10

Modelling Areal Data

10.1 Introduction

We have seen in Chap. 9 that the lack of independence between observations in spatial data – spatial autocorrelation – is commonplace, and that tests are available. In an ideal world, one would prefer to gather data in which the observations were mutually independent, and so avoid problems in inference from analytical results. Most applied data analysts, however, do not have this option, and must work with the data that are available, or that can be collected with available technologies. It is quite often the case that observations on relevant covariates are not available at all, and that the detection of spatial autocorrelation in data or model residuals in fact constitutes the only way left to model the remaining variation.

In this chapter, we show how spatial structure in dependence between observations may be modelled, in particular for areal data, but where necessary also using alternative representations. We look at spatial econometrics approaches separately, because the terminology used in that domain differs somewhat from other areas of spatial statistics. We cover spatial filtering using Moran eigenvectors and geographically weighted regression in this chapter, but leave Bayesian hierarchical models until Chap. 11.

The problems we face when trying to fit models in the presence of spatial autocorrelation are challenging, not least because the spatial autocorrelation that we seem to have found may actually come from model misspecification (see Sect. 9.4). If this is the case, effort spent on modelling the spatial structure would be better used on improving the model itself, perhaps by handling heteroskedasticity, by adding a missing covariate, by revisiting the functional form of included covariates, or by reconsidering the distributional representation of the response variable.

10.2 Spatial Statistics Approaches

Spatial dependence can be modelled in different ways using statistical models. In many cases, it is common to assume that observations are independent and identically distributed, but this may not be the case when working with spatial data. Observations are not independent because there may exist some correlation between neighbouring areas. It may also be difficult to pick apart the impact of spatial autocorrelation and spatial differences in the distribution of the observation. Cressie (1993, pp. 402–448, 458–477, 548–568) provides a very wide discussion of these approaches, including reviews of the background for their development and comprehensive worked examples. Schabenberger and Gotway (2005, pp. 335–348) and Waller and Gotway (2004, pp. 362–380) concentrate on the spatial autoregressive models to be used in this section. Wall (2004) provides a useful comparative review of the ways in which spatial processes for areal data are modelled. Banerjee et al. (2004, pp. 79–87) also focus on these models, because the key features carry through to hierarchical models. Fortin and Dale (2005, pp. 229–233) indicate that spatial autoregressive models may play a different role in ecology, although reviews like Dormann et al. (2007) suggest that they may be of use.

In this section, we have followed Waller and Gotway (2004, Chap. 9) quite closely, as their examples highlight issues such as transforming the response variable and using weights to try to handle heteroskedasticity.

From a statistical point of view, it is possible to account for correlated observations by considering a structure of the following kind in the model. If the vector of response variables is multivariate normal, we can express the model as follows:

$$Y = \mu + e,$$

where μ is the vector of area means, which can be modelled in different ways and e is the vector of random errors, which we assume is normally distributed with zero mean and generic variance V. The mean is often supposed to depend on a linear term on some covariates X, so that we will substitute the mean by $X^{\mathrm{T}}\beta$ in the model. On the other hand, correlation between areas is taken into account by considering a specific form of the variance matrix V.

For the case of non-Normal variables, we could transform the original data to achieve the desired Normality. Hence, the techniques described below can still be applied on the transformed data. In principle, many correlation structures could be feasible in order to account for spatial correlation. However, we focus on two approaches that are commonly used in practise,such as SAR (Simultaneous Autoregressive) and CAR (Conditionally Autoregressive) models.

In Chap. 9, we took the mean of the counts of leukaemia cases by tract as our best understanding of the data generation process, supplementing this with the constant risk approach to try to handle heterogeneity coming from variations in tract populations. One of the alternatives examined by Waller and Gotway (2004, p. 348) is to take a log transformation of the rate:

$$Z_i = \log \frac{1000(Y_i + 1)}{n_i}.$$

The transformed incidence proportions are not yet normal, with three outliers, tracts with small populations but unexpectedly large case counts. They could be smoothed away, but may in fact be interesting, as the patterns they display may be related to substantive covariates, such as closeness to TCE locations. As covariates, we have used the inverse distance to the closest TCE (PEXPOSURE), the proportion of people aged 65 or higher (PCTAGE65P) and the proportion of people who own their own home (PCTOWNHOME).

To set the scene, let us start with a linear model of the relationship between the transformed incidence proportions and the covariates. Note that most model fitting functions accept `Spatial*DataFrame` objects as their `data` argument values, and simply treat them as regular `data.frame` objects. This is not by inheritance, but because the same access methods are provided (see p. 35).

```
> library(spdep)

> nylm <- lm(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME, data = NY8)
> summary(nylm)

Call:
lm(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME, data = NY8)

Residuals:
    Min      1Q  Median      3Q     Max
-1.7417 -0.3957 -0.0326  0.3353  4.1398

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.5173     0.1586   -3.26   0.0012 **
PEXPOSURE     0.0488     0.0351    1.39   0.1648
PCTAGE65P     3.9509     0.6055    6.53  3.2e-10 ***
PCTOWNHOME   -0.5600     0.1703   -3.29   0.0011 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.657 on 277 degrees of freedom
Multiple R-squared: 0.193,  Adjusted R-squared: 0.184
F-statistic: 22.1 on 3 and 277 DF,  p-value: 7.3e-13

> NY8$lmresid <- residuals(nylm)
```

Figure 10.1 shows the spatial distribution of residual values for the study area census tracts. The two census variables appear to contribute for explaining the variance in the response variable, but exposure to TCE does not. Moreover, although there is less spatial autocorrelation in the residuals from

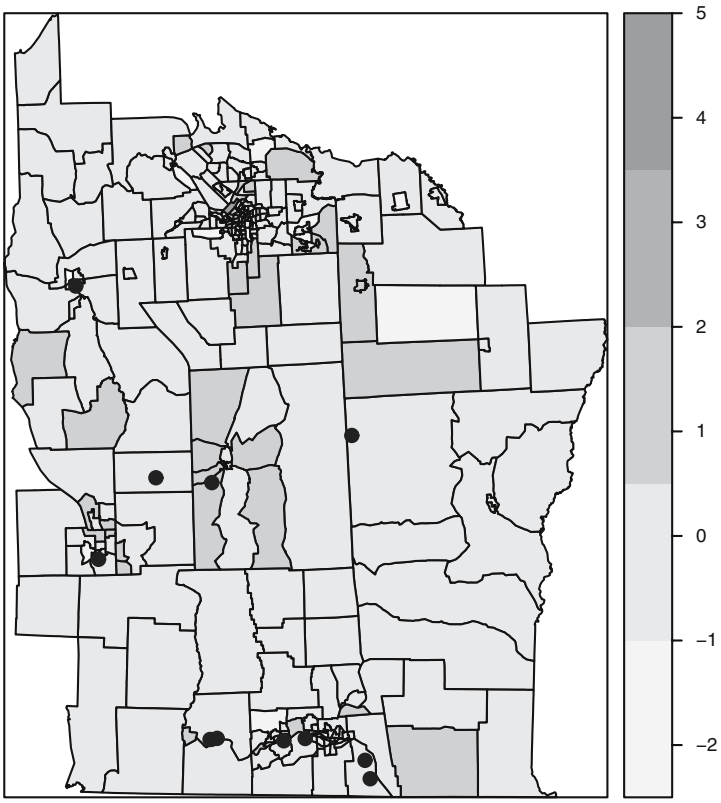

Fig. 10.1. Residuals from the linear model of transformed incidence proportions; TCE site locations shown for comparative purposes

the model with covariates than in the null model, it is clear that there is information in the residuals that we should try to use. An exact test for spatial autocorrelation in the residuals leads to similar conclusions.

Since the Moran test is intended to detect spatial autocorrelation, we can try to fit a model taking this into account. We should not, however, forget that the misspecifications detected by Moran's I can have a range of causes (see Sect. 9.4). It is also the case that if the fitted model exhibits multi-collinearity, the results of the test may be affected because of the numerical consequences of the model matrix not being of full rank for the expectation and variance of the statistic.

```
> NYlistw <- nb2listw(NY_nb, style = "B")
> lm.morantest(nylm, NYlistw)

    Global Moran's I for regression residuals

data:
```

```
model: lm(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
data = NY8)
weights: NYlistw

Moran I statistic standard deviate = 2.638, p-value = 0.004169
alternative hypothesis: greater
sample estimates:
Observed Moran's I        Expectation           Variance
          0.083090          -0.009891           0.001242
```

10.2.1 Simultaneous Autoregressive Models

The SAR specification uses a regression on the values from the other areas to account for the spatial dependence. This means that the error terms ε are modelled so that they depend on each other in the following way:

$$e_i = \sum_{i=1}^{m} b_{ij} e_i + \varepsilon_i.$$

Here, ε_i are used to represent residual errors, which are assumed to be independently distributed according to a Normal distribution with zero mean and diagonal covariance matrix Σ_ε with elements $\sigma^2_{\varepsilon_i}, i = 1, \ldots, m$ (the same variance σ^2_ε is often considered though). The b_{ij} values are used to represent spatial dependence between areas. b_{ii} must be set to zero so that each area is not regressed on itself.

Note that if we express the error terms as $e = B(Y - X^{\mathrm{T}}\beta) + \varepsilon$, the model can also be expressed as

$$Y = X^{\mathrm{T}}\beta + B(Y - X^{\mathrm{T}}\beta) + \varepsilon.$$

Hence, this model can be formulated in a matrix form as follows:

$$(I - B)(Y - X^{\mathrm{T}}\beta) = \varepsilon,$$

where B is a matrix that contains the dependence parameters b_{ij} and I is the identity matrix of the required dimension. It is important to point out that in order for this SAR model to be well defined, the matrix $I - B$ must be non-singular.

