

SISMID Spatial Statistics in Epidemiology and Public Health

2016 R Notes: Infectious Disease Data

Jon Wakefield

Departments of Statistics, University of Washington

2016-07-03

Lower Saxony Measles Data

As a first example of the Held et al. (2005) approach we examine data on measles considered in this paper (actually an updated version).

These data are in the `surveillance` package.

The data consist of weekly measles counts over 2001 and 2002, for each of 17 administrative district Weser-Ems region of Lower Saxony, Germany.

Included in the dataset are a 17×17 matrix of 0/1 entries indicating which areas share a common boundary.

There is also the population that is contained in each area, and various other data including vaccination information.

Lower Saxony Measles: Reading in the Data

`data("measlesWeserEms")` is of `sts` class.

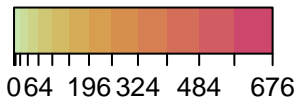
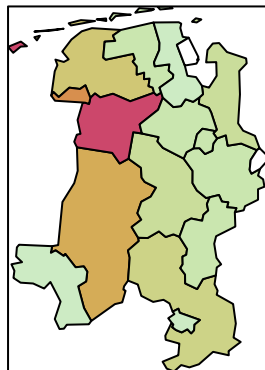
The data object also contains a map of the region, as a `SpatialPolygonsDataFrame`

```
library(surveillance)
data("measlesWeserEms")
```

Map of total counts

```
plot(measlesWeserEms, type = observed ~ unit)
```

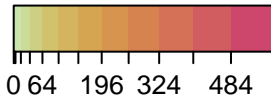
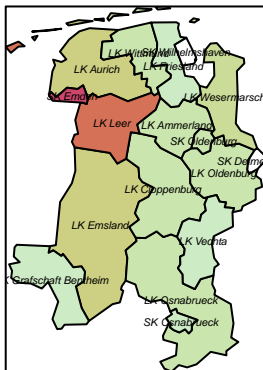
2001/1 – 2002/52



Map of total counts with labels

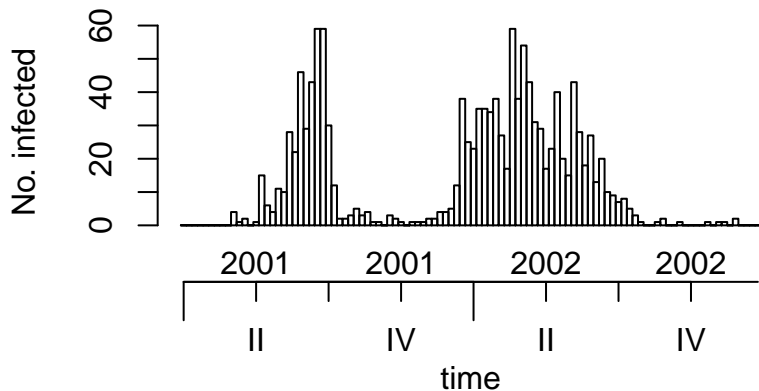
```
plot(measlesWeserEms, type = observed ~ unit, population = measl  
     labels = list(labels = "GEN", cex = 0.3, font = 3))
```

