Chapter 7
Violation of Independence – Part II

In the previous chapter, we discussed violation of independence for measurements taken repeatedly over time and how temporal correlation structures can be added to linear regression and additive models. We used a regular spaced data set. In this chapter, we consider data measured at multiple spatial locations, and we show how similar correlation structures can be used. The 'Part II' in the title refers to irregular spaced data, either in space, time, or along an age or depth gradient. The general principle with spatial data is that things that are close to each other are likely to be more similar than things that are further apart (Tobler, 1979).

In this chapter, we use various examples. The first example uses data obtained at multiple spatial locations. We then revisit the Hawaiian bird data and show how to add spatial correlation to time series models. We also present an example where a correlation structure along an age gradient is required. In Section 7.5, we discuss the possibility that spatial correlation may be due to missing covariates. In the final section, we analyse data from a bird behavioural study, but this time with short longitudinal (temporal) measurements.

7.1 Tools to Detect Violation of Independence

In this section, we use data from Chapter 37 in Zuur et al. (2007). The case study in that chapter illustrates the application of spatial analysis methods on a boreal forest in Tatarstan, Russia. Using remotely sensed data and spatial statistical methods, they explored the influence of relief, soil, and climatic factors on the forests of the Raifa section of Volzhsko-Kamsky State Nature Biosphere. The response variable is a boreal forest index and is defined as the number of species that belong to a set of boreal species divided by the total number of species at a site. Several remotely sensed variables derived from the LANDSAT 5 satellite images were used as explanatory variables: (i) the normalised difference vegetation index, (ii) temperature, (iii) an index of wetness, and (iv) an index of greenness. A data exploration indicated high collinearity between these variables, and we therefore only used wetness. In addition to these variables, we also know the latitude (X) and longitude (Y) of each site. Boreality was transformed using the following transformation:

A.F. Zuur et al., *Mixed Effects Models and Extensions in Ecology with R*,
Statistics for Biology and Health, DOI 10.1007/978-0-387-87458-6_7,

$$z_i = (1000 \times (S_i + 1)/n_i)^{1/2}$$

where z_i is transformed boreality, S_i is the number of species that belong to boreal coenosis species, and n_i is the number of all species at the site i. See Cressie (p. 395, 1993) for a discussion of this transformation.

In Chapter 6, we started by applying a model without a temporal correlation structure and used graphical tools to asses violation of independence. As a second step, we made an auto-correlation (ACF) of the residuals, and finally we added a temporal correlation structure to the regression and GAM models. The same can be done for spatial data. We first fit the following linear regression model.

$$z_i = \alpha + \beta \times \text{Wetness}_i + \varepsilon_i$$

where z_i is the transformed boreality, Wetness_i is the wetness at site i, and the index $i = 1, \ldots, 533$. The following R code imports the data and applies the transformation and linear regression.

```
> library(AED); data(Boreality)
> Boreality$Bor <- sqrt(1000 * (Boreality$nBor + 1) /
                                (Boreality$nTot))
> B.lm <- lm(Bor ~ Wet, data = Boreality)
> summary(B.lm)
```

The results from the `summary` command are not given here, but the explanatory variable Wetness is highly significant ($t = 15.64$, $df = 532$, $p < 0.001$). Based on residual graphs (not shown here), homogeneity is a reasonable assumption. As a first step to verify independence, we plot the residuals versus their spatial coordinates. The package `gstat` (Pebesma, 2004) has a nice tool for this called a bubble plot, see Fig. 7.1. This package is not part of the base installation and you will need to install it from the R website. The size of the dots is proportional to the value of the residuals. This graph should not show any spatial pattern (e.g. groups of negative or positive residuals close to each other). If it does, then their may be a missing covariate or spatial correlation. In this case, there seems to be some spatial pattern as most of the positive residuals as well as the negative residuals are showing some clustering. The following R code was used to create the graph:

```
> E <- rstandard(B.lm)
> library(gstat)
> mydata <- data.frame(E, Boreality$x, Boreality$y)
> coordinates(mydata) <- c("Boreality.x","Boreality.y")
> bubble(mydata, "E", col = c("black","grey"),
       main = "Residuals", xlab = "X-coordinates",
       ylab = "Y-coordinates")
```

The first part of the R code extracts the standardised residuals, loads the `gstat` package, and creates a data frame containing the residuals and the coordinates.

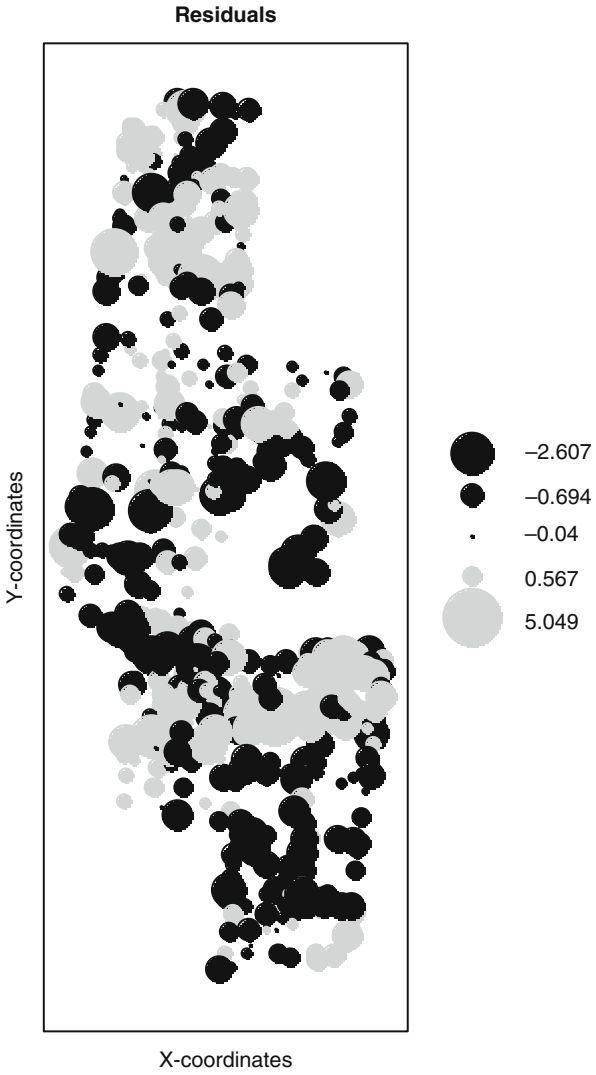

Fig. 7.1 Standardised residuals obtained by the linear regression model plotted versus their spatial coordinates. Black dotes are negative residuals, and grey dots are positive residuals

The `coordinates` command is from the `gstat` package and ensures that the `bubble` functions know that *x* and *y* are spatial coordinates. The names of the *x* and *y* columns in the `coordinates` command must match the ones from the data frame, hence the 'Boreality' in 'Boreality.x'.

As an alternative to the informal approach of making a bubble plot of residuals and judging whether there is spatial dependence, you can make a variogram of the

residuals. In Chapter 6, we used the ACF to judge whether there was dependence over time. For this, we assumed stationarity of the residuals and calculated the correlation between ε_s and ε_{s+k}, where k is the time lag. So residuals that are separated by k time units are aggregated to calculate the ACF.

In the forest data example, we do not have residuals at time s and t, but we have residuals at sites i and j and instead of using the ACF, we use the variogram. It is defined by:

$$\gamma(\mathbf{x}_1, \mathbf{x}_2) = \frac{1}{2} E\left[(Z(\mathbf{x}_1) - Z(\mathbf{x}_2))^2\right]$$

This is a function that measures the spatial dependence between two sites with coordinates $\mathbf{x}_1$ and $\mathbf{x}_2$. If these two sites are located close to each other, then you would expect the values of the variables of interest (residuals in this case) are similar. A low value of $\gamma(\mathbf{x}_1, \mathbf{x}_2)$ indicates that this is indeed the case (dependence), whereas a large value indicates independence. Spatial statistics tends to be rather complicated and intimidating. Zuur et al. (2007) discussed various aspects like ergodicity, stationarity, and weak stationarity. Without going into detail here, weak stationarity leads to the following variogram.

$$\gamma(\mathbf{h}) = \frac{1}{2} Var\left[(Z(\mathbf{x} + \mathbf{h}) - Z(\mathbf{x})\right].$$

In the same way as the ACF measures the temporal dependence by comparing the value of Z at times t and $t + k$, so does the variogram in space. Instead of comparing all time points that are separated by k units, it takes all points that are separated by a vector $\mathbf{h}$, and it uses these to calculate the sample (or experimental) variogram:

$$\hat{\gamma}(\mathbf{h}) = \frac{1}{2\,N(\mathbf{h})} \sum_{i=1}^{N(\mathbf{h})} [z(\mathbf{x}_i + \mathbf{h}) - z(\mathbf{x}_i)]^2$$

The hat notation $\hat{}$ is used because it is an estimator based on sample data. If there is spatial dependence, points close to each other tend to have similar values and the experimental variogram will be small. Large values for the experimental variogram indicate spatial independence. There are all kinds of 'little' details that play a role here, for example, the number of points that are exactly separated by a distance $\mathbf{h}$. This is the $N(\mathbf{h})$. In reality only a few points, if any, are separated exactly by a distance $\mathbf{h}$. The software code used to estimate the variogram puts a small range around $\mathbf{h}$ so that enough points are available for analysis. Another important issue is that we assume isotropy. This means that the spatial dependence of the residuals is the same in any direction. If this is not the case, we cannot calculate the variogram using sites that are separated by a distance $\mathbf{h}$ in any direction. If you do not have isotropy in the residuals, you may try to add more covariates and model the anisotropy.