Under this model, Y is distributed according to a multivariate normal with mean

$$E[Y] = X^{\mathrm{T}}\beta$$

and covariance matrix

$$\mathrm{Var}[Y] = (I - B)^{-1}\Sigma_\varepsilon(I - B^{\mathrm{T}})^{-1}.$$

Often Σ_ε is taken to depend on a single parameter σ^2, so that $\Sigma_\varepsilon = \sigma^2 I$ and then $\mathrm{Var}[Y]$ simplifies to

$$\mathrm{Var}[Y] = \sigma^2 (I - B)^{-1} (I - B^{\mathrm{T}})^{-1}.$$

It is also possible to specify Σ_ε as a diagonal matrix of weights associated with heterogeneity among the observations.

A useful re-parametrisation of this model can be obtained by writing $B = \lambda W$, where λ is a spatial autocorrelation parameter and W is a matrix that represents spatial dependence – it is often assumed to be symmetric. These structures can be chosen among those described in Chap. 9. With this specification, the variance of Y becomes

$$\mathrm{Var}[Y] = \sigma^2 (I - \lambda W)^{-1} (I - \lambda W^{\mathrm{T}})^{-1}.$$

These models can be estimated efficiently by maximum likelihood. In R this can be done by using function `spautolm` in package `spdep`. The model can be specified using a formula for the linear predictor, whilst matrix W must be passed as a `listw` object. To create this object from the list of neighbours we can use function `nb2listw`, which will take an object of class `nb`, as explained in Chap. 9.

The following code shows how to fit a simultaneous autoregression to the chosen model. We have fitted the standard model and the weighted model using the population size in 1980 (according to the US Census) in the areas as weights. This reproduces the example developed in Waller and Gotway (2004, Chap. 9, pp. 375–379), and the reader is referred to their discussion for more information. In the call to `nb2listw`, we specified `style = "B"` to construct W using a binary indicator of neighbourhood.

```
> nysar <- spautolm(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     data = NY8, listw = NYlistw)
> summary(nysar)

Call:
spautolm(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME, data = NY8,
    listw = NYlistw)

Residuals:
      Min        1Q    Median        3Q       Max
-1.567536 -0.382389 -0.026430  0.331094  4.012191

Coefficients:
             Estimate Std. Error z value  Pr(>|z|)
(Intercept) -0.618193   0.176784 -3.4969 0.0004707
PEXPOSURE    0.071014   0.042051  1.6888 0.0912635
PCTAGE65P    3.754200   0.624722  6.0094 1.862e-09
PCTOWNHOME  -0.419890   0.191329 -2.1946 0.0281930
```

```
Lambda: 0.04049 LR test value: 5.244 p-value: 0.022026

Log likelihood: -276.1
ML residual variance (sigma squared): 0.4139, (sigma: 0.6433)
Number of observations: 281
Number of parameters estimated: 6
AIC: 564.2
```

According to the results obtained it seems that there is significant spatial correlation in the residuals because the estimated value of λ is 0.0405 and the p-value of the likelihood ratio test is 0.0220. In the likelihood ratio test we compare the model with no spatial autocorrelation (i.e. $\lambda = 0$) to the one which allows for it (i.e. the fitted model with non-zero autocorrelation parameter).

The proximity to a TCE seems not to be significant, although its p-value is close to being significant at the 95% level and it would be advisable not to discard a possible association and to conduct further research on this. The other two covariates are significant, suggesting that census tracts with larger percentages of older people and with lower percentages of house owners have higher transformed incidence rates.

However, this model does not account for the heterogeneous distribution of the population by tracts beyond the correction introduced in transforming incidence proportions. Weighted version of these models can be fitted so that tracts are weighted proportionally to the inverse of their population size. For this purpose, we include the parameter `weights=POP8` in the call to the function `lm`.

```
> nylmw <- lm(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME, data = NY8,
+     weights = POP8)
> summary(nylmw)

Call:
lm(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME, data = NY8,
    weights = POP8)

Residuals:
    Min      1Q  Median      3Q     Max
-129.07  -14.71    5.82   25.62   70.72

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.7784     0.1412   -5.51  8.0e-08 ***
PEXPOSURE     0.0763     0.0273    2.79   0.0056 **
PCTAGE65P     3.8566     0.5713    6.75  8.6e-11 ***
PCTOWNHOME   -0.3987     0.1531   -2.60   0.0097 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 33.5 on 277 degrees of freedom
Multiple R-squared: 0.198,  Adjusted R-squared: 0.189
F-statistic: 22.8 on 3 and 277 DF,  p-value: 3.38e-13
```

```
> NY8$lmwresid <- residuals(nylmw)
```

Starting with the weighted linear model, we can see that the TCE exposure variable has become significant with the expected sign, indicating that tracts closer to the TCE sites have slightly higher transformed incidence proportions. The other two covariates now also have more significant coefficients. Figure 10.2 shows that information has been shifted from the model residuals to the model itself, with little remaining spatial structure visible on the map.

```
> lm.morantest(nylmw, NYlistw)

    Global Moran's I for regression residuals

data:
```

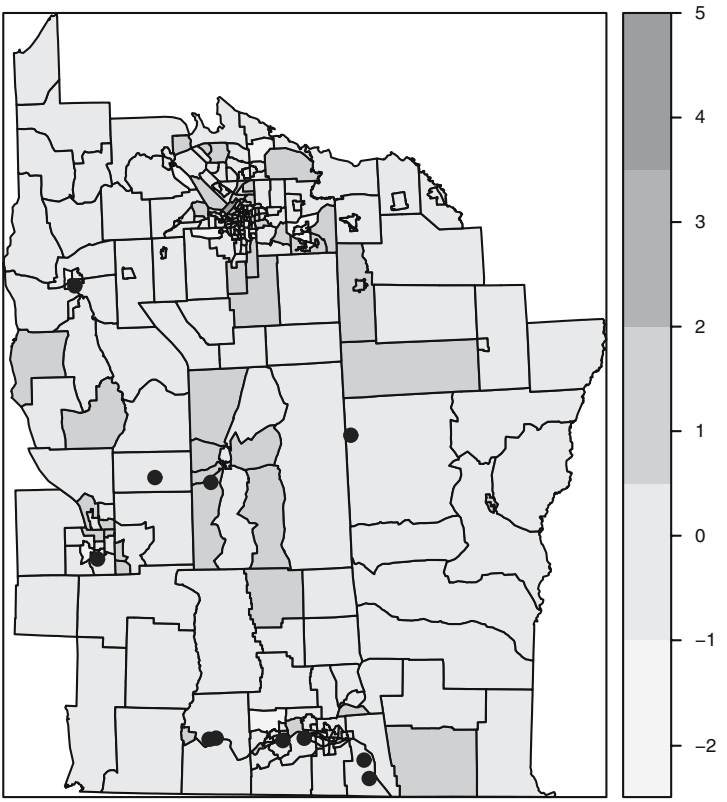

Fig. 10.2. Residuals from the weighted linear model of transformed incidence proportions; TCE site locations shown for comparative purposes

```
model: lm(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
data = NY8, weights = POP8)
weights: NYlistw

Moran I statistic standard deviate = 0.4773, p-value = 0.3166
alternative hypothesis: greater
sample estimates:
Observed Moran's I        Expectation           Variance
          0.007533          -0.009310           0.001245
```

The Moran tests for regression residuals can also be used with a weighted linear model object. The results are interesting, suggesting that the misspecification detected by Moran's I is in fact related to heteroskedasticity more than to spatial autocorrelation. We can check this for the SAR model too, since `spautolm` also takes a `weights` argument:

```
> nysarw <- spautolm(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     data = NY8, listw = NYlistw, weights = POP8)
> summary(nysarw)

Call:
spautolm(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME, data = NY8,
    listw = NYlistw, weights = POP8)

Residuals:
     Min       1Q   Median       3Q      Max
-1.48488 -0.26823  0.09489  0.46552  4.28343

Coefficients:
             Estimate Std. Error z value  Pr(>|z|)
(Intercept) -0.797063   0.144054 -5.5331 3.146e-08
PEXPOSURE    0.080545   0.028334  2.8428  0.004473
PCTAGE65P    3.816731   0.576037  6.6258 3.453e-11
PCTOWNHOME  -0.380778   0.156507 -2.4330  0.014975

Lambda: 0.009564 LR test value: 0.3266 p-value: 0.56764

Log likelihood: -251.6
ML residual variance (sigma squared): 1104, (sigma: 33.23)
Number of observations: 281
Number of parameters estimated: 6
AIC: 515.2
```

The coefficients of the covariates change slightly in the new model, and all the coefficient p-values drop substantially. In this weighted SAR fit, proximity to a TCE site becomes significant. However, there are no traces of spatial autocorrelation left after adjusting for the heterogeneous size of the population. This suggests that the spatial variation in population between tracts is responsible for the observed residual spatial correlation after adjusting for covariates.

To compare both models and choose the best one, we use Akaike's Information Criterion (AIC) reported in the model summaries. The AIC is a weighted sum of the log-likelihood of the model and the number of fitted coefficients; according to the criterion, better models are those with the lower values of the AIC. Hence, the weighted model provides a better fitting since its AIC is considerably lower. This indicates the importance of accounting for heterogeneous populations in the analysis of this type of lattice data.

