0.002570</td>
</tr>
</tbody>
</table>

### 28.4.2 Long Output for the IV Estimates

In the long output case, the first screen of estimates is followed by the asymptotic variance matrix for the coefficients, as listed in Table 28.3. The matrix is organized in the same fashion as for the OLS estimates. The square roots of the elements on the diagonal yield the figures for the standard errors given in Table 28.2.

In the same fashion as for the other regression methods, a Report File will be created with as filename the one you specified in the Problem File (REP), with the extension .DOC added to it. This file will contain an indicator variable (or observation sequence number if none was specified in the Indicator Variable option), the dependent variable, predicted values and residuals for the model. These results are also listed to the screen.

### Table 28.3 Asymptotic Variance Matrix for IV Estimates in Spatial Lag Model

<table>
<thead>
<tr>
<th>COEFFICIENT VARIANCE MATRIX</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_CRIME</td>
</tr>
<tr>
<td>0.0328544  -1.79429  0.0431195  0.000610112</td>
</tr>
<tr>
<td>CONSTANT</td>
</tr>
<tr>
<td>-1.79429   114.320  -3.04101  -0.150954</td>
</tr>
<tr>
<td>INCOME</td>
</tr>
<tr>
<td>0.0431195  -3.04101  0.137877  -0.0117490</td>
</tr>
<tr>
<td>HOUSING</td>
</tr>
<tr>
<td>0.000610112 -0.150954 -0.0117490  0.00776536</td>
</tr>
</tbody>
</table>
28.4.3 Output for Bootstrap Estimation

The only screen of the standard output lists the estimates, their asymptotic standard deviations, associated z-values and corresponding probability levels, as illustrated in Table 28.4. This follows the same format as for IV estimation. The results in Table 28.4 are for 1000 bootstrap replications. Note the similarity between these results and the ones listed in Table 28.2. In situations where heteroskedasticity may be present and where the assumption of normality may be invalid, the bootstrap estimates provide a viable alternative to ML estimation.

When the Long Output option is set to YES, the estimates are followed by a table with the estimated variance matrix, as well as the complete listing of parameter estimates obtained for each replication. In the usual way, the observed, predicted values and residuals are given as well and written to the Report File.

Table 28.4 Bootstrap Estimates in Spatial Lag Model

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>COEFF</th>
<th>S.D.</th>
<th>z-value</th>
<th>Prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_CRIME</td>
<td>0.4463</td>
<td>0.1863</td>
<td>2.3946</td>
<td>0.0166</td>
</tr>
<tr>
<td>CONSTANT</td>
<td>43.66</td>
<td>10.14</td>
<td>4.3064</td>
<td>0.0001</td>
</tr>
<tr>
<td>INCOME</td>
<td>-0.98</td>
<td>0.34</td>
<td>-2.88</td>
<td>0.0040</td>
</tr>
<tr>
<td>HOUSING</td>
<td>-0.26</td>
<td>0.09</td>
<td>-2.99</td>
<td>0.0027</td>
</tr>
</tbody>
</table>

28.4.4 IDRISI Interface

If the Idrisi Interface option is set to YES, and irrespective of the settings for the Output to a File (Option 2) or Long Output (Option 3) options, then two special files are created, with as file names the name of the dependent variable with extensions _YP.VAL and _E.VAL. These files contain the values for the indicator variable in the first column (or the observation sequence number, if no Indicator Variable was specified) and respectively the predicted value and residual for the spatial lag model. As is the case for the other regression methods, these files would be named CRIME_YP.VAL and CRIME_E.VAL and will overwrite any existing files with the same name.

28.5 Exercise

You may now take a similar approach, specifying the spatially lagged explanatory variables as instruments, to estimate a spatial lag model with your own data, or for the African vegetation index or Cliff-Ord Irish data. Pay particular attention to the extent of the change
in parameter values for the model. You may experiment with including higher order spatial lags as instruments and assess the effect on the bootstrap estimates of an increased number of replications.

References


MAXIMUM LIKELIHOOD ESTIMATION OF THE SPATIAL ERROR MODEL

29.1 Introduction

The estimation of a model with a spatially dependent error term is based on an underlying normal distribution, as an application of Maximum Likelihood (ML) estimation. It is invoked by the first command in the Space Err Model menu of the Regress module:

- **Spatial AR** ($R^{-2-1}$)

This function is started by entering the appropriate command sequence, followed by the name of the Problem File in batch mode (option 1), or by entering the requested information in interactive mode (option 2). A Problem File for the estimation of a spatial error model must contain the name of a dataset, the file name for a spatial weights matrix, the type of model specification, a dependent variable and at least one explanatory variable. If no spatial weights matrices are specified, an error message will be generated. When more than one spatial weights matrix is included, only the first one is used to estimate the model, while the others are ignored. Variables to be used in an additive heteroskedastic specification may be included as well. For details on the creation of Problem Files in the Regress module, see Chapter 25.

29.2 Methodology

29.2.1 The Spatial Error Model

The spatial error model is a special case of a so-called non-spherical error model, i.e., a regression specification for which the assumptions of homoskedastic (constant variance) and uncorrelated errors are not satisfied. The spatial dependence in the error term can take on a number of different forms. In the current version of SpaceStat, only a spatial autoregressive process for the error term can be estimated. This model is the standard regression specification with a spatial autoregressive error term:

\[
Y = X\beta + \epsilon \\
\epsilon = \lambda We + \xi
\]

where $Y$ is a $N$ by 1 vector of observations on the dependent variable, $X$ is a $N$ by $K$ matrix of observations on the explanatory variables, $\beta$ is a $K$ by 1 vector of regression coefficients, $\epsilon$ is a $N$ by 1 vector of error terms, $W_e$ is a spatial lag for the errors, $\lambda$ is the autoregressive coefficient and $\xi$ is a "well-behaved" error, with mean 0 and variance matrix $\sigma^2 I$. 
As a consequence of the spatial dependence, the error term no longer has the usual diagonal variance matrix, but instead takes on the following form:

\[ E[\epsilon' \epsilon] = \Omega = \sigma^2 [(I - \lambda W)'(I - \lambda W)]^{-1} \]

It is easy to show that if the autoregressive coefficient \( \lambda \) were known, the regression coefficients could be estimated by means of OLS in a model with spatially filtered variables \( y - \lambda Wy \) and \( X - \lambda WX \):

\[ y - \lambda Wy = (X - \lambda WX)\beta + \epsilon \]

where \( Wy \) and \( WX \) are spatially lagged dependent and explanatory variables, and the error term \( \xi \) follows the assumptions of the classic model.

This approach is referred to as Generalized Least Squares (GLS). However, the \( \lambda \) coefficient is typically not known and must be estimated jointly with the regression coefficients. For serial autocorrelation in the time domain, a number of two-step so-called Feasible or Estimated Generalized Least Squares (FGLS or EGLS) procedures have been developed, such as the familiar Cochrane-Orcutt estimator. However, due to the simultaneity implied by the spatial nature of the dependence, these procedures are not applicable in the spatial case and a full maximum likelihood estimation must be carried out.\(^1\)

As pointed out before, the consequences of ignoring spatial error dependence are not quite as severe as those of ignoring spatial lag dependence. The main problem is that the OLS estimates become inefficient, but they are still unbiased. The maximum likelihood estimates are consistent, and, in most cases (including the spatial case) also yield unbiased estimates for the regression coefficients \( \beta \) (but not for the so-called nuisance parameter \( \lambda \)). It is still somewhat a matter of debate whether the superior asymptotic efficiency of the maximum likelihood approach also translates into higher efficiency in finite samples.\(^2\)

**Maximum Likelihood Estimation**

As is the case for the spatial lag model, maximum likelihood estimation of the spatial error model is based on the assumption of normal error terms. Given this assumption, a likelihood function can be derived that is a nonlinear function of the parameters and must be maximized.\(^3\) In general, a solution can be found as a straightforward application of nonlinear optimization. As pointed out, the estimates for \( \beta \) and \( \sigma^2 \) can be expressed in function of

---

1. For a more extensive technical discussion of the relative merits of the various estimators suggested in the literature, see Anselin (1988a, Chapter 8).
2. See for example Florax and Folmer (1992) for a recent examination of small sample properties of various estimators for this model.
a value for the autoregressive parameter $\lambda$ (in a GLS approach). After substituting these expressions into the likelihood, a concentrated likelihood can be found, which is solely a function of the autoregressive parameter $\lambda$, similar to the result for a spatial lag.\(^4\) A maximum likelihood estimate for $\lambda$ can be found by means of a simple search over the acceptable interval $1/\omega_{\min}$ to $1/\omega_{\max}$, as in the spatial lag case. However, in some instances of ill-behaved data matrices, the intermediate EGLS step may lead to problems with unacceptable rounding errors and even perfect multicollinearity. In SpaceStat, all interim estimations are carried out with sufficient precision to avoid these problems.

**Measures of Fit**

The standard R\(^2\) measure of fit is not applicable to models with non-spherical errors. SpaceStat reports three pseudo-R\(^2\) measures in its output. The first two are the same as in the spatial lag model: the ratio of the variance of the predicted values over the variance of the observed values for the dependent variable (listed as R\(^2\)) and the squared correlation between these two magnitudes (reported as Sq. Corr.). The third measure is an application of the adjustments suggested by Buse (1973) to the spatial case. It is reported as R\(^2\)(Buse) in the SpaceStat output. In principle, the Buse measures are comparable across models with non-spherical errors, but they are sometimes difficult to interpret.

Three additional measures of goodness-of-fit are based on the maximum likelihood framework: the log likelihood, the Akaike information criterion (AIC), and the Schwartz criterion (SC). These measures are comparable between the standard regression model and the spatial lag and spatial error models. Note that the correction used for AIC and SC in the spatial error model does not count the spatial autoregressive parameter (which may be interpreted as a nuisance parameter). This will tend to favor the spatial error model over the spatial lag model in terms of the information criteria. You can easily compute an adjustment that would include the additional parameter, using the expressions given in Chapter 26.

---

3. The likelihood is of the familiar form for non-spherical errors:

$$L = \Sigma \ln(1 - \lambda \omega_i) - N/2 \ln(2\pi) - N/2 \ln(\sigma^2) - (y - X\beta)'(I - \lambda W)(y - X\beta)/2\sigma^2$$

with $\omega_i$ as the eigenvalues of the weights matrix and the rest of the notation as in the main text. For full details on the derivation and implementation considerations, see Ord (1975), Anselin (1980), Anselin (1988a, Chapter 8), and Anselin and Hudak (1992).

4. The concentrated likelihood function takes on the form:

$$L_C = -N/2 \ln((e'e)/N) + \Sigma \ln(1 - \lambda \omega_i)$$

where $e'e$ is the residual sum of squares in the regression of the spatially filtered dependent and explanatory variables $y - \lambda W y$ and $X - \lambda WX$ (of course, these are a function of $\lambda$). For further details, see Anselin (1980), Anselin (1988a, Chapter 8), and Anselin and Hudak (1992).

5. The expression for the Buse R\(^2\) in the spatial case is:

$$R_B^2 = 1 - (e - \lambda W e)'(e - \lambda W e)/(y - y_w)'(I - \lambda W)(y - y_w)$$

with $y_w = (1 - \lambda W y)/(1 - \lambda W)$ and $1$ as a N by 1 vector of ones. See Anselin (1988a, Chapter 14) for technical details.
**Hypothesis Tests**

As is the case for the spatial lag model, statistical inference in the spatial error model is based on asymptotic considerations. The analytical expressions for the asymptotic standard errors are complex. In contrast to what is the case for many “canned” nonlinear optimization routines, where these asymptotic standard errors are often found from a numerical approximation, **SpaceStat** uses the correct analytical expressions. The standard errors are obtained by substituting the ML estimates for the parameters into these expressions.⁶

Given the asymptotic nature of the standard errors, the significance of the individual model coefficients is based on a standard normal distribution (the asymptotic result) and not on the Student t distribution (as is the case for OLS). The so-called asymptotic t-test (which is actually a z-value since the limiting distribution is the standard normal) is obtained as the ratio of the estimate to its asymptotic standard error. **SpaceStat** lists the estimates, their asymptotic standard errors, associated z-value and corresponding probability.

### 29.2.2 Specification Diagnostics

**SpaceStat** includes a number of specification diagnostics to test the assumptions on which the maximum likelihood estimation in the spatial error model is based. Similar to the approach taken for the spatial lag model, there are Lagrange Multiplier tests against heteroskedasticity and a Likelihood Ratio (LR) test on the spatial error autoregressive coefficient. In addition, there also are a LR and a Wald test on the common factor hypothesis, which is a test on the internal consistency of the spatial error specification.

**Heteroskedasticity**

The two tests against heteroskedasticity reported in the **SpaceStat** output for the spatial error model are the unadjusted and spatially adjusted Breusch-Pagan statistics, based on the same principles as the BP tests in the spatial lag model.⁷ **SpaceStat** reports each statistic as well as its degrees of freedom and corresponding probability according to a $\chi^2$ distribution.

