Trabajo N.3

HUERTAS, ANTHONY[1,*]

[1] *Maestría en Estadística, Escuela de Posgrado, Pontificia Universidad Católica del Perú, Lima, Perú*
[*] *Cod: 20173728*

Compiled November 14, 2018

Profesora: Zaida Quiroz

---

1. BASE DE DATOS IRIS - CÓDIGO: IRIS.R

En primera instancia se necesita instalar el paquete INLA, de la forma que se indica en http://www.r-inla.org/download

Listing 1. Paquete INLA

```
install.packages("INLA", repos=c(getOption("repos"),
                 INLA="https://inla.r-inla-download.org/R/stable"), dep=TRUE)
inla.upgrade(testing=TRUE) # Actualizacion del paquete a su version mas reciente

library(INLA) # Cargar el paquete
```

En un modelo Gaussiano latente, sobre los datos observados $\mathbf{y} = (y_1, \ldots, y_n)$, se tiene que cada y_i se rige bajo una distribución en función de un parámetro ϕ_i, la cual cumple con

$$g(\phi_i) = \beta_0 + \sum_{j=1}^{m} \beta_j x_{ij} + \sum_{k=1}^{p} f_k(z_{ik}) \tag{S1}$$

donde $g(\cdot)$ representa la funcion link, $\mathbf{x_i} = (x_{i1}, \ldots, x_{im})$ y $\mathbf{z_i} = (x_{i1}, \ldots, x_{ip})$ covariables respecto al valor observado y_i, y los f_k funciones sobre los $\mathbf{z_i}$, las cuales son descritas por

Listing 2. Modelos latentes

```
names(inla.models()$latent)
```

```
 [1] "linear"      "iid"        "mec"        "meb"        "rgeneric"   "rw1"        "rw2"
 [8] "crw2"        "seasonal"   "besag"      "besag2"     "bym"        "bym2"       "besagproper"
[15] "besagproper2" "fgn"       "fgn2"       "ar1"        "ar1c"       "ar"         "ou"
[22] "generic"     "generic0"   "generic1"   "generic2"   "generic3"   "spde"       "spde2"
[29] "spde3"       "iid1d"      "iid2d"      "iid3d"      "iid4d"      "iid5d"      "2diid"
[36] "z"           "rw2d"       "rw2diid"    "slm"        "matern2d"   "copy"       "clinear"
[43] "sigm"        "revsigm"    "log1exp"    "logdist"
```

Fig. S1. Modelos latentes.

De forma similar se pueden observar las distintas distribuciones, descritas en **Listing 4** que deseemos modelar para los datos

Listing 3. Distribución de los datos

```
names(inla.models()$likelihood)
```

En los códigos posteriores, se trabajará sobre los *datos de Iris* (de Fisher o Anderson), especificados en ?iris en R, los cuales son datos que contienen a las variables: **Sepal Length** (Longitud de sépalo), **Sepal Width** (Ancho de sépalo), **Petal Length** (Longitud de pétalo) y **Petal Length** (Ancho de pétalo), medidas en centímetros, para una cantidad de 50 flores por cada 3 especies de iris, siendas estas especies: *Iris Setosa*, *Iris Versicolor* y *Iris Virgínica*.

Listing 4. base de datos Iris.

```
?iris
str(iris)
```

```
  [1] "poisson"                    "contpoisson"                 "qcontpoisson"
  [4] "cenpoisson"                 "gpoisson"                    "binomial"
  [7] "testbinomial1"              "pom"                         "gamma"
 [10] "gammacount"                 "qkumar"                      "qloglogistic"
 [13] "qloglogisticsurv"           "beta"                        "betabinomial"
 [16] "cbinomial"                  "nbinomial"                   "simplex"
 [19] "gaussian"                   "normal"                      "circularnormal"
 [22] "wrappedcauchy"              "iidgamma"                    "iidlogitbeta"
 [25] "loggammafrailty"            "logistic"                    "skewnormal"
 [28] "sn"                         "sn2"                         "gev"
 [31] "lognormal"                  "lognormalsurv"               "exponential"
 [34] "exponentialsurv"            "coxph"                       "weibull"
 [37] "weibullsurv"                "loglogistic"                 "loglogisticsurv"
 [40] "weibullcure"                "stochvol"                    "stochvolt"
 [43] "stochvolnig"                "zeroinflatedpoisson0"        "zeroinflatedpoisson1"
 [46] "zeroinflatedpoisson2"       "zeroinflatedbetabinomial0"   "zeroinflatedbetabinomial1"
 [49] "zeroinflatedbinomial0"      "zeroinflatedbinomial1"       "zeroinflatedbinomial2"
 [52] "zeroninflatedbinomial2"     "zeroninflatedbinomial3"      "zeroinflatedbetabinomial2"
 [55] "zeroinflatednbinomial0"     "zeroinflatednbinomial1"      "zeroinflatednbinomial1strata2"
 [58] "zeroinflatednbinomial1strata3" "zeroinflatednbinomial2"   "t"
 [6 names(x) rata"                 "nmix"                        "nmixnb"
 [64] "gp"                         "logperiodogram"
```

