

# Small Area Estimation with R

## Unit 5: Bayesian Small Area Estimation

V. Gómez-Rubio

Department of Mathematics  
Universidad de Castilla-La Mancha, Spain

# What is this 'Bayesian' thing?

## Formulation

$$\pi(\theta|x) \propto f(x|\theta)\pi(\theta)$$

$\pi(\theta|x)$  is the *posterior distribution* of the parameters  $\theta$  given the data  $x$

$f(x|\theta)$  is the *likelihood* of the model

$\pi(\theta)$  is the *prior distribution* of the parameters  $\theta$

## Introduction

- Bayesian Hierarchical Models are Multilevel Models
- All unknown quantities and parameters of the model  $\theta$  are considered as random variables
- Inference is based on the distribution of  $\theta$  given the observed data
- Complex models must be fitted using computational procedures (Markov Chain Monte Carlo methods) to obtain a sample from the posterior distribution of  $\theta$

# So, what is all this about?

## Main ideas

- BHM are formulated in the same way as the models that we have been so far
- However, a *prior* distribution is assigned to each parameter to reflect our previous information. Usually, these priors are taken as vague as possible to reflect that there is no previous information
- In some cases, the priors are taken so that they elicit information provided by experts
- The choice of the prior will affect the posterior distribution of the parameters of the model and the results

## Some benefits of Bayesian Inference

- Probability statements like  $P(\theta_L < \text{Av. Income} < \theta_U)$
- Results can be summarised as posterior probabilities: What is the probability of having an income higher than ■1000/week?

# Area Level Models

## Fay-Herriott Estimator

$$\begin{aligned}\hat{Y}_{D,i} &= \mu_i + e_i \\ e_i &\sim N(0, \hat{\sigma}_i^2)\end{aligned}$$

$$\mu_i = \alpha + \beta \bar{X}_i + u_i + v_i$$

$$u_i \sim N(0, \sigma_u^2)$$

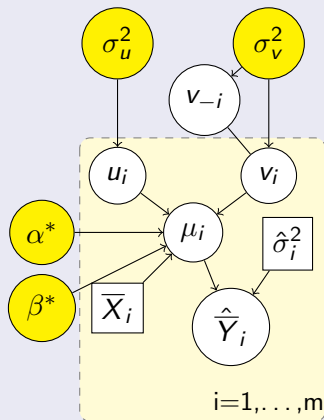
$$v_i | v_{-i} \sim N\left(\sum_{j \in \delta_i} \frac{v_j}{|\delta_i|}, \frac{\sigma_v^2}{|\delta_i|}\right)$$

$$\sigma_u^2, \sigma_v^2 \sim Ga^{-1}(0.001, 0.001)$$

## Small Area Estimation

$$\hat{Y}_{A,i} = \hat{\mu}_i$$

## Graphical Model



# Unit Level Models

## Model description

$$y_{ij} = \mu_{ij} + e_{ij}$$

$$e_{ij} \sim N(0, \sigma_e^2)$$

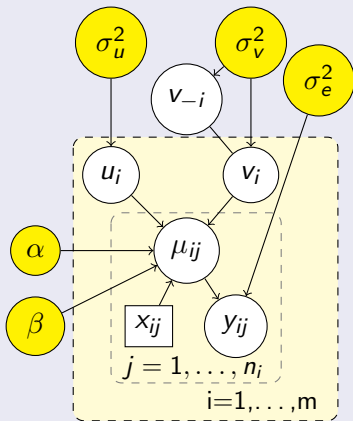
$$\sigma_e^2 \sim Ga^{-1}(0.001, 0.001)$$

$$\mu_{ij} = \alpha + \beta x_{ij} + u_i + v_i$$

## Small Area Estimation

$$\hat{\bar{Y}}_{u,i} = \hat{\alpha} + \hat{\beta} \bar{X}_i + \hat{u}_i + \hat{v}_i$$

## Graphical Model



# Unit Level Models

## Model description

$$y_{ij} = \mu_{ij} + e_{ij}$$

$$e_{ij} \sim N(0, \sigma_i^2)$$

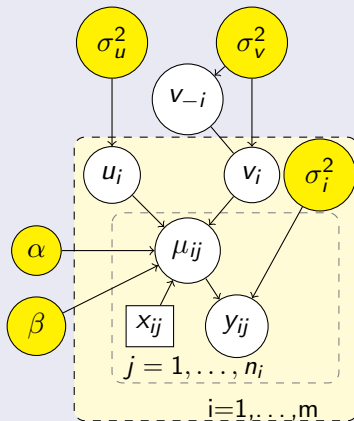
$$\sigma_i^2 \sim Ga^{-1}(0.001, 0.001)$$

$$\mu_{ij} = \alpha + \beta x_{ij} + u_i + v_i$$

## Small Area Estimation

$$\hat{\bar{Y}}_{u,i} = \hat{\alpha} + \hat{\beta} \bar{X}_i + \hat{u}_i + \hat{v}_i$$

