Spatial econometrics 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

Introduction

1. Spatial models
2. Regression models
3. Spatial Econometrics
4. Bayesian Inference
5. Approximate Inference

Spatial models

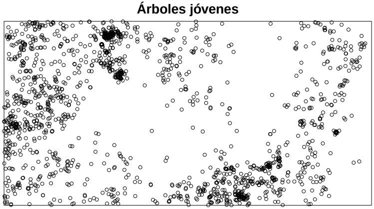
Modelling spatial dependence is important when events close in space are thought to have a similar behaviour

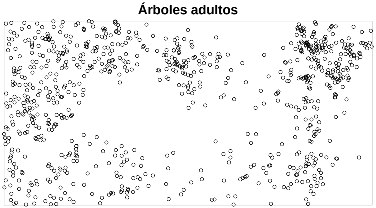
Spatial statistics

- Point patterns
- Geostatistics
- Lattice data

Point patterns

- Point patterns record the locations of a series of events in a study area
- The aim is to determine the distribution of points in the study area, i.e., their probability distribution in space
- Also, it is of interest whether the events appear independently (Complete Spatial Randomness), clustered or repelled



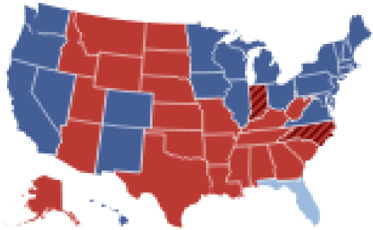


Geostatistics

- Some environmental processes show a strong spatial pattern (e.g., temperature, wind speed, concentration of heavy metals, etc.)
- If a survey is conducted in the study area there is a strong spatial dependence
- Samples close in space will have closer values than values that are further apart

Lattice data

- Lattice data involves data measured at different areas, e.g., neighbourhoods, cities, provinces, states, etc.
- Spatial dependence appears because neighbour areas will show similar values of the variable of interest



Approaches to spatial modelling

- In point patterns, we will estimate the density of the point pattern
- In geostatistics, we will model how the variable changes with distance
- In the analysis of lattice data we will model the correlation between the outcome at the different areas
- **This part focuses on the analysis of lattice data using different models**

Multivariate Normal

- The vector of observed values $\mathbf{y}$ can be supposed to have a multivariate Normal distribution:

$$\mathbf{y} \sim MVN(\mu, \Sigma)$$

- Note that each observation y_i has been measured at a different location
- The mean can be modelled to depend on relevant covariates
- The variance covariance matrix can be defined so that observations close in space have higher covariance/correlation
- In the case of lattice data, a common approach is that if two regions are neighbours they will have a higher correlation
- There are many ways of definign Σ to account for spatial dependence

Continuous processes

- In geostatistics we have a continuous process Z(x)
- Observations are taken at some survey sites $\{x_i\}_{i=1}^n$
- Many times we have multivariate data. For example, concentration of heavy metals (zinc, copper, etc.)
- A common approach is to measure how the process change, i.e, $Z(x) - Z(y)$
- We can compute the (semi-)variogram to measure spatial dependence

$$\gamma(x, y) = \frac{1}{2} E[(Z_(x) - Z(y))^2]$$

- Kriging uses the semivariogram to produce estimates such as

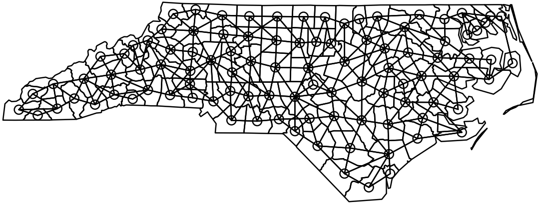
$$\hat{Z}(x) = \sum_{i=1}^{n} \lambda_i(x) Z(x_i)$$

where $\lambda_i(x)$ are a number of weights
- $\lambda_i(x)$ is higher for points x_i closer to x

Models for lattice data

- We have observations $y = \{y_i\}_{i=1}^n$ from the n areas
- y is assigned a multivariate distribution that *accounts for spatial dependence*
- A common way of describing spatial proximity in lattice data is by means of an *adjacency matrix* W
- $W[i, j]$ is non-zero if areas i and j are neighbours
- Usually, two areas are neighbours if the share a common boundary
- There are other definitions of neighbourhood

Adjacency matrix



Regression models

- It is often the case that, in addition to y_i, we have a number of covariates x_i
- Hence, we may want to regress y_i on x_i
- In addition, to the covariates we may want to account for the spatial structure of the data
- Different types of regression models can be used to model lattice data:
  - Generalized Linear Models (with spatial random effects)
  - Spatial econometrics models