Fig. S2. Distribución de los datos.

```
'data.frame':   150 obs. of  5 variables:
 $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
 $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
 $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
 $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
 $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

Fig. S3. Datos de Iris.

Para este conjunto de datos, se modelará una regresión lineal, donde los datos observados sean representados por Petal.Length como en S1, de la forma

$$g(\phi) \quad = \quad \beta_0 + \beta_1 \times Petal.Width \tag{S2}$$

bajo una distribución **Gaussiana** sobre los datos observados, mediante el siguiente código

Listing 5. Modelo mediante INLA.

```
formula <- Petal.Length ~  1 + Petal.Width
output <- inla(formula, family="gaussian", data=iris)
summary(output)
```

Teniendose los siguientes resultados

```
Call:
"inla(formula = formula, family = \"gaussian\", data = iris)"

Time used:
 Pre-processing    Running inla Post-processing           Total
         2.7886          1.9413          0.3126          5.0424

Fixed effects:
              mean     sd 0.025quant 0.5quant 0.975quant   mode kld
(Intercept) 1.0836 0.0729     0.9402   1.0836     1.2268 1.0836   0
Petal.Width 2.2299 0.0514     2.1289   2.2299     2.3308 2.2299   0

The model has no random effects

Model hyperparameters:
                                         mean    sd 0.025quant 0.5quant 0.975quant  mode
Precision for the Gaussian observations 4.433 0.511      3.488    4.413       5.49 4.373

Expected number of effective parameters(std dev): 2.004(5e-04)
Number of equivalent replicates : 74.85

Marginal log-Likelihood:  -119.59
```

Fig. S4. Resumen del modelo.

Interpretaciones:

- La media de los efectos fijos son: 1.0836 (para β_0) y 2.2299 (para β_1). Así mismo se observa la desviación estándar y quantiles de estos efectos.
- La media de la precisión $\hat{\phi}$ para las observaciones Gaussianas es de 4.433.

Cabe indicar que los resultados del modelo para los efectos e hiperparámetros, se pueden visualizar por separado mediante

Listing 6. Modelo mediante INLA.

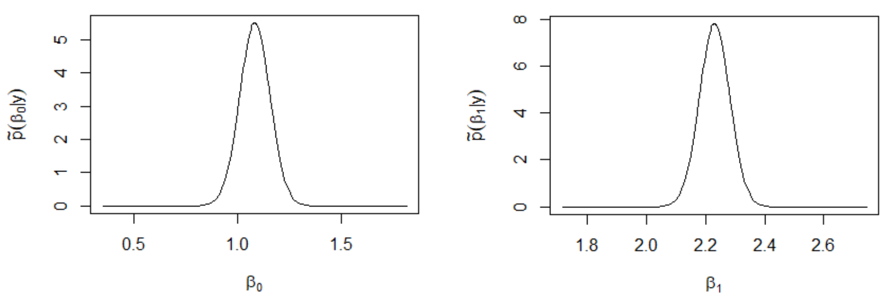
```
# ———— Efectos fijos
output$summary.fixed              # Resumen
names(output$marginals.fixed)     # Variables sobre las cuales actuan los efectos

# ———— Precision para las observaciones
output$summary.hyperpar           # Resumen
names(output$marginals.hyperpar) # Nombres los hiperparametros
```

cuyas distribuciones respectivas puedes ser obtenidas mediante

Listing 7. Distribución para los efectos β_0, β_1.

```
output$marginals.fixed
beta0_post <- output$marginals.fixed$Petal.Width
plot(beta0_post, type="l",xlab=expression(beta[0]), ylab=expression(tilde(p)(paste(beta[0],"|",y)
    )))
beta1_post <- output$marginals.fixed$Petal.Width
plot(beta1_post, type="l",xlab=expression(beta[1]), ylab=expression(tilde(p)(paste(beta[1],"|",y)
    )))
```



(a) Distribución a posteriori de β_0 (b) Distribución a posteriori de β_1

Fig. S5. Distribución de los efectos fijos.

Listing 8. Distribución de la precisión de las observaciones Gaussianas .

