

Modelos Estadísticos Avanzados Aplicados a la Investigación en Salud y Ambiente

Facultad de Ciencias Exactas y Naturales, Universidad Nacional de
Buenos Aires

*Raúl E. Macchiavelli, Ph.D. (Departamento de Ciencias Agroambientales,
Universidad de Puerto Rico, Mayaguez), Pedro A. Torres-Saavedra,
Ph.D., (Departamento de Ciencias Matemáticas, Universidad de Puerto
Rico, Mayaguez)*

2018-08-06

Índice general

Información Importante	4
Introducción	5
Pre-requisitos	5
Agenda del Curso	6
1. Modelos Lineales Mixtos	8
1.1. Modelos Lineales	8
1.2. Modelo Lineal (Clásico)	8
1.3. Modelos Mixtos	21
1.4. Estimación en Modelos Lineales Mixtos	34
1.5. Estimación por Máxima Verosimilitud (<i>ML</i>)	36
1.6. Estimación por Máxima Verosimilitud Restringida (<i>REML</i>)	36
1.7. Propiedades del Estimador de Efectos Fijos	37
1.8. Predicción de Efectos Aleatorios	37
1.9. Prueba de Hipótesis para Efectos Fijos	38
1.10. Prueba de verosimilitud para Efectos Fijos	39
1.11. Prueba de verosimilitud para Varianzas	39
2. Modelos Lineales Generalizados y Mixtos	54
2.1. Modelos Mixtos	54
2.2. Modelos Lineales Generalizados (MLG)	54
2.3. Regresión logística para datos binarios o proporciones binomiales	56
2.4. Regresión Poisson	63
2.5. Estimación por Máxima Verosimilitud	69
2.6. Sobredispersión en modelos lineales generalizados	69
2.7. Cuasiverosimilitud	71
2.8. Distribución Comway-Maxwell-Poisson	77
2.9. Modelos Lineales Generalizados Mixtos	84
2.10. Modelo LogLineal Poisson-Normal (o Poisson-Lognormal)	92
2.11. Modelo Negativo Binomial-Normal	93
3. Modelado de la Dependencia Temporal y Espacial. Datos Longitudi-	

nales y Curvas de Crecimiento	102
3.1. Datos Longitudinales	102
3.2. Datos Espaciales	103
3.3. Análisis de datos correlacionados en el tiempo	104
3.4. Estructuras de correlación espacial	106
3.5. Enfoques para datos longitudinales	107
3.6. Selección de la estructura de covarianza	107
4. Introducción a los Modelos No Lineales Mixtos. Modelos Semiparamétricos y GAMM	128
4.1. Modelos Estadísticos	128
4.2. Modelos No Lineales	128
4.3. Familias de funciones no lineales: curvas de crecimiento	129
4.4. Ejemplos de otras familias de funciones no lineales	130
4.5. Ejemplos de Modelos No Lineales Mixtos	130
4.6. Definición del Modelo	131
4.7. Objetivos de la inferencia	133
4.8. Inferencia sujeta específica o promedio poblacional	133
4.9. Inferencia Basada en Verosimilitud	134
4.10. Estimación por ML (<i>Maximum Likelihood</i>)	135
4.11. Métodos de estimación	135
4.12. Ejemplo de modelos no lineales: Teofilina	138
4.13. Modelos Semiparamétricos	140
5. Introducción a Estadística Bayesiana. Principios de Inferencia Bayesiana, MCMC, Gibbs Sampling	154
5.1. Teorema de Bayes	155
5.2. Inferencia Bayesiana	156
5.3. Distribuciones a priori	158
5.4. Métodos de Monte Carlo con Cadenas de Markov (MCMC)	162
5.5. Diagnósticos de convergencia	163
5.6. Inferencia Bayesiana	165
5.7. Comparación de modelos	166
5.8. Introducción a JAGS	166
6. Modelos Lineales y Modelos Lineales Generalizados Bayesianos	181
6.1. Modelos Lineales Bayesianos	181
6.2. Modelos Lineales Generalizados (MLG) Bayesianos	192
7. Introducción a INLA (Integrated Nested Laplace Approximation)	199
7.1. Modelos Gaussianos Latentes (MGL)	199
7.2. <i>Integrated Nested Laplace Approximation (INLA)</i>	203
7.3. Modelos Lineales en R-INLA	204

7.4. Modelos Lineales Mixtos en R-INLA	213
7.5. Modelos Espaciales: Cáncer	221
8. Temas Especiales	226
8.1. Selección de Variables usando LASSO Bayesiano	226
8.2. Suavizamiento en R-INLA	230
8.3. Datos Geoespaciales en R-INLA: Malaria en Gambia	230

Información Importante

Esta versión de las notas de clase fueron elaboradas usando el paquete **bookdown** (basado en R Markdown y **knitr**) (Xie, 2018). Los análisis fueron corridos en el programa estadístico R (R Core Team, 2016) y JAGS (Plummer, 2003).

Estas notas sirven de material de apoyo para el material que se cubrirá en el curso: **Modelos Estadísticos Avanzados Aplicados a la Investigación en Salud y Ambiente** a llevarse a cabo del 6 al 16 de agosto del 2018 en la **Facultad de Ciencias Exactas y Naturales, Universidad Nacional de Buenos Aires**.

Este curso ha sido financiado por el **Centro Latinoamericano de Formación Interdisciplinaria (CELFI)**, Ministerio de Ciencia, Tecnología e Innovación Productiva, República Argentina.

Esta es una versión que podría contener errores. Cualquier comentario por favor enviarlo a pedro.torres14@upr.edu o raul.macchiavelli@upr.edu.

Esta versión de las notas se considera PRELIMINAR.

Introducción

Pre-requisitos

Se recomienda usar Google Chrome para ver las notas.

Antes de comenzar a cubrir el material del curso es necesario que instalen en sus computadoras personales los siguientes programas y librerías que serán usadas para manipular los datos y ajustar los modelos estadísticos durante el curso.

1. Instale las últimas versiones de R y RStudio.
2. Instale las siguientes librerías de R.

```
# Instala librerías necesarias
install.packages(c("readr", "dplyr", "ggplot2", "lattice",
                  "haven", "GGally", "nlme", "nlmeU", "rjags",
                  "compGLM", "lme4", "COMPOissonReg", "sjPlot",
                  "faraway", "R2jags"),
                repos="https://cloud.r-project.org")
```

```
# Carga librerías necesarias
library(dplyr, readr, ggplot2, lattice, haven, GGally,
        nlme, nlmeU, faraway, multcomp, lme4, rjags, compGLM,
        COMPOissonReg, sjPlot, gam, splines, R2jags)
```

Las demás librerías que se usarán en el curso se instalarán a medida que avancemos con el material.

3. Instale el programa **JAGS (Just Another Gibbs Sampler)**. Este programa sirve para ajustar modelos Bayesianos jerárquicos usando el método de Monte Carlo con Cadenas de Markov (MCMC). En este enlace encontrará los manuales, ejemplos y las notas de un curso, entre otras cosas. Para bajar los instaladores de la versión 4.x de JAGS del sistema operativo Windows o Mac OS X ir a este enlace. Baje el archivo **JAGS-4.3.0.exe** y haga doble click sobre el archivo empezar la instalación (si usa el sistema operativo Mac OS X baje el archivo **JAGS-4.3.0.dmg**).

4. Asegúrese de tener instalada las librerías `rjags` y `R2jags` de R. Esta librería servirá para correr JAGS dentro de R.
5. Instale las librerías R-INLA y `brinla`:

```
install.packages("INLA", repos=c(getOption("repos"),
                                INLA="https://inla.r-inla-download.org/R/stable"),
                dep=TRUE)
install.packages("devtools",
                repos="https://cloud.r-project.org")
library(devtools)
install_github("julianfaraway/brinla")
```

Agenda del Curso

1. Día 1 – Lunes 6 de agosto (Capítulo 1)

- 9:00 a 10:00 hs.: Clase teórica 1 (abierta). Fundamentos de los modelos lineales mixtos generales y generalizados, ventajas y desventajas. Principales aplicaciones.
- 10:30 a 12:30 hs.: Clase teórica 1 (Continuación). Modelos con factores aleatorios para datos normales. Modelos marginales versus modelos sujetos específicos. Modelos para diseños jerárquicos (multinivel). Método REML.
- 14:00 a 18:00 hs. Práctica 1. Resolución de casos de estudio.

2. Día 2: Martes 7 de agosto (Capítulo 2)

- 9:00 a 12:30 hs.: Clase teórica 2. Modelos lineales generalizados y mixtos con distribución (Poisson, binomial y binomial negativa). Modelado de sub y sobredispersión. Inferencia con métodos de verosimilitud y pseudo-verosimilitud.
- 14:00 a 18:00 hs. Práctica 2. Resolución de casos de estudio.

3. Día 3 - Miércoles 8 de agosto (Capítulo 3)

- 9:00 a 12:30 hs. Clase teórica 3. Modelado de la dependencia temporal y espacial. Datos longitudinales y curvas de crecimiento. Aplicaciones a datos normales. Introducción a modelos espaciales para datos binomiales y Poisson.
- 14:00 a 18:00 hs. Práctica 3. Resolución de casos de estudio

4. Día 4 - Jueves 9 de agosto (Capítulo 4)

- *9:00 a 12:30 hs.* Clase teórica 4. Nuevos desafíos. Introducción a los modelos no lineales mixtos. Modelos semiparamétricos y GAMM.
- *14:00 a 18:00 hs.* Práctica 4. Resolución de casos de estudio

5. Día 5 - Viernes 10 de agosto (Capítulo 5)

- *9:00 a 12:30 hs.* Clase teórica 6. Introducción a estadística Bayesiana. Principios de Inferencia Bayesiana, MCMC, Gibbs sampling (abierta).
- *14:00 a 18:00 hs.* Práctica 5. Introducción a JAGS, diagnósticos de convergencia. Comparación entre inferencia Bayesiana y frecuentista. Resolución de casos de estudio.

6. Día 6- Lunes 13 de agosto (Capítulo 6)

- *9:00 a 12:30 hs.* Clase teórica 6. Modelos lineales y modelos lineales generalizados Bayesianos. Análisis de sensibilidad, predicción y selección de modelos.
- *14:00 a 18:00 hs.* Práctica 6. Resolución de casos de estudio

7. Día 7 - Martes 14 de agosto (Capítulo 7)

- *12.00 a 14.00 hs.* Clase teórica 7. Introducción a INLA (Integrated Nested Laplace Approximation). Aplicaciones a modelos espaciales.
- *14:00 a 18:00 hs.* Práctica 7. Resolución de casos de estudio

8. Día 8 - Miércoles 15 de agosto (Capítulo 8)

- *9:00 a 12:30 hs.* Clase teórica 8. Temas Especiales: INLA vs. MCMC. Selección de modelos en datos de gran dimensión, datos faltantes, etc.
- *14:00 a 18:00 hs.* Práctica 8. Uso de R-INLA. Resolución de casos de estudio

Capítulo 1

Modelos Lineales Mixtos

1.1. Modelos Lineales

- Tradicionalmente ha sido la manera de modelar la relación entre variables independientes (numéricas o categóricas) y una variable dependiente (numérica)
- Regresión lineal simple, múltiple y polinomial
- Análisis de varianza para uno o más factores
- Análisis de covarianza
- Estimación basada en mínimos cuadrados
- Pruebas válidas si los supuestos se cumplen (validez de la ecuación para la media, independencia, normalidad, varianzas homogéneas)

1.2. Modelo Lineal (Clásico)

Suponga que tiene n observaciones independientes. Para cada individuo i se observa una variable respuesta Y_i y p covariables $\mathbf{x}_i = (x_{i1}, \dots, x_{ip})^t$. Note que \mathbf{x}_i es un vector columna de dimensión $p < n$. En el modelo lineal clásico se asume que

$$Y_i = \mathbf{x}_i^t \beta + \varepsilon_i, \quad (1.1)$$

donde β es un vector columna con p parámetros y ε_i satisface que $\varepsilon_i \stackrel{i.i.d.}{\sim} N(0, \sigma^2)$ (i.i.d. significa independiente e idénticamente distribuidos). Los supuestos del modelo anterior (1.1) implican que:

1. $E(Y_i | \mathbf{x}_i) = \mathbf{x}_i^t \beta$.

2. $Var(Y_i|\mathbf{x}_i) = \sigma^2$.
3. $Y_i|\mathbf{x}_i \stackrel{indep.}{\sim} N(\mathbf{x}_i^t\beta, \sigma^2)$.

Note que para dos individuos i y j , $i \neq j$, se tiene que $Cov(Y_i, Y_j) = 0$ ya que asumimos errores independientes (es más, Y_i y Y_j son independientes). Esto nos indica que los errores son los que definen la distribución de la respuesta dado el modelo lineal aditivo.

Ahora vamos a plantear el modelo anterior usando matrices. Defina la siguiente matriz del modelo de dimensión $n \times p$

$$\mathbf{X}_{n \times p} = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,p} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,p} \end{pmatrix}.$$

Cada fila en \mathbf{X} representa el vector de covariables de cada uno de los n individuos. Es decir,

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^t \\ \mathbf{x}_2^t \\ \vdots \\ \mathbf{x}_n^t \end{pmatrix}.$$

De esta manera el modelo (1.1) se puede escribir como

$$\mathbf{Y} = \mathbf{X}\beta + \varepsilon, \tag{1.2}$$

donde $\mathbf{Y} = (Y_1, \dots, Y_n)^t$ y $\varepsilon = (\varepsilon_1, \dots, \varepsilon_n)^t$ es un vector columna con los n errores del modelo. También se asume que $\varepsilon \sim N(\mathbf{0}, \sigma^2 \mathbf{I}_n)$, donde \mathbf{I}_n es la matriz identidad de orden n . Los supuestos para el modelo (1.2) implican que:

1. $E(\mathbf{Y}|\mathbf{X}) = \mathbf{X}\beta$.
2. $Var(\mathbf{Y}|\mathbf{X}) = \sigma^2 \mathbf{I}_n$.
3. $\mathbf{Y}|\mathbf{X} \sim N(\mathbf{X}\beta, \sigma^2 \mathbf{I}_n)$.

1.2.1. Modelo de Efectos Fijos

■ Modelo de Clasificación

$$E(y_{ij}) = \mu + \tau_i.$$

■ Modelo de Regresión Simple

$$E(y_{ij}) = \mu_i = \beta_0 + \beta_1 x_i$$

En ambos casos

$$y_{ij} \stackrel{indep.}{\sim} N(\mu_i, \sigma^2) \leftarrow \textit{Inferencia.}$$

Ejemplo 1.1 (Esponjas (Qian, 2010)). Se realizó un experimento para estudiar la interacción entre esponjas y mangle rojo. Se deseaba probar si el mangle se beneficiaba del crecimiento de organismos alrededor de las raíces.

Parcelas con mangle rojo se asignaron aleatoriamente a cuatro tratamientos: 1. Control sin manipular.

2. Esponja plástica colocada alrededor de la raíz.
3. Colonias de esponja roja (*Tedania ignis*) y transplantadas a las raíces.
4. Colonias de esponja púrpura (*Haliclona implexiformis*) transplantadas a la raíces.

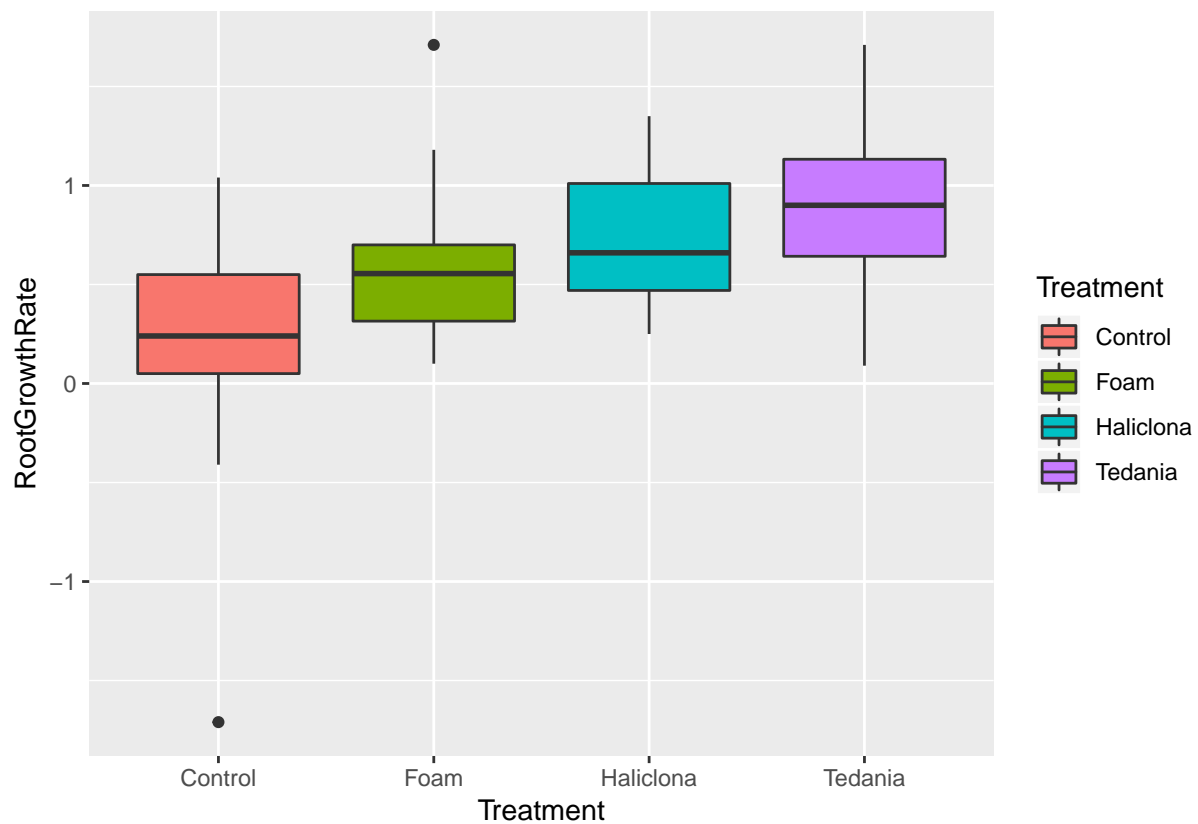
La variable de respuesta es crecimiento de las raíces (mm/día). Los datos se tomaron en cuatro localidades diferentes, por ahora ignoraremos esto.

```
head(SpongeData)
```

```
# A tibble: 6 x 3
  Location Treatment RootGrowthRate
  <chr>      <chr>          <dbl>
1 lcn       Control          0.240
2 lcn       Control         -0.0500
3 lcn       Control          0.190
4 etb       Control          1.02
5 lcs       Control          0.840
6 bbs       Control          0.110
```

```
library(ggplot2)
```

```
ggplot(SpongeData, aes(x=Treatment, y=RootGrowthRate, fill=Treatment)) + geom_boxplot
```



El modelo para analizar este conjunto de datos es el siguiente:

$$Y_i = \mu + \tau_i x_i + e_i, \quad i = 2, 3, 4.$$

$$x_i = \begin{cases} 1, & \text{Tratamiento} = i \\ 0, & \text{Tratamiento} \neq i. \end{cases}$$

$$e_{ij} \stackrel{iid}{\sim} N(0, \sigma^2).$$

$$E(\mathbf{y}) = \mathbf{X}\beta, \quad \mathbf{y} \sim N(E(\mathbf{y}), \sigma^2 \mathbf{I}).$$

```
mod <- lm(RootGrowthRate ~ Treatment, data=SpongeData)
summary(mod)
```

Call:

```
lm(formula = RootGrowthRate ~ Treatment, data = SpongeData)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.94714	-0.26905	-0.03754	0.25893	1.11850

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	0.2371	0.1008	2.352	0.02156	*
TreatmentFoam	0.3544	0.1443	2.455	0.01665	*
TreatmentHaliclona	0.4911	0.1507	3.258	0.00175	**
TreatmentTedania	0.6764	0.1594	4.244	6.82e-05	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.462 on 68 degrees of freedom

Multiple R-squared: 0.2327, Adjusted R-squared: 0.1989

F-statistic: 6.875 on 3 and 68 DF, p-value: 0.0004111

```
anova(mod)
```

Analysis of Variance Table

Response: RootGrowthRate

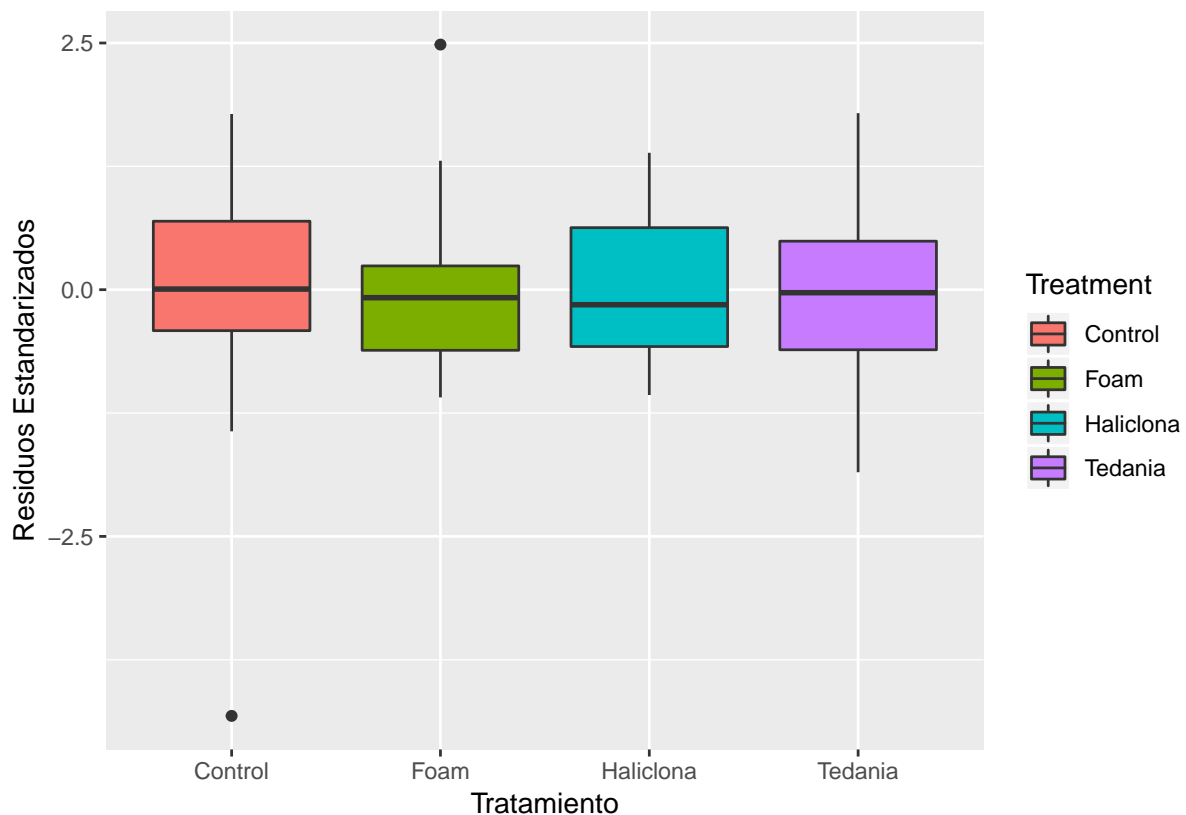
	Df	Sum Sq	Mean Sq	F value	Pr(>F)	
Treatment	3	4.4019	1.46729	6.8749	0.0004111	***
Residuals	68	14.5131	0.21343			

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Ejercicio 1.1. Escribir las componentes de cada uno de los siguientes objetos:

$$\mathbf{y} = \begin{bmatrix} \\ \\ \\ \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} \\ \\ \\ \end{bmatrix}, \quad \boldsymbol{\beta} = \begin{bmatrix} \\ \\ \\ \end{bmatrix}, \quad \mathbf{e} = \begin{bmatrix} \\ \\ \\ \end{bmatrix}$$

```
par(mfrow=c(2,2))
plot(fitted(mod), rstandard(mod), xlab="Ajustados", ylab="Residuos Estandarizados")
SpongeData$stdres <- rstandard(mod)
qqnorm(SpongeData$stdres)
qqline(SpongeData$stdres)
ggplot(SpongeData, aes(x=Treatment, y=stdres, fill=Treatment)) +
  geom_boxplot() + xlab("Tratamiento") +
  ylab("Residuos Estandarizados")
```



Ejemplo 1.2 (Actividad Sexual en Moscas). Estos datos provienen de un estudio de actividad sexual en machos de *Drosophila* (Faraway, 2005). Se usaron 125 moscas, divididas en 5 grupos de actividad sexual y se midió la longevidad. Como se conoce que la longevidad se relaciona con el tamaño del tórax, también se midió el tamaño del tórax en cada mosca. Los tratamientos fueron los siguientes:

1. Cada macho se mantuvo aislado (*isolated*).
2. Cada macho se mantuvo con una hembra virgen cada día (*low*).
3. Cada macho se mantuvo con 8 hembras vírgenes cada día (*high*).
4. Cada macho se mantuvo con una hembra preñada cada día (*one*).
5. Cada macho se mantuvo con 8 hembras preñadas cada día (*many*).

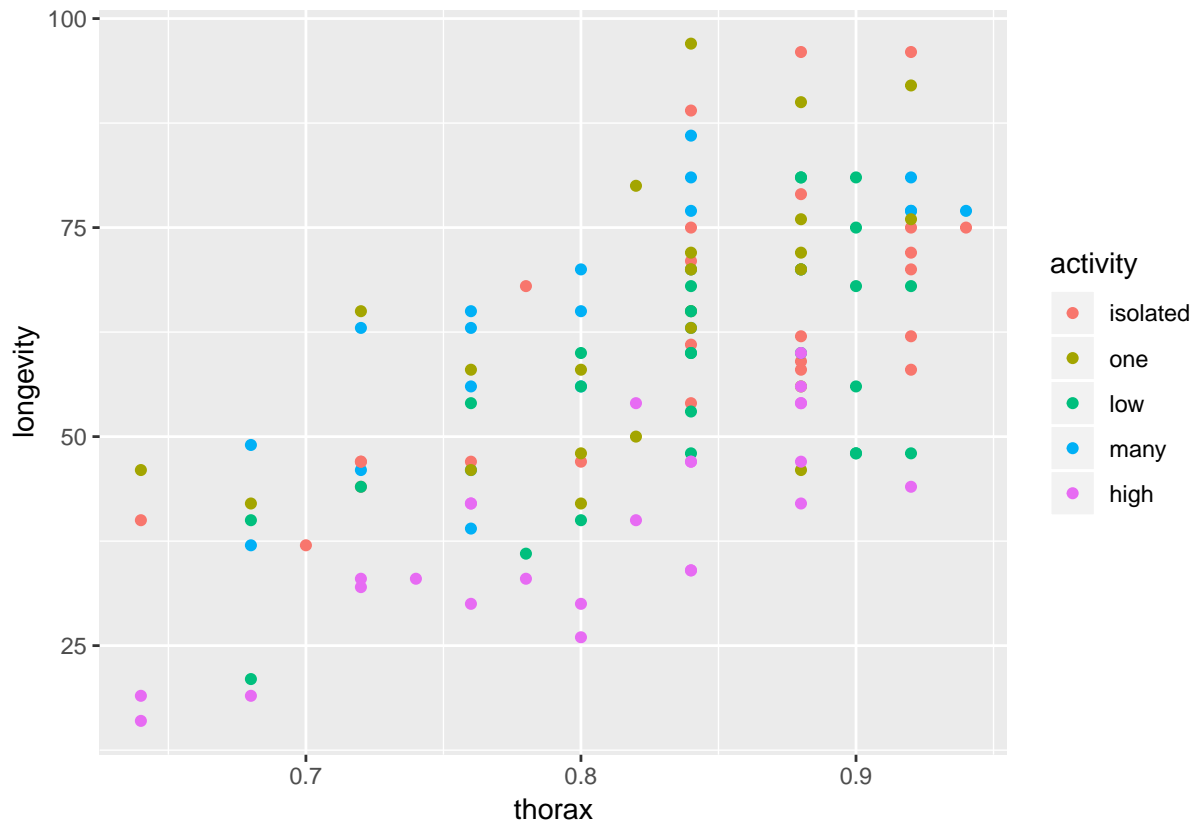
El objetivo es comparar la longevidad entre los niveles de actividad sexual.

```
library(faraway)
data(fruitfly)
head(fruitfly)
```

```
  thorax longevity activity
1   0.68         37    many
```

2	0.68	49	many
3	0.72	46	many
4	0.72	63	many
5	0.76	39	many
6	0.76	46	many

```
ggplot(fruitfly, aes(x=thorax, y=longevity, colour=activity)) +  
  geom_point()
```



```
mod1 <- lm( longevity ~ thorax + activity + thorax:activity,  
            data=fruitfly)  
summary(mod1)
```

Call:

```
lm(formula = longevity ~ thorax + activity + thorax:activity,  
    data = fruitfly)
```

Residuals:

Min	1Q	Median	3Q	Max
-25.9509	-6.7296	-0.9103	6.1854	30.3071

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-50.2420	21.8012	-2.305	0.023 *
thorax	136.1268	25.9517	5.245	7.27e-07 ***
activityone	6.5172	33.8708	0.192	0.848
activitylow	-7.7501	33.9690	-0.228	0.820
activitymany	-1.1394	32.5298	-0.035	0.972
activityhigh	-11.0380	31.2866	-0.353	0.725
thorax:activityone	-4.6771	40.6518	-0.115	0.909
thorax:activitylow	0.8743	40.4253	0.022	0.983
thorax:activitymany	6.5478	39.3600	0.166	0.868
thorax:activityhigh	-11.1268	38.1200	-0.292	0.771

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.71 on 114 degrees of freedom

Multiple R-squared: 0.6534, Adjusted R-squared: 0.626

F-statistic: 23.88 on 9 and 114 DF, p-value: < 2.2e-16

```
anova(mod1)
```

Analysis of Variance Table

Response: longevity

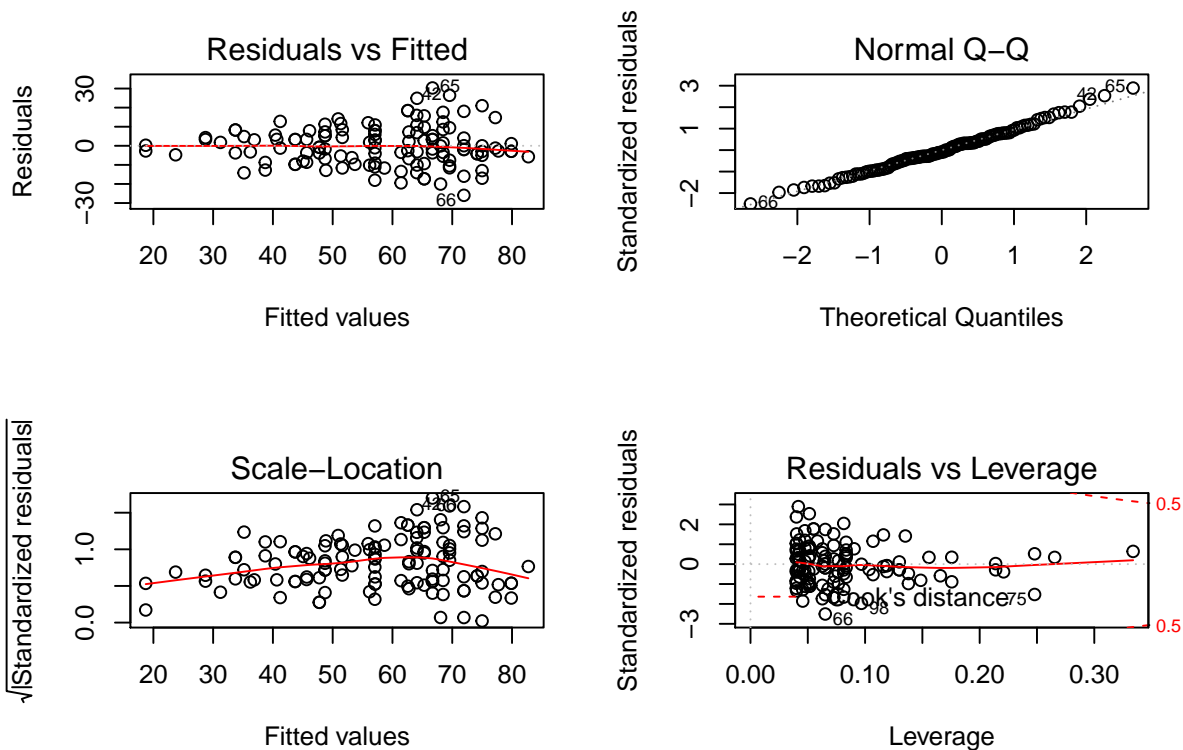
	Df	Sum Sq	Mean Sq	F value	Pr(>F)
thorax	1	15003.3	15003.3	130.733	< 2.2e-16 ***
activity	4	9634.6	2408.6	20.988	5.503e-13 ***
thorax:activity	4	24.3	6.1	0.053	0.9947
Residuals	114	13083.0	114.8		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

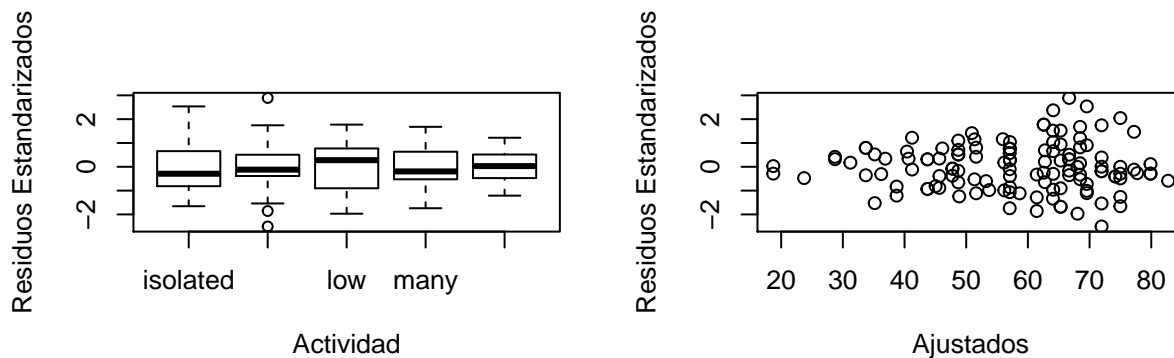
```
# Algunos gráficos diagnóstico
```

```
par(mfrow=c(2,2))
```

```
plot(mod1)
```

```
fruitfly$stdres <- rstandard(mod1)
fruitfly$fitted <- fitted(mod1)
plot(stdres ~ activity, data=fruitfly, xlab="Actividad",
      ylab="Residuos Estandarizados")
plot(stdres ~ fitted, data=fruitfly, xlab="Ajustados",
      ylab="Residuos Estandarizados")
```



```
# Matriz del modelo X
head( model.matrix(mod1) )
```

```
(Intercept) thorax activityone activitylow activitymany activityhigh
1           1    0.68             0             0             1             0
2           1    0.68             0             0             1             0
3           1    0.72             0             0             1             0
```

4	1	0.72	0	0	1	0
5	1	0.76	0	0	1	0
6	1	0.76	0	0	1	0
thorax:activityone thorax:activitylow thorax:activitymany						
1		0		0		0.68
2		0		0		0.68
3		0		0		0.72
4		0		0		0.72
5		0		0		0.76
6		0		0		0.76
thorax:activityhigh						
1		0				
2		0				
3		0				
4		0				
5		0				
6		0				

```
# Elimina término de interacción
mod2<-update(mod1, . ~ . - thorax:activity)
anova(mod2)
```

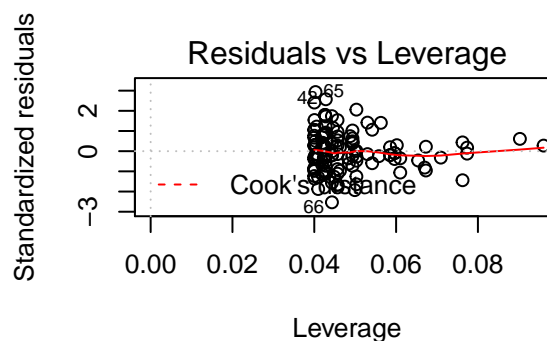
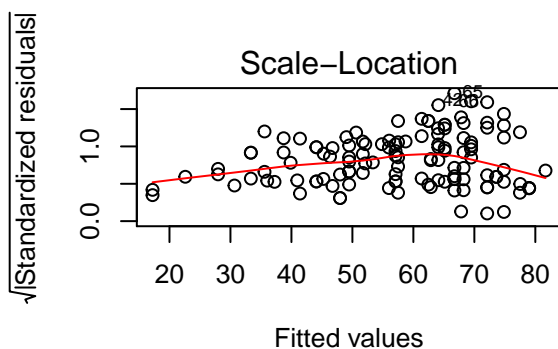
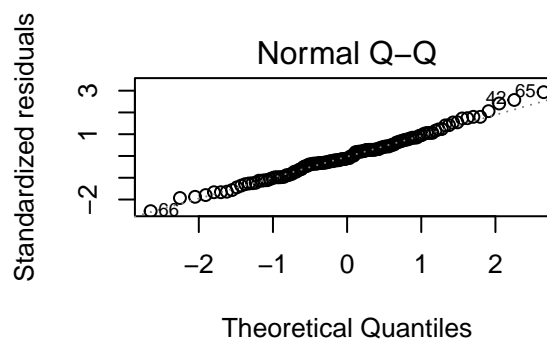
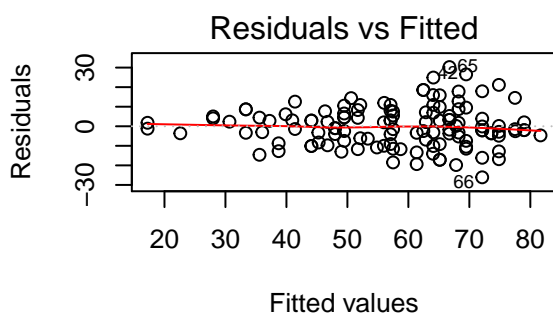
Analysis of Variance Table

Response: longevity

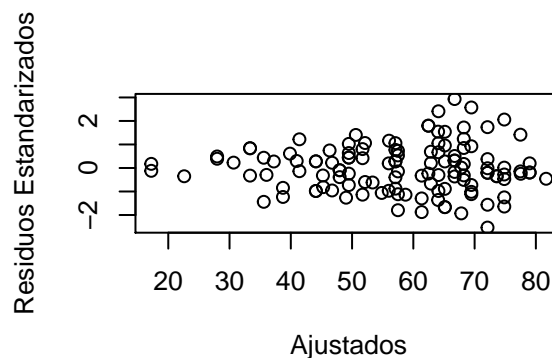
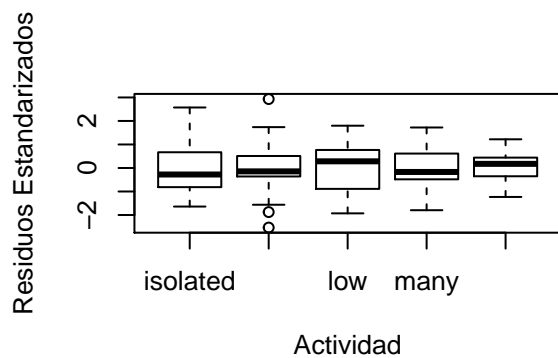
	Df	Sum Sq	Mean Sq	F value	Pr(>F)
thorax	1	15003.3	15003.3	135.069	< 2.2e-16 ***
activity	4	9634.6	2408.6	21.684	1.974e-13 ***
Residuals	118	13107.3	111.1		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
# Algunos gráficos diagnóstico
par(mfrow=c(2,2))
plot(mod2)
```



```
fruitfly$stdres2 <- rstandard(mod2)
fruitfly$fitted2 <- fitted(mod2)
plot(stdres2 ~ activity, data=fruitfly, xlab="Actividad",
     ylab="Residuos Estandarizados")
plot(stdres2 ~ fitted2, data=fruitfly, xlab="Ajustados",
     ylab="Residuos Estandarizados")
```



```
# Comparaciones múltiples
library(multcomp)
compmul <- glht(mod2, linfct = mcp(activity = "Tukey"))
summary(compmul)
```

Simultaneous Tests for General Linear Hypotheses

Multiple Comparisons of Means: Tukey Contrasts

Fit: `lm(formula = longevity ~ thorax + activity, data = fruitfly)`

Linear Hypotheses:

	Estimate	Std. Error	t value	Pr(> t)
one - isolated == 0	2.637	2.984	0.884	0.902364
low - isolated == 0	-7.015	2.981	-2.353	0.135796
many - isolated == 0	4.139	3.027	1.367	0.649624
high - isolated == 0	-20.004	3.016	-6.632	< 1e-04 ***
low - one == 0	-9.652	2.985	-3.234	0.013427 *
many - one == 0	1.502	3.016	0.498	0.987431
high - one == 0	-22.641	2.999	-7.550	< 1e-04 ***
many - low == 0	11.154	3.029	3.683	0.003170 **
high - low == 0	-12.989	3.019	-4.302	0.000328 ***
high - many == 0	-24.142	3.016	-8.005	< 1e-04 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
(Adjusted p values reported -- single-step method)

Modello con varianza residual non costante

```
library(nlme)
```

```
mod3 <- gls(longevity ~ thorax + activity, weights=varComb(varExp(form=~fitted(.))),  
            method="REML")
```

```
summary(mod3)
```

Generalized least squares fit by REML

Model: `longevity ~ thorax + activity`

Data: `fruitfly`

AIC	BIC	logLik
912.3038	934.4693	-448.1519

Combination of variance functions:

Structure: Exponential of variance covariate

Formula: `~fitted(.)`

Parameter estimates:

expon
0.01732811

Coefficients:

Value	Std.Error	t-value	p-value
-------	-----------	---------	---------

```

(Intercept)  -49.84787  8.638814 -5.770221  0.0000
thorax        135.51245 10.459685 12.955691  0.0000
activityone   3.01326  3.088185  0.975737  0.3312
activitylow  -7.08339  2.879451 -2.459981  0.0153
activitymany  4.00209  3.100306  1.290868  0.1993
activityhigh -19.56710  2.581960 -7.578392  0.0000

```

Correlation:

```

              (Intr) thorax actvtyn actvtyl actvtym
thorax       -0.969
activityone  -0.173  0.002
activitylow  -0.142 -0.043  0.515
activitymany -0.214  0.044  0.478  0.511
activityhigh -0.323  0.122  0.575  0.611  0.577

```

Standardized residuals:

```

              Min              Q1              Med              Q3              Max
-2.08038023 -0.66362420 -0.09970742  0.69992606  2.53139237

```

Residual standard error: 3.712311

Degrees of freedom: 124 total; 118 residual

```
anova(mod3, mod2)
```

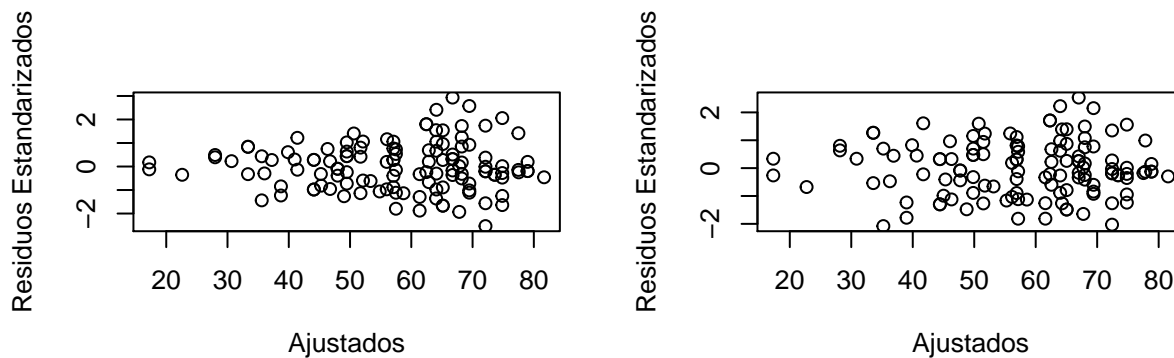
	Model	df	AIC	BIC	logLik	Test	L.Ratio	p-value
mod3	1	8	912.3038	934.4693	-448.1519			
mod2	2	7	920.3535	939.7483	-453.1767	1 vs 2	10.04964	0.0015

Algunos gráficos diagnóstico

```

STDRES <- residuals(mod3, type="pearson")
par(mfrow=c(2,2))
plot(stdres(mod2)~fitted(mod2), xlab="Ajustados",
     ylab="Residuos Estandarizados")
plot(STDRES~fitted(mod3), xlab="Ajustados",
     ylab="Residuos Estandarizados")

```



1.3. Modelos Mixtos

- **Modelo Lineal:**

$$E(\mathbf{y}) = \mathbf{X}\beta, \quad \mathbf{y} \sim N(E(\mathbf{y}), \sigma^2 \mathbf{I}).$$

- **Modelo Lineal Mixto:**

$$\mathbf{y}|\mathbf{u} \sim N(\mathbf{X}\beta + \mathbf{Z}\mathbf{u}, \mathbf{R}), \quad \mathbf{u} \sim N(\mathbf{0}, \mathbf{G}).$$

- **Modelo Lineal Generalizado:**

$$g(E(y_i)) = \mathbf{X}_i\beta, \quad y_i \text{ i.d. } dist(\beta, \xi).$$

- **Modelo Lineal Generalizado Mixto:**

$$g(E(y_i|\mathbf{u})) = \mathbf{X}_i\beta + \mathbf{Z}_i\mathbf{u}, \quad y_i|\mathbf{u} \text{ indep. } dist(\beta, \xi); \mathbf{u} \sim N(\mathbf{0}, \mathbf{G}).$$

1.3.1. ¿Porqué Modelos Mixtos?

- Ambiente flexible para modelar correlaciones y varianzas heterogéneas.
- Modelos heteroscedásticos.
- Modelos jerárquicos con efectos aleatorios.
- Modelos de regresión específico de sujetos.
- Modelos de covarianza y correlación (temporal, espacial).
- Estimación basada en verosimilitud.
- Pruebas válidas en diseños complejos con datos desbalanceados.
- Estimación REML de componentes de varianza.

- Inferencia sobre efectos aleatorios (predictores encogidos hacia la media).

-
- $Y \rightarrow$ función de factores y otras variables independientes con efectos fijos (constantes) y aleatorios.
 - Fijo vs. Aleatorio \rightarrow niveles arbitrariamente determinados vs. muestra aleatoria de niveles (asociado a una distribución de probabilidad).
 - Fijo vs. Aleatorio \rightarrow contexto; mecanismos de colección; espacio de inferencia; estrategia de modelación; agrupamientos jerárquicos en estudios observacionales (regiones, lagos dentro de regiones, muestras dentro de lagos, etc.)

1.3.2. Diseño en bloques completos aleatorizados

- **EXPERIMENTO 1:**

1. Elegimos 4 localidades de una región (cada una bastante homogénea) aleatoriamente.
2. En cada localidad elegimos aleatoriamente cinco parcelas.
3. En cada parcela asignamos un tratamiento distinto (5 tratamientos en total). Repetimos el proceso en las otras áreas.

- **EXPERIMENTO 2:**

1. Elegimos 20 vacas de 3-8 años de edad
2. Agrupamos a las vacas en grupos de 5 con aproximadamente el mismo peso.
3. En el primer grupo de vacas, asignamos cada animal (aleatoriamente) a una de 5 dietas. Repetimos el proceso en los otros grupos.

1.3.3. Modelo para las observaciones de estos dos experimentos

$$Y_{ij} = \mu + \tau_i + b_j + e_{ij},$$

$$b_i \stackrel{i.i.d.}{\sim} N(0, \sigma_b^2), \quad e_{ij} \stackrel{i.i.d.}{\sim} N(0, \sigma^2), \quad b_j, e_{ij} \text{ indep.}$$

$$Y_{ij} | b_j \stackrel{i.d.}{\sim} N(\mu + \tau_i + b_j, \sigma^2) \quad (\text{dist. condicional}).$$

$$Y_{ij} \stackrel{i.d.}{\sim} N(\mu + \tau_i, \sigma_b^2 + \sigma^2) \quad (\text{dist. marginal}).$$

Ejercicio 1.2. Use el siguiente modelo

$$Y_{ij} = \mu + \tau_i + b_j + e_{ij},$$

$$b_i \stackrel{i.i.d.}{\sim} N(0, \sigma_b^2), \quad e_{ij} \stackrel{i.i.d.}{\sim} N(0, \sigma^2), \quad b_j, e_{ij} \text{ indep..}$$

para encontrar las expresiones:

1. $E(Y) =$
2. $E(Y|b) =$
3. $Var(Y) =$
4. $Var(Y|b) =$
5. $Cov(Y_{1j}, Y_{3j}) =$
6. $Corr(Y_{1j}, Y_{3j}) =$
7. $Cov(Y_{11}, Y_{34}) =$
8. $Cov(Y_{11}, Y_{34}) =$

1.3.4. ¿Cuándo un efecto es fijo o es aleatorio?

- Si los niveles estudiados son los únicos que nos interesan, el **efecto es fijo**.
- Si los niveles estudiados son una muestra aleatoria de posibles niveles, el **efecto es aleatorio**.
- Si los niveles estudiados forman una estructura jerárquica (multinivel), como lagos en una región, muestras dentro de cada lago, etc., entonces los **efectos son aleatorios**.
- A veces hay otras consideraciones: contexto, espacio de inferencia (sujeto-específico o promedio-poblacional), estrategia de modelación, etc.

1.3.5. Interpretación en modelos mixtos

Se modela el valor esperado de y_{ij} cuando la variable aleatoria “efecto de...” asume el valor observado en el experimento (Esperanza condicional). Tomando esperanza sobre la variable aleatoria “efecto de...” se tiene la Esperanza marginal o incondicional de Y .

1.3.6. Formulación de un Modelo Lineal Mixto

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{Z}\mathbf{u} + \mathbf{e},$$

$$\begin{bmatrix} \mathbf{u} \\ \mathbf{e} \end{bmatrix} \sim N \left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{G} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{bmatrix} \right)$$

$$E(\mathbf{y}) = \mathbf{X}\beta, \quad Var(\mathbf{y}) = \mathbf{V} = \mathbf{ZGZ}^t + \mathbf{R} \quad (\text{dist. marginal}).$$

$$E(\mathbf{y}|\mathbf{u}) = \mathbf{X}\beta + \mathbf{Z}\mathbf{u}, \quad Var(\mathbf{y}|\mathbf{u}) = \mathbf{R} \quad (\text{dist. condicional}).$$

1.3.7. Otra Formulación de un Modelo Lineal Mixto

$$\mathbf{y}|\mathbf{u} \stackrel{i.d.}{\sim} N(\mathbf{X}\beta + \mathbf{Z}\mathbf{u}, \mathbf{R})$$

$$\mathbf{u} \stackrel{i.d.}{\sim} N(\mathbf{0}, \mathbf{G}).$$

$$E(\mathbf{y}) = \mathbf{X}\beta, \quad Var(\mathbf{y}) = \mathbf{V} = \mathbf{ZGZ}^t + \mathbf{R} \quad (\text{dist. marginal}).$$

$$E(\mathbf{y}|\mathbf{u}) = \mathbf{X}\beta + \mathbf{Z}\mathbf{u}, \quad Var(\mathbf{y}|\mathbf{u}) = \mathbf{R} \quad (\text{dist. condicional}).$$

Esta formulación se especifica en R con los siguientes comandos.

```
library(nlme)
modelo <- lme(Y ~ 1 + factor(Trat) + X, random=list(U=pdIdent(~1)),
             method='REML', data=Datos)
```

La matrix de varianzas y correlación residual \mathbf{R} se puede escribir de la siguiente manera:

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_2 & \cdots & \mathbf{0} \\ \cdots & \cdots & \ddots & \cdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{R}_n \end{bmatrix}$$

donde

$$\mathbf{R}_i = \text{Diag}^{1/2}(\sigma_{ij}) \text{Corr}_i \text{Diag}^{1/2}(\sigma_{ij}).$$

```
Modelo <- lme(Y ~ 1 + factor(Trat) + X,
             random=list(U=pdIdent(~1)), # efectos aleatorios
             weight=varComb(varIdent(form=~1|Trat), # varianza residual
                             correlation=corCompSymm(form=~1|Trat), # estructura de correlación
             method='REML', data=Datos)
```

1.3.8. Problemas a resolver

- Formulación del modelo
- Efectos fijos (vector β).
- Efectos aleatorios (vector \mathbf{u} , matriz \mathbf{G}).
- Estructura de varianzas y correlación residual (\mathbf{R}).
- Estimación e interpretación de parámetros.
- Diagnóstico sobre la adecuación del modelo.
- Pruebas de hipótesis (principalmente sobre efectos fijos).
- Predicción de efectos aleatorios.