10.2.2 Conditional Autoregressive Models

The CAR specification relies on the conditional distribution of the spatial error terms. In this case, the distribution of e_i conditioning on e_{-i} (the vector of all random error terms minus e_i itself) is given. Instead of the whole e_{-i} vector, only the neighbours of area i, defined in a chosen way, are used. We represent them by $e_{j\sim i}$. Then, a simple way of putting the conditional distribution of e_i is

$$e_i|e_{j\sim i} \sim N\Big(\sum_{j\sim i} \frac{c_{ij}e_j}{\sum_{j\sim i} c_{ij}}, \frac{\sigma^2_{e_i}}{\sum_{j\sim i} c_{ij}}\Big),$$

where c_{ij} are dependence parameters similar to b_{ij}. However, specifying the conditional distributions of the error terms does not imply that the joint distribution exists. To have a proper distribution some constraints must be set on the parameters of the model. The reader is referred to Schabenberger and Gotway (2005, pp. 338–339) for a detailed description of CAR specifications. For our modelling purposes, the previous guidelines will be enough to obtain a proper CAR specification in most cases.

To fit a CAR model, we can use function `spautolm` again. This time we set the argument `family="CAR"` to specify that we are fitting this type of models.

```
> nycar <- spautolm(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     data = NY8, family = "CAR", listw = NYlistw)
> summary(nycar)

Call:
spautolm(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME, data = NY8,
    listw = NYlistw, family = "CAR")

Residuals:
      Min        1Q    Median        3Q       Max
-1.539732 -0.384311 -0.030646  0.335126  3.808848

Coefficients:
             Estimate Std. Error z value  Pr(>|z|)
(Intercept) -0.648362   0.181129 -3.5796 0.0003442
PEXPOSURE    0.077899   0.043692  1.7829 0.0745986
```

```
PCTAGE65P    3.703830   0.627185  5.9055 3.516e-09
PCTOWNHOME  -0.382789   0.195564 -1.9574 0.0503053

Lambda: 0.08412 LR test value: 5.801 p-value: 0.016018

Log likelihood: -275.8
ML residual variance (sigma squared): 0.4076, (sigma: 0.6384)
Number of observations: 281
Number of parameters estimated: 6
AIC: 563.7
```

The estimated coefficients of the covariates in the model are very similar to those obtained with the SAR models. Nevertheless, the p-values of two covariates, the distance to the nearest TCE and the percentage of people owning a home, are slightly above the 0.05 threshold. The likelihood ratio test indicates that there is significant spatial autocorrelation and the estimated value of λ is 0.0841.

Considering a weighted regression, using the population size as weights, for the same model to account for the heterogeneous distribution of the population completely removes the spatial autocorrelation in the data. The coefficients of the covariates do not change much and all of them become significant. Hence, modelling spatial autocorrelation by means of SAR or CAR specifications does not change the results obtained; Waller and Gotway (2004, pp. 375–379) give a complete discussion of these results.[1]

```
> nycarw <- spautolm(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     data = NY8, family = "CAR", listw = NYlistw, weights = POP8)
> summary(nycarw)

Call:
spautolm(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME, data = NY8,
    listw = NYlistw, weights = POP8, family = "CAR")

Residuals:
      Min        1Q    Median        3Q       Max
-1.491042 -0.270906  0.081435  0.451556  4.198134

Coefficients:
             Estimate Std. Error z value  Pr(>|z|)
(Intercept) -0.790154   0.144862 -5.4545 4.910e-08
PEXPOSURE    0.081922   0.028593  2.8651  0.004169
PCTAGE65P    3.825858   0.577720  6.6223 3.536e-11
PCTOWNHOME  -0.386820   0.157436 -2.4570  0.014010
```

[1] The fitted coefficient values of the weighted CAR model do not exactly reproduce those of Waller and Gotway (2004, p. 379), although the spatial coefficient is reproduced. In addition, the model cannot be fit with S-PLUS™ SpatialStats module `slm`, as the product of the two components of the model covariance matrix is not symmetric, while the two components taken separately are. This suggests that caution in using current implementations of weighted CAR models is justified.

```
Lambda: 0.02242 LR test value: 0.3878 p-value: 0.53343

Log likelihood: -251.6
ML residual variance (sigma squared): 1103, (sigma: 33.21)
Number of observations: 281
Number of parameters estimated: 6
AIC: 515.1
```

10.2.3 Fitting Spatial Regression Models

The `spautolm` function fits spatial regression models by maximum likelihood, by first finding the value of the spatial autoregressive coefficient, which maximises the log likelihood function for the model family chosen, and then fitting the other coefficients by generalised least squares at that point. This means that the spatial autoregressive coefficient can be found by line search using `optimize`, rather than by optimising over all the model parameters at the same time.

The most demanding part of the functions called to optimise the spatial autoregressive coefficient is the computation of the Jacobian, the log determinant of the $n \times n$ matrix $|I - B|$, or $|I - \lambda W|$ in our parametrisation. As n increases, the use of the short-cut of

$$\log(|I - \lambda W|) = \log\Big(\prod_{i=1}^{n}(1 - \lambda\zeta_i)\Big),$$

where ζ_i are the eigenvalues of W, becomes more difficult. The default method of `method="full"` uses eigenvalues, and can thus also set the lower and upper bounds for the line search for λ accurately (as $[1/\min_i(\zeta_i), 1/\max_i(\zeta_i)]$), but is not feasible for large n. It should also be noted that although eigenvalues are computed for intrinsically asymmetric spatial weights matrices, their imaginary parts are discarded, so that even for `method="full"`, the consequences of using such asymmetric weights matrices are unknown.

Alternative approaches involve finding the log determinant of a Cholesky decomposition of the sparse matrix $(I - \lambda W)$ directly. Here it is not possible to pre-compute eigenvalues, so one log determinant is computed for each value of λ used, but the number needed is in general not excessive, and much larger n become feasible on ordinary computers. A number of different sparse matrix approaches have been tried, with the use of **Matrix** and `method="Matrix"`, the one suggested currently. All of the sparse matrix approaches to computing the Jacobian require that matrix W be symmetric or at least similar to symmetric, thus providing for weights with `"W"` and `"S"` styles based on symmetric neighbour lists and symmetric general spatial weights, such as inverse distance. Matrices that are similar to symmetric have the same eigenvalues, so that the eigenvalues of symmetric $W^* = D^{1/2}WD^{1/2}$ and row-standardised

$W = DB$ are the same, for symmetric binary or general weights matrix B, and D a diagonal matrix of inverse row sums of B, $d_{ii} = 1/\sum_{j=1}^{n} b_{ij}$ (Ord, 1975, p. 125).

```
> nysarwM <- spautolm(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     data = NY8, family = "SAR", listw = NYlistw, weights = POP8,
+     method = "Matrix")

> summary(nysarwM)

Call:
spautolm(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME, data = NY8,
    listw = NYlistw, weights = POP8, family = "SAR", method = "Matrix")

Residuals:
     Min       1Q   Median       3Q      Max
-1.48488 -0.26823  0.09489  0.46552  4.28343

Coefficients:
             Estimate Std. Error z value  Pr(>|z|)
(Intercept) -0.797063   0.144054 -5.5331 3.146e-08
PEXPOSURE    0.080545   0.028334  2.8428  0.004473
PCTAGE65P    3.816731   0.576037  6.6258 3.453e-11
PCTOWNHOME  -0.380778   0.156507 -2.4330  0.014975

Lambda: 0.009564 LR test value: 0.3266 p-value: 0.56764

Log likelihood: -251.6
ML residual variance (sigma squared): 1104, (sigma: 33.23)
Number of observations: 281
Number of parameters estimated: 6
AIC: 515.2
```

The output from fitting the weighted SAR model using functions from the **Matrix** package is identical with that from using the eigenvalues of W. Thanks to help from the **Matrix** package authors, Douglas Bates and Martin Mächler; additional facilities have been made available allowing the Cholesky decomposition to be computed once and updated for new values of the spatial coefficient. An internal vectorised version of this update method has also been made available, making the look-up time for many coefficient values small.

If it is of interest to examine values of the log likelihood function for a range of values of λ, the `llprof` argument may be used to give the number of equally spaced λ values to be chosen between the inverse of the smallest and largest eigenvalues for `method="full"`, or a sequence of such values more generally.

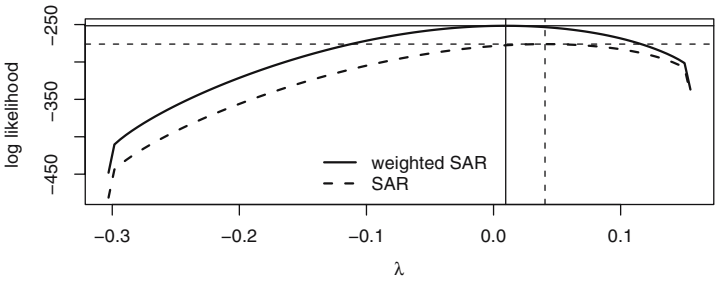

Fig. 10.3. Log likelihood values for a range of values of λ, weighted and unweighted SAR models; fitted spatial coefficient values and maxima shown

```
> 1/range(eigenw(NYlistw))
[1] -0.3029  0.1550
> nysar_ll <- spautolm(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     data = NY8, family = "SAR", listw = NYlistw, llprof = 100)
> nysarw_ll <- spautolm(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     data = NY8, family = "SAR", listw = NYlistw, weights = POP8,
+     llprof = 100)
```

Figure 10.3 shows the shape of the values of the log likelihood function along the feasible range of λ for the weighted and unweighted SAR models. We can see easily that the curves are very flat at the maxima, meaning that we could shift λ a good deal without impacting the function value much. The figure also shows the sharp fall-off in function values as the large negative values of the Jacobian kick in close to the ends of the feasible range.