```
output$marginals.hyperpar
prec_post <- output$marginals.hyperpar$`Precision for the Gaussian observations`
plot(prec_post, type="l",xlab=expression(phi), ylab=expression(tilde(p)(paste(phi,"|",y))))
```

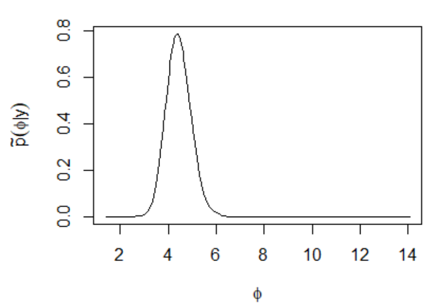


Fig. S6. Distribución de la precisión de las observaciones Gaussianas.

Pudiéndose además determinar Intervalos de confianza al 95%, en particular del efecto $\beta_1 \mid \mathbf{y}$ (distribución a posteriori de β_1) mediante funciones propias del paquete INLA,

Listing 9. IC para $\beta_1 \mid \mathbf{y}$.

```
IC_beta1 <- data.frame(inla.qmarginal(0.025,beta1_post),inla.qmarginal(0.975,beta1_post))
colnames(IC_beta1) <- c("2.5%","97.5%")
rownames(IC_beta1) <- c("beta1")
IC_beta1

# Lo anterior puede resumirse mediante esta funcion:
# inla.hpdmarginal(0.95,beta1_post)
```

	2.5%	97.5%
$\beta_1 \mid \mathbf{y}$	2.128938	2.330529

Table S1. Intervalo de confianza al 95% para $\beta_1 \mid \mathbf{y}$.

Ahora, sabiendo que $\sigma^2 = 1/\hat{\phi}$ (varianza del modelo Gaussiano), entonces

$$Var(\sigma^2) \quad = \quad E[(\sigma^2)^2] - E[\sigma^2]^2 = E[(1/\hat{\phi})^2] - E[\hat{\phi}]^2 \tag{S3}$$

por lo que podemos obtener la desviación estándar, además de la distribución respectiva, de este estimador σ^2, haciendo uso de la distribución obtenida para $\hat{\phi}$, como se ve a continuación

Listing 10. Distribución para $\sigma^2 \mid \mathbf{y}$.

```
inla.emarginal(fun=function(x) 1/x, marg=prec_post)
m1 <- inla.emarginal(function(x) 1/x, prec_post)
m2 <- inla.emarginal(function(x) (1/x)^2, prec_post)
sd <- sqrt(m2 - m1^2)
sd

# Grafica
plot.default(inla.tmarginal(function(x) 1/x,prec_post),
             type="l",xlab=expression(sigma^2),
             ylab=expression(tilde(p)(paste(sigma^2,"|",y))))
```

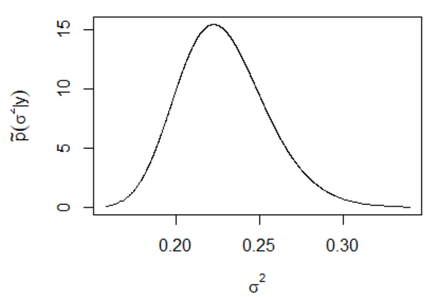


Fig. S7. Distribución de $\sigma^2 \mid \mathbf{y}$. Además, sd($\sigma^2 \mid \mathbf{y}$) = 0.02674407 (desviación estándar).

2. STEP BY STEP - CÓDIGO: INLASTEPBYSTEP.R

Se asume un modelo, bajo $\mathbf{y} = (y_1, \ldots, y_n)$, de la forma

$$y_i \mid \mu, \sigma^2 \sim N(\mu, \sigma^2) \tag{S4}$$

con distribuciones a priori:

$$\begin{aligned} \mu \quad &| \quad N(-3, 4^2) && \text{(S5)} \\ \phi = 1/\sigma^2 \quad &\sim \quad Gamma(1.6, 0.4) && \text{(S6)} \end{aligned}$$

El paquete INLA, trabaja de la forma siguiente

$$y_i \mid \theta, \phi \sim N(\eta_i, \sigma^2) \tag{S7}$$

con $\theta = \eta_i = \mu$, $\phi = 1/\sigma^2$.
Se presentan datos provenientes de una simulación por parte del modelo en estudio.

Listing 11. Datos