As before, for the standard model, the tests against heteroskedasticity either use a set of variables specified by you, or take the random coefficients model as the default.

**Spatial Error Dependence**

**SpaceStat** includes a Likelihood Ratio (LR) test on the spatial autoregressive coefficient $\lambda$. This LR test corresponds to twice the difference between the log likelihood in the spatial error model and the log likelihood in a standard regression model with the same set of explanatory variables (i.e., with $\lambda$ set to zero). It is distributed as a $\chi^2$ variate with one degree of freedom. The statistic, its degrees of freedom (always 1) and corresponding probability are

---

⁶ The complete expression for the asymptotic variance is given in Ord (1975) and Anselin (1980).
⁷ For technical details, see Anselin (1988a, Chapter 8) and Anselin (1988b).
given. Note that you can also compute this test in a straightforward way from the outputs for the two models, but its magnitude is listed explicitly for ease of interpretation.

**Common Factor Hypothesis**

The spatial error model is equivalent to a spatial lag model of a special form, often referred to as the common factor or spatial Durbin model: ⁸

\[ y = \lambda Wy + X\beta - \lambda WX\beta + \xi \]

in the same notation as above. In an unconstrained form, this model would be:

\[ y = \lambda Wy + X\beta - \lambda WX\gamma + \xi \]

What makes this common factor model special is the nonlinear constraint implied on the coefficients of the spatial lag (\(\lambda\)), the explanatory variables (\(\beta\)) and the spatially lagged explanatory variables (\(\gamma\)). To be consistent with a spatial error formulation, the coefficients of the WX terms should equal the negative of the product of the coefficient of the Wy term with the coefficients of the X terms. This is formally expressed as the so-called common factor hypothesis:

\[ H_{0}: \lambda \cdot \beta = \gamma \]

**SpaceStat** implements two ways to test this hypothesis. In one, a Wald test on the set of nonlinear constraints implied by the common factor model is carried out. This test asymptotically has a \(\chi^2\) distribution with as many degrees of freedom as regression coefficients, but not including the constant term. The test is computed from an auxiliary maximum likelihood estimation of the unconstrained spatial lag model (i.e., the nonlinear constraints are not enforced). An alternative is to compute a Likelihood Ratio test from the maximized likelihood of the spatial error model (i.e., the constrained model) and the likelihood of the unconstrained spatial lag model. The LR test also follows a \(\chi^2\) distribution asymptotically, with the same degrees of freedom as the Wald test. Both statistics, their degrees of freedom and associated probabilities are reported.

In some instance, the lagged explanatory variables WX lead to perfect multicollinearity (e.g., when dummy variables are included). In those instances, the test on the common factor hypothesis is not computed.

---

⁸. The model is referred to as the spatial Durbin model in analogy to a similar specification for time series data. See Burridge (1981) and Bivand (1984) for technical details, and also Anselin (1988a, Chapter 13).
29.3 Example Problem File

As an example, you will continue the analysis of the regression model that relates the CRIME variable in the Columbus data set to a constant term and the explanatory variables INCOME and HOUSING. Make sure that the data set (the files COL.DAT and COL.DHT) is on your current directory. In addition, the problem file COL7.BTC and the weights matrices COLWS_1.FMT, COLWS_2.FMT, and COLWS_3.FMT should be on the current directory as well. The Problem File COL7.BTC, created in Chapter 25 contains all the information to carry out this example. It is contained on the \EXAMPLES directory and also listed below, as Table 29.1.

You start the spatial error model by means of the R-2-1 command sequence. Enter col7.btc as the name for the Problem File (followed by Return).

If you have the Long Output option set to NO and you did not earlier estimate a spatial lag or spatial error model using the weights matrix COLWS_1, you will get the following series of messages:

- Reading in data ...
- Computing eigenvalues of the weights matrix ...
- Starting analysis ...
- Bisection search for lambda...
- Computing asymptotic variance matrix ...

These messages keep you up to date on the progress of the program. When your data set is small, as is the case for the Columbus example, these messages come up fairly quickly, with hardly a delay between them. However, when you carry out the estimation for moderately sized data sets, there can be quite some time between them, especially for the eigenvalue computation and the computation of the asymptotic variance matrix. When the eigenvalues for the spatial weights matrix are stored in the weights file (i.e., they were computed as part of an earlier command), the second message is skipped.

The messages are followed by the results, one screen at a time. These results will also be written to the file you specified in the Output File option.

<table>
<thead>
<tr>
<th>Table 29.1 Problem file COL7.BTC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>1 1 3 1 1 0 2 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>COL REP COLWS_1 COLWS_2 COLWS_3 CRIME INCOME HOUSING</td>
</tr>
</tbody>
</table>
29.4 Program Output

29.4.1 Standard Output

The first screen of output lists the estimates, their asymptotic standard deviations, associated z-values and corresponding probability levels, as illustrated in Table 29.2. Note that all coefficients are highly significant, including the spatial autoregressive parameter (listed as LAMBDA). Again, you will find some changes in the coefficient values compared to the results of the standard model. These differences should not be too large, since OLS is unbiased for this case. The main impact of the spatial error model should be on the standard errors (and thus on the indication of significance).

At the top of the table of results is some general information on the model, using the exact same format as for the spatial lag model. There are three pseudo $R^2$ measures of fit, which should be interpreted with caution. In contrast, the three likelihood based measures are comparable between models. Relative to the OLS estimates, the spatial error model achieves a higher likelihood (-183.4 vs -187.4 for OLS), which is to be expected, given the indications of the various diagnostics for spatial error dependence in the standard model and the high significance of $\lambda$. However, this fit is slightly inferior to that of the spatial lag model ($L=-182.4$), confirming the indication given by the LM tests in the standard model. The AIC is basically the same as for the spatial lag model (372.8) while the SC is slightly better (378.4 vs 380.3). The discrepancy between the two information criteria is due to the way in which the fit is penalized for additional parameters. Note that these measures do not count the spatial autoregressive coefficient in the spatial error model, while it is included in the AIC and SC for the spatial lag model.

The estimates are followed by a screen of diagnostics, as illustrated in Table 29.3. First are the two tests for heteroskedasticity. As is the case for the OLS regression output, the variables used in the heteroskedastic tests are listed only if you specified them explicitly. In the default case, the random coefficients model is used, as in the example in Table 29.3. It is evident that there still may be quite a bit of ignored heteroskedasticity in the model, which merits further attention.\footnote{The estimation of regression model with joint heteroskedasticity and spatial error dependence is not implemented in the current release of \textit{SpaceStat}.}

The second item in the table pertains to the LR test on the spatial autoregressive coefficient. The file name for the weights matrix is listed, as well as whether or not it has been row-standardized. The LR statistic of 7.99 corresponds to twice the difference between the log likelihood of the OLS and spatial lag models (2x3.997) and is highly significant. Note that the order of the Wald, LR and LM tests on the spatial lag is as expected:

\[ 17.61 > 7.99 > 5.72 \]
The value for the Wald test (17.61) is the square of the asymptotic t-test on $\lambda$ (4.20 in Table 29.2) and the value for the LM test (5.72) is from Table 26.6 in chapter 26.

The last two items are the results for the LR and Wald test on the common factor hypothesis. Here again, the order $W > LR$ (4.23 > 3.97) is satisfied. Neither are strongly significant, indicating no inherent inconsistency in the spatial error specification. If these statistics had been highly significant, the implication would be that the spatial error model is inappropriate. In this instance, this is not the case.

<table>
<thead>
<tr>
<th>Table 29.2 Maximum Likelihood Estimates in the Spatial Error Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPATIAL ERROR MODEL - MAXIMUM LIKELIHOOD ESTIMATION</td>
</tr>
<tr>
<td>DATA SET</td>
</tr>
<tr>
<td>DEPENDENT VARIABLE</td>
</tr>
<tr>
<td>R2</td>
</tr>
<tr>
<td>LIK</td>
</tr>
<tr>
<td>SIG-SQ</td>
</tr>
<tr>
<td>VARIABLE</td>
</tr>
<tr>
<td>CONSTANT</td>
</tr>
<tr>
<td>INCOME</td>
</tr>
<tr>
<td>HOUSING</td>
</tr>
<tr>
<td>LAMBDA</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 29.3 Specification Diagnostics for the Spatial Error Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>REGRESSION DIAGNOSTICS</td>
</tr>
<tr>
<td>DIAGNOSTICS FOR HETEROSEDASTICITY</td>
</tr>
<tr>
<td>RANDOM COEFFICIENTS</td>
</tr>
<tr>
<td>TEST</td>
</tr>
<tr>
<td>Breusch-Pagan test</td>
</tr>
<tr>
<td>Spatial B-P test</td>
</tr>
<tr>
<td>DIAGNOSTICS FOR SPATIAL DEPENDENCE</td>
</tr>
<tr>
<td>SPATIAL ERROR DEPENDENCE FOR WEIGHTS MATRIX COLWS_1 (row-standardized weights)</td>
</tr>
<tr>
<td>TEST</td>
</tr>
<tr>
<td>Likelihood Ratio Test</td>
</tr>
<tr>
<td>TEST ON COMMON FACTOR HYPOTHESIS</td>
</tr>
<tr>
<td>TEST</td>
</tr>
<tr>
<td>Likelihood Ratio Test</td>
</tr>
<tr>
<td>Wald Test</td>
</tr>
</tbody>
</table>

29.4.2 Long Output

If you have set the Long Output option to YES, the bisection search message will not appear, since you will see the intermediate results for each iteration of the search, as in Table
29.4. Note that the precision of the search and its stopping point are determined by the value for the \textbf{Convergence Criterion} option (6 in the \textbf{Options} menu). The iterations operate in the same fashion as for the spatial lag model. For each iteration, the new value for $\lambda$, the corresponding partial derivative for the concentrated log likelihood, estimate for the error variance $\sigma^2$, and the associated trace of an auxiliary matrix are listed.

\begin{table}[h]
\centering
\caption{Bisection for the Spatial Error Model}
\begin{tabular}{lcccc}
\hline
\multicolumn{1}{c}{} & \multicolumn{1}{c}{MAXIMUM LIKELIHOOD ESTIMATION OF SPATIAL ERROR MODEL} \\
& \multicolumn{1}{c}{(BISECTION METHOD)} \\
\hline
\textbf{STARTING VALUES} \\
Partial Derivative for Lambda=0: & 11.5462 \\
Lower and Upper Limits of Acceptable Parameter Range: \\
Lower: & -1.536177 & Upper: & 1.000000 \\
Lower and Upper Limits to Start Bisection: \\
Lower: & 0.000000 & Upper: & 0.999900 \\
\hline
\textbf{ITERATIONS} \\
Iter & Lambda & Dlik & Sig2 & tr \\
\hline
1 & 0.499950 & 2.384666 & 97.879644 & 7.451926 \\
2 & 0.749925 & -11.884295 & 90.497234 & 16.639800 \\
3 & 0.624937 & -3.061405 & 93.510347 & 10.944946 \\
4 & 0.562444 & -0.028351 & 95.551310 & 9.025731 \\
5 & 0.531197 & 1.246019 & 96.682622 & 8.203141 \\
6 & 0.546820 & 0.626896 & 96.108381 & 8.604719 \\
7 & 0.554632 & 0.303933 & 95.827651 & 8.812686 \\
8 & 0.558538 & 0.138975 & 95.688926 & 8.918559 \\
9 & 0.560491 & 0.055610 & 95.619979 & 8.971981 \\
10 & 0.561467 & 0.013704 & 95.585609 & 8.998815 \\
11 & 0.561956 & -0.007305 & 95.568451 & 9.012262 \\
12 & 0.561711 & 0.003204 & 95.577028 & 9.005536 \\
13 & 0.561833 & -0.002049 & 95.572739 & 9.008899 \\
14 & 0.561772 & 0.000578 & 95.574883 & 9.007217 \\
15 & 0.561803 & -0.000735 & 95.573811 & 9.008058 \\
16 & 0.561788 & -0.000079 & 95.574347 & 9.007637 \\
17 & 0.561780 & 0.000250 & 95.574615 & 9.007427 \\
18 & 0.561784 & 0.000086 & 95.574481 & 9.007532 \\
19 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
\hline
\end{tabular}
\end{table}

\begin{table}[h]
\centering
\caption{Bisection for the Spatial Error Model (continued)}
\begin{tabular}{lcccc}
\hline
\multicolumn{1}{c}{} & \multicolumn{1}{c}{MAXIMUM LIKELIHOOD ESTIMATION OF SPATIAL ERROR MODEL} \\
& \multicolumn{1}{c}{(BISECTION METHOD)} \\
\hline
\textbf{ITERATIONS} \\
Iter & Lambda & Dlik & Sig2 & tr \\
\hline
1 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
2 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
3 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
4 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
5 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
6 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
7 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
8 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
9 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
10 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
11 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
12 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
13 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
14 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
15 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
16 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
17 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
18 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
19 & 0.561786 & 0.000003 & 95.574414 & 9.007585 \\
\hline
\end{tabular}
\end{table}

In the \textbf{Long Output} case, the first screen of estimates is followed by the asymptotic variance matrix for the coefficients, as listed in Table 29.5. The matrix is organized in the same fashion as for the OLS estimates, except that two additional rows are included, i.e., one for the autoregressive coefficient $\lambda$ (referred to as LAMBDA), and one for the error variance $\sigma^2$ (referred to as SIGMA-SQ). Note that this matrix is block diagonal between the regression-
coefficients of the model and the nuisance parameters $\lambda$ and $\sigma^2$. The square roots of the elements on the diagonal yield the figures for the standard errors given in the previous table.