Finally, `family="SMA"` for simultaneous moving average models is also available within the same general framework, but always involves handling dense matrices for fitting.

```
> nysmaw <- spautolm(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     data = NY8, family = "SMA", listw = NYlistw, weights = POP8)
> summary(nysmaw)

Call:
spautolm(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME, data = NY8,
    listw = NYlistw, weights = POP8, family = "SMA")

Residuals:
      Min        1Q    Median        3Q       Max
-1.487080 -0.268990  0.093956  0.466055  4.284087

Coefficients:
             Estimate Std. Error z value  Pr(>|z|)
(Intercept) -0.795243   0.143749 -5.5321 3.163e-08
PEXPOSURE    0.080153   0.028237  2.8386  0.004531
PCTAGE65P    3.820316   0.575463  6.6387 3.165e-11
PCTOWNHOME  -0.382529   0.156160 -2.4496  0.014302
```

```
Lambda: 0.009184 LR test value: 0.3077 p-value: 0.57909

Log likelihood: -251.6
ML residual variance (sigma squared): 1105, (sigma: 33.24)
Number of observations: 281
Number of parameters estimated: 6
AIC: 515.2
```

Although there may be computing environments within which it seems easier to fit spatial regression models, arguably few give the analyst both reasonable defaults and the opportunity to examine in as much detail as is needed in the internal workings of the methods used, and of their implementations in software. Naturally, improvements will need to be made, perhaps including the fitting of more than one spatial autocorrelation parameter.

10.3 Mixed-Effects Models

The errors e_i which appear in the previous models are used to account for between-area variation, following a specified correlation structure. These terms are usually known as *random effects* because, contrary to what happens with fixed effects (the covariates), the value of the random effect can change from area to area. The range of application of random effects is quite wide, and they are often used to model different types of interaction between the observations. Although mixed-effects models belong to a different tradition from the spatial models discussed above, they are central to multi-level models and small area estimation, both of which can also be used in the analysis of spatial data. In the spatial context, Schabenberger and Gotway (2005, pp. 325–334) discuss linear mixed-effects models; other coverage is to be found in Pinheiro and Bates (2000, pp. 230–232, 237–238) for the implementation used here.

Using a similar notation as in previous sections, mixed-effect models (McCulloch and Searle, 2001) can be formulated as

$$Y = X\beta + Ze + \varepsilon.$$

Vector e represents the random effects, whilst Z is used to account for their structure. The distribution of e is assumed to be Normal with mean zero and generic covariance matrix Σ_e. This structure can reflect the influence of several elements of e on a single observation. Z is a design matrix that may be fixed or depend on any parameter. For example, Z can be set to a specific value to reproduce a SAR or CAR specification but, in this case, Z also depends on λ, which is another parameter to be estimated. Similar models may also be specified for areal data with point support using functions in the **spBayes** package.

Maximum Likelihood or Restricted Maximum Likelihood (McCulloch and Searle, 2001) are often employed to fit mixed-effects models. Packages `nlme` and `lme4` (Pinheiro and Bates, 2000) can fit these types of models. These

packages allow the specification of different types of covariance matrices of the random effects, including spatial structure.

The following example illustrates how to fit a mixed-effects model using a correlation matrix, which depends on the distance between the centroids of the areas. First, we need to specify the correlation structure between the areas. This correlation structure is similar to those used in geostatistics and we have chosen a Gaussian variogram based on the Euclidean distances between the centroids of the regions.

```
> library(nlme)
> NY8$x <- coordinates(NY8)[, 1]/1000
> NY8$y <- coordinates(NY8)[, 2]/1000
> sp1 <- corSpatial(1, form = ~x + y, type = "gaussian")
> scor <- Initialize(sp1, as(NY8, "data.frame")[, c("x",
+     "y")], nugget = FALSE)
```

Once we have specified the correlation structure using `corSpatial`, we need to set up the model. The fixed part of the model is as in the previous SAR and CAR models. In the random part of the model we need to include a random effect per area. This is done by including `random=` $\sim$ `1|AREAKEY` in the call to `lme`. The fitting functions require that the `Spatial*DataFrame` object be coerced to a `data.frame` object in this case.

```
> spmodel <- lme(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     random = ~1 | AREAKEY, data = as(NY8, "data.frame"),
+     correlation = scor, method = "ML")

> summary(spmodel)

Linear mixed-effects model fit by maximum likelihood
 Data: as(NY8, "data.frame")
    AIC   BIC logLik
  571.5 596.9 -278.7

Random effects:
 Formula: ~1 | AREAKEY
        (Intercept) Residual
StdDev:      0.6508  0.04671

Correlation Structure: Gaussian spatial correlation
 Formula: ~x + y | AREAKEY
 Parameter estimate(s):
  range
0.01929
Fixed effects: Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME
             Value Std.Error  DF t-value p-value
(Intercept) -0.517    0.1586 277  -3.262  0.0012
PEXPOSURE    0.049    0.0351 277   1.393  0.1648
PCTAGE65P    3.951    0.6055 277   6.525  0.0000
```

```
PCTOWNHOME  -0.560    0.1703 277  -3.288  0.0011
 Correlation:
           (Intr) PEXPOS PCTAGE
PEXPOSURE  -0.411
PCTAGE65P  -0.587 -0.075
PCTOWNHOME -0.741  0.082  0.147

Standardized Within-Group Residuals:
      Min        Q1       Med        Q3       Max
-0.191116 -0.043422 -0.003575  0.036788  0.454251

Number of Observations: 281
Number of Groups: 281
```

In this case for the un-weighted model, the coefficients of the fixed part are the same as for the linear model. The random effects can be arranged so that they follow a SAR or CAR specification, and it can be seen as a particular structure for Z. Note that when a SAR specification is added, Z, which models the structure of the random effects, may depend on further parameters.

10.4 Spatial Econometrics Approaches

One of the attractions of spatial data analysis is the wide range of scientific disciplines involved. Naturally, this leads to multiple approaches to many kinds of analysis, including accepted ways of applying tests and model fitting methods. It also leads to some sub-communities choosing their own sets of tools, not infrequently diverging from other sub-communities. During the 2003 Distributed Computational Statistics meeting, surprise and amusement was caused by the remark that the Internet domain `www.spatial-statistics.com` contains material chiefly relating to real estate research. But this connection is in fact quite reasonable, as real estate generates a lot of spatial data, and requires suitable methods. Indeed, good understanding of real estate markets and financing is arguably as important to society as a good understanding of the spatial dimensions of disease incidence.

Spatial econometrics is authoritatively described by Anselin (1988, 2002), with additional comments by Bivand (2002, 2006) with regard to doing spatial econometrics in R. While the use of weights, as we have seen above, has resolved a serious model mis-specification in public health data, it would be more typical for econometricians to test first for heteroskedasticity, and to try to relieve it by adjusting coefficient standard errors:

```
> library(lmtest)
> bptest(nylm)

    studentized Breusch-Pagan test

data:  nylm
BP = 9.214, df = 3, p-value = 0.02658
```

The Breusch–Pagan test (Johnston and DiNardo, 1997, pp. 198–200) results indicate the presence of heteroskedasticity when the residuals from the original linear model are regressed on the right-hand-side variables – the default test set. This might suggest the need to adjust the estimated coefficient standard errors using a variance–covariance matrix (Zeileis, 2004) taking heteroskedasticity into account:

```
> library(sandwich)
> coeftest(nylm)

t test of coefficients:

            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.5173     0.1586   -3.26   0.0012 **
PEXPOSURE     0.0488     0.0351    1.39   0.1648
PCTAGE65P     3.9509     0.6055    6.53  3.2e-10 ***
PCTOWNHOME   -0.5600     0.1703   -3.29   0.0011 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> coeftest(nylm, vcov = vcovHC(nylm, type = "HC4"))

t test of coefficients:

            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.5173     0.1617   -3.20  0.00154 **
PEXPOSURE     0.0488     0.0343    1.42  0.15622
PCTAGE65P     3.9509     0.9992    3.95  9.8e-05 ***
PCTOWNHOME   -0.5600     0.1672   -3.35  0.00092 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

There are only minor changes in the standard errors, and they do not affect our inferences.[2]

In spatial econometrics, Moran's I is supplemented by Lagrange Multiplier tests fully described in Anselin (1988, 2002) and Anselin et al. (1996). The development of these tests, as more generally in spatial econometrics, seems to assume the use of row-standardised spatial weights, so we move from symmetric binary weights used above to row-standardised similar to symmetric weights. A key concern is to try to see whether the data generating process is a spatial error SAR or a spatial lag SAR. The former is the SAR that we have already met, while the spatial lag model includes only the endogenous spatially lagged dependent variable in the model.

```
> NYlistwW <- nb2listw(NY_nb, style = "W")
> res <- lm.LMtests(nylm, listw = NYlistwW, test = "all")
> tres <- t(sapply(res, function(x) c(x$statistic, x$parameter,
```

[2] Full details of the test procedures can be found in the references to the function documentation in **lmtest** and **sandwich**.