1.3.9. Problemas a resolver

1. Formular un modelo para la parte fija (sobreajustando si es necesario).
2. Especificar los efectos aleatorios (pueden provenir del diseño del estudio, de la estructura de los datos, etc.)
3. Seleccionar estructuras de varianzas y correlación residuales (\mathbf{R}) apropiadas.
4. Trabajando con la estructura definida en las partes 2 y 3 (matriz $\mathbf{V} = \mathbf{ZGZ}^t + \mathbf{R}$), realizar inferencias para la parte fija del modelo y predicciones para los efectos aleatorios de interés.

1.3.10. Selección de la estructura de covarianza

1. Pruebas de cociente de verosimilitud:

$$2(\text{Log}(L_a) - \text{Log}(L_0)),$$

donde L_a y L_0 son las verosimilitudes (o las verosimilitudes restringidas) maximizadas bajo las hipótesis alternativa y nula, respectivamente.

- Si la hipótesis nula es un caso especial de la alternativa, este estadístico tiene, bajo H_0 , una distribución χ^2 .
- **Grados de libertad:** diferencia entre cantidad de parámetros bajo la alternativa y la nula.

2. Criterios de verosimilitud penalizada:

■ **Criterio de Akaike:**

$$AIC = -2 * \log(verosimilitud) + q.$$

■ **Criterio de Schwarz:**

$$BIC = -2 * \log(verosimilitud) + q \log(N).$$

Las cantidades q y N son el número de parámetros en el modelo y el número de observaciones, respectivamente.

Ejemplo 1.3 (Humedales). Con el objetivo de determinar si el tipo de un humedal (dos tipos: alimentado con agua de precipitación o con agua subterránea) influía sobre las tasas de producción de metano, se muestrearon tres humedales de cada tipo de agua (precipitación o agua subterránea).

En cada humedal seleccionado, se obtuvieron 3 muestras de suelo (en lugares aleatoriamente elegidos dentro de cada humedal).

Estas muestras se transportaron al laboratorio y se incubaron a temperatura constante.

Las tasas de producción de metano (m mol/l/hr) que se obtuvieron se acompañan en el archivo *metano.txt*.

```
head(Metano)
```

```
# A tibble: 6 x 4
  Tipo Humedal Muestra Metano
<chr>   <int>   <int>   <dbl>
1 ST         1       1    6.63
2 ST         1       2    6.77
3 ST         1       3    5.64
4 ST         2       1   12.4
5 ST         2       2   13.5
6 ST         2       3   11.9
```

El modelo para analizar las tasas de producción de metano en humedales es el siguiente:

$$Y_{ij} = \mu + \tau_i + b_j + e_{ij},$$

$$b_i \stackrel{i.i.d.}{\sim} N(0, \sigma_b^2), \quad e_{ij} \stackrel{i.i.d.}{\sim} N(0, \sigma^2), \quad b_j, e_{ij} \text{ indep.}$$

$$\mathbf{y} | \mathbf{u} \stackrel{i.d.}{\sim} N(\mathbf{X}\beta + \mathbf{Z}\mathbf{u}, \mathbf{R})$$

$$\mathbf{u} \stackrel{i.d.}{\sim} N(\mathbf{0}, \mathbf{G}).$$

```
library(nlme)
Modelo <-lme(Metano~ 1 + factor(Tipo), random=list(Humedal=pdIdent(~1)),
             method="REML", data=Metano)
summary(Modelo)
```

Linear mixed-effects model fit by REML

Data: Metano

	AIC	BIC	logLik
	66.12848	69.21883	-29.06424

Random effects:

Formula: ~1 | Humedal

	(Intercept)	Residual
--	-------------	----------

StdDev:	2.686734	0.8428853
---------	----------	-----------

Fixed effects: Metano ~ 1 + factor(Tipo)

	Value	Std.Error	DF	t-value	p-value
(Intercept)	2.033333	1.576426	12	1.289837	0.2214
factor(Tipo)ST	6.420000	2.229404	4	2.879694	0.0450

Correlation:

	(Intr)
--	--------

factor(Tipo)ST	-0.707
----------------	--------

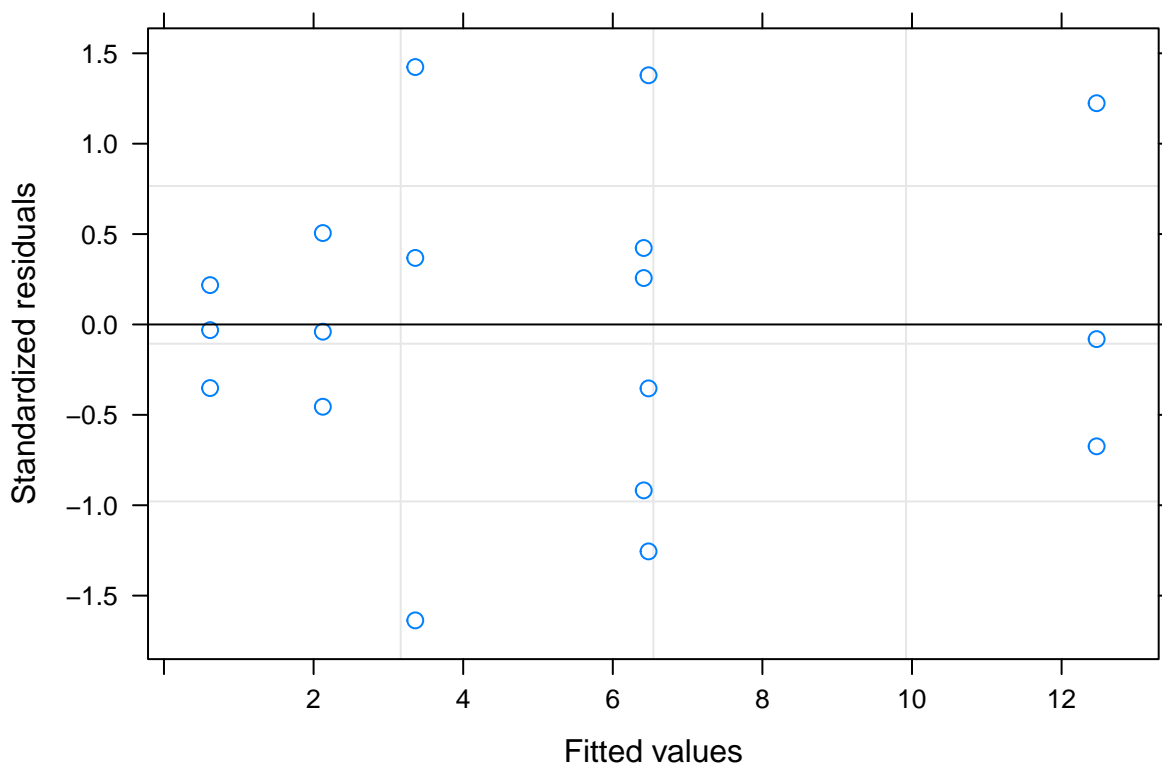
Standardized Within-Group Residuals:

	Min	Q1	Med	Q3	Max
	-1.6370145	-0.4298448	-0.0357015	0.4091395	1.4239002

Number of Observations: 18

Number of Groups: 6

```
plot(Modelo)
```



Ejercicio 1.3. Escribir las componentes de cada uno de los siguientes objetos del modelo propuesto para analizar los datos de humedales:

$$\mathbf{y} = \begin{bmatrix} \\ \\ \\ \\ \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} \\ \\ \\ \\ \end{bmatrix}, \quad \beta = \begin{bmatrix} \\ \\ \\ \\ \end{bmatrix}, \quad \mathbf{Z} = \begin{bmatrix} \\ \\ \\ \\ \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} \\ \\ \\ \\ \end{bmatrix}, \quad \mathbf{e} = \begin{bmatrix} \\ \\ \\ \\ \end{bmatrix}$$

$$\mathbf{G} = \begin{bmatrix} \\ \\ \\ \\ \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} \\ \\ \\ \\ \end{bmatrix}, \quad Var(\mathbf{y}) = \mathbf{V} = \mathbf{ZGZ}^t + \mathbf{R} = \begin{bmatrix} \\ \\ \\ \\ \end{bmatrix}$$

Ejemplo 1.4 (Esponjas (Qian, 2010) - Revisitado). Se realizó un experimento para estudiar la interacción entre esponjas y mangle rojo. Se deseaba probar si el mangle se beneficiaba del crecimiento de organismos alrededor de las raíces.

Parcelas con mangle rojo se asignaron aleatoriamente a cuatro tratamientos: control sin manipular, esponja plástica colocada alrededor de la raíz, colonias de esponja roja (*Tedania ignis*) transplantadas a las raíces y colonias de esponja púrpura (*Haliclona implexiformis*) transplantadas a las raíces.

La variable de respuesta es crecimiento de las raíces (mm/día).

```
head(SpongeData)
```

```
# A tibble: 6 x 3
  Location Treatment RootGrowthRate
  <chr>      <fct>          <dbl>
1 lcn       Control        0.240
2 lcn       Control       -0.0500
3 lcn       Control        0.190
4 etb       Control        1.02
5 lcs       Control        0.840
6 bbs       Control        0.110
```

El modelo para analizar la respuesta es crecimiento de las raíces es el siguiente:

$$Y_{ijk} = \mu + \tau_i + loc_j + e_{ijk},$$

$$loc_j \stackrel{i.i.d.}{\sim} N(0, \sigma_l^2); e_{ijk} \stackrel{i.i.d.}{\sim} N(0, \sigma_e^2); loc_j, e_{ijk} \text{ independientes.}$$

```
library(nlme)
M <- lme(RootGrowthRate ~ Treatment, random=list(Location=pdIdent(~1)),
        method="REML", data=SpongeData)
summary(M)
```

Linear mixed-effects model fit by REML

```
Data: SpongeData
      AIC      BIC    logLik
111.3824 124.6995 -49.69122
```

Random effects:

```
Formula: ~1 | Location
      (Intercept)  Residual
StdDev:   0.0564957 0.4593281
```

Fixed effects: RootGrowthRate ~ Treatment

	Value	Std.Error	DF	t-value	p-value
(Intercept)	0.2401578	0.1042144	65	2.304459	0.0244
TreatmentFoam	0.3537240	0.1435280	65	2.464495	0.0164
TreatmentHaliclona	0.4888779	0.1499451	65	3.260378	0.0018
TreatmentTedania	0.6605018	0.1598051	65	4.133170	0.0001

Correlation:

	(Intr)	TrtmnF	TrtmnH
TreatmentFoam	-0.672		
TreatmentHaliclona	-0.644	0.467	

```
TreatmentTedania    -0.608  0.439  0.423
```

Standardized Within-Group Residuals:

	Min	Q1	Med	Q3	Max
	-4.22569820	-0.56596489	-0.06472748	0.53490049	2.46402438

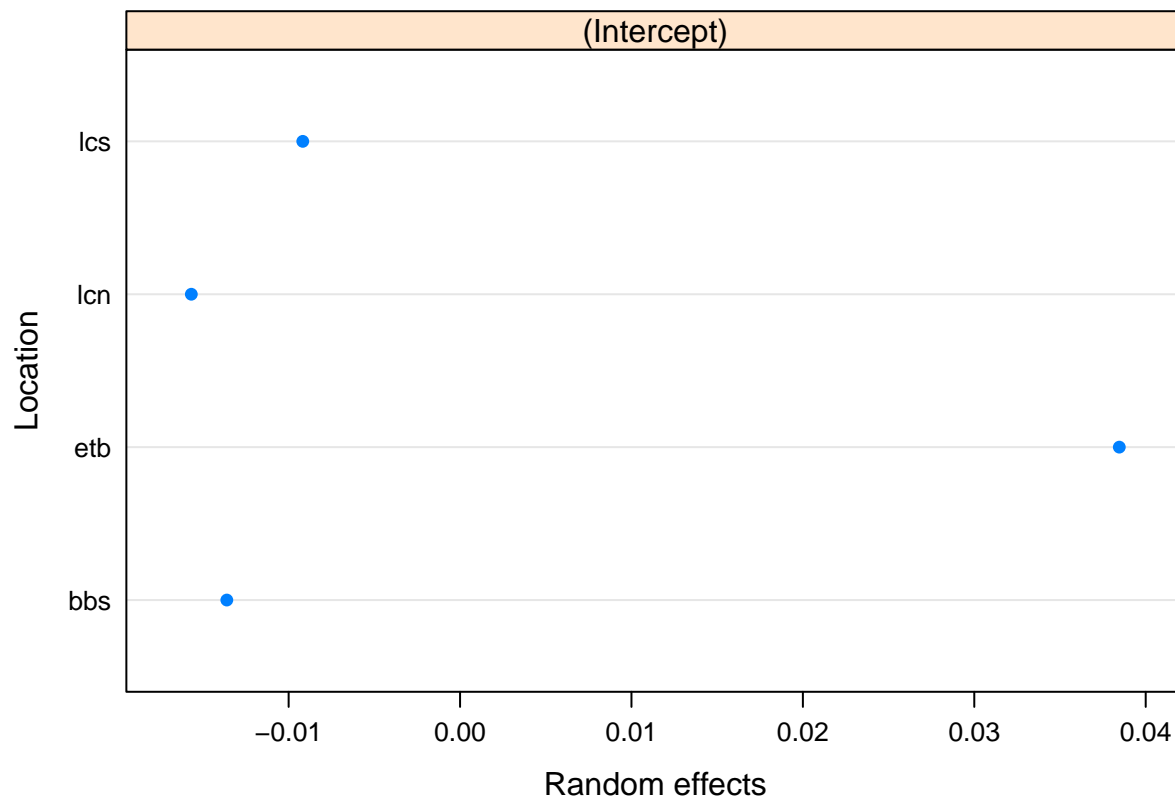
Number of Observations: 72

Number of Groups: 4

```
ranef(M)
```

```
(Intercept)
bbs -0.01360736
etb  0.03846065
lcn -0.01567753
lcs -0.00917575
```

```
plot(ranef(M))
```



```
# Comparaciones múltiples
library(multcomp)
compmulM <- glht(M, linfct = mcp(Treatment = "Tukey"))
summary(compmulM)
```

Simultaneous Tests for General Linear Hypotheses

Multiple Comparisons of Means: Tukey Contrasts

```
Fit: lme.formula(fixed = RootGrowthRate ~ Treatment, data = SpongeData,
  random = list(Location = pdIdent(~1)), method = "REML")
```

Linear Hypotheses:

	Estimate	Std. Error	z value	Pr(> z)
Foam - Control == 0	0.3537	0.1435	2.464	0.06541 .
Haliclona - Control == 0	0.4889	0.1499	3.260	0.00605 **
Tedania - Control == 0	0.6605	0.1598	4.133	< 0.001 ***
Haliclona - Foam == 0	0.1352	0.1516	0.892	0.80875
Tedania - Foam == 0	0.3068	0.1612	1.903	0.22617
Tedania - Haliclona == 0	0.1716	0.1666	1.030	0.73105

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
(Adjusted p values reported -- single-step method)

Ejemplo 1.5 (Prevención de Tabaco en Jóvenes). El siguiente conjunto de datos proviene de un estudio comparativo para evaluar la eficacia de programas de prevención de tabaco en jóvenes. Se compararon 4 grupos, provenientes de un diseño 2x2 (con/sin currículo escolar, con/sin programa basado en TV).

Se impactaron 1600 estudiantes de séptimo grado de 135 grupos de 28 escuelas de Los Angeles (datos de Hedeker et al., 1994 y Agresti, 2015, p.297). Las escuelas se asignaron aleatoriamente a los tratamientos, y la respuesta fue una escala *tobacco and health knowledge* (THK) al finalizar el estudio. También se usó la misma escala al comenzarlo (*Pre THK*) y se usó como covariable (*PTHK*).

```
head(smoking)
```

```
# A tibble: 6 x 6
  school class   SC   TV  PTHK    y
  <int> <int> <int> <int> <int> <int>
1   403 403101     1     0     2     3
2   403 403101     1     0     4     4
3   403 403101     1     0     4     3
4   403 403101     1     0     3     4
5   403 403101     1     0     3     4
6   403 403101     1     0     4     3
```



```
library(lme4)
Model1 <- lmer(y~PTHK+SC+TV+(1|school)+(1|class),
               data=smoking)
summary(Model1)
```

Linear mixed model fit by REML ['lmerMod']
 Formula: y ~ PTHK + SC + TV + (1 | school) + (1 | class)
 Data: smoking

REML criterion at convergence: 5374.3

Scaled residuals:

	Min	1Q	Median	3Q	Max
	-2.5202	-0.6975	-0.0177	0.6875	3.1630

Random effects:

Groups	Name	Variance	Std.Dev.
class	(Intercept)	0.06853	0.2618
school	(Intercept)	0.03925	0.1981
Residual		1.60108	1.2653

Number of obs: 1600, groups: class, 135; school, 28

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	1.78493	0.11295	15.803
PTHK	0.30524	0.02590	11.786
SC	0.47147	0.11330	4.161
TV	0.01956	0.11330	0.173

Correlation of Fixed Effects:

	(Intr)	PTHK	SC
PTHK	-0.493		
SC	-0.503	0.025	
TV	-0.521	0.015	-0.002

```
Model2 <- lme(y~PTHK+SC+TV, random=list(school=pdIdent(~1),
                                         class=pdIdent(~1)),
              data=smoking)
summary(Model2)
```

Linear mixed-effects model fit by REML
 Data: smoking

	AIC	BIC	logLik
--	-----	-----	--------

5388.337 5425.964 -2687.168

Random effects:

Formula: ~1 | school
(Intercept)

StdDev: 0.1981275

Formula: ~1 | class %in% school
(Intercept) Residual

StdDev: 0.261783 1.265338

Fixed effects: y ~ PTHK + SC + TV

	Value	Std.Error	DF	t-value	p-value
(Intercept)	1.7849270	0.11294585	1464	15.803387	0.0000
PTHK	0.3052393	0.02589749	1464	11.786441	0.0000
SC	0.4714724	0.11330445	25	4.161111	0.0003
TV	0.0195612	0.11330016	25	0.172649	0.8643

Correlation:

(Intr) PTHK SC

PTHK -0.493

SC -0.503 0.025

TV -0.521 0.015 -0.002

Standardized Within-Group Residuals:

Min	Q1	Med	Q3	Max
-2.52020155	-0.69745998	-0.01770643	0.68751261	3.16298132

Number of Observations: 1600

Number of Groups:

school	class %in% school
28	135

ignorando las correlaciones

```
Modeloind <- lm(y~PTHK+SC+TV, data=smoking)
```

```
summary(Modeloind)
```

Call:

```
lm(formula = y ~ PTHK + SC + TV, data = smoking)
```

Residuals:

Min	1Q	Median	3Q	Max
-3.5635	-0.9130	-0.0626	0.8981	4.2416

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	1.73734	0.07866	22.088	< 2e-16 ***
PTHK	0.32525	0.02589	12.561	< 2e-16 ***
SC	0.47987	0.06529	7.350	3.15e-13 ***
TV	0.04534	0.06518	0.696	0.487

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.303 on 1596 degrees of freedom

Multiple R-squared: 0.1136, Adjusted R-squared: 0.112

F-statistic: 68.21 on 3 and 1596 DF, p-value: < 2.2e-16

1.3.11. Algunas conclusiones del análisis de los datos de tabaco

- Estimados de correlación entre estudiantes de la misma clase.
- Estimados de correlación entre estudiantes de la misma escuela pero diferentes clases.
- Observar los errores estándar de los efectos fijos si no tenemos en cuenta los efectos aleatorios.

1.4. Estimación en Modelos Lineales Mixtos

$$E(\mathbf{y}) = \mathbf{X}\beta, \quad \text{Var}(\mathbf{y}) = \mathbf{V} = \mathbf{ZGZ}^t + \mathbf{R} \quad (\text{dist. marginal}).$$

- Para realizar inferencias en modelo mixto debemos estimar β , \mathbf{G} y \mathbf{R} . Para poder estimar separadamente a \mathbf{G} y \mathbf{R} (nosotros solo observamos \mathbf{V}), debemos también predecir los valores de la variable aleatoria \mathbf{u} .
- Primero asumimos \mathbf{G} y \mathbf{R} conocidos, y se desarrollan ecuaciones de estimación / predicción para β y \mathbf{u} .
- Las ecuaciones de estimación para β son las correspondientes a mínimos cuadrados generalizados (o máxima verosimilitud).
- Luego veremos cómo estos estimadores/predictores se afectan por tener \mathbf{G} y \mathbf{R} estimados.

1.4.1. Ecuaciones de Modelos Mixtos (Henderson)

La función de densidad conjunta de \mathbf{u}, \mathbf{e} esta dada por:

$$f(\mathbf{u}, \mathbf{e}) = (2\pi)^{-(n+g)/2} \begin{vmatrix} \mathbf{G} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{vmatrix}^{-1/2} \exp \left\{ -\frac{1}{2} \begin{bmatrix} \mathbf{y} - \mathbf{X}\beta - \mathbf{Zu} \end{bmatrix}^t \begin{bmatrix} \mathbf{G}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{y} - \mathbf{X}\beta - \mathbf{Zu} \end{bmatrix} \right\}$$

Las ecuaciones de estimación de Herdenson son las siguientes:

$$\begin{bmatrix} \mathbf{b} \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} \mathbf{X}^t \mathbf{R}^{-1} \mathbf{X} & \mathbf{X}^t \mathbf{R}^{-1} \mathbf{Z} \\ \mathbf{Z}^t \mathbf{R}^{-1} \mathbf{X} & \mathbf{Z}^t \mathbf{R}^{-1} \mathbf{Z} + \mathbf{G}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{X}^t \mathbf{R}^{-1} \mathbf{y} \\ \mathbf{Z}^t \mathbf{R}^{-1} \mathbf{y} \end{bmatrix}$$

Solución:

$$\begin{bmatrix} \hat{\beta} \\ \hat{\mathbf{u}} \end{bmatrix} = \begin{bmatrix} (\mathbf{X}^t \mathbf{V}^{-1} \mathbf{X})^{-1} \mathbf{X}^t \mathbf{V}^{-1} \mathbf{y} \\ \mathbf{G} \mathbf{Z}^t \mathbf{V}^{-1} [\mathbf{y} - \mathbf{X}(\mathbf{X}^t \mathbf{V}^{-1} \mathbf{X})^{-1} \mathbf{X}^t \mathbf{V}^{-1} \mathbf{y}] \end{bmatrix}$$

Si los \mathbf{u} fuesen parámetros (cantidades fijas a estimar), entonces las ecuaciones por mínimos cuadrados serían:

$$\begin{bmatrix} \mathbf{b} \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} \mathbf{X}^t \mathbf{R}^{-1} \mathbf{X} & \mathbf{X}^t \mathbf{R}^{-1} \mathbf{Z} \\ \mathbf{Z}^t \mathbf{R}^{-1} \mathbf{X} & \mathbf{Z}^t \mathbf{R}^{-1} \mathbf{Z} \end{bmatrix} \begin{bmatrix} \mathbf{X}^t \mathbf{R}^{-1} \mathbf{y} \\ \mathbf{Z}^t \mathbf{R}^{-1} \mathbf{y} \end{bmatrix}$$

Los predictores para el caso de \mathbf{u} aleatorio “encogen” el estimador de \mathbf{u} fijo y lo acercan a la media. Este “encogimiento” (*shrinkage*) es mayor a medida que \mathbf{G} es más “chica” (tiene sus autovectores más chicos, o sea, los efectos aleatorios tienen una varianza más pequeña).

1.4.2. Propiedades para \mathbf{G} y \mathbf{R} conocidas

- $\hat{\beta}$ o funciones estimables de este vector si \mathbf{X} no es de rango completo por columnas) es el mejor estimador lineal insesgado (**BLUE**), y es igual al estimador máximo verosímil.
- $\hat{\mathbf{u}}$ es el mejor predictor lineal insesgado (**BLUP**).

1.4.3. Propiedades para \mathbf{G} y \mathbf{R} estimadas

- En la práctica \mathbf{V} es desconocida y se reemplaza por su estimador

$$\hat{\mathbf{V}} = \mathbf{Z}\hat{\mathbf{G}}\mathbf{Z}^t + \hat{\mathbf{R}}.$$

- Los métodos de estimación más usados son máxima verosimilitud (ML) y máxima verosimilitud restringida (REML). Otros métodos son MIVQUE, MIVQUE2, etc.

1.5. Estimación por Máxima Verosimilitud (*ML*)

$\theta \leftarrow$ vector conteniendo los parámetros de varianza.

La ecuación de log verosimilitud es

$$-2\log(\theta, \beta; \mathbf{y}) = \log |\mathbf{V}| + (\mathbf{y} - \mathbf{X}\beta)^t \mathbf{V}^{-1} (\mathbf{y} - \mathbf{X}\beta) + c.$$

Podemos resolver para

$$\hat{\beta} = (\mathbf{X}^t \mathbf{V}(\theta)^{-1} \mathbf{X})^{-1} \mathbf{X}^t \mathbf{V}(\theta)^{-1} \mathbf{y},$$

quedando la verosimilitud solo en función de θ .

$$-2\log(\theta, \beta(\theta); \mathbf{y}) = \log |\mathbf{V}(\theta)| + (\mathbf{y} - \mathbf{X}\beta(\theta))^t \mathbf{V}(\theta)^{-1} (\mathbf{y} - \mathbf{X}\beta(\theta)) + c.$$

1.6. Estimación por Máxima Verosimilitud Restringida (*REML*)

- El simple ejemplo del estimador ML de la varianza de una muestra aleatoria de variables normales, sugiere que cuando μ no es conocida y debe estimarse, dicha estimación introduce un sesgo en el estimador ML de la varianza.
- La pregunta entonces es, ¿cómo estimar las componentes de varianza sin tener que estimar los parámetros correspondientes a los efectos fijos?
- La respuesta conduce al estimador REML, sugerido por Patterson y Thompson (1971).
- En esta aproximación el vector de efectos fijos es eliminado de la log-verosimilitud.

En general, lo único que necesitamos es una transformación de los datos ortogonal a \mathbf{X} , es decir, necesitamos residuos:

$$\mathbf{r} = \mathbf{A}^t \mathbf{y} \sim N(0, \mathbf{A}^t \mathbf{V}(\theta) \mathbf{A}).$$

La función de verosimilitud restringida no depende de β :

$$-2 \log_R(\theta; \mathbf{r}) = \log |\mathbf{V}(\theta)| + \log |\mathbf{X}^t \mathbf{V}(\theta) \mathbf{X}| + \mathbf{r}^t \mathbf{V}(\theta)^{-1} \mathbf{r}.$$

Sustituyendo en el estimador mínimo-cuadrático (generalizado) de β el estimador REML de θ obtenemos el estimador de efectos fijos $\hat{\beta}(\hat{\theta}_{REML})$, que suele denotarse $\hat{\beta}_{REML}$.

1.7. Propiedades del Estimador de Efectos Fijos

- El estimador del vector de efectos fijos se obtiene por mínimos cuadrados generalizados usando $\hat{\theta}$ en lugar de θ para construir \mathbf{V} .
- Si $E(\mathbf{y}|\mathbf{u}) = \mathbf{X}\beta$, entonces este estimador es insesgado (sujeto, por supuesto a estimabilidad).
- Luego, para obtener estimaciones insesgadas relacionadas a los efectos fijos es suficiente que la media de la respuesta sea correctamente especificada.
- Ya no les llamaremos BLUE y BLUP a las soluciones de las ecuaciones de modelos mixtos, sino EBLUE y EBLUP (observar que ni siquiera son lineales).

1.8. Predicción de Efectos Aleatorios

- Las soluciones para los efectos aleatorios de las ecuaciones de modelos mixtos suelen llamarse Estimadores Bayesianos Empíricos (*Empirical Bayes Estimators*).
- Para un efecto aleatorio b_j o un vector de efectos aleatorios \mathbf{u} recordemos que estamos suponiendo que

$$b_j \stackrel{i.i.d.}{\sim} N(0, \sigma_b^2), \quad \mathbf{u} \stackrel{i.i.d.}{\sim} N(\mathbf{0}, \mathbf{G}).$$

- Es decir, $E(b_j) = 0$. Si pensamos desde el punto de vista Bayesiano, podemos interpretar esto como que la distribución a priori tiene esperanza 0. La esperanza de la distribución a posteriori del efecto aleatorio (dados los datos) será:

$$f(b_j|\mathbf{y}) \propto f(\mathbf{y}|b_j)f(b_j)$$

$$E(b_j|\mathbf{y}) = \int b_j f(b_j|\mathbf{y}) db_j.$$

Para el caso especial de

$$\mathbf{y}|\mathbf{u} \stackrel{i.d.}{\sim} N(\mathbf{X}\beta + \mathbf{Z}\mathbf{u}, \sigma^2\mathbf{I}), \quad \mathbf{u} \stackrel{i.d.}{\sim} N(\mathbf{0}, \mathbf{G}).$$

La media a posteriori es

$$\hat{\mathbf{u}} = \mathbf{G}\mathbf{Z}^t\mathbf{V}^{-1}(\mathbf{y} - \mathbf{X}\beta).$$

Si reemplazamos los estimadores REML de \mathbf{G} , \mathbf{V} , β tenemos predictores de los efectos aleatorios, que ahora tienen una justificación como estimadores Bayesianos (empíricos, ya que no hemos especificado la distribución a priori) de la media a posteriori de los efectos aleatorios.

1.9. Prueba de Hipótesis para Efectos Fijos

Supongamos que queremos probar

$$H_0 : \mathbf{A}_{q \times k}\beta = \mathbf{d}.$$

Es decir, tenemos q contrastes de β linealmente independientes. Por ejemplo, si $\beta = (\beta_0, \beta_1, \beta_2)^t$ y queremos probar que $\beta_0 = 0$ y que $\beta_1 = \beta_2$ entonces

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \end{bmatrix}, \quad \mathbf{d} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

- Si \mathbf{V} es conocida, entonces un estadístico de prueba de *Wald* es:

$$W = \frac{(\mathbf{A}\hat{\beta} - \mathbf{d})^t [\mathbf{A}(\mathbf{X}^t\mathbf{V}^{-1}\mathbf{X}) - \mathbf{A}^t]^{-1} (\mathbf{A}\hat{\beta} - \mathbf{d})}{\sigma^2} \sim \chi_q^2.$$

- Si \mathbf{V} es conocida excepto por σ^2 , entonces un estadístico de prueba de Wald es:

$$F = \frac{(\mathbf{A}\hat{\beta} - \mathbf{d})^t [\mathbf{A}(\mathbf{X}^t\mathbf{V}^{-1}\mathbf{X}) - \mathbf{A}^t]^{-1} (\mathbf{A}\hat{\beta} - \mathbf{d})/q}{(\mathbf{y} - \mathbf{X}\hat{\beta})^t\mathbf{V}^{-1}(\mathbf{y} - \mathbf{X}\hat{\beta})/(n - r(\mathbf{X}))} \sim F_{q, n-r(\mathbf{X})}.$$

- En la práctica, podemos poner el estimador de \mathbf{V} :

$$W^* = \frac{(\mathbf{A}\hat{\beta} - \mathbf{d})^t [\mathbf{A}(\mathbf{X}^t\mathbf{V}(\hat{\theta})^{-1}\mathbf{X}) - \mathbf{A}^t]^{-1} (\mathbf{A}\hat{\beta} - \mathbf{d})}{\sigma^2}.$$

$$F^* = \frac{(\mathbf{A}\hat{\beta} - \mathbf{d})^t [\mathbf{A}(\mathbf{X}^t \mathbf{V}(\hat{\theta})^{-1} \mathbf{X}) - \mathbf{A}^t]^{-1} (\mathbf{A}\hat{\beta} - \mathbf{d})/q}{(\mathbf{y} - \mathbf{X}\hat{\beta})^t \mathbf{V}(\hat{\theta})^{-1} (\mathbf{y} - \mathbf{X}\hat{\beta}) / (n - r(\mathbf{X}))}.$$

- Si el estimador de θ es consistente, entonces W^* es asintóticamente chi-cuadrado. En general se prefiere usar F^* , que aunque no tiene una distribución asintótica F , suele mostrar mejor comportamiento para n pequeño.
- Se realizan correcciones al sesgo introducido por la estimación de θ (Kenward-Roger). Para mejorar la aproximación, los grados de libertad del denominador suelen también corregirse (Satterthwaite, Kenward-Roger, etc.).

1.10. Prueba de verosimilitud para Efectos Fijos

- Otra opción para probar hipótesis sobre efectos fijos es la prueba del cociente de verosimilitud.

$$\chi_{LR}^2 = -2[L(\hat{\beta}_{H_0}) - L(\hat{\beta}_{H_a})].$$

- Para esto ajustamos un modelo completo y uno reducido, y calculamos la diferencia. [Esto se debe hacer solamente con ML, nunca con REML \(recordemos que en REML los efectos fijos no intervienen, ya que se usan los residuos\).](#)

1.11. Prueba de verosimilitud para Varianzas

- En el caso de probar hipótesis sobre parámetros de varianzas y covarianzas (elementos de \mathbf{G} y \mathbf{R}) es posible también usar prueba de cociente de verosimilitud.
- En este caso podemos hacerlo con ML o REML, ya que estamos comparando modelos con la misma parte fija.
- Debemos tener en cuenta que al probar si una o más varianzas son cero estamos en la frontera del espacio de parámetros, y por lo tanto la distribución asintótica del estadístico de la prueba no es chi-cuadrado.
- Molenberghs (2007) recomienda usar como distribución asintótica una mezcla de distribuciones chi-cuadrado. Por ejemplo, para probar $\sigma_b^2 = 0$, el cociente de verosimilitud se prueba con una mezcla de χ_0^2 y χ_1^2 .

Ejemplo 1.6 (Regresión Multinivel (Qian, p.362)). Estudio conducido por USGS para ver los efectos de la urbanización sobre los ecosistemas de arroyos (*EUSE, Effects of Urbanization on Stream Ecosystems*).

Serie de estudios conducidos en 9 ciudades de Estados Unidos para ver el efecto sobre biota acuáticos (peces, invertebrados y algas) y características físicas y químicas.

Para identificar gradientes representativos de urbanización con diferentes niveles de urbanización dentro de un ambiente relativamente homogéneo se usó un índice multimétrico de intensidad urbana (UII), calculado a nivel nacional (NUII).

Una de las variables que se desea relacionar con el UII es la tolerancia promedio de taxa de macroinvertebrados (TOLr), indicador de si una especie o taxa puede sobrevivir en un ambiente contaminado. En general, valores altos de TOLr se encuentran en aguas de calidad mala.

La relación entre TOLr y NUII parece ser lineal, aunque con rectas diferentes por ciudad.

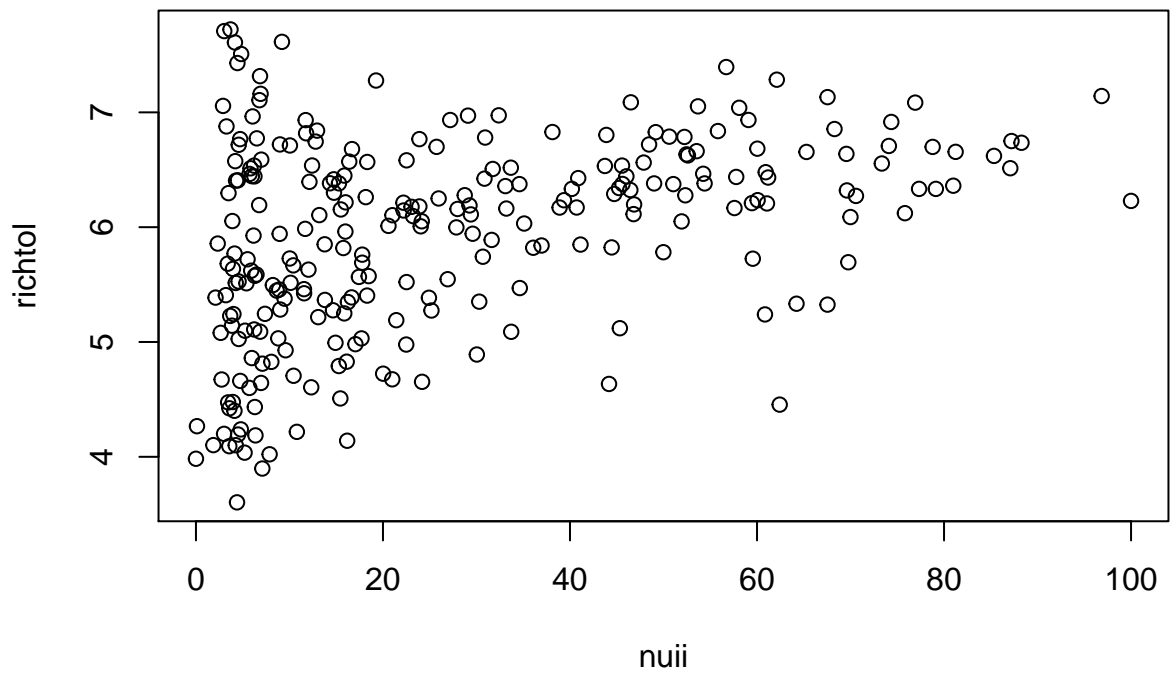
- ¿Cómo interpretamos el intercepto? Recordemos que NUII=0 es no urbanizado
- ¿Cómo interpretamos la pendiente?

Vamos a analizar un modelo con recta única y uno con una recta diferente para cada ciudad.

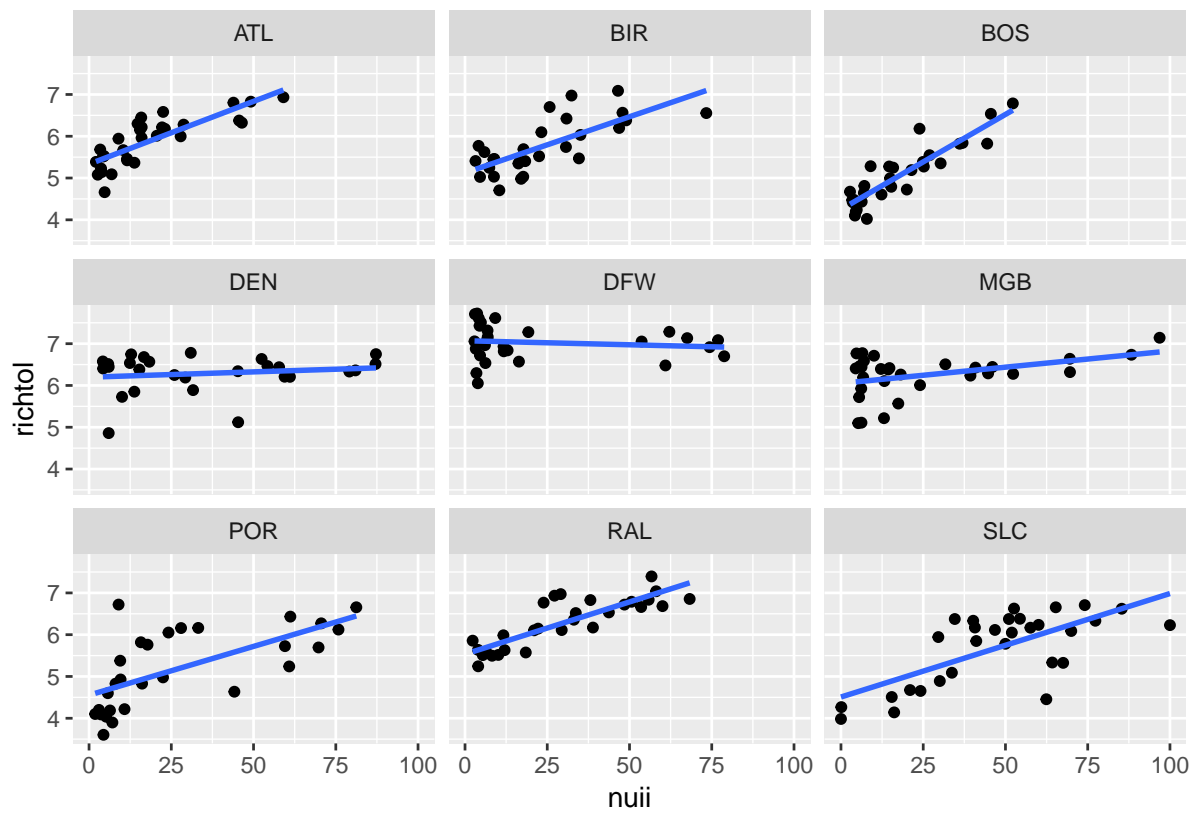
```
head(rtol2)
```

```
# A tibble: 6 x 9
  X1 city   suid   sample   agr a_temp a_prec richtol  nuui
<int> <chr> <chr>   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1     1 ATL   ACFB  2204230. 10.2   16.5   128.    6.00 27.8
2     2 ATL   ACFB  2204468. 22.7   16.6   126.    5.96 16.0
3     3 ATL   ACFB  2206314.  2.74   16.0   138.    6.38 45.6
4     4 ATL   ACFB  2208150. 11.7   15.9   133.    6.01 20.6
5     5 ATL   ACFB  2213450. 22.5   17.3   124.    5.68  3.40
6     6 ATL   ACFB  2217293. 23.3   15.9   133.    5.37 13.8
```

```
plot(richtol~nuui, data=rtol2)
```



```
p <- ggplot(rtol2, aes(x=nuui, y=richtol))+
  geom_point()
p + facet_wrap(~city) + geom_smooth(method=lm, se=FALSE)
```



```
recta.unica <- lm(richtol~nuii, data=rtol2)
rectas.dif <- lm(richtol~nuii*factor(city)-1-nuii,
                data=rtol2)
summary(rectas.dif)
```

Call:

```
lm(formula = richtol ~ nuii * factor(city) - 1 - nuii, data = rtol2)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.5997	-0.2440	-0.0002	0.2790	1.9582

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
factor(city)ATL	5.331757	0.135463	39.359	< 2e-16 ***
factor(city)BIR	5.122756	0.154388	33.181	< 2e-16 ***
factor(city)BOS	4.248632	0.139184	30.525	< 2e-16 ***
factor(city)DEN	6.197783	0.149944	41.334	< 2e-16 ***
factor(city)DFW	7.070354	0.116733	60.569	< 2e-16 ***
factor(city)MGB	6.050082	0.122666	49.321	< 2e-16 ***
factor(city)POR	4.552942	0.130455	34.901	< 2e-16 ***
factor(city)RAL	5.533967	0.154333	35.857	< 2e-16 ***
factor(city)SLC	4.507952	0.193590	23.286	< 2e-16 ***
nuii:factor(city)ATL	0.030060	0.005584	5.384	1.72e-07 ***
nuii:factor(city)BIR	0.026880	0.005297	5.075	7.72e-07 ***
nuii:factor(city)BOS	0.045481	0.006171	7.370	2.67e-12 ***
nuii:factor(city)DEN	0.002546	0.003369	0.756	0.4505
nuii:factor(city)DFW	-0.001928	0.003340	-0.577	0.5642
nuii:factor(city)MGB	0.007762	0.003336	2.327	0.0208 *
nuii:factor(city)POR	0.023329	0.003470	6.723	1.26e-10 ***
nuii:factor(city)RAL	0.024990	0.004350	5.744	2.75e-08 ***
nuii:factor(city)SLC	0.024770	0.003664	6.760	1.02e-10 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4744 on 243 degrees of freedom

Multiple R-squared: 0.9942, Adjusted R-squared: 0.9937

F-statistic: 2299 on 18 and 243 DF, p-value: < 2.2e-16

```
anova(recta.unica, rectas.dif)
```

Analysis of Variance Table

```

Model 1: richtol ~ nuii
Model 2: richtol ~ nuii * factor(city) - 1 - nuii
  Res.Df    RSS Df Sum of Sq    F    Pr(>F)
1     259 164.245
2     243  54.685 16    109.56 30.428 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

1.11.0.1. Modelo Mixto para datos de ecosistemas

- Para esta ejemplo podemos pensar en un modelo con una recta “base” común que describe la relación entre NUII y RICHTOL, al que se le agregan desviaciones de la línea base debidas al efecto de la ciudad.
- Si este efecto de la ciudad lo consideramos aleatorio, tenemos un modelo mixto.
- En este caso es algo intermedio entre los dos modelos que hemos ajustado: rectas diferentes para cada ciudad y una recta única para todas las ciudades.
- Otros nombres para este modelo son: regresión multinivel, o modelo de regresión con coeficientes aleatorios.

Modelo:

$$Y_{ij} = (\beta_0 + b_{0i}) + (\beta_1 + b_{1i})x_{ij} + e_{ij}$$

$$\begin{bmatrix} b_{0i} \\ b_{1i} \end{bmatrix} \stackrel{i.i.d.}{\sim} N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_0^2 & \sigma_{01}^2 \\ \sigma_{01}^2 & \sigma_1^2 \end{bmatrix}\right), \quad e_{ij} \stackrel{i.i.d.}{\sim} N(0, \sigma_e^2), \quad \mathbf{b}_i, e_{ij} \text{ independientes.}$$

$$\mathbf{y}|\mathbf{u} \stackrel{i.d.}{\sim} N(\mathbf{X}\beta + \mathbf{Z}\mathbf{u}, \mathbf{R}),$$

$$\mathbf{u} \stackrel{i.d.}{\sim} N(\mathbf{0}, \mathbf{G}).$$

Ejercicio 1.4. Escribir las componentes de cada uno de los siguientes objetos del modelo propuesto para analizar los datos de ecosistemas:

$$\mathbf{y} = \begin{bmatrix} \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} \end{bmatrix}, \quad \beta = \begin{bmatrix} \end{bmatrix}, \quad \mathbf{Z} = \begin{bmatrix} \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} \end{bmatrix}, \quad \mathbf{e} = \begin{bmatrix} \end{bmatrix}$$

$$\mathbf{G} = \begin{bmatrix} \\ \\ \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} \\ \\ \end{bmatrix}, \quad \text{Var}(\mathbf{y}) = \mathbf{V} = \mathbf{ZGZ}^t + \mathbf{R} = \begin{bmatrix} \\ \\ \end{bmatrix}$$

```
plot(richtol~nuii, data=rtol2)
p <- ggplot(rtol2, aes(x=nuii, y=richtol))+
  geom_point()
p + facet_wrap(~city) + geom_smooth(method=lm, se=FALSE)

recta.unica <- lm(richtol~nuii, data=rtol2)
rectas.dif <- lm(richtol~nuii*factor(city)-1-nuii,
  data=rtol2)
summary(recta.unica)
summary(rectas.dif)

anova(recta.unica,rectas.dif)

# Modelo Mixto
library(lme4)
library(lattice)
mod.mix <- lmer(richtol~1+nuii+(1+nuii|city),
  data=rtol2)
summary(mod.mix)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: richtol ~ 1 + nuii + (1 + nuii | city)
Data: rtol2
```

REML criterion at convergence: 411.8

Scaled residuals:

Min	1Q	Median	3Q	Max
-3.4289	-0.5610	0.0144	0.5799	4.1053

Random effects:

Groups	Name	Variance	Std.Dev.	Corr
city	(Intercept)	0.8171420	0.90396	
	nuii	0.0001882	0.01372	-0.89
Residual		0.2253116	0.47467	

Number of obs: 261, groups: city, 9

Fixed effects:

Estimate	Std. Error	t value
----------	------------	---------

```
(Intercept) 5.418388 0.305148 17.757
nuii         0.019431 0.004789 4.057
```

Correlation of Fixed Effects:

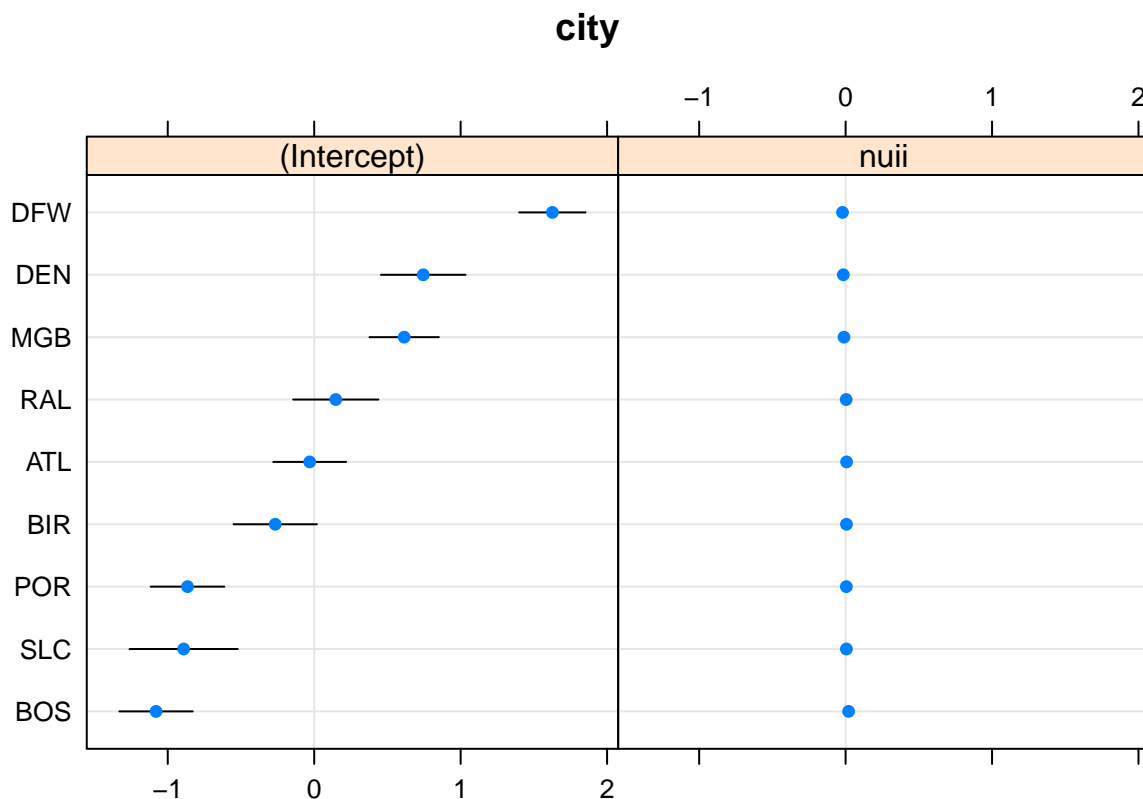
```
(Intr)
nuii -0.877
```

```
EBE <- ranef(mod.mix, condVar=TRUE)
#condVar(EBE)
VarCorr(mod.mix)
```

```
Groups   Name          Std.Dev. Corr
city     (Intercept) 0.903959
          nuii       0.013719 -0.893
Residual                0.474670
```

```
#se.ranef(mod.mix)
dotplot(EBE)
```

\$city



```
library(nlme)
mod.mix2 <- lme(richtol ~ 1+nuii, random=list(city=pdSymm(~1+nuii)),
               method="REML", data=rtol2)
```

```
summary(mod.mix2)
```

Linear mixed-effects model fit by REML

Data: rtol2

	AIC	BIC	logLik
	423.8282	445.1692	-205.9141

Random effects:

Formula: ~1 + nuii | city

Structure: General positive-definite

	StdDev	Corr
(Intercept)	0.90400645	(Intr)
nuii	0.01371996	-0.893
Residual	0.47466924	

Fixed effects: richtol ~ 1 + nuii

	Value	Std.Error	DF	t-value	p-value
(Intercept)	5.418387	0.30516361	251	17.75568	0e+00
nuii	0.019431	0.00478958	251	4.05695	1e-04

Correlation:

(Intr)	
nuii	-0.877

Standardized Within-Group Residuals:

	Min	Q1	Med	Q3	Max
	-3.42887458	-0.56104881	0.01444248	0.57988474	4.10530930

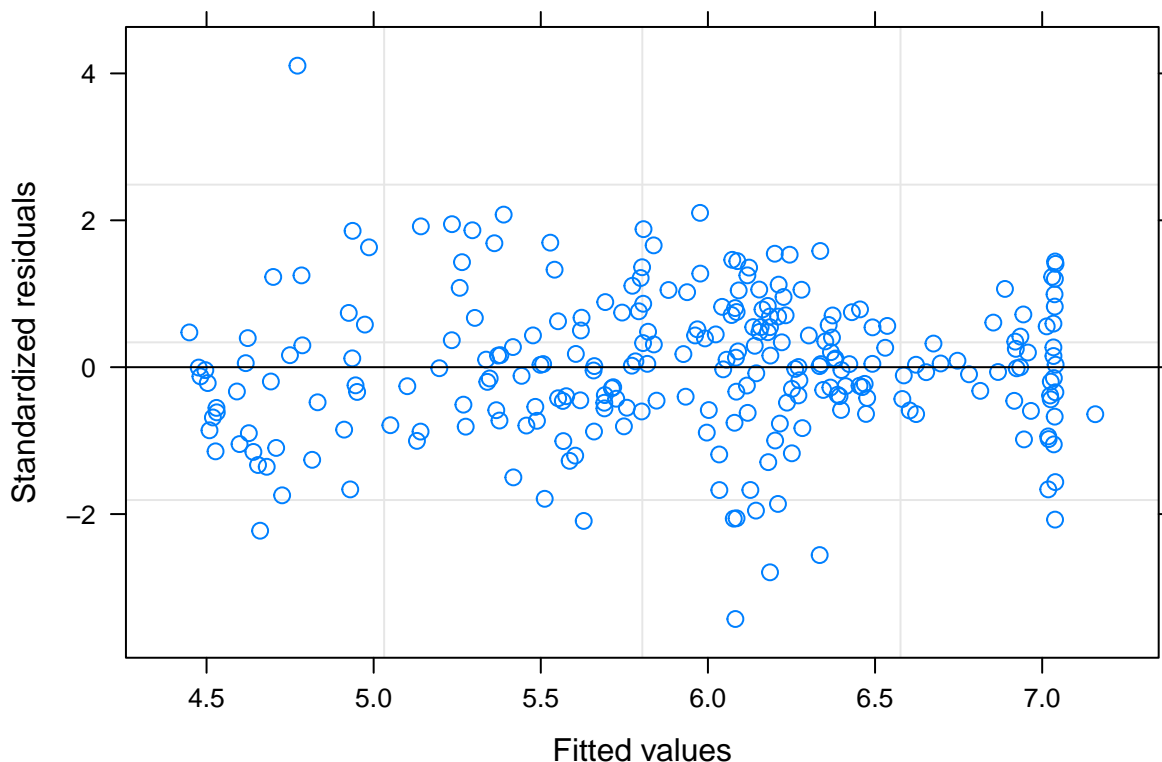
Number of Observations: 261

Number of Groups: 9

```
ranef(mod.mix2)
```

	(Intercept)	nuii
ATL	-0.03074633	0.006745038
BIR	-0.26613286	0.006025201
BOS	-1.07984752	0.020616795
DEN	0.74487448	-0.015656374
DFW	1.62647812	-0.021060559
MGB	0.61444883	-0.010954518
POR	-0.86495303	0.004903514
RAL	0.14726265	0.003899805
SLC	-0.89138435	0.005481098

```
plot(mod.mix2)
```



```
# Matrices de varianza-covarianza marginales y condicionales
getVarCov(mod.mix2, type="conditional", individual="ATL")
getVarCov(mod.mix2, type="marginal", individual="ATL")
```

1.11.0.2. Modelo para la parte fija (ecosistemas)

Supongamos que queremos modelar cómo el intercepto y la pendiente comunes (la parte fija del modelo) depende de otras variables, como la temperatura promedio de la ciudad.

