# Analysis of spatio-temporal data with R

V. Gómez-Rubio

Departamento de Matemáticas
Escuela de Ingenieros Industriales de Albacete
U. de Castilla-La Mancha

Imperial College London, 16 May 2014

based on work by Roger S. Bivand, Edzer Pebesma and H. Rue

# R classes for Time Series

## **xts** package

- Extends the **zoo** package
- Temporal index must be unique and ordered, and of a time-based class, i.e., every observation has a time-tag attached.
- Class **xts** allows for meta-data, so that it can be easily extended

## Some nice features

- Set of `apply` functions to compute results over time periods (monthly, yearly, ...)
- Location of endpoints by time, to obtain the observations in a period of time

# R clases for Space-Time Data

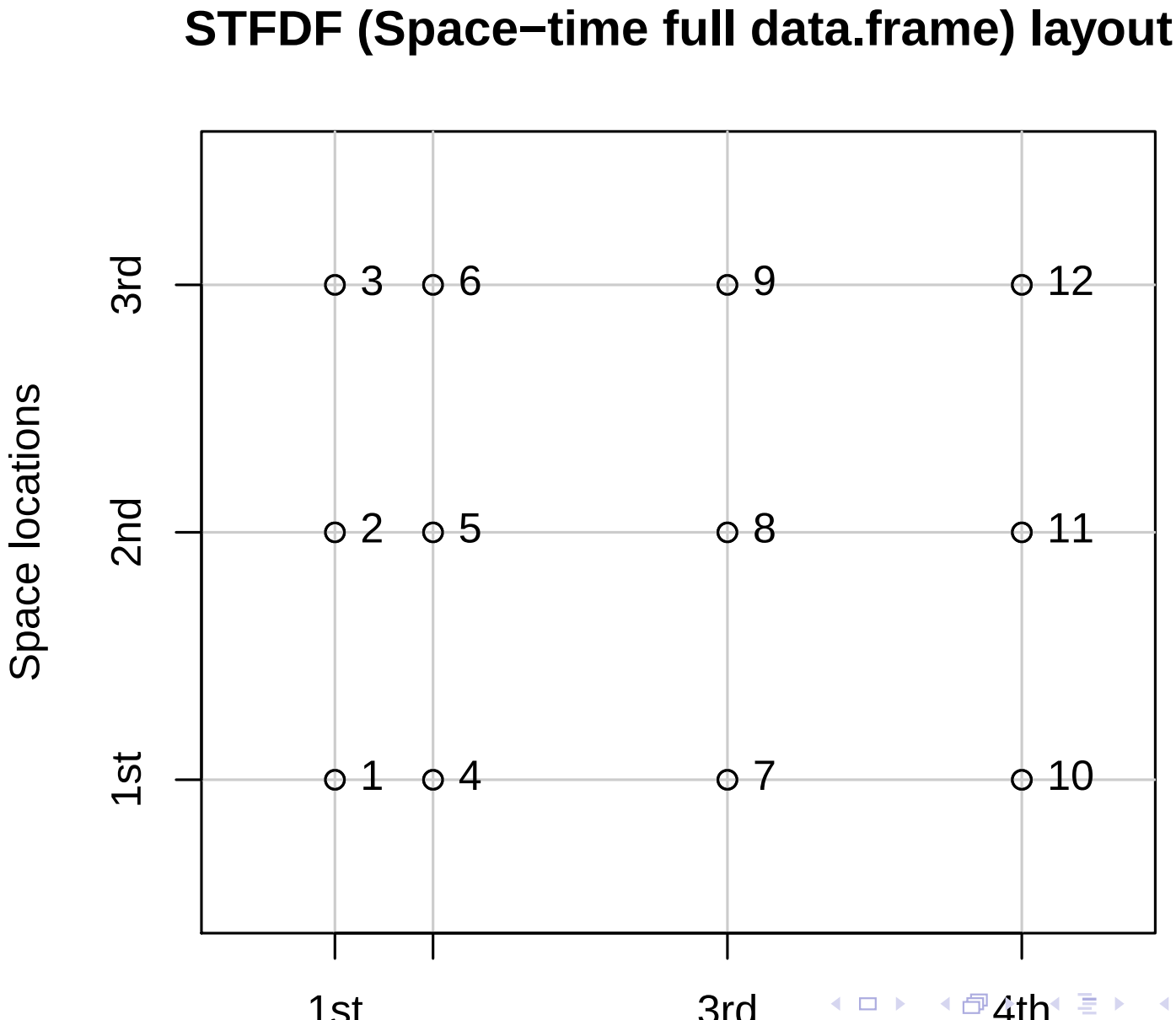
## **spacetime** package

- Extends spatial objects in the **sp** package using **xts** classes
- Differet types of space-time objects:
  - Full-space time grid
  - Sparse space-time grid
  - Irregular space-time grid

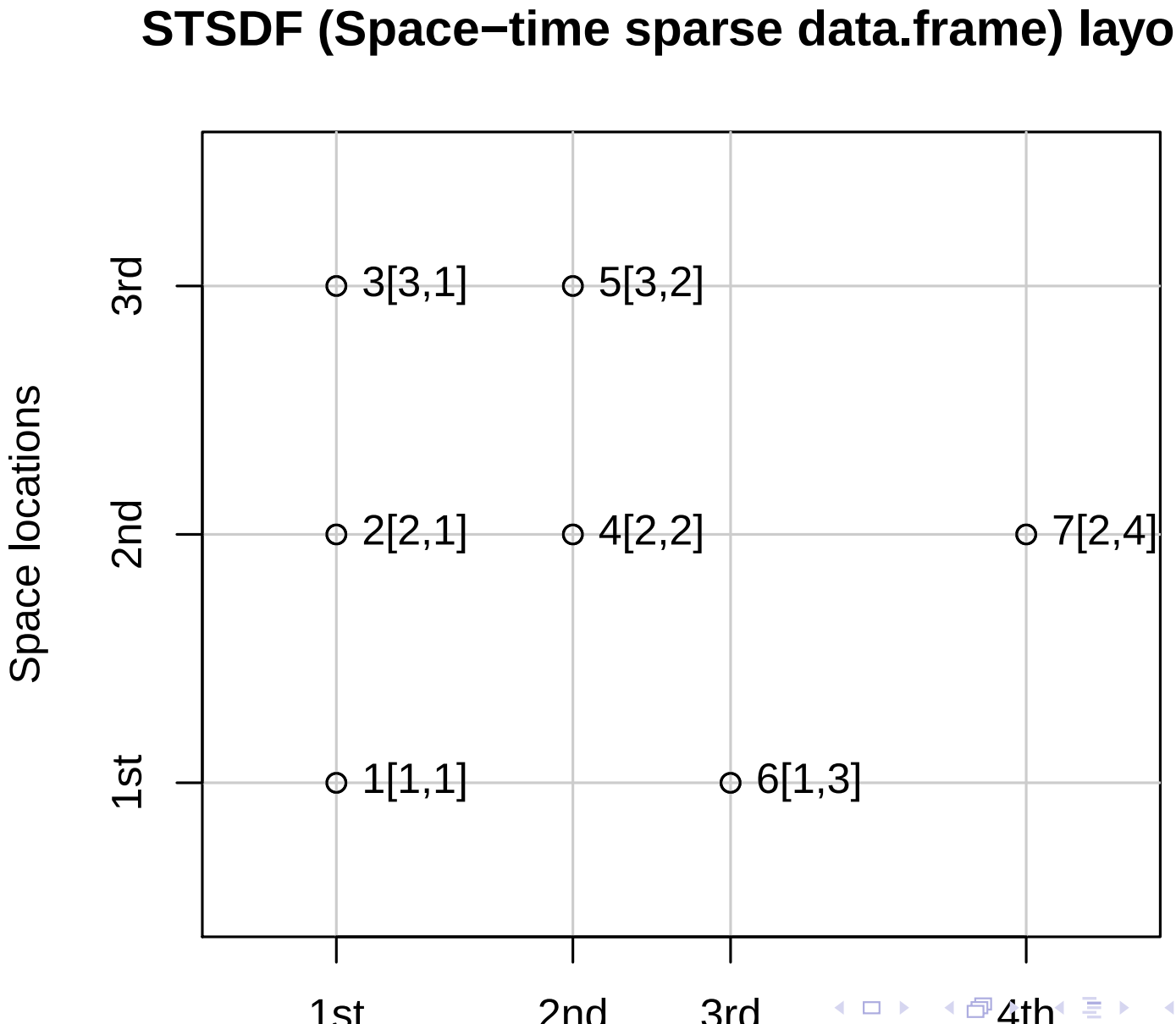
## Nice features

- Converts data into `sp`-objects when a particular time period is selected
- Temporal data can be used as in a `xts`-object
- Provides a `stplot` function similar to spplot for plotting space-time data