```
+     x$p.value)))
> colnames(tres) <- c("Statistic", "df", "p-value")
> printCoefmat(tres)

       Statistic    df p-value
LMerr       5.17  1.00    0.02
LMlag       8.54  1.00  0.0035
RLMerr      1.68  1.00    0.20
RLMlag      5.05  1.00    0.02
SARMA      10.22  2.00    0.01
```

The robust LM tests take into account the alternative possibility, that is the `LMerr` test will respond to both an omitted spatially lagged dependent variable and spatially autocorrelated residuals, while the robust `RLMerr` is designed to test for spatially autocorrelated residuals in the possible presence of an omitted spatially lagged dependent variable. The `lm.LMtests` function here returns a list of five LM tests, which seem to point to a spatial lag specification. Further variants have been developed to take into account both spatial autocorrelation and heteroskedasticity, but are not yet available in R. Again, it is the case that if the fitted model exhibits multicollinearity, the results of the tests will be affected.

The spatial lag model takes the following form:

$$\mathbf{y} = \rho\mathbf{W}\mathbf{y} + \mathbf{X}\beta + \mathbf{e},$$

where $\mathbf{y}$ is the endogenous variable, $\mathbf{X}$ is a matrix of exogenous variables, and $\mathbf{W}$ is the spatial weights matrix. This contrasts with the spatial Durbin model, including the spatial lags of the covariates (independent variables) with coefficients γ:

$$\mathbf{y} = \rho\mathbf{W}\mathbf{y} + \mathbf{X}\beta + \mathbf{W}\mathbf{X}\gamma + \mathbf{e},$$

and the spatial error model:

$$\mathbf{y} - \lambda\mathbf{W}\mathbf{y} = \mathbf{X}\beta - \lambda\mathbf{W}\mathbf{X}\beta + \mathbf{e},$$

$$(\mathbf{I} - \lambda\mathbf{W})\mathbf{y} = (\mathbf{I} - \lambda\mathbf{W})\mathbf{X}\beta + \mathbf{e},$$

which can also be written as

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{u},$$

$$\mathbf{u} = \lambda\mathbf{W}\mathbf{u} + \mathbf{e}.$$

First let us fit a spatial lag model by maximum likelihood, once again finding the spatial lag coefficient by line search, then the remaining coefficients by generalised least squares:

```
> nylag <- lagsarlm(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     data = NY8, listw = NYlistwW)
> summary(nylag)

Call:
lagsarlm(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME, data = NY8,
    listw = NYlistwW)

Residuals:
      Min        1Q    Median        3Q       Max
-1.626029 -0.393321 -0.018767  0.326616  4.058315

Type: lag
Coefficients: (asymptotic standard errors)
             Estimate Std. Error z value  Pr(>|z|)
(Intercept) -0.505343   0.155850 -3.2425  0.001185
PEXPOSURE    0.045543   0.034433  1.3227  0.185943
PCTAGE65P    3.650055   0.599219  6.0914 1.120e-09
PCTOWNHOME  -0.411829   0.169095 -2.4355  0.014872

Rho: 0.2252 LR test value: 7.75 p-value: 0.0053703
Asymptotic standard error: 0.07954 z-value: 2.831 p-value: 0.0046378
Wald statistic: 8.015 p-value: 0.0046378

Log likelihood: -274.9 for lag model
ML residual variance (sigma squared): 0.41, (sigma: 0.6403)
Number of observations: 281
Number of parameters estimated: 6
AIC: 561.7, (AIC for lm: 567.5)
LM test for residual autocorrelation
test value: 0.6627 p-value: 0.41561

> bptest.sarlm(nylag)

    studentized Breusch-Pagan test

data:
BP = 7.701, df = 3, p-value = 0.05261
```

The spatial econometrics model fitting functions can also use sparse matrix techniques, but when the eigenvalue technique is used, asymptotic standard errors are calculated for the spatial coefficient. There is a numerical snag here, that if the variables in the model are scaled such that the other coefficients are scaled differently from the spatial autocorrelation coefficient, the inversion of the coefficient variance–covariance matrix may fail. The correct resolution is to re-scale the variables, but the tolerance of the inversion function called internally may be relaxed. In addition, an LM test on the residuals is carried out, suggesting that no spatial autocorrelation remains, and a spatial Breusch–Pagan test shows a lessening of heteroskedasticity.

Fitting a spatial Durbin model, a spatial lag model including the spatially lagged explanatory variables (but not the lagged intercept when the spatial weights are row standardised), we see that the fit is not improved significantly.

```
> nymix <- lagsarlm(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     data = NY8, listw = NYlistwW, type = "mixed")
> nymix

Call:
lagsarlm(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME, data = NY8,
    listw = NYlistwW, type = "mixed")
Type: mixed

Coefficients:
          rho    (Intercept)      PEXPOSURE      PCTAGE65P
      0.17578       -0.32260        0.09039        3.61356
   PCTOWNHOME  lag.PEXPOSURE  lag.PCTAGE65P lag.PCTOWNHOME
     -0.02687       -0.05188        0.13123       -0.69950

Log likelihood: -272.7

> anova(nymix, nylag)

      Model   df  AIC logLik Test L.Ratio p-value
nymix     1    9  563   -273    1
nylag     2    6  562   -275    2       4    0.22
```

If we impose the Common Factor constraint on the spatial Durbin model, that $\gamma = -\lambda\beta$, we fit the spatial error model:

```
> nyerr <- errorsarlm(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     data = NY8, listw = NYlistwW)
> summary(nyerr)

Call:errorsarlm(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
    data = NY8, listw = NYlistwW)

Residuals:
      Min        1Q    Median        3Q       Max
-1.628589 -0.384745 -0.030234  0.324747  4.047906

Type: error
Coefficients: (asymptotic standard errors)
            Estimate Std. Error z value  Pr(>|z|)
(Intercept) -0.58662    0.17471 -3.3577  0.000786
PEXPOSURE    0.05933    0.04226  1.4039  0.160335
PCTAGE65P    3.83746    0.62345  6.1552 7.496e-10
PCTOWNHOME  -0.44428    0.18897 -2.3510  0.018721

Lambda: 0.2169 LR test value: 5.425 p-value: 0.019853
```

```
Asymptotic standard error: 0.08504 z-value: 2.551 p-value: 0.010749
Wald statistic: 6.506 p-value: 0.010749

Log likelihood: -276 for error model
ML residual variance (sigma squared): 0.4137, (sigma: 0.6432)
Number of observations: 281
Number of parameters estimated: 6
AIC: 564, (AIC for lm: 567.5)
```

Both the spatial lag and Durbin models appear to fit the data somewhat better than the spatial error model. However, in relation to our initial interest in the relationship between transformed incidence proportions and exposure to TCE sites, we are no further forward than we were with the linear model, and although we seem to have reduced the mis-specification found in the linear model by choosing the spatial lag model, the reduction in error variance is only moderate.

Spatial econometrics has also seen the development of alternatives to maximum likelihood methods for fitting models. Code for two of these has been contributed by Luc Anselin, and is available in **spdep**. For example, the spatial lag model may be fitted by analogy with two-stage least squares in a simultaneous system of equations, by using the spatial lags of the explanatory variables as instruments for the spatially lagged dependent variable.

```
> nystsls <- stsls(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     data = NY8, listw = NYlistwW)
> summary(nystsls)

Call:
stsls(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME, data = NY8,
    listw = NYlistwW)

Residuals:
      Min        1Q    Median        3Q       Max
-1.593609 -0.368930 -0.029486  0.335873  3.991544

Coefficients:
             Estimate Std. Error t value  Pr(>|t|)
Rho          0.409651   0.171972  2.3821  0.017215
(Intercept) -0.495567   0.155743 -3.1820  0.001463
PEXPOSURE    0.042846   0.034474  1.2428  0.213924
PCTAGE65P    3.403617   0.636631  5.3463 8.977e-08
PCTOWNHOME  -0.290416   0.201743 -1.4395  0.149998

Residual variance (sigma squared): 0.4152, (sigma: 0.6444)
```

The implementation acknowledges that the estimate of the spatial coefficient will be biased, but because it can be used with very large data sets and does provide an alternative, it is worth mentioning. It is interesting that when the `robust` argument is chosen, adjusting not only standard errors but also

coefficient values for heteroskedasticity over and above the spatial autocorrelation already taken into account, we see that the coefficient operationalising TCE exposure moves towards significance:

```
> nystslsR <- stsls(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     data = NY8, listw = NYlistwW, robust = TRUE)
> summary(nystslsR)

Call:
stsls(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME, data = NY8,
    listw = NYlistwW, robust = TRUE)

Residuals:
      Min        1Q    Median        3Q       Max
-1.559044 -0.361838 -0.016518  0.353569  4.092810

Coefficients:
             Estimate Robust std. Error z value Pr(>|z|)
Rho          0.411452          0.184989  2.2242 0.026135
(Intercept) -0.499489          0.156801 -3.1855 0.001445
PEXPOSURE    0.056973          0.029993  1.8995 0.057494
PCTAGE65P    3.030160          0.955171  3.1724 0.001512
PCTOWNHOME  -0.267249          0.203269 -1.3148 0.188591

Asymptotic robust residual variance: 0.409, (sigma: 0.6395)
```

Finally, `GMerrorsar` is an implementation of the Kelejian and Prucha (1999) Generalised Moments (GM) estimator for the autoregressive parameter in a spatial model. It uses a GM approach to optimise λ and σ^2 jointly, and where the numerical search surface is not too flat, can be an alternative to maximum likelihood methods when n is large.

