4

Specifying Spatial Relationships on the Map: The Weights Matrix

4.1 Introduction

In this chapter we describe various ways of representing spatial relationships between areas (or fixed points, such as the locations of petrol stations or the centroids of census tracts) on a map using a spatial weights matrix. This matrix is often referred to as the $\boldsymbol{W}$ matrix. When modelling spatial data, the spatial weights matrix, $\boldsymbol{W}$, plays an important role because it provides a way of summarising the spatial relationships between the areas (or points) to which the observed data are georeferenced.[1] As we shall discuss in this chapter, the spatial relationships amongst areas constitute an important data element, and the construction of the $\boldsymbol{W}$ matrix allows us to turn these relationships into numbers. However, different from conventional data (such as the observed numbers of crime or disease cases), this form of data is often based on our assumptions about how the areas are connected to each other. A purpose of this chapter, then, is to introduce different ways of defining connections, as well as the strength of those connections, amongst pairs of areas. Some of these methods are based on the geographical configuration of the areas, and some methods are attribute-based, in which distance between any two areas is defined by how similar (or dissimilar) the two areas are in terms of a chosen attribute.

When modelling a spatial dataset that comprises N areas (or fixed points), a spatial weights matrix $\boldsymbol{W}$ is an $N \times N$ (N rows by N columns) matrix that defines how we choose to specify the spatial relationships between the areas (or the points). In this chapter (also throughout this book), we use w_{ij} to denote the element of $\boldsymbol{W}$ on the ith row and jth column ($i=1,\ldots,N$ and $j=1,\ldots,N$). Often the elements of $\boldsymbol{W}$ take the value either 0 or 1. If $w_{ij}=1$, then we say that the two areas (or points) i and j are *neighbours* of each other. If $w_{ij}=0$, then the two areas (or points) i and j are not neighbours. Each diagonal element w_{ii} in $\boldsymbol{W}$ is set to 0 (for all i), meaning that an area (or point) is not allowed to be a neighbour of itself. However, in some situations where, for example, we are applying some spatial smoothing operation to the data, we may want to set w_{ii} to 1 (for all i) so that the value in area i, together with the values in the neighbours of i, are used in the calculation. We discuss this option in more detail later in the chapter. The elements in $\boldsymbol{W}$ can also be non-binary. That is, they can be real numbers (not necessarily integers) greater than or equal to zero. This option allows us to vary the strength of "neighbourliness" from one pair of neighbouring areas to another, for example as a function of how close two areas (or points) are geographically. The focus

[1] The areas (or points) are the nodes, whilst the elements of $\boldsymbol{W}$ specify the edges that connect the nodes. We use the term "matrix" here for explanatory purposes and because formulae typically use matrix notation. In computer software, spatial relationships are usually presented as lists of areas with their defined "neighbours".

of this chapter is on defining spatial weights matrices and discussing the important consequences of any definition. We defer the discussion of weights matrices for modelling temporal and spatial-temporal datasets until Chapters 12 and 15, respectively.

Later in the chapter we discuss the roles that the $\boldsymbol{W}$ matrix plays in various forms of statistical analyses of spatial data and the statistical implications arising from any choice made for the form of this matrix. As we shall discuss, the specification of $\boldsymbol{W}$ has been a particular subject of study in spatial econometrics, a collection of models for spatial data that we will discuss in depth in Chapter 10. But one of the principal goals in spatial econometric modelling is to obtain reliable estimates of spatial spillover and feedback effects – for example, how economic decisions or activity in one area impact on outcomes in neighbouring areas, both directly and indirectly (i.e. through third-party areas), as well as in the originating area. Thus, an important aspect of that modelling is how the spatial relationships between areas are defined through the $\boldsymbol{W}$ matrix. As we shall also see, there are implications for other areas of spatial modelling, including small area estimation and the estimation of covariate effects.

In the first five sections in this chapter (Sections 4.2–4.6), we describe five main methods for defining the elements in the $\boldsymbol{W}$ matrix: contiguity (Section 4.2), geographical distance (4.3), graph-based methods (4.4), attribute-based methods (4.5) and interaction-based methods (4.6). The first three methods, contiguity, geographical distance and graph-based methods, construct $\boldsymbol{W}$ based on purely geometrical or topological relationships amongst the set of areas (or points). The other two methods, attribute-based and interaction-based, define $\boldsymbol{W}$ based on data (observed values) that describe certain characteristics of the areas where such characteristics are thought to underlie the interconnectedness of these areas. In Section 4.7, we discuss row standardisation of the $\boldsymbol{W}$ matrix, and in Section 4.8 we describe higher order weights matrices (such as $\boldsymbol{W}^2$, $\boldsymbol{W}^3$ and so on). In Section 4.9 we draw the reader's attention to the roles of the $\boldsymbol{W}$ matrix in the statistical modelling of spatial data and some of the statistical consequences arising from the way that $\boldsymbol{W}$ is specified.

Although different ways of defining the $\boldsymbol{W}$ matrix exist, the reader should note that in much statistical modelling to date it is the contiguity and geographical distance criteria that have been most frequently employed. Whilst Tobler's First Law (see Section 3.3.2.1) is often invoked to justify a choice of $\boldsymbol{W}$ based on geometrical or topological relationships between the areas, or the choice of $\boldsymbol{W}$ is based just on what software makes possible, we argue that the specification of $\boldsymbol{W}$ really ought to be placed in the context of the problem in hand. In some situations, as we shall describe in Section 4.9, we may even need to model the elements in the $\boldsymbol{W}$ matrix, as opposed to treating them as known quantities. Therefore, in Section 4.10 we briefly discuss approaches to estimating the weights matrix. Some of these methods will be discussed in more detail in Part II of the book. The two appendices at the end of this chapter deal with some practical problems: how to combine a spatial dataset with a shapefile; how to create various spatial weight matrices from a shapefile; and how to manipulate the data values stored in a shapefile.

4.2 Specifying Weights Based on Contiguity

There are two methods that lie within this category. Under the first method, if areas i and j share a common border, then $w_{ij} \neq 0$. In other words, these two areas are defined as neighbours. Typically, $w_{ij} = 1$. If two areas do not share a common boundary, then $w_{ij} = 0$ and they are not neighbours. By analogy with a chess board, this first method is sometimes referred to as the "rook's move" definition for contiguity.

The second method defines two areas i and j to be neighbours, i.e. $w_{ij}=1$, if they share either a common border *or* a common vertex (two areas touch at just one point). Otherwise they are considered to be non-neighbours ($w_{ij}=0$). By analogy again with a chess board, this is sometimes referred to as the "queen's move" definition for contiguity.

In both methods, $w_{ij}=w_{ji}$ so that $\mathbf{W}$ is a symmetric matrix. This distinction between rook's move and queen's move is important in the case of data reported on square pixels. Compared to rook's move contiguity, queen's move contiguity increases the number of neighbours for each pixel on the map. In particular, for the pixels that are not at the boundary of the map, the increase is from four to eight.

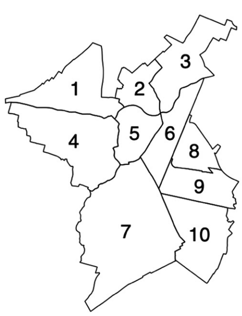
In the case of data reported on census units, which are typically irregular in shape, this distinction is usually of much lesser significance. Figure 4.1 shows a binary weights matrix for a small set of spatial units using queen's move contiguity as the criterion. Note that areas 2 and 6 are treated as neighbours ($w_{26}=w_{62}=1$) and so are areas 3 and 5 ($w_{35}=w_{53}=1$) under queen's move, but they would not be under rook's move.

Constructing the weights matrix via spatial contiguity can be refined by specifying each weight (the strength of the "neighbourliness") as a function of the length of the common border as a proportion of the total border of i. That is,

$$w_{ij}=\left(\frac{l_{ij}}{l_i}\right)^a \tag{4.1}$$

where l_i is the length of the border of spatial unit i and l_{ij} is the length of the shared border between i and j. The power (exponent) a is usually specified by the user rather than estimated. Applying this weight specification to an irregular spatial partition, generally $w_{ij}\neq w_{ji}$, and thus $\mathbf{W}$ is no longer a symmetric matrix.

A question that may need resolving is what to do with islands or if the spatial system comprises blocks of areas that are non-contiguous. If spatial unit i is an island, then the ith row of the $\mathbf{W}$ matrix under spatial contiguity would consist entirely of zeros. When using the weights matrix as part of a spatial model, this is not permitted, and a typical solution is to "join" islands to the spatial units on the mainland that are nearest (see an example in Section 11.1.2 in Chapter 11).[2]



	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
A1	0	1	0	1	1	0	0	0	0	0
A2	1	0	1	0	1	1	0	0	0	0
A3	0	1	0	0	1	1	0	0	0	0
A4	1	0	0	0	1	0	1	0	0	0
A5	1	1	1	1	0	1	1	0	0	0
A6	0	1	1	0	1	0	1	1	1	0
A7	0	0	0	1	1	1	0	0	1	1
A8	0	0	0	0	0	1	0	0	1	0
A9	0	0	0	0	0	1	1	1	0	1
A10	0	0	0	0	0	0	1	0	1	0

FIGURE 4.1
A map of spatial units and an associated binary (0/1) weights matrix based on queen's move contiguity.

[2] This is not a problem if the weights matrix is only being used for smoothing purposes, although the user might wish to smooth the island value with one or more of the spatial units close by on the mainland.

4.3 Specifying Weights Based on Geographical Distance

This method of defining the weights matrix $\boldsymbol{W}$ can be applied quite naturally to the case where data values are attached to fixed point sites. But any collection of spatial units in the form of polygons can also be represented by a collection of points. For example, we could take the geometric centroid of each polygon or, in other circumstances, we might use the population weighted centroid where the location of the centroid depends on the distribution of population within the polygon. This might be particularly appropriate if spatial interaction is defined through resident populations (in each spatial unit) and these populations are not uniformly distributed. In any of these cases, we can specify spatial relationships in terms of Euclidean (straight-line) distance. Other distance metrics could be used, and in small area analyses of urban data the Manhattan metric is sometimes chosen, which measures distance by following the street network.