## Graphical Model



# Unit Level Models

## Model description

$$y_{ij} = \mu_{ij} + e_{ij}$$

$$e_{ij} \sim N(0, \sigma_i^2)$$

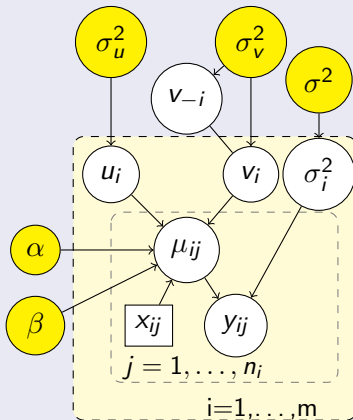
$$\log(\sigma_i^2) \sim N(0, \sigma^2)$$

$$\mu_{ij} = \alpha + \beta x_{ij} + u_i + v_i$$

## Small Area Estimation

$$\hat{\bar{Y}}_{u,i} = \hat{\alpha} + \hat{\beta} \bar{X}_i + \hat{u}_i + \hat{v}_i$$

## Graphical Model



# Fitting Bayesian models with WinBUGS

- WinBUGS provides a graphical user interface to deal with the fitting of Bayesian models
- The information required is
  - Model definition
  - Data
  - Initial Values
- Model fitting is mainly done using Gibbs Sampling
- Summary statistics and plots of the output can be produced
- In practice, we will call WinBUGS from R using function from package **R2WinBUGS**
- Analysis of the output will be done with package **coda**



# Example: Mixed effects models

## Average income per household

- The survey covers a number of households in the country
- Variables measured are:
  - Average income per household (target variable)
  - Number of persons in the household
  - Number of persons employed in the household
  - Age of head of household
  - Gender of head of household
  - Higher education status of head of household
- For each municipality, we have area level means/proportions of the covariates

# WinBUGS area level model

```
model
{
  for(i in 1:N)
  {
    prece[i]<-1/desvar[i]
    Y[i] ~ dnorm(mu[i], prece[i])
    mu[i] <- alpha + bhhpers*HHPERS[i]+...+ bsexhh*SEXHH[i]+ u[i]

    u[i] ~ dnorm(0, precu)
  }

  precu ~ dgamma (a0,b0)

  alpha ~ dflat()

  bhhpers ~ dflat()
  bhhemp1 ~ dflat()
  bedterhh ~ dflat()
  bagehh ~ dflat()
  bsexhh ~ dflat()

  sigmau<-1/precu
}
```

# WinBUGS unit level model

```
model
{
  for(i in 1:N)
  {
    for(j in 1:n[i])
    {
      y[cumn[i]+j] ~ dnorm(mu[cumn[i]+j], prece[i])

      mu[cumn[i]+j] <- alpha + bhhpers*hhpers[cumn[i]+j] + ... +bsexhh*sexhh[cumn[i]+j] +u[i]
    }
    u[i] ~ dnorm(0, precu)

    mug[i]<-alpha+u[i]+bhhpers*HHPERS[i]+...+bsexhh*SEXHH[i]

    logsigmae[i] ~ dnorm(0, prec)
    sigmae[i] <-exp(logsigmae[i])
    prece[i]<- 1/sigmae[i]
  }

  precu ~ dgamma (a0,b0)
  prec ~ dgamma (a2,b2)

  alpha ~ dflat()
  bhhpers ~ dflat()
  bhhemp1 ~ dflat()
  bedterhh ~ dflat()
  bagehh ~ dflat()
  bsexhh ~ dflat()
}
```

# Running the area level model

The following code reads that data for the area level model and two sets of initial values to start two different Markov Chains.