Linear Mixed Models

- A common approach (for Gaussian data) is to use a linear regression with random effects

$$Y = X\beta + Zu + \varepsilon$$

- The vector random effects u is modelled as a MVN:

$$u \sim N(0, \sigma_u^2 \Sigma)$$

- Σ is defined such as it induces higher correlation in adjacent areas
- Z is a design matrix for the random effects
- $\varepsilon_i \sim N(0, \sigma^2), i = 1, \ldots, n$: error term
- The full model can be written as:

$$Y \sim N(X\beta, \sigma^2 I + \sigma_u^2 Z \Sigma Z^T)$$

Generalized Linear Models

- Y_i random variable of the Exponential Family

$$f_Y(y;\theta,\phi) = exp\{(y\theta - b(\theta))/a(\phi) + c(y,\phi)\}$$

  ϕ is a *scale* parameter and θ is the *canonical* parameter

- Linear predictor

$$\eta = \beta_1 x_1 + \ldots + \beta_k x_k$$

- *Link* function

$$g(\mu) = \eta = \beta_1 x_1 + \ldots + \beta_k x_k; \quad \mu \equiv E[Y]$$

Generalized Linear Models

Parameters of some common distributions:

Distribution	Range y	$\mu(\theta)$	Canon. link	ϕ
$\mathrm{N}(\mu, \sigma^2)$	$(-\infty, \infty)$	θ	identity	σ^2
$\mathrm{P}(\mu)$	$0, 1, \ldots, \infty$	$\exp(\theta)$	log	1
B(n, p)/n	$(0, 1, \ldots, n)/n$	$\frac{\exp^{\theta}}{1+\exp^{\theta}}$	logit	1/n

Some link functions:

$$\eta = \log(\mu)$$

$$\eta = \text{logit}(\mu) = \log(\frac{\mu}{1-\mu})$$

$$\eta = \text{probit}(\mu) = \Phi^{-1}(\mu)$$

$$\eta = \text{cloglog}(\mu) = \log(-\log(1-\mu))$$

GLMs with random effects

- Y_i random variable of the exponential family

$$f_Y(y;\theta,\phi) = exp\{(y\theta - b(\theta))/a(\phi) + c(y,\phi)\}$$

- Linear predictor

$$\eta = \beta_1 x_1 + \ldots + \beta_k x_k + Zu$$

- *Link* function

$$g(\mu) = \eta = \beta_1 x_1 + \ldots + \beta_k x_k + Zu; \quad \mu \equiv E[Y]$$

- u is a vector of random effects, which are MVN-distributed

$$u \sim N(0, \sigma^2 \Sigma)$$

- Z is a design matrix for the random effects

Structure of spatial random effects

There are **many** different ways of including spatial dependence in Σ:

- Simultaneous autoregressive (SAR):

$$\Sigma = [(I - \rho W)'(I - \rho W)]^{-1}$$

- Conditional autoregressive (CAR):

$$\Sigma = (I - \rho W)^{-1}$$

- $\Sigma_{i,j}$ depends on a function of $d(i,j)$. For example:

$$\Sigma_{i,j} = \exp\{-d(i,j)/\phi\}$$

SAR

- The variance-covariance matrix has the following structure:

$$\Sigma = [(I - \rho W)'(I - \rho W)]^{-1}$$

- W can have many different forms.
- Any of them will produce a symmetric variace-covariance matrix (but some may be singular!)
- W can be the usual $0/1$ matrix
- W is often taken as a row-standardised matrix
- This implies that in many cases $\rho \in (-1, 1)$
- See (Haining, 2003) for details

CAR

- The variance-covariance matrix has the following structure:

$$\Sigma = (I - \rho W)^{-1}$$

- In order to have a valid variance covariance matrix W must be symmetric
- W is often taken as a binary matrix
- If $\rho = 1$ we have an intrinsic CAR model but then Σ is singular
- Fitting this model will require the use of a generalized inverse
- Seldom a problem if you are Bayesian!!