Let d_{ij} be the distance (Euclidean, Manhattan or other distance measure) between points i and j with $i \neq j$. Listed below are three different functions to define the weight between those two points:

$$\begin{aligned}
&\text{(i)} \rightarrow \text{Inverse distance}: \ w_{ij} = d_{ij}^{-1} \\
&\text{(ii)} \rightarrow \text{Inverse distance raised to the power } \gamma > 0: \ w_{ij} = d_{ij}^{-\gamma} \\
&\text{(iii)} \rightarrow \text{Negative exponential with } \lambda > 0: \ w_{ij} = \exp\left(-\lambda \cdot d_{ij}\right)
\end{aligned} \tag{4.2}$$

The diagonal elements in $\boldsymbol{W}$ are all equal to zero. In both (ii) and (iii), the two parameters γ and λ are often specified by the analyst as opposed to being estimated using data. All the above three functions have the same properties that (a) as distance increases the weight decreases, and thus neighbourliness gets weaker; and (b) the resulting weights are symmetric ($w_{ij} = w_{ji}$ for all i and j). All the off-diagonal elements in $\boldsymbol{W}$ will be non-zero (albeit declining close to zero as distance increases) unless we impose a threshold. Two commonly applied thresholds are:

(i) Absolute distance threshold: if $d_{ij} \geq x$ then $w_{ij} = 0$.
(ii) Based on k nearest neighbours: allowing only the k nearest neighbours of spatial unit i to be non-zero. In this case, weights will not be symmetric.

The distance and common border definitions of neighbourliness can be combined, for example, using the following form:

$$w_{ij} = \left(\frac{l_{ij}}{l_i}\right)^a \cdot d_{ij}^{-\gamma} \tag{4.3}$$

As before, both a and γ are positive, and their values are specified by the analyst.

4.4 Specifying Weights Based on the Graph Structure Associated with a Set of Points

Where the spatial units on which the data have been collected are, or can be treated as, a set of fixed points (such as a set of retail outlets in a city, or a set of urban places across a large region), then we can impose a graph on the set of points and use that as a basis for defining a set of weights. In the case of the Gabriel graph (Matula and Sokal, 1980), as illustrated in Figure 4.2, any two points A and B are neighbours if and only if all other points are outside the circle on whose circumference A and B lie and whose centre lies on the line segment joining A and B.

Another method of defining neighbours for a set of fixed points based on graph structure is through a Dirichlet tessellation (also referred to as a Voronoi partition or Voronoi decomposition). To illustrate the method, Figure 4.3(a) shows a set of points which are called fixed seed points, and Figure 4.3(b) shows a Dirichlet partition defined on those fixed seed points. A Dirichlet partition for a set of fixed seed points is a partitioning of the space into regions with each region containing only one of those fixed seed points. These regions (called Voronoi polygons, Thiessen polygons or Dirichlet cells) are formed such that if you are standing at any position inside a region, you are always closer to the seed point in that region than to any of the other seed points. This partitioning can then be used to provide a set of contiguity-based weights so that, for example, in Figure 4.3(b), the two fixed points A and C are neighbours, but A and F are not. Alternatively, a Delaunay triangulation, as illustrated in Figure 4.3(c), can be used to connect each pair of fixed points if the respective regions that these two points are in share a common border. Then a set of distance-based weights (combined with contiguity because of the use of the Delaunay triangulation) can be defined.

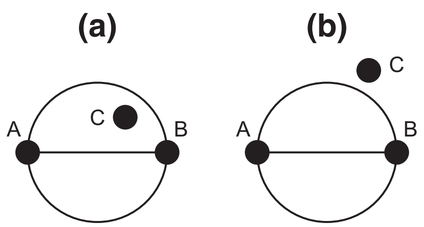


FIGURE 4.2
Defining neighbours on a Gabriel graph. In (a), the two points A and B are not defined to be neighbours, because point C is inside the circle, whereas in (b), A and B are defined to be neighbours since C is outside the circle.

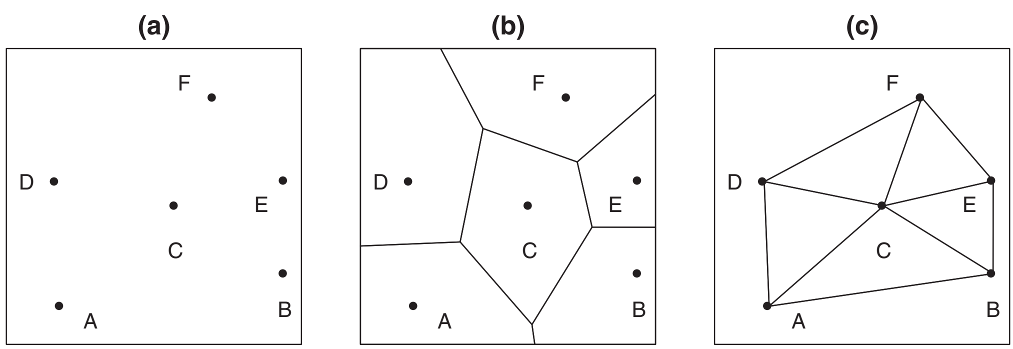


FIGURE 4.3
(a) A set of point sites; (b) a Dirichlet tessellation defined on the point sites, where the line that separates each pair of points dissects perpendicularly the line segment that joins those two points (for example, the nearly horizontal line that separates points B and E perpendicularly dissects the line joining B and E); and (c) the Delaunay triangulation based on the Dirichlet tessellation in (b).

4.5 Specifying Weights Based on Attribute Values

Neighbourliness can be defined in terms of how similar two areas are based on some attribute or set of attributes. Case et al. (1993), in a study of budget spillovers amongst US states, defined w_{ij} based on how similar two states were in terms of their socio-economic composition. Their definition of the weight between states i and j is given by

$$w_{ij} = \frac{1}{\left|x_i - x_j\right|} \tag{4.4}$$

where x_i and x_j are the values of a selected socio-economic variable in the two states and $\left|x_i - x_j\right|$ is the absolute value of the difference. Law (2016), in a study of fall injuries amongst senior citizens by small area in a part of Eastern Canada, specified the weight between two areas as

$$w_{ij} = E_i \cdot E_j \tag{4.5}$$

where E_i and E_j are the expected numbers of fall injuries in areas i and j, respectively, taking into account the age and sex composition of the areas (see Section 9.3.2 in Chapter 9). This formulation of weights implies that the neighbourliness of two areas is strong if the expected numbers of fall injuries in both areas are large. These weights can be further modified to include spatial separation of the two areas by, for example, dividing by d_{ij}. We will see some examples of the use of attribute values to define the $\boldsymbol{W}$ matrix in Chapter 8 (in particular, Sections 8.3.3 and 8.4.2).

4.6 Specifying Weights Based on Evidence about Interactions

Interaction data such as data on flows of goods or people or communications can be used to specify weights. The underlying idea is that real flows may be indicative of the strength of neighbourliness (contact) that may exist between different places and which may therefore help in explaining spatial variation.

Bavaud (1998) proposed two different ways to define interaction weights:

$$\begin{aligned} &\text{Export} - \text{based weights}: \; w_{ij} = f_{i\to j} \,/\, f_{i\to\cdot} \\ &\text{Import} - \text{based weight}: \; w_{ij} = f_{j\to i} \,/\, f_{\cdot\to i} \end{aligned} \tag{4.6}$$

where

- $f_{i\to j}$ measures the export from area i to area j.
- $f_{j\to i}$ measures the import from area j to area i.
- $f_{i\to\cdot}$ is the total export from i to all other areas in the study region.
- $f_{\cdot\to i}$ measures the total import from all other areas to i.

In general, weights defined through interaction between areas will not be symmetric, i.e. $w_{ij} \neq w_{ji}$.

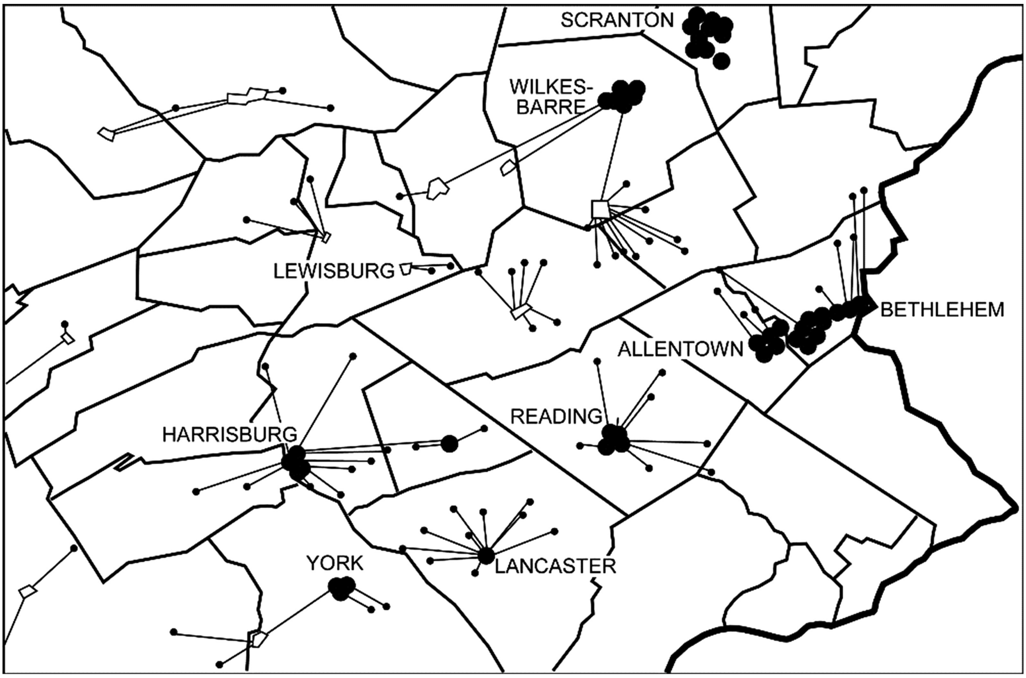


FIGURE 4.4
Urban places and their economic links based on interaction data: Pennsylvania, USA (after Haining, 1987).