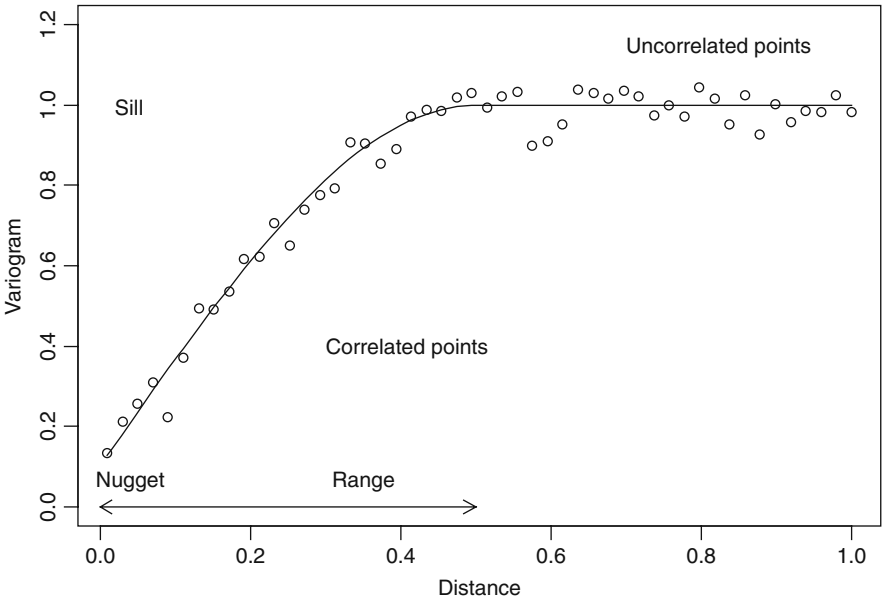

Fig. 7.2 Variogram with fitted line. The sill is the asymptotic value and the range is the distance where this value occurs. Pairs of points that have a distance larger than the range are uncorrelated. The nugget effect occurs if $\hat{\gamma}(\mathbf{h})$ is far from 0 for small $\boldsymbol{h}$

Figure 7.2 shows a theoretical variogram (line) plus some simulated data (dots). Along the *x*-axis, the distances between the sites are plotted, and along the *y*-axis, the estimated values of the variogram are plotted. Spatial dependence shows itself as an increasing band of points, which then levels off at a certain distance. The point along the *x*-axis at which this pattern levels off is called the range, and the *y*-value at the range is the sill. The nugget is the *y*-value when the distance is 0. It represents the discontinuity of the variable caused by spatial structures at distances less than the minimum distance between points.

Figure 7.3A shows the experimental variogram for the residuals of the linear regression model applied on the forest boreality data. Note that there is a clear spatial correlation up to about 1000 m. There is also a nugget effect of approximately 0.75. The following R code was used to create the experimental variogram.

```
> Vario1 <- variogram(E ~ 1, mydata)
> plot(Vario1)
```

Note that this variogram assumes isotropy; the strength of the spatial correlation is the same in each direction. We can verify this by making experimental variograms in multiple directions; see Fig. 7.3B. It seems that isotropy is a reasonable assumption as the strength, and pattern, of the spatial correlation seems to be broadly the

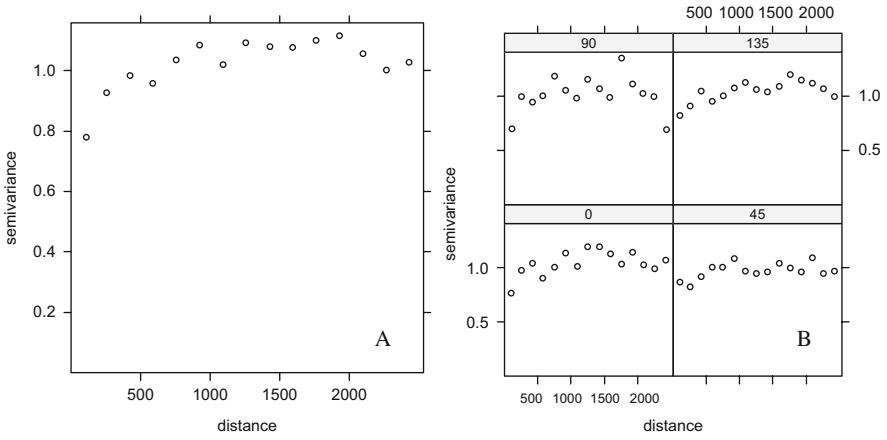

Fig. 7.3 **A**: Semi-variogram of the standardised residuals obtained by the linear regression model. The semi-variogram assumes isotropy. Note that there is spatial correlation up to 1000 m. **B**: Experimental variograms for four different directions

same in all four directions. The code to produce this graph is similar as above, except that the argument `alpha = c(0, 45, 90, 135)` is added to the variogram function.

```
> Vario2 <- variogram(E ~ 1, mydata,
                      alpha = c(0, 45, 90, 135))
> plot(Vario2)
```

7.2 Adding Spatial Correlation Structures to the Model

Both the bubble plot and the experimental variogram indicate that there is spatial correlation in the residuals, and Fig. 7.3 seems to suggest that isotropy is a reasonable assumption. We are now going from an informal assessment (looking at the bubble plot or experimental variogram) to a more formal approach. Now we include the correlation structure and use the AIC, BIC, or likelihood ratio test to judge the best model, the one with or without spatial correlation. This process works in a similar way as in the previous chapter. The only conceptual difference is that time goes in only one direction and space goes in multiple directions.

The question now is how do we include a spatial residual correlation structure in a linear regression, additive model, or (additive) mixed model? In Chapter 6, temporal dependences were included using the AR-1 or ARMA structures. Recall that these were used to parameterise the correlation function $h()$ in

$$\text{cor}(\varepsilon_s, \varepsilon_t) = \begin{cases} 1 & \text{if } s = t \\ h(\varepsilon_s, \varepsilon_t, \rho) & \text{else} \end{cases} \tag{7.1}$$

We now need to do the same trick we used with the time series, but this time, based on the shape of the variogram we need to choose a parameterisation for the correlation function $h()$. Options available in the R package `nlme` are as follows:

- Exponential correlation using the function `corExp`.
- Gaussian correlation using the function `corGaus`.
- Linear correlation using the function `corLin`.
- Rational quadratic correlation using the function `corRatio`.
- Spherical correlation using the function `corSpher`.

Each of these options implies a specific mathematical structure for the function $h()$, and a good overview is given in Schabenberger and Pierce (2002). Instead of diving straight into these formulae, it is perhaps more useful to first look at a couple of typical shapes implied by these spatial correlation structures. In Fig. 7.4, we show several theoretical variograms; the Gaussian, linear, rational quadratic, exponential, and the spherical correlation. Lines in the same panel were obtained by using a different range and sill. Some of these curves look similar and selecting the right one is a matter of expertise and pre-knowledge.

The R code for Fig. 7.4 is not given in full here. Instead, we show how to make one particular variogram, which should provide the background required to build the others.

```
> library(nlme)
> D <- seq(from = 0, to = 1, by = 0.1)
> Mydata2 <- data.frame(D = D)
> cs1C <- corSpher(c(0.8, 0.1), form = ~ D, nugget = TRUE)
> cs1C <- Initialize(cs1C, data = Mydata2)
> v1C <- Variogram(cs1C)
> plot(v1C, smooth = FALSE, type = "l", col = 1)
```

The first line creates an artificial distance gradient from 0 to 1, which we store in the data frame `mydata`. It is used in the function `corSpher`, which takes as arguments the range and the sill (optional) and the `form` option that specifies the gradient. Note that the sill is scaled to 1 by this particular `Variogram` function from the `nlme` package. The function uses the specified range and sill, substitutes these in the formulae for the spherical correlation (Schabenberger and Pierce, 2002), and calculates the corresponding variogram values. The multipanel plot in Fig. 7.4 is then a matter of repeating this with different ranges and sills and correlation functions, and then, with a bit of R magic, using the `rep` function and `xyplot`.

Some of the underlying formulae for the variogram are intimidating and some are surprisingly simple. For example, the exponential correlation structure is given by

$$\gamma(s, \rho) = 1 - e^{\frac{s}{\rho}}$$

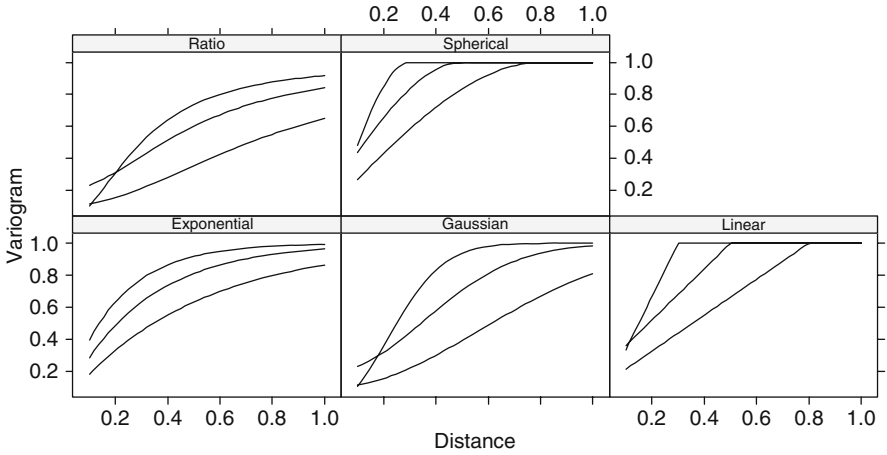

Fig. 7.4 Different variogram patterns. The three lines in the same panel were obtained using different values for the range and nugget

where ρ is the range and s the distance. If you decide to add a nugget effect c_0, the formulation changes to

$$\gamma(s,\rho) = \begin{cases} c_0 + (1-c_0)(1-e^{\frac{s}{\rho}}) & \text{if } s > 0 \\ 0 & \text{if } s = 0 \end{cases}$$

All this does is specify that the variogram is 0 for $s = 0$, shifts up the curve with c_0, and ensures it is not larger than 1. The Gaussian model is similar, but it squares the s/ρ term. The linear, rational quadratic, and spherical correlations are slightly more complicated and are not given here, but the principle is the same. The function $\gamma(s,\rho)$ is actually called the correlogram; you need to multiply it with the variance to get the variogram.