Spatial Econometrics Models

- Slightly different approach to spatial modelling
- Instead of using latent effects, spatial dependence is modelled explicitly
- Autoregressive models are used to make the response variable to depend on the values at its neighbours

Spatial autoregressive models

- Autoregression on the response term

$$y = \rho W y + \alpha + e;\ \ e \sim N(0, \sigma^2 I)$$

- Autoregression on the error term

$$y = \alpha + u; u = \rho W u + e;\ \ e \sim N(0, \sigma^2 I)$$

- ρ measures spatial dependence
- If $\rho = 0$ there is no spatial dependence

Regression models on the response

$$y = \alpha + \rho W y + e;\ \ e \sim N(0, \sigma^2 I)$$

$$y = (I - \rho W)^{-1}(\alpha + e);\ \ e \sim N(0, \sigma^2 I)$$

$$y = (I - \rho W)^{-1}\alpha + \varepsilon;\ \ \varepsilon \sim N(0, \sigma^2[(I - \rho W')(I - \rho W)]^{-1})$$

Regression models on the error term

$$y = \alpha + u; u = Wu + \varepsilon;\ \varepsilon \sim N(0, \sigma^2 I)$$

$$y = \alpha + u; u = (I - \rho W)^{-1}\varepsilon;\ \varepsilon \sim N(0, \sigma^2 I)$$

$$y = \alpha + u; u \sim N(0, \sigma^2[(I - \rho W')(I - \rho W)]^{-1})$$

Simultaneous Autoregresive Model (SEM)

- This model includes covariates
- Autoregressive on the error term

$$y = X\beta + u; u = \rho W u + e; e \sim N(0, \sigma^2)$$

$$y = X\beta + \varepsilon; \varepsilon \sim N(0, \sigma^2 (I - \rho W)^{-1} (I - \rho W')^{-1})$$

Spatial Lag Model (SLM)

- This model includes covariates
- Autoregressive on the response

$$y = \rho W y + X\beta + e; e \sim N(0, \sigma^2)$$

$$y = (I - \rho W)^{-1} X\beta + \varepsilon;\ \varepsilon \sim N(0, \sigma^2 (I - \rho W)^{-1} (I - \rho W')^{-1})$$

Spatial Durbin Model (SDM)

- This model includes covariates
- Autoregressive on the response
- In addion, we include the lagged-covariates WX as another extra term in the regression

$$y = \rho W y + X\beta + WX\gamma + e = [X, WX][\beta, \gamma] + e; \ ; e \sim N(0, \sigma^2)$$

$$y = \rho W y + XWX\mathcal{B} + e; \ XWX = [X, WX]; \ \mathcal{B} = [\beta, \gamma]$$

$$y = (I - \rho W)^{-1} XWX\ \mathcal{B} + \varepsilon; \ \varepsilon \sim N(0, \sigma^2 (I - \rho W)^{-1} (I - \rho W')^{-1})$$

Probit models

- LeSage et al. (2011) use a non-Guassian model for the probability of reopening a business in the aftermath of hurricane Katrina in New Orleans.
- A non-linear link function is used between the response y (=0/1) and a latent variable y^* which represents (unobserved) net profits.
- y^* is modelled with a SLM
- Other models (SEM, SDM) could be used for the latent variable y^*
- Note also that a GLM can be fit to this data

Software

- The Spatial Econometrics Toolbox (`http://www.spatial-econometrics.com/`) provides Matlab code to fit a wealth of Spatial Econometrics models
- The R software provides number of functions to fit LMs, LMMs, GLMs and GLMMs
- The spdep package implements some functions for spatial econometrics. In particular, it includes functions to fit SEM, SLM and SDM
- Other generic software packages can be used to fit some of the models presented so far
- **INLA** recently added a new `slm` latent effect to fit Spatial Econometrics models

Bayes Inference for Spatial Models

- Bayesian inference is based on Bayes' rule to compute the probability of the parameters in the model (θ) given the observed data (y):

$$\pi(\theta|y) = \frac{\pi(y|\theta)\pi(\theta)}{\pi(y)}$$

- $\pi(y|\theta)$ is the likelihood of the model
- $\pi(\theta)$ is the prior distribution of the parameters in the model
- $\pi(y)$ is a normalising constant that is often ignored
- In spatial statistics, the prior distribution of the random effects can be used to encode spatial dependence
- Vague priors are often used for most parameters in the model

Overview of Bayesian inference

- The aim is computing the (multivarite) posterior probability of θ
- Given that $\pi(\theta|y)$ is a probability distribution, all statements are made in terms of probabilities
- Bayesian inference is 'exact'
- Obtaining $\pi(\theta|y)$ is usually hard
- However, recent computational approaches have made Bayesian inference easier