Figures 4.4 and 4.5 illustrate two other examples of defining weights based on interaction. In Figure 4.4, urban places in Pennsylvania are linked together based on a set of hierarchical relationships drawn from Central Place Theory (Losch, 1957). Smaller urban places close to larger urban places are assumed to be economically dependent on those larger places (through wage transfers and expenditure patterns), but larger places are not dependent on smaller places. In the case of Figure 4.5, managers of petrol retail outlets in a city were asked to name the sites they believed were their principal competitors in the urban market. Linkages were then drawn using that information.

Conway and Rork (2004) used migration data to construct weights in their study of US state taxes. Davies and Voget (2008), in their study of tax competition within the European Union, constructed weights based on relative market potential, a theoretical concept from the new economic geography literature (see Baldwin and Krugman, 2004; Krugman, 1996, 1998). Their paper includes a review of other approaches to specifying spatial weights (including distance and levels of GDP) in studies of tax rates, some of which can be classified under the methods described in Sections 4.3 and 4.4.

4.7 Row Standardisation

Where polygons have different numbers of neighbours that we wish to control for, then the weights matrix needs to be row standardised. In some areas of spatial modelling, working with a row-standardised matrix helps in the interpretation of model parameters. For

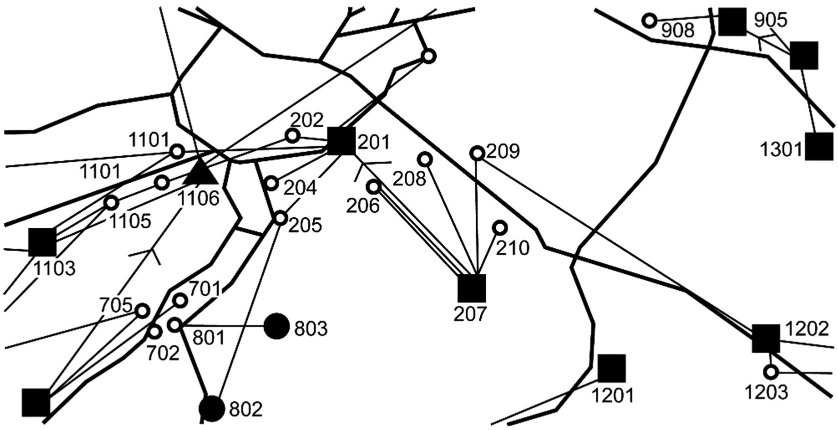


FIGURE 4.5
Competitive links between petrol retail sites based on proximity and evidence from a questionnaire survey of site managers in Sheffield, England. Each number refers to a retail site, the shapes refer to their corporate type (solid square = "Major" retailer; solid circle = "Minor" retailer; solid triangle = supermarket retailer; open circle = a reference station for the retailer, to which it is joined by a line. If no arrow, both sites consider the other a reference station in price setting; arrow head indicates direction of referencing). From Ning and Haining, 2003, Figure 4.

example, in the context of spatial smoothing, we may assume that the attribute value of a spatial unit is similar to an average (as opposed to the sum) of the attribute values of its neighbouring units.

To row standardise the ***W*** matrix, each element on a row is divided by the corresponding row sum. So the sum of all elements on a row after row standardisation equals 1, and the sum of all values in the row-standardised matrix, often referred to as ***W****, equals N, the number of spatial units in the study region. It follows that row-standardised weights matrices are no longer symmetric, as shown in Figure 4.6. For the conditional autoregressive (CAR) models, the asymmetry of the row-standardised weights matrix has several implications, which will be discussed in Chapter 8.

As part of exploratory spatial data analysis, row-standardised matrices are used to obtain spatially smoothed maps in order to iron out the noises (the sudden "jumps") in the

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	Sum
A1	0	1	0	1	1	0	0	0	0	0	3
A2	1	0	1	0	1	1	0	0	0	0	4
A3	0	1	0	0	1	1	0	0	0	0	3
A4	1	0	0	0	1	0	1	0	0	0	3
A5	1	1	1	1	0	1	1	0	0	0	6
A6	0	1	1	0	1	0	1	1	1	0	6
A7	0	0	0	1	1	1	0	0	1	1	5
A8	0	0	0	0	0	1	0	0	1	0	2
A9	0	0	0	0	0	1	1	1	0	1	4
A10	0	0	0	0	0	0	1	0	1	0	2

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	Sum
A1	0	0.33	0	0.33	0.33	0	0	0	0	0	1
A2	0.25	0	0.25	0	0.25	0.25	0	0	0	0	1
A3	0	0.33	0	0	0.33	0.33	0	0	0	0	1
A4	0.33	0	0	0	0.33	0	0.33	0	0	0	1
A5	0.17	0.17	0.17	0.17	0	0.17	0.17	0	0	0	1
A6	0	0.17	0.17	0	0.17	0	0.17	0.17	0.17	0	1
A7	0	0	0	0.20	0.20	0.20	0	0	0.20	0.20	1
A8	0	0	0	0	0	0.50	0	0	0.50	0	1
A9	0	0	0	0	0	0.25	0.25	0.25	0	0.25	1
A10	0	0	0	0	0	0	0.50	0	0.50	0	1

FIGURE 4.6
An original, unstandardised, contiguity ***W*** matrix (left; as in Figure 4.1) and its row-standardised form (right).

map so that spatial trends and patterns in the data can be easier to identify (Section 6.2.2). Let $\boldsymbol{z}$ denote a column vector of length N containing the data values to be smoothed. Let $\boldsymbol{W}$ denote an $N \times N$ binary (0/1) weights matrix, and let $\boldsymbol{I}_N$ denote the identity matrix of size N. When the purpose of spatial smoothing is to remove noise on the map, we want to include the value from area i in calculating the spatial average for area i. That can be done by adding the identity matrix to $\boldsymbol{W}$, i.e. $(\boldsymbol{I}_N + \boldsymbol{W})$. So, denoting $(\boldsymbol{I}_N + \boldsymbol{W})^*$ as the row-standardised version of $(\boldsymbol{I}_N + \boldsymbol{W})$, and $\left(\boldsymbol{I}_N + \boldsymbol{W}\right)_i^*$ as the ith row of $(\boldsymbol{I}_N + \boldsymbol{W})^*$, then the spatial average for area i is given by

$$z_i^{(SL1)} = \left(\boldsymbol{I}_N + \boldsymbol{W}\right)_i^* \boldsymbol{z} = \sum_{j=1}^{N} h_{ij}^* z_j \tag{4.7}$$

where h_{ij}^* is the element on row i, column j of the row-standardised matrix $(\boldsymbol{I}_N + \boldsymbol{W})^*$. We use the superscript ($SL1$) in $z_i^{(SL1)}$ to signify that it is a spatially smoothed (also referred to as a *spatially-lagged*) value. Then $\boldsymbol{z}^{(SL1)}$ is a spatially-smoothed version of $\boldsymbol{z}$. Eq. 4.7 is one of many ways of spatial smoothing, and thus we label it as $SL1$.

Another way to calculate a spatially-lagged value for area i is to exclude its own value in the calculation. For example, in spatial epidemiology, we may wish to distinguish between the impacts on health from exposure to some environmental hazard (atmospheric pollution, for example) in an area from exposure arising in neighbouring areas. The latter aims to capture the residents' exposure arising from short distance movements (for example, for commuting and shopping). See Section 1.3.2.1. Then the first kind of effect could be quantified via the inclusion of the exposure covariate $\boldsymbol{x}$, whilst the second kind of effects could be investigated through a spatially-lagged version of the exposure covariate:

$$x_i^{(SL2)} = \boldsymbol{W}_i^* \boldsymbol{x} = \sum_{j=1}^{N} w_{ij}^* x_j \tag{4.8}$$

Different from Eq. 4.7, this version of spatial smoothing (labelled as $SL2$) excludes the exposure level in area i so that $x_i^{(SL2)}$ represents an average exposure level of the neighbours of i. In Eq. 4.8, $\boldsymbol{W}_i^*$ denotes the ith row of the row-standardised $\boldsymbol{W}$. This way of calculating a spatially-lagged version of an observable covariate is also used in the spatially-lagged covariates (SLX) model in spatial econometric modelling (see Section 10.2.3 for more detail).

4.8 Higher Order Weights Matrices

The binary (0/1) matrix $\boldsymbol{W}$ based on contiguity is sometimes referred to as the first order contiguity weights matrix because it defines the immediate or first order neighbours. Spatial weights matrices can be defined for second, third and higher orders of contiguity. As the order increases, the spatial separation between an area and its neighbours at the corresponding order increases. Here, spatial separation between two areas is measured by the number of steps along the shortest path. As illustrated in Figure 4.7, the second order neighbours of area 6 are generally further away from area 6 than the first order neighbours. For this map, there are no third order neighbours for area 6. A purpose in defining

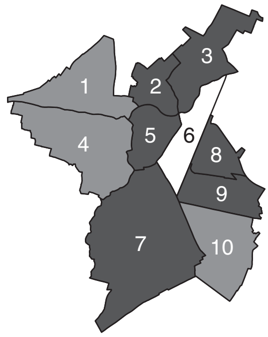


FIGURE 4.7
The first order (the polygons in a darker grey colour) and the second order neighbours (the polygons in a lighter grey colour) of area 6 based on queen's move contiguity. Note that the two-colour shadings do not represent the strength of relationships; they are only used to distinguish the first and the second order neighbours.

higher order weights matrices is to assess how similarity (measured by the strength of autocorrelation) decays with increasing spatial separation (see Chapter 6 and Section 6.2.4 in particular). Spatial smoothing can be carried out by including higher order neighbours (e.g. averaging values from both the first and the second order neighbours) so that the resulting map becomes smoother than if only first order neighbours are included in the smoothing procedure.

Different orders of contiguity can be obtained by raising the first order contiguity binary (0/1) matrix, $\boldsymbol{W}$ (e.g. as in Figure 4.1), to a positive integer power e.g. $\boldsymbol{W}^2 = \boldsymbol{W} \times \boldsymbol{W}$ for the second order contiguity and $\boldsymbol{W}^3 = \boldsymbol{W}^2 \times \boldsymbol{W} = \boldsymbol{W} \times \boldsymbol{W} \times \boldsymbol{W}$ for the third order and so on. To look at these powered $\boldsymbol{W}$ matrices in more detail, let $w_{ij}^{(k)}$ denote the element in row i and column j of the $\boldsymbol{W}$ matrix of order k. The value of $w_{ij}^{(k)}$ equals the *total* number of paths going from i to j in k steps. For example, using queen's move contiguity, we can go from area 6 in Figure 4.7 to any one of the following areas, 2, 3, 5, 7, 8 and 9, in one step, and there is only one pathway leading to one of those areas from area 6. Thus, $w_{6j} = 1$, with $j = 2,3,5,7,8$ and 9. Note that the superscript (k) in $w_{ij}^{(k)}$ is removed when $k = 1$. However, since we cannot go from area 6 to the other three areas (1, 4 and 10) in one step, $w_{6j} = 0$, with $j = 1,4$ and 10. The diagonal element $w_{66} = 0$, since by taking one step, we have to move away from area 6. Therefore, we can write down all the first order neighbours of an area directly from the binary $\boldsymbol{W}$ matrix.