```
> nyGMerr <- GMerrorsar(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     data = NY8, listw = NYlistwW)
> summary(nyGMerr)

Call:GMerrorsar(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
    data = NY8, listw = NYlistwW)

Residuals:
      Min        1Q    Median        3Q       Max
-1.640399 -0.384014 -0.031843  0.318732  4.057979

Type: GM SAR estimator
Coefficients: (GM standard errors)
             Estimate Std. Error z value  Pr(>|z|)
(Intercept) -0.577906   0.172566 -3.3489 0.0008114
PEXPOSURE    0.057984   0.041303  1.4039 0.1603604
PCTAGE65P    3.848771   0.621105  6.1967 5.768e-10
PCTOWNHOME  -0.458145   0.186666 -2.4544 0.0141138
```

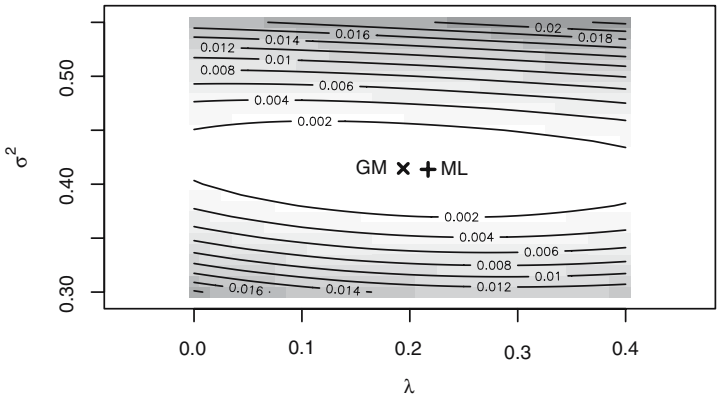

Fig. 10.4. Numerical search surface of the generalised moments estimator with GM and ML optima marked

```
Lambda: 0.1939 LR test value: 5.361 p-value: 0.020594

Log likelihood: -276.0 for GM model
ML residual variance (sigma squared): 0.4146, (sigma: 0.6439)
Number of observations: 281
Number of parameters estimated: 6
AIC: 564.1, (AIC for lm: 567.5)
```

Figure 10.4 shows, however, that there is much more variability in the surface on the σ^2 axis than on the λ axis, and so the optimiser may stop its search when the default joint criterion for termination is satisfied, rather than searching harder along λ. Of course, non-default settings may be passed to the optimiser to tune its performance, but this too requires care and insight.

10.5 Other Methods

Other methods can be used to model dependency between areas. In this section we introduce some of them, based in part on the recent applied survey reported by Dormann et al. (2007). A specific difficulty that we met above when considering mixed-effects models is that available functions for model fitting use point support rather than polygon support. This means that our prior description of the relationships between observations are distance-based, and so very similar to those described in detail in Chap. 8, where the focus was more on interpolation than modelling. These methods are discussed in the spatial context by Schabenberger and Gotway (2005, pp. 352–382) and Waller and Gotway (2004, pp. 380–409), and hierarchical methods are being employed with increasing frequency (Banerjee et al., 2004).

10.5.1 GAM, GEE, GLMM

Generalised Additive Models (GAM) are very similar to generalised linear models, but they also allow for including non-linear terms in the linear predictor term (Hastie and Tibshirani, 1990; Wood, 2006). It is worth noting that the `formula` argument to linear, generalised linear, spatial, and many other models may contain polynomial and spline terms if desired, but these need to be configured manually. Different types of non-linear functions are available, and may be chosen in the `s()` function in the formula. Here, an isotropic thin plate regression spline is used effectively as a semi-parametric trend surface to add smooth spatial structure from the residuals to the fit, as in Chap. 7 (p. 180).

```
> library(mgcv)

This is mgcv 1.3-29

> nyGAM1 <- gam(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME +
+     s(x, y), weights = POP8, data = NY8)
> anova(nylmw, nyGAM1, test = "Chisq")

Analysis of Variance Table

Model 1: Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME
Model 2: Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME + s(x, y)
  Res.Df    RSS     Df Sum of Sq P(>|Chi|)
1 277.00 310778
2 273.19 305229   3.81      5550      0.27
```

This does not add much to what we already knew from the weighted linear model, with the differences in the residual degrees of freedom showing that the thin plate regression spline term only takes 3.810 estimated degrees of freedom. This does not, however, exploit the real strengths of the technique. Because it can fit generalised models, we can step back from using the transformed incidence proportions to use the case counts (admittedly not integer because of the sharing-out of cases with unknown tract within blocks), offset by the logarithm of tract populations. Recall that we have said that distributional assumptions about the response variable matter – our response variable perhaps ought to be treated as discrete, so methods respecting this may be more appropriate.

Using the Poisson Generalised Linear Model (GLM) fitting approach, we fit first with `glm`; the Poisson model is introduced in Chap. 11. We can already see that this GLM approach yields interesting insights and that the effects of TCE exposure on the numbers of cases are significant.

```
> nyGLMp <- glm(Cases ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME +
+     offset(log(POP8)), data = NY8, family = "poisson")

> summary(nyGLMp)
```

```
Call:
glm(formula = Cases ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME +
    offset(log(POP8)), family = "poisson", data = NY8)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-2.678  -1.057  -0.198   0.633   3.266

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -8.1344     0.1826  -44.54  < 2e-16 ***
PEXPOSURE     0.1489     0.0312    4.77  1.8e-06 ***
PCTAGE65P     3.9982     0.5978    6.69  2.3e-11 ***
PCTOWNHOME   -0.3571     0.1903   -1.88     0.06 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 428.25  on 280  degrees of freedom
Residual deviance: 353.35  on 277  degrees of freedom
AIC: Inf

Number of Fisher Scoring iterations: 5
```

The use of the Moran's I test for regression residuals is speculative and provisional. Based on Lin and Zhang (2007), it takes the deviance residuals and the linear part of the GLM and provides an indication that the Poisson regression, like the weighted linear regression, does not have strong residual spatial autocorrelation (see Fig. 10.5). Much more work remains to be done, perhaps based on Jacqmin-Gadda et al. (1997), to reach a satisfactory spatial autocorrelation test for the residuals of GLM.

```
> NY8$lmpresid <- residuals(nyGLMp, type = "deviance")
> lm.morantest(nyGLMp, listw = NYlistwW)

    Global Moran's I for regression residuals

data:
model: glm(formula = Cases ~ PEXPOSURE + PCTAGE65P +
PCTOWNHOME + offset(log(POP8)), family = "poisson", data =
NY8)
weights: NYlistwW

Moran I statistic standard deviate = 0.7681, p-value = 0.2212
alternative hypothesis: greater
sample estimates:
Observed Moran's I        Expectation           Variance
          0.024654          -0.004487           0.001439
```

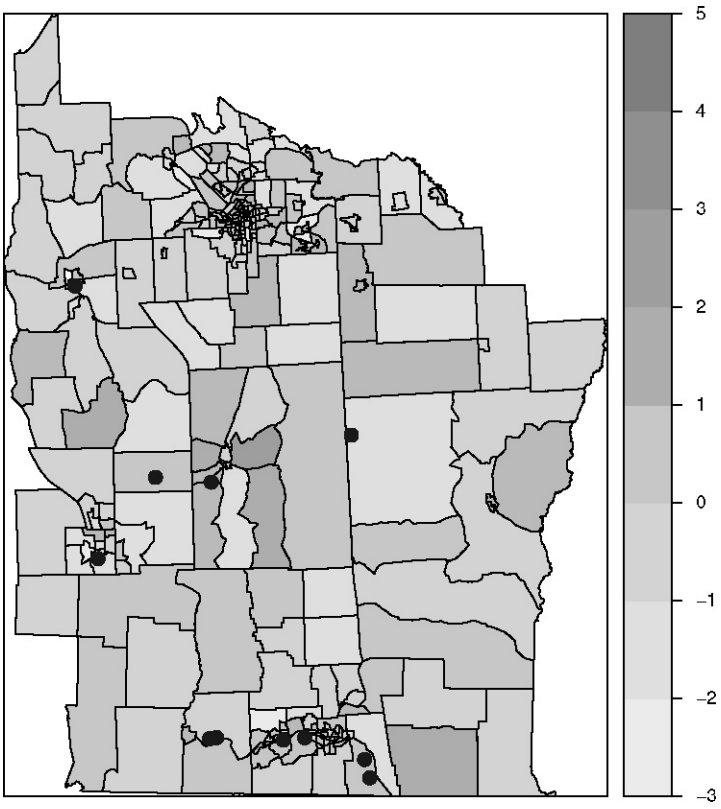

Fig. 10.5. Residuals from the Poisson regression model; TCE site locations shown for comparative purposes

With the GLM to start from, we again add an isotropic thin plate regression spline in `gam`. There is little over-dispersion present – fitting with `family=quasipoisson`, in which the dispersion parameter is not fixed at unity, so they can model over-dispersion that does not result in large changes. Model comparison shows that the presence of the spline term is now significant. While the coefficient values of the Poisson family fits are not directly comparable with the linear fits on the transformed incidence proportions, we can see that exposure to TCE sites is clearly more significant.

```
> nyGAMp <- gam(Cases ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME +
+     offset(log(POP8)) + s(x, y), data = NY8, family = "poisson")
> summary(nyGAMp)

Family: poisson
Link function: log
```

```
Formula:
Cases ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME + offset(log(POP8)) +
    s(x, y)

Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -8.1366     0.2070  -39.31  < 2e-16 ***
PEXPOSURE     0.1681     0.0558    3.01   0.0026 **
PCTAGE65P     3.7199     0.6312    5.89  3.8e-09 ***
PCTOWNHOME   -0.3602     0.1951   -1.85   0.0649 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
        edf Est.rank Chi.sq p-value
s(x,y) 7.71       16     24   0.089 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.394   Deviance explained = 21.4%
UBRE score = 0.2815  Scale est. = 1         n = 281
```

```
> anova(nyGLMp, nyGAMp, test = "Chisq")

Analysis of Deviance Table

Model 1: Cases ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME +
    offset(log(POP8))

Model 2: Cases ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME +
    offset(log(POP8)) + s(x, y)

  Resid. Df Resid. Dev     Df Deviance P(>|Chi|)
1    277.00        353
2    269.29        337   7.71       17     0.029
```

Generalised Estimating Equations (GEE) are an alternative to the estimation of GLMs when we have correlated data. They are often used in the analysis of longitudinal data, when we have several observations for the same subject. In a spatial setting, the correlation arises between neighbouring areas. The treatment in Dormann et al. (2007) is promising for the restricted case of clusters of grid cells, but has not yet been extended to irregular point or polygon support.

Generalised linear mixed-effect models (GLMM) extend GLMs by allowing the incorporation of mixed effects into the linear predictor; see Waller and Gotway (2004, pp. 387–392) and Schabenberger and Gotway (2005, pp. 359–369). These random effects can account for correlation between observations. Here we use `glmmPQL` from **MASS**, described in Venables and Ripley (2002, pp. 292–298), and a Gaussian spatial correlation structure as above when applying linear mixed-effect models. The `glmmPQL` function calls `lme` internally,

so we can use the values of the `random` and `correlation` arguments used above on p. 288. Dormann et al. (2007) suggest the use of a single group, because the spatial correlation structure is applied group-wise,[3] but admit that this is an 'abuse' of the procedure.