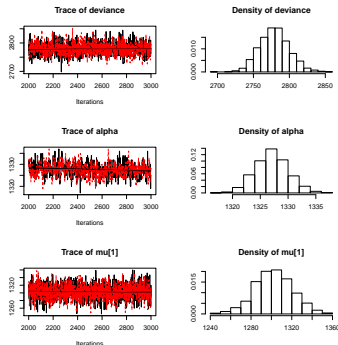
```
> library(R2WinBUGS)
> areamodel<-paste(getwd(), "/Income/models/area_model.txt", sep="")
> dataarea<-source("Income/WBdata/area_data.txt")$value
> initsarea1<-source("Income/area_inits/inits-1.txt")$value
> initsarea2<-source("Income/area_inits/inits-2.txt")$value
> wdir<-paste(getwd(), "/Income_area", sep="")
> if(!file.exists(wdir)){dir.create(wdir)}
> BugsDir <- "/home/asdar/.wine/dosdevices/c:/Program Files/WinBUGS14"
> areares<- bugs(data=dataarea, inits=list(initsarea1, initsarea2),
+   working.directory=wdir,
+   parameters.to.save=c("mu", "sigmau", "u", "alpha", "bedterhh"),
+   n.chains=2, n.iter=3000, n.burnin=2000, n.thin=1,
+   model.file=areamodel,
+   bugs.directory=BugsDir,
+   WINEPATH="/usr/bin/winepath")
```

# Running the unit level model

The following code reads that data for the unit level model and two sets of initial values to start two different Markov Chains.

```
> unitmodel<-paste(getwd(), "/Income/models/unit_model.txt", sep="")
> dataunit<-source("Income/WBdata/unit_data.txt")$value
> initsunit1<-source("Income/unit_inits/inits-1.txt")$value
> initsunit2<-source("Income/unit_inits/inits-2.txt")$value
> wdir<-paste(getwd(), "/Income_unit", sep="")
> if(!file.exists(wdir)){dir.create(wdir)}
> BugsDir <- "/home/asdar/.wine/dosdevices/c:/Program Files/WinBUGS14"
> unitres<- bugs(data=dataunit, inits=list(initsunit1, initsunit2),
+   working.directory=wdir,
+   parameters.to.save=c("mug", "sigmau", "u", "alpha", "bedterhh", "sigmae"),
+   n.chains=2, n.iter=3000, n.burnin=2000, n.thin=1,
+   model.file=unitmodel,
+   bugs.directory=BugsDir,
+   WINEPATH="/usr/bin/winepath")
>
```

# Assessment of the convergence



WinBUGS will create two files with the output of the Markov Chains that can be read. These chains can be plotted and the posterior density of the parameters displayed.

The posterior deviance of the model can be monitored as a guidance to the overall convergence.

```
> library(coda)
> chain1<-read.coda("Income_area/coda1.txt", "Income_area")
> chain2<-read.coda("Income_area/coda2.txt", "Income_area")
> chains<-mcmc.list(chain1, chain2)
> plot(chains[, c("deviance", "alpha", "mu[1]")])
```

## CAR specification

We can include spatial random effects by means of a Conditionally Autoregressive specification:

$$v_i | v_{-i} \sim N\left(\sum_{j \in \delta_i} \frac{v_j}{|\delta_i|}, \frac{\sigma_v^2}{|\delta_i|}\right)$$

- $\delta_i$  denotes the list of neighbours of the area  $i$ . It can be obtained using GeoBUGS or function `nb2WB` (in package **spdep**)
- $\sigma_v^2$  is the *conditional* variance
- The CAR specification has the particularity that it can be conveniently fitted by Gibbs Sampling
- This prior is improper and may lead to improper priors

# Code for area level model with spatial random effects

```
model
{
  for(i in 1:N)
  {
    prece[i]<-1/desvar[i]
    Y[i] ~ dnorm(mu[i], prece[i])
    mu[i] <- alpha + bhhpers*HHPERS[i] + ... + bsexhh*SEXHH[i]+ u[i] +v[i]

    u[i] ~ dnorm(0, precu)
  }

  v[1:N] ~ car.normal(adj[], weights[], num[], precv)

  precu ~ dgamma (a0,b0)
  precv ~ dgamma (a1,b1)

  alpha ~ dflat()
  bhhpers ~ dflat()
  bhhemp1 ~ dflat()
  bedterhh ~ dflat()
  bagehh ~ dflat()
  bsexhh ~ dflat()

  sigmau<-1/precu
  sigmav<-1/precv
}
```