Model fitting and computational issues

- Fitting a Bayesian model means computing $\pi(\theta|y)$
- θ contains all parameters in the model and, possibly, other derived quantities
- For example, we could compute posterior probabilities of linear predictors, random effects, sums of random effects, etc.
- Depending on the likelihood and the prior distribution computing $\pi(\theta|y)$ can be very difficult
- In the last 20-30 years some computational approaches have been proposed to estimate $\pi(\theta|y)$ with Monte Carlo methods

Markov Chain Monte Carlo

- MCMC is a family of algorithms to obtain draws from the posterior distribution
- In all cases, a starting point is chosen to start the simulation
- At every iteration k we draw a sample of the model parameters $\hat{\theta}_i^{(k)}$ using a particular rule
- After a number of iterations (burn-in period) the algorithm is in fact sampling from $\pi(\theta|y)$
- The iterations generated during the burn-in period are discarded
- A large number of simulations is generated and the posterior distribution is estimated from these samples
- Summary statistics for the model parameters can be easily computed from the simulations

Metropolis-Hasting Sampling

- Generic algorithm to sample from any probablity density $f(y)$
- A candidate-generating probability density $q(v|u)$ is required for every parameter in the model
- This will give us the probabilities of sampling v given that we are at u
- We draw a value from this density
- This new value is only accepted with a certain probability, which is

$$\min\{1, \frac{\pi(v|y)q(u|v)}{\pi(u|y)q(v|u)}\}$$

- Note that

$$\frac{\pi(v|y)q(u|v)}{\pi(u|y)q(v|u)} = \frac{\pi(y|v)\pi(v)q(u|v)}{\pi(y|u)\pi(u)q(v|u)}$$

and that the probability can be computed

Gibbs Sampling

- Particular case of Metropolis-Hastings algorithm
- The proposal distribution is the conditional distribution given the :

$$\pi(\theta_i^{(k+1)} | \theta_1^{(k+1)}, \ldots, \theta_{i-1}^{(k+1)}, \theta_{i+1}^{(k)}, \ldots \theta_N^{(k)})$$

- This ensures that the acceptance probability is always 1
- This means that we always accept a new candidate point, i.e., ”we always move to a new point”
- Sampling from the conditional probability distribution is usually very easy

Inference with MCMC

- MCMC provides simulations from the ensemble of model parameters, i.e., a multivariate distribution
- This will allow us to estimate the joint posterior distribution
- However, we may be interested in a single parameter or a subset of the parameters
- Inference for this subset of parameters can be done by simply ignoring the samples for the other parameters
- Using the samples it is possible to compute the posterior distribution of any function on the model parameters
- MCMC may require lots of simulations to make valid inference
- Also, we must check that the burn-in period has ended, i.e., we have reached the posterior distribution

Integrated Nested Laplace Approximation

- Sometimes we only need marginal inference on some parameters, i.e., we need $\pi(\theta_i|y)$
- Rue et al. (2009) propose a way of approximating the marginal distributions
- Now we are dealing with dealing with (many) univariate distributions
- This is computationally faster because numerical integration techniques are used instead of Monte Carlo sampling

Integrated Nested Laplace Approximation

- We assume that observations $\mathbf{y}$ are independent given $\mathbf{x}$ (latent effects) and $\theta = (\theta_1, \theta_2)$ (two sets of hyperparameters)
- The model likelihood can be written down as

$$\pi(\mathbf{y}|\mathbf{x}, \theta) = \prod_{i \in \mathcal{I}} \pi(y_i | x_i, \theta)$$

- x_i is the latent linear predictor η_i and other latent effects
- $\mathcal{I}$ represents the indices of the observations (missing observations are not include here, for example)
- $\theta = (\theta_1, \theta_2)$ is a vector of hyperparameters for the likelihood and the distribution of the latent effects

Integrated Nested Laplace Approximation

- $\mathbf{x}$ is assumed to be distributed as a Gaussian Markov Random Field with precission matrix $Q(\theta_2)$
- The posterior distribution of the model parameters and hyperparameters is:

$$\pi(\mathbf{x},\theta|\mathbf{y}) \propto \pi(\theta)\pi(\mathbf{x}|\theta)\prod_{i\in\mathcal{I}}\pi(y_i|x_i,\theta) \propto$$

$$\pi(\theta)|\mathbf{Q}(\theta)|^{n/2}\exp\{-\frac{1}{2}\mathbf{x}^T\mathbf{Q}(\theta)\mathbf{x}+\sum_{i\in\mathcal{I}}\log(\pi(y_i|x_i,\theta)\}$$