With the Long Output set to YES, a Report File will be created with as filename the one you specified in the problem file (REP), with the extension .DOC added to it. This file will contain an indicator variable (or observation sequence number if none was specified in theIndicator Variable option), the dependent variable, predicted values and residuals for the model.

Table 29.5 Asymptotic variance matrix for ML Estimates in Spatial Error Model

<table>
<thead>
<tr>
<th>COEFFICIENT VARIANCE MATRIX</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CONSTANT</td>
<td>28.7955</td>
<td>-0.998812</td>
<td>-0.124423</td>
<td>0.00000</td>
</tr>
<tr>
<td>INCOME</td>
<td>-0.998812</td>
<td>0.109275</td>
<td>-0.0135821</td>
<td>0.00000</td>
</tr>
<tr>
<td>HOUSING</td>
<td>-0.124423</td>
<td>-0.0135821</td>
<td>0.00818591</td>
<td>0.00000</td>
</tr>
<tr>
<td>LAMBDA</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.0179211</td>
</tr>
<tr>
<td>SIGMA-SQ</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>-0.629720</td>
</tr>
</tbody>
</table>

29.4.3 IDRISI Interface

If the Idrisi Interface option is set to YES, and irrespective of the settings for the Output to a File (Option 2) or Long Output (Option 3) options, then two special files are created, with as file names the name of the dependent variable with extensions _YP. VAL and _E. VAL. These files contain the values for the indicator variable in the first column (or the observation sequence number, if no Indicator Variable was specified) and respectively the predicted value and residual for the spatial error model. In the Columbus example, these files would be named CRIME_YP. VAL and CRIME_E. VAL. Note that this is the same filename as used for the OLS and spatial lag regression. Files with identical file names will be generated for all types of regression models. It is up to you to keep track of them. You may therefore want to change the file name for these files if you intend to keep several of them on the same directory (if you don’t change the names, only the latest result will be kept, the rest will be overwritten).

10. This is a standard result for regression with non-spherical error terms. See Anselin (1988a, Chapter 8) for further details.
29.5 Exercise

You may now use the same commands with the Problem File you constructed for the least squares regression exercise to estimate a spatial error model with your own data, or for the African vegetation index or Cliff-Ord Irish data. Pay particular attention to the extent of the change in parameter values for the model. You can compare the fit of the spatial error model to that of the spatial lag and least squares solution by means of the likelihood measures. You can also check whether the test on the common factor hypothesis is satisfied in your model.

References

CHAPTER 30

ESTIMATION OF THE HETEROSKEDASTIC ERROR MODEL

30.1 Introduction

The estimation of a model with heteroskedastic error terms is carried out in the Heterosked Err Model menu of the Regress module. Three different model specifications are allowed:

- **Generic Heteroskedasticity** (*R*-3-1 and *R*-3-2)
- **Groupwise Heteroskedasticity** (*R*-3-3 and *R*-3-4)
- **Random Coefficients** (*R*-3-5 and *R*-3-6)

These models differ in the way in which the heteroskedastic variables are constructed. For the **Generic** and **Groupwise** forms, the heteroskedastic variables must be specifically included in the Problem File. If they are not, an error message is generated. The difference between the two cases is the type of variable that is used to construct the heteroskedasticity. In the **Generic** case, any variable from the specified data set may be included. These variables are squared in the specification of the heteroskedastic error variance. In the **Groupwise** case, the heteroskedastic variable must be a categorical indicator variable that takes on integer values corresponding to the different “groups” or regimes in the data. In the **Generic** case, the heteroskedastic function always includes a constant term, while in the **Groupwise** case it does not. In the **Random Coefficients** model, the heteroskedastic variables are constructed as the squares of the explanatory variables in the model, and they need not be specified in the Problem File. If some heteroskedastic variables are specified, they are ignored.

For each of the three models, two estimation methods are included. The first is a two or three-step Feasible Generalized Least Squares (FGLS) estimator, the second an iterated form of FGLS, which is equivalent to maximum likelihood when the error terms are normally distributed. Diagnostics for spatial dependence are included for the ML estimates only.

Each function is started by entering the appropriate command sequence, followed by the name of the Problem File in batch mode (option 1), or by entering the requested information in interactive mode (option 2). A Problem File for the estimation of a heteroskedastic error model must contain the name of a dataset, the type of model specification, a dependent variable and at least one explanatory variable. For the **Generic** and **Groupwise** cases, one or more heteroskedastic variables must be specified. When a spatial weights matrix is included, diagnostics for spatial dependence will be computed for the ML estimates. For details on the creation of Problem Files in the Regress module, see Chapter 25.
30.2 Methodology

30.2.1 The Heteroskedastic Error Model

The heteroskedastic error model is a special case of a so-called non-spherical error model. The variance of the error term is no longer constant, but instead varies with each observation:

\[ y = X\beta + \varepsilon \]
\[ \text{Var}[\varepsilon_i] = \sigma_i^2 \]

or

\[ E[\varepsilon \varepsilon'] = \Omega \]

where, as before, \( y \) is a \( N \) by 1 vector of observations on the dependent variable, \( X \) is a \( N \) by \( K \) matrix of observations on the explanatory variables, \( \beta \) is a \( K \) by 1 vector of regression coefficients, \( \varepsilon \) is a \( N \) by 1 vector of error terms, with variance \( \sigma_i^2 \) for observation \( i \) and diagonal covariance matrix \( \Omega \).

In order to be identifiable, the non-constant variance must be given some structure. While there are many possibilities, one common specification is so-called additive heteroskedasticity, where the error variance is expressed as a linear function of a set of explanatory variables:

\[ \text{Var}[\varepsilon] = Z\gamma \]

where \( \text{Var}[\varepsilon] \) is a \( N \) by 1 column vector of the error variances, \( Z \) is a \( N \) by \( P \) matrix with the heteroskedastic variables as columns, and \( \gamma \) is a corresponding vector of coefficients. Typically, the first variable is a constant term, so that a simple test for heteroskedasticity may be formulated as a test on the joint significance of the other coefficients. The constant itself is then the homoskedastic error variance. This is the principle behind the Breusch-Pagan test for heteroskedasticity outlined in Chapter 26.

In the Generic and Random Coefficients specifications, the elements of the \( Z \) matrix are formed by the squares of the heteroskedastic variables. In the Random Coefficients model, these are the squares of the explanatory variables \( X \). The coefficients \( \gamma \) can then be interpreted as the variance associated with the corresponding random regression coefficient. As a consequence, this estimate should be positive, which may not be the case in practice.\(^1\)

In the Groupwise model, a number of regimes are considered, for which the error variance is different, but constant within each regime. As a result, the variance may be estimated
directly from the residuals for each regime, provided that enough observations are available. In SpaceStat, the regimes are constructed from a categorical indicator variable that takes on integer values. With each value corresponds a regime. There is no constant term in this heteroskedastic specification, in contrast to the approach taken for the two other models.

Feasible Generalized Least Squares (FGLS) Estimation

The principle behind FGLS estimation of the heteroskedastic error model is to obtain consistent estimates for the elements of the error variance $\Omega$. These estimates may then be used in the familiar FGLS estimator:

$$b_{FGLS} = [X' \Omega^{-1} X]^{-1} X' \Omega^{-1} y$$

For the Generic and Random Coefficients specifications, the FGLS estimates are obtained by means of a three-step procedure suggested by Amemiya (1977, 1985). This procedure starts with the residuals of an ordinary least squares regression, $e$. In the first step, the squared residuals $e^2$ are regressed on the heteroskedastic variables $Z$, to yield a first set of estimates $\gamma_1$:

$$\gamma_1 = (Z'Z)^{-1} Z' e^2$$

In the second step, the predicted error variances constructed by means of the $\gamma_1$ estimates are used to obtain a more efficient FGLS estimate in a regression of the squared residuals on the $Z$ variables:

$$\gamma_2 = [ZD^{-2}Z]^T Z D^{-2} e^2$$

where $D$ is a diagonal matrix with as elements $Z\gamma_1$. In the third step, the $\gamma_2$ estimates are used to construct a consistent estimate for $\Omega$, by taking the diagonal elements of $\Omega$ as $Z\gamma_2$. This $W$ matrix is then used to obtain the FGLS estimate.

In the Groupwise case, the FGLS estimation is much simpler. The OLS residuals are grouped corresponding to the regimes defined by the indicator variable and for each group $g$ the error variance $\sigma^2_g$ is estimated as $e_g'e_g/N_g$, where $e_g$ is a vector of residuals for group $g$ and $N_g$ is the number of observations in the group. The estimates for $\sigma^2_g$ are then substituted in the proper diagonal elements of $\Omega$ to obtain the FGLS estimate.

Maximum Likelihood Estimation

---

A detailed discussion of the random coefficients model is beyond the scope of this tutorial. Extensive treatments may be found in Judge et al. (1985) and Amemiya (1985), among others.
When the error terms are normally distributed, an iterated FGLS procedure can be shown to be equivalent to maximum likelihood. This principle is applied in SpaceStat for both the Amemiya and the groupwise estimators. The iteration consists of computing the residuals for the FGLS estimates in the previous step and to use these in the auxiliary regressions or groupwise variance estimates.

The iteration stops when either of two criteria are satisfied. The first is a minimax criterion of convergence of the \( b \) estimates. When the largest deviation between iterations is less than the Convergence Criterion specified in the options, the iterations end. The second criterion sets a maximum limit on the number of iterations, as specified by the Maximum Iterations option. This is to avoid problems with difficult convergence. When the estimation fails to converge, the results are listed for the last iteration, together with a warning message.

One problem that often occurs in the iterated FGLS estimation of the Generic and Random Coefficients models is that the error variance matrix fails to be positive definite for some values of the parameters. If this is the case, the iteration stops with an error message. Typically, this indicates a misspecification of the additive heteroskedastic model.

**Measures of Fit**

The standard \( R^2 \) measure of fit is not applicable to models with non-spherical errors. SpaceStat reports three pseudo-\( R^2 \) measures in its output. The first two are the same as in the spatial error model: the ratio of the variance of the predicted values over the variance of the observed values for the dependent variable (listed as R2) and the squared correlation between these two magnitudes (reported as Sq. Corr.). The third measure is an application of the adjustments suggested by Buse (1973). It is reported as R2(Buse) in the SpaceStat output. In principle, the Buse measures are comparable across models with non-spherical errors, but they are sometimes difficult to interpret.

Three additional measures of goodness-of-fit are based on the maximum likelihood framework: the log likelihood, the Akaike information criterion (AIC), and the Schwartz criterion (SC). These measures are only reported for the ML estimates (i.e., the iterated FGLS), but not for the FGLS.

**Hypothesis Tests**

As is the case for the spatial error model, statistical inference in the heteroskedastic error model is based on asymptotic considerations. The significance of the individual model

---

2. The formal log likelihood function for the heteroskedastic error model is of the form:
\[
L = -N/2(\ln 2\pi) - 0.5 \ln \det(\Omega) - 0.5 e' \Omega^{-1} e
\]
where \( \det \) is the determinant and the other notation is as before.

3. The expression for the Buse \( R^2 \) in the heteroskedastic case is:
\[
R^2_B = 1 - (e' \Omega^{-1} e) / (y_y)' / (y_y)
\]
with \( y_y = y - \Omega^{1/2} \Omega^{1/2} \) and \( y_y \) as a \( N \) by 1 vector of ones.
coefficients is based on a standard normal distribution (the asymptotic result) and not on the Student t distribution (as is the case for OLS). The so-called asymptotic t-test (which is actually a z-value since the limiting distribution is the standard normal) is obtained as the ratio of the estimate to its asymptotic standard error. \textit{SpaceStat} lists the estimates, their asymptotic standard errors, associated z-value and corresponding probability.

30.2.2 Specification Diagnostics

\textit{SpaceStat} includes a number of specification diagnostics to test the assumptions on which the estimation in the heteroskedastic error model is based. For the FGLS estimation, there is only a Wald (W) test against heteroskedasticity. For the ML estimation, there are both a Wald and Likelihood Ratio (LR) test on heteroskedasticity, and LM tests on spatial error and spatial lag dependence.

\textit{Heteroskedasticity}

The two tests against heteroskedasticity reported in the \textit{SpaceStat} output are a Wald test and a Likelihood Ratio test. For the \textit{Generic} and \textit{Random Coefficients} models, where the constant term may be identified with the homoskedastic variance, the Wald test consists of a test on the joint significance of the other coefficients in the heteroskedastic specification. In the \textit{Groupwise} model, the Wald test corresponds to a test on the equality of the variances in each group.