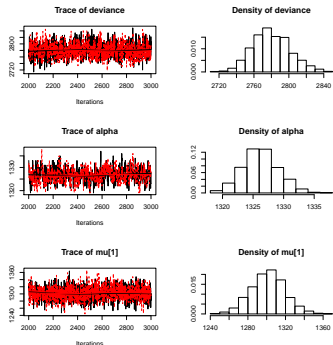


# Running the area level model

The following code reads that data for the area level model and two sets of initial values to start two different Markov Chains.

```
> library(R2WinBUGS)
> areamodelsp<-paste(getwd(), "/Income/models/area_modelsp.txt", sep="")
> dataareasp<-source("Income/WBdata/area_datasp.txt")$value
> neighbours<-source("Income/WBdata/spdata.txt")$value
> initsareasp1<-source("Income/area_inits/initssp-1.txt")$value
> initsareasp2<-source("Income/area_inits/initssp-2.txt")$value
> wdir<-paste(getwd(), "/Income_areasp", sep="")
> if(!file.exists(wdir)){dir.create(wdir)}
> BugsDir <- "/home/asdar/.wine/dosdevices/c:/Program Files/WinBUGS14"
> areasp<- bugs(data=c(dataareasp, neighbours),
+   inits=list(initsareasp1, initsareasp2),
+   working.directory=wdir,
+   parameters.to.save=c("mu", "sigmau", "u", "sigmav", "v", "alpha", "bedterhh"),
+   n.chains=2, n.iter=3000, n.burnin=2000, n.thin=1,
+   model.file=areamodelsp,
+   bugs.directory=BugsDir,
+   WINEPATH="/usr/bin/winepath")
```

# Assessment of the convergence



WinBUGS will create two files with the output of the Markov Chains that can be read. These chains can be plotted and the posterior density of the parameters displayed.

The posterior deviance of the model can be monitored as a guidance to the overall convergence.

```
> library(coda)
> chainsp1<-read.coda("Income_areasp/coda1.txt", "Income_
> chainsp2<-read.coda("Income_areasp/coda2.txt", "Income_
> chainssp<-mcmc.list(chainsp1, chainsp2)
> plot(chainssp[, c("deviance", "alpha", "mu[1]")])
```

# Spatial approach and Data 'missing' by design

- As in the non-Bayesian case, when there are survey data 'missing' from some areas the estimator in those areas reduces to a (Bayesian) synthetic estimator when random effect are not correlated
- If we use the CAR specification for the random effects, the values of the random effects can be predicted in the off-sample areas and the estimation is improved
- If there are too many off-sample areas the estimation procedure may be unstable
- However, a spatial structure can be imposed at a higher administrative level (for example, regions) so that information from several areas is put together to obtain an estimate of the regional random effect

# Spatial models at higher administrative levels

```
model
{
  for(i in 1:ns)
  {
    prece[i]<-1/desvar[s[i]]
    Y[s[i]] ~ dnorm(mu[s[i]], prece[i])

    u[s[i]] ~ dnorm(0, precu)
  }

  for(i in 1:N)
  {
    mu[i] <- alpha + bhhpers*HHPERS[i]+...+ bsexhh*SEXHH[i]+ u[i] +r[region[i]]
  }

  precu ~ dgamma (a0,b0)

  r[1:nregions] ~ car.normal(region.adj[], region.weights[], region.num[], precregion.v)

  precregion.v ~ dgamma (a1,b1)

  alpha ~ dflat()
  bhhpers ~ dflat()
  bhhemp1 ~ dflat()
  bedterhh ~ dflat()
  bagehh ~ dflat()
  bsexhh ~ dflat()

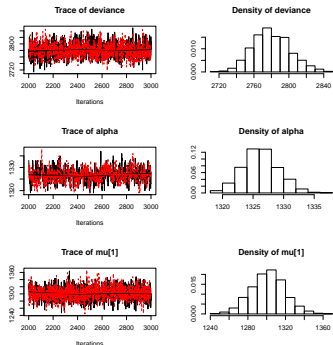
  sigmau<-1/precu
  sigmaaregion.v<-1/precregion.v
}
```

# Running the area level model

The following code reads that data for the area level model and two sets of initial values to start two different Markov Chains.

```
> library(R2WinBUGS)
> regionmodel<-paste(getwd(), "/Income/models-m/area_modelregion.txt", sep="")
> dataregion<-source("Income/WBdata-m/area_regiondata.txt")$value
> regneighbours<-source("Income/WBdata-m/spregiondata.txt")$value
> initsregion1<-source("Income/area_inits-m/initsregion-1.txt")$value
> initsregion2<-source("Income/area_inits-m/initsregion-2.txt")$value
> wdir<-paste(getwd(), "/Income_region", sep="")
> if(!file.exists(wdir)){dir.create(wdir)}
> BugsDir <- "/home/asdar/.wine/dosdevices/c:/Program Files/WinBUGS14"
> regionres<- bugs(data=c(dataregion, regneighbours),
+   inits=list(initsregion1, initsregion2),
+   working.directory=wdir,
+   parameters.to.save=c("mu", "sigmau", "u", "sigmaregion.v", "r", "alpha", "bedterhh"),
+   n.chains=2, n.iter=3000, n.burnin=2000, n.thin=1,
+   model.file=regionmodel,
+   bugs.directory=BugsDir,
+   WINEPATH="/usr/bin/winepath")
```