Integrated Nested Laplace Approximation

The marginal distributions for the latent effects and hyper-parameters can be written as

$$\pi(x_i|\mathbf{y}) \propto \int \pi(x_i|\theta, \mathbf{y})\pi(\theta|\mathbf{y})d\theta$$

and

$$\pi(\theta_j|\mathbf{y}) \propto \int \pi(\theta|\mathbf{y})d\theta_{-j}$$

Integrated Nested Laplace Approximation

Rue et al. (2009) provide a simple approximation to $\pi(\theta|\mathbf{y})$, denoted by $\tilde{\pi}(\theta|\mathbf{y})$, which is then used to compute the approximate marginal distribution of a latent parameter x_i:

$$\tilde{\pi}(x_i|\mathbf{y}) = \sum_k \tilde{\pi}(x_i|\theta_k, \mathbf{y}) \times \tilde{\pi}(\theta_k|\mathbf{y}) \times \Delta_k$$

Δ_k are the weights of a particular vector of values θ_k in a grid for the ensemble of hyperparameters .

INLA & Spatial econometrics models

- In principle, INLA can handle a large number of models
- The R-INLA package for the R software implements a number of likelihoods and latent effects
- This includes GLMs
- This includes several latent effects for lattice data
- The CAR model is implemented
- SEM, SLM and SDM are not implemented
- The SAR specification is not implemented as a random effects
- Linear predictors are multiplied by $(I - \rho W)^{-1}$, and this is not implemented either
- What to do then?

New `slm` latent class

- **INLA** includes now a new latent effect:

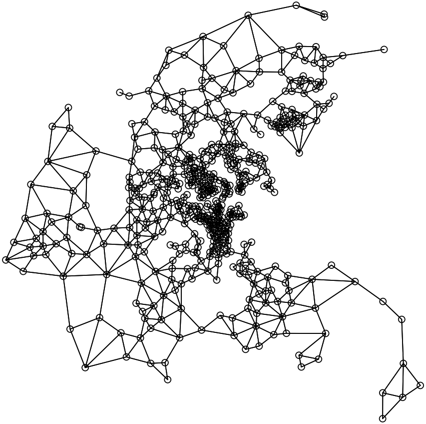
$$\mathbf{x} = (I_n - \rho W)^{-1}(X\beta + e)$$

- W is a row-standardised adjacency matrix
- ρ is a spatial autocorrelation parameter
- X is a matric of covariates, with coefficients β
- e are Gaussian i.i.d. errors

Example: Boston housing data

In our first example we will re-analyse the Boston housing data (Harrison and Rubinfeld, 1978). Here the interest is in estimating the median of owner-occupied houses using relevant covariates and the spatial structure of the data (Pace and Gilley, 1997). We have fitted the 3 spatial econometrics models described in this paper plus a spatial model with a CAR error term. In addition, we have fitted the spatial econometrics models using maximum likelihood to compare the estimates of the model parameters

Boston housing data: Adjacency matrix



Boston housing data: Load data

```
> #Load libraries
> library(INLA)
> library(spdep)
> library(parallel)
> #Load data
> #data(boston)
> library(maptools)
> boston.tr <- readShapePoly(system.file("etc/shapes/boston_tracts.shp",
+   package="spdep")[1], ID="poltract",
+   proj4string=CRS(paste("+proj=longlat +datum=NAD27 +no_defs +ellps=clrk66",
+   "+nadgrids=@conus,@alaska,@ntv2_0.gsb,@ntv1_can.dat")))
> boston_nb <- poly2nb(boston.tr)
> censored <- boston.tr$CMEDV == 50
> boston.c <- boston.tr[!censored,]
> boston_nb_1 <- subset(boston_nb, !censored)
> lw <- nb2listw(boston_nb_1, style="W")
> #Define some indices used in the models
> n<-nrow(boston.c)
> boston.c$idx<-1:n
> #Adjcency matrix
> #W<-nb2mat(boston.soi)
> W <- as(as_dgRMatrix_listw(lw), "CsparseMatrix")
```

Boston housing data: ML estimation

- First of all, we will fit the spatial econometrics models using maximum likelihood
- Bayesian estimates should be similar (under vague priors)

```
> #Model matrix for SLM models
> f1<-log(CMEDV) ~ CRIM + ZN + INDUS + CHAS + I(NOX^2)+ I(RM^2) +  AGE + log(DIS) + log(RAD) + TAX + PTRATIO + B
> mmatrix <- model.matrix(f1, boston.c)
> mmatrix2 <- cbind(mmatrix, create_WX(mmatrix, lw, prefix="lag"))
> #Compute some Spatial Econometrics models using ML estimation
> m1<-errorsarlm(f1, boston.c, lw)
> m2<-lagsarlm(f1, boston.c, lw)
> m3<-lagsarlm(f1, boston.c, lw, type="mixed")

> summary(m1)
> summary(m2)
> summary(m3)
```