The Likelihood Ratio test corresponds to twice the difference between the log likelihood in the heteroskedastic error model and the log likelihood in a standard regression model with the same set of explanatory variables.

\textit{SpaceStat} reports each statistic as well as its degrees of freedom and corresponding probability according to a $\chi^2$ distribution.

\textit{Spatial Error Dependence}

\textit{SpaceStat} includes a Lagrange Multiplier test suggested in Anselin (1988b) for remaining spatial error dependence in the heteroskedastic model. The test is asymptotic and is distributed as a $\chi^2$ variate with one degree of freedom. The statistic, its degrees of freedom (always 1) and corresponding probability are given.\footnote{5}

---

\footnote{4. The variance matrix for the $b$ estimates is $\left( X' \Omega^{-1} X \right)^{-1}$, with the estimates substituted in the $\Omega$ matrix. The variance matrix for the $\gamma$ coefficients in the Amemiya procedure is $\left( 2(ZD^{-1}Z) \right)^{-1}$, where the diagonal matrix D is based on the $\gamma$ estimates. The variance for the estimates of groupwise heteroskedasticity is $2\sigma_g^2 / N_g$.

5. Formally, the LM test is:
\[ LM_{stat} = \left( e' \Omega^{-1} W e \right) / tr(W'W + W') \]
where $e$ are the residuals in the iterated FGLS estimation, $\Omega$ is the estimated error variance matrix and tr is the matrix trace operator. See Anselin (1988a, Chapter 8) and Anselin (1988b) for technical details.}
**Spatial Lag Dependence**

A Lagrange Multiplier test for a spatial lag in the heteroskedastic error model is included as well. This test follows from the general principles outlined in Anselin (1988b). It is also asymptotic and is distributed as a $\chi^2$ variate with one degree of freedom. The statistic, its degrees of freedom (always 1) and corresponding probability are given.\(^6\)

### 30.3 Example Problem File

As an example, you will continue the analysis of the regression model that relates the CRIME variable in the Columbus data set to a constant term and the explanatory variables INCOME and HOUSING. Make sure that the data set (the files COL.DAT and COL.DHT) is on your current directory. You will need to construct a new Problem File, that includes the variable EW as the heteroskedastic variable. This is the case for the file COL9.BTC, listed below in Table 30.1 and included on the \EXAMPLES directory.

You can use this Problem File to estimate a Generic heteroskedastic model as well as the Groupwise specification. The difference between the two is that in the former, a constant term is the estimate for the overall homoskedastic error variance, while there is no constant term in the latter. You can use the Problem File COL7.BTC to estimate a Random Coefficients model. You start the estimation of these models by entering the proper command sequence, followed by `col9.btc` or `col7.btc` as the name for the Problem File. In what follows, I will illustrate results for both FGLS and ML estimation.

#### Table 30.1 Problem file COL9.BTC

<p>| | | | | | | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

**COL**  **REP**  **COLWS_1**  **CRIME**  **INCOME**  **HOUSING**  **EW**

### 30.4 Program Output

#### 30.4.1 Standard Output for Generic Heteroskedasticity

After you enter the name of the Problem File, a screen with intermediate results briefly appears, which lists the estimates obtained in the first and second steps of the Amemiya FGLS estimation.

---

\(^6\) Formally, the LM test is:

$$LM_{LAG} = (e'\Omega^{-1}W^e)'[D + tr(WW + W^e)]$$

with

$$D = (WXb)'\Omega^{-1}(WXb) - (WXb)'\Omega^{-1}(X'\Omega^{-1}X)^{-1}X'\Omega^{-1}(WXb)$$

where $e$ are the residuals in the iterated FGLS estimation, $\Omega$ is the estimated error variance matrix, WXb are the spatially lagged predicted values, and $tr$ is the matrix trace operator. This test is a straightforward extension of the spatial lag test outlined in Chapter 26.
estimation. For small models such as the Columbus example, these results disappear too quickly to allow a careful consideration. However, they are written to the Output File, if this option is set.

The first screen of output lists the estimates for the regression coefficients, their asymptotic standard deviations, associated z-values and corresponding probability levels, as illustrated in the top part of Table 30.2 for the FGLS estimation. Note that the main difference with the OLS results in Table 26.3 are the standard deviations of the estimates. These are smaller for the FGLS, indicating the lack of efficiency of OLS when heteroskedasticity is present (the latter was also suggested by the various diagnostics in Table 26.5). The next screen, shown as the bottom part of Table 30.2 for the FGLS case, lists the estimates for the heteroskedastic coefficients, their standard deviations, z-values and associated probability levels. The estimate for the CONSTANT is the overall error variance. The value of 202.07 is considerably higher than the ML estimate for error variance in Table 26.3 (122.75). However, the second coefficient indicates that the error variance is substantially lower (-134.02) in the locations that correspond to a value of EW of 1. Both heteroskedastic coefficients are significant, although the EW coefficient is only weakly so (p=0.04). The only diagnostic given with FGLS estimation is a Wald test on heteroskedasticity. The value of 4.080 listed at the very bottom of Table 30.2 is nothing but the square of the estimate for EW (-2.01982), and has the same probability.

The results for an iterated FGLS estimation of the same model are listed in Table 30.3. Using the default value for the convergence criterion, the estimation process stops after 6 iterations, as indicated above the coefficient estimates. The same format is followed as for FGLS. The estimates differ slightly from the ones in Table 30.2, but the standard errors are consistently smaller. The estimate for the overall error variance is now 223.87 with a coefficient of -164.18 for EW. The latter is strongly significant (p=0.02). For the iterated FGLS, there are diagnostics for spatial dependence, as illustrated in Table 30.4. Following the Wald test on the coefficient of EW (5.13) and an LR test on heteroskedasticity (8.89, p<0.01), the results for the LM tests on spatial error dependence and spatial lag dependence are given. Both of these indicate that substantial problems of spatial dependence remain present, even after heteroskedasticity is taken into account.

In the header of the listing of results is some general information on the model, using the same format as for the standard regression model. For both FGLS and iterated FGLS, there are three pseudo R\textsuperscript{2} measures of fit, which should be interpreted with caution. For the iterated FGLS results, the usual three likelihood based measures are listed as well. The log likelihood of -182.9 is better than the -187.4 for OLS (compare to Table 26.3), slightly inferior to the result for the spatial lag model (-182.4 in Table 27.2), but better than the result for the spatial error model (-183.4 in Table 29.2).
30.4.2 Standard Output for Groupwise Heteroskedasticity

The output for the Groupwise model follows the same format as in the Generic model. The only difference is in the estimates for the heteroskedastic coefficients. In the Groupwise model, each regime has an estimate for its error variance, as illustrated in Table 30.5 for the iterated FGLS results. The error variance in the locations with \( EW = 0 \) is estimated to be 223.87,
while the variance in the other locations is 59.69. Note that the difference between the two variances (164.18) corresponds exactly to the estimate of the EW coefficient in the **Generic** model, listed in Table 30.3. The regression coefficients, standard errors and diagnostics are identical for the two models.

Table 30.4 Diagnostics for the Generic Heteroskedastic Error Model

<table>
<thead>
<tr>
<th>REGRESSION DIAGNOSTICS</th>
</tr>
</thead>
<tbody>
<tr>
<td>TESTS ON HETEROSKEDASTICITY</td>
</tr>
<tr>
<td>TEST</td>
</tr>
<tr>
<td>Wald test</td>
</tr>
<tr>
<td>LR test</td>
</tr>
</tbody>
</table>

Table 30.5 Iterated FGLS Estimates for the Error Variance in the Groupwise Heteroskedastic Error Model

<table>
<thead>
<tr>
<th>HETEROSKEDASTIC COEFFICIENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>VARIABLE</td>
</tr>
<tr>
<td>EW</td>
</tr>
<tr>
<td>EW_1</td>
</tr>
</tbody>
</table>

### 30.4.3 Standard Output for Random Coefficients

The output for the **Random Coefficients** model follows the same format as above. The only difference is that the heteroskedastic coefficients correspond to the explanatory variables in the model. The results of the FGLS estimation of the model in **Problem File** COL7.BTC are listed in Table 30.6. The heteroskedastic coefficient of INCOME is negative and strongly significant, while the coefficient of HOUSING is not significant. The negative coefficient of INCOME conflicts with its usual interpretation as the variance of the random regression coefficient. This particular specification will also lead to problems in the iterated FGLS estimation. After one iteration, the estimation process aborts with an error message indicating that the estimated variance matrix is not positive definite.
30.4.4 Long Output

When the Long Output option is set to YES, the results for each iteration in the iterated FGLS are listed to the screen and written to the Output File. Also, similar to the format used for the other regression models, the complete asymptotic variance matrix is listed, for both the regression and the heteroskedastic coefficients. In addition, the observed, predicted values and residuals will be listed to the screen and written to the Report File.

30.4.5 IDRISI Interface

If the Idrisi Interface option is set to YES, and irrespective of the settings for the Output to a File (Option 2) or Long Output (Option 3) options, then two special files are created, with as file names the name of the dependent variable with extensions _YP.VAL and _E.VAL. Similar to the format used for the other regression models, these files contain the values for the indicator variable in the first column (or the observation sequence number, if no Indicator Variable was specified) and respectively the predicted value and residual for the spatial error model.

30.5 Exercise

You may now further experiment with specification for a heteroskedastic model with your own data, or for the African vegetation index or Cliff-Ord Irish data. Pay particular attention to the extent of the change in parameter values for the model. When you use iterated FGLS, you can compare the fit of the heteroskedastic models to that of the spatial lag and spatial error models.

Table 30.6 FGLS Estimates for the Random Coefficients Model

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>COEFF</th>
<th>S.D.</th>
<th>z-value</th>
<th>Prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONSTANT</td>
<td>62.6087</td>
<td>3.32753</td>
<td>18.815375</td>
<td>0.000000</td>
</tr>
<tr>
<td>INCOME</td>
<td>-1.2252</td>
<td>0.229287</td>
<td>-5.343510</td>
<td>0.000000</td>
</tr>
<tr>
<td>HOUSING</td>
<td>-0.248179</td>
<td>0.0986935</td>
<td>-2.514647</td>
<td>0.011915</td>
</tr>
</tbody>
</table>

HETEROSKEDASTIC COEFFICIENTS

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>COEFF</th>
<th>S.D.</th>
<th>z-value</th>
<th>Prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONSTANT</td>
<td>126.674</td>
<td>28.0194</td>
<td>4.520937</td>
<td>0.000006</td>
</tr>
<tr>
<td>INCOME</td>
<td>-0.178881</td>
<td>0.0551075</td>
<td>-3.246030</td>
<td>0.001170</td>
</tr>
<tr>
<td>HOUSING</td>
<td>0.0102752</td>
<td>0.010037</td>
<td>1.023730</td>
<td>0.305963</td>
</tr>
</tbody>
</table>
References

CHAPTER 31

TREND SURFACE MODEL

31.1 Introduction

A trend surface model is a special regression model that has as its explanatory variables the elements of a polynomial in the coordinates of the observations, say x and y. These coordinates correspond to data points, or represent meaningful points associated with an areal unit, such as a centroid. For example, a second order trend surface model would have the following specification:

\[ z = \alpha + \beta_1 x + \beta_2 y + \beta_3 x^2 + \beta_4 y^2 + \beta_5 xy + \epsilon \]

where, in contrast to the notational convention used in the rest of this tutorial, the dependent variable is not represented by y, but by the symbol z. The explanatory variables x and y correspond to the coordinates. In addition, \( \alpha \) is the constant term, the \( \beta_i \) are regression coefficients for each term in the polynomial and \( \epsilon \) is a random error. Except for the particular choice of explanatory variables, the trend surface model is otherwise treated exactly like any other regression specification. You invoke the estimation of a trend surface model by means of the command sequence for a particular regression model, either OLS, spatial error, heteroskedastic error, or spatial lag.

A trend surface model is particularly useful if you wish to filter out large scale spatial trends in order to focus your attention to the smaller scale variation, i.e., on the residuals of the trend surface regression. Another common application of this model is to obtain spatial interpolations. Since the trend surface is only a function of the absolute location of points (i.e., their coordinates), predicted values can be obtained for any location for which the coordinates are known. This can be exploited to produce a variety of smoothed maps.

The Problem File for a trend surface regression must contain the name of the data set, the name of the dependent variable and one variable name each for the x and y coordinates (a constant term is optional, but recommended). If you wish to test for spatial dependence or estimate one of the spatial regression models, you must also include a file name for at least one weights matrix.

Three additional features distinguish the treatment of a trend surface regression from that of a generic model. First, the model type must be identified by the first flag in the problem file flag list. For a trend surface, this flag should take on a value of 2.1. Secondly, you must specify the order for the polynomial that will be used in the trend surface. This is the 13th
element in the list of flags in the problem file. This flag must be non-zero, or else an error message will be generated. Thirdly, the default for the heteroskedastic variables in the Breusch-Pagan and Koenker-Bassett tests is no longer the random coefficient model, but a linear specification in the squares of the $x$ and $y$ coordinates (and a constant). If instead you want to implement the tests for the random coefficient hypothesis, you must therefore specify all heteroskedastic variables explicitly. Also note that the White test against heteroskedasticity often cannot be computed for the trend surface model, due to the loss in degrees of freedom.