```
y <- c(1.2697,7.7637,2.2532,3.4557,4.1776,6.4320,-3.6623,7.7567,
5.9032,7.2671,-2.3447,8.0160,3.5013,2.8495,0.6467,3.2371,
5.8573,-3.3749,4.1507,4.3092,11.7327,2.6174,9.4942,-2.7639,
-1.5859,3.6986,2.4544,-0.3294,0.2329,5.2846)
n <- length(y)
ybar <- mean(y)

# Valores que definen las distribuciones a prioris
mu0 <- -3
sigma2_0 <- 4
a <- 1.6
b <- 0.4
```

Teniéndose las distribuciones

$$\begin{aligned} \mathbf{y} \mid \theta, \phi \quad &\sim \quad \prod_{i=1}^{n} f(y_i \mid\mid \theta, \phi) \\ \theta \quad &\sim \quad N(-3, 4^2) \\ \phi = 1/\sigma^2 \quad &\sim \quad Gamma(1.6, 0.4) \\ \theta \mid \phi, \mathbf{y} \quad &\sim \quad N\left(\frac{\phi \sum_{i=1}^{n} y_i - 3/4^2}{n\phi + 1/4^2}, \frac{1}{n\phi + 1/4^2} \right) \end{aligned}$$

es de interés determinar la distribución a posteriori de ϕ, representada por

$$\begin{aligned} p(\phi \mid \mathbf{y}) \quad &\propto \quad \frac{p(y \mid \theta, \phi) p(\theta) p(\phi)}{p(\theta \mid \phi, \mathbf{y})} \\ &= \quad \frac{1}{(2\pi\sigma^2)^{-1/2}} p(y \mid \theta, \phi) p(\theta) p(\phi) && \text{(S8)} \end{aligned}$$

Por tanto, se logra obtener la siguiente representación recursiva

$$p(\phi^{(k+1)} \mid \mathbf{y}) \propto \frac{1}{(2\pi\sigma^{2(k)})^{-1/2}} p(\theta^{(k)}, 1/\phi^{(k)}) p(\theta^{(k)}) p(\phi^{(k)}) \tag{S9}$$

donde

$$\begin{aligned} \theta^{(k)} \quad &= \quad \frac{\phi^{(k)} \sum_{i=1}^{n} y_i - 3/4^2}{n\phi^{(k)} + 1/4^2} \\ \sigma^{2(k)} \quad &= \quad \frac{1}{n\phi^{(k)} + 1/4^2} \end{aligned}$$

Listing 12. Distribucion a posteriori de la precisión phi.

```
H <- 25
theta.min <- 0.001
theta.max <- 0.5
theta.grid <- seq(theta.min,theta.max,length.out=H)
hprior <- dgamma(theta.grid,shape=a,rate=b)

x.n <- sigma2.n <- lik <- num <- den <- prior <- c()
for (h in 1:H) {
        x.n[h]   <- (theta.grid[h]*n*ybar + mu0/sigma2_0) / (theta.grid[h]*n + 1/sigma2_0)
        sigma2.n[h]  <- 1 / (n*theta.grid[h] + 1/sigma2_0)
        prior[h]  <- dnorm(x.n[h], mu0, sd=sqrt(sigma2_0))
        lik[h]    <- prod(dnorm(y, x.n[h], sd=1/sqrt(theta.grid[h])))
        num[h]    <- hprior[h] * prior[h] * lik[h]
        den[h]    <- dnorm(x.n[h], x.n[h], sd=sqrt(sigma2.n[h]))
}

# Distribucion marginal (no normalizada para la precision phi)
post.theta <- num/den

plot(theta.grid, post.theta, xlab=expression(phi), ylab="",pch=20,cex=1.3, ylim=range(post.theta,
    spline(theta.grid,post.theta)$y))
lines(smooth.spline(theta.grid,post.theta,all.knots=T))
```

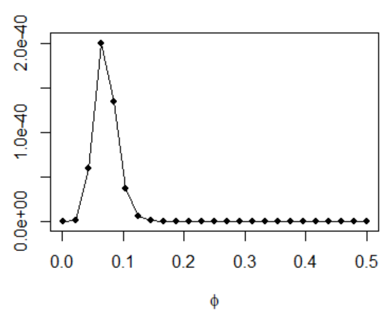


Fig. S8. Distribución de $\phi \mid \mathbf{y}$.

El normalizar la distribución de $\phi \mid \mathbf{y}$, se determina por

Listing 13. Distribucion a posteriori (normalizada) de la precisión phi.