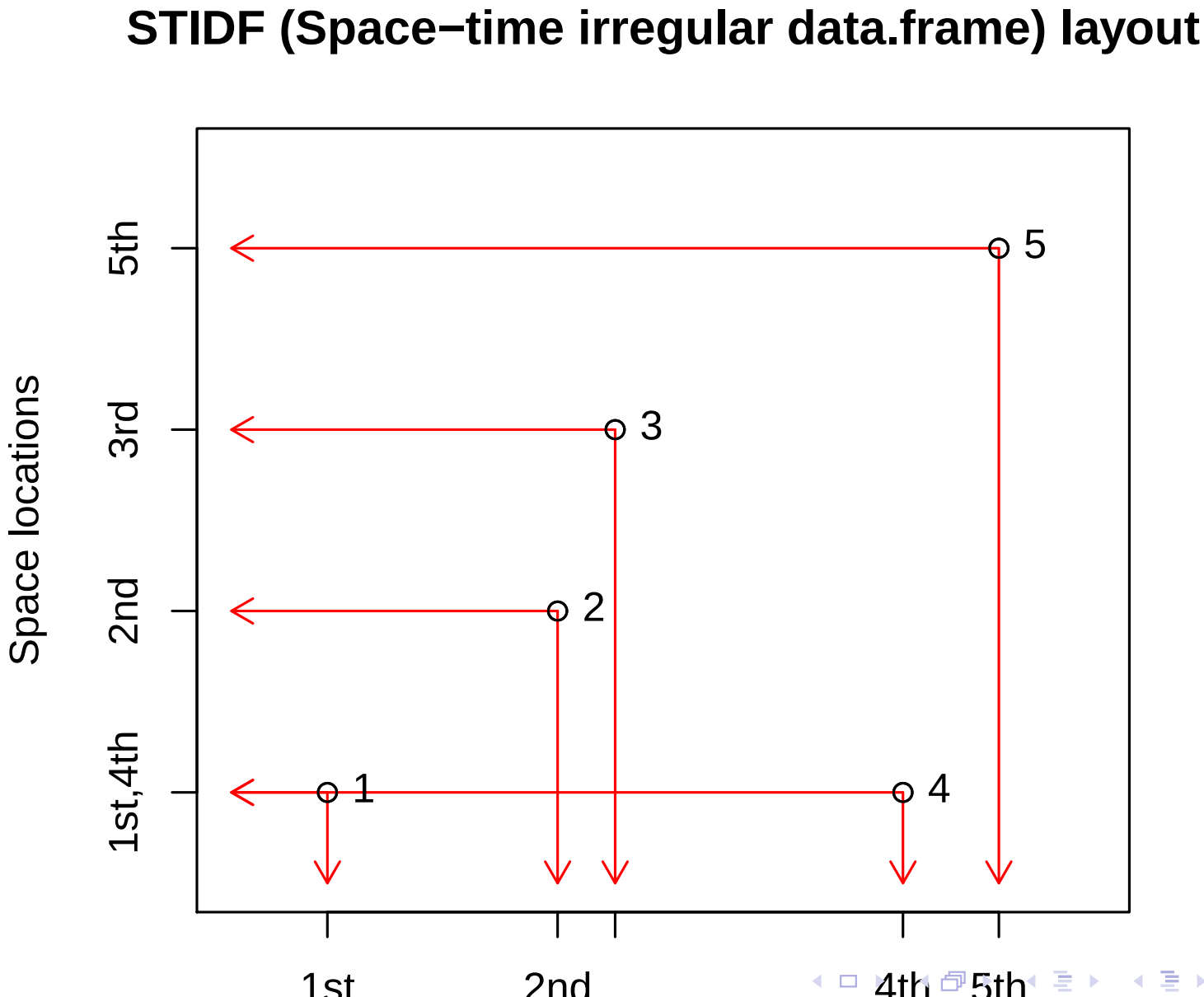
# Space-Time Layouts in **spacetime**



# Space-Time Layouts in **spacetime**



# Space-Time Layouts in **spacetime**



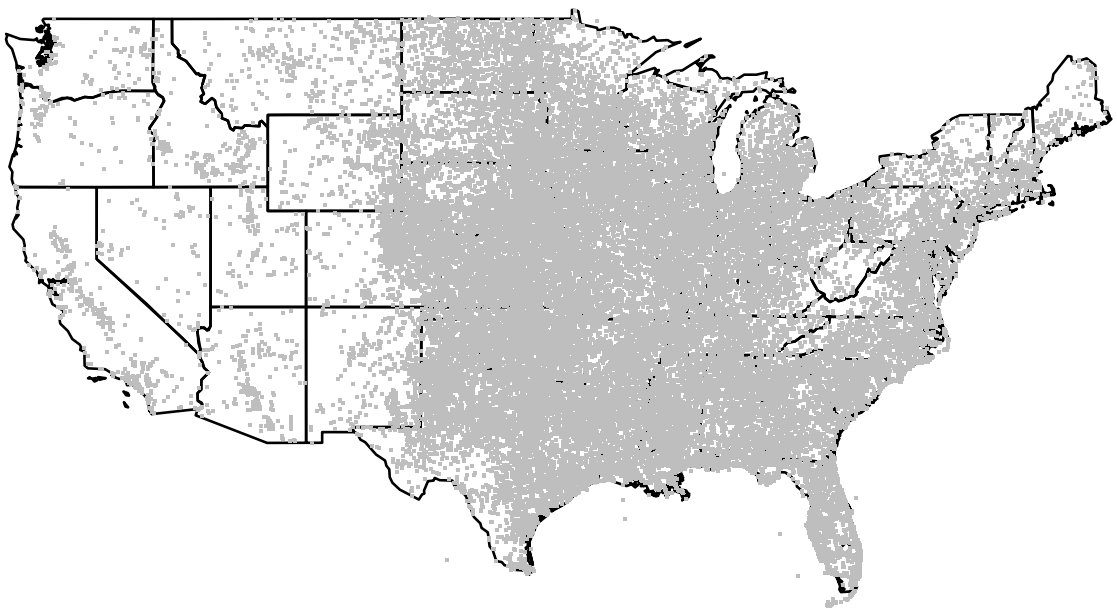
# Tornado Data 1950-2009

- We will use some Tornado data to show the analysis of point patterns
- These data have been obtained from the *Storm Prediction Center*[a]
- Tornado data from 1950 until 2009 are available
- In addition to the coordinates, we have a wealth of related information for each tornado

[a] http://www.spc.noaa.gov/wcm/index.html#data

# Location of Tornados

- 54123 tornados recorded in the 1950-2009 period
- 53217 occurred in the *Continental States* (without Alaska)
- The starting point of the tornado will be used
- Information about the ending point, trajectories and strength of the tornado is also available



# Inhomogeneous Poisson Processes (IPP)

## Intensity

The intensity $\lambda(x)$ provides the average number of events at location $x$. Events occur independently of each other. The total number of events in the study region $A$ is Poisson with mean

$$\int_A \lambda(x)dx$$

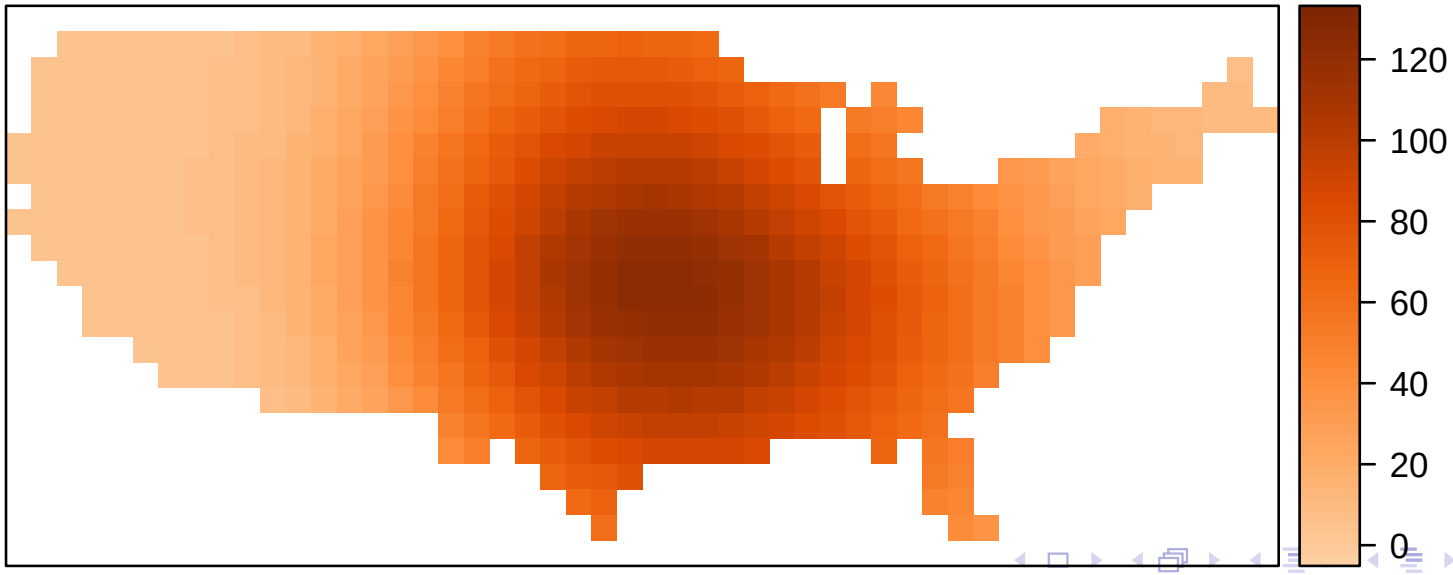
## Modelling

- Non-parametric methods (kernel smoothing, etc.)
- Parametric methods (log-intensity as polynomials, etc.)
- Semi-parametric

# Kernel Density Plots

- Kernel smoothing is a popular non-parametric way of estimating the intensity at a given point $x$:

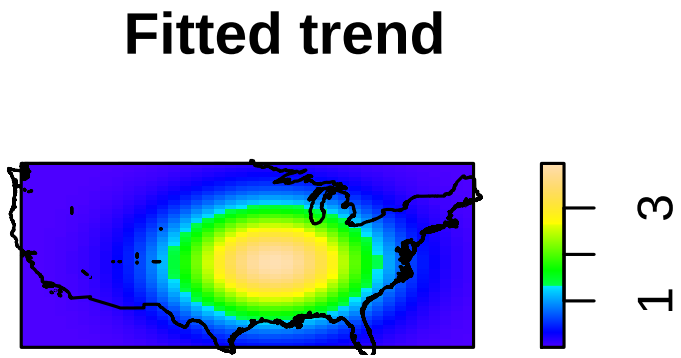
$$\hat{\lambda}(x) = \sum_{i=1}^{n} \frac{1}{h^2} \kappa\left(\frac{|x - x_i|}{h}\right); \kappa(\cdot) \text{ is a kernel function}$$