The powers and cross products for the terms in the trend surface need not be specified as explanatory variables, since they are computed internally by SpaceStat. The variable names used for the powers and cross-products in the output are $XX_i$, $YY_i$ and $XX_iYY_j$, where the $XX$ corresponds to the $x$ coordinate, the $YY$ to the $y$ coordinate and the $i,j$ are integer powers.\(^2\)

### 31.2 Methodology

The same estimation methods, diagnostics and interpretations apply as for the generic regression discussed up to this point. In other words, you can estimate a trend surface as a standard OLS regression, or you can implement a spatial lag, spatial error or one of the heteroskedastic models.

One issue to keep in mind when interpreting the results of a trend surface regression is that there is likely to be a high degree of multicollinearity. This is due to the strong functional relation between the various terms in the polynomial. As a consequence, the indication of significance (t-values) and fit ($R^2$) may be suspect, as discussed in more detail in Chapter 26. It is generally not a very good idea to use a trend surface model for anything besides simple smoothing, filtering or interpolation of data.

### 31.3 Example Problem File

As an illustration of a trend surface specification, you can now re-analyze the Columbus crime data in function of a second order polynomial in the $x$ and $y$ coordinates. The coordinates are included as variables $X$ and $Y$ in the COL data set. In order to run the example, you must have this data set on your current directory (COL.DAT and COL.DHT), as well as the weights matrix COLWS_1.FMT. A Problem File to carry out this regression is listed in Table 31.1. It is also contained on the \EXAMPLES directory as file COL10.BTC. For this example,

---

1. Note that for a problem file constructed in the Explore module, the default value for this flag will be 0. This is interpreted as a generic regression model in the Regress module.

2. You can also create the various powers and cross-products explicitly, by means of the SpaceStat Data-Var Algebra-Trend Surface command (D-6-7). The variable names for the powers and cross-products will be the same as in the regression command. If you construct the terms of the polynomial explicitly, and include them in the problem file for a regression, you cannot use the Trend Surface model form, since all but the linear terms will be ignored in the regression. Instead, you must treat this as a generic model and set the model flag to 1 (first flag).
I will use the **Classic Model OLS** regression. You start this by typing the `R-1-1` command sequence, followed by `col10.btc` in response to the prompt for the **Problem File** in the **Batch** option.

### Table 31.1 Problem File for the Trend Surface Regression

<table>
<thead>
<tr>
<th>COL</th>
<th>REP</th>
<th>COLWS_1</th>
<th>CRIME</th>
<th>X</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**31.4 Program Output**

The output follows the same format as outlined for the generic regression example. The first screen shows the estimates, as illustrated in Table 31.2. All except the cross product term in the polynomial are highly significant, and the model fit is only slightly worse than that achieved in the regression with INCOME and HOUSING as explanatory variables. Note that this worse fit is more emphasized in the adjusted $R^2$ (0.46 vs. 0.53) which penalizes for the additional explanatory variables (compare to the unadjusted $R^2$ of 0.52 vs 0.55). The more interesting aspects of this model are the results for the diagnostics, as listed in Table 31.3. The very high score for the multicollinearity condition number (372.2) is characteristic for trend surface models (or any polynomial regression) as pointed out before. The residuals for the trend surface model are also no longer normally distributed. As a result, the test listed for heteroskedasticity is the Koenker-Bassett test (which is robust against non-normality) and not the Breusch-Pagan test shown in Chapter 26. In contrast to what you found for the standard regression, there is no evidence for heteroskedasticity here. Due to the lack of degrees of freedom, the White test is not computed. The tests for spatial dependence illustrate the unreliability of the Moran’s I test. It is the only one out of the four tests for spatial dependence listed to show a significant value. Rather than to attribute this to a superior power to detect spatial error autocorrelation, this is likely due to the sensitivity of Moran’s I to other forms of misspecification, notably the non-normality indicated by the Kiefer-Salmon test.

**31.5 Exercise**

You can now easily implement a trend surface regression using your own data or one of the other data sets provided. Of course, you must have the values for the x and y coordinates associated with the spatial units of observation contained in the data set. You may also want to go further than in the example above and also implement a spatial lag, spatial error, or heteroskedastic model, if warranted by the regression diagnostics.
### Table 31.2 OLS Estimates for Trend Surface Regression

| DATA SET | COL | DEPENDENT VARIABLE | CRIME | OBS 49 | VARS 6 | DF 43 | R2          | R2-adj      | LIK        | AIC         | SC         | RSS         | F-test      | Prob        | SIG-SQ      | SIG-SQ(ML)   |
|----------|-----|--------------------|-------|--------|--------|-------|------------|-------------|-----------|-------------|------------|------------|-------------|------------|------------|-------------|-------------|
|          |     |                    |       |        |        |       | 0.5176     | 0.4615      | -189.211  | 390.422     | 401.773    | 6482.40    | 9.22796     | 5.04058E-06 | 150.753     | 12.2782     |

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>COEFF</th>
<th>S.D.</th>
<th>t-value</th>
<th>Prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONSTANT</td>
<td>-487.22</td>
<td>126.843</td>
<td>-3.84119</td>
<td>0.000388</td>
</tr>
<tr>
<td>X</td>
<td>12.8906</td>
<td>3.07301</td>
<td>4.194787</td>
<td>0.000124</td>
</tr>
<tr>
<td>Y</td>
<td>17.6748</td>
<td>6.5797</td>
<td>2.686266</td>
<td>0.010219</td>
</tr>
<tr>
<td>XX_2</td>
<td>-0.171987</td>
<td>0.0389511</td>
<td>-4.415463</td>
<td>0.000057</td>
</tr>
<tr>
<td>YY_2</td>
<td>-0.276019</td>
<td>0.0730785</td>
<td>-3.777016</td>
<td>0.000473</td>
</tr>
<tr>
<td>XX_1YY_1</td>
<td>0.0093572</td>
<td>0.0823689</td>
<td>0.113601</td>
<td>0.910082</td>
</tr>
</tbody>
</table>

### Table 31.3 Diagnostics in the Trend Surface Regression

<table>
<thead>
<tr>
<th>REGRESSION DIAGNOSTICS</th>
</tr>
</thead>
<tbody>
<tr>
<td>MULTICOLLINEARITY CONDITION NUMBER</td>
</tr>
<tr>
<td>TEST ON NORMALITY OF ERRORS</td>
</tr>
<tr>
<td>TEST</td>
</tr>
<tr>
<td>Kiefer-Salmon</td>
</tr>
<tr>
<td>DIAGNOSTICS FOR HETEROSKEDASTICITY</td>
</tr>
<tr>
<td>LINEAR SPECIFICATION USING VARIABLES</td>
</tr>
<tr>
<td>CONSTANT</td>
</tr>
<tr>
<td>TEST</td>
</tr>
<tr>
<td>Koenker-Bassett test</td>
</tr>
<tr>
<td>DIAGNOSTICS FOR SPATIAL DEPENDENCE</td>
</tr>
<tr>
<td>FOR WEIGHTS MATRIX COLWS_1 (row-standardized weights)</td>
</tr>
<tr>
<td>TEST</td>
</tr>
<tr>
<td>Moran's I (error)</td>
</tr>
<tr>
<td>Lagrange Multiplier (error)</td>
</tr>
<tr>
<td>Kelejian-Robinson (error)</td>
</tr>
<tr>
<td>Lagrange Multiplier (lag)</td>
</tr>
</tbody>
</table>
CHAPTER 32

SPATIAL REGIMES

32.1 Introduction

In many instances, the assumption of a fixed relation between the explanatory variables and the dependent variable that holds over the complete dataset is not tenable. Instead, heterogeneity may be present, in the form of different intercepts and/or slopes in the regression equation for subsets of the data.\(^1\) This is often referred to as structural instability or structural change in the econometric literature, and may be expressed in the form of switching regression models, originally suggested by Quandt (1958, 1972). When the different subsets in the data correspond to regions or spatial clusters, I refer to a switching regression specification as Spatial Regimes (see Anselin, 1988, Chapter 9).

For example, if two regimes were distinguished, following the value of an indicator (dummy) variable \(d\), the constant term and slope coefficients would take on two different sets of values, depending on the regime:

\[
y_1 = \alpha_1 + X_1\beta_1 + \epsilon_1 \quad \text{for } d = 0
\]
\[
y_2 = \alpha_2 + X_2\beta_2 + \epsilon_2 \quad \text{for } d = 0
\]

where \(y_1\) and \(X_1\) are subsets of the dependent and explanatory variables corresponding to the first regime, and \(y_2\) and \(X_2\) are those for the second regime, with the \(\alpha_1\), \(\alpha_2\), and \(\beta_1\), \(\beta_2\) as matching regression coefficients, and the \(\epsilon_1\), \(\epsilon_2\) as error vectors.

In SpaceStat, models of spatial regimes are implemented by jointly estimating the coefficients for both regimes. An augmented matrix of observations on the explanatory variables is constructed, of dimension \(N\) by \(MK\) (with \(M\) as the number of regimes), by transforming each explanatory variable into as many new variables as there are regimes. The new variables are zero for all observations that do not fall in the regime to which they correspond.\(^2\)

In all other respects, a model with spatial regimes is treated as a regression model, allowing the full range of estimation methods (OLS, spatial error, heteroskedastic error and spatial lag). The proper Problem File must contain the data set filename, the variable name for the dependent variable and at least one explanatory variable, and the variable name for a categorical indicator variable. This indicator variable may only take on consecutive integer

---

1. Different intercepts may be easily taken into account by including dummy variables in the regression specification.
2. This is the same convention as taken in the Spatial Regimes command of the Data-Var Algebra menu (D-6-8).
values, with each value corresponding to a regime. **SpaceStat** internally allocates the observations to their proper regimes. If you wish to consider spatial effects, the filename of at least one spatial weights matrix must be included as well. You start the estimation of a spatial regimes model by means of the command sequence for one of the regression methods.

To ensure the proper treatment of the **Spatial Regimes** model, the first flag in the flag list of the **Problem File** must be set to 3 (instead of the default of 1 for a generic model or 0 for the **Explore** module).

You may also create the variables corresponding to each regime explicitly, by means of the **Data-Var Algebra-Spatial Regimes** (D-6-8) command sequence. However, you must then use the generic regression model (option 1) and will not be able to carry out the tests on structural stability that are part of the regimes specification.

### 32.2 Methodology

#### 32.2.1 Test for Structural Instability

Since the **Spatial Regimes** specification is treated as a standard regression model, the full range of estimation methods and specification diagnostics are carried out, as outlined in Part V. In addition, a test is implemented on the stability of the regression coefficients over the regimes. This is a test on the null hypothesis that the coefficients are the same in all regimes, e.g., for the two-regime case:

\[ H_0: \alpha_1 = \alpha_2 \text{ and } \beta_1 = \beta_2 \]

This test is implemented for all coefficients jointly, as well as for each coefficient separately.

In the classic regression model, this is the familiar Chow (1960) test on the stability of regression coefficients. The Chow statistic is distributed as an F variate with K,N-MK degrees of freedom (M as the number of regimes). It has been extended to spatial models in Anselin (1990) in the form of a so-called spatial Chow test. In both spatial models, but also for robust and heteroskedastic regression (see, e.g., White, 1980, and MacKinnon and White, 1985), the test is based on an asymptotic Wald statistic, distributed as \( \chi^2 \) with \((M-1).K\) degrees of freedom.3

**SpaceStat** lists the statistic, its degrees of freedom and associated probability level, for both the joint test and the tests on each individual coefficient.

---
3. In its most general form, the test for two regimes may be expressed as a test on the null hypothesis \( H_0: g^\prime \hat{\beta} = 0 \), where \( \hat{\beta} = [\hat{\beta}_0; \hat{\beta}_1] \) is a stacked vector of all the regression coefficients (including the constant terms) and \( g^\prime \) is a K by 2K matrix \([I_K; -I_K]\), with \( I_K \) as a K by K identity matrix. The corresponding Wald test is of the form:

\[
W = (g^\prime \hat{b})(g^\prime \text{var}(b))^{-1}g^\prime (g^\prime \hat{b})
\]

where \( \hat{b} \) are the estimates for the regression coefficients and \( \text{var}(b) \) is the corresponding (asymptotic) variance matrix. Simpler expressions may be derived for the standard regression (Chow test) and the spatial error model (i.e., the spatial Chow test in Anselin, 1990).
32.3 Example Problem File

To practice the spatial regimes model, you may use the Problem File listed in Table 32.1. This is also contained as the file COL11.BTC on the \EXAMPLES directory. Note that the Problem File includes EW twice, once as the indicator variable for structural change (flag 9) and once as the indicator variable for groupwise heteroskedasticity (flag 11). This time, you will first estimate the model as a groupwise heteroskedastic regression model. This takes into account the possibility of not only having different regression coefficients in each regime determined by EW, but different error variances as well. Type R-3-4 to invoke the ML estimation of this model, and enter col11.btc in response to the prompt for the Problem File.