Esto se realiza de manera muy simple, agregando posibles variables explicativas (regresoras) a la parte fija (es decir, a nivel de ciudad).

$$Y_{ij} = (\beta_0 + \beta_{0ti}t_i) + b_{i0} + ((\beta_1 + \beta_{1ti}t_i) + b_{1i})x_{ij} + e_{ij}$$

$$\begin{bmatrix} b_{0i} \\ b_{1i} \end{bmatrix} \stackrel{i.i.d.}{\sim} N \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_0^2 & \sigma_{01}^2 \\ \sigma_{01}^2 & \sigma_1^2 \end{bmatrix} \right), \quad e_{ij} \stackrel{i.i.d.}{\sim} N(0, \sigma_e^2), \quad \mathbf{b}_i, e_{ij} \text{ independientes.}$$

```
euse.env <- read_csv("~/Google Drive/12-Spring2018/Curso Argentina/CursoBuenosAires2018/
AveTemp <- tapply(euse.env$AnnMeanT, euse.env$CITY, mean)
```



```

site <- as.numeric(ordered(rtol2$city))
temp.full <- as.vector(AveTemp[site])
mod.mix3 <- lmer(richtol~1+nuii+temp.full+nuii:temp.full +
                (1+nuii|city), data=rtol2)
summary(mod.mix3)

```

Linear mixed model fit by REML ['lmerMod']

Formula: richtol ~ 1 + nuii + temp.full + nuii:temp.full + (1 + nuii | city)

Data: rtol2

REML criterion at convergence: 424

Scaled residuals:

Min	1Q	Median	3Q	Max
-3.4336	-0.5517	0.0236	0.6143	4.0996

Random effects:

Groups	Name	Variance	Std.Dev.	Corr
city	(Intercept)	0.789369	0.88846	
	nuii	0.000207	0.01439	-0.93
Residual		0.225589	0.47496	

Number of obs: 261, groups: city, 9

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	4.3192210	1.0397345	4.154
nuii	0.0231601	0.0172792	1.340
temp.full	0.0885867	0.0802678	1.104
nuii:temp.full	-0.0002978	0.0013378	-0.223

Correlation of Fixed Effects:

	(Intr)	nuii	tmp.fl
nuii	-0.918		
temp.full	-0.957	0.879	
nui:tmp.fll	0.877	-0.957	-0.916

```
ranef(mod.mix3)
```

\$city

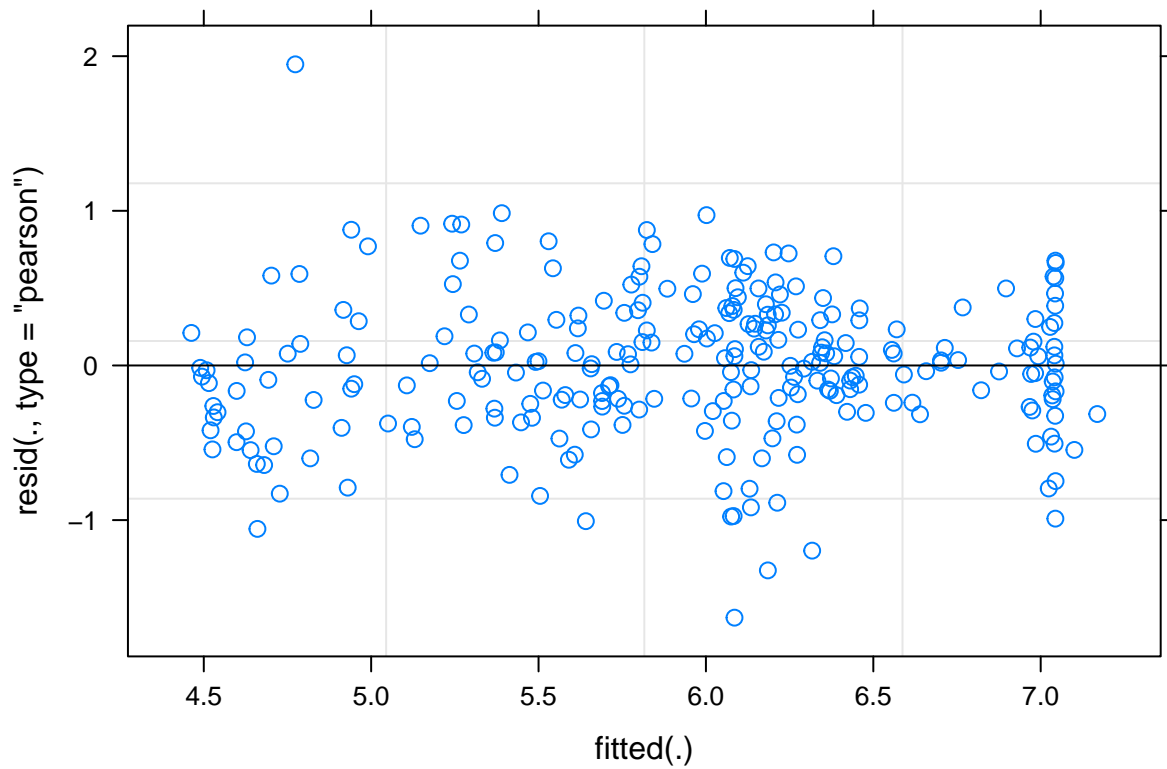
	(Intercept)	nuii
ATL	-0.38448383	0.008894917
BIR	-0.60307519	0.008427514

```

BOS -0.73307270  0.017563937
DEN  1.03155525 -0.017060022
DFW  1.10797652 -0.018738261
MGB  1.04076556 -0.013338205
POR -0.72509854  0.004720425
RAL -0.07963226  0.004812283
SLC -0.65493480  0.004717412

```

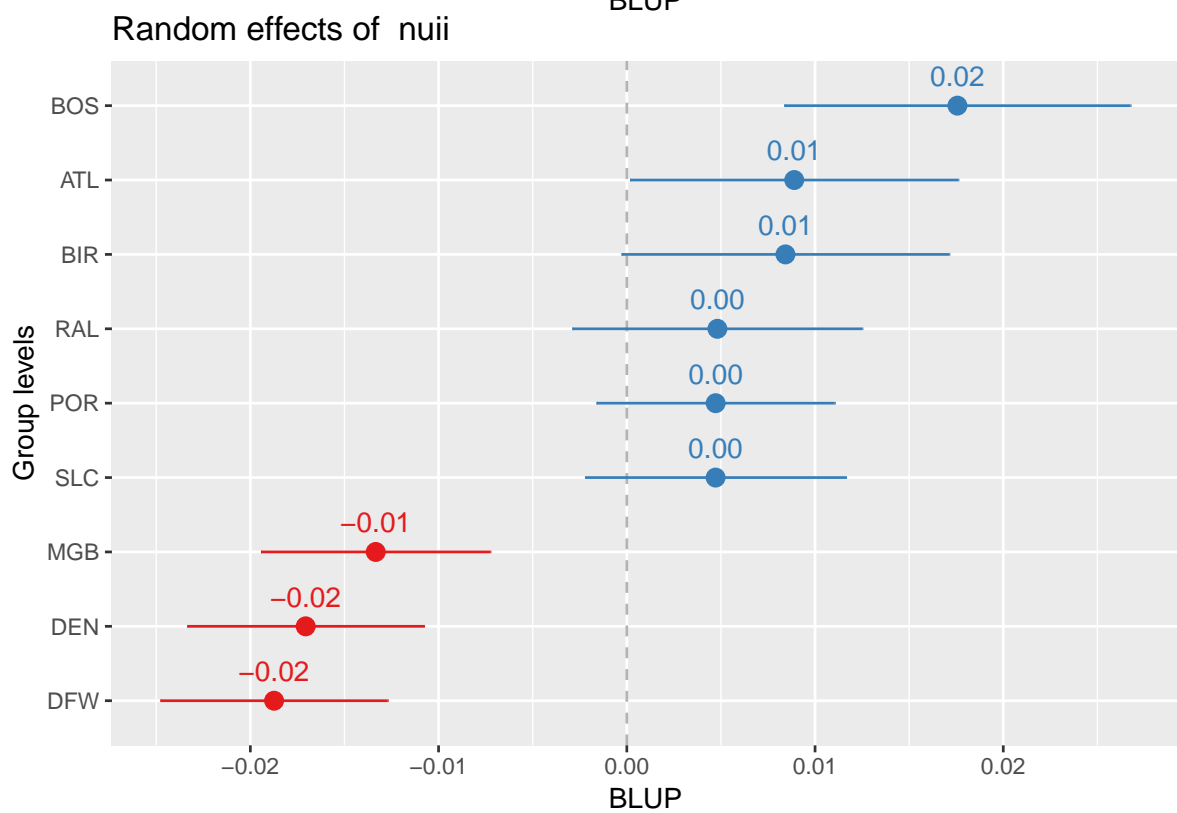
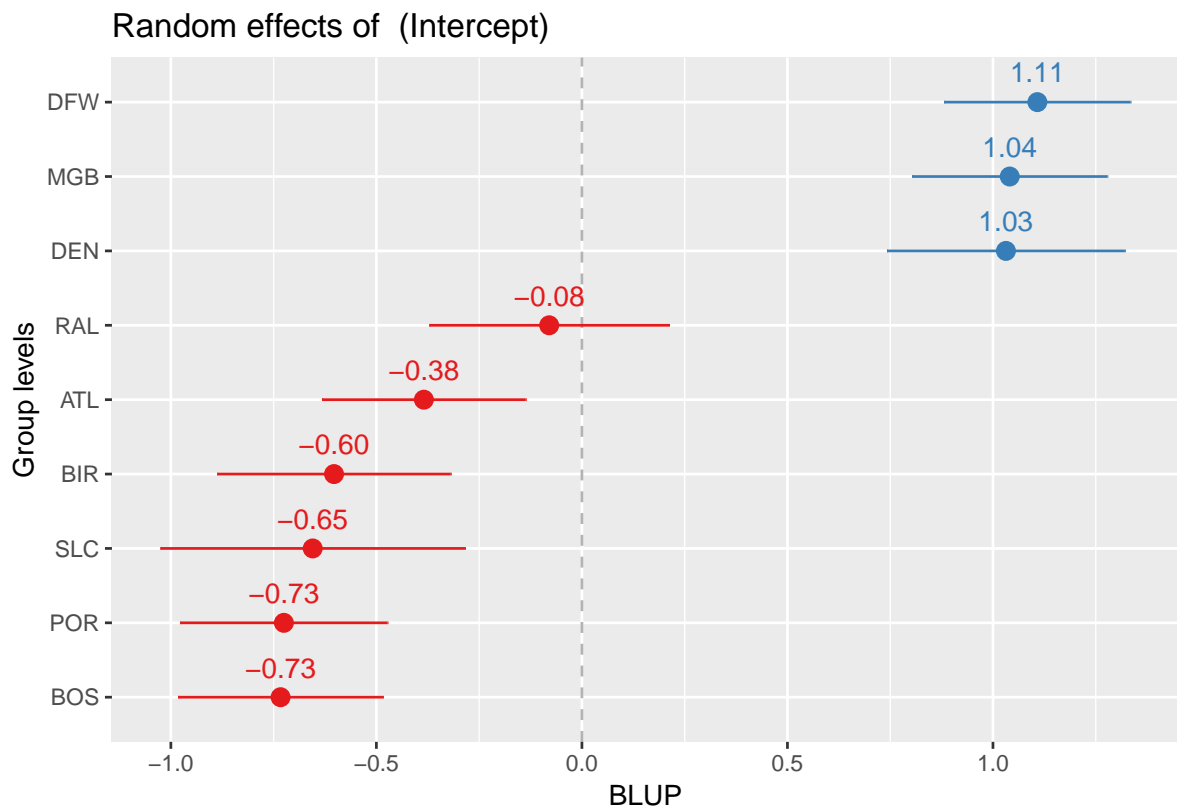
```
plot(mod.mix3)
```



```

library(sjPlot)
sjp.lmer(mod.mix3, facet.grid = FALSE,
  sort.est = "sort.all",
  y.offset = .4)

```



```
mod.mix4 <- lmer(richtol~1+nuii+(1+nuii|city), data=rtol2,
                method="ML")
mod.mix5 <- lmer(richtol~1+nuii+temp.full+nuii:temp.full +
                (1+nuii|city), data=rtol2,
                method="ML")
anova(mod.mix4, mod.mix5)
```

Data: rtol2

Models:

mod.mix4: richtol ~ 1 + nuii + (1 + nuii | city)

mod.mix5: richtol ~ 1 + nuii + temp.full + nuii:temp.full + (1 + nuii |

mod.mix5: city)

	Df	AIC	BIC	logLik	deviance	Chisq	Chi	Df	Pr(>Chisq)
mod.mix4	6	412.85	434.24	-200.43	400.85				
mod.mix5	8	411.84	440.36	-197.92	395.84	5.0084		2	0.08174 .

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
city_ag <- read_csv("~/Google Drive/12-Spring2018/Curso Argentina/CursoBuenosAires2018")
city_ag<-cbind(city_ag[,1:2],city_ag[,3:5]/100)
city_ag[order(city_ag[,1]), ]
```

	CITY	SUID	PNLCD7	PNLCD8	PNLCD78
3	ATL	ACFB	0.071979288	0.16511416	0.23709344
4	BIR	MOBL	0.036328628	0.16987809	0.20620672
1	BOS	NECB	0.003375359	0.07521565	0.07859101
6	DEN	SPLT	0.635333298	0.22640489	0.86173819
7	DFW	TRIN	0.208222135	0.54798380	0.75620594
5	MGB	WMIC	0.009334345	0.76057259	0.76990694
9	POR	WILL	0.074810446	0.12115522	0.19596567
2	RAL	ALBE	0.050754986	0.21494498	0.26569997
8	SLC	GRSL	0.000000000	0.02252253	0.02252253

```
ag<-city_ag[order(city_ag[,1]), ][,5]
```

```
ag.full <- as.vector(ag[site])
```

```
ag.cat <- ag.full>0.5
```

adding background agriculture land use

```
euse.lmer2 <- lmer(richtol ~ nuii+ag.full+nuii:ag.full+
                  (1+nuii|site), data=rtol2)
```

```
summary(euse.lmer2) ## display no plonger works for lmer
```

Linear mixed model fit by REML ['lmerMod']

Formula: richtol ~ nuii + ag.full + nuii:ag.full + (1 + nuii | site)

Data: rtol2

REML criterion at convergence: 404.6

Scaled residuals:

	Min	1Q	Median	3Q	Max
	-3.4773	-0.5130	0.0242	0.5776	4.1154

Random effects:

Groups	Name	Variance	Std.Dev.	Corr
site	(Intercept)	1.844e-01	0.429440	
	nuii	2.741e-05	0.005236	-0.34
Residual		2.253e-01	0.474617	

Number of obs: 261, groups: site, 9

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	4.458446	0.241639	18.451
nuii	0.034116	0.003663	9.315
ag.full	2.529382	0.493903	5.121
nuii:ag.full	-0.038842	0.007068	-5.496

Correlation of Fixed Effects:

	(Intr)	nuii	ag.fll
nuii		-0.419	
ag.full	-0.781	0.319	
nuii:ag.fll	0.337	-0.792	-0.406

```
M2.coef <- coef (euse.lmer2)
```

```
a.hat.M2 <- M2.coef[[1]][,1] + M2.coef[[1]][,3]*ag
```

```
b.hat.M2 <- M2.coef[[1]][,2] + M2.coef[[1]][,4]*ag
```

```
# plot estimated intercepts and slopes
```

```
# Figure 10.14
```

```
# postscript(file=paste(plotDIR, "richtol2.eps", sep="/"), width=4.75, height=3, hor
```

```
par (mfrow=c(1,2), mar=c(3,3,3,0.25), mgp=c(1.25,0.125,0),  
     las=1, tck=0.01)
```

```
plot (ag, a.hat.M2, cex.lab=0.75, cex.axis=0.75,  
      xlab="Background ag", ylab="regression intercept", pch=20)
```

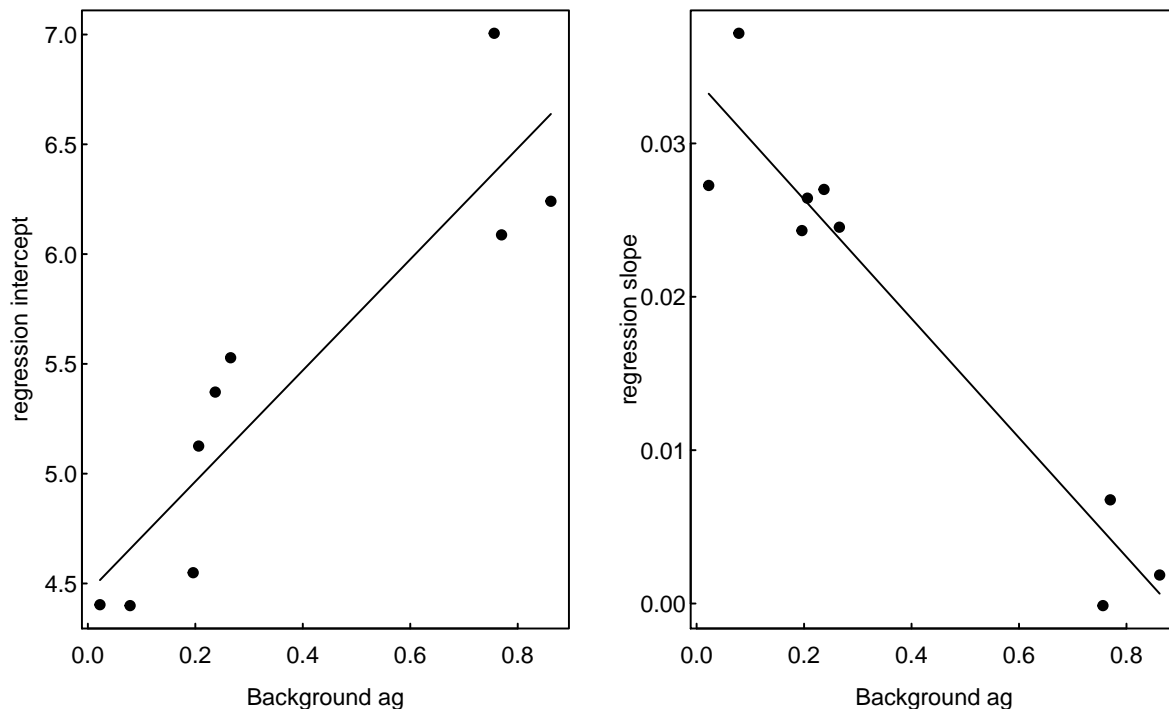
```
curve (fixef(euse.lmer2)["(Intercept)"] +  
       fixef(euse.lmer2)["ag.full"]*x, lwd=1,
```

```

      col="black", add=TRUE)
text(ag, levels(rtol2$city), adj=c(.5,1),cex=0.5)

plot (ag, b.hat.M2, cex.lab=0.75, cex.axis=0.75,
      xlab="Background ag", ylab="regression slope", pch=20)
curve (fixef(euse.lmer2)["nuii"] +
      fixef(euse.lmer2)["nuii:ag.full"]*x, lwd=1,
      col="black", add=TRUE)
text(ag, levels(rtol2$city), adj=c(.5,1),cex=0.5)

```



```

mod.mix6<- lmer(richtol ~ nuii+ag.full+nuii:ag.full+(1+nuii|site),
               data=rtol2,method="ML")
anova(mod.mix4,mod.mix6)

```

Data: rtol2

Models:

mod.mix4: richtol ~ 1 + nuii + (1 + nuii | city)

mod.mix6: richtol ~ nuii + ag.full + nuii:ag.full + (1 + nuii | site)

	Df	AIC	BIC	logLik	deviance	Chisq	Chi Df	Pr(>Chisq)
mod.mix4	6	412.85	434.24	-200.43	400.85			
mod.mix6	8	399.67	428.18	-191.83	383.67	17.186	2	0.0001854 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Capítulo 2

Modelos Lineales Generalizados y Mixtos

2.1. Modelos Mixtos

- **Modelo Lineal:**

$$E(\mathbf{y}) = \mathbf{X}\beta, \quad \mathbf{y} \sim N(E(\mathbf{y}), \sigma^2 \mathbf{I}).$$

- **Modelo Lineal Mixto:**

$$\mathbf{y}|\mathbf{u} \sim N(\mathbf{X}\beta + \mathbf{Z}\mathbf{u}, \mathbf{R}), \quad u \sim N(\mathbf{0}, \mathbf{G}).$$

- **Modelo Lineal Generalizado:**

$$g(E(y_i)) = \mathbf{X}_i\beta, \quad y_i \text{ i.d. } \textit{dist}(\beta, \xi).$$

- **Modelo Lineal Generalizado Mixto:**

$$g(E(y_i|\mathbf{u})) = \mathbf{X}_i\beta + \mathbf{Z}_i\mathbf{u}, \quad y_i|\mathbf{u} \text{ indep. } \textit{dist}(\beta, \xi); \mathbf{u} \sim N(\mathbf{0}, \mathbf{G}).$$

2.2. Modelos Lineales Generalizados (MLG)

Un modelo lineal generalizado es un modelo que vincula las respuestas (variables “dependientes”) con otras variables “independientes” o “explicativas”. Tenemos que considerar tres componentes (Agresti, 2015):

1. La componente aleatoria (la distribución de las Y_i). En general, se supone que las Y_i 's son independientes, con una distribución que sea una familia exponencial lineal (por ejemplo: normal, binomial, Poisson, gamma, etc.)

Por ejemplo, la distribución Poisson cuya densidad es igual

$$f(y; \theta) = \frac{e^{-\theta} \theta^y}{y!}, \quad y = 0, 1, \dots,$$

se puede re-escribir como

$$f(y; \theta) = \exp(y \log \theta - \theta - \log y!).$$

Por lo tanto, la distribución Poisson pertenece a la familia exponencial. Además, tiene forma canónica ya que $a(y) = y$ y el parámetro natural es $\log \theta$.

2. La componente sistemática, que indica la relación entre las variables independientes. Éste es un modelo lineal (es decir, los parámetros entran linealmente al modelo). Por ejemplo,

$$\eta_i = \alpha + \beta_1 x_{1i} + \beta_2 x_{2i}.$$

3. La función de enlace, que es la que vincula la media (esperanza) de la distribución de las Y_i con la componente sistemática. Por ejemplo,

$$g(\mu_i) = \log(\mu_i) = \alpha + \beta_1 x_{1i} + \beta_2 x_{2i}.$$

En el modelo lineal se tiene que $E(Y_i | \mathbf{x}_i) = \mathbf{x}_i^t \boldsymbol{\beta}$. Esto es, el interés es modelar la media de la variable respuesta como una función lineal de las variables explicativas. La varianza se asume constante o no constante. En un MLG el interés es modelar una función no lineal de la media a través de una combinación lineal de las variables explicativas. Una de las razones para hacer esto es que la media de una variable respuesta en general podría tener restricciones tales como estar en el intervalo $(0, 1)$ (Bernoulli) o positiva (Poisson). Una forma de asegurar esto es usando transformaciones de la media (la cantidad $\mathbf{x}_i^t \boldsymbol{\beta}$ no tiene restricciones). Por ejemplo, en el caso de una distribución Poisson si modelamos la media como $E(Y_i | \mathbf{x}_i) = \lambda_i = \mathbf{x}_i^t \boldsymbol{\beta}$ esta cantidad podría ser negativa, o en el caso de la distribución Bernoulli $E(Y_i | \mathbf{x}_i) = p_i = \mathbf{x}_i^t \boldsymbol{\beta}$ podría no estar en el intervalo $(0, 1)$.

La siguiente tabla presenta algunas de las distribuciones conocidas que pertenecen a la familia exponencial con sus respectivas medias, varianzas y funciones de enlace canónicas. Es importante mencionar que a diferencia de la distribución normal donde la varianza se puede modelar separadamente de la media en las distribuciones de la familia exponencial la varianza queda definida como una función de la media. Por ejemplo, en la distribución Bernoulli $Var(Y_i) = p_i(1 - p_i) = \mu_i(1 - \mu_i)$ y en la Poisson $Var(Y_i) = \lambda_i = \mu_i$. Esta relación entre la media y la varianza es característica de los MLG. Es más, $Var(Y_i)$ suele escribirse como $Var(Y_i) = \nu(\mu_i)$ para enfatizar en la relación media-varianza en un MLG.

Distribuciones comúnmente usadas con sus medias, varianzas y funciones de enlace canónicas

Distribución	Media	Varianza	Función de enlace $g(\mu) = \mathbf{x}_i^t \boldsymbol{\beta} = \eta_i$	$\mu = g^{-1}(\eta_i)$
Normal(μ, σ^2)	μ	σ^2	μ (Identidad)	η_i
Gamma(α, β)	α/β	α/β^2	$\log(\mu)$	$1/\eta_i$
Poisson(λ)	λ	λ	$\log(\lambda)$	e^{η_i}
Bernoulli(p)	p	$p(1-p)$	$\text{logit}(p) = \log[p/(1-p)]$	$e^{\eta_i}/(1 + e^{\eta_i})$

2.3. Regresión logística para datos binarios o proporciones binomiales

En muchas aplicaciones tenemos una variable respuesta que puede tener solamente dos posibles resultados (vivo o muerto, macho o hembra, enfermo o sano, mejora o no mejora, etc.).

Típicamente uno de los resultados se codifica como 0 (“fracaso”) y el otro como 1 (“éxito”). La codificación es arbitraria, pero si usamos 0, 1 al calcular la media de los datos obtenemos la proporción de éxitos.

Como en modelos de regresión lineal o análisis de varianza, nos interesa estudiar cómo el promedio (valor esperado) depende de una o más variables explicatorias o factores.

El modelo más simple que podríamos usar es el de regresión lineal simple, en la escala de los datos:

$$E(Y) = \beta_0 + \beta_1 x.$$

Se puede probar que la media de la variable dependiente es la proporción de éxitos en la población. Por lo tanto, lo que estamos modelando es cómo la proporción de éxitos cambia con las x . Este modelo tiene tres problemas fundamentales:

- Las Y 's no tienen distribución normal.
- Las varianzas no son constantes: $\text{Var}(Y) = E(Y)[1 - E(Y)]$.
- El modelo podría predecir valores mayores que 1 o menores que 0 (recordemos que estamos modelando proporciones o probabilidades!)

Por todo esto, el modelo lineal no se usa, y se usa un modelo lineal generalizado. Esto es equivalente a modelar la media como

$$E(Y) = \frac{\exp(\beta_0 + \beta_1 x)}{1 + \exp(\beta_0 + \beta_1 x)}$$

Esta curva tiene las siguientes propiedades:

- A medida que x crece, $E(Y)$ se acerca a 1 (si $\beta_1 > 0$) o a 0 (si $\beta_1 < 0$).
- La curva tiene forma de S. Es monótona (o crece o decrece siempre). $E(Y) = 1/2$ cuando $x = -\beta_0/\beta_1$.

Una interpretación de esta curva es mediante la función de enlace “logit”:

$$\log \left(\frac{E(Y)}{1 - E(Y)} \right) = \beta_0 + \beta_1 x.$$

donde el logaritmo es el logaritmo natural, base e (a veces se denota como \ln). Podemos ver que en esta escala el modelo es lineal. El cociente entre las probabilidades de éxito y de fracaso se denomina *odds* (chances).

Probabilidad de Éxito $E(Y)$	<i>Odds</i> : $\frac{E(Y)}{1-E(Y)}$	Logits: $\log \left(\frac{E(Y)}{1-E(Y)} \right)$
0	0	$-\infty$
0.5	1	0
1	∞	∞

Usando la escala de odds podemos interpretar la pendiente: el valor de $\exp(\beta_1)$ indica por cuánto se multiplican las chances de éxito cuando aumentamos la variable independiente en una unidad. Por ejemplo, si la relación entre dosis y probabilidad de mejorar una condición está dada por

$$E(Y) = \frac{\exp(3 + 2x)}{1 + \exp(3 + 2x)},$$

o

$$\log \left(\frac{E(Y)}{1 - E(Y)} \right) = 3 + 2x.$$

La pendiente indica que si incrementamos la dosis en una unidad, las chances de mejorar se multiplican por $\exp(2) = 7,39$ (es decir, aumentan un 739 %). Observar que esto no es lo mismo que decir que la probabilidad de mejorar se incrementa 7.39 unidades (recuerden que este modelo es no lineal en la escala de probabilidad).

Cuando usamos este modelo para datos binomiales (en vez de binarios), la idea es la misma, ya que debemos recordar que los datos binomiales representan la cantidad total

de éxitos en n ensayos independientes. Seguimos modelando la probabilidad de éxito, y por lo tanto interpretamos los resultados de la misma manera.

Ejemplo 2.1 (Satélites (Agresti, 2015)). Los siguientes datos representan la cantidad de satélites (machos residiendo cerca) en cangrejos hembras (“horseshoe crabs”). Datos en Agresti (2015). Se desea saber si la presencia de satélites depende de ciertas características (peso, color, ancho, etc.)

Podemos modelar la probabilidad de que un cangrejo tenga satélite como un modelo de regresión logística múltiple.

La variable color es ordinal (1=medio claro, 2=medio, 3=medio oscuro, 4=oscura).

La variable espina es ordinal (1=ambas espinas en buenas condiciones, 2=una gastada o quebrada, 3=ambas gastadas o quebradas).

La variable respuesta y es 1 si el número satélites es mayor que 0, y 0 si el número de satélites es 0.

Podemos usar el siguiente modelo para estos datos:

$$\text{logit}(\mu_i) = \beta_0 + \beta_1 \text{peso}_i + \beta_2 \text{color}_i,$$

donde $Y_i \stackrel{\text{Indep.}}{\sim} \text{Binaria}(\mu_i)$.

```
head(crab)
```

```
# A tibble: 6 x 7
```

	crab	satellites	weight	width	color	spine	y
	<int>	<int>	<dbl>	<dbl>	<int>	<int>	<int>
1	1	8	3.05	28.3	2	3	1
2	2	0	1.55	22.5	3	3	0
3	3	9	2.30	26.0	1	1	1
4	4	0	2.10	24.8	3	3	0
5	5	4	2.60	26.0	3	3	1
6	6	0	2.10	23.8	2	3	0

```
mod1 <- glm(y ~ weight + color, family="binomial", data=crab)
summary(mod1)
```

Call:

```
glm(formula = y ~ weight + color, family = "binomial", data = crab)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-----	----	--------	----	-----

-2.1596 -0.9998 0.5237 0.8825 1.9109

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-2.0316	1.1161	-1.820	0.0687 .
weight	1.6531	0.3825	4.322	1.55e-05 ***
color	-0.5142	0.2234	-2.302	0.0213 *

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 225.76 on 172 degrees of freedom

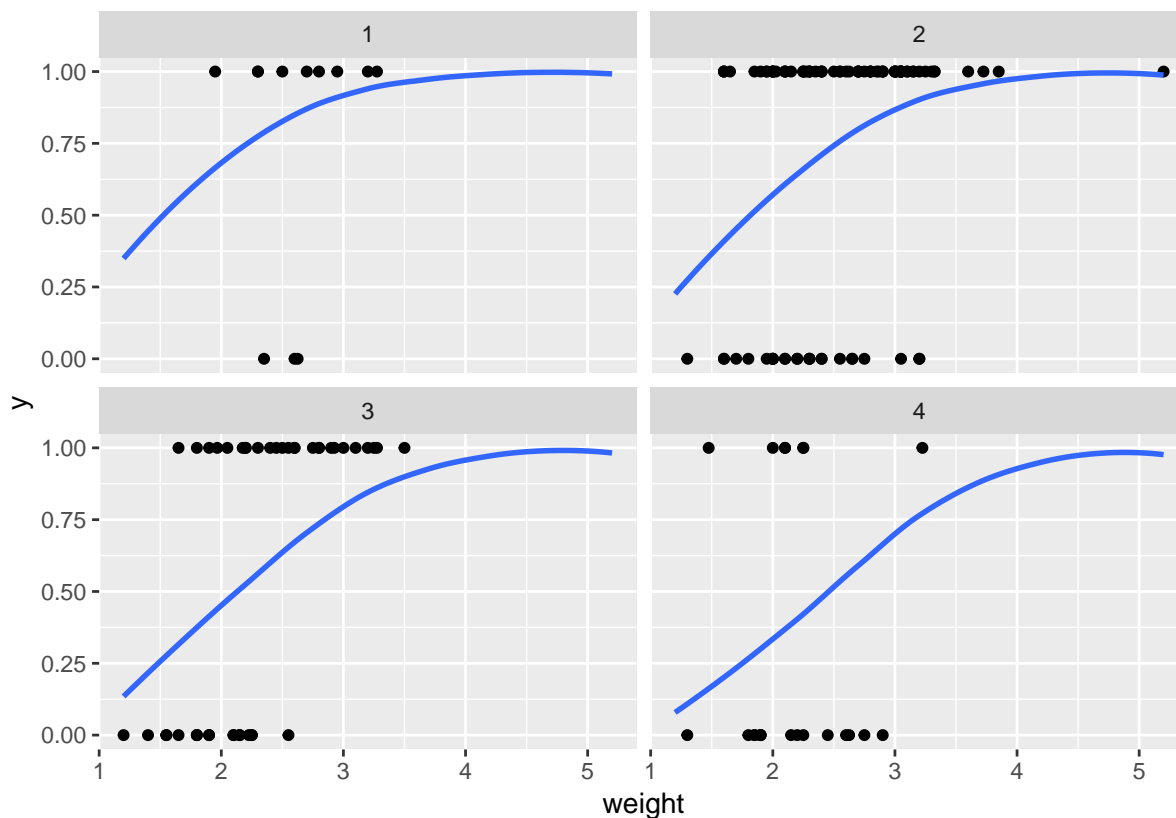
Residual deviance: 190.27 on 170 degrees of freedom

AIC: 196.27

Number of Fisher Scoring iterations: 4

Predicciones

```
library(ggplot2)
xmin <- min(crab$weight)
xmax <- max(crab$weight)
predicted1 <- data.frame(cbind(weight=seq(xmin,xmax,length.out=100)),
                          color=1)
predicted2 <- data.frame(cbind(weight=seq(xmin,xmax,length.out=100)),
                          color=2)
predicted3 <- data.frame(cbind(weight=seq(xmin,xmax,length.out=100)),
                          color=3)
predicted4 <- data.frame(cbind(weight=seq(xmin,xmax,length.out=100)),
                          color=4)
predicted <- rbind(predicted1, predicted2, predicted3, predicted4)
predicted$logity <- predict(mod1, predicted)
predicted$y <- exp(predicted$logity)/(1+exp(predicted$logity))
p <- ggplot(crab, aes(x=weight, y=y)) + geom_point()
p + facet_wrap(~factor(color)) +
  geom_smooth(data=predicted, se=FALSE)
```



```
mod2 <- glm(y~weight+as.factor(color), family="binomial",
            data=crab)
summary(mod2)
```

Call:

```
glm(formula = y ~ weight + as.factor(color), family = "binomial",
    data = crab)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-2.1908	-1.0144	0.5101	0.8683	2.0751

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-3.2572	1.1985	-2.718	0.00657 **
weight	1.6928	0.3888	4.354	1.34e-05 ***
as.factor(color)2	0.1448	0.7365	0.197	0.84410
as.factor(color)3	-0.1861	0.7750	-0.240	0.81019
as.factor(color)4	-1.2694	0.8488	-1.495	0.13479

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 225.76 on 172 degrees of freedom
Residual deviance: 188.54 on 168 degrees of freedom
AIC: 198.54

Number of Fisher Scoring iterations: 4

```
anova(mod2)
```

Analysis of Deviance Table

Model: binomial, link: logit

Response: y

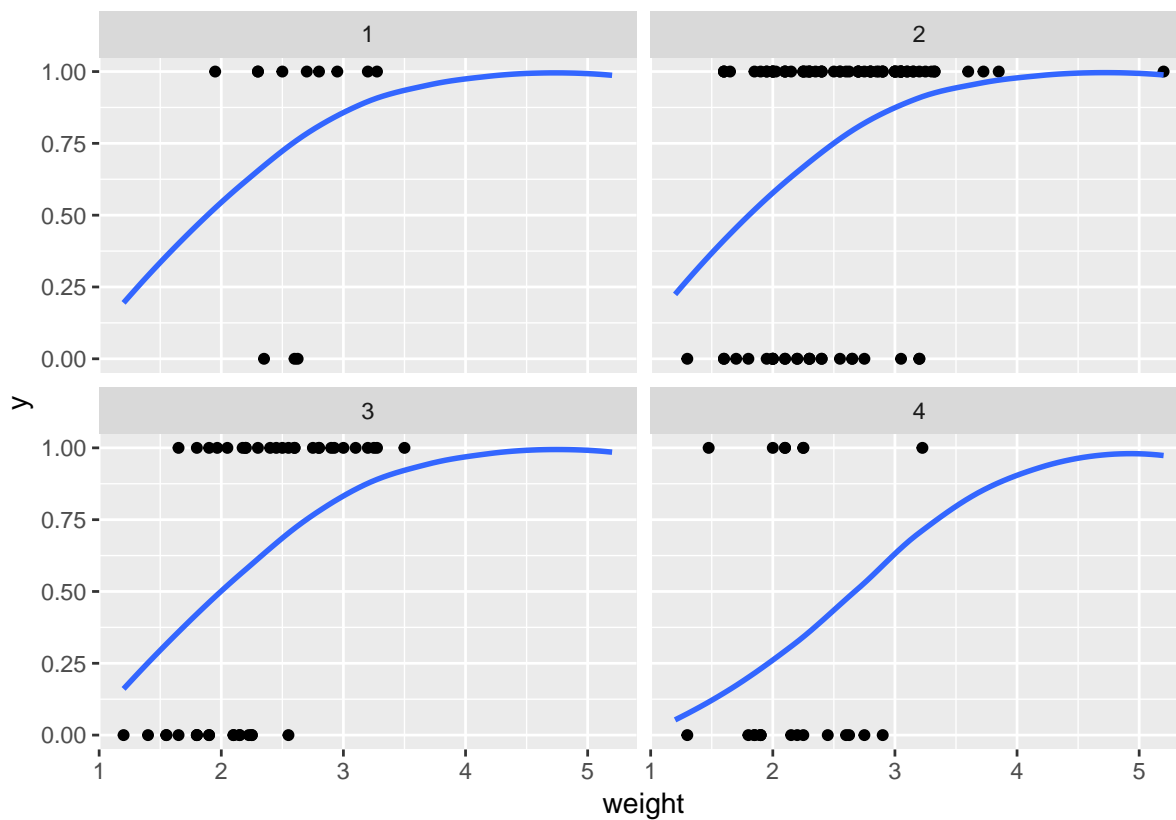
Terms added sequentially (first to last)

	Df	Deviance	Resid. Df	Resid. Dev
NULL			172	225.76
weight	1	30.0214	171	195.74
as.factor(color)	3	7.1949	168	188.54

```
1-pchisq(7.1949, 3)
```

```
[1] 0.06593839
```

```
predictedmod2<- rbind(predicted1, predicted2, predicted3, predicted4)
predictedmod2$logity <- predict(mod2, predictedmod2)
predictedmod2$y <- exp(predictedmod2$logity)/(1+exp(predictedmod2$logity))
p2 <- ggplot(crab, aes(x=weight, y=y))+geom_point()
p2 + facet_wrap(~factor(color)) + geom_smooth(data=predictedmod2, se=FALSE)
```



```
anova(mod1, mod2)
```

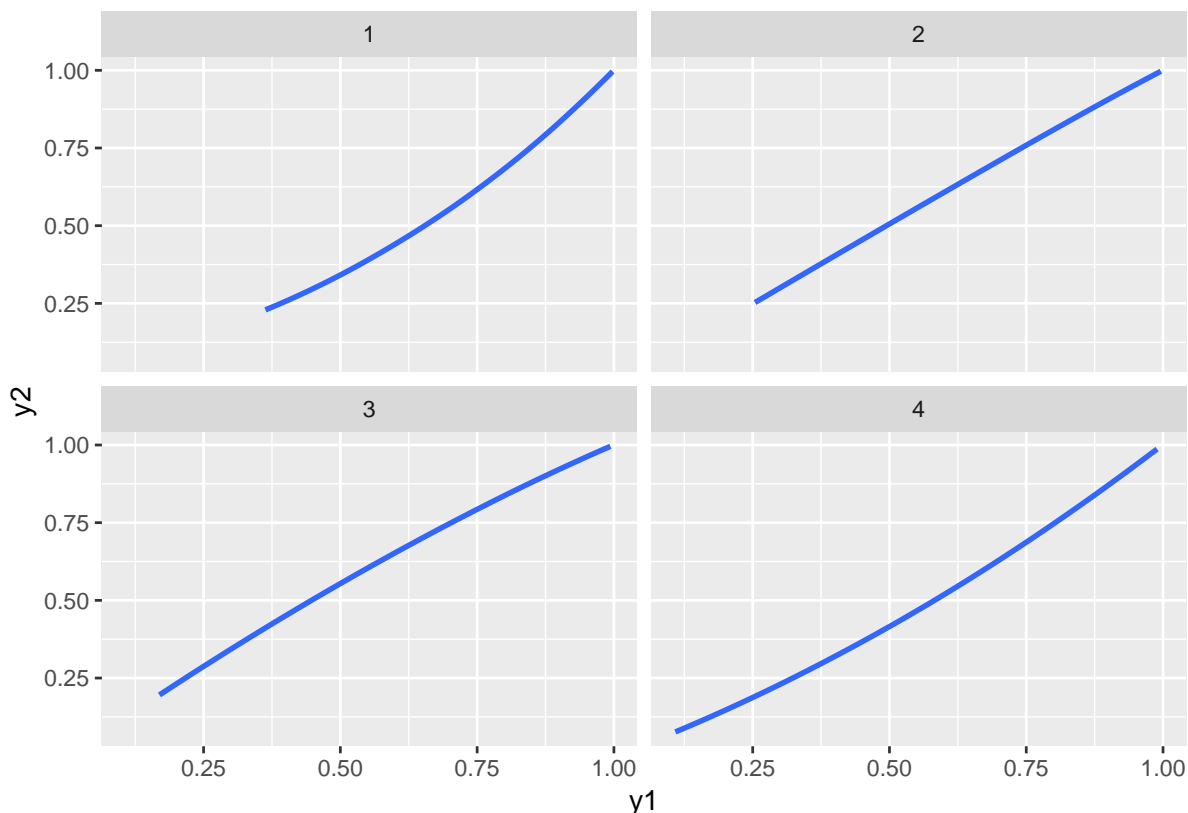
Analysis of Deviance Table

Model 1: $y \sim \text{weight} + \text{color}$

Model 2: $y \sim \text{weight} + \text{as.factor}(\text{color})$

	Resid. Df	Resid. Dev	Df	Deviance
1	170	190.27		
2	168	188.54	2	1.7265

```
comp <- data.frame(y1=predicted$y, y2=predictedmod2$y, color=predicted$color )
p3 <- ggplot(comp, aes(x=y1, y=y2))
p3 + facet_wrap(~factor(color)) + geom_smooth(method=loess, se=FALSE)
```



2.4. Regresión Poisson

Si la variable respuesta son conteos una forma de modelar estos datos usando un modelo lineal es aplicando la transformación \sqrt{y} para tratar de obtener varianza constante en los residuales (usando el método delta se puede mostrar que $Var(\sqrt{Y}) \approx 1/4$, cuando $Y \sim Poisson(\lambda)$; esta aproximación funciona mejor cuando μ es grande). Sin embargo, tal como hemos mencionado anteriormente la dificultad en las interpretaciones y las implicaciones de modelar datos transformados limitan esta práctica. Por esta razón, una forma más razonable de modelar datos de conteos es asumir que $Y_i \sim Poisson(\mu_i)$.

La idea es modelar la media μ_i en función de las covariables \mathbf{x}_i , por ejemplo,

$$\log(\mu_i) = \beta_0 + \beta_1 x_i.$$

Recuerde que si $Y_i \sim Poisson(\mu_i)$ entonces la función de densidad de Y_i esta dada por

$$f(y_i; \mu_i) = \frac{e^{-\mu_i} \mu_i^{y_i}}{y_i!}, \quad y_i = 0, 1, 2, \dots$$

Ejemplo 2.2 (Satélites (revisitado)). Los siguientes datos representan la cantidad de satélites (machos residiendo cerca) en cangrejos hembras (“horseshoe crabs”). Datos en

Agresti (2015). Se desea saber si la presencia de satélites depende de ciertas características (peso, color, ancho, etc.)

Ahora podemos modelar la cantidad de satélites y (machos residiendo cerca) en cangrejos hembras (“horseshoe crabs”).

La variable color es ordinal (1=medio claro, 2=medio, 3=medio oscuro, 4=oscura).

La variable espina es ordinal (1=ambas espinas en buenas condiciones, 2=una gastada o quebrada, 3=ambas gastadas o quebradas).

Vamos a modelar la cantidad de satélites que un cangrejo hembra tiene en función de las variables disponibles.