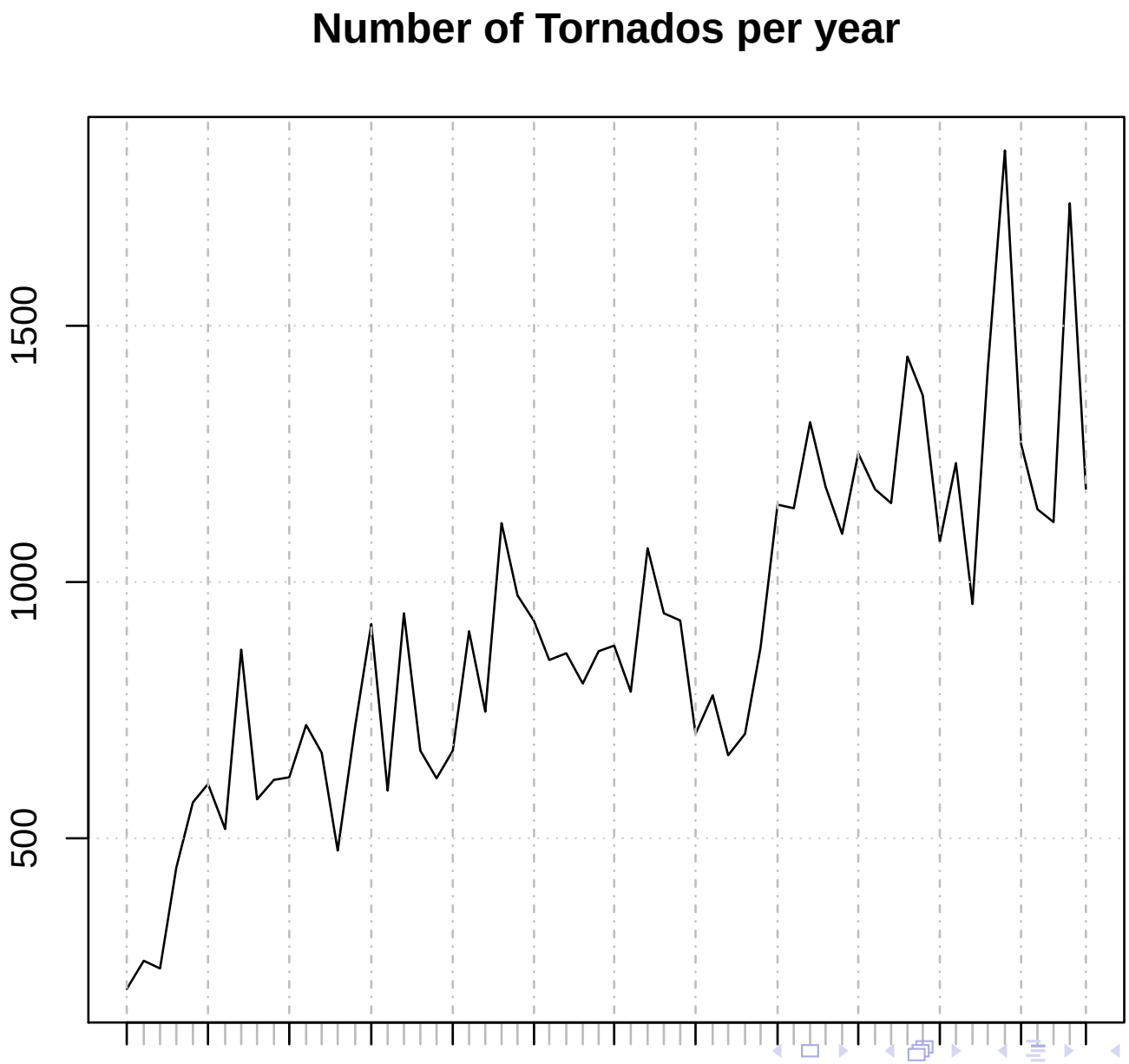
# Parametric Estimation

- Parametric models can be used to model the log-intensity. For example:

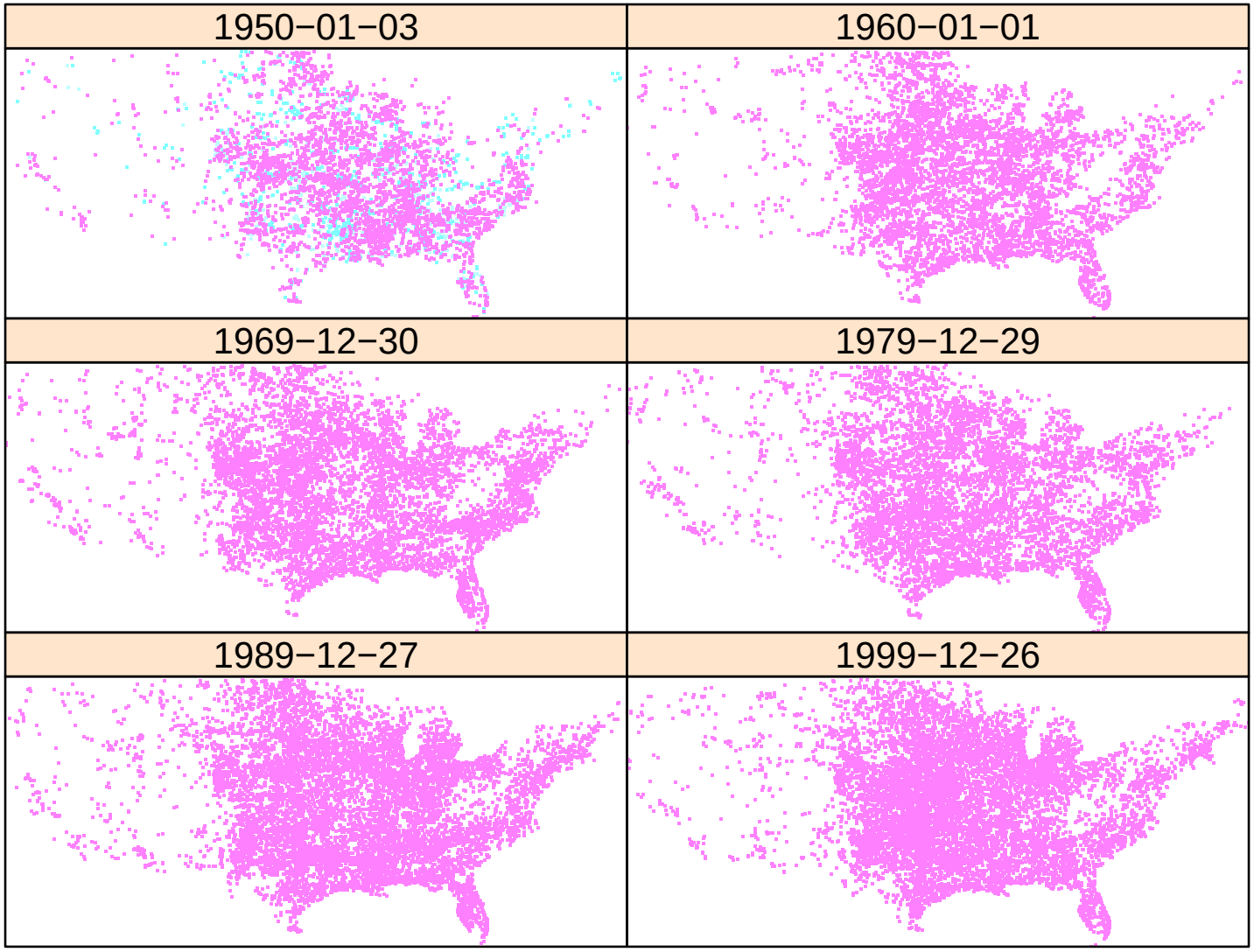
$$log(\lambda(x)) = 1 + x + y + x^2 + y^2$$