However, as far as obtaining a higher order contiguity matrix of, say, second (or third) order is concerned, the problem with raising $\boldsymbol{W}$ to the power two (or three) is that the resulting matrix will count *all* the two- (or *all* the three-) step paths from i to j. Consider again area 6 in Figure 4.7. There are two ways of stepping from area 6 to area 1 in two steps: one way is via area 2, and the other way is via area 5. Thus, $w_{61} = 2$. But if all we want to know is whether area 2 is a second order neighbour of area 6, the actual *number* of ways of going from 6 to 1 in two steps is redundant information.[3] Then there is the problem of the

[3] There are circumstances where we want to know the number of pathways between two areas, for example if we are interested in constructing a measure of influence of areas on each other. Deriving the number of pathways becomes important when we come to explain the properties of direct and indirect effects in spatial econometric modelling (see Section 10.3.3.3 for more detail). Here, however, we are interested in constructing a way of identifying second and higher order neighbours for the purpose of calculating a second or higher order spatial autocorrelation statistic, for example.

pathways themselves and whether they involve backtracking or circularity (or following a "scenic" route) – all of which would be undesirable if we are seeking a proper measure of second or third, or higher order neighbourliness. To take an example, suppose we calculate $\boldsymbol{W}^2$ (see Exercise 4.4). We will see that w_{66} is not zero but six, meaning that area 6 is a "neighbour" of itself in two steps. The six two-step pathways involve going from area 6 to one of its six first order neighbours, then back to itself. However, area 6 should not really be considered a second order neighbour of itself. It is not a "proper" second order neighbour since it involves a form of backtracking. The possibilities for taking "scenic" pathways between areas tend to increase as order increases. When raising $\boldsymbol{W}$ to order three, there are many pathways linking area 6 and area 1 in three steps, but we have already defined area 1 as a second order neighbour of area 6, so we do not also want to define it as a third order neighbour too. Anselin and Smirnov (1996) develop an efficient algorithm for sweeping out redundant, backtracking, "scenic" and circular paths for the purpose of constructing the proper second and higher order contiguity matrix from the powered $\boldsymbol{W}$ matrix.

It is also worth noting that raising a symmetric $\boldsymbol{W}$ matrix to any positive integer power yields a symmetric matrix. However, since the row-standardised $\boldsymbol{W}$ is not symmetric, $(\boldsymbol{W}^*)^2$, $(\boldsymbol{W}^*)^3$ and so on are also not symmetric. But when $w_{ij} \geq 0$ for all i and j, and if $w_{ij}^{(k)} = 0$, then $\left(w_{ij}^*\right)^{(k)} = 0$, where $\left(w_{ij}^*\right)^{(k)}$ is the element in row i and column j of $(\boldsymbol{W}^*)^k$ (see Exercise 4.7).

All the above observations on the elements of powered $\boldsymbol{W}$ matrices have significance beyond the calculation of higher order contiguity. As we shall see in the next section, these powered $\boldsymbol{W}$ matrices give rise to complex ripple effects when using the $\boldsymbol{W}$ matrix for local information sharing and assessing the effects of spatial spillover and feedback in spatial econometric modelling.

4.9 Choice of *W* and Statistical Implications

An important role for the $\boldsymbol{W}$ matrix is to turn spatial relationships amongst the N areas (or points) into "data". Through modelling, this form of data supplements the conventional form of data (e.g. data on crime/disease case counts observed across the census tracts in a city, or data on levels of income amongst a set of geo-referenced households within a survey), allowing us to strengthen the estimation of quantities at the small area level, to study how the characteristics of one area might affect those in other areas (i.e. the notion of spatial spillovers and feedbacks), and to obtain a more accurate estimate of the effect of an observable covariate on an outcome. The $\boldsymbol{W}$ matrix plays an integral part in any form of spatial analysis in this book. Whilst we defer the more technical details of the modelling to Part II, here we discuss the implications of the choice of $\boldsymbol{W}$ on specific statistical analyses and problems.

4.9.1 Implications for Small Area Estimation

"Borrowing strength" is a methodology for improving the precision of estimates of small area characteristics such as crime/disease rates or average income levels. If we represent the unknown quantities of interest, say the unknown crime rates in two areas i and j as random variables, and if these two random variables are *positively correlated*, then the information that we have on the crime rate in one area tells us something about the crime rate in the other area. For example, if area i has a low crime rate, as suggested by its small number of reported crime events, then this information can be used (or borrowed/shared) when estimating the crime rate in area j, in addition to using the count of reported crime events in j and vice versa. This is the idea behind *local information sharing*. The stronger the correlation between two random variables, the more information is shared locally between the two areas. Note that negative correlation is also allowed but not often used.

Now, what is the role that the $\boldsymbol{W}$ matrix plays? In Chapter 8, to operationalise local information sharing, we will introduce a class of spatial models based on the conditional autoregressive (CAR) structure, a modelling structure that allows us to impose a set of spatial relationships onto the random variables that represent the quantities of interest (e.g. the small area crime rates). These random variables are also called random effects. The so-called proper CAR (pCAR) model structures these random effects such that the variance-covariance matrix of these random effects takes the following form,

$$\Sigma = \sigma^2 \boldsymbol{D}_w^{-1}\left(\boldsymbol{I}_N - \rho \boldsymbol{W}^*\right)^{-1}, \tag{4.9}$$

where we now see the appearance of $\boldsymbol{W}^*$, the row-standardised $\boldsymbol{W}$ matrix. In Eq. 4.9, σ^2 is a variance parameter, $\boldsymbol{D}_w^{-1}$ is the inverse of $\boldsymbol{D}_w$, an $N \times N$ diagonal matrix (all off-diagonal elements equal to zero) with the diagonal elements equal to the row sums of the $\boldsymbol{W}$ matrix, and ρ is called the spatial autocorrelation parameter. The variance-covariance matrix $\boldsymbol{\Sigma}$, a function of $\boldsymbol{W}$, determines the properties of local information sharing. There are two ways to study the form of $\boldsymbol{\Sigma}$: one is through an infinite regular array (see Besag and Kooperberg, 1995 and Künsch, 1987) and the other is by using a torus (Held and Rue, 2010, p.205–207). Here we describe the latter in more detail whilst referring the reader to the reference provided for details about the former. A torus illustrated in Figure 4.8 is formed, as Held and Rue (2010, p.206) instruct, by wrapping a regular lattice into a "sausage" then joining the two ends of the sausage together. Whilst the geometric shape of a torus hardly ever appears in any social science applications, it has the benefit that when using contiguity (Section 4.2) to construct the $\boldsymbol{W}$ matrix, every pixel on the torus has the same number of neighbours (four in the case of the rook's move or eight in the case of the queen's move). As a result, every single pixel has the same set of correlations with all the other pixels on

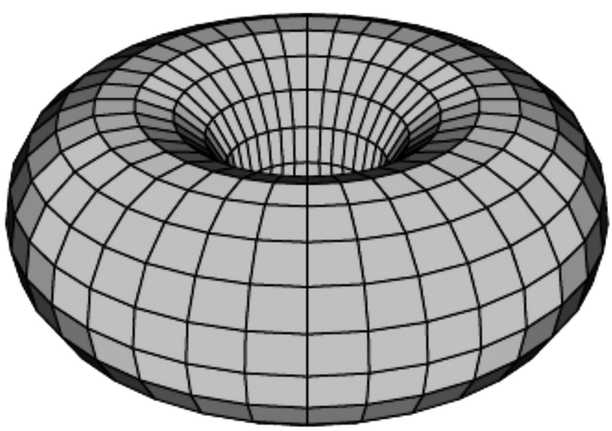


FIGURE 4.8
An illustration of a torus.

the torus. So it is sufficient to study the spatial correlation structure of just one pixel. In the setting considered by Held and Rue (2010), their torus is formed using a 29 × 29 lattice so the $\boldsymbol{W}$ matrix is of size 841 × 841. Their $\boldsymbol{W}$ matrix is defined as a binary matrix based on first order rook's contiguity. Figure 4.9 shows the correlation coefficients of pixel (15, 15) with all other 840 pixels on the torus (see Held and Rue, 2010, p.205–207 for further details).

Three observations can be drawn from Figure 4.9. First, all correlations are non-zero and positive. This means that local information sharing is not restricted to between each pixel and its immediate neighbours defined by $\boldsymbol{W}$, but instead this process spans the entire map, mimicking a "ripple effect". Second, although the random effect in one area is positively correlated with the random effects in all other areas, the strongest correlations are with its immediate neighbours defined by the binary $\boldsymbol{W}$ matrix. For example, in Figure 4.9, the strongest correlation is between pixel (15, 15) and its immediate neighbours (15, 14), (14, 15), (16, 15) and (15, 16). When a non-binary weights matrix is used, those neighbours are the ones that have the largest values within the $\boldsymbol{W}$ matrix. Finally, Figure 4.9 clearly shows the tailing off in spatial correlation with increasing distance separation. In terms of information sharing, the information that area i borrows from area j ($i \neq j$) becomes less as j moves away from i.

These three observations can also be explained mathematically by expanding the matrix inversion, $(\boldsymbol{I}_N - \rho\boldsymbol{W}^*)^{-1}$, in Eq. 4.9. Note that both σ^2, a scalar parameter, and $\boldsymbol{D}_w^{-1}$, a diagonal matrix with positive, non-zero diagonal elements, do not affect the discussion of the off-diagonal elements in $\boldsymbol{\Sigma}$ (the focus of our discussion here). That is because if an off-diagonal element in $(\boldsymbol{I}_N - \rho\boldsymbol{W}^*)^{-1}$ is non-zero and positive, the element at the corresponding position in the variance-covariance matrix is also non-zero and positive.