```
f.theta <- approxfun(theta.grid, post.theta, yleft=min(theta.grid), yright=max(theta.grid))
const <- integrate(f.theta, min(theta.grid), max(theta.grid))
post.theta <- post.theta/const$value

plot(theta.grid, post.theta, xlab=expression(theta), ylab="",pch=20,cex=1.3, ylim=range(post.
    theta,spline(theta.grid,post.theta)$y))
lines(smooth.spline(theta.grid,post.theta,all.knots=T))
```

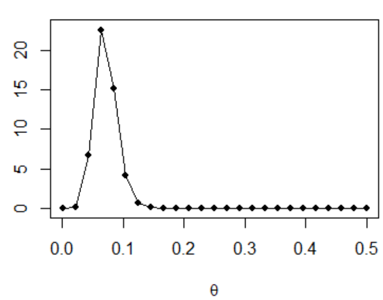


Fig. S9. Distribución normalizada de $\phi \mid \mathbf{y}$.

Observandose aún una cola muy extensa a la derecha de la distribución.
Ahora, lo que se busca es generar la distribución de $\theta \mid \phi, \mathbf{y}$ determinada por

$$\theta^{(k+1)} \mid \phi, \mathbf{y} \quad \propto \quad p(\theta^{(k)}, 1/\phi^{(k)}) \tag{S10}$$

Luego mediante una sumatoria finita con pesos respectivos para $p(\theta \mid \phi, \mathbf{y})$, tenemos que

$$p(\theta^{(k+1)} \mid \mathbf{y}) \quad \propto \quad \sum_j p(\theta^k \mid \phi^{(j)}, \mathbf{y}) p(\phi^{(j)} \mid \mathbf{y}) \underbrace{\frac{1}{\sum_j p(\phi^{(j)} \mid \mathbf{y})}}_{Pesos} \tag{S11}$$

Listing 14. Distribución para $\theta \mid \phi, \mathbf{y}$.

```
J <- 50
min.x <- -8
max.x <- 5
x.grid <- seq(min.x,max.x,length.out=J)

# Distribucion condicional theta | phi,y
full.cond.x <- matrix(NA,J,H)
for (j in 1:J) {
        for (h in 1:H) {
                full.cond.x[j,h] <- dnorm(x.grid[j], x.n[h], sd=sqrt(sigma2.n[h]))
        }
}

plot(x.grid,full.cond.x[,1],t="l",lwd=.7,ylab="",xlab=expression(theta), ylim=range(full.cond.x),
    cex=1.3)
for (h in 1:H) {
        lines(x.grid,full.cond.x[,h],lwd=.7)
}
```

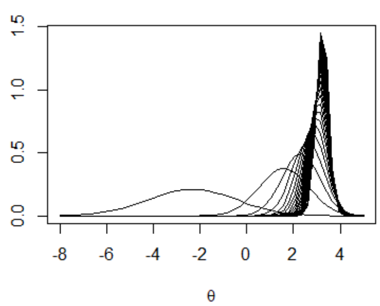


Fig. S10. Distribución para $\theta \mid \phi, \mathbf{y}$.

Listing 15. Distribución conjunta de $\theta, \phi \mid \mathbf{y}$.

```
# Insercion de los pesos
Delta <- 1/sum(post.theta)
joint.post.x.theta <- matrix(NA,J,H)
for (h in 1:H) {
        joint.post.x.theta[,h] <- full.cond.x[,h] * post.theta[h] * Delta
}

# Integracion bajo phi para obterner la distribucion marginal posterior para theta
marg.post.x <- rowSums(joint.post.x.theta)

rg <- c( -0.839603 , 5.836238)
plot(x.grid,marg.post.x,t="l",xlab=expression(theta),ylab="",
xlim=c(rg[1],max(x.grid)),lty=2,lwd=3)
for (h in 1:H) {
        lines(x.grid,joint.post.x.theta[,h],lwd=.5)
}
```

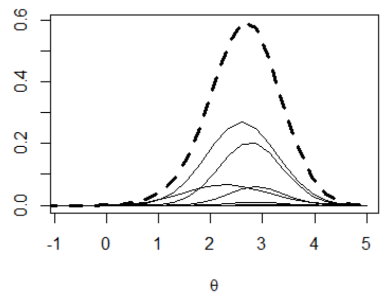


Fig. S11. Distribución conjunta de $\theta, \phi \mid \mathbf{y}$.

En Fig S11, visualizamos la distribución de $\theta \mid \mathbf{y}$ como una curva punteada que viene determinada como la suma de las distribuciones a posteriori de $\theta, \phi \mid \mathbf{y}$ multiplicadas por sus pesos respectivos y representadas cada una por las curvas solidas.