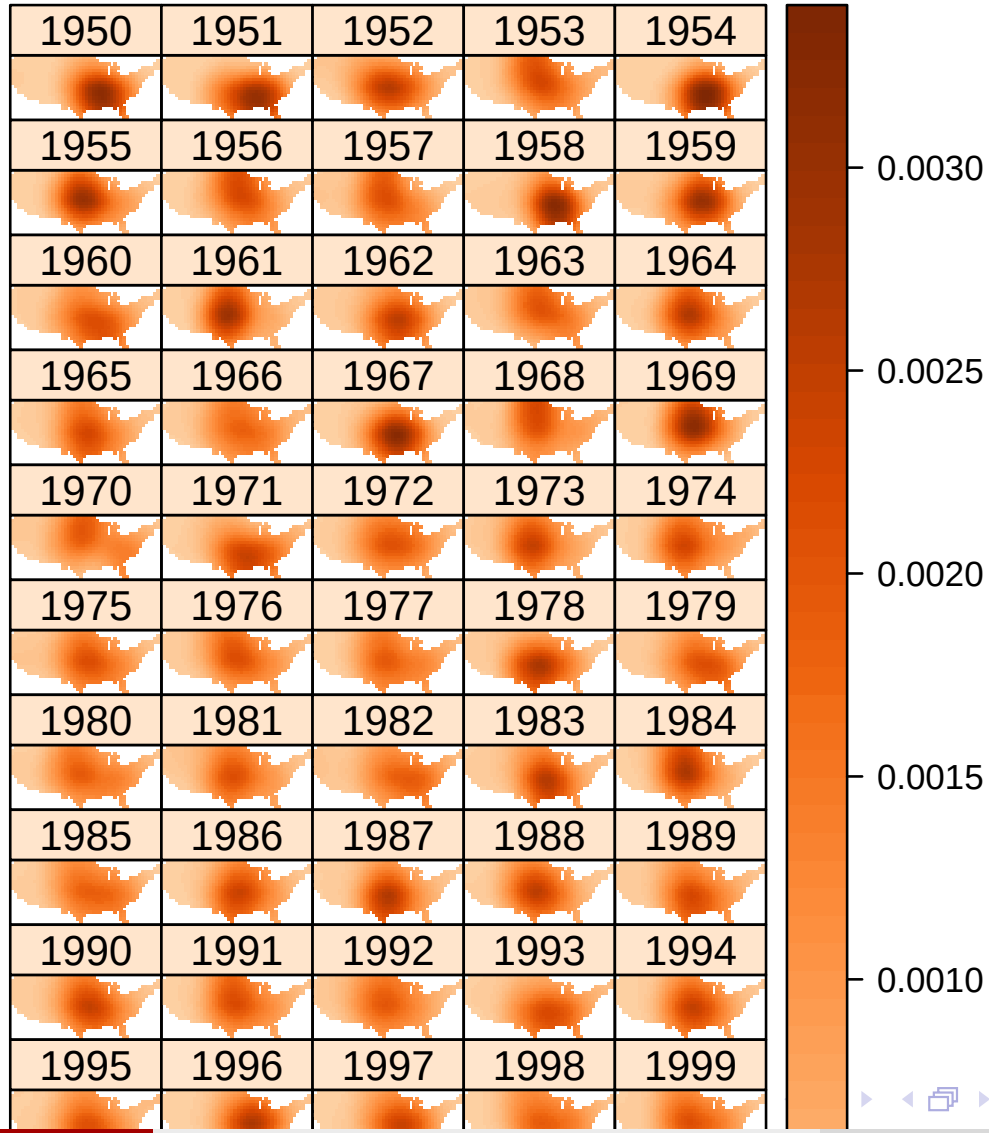
# Temporal Analysis



# Spatio-Temporal Analysis



# Spatio-Temporal Density Plots



# From Point Patterns to Lattice Data

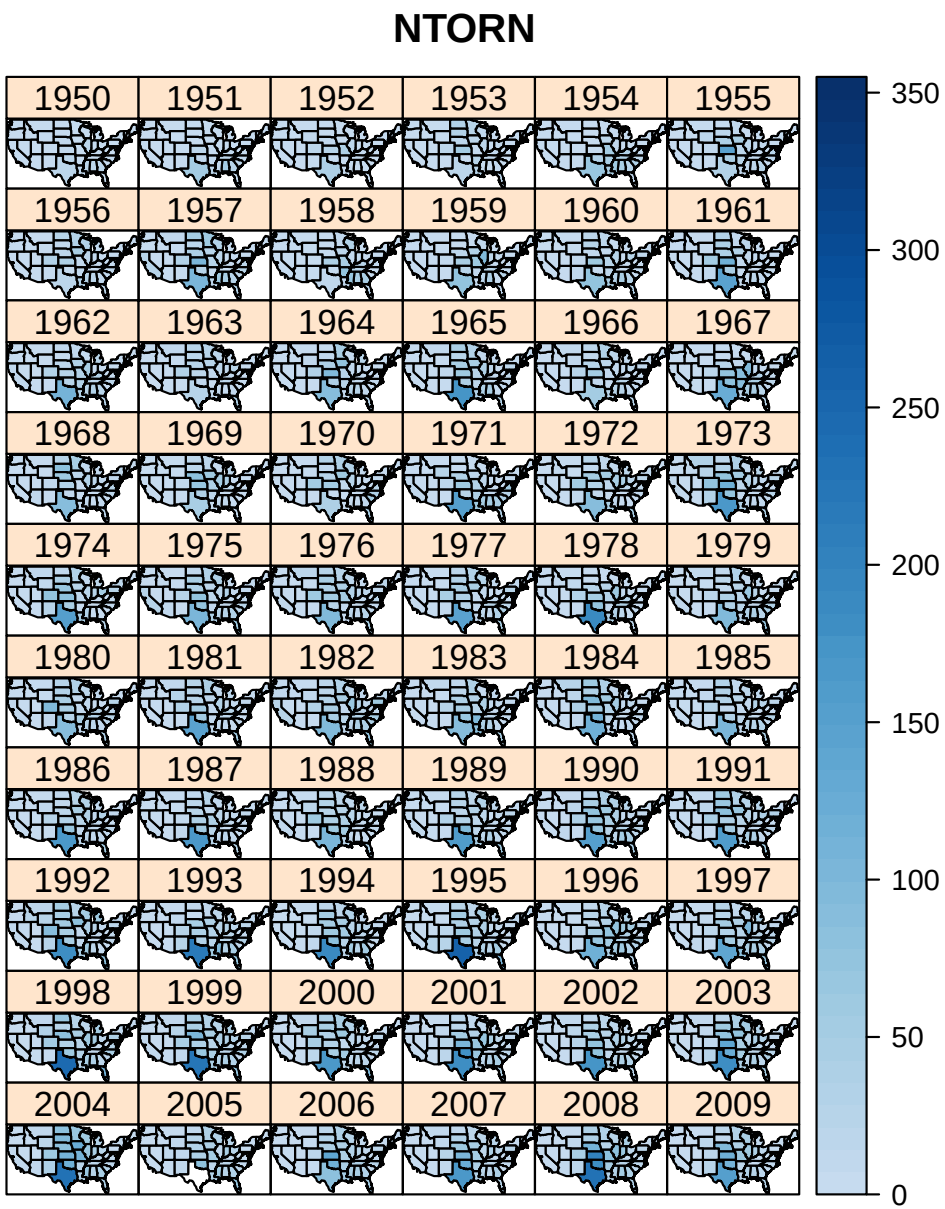
If events generated from an IPP are counted over a subregion $A_i$, the observed number of events is

$$O_i \sim Po(\mu_i);\ \mu_i = \int_{A_i} \lambda(x)dx$$

In our example, the "subregions" are State $\times$ Year:

$$O_{i,t} \sim Po(\mu_{i,t});\ \mu_{i,t} = \int_{Y_t} \int_{A_i} \lambda(x,t)dxdt$$

# Example: Number of Tornados per State



# Spatial Models for Lattice Data

- We will focus on the use of Generalized Linear Models
- In particular, Poisson models

$$O_i \sim Po(\mu_i) \quad \log(\mu_i) = \alpha + \beta x_i + u_i + v_i$$

- $u_i \sim N(0, \sigma_u^2)$ is a random effect that accounts for non-spatial variation
- $v_i \sim N(0, G)$ is a random effects that accounts for spatial variation, encoded in variance-covariance matrix $G$:
  - Spatially Autoregressive Specification (SAR models)

$$G = \sigma_v^2[(I - \rho W)^T (I - \rho W)]^{-1}$$

  - Conditionally Autoregressive Specification (CAR models)

$$G = \sigma_v^2[I - \rho W]^{-1}$$

# Spatio-Temporal Models for Lattice Data

## Separable models

No interaction between the spatial and temporal components:

$$\log(\mu_{i,t}) = \alpha + \beta X_{i,t} + u_i + v_i + w_t$$

Tempral random effects $w_t$ are often modelled as a random walk:

$$w_t = w_{t-1} + \varepsilon;\ \varepsilon \sim N(0, \sigma_w^2)$$

## Non-separable models

Some terms model space-time interaction:

$$\log(\mu_{i,t}) = \alpha + \beta X_{i,t} + u_i + v_i + w_t + z_{i,t}$$

$z_{i,t}$ is a spatio-temporal random effect.

# Example: Number of Tornados per State

- The number of tornados per State and year are modelled ($O_{i,t}$)
- As this may depend on the area of the State, it is used as a covariate
- Spatial correlation is considered using a CAR specification
- Temporal variation is modelled as second-order random walk
- The final model is:

$$O_{i,t} \sim Po(\mu_{i,t})$$

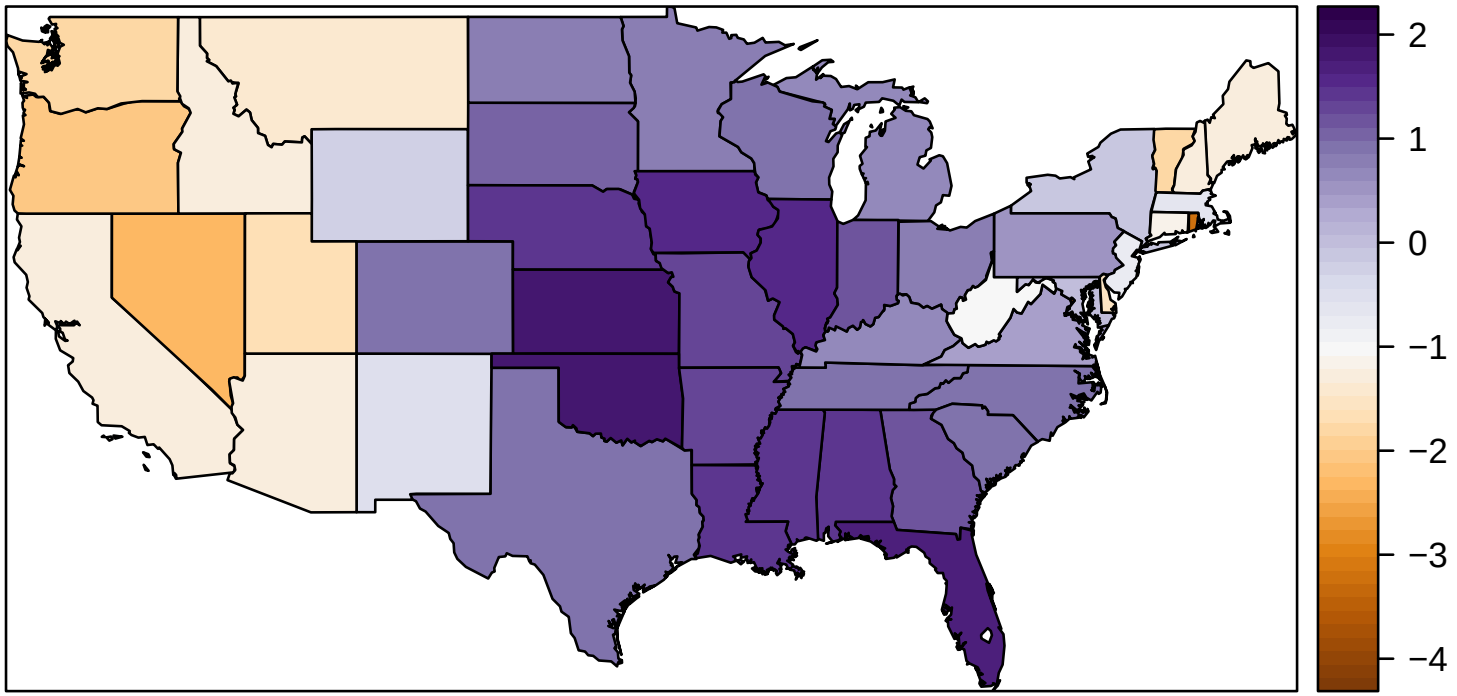
$$\log(\mu_{i,t}) = \alpha + \beta AREA_i + v_i + w_t$$

- But in the case a Bayesian model will be used, so we should include the specification of the prior distributions of the parameters in the model
- INLA can fit this model in a few seconds

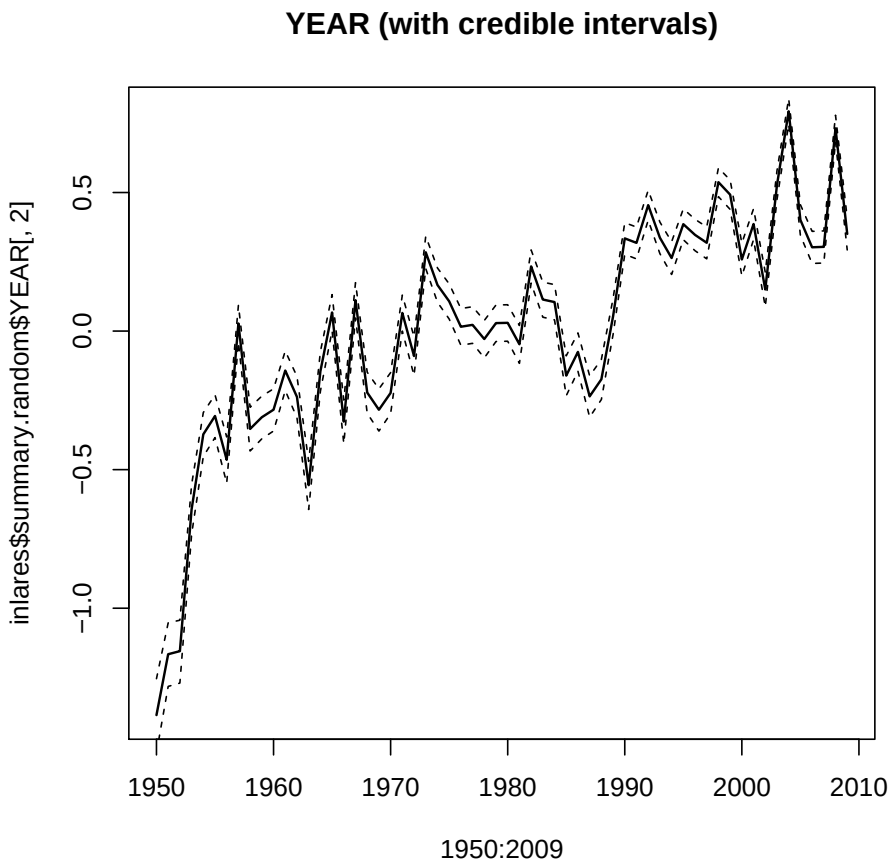
# Example: Bayesian Models with INLA

```
c("inla(formula = form, family = "Poisson", data = as.data.frame(stfdf), ",  "    control.predictor = list(comp

Time used:
 Pre-processing    Running inla Post-processing           Total
      3.9054101      10.1060719       0.5856271      14.5971091

Fixed effects:
                mean         sd 0.025quant   0.5quant 0.975quant          kld
(Intercept) 1.3477680 0.20635177 0.93963480 1.34814000 1.75334203 4.431948e-04
AREA        0.0386048 0.01232698 0.01430088 0.03859873 0.06292423 1.464159e-05

Random effects:
Name      Model          Max KLD
YEAR   RW2 model    0.00035
STATEID   Besags ICAR model    0.01123

Model hyperparameters:
                      mean    sd      0.025quant 0.5quant 0.975quant
Precision for YEAR     7.0674  1.4144  4.6634     6.9428  10.2018
Precision for STATEID  0.5458  0.1127  0.3542     0.5361   0.7950

Expected number of effective parameters(std dev): 105.47(0.5218)
Number of equivalent replicates : 27.87


Marginal Likelihood:  -15091.74
Warning: Interpret the marginal likelihood with care if the prior model is improper.
Posterior marginals for linear predictor and fitted values computed
```