```
mod3 <- glm(satellites~weight+width+color+spine, family="poisson"(link="log"), data=crab)
summary(mod3)
```

Call:

```
glm(formula = satellites ~ weight + width + color + spine, family = poisson(link = "log"),
     data = crab)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-3.0126	-1.8846	-0.5406	0.9448	4.9602

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-0.52848	0.94867	-0.557	0.57748
weight	0.47246	0.16490	2.865	0.00417 **
width	0.02753	0.04794	0.574	0.56588
color	-0.18493	0.06652	-2.780	0.00544 **
spine	0.03998	0.05681	0.704	0.48160

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 632.79 on 172 degrees of freedom
Residual deviance: 551.85 on 168 degrees of freedom
AIC: 917.15

Number of Fisher Scoring iterations: 6

```
mod4 <- update(mod3, .~ . - width)
summary(mod4)
```

```
Call:
glm(formula = satellites ~ weight + color + spine, family = poisson(link = "log"),
    data = crab)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-3.0359	-1.8986	-0.5127	0.9335	4.9750

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-0.009716	0.286144	-0.034	0.97291
weight	0.558179	0.069817	7.995	1.3e-15 ***
color	-0.190301	0.065806	-2.892	0.00383 **
spine	0.043999	0.056439	0.780	0.43563

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 632.79 on 172 degrees of freedom
 Residual deviance: 552.18 on 169 degrees of freedom
 AIC: 915.48

Number of Fisher Scoring iterations: 6

```
mod5 <- update(mod4, .~ . - spine + factor(spine))
summary(mod5)
```

Call:

```
glm(formula = satellites ~ weight + color + factor(spine), family = poisson(link = "log"),
    data = crab)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-3.0337	-1.8885	-0.5548	0.8798	4.8910

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	0.11096	0.27411	0.405	0.68561
weight	0.54099	0.07150	7.567	3.83e-14 ***
color	-0.19068	0.06590	-2.894	0.00381 **

```
factor(spine)2 -0.18531    0.20981   -0.883    0.37712
factor(spine)3  0.06567    0.11357    0.578    0.56311
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(Dispersion parameter for poisson family taken to be 1)

```
Null deviance: 632.79  on 172  degrees of freedom
Residual deviance: 550.83  on 168  degrees of freedom
AIC: 916.13
```

Number of Fisher Scoring iterations: 6

```
mod6 <- update(mod4, .~ . - spine )
summary(mod6)
```

Call:

```
glm(formula = satellites ~ weight + color, family = poisson(link = "log"),
     data = crab)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-2.9785	-1.9159	-0.5471	0.9181	4.8338

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	0.08855	0.25443	0.348	0.72783
weight	0.54588	0.06749	8.088	6.05e-16 ***
color	-0.17282	0.06155	-2.808	0.00499 **

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(Dispersion parameter for poisson family taken to be 1)

```
Null deviance: 632.79  on 172  degrees of freedom
Residual deviance: 552.79  on 170  degrees of freedom
AIC: 914.09
```

Number of Fisher Scoring iterations: 6

Interpretación:

- Coeficiente asociado con peso

- Coeficiente asociado con color
- Predicción del número promedio de satélites para una hembra de color 2 que pese 3.5. Intervalo de confianza para la predicción de la media?

```
predict(mod6, data.frame(weight=3.5, color=2), se.fit=TRUE,
        type="link" )
```

```
$fit
      1
1.653488
```

```
$se.fit
[1] 0.07092063
```

```
$residual.scale
[1] 1
```

```
predict(mod6, data.frame(weight=3.5, color=2), se.fit=TRUE,
        type="response" )
```

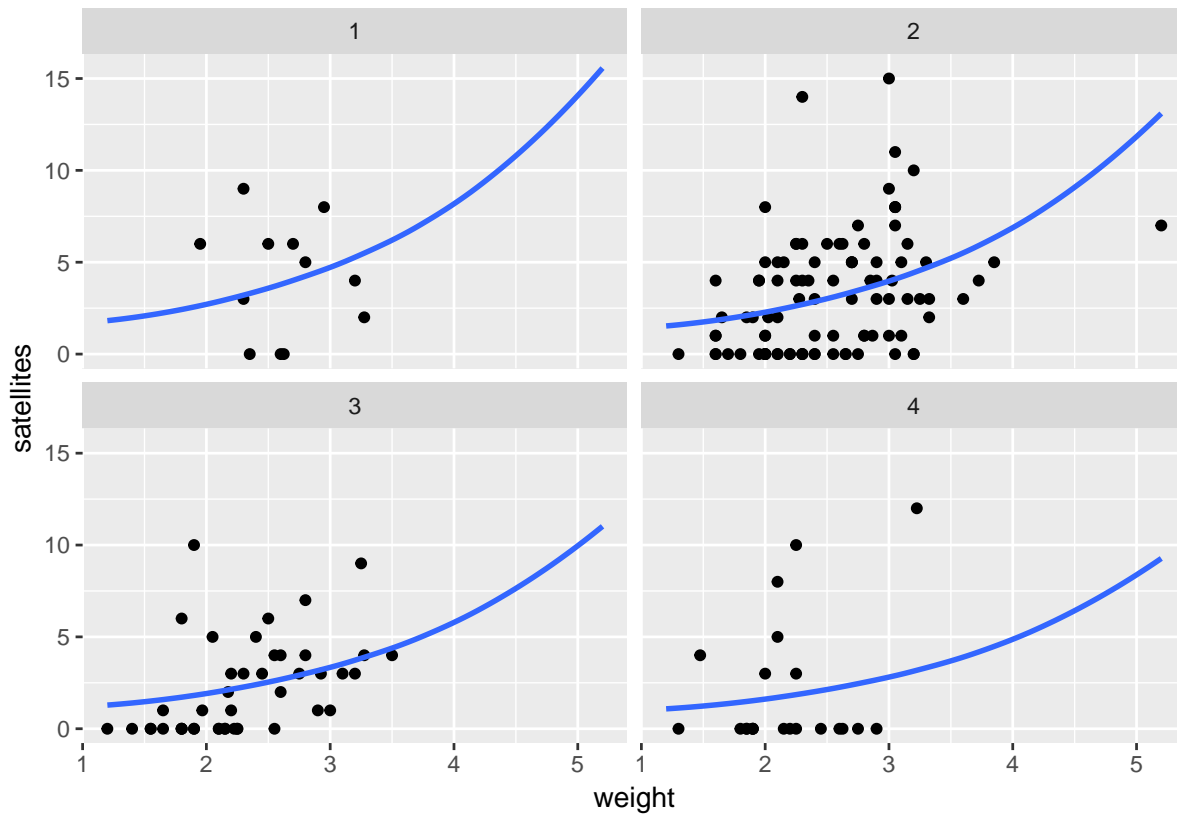
```
$fit
      1
5.225173
```

```
$se.fit
      1
0.3705726
```

```
$residual.scale
[1] 1
```

```
xmin<-min(crab$weight)
xmax<-max(crab$weight)
predicted1<-data.frame(cbind(weight=seq(xmin,xmax,length.out=100)),
                       color=1)
predicted2<-data.frame(cbind(weight=seq(xmin,xmax,length.out=100)),
                       color=2)
predicted3<-data.frame(cbind(weight=seq(xmin,xmax,length.out=100)), color=3)
predicted4<-data.frame(cbind(weight=seq(xmin,xmax,length.out=100)),
                       color=4)
predicted<- rbind(predicted1, predicted2, predicted3, predicted4)
predicted$logy<-predict(mod6,predicted)
predicted$satellites<-exp(predicted$logy)
p2 <- ggplot(crab, aes(x=weight, y=satellites))+geom_point()
```

```
p2 + facet_wrap(~factor(color)) + geom_smooth(data=predicted, se=FALSE)
```



```
AIC(mod6)
```

```
[1] 914.0914
```

```
BIC(mod6)
```

```
[1] 923.5513
```

```
Devchi2<-deviance(mod6)
```

```
Devchi2 / 170
```

```
[1] 3.251728
```

```
Pearsonchi2<-sum(residuals(mod6,type="pearson")^2)
```

```
Pearsonchi2 / 170
```

```
[1] 3.18719
```

2.5. Estimación por Máxima Verosimilitud

En un MLG especificamos una componente aleatoria (distribución) que pertenece a la familia exponencial, por lo tanto, podemos encontrar la función de verosimilitud L con las n observaciones independientes

$$L(\theta_i, \phi) = \prod_{i=1}^n f(y_i; \theta_i, \phi) \quad (2.1)$$

Recuerde que los parámetros θ_i dependen del vector de coeficientes β del modelo y ϕ representa algún parámetro de dispersión en la familia. El logaritmo de la función de verosimilitud (2.1) se debe maximizar y posteriormente se deben solucionar las ecuaciones de estimación en ocasiones usando métodos numéricos. Bajo ciertas condiciones de regularidad y si el tamaño de muestra n es suficientemente grande se tiene que la distribución de $\hat{\beta}_{ML}$ es aproximadamente normal con media β y varianza $(\mathbf{X}^t \mathbf{W} \mathbf{X})^{-1}$, donde \mathbf{W} es una matriz diagonal con elementos $w_i = (\partial \mu_i / \partial \eta_i)^2 / \text{Var}(Y_i)$, donde $\eta_i = \mathbf{x}_i^t \beta$.

- El método de máxima verosimilitud (ML) estima los valores de los parámetros de manera de maximizar la probabilidad (o algo equivalente) de observar nuestros datos.
- Es un método paramétrico (debemos conocer la distribución de probabilidad) que estima parámetros para modelos complejos (no se concentra solamente en la media).
- Si los datos son normales, los estimadores ML de la media son idénticos a los OLS (y los de la varianza lo son el caso de REML).
- Es posible comparar dos modelos encajados mediante una prueba del cociente de verosimilitud, cuya distribución es aproximadamente chi-cuadrado.
- El problema que pueden tener es que no necesariamente existen, ni son necesariamente únicos, y muchas veces son sesgados.
- Estos problemas pueden ser graves si hay demasiados parámetros (cientos o miles en algunos casos).

2.6. Sobredispersión en modelos lineales generalizados

- ¿Cómo sabemos si un modelo lineal generalizado ajusta razonablemente?

- Un aspecto que debemos observar es si la dispersión que presentan es la que esperaríamos bajo el modelo.
- Por ejemplo, en datos *Poisson*, $Var(Y) = \mu$. Si observamos que $Var(Y) > \mu$, entonces estamos ante lo que llamamos **sobredispersión**.
- Hay varias razones para la presencia de sobredispersión: la distribución que asumimos no es la correcta, hay heterogeneidad debida a alguna causa conocida, modelo especificado incorrectamente (variable importantes omitidas en el modelo), datos atípicos, etc.
- En modelos con distribución normal nunca es un problema, ya que la varianza es un parámetro adicional que “se lleva” toda la heterogeneidad, pero con Poisson, binomial, binomial negativa, no hay un parámetro separado para la dispersión.
- El problema principal es que si hay sobredispersión la varianza verdadera es mucho mayor que la que el modelo dice, y las pruebas de hipótesis encontrarán diferencias que no existen.
- ¿Cómo verificamos sobredispersión o subdispersión?
 - El cociente **Deviance/gl residuales** debería ser cercano a 1. Si es mucho mayor que 1, entonces debemos considerar alternativas.
 - Un modelo con cuasi verosimilitud en el cual ponemos $Var(Y_i) = \phi\mu_i$ o $Var(Y_i) = \phi\mu_i^2$. El multiplicador de la varianza ϕ se estima de manera que el cociente $Deviance/gl = 1$ o el cociente $X^2Pearson/gl = 1$.
 - Un modelo con enlace logarítmico pero distribución binomial negativa.
 - Un modelo lineal generalizado mixto.
 - Un modelo con enlace logarítmico pero distribución Conway-Maxwell-Poisson (esta distribución también permite modelar datos con subdispersión).

2.6.1. Devianza en la Regresión Poisson

En la regresión de Poisson se asume que Y_1, \dots, Y_n son independientes con $Y_i \sim Poisson(\lambda_i)$. Sin entrar en detalles, el logaritmo de la función de verosimilitud para estos datos usando el modelo está dado por

$$\log(\boldsymbol{\beta}; \mathbf{y}) = \sum_i y_i \log \lambda_i - \sum_i \lambda_i - \sum_i \log y_i!$$

Recuerde que en un MLG asumimos que $\lambda_i = \exp(\mathbf{x}_i^t \boldsymbol{\beta})$, es decir, los n parámetros iniciales λ_i se reducen a p coeficientes $\boldsymbol{\beta}$. Una vez obtenemos el MLE de $\boldsymbol{\beta}$ tenemos el estimador $\hat{\lambda}_i = \hat{y}_i = \hat{E}(Y_i) = \exp(\mathbf{x}_i^t \hat{\boldsymbol{\beta}})$.

Ahora, en lo que se conoce como un modelo saturado, todos los λ_i son diferentes y se deben estimar. Después de algún trabajo algebraico se llega a que el MLE de λ_i es igual a $\hat{\lambda}_i = y_i$. Por lo tanto, el logaritmo de la verosimilitud se reduce a

$$\log(\beta_{max}; \mathbf{y}) = \sum_i y_i \log y_i - \sum_i y_i - \sum_i \log y_i!$$

En este modelo, la devianza es entonces igual a

$$D = 2 \sum_i o_i \log(o_i/e_i),$$

donde o_i es el valor observado y_i y e_i es el valor esperado (ajustado) \hat{y}_i .

2.7. Cuasiverosimilitud

Las estimaciones serán las mismas que en el modelo anterior, pero los errores estándar serán mayores, reflejando la sobredispersión (o menores si hubiese subdispersión).

No es posible aplicar AIC, BIC o cualquier método que se base en verosimilitud.

Para los datos de cangrejos, el modelo que seleccionamos tiene:

Residual deviance: 552.79 on 170 degrees of freedom

Por lo tanto, el cociente es 3.37, y debemos considerar sobredispersión.

```
mod7 <- glm(satellites~weight+color, family="quasi"(link="log",variance="mu"), data=crab)
summary(mod7)
```

Call:

```
glm(formula = satellites ~ weight + color, family = quasi(link = "log",
  variance = "mu"), data = crab)
```

Deviance Residuals:

	Min	1Q	Median	3Q	Max
	-2.9785	-1.9159	-0.5471	0.9181	4.8338

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.08855	0.45424	0.195	0.846
weight	0.54588	0.12049	4.531	1.1e-05 ***
color	-0.17282	0.10988	-1.573	0.118

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasi family taken to be 3.18719)

Null deviance: 632.79 on 172 degrees of freedom
Residual deviance: 552.79 on 170 degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 6

```
mod8 <- glm(satellites~weight+color, family="quasi"(link="log",variance="mu^2"), data=
summary(mod8)
```

Call:

```
glm(formula = satellites ~ weight + color, family = quasi(link = "log",
variance = "mu^2"), data = crab)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-1.2363	-0.1958	0.0000	0.5073	2.6166

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-0.4710	0.5586	-0.843	0.400
weight	0.7620	0.1649	4.621	7.51e-06 ***
color	-0.1693	0.1186	-1.427	0.155

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasi family taken to be 1.459119)

Null deviance: 80.907 on 172 degrees of freedom
Residual deviance: 91.583 on 170 degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 19

2.7.1. Regresión Binomial Negativa

La función de densidad de la distribución binomial negativa está dada por

$$f(y_i; \mu_i, \alpha) = \frac{\Gamma(y + \alpha)}{\Gamma(\alpha)\Gamma(y + 1)} \left(\frac{\mu_i}{\mu_i + \alpha} \right)^y \left(\frac{\alpha}{\mu_i + \alpha} \right)^\alpha, \quad y = 0, 1, \dots$$

Esta distribución resulta de la distribución marginal de Y_i si se asume que $Y_i|\lambda_i \sim \text{Poisson}(\lambda_i)$ y $\lambda_i \sim \text{Gama}(\alpha, \alpha/\mu_i)$ (parametrización tradicional con $E(\lambda_i) = \mu_i$ y $\text{Var}(\lambda_i) = \mu_i^2/\alpha$).

En la distribución BN se tiene entonces que

$$E(Y_i) = \mu_i, \quad \text{Var}(Y_i) = \mu_i + \gamma\mu_i^2,$$

donde $\gamma = 1/\alpha$ se conoce como el parámetro de dispersión. Note que para $\gamma > 0$ se tiene que $\text{Var}(Y_i) > \mu_i$, lo cual justifica el uso de la distribución BN para sobredispersión. Además, cuando γ tiende a 0 entonces la BN tiende a una Poisson. Es decir, si γ no es significativamente diferente de 0 entonces este resultado sugiere que el modelo Poisson es mejor para el conjunto de datos. De la misma manera, se pueden llevar a cabo una prueba LRT para comparar el modelo Poisson y el BN, pero el valor p se debe ajustar debido a las características de los modelos.

```
library(MASS)

mod9 <- glm.nb(satellites~weight+color, data=crab)
summary(mod9)
```

Call:

```
glm.nb(formula = satellites ~ weight + color, data = crab, init.theta = 0.9555407028,
       link = log)
```

Deviance Residuals:

	Min	1Q	Median	3Q	Max
	-1.8635	-1.3849	-0.3215	0.4446	2.1361

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-0.3220	0.5540	-0.581	0.561
weight	0.7072	0.1612	4.387	1.15e-05 ***
color	-0.1734	0.1199	-1.445	0.148

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for Negative Binomial(0.9555) family taken to be 1)

Null deviance: 219.50 on 172 degrees of freedom
 Residual deviance: 196.64 on 170 degrees of freedom
 AIC: 754.45

Number of Fisher Scoring iterations: 1

Theta: 0.956
Std. Err.: 0.174

2 x log-likelihood: -746.452

```
Devchi2<-deviance(mod9)  
Devchi2 / 170
```

[1] 1.156701

```
Pearsonchi2 <- sum(residuals(mod9,type="pearson")^2)  
Pearsonchi2 / 170
```

[1] 0.924807

```
mod10<-update(mod9, .~. -color)  
summary(mod10)
```

Call:

```
glm.nb(formula = satellites ~ weight, data = crab, init.theta = 0.9310592351,  
link = log)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-1.8394	-1.4122	-0.3247	0.4744	2.1279

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-0.8647	0.4048	-2.136	0.0327 *
weight	0.7603	0.1578	4.817	1.45e-06 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for Negative Binomial(0.9311) family taken to be 1)

Null deviance: 216.43 on 172 degrees of freedom
Residual deviance: 196.16 on 171 degrees of freedom
AIC: 754.64

Number of Fisher Scoring iterations: 1

Theta: 0.931
Std. Err.: 0.168

2 x log-likelihood: -748.644

```
predict(mod10,data.frame(weight=3),se.fit=TRUE,  
        type="link" )
```

\$fit

1
1.416179

\$se.fit

[1] 0.1214045

\$residual.scale

[1] 1

```
predict(mod10,data.frame(weight=3),se.fit=TRUE,  
        type="response" )
```

\$fit

1
4.121344

\$se.fit

1
0.5003498

\$residual.scale

[1] 1

```
predict(mod10,data.frame(weight=2),se.fit=TRUE,  
        type="link" )
```

\$fit

1
0.6559005

\$se.fit

[1] 0.1209425

\$residual.scale

[1] 1

```
predict(mod10,data.frame(weight=2),se.fit=TRUE,  
        type="response" )
```

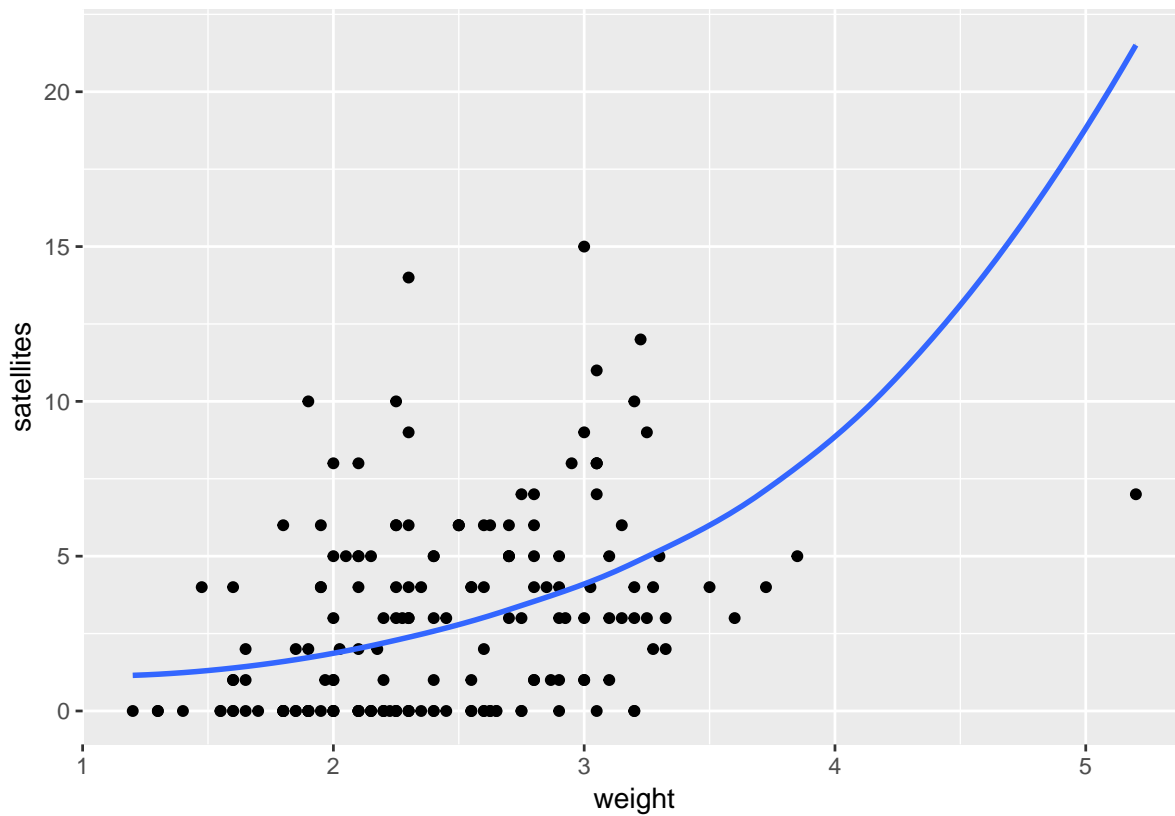
```
$fit
      1
1.926877
```

```
$se.fit
      1
0.2330412
```

```
$residual.scale
[1] 1
```

```
xmin<-min(crab$weight)
xmax<-max(crab$weight)
predicted10 <- data.frame(weight=seq(xmin,xmax,length.out=100))
predicted10$satellites<-predict(mod10,predicted10, type="response")

p3 <- ggplot(crab, aes(x=weight, y=satellites))+geom_point()
p3 + geom_smooth(data=predicted10)
```



2.7.2. Interpretación

- Coeficiente asociado con peso.
- Predicción del número promedio de satélites para una hembra que pese 3. ¿Intervalo de confianza para la predicción de la media?
- Predicción del número promedio de satélites para una hembra que pese 2. ¿Intervalo de confianza para la predicción de la media?
- ¿Cociente entre las predicciones en partes b y c?

2.8. Distribución Comway-Maxwell-Poisson

La distribución Comway-Maxwell-Poisson tiene función de probabilidad dada por

$$f(y; \lambda, \nu) = \frac{\lambda^y}{(y!)^\nu Z(\lambda, \nu)}, \quad y = 0, 1, 2, \dots$$

donde

$$Z(\lambda, \nu) = \sum_{j=0}^{\infty} \frac{\lambda^j}{(j!)^\nu}.$$

Esta suma se aproxima con truncándola en un valor específico.

Esta distribución permite modelar sub- y sobre-dispersión, ya que su parámetro de dispersión ν puede tomar valores mayores de 1 (subdispersión), menores de 1 (sobredispersión) e iguales a 1 (Poisson).

Bajo esta parametrización,

$$E(Y) \approx \lambda^\nu + \frac{1}{2\nu} - \frac{1}{2},$$
$$Var(Y) \approx \frac{1}{\nu} \lambda^{1/\nu}.$$

- Es posible modelar tanto el parámetro λ como el parámetro ν con un modelo loglineal.
- Existen otras parametrizaciones de esta distribución (por ejemplo, en PROC COUNTREG en SAS) en términos de media y dispersión.

2.8.1. COMPOisson en R

- Los paquetes `COMPOissonReg` y `CompGLM` permiten ajustar modelos loglineales bajo la parametrización presentada (`CompGLM` tiene algún problema de organización de la salida, por lo que no la usaremos por ahora).
- Ambas ajustan un modelo loglineal para λ (se le puede agregar un *offset*) y un modelo loglineal para μ .
- Observar que no es estrictamente un modelo lineal generalizado, ya que no se modela un predictor lineal para la media en la escala logarítmica.

Ejemplo 2.3 (Variaciones en la riqueza de los ensambles de roedores y su relación con variables ambientales). La lechuza de los campanarios (*Tyto alba*) es una rapaz que se alimenta principalmente de roedores y consume diferentes especies de manera oportunista (en función de su disponibilidad). Por lo tanto, se considera que las especies de roedores consumidas por las lechuzas son un reflejo de los ensambles de roedores que habitan en el área donde la lechuza se alimentó. La composición de la dieta de las lechuzas se obtiene a partir del análisis del contenido de las egagrópilas, bolas formadas por restos de alimentos no digeridos (huesos, pelos, etc.) que regurgitan las aves.

Para analizar cuáles son los procesos que estructuran a las comunidades de roedores, un grupo de investigadores determinó la riqueza de roedores (número de especies), estimada a partir de egagrópilas de *T. alba* obtenidas en 111 sitios ubicados en distintas provincias del centro y norte de Argentina y pertenecientes a 5 ecorregiones.

Datos cedidos por Gerardo Cueto, disponibles en “Riqueza Roedores y Ambiente.csv”

Utilizando a cada sitio de colecta de egagrópilas como centro, los investigadores trazaron un círculo de 2,5 km de radio que representó el área de acción de la lechuza. Registraron para cada círculo 4 variables climáticas:

- Temperatura media anual (C): promedio de los 12 registros de temperatura media mensual.
- Variación en la temperatura (C): desvío estándar de los 12 registros temperatura media mensual
- Precipitación anual (mm acumulados por año).
- CV de la precipitación anual.

Dos variables topográficas:

- Altura media (metros): promedio de las alturas de los pixeles del círculo.
- Variación en la altura (metros): desvío estándar de las alturas de los pixeles del círculo.



Figura 2.1: Datos de Lechuza

```
head(Riqueza.Roedores.y.Ambiente)
```

	ID.Lat		Sitio	Provincia	Ecorregion	Longitud
1	1	Muestra 5	San Hilario	Formosa	Chaco Humedo	-58.65182
2	2	Muestra 4	Mariano Boedo	Formosa	Chaco Humedo	-58.48617
3	3	Muestra 3	Textil	Formosa	Chaco Humedo	-58.17420
4	4	Muestra 2	Tanque Ejercito	Formosa	Chaco Humedo	-58.16946
5	5	Muestra	Tanque 1	Formosa	Chaco Humedo	-58.16491
6	6	Muestra 6	Algodonera	Formosa	Chaco Humedo	-58.15437

	Latitud	Riqueza	Altura	Altura.STD	Temp.media.anual	SD_temp
1	-26.02988	8	62.77	4.48	21.920	3.906
2	-26.11137	7	57.59	4.30	22.029	3.847
3	-26.16594	8	50.53	4.65	22.400	3.728
4	-26.16622	8	51.00	5.24	22.400	3.727
5	-26.17856	8	51.25	5.69	22.400	3.726
6	-26.19409	8	50.83	5.84	22.400	3.749

	Precip.anual	CV_pp
1	1212.5	39.0
2	1290.0	37.6
3	1390.0	35.0
4	1390.0	35.0
5	1411.2	34.9
6	1417.4	34.5


```

library(CompGLM)
library(COMPoissonReg)
library(ggplot2)

Altura <- ggplot(Riqueza.Roedores.y.Ambiente, aes(Altura, log(Riqueza))) +geom_point(
AlturaSTD <- ggplot(Riqueza.Roedores.y.Ambiente, aes(Altura.STD, log(Riqueza))) +geom_
Temp <- ggplot(Riqueza.Roedores.y.Ambiente, aes(Temp.media.anual, log(Riqueza))) +geom_
TempSTD <- ggplot(Riqueza.Roedores.y.Ambiente, aes(SD_temp, log(Riqueza))) +geom_poin
Precip <- ggplot(Riqueza.Roedores.y.Ambiente, aes(Precip.anual, log(Riqueza))) +geom_l
PrecipCV <- ggplot(Riqueza.Roedores.y.Ambiente, aes(CV_pp, log(Riqueza))) +geom_point

# Multiple plot function
#
# ggplot objects can be passed in ..., or to plotlist (as a list of ggplot objects)
# - cols: Number of columns in layout
# - layout: A matrix specifying the layout. If present, 'cols' is ignored.
#
# If the layout is something like matrix(c(1,2,3,3), nrow=2, byrow=TRUE),
# then plot 1 will go in the upper left, 2 will go in the upper right, and
# 3 will go all the way across the bottom.
#
multiplot <- function(..., plotlist=NULL, file, cols, layout=NULL) {
  library(grid)

  # Make a list from the ... arguments and plotlist
  plots <- c(list(...), plotlist)

  numPlots = length(plots)

  # If layout is NULL, then use 'cols' to determine layout
  if (is.null(layout)) {
    # Make the panel
    # ncol: Number of columns of plots
    # nrow: Number of rows needed, calculated from # of cols
    layout <- matrix(seq(1, cols * ceiling(numPlots/cols)),
                     ncol = cols, nrow = ceiling(numPlots/cols))
  }

  if (numPlots==1) {
    print(plots[[1]])

  } else {

```

```

# Set up the page
grid.newpage()
pushViewport(viewport(layout = grid.layout(nrow(layout), ncol(layout))))

# Make each plot, in the correct location
for (i in 1:numPlots) {
  # Get the i,j matrix positions of the regions that contain this subplot
  matchidx <- as.data.frame(which(layout == i, arr.ind = TRUE))

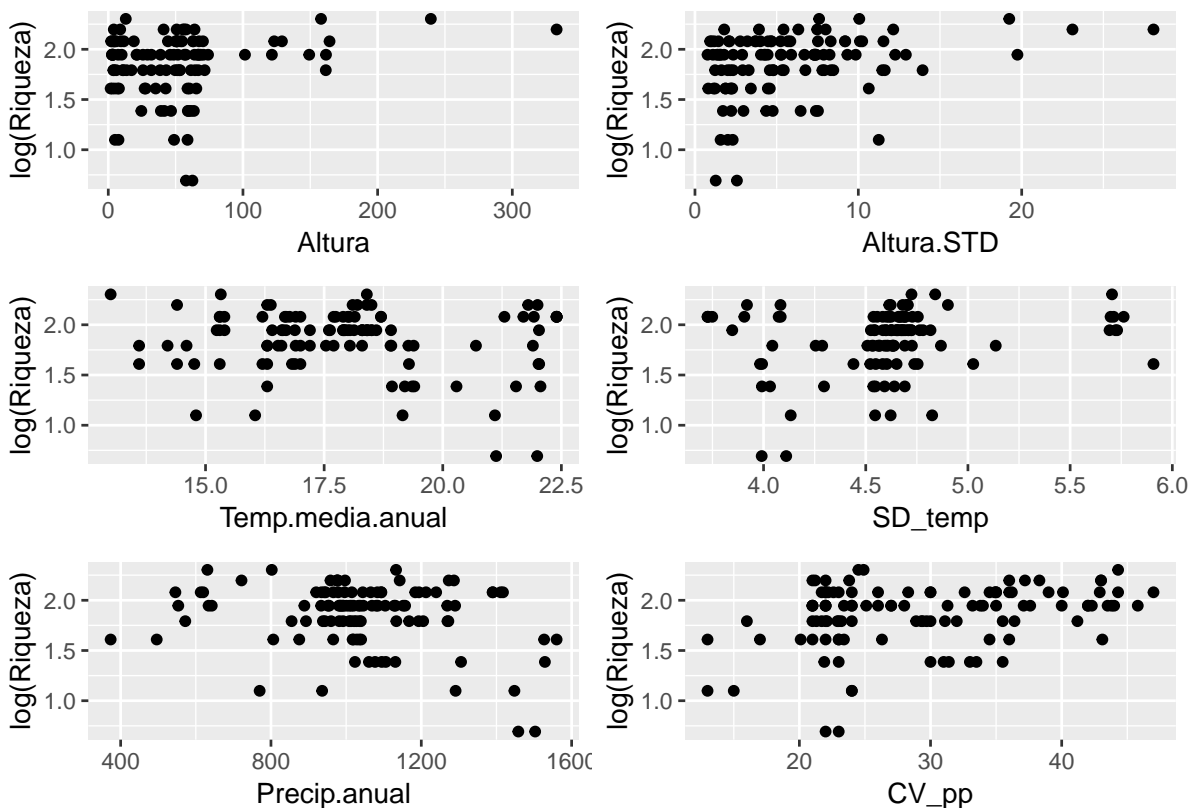
  print(plots[[i]], vp = viewport(layout.pos.row = matchidx$row,
                                   layout.pos.col = matchidx$col))
}
}
}

```

```

multiplot(Altura, Temp, Precip, AlturaSTD, TempSTD, PrecipCV, cols=2)

```



```

fit1 <- glm(Riqueza~CV_pp,family=poisson, data=Riqueza.Roedores.y.Ambiente)
summary(fit1)

```

```
Call:
glm(formula = Riqueza ~ CV_pp, family = poisson, data = Riqueza.Roedores.y.Ambiente)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-1.9732	-0.3344	0.0171	0.3870	1.3558

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	1.610919	0.137917	11.680	<2e-16 ***
CV_pp	0.009386	0.004437	2.115	0.0344 *

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 53.419 on 110 degrees of freedom
Residual deviance: 48.979 on 109 degrees of freedom
AIC: 465.1

Number of Fisher Scoring iterations: 4

```
fit2 <- glm(Riqueza~CV_pp,family=quasipoisson, data=Riqueza.Roedores.y.Ambiente)
summary(fit2)
```

Call:

```
glm(formula = Riqueza ~ CV_pp, family = quasipoisson, data = Riqueza.Roedores.y.Ambien
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-1.9732	-0.3344	0.0171	0.3870	1.3558

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	1.610919	0.089433	18.013	< 2e-16 ***
CV_pp	0.009386	0.002878	3.262	0.00148 **

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasipoisson family taken to be 0.4204923)

Null deviance: 53.419 on 110 degrees of freedom

Residual deviance: 48.979 on 109 degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 4

```
fit3 <- glm.cmp(Riqueza~CV_pp, data=Riqueza.Roedores.y.Ambiente, max=11)
fit3
```

Fit for CMP coefficients

	Estimate	SE	z.value	p.value
X:(Intercept)	3.8301	0.6289	6.0900	1.129e-09
X:CV_pp	0.0220	0.0074	2.9770	0.002911
S:(Intercept)	0.8387	0.1490	5.6277	1.827e-08

--

Transformed intercept-only parameters

	Estimate	SE
nu	2.3134	0.3448

--

Chi-squared test for equidispersion

X² = 32.3407, df = 1, p-value = 1.2937e-08

--

Elapsed Sec: 0.10 Sample size: 111

LogLik: -214.3817 AIC: 434.7634 BIC: 442.8920

```
summary(fit3)
```

\$DF

	Estimate	SE	z.value	p.value
X:(Intercept)	3.8301	0.6289	6.0900	1.129e-09
X:CV_pp	0.0220	0.0074	2.9770	0.002911
S:(Intercept)	0.8387	0.1490	5.6277	1.827e-08

\$DF.lambda

NULL

\$DF.nu

	Estimate	SE
nu	2.3134	0.3448

\$n

[1] 111

\$loglik

[1] -214.3817

```
$aic  
[1] 434.7634
```

```
$bic  
[1] 442.892
```

```
$opt.message  
NULL
```

```
$opt.convergence  
NULL
```

```
$elapsed.sec  
[1] 0.09848213
```

```
dev <- deviance(fit3)  
res <- residuals(fit3)  
pred <- predict(fit3)
```

2.9. Modelos Lineales Generalizados Mixtos

$$\begin{aligned}y_{ij}|\mathbf{u}_i & \text{ i.d. } f(\boldsymbol{\beta}, \boldsymbol{\theta}), \\ \mathbf{u}_i & \sim N(\mathbf{0}, \mathbf{G}). \\ g(E(y_{ij}|\mathbf{u}_i)) & = \mathbf{x}_{ij}\boldsymbol{\beta} + \mathbf{z}_{ij}\mathbf{u}_i.\end{aligned}$$

f , la distribución condicional de y , es una familia exponencial con dispersión y g es la función de enlace (monótona y con segunda derivada).

2.9.1. Métodos de estimación para MLGM

- Métodos basados en **aproximaciones lineales**: realizan aproximaciones de los datos, llamados pseudo-datos, usando expansión en serie de Taylor alrededor de una estimación razonable de \mathbf{u}_i . No dan verosimilitud (no AIC, BIC, etc.)
- Métodos basados en **aproximaciones numéricas de las integrales**, p.ej., cuadratura de Gauss-Hermite, Laplace. Estos métodos se suelen llamar “exactos” ya que consideran la verosimilitud completa de los datos originales a pesar de su aproximación numérica. Obtienen una verosimilitud.

- **Métodos Bayesianos:** aproximan las integrales usando integración vía Monte Carlo y consideran otro nivel adicional del modelo donde se especifican las distribuciones a priori de los parámetros. Se deben interpretar con métodos Bayesianos.

Ejemplo 2.4 (Teratología (p. 275; Agresti, 2015)). Se compararon 4 tratamientos en ratas con deficiencia de hierro para ver si había diferencias en la proporción de fetos muertos.

Los cuatro tratamientos fueron inyecciones con hierro (tratamiento 1, placebo; tratamiento 2, a los días 7 y 10; tratamiento 3, a los días 0 y 7; tratamiento 4, semanalmente).

Las 58 ratas preñadas se sacrificaron a las 3 semanas, y se contó el número de fetos totales y muertos en cada caso. Se midió además el nivel de hemoglobina de la madre.

Primero analizaremos de la manera más simple posible: un MLG para la cantidad de fetos muertos suponiendo distribución binomial y enlace logit.

$$y_{ij} \stackrel{i.d.}{\sim} \text{Binomial}(n_{ij}, \beta)$$

$$\text{logit} \left\{ E \left(\frac{y_{ij}}{n_{ij}} \right) \right\} = \beta_0 + \tau_i + \beta_1 h_{ij} + u_{ij}.$$

```
head(Rats)
```

```
# A tibble: 6 x 5
  litter group    h     n     s
  <int> <int> <dbl> <int> <int>
1     1     1  4.10    10     1
2     2     1  3.20    11     4
3     3     1  4.70    12     9
4     4     1  3.50     4     4
5     5     1  3.20    10    10
6     6     1  5.90    11     9
```

```
fit.ML <- glm(s/n ~ as.factor(group) + h, weights=n, data=Rats,
              family=binomial)
summary(fit.ML)
```

Call:

```
glm(formula = s/n ~ as.factor(group) + h, family = binomial,
     data = Rats, weights = n)
```

Deviance Residuals:

```
Min      1Q   Median      3Q      Max
```

-4.6331 -0.9684 -0.0471 1.3945 2.8256

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)	
(Intercept)	2.1795	0.5056	4.311	1.63e-05	***
as.factor(group)2	-2.4748	0.5045	-4.906	9.30e-07	***
as.factor(group)3	-3.1527	0.9433	-3.342	0.000832	***
as.factor(group)4	-2.0520	1.0629	-1.931	0.053525	.
h	-0.2190	0.1020	-2.148	0.031728	*

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 509.43 on 57 degrees of freedom
Residual deviance: 168.91 on 53 degrees of freedom
AIC: 250.38

Number of Fisher Scoring iterations: 5

Supuestos para que el modelo binomial sea válido: ¿Cómo medimos en un MLG si el ajuste es correcto?

La manera más directa de eliminar la sobredispersión es mediante un factor multiplicador de la varianza. En este caso no existe una verosimilitud, y ya no ajustamos un MLG sino que solo definimos media y varianza (cuasiverosimilitud).

$$y_{ij} \quad i.d.$$
$$\text{logit} \left\{ E \left(\frac{y_{ij}}{n_{ij}} \right) \right\} = \beta_0 + \tau_i + \beta_1 h_{ij} + u_{ij}.$$
$$\text{Var} \left(\frac{y_{ij}}{n_{ij}} \right) = \phi \frac{\mu_{ij}(1 - \mu_{ij})}{n_{ij}}.$$

```
fit.QL <- glm(s/n ~ as.factor(group) + h, weights=n, data=Rats,  
             family=quasi(link="logit", variance="mu(1-mu)" ))  
summary(fit.QL)
```

Call:

```
glm(formula = s/n ~ as.factor(group) + h, family = quasi(link = "logit",  
  variance = "mu(1-mu)", data = Rats, weights = n)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-4.6331	-0.9684	-0.0471	1.3945	2.8256

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	2.1795	0.8669	2.514	0.01501 *
as.factor(group)2	-2.4748	0.8650	-2.861	0.00603 **
as.factor(group)3	-3.1527	1.6175	-1.949	0.05658 .
as.factor(group)4	-2.0520	1.8224	-1.126	0.26524
h	-0.2190	0.1749	-1.253	0.21583

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasi family taken to be 2.939931)

Null deviance: 509.43 on 57 degrees of freedom
 Residual deviance: 168.91 on 53 degrees of freedom
 AIC: NA

Number of Fisher Scoring iterations: 6

```
library(lme4)
Rats$lit<-as.factor(Rats$litter)
fit.GLMMLapl <- glmer(s/n ~ as.factor(group) + h +(1|lit), weights=n, data=Rats,
                     family=binomial, AGQ=1)
summary(fit.GLMMLapl)
```

Generalized linear mixed model fit by maximum likelihood (Laplace
 Approximation) [glmerMod]
 Family: binomial (logit)
 Formula: s/n ~ as.factor(group) + h + (1 | lit)
 Data: Rats
 Weights: n

AIC	BIC	logLik	deviance	df.resid
195.6	208.0	-91.8	183.6	52

Scaled residuals:

Min	1Q	Median	3Q	Max
-1.01937	-0.36233	-0.05219	0.61911	0.96894

Random effects:

Groups	Name	Variance	Std.Dev.
--------	------	----------	----------


```
lit      (Intercept) 2.242    1.497
Number of obs: 58, groups:  lit, 58
```

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	2.6362	1.1939	2.208	0.02724 *
as.factor(group)2	-3.7812	1.2339	-3.064	0.00218 **
as.factor(group)3	-4.7752	1.8732	-2.549	0.01080 *
as.factor(group)4	-3.8256	2.5361	-1.508	0.13143
h	-0.1782	0.2419	-0.737	0.46128

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:

	(Intr)	as.()2	as.()3	as.()4
as.fctr(g)2	0.670			
as.fctr(g)3	0.673	0.715		
as.fctr(g)4	0.838	0.820	0.777	
h	-0.953	-0.807	-0.780	-0.935

`deviance`(fit.GLMMLapl)

[1] 24.34715

Otra manera de considerar la sobredispersión es mediante un efecto aleatorio de camada. Ahora tenemos un MLGM:

$$y_{ij}|u_{ij} \stackrel{i.d.}{\sim} \text{Binomial}(n_{ij}, \beta)$$

$$u_{ij} \sim N(0, \sigma_u^2)$$

$$\text{logit} \left\{ E \left(\frac{y_{ij}}{n_{ij}} | u_{ij} \right) \right\} = \beta_0 + \tau_i + \beta_1 h_{ij} + u_{ij}.$$

```
fit.GLMM2 <- glmer(s/n ~ as.factor(group) +(1|lit), weights=n, data=Rats, family=binor
summary(fit.GLMM2)
```

Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) [glmerMod]
 Family: binomial (logit)
 Formula: s/n ~ as.factor(group) + (1 | lit)
 Data: Rats
 Weights: n

AIC	BIC	logLik	deviance	df.resid
-----	-----	--------	----------	----------

```

      194.2      204.5      -92.1      184.2      53

Scaled residuals:
      Min       1Q   Median       3Q      Max
-0.98723 -0.35727 -0.08385  0.64553  0.82287

Random effects:
 Groups Name      Variance Std.Dev.
 lit      (Intercept) 2.284    1.511
Number of obs: 58, groups:  lit, 58

Fixed effects:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)      1.8094    0.3616   5.004 5.62e-07 ***
as.factor(group)2 -4.5396    0.7345  -6.180 6.40e-10 ***
as.factor(group)3 -5.8833    1.1754  -5.005 5.58e-07 ***
as.factor(group)4 -5.6063    0.9077  -6.177 6.55e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
              (Intr) as.()2 as.()3
as.fctr(g)2  -0.562
as.fctr(g)3  -0.373  0.235
as.fctr(g)4  -0.496  0.316  0.221

```

```
deviance(fit.GLMM2)
```

```
[1] 24.03924
```

- Si una camada tiene efecto aleatorio 0, ¿cuál es la proporción esperada de fetos muertos en cada uno de los cuatro grupos?
- Interpretar los efectos estimados de cada camada (estimadores Bayesianos empíricos) y los valores predichos de proporción de fetos muertos en cada rata.

```
library(sjPlot)
ranef(fit.GLMM2)
```

```
$lit
  (Intercept)
1 -3.00801372
2 -2.03217685
3 -0.59201701
4  0.69200802

```

5 1.13827670
6 -0.23979728
7 1.08132636
8 1.19092875
9 1.13827670
10 -0.79109992
11 1.23990639
12 0.26503262
13 1.01927637
14 -0.23979728
15 -0.82906707
16 -0.43093547
17 1.32872559
18 -1.27740993
19 -0.23979728
20 -1.16931968
21 -2.11824095
22 1.13827670
23 -0.15789582
24 -1.16931968
25 1.13827670
26 -2.66756364
27 1.28570520
28 -0.43673215
29 1.01927637
30 -2.00181181
31 1.23990639
32 0.34984411
33 1.13153819
34 0.16455565
35 -0.78864820
36 1.56050377
37 1.12623881
38 0.80120296
39 0.01510852
40 -0.75032588
41 -0.38722776
42 -0.12433117
43 -0.78864820
44 -0.24098884
45 1.08250194
46 -0.37093559
47 0.94415992

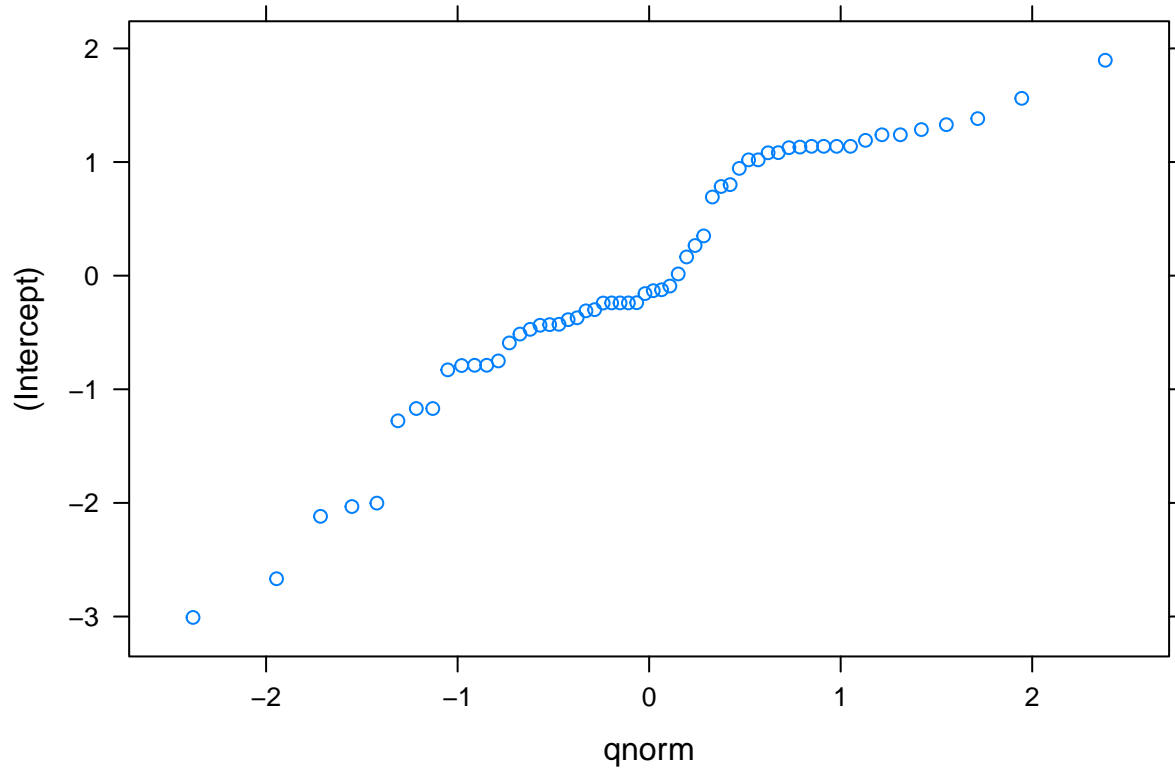
```

48 -0.30964438
49 -0.13211584
50 -0.42798808
51  1.89573298
52  1.38200091
53 -0.47258843
54 -0.09164216
55  0.78474804
56 -0.29903446
57 -0.23811786
58 -0.51410676

```

```
plot(ranef(fit.GLMM2))
```

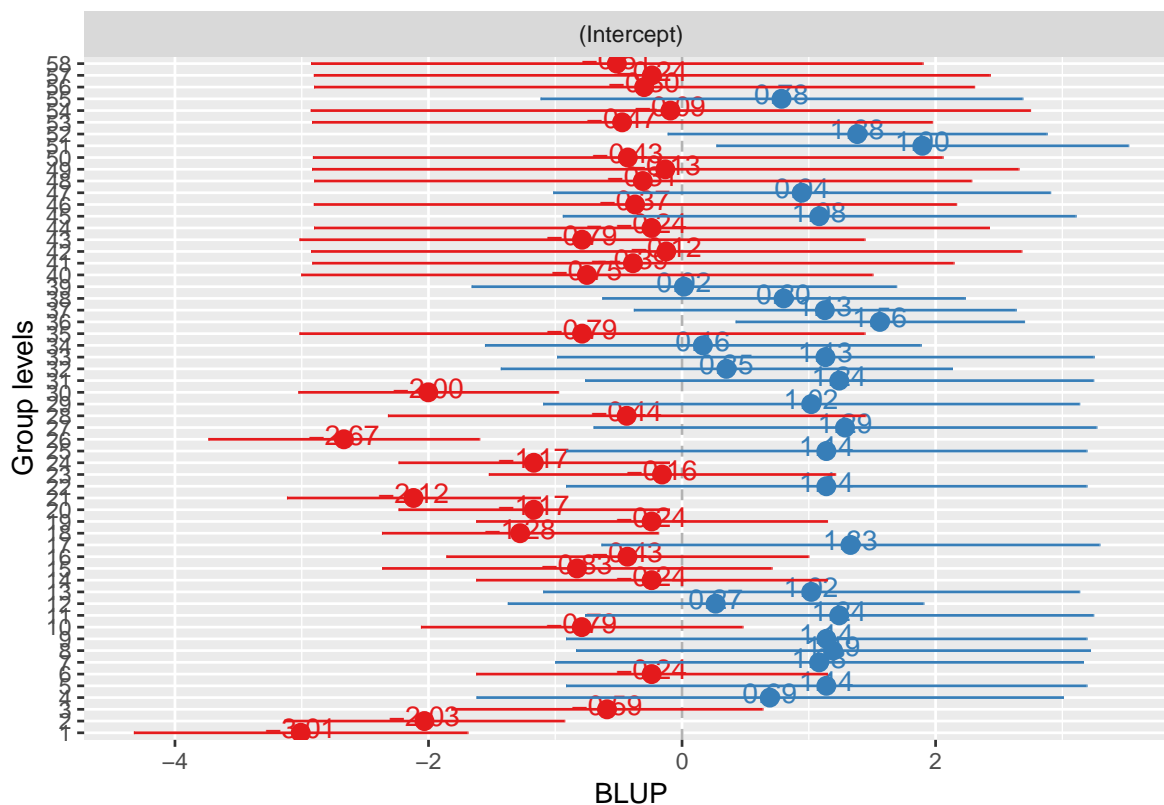
\$lit



```

sjp.lmer(fit.GLMM2,
  sort.est = "sort.all",
  y.offset = .4)

```



```
Rats$proppred<- predict(fit.GLMM2,Rats,type="response")
Rats$proppred
```

```
[1] 0.23172334 0.44453696 0.77160408 0.92424103 0.95015395 0.82772811
[7] 0.94738650 0.95258934 0.95015395 0.73464290 0.95475292 0.88839401
[13] 0.94420633 0.82772811 0.72717586 0.79874557 0.95843860 0.62994901
[19] 0.82772811 0.65477342 0.42339963 0.95015395 0.83909532 0.65477342
[25] 0.95015395 0.29772483 0.95669072 0.79781213 0.94420633 0.45204689
[31] 0.95475292 0.08468004 0.16816344 0.07137998 0.02877962 0.23690306
[37] 0.16742344 0.12685744 0.06208649 0.02987034 0.04239254 0.05444562
[43] 0.02877962 0.01319137 0.04781488 0.01160255 0.04189606 0.01232689
[49] 0.01928486 0.01441688 0.12998258 0.08204769 0.01379669 0.02006542
[55] 0.04688232 0.01636870 0.01737895 0.01324303
```

2.10. Modelo LogLineal Poisson-Normal (o Poisson-Lognormal)

$$y|\mathbf{u} \sim \text{Poisson}(\boldsymbol{\beta})$$

$$\mathbf{u} \sim N(\mathbf{0}, \mathbf{G})$$

$$\log(E(y|\mathbf{u})) = \mathbf{x}^t \boldsymbol{\beta} + \mathbf{z}^t \mathbf{u}.$$

Media marginal:

$$E(y) = E_{\mathbf{u}}[\exp(\mathbf{x}^t \boldsymbol{\beta} + \mathbf{z}^t \mathbf{u})] \quad (2.2)$$

$$= \exp(\mathbf{x}^t \boldsymbol{\beta}) E_{\mathbf{u}}(\exp(\mathbf{z}^t \mathbf{u})) \quad (2.3)$$

$$= \exp(\mathbf{x}^t \boldsymbol{\beta}) \exp\left(\frac{1}{2} \mathbf{z}^t \mathbf{u} \mathbf{G} \mathbf{u}\right) \quad (2.4)$$

Por ejemplo, si solo tenemos un efecto aleatorio en el intercepto,

$$\mathbf{z} = (1, 0, 0, \dots, 0), \quad \mathbf{G} = \sigma_u^2 \mathbf{I}$$

$$E(y) = \exp\left(\mathbf{x}^t \boldsymbol{\beta} + \frac{1}{2} \sigma_u^2\right).$$

Varianza marginal: La distribución marginal siempre tiene sobredispersión.

$$Var(y) = E_{\mathbf{u}}[Var(y|\mathbf{u})] + Var_{\mathbf{u}}[E(y|\mathbf{u})] \quad (2.5)$$

$$= E(y)[\exp(\mathbf{x}^t \boldsymbol{\beta}) \{\exp(3\sigma^2/2) - \exp(\sigma^2/2)\} + 1] \quad (2.6)$$

La distribución es una mezcla de distribuciones que no tiene una forma cerrada (involucra una serie infinita):

$$f(y) = \int_{\mathbf{u}} f(y|\mathbf{u}) f(\mathbf{u}) d\mathbf{u}.$$

2.11. Modelo Negativo Binomial-Normal

A veces la sobredispersión inducida por el modelo Poisson-normal no es suficiente para explicar la variabilidad observada.

Un modelo alternativo es

$$y|\mathbf{u} \sim NB(\boldsymbol{\beta}, \alpha)$$

$$\mathbf{u} \sim N(\mathbf{0}, \mathbf{G})$$

$$\log(E(y|\mathbf{u})) = \mathbf{x}^t \boldsymbol{\beta} + \mathbf{z}^t \mathbf{u}.$$

Ejemplo 2.5 (Ensayo clínico de progabide). Para probar la efectividad de una droga anti-epiléptica (progabide) se realizó un ensayo clínico con 59 pacientes. Los datos están analizados en Thall y Vail (1990), Breslow y Clayton (1993) y Diggle et al. (2002). Para cada paciente se registró el número de convulsiones durante un periodo de 8 semanas (línea de base). Luego se aleatorizaron en dos grupos: tratado (progabide) y placebo, además de darles a todos la quimioterapia tradicional. Se registró el número de convulsiones de cada paciente en cuatro periodos consecutivos de dos semanas.

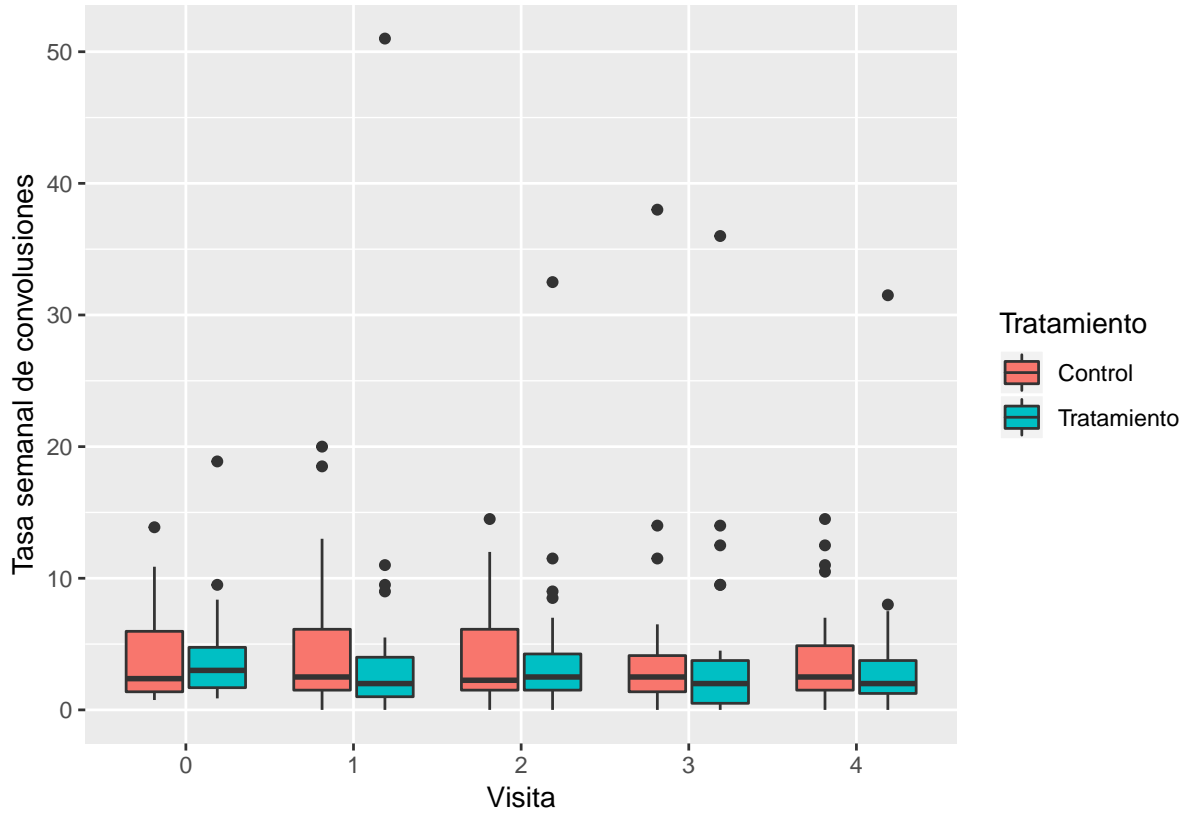
La pregunta de investigación es si la droga reduce la tasa de convulsiones epilépticas (cantidad de convulsiones/semana).