# Assessment of the convergence



WinBUGS will create two files with the output of the Markov Chains that can be read. These chains can be plotted and the posterior density of the parameters displayed.

The posterior deviance of the model can be monitored as a guidance to the overall convergence.

```
> library(coda)
> chainreg1<-read.coda("Income_region/coda1.txt", "Income
> chainreg2<-read.coda("Income_region/coda2.txt", "Income
> chainsreg<-mcmc.list(chainreg1, chainreg2)
> plot(chainsreg[, c("deviance", "alpha", "mu[1]")])
```

## Deviance Information Criterion (DIC)

- The DIC is similar to Akaike's Information Criterion:
- It is usually computed as

$$DIC = D(\hat{\theta}) + 2p_D$$

where  $D(\hat{\theta})$  is the deviance of the model evaluated at the posterior estimates of the parameters and  $p_D$  is the effective number of parameters.

- As with the AIC, lower values of the DIC are preferred
- The DIC is provided as part of the output generated by `bugs()`
- It cannot be used to compare models with different data, i.e., it cannot be used to compare area and unit level models

- The AEMSE can be used with Bayesian models using a reasonable posterior point estimate (i.e., mean or median)
- Other specific Bayesian criteria may be used
- It is worth noting that:
  - The AEMSE cannot usually be computed
  - The DIC can always be computed
  - In practise, we will usually rely on the AEMSE for simulation studies
  - and we will use the DIC to select models in a real situation
- How do the AEMSE and DIC behave?
- Gómez-Rubio et al. (2008) have shown that the way the AEMSE and the DIC rank the models is very similar



## Example: Spatial vs non-Spatial

```
> bayesres<-data.frame(AEMSE=c(NA, NA),  
+   DIC=c(areares$DIC, arearessp$DIC))  
> bayesres$AEMSE[1]<-mean((areares$mean$mu-0)^2)  
> bayesres$AEMSE[2]<-mean((arearessp$mean$mu-0)^2)  
> bayesres
```

	AEMSE	DIC
1	1762643	2935.84
2	1761129	2927.89

# Other applications of Bayesian estimators

- Temporal and spatio-temporal models can be easily developed by extended the models developed so far
- WinBUGS provides a suitable framework for writing and fitting new models
- The output (i.e., the chains generated during the Gibbs Sampling) can be further exploited for other purposes. In general, complex statistics based on the data can be computed from the output and inference can be made about them
- For example, areas can be 'classified' according to different criteria
- In the example of the average income per household:
  - Posterior relative ranks
  - Posterior probabilities of being among a, say, 10% of the areas with the lowest income
  - Posterior probabilities of being below the poverty line

# References

- BIAS Project. <http://www.bias-project.org.uk>
- Ghosh, M. and J. N. K. Rao (1994). Small area estimation: An appraisal. *Statistical Science* 9(1), 55–76.
- Gilks, WR, S. Richardson and D. Spiegelhalter (1995). Markov Chain Monte Carlo in Practice. John Wiley & Sons.
- Goldstein, H. and D. J. Spiegelhalter (1996). League tables and their limitations: Statistical issues in comparisons of institutional performance (with discussion). *Journal of the Royal Statistical Society, Series A* 159(3), 385–443.
- Gómez-Rubio, V., N. Best, S. Richardson and P. Clarke. Bayesian Statistics for Small Area Estimation. Submitted for publication.
- Lunn, D.J., Thomas, A., Best, N., and Spiegelhalter, D. (2000) WinBUGS – a Bayesian modelling framework: concepts, structure, and extensibility. *Statistics and Computing*, 10:325–337
- Rao, J. N. K. (2003). *Small Area Estimation*. John Wiley & Sons, Inc., Hoboken, New Jersey.
- Spiegelhalter, D. J., N. G. Best, B. P. Carlin, and A. van der Linde (2002). Bayesian measures of model complexity and fit (with discussion). *J. R. Statist. Soc. B* 64, 583–639.