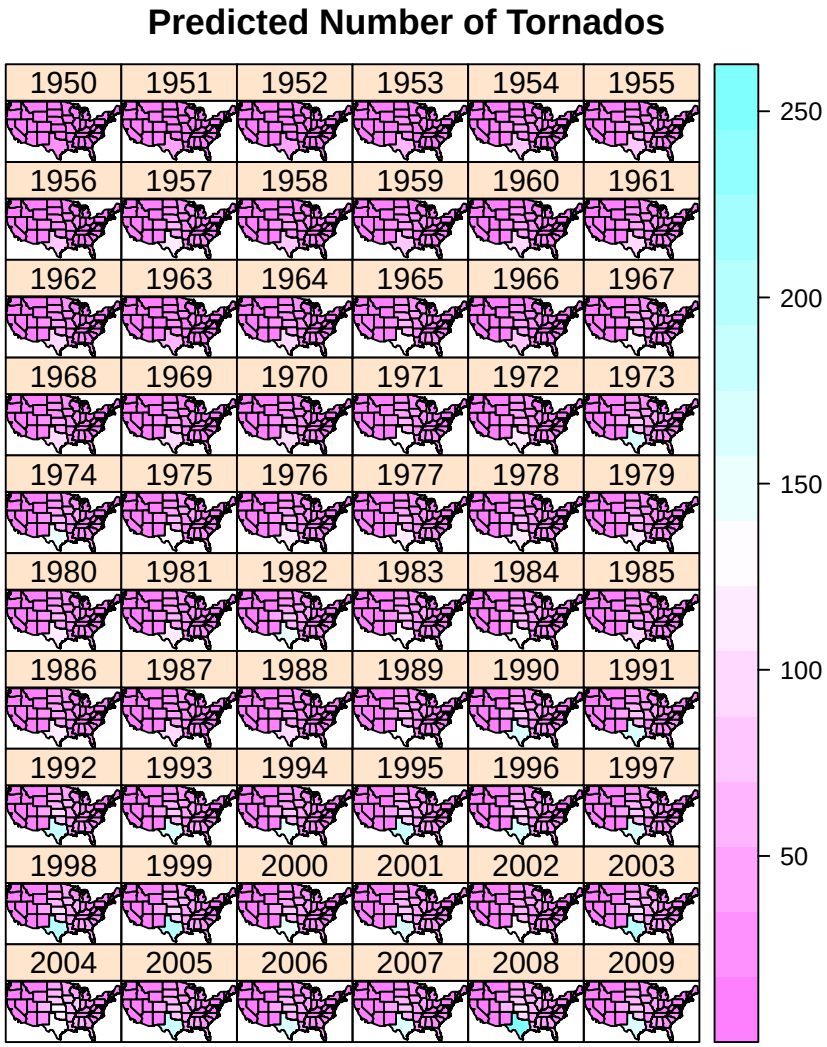
# Results: Spatial Random Effects



# Results: Temporal Random Effects



# Results: Predicted Number of Tornados



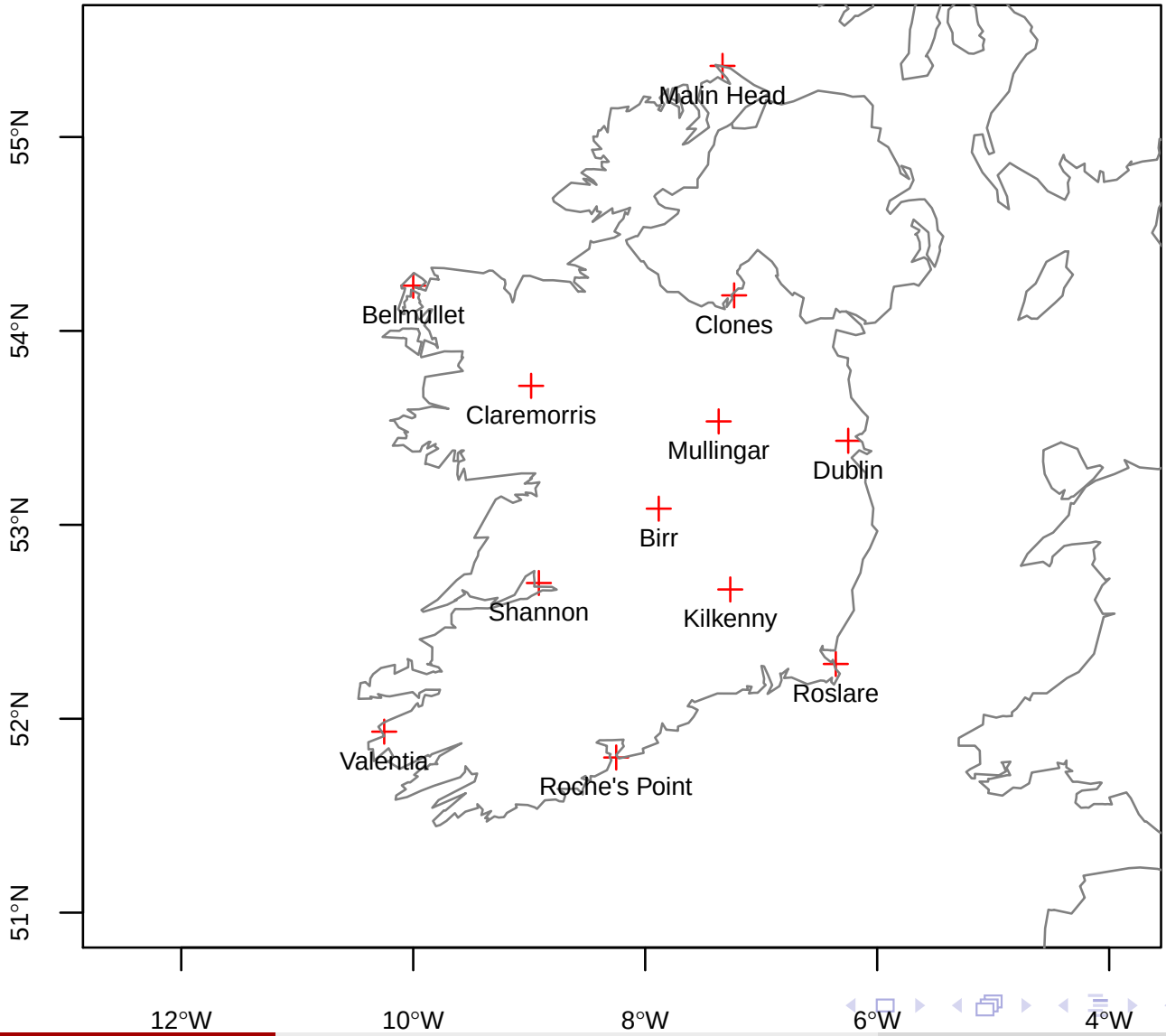
# Wind speed in Ireland

## Description

- Daily average wind speeds for 1961-1978 at 12 synoptic meteorological stations in the Republic of Ireland
- Wind speeds are in knots (1 knot = 0.5418 m/s)
- Complete analysis in Haslett and Raftery (1989)
- Example from the **spacetime** vignette

```
> library(sp)
> library(spacetime)
> library(gstat)
> data(wind)
> wind.loc$y = as.numeric(char2dms(as.character(wind.loc[["Latitude"]])))
> wind.loc$x = as.numeric(char2dms(as.character(wind.loc[["Longitude"]])))
> coordinates(wind.loc) = ~x+y
> proj4string(wind.loc) = "+proj=longlat +datum=WGS84"
```

# Wind speed in Ireland: Location of station



# Wind speed in Ireland

- Next we will recode time to an appropriate format
- The daily trend is removed (using a lowess regression)

```
> #Date transformation
> wind$time = ISOdate(wind$year+1900, wind$month, wind$day)
> wind$jday = as.numeric(format(wind$time, '%j'))
> stations = 4:15
> # knots -> m/s
> windsqrt = sqrt(0.5148 * as.matrix(wind[stations]))
> #Trend removal
> Jday = 1:366
> daymeans = sapply(split(windsqrt, wind$jday), mean)
> meanwind = lowess(daymeans ~ Jday, f = 0.1)$y[wind$jday]
> velocities = apply(windsqrt, 2, function(x) { x - meanwind })
```

# Wind speed in Ireland

- Match wind speeds to locations

```
> # order locations to order of columns in wind;
> # connect station names to location coordinates
> wind.loc = wind.loc[match(names(wind[4:15]), wind.loc$Code),]
> pts = coordinates(wind.loc[match(names(wind[4:15]), wind.loc$Code),])
> rownames(pts) = wind.loc$Station
> pts = SpatialPoints(pts)
> # convert to utm zone 29, to be able to do interpolation in
> # proper Euclidian (projected) space:
> proj4string(pts) = "+proj=longlat +datum=WGS84"
> library(rgdal)
> utm29 = CRS("+proj=utm +zone=29 +datum=WGS84")
> pts = spTransform(pts, utm29)
> # construct from space-wide table:
> wind.data = stConstruct(velocities,
+    space = list(values = 1:ncol(velocities)),
+          time = wind$time, SpatialObj = pts)
```

# Wind speed in Ireland

- Reproject coordinates of stations and map boundaries to UTM zone

```
> library(maptools)
> m = map2SpatialLines(
+         map("worldHires", xlim = c(-11,-5.4), ylim = c(51,55.5), plot=F))
> proj4string(m) = "+proj=longlat +datum=WGS84"
> m = spTransform(m, utm29)
> # setup grid
> grd = SpatialPixels(SpatialPoints(makegrid(m, n = 300)),
+         proj4string = proj4string(m))
> # select april 1961:
> wind.data = wind.data[, "1961-04"]
> # 10 prediction time points, evenly spread over this month:
> library(xts)
> n = 10
> tgrd = seq(min(index(wind.data)), max(index(wind.data)), length=n)
> pred.grd = STF(grd, tgrd)
>
>
```