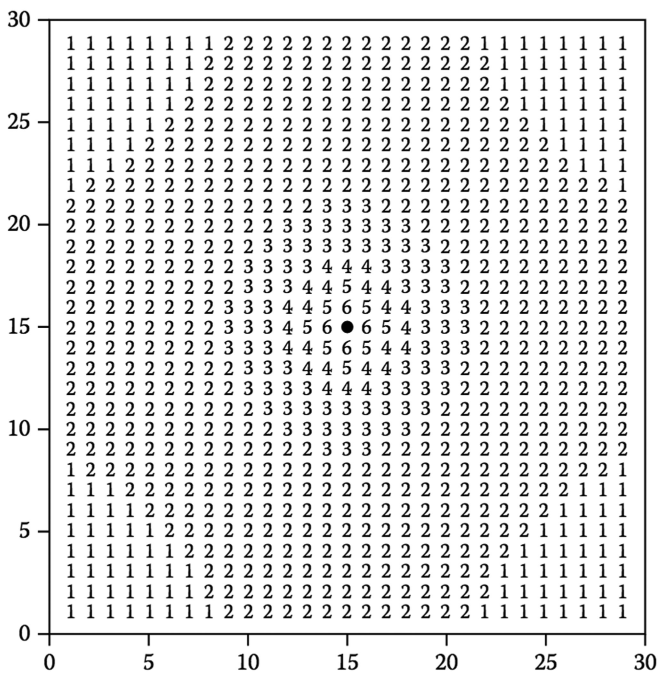


FIGURE 4.9
Plot of the correlation between pixel (15, 15) and all other pixels (i, j) (with $i \neq 15$ and $j \neq 15$) for a first order proper CAR model with parameter $\rho = 0.2496$ on a torus formed by wrapping a 29 × 29 lattice. Shown are the spatial correlations (x10) truncated to an integer (Held and Rue, 2010, p.207).

Providing ρ lies between –1 and 1, i.e. $|\rho| < 1$, a constraint that we will return to in Section 10.3.3.3, $(\boldsymbol{I}_N - \rho \boldsymbol{W}^*)^{-1}$ can be written as the sum of an infinite series:

$$\left(\boldsymbol{I}_N - \rho \boldsymbol{W}^*\right)^{-1} = \boldsymbol{I}_N + \rho \cdot \boldsymbol{W}^* + \rho^2 \cdot \left(\boldsymbol{W}^*\right)^2 + \rho^3 \cdot \left(\boldsymbol{W}^*\right)^3 + \cdots \tag{4.10}$$

From Eq. 4.10, it becomes clear that not only is information borrowed from the first order neighbours it is also borrowed from higher order neighbours due to the terms involving higher orders of the row-standardised $\boldsymbol{W}$ matrix (see discussion in Section 4.8). However, the amount of information borrowing reduces as the order increases, since ρ^k gets smaller as k increases. Therefore, the key points obtained from the expansion are: (a) under the conditional autoregressive structure, information is not only borrowed from the immediate (first order) neighbours as defined by the chosen $\boldsymbol{W}$ but ripples across higher order neighbours and (b) the amount of borrowing reduces as order increases. Dubin (2009) discusses the spatial correlation structures associated with different spatial models and different forms of $\boldsymbol{W}$.

However, the situation becomes more complicated when $\boldsymbol{W}$ refers to the spatial relationships amongst a set of irregular polygons, as Wall (2004) demonstrates using the states of the USA. Table 4.1 shows the correlations between two selected states (Missouri and Tennessee) and their first order contiguous neighbours derived from the proper CAR model (for specification of the model see Wall (2004, p.316)). Although both states have eight neighbours, the correlation varies within each state. For example, the correlation between Missouri and Kansas is stronger compared to that between Missouri and Kentucky. Similarly, Tennessee is more correlated with Alabama than it is with Arkansas. Since both states have the same number of first order neighbours, such differences do not reflect the influence of differences in the numbers of first order neighbours. We should ask the same question that Wall (2004, p.318) poses: "is this reasonable?" Wall also remarks that the variability in these correlations shows no systematic structure and also varies with the size of ρ, the spatial autocorrelation parameter in Eq. 4.9.

The specification of the spatial weights matrix, $\boldsymbol{W}$, determines how information is shared locally. However, we seldom know the *true* spatial relationships of the areas, and thus we need to question the appropriateness of the choice of $\boldsymbol{W}$. For example, if the $\boldsymbol{W}$ matrix is specified based on geographical proximity, are the adjacent areas the most appropriate to

TABLE 4.1

Implied Correlations Between Tennessee and Missouri and their First-Order Neighbours Under the Proper CAR Model

Missouri First Order Neighbours		Tennessee First Order Neighbours	
Arkansas	0.238	Alabama	0.324
Illinois	0.247	Arkansas	0.257
Iowa	0.244	Georgia	0.327
Kansas	0.263	Kentucky	0.229
Kentucky	0.223	Mississippi	0.300
Nebraska	0.248	Missouri	0.216
Oklahoma	0.251	N. Carolina	0.312
Tennessee	0.216	Virginia	0.265

Source: Wall, 2004, p.319.

borrow information from? If we are seeking to improve the precision of small area estimates of social or economic attributes, and if geographically nearby areas, at least on some parts of the map, differ markedly in terms of social or economic characteristics, a better basis for "information borrowing" might be to define the set of neighbours of any area *i* with reference to those attributes that suggest the areas will have similar characteristics. This is the argument for defining the ***W*** matrix using attributes of areas (Section 4.5). For example, Law (2016) defines "distance" between areas by the expected number of fall-induced injuries. The larger the expected number of falls in both areas *i and j*, the "closer" they are considered to be in terms of spatial epidemiological "distance". The larger the weighting (w_{ij}), the stronger the information borrowing from those areas as opposed to other areas, even including those that are geographically adjacent. We shall see some other examples of such attribute-based definitions in Chapter 8.

A consequence of local information sharing is that the resulting map is made smoother looking. However, where pronounced differences exist between adjacent areas, which can be seen by mapping the observed data (Chapter 6), sharing information locally runs the risk of oversmoothing. Figure 4.10 provides an illustration where, if information

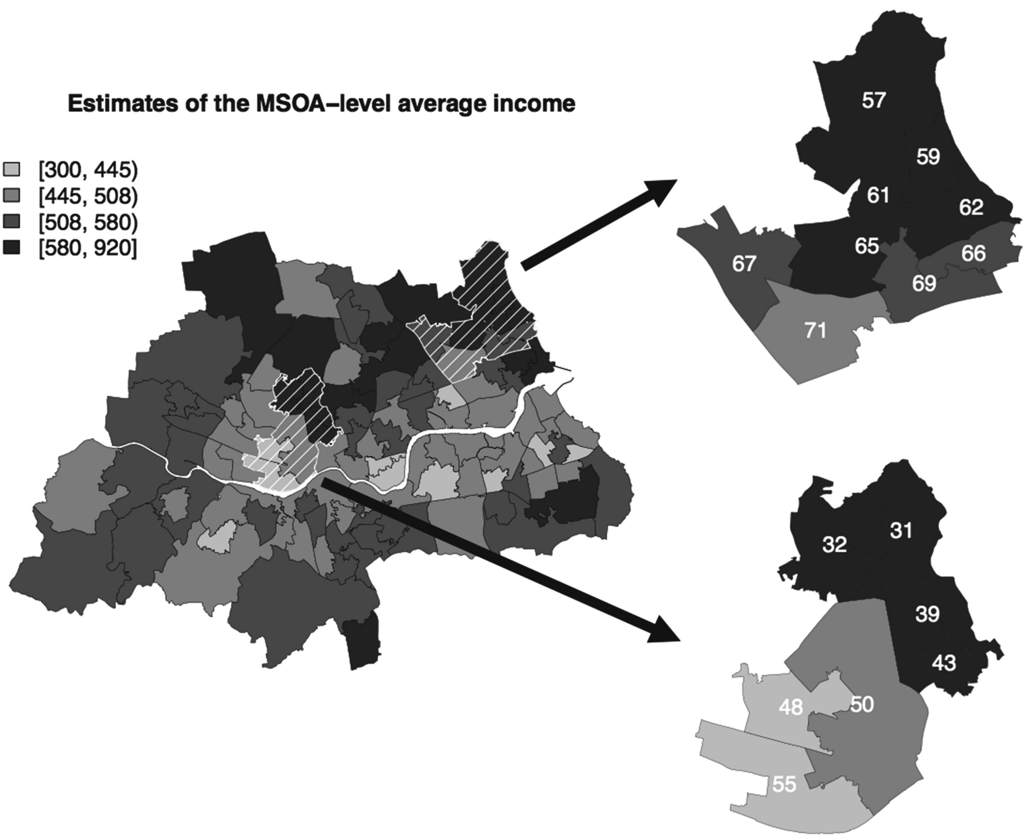


FIGURE 4.10
A map of the estimated average income at the middle super output area (MSOA) level in Newcastle. The income estimates are obtained from a version of the conditional autoregressive model. Whilst the resulting map is generally smooth-looking, there is evidence of the presence of pronounced differences amongst some subsets of spatially-contiguous areas. See Section 8.4 for further detail.

sharing is applied to these data, in some areas of the map (highlighted) there is a risk of masking real differences between adjacent areas. Other forms of bias are possible too, including undersmoothing. Various methods for adaptive spatial smoothing have been proposed to address these problems. Common to these adaptive smoothers is the modelling of the $\boldsymbol{W}$ matrix as opposed to treating it as fixed (or given). We will discuss the modelling of the $\boldsymbol{W}$ matrix in Section 4.10 and will detail some of the spatial adaptive models in Section 8.4.