So, our task is to extract the (standardised; observed, minus fitted, and potentially corrected for heterogeneity) residuals from the linear regression or GLS model, make an experimental variogram of the residuals, and judge which correlation structure is the most appropriate. We look at this process using the boreality forest data. Instead of the function `variogram` from the `gstat` package, we use the `Variogram` function from the `nlme` package as it takes as input the object from a `gls`, `lme`, or `gamm` command. The following R code produces a similar experimental variogram for the residuals of the linear regression model as in Fig. 7.3A.

```
> library(nlme)
> f1 <- formula(Bor ~ Wet)
> B1.gls <- gls(f1, data = Boreality)
```

```
> Vario.gls <- Variogram(B1.gls, form =~ x + y,
                         robust = TRUE, maxDist = 2000,
                         resType = "pearson")
> plot(Vario.gls, smooth = TRUE)
```

The first three lines apply the same linear regression model as above (transformed boreality as a function of wetness), but now with the `gls` command. The function `Variogram` takes the object from the `gls` function and extracts the standardised residuals (because we specified this type of residuals with the `resType` option). It then calculates the experimental variogram. The x and y-coordinates are used to calculate distances (using Pythagoras theorem) between points. To aid visual interpretation, a LOESS smoother was added, but can be suppressed using `smooth = FALSE`. Sometimes it is handy to add it, and sometimes it is not. The `robust` and `maxDist` are further parameters for calculating the experimental variogram (Cressie, 1993). Spatial independence is a likely assumption if the experimental variogram shows a band of horizontal points, but this is not the case here.

Instead of judging from the experimental variogram whether residual independence can be assumed, we can add a spatial correlation structure to the GLS model and compare it with the model without the spatial correlation. The following R code adds the various correlation structures to the GLS model.

```
> B1A <- gls(f1, correlation = corSpher(form =~ x + y,
              nugget = TRUE), data = Boreality)
> B1B <- gls(f1, correlation = corLin(form =~ x + y,
              nugget = TRUE), data = Boreality)
> B1C <- gls(f1, correlation = corRatio(form =~ x + y,
              nugget = TRUE), data = Boreality)
> B1D <- gls(f1, correlation = corGaus(form =~ x + y,
              nugget = TRUE), data = Boreality)
> B1E <- gls(f1, correlation = corExp(form =~ x + y,
              nugget = TRUE), data = Boreality)
> AIC(B1, B1A, B1B, B1C, B1D, B1E)
```

We could have used the `update` command, but in this case, it does not shorten the code. If there are convergence problems (and this can happen quite often), then it may help to modify the `lmeControl` settings (see its help file). The `anova` or `AIC` command can be used to obtain the AIC values, and these are given in Table 7.1. The AIC of the model with no correlation is 2844.54, but the models with the `corLin`, `corGaus`, and `corExp` correlation structures have considerable lower AIC values, making them all candidate models. So adding a spatial correlation structure improves the model, as judged by the AIC.

We can also apply a hypothesis test with the `anova(B1, B1E)` command (we could have used any of the other candidate models). It gives $L = 116.31$, ($df = 2$, $p < 0.001$), indicating that adding a spatial correlation structure gives a significantly better model.

Table 7.1 AIC values obtained by adding various correlation structures to the linear regression model. The first column shows which correlation structure is added, the second column the object name, all models with spatial correlation use two extra parameters, and the last column gives the AIC

Model	Object	df	AIC
No correlation	B1	3	2844.54
corSpher	B1A	5	2737.01
corLin	B1B	5	2848.51
corRatio	B1C	5	2732.93
corGaus	B1D	5	2736.29
corExp	B1E	5	2732.22

From the AICs and likelihood ratio test, we can conclude that we are violating the independence assumption in the linear regression model. So the remaining question is now whether adding any of these spatial correlation structures can solve the independence problem. The commands

```
> Vario1E <- Variogram(B1E, form =~ x + y,
                       robust = TRUE, maxDist = 2000,
                       resType = "pearson")
> plot(Vario1E, smooth = FALSE)
```

will show the experimental variogram with the fitted spatial correlation (results are not shown here), and the following code

```
> Vario2E <- Variogram(B1E, form =~ x + y,
                       robust = TRUE, maxDist = 2000,
                       resType = "normalized")
> plot(Vario2E, smooth = FALSE)
```

does the same for the normalised residuals. The later ones should no longer show a spatial correlation (you should see a horizontal band of points). Results are not presented here, but the experimental variogram of the normalised residuals indeed form a horizontal band of points, indicating spatial independence.

Note that we should apply the same 10-step protocol we used in Chapters 4 and 5. First determine the optimal random structure using REML estimation, using as many fixed covariates as possible. (However, here all covariates are highly collinear; so there is effectively only one variable.) Once the optimal random structure has been found, the optimal fixed structure can be found using the tools described in Chapters 4 and 5. So, the whole REML and ML process used earlier also applies here.

For this chapter, we used the GLS model. If a random effects model is used, the spatial correlation structure is applied within the deepest level of the data. See also Chapters 16 and 17 where we impose a correlation structure on nested data.

7.3 Revisiting the Hawaiian Birds

Now we return to the Hawaiian bird data set, which we left with an AR1 autocorrelation structure. In the previous section, we used the `form =~ x + y` argument in the correlation option. If included in the `gls`, `lme`, or `gamm` function, it ensures that R calculates distances between the sampling points with coordinates given by *x* and *y*. The default option to calculate distances is Euclidean distances (using Pythagoras) and alternatives are Manhattan and maximum distances (Pinheiro and Bates, 2000). In the Hawaiian data, we used `form =~ Time | ID` in the `corAR1` function. Nothing stops us using for example a spatial correlation function like `corSpher` for time series. It can cope better with missing values and irregularly spaced data. In fact, the `corExp` structure is closely related to the `corAR1` (Diggle et al., 2002). The following code applies the model with the `corAR1` structure and all four spatial correlation functions. We copied and pasted the code from Chapter 6 to access the data.

```
> library(AED); data(Hawaii)
> Birds <- c(Hawaii$Stilt.Oahu, Hawaii$Stilt.Maui,
             Hawaii$Coot.Oahu, Hawaii$Coot.Maui)
> Time <- rep(Hawaii$Year, 4)
> Rain <- rep(Hawaii$Rainfall, 4)
> ID <- factor(rep(c("Stilt.Oahu", "Stilt.Maui",
                     "Coot.Oahu", "Coot.Maui"),
                   each = length(Hawaii$Year)))
> library(mgcv); library(nlme)
> #Define the fixed part of the model
> f1 <- formula(Birds ~ Rain + ID +
        s(Time, by = as.numeric(ID == "Stilt.Oahu"))+
        s(Time, by = as.numeric(ID == "Stilt.Maui"))+
        s(Time, by = as.numeric(ID == "Coot.Oahu"))+
        s(Time, by = as.numeric(ID == "Coot.Maui")))
> #Fit the gamms
> HawA <- gamm(f1, method = "REML", correlation =
             corAR1(form =~ Time | ID),
             weights = varIdent(form =~ 1 | ID))
> HawB <- gamm(f1, method = "REML", correlation =
             corLin(form =~ Time | ID, nugget = TRUE),
             weights = varIdent(form =~ 1| ID))
> HawC <- gamm(f1, method = "REML", correlation =
             corGaus(form =~ Time | ID, nugget = TRUE),
               weights = varIdent(form =~ 1| ID))
> HawD <- gamm(f1, method = "REML", correlation =
             corExp(form =~ Time | ID, nugget = TRUE),
             weights = varIdent(form =~ 1 | ID))
```

```
> HawE <- gamm(f1, method = "REML", correlation =
              corSpher(form =~ Time | ID, nugget = TRUE),
              weights = varIdent(form =~ 1| ID))
> #Compare the models
> AIC(HawA$lme, HawB$lme, HawC$lme, HawD$lme, HawE$lme)
          df      AIC
HawA$lme 18 2277.677
HawB$lme 19 2281.336
HawC$lme 19 2279.182
HawD$lme 19 2279.677
HawE$lme 19 2278.898
```

The results of the `AIC` command indicate that the model with the `corAR1` structure should be chosen.

7.4 Nitrogen Isotope Ratios in Whales

In this section, we analyse the nitrogen isotopic data of teeth growth layers of 11 whales. We start with one whale and then analyse the data from all whales.