2001/1 – 2002/52



Time series of all data

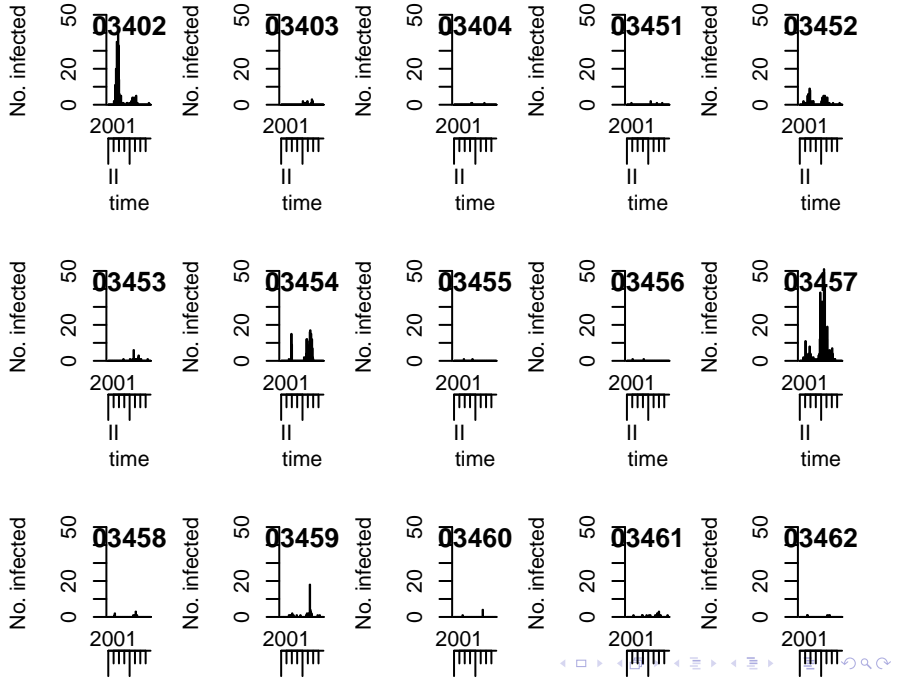
```
plot(measlesWeserEms, type = observed ~ time,  
     legend.opts = NULL)
```



Time series with areas of all zeroes excluded

Two areas contain all zeroes

```
measlesWeserEms15 <- measlesWeserEms[, colSums(observed(measlesW  
0)]  
plot(measlesWeserEms15)
```



Animation

The commands

```
library(gridExtra)
```

```
animate(measlesWeserEms)
```

produce an animation.

Try them yourself!

Model framework

We have a negative binomial model with

$$E[Y_{it}|\mu_{it}] = \mu_{it}$$

and

$$\text{var}(Y_{it}|\mu_{it}) = \mu_{it}(1 + \psi\mu_{it})$$

so that $\psi = 0$ corresponds to a Poisson model.

First model

We first fit the model

$$\mu_{it} = \underbrace{\lambda^{\text{AR}}}_{\exp(\alpha_0^{\text{AR}})} y_{i,t-1} + N_i \lambda_t^{\text{EN}},$$

with endemic term:

$$\log(\lambda_t^{\text{EN}}) = \alpha_0^{\text{EN}} + \alpha_1 t + \gamma \sin(\omega t) + \delta \cos(\omega t)$$

where

- ▶ λ^{AR} is the epidemic force,
- ▶ N_t are population counts in area i ,
- ▶ λ_t^{EN} is the endemic term,
- ▶ α_1 is a slope parameter describing the large scale endemic temporal trend,
- ▶ γ and δ are seasonal parameters and do not vary across areas, $\omega = (2\pi)/52$.

First model

NegBin1 gives a single overdispersion parameter, i.e., common to all areas.

```
f.end1 <- addSeason2formula(f = ~1 + t, S = 1, period = 52)
measles.mod1 <- list(ar = list(f = ~1), end = list(f = f.end1,
  offset = population(measlesWeserEms)), family = "NegBin1")
results.mod1 <- hhh4(measlesWeserEms, control = measles.mod1)
confint(results.mod1, parm = "overdisp")
##                2.5 %    97.5 %
## overdisp 1.608914 2.840724
```

This interval suggests a Poisson model, with $\psi = 0$ would

Results for model 1

```
summary(results.mod1)
##
## Call:
## hhh4(stsObj = measlesWeserEms, control = measles.mod1)
##
## Coefficients:
##              Estimate      Std. Error
## ar.1             -0.384089      0.122443
## end.1              0.397918      0.197906
## end.t              0.002941      0.003321
## end.sin(2 * pi * t/52)  1.027359      0.146790
## end.cos(2 * pi * t/52) -0.526122      0.130008
## overdisp           2.224819      0.314243
##
## Log-likelihood:   -991.34
## AIC:              1994.68
## BIC:              2027.49
##
## Number of units:      17
## Number of time points: 103
```