Boston housing data: INLA

```
> #DEFINE PRIORS TO BE USED WITH R-INLA
>
> #Zero-variance for Gaussian error term
> zero.variance = list(prec=list(initial = 25, fixed=TRUE))
> #Compute eigenvalues for SLM model (as in Havard's code)
> e = eigenw(lw)
> re.idx = which(abs(Im(e)) < 1e-6)
> rho.max = 1/max(Re(e[re.idx]))
> rho.min = 1/min(Re(e[re.idx]))
> rho = mean(c(rho.min, rho.max))
> #
> #Variance-covarinace matrix for beta coeffients' prior
> #
> betaprec<-.0001
> #Standard regression model
> Q.beta = Diagonal(n=ncol(mmatrix), x=1)
> Q.beta = betaprec*Q.beta
> #Regression model with lagged covariates
> Q.beta2 = Diagonal(n=ncol(mmatrix2), x=1)
> Q.beta2 = betaprec*Q.beta2
>
```

Boston housing data: INLA

```
> #This defines the range for the spatial autocorrelation parameters
> # the spatial adjacenty matrix W, the associated
> #matrix of covariates X and the precision matrix Q.beta for the prior
> #on the coefficients of the covariates
> #
> args.slm = list(
+    rho.min = rho.min ,
+    rho.max = rho.max,
+    W = W,
+    X = matrix(0, nrow(mmatrix),0),
+    Q.beta = matrix(1,0,0)
+ )
> #Definition of priors for precision of the error effect in the slm latent
> #effect and the spatial autocorrelation parameter (in the (0,1) interval).
> #
> hyper.slm = list(
+    prec = list(
+       prior = "loggamma", param = c(0.01, 0.01)),
+       rho = list(initial=0, prior = "logitbeta", param = c(1,1))
+ )
>
```

Boston housing data: INLA

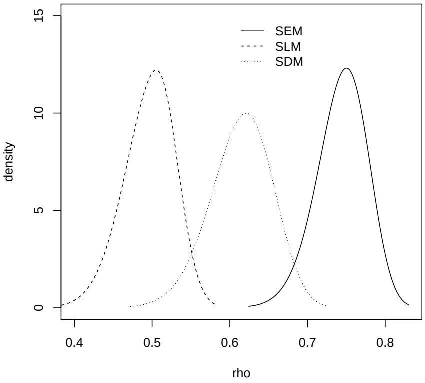
```
> #SEM model
> semm1<-inla(log(CMEDV) ~ CRIM + ZN + INDUS + CHAS + I(NOX^2)+ I(RM^2) +
+    AGE + log(DIS) + log(RAD) + TAX + PTRATIO + B + log(LSTAT)+
+    f(idx, model="slm", args.slm=args.slm, hyper=hyper.slm),
+    data=as(boston.c, "data.frame"), family="gaussian",
+    control.family = list(hyper=zero.variance),
+    control.compute=list(dic=TRUE, cpo=TRUE)
+ )
> #SLM model
> slmm1<-inla( log(CMEDV) ~ -1 +
+    f(idx, model="slm",
+       args.slm=list(rho.min = rho.min, rho.max = rho.max, W=W, X=mmatrix,
+          Q.beta=Q.beta),
+       hyper=hyper.slm),
+    data=as(boston.c, "data.frame"), family="gaussian",
+    control.family = list(hyper=zero.variance),
+    control.compute=list(dic=TRUE, cpo=TRUE)
+ )
> #SDM model
> sdmm1<-inla( log(CMEDV) ~ -1 +
+    f(idx, model="slm",
+       args.slm=list(rho.min = rho.min, rho.max = rho.max, W=W, X=mmatrix2,
+          Q.beta=Q.beta2),
+       hyper=hyper.slm),
+    data=as(boston.c, "data.frame"), family="gaussian",
+    control.family = list(hyper=zero.variance),
+    control.compute=list(dic=TRUE, cpo=TRUE)
+ )
>
```