7.4.1 Moby

In Chapter 2, we applied linear regression on the nitrogen isotope values of a whale nicknamed Moby, and we discussed two potential sources of violating the independence assumption. The first was a potential improper model specification (a linear relationship when the real relationship may be non-linear). The second one was due to the nature of the data; nitrogen concentrations at a certain age s may depend on the concentrations at age $s-1$, $s-2$, $s-3$, etc. To deal with the first problem, we applied a Gaussian additive model on the data for Moby:

$$y_s = \alpha + f(age_s) + \varepsilon_s \qquad \boldsymbol{\varepsilon} \sim N(0, \sigma^2 \times \mathbf{V}), \text{ where } \boldsymbol{\varepsilon} = (\varepsilon_1, \varepsilon_2 \ldots, \varepsilon_T)'$$

The index s represents year and runs from 3 to 44 for Moby. The variable y_s contains the isotopic value in year s, α is the intercept, age_s is the age in year s, $f(age_s)$ is the smoothing function of age, and ε_s are the residuals. In an ordinary Gaussian additive model (or linear regression model), we assume that the residuals are independent and normally distributed with mean 0 and variance σ^2. This means that $\mathbf{V}$ is a 42-by-42 identity matrix. (This is matrix full of zeros, except for the diagonal; these are all equal to 1.)

To allow for a dependence structure between the observations, we can use any of the correlation structures discussed earlier in Chapter 6 or in this chapter. Instead of

temporal or geographical coordinates, age is now the variable that we use to set up the variogram. As a consequence, **V** is no longer a diagonal matrix. Its off-diagonal elements give the residual covariance at different ages. The key question is now, how we should parameterise this matrix. Clearly, using a completely unspecified matrix results in too many unknown parameters. We can use the variogram or the AR1 residual correlation structures. These will specify that observations that are separated by an age of k years have a correlation as specified by, for example, the linear, spherical, exponential, or Gaussian variogram structure. All we have to do is to apply models with different covariance structures and assess which one is the most appropriate using, for example, the AIC.

The model selection process is identical to mixed modelling; (i) start with a model that contains as many explanatory variables as possible, (ii) find the optimal random structure, and (iii) find the optimal fixed structure. If we have data on only one whale, the first step is rather simple: use age. The following code imports the data, extracts the data from Moby, and applies the models.

```
> library(AED); data(TeethNitrogen)
> TN <- TeethNitrogen
> N.Moby <- TN$X15N[TN$Tooth == "Moby"]
> Age.Moby <- TN$Age[TN$Tooth == "Moby"]
> library(mgcv); library(nlme)
> f <- formula(N.Moby ~ s(Age.Moby))
> #Apply gamm models
> Mob0 <- gamm(f, method = "REML")
> Mob1 <- gamm(f, method = "REML", cor =
             corSpher(form =~ Age.Moby, nugget = TRUE))
> Mob2 <- gamm(f, method = "REML", cor =
             corLin(form =~ Age.Moby, nugget = TRUE))
> Mob3 <- gamm(f, method = "REML", cor =
             corGaus(form =~ Age.Moby, nugget = TRUE))
> Mob4 <- gamm(f, method = "REML", cor =
             corExp(form =~ Age.Moby, nugget = TRUE))
> Mob5 <- gamm(f, method = "REML", cor =
             corRatio(form =~ Age.Moby, nugget = TRUE))
> Mob6 <- gamm(f, method = "REML", cor =
             corAR1(form =~ Age.Moby))
> AIC(Mob0$lme, Mob1$lme, Mob1$lme, Mob4$lme, Mob5$lme,
      Mob6$lme)

Mob0$lme  4 64.52995
Mob1$lme  6 67.02795
Mob2$lme  6 67.02795
Mob4$lme  6 63.38405
Mob5$lme  6 63.09320
Mob6$lme  5 63.60480
```

The model with the `corGaus` correlation structure did not converge and is therefore not used in the `AIC` command. Except for the `corSpher` correlation structure, all AICs are similar; hence, we might as well choose the simplest model, which is the one without a correlation structure (the linear regression model, `Mob0`). Comparing model `Mob0` with `Mob5` (`Mob0` is the model without a correlation structure and `Mob5` is the best *potential* model with respect to spatial correlation) using a likelihood ratio test gave a p-value of 0.06 (just type: `anova(Mob0$lme, Mob5$lme)`). Hence, there is no convincing evidence to use a correlation structure for the data of this whale. Furthermore, the estimated smoother in model `Mob5` is a straight line. This indicates that for the Moby data, the linear regression model that was presented in Chapter 2 suffices. This is rather confusing as the model did have residual patterns!

7.4.2 All Whales

What about the other whales? Instead of applying the above method on each individual whale data, we apply one additive model on the data of all animals and estimate one underlying 'spatial' correlation structure. This is the same approach we applied on the Hawaiian time series in Chapter 6. The following model was applied:

$$N_{is} = \alpha_i + f_i(Age_{is}) + \varepsilon_{is}$$

The subscript i refers to whale ($i = 1, \ldots, 11$) and s to year. Here, we assume that each whale i has a potentially different age-effect on isotopic nitrogen values, hence the subscript i for the smoothing function f. Later, in the case studies, we show how we can test whether multiple smoothers can be replaced by one or a few smoothers using a deep sea research data set.

In a standard application of this model, the residuals ε_{is} are assumed to be independent and normally distributed with mean 0 and covariance matrix $\sigma^2\mathbf{V}_i$, where $\mathbf{V}_i$ is an identity matrix. The size of this matrix depends on the number of observations for whale i. This is perhaps clearer if we switch to a vector notation.

$$\mathbf{N}_i = \boldsymbol{\alpha} + \mathbf{f}(\mathbf{Age}_i) + \boldsymbol{\varepsilon}_i$$

Each vector contains all the age data for one whale. A dependence structure between residuals of different ages can be introduced by using a non-diagonal matrix $\mathbf{V}_i$, just as we did for the Moby data earlier in this section. We use the data from all the 11 whales and apply the correlation structure at the deepest level within a time series for each whale. All whales are assumed to have the same spatial correlation structure.

A model that contains as many explanatory variables as possible is a model that has one smoother per whale. This means that we have to use 11 smoothers, and this could potentially take considerable computing power (even with modern

computers). We therefore use cubic regression splines as these are less time consuming to calculate than the default thin plate regression spline smoother (Wood, 2006).

The following R code applies the model in R. Note the use of the `by` command in the smoother; it ensures we have one smoother per whale.

```
> lmc <- lmeControl(niterEM = 5200, msMaxIter = 5200)
> AllWhales.corGaus <- gamm( X15N ~
   s(Age,by=as.numeric(Tooth=="M2679/93"),bs="cr") +
   s(Age,by=as.numeric(Tooth=="M2683/93"),bs="cr") +
   s(Age,by=as.numeric(Tooth=="M2583/94(1)"),bs="cr") +
   s(Age,by=as.numeric(Tooth=="M2583/94(7)"),bs="cr") +
   s(Age,by=as.numeric(Tooth=="M2583/94(10)"),bs="cr")+
   s(Age,by=as.numeric(Tooth=="M546/95"),bs="cr") +
   s(Age,by=as.numeric(Tooth=="M143/96E"),bs="cr") +
   s(Age,by=as.numeric(Tooth=="Moby"),bs="cr") +
   s(Age,by=as.numeric(Tooth=="M447/98"),bs="cr") +
   s(Age,by=as.numeric(Tooth=="I1/98"),bs="cr") +
   factor(Tooth), control = lmc, method = "REML",
   correlation = corGaus(form =~ Age|Tooth, nugget=T),
   data = TN)
```

Besides the `corGaus` correlation structure, we also applied all the other correlation structures we discussed earlier in this chapter. Using no correlation structure gave AIC = 529.16. The lowest AIC value was obtained by the `corGaus` structure with a value of 478.82, closely followed by the corRatio. Other correlation structures were all slightly higher (around 485). This shows that a correlation structure improves the model considerably! The estimated range by the `corGaus` structure was 2.9 years. This means that after removing the age effect, the nitrogen isotopic values are correlated up to 2.9 years.

An interesting question is then what are the differences between the models with and without the `corGaus` correlation structure? The results of the model without the correlation structure are presented below. The object `AllWhales.0$gam` was fitted with the code below, except that the correlation option was removed.

```
> anova(AllWhales.0$gam)

Approximate significance of smooth terms:
                                          edf       F      p-value
s(Age):as.numeric(Tooth=="M2679/93")    6.055    58.440    < 2e-16
s(Age):as.numeric(Tooth=="M2683/93")    1.000    39.421   1.42e-09
s(Age):as.numeric(Tooth=="M2583/94(1)") 1.000   175.088    < 2e-16
s(Age):as.numeric(Tooth=="M2583/94(7)") 4.215    12.742   1.30e-12
s(Age):as.numeric(Tooth=="M2583/94(10)") 3.839    6.103   5.32e-06
s(Age):as.numeric(Tooth=="M546/95")     4.039    18.847    < 2e-16
s(Age):as.numeric(Tooth=="M143/96E")    1.000    32.316   3.50e-08
s(Age):as.numeric(Tooth=="Moby")        4.272    44.760    < 2e-16
```

```
s(Age):as.numeric(Tooth=="M447/98")      4.408     21.910   < 2e-16
s(Age):as.numeric(Tooth=="I1/98")        5.244     14.361   < 2e-16
```

All the smoothers are highly significant at the 5% level. However, this model ignores the potential dependence. The following results were obtained by the model with the `corGaus` correlation structure.

```
> anova(AllWhales.corGaus$gam)

Approximate significance of smooth terms:
                                          edf       F      p-value
s(Age):as.numeric(Tooth=="M2679/93")    1.000    86.928    < 2e-16
s(Age):as.numeric(Tooth=="M2683/93")    1.000    10.746   0.001178
s(Age):as.numeric(Tooth=="M2583/94(1)") 1.000    48.341   2.56e-11
s(Age):as.numeric(Tooth=="M2583/94(7)") 2.414     4.983   0.000218
s(Age):as.numeric(Tooth=="M2583/94(10)") 3.290    4.715   0.000137
s(Age):as.numeric(Tooth=="M546/95")     3.371     5.071   5.89e-05
s(Age):as.numeric(Tooth=="M143/96E")    1.000     7.896   0.005307
s(Age):as.numeric(Tooth=="Moby")        1.000    73.198   8.08e-16
s(Age):as.numeric(Tooth=="M447/98")     1.000    32.954   2.47e-08
s(Age):as.numeric(Tooth=="I1/98")       3.336     3.035   0.004317
```

Note that there are considerable differences in the *p*-values, and the model without the correlation structure giving a rosier but misleading picture in terms of significance levels!