Results for model 1

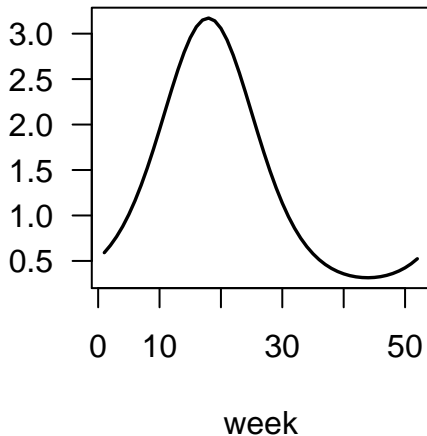
`idx2Exp = 1:4` gives exponentiated parameters and `amplitudeShift` transforms to the amplitude and phase shift.

```
summary(results.mod1, idx2Exp = 1:4, amplitudeShift = TRUE)
##
## Call:
## hhh4(stsObj = measlesWeserEms, control = measles.mod1)
##
## Coefficients:
##
##              Estimate      Std. Error
## exp(ar.1)          0.681071      0.083392
## exp(end.1)          1.488722      0.294627
## exp(end.t)          1.002945      0.003331
## end.A(2 * pi * t/52)  1.154241      0.147789
## end.s(2 * pi * t/52) -0.473290      0.103303
## overdisp            2.224819      0.314243
##
## Log-likelihood:    -991.34
## AIC:                1994.68
## BIC:                2027.49
##
## Number of units:      17
## Number of time points: 103
```

Plot of endemic component

Plot for the 7th area:

```
plot(results.mod1, type = "season", components = "end",  
      main = "")
```



A second model

We now add a neighborhood term, with simple binary weights given by

$$\mu_{it} = \underbrace{\lambda^{\text{AR}}}_{\exp(\alpha_0^{\text{AR}})} y_{i,t-1} + \underbrace{\lambda^{\text{NE}}}_{\exp(\alpha_0^{\text{NE}})} \sum_{j \in \text{ne}(i)} y_{j,t-1} + N_{it} \lambda_{it}^{\text{EN}},$$

with endemic term:

$$\log(\lambda_{it}^{\text{EN}}) = \alpha_0^{\text{EN}} + \alpha_1 t + \gamma \sin(\omega t) + \delta \cos(\omega t)$$

where

- ▶ λ^{AR} is the epidemic force,
- ▶ λ^{NE} is the neighborhood effect,
- ▶ N_{it} are (possibly standardized) population counts in area i at time t ,
- ▶ λ_{it}^{EN} is the endemic term,
- ▶ γ and δ are seasonal parameters and do not vary across areas, $\omega = (2\pi)/52$.

Lower Saxony Measles Data

The neighbourhood matrix is a distance matrix (neighbourhood orders)

wji is a standardized weight

```
wji <- neighbourhood(measlesWeserEms)/rowSums(neighbourhood(measlesWeserEms))
measles.mod2 <- list(ar = list(f = ~1), ne = list(f = ~1,
  weights = wji), end = list(f = f.end1, offset = population(measlesWeserEms)),
  family = "NegBin1")
results.mod2 <- hhh4(measlesWeserEms, control = measles.mod2)
```

Lower Saxony Measles Data

```
summary(results.mod2)
##
## Call:
## hhh4(stsObj = measlesWeserEms, control = measles.mod2)
##
## Coefficients:
##              Estimate      Std. Error
## ar.1             -3.841e-01    1.224e-01
## ne.1             -2.094e+01    3.864e+03
## end.1             3.979e-01    1.979e-01
## end.t             2.941e-03    3.321e-03
## end.sin(2 * pi * t/52) 1.027e+00    1.468e-01
## end.cos(2 * pi * t/52) -5.261e-01    1.300e-01
## overdisp          2.225e+00    3.142e-01
##
## Log-likelihood:   -991.34
## AIC:              1996.68
## BIC:              2034.96
##
## Number of units:      17
## Number of time points: 103
```