<table>
<thead>
<tr>
<th>Table 32.1 Problem File for Spatial Regimes</th>
</tr>
</thead>
</table>
| 1
| 3 1 1 1 1 0 2 0 1 0 1 0 0 0 0
| COL REP COLWS_1 CRIME INCOME HOUSING EW EW

32.4 Program Output

32.4.1 Groupwise Heteroskedastic Spatial Regimes

The estimates of the coefficients in the spatial regimes model follow the general format for the method by which they are obtained. As illustrated in Table 32.2, the estimates are listed for the coefficients in each regime, with the regime designated by a _i postfix. For each coefficient, the usual information is given. The header contains measures of fit and an indication of which variable was used to determine the regimes. Note that the fit of the model is only slightly better than that of the specification without regimes (compare to the results in Table 30.3): while the log likelihood of -180.47 (vs. -182.93) shows a marginally better fit, the AIC of 372.94 (vs. 372.86) indicates that this may not be sufficient for the decrease in degrees of freedom. The coefficients have the same signs in both regimes, but HOUSING is no longer significant in the second regime (while it remains significant in the first).

32.4.2 Test for Structural Stability

The extent to which the differences between coefficients in the two regimes is statistically significant is indicated by the results of the tests in Table 32.3. The null hypothesis on the joint equality of coefficients cannot be rejected by the Chow-Wald test. Its value of 5.35 is not extreme for a $\chi^2$ distribution with 3 degrees of freedom. The same indication is provided by the tests on the individual coefficients, except for HOUSING (5.16 for a $\chi^2$ with 1 degree
of freedom yields $p=0.02$). In other words, there is a significant difference in the relation between HOUSING and CRIME in each of the regimes defined by the EW variable.

### Table 32.2 Groupwise Heteroskedastic Regression of Spatial Regimes

<table>
<thead>
<tr>
<th>HETEROSKEDASTIC ERROR MODEL (GROUPWISE)</th>
<th>ITERATED FGLS ESTIMATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>STRUCTURAL CHANGE - DIFFERENT SLOPES FOLLOWING VARIABLE</td>
<td>EW</td>
</tr>
<tr>
<td>DATA SET</td>
<td>COL</td>
</tr>
<tr>
<td>DEPENDENT VARIABLE</td>
<td>CRIME</td>
</tr>
<tr>
<td>R2</td>
<td>0.6029</td>
</tr>
<tr>
<td>LIK</td>
<td>-180.471</td>
</tr>
</tbody>
</table>

Convergence after 1 iterations

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>COEFF</th>
<th>S.D.</th>
<th>z-value</th>
<th>Prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONST_1</td>
<td>76.6494</td>
<td>9.85367</td>
<td>7.778775</td>
<td>0.000000</td>
</tr>
<tr>
<td>INCOM_1</td>
<td>-1.45524</td>
<td>0.605153</td>
<td>-2.404747</td>
<td>0.016184</td>
</tr>
<tr>
<td>HOUSI_1</td>
<td>-0.545486</td>
<td>0.189878</td>
<td>-2.872823</td>
<td>0.004068</td>
</tr>
<tr>
<td>CONST_2</td>
<td>67.2942</td>
<td>3.96016</td>
<td>16.992780</td>
<td>0.000000</td>
</tr>
<tr>
<td>INCOM_2</td>
<td>-2.01405</td>
<td>0.313226</td>
<td>-6.430021</td>
<td>0.000000</td>
</tr>
<tr>
<td>HOUSI_2</td>
<td>-0.0637665</td>
<td>0.0943107</td>
<td>-0.676132</td>
<td>0.498957</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HETEROSKEDASTIC COEFFICIENTS</th>
<th>VARIABLE</th>
<th>COEFF</th>
<th>S.D.</th>
<th>z-value</th>
<th>Prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>EW</td>
<td>182.803</td>
<td>57.8075</td>
<td>3.162278</td>
<td>0.001565</td>
<td></td>
</tr>
<tr>
<td>EW_1</td>
<td>57.9317</td>
<td>15.2136</td>
<td>3.807887</td>
<td>0.000140</td>
<td></td>
</tr>
</tbody>
</table>

### Table 32.3 Test on Structural Stability

<table>
<thead>
<tr>
<th>TEST ON STRUCTURAL INSTABILITY FOR 2 REGIMES DEFINED BY EW</th>
<th>DF</th>
<th>VALUE</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chow – Wald</td>
<td>3</td>
<td>5.350547</td>
<td>0.147856</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>STABILITY OF INDIVIDUAL COEFFICIENTS</th>
<th>TEST</th>
<th>DF</th>
<th>VALUE</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONST_1</td>
<td>1</td>
<td>0.776048</td>
<td>0.378353</td>
<td></td>
</tr>
<tr>
<td>INCOM_1</td>
<td>1</td>
<td>0.672525</td>
<td>0.412173</td>
<td></td>
</tr>
<tr>
<td>HOUSI_1</td>
<td>1</td>
<td>5.162697</td>
<td>0.023077</td>
<td></td>
</tr>
</tbody>
</table>

### 32.4.3 Diagnostics for Spatial Dependence

As part of the standard set of diagnostics provided for the groupwise heteroskedastic model, tests for a spatial lag and spatial error dependence are carried out. The results, listed in Table 32.4, provide strong evidence for both, with an edge towards the spatial lag. Consequently, the proper analysis of the effects of spatial regimes should include a spatial lag in the model.
32.5 Exercise

You can now continue this analysis of the Columbus example, and explore the effect of using different specifications upon the evidence of structural instability. In particular, you should consider a model with a spatial lag, estimated by means of ML or IV/Bootstrap. To implement the latter, a Problem File with instrumental variables is contained as the file COL11A.BTC on the \EXAMPLES directory. You will notice how the indication of structural stability changes once the spatial dependence is taken into account. Of course, you may also compare this to the results of the classic OLS regression model with the standard Chow test.

Instead, you may want to carry out an analysis of spatial regimes for your own data set or one of the other example data sets. Note that neither the Irish data set nor the African one include dummy variables. To implement spatial regimes models, you may need to first create a relevant indicator variable and add it to the existing data set.

References


<table>
<thead>
<tr>
<th>Table 32.4 Diagnostics for Spatial Dependence in Spatial Regimes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DIAGNOSTICS FOR SPATIAL DEPENDENCE</strong></td>
</tr>
<tr>
<td>FOR WEIGHTS MATRIX COLWS_1 (row-standardized weights)</td>
</tr>
<tr>
<td>TEST</td>
</tr>
<tr>
<td>Lagrange Multiplier (error)</td>
</tr>
<tr>
<td>Lagrange Multiplier (lag)</td>
</tr>
</tbody>
</table>
CHAPTER 33

SPATIAL EXPANSION METHOD

33.1 Introduction

Instead of following discrete regimes, as in the previous chapter, the variability or spatial heterogeneity in regression coefficients may also be expressed in the form of continuous variation. In the Spatial Expansion Method, originally suggested by Casetti (1972), the continuous drift in the parameters is formulated in function of a set of auxiliary variables, the so-called expansion variables. Recently, this expansion paradigm has been extended to become a general framework for model development (e.g., Casetti, 1986; Casetti and Jones, 1988; Jones and Casetti, 1992).

In a basic implementation of the expansion method, each regression coefficient, say $b_k$, becomes a linear function of a set of expansion variables, say $z_1, z_2, \ldots, z_m$:

$$\beta_k = \gamma_0 + \gamma_1 z_1 + \gamma_2 z_2 + \ldots + \gamma_m z_m$$

After substitution of this expression in the original (initial) model, $m$ new variables result, consisting of the product of $x_k$ (the variable associated with $b_k$ in the initial model) with each of the expansion variables, e.g., $x_k z_1, x_k z_2$, etc. The expansion variables may be any set of variables, including polynomial expressions such as a trend surface.

In SpaceStat, a model that incorporates the spatial expansion method is treated as a regression model, allowing the full range of estimation methods (OLS, spatial error, heteroskedastic error and spatial lag) and diagnostics. The proper Problem File must contain the data set filename, the variable name for the dependent variable and at least one explanatory variable (initial variable), the variable name for at least one expansion variable, and the order for the polynomial expansion (linear or quadratic). SpaceStat creates the expanded variables (i.e., the products of the initial variables with the expansion variables) internally. If you wish to consider spatial effects, the filename of at least one spatial weights matrix must be included as well. You start the estimation of a spatial expansion model by means of the command sequence for one of the regression methods.

To ensure the proper treatment of the Spatial Expansion model, the first flag in the flag list of the Problem File must be set to 4 (instead of the default of 1 for a generic model or 0 for the Explore module).

You may also create the expansion variables explicitly, by means of the Data-Var Algebra-Expansion (D-6-6) command sequence. However, you must then use the generic regression
model (option 1) and will not be able to carry out the tests on coefficient drift that are part of the expansion method specification.

The specification of an expanded regression model may lead to two types of problems that merit some attention. When many expansion variables are used, a certain degree of multicollinearity is inevitable (e.g., when these are trend surfaces). This may complicate the interpretation of the regression results (see Chapter 26, for details). One approach suggested by Casetti and Jones (1988) to avoid this problem is to replace the original set of expansion variables by their principal components, in the form of a so-called orthogonal expansion method. This is not implemented specifically in **SpaceStat**, but may be carried out by constructing the principal components and expanded variables by means of the **Data-Var Algebra** commands R-6-9 and R-6-6. A second set of problems consists of potential heteroskedasticity when the expansion is not specified correctly (Anselin, 1988 and 1992). This is easily taken into account by means of the **OLS-Robust** (R-1-2) or **Heteroskedastic** (R-3) regression methods, and does not require any changes to the expansion specification.

### 33.2 Methodology

#### 33.2.1 Test for Coefficient Drift

Since the **Expansion Method** specification is treated as a standard regression model, the full range of estimation methods and specification diagnostics are carried out, as outlined in Part V. The single difference with the standard approach is that the default for the tests against heteroskedasticity consists of the initial variables, and is not the usual random coefficient specification. This is consistent with the likely structure of heteroskedasticity in expanded models (see Anselin, 1992, for technical details).

In addition to the standard repertoire of tests, a test is implemented on the stability of the regression coefficients, in the form of a test on the joint significance of the expanded coefficients. The null hypothesis for this test is:

$$H_0: \gamma_{1k} = \gamma_{2k} = \ldots = \gamma_{mk} = 0$$

for all expanded variables k. In ordinary least squares regression, this test results in the usual F statistic, with m.(K-1), N - m.(K-1) degrees of freedom (with m as the number of expansion variables and K as the number of explanatory variables in the initial model). For all other estimation methods, the test is an asymptotic Wald test, distributed as $\chi^2$ with m.(K-1) degrees of freedom. In addition to the joint test, a test on the significance of the expansion for each coefficient in the initial model is implemented as well. Again, this yields an F statistic for
OLS (with \(m, N - m\) degrees of freedom) and an asymptotic Wald statistic for all other methods (\(\chi^2\) with \(m\) degrees of freedom).

*SpaceStat* lists the statistic, its degrees of freedom and associated probability level, for both the joint test and the tests on each individual coefficient.

### 33.3 Example Problem File

To practice the spatial expansion method, you may use the *Problem File* listed in Table 33.1. This is also contained as the file COL12.BTC on the \EXAMPLES directory. Note that the *Problem File* includes the coordinates \(X\) and \(Y\) as the two expansion variables (flag 10) and indicates a second order polynomial in these variables (flag 14). This time, you will proceed immediately to the spatial lag model (you may check for yourself that the OLS regression indicates strong spatial lag dependence). Type \(R-4-1\) to invoke the ML estimation of this model, and enter *col12.btc* in response to the prompt for the *Problem File*.

#### Table 33.1 Problem File for Spatial Expansion

<p>| | | | | | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>COL</td>
<td>REP</td>
<td>COLWS_1</td>
<td>CRIME</td>
<td>INCOME</td>
<td>HOUSING</td>
<td>X</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 33.4 Program Output

#### 33.4.1 Spatially Expanded Model with a Spatial Lag

The estimates of the coefficients in the spatial expansion model follow the general format for the method by which they are obtained. As illustrated in Table 33.2, the estimates are listed for the spatial lag first, followed by the initial coefficients, and the expansions for each of the variables in the initial model. The expanded coefficients are denoted in the same fashion as implemented in the *Data-Var Algebra-Expansion* (D-6-6) command: the first order expansions have a prefix of \(A_\) for the first expansion variable, \(B_\) for the second, etc., the second powers have a prefix of \(AA_\) for the first variable, \(BB_\) for the second, etc., and the cross products have the prefix \(AB_\), etc. For each coefficient, the usual information is given. The header contains measures of fit and an indication of which expansion variables were used.