These models assume the same residual spread per whale and over time. A model validation did not reveal any immediate problems with homogeneity, but the analysis may be extended by allowing for different spread per whale, which means the use of the `weights` and `varIdent` functions. The reason that we mention this is that most examples used in this book contain some form of heterogeneity. It would be a small miracle if this is not also the case here.

To save some parameters, it is also possible to use Tooth as random effect instead of a fixed nominal variable with 11 levels. It is also interesting to compare the compound symmetric correlation structure (by using a random intercept) versus the spatial correlation model. Or perhaps, use both correlation structures: a spatial correlation within the random effect tooth. We leave this an exercise for the reader.

Mendes et al. (2007) analysed the same data and looked at sudden changes in nitrogen isotopic values. Multivariate time series techniques like chronological clustering were used (Legendre and Legendre, 1998). Such an analysis can also be carried out within the additive mixed modelling framework. A dummy variable (also called intervention variable in this context) is an explanatory variable of the form 0 0 0 0 0 0 . . . 1 1 1 1 1 (Harvey, 1989). These can be used to test for sudden changes using a model of the form

$$N_{is} = \alpha_i + f_i(Age_{is}) + \beta_i \times D_{is} + \varepsilon_{is}$$

where D_{is} is a vector of zeros and ones. A sudden increase in nitrogen isotope ratios for a particular whale can be tested by looking at the significance of the regression parameter β_i. The only problem is at which age the dummy variable D_{is} should switch from a zero to a one. Adding an optimisation routine that tries different switching points per whale and compares them using the AIC may be an option. This is also called intervention analysis (Harvey, 1989; Zuur et al., 2007).

Something we have ignored so far is the assumption of a fixed X. Recall that this means that before sampling, we know the value of the explanatory variables. For the whale data, this assumption is clearly violated as there may be an error of 1–2 years on an age reading. Bootstrapping (Efron and Tibshirani, 1993) may be a tool to deal with this. There are many ways to carry out a bootstrap, and one of these, to apply an ordinary bootstrap, is as follows.

1. Apply the smoothing model for the given data, and estimate the smoothers, etc. Obtain the fitted values and the residuals for the original data.
2. Permute the residuals from step 1, or apply a parametric bootstrap on the residuals. Add the permuted residuals to the fitted values from step 1. This gives bootstrapped data (response variable).
3. Apply the smoothing model on the new data.
4. Repeat steps 2 and 3 1000 times.
5. Use the 1000 estimated smoothers to create confidence bands.

More details can be found in Davison and Hinkley (1997). To take account of the age of 20 years being anything between 18 and 22, we can add an extra permutation step to the algorithm described above that will slightly modify the age in each bootstrap iteration. Note that this 5-step scheme is not a full recipe. Details on bootstrapping GAMs can be found in Davison and Hinkley (1997) and Keele (2008).

7.5 Spatial Correlation due to a Missing Covariate

In this section, we show how a missing covariate may cause spatial correlation. The data used are a subset of the data analysed in Cruikshanks et al. (2006), a technical report by the Environmental Protection Agency, Wexford, Ireland). We only use the 2003 data, and several recordings were dropped. So, our results may be different from those presented in the original report.

The original research sampled 257 rivers in Ireland during 2002 and 2003. One of the aims was to find a different tool for identifying acid-sensitive waters, which currently uses measures of pH. The problem with pH is that it is extremely variable within a catchment and depends on both flow conditions and underlying geology. As an alternative measure, the Sodium Dominance Index (SDI) is proposed as an indicator of the acid sensitivity of rivers. SDI is defined as the contribution of sodium (Na+) to the sum of the major cations. The motivation for this research is

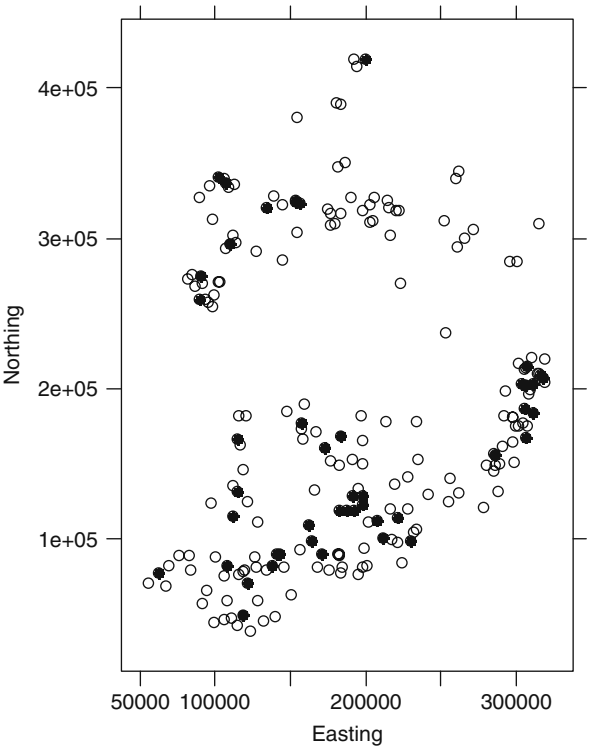

Fig. 7.5 Positions of the sites in Ireland that were sampled in 2003. Filled circles are forested sites and open circles the non-forested sites

the increase in plantation forestry cover in Irish landscapes and its potential impacts on aquatic resources. Of the 257 sites, 192 were non-forested and 65 were forested.

In this section, we model pH as a function of SDI, whether a site is forested or not, and altitude. Figure 7.5 shows the geographical position of the sites in Ireland that were sampled in 2003. The following code accesses the data and makes the graph.

```
> library(AED); data(SDI2003);
> library(lattice)
> MyPch <- vector(length = dim(SDI2003)[1])
> MyPch[SDI2003$Forested == 1] <- 16
> MyPch[SDI2003$Forested == 2] <- 1
> xyplot(Northing ~ Easting, aspect = "iso", col = 1,
         pch = MyPch, data = SDI2003)
```

The `xyplot` from the `lattice` package is used to ensure that the units along the vertical and horizontal axes are the same; see also Chapter 16 for other ways of doing this. The variable `MyPch` is used to plot different types of dots for forested and non-forested sites.

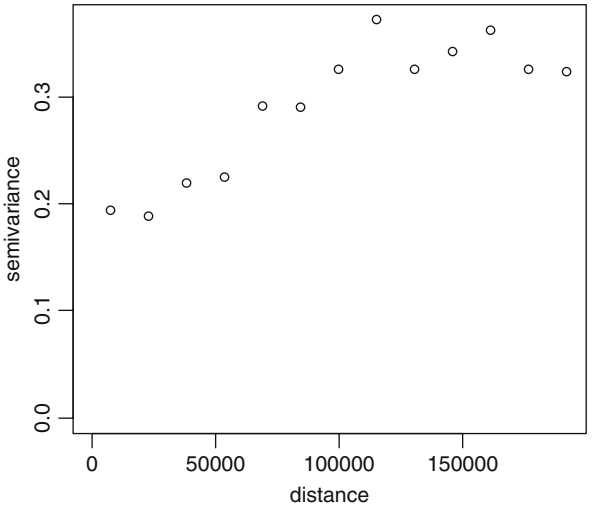

Fig. 7.6 Experimental variogram for the pH data sampled in Ireland in 2003. Note that there is spatial dependence because there is an increase in the experimental variogram

We first show that there is spatial correlation in the pH data with help of an experimental variogram (Fig. 7.6). Results clearly indicate that there is spatial dependence as the pattern slowly increases and then levels off. Earlier in this chapter, we used the variogram function from the `gstat` package. Here, we use yet another package to make variograms, namely `geoR`. In practise, these packages tend to give similar results, but it is useful to know (and be able to use) that there are multiple packages for the same thing.