When borrowing information in estimating a set of spatial-temporal random effects (parameters), the same principles apply. For a dataset with N spatial units and T time periods, if the aim is to improve the precision of the estimates associated with the parameter in each of the $N \times T$ space-time cells (say the crime/disease rate in area i at time t), then it is natural to consider not only how information should be borrowed spatially, but also how to borrow information temporally across the T time periods. As we shall see, borrowing information in time is *bi-directional*, meaning that the estimate at time t will borrow information from *both* $t-1$ *and* $t+1$ – that is, from both the past and the future time points with respect to t. This definition of temporal neighbours is different from the situation where we are modelling the underlying space-time *process*. In that case, we might reasonably assume the present time point t depends on the past time point $t-1$ but not the future time point $t+1$. However, in the context of small area estimation, bi-directional information sharing over time is justifiable and appropriate because we are exploiting the temporal dependence structure present in the dataset in order to improve the estimation of these $N \times T$ parameters. We can extend such an idea further to borrow information *both* spatially *and* temporally. We shall discuss these topics in detail in Part III of the book.

4.9.2 Implications for Spatial Econometric Modelling

An important reason for fitting spatial econometric models is to empirically estimate spatial spillovers and feedbacks (see Section 2.3.1). One of the models in spatial econometrics is the so-called spatial lag (SLM) model, and the mathematical form of this model helps clarify the fundamental role played by the $\boldsymbol{W}$ matrix. Suppose we are modelling a set of random variables, $Y_1,\ldots,Y_N$. Each Y_i represents the outcome of interest in area i within a study region that has N areas. The SLM model is formulated as

$$Y_i = \alpha + \sum\nolimits_{j=1}^{N} b_{ij} Y_j + \beta \cdot x_i + e_i \tag{4.11}$$

Eq. 4.11 essentially defines a regression model with a single observable covariate x_i (although the SLM model can include more than one covariate), and e_i representing an independent error term. The distinctive feature of the SLM model lies within the summation term, $\sum\nolimits_{j=1}^{N} b_{ij} Y_j$. The inclusion of that term allows the outcome variable of each area i to be affected by the outcome variables of other areas. The parameter b_{ij} measures the effect of the outcome variable in area j on the outcome variable in area i. Typically, $b_{ii}=0$, for all $i=1,\ldots,N$, so that the outcome of each area is not allowed to affect itself. It is the estimation of these $N \cdot (N-1)$ parameters (i.e. b_{ij} for $i \neq j$) that enables us to study the spatial interdependence of these areas. However, that poses a challenge: there are more parameters to estimate than available observations. One way to solve the problem is to replace b_{ij} by $\delta \cdot w_{ij}^{*}$, where w_{ij}^{*} is the element from the row-standardised version of a chosen $\boldsymbol{W}$ matrix.

The number of unknown parameters is reduced dramatically from $N \cdot (N-1)$ to just one, namely, δ. The SLM model then becomes

$$Y_i = \alpha + \delta \sum_{j=1}^{N} w_{ij}^{*} Y_j + \beta \cdot x_i + e_i \tag{4.12}$$

As we shall see in Section 10.2.2.3, the SLM model in Eq. 4.12 can be rewritten into the so-called reduced form[4]:

$$\boldsymbol{Y} = \alpha \left(\boldsymbol{I}_N - \delta\, \boldsymbol{W}^{*} \right)^{-1} + \left(\boldsymbol{I}_N - \delta\, \boldsymbol{W}^{*} \right)^{-1} \beta \boldsymbol{x} + \left(\boldsymbol{I}_N - \delta\, \boldsymbol{W}^{*} \right)^{-1} \boldsymbol{e} \tag{4.13}$$

Whilst we shall describe each of the terms in Eq. 4.13 in Section 10.2.2.3, we can immediately see that the matrix inverse $(\boldsymbol{I}_N - \delta \boldsymbol{W}^{*})^{-1}$, the same term as we discussed in Section 4.9.1, comes into play. At this point, we shall highlight some of the properties of this model induced by the $\boldsymbol{W}$ matrix.

First, in Eq. 4.13, the independent error terms (represented by $\boldsymbol{e}$) are multiplied by $(\boldsymbol{I}_N - \delta \boldsymbol{W}^{*})^{-1}$ so that the error terms in the SLM model are no longer independent. In fact, some spatial econometric models (the SLM model is an example) incorporate spatial dependence in the observed outcome values through the likelihood function. This contrasts with hierarchical modelling where spatial dependence in data is modelled through the process model. But we shall defer further discussion of these two approaches to Chapters 7, 8 and 10.

Second, in contrast to a standard regression model, the covariate effect, β, is multiplied by $(\boldsymbol{I}_N - \delta \boldsymbol{W}^{*})^{-1}$. As a result, a change to a covariate's level in area i not only leads to changes to the outcome in the same area (the "direct impact" of LeSage and Pace, 2009), it also leads to changes to the outcomes in all the other areas due to the ripple effect – as we illustrated in Figure 4.9. However, the closer (further away) an area is to the originating area i, the stronger (weaker) is the spillover effect it experiences. This can be seen through the power expansion of $(\boldsymbol{I}_N - \delta \boldsymbol{W}^{*})^{-1}$ in Eq. 4.10.

Finally, the interrelatedness of the outcome variables, as in Eq. 4.12, gives rise to a feedback mechanism due to backtracking. For example, a change in a covariate's level in area i can lead to changes in the outcomes in other neighbouring areas, which in turn affect the outcome in the originating area i (the "indirect impact" of LeSage and Pace, 2009).

Quantifying the direct and indirect impacts of an observable covariate is a topic of importance in certain spatial econometric models. We shall investigate their calculation in Chapter 10 (Section 10.3). However, in identifying these direct and indirect impacts, the above arguments have demonstrated the crucial role played by the (assumed) $\boldsymbol{W}$ matrix.

It is also worth highlighting two identification problems,[5] as listed below, in spatial econometric modelling, both of which are associated with the $\boldsymbol{W}$ matrix:

1. Whilst the specification of the $\boldsymbol{W}$ matrix eases the estimation problem, it cannot be said to test directly which regions interact with one another nor the strength of their interactions (Harris, 2011, p.263). The *true* $\boldsymbol{W}$ matrix is unknown – however

[4] The reduced form of a spatial econometric model is the form of the model in which the outcome variables are expressed as a function of the covariate(s) and the error terms.

[5] An identification problem is the inability in principle to identify the best estimate of the values of one or more parameters in a regression model. Put differently, more than one set of parameters (including those associated with spillover effects) can generate the same distribution of observations.

reasonable it might seem to base its construction on contiguity or distance criteria or interaction data, for example. This has led some authors to stress the importance of theory for specifying $\boldsymbol{W}$ (Davies and Voget, 2008) or at least to assess the robustness of model results to alternative and equally plausible specifications of $\boldsymbol{W}$. There is uncertainty associated with specifying $\boldsymbol{W}$.

2. Even if the model and the $\boldsymbol{W}$ matrix are correctly specified, a second identification problem arises when the unknown parameters of the model cannot be uniquely recovered from the reduced form of the model. Vega and Elhorst (2015, p.341) observe that this type of identification problem can arise with what they call the general nesting spatial model (Section 10.5.1) that contains spatial interaction terms in the outcome variables, the explanatory variables and the errors. An identification problem can also arise if the $\boldsymbol{W}$ matrix is specified using an economic variable (using for example the attribute-based methods discussed in Sections 4.5 and 4.6), and the variation of that economic variable can be explained by any of the variables (covariates or the outcome variable) in the model. This last form of identification problem is avoided if, for example, contiguity or distance criteria can legitimately be used (on economic grounds) to specify $\boldsymbol{W}$.

Vega and Elhorst (2015, p.341) conclude: "the basic identification problem in spatial econometrics is the difficulty to distinguish different models and different specifications of $\boldsymbol{W}$ from each other without reference to specific economic theories."

4.9.3 Implications for Estimating the Effects of Observable Covariates on the Outcome

As we have already noted in Chapter 1 (see examples 1 and 3 in Section 1.3.2), regression modelling can be employed to provide estimates of fixed effect parameters, i.e. the regression coefficients that quantify the effects of observable covariates on the outcome. But suppose, after fitting the model, spatial dependence is still encountered in the model's residuals. Remedial steps include adding a set of spatially structured random effects in the case of hierarchical modelling or specifying a spatially structured error term in the case of spatial econometric modelling. This is often done to adjust the fixed effect estimates for the (unobserved) presence of spatially structured missing covariates that are impacting the outcome. Suppose the fixed effect of interest is the association between an environmental exposure (e.g. air pollution) and stroke deaths, as in Section 1.3.2, Example 1. Suppose, as is often the case, this environmental exposure is itself spatially structured. Having accounted for the spatial dependence structure in the model's residuals by including spatially-structured random effects (or error terms), the estimated exposure-outcome fixed effects relationship will change. Yet the choice of the spatial structure for the residuals, through the use of the $\boldsymbol{W}$ matrix, is typically a modelling assumption, as opposed to something determined from the data. If the spatially structured missing covariates are confounders in the relationship, it will be difficult to assess the true impact of the environmental exposure on the health outcome. This is known as the *problem of spatial confounding* (see Hodges and Reich, 2010; Wakefield, 2007; Paciorek, 2010). Currently, a response is to assess the sensitivity of findings to different forms of the spatial model for the errors, including the choice of the $\boldsymbol{W}$ matrix (Haining et al., 2010).

4.10 Estimating the *W* Matrix

Section 4.9 discusses the importance of the $\boldsymbol{W}$ matrix in spatial modelling. Yet its true form is unknown to us, and hence there is uncertainty associated with its specification. Instead of treating the $\boldsymbol{W}$ matrix as a matrix of fixed numbers, various methods, in the literature of both hierarchical modelling and spatial econometrics, have been proposed to estimate the $\boldsymbol{W}$ matrix from data.

Methods for estimating $\boldsymbol{W}$ have been described in the spatial econometrics literature in the context of model selection. One strategy proposed is to fit the model in hand with the $\boldsymbol{W}$ matrix specified by a set of plausible configurations and then select the best configuration through model comparisons (see, for example, Seya et al., 2013 and LeSage and Pace, 2009). When the $\boldsymbol{W}$ matrix is specified using distance-based methods (such as those in Sections 4.2 and 4.3), the parameters, for example, a in Eq. 4.1 and λ and γ in Eq. 4.2, could be estimated rather than simply prespecified by the analyst. Such estimation methods have been implemented by and within a wider class of geostatistical models by Diggle et al. (1998). See also Vega and Elhorst (2015), who specify the elements in $\boldsymbol{W}$ using method (ii) in Eq. 4.2, then estimate the power γ from data.