Lower Saxony Measles Data

Hence, the mean model is

$$\begin{aligned} E[Y_{i,t+1}|y_{it}, y_{jt}] &= \exp(-0.38)y_{it} + \exp(-25.05) \sum_{j \in \text{ne}(i)} y_{jt} \\ &+ \exp[0.40 + 0.0029 \times t + 1.0 \sin(\omega t) - 0.53 \cos(\omega t)] \end{aligned}$$

The variance is

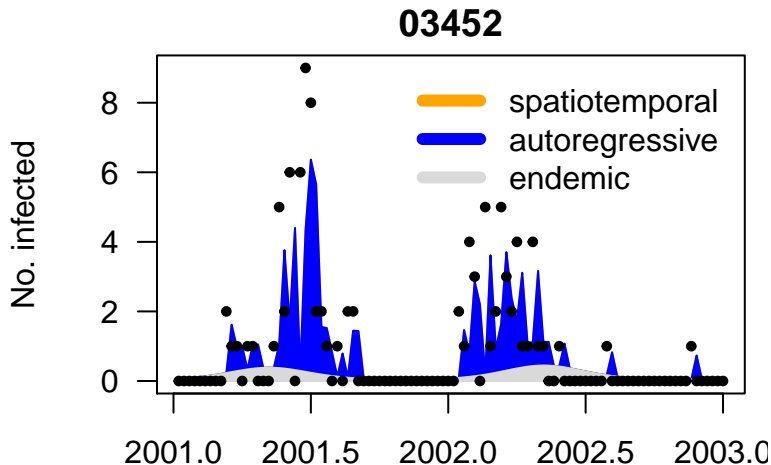
$$\text{var}(Y_{i,t+1}|y_{it}) = E[Y_{it}|y_{it}](1 + 2.23 \times E[Y_{it}|y_{it}]).$$

The fits are different in each of the areas due to the two epidemic components (AR and NE).

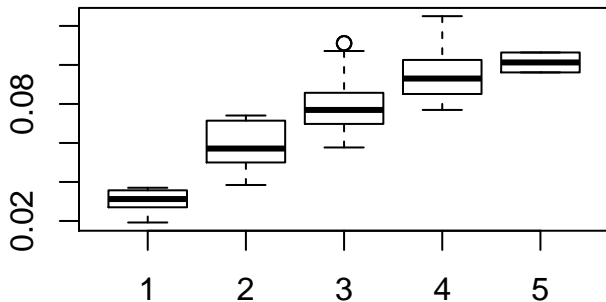
Lower Saxony Measles Data

Plot for the 7th area:

```
plot(results.mod2, unit = 7, legend = TRUE)
```



Neighbors summary



Lower Saxony Measles Data

We now extend the model to add area-specific intercepts (fixed effects) to the endemic component.

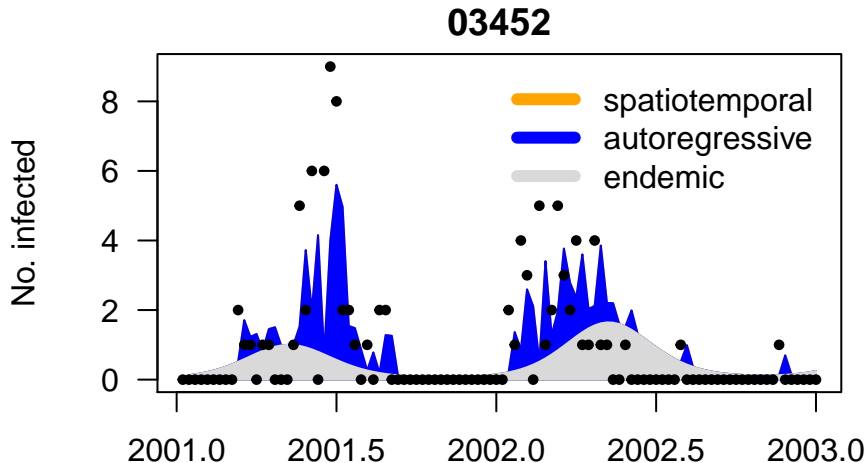
```
f.end2 <- addSeason2formula(f = ~-1 + fe(1, which = rep(TRUE,
  ncol(measlesWeserEms))) + t, S = 1, period = 52)
measles.mod3 <- list(ar = list(f = ~1), ne = list(f = ~1,
  weights = wji), end = list(f = f.end2, offset = population(m
  family = "NegBin1")
results.mod3 <- hhh4(measlesWeserEms, control = measles.mod3)
```