The code to make Fig. 7.6 is given below.

```
> library(geoR)
> cords <- matrix(0, length(SDI2003$pH), 2)
> coords[, 1] <- SDI2003$Easting;
> coords[, 2] <- SDI2003$Northing
> gb <- list(data = SDI2003$pH, cords = coords)
> plot(variog(gb, max.dist = 200000))
```

Before adding spatial correlation structures, we should first apply a model without spatial correlation structures, extract its residuals, and see whether these residuals show spatial dependence. After all, we may be able to explain the spatial patterns in pH with SDI or altitude. The following linear regression model (in words) is applied.

$$\text{pH}_i = \alpha + \text{SDI}_i \times \text{Altitude}_\text{i} \times \text{factor}(\text{Forested}_i) + \varepsilon_i$$

Actually, we used the log-transformed altitude. The model contains 3 main terms, all 2-way interactions, and one 3-way interaction term, and the residuals

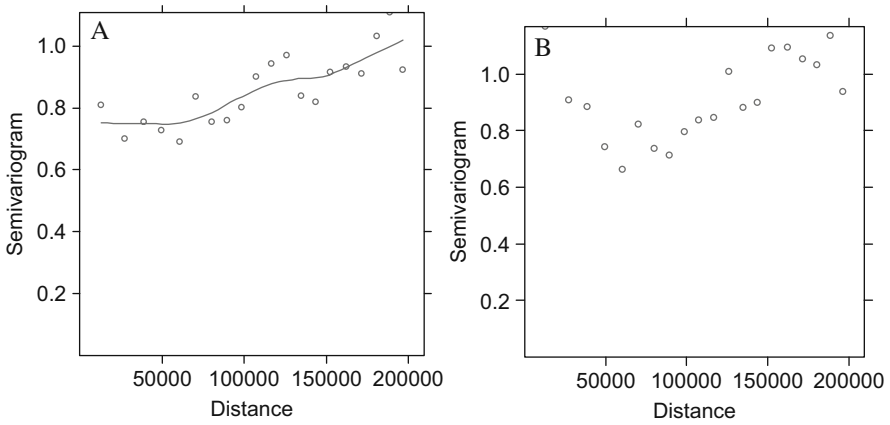

Fig. 7.7 **A**: Experimental variogram of the residuals obtained by applying a linear regression model on pH. Note that there is evidence that the independence assumption is violated. The line is a smoother and can be suppressed by adding `smooth = FALSE` to the `plot(Vario1)` command. **B**: Experimental variogram of the normalised residual using the `corRatio` structure. The `corExp` is equally bad

are assumed to be independent and normally distributed with mean 0 and variance σ^2. Homogeneity and normality are valid assumptions, and the numerical output indicates that we may expect a significant SDI effect and a significant altitude × Forested effect. The following R code applies the linear regression model and draws an experimental variogram of the residuals (Fig. 7.7A). A smoother was added to aid visual interpretation.

```
> library(nlme)
> SDI2003$fForested <- factor(SDI2003$Forested)
> SDI2003$LAltitude <- log(SDI2003$Altitude)
> M1 <- gls(pH ~ SDI * fForested * LAltitude,
            data =  SDI2003)
> Vario1 <- Variogram(M1, form =~ Easting + Northing,
          data = SDI2003, nugget = TRUE, maxDist = 200000)
> plot(Vario1)
```

The AIC of the GLS model without auto-correlation is 248.34. Just as in previous sections, we can add any of the five available correlation structures to the GLS and the `corRatio` and `corExp` structures give considerable lower AICs: 205.95 and 208.57, respectively. These models are implemented with the following code:

```
> M1C <- gls(pH ~ SDI * fForested * LAltitude,
             correlation = corRatio(form =~ Easting +
             Northing, nugget = TRUE), data = SDI2003)
```

```
> M1E <- gls(pH ~ SDI * fForested * LAltitude,
             correlation = corExp(form =~ Easting +
             Northing, nugget = TRUE), data = SDI2003)
```

However, neither of these correlation structures produces a fitted line that matches the experimental variogram. Figure 7.7B shows the experimental variogram of the normalised residuals and shows a clear pattern. It was made with the following R code. If you remove the `resType` option, the plot function shows the fitted experimental variogram.

```
> Vario1C <- Variogram(M1C, form =~ Easting + Northing,
       data = SDI2003, nugget = TRUE, maxDist = 200000,
       resType = "normalized")
> plot(Vario1C, smooth = FALSE)
```

We could try and choose fixed values for the nugget and range, but the real problem is that we are missing a covariate. This can be seen from a bubble plot of the normalised residuals of the linear regression model (Fig. 7.8). The negative residuals are mainly clustered along the south and south-east coastline, and the western coastline mainly contains positive residuals. So there is a clear pattern in these residuals. To solve this problem, we need to think very carefully about which missing covariate could be causing this type of pattern and hope that it can (still) be quantified

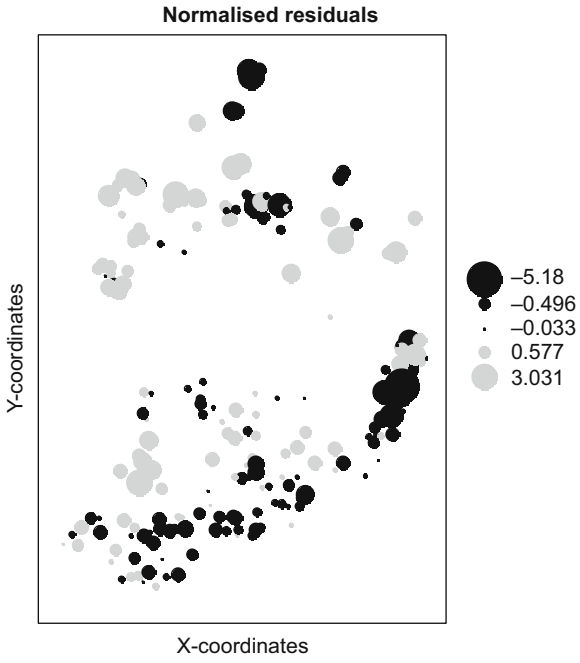

Fig. 7.8 Normalised residuals of the linear regression model plotted against spatial coordinates. The size of the dot is proportional to its value, and the colour refers to the sign. Note that most negative residuals are clustered along the south-east coast, and the west coast mainly contains positive residuals

and included in the model. In the meantime, we should refrain from making any inferential conclusions from these models, and we cannot say (yet) whether there is an altitude × Forested interaction or an SDI effect on pH.

The following R code was used to create the bubble plot.

```
> library(gstat)
> E <- resid(M1, type = "normalized")
> mydata <- data.frame(E, SDI2003$Easting,
                       SDI2003$Northing)
> coordinates(mydata) <- c("SDI2003.Easting",
                           "SDI2003.Northing")
> bubble(mydata, "E", col = c("black", "grey"),
        main = "Normalised residuals",
        xlab = "X-coordinates", ylab = "Y-coordinates")
```

7.6 Short Godwits Time Series

In the previous chapter, we showed how to include a temporal correlation structure using relatively long and regularly spaced time series with the `corAR1` and `corARMA` functions. In earlier sections in this chapter, we had spatial data and data along an age gradient. In all cases, the length of the gradient was long. We now use an example that consists of rather short and irregularly spaced time series of feeding behaviour patterns in the godwits (*Limosa haemastica*) data (Ieno, unpublished data).

7.6.1 Description of the Data

Food intake rates of migrating godwits were observed at a tidal channel, on a section of a South Atlantic mudflat system in Argentina (Samborombón Bay). Sampling took place on 20 (non-sequential) days, divided over three consecutive periods. On the basis of plumage and time of the year, birds were classified as 'birds preparing for migration' (southern late summer/fall) and 'birds not preparing for migration'. The second group can be further divided in southern spring/early summer, and southern winter. Measurements took place during the low water period on at least two days per month during 15 consecutive months.

On each sampling day, between 7 and 19 observations were taken, which gives us short longitudinal time series per day.

The observations consist of the food intake rates, which is the mg of Ash free dried prey (nereid worm) weight per second of feeding (mg AFDW/s). The time when the godwits took food was also recorded. Because time itself has no ecological meaning for the birds, it is expressed in hours before and after the low tide.

The underlying question is whether intake rate depends on period of migration, time with respect to low tide (does food consumption depend on the tide), and sex. What we have in mind is a model of the form:

$$\text{IntakeRate}_{ij} = \textit{function}(\text{Time}_{ij}, \text{Sex}_{ij}, \text{Period}_{ij}) + \varepsilon_{ij}$$

IntakeRate_{ij} is the intake rate of observation j on day i. Time_{ij} is the corresponding time. It tells you how many minutes before or after low tide an observation was made. Sex has the values unknown, female or male. Period is a nominal variable with three levels; 0 if an observation was made in January, September–December; 1 if an observation was made during February, March, or April; and 3 for May–August. These three periods represent the migration 'status' of godwits as explained above.

The potential complicating factor is that the intake rate at a particular time on a particular day may depend on the intake rate at an earlier time on the same day. Your alarm bells for violation of independence should now make a lot of noise!

7.6.2 Data Exploration

As always, we started the statistical analysis with a detailed graphical data exploration. Results are not presented here, but none of the data exploration tools (boxplots, Cleveland dotplots, and pairplots) indicated any outliers. The coplot in Fig. 7.9

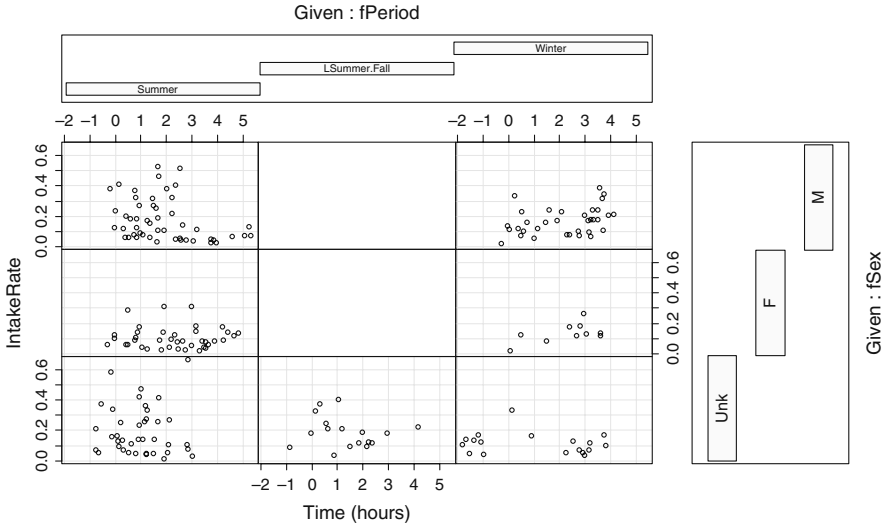

Fig. 7.9 Coplot of intake rate versus time (time since low tide in hours), conditional on sex and period. Note that in late summer and fall, not all sexes were measured