```
head(epilepsia)
```

	id	convulsiones	base	trat	Despues.Aleat	visita	log.tiempo	tasa_sem
1	101	76	76	1	0	0	2.0794415	9.50
2	101	11	76	1	1	1	0.6931472	5.50
3	101	14	76	1	1	2	0.6931472	7.00
4	101	9	76	1	1	3	0.6931472	4.50
5	101	8	76	1	1	4	0.6931472	4.00
6	102	38	38	1	0	0	2.0794415	4.75

```
sqrtasa
1 3.082207
2 2.345208
3 2.645751
4 2.121320
5 2.000000
6 2.179449
```

```
library(ggplot2)
ggplot(epilepsia, aes(y=tasa_sem, x=factor(visita), fill=factor(trat))) + geom_boxplot()
  scale_fill_discrete(name="Tratamiento",
                      breaks=c("0", "1"),
                      labels=c("Control", "Tratamiento"))
```



Se puede ver que en los pacientes tratados hay una tasa levemente menor que en los pacientes con placebo, especialmente en las semanas 1 y 4.

Inicialmente plantearemos un modelo sin tener en cuenta la naturaleza longitudinal de esta información (es decir, suponiendo que todas las observaciones son independientes).

El más simple es un modelo loglineal que supone que los recuentos Y_{ij} del paciente i en la visita j siguen una distribución Poisson cuya media depende del tiempo de exposición periodo (*offset*, $\log(t_{ij})$), del tratamiento (placebo/tratado), de la visita (pre-aleatorización o post-aleatorización) y de la interacción *tratamiento * visita*:

$$Y_{ij} \sim \text{Poisson}(\mu_{ij})$$

$$\log(\mu_{ij}) = \log(t_{ij}) + \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2j} + \beta_3 x_{1i} x_{2j}.$$

Debemos observar que para modelar tasas debemos usar $\log(t_{ij})$ como “offset”, para que los parámetros de la componente lineal puedan interpretarse en términos de tasas esperadas y no de recuentos esperados:

$$tasa_{ij} = \frac{\mu_{ij}}{t_{ij}}, \quad \log(tasa_{ij}) = \log(\mu_{ij}) - \log(t_{ij})$$

$$Y_{ij} \sim \text{Poisson}(\mu_{ij})$$

$$\begin{aligned}\log(\mu_{ij}/t_{ij}) &= \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2j} + \beta_3 x_{1i} x_{2j}. \\ \log(\mu_{ij}/t_{ij}) &= \log(t_{ij}) + \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2j} + \beta_3 x_{1i} x_{2j}.\end{aligned}$$

$$x_{1i} = \begin{cases} 0, & \text{Placebo} \\ 1, & \text{Tratado.} \end{cases} \quad x_{2j} = \begin{cases} 0, & \text{visita } j = 0 \\ 1, & \text{visita } j > 0. \end{cases}$$

- β_0 es el logaritmo de la tasa esperada de convulsiones por semana para un paciente del grupo placebo antes de la aleatorización (i.e., asignación de tratamientos).
- β_1 es el logaritmo del cociente entre la tasa esperada de convulsiones por semana para un paciente tratado y la tasa esperada de convulsiones por semana para un paciente en el placebo antes de la aleatorización (i.e., asignación de tratamientos).
- Observar que x_{2j} y $x_{1i}x_{2j}$ antes de la aleatorización toman el valor cero, y por lo tanto, el modelo para $\log(tasa_{ij})$ antes de la aleatorización será:

$$\log(tasa_{i0}) = \beta_0 + \beta_1 x_{1i}.$$

Luego, un para un paciente tratado, el logaritmo de la tasa esperada antes de la aleatorización es $\log(tasa_{tratado,0}) = \beta_0 + \beta_1$, mientras que para un placebo es $\log(tasa_{placebo,0}) = \beta_0$. Luego, el logaritmo de la esperanza del cociente de tasas será

$$\log \left[\frac{tasa_{tratado,0}}{tasa_{placebo,0}} \right] = \log(tasa_{tratado,0}) - \log(tasa_{placebo,0}) = \beta_0 + \beta_1 - \beta_0 = \beta_1.$$

- β_2 es el logaritmo del cociente entre la tasa esperada de convulsiones por semana en pacientes con placebo después y antes de la aleatorización. Observar que el modelo para el logaritmo de la tasa esperada en el placebo antes de la aleatorización es

$$\log(tasa_{placebo,0}) = \beta_0,$$

y después de la aleatorización

$$\log(tasa_{placebo,1}) = \beta_0 + \beta_2,$$

luego,

$$\log \left[\frac{tasa_{placebo,1}}{tasa_{placebo,0}} \right] = \log(tasa_{placebo,1}) - \log(tasa_{placebo,0}) = \beta_0 + \beta_2 - \beta_0 = \beta_2.$$

- β_3 es “la ganancia de estar tratado” cuando se evalúa la esperanza del logaritmo de la tasa antes y después de la aleatorización en comparación con lo que ocurre en el placebo. Si $\beta_3 = 0$, entonces entonces la esperanza del logaritmo del cociente de tasas antes y después aleatorización en pacientes tratados es igual que en pacientes que recibieron el placebo (β_2). Si β_3 es significativamente menor que 0, entonces el tratamiento es efectivo.

```
library(lme4)
library(MASS)
convulsiones_ML <- glm(convulsiones~1+trat+Despues.Aleat+trat:Despues.Aleat,
                      family=poisson(link=log), offset=log.tiempo,
                      data=epilepsia)
summary(convulsiones_ML)
```

Call:

```
glm(formula = convulsiones ~ 1 + trat + Despues.Aleat + trat:Despues.Aleat,
    family = poisson(link = log), data = epilepsia, offset = log.tiempo)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-5.4725	-2.5287	-1.5539	0.5805	18.2274

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	1.34761	0.03406	39.566	<2e-16 ***
trat	0.02651	0.04670	0.568	0.5702
Despues.Aleat	0.11080	0.04689	2.363	0.0181 *
trat:Despues.Aleat	-0.10368	0.06505	-1.594	0.1110

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 3584.1 on 294 degrees of freedom
 Residual deviance: 3577.8 on 291 degrees of freedom
 AIC: 4646.9

Number of Fisher Scoring iterations: 6

A continuación un primer intento de modelar la sobredispersión:

```
# Quasi-Poisson
convulsiones_PL<-glm(convulsiones~1+trat+Despues.Aleat+trat:Despues.Aleat,
                    family=quasipoisson(link=log),
                    offset=log.tiempo,
                    data=epilepsia)
summary(convulsiones_PL)
```

Call:

```
glm(formula = convulsiones ~ 1 + trat + Despues.Aleat + trat:Despues.Aleat,
     family = quasipoisson(link = log), data = epilepsia, offset = log.tiempo)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-5.4725	-2.5287	-1.5539	0.5805	18.2274

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	1.34761	0.15119	8.914	<2e-16 ***
trat	0.02651	0.20727	0.128	0.898
Despues.Aleat	0.11080	0.20813	0.532	0.595
trat:Despues.Aleat	-0.10368	0.28876	-0.359	0.720

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasipoisson family taken to be 19.70269)

Null deviance: 3584.1 on 294 degrees of freedom
 Residual deviance: 3577.8 on 291 degrees of freedom
 AIC: NA

Number of Fisher Scoring iterations: 6

```
# Modelo lineal generalizado mixto: intercepto aleatorio y Distribución Poisson
convulsiones_GLMM <- glmer(convulsiones~1+trat+Despues.Aleat+
                           trat:Despues.Aleat+(1|id),
                           family=poisson(link=log),
                           offset=log.tiempo,
                           data=epilepsia)
summary(convulsiones_GLMM)
```

Generalized linear mixed model fit by maximum likelihood (Laplace
 Approximation) [glmerMod]
 Family: poisson (log)
 Formula: convulsiones ~ 1 + trat + Despues.Aleat + trat:Despues.Aleat +
 (1 | id)
 Data: epilepsia
 Offset: log.tiempo

AIC	BIC	logLik	deviance	df.resid
2031.4	2049.8	-1010.7	2021.4	290

Scaled residuals:

Min	1Q	Median	3Q	Max
-4.8659	-0.9377	-0.1678	0.5853	9.8781

Random effects:

Groups	Name	Variance	Std.Dev.
id	(Intercept)	0.6083	0.7799

Number of obs: 295, groups: id, 59

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	1.03265	0.15255	6.769	1.29e-11 ***
trat	-0.02385	0.21047	-0.113	0.9098
Despues.Aleat	0.11080	0.04672	2.371	0.0177 *
trat:Despues.Aleat	-0.10368	0.06482	-1.599	0.1097

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:

	(Intr)	trat	Dsps.A
trat	-0.724		
Despues.Alt	-0.162	0.117	
trt:Dsps.Al	0.116	-0.159	-0.721

```
deviance(convulsiones_GLMM)
```

```
[1] 705.2927
```

Modelo lineal generalizado mixto: intercepto aleatorio y Distribución Binomial Neg

```
convulsiones_NB <- glmer.nb(convulsiones~1+trat+
                             Despues.Aleat+trat:Despues.Aleat+
                             (1|id),
                             offset=log.tiempo,
                             data=epilepsia)
```

```
summary(convulsiones_NB)
```

Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) [glmerMod]

Family: Negative Binomial(6.7803) (log)

Formula: convulsiones ~ 1 + trat + Despues.Aleat + trat:Despues.Aleat + (1 | id)

Data: epilepsia

Offset: log.tiempo

AIC	BIC	logLik	deviance	df.resid
-----	-----	--------	----------	----------

```

1789.5    1811.6    -888.7    1777.5        289

Scaled residuals:
    Min       1Q   Median       3Q      Max
-1.8269 -0.5474 -0.0971  0.4245  3.9026

Random effects:
 Groups Name      Variance Std.Dev.
 id      (Intercept) 0.661    0.813
Number of obs: 295, groups: id, 59

Fixed effects:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)      1.08031   0.17542   6.159 7.34e-10 ***
trat              0.07256   0.24205   0.300  0.7644
Despues.Aleat     0.02349   0.10029   0.234  0.8149
trat:Despues.Aleat -0.31042   0.14052  -2.209  0.0272 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
              (Intr) trat    Dsps.A
trat          -0.725
Despues.Alt  -0.402  0.291
trt:Dsps.Al   0.287 -0.399 -0.712

deviance(convulsiones_NB)

```

```
[1] 274.0781
```

Podemos apreciar que este modelo ajusta mejor que los anteriores usando Poisson (los valores de AIC y BIC son menores y el cociente deviance/GL es $274.08/291=0.94$).

Se puede verificar que β_3 es significativo ($p=0.0272$ en la prueba de Wald), y su valor estimado tiene un signo negativo (-0.31), por lo que se puede concluir que el tratamiento tiene el efecto de disminuir la tasa promedio de convulsiones para un sujeto dado.

Una estimación de la tasa esperada de convulsiones por semana para un sujeto “típico” (con efectos aleatorios $u = 0$, $v = 0$) del grupo tratado en el periodo post-aleatorización (es decir, una vez que está tomando el medicamento) es:

$$tasa_{tratado,1} = \exp(1,08031 + 0,07256 + 0,02349 - 0,31042) = 2,38.$$

La tasa análoga para un sujeto “típico” del grupo placebo es:

$$tasa_{placebo,1} = \exp(1,08031 + 0,02349) = 3,02.$$

Capítulo 3

Modelado de la Dependencia Temporal y Espacial. Datos Longitudinales y Curvas de Crecimiento

3.1. Datos Longitudinales

- Los datos longitudinales se originan en distintos contextos (estudios observacionales, experimentales, prospectivos, retrospectivos, etc.).
- La característica fundamental es que siempre tendremos varias observaciones tomadas secuencialmente sobre la misma unidad (persona, animal, parcela, árbol, sujeto, etc.).
- Los datos longitudinales también se denominan “mediciones repetidas”, especialmente en el contexto experimental; o datos de “panel”, en estudios económicos.
- Generalmente las observaciones se toman a lo largo del tiempo, pero podemos tener mediciones repetidas en una secuencia espacial, como por ejemplo a lo largo de una transecta.
- La característica fundamental es que debemos modelar la correlación existente entre observaciones longitudinales.
- Una manera de hacerlo es explícitamente mediante la matriz R (matriz de covarianza de los errores, o matriz de covarianza condicional de las observaciones dados los efectos aleatorios).
- La otra manera de hacerlo es mediante la incorporación de efectos aleatorios de

los sujetos. Esto genera correlación positiva entre observaciones longitudinales.

Ejemplo 3.1 (Diseño con medidas repetidas). Se prueban dos tratamientos, A y B, en 20 sujetos elegidos al azar (10 sujetos aleatoriamente asignados a cada tratamiento). La respuesta Y_{ijk} (i representa tratamiento, j tiempo y k sujeto) se observa en 2 momentos en el tiempo en cada sujeto (al mes y a los 6 meses de aplicar el tratamiento).

Efecto del tratamiento: $B =$

Efecto del tiempo: $W =$

Si suponemos que tenemos todas las observaciones independientes:

$$Var(B) = Var(W) = \frac{\sigma^2}{10}$$

Sin embargo, podemos probar que

$$Var(B) = \frac{\sigma^2(1 + \rho)}{10}, \quad Var(W) = \frac{\sigma^2(1 - \rho)}{10}.$$

3.2. Datos Espaciales

Los datos espaciales se originan en distintos contextos (estudios observacionales, experimentales).

La característica fundamental es que los datos están georreferenciados, y suponemos que están correlacionados de manera que las observaciones más cercanas tienen una correlación más alta que observaciones más alejadas.

Para considerar esta correlación espacial se han propuesto procedimientos estadísticos que van desde el ajuste de medias de tratamientos en función de lo observado en las parcelas vecinas más cercanas (Papadakis 1937), hasta el uso de modelos que contemplan las correlaciones espaciales en términos del error y que también producen ajustes de medias de tratamientos (Mead 1971, Besag 1974, 1977, Ripley 1981).

La variabilidad espacial se puede particionar en variabilidad espacial local y global. La variabilidad espacial local hace referencia a las diferencias entre parcelas a pequeña escala, donde se contemplan las variaciones intra-bloque. La tendencia espacial local y la heterogeneidad residual se modelan mediante la matriz de varianza y covarianza residual. A través de un sistema de coordenadas bidimensionales es posible definir la ubicación de las observaciones en el campo.

La modelación de la estructura espacial de parcelas a partir de funciones de distancia puede realizarse en el contexto de los modelos lineales mixtos, donde además de contem-

plar la estructura de correlación entre observaciones provenientes de distintas parcelas es posible modelar heterogeneidad de varianza residual.

Si la correlación solo depende de la distancia (magnitud y/o dirección de las distancias), los modelos que estiman las covarianzas entre observaciones se denominan estacionarios.

Las funciones de correlación para modelos estacionarios pueden ser isotrópicas o anisotrópicas. Las primeras son idénticas en cualquier dirección (sólo dependen de la magnitud de las distancias) mientras que las segundas permiten diferentes valores de sus parámetros en diferentes direcciones (i.e. dependen también de la dirección sobre la cual se calculan las distancias).

3.3. Análisis de datos correlacionados en el tiempo

- Aspecto más importante a considerar es la estructura de la matriz de covarianza residual (correlaciones y varianzas residuales).
- Las varianzas residuales suelen mostrar heterogeneidad.
- Es común la presencia de datos faltantes.
- Existe correlación residual entre observaciones que comparten el mismo valor del criterio de estratificación, conocido también como sujeto (persona, parcela, animal, árbol, etc.).
- La matriz \mathbf{R} será diagonal por bloques, y en cada bloque se reflejará la estructura deseada, i.e. simetría compuesta, autorregresiva de orden 1, etc.

3.3.1. Matriz de covarianza residual

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{Z}\mathbf{u} + \mathbf{e},$$

$$\begin{bmatrix} \mathbf{u} \\ \mathbf{e} \end{bmatrix} \sim N \left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{G} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{bmatrix} \right)$$

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_2 & \cdots & \mathbf{0} \\ \cdots & \cdots & \ddots & \cdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{R}_N \end{bmatrix}$$

donde

$$\mathbf{R}_i = \text{Diag}^{1/2}(\sigma_{ij}) \text{Corr}_i \text{Diag}^{1/2}(\sigma_{ij}).$$

Supongamos por un momento que no tenemos efectos aleatorios. Si en el modelo lineal mixto pensamos que tenemos N unidades independientes, entonces la matriz \mathbf{R} tendrá una estructura diagonal por bloques:

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_2 & \cdots & \mathbf{0} \\ \cdots & \cdots & \ddots & \cdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{R}_N \end{bmatrix}$$

Cada uno de estos bloques será típicamente igual, y tendrá una estructura que refleje las varianzas y correlaciones entre las observaciones de la misma unidad.

3.3.2. Estructura de Errores Independientes

$$\mathbf{R}_i = \sigma_e^2 \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \ddots & \cdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$

3.3.3. Estructura de Simetría Compuesta

$$\mathbf{R}_i = \sigma_e^2 \begin{bmatrix} 1 & \rho & \cdots & \rho \\ \rho & 1 & \cdots & \rho \\ \cdots & \cdots & \ddots & \cdots \\ \rho & \rho & \cdots & 1 \end{bmatrix}$$

Si $\rho > 0$, es equivalente a pensar en \mathbf{R} independiente y añadir un intercepto aleatorio a nivel de sujeto.

3.3.4. Estructura autorregresiva de primer orden

$$\mathbf{R}_i = \sigma_e^2 \begin{bmatrix} 1 & \rho & \rho^2 & \cdots & \rho^{T-1} \\ \rho & 1 & \rho & \cdots & \rho^{T-2} \\ \cdots & \cdots & \cdots & \ddots & \cdots \\ \rho^{T-1} & \rho^{T-2} & \cdots & \cdots & \rho \end{bmatrix}$$

Si las ocasiones no están equiespaciadas, existe una estructura equivalente

$$\text{corr}(e_i, e_j) = \rho^{|t_i - t_j|}.$$

3.3.5. Sin estructura

$$\mathbf{R}_i = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \sigma_{13} & \cdots & \sigma_{1T} \\ \sigma_{12} & \sigma_2^2 & \sigma_{23} & \cdots & \sigma_{2T} \\ \cdots & \cdots & \cdots & \ddots & \cdots \\ \sigma_{1T} & \sigma_{2T} & \cdots & \cdots & \sigma_T^2 \end{bmatrix}$$

También puede parametrizarse en términos de varianzas y correlaciones (más intuitivo).

3.4. Estructuras de correlación espacial

Al igual que en las medidas repetidas en el tiempo, típicamente las observaciones más cercanas estarán más correlacionadas entre sí.

Por lo tanto, la correlación entre los errores (o las observaciones) será una función de la distancia entre éstos.

La distancia Euclídea (Euclidiana) entre dos puntos P y Q , referenciados en un sistema de coordenadas de dos dimensiones (x, y) , es

$$d_{PQ} = \sqrt{(x_P - x_Q)^2 + (y_P - y_Q)^2}$$

- **Esférica:**

$$Corr(e_i, e_j) = f(d_{ij}) = \left[1 - 1,5 \left(\frac{d_{ij}}{\rho} \right) + 0,5 \left(\frac{d_{ij}}{\rho} \right)^3 \right], \quad d_{ij} < \rho.$$

- **Exponencial:**

$$Corr(e_i, e_j) = f(d_{ij}) = \exp(-d_{ij}/\rho).$$

- **Gaussiana:**

$$Corr(e_i, e_j) = f(d_{ij}) = \exp(-d_{ij}^2/\rho^2).$$

- **Lineal:**

$$Corr(e_i, e_j) = f(d_{ij}) = 1 - \rho d_{ij}, \quad d_{ij} < 1/\rho.$$

Existen situaciones en las que localmente se observa más variabilidad que la que esperaríamos con los modelos anteriores, aunque la correlación sigue dependiendo de la distancia entre las observaciones.

Este efecto puede modelarse como una varianza adicional (efecto “nugget”):

$$\begin{aligned} \text{Var}(e_i) &= \sigma^2 + \sigma_1^2, \\ \text{Cov}(e_i, e_j) &= \sigma^2 f(d_{ij}). \end{aligned}$$

3.5. Enfoques para datos longitudinales

- Si las ocasiones de medición son pocas y siempre (o casi siempre) las mismas para todos los sujetos, se puede formular directamente un modelo mixto con una estructura de covarianzas para los errores (**Modelo marginal**).
- Si las ocasiones de medición son muchas, o no son las mismas, entonces es posiblemente más práctico inducir la correlación indirectamente mediante efectos aleatorios de sujeto, o curvas aleatorias sujeto-específicas (**Modelo condicional**).

3.6. Selección de la estructura de covarianza

- Seleccionar modelos para la media (sobreajustando si es necesario: ¡necesitamos buenos residuos!)
- Ajustar distintos efectos aleatorios y estructuras de covarianza (matrices **G** y **R**).
- Seleccionar la estructura de covarianza.
- Usando pruebas de cociente de verosimilitud (ML o REML) si son modelos anidados.
- Usando criterios de verosimilitud penalizada.
- Se debe tener cuidado de no ajustar modelos no identificables (por ejemplo un intercepto aleatorio y una simetría compuesta al mismo nivel).

Ejemplo 3.2 (Ensayo clínico para tratamiento de asma). Una compañía farmacéutica ha examinado los efectos de dos drogas (A y B) sobre la capacidad respiratoria de pacientes de asma (Littell et ál. 2006).

Las dos drogas y un placebo (P) fueron administradas a un grupo de pacientes en forma aleatoria. Se contó con 24 pacientes por cada uno de los tres tratamientos. A cada paciente se le midió la capacidad respiratoria basal (Cap_Rep_Bas) inmediatamente antes de aplicarle el tratamiento y la capacidad respiratoria cada hora durante las 8 horas siguientes (Cap_Respirat).

Usando nuevamente la estrategia discutida antes, primero se ajustarán modelos con distintas estructuras de covarianza, combinando apropiadamente estructuras de correlación residual, heteroscedasticidad residual y efectos aleatorios.

Mediante criterios de verosimilitud penalizada (AIC y BIC) y pruebas de cociente de verosimilitud se elegirá el modelo que mejor describa los datos.

Una vez seleccionado el modelo de estructura de covarianza adecuado se realizarán inferencias acerca de las medias.

Es importante destacar que toda la inferencia sobre las medias estará basada en el modelo de estructura de covarianza seleccionado.

3.6.1. Modelos a considerar

1. Errores independientes y varianzas residuales heterogéneas.
2. Simetría compuesta y varianzas residuales homogéneas.
3. Auto regresivo de orden 1 y varianzas residuales homogéneas.
4. Auto regresivo de orden 1 y varianzas residuales heterogéneas.
5. Auto regresivo de orden 1, varianzas residuales homogéneas y efecto aleatorio de paciente.
6. Auto regresivo de orden 1, varianzas residuales heterogéneas y efecto aleatorio de paciente.
7. Matriz de correlación sin estructura y varianzas residuales heterogéneas.

```
head(CapacidadRespiratoria)
```

```
# A tibble: 6 x 6
  Paciente Cap_Resp_base Droga Hora Cap_Respirat Paciente_Droga
  <int>      <dbl> <chr> <int>      <dbl> <chr>
1     201        2.46 A         1        2.68 201_A
2     202        3.50 A         1        3.95 202_A
3     203        1.96 A         1        2.28 203_A
4     204        3.44 A         1        4.08 204_A
5     205        2.80 A         1        4.09 205_A
6     206        2.36 A         1        3.79 206_A
```

```
library(nlme)
fit_Cap_Respirat6<-lme(Cap_Respirat~1+Droga+factor(Hora)
  +Droga:factor(Hora)+Cap_Resp_base,
  weight=varComb(varIdent(form=~1|Hora)),
  random=list(Paciente_Droga=pdIdent(~1)),
  correlation=corAR1(form=~Hora|Paciente_Droga),
  method="REML",
  control=lmeControl(niterEM=150,msMaxIter=200),
```

```
data=CapacidadRespiratoria)
summary(fit_Cap_Respirat6)
```

Linear mixed-effects model fit by REML

Data: CapacidadRespiratoria

AIC	BIC	logLik
287.7959	438.7067	-108.898

Random effects:

Formula: ~1 | Paciente_Droga
(Intercept) Residual

StdDev: 0.4358967 0.243054

Correlation Structure: AR(1)

Formula: ~Hora | Paciente_Droga

Parameter estimate(s):

Phi
0.5896629

Combination of variance functions:

Structure: Different standard deviations per stratum

Formula: ~1 | Hora

Parameter estimates:

	1	2	3	4	5	6	7
1.0000000	1.0743275	0.8730669	1.2527525	1.3456543	1.2297396	1.5931443	
8							
1.3088023							

Fixed effects: Cap_Respirat ~ 1 + Droga + factor(Hora) + Droga:factor(Hora) +

	Value	Std.Error	DF	t-value	p-value
(Intercept)	1.1017800	0.29442943	483	3.742085	0.0002
DrogaB	0.2182411	0.14409455	68	1.514569	0.1345
DrogaP	-0.6446770	0.14411135	68	-4.473464	0.0000
factor(Hora)2	-0.0766667	0.04673123	483	-1.640587	0.1015
factor(Hora)3	-0.2895833	0.05332231	483	-5.430810	0.0000
factor(Hora)4	-0.4270833	0.07113382	483	-6.003942	0.0000
factor(Hora)5	-0.4200000	0.07821617	483	-5.369734	0.0000
factor(Hora)6	-0.4941667	0.07584373	483	-6.515590	0.0000
factor(Hora)7	-0.6050000	0.09153827	483	-6.609258	0.0000
factor(Hora)8	-0.6154167	0.08073516	483	-7.622660	0.0000
Cap_Resp_base	0.8945546	0.10352650	68	8.640828	0.0000
DrogaB:factor(Hora)2	0.0120833	0.06608794	483	0.182837	0.8550
DrogaP:factor(Hora)2	0.1425000	0.06608794	483	2.156218	0.0316
DrogaB:factor(Hora)3	0.1758333	0.07540913	483	2.331725	0.0201

DrogaP:factor(Hora)3	0.3608333	0.07540913	483	4.785008	0.0000		
DrogaB:factor(Hora)4	0.1795833	0.10059842	483	1.785151	0.0749		
DrogaP:factor(Hora)4	0.4716667	0.10059842	483	4.688609	0.0000		
DrogaB:factor(Hora)5	-0.0216667	0.11061436	483	-0.195876	0.8448		
DrogaP:factor(Hora)5	0.3616667	0.11061436	483	3.269618	0.0012		
DrogaB:factor(Hora)6	-0.1116667	0.10725923	483	-1.041091	0.2984		
DrogaP:factor(Hora)6	0.4829167	0.10725923	483	4.502332	0.0000		
DrogaB:factor(Hora)7	-0.1091667	0.12945466	483	-0.843281	0.3995		
DrogaP:factor(Hora)7	0.5629167	0.12945466	483	4.348369	0.0000		
DrogaB:factor(Hora)8	-0.0654167	0.11417676	483	-0.572942	0.5670		
DrogaP:factor(Hora)8	0.5208333	0.11417676	483	4.561641	0.0000		
Correlation:							
	(Intr)	DrogaB	DrogaP	fc(H)2	fc(H)3	fc(H)4	fc(H)5
DrogaB	-0.261						
DrogaP	-0.267	0.500					
factor(Hora)2	-0.066	0.134	0.134				
factor(Hora)3	-0.109	0.223	0.223	0.609			
factor(Hora)4	-0.087	0.178	0.178	0.428	0.704		
factor(Hora)5	-0.089	0.183	0.183	0.337	0.556	0.697	
factor(Hora)6	-0.101	0.205	0.205	0.305	0.505	0.543	0.716
factor(Hora)7	-0.085	0.174	0.174	0.243	0.402	0.410	0.521
factor(Hora)8	-0.100	0.205	0.205	0.257	0.426	0.390	0.454
Cap_Resp_base	-0.938	0.018	0.023	0.000	0.000	0.000	0.000
DrogaB:factor(Hora)2	0.046	-0.189	-0.095	-0.707	-0.430	-0.302	-0.238
DrogaP:factor(Hora)2	0.046	-0.095	-0.189	-0.707	-0.430	-0.302	-0.238
DrogaB:factor(Hora)3	0.077	-0.316	-0.158	-0.430	-0.707	-0.498	-0.393
DrogaP:factor(Hora)3	0.077	-0.158	-0.315	-0.430	-0.707	-0.498	-0.393
DrogaB:factor(Hora)4	0.062	-0.252	-0.126	-0.302	-0.498	-0.707	-0.493
DrogaP:factor(Hora)4	0.062	-0.126	-0.252	-0.302	-0.498	-0.707	-0.493
DrogaB:factor(Hora)5	0.063	-0.259	-0.129	-0.238	-0.393	-0.493	-0.707
DrogaP:factor(Hora)5	0.063	-0.129	-0.259	-0.238	-0.393	-0.493	-0.707
DrogaB:factor(Hora)6	0.071	-0.291	-0.145	-0.215	-0.357	-0.384	-0.506
DrogaP:factor(Hora)6	0.071	-0.145	-0.291	-0.215	-0.357	-0.384	-0.506
DrogaB:factor(Hora)7	0.060	-0.246	-0.123	-0.172	-0.284	-0.290	-0.368
DrogaP:factor(Hora)7	0.060	-0.123	-0.246	-0.172	-0.284	-0.290	-0.368
DrogaB:factor(Hora)8	0.071	-0.290	-0.145	-0.181	-0.301	-0.275	-0.321
DrogaP:factor(Hora)8	0.071	-0.145	-0.289	-0.181	-0.301	-0.275	-0.321
	fc(H)6	fc(H)7	fc(H)8	Cp_Rs_	DB:(H)2	DP:(H)2	DB:(H)3
DrogaB							
DrogaP							
factor(Hora)2							
factor(Hora)3							
factor(Hora)4							

```

factor(Hora)5
factor(Hora)6
factor(Hora)7      0.709
factor(Hora)8      0.579  0.709
Cap_Resp_base      0.000  0.000  0.000
DrogaB:factor(Hora)2 -0.215 -0.172 -0.181  0.000
DrogaP:factor(Hora)2 -0.215 -0.172 -0.181  0.000  0.500
DrogaB:factor(Hora)3 -0.357 -0.284 -0.301  0.000  0.609  0.304
DrogaP:factor(Hora)3 -0.357 -0.284 -0.301  0.000  0.304  0.609  0.500
DrogaB:factor(Hora)4 -0.384 -0.290 -0.275  0.000  0.428  0.214  0.704
DrogaP:factor(Hora)4 -0.384 -0.290 -0.275  0.000  0.214  0.428  0.352
DrogaB:factor(Hora)5 -0.506 -0.368 -0.321  0.000  0.337  0.168  0.556
DrogaP:factor(Hora)5 -0.506 -0.368 -0.321  0.000  0.168  0.337  0.278
DrogaB:factor(Hora)6 -0.707 -0.502 -0.409  0.000  0.305  0.152  0.505
DrogaP:factor(Hora)6 -0.707 -0.502 -0.409  0.000  0.152  0.305  0.252
DrogaB:factor(Hora)7 -0.502 -0.707 -0.502  0.000  0.243  0.121  0.402
DrogaP:factor(Hora)7 -0.502 -0.707 -0.502  0.000  0.121  0.243  0.201
DrogaB:factor(Hora)8 -0.409 -0.502 -0.707  0.000  0.257  0.128  0.426
DrogaP:factor(Hora)8 -0.409 -0.502 -0.707  0.000  0.128  0.257  0.213

```

DP: (H)3 DB: (H)4 DP: (H)4 DB: (H)5 DP: (H)5 DB: (H)6

```

DrogaB
DrogaP
factor(Hora)2
factor(Hora)3
factor(Hora)4
factor(Hora)5
factor(Hora)6
factor(Hora)7
factor(Hora)8
Cap_Resp_base
DrogaB:factor(Hora)2
DrogaP:factor(Hora)2
DrogaB:factor(Hora)3
DrogaP:factor(Hora)3
DrogaB:factor(Hora)4  0.352
DrogaP:factor(Hora)4  0.704  0.500
DrogaB:factor(Hora)5  0.278  0.697  0.348
DrogaP:factor(Hora)5  0.556  0.348  0.697  0.500
DrogaB:factor(Hora)6  0.252  0.543  0.272  0.716  0.358
DrogaP:factor(Hora)6  0.505  0.272  0.543  0.358  0.716  0.500
DrogaB:factor(Hora)7  0.201  0.410  0.205  0.521  0.261  0.709
DrogaP:factor(Hora)7  0.402  0.205  0.410  0.261  0.521  0.355
DrogaB:factor(Hora)8  0.213  0.390  0.195  0.454  0.227  0.579

```



```
DrogaP:factor(Hora)8  0.426  0.195  0.390  0.227  0.454  0.289
                      DP:(H)6 DB:(H)7 DP:(H)7 DB:(H)8
```

```
DrogaB
DrogaP
factor(Hora)2
factor(Hora)3
factor(Hora)4
factor(Hora)5
factor(Hora)6
factor(Hora)7
factor(Hora)8
Cap_Resp_base
DrogaB:factor(Hora)2
DrogaP:factor(Hora)2
DrogaB:factor(Hora)3
DrogaP:factor(Hora)3
DrogaB:factor(Hora)4
DrogaP:factor(Hora)4
DrogaB:factor(Hora)5
DrogaP:factor(Hora)5
DrogaB:factor(Hora)6
DrogaP:factor(Hora)6
DrogaB:factor(Hora)7  0.355
DrogaP:factor(Hora)7  0.709  0.500
DrogaB:factor(Hora)8  0.289  0.709  0.355
DrogaP:factor(Hora)8  0.579  0.355  0.709  0.500
```

```
Standardized Within-Group Residuals:
      Min      Q1      Med      Q3      Max
-2.88591041 -0.49045533  0.01443519  0.51159757  3.34595767
```

```
Number of Observations: 576
Number of Groups: 72
```

```
fit_Cap_Respirat7<-gls(Cap_Respirat~1+Droga+factor(Hora)
  +Droga:factor(Hora)+Cap_Resp_base,
  weight=varComb(varIdent(form=~1|Hora)),
  correlation=corSymm(form=~Hora|Paciente_Droga),
  method="REML",
  data=CapacidadRespiratoria)
summary(fit_Cap_Respirat7)
```

Generalized least squares fit by REML

Model: Cap_Respirat ~ 1 + Droga + factor(Hora) + Droga:factor(Hora) +
 Data: CapacidadRespiratoria
 AIC BIC logLik
 270.2702 533.2861 -74.13512

Cap_Resp_

Correlation Structure: General

Formula: ~Hora | Paciente_Droga

Parameter estimate(s):

Correlation:

	1	2	3	4	5	6	7
2	0.893						
3	0.880	0.908					
4	0.785	0.872	0.915				
5	0.688	0.807	0.813	0.822			
6	0.675	0.698	0.745	0.735	0.855		
7	0.517	0.590	0.643	0.670	0.733	0.812	
8	0.646	0.701	0.742	0.755	0.845	0.882	0.820

Combination of variance functions:

Structure: Different standard deviations per stratum

Formula: ~1 | Hora

Parameter estimates:

	1	2	3	4	5	6	7	8
1.000000	1.070385	1.058554	1.148414	1.123668	1.066432	1.092585	1.149450	

Coefficients:

	Value	Std.Error	t-value	p-value
(Intercept)	1.0420980	0.27222209	3.828117	0.0001
DrogaB	0.2187910	0.13740939	1.592256	0.1119
DrogaP	-0.6439501	0.13742432	-4.685852	0.0000
factor(Hora)2	-0.0766667	0.04698213	-1.631826	0.1033
factor(Hora)3	-0.2895833	0.04922182	-5.883231	0.0000
factor(Hora)4	-0.4270833	0.06983476	-6.115627	0.0000
factor(Hora)5	-0.4200000	0.08219991	-5.109495	0.0000
factor(Hora)6	-0.4941667	0.08114386	-6.090007	0.0000
factor(Hora)7	-0.6050000	0.10024183	-6.035405	0.0000
factor(Hora)8	-0.6154167	0.08880450	-6.930017	0.0000
Cap_Resp_base	0.9169214	0.09530179	9.621240	0.0000
DrogaB:factor(Hora)2	0.0120833	0.06644276	0.181861	0.8558
DrogaP:factor(Hora)2	0.1425000	0.06644276	2.144703	0.0324
DrogaB:factor(Hora)3	0.1758333	0.06961016	2.525972	0.0118
DrogaP:factor(Hora)3	0.3608333	0.06961016	5.183630	0.0000
DrogaB:factor(Hora)4	0.1795833	0.09876126	1.818358	0.0696
DrogaP:factor(Hora)4	0.4716667	0.09876126	4.775827	0.0000

DrogaB:factor(Hora)5	-0.0216667	0.11624822	-0.186383	0.8522
DrogaP:factor(Hora)5	0.3616667	0.11624822	3.111159	0.0020
DrogaB:factor(Hora)6	-0.1116667	0.11475474	-0.973090	0.3309
DrogaP:factor(Hora)6	0.4829167	0.11475474	4.208250	0.0000
DrogaB:factor(Hora)7	-0.1091667	0.14176335	-0.770063	0.4416
DrogaP:factor(Hora)7	0.5629167	0.14176335	3.970819	0.0001
DrogaB:factor(Hora)8	-0.0654167	0.12558853	-0.520881	0.6027
DrogaP:factor(Hora)8	0.5208333	0.12558853	4.147141	0.0000

Correlation:

	(Intr)	DrogaB	DrogaP	fc(H)2	fc(H)3	fc(H)4	fc(H)5
DrogaB	-0.268						
DrogaP	-0.273	0.500					
factor(Hora)2	-0.033	0.064	0.064				
factor(Hora)3	-0.048	0.095	0.095	0.577			
factor(Hora)4	-0.049	0.097	0.097	0.619	0.767		
factor(Hora)5	-0.096	0.189	0.189	0.591	0.611	0.635	
factor(Hora)6	-0.120	0.237	0.237	0.300	0.448	0.465	0.751
factor(Hora)7	-0.151	0.298	0.298	0.340	0.472	0.505	0.644
factor(Hora)8	-0.100	0.199	0.199	0.370	0.492	0.536	0.743
Cap_Resp_base	-0.934	0.017	0.023	0.000	0.000	0.000	0.000
DrogaB:factor(Hora)2	0.023	-0.091	-0.046	-0.707	-0.408	-0.438	-0.418
DrogaP:factor(Hora)2	0.023	-0.046	-0.091	-0.707	-0.408	-0.438	-0.418
DrogaB:factor(Hora)3	0.034	-0.134	-0.067	-0.408	-0.707	-0.542	-0.432
DrogaP:factor(Hora)3	0.034	-0.067	-0.134	-0.408	-0.707	-0.542	-0.432
DrogaB:factor(Hora)4	0.035	-0.138	-0.069	-0.438	-0.542	-0.707	-0.449
DrogaP:factor(Hora)4	0.035	-0.069	-0.138	-0.438	-0.542	-0.707	-0.449
DrogaB:factor(Hora)5	0.068	-0.268	-0.134	-0.418	-0.432	-0.449	-0.707
DrogaP:factor(Hora)5	0.068	-0.134	-0.268	-0.418	-0.432	-0.449	-0.707
DrogaB:factor(Hora)6	0.085	-0.335	-0.168	-0.212	-0.317	-0.329	-0.531
DrogaP:factor(Hora)6	0.085	-0.168	-0.335	-0.212	-0.317	-0.329	-0.531
DrogaB:factor(Hora)7	0.106	-0.422	-0.211	-0.240	-0.334	-0.357	-0.455
DrogaP:factor(Hora)7	0.106	-0.211	-0.422	-0.240	-0.334	-0.357	-0.455
DrogaB:factor(Hora)8	0.071	-0.281	-0.141	-0.262	-0.348	-0.379	-0.526
DrogaP:factor(Hora)8	0.071	-0.141	-0.281	-0.262	-0.348	-0.379	-0.526
	fc(H)6	fc(H)7	fc(H)8	Cp_Rs_	DB:(H)2	DP:(H)2	DB:(H)3
DrogaB							
DrogaP							
factor(Hora)2							
factor(Hora)3							
factor(Hora)4							
factor(Hora)5							
factor(Hora)6							

factor(Hora)7	0.768						
factor(Hora)8	0.809	0.766					
Cap_Resp_base	0.000	0.000	0.000				
DrogaB:factor(Hora)2	-0.212	-0.240	-0.262	0.000			
DrogaP:factor(Hora)2	-0.212	-0.240	-0.262	0.000	0.500		
DrogaB:factor(Hora)3	-0.317	-0.334	-0.348	0.000	0.577	0.289	
DrogaP:factor(Hora)3	-0.317	-0.334	-0.348	0.000	0.289	0.577	0.500
DrogaB:factor(Hora)4	-0.329	-0.357	-0.379	0.000	0.619	0.310	0.767
DrogaP:factor(Hora)4	-0.329	-0.357	-0.379	0.000	0.310	0.619	0.383
DrogaB:factor(Hora)5	-0.531	-0.455	-0.526	0.000	0.591	0.296	0.611
DrogaP:factor(Hora)5	-0.531	-0.455	-0.526	0.000	0.296	0.591	0.306
DrogaB:factor(Hora)6	-0.707	-0.543	-0.572	0.000	0.300	0.150	0.448
DrogaP:factor(Hora)6	-0.707	-0.543	-0.572	0.000	0.150	0.300	0.224
DrogaB:factor(Hora)7	-0.543	-0.707	-0.541	0.000	0.340	0.170	0.472
DrogaP:factor(Hora)7	-0.543	-0.707	-0.541	0.000	0.170	0.340	0.236
DrogaB:factor(Hora)8	-0.572	-0.541	-0.707	0.000	0.370	0.185	0.492
DrogaP:factor(Hora)8	-0.572	-0.541	-0.707	0.000	0.185	0.370	0.246

DP: (H)3 DB: (H)4 DP: (H)4 DB: (H)5 DP: (H)5 DB: (H)6

DrogaB
DrogaP
factor(Hora)2
factor(Hora)3
factor(Hora)4
factor(Hora)5
factor(Hora)6
factor(Hora)7
factor(Hora)8
Cap_Resp_base
DrogaB:factor(Hora)2
DrogaP:factor(Hora)2
DrogaB:factor(Hora)3
DrogaP:factor(Hora)3
DrogaB:factor(Hora)4 0.383
DrogaP:factor(Hora)4 0.767 0.500
DrogaB:factor(Hora)5 0.306 0.635 0.317
DrogaP:factor(Hora)5 0.611 0.317 0.635 0.500
DrogaB:factor(Hora)6 0.224 0.465 0.232 0.751 0.376
DrogaP:factor(Hora)6 0.448 0.232 0.465 0.376 0.751 0.500
DrogaB:factor(Hora)7 0.236 0.505 0.253 0.644 0.322 0.768
DrogaP:factor(Hora)7 0.472 0.253 0.505 0.322 0.644 0.384
DrogaB:factor(Hora)8 0.246 0.536 0.268 0.743 0.372 0.809
DrogaP:factor(Hora)8 0.492 0.268 0.536 0.372 0.743 0.405

DP: (H)6 DB: (H)7 DP: (H)7 DB: (H)8

```

DrogaB
DrogaP
factor(Hora)2
factor(Hora)3
factor(Hora)4
factor(Hora)5
factor(Hora)6
factor(Hora)7
factor(Hora)8
Cap_Resp_base
DrogaB:factor(Hora)2
DrogaP:factor(Hora)2
DrogaB:factor(Hora)3
DrogaP:factor(Hora)3
DrogaB:factor(Hora)4
DrogaP:factor(Hora)4
DrogaB:factor(Hora)5
DrogaP:factor(Hora)5
DrogaB:factor(Hora)6
DrogaP:factor(Hora)6
DrogaB:factor(Hora)7 0.384
DrogaP:factor(Hora)7 0.768 0.500
DrogaB:factor(Hora)8 0.405 0.766 0.383
DrogaP:factor(Hora)8 0.809 0.383 0.766 0.500

```

Standardized residuals:

	Min	Q1	Med	Q3	Max
	-2.57843870	-0.69613225	-0.08948268	0.58940159	3.09341703

Residual standard error: 0.4759309
Degrees of freedom: 576 total; 551 residual

3.6.2. Comparación de modelos

Modelo	Paciente (Aleatorio)*	Corr. Residual**	Var. Residual por Tiempo***	AIC	BIC	Log Lik
1	No	-	No	968.94	1081.04	- 458.47
2	No	corCompSymm	No	401.29	517.71	- 173.65

Modelo	Paciente (Aleatorio)*	Corr. Residual**	Var. Residual por Tiempo***	AIC	BIC	Log Lik
3	No	corAR1	No	329.04	445.45	- 137.52
4	No	corAR1	Si	324.57	471.17	- 128.28
5	Si	corAR1	No	303.03	423.76	- 123.52
6	Si	corAR1	Si	287.80	438.71	- 108.90
7	No	corSymm	Si	270.27	533.29	-74.14

Comandos en R:

```
* random=list(Paciente_Droga=pdIdent(~1))
```

```
** correlation=corAR1(form=~Hora|Paciente_Droga)
```

```
*** weight=varComb(varIdent(form=~1|Hora))
```

```

              df      AIC      BIC    logLik L.Ratio p-value
fit_Cap_Respirat5 28 303.0306 423.7592 -123.5153
fit_Cap_Respirat6 35 287.7960 438.7067 -108.8980 29.2347 1e-04

```

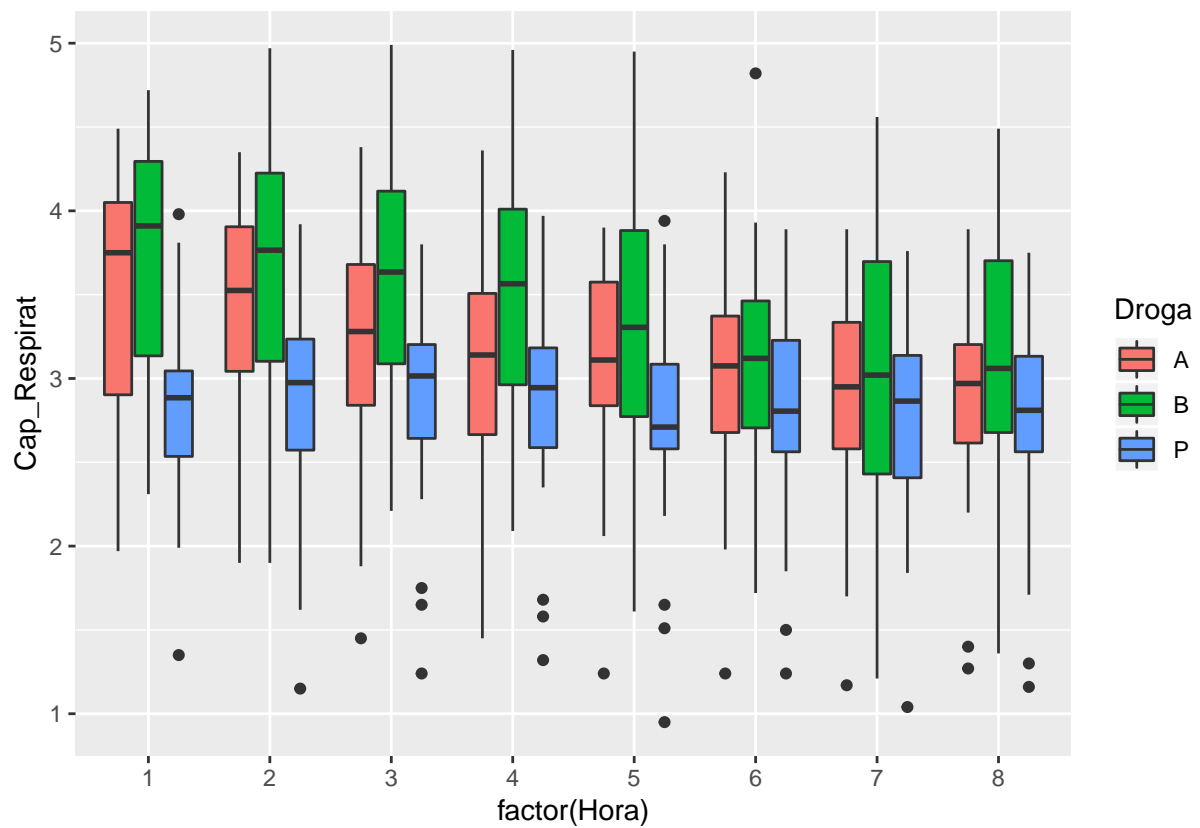
```

              df      AIC      BIC    logLik L.Ratio p-value
fit_Cap_Respirat6 35 287.7960 438.7067 -108.8980
fit_Cap_Respirat7 61 270.2702 533.2861 -74.13512 69.5257 <.0001