Lower Saxony Measles Data

```
cbind(results.mod3$coeff, results.mod3$se)
##                                [,1]          [,2]
## ar.1                        -0.580739100 1.289739e-01
## ne.1                       -23.379640570 7.734442e+03
## end.t                        0.009601913 3.476919e-03
## end.sin(2 * pi * t/52)      1.282329563 1.571930e-01
## end.cos(2 * pi * t/52)     -0.638093991 1.333572e-01
## end.1.03401                 -21.481281768 1.628901e+04
## end.1.03402                   2.261499246 3.210989e-01
## end.1.03403                 -0.278109315 3.987178e-01
## end.1.03404                 -1.717665726 6.406479e-01
## end.1.03405                 -21.543083205 1.592327e+04
## end.1.03451                 -0.829303555 5.242275e-01
## end.1.03452                   0.968107877 3.098019e-01
## end.1.03453                   0.124479801 3.683747e-01
## end.1.03454                   0.255424890 3.091657e-01
## end.1.03455                 -1.646575261 7.590931e-01
## end.1.03456                 -1.918957639 7.590523e-01
## end.1.03457                   1.778743964 2.957993e-01
## end.1.03458                 -0.577112364 4.694557e-01
## end.1.03459                 -0.406827611 3.602242e-01
## end.1.03460                 -1.056395570 5.294573e-01
## end.1.03461                   0.514041619 3.795217e-01
## end.1.03462                 -0.406028743 5.731908e-01
## -log(overdisp)              -0.388969337 1.408852e-01
```

Lower Saxony Measles Data

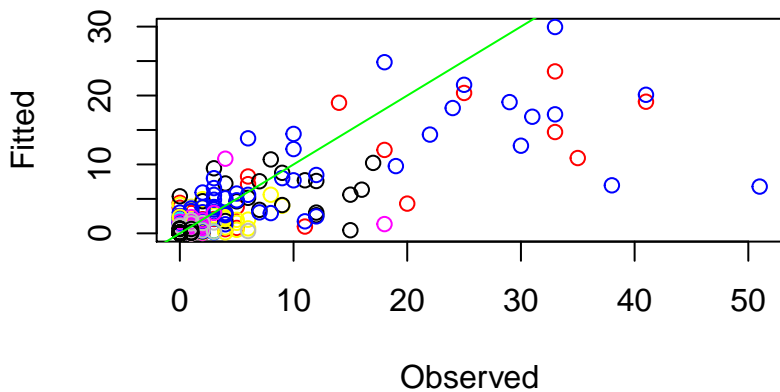
```
plot(results.mod3, unit = 7, legend = TRUE)
```



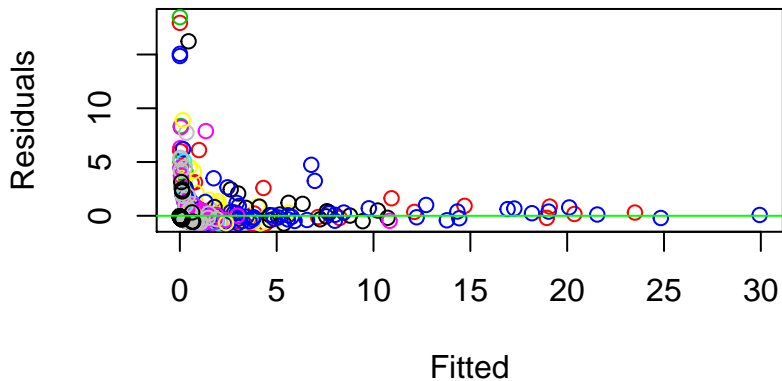
Residuals

```
y <- measlesWeserEms@observed[2:104, ] # Notice no first
time <- matrix(rep(measlesWeserEms@epoch, 17), nrow = 104,
               ncol = 17)
time <- time[-1, ]
#
mu3 <- fitted(results.mod3)
res3 <- (y - mu3)/sqrt(mu3 * (1 + 1.74 * mu3))
par(mfrow = c(1, 3))
plot(mu3 ~ y, xlab = "Observed", ylab = "Fitted", type = "n")
for (i in 1:17) points(mu3[, i] ~ y[, i], col = i)
abline(0, 1, col = "green")
plot(res3 ~ mu3, xlab = "Fitted", ylab = "Residuals",
      type = "n")
for (i in 1:17) points(res3[, i] ~ mu3[, i], col = i)
abline(0, 0, col = "green")
plot(res3 ~ time, xlab = "Time (week)", ylab = "Residuals",
      type = "n")
for (i in 1:17) points(res3[, i] ~ time[, i], col = i)
abline(0, 0, col = "green")
```