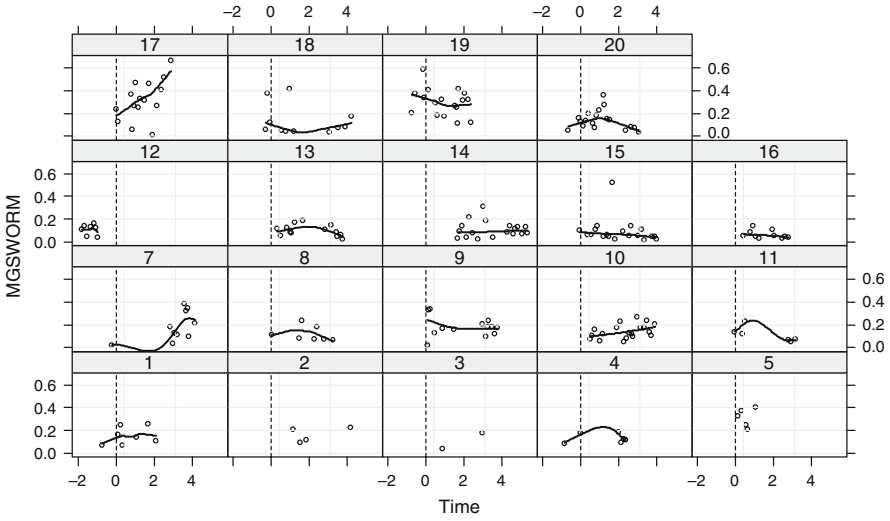

Fig. 7.10 An `xyplot` from the lattice package. The *y*-axis shows the intake rate (mg AFDW/s) of godwits, and the *x*-axis the time since low tide in hours. The numbers 1–20 refer to the sampling day. The vertical dotted line is the moment of low tide

shows that in some periods (late summer and fall), not all sexes are measured. Hence, we cannot include a sex–period interaction term.

The coplot accumulates the data from all sampling days. To show how intake rate changes on each day, we made an xyplot from the lattice package (Fig. 7.10). We added a LOESS smoother to aid visual interpretation. At some days, there seems to be a non-linear time effect; hence, we should perhaps model time as a quadratic function.

7.6.3 Linear Regression

Based on the data exploration, we think that a reasonably starting model is

$$\text{IntakeRate}_{ij} = \alpha + \beta_1 \times \text{Time}_{ij} + \beta_2 \times \text{Time}_{ij}^2 + \beta_3 \times \text{Sex}_{ij} + \beta_4 \times \text{Period}_{ij} + \varepsilon_{ij}$$

where the residuals are independently and normally distributed with mean 0 and variance σ^2. The R code to import the data, make the two graphs, and apply the linear regression model is given below.

```
> library(AED); data(Limosa)
> Limosa$fID <- factor(Limosa$ID)
> Limosa$fPeriod <- factor(Limosa$Period,
                           levels = c(0, 1, 2),
```

```
                    labels = c("Summer", "LSummer.Fall",
                               "Winter"))
> Limosa$fSex <- factor(Limosa$Sex, levels = c(0, 1, 2),
                        Labels = c("Unk", "F", "M"))
> coplot(IntakeRate ~ Time | fPeriod * fSex,
         data = Limosa, xlab = c("Time (hours)"))
> library(lattice)
> xyplot(IntakeRate ~ Time | fID, data = Limosa,
    panel=function(x, y){
       panel.xyplot(x, y, col = 1, cex = 0.5, pch = 1)
       panel.grid(h = -1, v = 2)
       panel.abline(v = 0, lty = 2)
       if (length(x) > 5) panel.loess(x, y, span = 0.9,
                                      col = 1, lwd = 2)
  })
```

The first line accesses the data from our package. Because the nominal variables Sex and Period were coded as 0, 1, and 2, we relabelled them; this will make the numerical output of the models easier to understand. The `coplot` command is straightforward and the `xyplot` has some fancy commands in the panel function to draw the LOESS smoother (a smoother is only added if there are at least 5 observations on a particular day). With so few data points, we choose a large span width. The linear regression is applied with the following code. We also produce some numerical output.

```
> Limosa$Time2 <- Limosa$Time^2 - mean(Limosa$Time^2)
> M.lm <- lm(IntakeRate ~ Time + Time2 + fPeriod +
             fSex, data = Limosa)
> drop1(M.lm, test = "F")

Single term deletions
Model: IntakeRate ~ Time + Time2 + fPeriod + fSex

        Df Sum of Sq     RSS     AIC F value    Pr(F)
<none>                  2.74 -881.37
Time     1      0.01    2.75 -882.51  0.8330 0.362515
Time2    1      0.03    2.77 -881.10  2.2055 0.139095
fPeriod  2      0.01    2.75 -884.25  0.5460 0.580142
fSex     2      0.13    2.87 -875.73  4.7675 0.009491
```

We centred the quadratic time component to reduce the collinearity. Note that there is a significant sex effect; the F statistic is 4.76 with a corresponding p-value of 0.009. Good enough to start thinking about writing a paper! But to spoil the fun, let us plot the residuals versus the fitted values (Fig. 7.11) with the command `plot(M.lm, which = c (1))`. Note that there is clear violation of homogeneity. It is now time to go back to the protocols from Chapters 4 and 5.

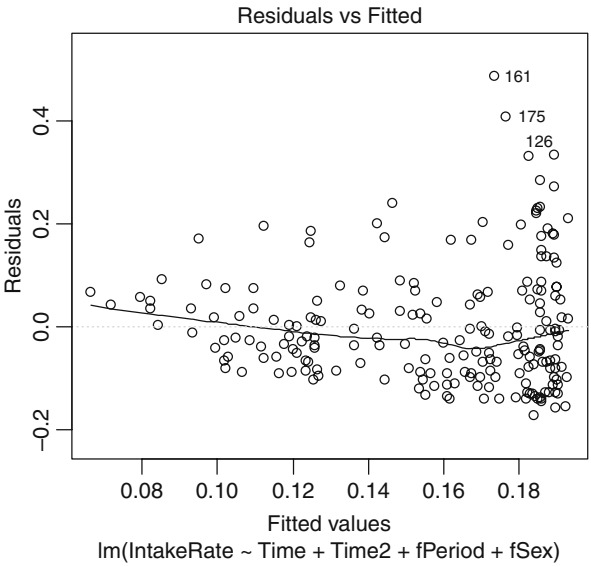

Fig. 7.11 Residuals versus fitted values for the linear regression model. Note that there is heterogeneity

7.6.4 Protocol Time

In the previous subsection, we detected heterogeneity in the residuals of the linear regression model (which is step 1 of the protocol). We can now do two things. We can either mess around with variance covariates and then discover that there is still misery (in terms of correlation) or be clever and do everything at once. Assuming that you read this book from A to Z (and are therefore familiar with the material in Chapters 4 and 5), we follow the second approach. We will use the 10-step protocol from Chapter 4.

7.6.4.1 Step 2 of the Protocol: Refit with gls

In this step, we refit the linear regression with the `gls` function (so that we have a base model) and make some fancy graphical validation graphs; see Fig. 7.12. The R code does not contain any new aspects.

```
> library(nlme)
> M1.gls <- gls(IntakeRate ~ Time + Time2 + fPeriod +
                fSex, data = Limosa)
> E <- resid(M1.gls)
> op <- par(mfrow = c(2, 2))
> boxplot(E  ~ Limosa$fPeriod, main = "Period")
> abline(0, 0)
> boxplot(E ~ Limosa$fSex, main = "Sex")
```

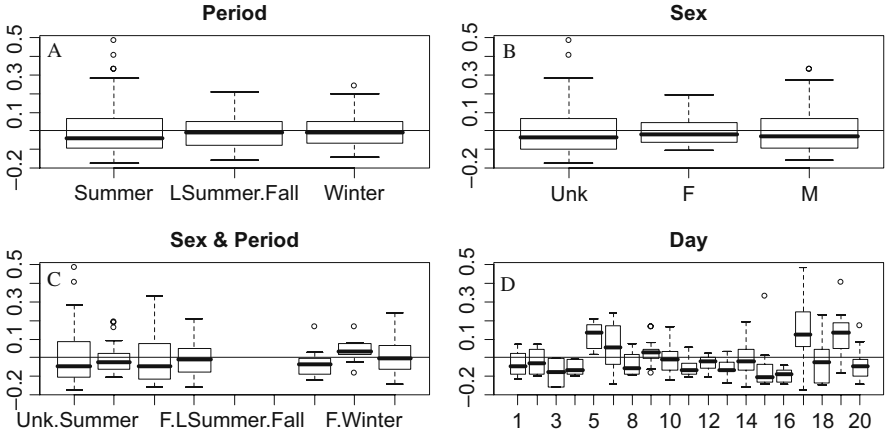

Fig. 7.12 Graphical validation plots for the linear regression model fitted with the `gls` function. **A**: Residuals versus period. **B**: Residuals versus sex. **C**: Residuals versus sex and period. **D**: Residuals versus day (coded by the variable ID). Due to lack of space, not all labels are presented on panels C and D

```
> abline(0, 0)
> boxplot(E ~ Limosa$fSex * Limosa$fPeriod,
          main = "Sex & Period")
> abline(0, 0)
> boxplot(E ~ Limosa$ID, main = "Day")
> abline(0, 0)
> par(op)
```

Note that the variation in residual spread is larger for the unknown sex, and it is also larger for the summer period. This means that in step 3 of the protocol, we could do with a `varIdent` variance structure with the variance covariates Period and Sex. Figure 7.12D shows that we need the term ID (sampling day) as an explanatory variable; at some days, all the residuals are above or below zero. We can either use ID as a fixed effect or as a random effect. In this example, it is obvious to use it as a random effect (it allows for correlation between observations from the same day; it avoids estimating lots of parameters and it allows us to generalise the conclusions); see also Chapter 5.