# Wind speed in Ireland

## Covariance function

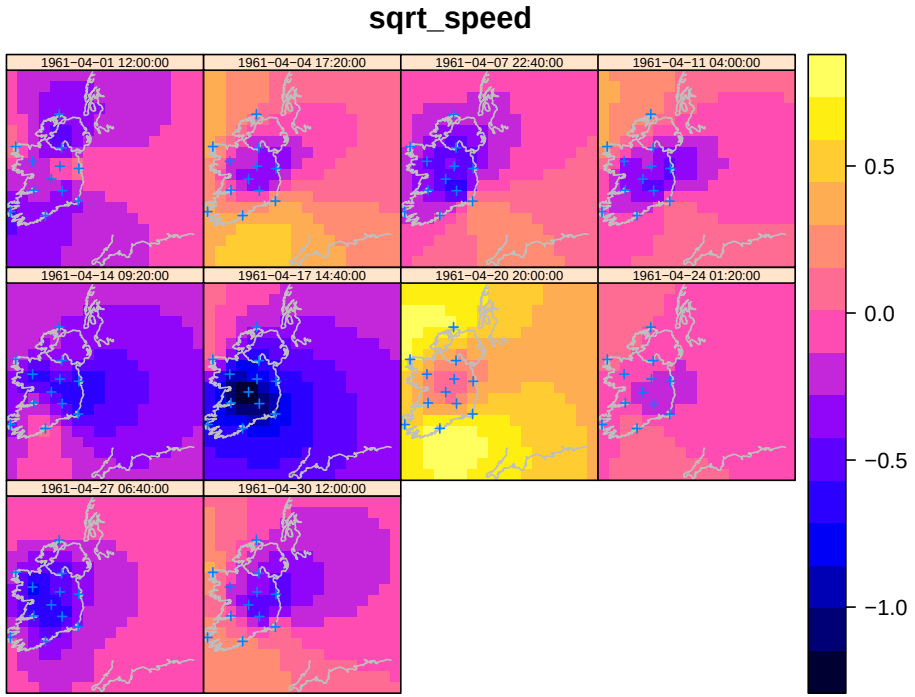
We will use a separable covariogram:

$$\gamma_{x,t}((x_i, t_i), (x_j, t_j)) = \gamma_x(x_i, x_j)\gamma_t(t_i, t_j) = \gamma_x(||x_i - x_j||)\gamma_t(|t_i - t_j|)$$

```
> # separable covariance model, exponential with ranges 750 km and 1.5 day:
> v = vgmST("separable", space = vgm(1, "Exp", 750000),
+    time = vgm(1, "Exp", 1.5 * 3600 * 24),
+          sill=0.6)
> wind.ST = krigeST(values ~ 1, wind.data, pred.grd, v)
> colnames(wind.ST@data) <- "sqrt_speed"
>
>
```

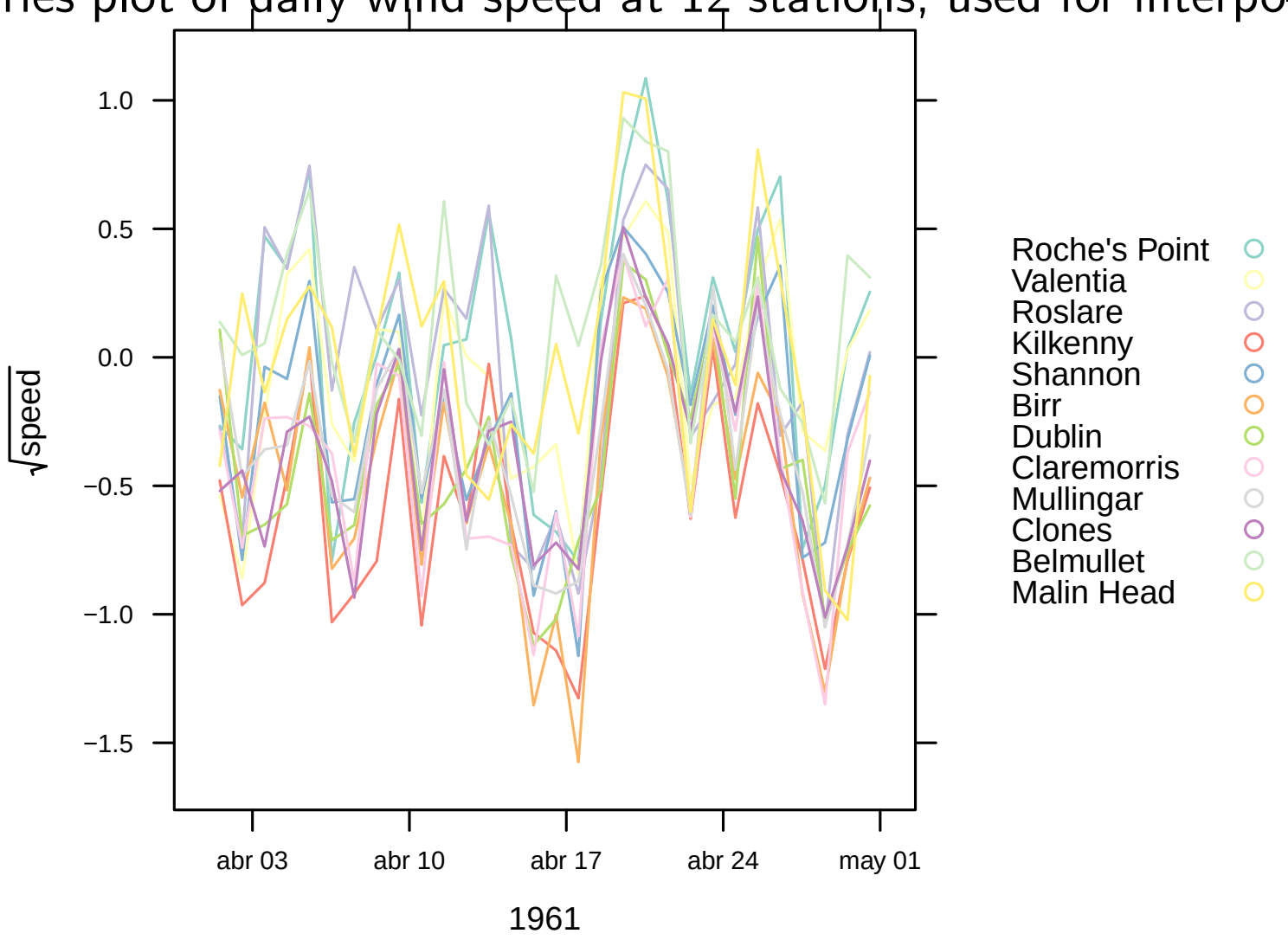
# Wind speed in Ireland

Space-time interpolations of wind (square root transformed, detrended)