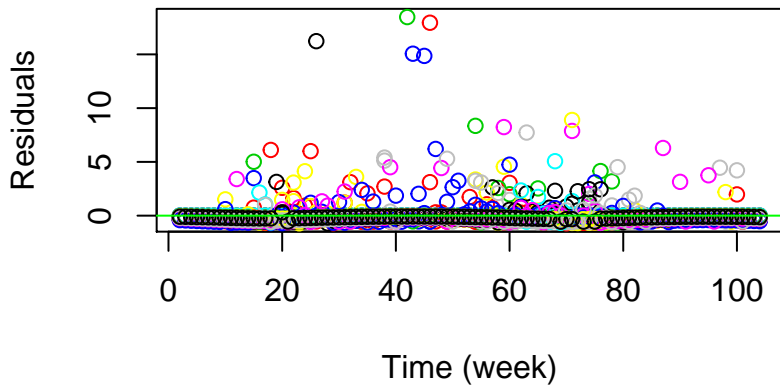
Residuals



Residuals



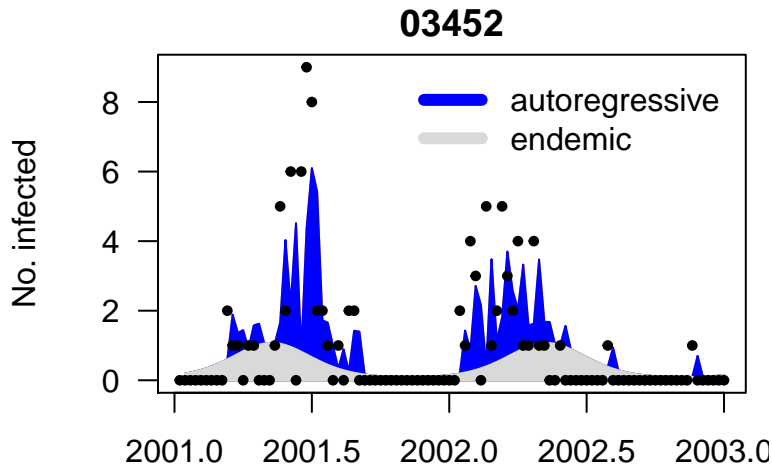
Residuals



Random effects model, no neighborhood

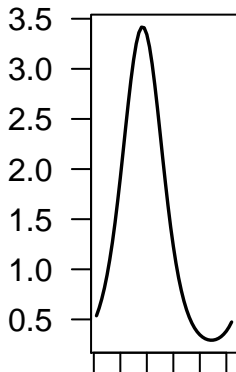
```
measles.mod4 <- list(ar = list(f = ~1), end = list(f = addSeason  
  ri(type = "iid"), S = 1, period = 52), offset = population(m  
  family = "NegBin1")  
measles.res4 <- hhh4(measlesWeserEms, measles.mod4)
```