7.6.4.2 Step 3 of the Protocol: Choose an Appropriate Variance Structure

We already discussed in the previous step that we need a `varIdent` variance structure and ID as random effect. Such a model is given by

```
> M1.lme <- lme(IntakeRate ~ Time + Time2 + fPeriod +
        fSex, data = Limosa,
```

```
        weights = varIdent(form =~ 1 | fSex * fPeriod),
        random =~ 1 | fID, method = "REML")
```

Perhaps it is useful to give the corresponding equation for this, just in case you find it difficult to see this from the R code.

$$IntakeRate_{ij} = \alpha + \beta_1 \times Time_{ij} + \beta_2 \times Time_{ij}^2 + \beta_3 \times Period_{ij} + \beta_4 \times Sex_{ij} + a_i + \varepsilon_{ij}$$
$$a_i \sim N(0, d^2)$$
$$\varepsilon_{ij} \sim N(0, \sigma^2_{Sex,Period})$$

We have seen most of this equation already in Section 7.6.1. The term a_i is the random intercept (Chapter 5). The subscripts for the σ are there because we allow for different residual variances depending on sex and period.

7.6.4.3 Steps 4–6 of the Protocol: Find the Optimal Random Structure

We are going to save some space by summarising a couple of model selection steps. The model that was fitted in step 3 is the optimal one in terms of the random structure. Leave out the random effect, refit the model, and compare both models with the likelihood ratio test, and you will get p-values smaller than 0.001. The same holds if you drop the `varIdent` variance structure if you use the `varIdent` with only sex or only with period. The R code to do these analyses was given in Chapters 4 and 5.

7.6.4.4 Steps 7–8 of the Protocol: Find the Optimal Fixed Structure

It is now time to find the optimal model in terms of the explanatory variables time, period, and sex. We use the likelihood ratio test with ML estimation. The starting model contains Time, Time^2, Period, and Sex. The last three can be dropped (Time is nested in Time^2 and cannot be dropped). The R code to do this is as follows.

```
> M1.lme <- lme(IntakeRate ~ Time + Time2 + fPeriod +
        fSex, data = Limosa,
        weights = varIdent(form =~ 1 | fSex * fPeriod),
        random =~ 1 | fID, method = "ML")
> M1.lmeA <- update(M1.lme, .~. -Time2)
> M1.lmeB <- update(M1.lme, .~. -fPeriod)
> M1.lmeC <- update(M1.lme, .~. -fSex)
> anova(M1.lme, M1.lmeA)
> anova(M1.lme, M1.lmeB)
> anova(M1.lme, M1.lmeC)
```

The output is not shown here, but the least significant term is Period ($L = 1.28$, $df = 2$, $p = 0.52$); hence, it can be dropped. In the next round, Time^2 is

dropped, followed by Time in the third round. In the fourth and last round, we have a model that only contains Sex. The following code gives us one *p*-value for the nominal variable Sex (the `update` command fits a model with only the intercept):

```
> M4.lme <- lme(IntakeRate ~ fSex, data = Limosa,
        weights = varIdent(form =~ 1 | fSex * fPeriod),
        random =~ 1 | fID, method = "ML")
> M4.lmeA <- update(M4.lme, .~. -fSex)
> anova(M4.lme, M4.lmeA)
```

```
        Model df       AIC       BIC   logLik   Test L.Ratio p-value
M4.lme      1 11 -359.3379 -322.6779 190.6689
M4.lmeA     2  9 -355.4784 -325.4839 186.7392 1 vs 2 7.85945  0.0196
```

Hence, the optimal model contains only Sex in the fixed part of the model. If we have to quote a *p*-value for this term, it will be 0.0196, which is not very impressive. A model validation shows that everything is now ok (no heterogeneity patterns in the normalised residuals).

7.6.4.5 Step 9 of the Protocol: Refit with REML

We now discuss the numerical output of the model. First we have to refit it with REML.

```
> M4.lme <- lme(IntakeRate ~ fSex, data = Limosa,
        weights = varIdent(form =~ 1 | fSex * fPeriod),
        random =~ 1 | fID, method = "REML")
> summary(M4.lme)
```

```
Linear mixed-effects model fit by REML. Data: Limosa
        AIC       BIC   logLik
  -340.1566 -303.6573 181.0783

Random effects:
 Formula: ~1 | fID
        (Intercept)  Residual
StdDev:  0.06425989 0.1369959

Variance function:
 Structure: Different standard deviations per stratum
 Formula: ~1 | fSex * fPeriod

 Parameter estimates:
Unk*Summer  Unk*LSummer.Fall   M*Winter  F*Winter  Unk*Winter
    1.0000             0.4938     0.6249    0.5566      0.5035
 M*Summer          F*Summer
   0.7971            0.4366
```

```
Fixed effects: IntakeRate ~ fSex
                  Value  Std.Error  DF   t-value p-value
(Intercept)  0.15051634 0.01897585 186  7.931993  0.0000
fSexF       -0.02507688 0.01962955 186 -1.277506  0.2030
fSexM        0.01999006 0.01863430 186  1.072756  0.2848

 Correlation:
      (Intr) fSexF
fSexF -0.491
fSexM -0.470  0.653

Number of Observations: 207. Number of Groups: 19
```

Let us discuss what this all means. Recall from Chapter 5 that in a mixed effects model with random intercept, the correlation between the observations from the same group (in this case: the same day), is given by

$$\frac{d^2}{d^2 + \sigma^2}$$

The problem is that in this case, we do not have one variance σ^2, but we have a σ^2 that depends on Sex and Period. This means that the within-day correlation is given by

$$\frac{d^2}{d^2 + (s_{ij} \times \sigma^2)} = \frac{0.064^2}{0.064^2 + (s_{ij} \times 0.136)^2}$$

The s_{ij}s are the multiplication factors denoted by 'Different standard deviations per stratum' in the numerical output. The largest value of s_{ij} is 1 for unknown sex in the summer, leading to a within-day correlation of 0.18. On the other extreme, for females in the summer, $s_{ij} = 0.436$, which leads to a within-day correlation of 0.54. Note that this correlation was called the intraclass correlation in Chapter 5.

As a final note, the p-values for the individual levels of sex (based on the t-statistic) are all larger than 0.05, but keep in mind that these p-values are with respect to the baseline level "Unknown". The fact that the likelihood ratio test showed that sex was significant (though only weakly, the p-value was 0.0196), means that males and females are having a different effect. Just change the baseline of the variable fSex to verify this.

7.6.5 Why All the Fuss?

You may wonder what the benefit is of the mixed modelling approach. Let us compare the optimal mixed effects model with the other models. Recall that the linear regression model in Section 7.6.3 gave us a p-value of 0.009 for Sex. That is rather a different p-value compared to the 0.0196 from the mixed model. Ok, you can argue that the linear regression model contained various non-significant terms.

No problem; let us drop them and refit the linear regression model with only Sex as explanatory variable.

```
> M2.lm <- lm(IntakeRate ~ fSex, data = Limosa)
> drop1(M2.lm, test = "F")

Single term deletions
Model: IntakeRate ~ fSex
       Df Sum of Sq     RSS     AIC F value    Pr(F)
<none>               2.80 -884.38
fSex    2      0.15    2.96 -877.56   5.475 0.004829
```

Hence, in the linear regression model in which we only use Sex, this term has a *p*-value of 0.0048. You may argue that you should not compare the linear regression with the linear mixed model as the linear regression model ignores the heterogeneity. Ok, let us fit a model that allows for heterogeneity, but without the random effect and obtain a *p*-value for sex.

```
> M5A.gls <- gls(IntakeRate ~ fSex, data = Limosa,
        weights = varIdent(form =~ 1 | fSex * fPeriod),
        method = "ML")
> M5B.gls <- gls(IntakeRate ~ 1, data = Limosa,
        weights = varIdent(form =~ 1 | fSex * fPeriod),
        method = "ML")
> anova(M5A.gls, M5B.gls)

        Model df       AIC       BIC   logLik   Test  L.Ratio p-value
M5A.gls     1 10 -321.8643 -288.5371 170.9322
M5B.gls     2  8 -311.9607 -285.2989 163.9803 1 vs 2 13.90364   0.001
```

The analysis of variance compares a model with sex and without sex. Both have the `varIdent` variance structure, but not the random intercept. We are still let to believe that sex is highly significant. What this means is that as soon as we include the random intercept, we allow for correlation between observations on the same day. For some sex–period combinations, this correlation can be as high as 0.54. Ignoring this correlation means that we end up with a *p*-value of 0.001. Taking it into account gives a *p*-value of 0.0196. The difference is a factor of 20. This example shows the danger of ignoring temporal correlation, something which happens in many scientific papers on ecology.

In case you enjoyed this analysis, try fitting the correlation structure with the compound symmetry correlation directly as an exercise. With this we mean that you can also use the `correlation = corCompSymm()` instead of random effects. And a more complicated approach would be to use any of the spatial correlation functions.