Methods for estimating the $\boldsymbol{W}$ matrix have also been considered in the spatial epidemiology literature in the context of obtaining small area estimates of disease rates in order to reduce the risk of oversmoothing. Lee and Mitchell (2013) describe a method using locally adaptive spatial smoothing. Their method seeks to take into account spatial heterogeneity in disease risk, where for some parts of the map a smooth transition (from one area to another) is evident between adjacent neighbours, whilst in other parts abrupt step changes (or boundaries) are observed.[6] In the former case, it is appropriate to borrow information from the adjacent neighbour; in the latter case, not. Identifying step changes may also be helpful in defining the location of disease clusters as well as the aetiological factors contributing to the variation in risk. The basis of their approach is to treat the non-zero elements in $\boldsymbol{W}$ as random variables (these are the pairs of areas that are contiguous) whilst the zero elements in $\boldsymbol{W}$ remain fixed at zero. If $w_{ij}=1$, this implies that areas i and j are correlated (see Section 4.9.1) and the random effects for both areas will be locally smoothed according to the neighbourhood structure. If $w_{ij}=0$ then areas i and j are conditionally independent. The approach of Lee and Mitchell (2013) is to define the $\boldsymbol{W}$ matrix as a matrix containing random elements, then to construct a decision rule based on the marginal posterior distributions of the corresponding random effects for deciding whether each non-zero weight element should be reset to zero. We will present details of their modelling in Section 8.4. But it is important to realise that resetting $w_{ij}=0$ (indeed under any of the methods described where $w_{ij}=0$) does *not* mean there will be no local smoothing (information sharing) between i and j, but rather the smoothing will be much reduced and will depend on, for example, the geography of step changes involving the neighbours of the neighbours. As Section 4.9.1 describes, there is a "ripple effect" associated with the smoothing process when using a spatial model such as the proper CAR model (see Chapter 8 for more detail).

[6] The detection of step changes on a map is referred to as "wombling", after Womble (1951).

4.11 Concluding Remarks

The $\boldsymbol{W}$ matrix summarizes our modelling assumptions about the spatial relationships between the areas (or fixed points or sites or polygons) to which data values refer in the database. Defining relationships between areas in geographical space raise many more challenges than the equivalent problem in time series analysis. Time series data are usually recorded at regular intervals, with time imposing a natural ordering. By contrast, as we have seen, there are many ways of defining spatial relationships, from those based purely on geometric properties to those that reflect, in some sense, real similarities in terms of attribute values or real interactions between places. We have also presented approaches based on estimating the non-zero elements in the $\boldsymbol{W}$ matrix. Many of these methods for defining $\boldsymbol{W}$ will appear again in Part II of the book.

Constructing the $\boldsymbol{W}$ matrix carries significant implications for map smoothing, small area estimation, spatial econometric modelling, as well as regression modelling more generally. Those implications differ between hierarchical modelling and spatial econometric modelling because of the way spatial dependence is specified in the two classes of models – in the process model in the first case, in the likelihood function in the second. As a consequence, the "rippling" effect associated with spatial models, such as the pCAR model, and which depend on how $\boldsymbol{W}$ is specified, have different consequences in the two methodologies. We shall have more to say about the two modelling approaches in Part II, especially in Section 10.5.2.

4.12 Exercises

Exercise 4.1. What is the "identifiability problem" associated with the weights matrix ($\boldsymbol{W}$) in spatial econometrics?

Exercise 4.2. Explain and give an example of the "problem of spatial confounding" in regression modelling.

Exercise 4.3. Describe circumstances where it is better to work with the row-standardised form of the weights matrix rather than the unstandardised form.

Exercise 4.4. Obtain $\boldsymbol{W}^2$, $\boldsymbol{W}^3$ and $\boldsymbol{W}^4$ for the map shown in Figure 4.1 based on binary weights under queen's move contiguity. Inspect a selection of the elements in each matrix and identify all the different pathways that link the corresponding pairs of polygons.

Exercise 4.5. For $\boldsymbol{W}^2$ and $\boldsymbol{W}^3$, obtained from Exercise 4.4, identify the second and third order neighbours of each polygon (if any), that would be used to compute second and third order measures of spatial autocorrelation.

Exercise 4.6. We presented a form of the spatial lag (SLM) model in Eq. 4.12. The model, without any covariates, has the form ($i=1,\ldots,N$):

$$Y_i = \alpha + \delta \sum_{j=1}^{N} w_{ij}^{*} Y_j + e_i .$$

(i) Rewrite the model in matrix notation (using the $\boldsymbol{W}$ matrix).

(ii) Specify the model on the map given by Figure 4.1 and for the associated binary queen's contiguity matrix shown in Figure 4.1. Trace the spillover and feedback effects (using both the mathematical form of the SLM model and the map) that spread from area 6 following a one unit increase in the value of Y_6.

(iii) Consider the full version of the SLM model, where a set of k observable covariates (exogenous variables), $\boldsymbol{X}_i = (x_{i1},\ldots,x_{ik})$, are included:

$$Y_i = \alpha + \delta \sum\nolimits_{j=1}^{N} w_{ij}^{*} Y_j + \boldsymbol{X}_i \boldsymbol{\beta} + e_i .$$

Write this model in matrix notation and re-arrange terms to give the reduced form of the model (in the form of Eq. 4.13).

Exercise 4.7. Let $w_{ij}^{(k)}$ and $\left(w_{ij}^{*}\right)^{(k)}$ be the elements on row i column j of $\boldsymbol{W}^k$ and $\left(\boldsymbol{W}^{*}\right)^{k}$, respectively. Show that if $w_{ij}^{(k)} = 0$, then $\left(w_{ij}^{*}\right)^{(k)} = 0$ when $w_{ij} \geq 0$ for all i and j.

4.13 Appendices

Appendix 4.13.1 Building a Geodatabase in R

This appendix introduces some of the aspects associated with constructing a geodatabase in R. Figure 4.11 illustrates the particular geodatabase that we are focusing on here. Each row in a dataset is linked to the corresponding geographical features in a shapefile. A shapefile is a commonly used file format for storing the geometric and attribute information of some geographical features. These geographical features can represent areas (such as census tracts) or points (such as the locations of petrol stations or retail units in a city). In the case of areas, the geometric information contains multiple sets of coordinates that form the boundaries of the areas, whereas in the case of points, the geometric information stored in a shapefile contains the coordinates of the points. A shapefile also contains some attribute information about the geographical features, for example, the names of the areas (or the petrol stations). There are several online resources from which shapefiles can

name	burgCases	houses	unemploy
1	754	3100	5.35
2	782	3420	6.74
…			

FIGURE 4.11
Linking a (external spatial or spatial-temporal) dataset to a map.

be obtained.[7] For example, the UK Data Service provides digitised boundary data for different layers within the census geography (https://census.ukdataservice.ac.uk/use-data/guides/boundary-data). The U.S. Census Bureau also provides a similar service (https://www.census.gov/geo/maps-data/data/tiger-cart-boundary.html).

In addition to the shapefile, we also have a dataset that contains our spatial or spatial-temporal data (e.g. the number of burglary events or disease cases observed in each area within the study region over a number of years), where the rows in this dataset are georeferenced to the areas (or points) in the given shapefile. However, quite often, such a dataset is given to us in a file that is separate from the shapefile. So our first task is to use R to link this dataset to the shapefile.

Solving this task involves the following four steps:

1. Read the shapefile into R.
2. Read the spatial (or spatial-temporal) dataset into R.
3. Link the shapefile with the spatial (or spatial-temporal) dataset.
4. Save the resulting shapefile.

To illustrate the above four steps, we use the example given in Figure 4.11. There are two files that we need to download from the book's website. The first file, `TenAreas.zip`, contains the relevant files to create the map in Figure 4.11. The second file, `data_for_TenAreas.csv`, is a spatial dataset where each row is georeferenced to each of the 10 areas in the above map. There are four variables (attributes) in this spatial dataset:

- `name`: the names of the areas
- `burgCases`: the number of burglary cases reported in each area
- `houses`: the number of houses at-risk in each area
- `unemploy`: the percentage of unemployed residents in each area

To proceed, create a folder called `sptmbook` on the `C` drive of your computer. Download the two files as described above and save them to the folder that you just created. Unzip the file `TenAreas.zip` so that three additional files appear in the folder: `TenAreas.shp`, `TenAreas.dbf` and `TenAreas.shx`.[8] The R code below reads the shapefile into R:

```
1  setwd('c:/sptmbook/') # specify the working directory
2  library(maptools) # load the maptools library in
3  shp <- readShapePoly('TenAreas.shp') # read the shapefile into R
4  plot(shp) # produce a simple map
5  str(shp@data) # show structure of the attribute data in the shapefile
```

Line 1 uses the `setwd` function to specify the so-called working directory where the shapefile and the data file are stored. This simplifies the coding later, because when we read a file into R, we only need to point R to the file name without specifying the full directory

7 There may be circumstances where you need to construct your own shapefile. Constructing shapefiles is beyond the scope of this book. For advice see, for example: http://gis.yohman.com/up206a/how-tos/how-to-create-your-own-shapefile/ or https://docs.qgis.org/2.8/en/docs/training_manual/create_vector_data/create_new_vector.html.

8 A shapefile consists of three mandatory files with the file extensions of .shp, .shx and .dbf. These three files must be in the same folder when reading a shapefile into R.

every time. For example, without setting the working directory, Line 3 would become `shp <- readShapePoly('c:/sptmbook/TenAreas.shp')`, where we have to specify the full directory in the function's argument. Line 2 loads the R package `maptools`, a package in R containing a collection of R functions for handling spatial objects. If this is the first time you are using this package (or indeed using R), you need to install the package by typing the command `install.packages('maptools')` prior to running the R code above. Line 3 uses the `readShapePoly` function (from the `maptools` package) to read the shapefile into R. Then Line 4 displays the map as shown in Figure 4.11 (without the labelling), and Line 5 shows the structure of the attribute data stored in that shapefile. In particular, the syntax `shp@data` accesses the attribute data, and the R function `str` displays the structure of the data. R returns the following output, showing that there is one variable called `areaName` stored in the shapefile, and this variable consists of the names of the areas (or polygons). The first polygon is called A1, the second polygon is called A2 and so on.

```
'data.frame': 10 obs. of  1 variable:
 $ areaName: chr  "A1" "A2" "A3" "A4" ...
```

It is important to make a note of the name of this variable, as it will be used as the variable to merge the shapefile with the other dataset. In other words, this variable is the *link* between the shapefile and the spatial dataset that we will look at next.