The spatial autoregressive coefficient of 0.49 is highly significant, confirming the indication given by the LM lag test for OLS regression (which you may check for yourself). Note that this autoregressive coefficient is not subject to the expansion. The coefficients of the initial variables INCOME and HOUSING are still significant, but HOUSING no longer has a negative sign. The expansions with the \(Y\) variable are significant, but the others are not, indicating
more of an East-West drift than a North-South drift, which confirms the indications given by the spatial regimes model. The fit of the spatial expansion model with a spatial lag is the best of all models illustrated: the log likelihood of -168.91 is by far the lowest, and even the AIC (which corrects for the large number of explanatory variables) of 365.8 is best. You may check for yourself that this model also eliminates all remaining heteroskedasticity and spatial error dependence.

33.4.2 Test on Spatial Drift

The extent to which the expansions of the coefficients are significant is indicated by the results of the tests in Table 33.3. Both the joint and the coefficient specific tests provide strong evidence for the significance of the expansion.

33.5 Exercise

You can now continue this analysis of the Columbus example, and explore the effect of using different expansion specifications. Instead, you may want to carry out an analysis of spatial expansion models for your own data set or one of the other example data sets.

References

Table 33.2 Spatial Expansion with a Spatial Lag

SPATIAL LAG MODEL - MAXIMUM LIKELIHOOD ESTIMATION

SPATIAL EXPANSION OF COEFFICIENTS - FOLLOWING EXPANSION VARIABLES

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA SET COL</td>
<td>SPATIAL WEIGHTS MATRIX COLWS_1</td>
</tr>
<tr>
<td>DEPENDENT VARIABLE CRIME</td>
<td>OBS 49 VARS 14 DF 35</td>
</tr>
</tbody>
</table>

| R2          | 0.7735     | Sq. Corr. | 0.8028 |
| LIK -168.908 | AIC 365.817 | SC 392.302 |
| SIG-SQ | 54.1620 (7.35949) |

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>COEFF</th>
<th>S.D.</th>
<th>z-value</th>
<th>Prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_CRIME</td>
<td>0.493124</td>
<td>0.12442</td>
<td>3.963398</td>
<td>0.000074</td>
</tr>
<tr>
<td>CONSTANT</td>
<td>42.8163</td>
<td>6.2531</td>
<td>6.847211</td>
<td>0.000000</td>
</tr>
<tr>
<td>INCOME</td>
<td>-32.9345</td>
<td>15.4605</td>
<td>-2.130238</td>
<td>0.033152</td>
</tr>
<tr>
<td>HOUSING</td>
<td>9.72855</td>
<td>4.01463</td>
<td>2.423278</td>
<td>0.015381</td>
</tr>
<tr>
<td>A_INCOME</td>
<td>0.022381</td>
<td>0.459709</td>
<td>0.048685</td>
<td>0.961170</td>
</tr>
<tr>
<td>B_INCOME</td>
<td>1.92667</td>
<td>0.71341</td>
<td>2.700643</td>
<td>0.006921</td>
</tr>
<tr>
<td>AA_INCOM</td>
<td>0.0107788</td>
<td>0.00612402</td>
<td>1.760087</td>
<td>0.078393</td>
</tr>
<tr>
<td>AB_INCOM</td>
<td>-0.0290948</td>
<td>0.00973851</td>
<td>-2.987605</td>
<td>0.002812</td>
</tr>
<tr>
<td>BB_INCOM</td>
<td>-0.0107795</td>
<td>0.00983186</td>
<td>-1.096386</td>
<td>0.272910</td>
</tr>
<tr>
<td>A_HOUSIN</td>
<td>0.0671222</td>
<td>0.175604</td>
<td>0.382236</td>
<td>0.702286</td>
</tr>
<tr>
<td>B_HOUSIN</td>
<td>-0.680417</td>
<td>0.213992</td>
<td>-3.179640</td>
<td>0.001475</td>
</tr>
<tr>
<td>AA_HOUSI</td>
<td>-0.0051155</td>
<td>0.00266636</td>
<td>-1.918535</td>
<td>0.055043</td>
</tr>
<tr>
<td>AB_HOUSI</td>
<td>0.0111586</td>
<td>0.00427113</td>
<td>2.612574</td>
<td>0.008986</td>
</tr>
<tr>
<td>BB_HOUSI</td>
<td>0.0031585</td>
<td>0.00354605</td>
<td>0.890710</td>
<td>0.373085</td>
</tr>
</tbody>
</table>

Table 33.3 Test on Spatial Drift

REGRESSION DIAGNOSTICS
TEST ON EXPANSION OF COEFFICIENTS

<table>
<thead>
<tr>
<th>TEST</th>
<th>DF</th>
<th>VALUE</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wald test</td>
<td>10</td>
<td>60.595678</td>
<td>0.000000</td>
</tr>
</tbody>
</table>

EXPANSION OF INDIVIDUAL COEFFICIENTS

<table>
<thead>
<tr>
<th>TEST</th>
<th>DF</th>
<th>VALUE</th>
<th>PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>INCOME</td>
<td>5</td>
<td>34.907602</td>
<td>0.000002</td>
</tr>
<tr>
<td>HOUSING</td>
<td>5</td>
<td>47.796221</td>
<td>0.000000</td>
</tr>
</tbody>
</table>
CHAPTER 34

SPATIAL ANOVA

34.1 Introduction

In many instances in exploratory spatial data analysis, you may be interested in the extent to which the mean for a variable differs significantly between spatial subsets of the data. For example, a quick glance at the map for the Columbus crime data in Appendix G would suggest that the crime rates in the central city are quite different from those in the "suburbs." The extent to which this is indeed "significant," and not just due to chance, is measured by means of Analysis of Variance (ANOVA). In ANOVA terminology, the interest is in significant differences between the means of a variable that is subject to a number of different "treatments".\footnote{A classic reference on ANOVA is Scheffé (1959).} I refer to this as "spatial" ANOVA in order to emphasize that the treatments are spatial, in the sense of being subregions of the data set (e.g., center-periphery, north-south, etc.).

In SpaceStat, the ANOVA model is treated as a regression model. The proper Problem File must contain the data set filename, the variable name for the dependent variable and the variable names for at least one treatment. If you wish to consider spatial effects, the filename of at least one spatial weights matrix must be included as well.

Also, to ensure the proper treatment of the ANOVA model, the first flag in the flag list of the Problem File must be set to 5 (instead of the default of 1 for a generic model or 0 for the Explore module).

A constant term is always included in the ANOVA estimation. If you construct the Problem File interactively, you will not be prompted for the usual constant term option, since the resulting file will always have the constant term flag set to 1 (this is the 4th item in the list of flags).

The explanatory variables for an ANOVA model must be categorical variables, i.e., variables that take on a finite number of consecutive integer values. For example, if there were 4 categories, the treatment variable could take on values 1, 2, 3 and 4. For binary categories, the values of the variable may be either 0-1 or 1-2. The only difference between the two is in the way the resulting dummy variables will be labeled. Each type of treatment is represented by one variable. SpaceStat internally converts the categories taken by that variable into the proper dummy variables. For binary categories represented by 0-1 values, the variable name for the dummy is always the original variable followed by _1.
egories, the variable name for each indicator variable is the original variable name followed by _i, where i stands for the i-th category. If the original variable is more than 5 characters long, its name will be truncated to allow for the inclusion of the _i.

Since a constant term is included in the specification, there will always be one less dummy than there are categories associated with a treatment (in order to avoid perfect multicollinearity). The base case, used to compute the overall mean is the first category (either represented by a value of 0 for binary categories, or 1 for multiple categories).

If you specify all indicator variables explicitly, you should use the generic regression model (option 1) instead of ANOVA (option 5). You start an ANOVA model by means of the command sequence for one of the regression models.

34.2 Methodology

Classic ANOVA is based on a number of simplifying assumptions that may not be satisfied in practice. Two important ones are similar to the assumptions of homoskedasticity and uncorrelatedness in a regression model. When heteroskedasticity and/or correlation are present, the interpretation of the results of the standard ANOVA tests is no longer valid.2

In SpaceStat, the ANOVA model is treated as a special case of a regression model, by associating the treatments with indicator variables in the form of a dummy variable regression. The model always includes a constant term, which corresponds to the overall mean of the dependent variable. Dummy variables measure the difference with the overall mean that is due to that particular “treatment.” Consequently, dummy variables that correspond to sub-regions in the data set measure the difference between the mean for the subregion and the overall mean.3 As is the case for the generic regression model, misspecification diagnostics may point to the failure of the assumptions of homoskedasticity and uncorrelatedness of the error terms. If this is the case, then the spatial ANOVA model should be estimated as a heteroskedastic model (i.e., a model with different variances for the treatments), as a spatial lag model, or as a spatial error model.4

34.3 Example Problem File

To practice the spatial ANOVA model, you may use the Problem File listed in Table 34.1. This is also contained as the file COL13.BTC on the \EXAMPLES directory. Note that the Problem File uses a new data set, COLANOVA. This data set only contains the neigh-

2. For an extensive discussion of spatial ANOVA, see Griffith (1978, 1992), and Legendre et al (1990)
3. Note that ANOVA in the form of a dummy variable regression can also be carried out without a constant term. However, this is not implemented in the SpaceStat Spatial ANOVA option. In order to implement this, you will have to run a generic regression model without a constant term and specify all dummy variables explicitly.
4. For a more detailed discussion, see Griffith (1992).
borhood indicator (NEIG), crime variable (CRIME) and three binary indicator variables associated with spatial subregions: EW (east-west), CP (center-periphery) and NS (north-south). This data set is included on the \COLUMBUS directory.

The Problem File shows CRIME as dependent variable and CP as the indicator variable for the ANOVA regimes (seventh flag). Again, you will first estimate the model as a Classic regression model. Type \texttt{R-1-1} as the command sequence and enter \texttt{col13.btc} in response to the prompt for the Problem File.

<table>
<thead>
<tr>
<th>Table 34.1 Problem File for Spatial ANOVA</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>5 1 1 1 1 0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>COLANOVA  REP  COLWS_1  CRIME  CP</td>
</tr>
</tbody>
</table>

34.4 Program Output

34.4.1 OLS ANOVA

The analysis consists of a dummy variable regression of CRIME on a constant term and the indicator CP_1, as listed in Table 34.2. The indicator CP_1 takes on a value of 1 for the "center" and 0 in the "periphery". The positive and highly significant value for its coefficient (24.9) indicates a strong discrepancy between the mean crime rates in the two subregions. Note that this simple model does surprisingly well, and achieves a slightly better $R^2$ (0.57 vs 0.55) and adjusted $R^2$ (0.56 vs 0.53) than the generic OLS regression of CRIME on INCOME and HOUSING. However, its likelihood of -186.7 is somewhat inferior to the value obtained in the spatial lag model of Chapter 27 (-182.4, in Table 27.2), which was the "best" model so far.

The diagnostics, listed in Table 34.3 don't indicate any problems in terms of collinearity, non-normality or heteroskedasticity. The latter is important, since a standard assumption in ANOVA is that the variances in the subgroups are constant.\(^5\) However, both Moran's I and the Lagrange Multiplier Lag test point to potential spatial dependence. Here again, the indication of spatial error autocorrelation given by Moran's I is not confirmed by either the Lagrange Multiplier Error test or by the Kelejian-Robinson test. It is thus likely that Moran's I picks up the spatial lag dependence rather than error dependence.

\(^5\) In practice, this assumption is often violated. If this were the case, you should proceed with one of the heteroskedastic estimation methods illustrated in Chapter 30.
34.4.2 Spatial Lag ANOVA

The indication of spatial lag dependence in the OLS estimation of the ANOVA model implies that these estimates may be biased, as discussed in Chapter 26. You should therefore follow up with a spatial lag estimation for the ANOVA model. You proceed in the exact same manner as outlined in Chapter 27, but using the Problem File COL13.BTC. The results for a spatial lag regression of the ANOVA model are listed in Table 34.4.

The autoregressive coefficient in Table 34.4 (0.32) is positive and highly significant, confirming the earlier indications of strong spatial clustering in crime rates. In addition, the CP_1 dummy remains significant as well, although with a slightly lower value (19.5 vs. 24.9 before). Note that the overall mean is lower as well and that the difference between the overall mean and the mean in the center actually increased (from a difference of 2 in Table 34.2 to 4.85 now). In other words, the correction for spatial dependence in the form of a spatial lag alters both the overall mean as well as the spatial difference. As pointed out before, this is a consequence of the biasedness of the OLS estimates in the model that ignores the spatial lag.6 The diagnostics for this model, shown in Table 34.5, do not reveal any further problems and confirm the significance of the spatial autoregressive term. Note that the overall fit of this model, as measured both by the likelihood (-184.5 vs -182.4) and by the information criteria (AIC: 375.0 vs 372.8 and SC: 380.6 vs 380.3) is still slightly inferior to the spatial lag in Chapter 27.