Boston housing data: spatial autocorrelation

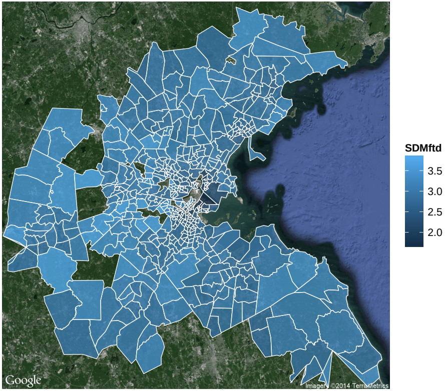
- **INLA** reports the values of ρ in the (0,1) interval
- They need to be re-scaled to the $(rho.min, rho.max)$ interval
- We will use `inla.tmarginal()` to transform the reported marginal
- `inla.zmarginal()` can be used to report summary statistics from a marginal

```
> #Transform Spatial autocorrelation parameters to be in  (rho.min, rho.max)
> #
> ff<-function(z){z*(rho.max-rho.min)+rho.min}
> semmarg<-inla.tmarginal(ff, semm1$marginals.hyperpar[[2]])
> slmmarg<-inla.tmarginal(ff, slmm1$marginals.hyperpar[[2]])
> sdmmarg<-inla.tmarginal(ff, sdmm1$marginals.hyperpar[[2]])
> inla.zmarginal(semmarg, TRUE)
> inla.zmarginal(slmmarg, TRUE)
> inla.zmarginal(sdmmarg, TRUE)
```

Boston housing data: Estimates of ρ



Boston housing data: Display results

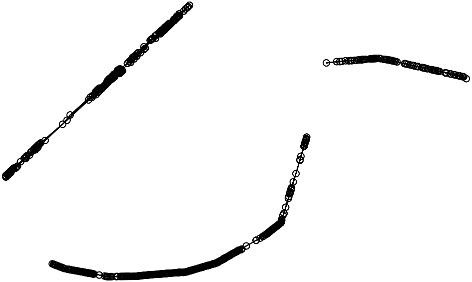


Example: New Orleans business data

In this example we will look at the data analysed in LeSage et al. (2011) regarding the probability of re-opening a business in the aftermath of hurricane Katrina. In this case we have a non-Gaussian model because we are modelling a probability and the response variable can take either 1 (the business re-opened) or 0 (the business didn't re-open). Similarly as in the previous example, we have fitted four models. However, now we have used a GLM with a Binomial family and a probit link.

LeSage et al. (2011) split the data into four periods according to different time frames. In our analysis we will focus on the first period, i.e., the business re-opened during the first 3 months (90 days). The model used therein is the one that we have termed Spatial Lag Model in this paper.

New Orleans business data: Adjacency matrix



New Orleans business data: Load data

```
> #
> #RE-analysis of the Katrina dataset using R-INLA
> #
> library(INLA)
> library(parallel)
> library(spdep)
> #Here I use the katrina dataset from spatialprobit
>
> library(spatialprobit)
> data(Katrina)
> #And index for slm model
> n<-nrow(Katrina)
> Katrina$idx<-1:n
> # (a) 0-3 months time horizon
> # LeSage et al. (2011) use k=11 nearest neighbors in this case
> nb <- knn2nb(knearneigh(cbind(Katrina$lat, Katrina$long), k=11))
> listw <- nb2listw(nb, style="W")
> W1 <- as(as_dgRMatrix_listw(listw), "CsparseMatrix")
>
>
>
```

New Orleans business data: INLA

```
> #Model matrix for SLM models
> f1<-y1~ 1+flood_depth+log_medinc+small_size+large_size+
+    low_status_customers+high_status_customers+owntype_sole_proprietor+
+    owntype_national_chain
> mm <- model.matrix(f1, Katrina)
> #With lagged covariates
> mm2 <-  cbind(mm,as.matrix(W1) %*% mm[,-1])
> #Zero-variance for Gaussian erros, not needed here(?)
> zero.variance = list(prec=list(initial = 25, fixed=TRUE))
> #DEFINE PRIORS TO BE USED WITH R-INLA
>
> #Compute eigenvalues for SLM model (as in Havard's code)
> e = eigen(W1)$values
> re.idx = which(abs(Im(e)) < 1e-6)
> rho.max = 1/max(Re(e[re.idx]))
> rho.min = 1/min(Re(e[re.idx]))
> rho = mean(c(rho.min, rho.max))
>
>
>
```