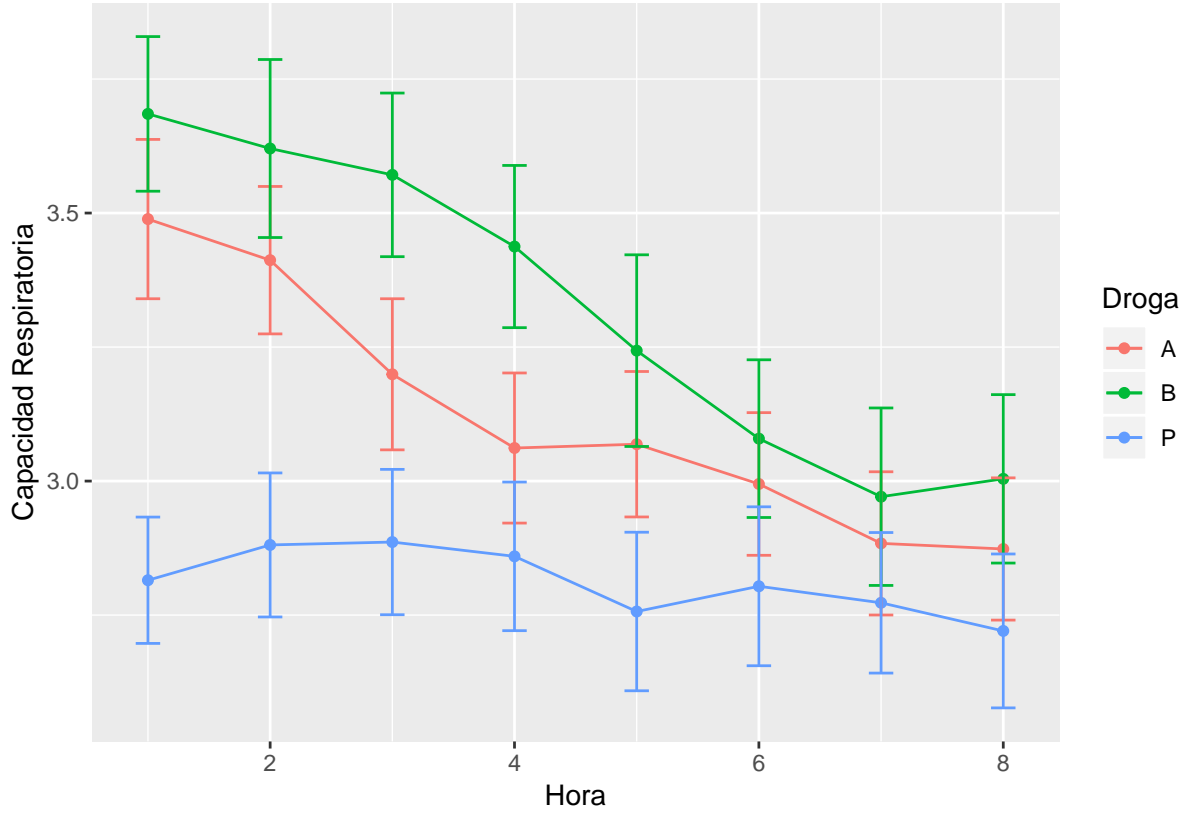
```

```
library(ggplot2)
```

```
ggplot(CapacidadRespiratoria, aes(x=factor(Hora), y=Cap_Respirat, fill=Droga))+geom_b
```



```
library(dplyr)
summary <- CapacidadRespiratoria %>% group_by(Hora, Droga)%>%
  summarize(media=mean(Cap_Respirat), sd=sd(Cap_Respirat), n=n() )
summary$se <- summary$sd / sqrt(summary$n)
library(ggplot2)
ggplot(summary, aes(x=Hora, y=media, colour=Droga))+geom_line()+geom_point()+geom_err
```



3.6.3. Correlación Espacial

El variograma de los residuos del modelo de regresión clásico muestra que hasta cierta distancia hay correlación. Recordemos para dos puntos \mathbf{x} y \mathbf{x}' el variograma es

$$\text{Variog}(\mathbf{x}, \mathbf{x}') = \frac{1}{2} \text{Var}[S(\mathbf{x}) - S(\mathbf{x}')].$$

Si las varianzas de $S(\mathbf{x})$ son constantes, y la correlación solo depende de la distancia entre los puntos \mathbf{x} y \mathbf{x}' , entonces

$$\text{Variog}(\mathbf{x}, \mathbf{x}') = \frac{1}{2} \{ \text{Var}(S(\mathbf{x})) + \text{Var}(S(\mathbf{x}')) - 2\text{Cov}(S(\mathbf{x}), S(\mathbf{x}')) \} = \sigma^2(1 - \rho(u)),$$

donde u es la distancia entre los puntos.

Ejemplo 3.3 (Animales capturados). Los datos en `habsel.txt` representan el número de animales capturados (`captures`) en 71 trampas permanentes ubicadas en coordenadas x y y en una isla noruega (Datos de Ottar Bjornstad, tomados de [<https://ms.mcmaster.ca/~bolker/eeid/lme.pdf>]). Se informan también varias variables del hábitat en la vecindad de la trampa. Nos concentraremos en `lichen` (cobertura de liquen), `veg` (cobertura de

vegetación total), **heather** (cobertura de brezos), **moss** (cobertura de musgos) y **stdp** (una medida de la complejidad estructural).

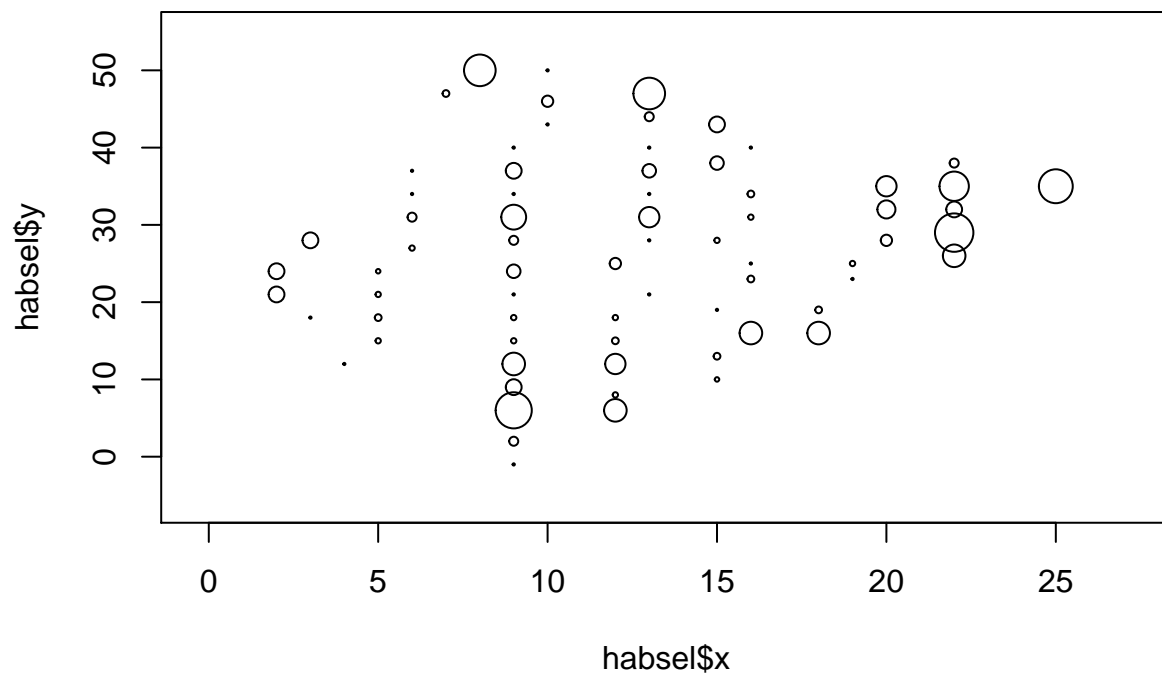
```
head(habsel)
```

```
# A tibble: 6 x 21
```

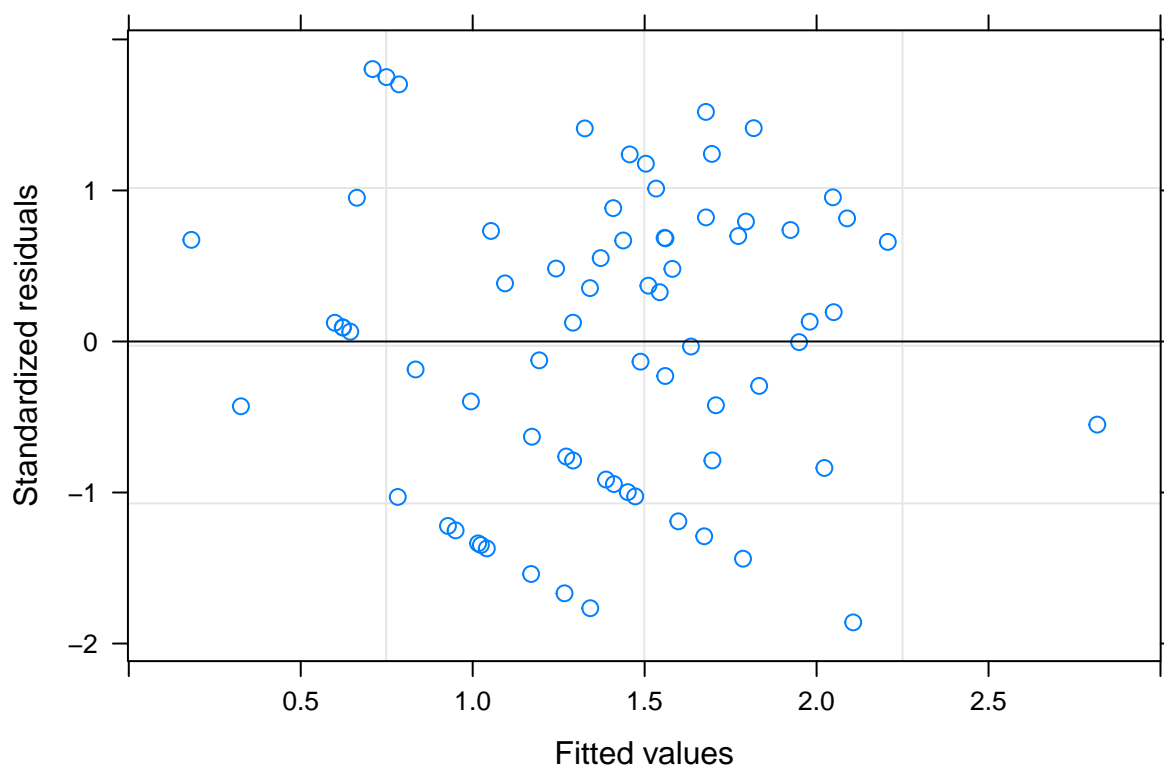
	Strap	captures	x	y	vm	vvi	heather	calluna	baer	moss	etgm
	<int>	<int>	<int>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	1	1	9	-1	1.38	1.25	0.625	0.	0.700	1.88	1.75
2	2	4	9	2	1.00	1.88	1.25	0.	0.700	1.88	1.12
3	3	16	9	6	1.62	1.50	0.750	0.	0.700	2.12	0.500
4	4	7	9	9	2.00	1.25	0.	0.250	0.750	2.25	0.
5	5	10	9	12	2.00	1.12	0.	0.	0.625	2.25	0.
6	6	0	9	15	0.375	0.625	0.	0.125	0.700	2.38	0.

```
# ... with 10 more variables: sigdm <dbl>, fjaer <dbl>, lichen <dbl>,  
#   veg <dbl>, pen <dbl>, stdp <dbl>, tueh <dbl>, vindf <dbl>, sti <dbl>,  
#   kratt <dbl>
```

```
symbols(x = habsel$x, y = habsel$y, circles = habsel$captures, inches = 0.1)
```



```
library(nlme)  
fit1<-gls(log(captures + 1) ~ lichen + veg + heather + moss + stdp,  
          data = habsel)  
plot(fit1)
```



```
summary(fit1)
```

Generalized least squares fit by REML

Model: $\log(\text{captures} + 1) \sim \text{lichen} + \text{veg} + \text{heather} + \text{moss} + \text{stdp}$

Data: habsel

AIC	BIC	logLik
187.449	202.6697	-86.72451

Coefficients:

	Value	Std.Error	t-value	p-value
(Intercept)	0.3197152	0.7333709	0.435953	0.6643
lichen	-0.4835256	0.2357549	-2.050967	0.0443
veg	0.0587848	0.0212540	2.765828	0.0074
heather	0.2895082	0.1654110	1.750236	0.0848
moss	-0.3127400	0.1446533	-2.161996	0.0343
stdp	0.1213571	0.0923399	1.314243	0.1934

Correlation:

	(Intr)	lichen	veg	heathr	moss
lichen	-0.413				
veg	-0.774	0.406			
heather	-0.131	0.029	0.173		
moss	-0.318	-0.131	-0.010	-0.165	

```
stdp      -0.514  0.185  0.120 -0.056 -0.217
```

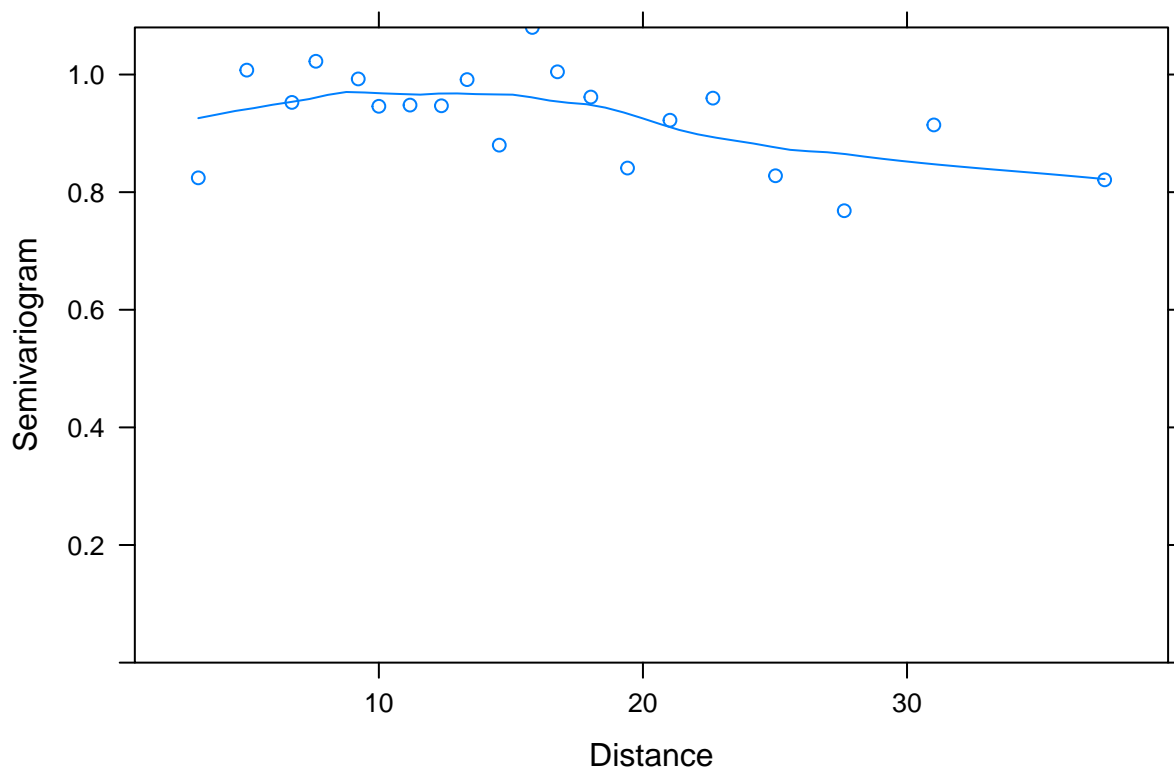
Standardized residuals:

	Min	Q1	Med	Q3	Max
	-1.85998486	-0.81284898	0.09348586	0.71497115	1.80374891

Residual standard error: 0.7598301

Degrees of freedom: 71 total; 65 residual

```
plot(Variogram(fit1,form=~x+y))
```



```
library(nlme)
fit3 <- gls(log(captures + 1) ~ lichen + veg + heather + moss + stdp,
            data = habsel,
            corr = corSpatial(form = ~x + y, type = "exponential", nugget = F),
            method = "REML")
summary(fit3)
```

Generalized least squares fit by REML

Model: $\log(\text{captures} + 1) \sim \text{lichen} + \text{veg} + \text{heather} + \text{moss} + \text{stdp}$

Data: habsel

AIC	BIC	logLik
183.9753	201.3704	-83.98764

Correlation Structure: Exponential spatial correlation

Formula: ~x + y

Parameter estimate(s):

range

2.233275

Coefficients:

	Value	Std.Error	t-value	p-value
(Intercept)	-0.0594373	0.7058155	-0.084211	0.9331
lichen	-0.3384882	0.2334688	-1.449822	0.1519
veg	0.0634851	0.0199702	3.178996	0.0023
heather	0.1437814	0.1640653	0.876367	0.3841
moss	-0.2770924	0.1357630	-2.041001	0.0453
stdp	0.1863430	0.0858756	2.169918	0.0337

Correlation:

	(Intr)	lichen	veg	heathr	moss
lichen	-0.450				
veg	-0.760	0.449			
heather	-0.164	0.142	0.244		
moss	-0.374	-0.052	0.048	-0.209	
stdp	-0.481	0.123	0.062	-0.100	-0.136

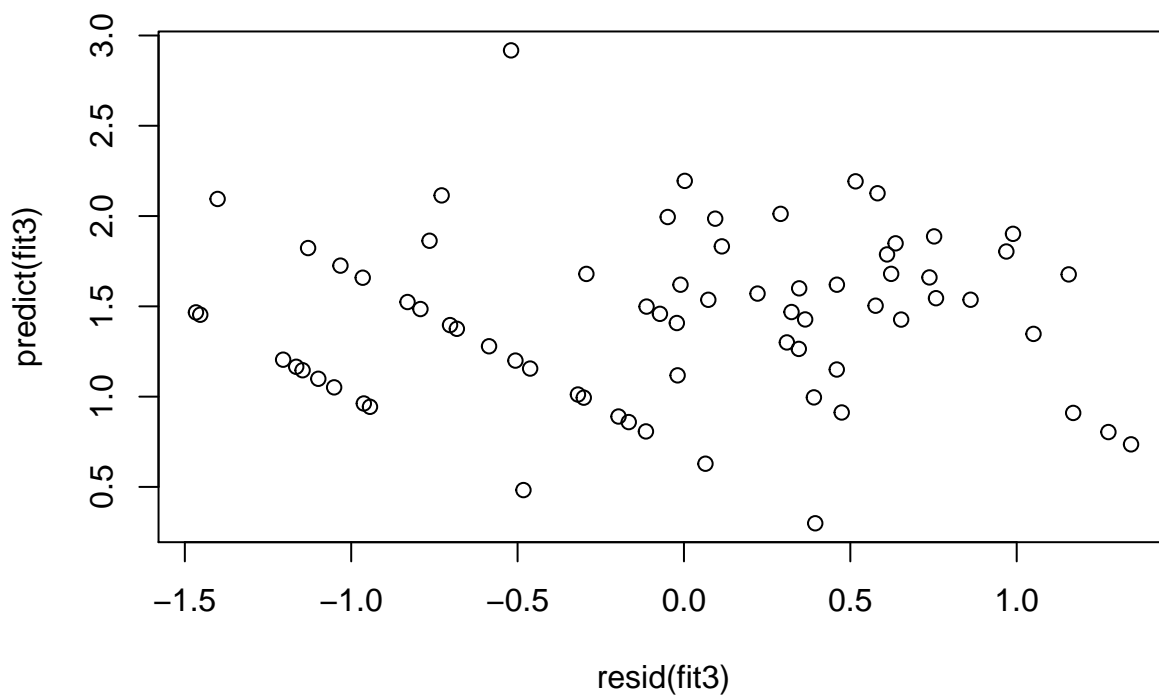
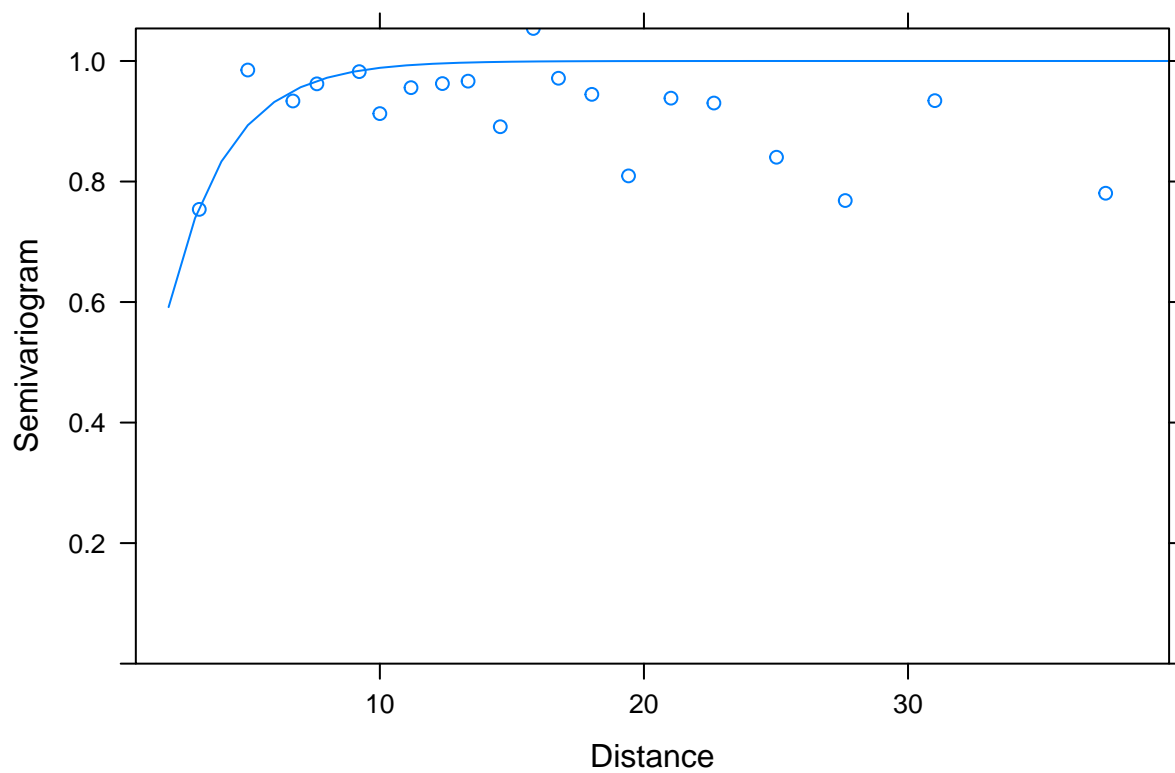
Standardized residuals:

	Min	Q1	Med	Q3	Max
	-1.88988670	-0.89270806	-0.01341163	0.63770014	1.73141952

Residual standard error: 0.7760408

Degrees of freedom: 71 total; 65 residual

```
plot(Variogram(fit3,form=~x+y))
```



```
fit4 <- gls(log(captures + 1) ~ lichen + veg + heather + moss + stdp,
            data = habssel,
            corr = corSpatial(form = ~x + y,
                              type = "exponential",
```

```

                                nugget = T),
                                method = "REML")
summary(fit4)

```

Generalized least squares fit by REML

Model: log(captures + 1) ~ lichen + veg + heather + moss + stdp

Data: habsel

	AIC	BIC	logLik
	185.9753	205.5448	-83.98764

Correlation Structure: Exponential spatial correlation

Formula: ~x + y

Parameter estimate(s):

	range	nugget
	2.233275e+00	6.647833e-09

Coefficients:

	Value	Std.Error	t-value	p-value
(Intercept)	-0.0594373	0.7058155	-0.084211	0.9331
lichen	-0.3384882	0.2334688	-1.449822	0.1519
veg	0.0634851	0.0199702	3.178996	0.0023
heather	0.1437814	0.1640653	0.876367	0.3841
moss	-0.2770924	0.1357630	-2.041001	0.0453
stdp	0.1863430	0.0858756	2.169919	0.0337

Correlation:

	(Intr)	lichen	veg	heathr	moss
lichen	-0.450				
veg	-0.760	0.449			
heather	-0.164	0.142	0.244		
moss	-0.374	-0.052	0.048	-0.209	
stdp	-0.481	0.123	0.062	-0.100	-0.136

Standardized residuals:

	Min	Q1	Med	Q3	Max
	-1.88988670	-0.89270806	-0.01341163	0.63770014	1.73141952

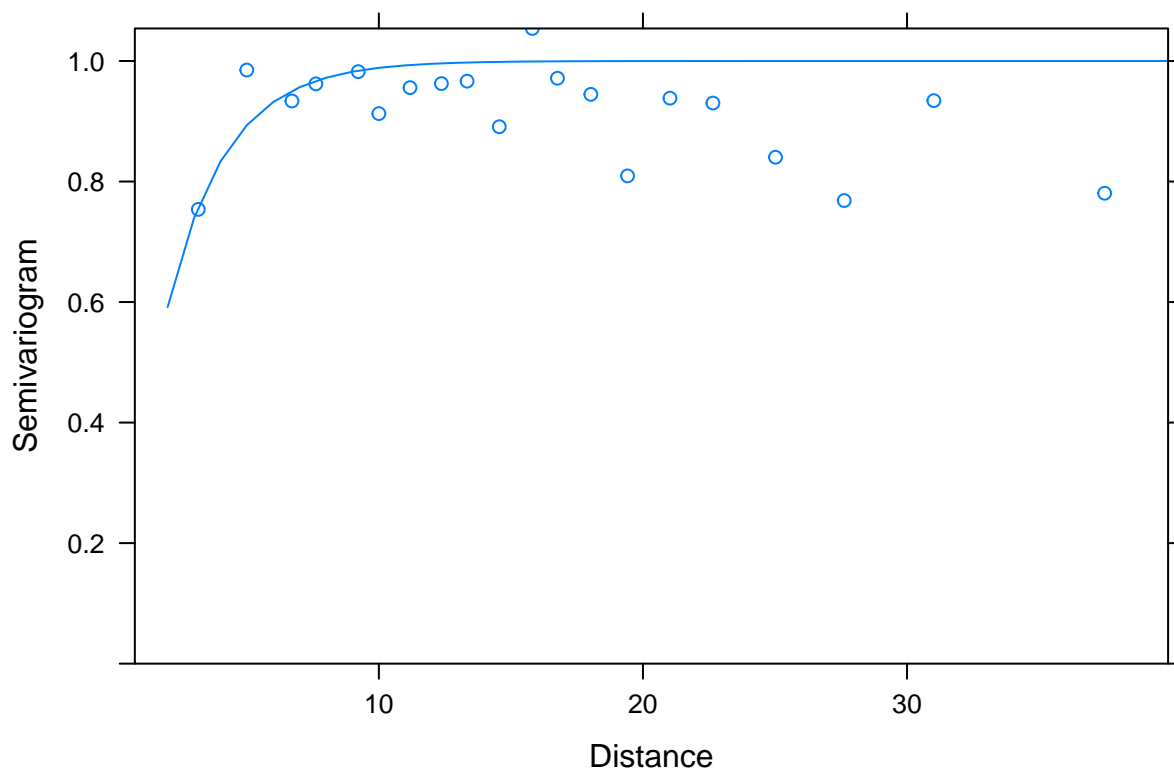
Residual standard error: 0.7760408

Degrees of freedom: 71 total; 65 residual

```

plot(Variogram(fit4,form=~x+y))

```



```
anova(fit1, fit3, fit4)
```

	Model	df	AIC	BIC	logLik	Test	L.Ratio	p-value
fit1	1	7	187.4490	202.6697	-86.72451			
fit3	2	8	183.9753	201.3704	-83.98764	1 vs 2	5.473738	0.0193
fit4	3	9	185.9753	205.5448	-83.98764	2 vs 3	0.000000	0.9999

Una vez tenemos un modelo aceptable de estructura de varianzas y correlaciones, podemos empezar un proceso de selección de variables independientes, por ejemplo, “backwards”.

Para ello debemos recordar SIEMPRE que si comparamos modelos cuyas partes fijas cambien, debemos hacerlo ajustando por máxima verosimilitud (ML), no REML.

```
fit5 <- gls(log(captures + 1) ~ lichen + veg + heather + moss + stdp,
            data = habsel,
            corr = corSpatial(form = ~x + y, type = "exponential",
                              nugget = F),
            method = "ML")
summary(fit5)
```

```
fit6 <- gls(log(captures + 1) ~ lichen + veg + moss + stdp,
            data = habsel,
            corr = corSpatial(form = ~x + y, type = "exponential",
                              nugget = F),
```

```
        method = "ML")  
summary(fit6)  
  
plot(fit6)
```


Capítulo 4

Introducción a los Modelos No Lineales Mixtos. Modelos Semiparamétricos y GAMM

4.1. Modelos Estadísticos

Los modelos estadísticos generalmente relacionan la esperanza de una variable (“respuesta”) a una o más variables independientes (“regresoras”):

$$E(Y) = f(x_1, \dots, x_k; \boldsymbol{\beta}).$$

Además se pueden hacer supuestos acerca de la distribución de Y . Por ejemplo:

$$Y \sim N(\beta_0 + \beta_1 x, \sigma^2).$$

4.2. Modelos No Lineales

Si los parámetros del modelo entran en la ecuación en forma no lineal, entonces tenemos un modelo no lineal.

Ejercicio 4.1 (¿Lineal o No Lineal?). Determine si los siguientes modelos son lineales o no-lineales.

1. $E(Y) = \beta_0 + \beta_1 x_1 + \dots + \beta_k x_k.$
2. $E(Y) = \frac{1}{\beta_0 + \beta_1 x_1 + \dots + \beta_k x_k}.$
3. $E(Y) = \beta_0 + \beta_1 x_1 + \beta_2 x_2^2 + \dots + \beta_k x_k^k.$

4. $E(Y) = \beta_0 \exp(-\exp(\beta_1 x_1)).$

5. $E(Y) = \frac{\beta_0}{[1 + e^{-(\beta_1 + \beta_2 x)}]^{\beta_3}}.$

4.2.1. Aplicaciones de modelos no lineales

- Muchas aplicaciones en biología, medicina, química, agricultura.
- En muchos casos surgen a partir de mecanismos físicos, químicos o biológicos conocidos (usando por ejemplo, ecuaciones diferenciales).
- Permiten modelar datos reales en forma “natural” con mucha flexibilidad (por ejemplo, con asíntotas, valores positivos de las Y , un único valor máximo).
- Muchos parámetros tienen interpretación práctica (tasas de degradabilidad, crecimiento máximo, velocidad máxima de absorción, etc.)

4.2.2. Problemas con los modelos no lineales

- Se deben usar métodos iterativos para estimar los parámetros por mínimos cuadrados, máxima verosimilitud, etc.
- Es común encontrar problemas de convergencia en los métodos iterativos.
- Distintas parametrizaciones pueden afectar no solo la interpretación sino también las propiedades de los estimadores (por ej., una parametrización puede ser muy interesante desde el punto de vista de su interpretación práctica, pero muy mala para lograr convergencia de estimadores).

4.3. Familias de funciones no lineales: curvas de crecimiento

- **Exponencial:**

$$\frac{\partial E(Y)}{\partial t} = \gamma E(Y); \quad E(Y) = \alpha \exp(\gamma t).$$

- **Monomolecular:**

$$\frac{\partial E(Y)}{\partial t} = \gamma(\beta - Y); \quad E(Y) = \beta - \alpha \exp(-\gamma t).$$

- **Logístico:**

$$\frac{\partial E(Y)}{\partial t} = \gamma E(Y)[\beta - E(Y)]; \quad E(Y) = \frac{\beta}{1 + \exp(-\alpha + \gamma t)}.$$

- **Gompertz:**

$$\frac{\partial E(Y)}{\partial t} = \gamma E(Y)[\log \beta - \log E(Y)]; \quad E(Y) = \beta \exp[-\alpha \exp(-\gamma t)].$$

- **Richards:**

$$\frac{\partial E(Y)}{\partial t} = -\frac{\gamma}{\kappa} \gamma E(Y) \left[\left(\frac{E(Y)}{\beta} \right)^\kappa - 1 \right], \quad E(Y) = \beta [1 + \kappa \exp(-\gamma(t - \xi))]^{-1/\kappa}.$$

4.4. Ejemplos de otras familias de funciones no lineales

- **Modelo de un compartimiento:**

$$E(Y) = \frac{dk_a}{V(k_a - k_e)} [\exp(-k_e t) - \exp(-k_a t)].$$

- **Modelo de difusión Browniana en dos dimensiones:**

$$E(Y) = \frac{\mu_0}{4\pi\delta t} \exp\left(-\frac{x^2 + y^2}{4\delta t}\right).$$

- **Modelo lineal con puntos de cambio:**

$$E(Y) = \beta_0 + \beta_1 x I(x \leq \alpha) + [\beta_1 \alpha + \beta_2 (x - \alpha)] I(x > \alpha).$$

- **Modelo polinomial recíproco (Holliday):**

$$E(Y) = (\alpha + \beta x + \gamma x^2)^{-1}.$$

4.5. Ejemplos de Modelos No Lineales Mixtos

4.5.1. Ejemplo 1: Desarrollo de una droga

- Durante el desarrollo de una droga nueva, es importante conocer cómo la droga se absorbe en la sangre a medida que pasa el tiempo.

- Para ello se administra una cantidad conocida de la droga, y se mide la concentración de la misma en la sangre a intervalos durante varias horas.
- En este caso se suelen ajustar modelos no lineales (p.ej., de un compartimiento) a los datos de cada individuo.
- Nos interesa saber cómo los parámetros de estos modelos dependen de ciertas covariables fijas (por ejemplo, edad, si el paciente está sano o enfermo, peso, etc.) y de efectos aleatorios de los pacientes.

4.5.2. Ejemplo 2: Ecología en Lagos

- Elegimos ocho lagos aleatoriamente, y en cada lago tomamos veinte muestras de agua (en distintas partes del lago) para estudiar la relación entre la densidad de un cierto microorganismo y la concentración de nitratos.
- Esta relación se modela mediante una curva exponencial cuyos parámetros dependerán del lago individual (efecto aleatorio).
- También puede depender de covariables que varíen:
 - solamente con los lagos (p.ej., superficie),
 - con las distintas observaciones en el mismo lago (p.ej., estación del año, nivel de eutroficación).

El primer ejemplo representan un caso de medidas repetidas, mientras que el segundo ejemplo es un caso de datos agrupados (muestreo en dos etapas). En ambos casos no hay independencia entre observaciones tomadas en el mismo sujeto (árbol, sujeto o lago). En general, habrá correlación positiva entre estas observaciones.

4.6. Definición del Modelo

Vamos a definir al vector de observaciones del i -ésimo sujeto (árbol, paciente, lago) como

$$\mathbf{y}_i = (y_{i1}, \dots, y_{in})^t.$$

Las covariables que no varían en el mismo sujeto se denotarán con el vector \mathbf{a}_i y las que varían con las distintas observaciones del sujeto se denotarán con el vector $\mathbf{x}_i = (t_{ij}, \mathbf{u}_i)$.

Podemos pensar en el modelo no lineal mixto como un modelo jerárquico con dos niveles:

- **Nivel 1 - Modelo Individual:** Este nivel del modelo explica las curvas individuales.

$$y_{ij} = f(x_{ij}, \beta_i) + e_{ij}, \quad j = 1, \dots, n_i.$$

- **Nivel 2 - Modelo Poblacional:** Este nivel del modelo explica cómo los elementos de β_i varían entre los individuos, debido a las covariables, los p parámetros β y a los efectos aleatorios \mathbf{b}_i .

$$\beta_i = d(\mathbf{a}_i, \beta, \mathbf{b}_i), \quad i = 1, \dots, m.$$

El caso más común del modelo poblacional es el del modelo lineal:

$$\beta_i = \mathbf{A}_i \beta + \mathbf{B}_i \mathbf{b}_i.$$

La matriz \mathbf{A} típicamente depende de las covariables \mathbf{a}_i , mientras que la matriz \mathbf{B} contiene solamente 0 y 1, aunque también podría depender de las covariables.

4.6.1. Resumen del modelo no lineal mixto

- Vamos a denotar $\mathbf{f}_i = \{f(x_{i1}, \beta_i), \dots, f(x_{in_i}, \beta_i)\}^t$.
- Las covariables serán $\mathbf{z}_i = (\mathbf{u}_i^t, \mathbf{a}_i^t)^t$.
- Modelo a nivel individual:

$$E(\mathbf{y}_i | \mathbf{z}_i, \mathbf{b}_i) = \mathbf{f}_i(\mathbf{u}_i, \beta_i) = \mathbf{f}_i(\mathbf{z}_i, \beta, \mathbf{b}_i).$$

$$Var(\mathbf{y}_i | \mathbf{z}_i, \mathbf{b}_i) = \mathbf{R}_i(\mathbf{u}_i, \beta_i, \xi) = \mathbf{R}_i(\mathbf{z}_i, \beta, \mathbf{b}_i, \xi).$$

- Modelo a nivel individual:

$$\beta_i = d(\mathbf{a}_i, \beta, \mathbf{b}_i), \quad \mathbf{b}_i = (\mathbf{0}, D).$$

4.6.2. Modelo marginal inducido

Si integramos a través de la distribución de los efectos aleatorios $F_{\mathbf{b}}(\mathbf{b}_i)$ (“promediamos” los efectos aleatorios), obtenemos el modelo marginal inducido por el modelo mixto:

$$E(\mathbf{y}_i | \mathbf{z}_i) = \int \mathbf{f}_i(\mathbf{z}_i, \beta, \mathbf{b}_i) dF_{\mathbf{b}}(\mathbf{b}_i)$$

$$Var(\mathbf{y}_i | \mathbf{z}_i) = E_{\mathbf{b}}(\mathbf{R}_i(\mathbf{z}_i, \beta, \mathbf{b}_i, \xi) | \mathbf{z}_i) + Var_{\mathbf{b}}(\mathbf{f}_i(\mathbf{z}_i, \beta, \mathbf{b}_i) | \mathbf{z}_i)$$

Aunque \mathbf{R} sea diagonal, $\text{var}(\mathbf{y}|\mathbf{z})$ no lo es, dado que los efectos aleatorios \mathbf{b} inducen correlaciones entre observaciones del mismo sujeto.

En general, el segundo término de $\text{var}(\mathbf{y}|\mathbf{z})$ es el más importante, por lo que es común en aplicaciones usar \mathbf{R} diagonal (es decir, suponer que, condicionado a los efectos aleatorios, las observaciones no están correlacionadas).

Esta estrategia es la sugerida por la mayoría de los autores en los libros de texto actuales (Diggle et al., Verbeke y Molenberghs, Davidian y Giltinan).

4.7. Objetivos de la inferencia

- Estimar valores típicos de los parámetros (efectos aleatorios = 0, el centro de su distribución) de \mathbf{f} .
- Estudiar cómo los parámetros varían en la población.
- Estudiar si esta variación está relacionada con características de los individuos (covariables, etc.).
- Estos objetivos se logran mediante inferencias acerca de los parámetros β y \mathbf{D} .
- Los componentes de β describen tanto valores típicos como la relación entre la respuesta y las covariables, manteniendo constante el efecto del sujeto.
- Como en muchos modelos de regresión, el objetivo también puede ser identificar una forma funcional parsimoniosa para describir la relación entre las observaciones y las covariables.
- Además puede ser de interés caracterizar a los sujetos (“predecir los efectos aleatorios”). En el ejemplo de los lagos, nos interesa caracterizar a los sitios según su potencial.

4.8. Inferencia sujeto específica o promedio poblacional

Los modelos no lineales mixtos son sujeto específicos (SS): los valores de β relacionan las observaciones con una covariable para un sujeto dado, no para el promedio de la población.

Por otra parte, los parámetros de un modelo promedio poblacional (PA) representan la relación entre el promedio marginal, $E(\mathbf{y}|\mathbf{z})$, con una covariable.

Si los modelos fuesen lineales, entonces los parámetros β pueden interpretarse tanto desde el punto de vista SS como PA:

$$E(\mathbf{y}_i|\mathbf{z}_i) = \int (\mathbf{X}_i\beta + \mathbf{Z}_i\mathbf{b}_i) dF_{\mathbf{b}}(\mathbf{b}_i) = \mathbf{X}_i\beta.$$

En modelos no lineales, los parámetros β de un modelo SS no pueden interpretarse como PA:

$$E(\mathbf{y}_i|\mathbf{z}_i) = \int \mathbf{f}_i(\mathbf{z}_i, \beta, \mathbf{b}_i) dF_{\mathbf{b}}(\mathbf{b}_i) \neq \mathbf{X}_i\beta.$$

Esta integral no se puede resolver analíticamente en general (por ejemplo, no tiene solución analítica para efectos aleatorios normales en un modelo no lineal). Por otro lado, si modelamos directamente la esperanza marginal (PA), entonces no podríamos interpretar sus parámetros como SS.

En modelos lineales generalizados, la no linealidad surge naturalmente como descripción empírica de datos. Para datos agrupados (longitudinales o “clustered”), la correlación se puede modelar explícitamente en un modelo PA o implícitamente mediante la introducción de efectos aleatorios en el modelo (SS).

Para modelos no lineales también se podrían usar ambas estrategias, aunque es más común el enfoque SS. Lee y Nelder (2004) sugieren que el enfoque SS es más “rico”, ya que a partir de un modelo SS es posible estudiar una relación marginal (PA), pero no es posible estudiar una relación SS a partir de un modelo PA.

4.9. Inferencia Basada en Verosimilitud

En un modelo no lineal con efectos aleatorios, la densidad conjunta es:

$$p(\mathbf{y}|\mathbf{x}, \beta, \gamma, \mathbf{D}) = \prod_{i=1}^m p(\mathbf{y}_i|\mathbf{x}_i, \beta, \gamma, \mathbf{D})$$

donde la densidad marginal de $\mathbf{Y}_i|\mathbf{x}_i$ toma la siguiente forma:

$$p(\mathbf{y}_i|\mathbf{x}_i, \beta, \gamma, \mathbf{D}) = \int p(\mathbf{y}_i|\mathbf{b}_i, \mathbf{z}_i, \mathbf{a}_i; \beta, \gamma) p(\mathbf{b}_i|\mathbf{D}) d\mathbf{b}_i.$$

Los efectos aleatorios de distintos sujetos se asumen i.i.d. (independientes e idénticamente distribuidos).

4.10. Estimación por ML (*Maximum Likelihood*)

- El proceso de estimación requiere evaluar m integrales de dimensión k en cada paso de optimización.
- Las integrales resultantes no tienen una solución analítica y se deben resolver usando aproximaciones numéricas.
- Aún cuando se use un método para aproximar la integral, por ejemplo, cuadratura Gaussiana, si el número de efectos aleatorios es muy grande, el problema computacional aumenta notablemente.
- Otra posible alternativa es aproximar las integrales usando técnicas de Monte-Carlo. Específicamente, la integral se puede considerar como el valor esperado de la función asumiendo una función de probabilidad. Esta alternativa requiere tamaños grandes de muestra y por lo tanto, un gran número de cálculos computacionales en el proceso de optimización para estimar los parámetros del modelo.

4.11. Métodos de estimación

- **Métodos basados en estimaciones individuales:** consideran los parámetros sujeto-específicos relacionados con el modelo poblacional como observaciones y luego ajustan un modelo para estas nuevas observaciones.
 - Si el número de observaciones por individuo ni es suficientemente grande, se puede resumir la respuesta de cada individuo \mathbf{y}_i en las estimaciones $\hat{\beta}_i$, y por medio de éstas hallar las estimaciones de β y D . Tomando los momentos condicionales como una función de β_i :

$$E(\mathbf{y}_i | \mathbf{z}_i, \beta_i) = \mathbf{f}_i(\mathbf{z}_i, \beta_i),$$

$$Var(\mathbf{y}_i | \mathbf{z}_i, \beta_i) = \mathbf{R}_i(\mathbf{z}_i, \beta_i, \gamma).$$

- El siguiente paso es ajustar un modelo para cada individuo con base en los dos momentos anteriores. Si la matriz $\mathbf{R}_i(\mathbf{z}_i, \beta_i, \gamma)$ tiene la forma $\sigma^2 I_{n_i}$, entonces se puede usar el método de mínimos cuadrados ordinarios para modelos no lineales. En otro caso, usando resultados asintóticos.
- **Métodos basados en aproximaciones lineales:** realizan aproximaciones de los datos, llamados pseudos-datos, usando expansión en serie de Taylor alrededor de una estimación razonable de \mathbf{b}_i .
 - En este grupo de métodos de estimación, el objetivo es aproximar las integrales implícitas en la función de verosimilitud mediante expresiones cerradas.

- Esto implica que la función objetivo a ser optimizada tiene una forma cerrada y por lo tanto, las ecuaciones de estimación se generan a partir de las derivadas del logaritmo de la verosimilitud aproximada.
- **Desventaja:** no se pueden aplicar métodos o criterios basados en la verosimilitud.
- Una forma de llevar a cabo la inferencia es usando linearización de primer orden. Se trabaja con el primer nivel del modelo, que involucra la parte individual:

$$E(\mathbf{y}_i | \mathbf{z}_i, \boldsymbol{\beta}_i) = \mathbf{f}_i(\mathbf{z}_i, \boldsymbol{\beta}_i)$$

$$Var(\mathbf{y}_i | \mathbf{z}_i, \boldsymbol{\beta}_i) = \mathbf{R}_i(\mathbf{z}_i, \boldsymbol{\beta}_i, \boldsymbol{\gamma})$$

- Asumiendo que \mathbf{R}_i es una matriz definida positiva, y usando su descomposición de Cholesky, se puede definir los pseudo-datos usando una expansión alrededor de $E(\boldsymbol{\beta}_i) = \mathbf{0}$.
 - En este caso, los pseudos-datos quedan definidos por los residuales más los efectos fijos (residuales+fijos).
 - Esta aproximación también se usa en la estimación de parámetros de un modelo lineal generalizado mixto mediante cuasi-verosimilitud marginal.
 - Bajo los supuestos de normalidad en los efectos aleatorios, las integrales de la función de verosimilitud se pueden aproximar con densidades normales y dada que $Var(Y)$ depende de $\boldsymbol{\beta}$, las derivadas del logaritmo de la función de verosimilitud conllevan a ecuaciones cuadráticas de estimación para $\boldsymbol{\beta}$ y \mathbf{D} (GEE2).
- La aproximación de las integrales en la función de verosimilitud considerando $\mathbf{b} = \mathbf{0}$ claramente es una aproximación no muy buena de su verdadero valor. Lindstrom y Bates propusieron hacer una aproximación usando la expansión en series de Taylor pero alrededor de un valor razonable de \mathbf{b}_i , llamémoslo \mathbf{b}_i^* , de tal manera que se definen nuevos pseudos-datos.
 - En este escenario, los pseudos-datos quedan definidos por los residuales, los efectos fijos y los aleatorios (residuales+fijos+aleatorios).
 - Al igual que con la aproximación anterior, se pueden deducir las ecuaciones de estimación (GEE1 o GEE2) para obtener las estimaciones de $\boldsymbol{\beta}$ y \mathbf{D} .
 - Un valor razonable para es el estimador empírico de Bayes (EBLUP).
 - Este tipo de aproximación es usada en la macro NLINMIX de SAS y la función nlme() de R.
- Estas rutinas asumen que la distribución condicional de Y_i es normal.

- Sin embargo, las estimaciones de \mathbf{b}_i en el proceso de iteración difieren entre NLINMIX y `nlme()`.
- Ambas aproximaciones ($\mathbf{b} = 0$ y $\mathbf{b} = \mathbf{b}_i^*$) también se usan en modelos lineales generalizados mixtos (por ejemplo, en `Proc GLIMMIX`).
- *Métodos condicionales de primer orden: aproximación de Laplace:* La aproximación mediante la expansión alrededor de un estimador de \mathbf{b}_i^* también puede derivarse usando la aproximación de Laplace para integrales que involucran densidades normales. La idea es aproximar la integral usando una aproximación cuadrática alrededor del punto en el cual el integrando toma su máximo (Demidenko, 2004).
 - Esta aproximación es válida cuando el tamaño de muestra n es grande. En teoría se necesita que tanto n_i como m sean grandes.
 - Este método es uno de los más usados en la actualidad. Esta aproximación se encuentra implementada en el procedimiento NLMIXED de SAS usando la opción `Method=FIR0` y `QPOINTS=1`, lo cual indica una aproximación de primer orden con un punto de cuadratura y en la función `nlme()` de R.
- **Métodos basados en aproximaciones numéricas de las integrales**, p.ej., cuadratura de Gauss-Hermite. Estos métodos se suelen llamar “exactos” ya que consideran la verosimilitud completa de los datos originales a pesar de su aproximación numérica.
 - Métodos basados sobre la verosimilitud “exacta”: Los métodos basados en la verosimilitud “exacta” han tenido un auge importante en los últimos años debido a los avances computacionales.
 - En estos métodos el objetivo es aproximar las integrales mediante métodos numéricos.
 - La idea es aprovechar la relación entre las densidades normales y la cuadratura de Gauss-Hermite para aproximar integrales de la forma:

$$\int f(x) e^{-x^2/2} dx.$$

- Consideremos una densidad normal $\phi(\mu, \sigma^2)$

$$\int f(x) \phi(\mu, \sigma^2) dx \approx \sum_{j=1}^Q w_j^* f(x_j),$$

siendo $w_j^* = w_j / \sqrt{\pi}$ y $x_j = \mu + \sqrt{2}\sigma q_j$ las abscisas. w_j^* y q_j son los Q pesos y puntos de cuadratura de Gauss-Hermite. Los puntos de cuadratura q_j están centrados alrededor de cero (cuadratura convencional).

- En la cuadratura adaptativa de Gauss-Hermite, los puntos de cuadratura no están alrededor de 0, sino que se seleccionan en una región apropiada del integrando.
- Esta aproximación con la cuadratura adaptativa con $Q = 1$ corresponde a la aproximación de Laplace.
- **Métodos Bayesianos:** aproximan las integrales usando integración vía Monte Carlo y consideran otro nivel adicional del modelo donde se especifican las distribuciones a priori de los parámetros.
 - Desde el punto de vista Bayesiano, los parámetros son tratados como vectores aleatorios.
 - Una vez se establecen las densidades a priori de $(\beta, \gamma, \mathbf{D})$, se procede a hallar la densidad a posteriori para $(\mathbf{b}_i, \beta, \gamma, \mathbf{D})$.
 - Luego, se usa MCMC para generar muestras de las distribuciones a posteriori para aproximar valores específicos de las distribuciones, tales como la moda. Este enfoque está completamente implementado en programas como WinBUGS o JAGS.

4.12. Ejemplo de modelos no lineales: Teofilina

Ejemplo 4.1 (Teofilina (Pinheiro y Bates)). Los siguientes datos representan concentración de teofilina (mg/l) luego de una dosis oral de la droga a 12 pacientes. Para esta sustancia se puede usar un modelo de un compartimiento, parametrizado de la siguiente manera:

$$E(Y) = \frac{Dk_a k_e}{Cl(k_a - k_e)} [\exp(-k_e t) - \exp(-k_a t)].$$

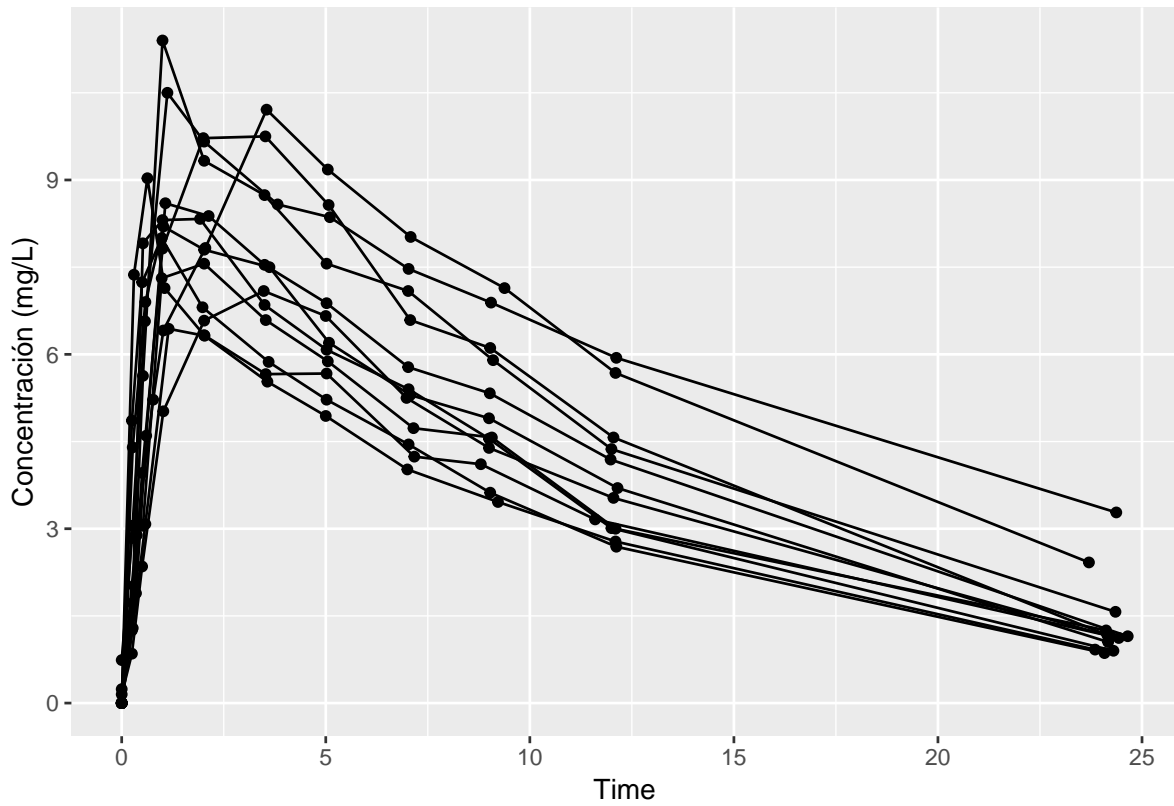
$C(t)$ es la concentración de la droga en el tiempo t luego de una dosis D dada en $t = 0$. k_a es la tasa de absorción, k_e es la tasa de eliminación, y Cl se denomina *clearance*, el volumen de sangre con el que la droga se elimina.