# Wind speed in Ireland

Time series plot of daily wind speed at 12 stations, used for interpo- lation

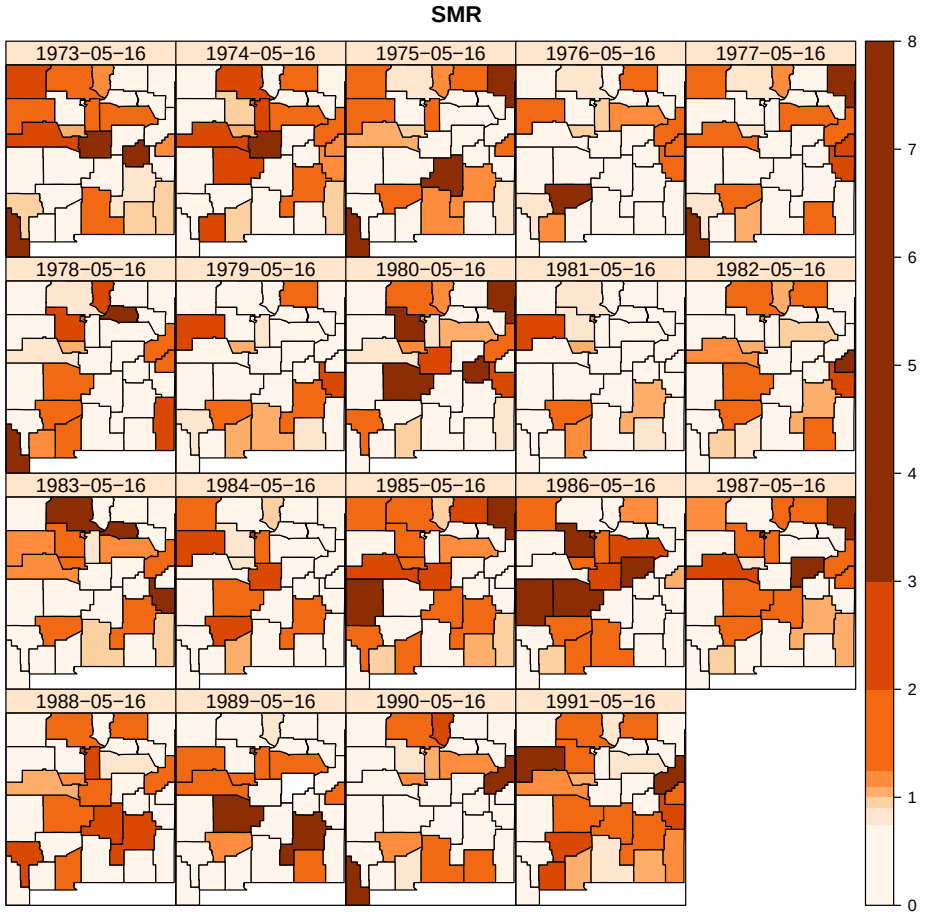


# Brain cancer in New Mexico

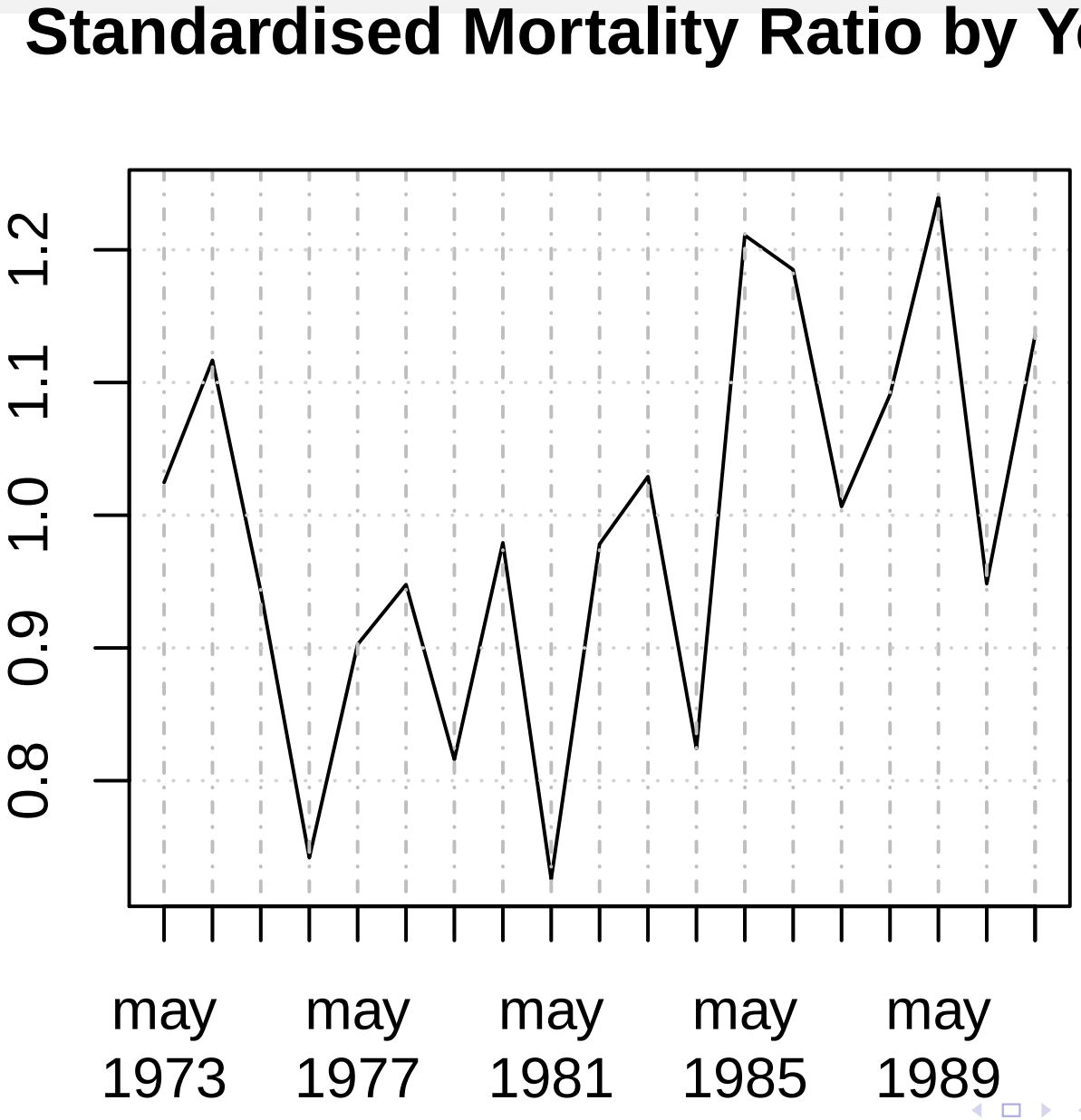
- Cases of brain cancer in New Mexico in the period 1973-1991
- Counts at the county level
- Data set obtained from the SatScan website
- Boundaries from the US Census Bureau
- Expected cases have bee computed using standardisation by age, race and sex
- Cibola county was merged with Valencia county
- As a covariate, we will use the (re-scaled) inverse distance to Los Alamos National Laboratory

```
> library(sp)
> library(spacetime)
> load("datasets/brain.RData")
> #Los Alamos National Laboratory (source Wikipedia)
> losalamos<-SpatialPoints(matrix(c(-106.298333, 35.881667), ncol=2))
> #Inverse distance to the laboratory
> nmf$IDLANL<-1/spDistsN1(coordinates(nmf), coordinates(losalamos), longlat
> nmf$IDLANLre<-nmf$IDLANL/mean(nmf$IDLANL) #Re-scale inverse distance
```

# Brain cancer in New Mexico



# Brain cancer in New Mexico: Temporal trend



# Brain cancer in New Mexico: Spatio-temporal model

- Inverse distance to LANL is used as a covariate
- Temporal trend $w_t$: first-order random walk
- Spatial term $v_i$: CAR structure

$$O_{i,t} \sim Po(\mu_{i,t})$$

$$\log(\mu_{i,t}) = \alpha + \beta IDLANL_i + v_i + w_t$$

$$v_i \sim CAR(\sigma_v^2)$$

$$w_t \sim RW(1, \sigma_w^2)$$

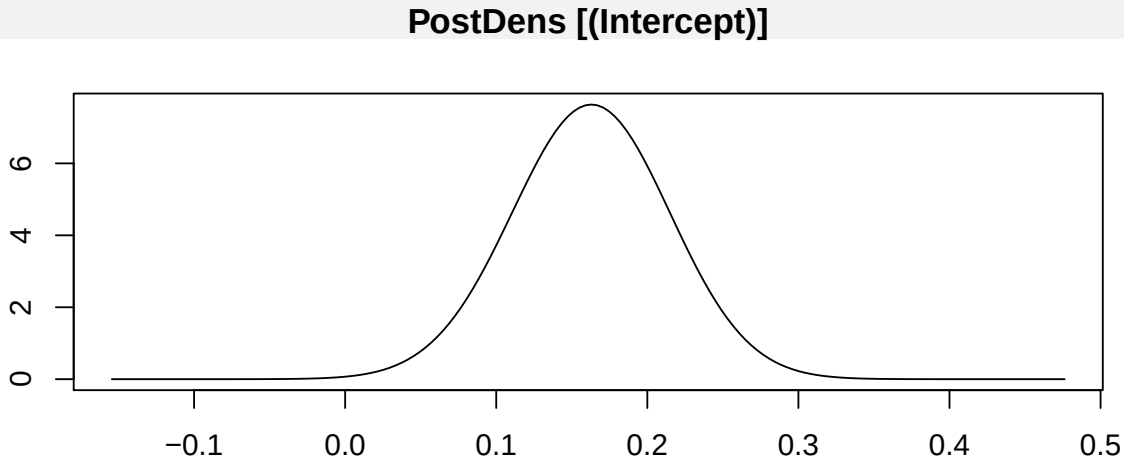
# Brain cancer in New Mexico: Fitting the model with INLA

```
> library(spdep)
> neib<-poly2nb(nmf)
> nb2INLA("results/nmf.adj", neib)
> #Fit spatio-temporal model with INLA
> library(INLA)
> #form<-NTORN~1+AREA+f(YEAR, model="rw2")+f(STATE, model="iid")+f(STATEID,
> form<-Observed~1+IDLANLre+f(Year, model="rw1")+f(ID, model="besag",
+    graph="results/nmf.adj")
> inlares<-inla(form, family="Poisson", data=as.data.frame(brainst),
+    E=Expected,
+ control.predictor=list(compute=TRUE),
+ #   quantiles=qnts,
+    control.results=list(return.marginals.predictor=TRUE)
+ )
```

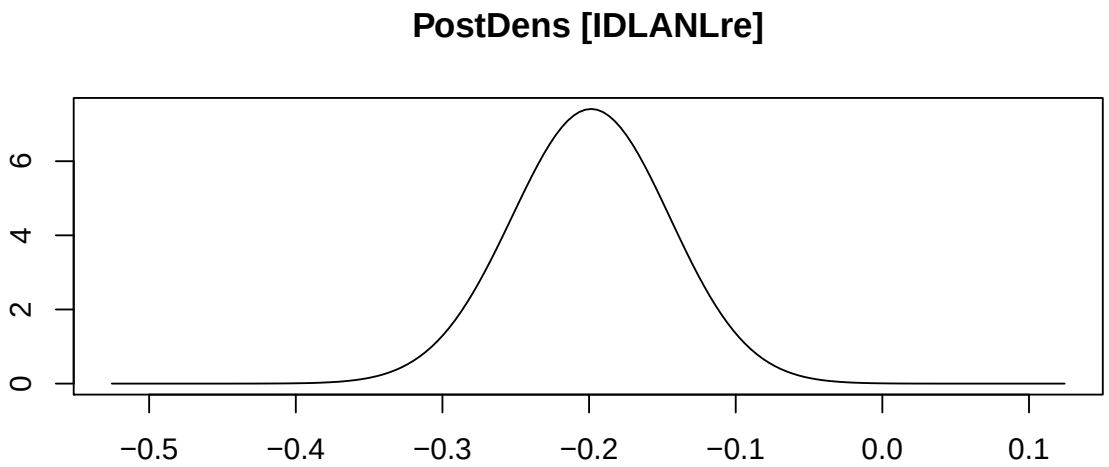
# Brain cancer in New Mexico

```
Call:
c("inla(formula = form, family = \"Poisson\", data = as.data.frame(brainst), ",  "    E = Expected, control.pre

Time used:
 Pre-processing    Running inla Post-processing           Total
         3.5222          7.2013          0.4688         11.1923

Fixed effects:
               mean     sd 0.025quant 0.5quant 0.975quant    mode kld
(Intercept) -0.0090 0.0303    -0.0689  -0.0088     0.0499 -0.0084   0
IDLANLre     0.0085 0.0092    -0.0106   0.0088     0.0256  0.0096   0

Random effects:
Name   Model
 Year   RW1 model
ID   Besags ICAR model

Model hyperparameters:
                    mean     sd       0.025quant 0.5quant 0.975quant mode
Precision for Year  8604.84 10595.75   119.82    4792.96 37374.83     109.22
Precision for ID   17406.80 17851.88   973.33   12005.23 64665.17    2495.35

Expected number of effective parameters(std dev): 2.697(0.8526)
Number of equivalent replicates : 225.41

Marginal Likelihood:  -823.89
Posterior marginals for linear predictor and fitted values computed
```