```
> library(MASS)
> attach(as(NY8, "data.frame"))
> nyGLMMp <- glmmPQL(Cases ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME +
+     offset(log(POP8)), data = NY8, family = poisson,
+     random = ~1 | AREAKEY, correlation = scor)
> detach("as(NY8, \"data.frame\")")

> summary(nyGLMMp)

Linear mixed-effects model fit by maximum likelihood
 Data: NY8
  AIC BIC logLik
   NA  NA     NA

Random effects:
 Formula: ~1 | AREAKEY
        (Intercept) Residual
StdDev:   7.325e-05    1.121

Correlation Structure: Gaussian spatial correlation
 Formula: ~x + y | AREAKEY
 Parameter estimate(s):
    range
0.0005343
Variance function:
 Structure: fixed weights
 Formula: ~invwt

Fixed effects: Cases ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME +
    offset(log(POP8))

             Value Std.Error  DF t-value p-value
(Intercept) -8.134    0.2062 277  -39.45  0.0000
PEXPOSURE    0.149    0.0352 277    4.23  0.0000
PCTAGE65P    3.998    0.6750 277    5.92  0.0000
PCTOWNHOME  -0.357    0.2148 277   -1.66  0.0976
 Correlation:
            (Intr) PEXPOS PCTAGE
PEXPOSURE   -0.472
PCTAGE65P   -0.634  0.030
PCTOWNHOME -0.768  0.134  0.230
```

[3] They report that results from PROC GLIMMIX in SAS can be reproduced using only a single group.

```
Standardized Within-Group Residuals:
    Min      Q1     Med      Q3     Max
-1.7839 -0.7476 -0.1731  0.6003  3.8928

Number of Observations: 281
Number of Groups: 281
```

The fitting functions require that the `Spatial*DataFrame` object be coerced to a `data.frame` object, and `attach` be used to make the variables visible in the global environment in this case. The outcome is very close to the GAM results, and again we find that closeness to the TCE sites is a significant covariate; again, the percentage owning their own homes is not significant. Since it is fitted by penalised quasi-likelihood, no log likelihood value is available, and the summary reports `NA` for AIC, BIC, and log likelihood.

10.5.2 Moran Eigenvectors

In the previous chapter, we touched on the use of eigenvalues in the Saddlepoint approximation and exact tests for Moran's I. The Moran eigenvector approach (Dray et al., 2006; Griffith and Peres-Neto, 2006) involved the spatial patterns represented by maps of eigenvectors; by choosing suitable orthogonal patterns and adding them to a linear or generalised linear model, the spatial dependence present in the residuals can be moved into the model.

It uses brute force to search the set of eigenvectors of the matrix **MWM**, where

$$\mathbf{M} = \mathbf{I} - \mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}$$

is a symmetric and idempotent projection matrix and $\mathbf{W}$ are the spatial weights. In the spatial lag form of `SpatialFiltering` and in the GLM `ME` form below, $\mathbf{X}$ is an n-vector of ones, that is the intercept only.

In its general form, `SpatialFiltering` chooses the subset of the n eigenvectors that reduce the residual spatial autocorrelation in the error of the model with covariates. The lag form adds the covariates in assessment of which eigenvectors to choose, but does not use them in constructing the eigenvectors. `SpatialFiltering` was implemented and contributed by Yongwan Chun and Michael Tiefelsdorf, and is presented in Tiefelsdorf and Griffith (2007); `ME` is based on Matlab code by Pedro Peres-Neto and is discussed in Dray et al. (2006) and Griffith and Peres-Neto (2006).

```
> nySFE <- SpatialFiltering(Z ~ PEXPOSURE + PCTAGE65P +
+     PCTOWNHOME, data = NY8, nb = NY_nb, style = "W",
+     verbose = FALSE)
> nylmSFE <- lm(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME +
+     fitted(nySFE), data = NY8)
> summary(nylmSFE)
```

```
Call:
lm(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME + fitted(nySFE),
    data = NY8)

Residuals:
    Min      1Q  Median      3Q     Max
-1.5184 -0.3523 -0.0105  0.3221  3.1964

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)         -0.5173     0.1461   -3.54  0.00047 ***
PEXPOSURE            0.0488     0.0323    1.51  0.13172
PCTAGE65P            3.9509     0.5578    7.08  1.2e-11 ***
PCTOWNHOME          -0.5600     0.1569   -3.57  0.00042 ***
fitted(nySFE)vec13  -2.0940     0.6053   -3.46  0.00063 ***
fitted(nySFE)vec44  -2.2400     0.6053   -3.70  0.00026 ***
fitted(nySFE)vec6    1.0298     0.6053    1.70  0.09007 .
fitted(nySFE)vec38   1.2928     0.6053    2.14  0.03361 *
fitted(nySFE)vec20   1.1006     0.6053    1.82  0.07015 .
fitted(nySFE)vec14  -1.0511     0.6053   -1.74  0.08366 .
fitted(nySFE)vec75   1.9060     0.6053    3.15  0.00183 **
fitted(nySFE)vec21  -1.0633     0.6053   -1.76  0.08014 .
fitted(nySFE)vec36  -1.1786     0.6053   -1.95  0.05258 .
fitted(nySFE)vec61  -1.0858     0.6053   -1.79  0.07399 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.605 on 267 degrees of freedom
Multiple R-squared: 0.34,   Adjusted R-squared: 0.308
F-statistic: 10.6 on 13 and 267 DF,  p-value: <2e-16
```

```
> anova(nylm, nylmSFE)
```

```
Analysis of Variance Table

Model 1: Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME
Model 2: Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME + fitted(nySFE)
  Res.Df   RSS  Df Sum of Sq    F Pr(>F)
1    277 119.6
2    267  97.8  10      21.8 5.94  4e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Since the `SpatialFiltering` approach does not allow weights to be used, we see that the residual autocorrelation of the original linear model is absorbed, or 'whitened' by the inclusion of selected eigenvectors in the model, but that the covariate coefficients change little. The addition of these eigenvectors – each representing an independent spatial pattern – relieves the residual autocorrelation, but otherwise makes few changes in the substantive coefficient values.

The ME function also searches for eigenvectors from the spatial lag variant of the underlying model, but in a GLM framework. The criterion is a permutation bootstrap test on Moran's I for regression residuals, and in this case, because of the very limited remaining spatial autocorrelation, is set at $\alpha = 0.5$. Even with this very generous stopping rule, only two eigenvectors are chosen; their combined contribution just improves only the fit of the GLM model.

```
> nyME <- ME(Cases ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     data = NY8, offset = log(POP8), family = "poisson",
+     listw = NYlistwW, alpha = 0.5)

> nyME

  Eigenvector ZI pr(ZI)
0          NA NA   0.26
1          24 NA   0.47
2         223 NA   0.52

> nyglmME <- glm(Cases ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME +
+     offset(log(POP8)) + fitted(nyME), data = NY8, family = "poisson")

> summary(nyglmME)

Call:
glm(formula = Cases ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME +
    offset(log(POP8)) + fitted(nyME), family = "poisson", data = NY8)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-2.569  -1.068  -0.212   0.610   3.166

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)         -8.1269     0.1834  -44.30  < 2e-16 ***
PEXPOSURE            0.1423     0.0314    4.53  5.8e-06 ***
PCTAGE65P            4.1105     0.5995    6.86  7.1e-12 ***
PCTOWNHOME          -0.3827     0.1924   -1.99    0.047 *
fitted(nyME)vec24    1.5266     0.7226    2.11    0.035 *
fitted(nyME)vec223   0.8142     0.7001    1.16    0.245
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 428.25  on 280  degrees of freedom
Residual deviance: 347.34  on 275  degrees of freedom
AIC: Inf

Number of Fisher Scoring iterations: 5

> anova(nyGLMp, nyglmME, test = "Chisq")
```

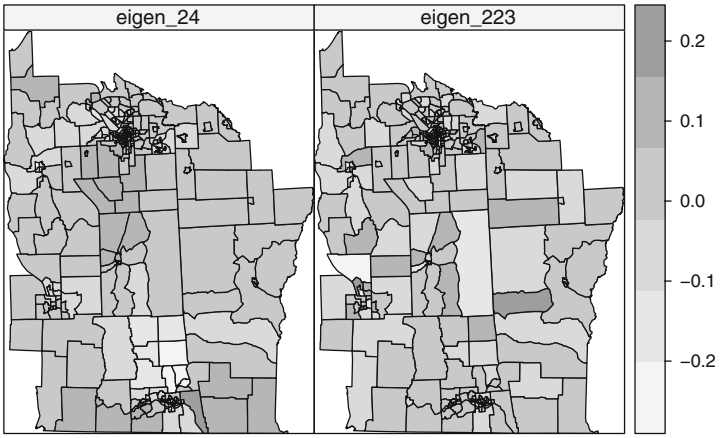

Fig. 10.6. Maps of the two eigenvalues selected for inclusion in the Poisson regression model

```
Analysis of Deviance Table

Model 1: Cases ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME +
    offset(log(POP8))

Model 2: Cases ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME +
    offset(log(POP8)) + fitted(nyME)

  Resid. Df Resid. Dev  Df Deviance P(>|Chi|)
1       277        353
2       275        347   2        6      0.05
```

Figure 10.6 shows the spatial patterns chosen to match the very small amount of spatial autocorrelation remaining in the model. As with the other Poisson regressions, the closeness to TCE sites is highly significant. Since, however, many TCE sites are also in or close to more densely populated urban areas with the possible presence of both point-source and non-point-source pollution, it would be premature to take such results simply at their face value. There is, however, a potentially useful contrast between the cities of Binghampton in the south of the study area with several sites in its vicinity, and Syracuse in the north without TCE sites in this data set.