New Orleans business data: INLA

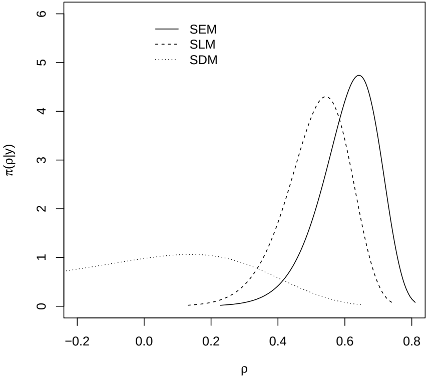
```
> #Variance-covarinace matrix for beta coeffients' prior
> #
> betaprec1<-.0001
> #Standard regression model
> Q.beta1 = Diagonal(n=ncol(mm), x=1)
> Q.beta1 = betaprec1*Q.beta1
> #Regression model with lagged covariates
> Q.beta2 = Diagonal(n=ncol(mm2), x=1)
> Q.beta2 = betaprec1*Q.beta2
> #This defines the range for the spatial autocorrelation parameters
> # the spatial adjacenty matrix W, the associated
> #matrix of covariates X and the precision matrix Q.beta for the prior
> #on the coefficients of the covariates
> #
> args.slm = list(
+    rho.min = rho.min ,
+    rho.max = rho.max,
+    W = W1,
+    X = matrix(0, nrow(mm),0),
+    Q.beta = matrix(1,0,0)
+ )
> #Definition of priors for precision of the error effect in the slm latent
> #effect and the spatial autocorrelation parameter (in the (0,1) interval).
> #
> hyper.slm = list(
+    prec = list(initial=log(1), fixed=TRUE),#prior = "loggamma", param = c(0.01, 0.01)),
+       rho = list(prior = "logitbeta", param = c(1, 1))#list(initial=0, prior = "logitbeta", param = c(1,1))
+ )
>
>
>
```

New Orleans business data: INLA

```
> #SEM model
> semm1<-inla(
+    update(f1, ~.+f(idx, model="slm", args.slm=args.slm, hyper=hyper.slm)),
+    data=Katrina, family="binomial",
+    control.family = list(link="probit", hyper=zero.variance),
+    control.compute=list(dic=TRUE, cpo=TRUE)
+ )
> #SLM model
> slmm1<-inla( y1 ~ -1 +
+    f(idx, model="slm",
+       args.slm=list(rho.min = rho.min, rho.max = rho.max, W=W1, X=mm,
+          Q.beta=Q.beta1),
+       hyper=hyper.slm),
+    data=Katrina, family="binomial",
+    control.family = list(link="probit", hyper=zero.variance),
+    control.compute=list(dic=TRUE, cpo=TRUE)
+ )
> #SDM model
> sdmm1<-inla( y1 ~ -1 +
+    f(idx, model="slm",
+       args.slm=list(rho.min = rho.min, rho.max = rho.max, W=W1, X=mm2,
+          Q.beta=Q.beta2),
+       hyper=hyper.slm),
+    data=Katrina, family="binomial",
+    control.family = list(link="probit", hyper=zero.variance),
+    control.compute=list(dic=TRUE, cpo=TRUE)
+ )
>
>
>
>
```

New Orleans business data: spatial autocorrelation

```
> #Transform Spatial autocorrelation parameters to be in  (rho.min, rho.max)
> ff<-function(z){z*(rho.max-rho.min)+rho.min}
> semmarg<-inla.tmarginal(ff, semm1$marginals.hyperpar[[1]])
> slmmarg<-inla.tmarginal(ff, slmm1$marginals.hyperpar[[1]])
> sdmmarg<-inla.tmarginal(ff, sdmm1$marginals.hyperpar[[1]])
```



Other issues and current work

- Impacts are difficult to compute as their are based on bivariate inference:

$$\frac{\partial y_i}{\partial x'_{v,j}}$$

- Exploit INLA for bivariate inference
- Computational issues with INLA with the Probit model
- Simulations to assess differences between GLMs and Spatial Econometrics models
- Develop R code similar to the Spatial Econometrics Toolbox (GSoc project) to compare with our results with INLA

References

Haining, R. (2003). *Spatial Data Analysis: Theory and Practice*. Cambridge University Press.

Harrison, D. and D. L. Rubinfeld (1978). Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management 5*, 81–102.

LeSage, J. P., K. R. Pace, N. Lam, R. Campanella, and X. Liu (2011). New Orleans business recovery in the aftermath of Hurricane Katrina. *Journal of the Royal Statistical Society: Series A (Statistics in Society) 174*, 1007–1027.

Pace, R. K. and O. W. Gilley (1997). Using the spatial configuration of the data to improve estimation. *Journal of the Real Estate Finance and Economics 14*, 333–340.

Rue, H., S. Martino, and N. Chopin (2009). Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of the Royal Statistical Society, Series B 71*(Part 2), 319–392.