# Brain cancer in New Mexico: Distance to LANL

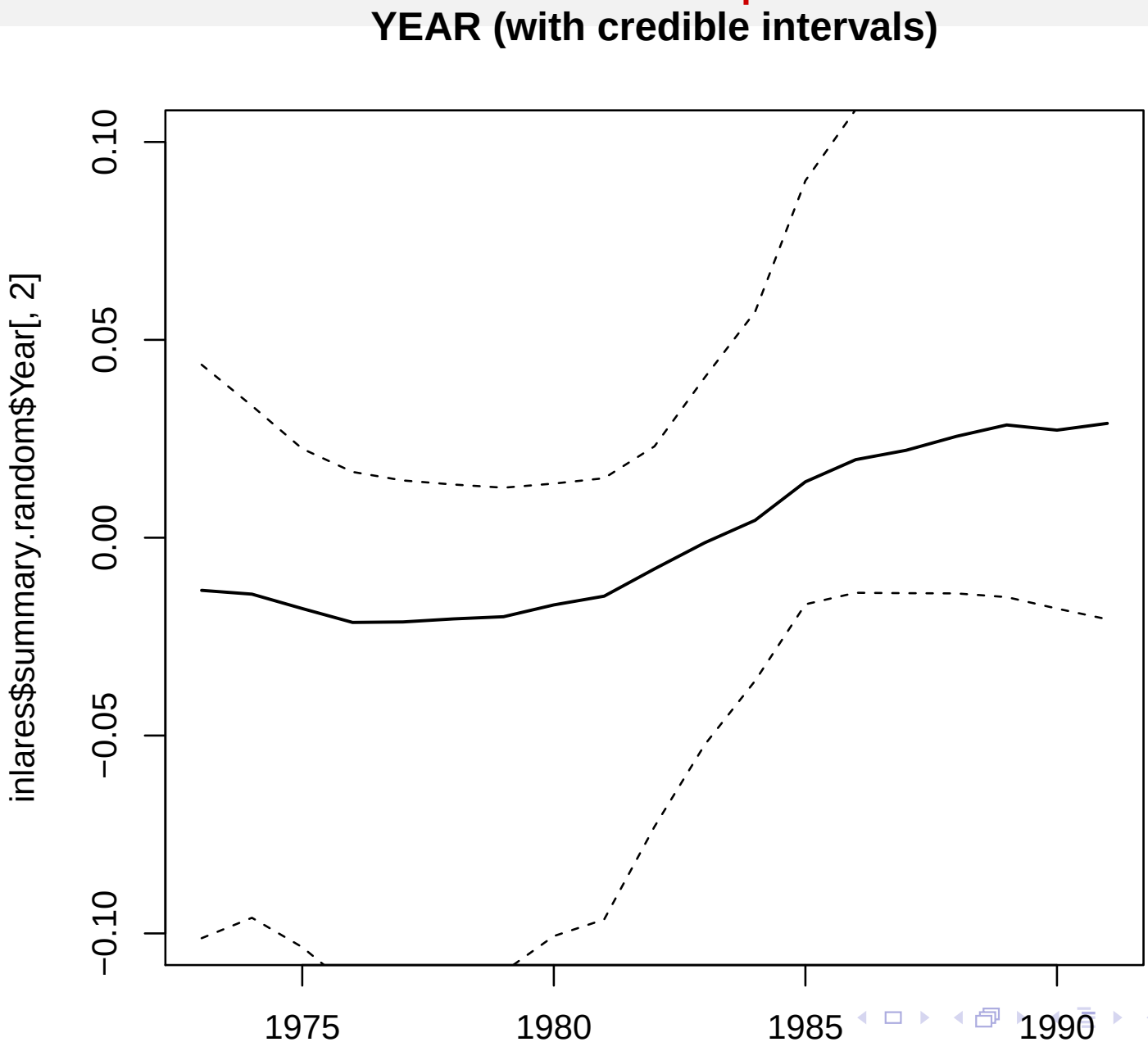



Mean = 0.162 SD = 0.052

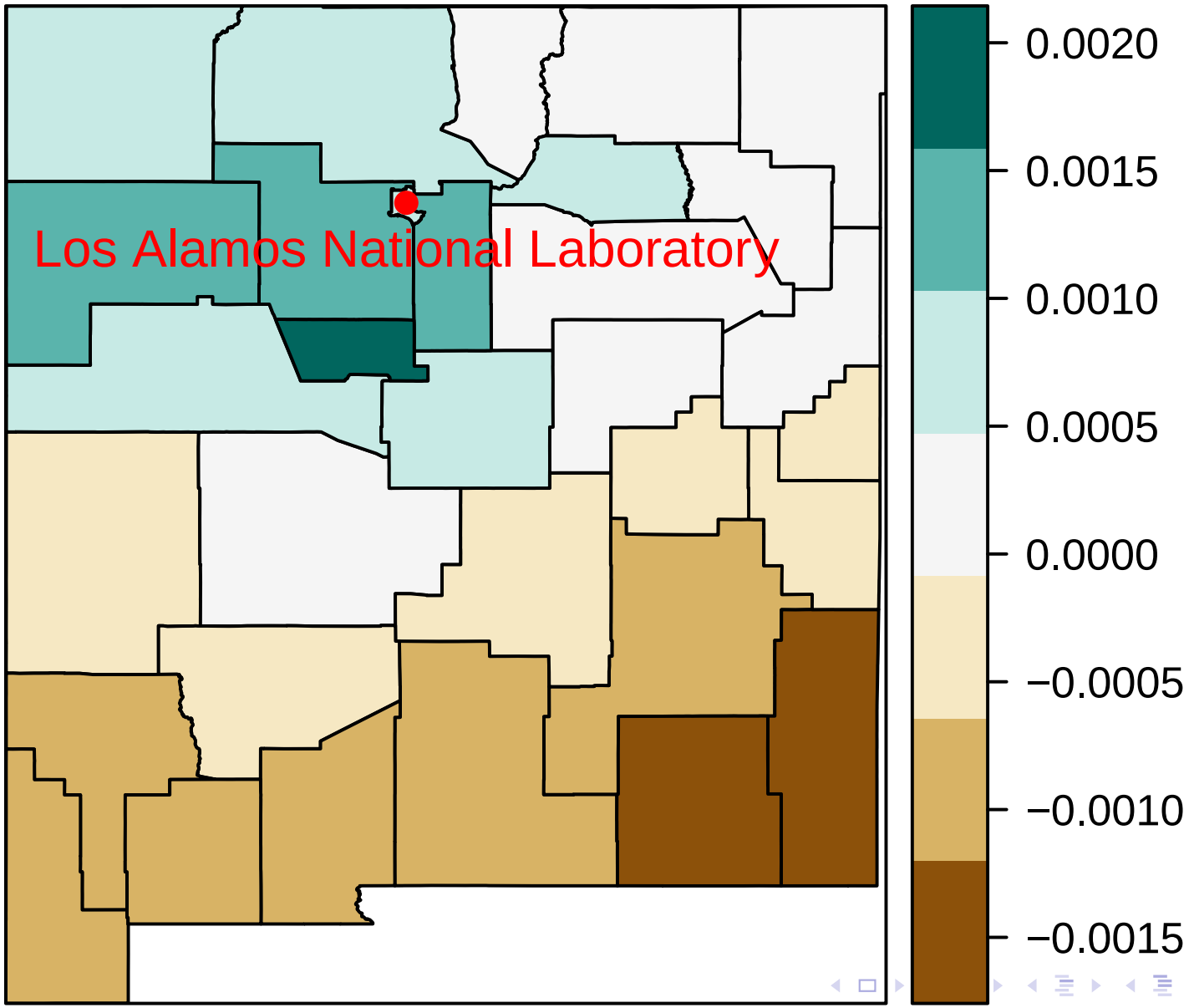



Mean = −0.199 SD = 0.054

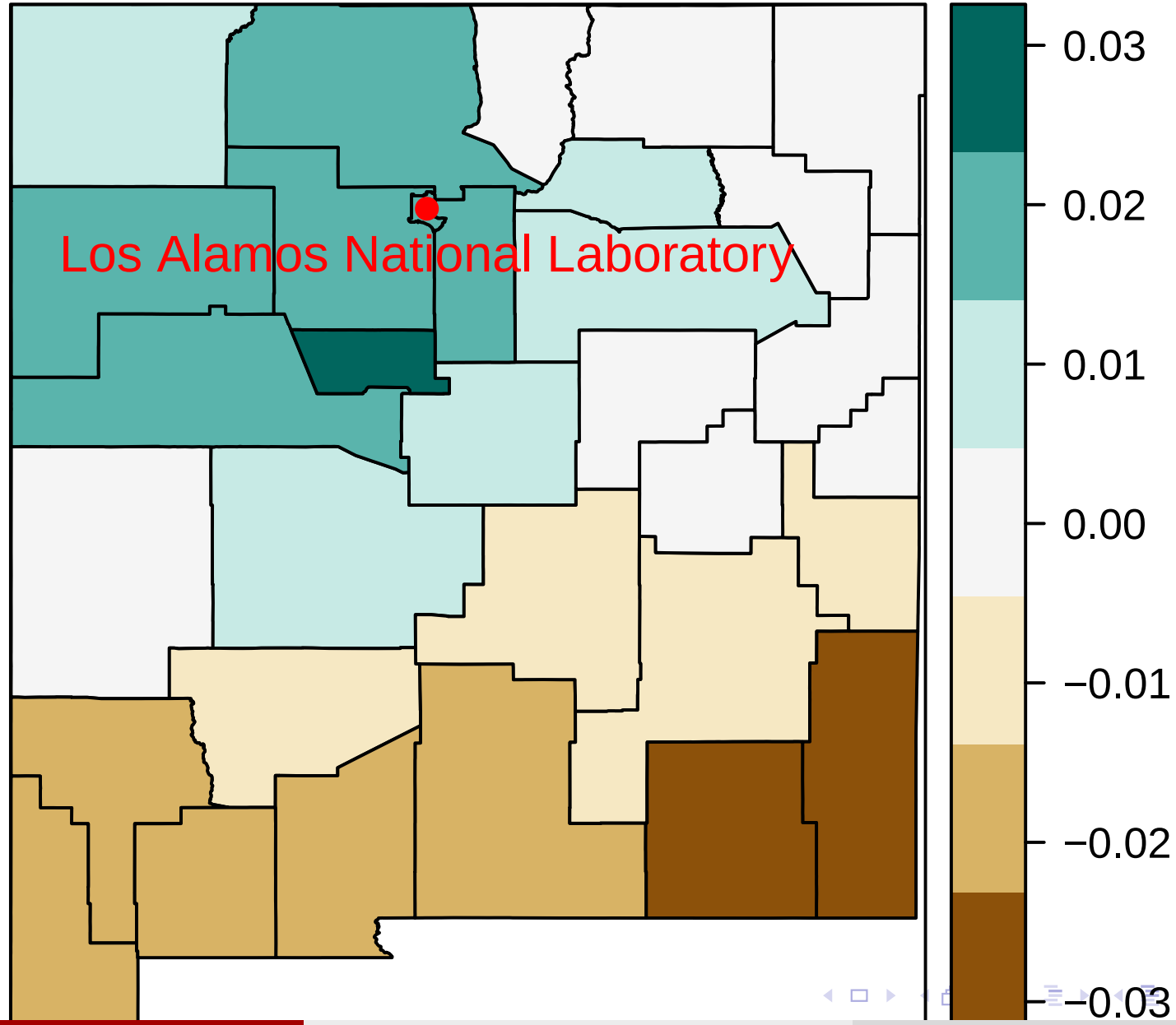
# Brain cancer in New Mexico: Temporal trend



# Brain cancer in New Mexico: Spatial random effects



# Brain cancer in New Mexico: Spatial effects (no covariates)



# Final Remarks

- Suitable models for analysing point patterns, geostatistical and lattice data are already available
- Most of them have also been implemented in R
- This is not the case for spatio-temporal models, BUT simple spatio-temporal models can be fitted using standard regression models
- Data visualization of space-time data is on the way (for example, in the `spacetime` package)
- External software can be used to fit spacetio-temporal models
- INLA provides a way of fitting models for point patterns and lattice data

# References


- S. Banerjee et al. (2003). *Hierarchical Modeling and Analysis for Spatial Data*. Chapman & Hall.
- J. Illian et al. (2008). *Statistical Analysis and MOdelling of Spatial Point Patterns*. Wiley.
- H. Rue et al. (2009). Approximate bayesian inference for latent gaussian models by using Integrated Nested Laplace Approximation (with Discussion). *Journal of the Royal Statistical Society, Series B* 71, 1–39.