10.5.3 Geographically Weighted Regression

Geographically weighted regression (GWR) is an exploratory technique mainly intended to indicate where non-stationarity is taking place on the map, that is where locally weighted regression coefficients move away from their global values. Its basis is the concern that the fitted coefficient values of a global model, fitted to all the data, may not represent detailed local variations in the

data adequately – in this it follows other local regression implementations. It differs, however, in not looking for local variation in 'data' space, but by moving a weighted window over the data, estimating one set of coefficient values at every chosen 'fit' point. The fit points are very often the points at which observations were made, but do not have to be. If the local coefficients vary in space, it can be taken as an indication of non-stationarity.

The technique is fully described by Fotheringham et al. (2002) and involves first selecting a bandwidth for an isotropic spatial weights kernel, typically a Gaussian kernel with a fixed bandwidth chosen by leave-one-out cross-validation. Choice of the bandwidth can be very demanding, as n regressions must be fitted at each step. Alternative techniques are available, for example for adaptive bandwidths, but they may often be even more compute-intensive. GWR is discussed by Schabenberger and Gotway (2005, pp. 316–317) and Waller and Gotway (2004, p. 434), and presented with examples by Lloyd (2007, pp. 79–86).

```
> library(spgwr)

> bwG <- gwr.sel(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME,
+     data = NY8, gweight = gwr.Gauss, verbose = FALSE)
> gwrG <- gwr(Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME, data = NY8,
+     bandwidth = bwG, gweight = gwr.Gauss, hatmatrix = TRUE)

> gwrG

Call:
gwr(formula = Z ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME, data = NY8,
    bandwidth = bwG, gweight = gwr.Gauss, hatmatrix = TRUE)
Kernel function: gwr.Gauss
Fixed bandwidth: 179943
Summary of GWR coefficient estimates:
                Min. 1st Qu.  Median 3rd Qu.    Max. Global
X.Intercept. -0.5220 -0.5210 -0.5200 -0.5140 -0.5110  -0.52
PEXPOSURE     0.0472  0.0480  0.0495  0.0497  0.0505   0.05
PCTAGE65P     3.9100  3.9300  3.9600  3.9600  3.9800   3.95
PCTOWNHOME   -0.5590 -0.5580 -0.5580 -0.5550 -0.5550  -0.56
Number of data points: 281
Effective number of parameters: 4.4
Effective degrees of freedom: 276.6
Sigma squared (ML): 0.4255
AICc (GWR p. 61, eq 2.33; p. 96, eq. 4.21): 568
AIC (GWR p. 96, eq. 4.22): 561.6
Residual sum of squares: 119.6
```

Once the bandwidth has been found, or chosen by hand, the `gwr` function may be used to fit the model with the chosen local kernel and bandwidth. If the `data` argument is passed a `SpatialPolygonsDataFrame` or a `SpatialPointsDataFrame` object, the output object will contain a component, which is an object of the same geometry populated with the local coefficient estimates.

If the input objects have polygon support, the centroids of the spatial entities are taken as the basis for analysis. The function also takes a `fit.points` argument, which permits local coefficients to be created by geographically weighted regression for other support than the data points.

The basic GWR results are uninteresting for this data set, with very little local variation in coefficient values; the bandwidth is almost 180 km. Neither `gwr` nor `gwr.sel` yet take a `weights` argument, as it is unclear how non-spatial and geographical weights should be combined. A further issue that has arisen is that it seems that local collinearity can be induced, or at least observed, in GWR applications. A discussion of the issues raised is given by Wheeler and Tiefelsdorf (2005).

As Fotheringham et al. (2002) describe, GWR can also be applied in a GLM framework, and a provisional implementation permitting this has been added to the `spgwr` package providing both cross-validation bandwidth selection and geographically weighted fitting of GLM models.

```
> gbwG <- ggwr.sel(Cases ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME +
+     offset(log(POP8)), data = NY8, family = "poisson",
+     gweight = gwr.Gauss, verbose = FALSE)
> ggwrG <- ggwr(Cases ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME +
+     offset(log(POP8)), data = NY8, family = "poisson",
+     bandwidth = gbwG, gweight = gwr.Gauss)

> ggwrG

Call:

ggwr(formula = Cases ~ PEXPOSURE + PCTAGE65P + PCTOWNHOME +
    offset(log(POP8)), data = NY8, bandwidth = gbwG, gweight =
    gwr.Gauss, family = "poisson")

Kernel function: gwr.Gauss
Fixed bandwidth: 179943
Summary of GWR coefficient estimates:
               Min. 1st Qu. Median 3rd Qu.   Max. Global
X.Intercept. -8.140  -8.140 -8.140  -8.130 -8.130  -8.13
PEXPOSURE     0.147   0.148  0.149   0.149  0.150   0.15
PCTAGE65P     3.980   3.980  3.980   4.010  4.020   4.00
PCTOWNHOME   -0.357  -0.355 -0.355  -0.349 -0.346  -0.36
```

The local coefficient variation seen in this fit is not large either, although from Fig. 10.7 it appears that slightly larger local coefficients for the closeness to TCE site covariate are found farther away from TCE sites than close to them. If, on the other hand, we consider this indication in the light of Fig. 10.8, it is clear that the forcing artefacts found by Wheeler and Tiefelsdorf (2005) in a different data set are replicated here.

Further ways of using R for applying different methods for modelling areal data are presented in Chap. 11. It is important to remember that the availability of implementations of methods does not mean that any of them are

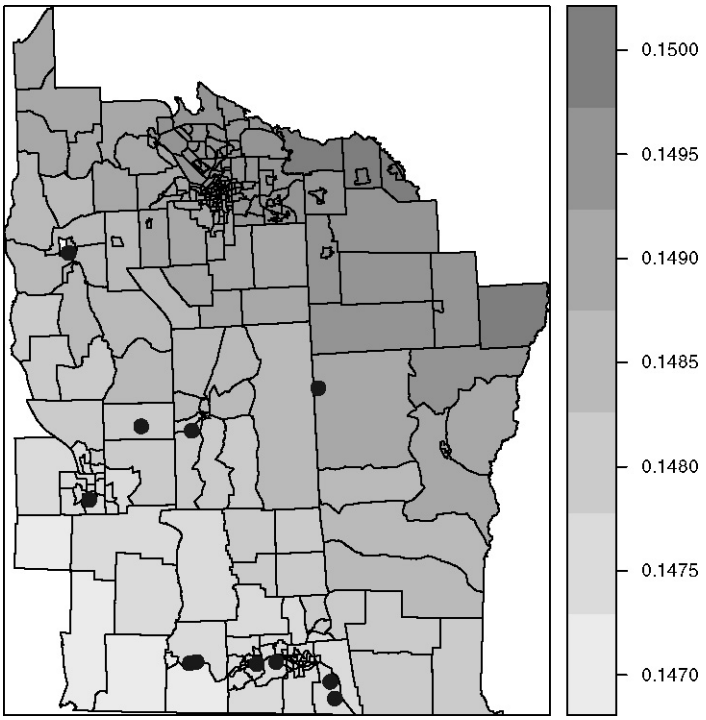

Fig. 10.7. GWR local coefficient estimates for the exposure to TCE site covariate

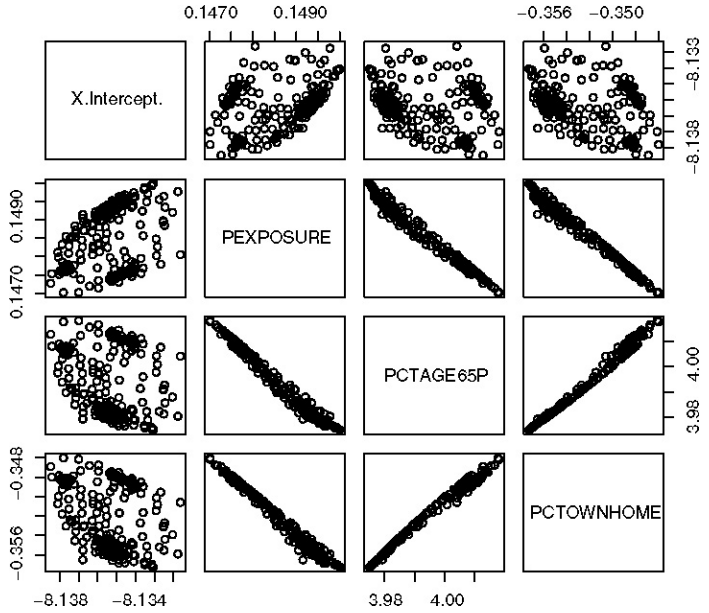

Fig. 10.8. Pairs plots of GWR local coefficient estimates showing the effects of GWR collinearity forcing

'best practice' as such. It is the analyst who has responsibility for choices of methods and implementations in relation to situation-specific requirements and available data. What the availability of a range of methods in R does make possible is that the analyst has choice and has tools for ensuring that the research outcomes are fully reproducible.