```
head(Teofilina)
```

```
# A tibble: 6 x 5
  Subject    Wt  Dose  Time  conc
  <int> <dbl> <dbl> <dbl> <dbl>
1       1  79.6  4.02  0.    0.740
2       1  79.6  4.02  0.250  2.84
3       1  79.6  4.02  0.570  6.57
4       1  79.6  4.02  1.12  10.5
5       1  79.6  4.02  2.02  9.66
```

```
6      1  79.6  4.02 3.82   8.58
```

```
library(ggplot2)
ggplot(Teofilina, aes(Time, conc, group=Subject)) + geom_point() +
  geom_line()+ylab("Concentración (mg/L)")
```



Modelo No Lineal Mixto:

$$y_{ij} = \frac{Dk_{ai}k_{ei}}{Cl_i(k_{ai} - k_{ei})} [\exp(-k_{ei}t) - \exp(-k_{ai}t)] + e_{ij},$$

$$Cl_i = \exp(\beta_1 + b_{1i}),$$

$$k_{ai} = \exp(\beta_2 + b_{2i}),$$

$$k_{ei} = \exp(\beta_3 + b_{3i}),$$

$$e_{ij} \stackrel{iid}{\sim} N(0, \sigma^2), \quad (b_{1i}, b_{2i}, b_{3i})^t \stackrel{iid}{\sim} N(\mathbf{0}, \mathbf{D}).$$

Los objetivos del análisis son:

- Determinar valores promedio de (k_a, k_e, Cl) .
- Ver cómo estos parámetros varían en la población de sujetos.

- Si la variación entre sujetos es grande, será difícil determinar una dosis general óptima para todos, sino que deberá estudiarse en más detalle en subgrupos de sujetos.

4.12.1. Estrategia de modelación

1. Comenzar ajustando el modelo con los efectos fijos y los parámetros de las distribuciones de efectos aleatorios.
2. Seleccionar efectos aleatorios importantes para simplificar la estructura.
3. Usar el modelo final para inferencia.

```
library(nlme)
modelo1 <- nls(conc~Dose*exp(lke+lka-lcl)/(exp(lka)-exp(lke))*(exp(-exp(lke)*Time)-exp(-exp(lka)*Time)),
               data=Teofilina,
               start=list(lke=-2.5, lka=0.5, lcl=-3))

modelo2 <- nlme(conc~Dose*exp(lke+lka-lcl)/(exp(lka)-exp(lke))*(exp(-exp(lke)*Time)-exp(-exp(lka)*Time)),
                data=Teofilina,
                fixed=list(lke~1, lka~1, lcl~1),
                random=list(lke~1), groups=~Subject,
                start=c(-2.5, 0.5, -3),
                method="REML")

modelo3 <- nlme(conc~Dose*exp(lke+lka-lcl)/(exp(lka)-exp(lke))*(exp(-exp(lke)*Time)-exp(-exp(lka)*Time)),
                data=Teofilina, fixed=list(lke~1, lka~1, lcl~1),
                random=list(lke~1, lka~1), groups=~Subject,
                start=c(-2.5, 0.5, -3),
                method="REML")

modelo4 <- nlme(conc~Dose*exp(lke+lka-lcl)/(exp(lka)-exp(lke))*(exp(-exp(lke)*Time)-exp(-exp(lka)*Time)),
                data=Teofilina, fixed=list(lke~1, lka~1, lcl~1),
                random=pdDiag(list(lke~1, lka~1, lcl~1)),
                groups=~Subject,
                start=c(-2.5, 0.5, -3),
                method="REML")
```

4.13. Modelos Semiparamétricos

En muchos problemas de regresión, el interés es interpretar pendientes, o efectos asociados a una o más variables independientes, factores, etc. En otros problemas el interés es predecir una variable o comparar curvas bajo condiciones distintas. En estos casos es

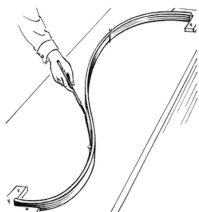


Figura 4.1: Splines

posible formular un modelo no paramétrico o semi-paramétrico, sin especificar la forma funcional de la relación entre la media y los predictors. Por ejemplo, si solo suponemos que f es una función suave y continua,

$$E(y_i) = f(x_i), \quad y_i \stackrel{i.d.}{\sim} \text{dist}(\beta, \xi).$$

4.13.1. Diferentes Suavizados

Hay diferentes estrategias para ajustar una curva suave y continua a los datos. La idea es obtener una curva que ajuste los datos bien, pero que no pase “demasiado” cerca de todos los puntos. Esto se logra mediante la optimización de algún criterio (por ejemplo, mínimos cuadrados) y el uso de un parámetro de suavizado (o penalización: cuando menos penalidad le demos al criterio que estamos usando menos suave será la curva estimada).

- Splines
- Estimación Kernel
- Polinomios locales (*loess*).
- Otros métodos: wavelets, vecino más cercano, medianas móviles, polinomios truncados, etc.

4.13.2. Splines

Un *spline* es una función polinomial por tramos que se unen suavemente en nodos predeterminados. El spline cúbico (el más comúnmente usado) es una curva suave cúbica por tramos. Los valores de cada tramo, y sus primera y segunda derivadas (pendiente y curvatura) coinciden en los nodos.

Supongamos que tenemos n nodos $k_1 \leq k_2 \leq \dots \leq k_n$. Un spline de grado $d \geq 0$ es la función con $d - 1$ derivadas continuas

$$S(x) = \begin{cases} P_0(x), & x < k_1 \\ P_i(x), & k_i \leq x < k_{i+1} \\ P_n(x), & x \geq k_n, \end{cases}$$

donde cada $P_i(x)$ es un polinomio de grado d .

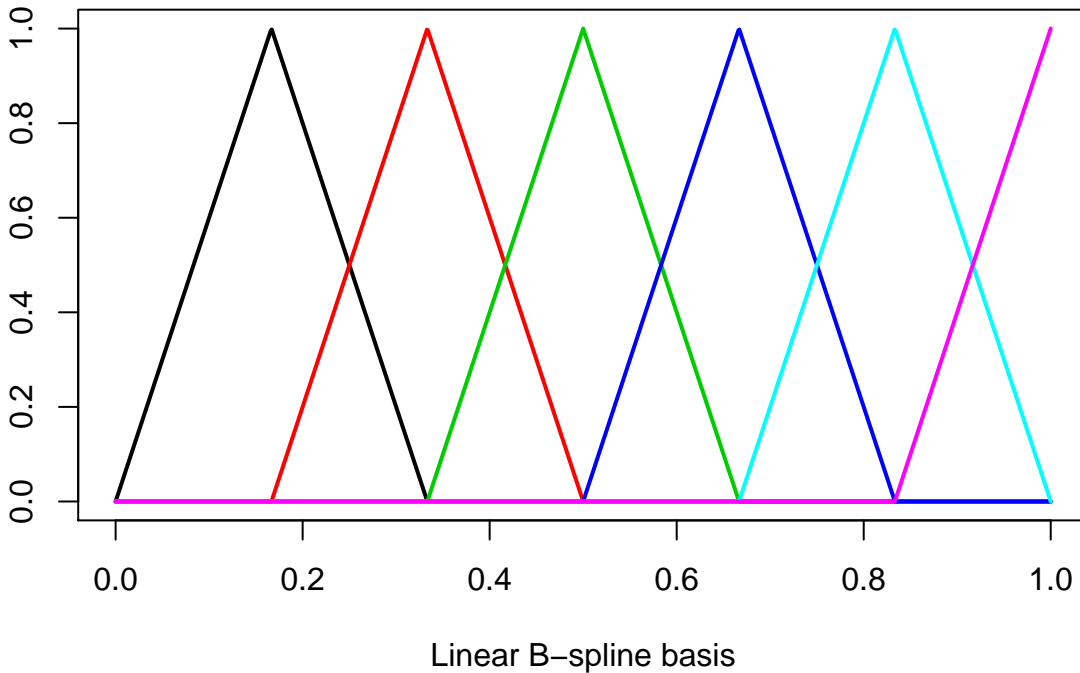
4.13.3. B-Splines

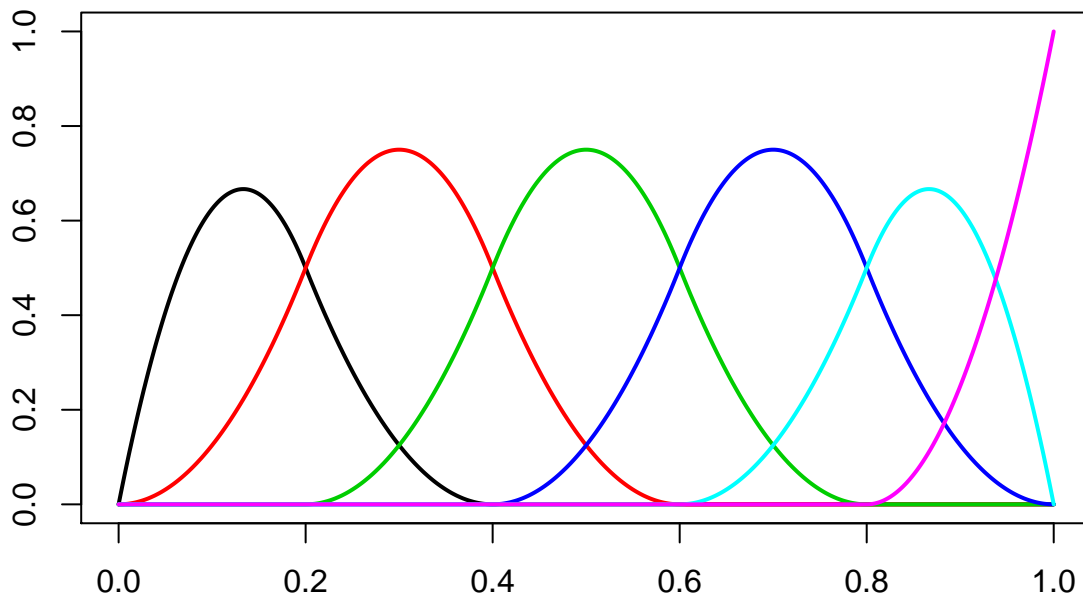
Si tenemos n nodos hay $n + 1$ polinomios de grado d , por lo que hay $(n + 1)(d + 1)$ coeficientes. Las restricciones de iguales valores y derivadas agregan d restricciones lineales en cada uno de los n nodos, por lo que el número de parámetros desconocidos es $n + d + 1$.

Otra manera de pensarlo es que la dimensión del espacio vectorial de splines de grado d en n nodos diferentes es $n + d + 1$. Es decir, si conocemos $n + d + 1$ vectores linealmente independientes que generen este espacio vectorial, podemos ajustar un spline a nuestros datos realizando una regresión lineal usando estos vectores base como columnas de nuestra matriz X .

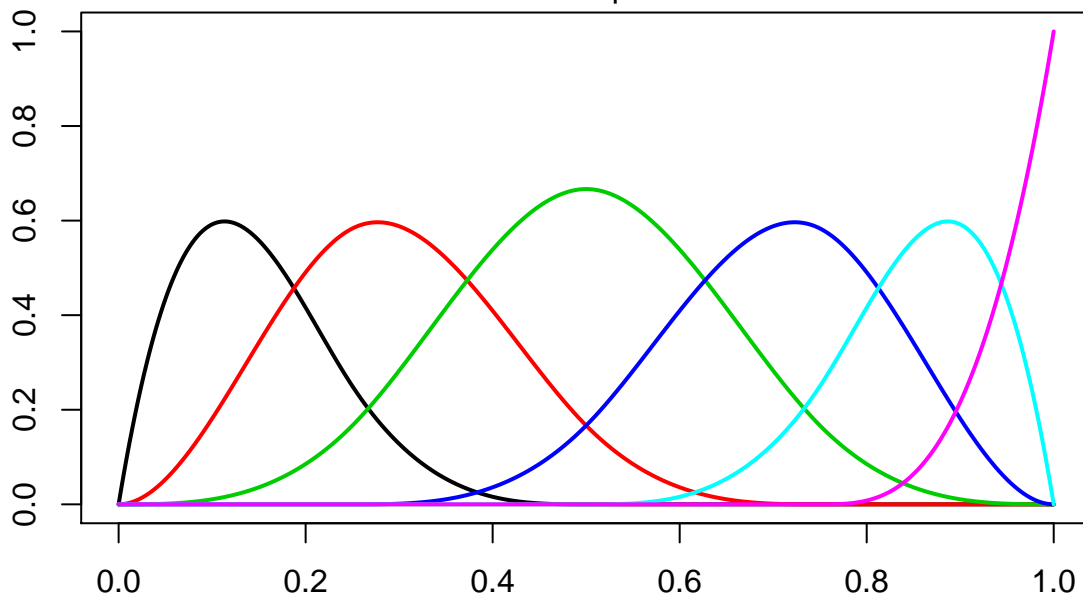
Por lo tanto, podemos ajustar splines usando toda la metodología de modelos lineales generalizados mixtos. Lo único “nuevo” en este proceso es generar una base para el espacio vectorial de splines. Hay varias alternativas (por ejemplo, B-splines y splines truncados).

Para elegir el número de nodos se puede usar AIC, BIC, etc.





Quadratic B-spline basis



Cubic B-spline basis

4.13.4. P-Splines

Otra alternativa es usar un factor de penalización, que define cuán suave uno quiere la curva. Para ello se optimiza una función objetivo y se le agrega una penalización, que será más grande a medida que la curva sea menos suave. Por ejemplo, para datos normales, a la optimización por máxima verosimilitud o mínimos cuadrados se le agrega

un término de penalización de forma que se busca minimizar

$$\frac{1}{n} \sum_{i=1}^n (Y_i - f(x_i))^2 + \lambda \int [f''(x)]^2 dx.$$

λ representa el parámetro de suavizado, y la integral es el factor de penalización (se conoce el *tuning parameter*). Cuando f es suave, la integral es cercana a 0, y cuando es muy “aserrada”, la integral es grande. Si λ es grande, el efecto de la penalidad es grande, y la solución será una curva suave. Si λ es pequeño, el efecto de la penalización es pequeño y la curva ajustada será poco suave (cuanto más nodos se usen, menos suave será la curva).

Las ecuaciones para resolver esto son similares a las ecuaciones de modelos mixtos.

4.13.5. Modelos aditivos, aditivos generalizados y mixtos

Los modelos aditivos permiten incorporar de manera suave efectos de distintas variables independientes:

$$y = \beta_0 + \sum_{j=1}^p f_j(X_j) + \varepsilon,$$

donde cada f_j es una función suave que se estima de manera no paramétrica. Aquí se pueden incluir partes lineales, polinomiales, etc., además de posibles interacciones. Si se usa una función de enlace para el valor esperado de la variable dependiente, tenemos un GAM. Si además hay términos aditivos aleatorios, tenemos un GAMM.

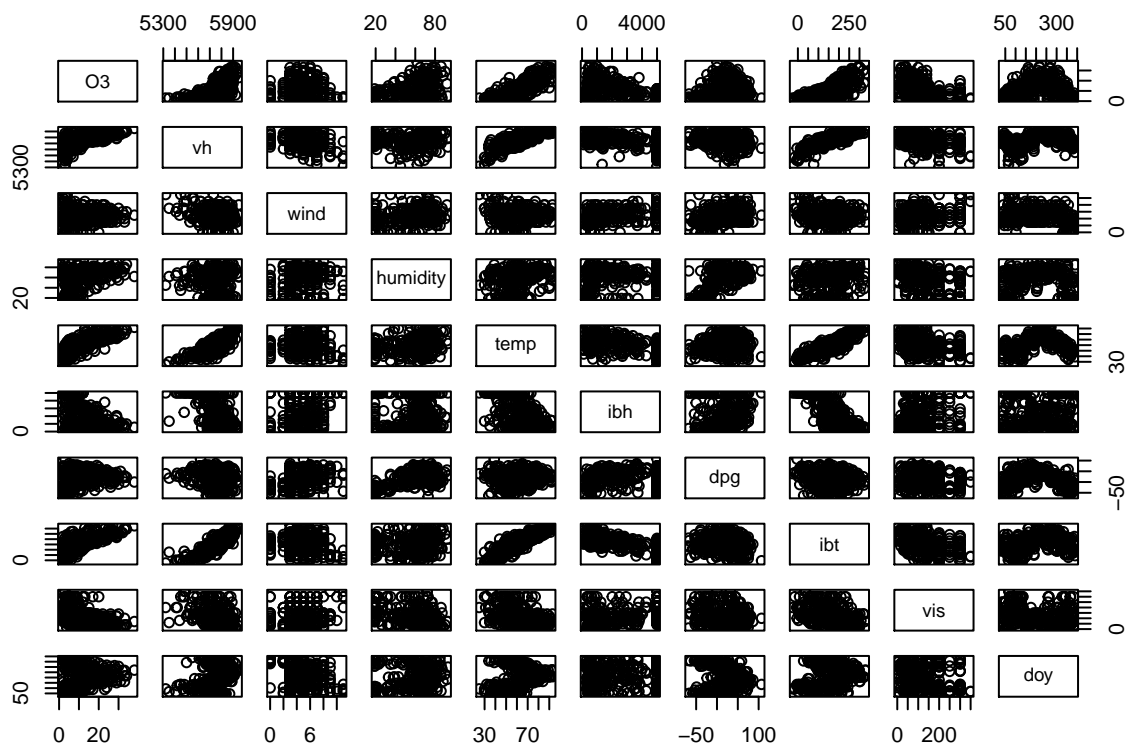
Estos modelos son muy flexibles, y permiten la interpretación de los efectos parciales de cada variable independiente o factor, al menos de manera gráfica.

El paquete `gam` usa un mecanismo de “backfitting”: va ajustando los residuos parciales de un ajuste sin usar una variable vs. esta variable hasta que converja. Permite diferentes suavizados para cada variable.

El paquete `mgcv` usa splines penalizados, y los términos de suavizado se eligen por validación cruzada para cada variable independiente.

Ejemplo 4.2 (Relación entre ozono y otras variables meteorológicas). Estos datos estudian la relación entre el ozono y distintas variables en la región de Los Angeles en 1976. Solo vamos a considerar tres predictores: temperatura medida en El Monte (`temp`), altura de la base de la inversión en el aeropuerto de Los Angeles (`ibh`) y temperatura de la inversión (`ibt`).

```
library(faraway)
data(ozone)
plot(ozone)
```



```
olm <- lm(O3~temp+ibh+ibt, ozone)
summary(olm)
```

Call:

```
lm(formula = O3 ~ temp + ibh + ibt, data = ozone)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-11.3224	-3.1913	-0.2591	2.9635	13.2860

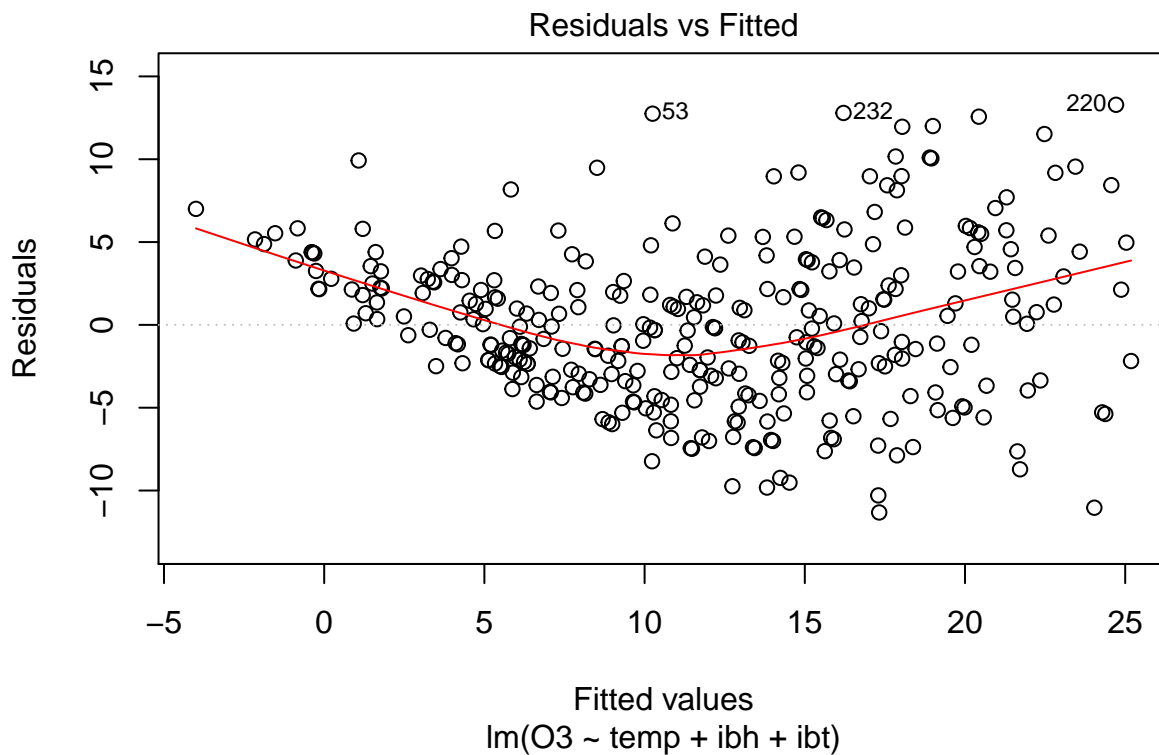
Coefficients:

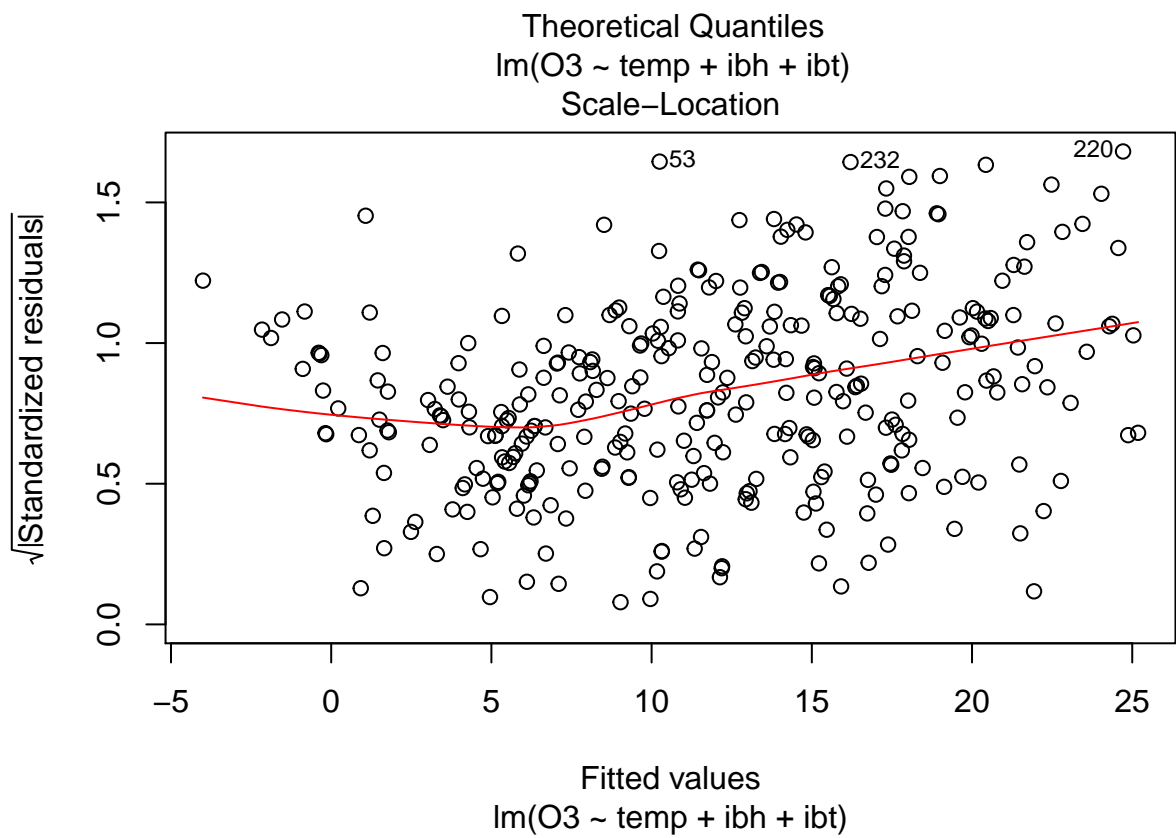
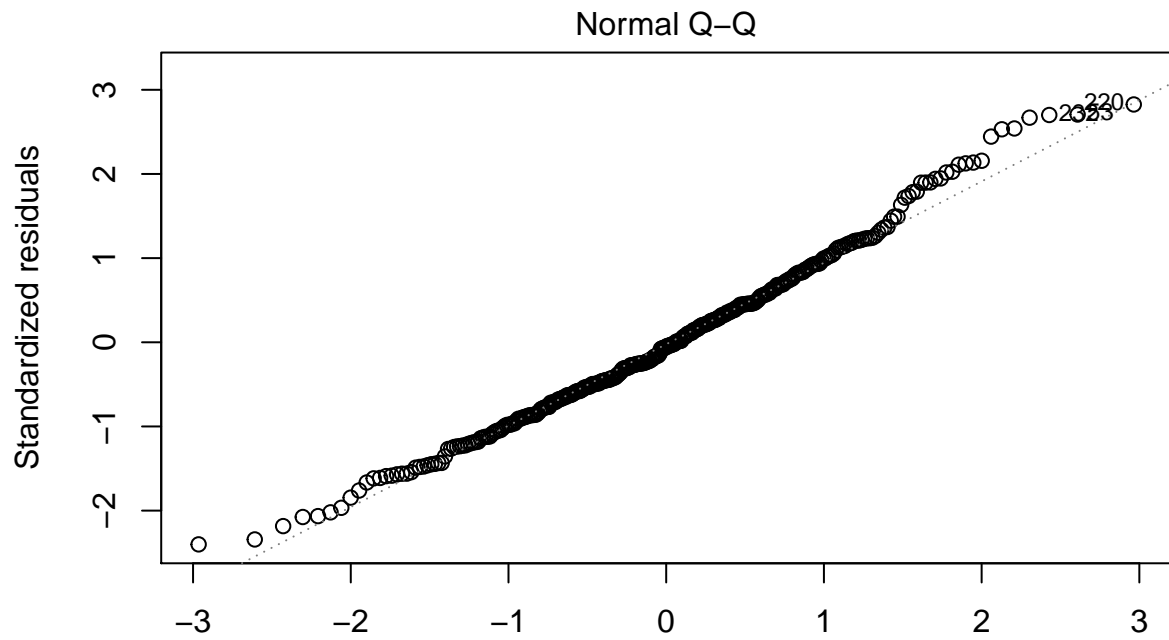
	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-7.7279822	1.6216623	-4.765	2.84e-06 ***
temp	0.3804408	0.0401582	9.474	< 2e-16 ***
ibh	-0.0011862	0.0002567	-4.621	5.52e-06 ***
ibt	-0.0058215	0.0101793	-0.572	0.568

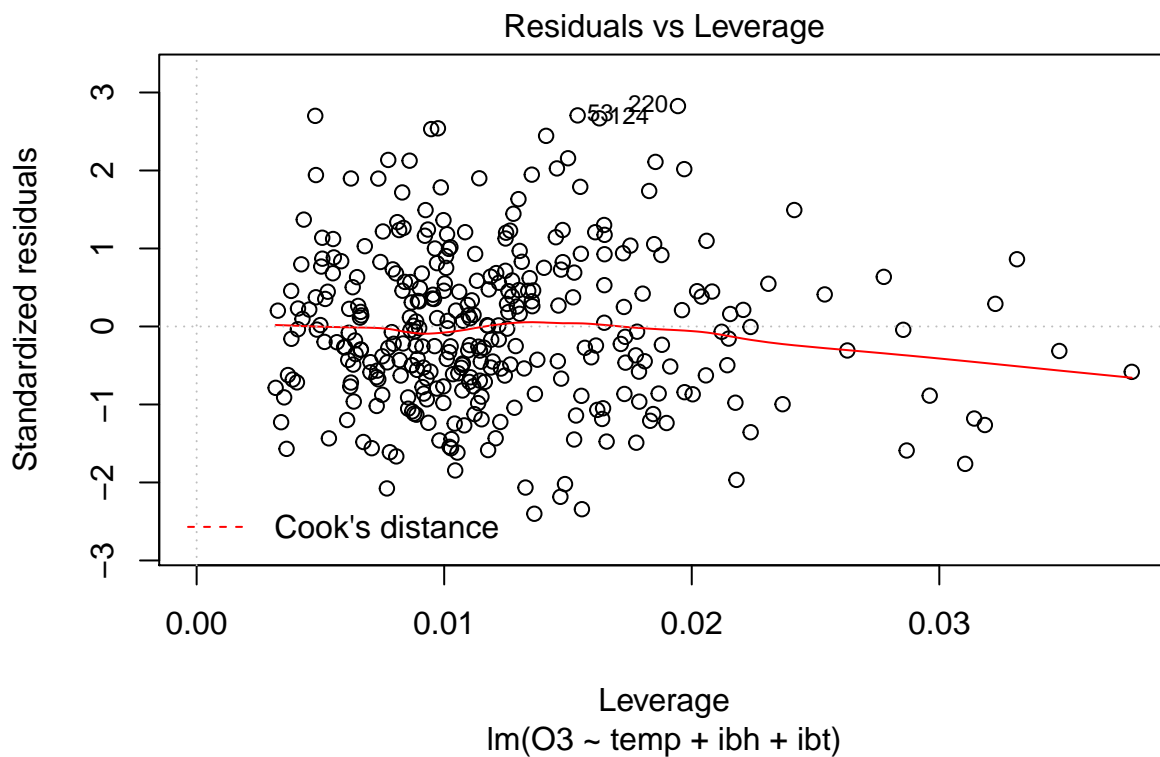
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.748 on 326 degrees of freedom
Multiple R-squared: 0.652, Adjusted R-squared: 0.6488
F-statistic: 203.6 on 3 and 326 DF, p-value: < 2.2e-16

```
plot(olm)
```







```
library(gam)
algaam <- gam( O3~lo(temp)+lo(ibh)+lo(ibt), data=ozone)
summary(algaam)
```

Call: gam(formula = O3 ~ lo(temp) + lo(ibh) + lo(ibt), data = ozone)

Deviance Residuals:

Min	1Q	Median	3Q	Max
-13.1146	-2.3624	-0.2092	2.1732	12.4447

(Dispersion Parameter for gaussian family taken to be 18.6638)

Null Deviance: 21115.41 on 329 degrees of freedom

Residual Deviance: 5935.096 on 318.0005 degrees of freedom

AIC: 1916.049

Number of Local Scoring Iterations: 2

Anova for Parametric Effects

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
lo(temp)	1	11958.2	11958.2	640.7153	< 2.2e-16 ***
lo(ibh)	1	1117.3	1117.3	59.8646	1.358e-13 ***
lo(ibt)	1	3.5	3.5	0.1898	0.6634

```

Residuals 318  5935.1    18.7
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Anova for Nonparametric Effects
      Npar Df Npar F      Pr(F)
(Intercept)
lo(temp)      2.5 7.4550 0.0002456 ***
lo(ibh)        2.9 7.6205 8.243e-05 ***
lo(ibt)        2.7 7.8434 9.917e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

algamr <- gam( 03~lo(temp)+lo(ibh), data=ozone)
summary(algamr)

```

```

Call: gam(formula = 03 ~ lo(temp) + lo(ibh), data = ozone)
Deviance Residuals:
      Min       1Q   Median       3Q      Max
-12.6284  -2.4766  -0.2668   2.2558  12.6436

```

(Dispersion Parameter for gaussian family taken to be 18.7915)

```

Null Deviance: 21115.41 on 329 degrees of freedom
Residual Deviance: 6044.567 on 321.6653 degrees of freedom
AIC: 1914.75

```

Number of Local Scoring Iterations: 2

```

Anova for Parametric Effects
      Df Sum Sq Mean Sq F value    Pr(>F)
lo(temp)  1.00 12373.6 12373.6 658.469 < 2.2e-16 ***
lo(ibh)    1.00   937.8   937.8  49.905 1.004e-11 ***
Residuals 321.67  6044.6    18.8
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Anova for Nonparametric Effects
      Npar Df  Npar F      Pr(F)
(Intercept)
lo(temp)      2.5 19.9446 2.629e-10 ***
lo(ibh)        2.9  5.4935  0.0013 **

```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
anova(algamr, algam)
```

Analysis of Deviance Table

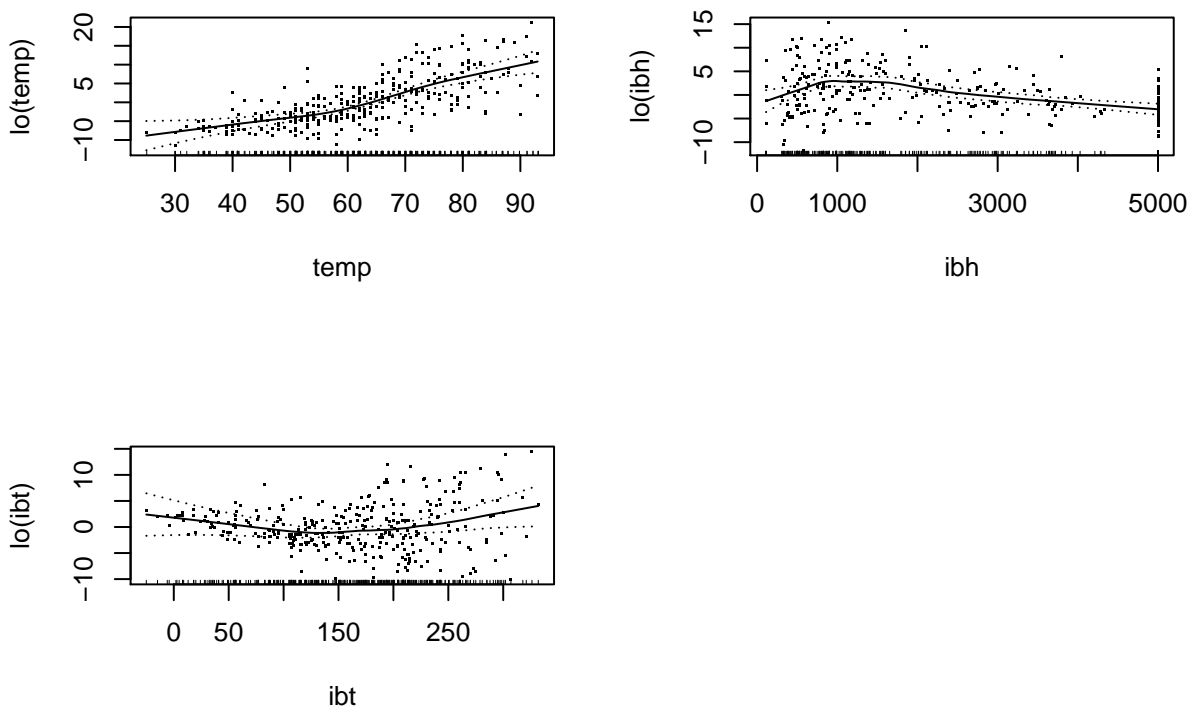
Model 1: $O_3 \sim \text{lo}(\text{temp}) + \text{lo}(\text{ibh})$

Model 2: $O_3 \sim \text{lo}(\text{temp}) + \text{lo}(\text{ibh}) + \text{lo}(\text{ibt})$

	Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
1	321.67	6044.6			
2	318.00	5935.1	3.6648	109.47	0.1763

```
par(mfrow=c(2,2))
```

```
plot(algam, residuals=T, se=T, pch=".")
```



```
asgam <- gam( O3~s(temp)+s(ibh)+s(ibt), data=ozone)  
summary(asgam)
```

Call: `gam(formula = O3 ~ s(temp) + s(ibh) + s(ibt), data = ozone)`

Deviance Residuals:

Min	1Q	Median	3Q	Max
-13.2725	-2.3447	-0.2385	2.1820	12.9764

(Dispersion Parameter for gaussian family taken to be 18.7364)

Null Deviance: 21115.41 on 329 degrees of freedom
Residual Deviance: 5939.441 on 316.9997 degrees of freedom
AIC: 1918.292

Number of Local Scoring Iterations: 2

Anova for Parametric Effects

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
s(temp)	1	12162.0	12162.0	649.1077	< 2.2e-16 ***
s(ibh)	1	1065.6	1065.6	56.8731	4.941e-13 ***
s(ibt)	1	2.0	2.0	0.1063	0.7446
Residuals	317	5939.4	18.7		

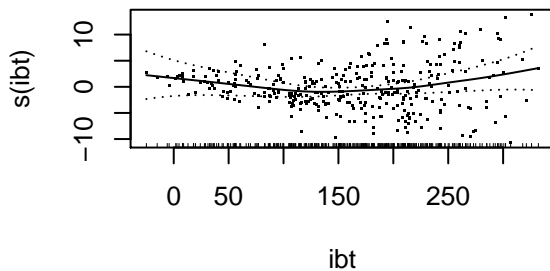
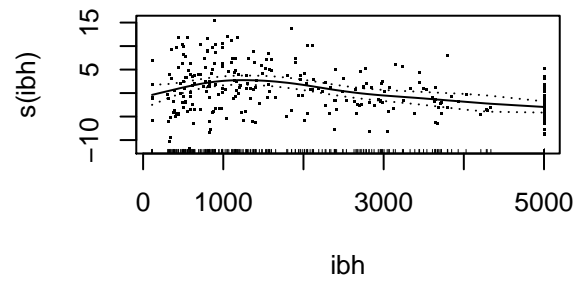
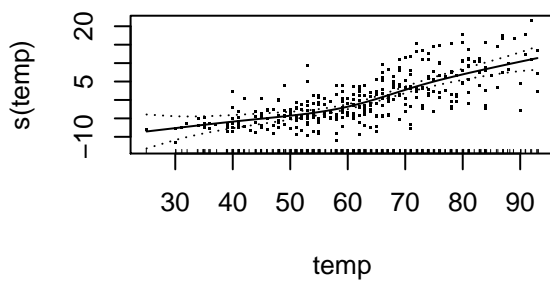
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Anova for Nonparametric Effects

	Npar	Df	Npar F	Pr(F)
(Intercept)				
s(temp)	3	7.2980	9.543e-05	***
s(ibh)	3	6.4328	0.000306	***
s(ibt)	3	5.5788	0.000968	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
par(mfrow=c(2,2))  
plot(asgam, residuals=T, se=T, pch=".")
```

```
library(mgcv)
asgam2<-gam( O3~s(temp)+s(ibh)+s(ibt), data=ozone)
summary(asgam2)
```

Family: gaussian
Link function: identity

Formula:
O3 ~ s(temp) + s(ibh) + s(ibt)

Parametric coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	11.7758	0.2382	49.44	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

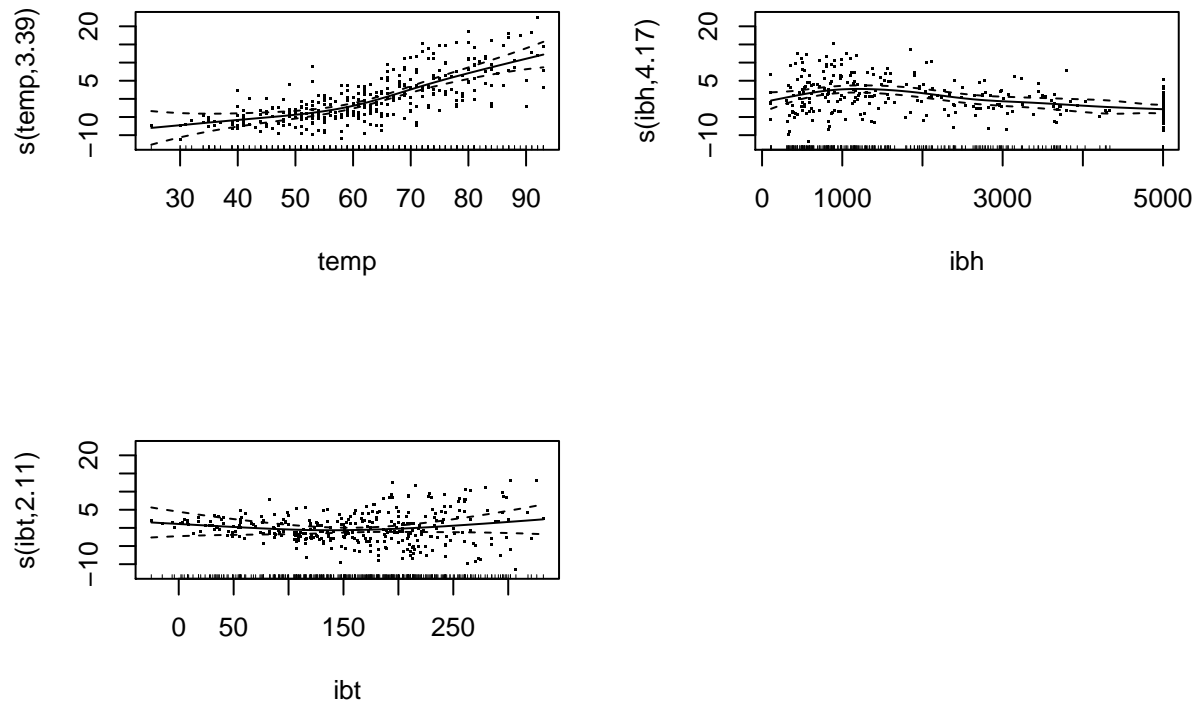
Approximate significance of smooth terms:

	edf	Ref.df	F	p-value
s(temp)	3.386	4.259	20.681	6.93e-16 ***
s(ibh)	4.174	5.076	7.338	1.38e-06 ***
s(ibt)	2.112	2.731	1.400	0.214

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) = 0.708 Deviance explained = 71.7%
GCV = 19.346 Scale est. = 18.72 n = 330

```
par(mfrow=c(2,2))  
plot(asgam2, residuals=T, se=T, pch=".")
```



Capítulo 5

Introducción a Estadística Bayesiana. Principios de Inferencia Bayesiana, MCMC, Gibbs Sampling

Existen dos formas en estadística para hacer inferencias mediante la cuantificación de la incertidumbre del proceso que genera los datos:

- **Estadística Frecuentista:**

- Considera los parámetros fijos pero desconocidos.
- Probabilidades se asignan a las observaciones dados los parámetros desconocidos.
- La probabilidad permitida en la inferencia de los parámetros se refiere a la frecuencia relativa con bastantes muestras (*nivel de confiabilidad*).
- Los procedimientos se juzgan en su desempeño en un número infinito de repeticiones hipotéticas de los experimentos.

- **Estadística Bayesiana:**

- Los parámetros son considerados variables aleatorias.
- Se puede asignar probabilidad tanto a observaciones como parámetros.
- Las probabilidades calculadas para parámetros se pueden interpretar como el *grado de credibilidad* y pueden ser subjetivas.
- Las reglas de probabilidad son usadas para revisar los *grados de credibilidad* acerca de los parámetros dado los datos observados.

5.1. Teorema de Bayes

Sean A y B dos eventos. Recuerde que

$$P(A \cap B) = P(B)P(A|B),$$

La probabilidad condicional de B dado A se define como

$$P(B|A) = \frac{P(A \cap B)}{P(A)}, \quad , P(A) > 0.$$

Pero, esta probabilidad condicional se puede re-escribir como

$$P(B|A) = \frac{P(B)P(A|B)}{P(A)}.$$

La interpretación de este teorema se debe a Laplace y Bayes. Preliminarmente, $P(B)$ (**a priori**) se calcula antes de observar A . Entonces la probabilidad de B se actualiza a $P(B|A)$ dado que se observó A (**a posteriori**).

Ahora, si se considera una partición del espacio muestral Ω entonces

$$P(\Omega) = P(\cup_{i=1}^n B_i) = \sum_{i=1}^n P(B_i) = 1.$$

Siguiendo las reglas de probabilidad se puede ver que

$$P(B_i|A) = \frac{P(B_i \cap A)}{P(A)} = \frac{P(A|B_i)P(B_i)}{\sum_{i=1}^n P(A|B_i)P(B_i)}.$$

Ejemplo 5.1. Una prueba nueva para VIH se supone que tiene 95 % de sensibilidad (probabilidad de que la prueba sea positiva dado que la persona tiene VIH) y una especificidad de 98 % (probabilidad de que la prueba sea negativa dado que la persona no tiene VIH). En la población Inglesa, la prevalencia (proporción) de VIH es 0,0015 y el interés es evaluar la probabilidad de que un paciente tenga VIH dado que la prueba fue positiva.

Sea $A :=$ “el paciente tiene VIH” (A^c es el evento complementario, es decir, que el paciente no tenga VIH) y sea $B :=$ “la prueba resultó positiva”.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B|A)P(A) + P(B|A^c)P(A^c)} = \frac{(0,95)(0,0015)}{(0,95)(0,0015) + (0,05)(0,9985)} = 0,067.$$

Es decir, se estima que cerca del 6.7 % de los pacientes que dan positivo en la prueba tendrán VIH. Compare este valor con la probabilidad previa de VIH: $P(A) = 0,0015$ (0,15 %).

Ejercicio 5.1. ¿Cómo cambia la probabilidad de tener VIH si la prevalencia (proporción) de VIH es igual a 1 %?

Ejercicio 5.2. Considere una prueba diagnóstica con un 90 % de sensibilidad y un 80 % de especificidad para una enfermedad con una incidencia de 1 en 10,000 individuos. Use el Teorema de Bayes para calcular la probabilidad de que un individuo tenga la enfermedad cuando la prueba da positiva y cuando da negativa.

5.2. Inferencia Bayesiana

Ahora considere una variable aleatoria X que sigue una distribución de probabilidad con función de densidad $f(x|\theta)$ y θ se asume aleatorio con función de densidad $\pi(\theta)$. Siguiendo argumentos similares de probabilidad, la función de densidad de θ dado x esta dada por

$$p(\theta|x) = \frac{f(x|\theta)\pi(\theta)}{\int f(x|\theta)\pi(\theta) d\theta} = \frac{f(x|\theta)\pi(\theta)}{m(x)}, \quad m(x) > 0.$$

- $\pi(\theta)$ se conoce como la *distribución a priori* (*prior distribution*). Más adelante se discutirán algunas opciones de $\pi(\theta)$ para ciertos problemas.
- $p(\theta|x)$ se conoce como la *distribución a posteriori* (*posterior distribution*).
- $m(x) = \int f(x|\theta)\pi(\theta) d\theta$ es la distribución marginal de x .
- Como $m(x)$ no depende de θ (es una constante de normalización), en ocasiones se expresa la distribución a posteriori como

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

La idea de la inferencia Bayesiana es usar la distribución a posteriori para hacer inferencias sobre el parámetro de interés. En muchos problemas, $p(\theta|x)$ no tiene una forma cerrada así que se recurre a métodos para aproximar dicha distribución. En el enfoque de MCMC se usan algoritmos para generar muestras aleatorias de dicha distribución. En otros enfoques tales como INLA (Wang et al., 2018) o métodos basados en variaciones (Lee and Wand, 2015) se utilizan métodos par aproximar dicha distribución sin necesidad de generar las muestras iterativamente.

5.2.1. Modelo Beta-Binomial

Ejemplo 5.2. Un estudio se llevó a cabo para evaluar la prevalencia de presión sanguínea alta entre personas viviendo a cinco millas del aeropuerto Heathrow (Londres, UK). La

variable respuesta es binaria: 1=“Presión sanguínea alta” (éxito) y 0=“Presión sanguínea normal o baja” (Blangiardo and Cameletti, 2015).

Asumiendo que los n individuos fueron muestreados independientemente, el número de éxitos y se puede modelar usando una distribución Binomial

$$y|\pi \sim \text{Binomial}(n, \pi).$$

La función de probabilidad esta dada por

$$f(y|\pi) = \binom{n}{y} \pi^y (1 - \pi)^{n - y}, \quad y = 0, 1, 2, \dots, n.$$

Una densidad candidata para distribución a priori es la siguiente:

$$\pi \sim \text{Beta}(a, b).$$

La distribución a posteriori de $\pi|y$ está dada por

$$p(\pi|y) \propto p(y|\pi)p(\pi) \propto \pi^{y+a-1}(1 - \pi)^{n-y+b-1},$$

la cual corresponde al núcleo de la distribución Beta

$$\pi|y \sim \text{Beta}(y + a, n - y + b).$$

Note que la media posterior está dada por

$$E(\pi|y) = \frac{y + a}{n + a + b}.$$

Compare esta cantidad a la media a priori

$$E(\pi) = \frac{a}{a + b}.$$

En el caso particular de observar $y = 0$ éxitos, la estimación frecuentista (MLE) de π sería $\hat{\pi} = 0$. Sin embargo, la estimación Bayesiana usando una distribución a priori $\text{Beta}(1, 1) \equiv U(0, 1)$ sería $E(\pi|y = 0) = \frac{1}{n+2} > 0$.

```
# Some examples
n=10
y=3
par(mfrow=c(3, 2))
a = 1; b = 1
curve(dbeta(x, a, b), main='A priori: Beta(a=1,b=1)',
      xlab=expression(pi), ylab=expression(p(pi)))
curve(dbeta(x, y+a, n-y+b), main='A posteriori: Beta(y+a, n-y+b)',
      xlab=expression(pi), ylab=expression(paste("p(", pi, "|y)")))
```

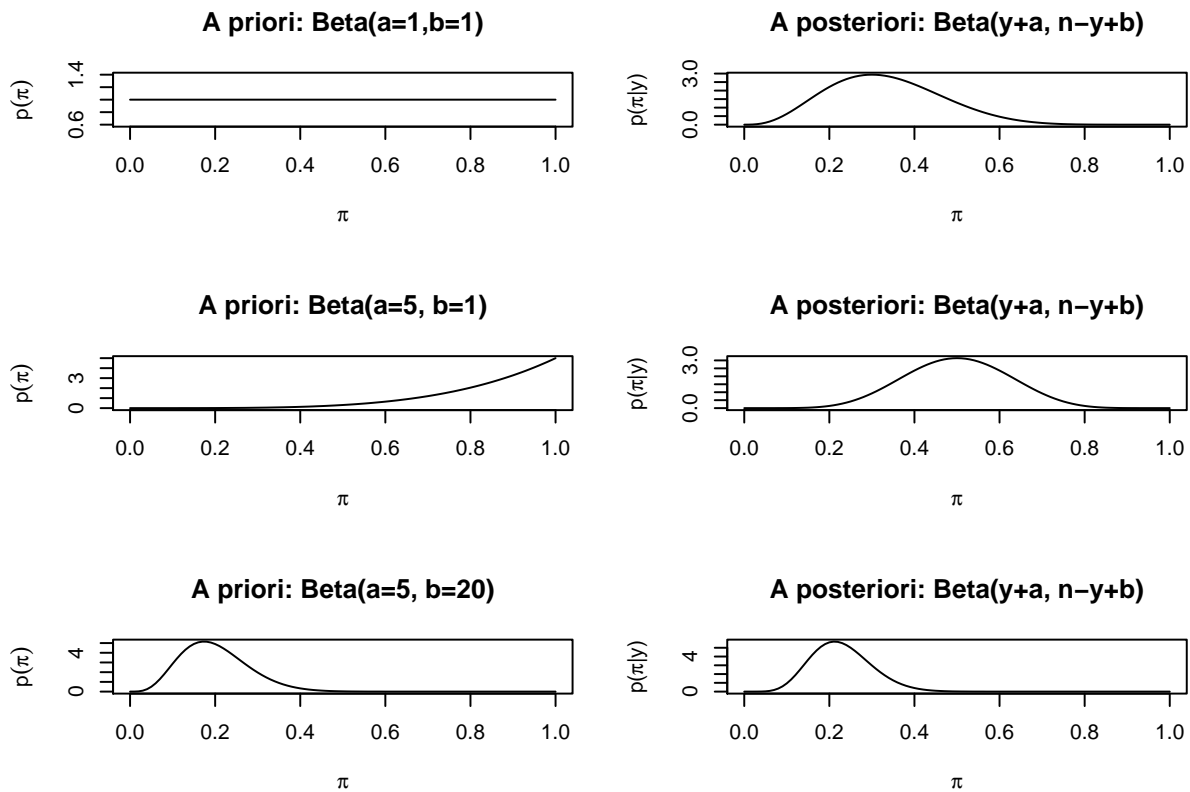


Figura 5.1: Distribuciones a priori y a posterior ($n=10$, $y=3$)

```
a = 5; b = 1
curve(dbeta(x, a, b), main='A priori: Beta(a=5, b=1)',
      xlab=expression(pi), ylab=expression(p(pi)))
curve(dbeta(x, y+a, n-y+b), main="A posteriori: Beta(y+a, n-y+b)",
      xlab=expression(pi), ylab=expression(paste("p(", pi, "|y)")))
a = 5; b = 20
curve(dbeta(x, a, b), main='A priori: Beta(a=5, b=20)',
      xlab=expression(pi), ylab=expression(p(pi)))
curve(dbeta(x, y+a, n-y+b), main="A posteriori: Beta(y+a, n-y+b)",
      xlab=expression(pi), ylab=expression(paste("p(", pi, "|y)")))
```

5.3. Distribuciones a priori

Existen diferentes formas para decidir que tipo de distribución a priori proponer para llevar a cabo inferencia Bayesiana. Las distribuciones a priori que se propongan dependerán del tipo de información disponible y del problema en cuestión (Carlin and Louis, 2009).