To read the data file, `data_for_TenAreas.csv`, into R, we run the following R code:

```
6  # read the data into R
7  crimeData <- read.csv('data_for_TenAreas.csv',header=TRUE)
8  crimeData # display the dataset on screen; you can also use
9   # str(crimeData) to show the structure of the dataset
```

Here, the spatial dataset is in a csv (comma-separated-values) file, so we use the R function `read.csv` to read the data in. The second argument `header` in the `read.csv` function is set to `TRUE` (all capital letters), because the first line in the `data_for_TenAreas.csv` file contains the column names. So, in practice, it is useful to inspect the data file before importing it into R. When the dataset is provided as an Excel file, one can export the dataset to a csv file (using the `Save As ...` option in Excel). Then run the R commands as given here. Another important point is that, in the spatial dataset, the column containing the names of the areas is labelled as `name`. However, the name of this column has to match the name of the column in the shapefile (which is labelled as `areaName` there). The following line of R code solves the problem by simply changing the name of the first column in `crimeData` to `areaName`:

```
10  colnames(crimeData)[1] <- 'areaName'
```

We are now in position to merge the shapefile with the dataset `crimeData`. This is done using the `merge` function. Note that the setting of the third argument, `by='areaName'`, tells R to match the rows in `crimeData` with the polygons in `shp` using the variable `areaName` so that the row for A1 in `crimeData` is assigned to the polygon called A1 and the row for A2 in `crimeData` is assigned to the polygon called A2 and so on.

```
11  new.shp <- merge(shp,crimeData,by='areaName')
12  new.shp@data
```

The R output below from running the code on Line 12 confirms the correct matching.

```
   areaName SP_ID burgCases houses unemploy
         A1    A1       754   3100     5.35
         A2    A2       782   3420     6.74
         A3    A3       700   4480     8.29
         A4    A4       470   1930     5.09
         A5    A5       557   2150     7.91
         A6    A6       896   4020     5.50
         A7    A7       305   3120     6.27
         A8    A8       274   3320     5.71
         A9    A9       182   3490     6.56
        A10   A10       287   3360     4.06
```

Now, run the following in R to save the resulting shapefile to the `sptmbook` folder:

```
writePolyShape(new.shp,'TenAreas_with_data.shp')
```

This completes our task of constructing a geodatabase.

Appendix 4.13.2 Constructing the *W* Matrix and Accessing Data Stored in a Shapefile

This appendix focuses on the following two tasks. The first task is to derive a spatial weights matrix ***W*** using a shapefile. The second task is to access some relevant columns of data stored in the shapefile in order to carry out some calculations and/or to export the data for WinBUGS modelling. Using the shapefile constructed in Appendix 4.13.1, we illustrate how to accomplish the first task.

First, we read the modified shapefile into R. Go through the material in Appendix 4.13.1 if you have not already done so.

```
setwd('c:/sptmbook/') # specify the working directory
library(maptools)     # load the maptools library
# read the modified shapefile into R
new.shp <- readShapePoly('TenAreas_with_data.shp')
```

The function `poly2nb` in the `spdep` package derives the neighbourhood structure from a shapefile. So Line 5 in the R code below loads the required package into R, then Line 6 obtains the first order neighbours of each area using queen's move contiguity. If `queen=FALSE` in the `polyg2nb` function, rook's move contiguity is then used. Line 7 (`str(nb)`) displays the structure of the resulting R object, `nb`.

```
5    library(spdep) # load the spdep library
6    nb <- poly2nb(new.shp,queen=TRUE)
7    str(nb)
```

As shown below, `nb` is a list with 10 elements, and each element is an array containing the first order neighbours of an area. For example, the first array in the list `nb` tells us that the first order neighbours of area 1 are areas 2, 4 and 5. The second array tells us that the first order neighbours of area 2 are areas 1, 3, 5 and 6.

```
List of 10
 $ : int [1:3] 2 4 5
 $ : int [1:4] 1 3 5 6
 $ : int [1:3] 2 5 6
 $ : int [1:3] 1 5 7
 $ : int [1:6] 1 2 3 4 6 7
 $ : int [1:6] 2 3 5 7 8 9
 $ : int [1:5] 4 5 6 9 10
 $ : int [1:2] 6 9
 $ : int [1:4] 6 7 8 10
 $ : int [1:2] 7 9
 - attr(*, "class")= chr "nb"
 - attr(*, "region.id")= chr [1:10] "0" "1" "2" "3" ...
 - attr(*, "call")= language poly2nb(pl = new.shp, queen = TRUE)
 - attr(*, "type")= chr "queen"
 - attr(*, "sym")= logi TRUE
```

Using the neighbourhood structure in `nb`, we can now derive various versions of the ***W*** matrix via the `nb2mat` function:

```
W <- nb2mat(nb, style='B')  # binary (0/1) weights
W                           # display W on screen
std.W <- nb2mat(nb, style='W') # row-standardised W
std.W                          # display std.W on screen
```

In the above code, `W` is the ***W*** matrix shown in Figure 4.1, and `std.W` is the row-standardised version given in Figure 4.6. The second argument `style` in the `nb2mat` function determines whether a set of binary (0/1) weights are used (by setting `style='B'`) or the weights are row-standardised (by setting `style='W'`). Higher order neighbours can be obtained using the `nblag` function. For example, the R code, `higher.order.neighbours <- nblag(nb,2)`, finds the neighbours of each area up to and including the second order. The second argument of the `nblag` function determines the maximum order of the neighbourhood structure to be constructed. Type `str(higher.order.neighbours)` into R to show these neighbours.

We can also calculate the ***W*** matrix raised to some power in R. For example, `W.squared <- W%*%W` calculates $\boldsymbol{W}^2$ and `W.power.three <- W%*%W%*%W` calculates $\boldsymbol{W}^3$ and so on. Note that the R syntax `%*%` performs a matrix multiplication.

As we shall discuss in Chapter 8, the intrinsic conditional autoregressive (ICAR) model and the proper conditional autoregressive (pCAR) model are two spatial models that allow us to impose a spatial structure on a set of unit-specific parameters, one parameter for each spatial unit. Whilst we will discuss the statistical properties of these spatial models in Chapter 8, the implementation of these models in WinBUGS requires us to input the chosen spatial weights matrix as data. In particular, when fitting the ICAR model, the `car.normal` function in WinBUGS is used and the ***W*** matrix enters the `car.normal` function via three data arrays, namely, `num[]`, `adj[]` and `weights[]`. The array `num[]` is of length N, which is the number of areas in the study region. The ith element in the array `num[]` shows the number of neighbours that area i has. The IDs of the neighbours and their associated weights are stored in `adj[]` and `weights[]`, respectively. Both `adj[]` and `weights[]` have the same length, which is the total number of neighbours across all areas.

These three arrays can be obtained in R using the `nb2WB` function. Running the following two lines of R code, we have

```
12  spatial.data.for.WinBUGS <- nb2WB(nb)
13  spatial.data.for.WinBUGS # display the results on screen
```

```
$adj
 [1]  2  4  5  1  3  5  6  2  5  6  1  5  7  1  2  3  4  6  7  2  3  5  7  8  9  4
[27]  5  6  9 10  6  9  6  7  8 10  7  9
$weights
 [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
$num
 [1] 3 4 3 3 6 6 5 2 4 2
```

The resulting R object, `spatial.data.for.WinBUGS`, is a list containing the three arrays that WinBUGS requires. We shall defer the detailed explanation of the values in each array to Section 8.2.1.2.

We now turn our attention to the second task: accessing data values stored in a shapefile. Suppose we want to produce a choropleth map of a variable (or attribute) that is some function of a variable (or several variables) stored in a shapefile. To do that, we first need to extract the relevant columns of data from the shapefile, then perform the calculation. Take the example of calculating the burglary rates – the burglary count in each area divided by the corresponding number of houses – across the 10 areas in Figure 4.11. The three lines of R code below carry out this calculation:

```
14  y <- new.shp@data$burgCases   # extract the burglary counts
15  pop <- new.shp@data$houses    # extract the numbers of houses
16  burgRate <- y/pop             # calculate the burglary rates
```

Again, the R syntax, `new.shp@data`, accesses the dataset (the attribute values) stored in the shapefile, and `new.shp@data$burgCases` accesses the column labelled as `burgCases` in that dataset. Line 16 then calculates the burglary rates as required. We will come back to the above procedure in Section 6.2.1, where we show how to create a choropleth map in R.

Another reason for accessing the data values in a shapefile is to export them from R to WinBUGS. Suppose we want to fit a model in WinBUGS taking the burglary counts, the numbers of houses and the spatial neighbourhood structure as input data. We need to put these data values into a list in R, then write the list of data to a file. The following lines of R code accomplish that:

```
17  # use the length function to obtain the number of areas
18  # in the study region
19  nareas <- length(new.shp@data$burgCases)
20  data.for.WinBUGS <- list(N=nareas,cases=new.shp@data$burgCases
21   ,pop=new.shp@data$houses)
22  data.for.WinBUGS <- c(data.for.WinBUGS,spatial.data.for.WinBUGS)
```

In particular, Lines 20 to 21 create a list with the burglary case counts, the numbers of houses and the number of areas. Line 22 combines the list `data.for.WinBUGS` with the list `spatial.data.for.WinBUGS`, where the latter contains the neighbourhood structure.

To write the resulting data list to a file, we use

```
23 library(R2WinBUGS)
24 write.datafile <- R2WinBUGS:::write.datafile
25 write.datafile(data.for.WinBUGS,towhere='WinBUGSdata.txt')
```

Line 25 writes the data list (`data.for.WinBUGS`) to a file called `WinBUGSdata.txt` in the working directory (which is `c:/sptmbook/`). The R function used there is `write.datafile`. However, that function is a hidden function (a non-visible function) in the R package `R2WinBUGS`. Before the hidden function can be used, Line 23 loads the package, then Line 24 accesses this hidden function from the package (`R2WinBUGS:::write.datafile`) and makes it visible. We can then use it on Line 25.