fitted values for a single unit

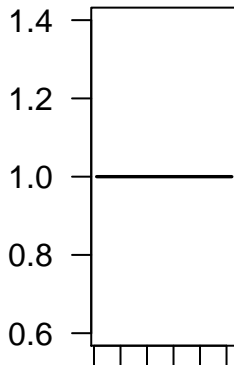


plot the multiplicative effect of seasonality

endemic compor **autoregressive com**



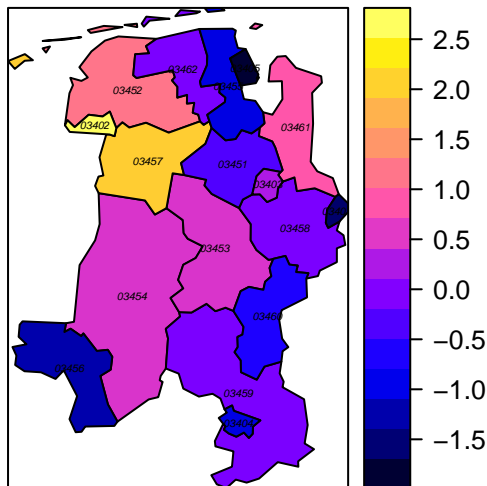
week



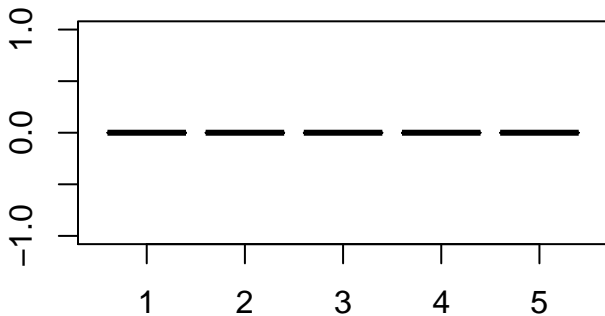
week

random intercepts of the endemic component

```
plot(measles.res4, type = "ri", component = "end",  
     labels = list(cex = 0.3, font = 3))
```



Neighbors summary



Random effects model

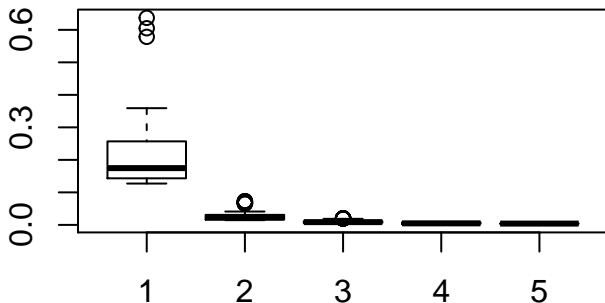
Without normalization, power-law weights are $w_{ji} = o_{ji}^{-d}$, where o_{ji} is the order of neighbourhood between regions i and j .

```
measles.mod5 <- list(end = list(addSeason2formula(~0 +  
  ri(type = "iid"), S = 1, period = 52)), ar = list(f = ~1),  
  ne = list(f = ~-1 + ri(type = "iid", corr = "all"),  
    weights = W_powerlaw(maxlag = 5)), family = "NegBin1")  
measles.res5 <- hhh4(measlesWeserEms, measles.mod5)
```

Neighborhood weights

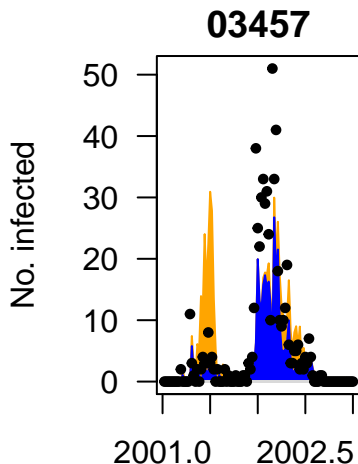
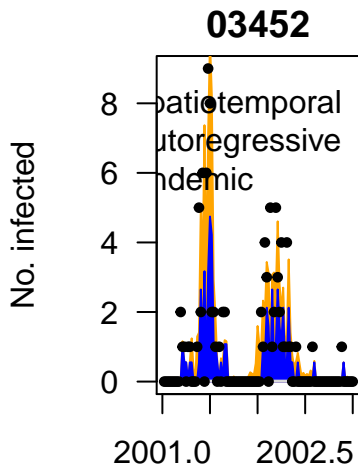
Plot the (estimated) neighbourhood weights (`newweights`) as a function of neighbourhood order (shortest-path distance between regions), i.e., $w_{ji} \sim o_{ji}$.

```
plot(measles.res5, type = "newweights")
```



random intercepts of the endemic component

```
plot(measles.res5, type = "fitted", units = c(7,  
12))
```