Table 34.2 OLS Regression of Spatial ANOVA

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>COEFF</th>
<th>S.D.</th>
<th>t-value</th>
<th>Prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONSTANT</td>
<td>22.932</td>
<td>2.23039</td>
<td>10.281641</td>
<td>0.000000</td>
</tr>
<tr>
<td>CP_1</td>
<td>24.9018</td>
<td>3.18693</td>
<td>7.813708</td>
<td>0.000000</td>
</tr>
</tbody>
</table>

---

6. See also Griffith (1978, 1992) for an extensive discussion of how spatial dependence affects the interpretation of ANOVA results.
Exercise

You can now continue this analysis of the Columbus example, and explore the significance of the two other indicator variables (EW and NS) instead of CP, as well as various combinations of the three. You may have to implement a heteroskedastic regression if indicated by the diagnostics. Alternatively, you could cope with the heteroskedasticity by using the OLS-Robust estimation option.

Instead, you may want to carry out an ANOVA for your own data set or one of the other example data sets. Note that neither the Irish data set nor the African one include dummy variables.
variables. To implement ANOVA, you may need to first create a relevant indicator variable and add it to the existing data set.

References

APPENDIX A

SPACESTAT MENU STRUCTURE

MAIN MENU

ALT-Q  QUIT SPACESTAT
F1  OPTIONS
F2  DOS
D  DATA MODULE
T  TOOLS MODULE
E  EXPLORE MODULE
R  REGRESS MODULE

OPTIONS (F1)

DEFAULT SETTING

1  INTERACTIVE OPERATION YES
2  OUTPUT TO A FILE NO
3  LONG OUTPUT NO
4  INDICATOR VARIABLE NO
5  IDRISI INTERFACE NO
6  CONVERGENCE CRITERION 0.000001
7  NUMBER OF PERMUTATIONS 99
8  MAXIMUM ITERATIONS 100
9  RANDOM NUMBER SEED 397204094

DATA MODULE (D)

1  INPUT

1  ASCII TO DATA SET
2  ASCII TO MATRIX
3  SPARSE ASCII TO BINARY CONTIGUITY (ROW-COL NUMBERS)
4  SPARSE ASCII TO BINARY CONTIGUITY (INDICATOR VARIABLES)
5  SPARSE ASCII TO SPATIAL WEIGHTS (ROW-COL NUMBERS)
6  SPARSE ASCII TO SPATIAL WEIGHTS (INDICATOR VARIABLES)

2  GIS INTERFACE

1  ARC/INFO BINARY CONTIGUITY
2  ARC/INFO BOUNDARY LENGTH WEIGHTS
3  IDRISI FORMAT BINARY CONTIGUITY
4  OSU-MAP FORMAT BINARY CONTIGUITY
5  GENERIC RASTER FORMAT BINARY CONTIGUITY

3  MERGE/SELECT

1  MERGE BY OBSERVATION (ADD VARS)
2  MERGE BY VARIABLE (ADD OBS)
3  SELECT VARIABLES FROM DATA SET
4 DELETE VARIABLES FROM DATA SET
5 SELECT OBSERVATIONS FROM DATA SET
6 DELETE OBSERVATIONS FROM DATA SET
7 SELECT ROWS/COLS FROM A WEIGHTS MATRIX
8 DELETE ROWS/COLS FROM A WEIGHTS MATRIX

4 VAR CREATE

1 RELABEL VARIABLES
2 RECODE VARIABLES (RANGE)
3 CREATE DUMMY VARIABLES (CATEGORIES)
4 CREATE DUMMY VARIABLES (RANGE)
5 CREATE CONSTANT
6 CREATE OBSERVATION NUMBERS
7 CREATE UNIFORM RANDOM VARIABLE
8 CREATE NORMAL RANDOM VARIABLE

5 VAR TRANSFORM

1 LN
2 EXP
3 INTEGER POWER
4 INVERSE
5 SQUARE ROOT
6 ABSOLUTE VALUE
7 DEVIATION FROM MEAN
8 STANDARDIZE

6 VAR ALGEBRA

1 ADD
2 SUBTRACT
3 MULTIPLY
4 DIVIDE
5 LINEAR COMBINATION
6 EXPANSION
7 TREND SURFACE
8 REGIMES
9 PRINCIPAL COMPONENTS

7 MAT ALGEBRA

1 ExE ADD
2 ExE SUBTRACT
3 ExE MULTIPLY
4 ExE DIVIDE
5 MATRIX MULTIPLY
6 MATRIX INVERSE
7 MATRIX DETERMINANT
8 MATRIX TRACE
8 LIST

1 SUMMARY DATA SET
2 LIST DATA SET
3 LIST SELECTED VARIABLES
4 LIST SELECTED OBSERVATIONS
5 SUMMARY MATRIX
6 LIST MATRIX
7 LIST BINARY CONTIGUITY IN SPARSE FORM
8 LIST SPATIAL WEIGHTS IN SPARSE FORM

TOOLS MODULE (T)

1 SPACE TRANS

1 SPATIAL LAG
2 WINDOW AVERAGE
3 SPATIAL AUTOREGRESSION FILTER
4 SPATIAL MOVING AVERAGE FILTER
5 SPATIAL AUTOREGRESSION TRANSFORM
6 SPATIAL MOVING AVERAGE TRANSFORM
7 NON-CONTIGUOUS RANDOM SAMPLE

2 WEIGHT TRANS

1 ROW STANDARDIZATION
2 HIGHER ORDER CONTIGUITY
3 GENERAL WEIGHTS TO BINARY CONTIGUITY
4 CONVERT WEIGHTS FORMAT TO MATRIX FORMAT
5 ExE POWER OF WEIGHTS MATRIX
6 ExE INVERSE OF WEIGHTS MATRIX
7 BOUNDARY SHARES OVER DISTANCE WEIGHTS

3 RASTER WTS

1 ROOK
2 BISHOP
3 QUEEN

4 DISTANCE WTS

1 DISTANCE TO BINARY WEIGHTS
2 INVERSE DISTANCE WEIGHTS
3 CREATE DISTANCE MATRIX
4 CREATE ARC DISTANCE MATRIX
5 CHARACTERISTICS OF DISTANCE MATRIX

5 WEIGHT CHARS

1 ROOTS
2 CONNECTIVITY
3 DOMINANT ROOT
4 S0 S1 S2
EXPLORE MODULE (E)

1 DESCRIBE
   1 DESCRIPTIVE STATS
   2 CORRELATION
   3 PRINCIPAL COMPONENTS
   4 MULTIVARIATE SPATIAL CORRELATION
   5 SPATIAL PRINCIPAL COMPONENTS

2 JOIN COUNT
   1 BB JOIN COUNTS
   2 BW JOIN COUNTS
   3 WW JOIN COUNTS

3 MORAN
   1 NORMAL
   2 RANDOMIZATION
   3 PERMUTATION
   4 CORRELOGRAM - NORMAL
   5 CORRELOGRAM - RANDOMIZATION
   6 CORRELOGRAM - PERMUTATION

4 GEARY
   1 NORMAL
   2 RANDOMIZATION
   3 PERMUTATION
   4 CORRELOGRAM - NORMAL
   5 CORRELOGRAM - RANDOMIZATION
   6 CORRELOGRAM - PERMUTATION

5 G-STATS
   1 G-STAT
   2 G-STAT - CORRELOGRAM
   3 G-I STAT
   4 G-I* STAT

6 QAP
   1 MATRIX AND PARTIAL MATRIX ASSOCIATION
   2 QAP - MORAN
   3 QAP - GEARY
   4 QAP - SOKAL
FOR ALL

1  BATCH
2  INTERACTIVE
3  MAKE PROBLEM FILE

REGRESS MODULE (R)

1  CLASSIC MODEL
   1  OLS
   2  OLS - ROBUST

2  SPACE ERR MODEL
   1  SPATIAL AR

3  HETEROSK ERR MODEL
   1  GENERIC HETEROSKEDASTICITY (FGLS)
   2  GENERIC HETEROSKEDASTICITY (ML)
   3  GROUPWISE HETEROSKEDASTICITY (FGLS)
   4  GROUPWISE HETEROSKEDASTICITY (ML)
   5  RANDOM COEFFICIENTS (FGLS)
   6  RANDOM COEFFICIENTS (ML)

4  SPACE LAG MODEL
   1  SAR - ML
   2  SAR - IV (2SLS)
   3  SAR - BOOT

FOR ALL

1  BATCH
2  INTERACTIVE
3  MAKE PROBLEM FILE
APPENDIX B

SPACESTAT PROBLEM FILE STRUCTURE

OVERALL STRUCTURE

first item: number of problems

for each problem:
- 15x1 list of flags
- list of file and variable names

FLAGS

1  model type
   0  explore
   1  generic regression
   2  trend surface regression
   3  regimes
   4  spatial expansion
   5  ANOVA

2  report file
   0  default, same as output file
   1  report file specified

3  spatial weights
   0  no spatial weights
   #  number of spatial weights

4  constant
   0  no constant
   1  constant = default

5  dependent variable (endogenous variables)
   0  for explore only (if desired)
   1  default

6  spatially lagged dependent variable
   0  no lag, will be computed if needed
   1  lag is in data set

7  explanatory variables
   0  only allowed if constant only, or for explore
   #  number of explanatory variables, not including constant

8  spatially lagged explanatory variables
   0  no lag, will be computed if needed
   #  if same as 7, lags present in data set, otherwise either ignored or added to explanatory variables
<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>structural change</td>
</tr>
<tr>
<td>0</td>
<td>no structural change (error for model=3)</td>
</tr>
<tr>
<td>1</td>
<td>structural change</td>
</tr>
<tr>
<td>10</td>
<td>expansion variables</td>
</tr>
<tr>
<td>0</td>
<td>no expansion variables (error for model=4)</td>
</tr>
<tr>
<td></td>
<td>number of expansion variables</td>
</tr>
<tr>
<td>11</td>
<td>heteroskedastic variables</td>
</tr>
<tr>
<td>0</td>
<td>none specified (default is random coeff)</td>
</tr>
<tr>
<td></td>
<td>number of heterosked variables (linear)</td>
</tr>
<tr>
<td>12</td>
<td>instruments</td>
</tr>
<tr>
<td>0</td>
<td>no instruments</td>
</tr>
<tr>
<td></td>
<td>number of instruments</td>
</tr>
<tr>
<td>13</td>
<td>order of trend surface</td>
</tr>
<tr>
<td>0</td>
<td>no trend (error for model=2)</td>
</tr>
<tr>
<td></td>
<td>order of polynomial</td>
</tr>
<tr>
<td>14</td>
<td>order of spatial expansion</td>
</tr>
<tr>
<td>0</td>
<td>no expansion (error for model=4)</td>
</tr>
<tr>
<td>1</td>
<td>linear</td>
</tr>
<tr>
<td>2</td>
<td>quadratic</td>
</tr>
<tr>
<td>15</td>
<td>unused in current version</td>
</tr>
</tbody>
</table>

**FILE AND VARIABLE NAMES**

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>data set</td>
</tr>
<tr>
<td>2</td>
<td>report file (if specified)</td>
</tr>
<tr>
<td>3</td>
<td>spatial weights (if specified)</td>
</tr>
<tr>
<td>4</td>
<td>dependent variable</td>
</tr>
<tr>
<td>5</td>
<td>spatial lag</td>
</tr>
<tr>
<td>6</td>
<td>explanatory variables</td>
</tr>
<tr>
<td>7</td>
<td>lagged explanatory variables</td>
</tr>
<tr>
<td>8</td>
<td>structural change indicator</td>
</tr>
<tr>
<td>10</td>
<td>expansion variables</td>
</tr>
<tr>
<td>11</td>
<td>heteroskedasticity variables</td>
</tr>
<tr>
<td>12</td>
<td>instruments</td>
</tr>
</tbody>
</table>
APPENDIX C

SPACESTAT SPATIAL WEIGHTS FILE STRUCTURE

OVERALL STRUCTURE

N by N+2 matrix, where N is the number of observations and must be at least equal to 20

first column: spatial weights characteristics (if computed)

second column: eigenvalues of the spatial weights matrix (if computed)

columns 3 through N+3: weights

CHARACTERISTICS OF SPATIAL WEIGHTS

1 dimension
2 flag for row-standardization (1=standardized, 0=not)
3 flag for presence of unconnected observations, i.e., a row consisting of only zeros in the spatial weights matrix (1=zero row present, 0=not)
4 flag for computation of eigenvalues of weights matrix (1=eigenvalues computed, 0=not)
5 flag for computation of special sums $S_0$ and $S_1$ (1=computed, 0=not)
6 flag for computation of special sum $S_2$ (1=computed, 0=not)
7 flag for computation of special matrix traces (1=computed, 0=not)
8 flag for computation of weight matrix connectivity measures (1=computed, 0=not)
9 maximum eigenvalue
10 minimum eigenvalue
11 % nonzero weights
12 average weight
13 number of nonzero links
14 average number of links
15 $S_0$
16 $S_1$
17 $S_2$
18 trace of $WW$
19 trace of $W^2$
20 trace of $(WW+W^2)$

For a formal definition of $S_0$, $S_1$, and $S_2$, see Part IV. For an extensive discussion of the role of the matrix traces, see Part V.