Random effects in all

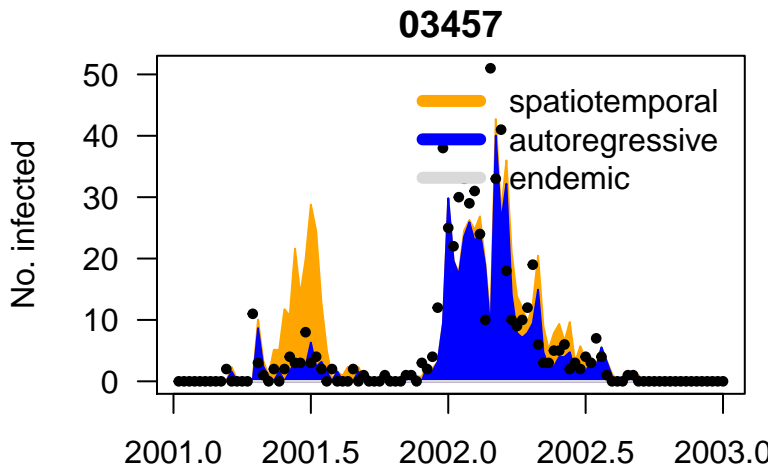
```
f.end <- addSeason2formula(~0 + ri(type = "iid", corr = "all"),  
  S = 1, period = 52)  
measles.mod6 <- list(end = list(f.end, offset = population(measles.mod6)),  
  ar = list(f = ~-1 + ri(type = "iid", corr = "all")),  
  ne = list(f = ~-1 + ri(type = "iid", corr = "all"),  
    weights = W_powerlaw(maxlag = 5)), family = "NegBin1")  
measles.res6 <- hhh4(measlesWeserEms, measles.mod6)
```

Results

```
summary(measles.res6)
##
## Call:
## hhh4(stsObj = measlesWeserEms, control = measles.mod6)
##
## Random effects:
##           Var      Corr
## ar.ri(iid) 1.5565
## ne.ri(iid) 2.5279 0.8981
##
## Fixed effects:
##           Estimate Std. Error
## ar.ri(iid)  -1.9596    0.3950
## ne.ri(iid)  -2.0552    0.4349
## end.1       -0.7821    0.2150
## newweights.d  3.0418    0.6195
## overdisp     1.3802    0.1987
##
## Penalized log-likelihood: -907.11
## Marginal log-likelihood:  -37.62
##
## Number of units:      17
## Number of time points: 103
```

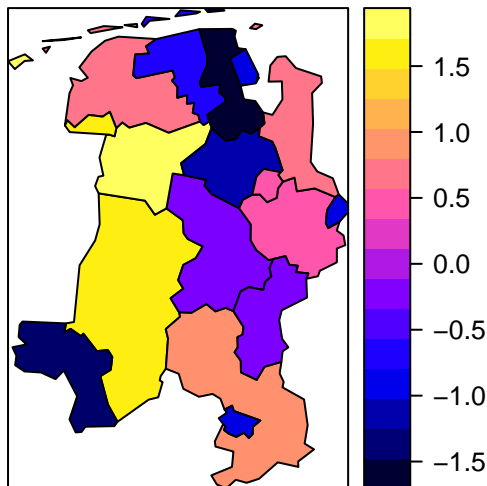
random intercepts of the endemic component

```
plot(measles.res6, type = "fitted", units = c(12))
```



Random intercepts of the endemic component

```
plot(measles.res6, type = "ri", component = "ar")
```



Random intercepts of the endemic component

```
plot(measles.res6, type = "ri", component = "ne")
```