5.3.1. Distribuciones a priori conjugadas

Cuando la distribución a priori $\pi(\theta)$ y la distribución a posterior $p(\theta|\mathbf{x})$ pertenecen a la misma familia.

Por ejemplo, considere X el número de mujeres que llegan a un hospital a tener su hijo en un mes dado (Carlin and Louis, 2009). Suponga que $X \sim \text{Poisson}(\theta)$, $\theta > 0$ y que θ sigue una distribución Gamma con parámetros α y β , esto es,

$$\pi(\theta|\alpha, \beta) = \frac{\theta^{\alpha-1} e^{-\theta/\beta}}{\Gamma(\alpha)\beta^\alpha}, \quad \theta > 0, \alpha > 0, \beta > 0.$$

En este caso se asume que α y β son conocidos. Usando el Teorema de Bayes se puede mostrar que

$$p(\theta|x) \propto \theta^{x+\alpha-1} e^{-\theta(1+1/\beta)}.$$

Excepto por una constante, la expresión anterior corresponde a la densidad de una distribución Gamma con parámetros $x + \alpha$ y $(1 + 1/\beta)^{-1}$.

A manera de ejemplo, suponga que $x = 42$ y se adopta una distribución a priori $\text{Gamma}(5, 6)$ (recuerde que la media de X es 30 y la varianza 180).

```
alpha <- 5
beta <- 6
theta <- seq(0, 100, length.out=100)
prior <- dgamma(theta, shape=alpha, scale=beta)
x <- 42
posterior <- dgamma(theta, shape=x+alpha, scale=1/(1+1/beta))
plot(theta, posterior, xlab=expression(theta), ylab="density")
lines(theta, prior, lty=3)
postdraw <- rgamma(2000, shape=x+alpha, scale=1/(1+1/beta))
r1 <- hist(postdraw, freq=F, breaks=20, plot=F)
```

Warning in hist.default(postdraw, freq = F, breaks = 20, plot = F):
argument 'freq' is not made use of

```
lines(r1, lty=3, freq=F, col="gray90")
```

Aunque las distribuciones conjugadas son convenientes, su utilidad es limitada en problemas de dimensiones grandes. Sin embargo, en estos casos es posible especificar distribuciones a priori conjugadas para las distribuciones condicionales de los parámetros.

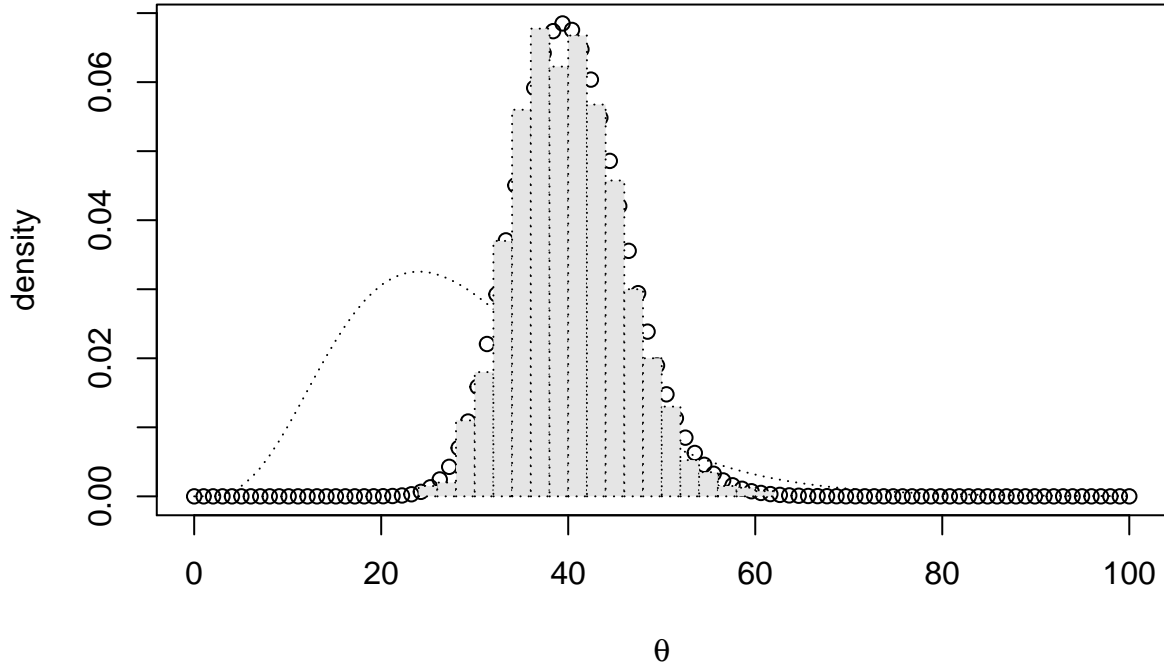


Figura 5.2: Distribución a priori y a posteriori conjugadas.

5.3.1.1. Inferencia para la media y la varianza con datos normales

Asuma que $y_1, \dots, y_n \stackrel{iid}{\sim} N(\mu, \sigma^2)$.

$$f(\mathbf{y}|\mu, \sigma^2) = \prod_{i=1}^n \left\{ \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left(-\frac{(y_i - \mu)^2}{2\sigma^2} \right) \right\}.$$

Considere la distribución a priori gamma-inversa-normal conjugada donde $f(\mu, \sigma^2) = f(\mu|\sigma^2)f(\sigma^2)$, donde $f(\mu|\sigma^2) = N(\mu_0, c\sigma^2)$ y $f(\sigma^2) = IG(a, b)$, donde $IG(a, b)$ es la distribución gamma-inversa. Si $\tau = 1/\sigma^2$, el parámetro de precisión, entonces la distribución a priori para $(\mu, \tau)^t$ esta dada por

$$f(\mu, \tau) = (2\pi)^{-1/2} \tau^{1/2} \exp \left(-\frac{\tau}{2c} (\mu - \mu_0)^2 \right) \frac{b^a}{\Gamma(a)} \tau^{a-1} e^{-b\tau}.$$

Esta parametrización con el parámetro de dispersión se implementa en la mayoría de programas estadísticos tales con WinBUGS.

La distribución a posteriori está dada por

$$f(\mu, \tau|\mathbf{y}) \propto \tau^{n/2+a+1/2-1} \exp \left\{ -\frac{\tau}{2} \frac{n}{w} (\mu - \tilde{\mu})^2 - \tau \left(b + \frac{SS}{2} \right) \right\},$$

donde $w = nc/(1 + nc)$, $\tilde{\mu} = w\bar{y} + (1 - w)\mu_0$, y

$$SS = (n - 1)s^2 + \frac{w}{c}(\bar{y} - \mu_0)^2.$$

Marginalmente,

$$f(\mu|\mathbf{y}) = t\left(\tilde{\mu}, \frac{n}{w} \frac{n+2a}{SS+2b}, n+2a\right),$$

y

$$f(\tau|\mathbf{y}) = \text{gamma}\left(\frac{n}{2} + a, \frac{SS}{2} + b\right).$$

5.3.2. Distribuciones a priori no informativas

Informalmente, una distribución se dice *no informativa* si es plana comparada a la verosimilitud. Es decir, si refleja bastante incertidumbre respecto a la información del parámetro. Otros nombres para estas distribuciones a priori son difusas, vagas o *flat*. Este tipo de distribuciones a priori pueden terminar en distribuciones a posterior no propias (no integran a 1) y en ocasiones no son invariantes a transformaciones.

A manera de ejemplo, suponga que $Y_1, \dots, Y_n \stackrel{\text{indep.}}{\sim} \text{Bernoulli}(p)$ y que $\pi(p) \propto 1$. Esto es equivalente a que $\pi(p) = \text{Beta}(1, 1)$. El MLE (*Maximum Likelihood Estimator*) de p es y/n , pero como se discutió en un ejercicio anterior la media posterior de p (estimador Bayesiano) es

$$\hat{p}_{\text{Bayes}} = \frac{y+1}{n+2},$$

lo cual indica que se agregó un éxito y dos ensayos a los datos originales usando una distribución a priori “no informativa”. Este estimador Bayesiano puede ser muy diferente al MLE si n y y son pequeños.

Una solución a la invarianza es usar distribuciones a priori de Jeffrey basadas en la matriz de Información de Fisher ($I(\theta)$):

$$\pi(\theta) \propto |I(\theta)|^{1/2}.$$

5.3.3. Distribuciones a priori informativas

Estas distribuciones se pueden proponer basadas en el conocimiento del investigador o en datos preliminares. Este tipo de distribuciones son muy usadas, por ejemplo, en estudios secuenciales donde se obtiene información de etapas anteriores. Lo importante es no usar información de la muestra para proponer una distribución a priori ya que se estaría usando la información dos veces (verosimilitud y a priori).

Una discusión sobre la elección de distribuciones a priori se puede encontrar aquí.

5.4. Métodos de Monte Carlo con Cadenas de Markov (MCMC)

Existen métodos de muestreo de una distribución tales como *rejection sampling*, *importance sampling* o *weighted bootstrap*. Sin embargo, estos métodos son no iterativos; es decir, se selecciona una muestra de tamaño N pero no existe el concepto de convergencia. Más aún, para problemas de dimensión alta no es fácil encontrar la distribución que sirva como envoltura (*envelope function*). Por lo tanto, se recurre a los métodos de Monte Carlo basados en Cadenas de Markov.

Recuerde que la idea básica es muestrear secuencialmente de una cadena de Markov cuya distribución estacionaria es exactamente la distribución conjunta a posteriori de los parámetros de interés. Un algoritmo para generar muestras de la distribución a posteriori usando MCMC es *Gibbs Sampling* o *Metropolis-Hasting*.

5.4.1. Gibbs Sampling

Suponga que se tiene un vector de parámetros $\theta = (\theta_1, \dots, \theta_k)^t$. Este algoritmo consiste en generar muestras de las distribuciones condicionales $f(\theta_j | \theta_{-j}, \mathbf{y})$, $j = 1, \dots, k$. Las muestras de estas distribuciones condicionales se pueden generar ya que tiene formas conocidas o se puede usar algún algoritmo de muestreo tales como muestreo adaptivo de rechazo (*adaptive rejection sampling*) o *Metropolis Hasting*. Bajo ciertas condiciones, la especificación de las distribuciones condicionales lleva a la distribución conjunta $f(\theta | \mathbf{y})$ de forma única y a las distribuciones marginales $f(\theta_j | \mathbf{y})$.

Valores iniciales: $(\theta_2^{(0)}, \theta_3^{(0)}, \dots, \theta_k^{(0)})$.

Pasos:

- **Paso 1:** Seleccione $\theta_1^{(t)}$ de $p(\theta_1 | \theta_2^{(t-1)}, \theta_3^{(t-1)}, \dots, \theta_k^{(t-1)}, \mathbf{y})$.
- **Paso 2:** Seleccione $\theta_2^{(t)}$ de $p(\theta_2 | \theta_1^{(t-1)}, \theta_3^{(t-1)}, \dots, \theta_k^{(t-1)}, \mathbf{y})$.
- **Paso k:** Seleccione $\theta_k^{(t)}$ de $p(\theta_k | \theta_1^{(t)}, \theta_3^{(t)}, \dots, \theta_k^{(t)}, \mathbf{y})$.

Bajo ciertas condiciones de regularidad, $(\theta_1^{(t)}, \dots, \theta_k^{(t)})$ converge en distribución a la muestra de la distribución a posterior verdadera $p(\theta_1, \dots, \theta_k | \mathbf{y})$.

5.4.2. Metropolis-Hasting (M-H)

Suponga que se quiere obtener una muestra de la distribución a posteriori $f(\theta | \mathbf{y})$. El algoritmo M-H se resume a continuación:

1. Genere valores iniciales $\theta^{(0)}$.

2. Genere un conjunto de valores candidatos θ' de la distribución de transición $q(\theta'|\theta)$.
3. Calcule

$$\alpha = \min \left(1, \frac{f(\theta'|\mathbf{y})q(\theta|\theta')}{f(\theta|\mathbf{y})q(\theta'|\theta)} \right)$$

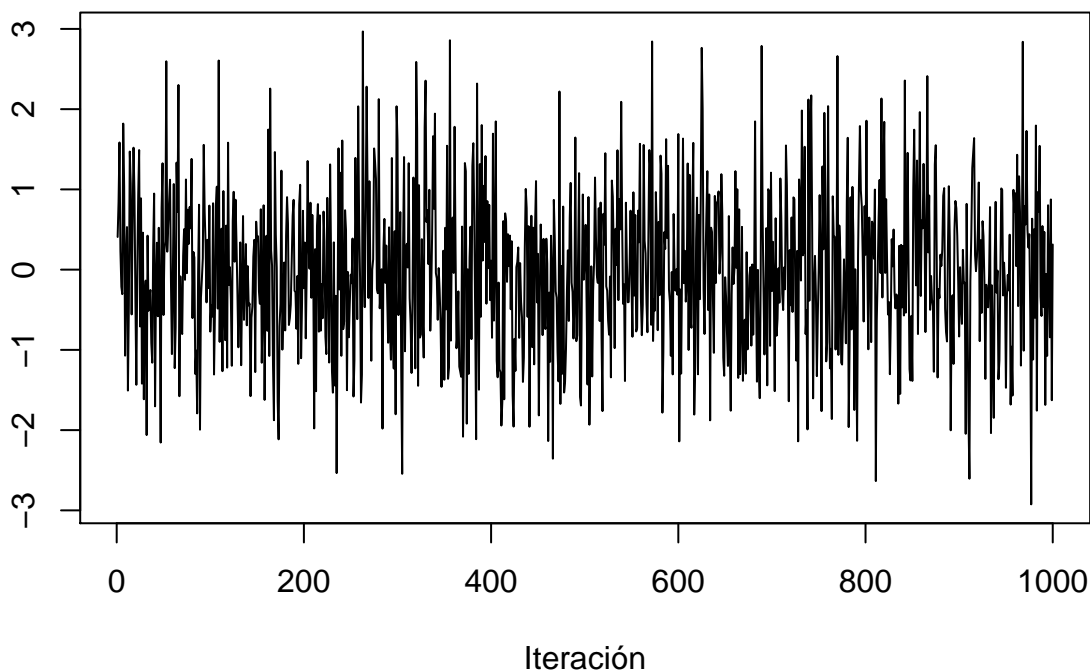
4. Actualice $\theta^{(t)} = \theta'$ con probabilidad α ; de lo contrario $\theta^{(t)} = \theta$.

A pesar de que *Gibbs Sampling* y el algoritmo de M-H se consideran separados en muchos libros, el *Gibbs Sampling* es un caso especial de M-H donde se usa la distribución condicional como la distribución de transición. Esta propuesta conlleva a una probabilidad de aceptación de $\alpha = 1$ (Ntzoufras, 2009). La ventaja de *Gibbs Sampling* es que en cada paso se generan muestras de distribuciones univariadas para las cuales existen bastantes herramientas. Este algoritmo se conoce como *Metropolis dentro de Gibbs*.

5.5. Diagnósticos de convergencia

La idea es verificar si el algoritmo ha alcanzado la distribución objetivo (equilibrio); en nuestro caso la distribución a posteriori. La convergencia se puede monitorear de diferentes maneras:

- Usando *trace plots*: una gráfica de las iteraciones vs. los valores generados (cadena). Si la convergencia se logra no deberían haber tendencias o periodicidades. Usualmente un período de descarte se establece para eliminar los valores generados al comienzo de la cadena (*burn-in period*). Cuando la cadena no muestra patrones se dice que la mezcla es buena: la cadena visita el soporte de la distribución (*good mixing*).



- Otras herramientas útiles son graficar la media ergódica (media hasta iteración k), o el promedio móvil.
- Otra forma de verificar convergencia es generando múltiples cadenas y observando la convergencia.
- Finalmente, se pueden calcular algunas estadísticas para verificar la convergencia:
 - *Estadística de Gelman-Rubin*: Compara las varianzas entre m cadenas. EL valor del estadístico debe ser cercano a 1.

$$\hat{R} = \sqrt{\frac{\widehat{Var}(\theta)}{W}},$$

donde $W = (1/n) = \sum_{j=1}^m s_j^2$ es la varianza dentro de las cadenas, B es la varianza entre cadenas y

$$\widehat{Var}(\theta) = \left(1 - \frac{1}{n}\right) W + \frac{1}{n} B.$$

Valores por encima de 1.05 podrían indicar problemas de convergencia.

- *Estadística de Geweke*: Prueba si la media del primer 10 % de la cadena difiera estadísticamente de la media del último 50 % de la cadena. El valor se interpreta como un puntaje z en una prueba de hipótesis de dos medias. Valores por fuera de -2.5 y 2.5 indican no estacionaridad de la cadena. En este caso se sugiere aumentar el período de *burn-in*.

- *Estadística de Heidel*: Usa el estadístico de Cramer-von-Mises para probar la hipótesis nula de que la muestra proviene de una distribución estacionaria. La prueba se aplica a toda la cadena y luego se vuelve a aplicar sucesivamente después de descartar el 10 %, 20 %, ... de la cadena hasta que la hipótesis nula se acepte, o hasta que el 50 % de la cadena se ha descartado. El último caso constituye una falla a la prueba de estacionaridad por lo que se recomienda una cadena más larga. En caso contrario, se muestra que por ciento de la cadena se debe eliminar.
- Monitoreando las autocorrelaciones. Es decir, la correlación entre las muestras separadas por un retardo $k = 0, 1, \dots$. Si las autocorrelaciones son muy altas se puede conservar las realizaciones cada cierto número, descartando las restantes. Por ejemplo, la número 1, 5, 10, 15, etc. Esto se conoce como *thinning*.

Estos diagnósticos se pueden obtener usando las librerías `ggmcmc` y `coda`. El objeto de JAGS se convierte en un objeto MCMC y luego se usan las funciones de estas librerías. Un ejemplo se discutirá más adelante.

```
library(ggmcmc)
jags1.mcmc <- as.mcmc(jags1) # coda
M <- ggs(jags1.mcmc) # ggmcmc
```

5.6. Inferencia Bayesiana

Para el caso univariado, una estimación puntual de $\hat{\theta}(\mathbf{y})$ de θ se puede obtener calculando una estadística (media, moda o mediana) para la distribución a posteriori $p(\theta|\mathbf{y})$.

Cuando la distribución a priori $\pi(\theta) \propto 1$ entonces la moda de $p(\theta|\mathbf{y})$ coincide con el estimador máximo verosímil.

Una medida de incertidumbre para $\hat{\theta}(\mathbf{y})$ se puede obtener calculando la varianza a posteriori. La media a posteriori minimiza la varianza a posteriori con respecto a todos los estimador puntuales $\hat{\theta}(\mathbf{y})$.

Intervalos de credibilidad de $100 \times (1 - \alpha) \%$ para los parámetros de interés θ se pueden calcular usando los respectivos percentiles de las muestras obtenidas mediante MCMC (intervalos con colas iguales). El intervalo de credibilidad se interpreta como “la probabilidad de que θ esté en el intervalo dado los datos observados \mathbf{y} es al menos $(1 - \alpha)$ ”.

5.7. Comparación de modelos

Una forma de comparar dos o más modelos es usando el DIC (*Deviance Information Criterion*).

$$DIC(m) = 2\overline{D(\boldsymbol{\theta}_m, m)} - D(\bar{\boldsymbol{\theta}}_m, m) = D(\bar{\boldsymbol{\theta}}_m, m) + 2p_m,$$

donde $D(\boldsymbol{\theta}, m) = -2 \log f(\mathbf{y}|\boldsymbol{\theta}, m)$ y $\overline{D(\boldsymbol{\theta}_m, m)}$ es su media a posteriori. p_m se puede interpretar como el “número efectivo de parámetros” dado por

$$p_m = \overline{D(\boldsymbol{\theta}_m, m)} - D(\bar{\boldsymbol{\theta}}, m).$$

Finalmente, $\bar{\boldsymbol{\theta}}_m$ es la media a posteriori de los parámetros en el modelo.

Menores valores de DIC indican mejor ajuste (al igual que AIC y BIC en el contexto frecuentista).

El DIC se debe usar con precaución ya que asume que la media a posteriori es una buena estadística descriptiva del centro de la distribución a posteriori. Así, se han reportado problemas donde la distribución a posteriori es asimétrica o no unimodal (Ntzoufras, 2009).

Otro criterio menos conocido es el *Watanabe AIC* (WAIC). Gelman y colaboradores (2014) sugieren que este criterio es preferido a DIC.

5.8. Introducción a JAGS

De acuerdo con su autor, Martyn Plummer (Plummer, 2003), JAGS (*Just Another Gibbs Sampling*) es una programa para el análisis de modelos jerárquicos Bayesianos usando simulaciones via MCMC. La sintaxis de JAGS es similar a la de BUGS.

Una ventaja de JAGS sobre los demás programas existentes tales como STAN, WinBUGS y NIMBLE es su facilidad de uso e interacción con R usando la librería `rjags` y `R2jags`. También corre en diferentes sistemas operativos y plataformas.

JAGS implementa el *Adaptive Rejection Metropolis sampler*. Su programación interna es en C++.

Para correr JAGS se necesitan los siguientes elementos:

- **Comandos para definir el modelo (verosimilitud) y las distribuciones a priori:** La sintaxis es similar a WinBUGS. El símbolo \sim define una relación estocástica mientras que el símbolo \leftarrow define una relación determinística. Se usan subíndices para vectores, matrices y arreglos de manera similar que en R. Es decir,

los datos pueden ingresar como un arreglo unidimensional o como una matriz, dependiendo del problema.

Las distribuciones a priori se definen en el manual del programa.

En la definición de distribuciones a priori se pueden usar subíndices o definir cada componente individualmente. Esta definición

```
for (k in 1:3){ a[k]~dnorm(0, 1.0E-11) }
```

es equivalente a

```
a[1]~dnorm(0, 1.0E-11 )
a[2]~dnorm(0, 1.0E-11 )
a[3]~dnorm(0, 1.0E-11 )
```

Este es un ejemplo de una definición de un modelo:

```
model {
  for (i in 1:N) {
    Y[i] ~ dnorm(mu[i], tau)
    mu[i] <- alpha + beta * (x[i] - x.bar)
  }
  x.bar <- mean(x)
  alpha ~ dnorm(0.0, 1.0E-4)
  beta ~ dnorm(0.0, 1.0E-4)
  sigma <- 1.0/sqrt(tau)
  tau ~ dgamma(1.0E-3, 1.0E-3)
}
```

- **Base datos para el modelo:** en esta parte se definen las variables involucradas en la definición del modelo como una lista, incluyendo las dimensión de los datos (n).

```
# Crear base de datos para el modelo jags
sim.dat.jags <-list(
  n = nrow(example1),
  x = x,
  y = y
)
```

- **Lista de valores iniciales:** Se pueden especificar valores iniciales para las cadenas de cada parámetro. Luego se especifica en la opción `inits`. Por ejemplo,

```
valores.ini <- list(beta0 = 1, beta1 = 1, beta2 = c(NA, -2, 4), tau = 2)
```

En este caso β_2 corresponde a los coeficientes de un factor con tres niveles donde el primero es el de referencia (también se especifica como `beta2[3] <- 0` en el programa del modelo).

5.8.1. Ejemplo simulado de regresión lineal

```
set.seed(1)
# Datos simulados
n <- 500
x <- 1:n
epsilon <- rnorm(n, 0, 1)
y <- 2-5*x + epsilon

example1 <- data.frame(x = x, y = y, Epsilon = epsilon)

# Modelo
modelo <- function()
{
  for (i in 1:n){
    y[i] ~ dnorm(y.hat[i], tau)
    y.hat[i] <- a + b * x[i]
  }
  # Distribuciones a priori
  a ~ dnorm(0, .0001)
  b ~ dnorm(0, .0001)
  tau <- pow(sigma, -2)
  sigma ~ dunif(0, 100)
}

# Crear base de datos para el modelo jags
sim.dat.jags <-list(
  n = nrow(example1),
  x = x,
  y = y
)

library(rjags)
library(R2jags)

parametros <- c("a","b", "tau", "sigma")
jags1 <- jags(data = sim.dat.jags,
              inits = NULL,
              parametros,
              n.iter = 1000,
              model.file = modelo,
```

```
n.burnin = 200,
n.chains = 1,
n.thin = 1)
```

```
Compiling model graph
  Resolving undeclared variables
  Allocating nodes
Graph information:
  Observed stochastic nodes: 500
  Unobserved stochastic nodes: 3
  Total graph size: 2013
```

```
Initializing model
```

```
summary(jags1)
```

	Length	Class	Mode
model	8	jags	list
BUGSoutput	24	bugs	list
parameters.to.save	5	-none-	character
model.file	1	-none-	character
n.iter	1	-none-	numeric
DIC	1	-none-	logical

```
# Estadísticas Descriptivas Distribución A Posteriori
```

```
jags1$BUGSoutput$summary
```

	mean	sd	2.5%	25%	50%
a	2.0677911	0.0941203852	1.8766890	2.0056541	2.0691375
b	-5.0001861	0.0003184123	-5.0007918	-5.0003903	-5.0001700
deviance	1432.3615990	2.4646774320	1429.6749268	1430.5404903	1431.7622197
sigma	1.0148547	0.0302793799	0.9595032	0.9940803	1.0150424
tau	0.9735216	0.0578673000	0.8585630	0.9350851	0.9705806
	75%	97.5%			
a	2.129222	2.252254			
b	-4.999987	-4.999552			
deviance	1433.457999	1438.379698			
sigma	1.034128	1.079230			
tau	1.011945	1.086193			

```
# Estadística Descriptivas Distribución A Posteriori (Individual)
```

```
jags1$BUGSoutput$summary["a",]
```

	mean	sd	2.5%	25%	50%	75%
a	2.06779106	0.09412039	1.87668897	2.00565413	2.06913750	2.12922244

```
jags1$BUGSoutput$summary["b",]
```

```
jags1$BUGSoutput$summary["tau",]
```

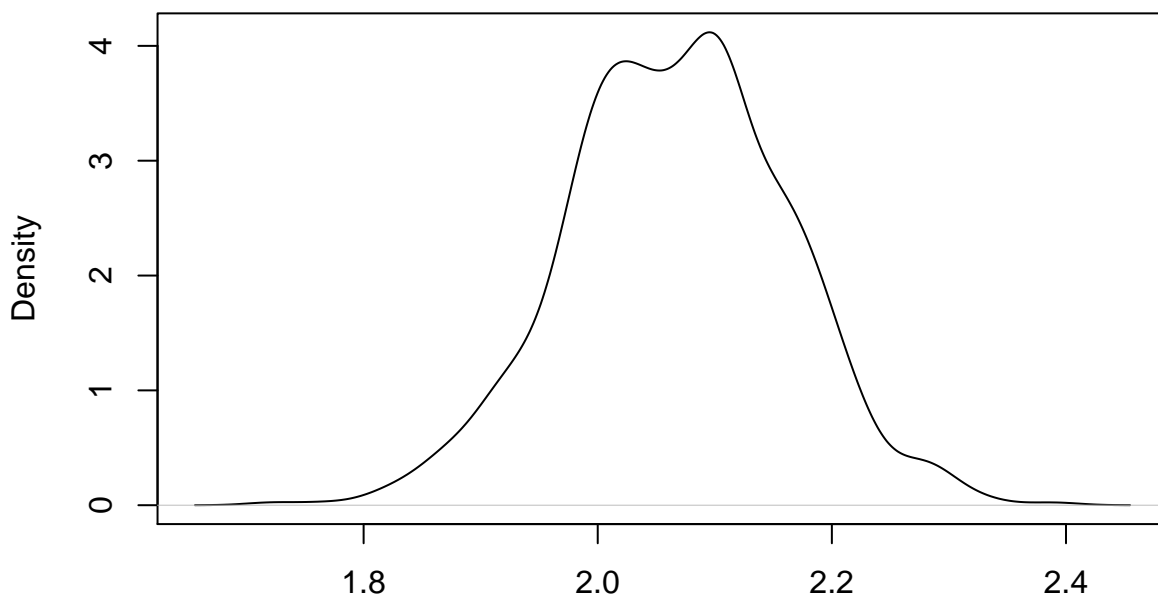
DIC y número efectivo de parámetros

```
jags1$BUGSoutput$pD
```

Gráficas de las distribuciones posteriores

```
plot(density(jags1$BUGSoutput$sims.list$a))
```

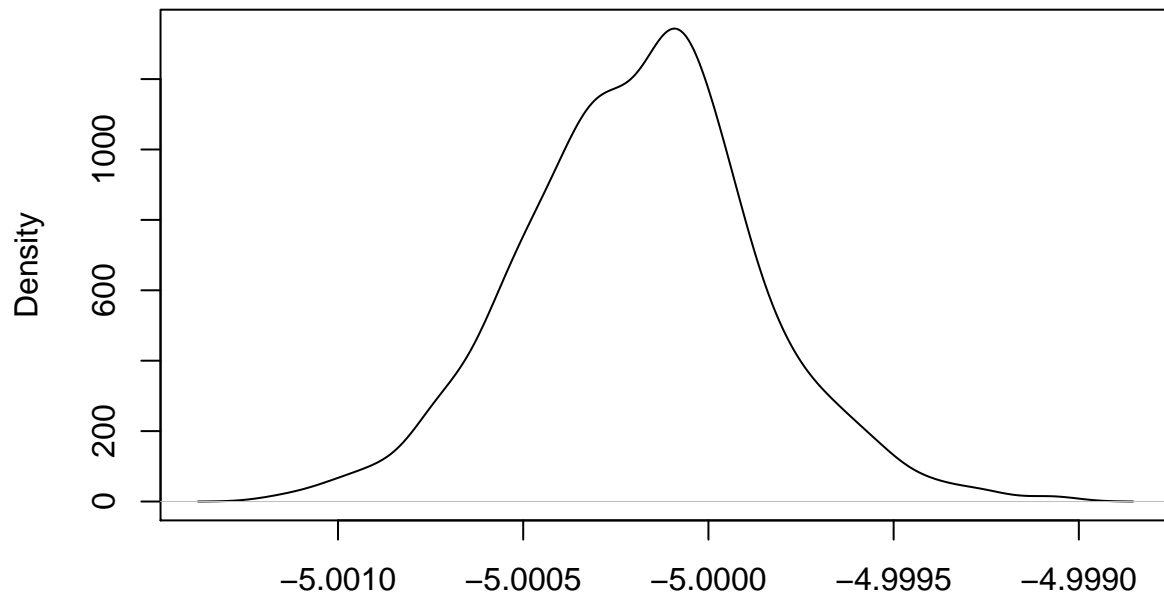
```
density.default(x = jags1$BUGSoutput$sims.list$a)
```



170

```
plot(density(jags1$BUGSoutput$sims.list$b))
```

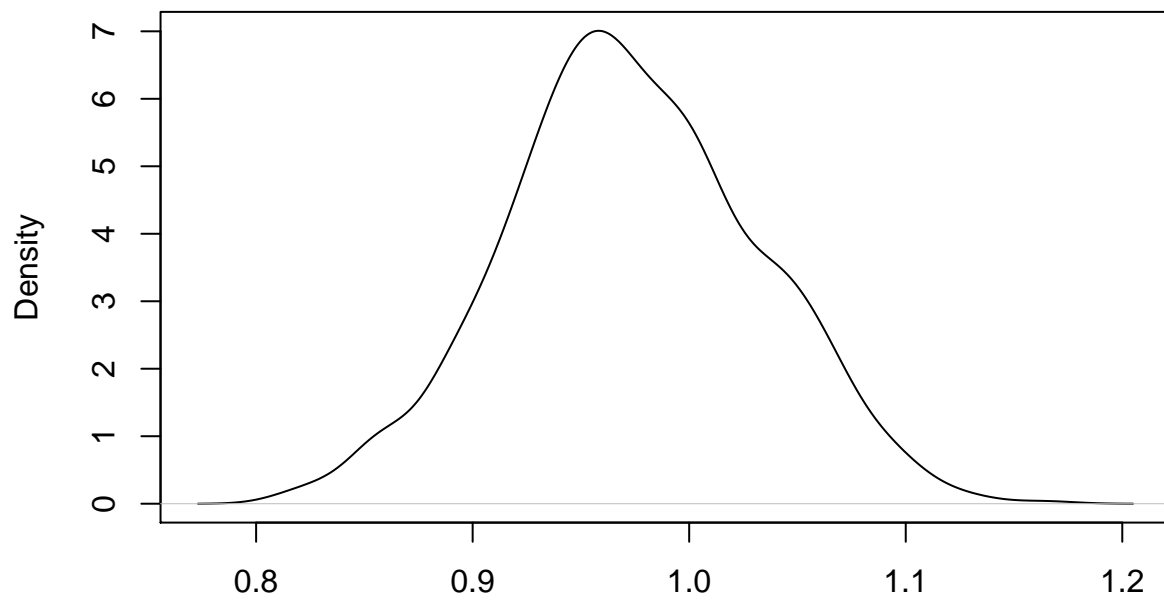
density.default(x = jags1\$BUGSoutput\$sims.list\$b)



N = 800 Bandwidth = 7.112e-05

```
plot(density(jags1$BUGSoutput$sims.list$tau))
```

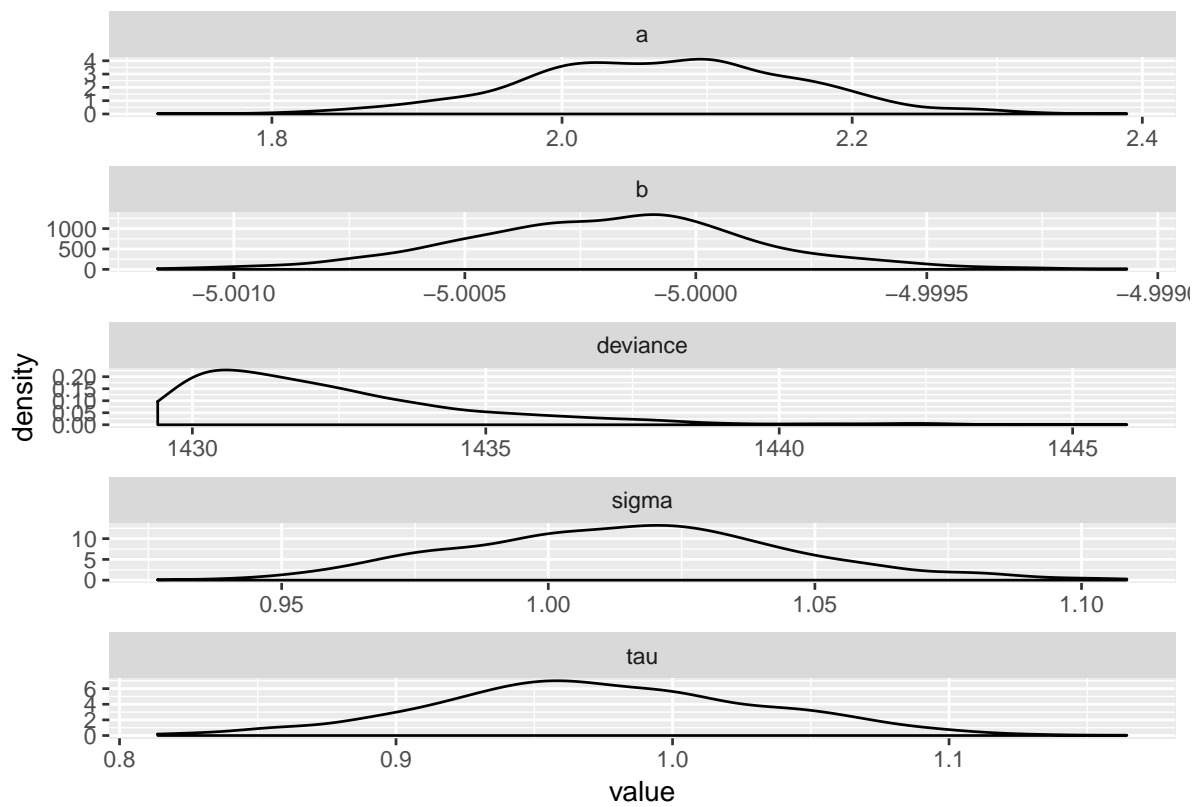
density.default(x = jags1\$BUGSoutput\$sims.list\$tau)



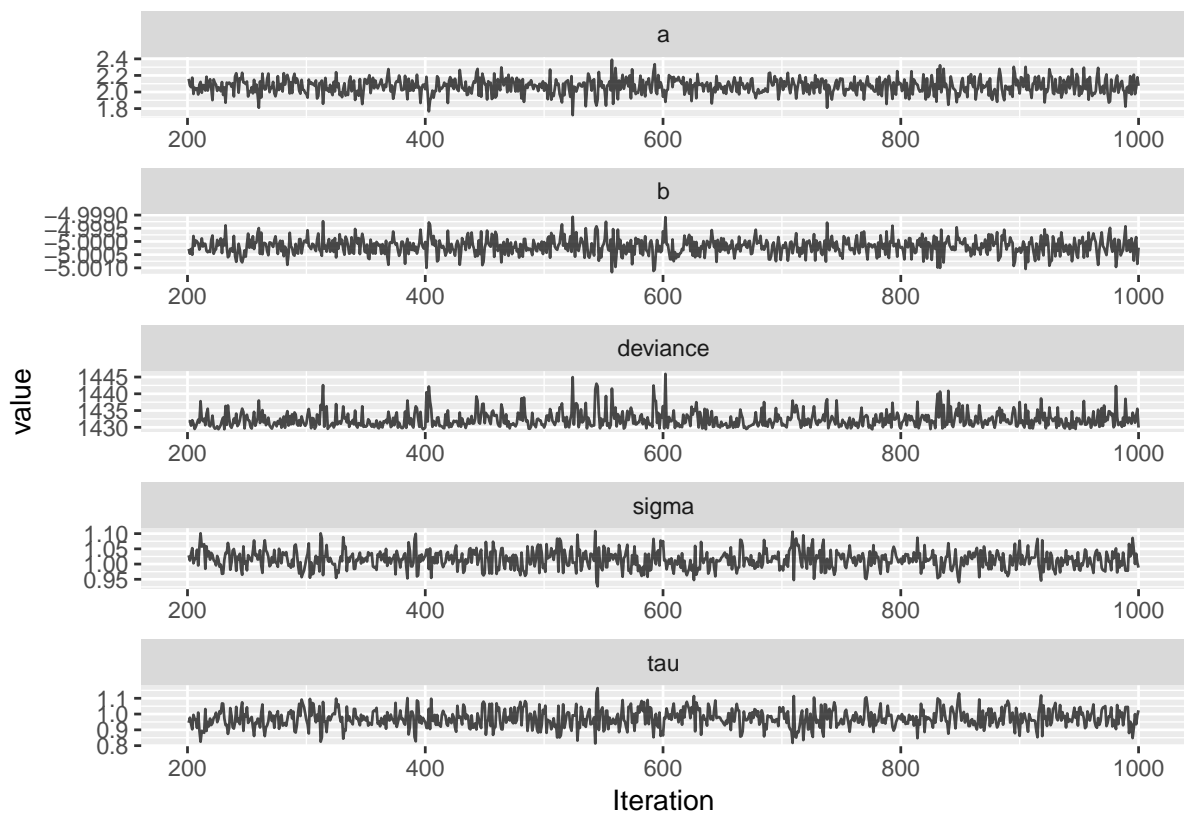
N = 800 Bandwidth = 0.01356

```
# Diagn3sticos de convergencia
library(ggmcmc)
library(coda)
jags1.mcmc <- as.mcmc(jags1)
M <- ggs(jags1.mcmc)

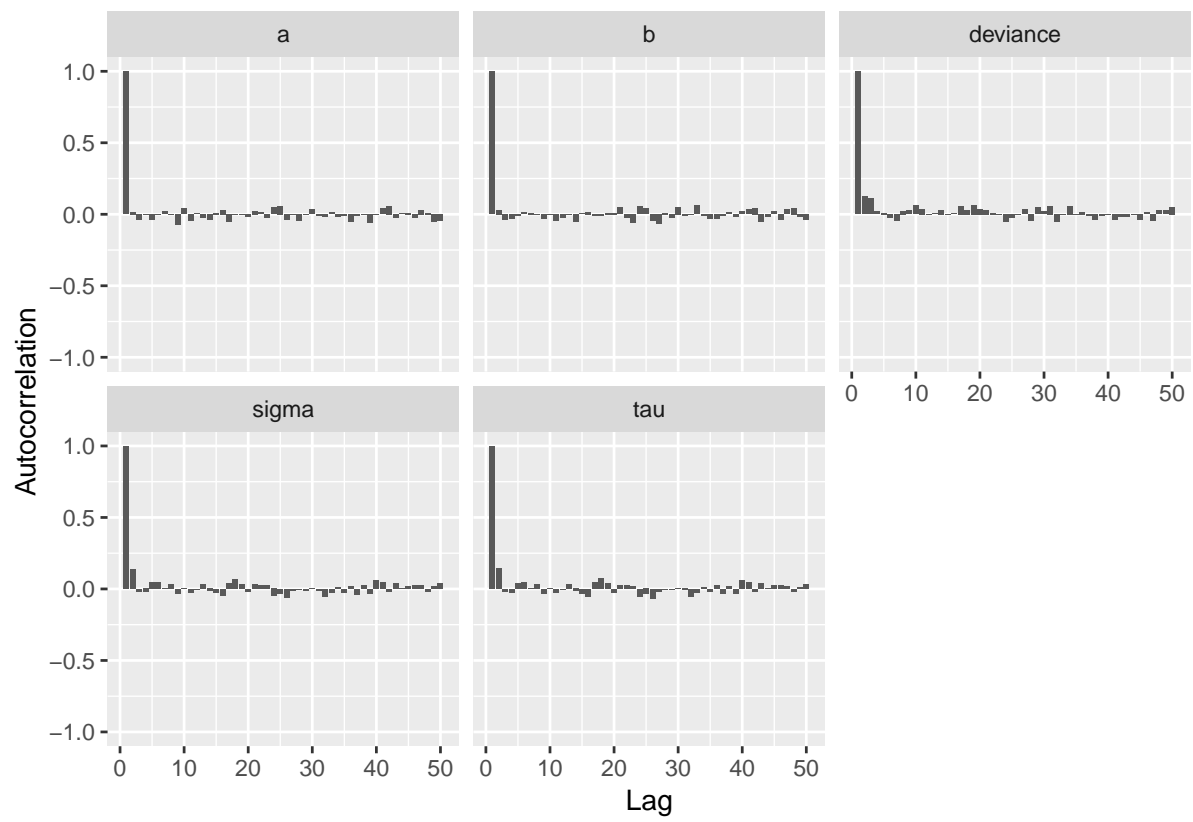
# Diagn3sticos graficos
density <- ggs_density(M)
density
```



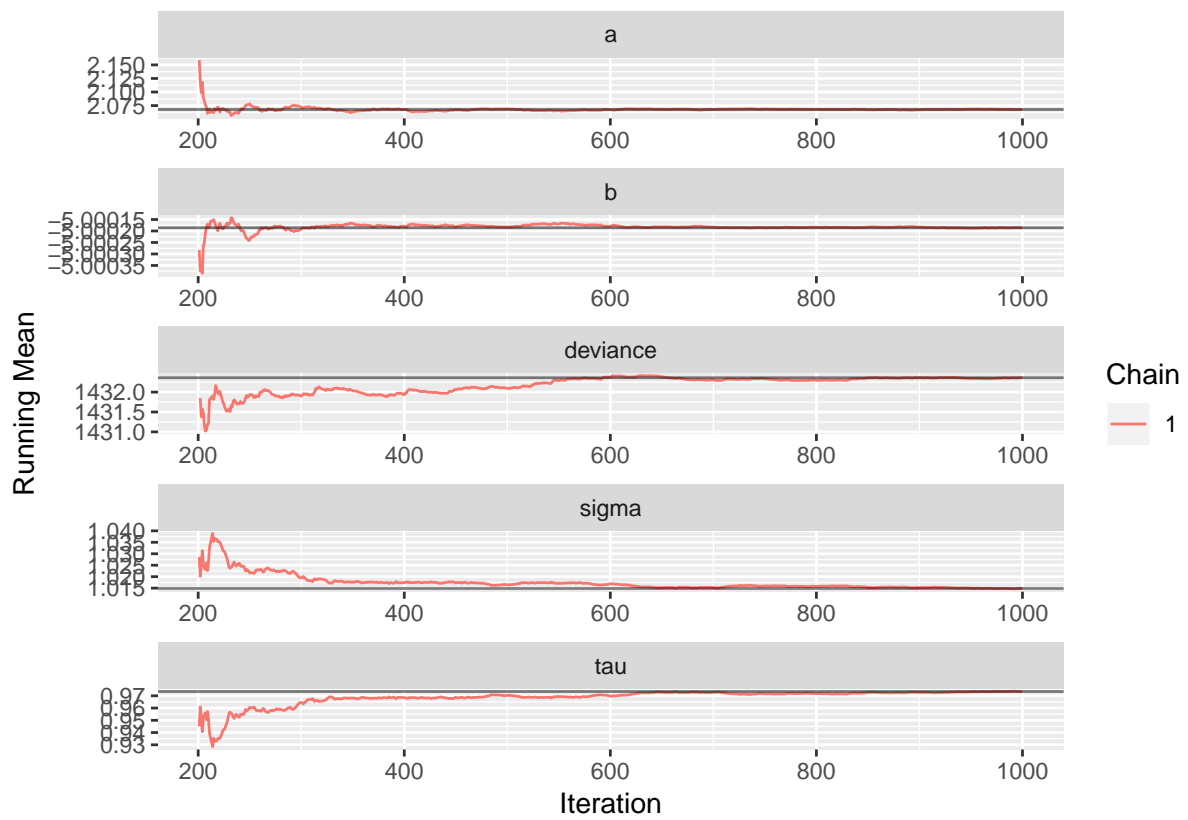
```
trace <- ggs_traceplot(M)
trace
```



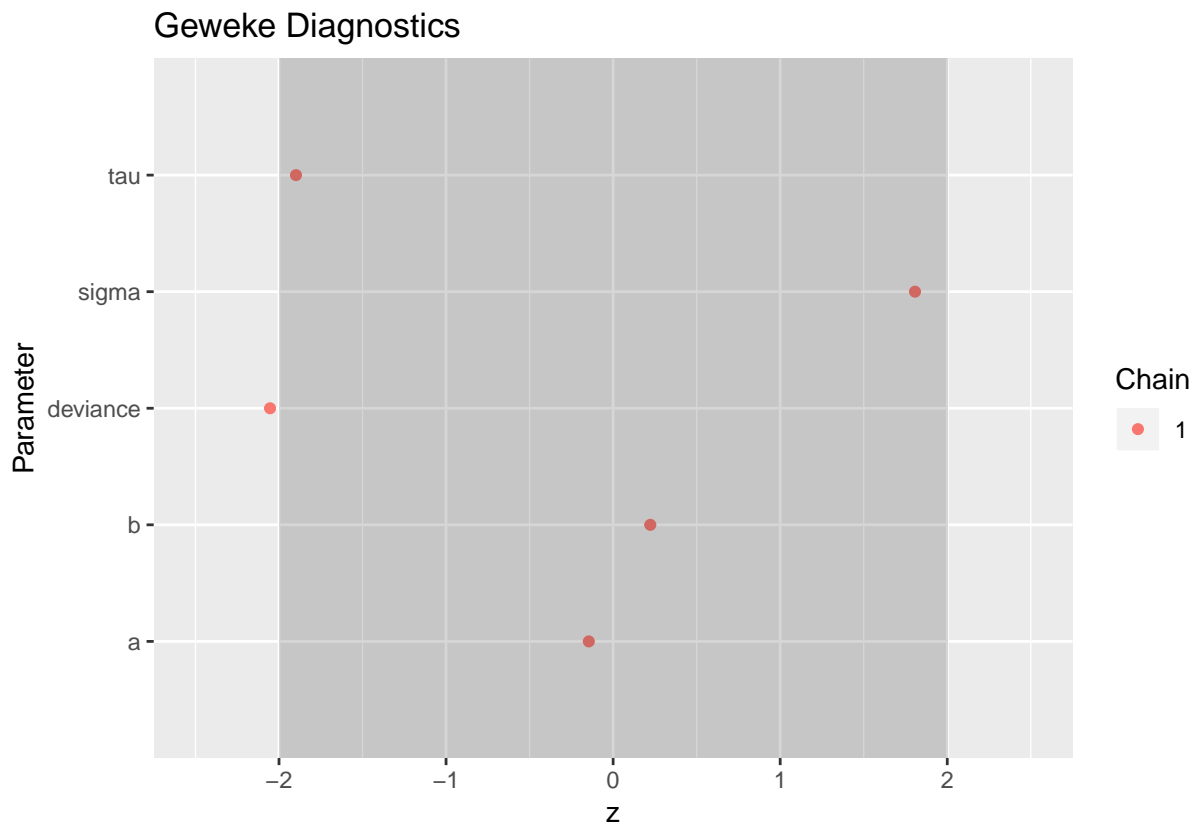
```
auto <- ggs_autocorrelation(M)
auto
```



```
run <- ggs_running(M)
run
```

`ggs_geweke(M)`



Ahora vamos a correr el MCMC con dos cadenas para poder calcular el estadístico de Gelman-Rubin.

```
set.seed(1)
# Datos simulados
n <- 500
x <- 1:n
epsilon <- rnorm(n, 0, 1)
y <- 2-5*x + epsilon

example1 <- data.frame(x = x, y = y, Epsilon = epsilon)

# Modelo
modelo <- function()
{
  for (i in 1:n){
    y[i] ~ dnorm(y.hat[i], tau)
    y.hat[i] <- a + b * x[i]
  }
  # Distribuciones a priori
```

```

a ~ dnorm(0, .0001)
b ~ dnorm(0, .0001)
tau <- pow(sigma, -2)
sigma ~ dunif(0, 100)
}

# Crear base de datos para el modelo jags
sim.dat.jags <-list(
  n = nrow(example1),
  x = x,
  y = y
)

library(rjags)
library(R2jags)

parametros <- c("a","b", "tau", "sigma")
jags1 <- jags(data = sim.dat.jags,
              inits = NULL,
              parametros,
              n.iter = 1000,
              model.file = modelo,
              n.burnin = 200,
              n.chains = 2,
              n.thin = 1)

```

```

Compiling model graph
  Resolving undeclared variables
  Allocating nodes
Graph information:
  Observed stochastic nodes: 500
  Unobserved stochastic nodes: 3
  Total graph size: 2013

```

Initializing model

```

library(ggmcmc)
library(coda)
jags1.mcmc <- as.mcmc(jags1)
M <- ggs(jags1.mcmc)

gelman.diag(jags1.mcmc)

```

Potential scale reduction factors:

	Point est.	Upper C.I.
a	1.01	1.02
b	1.00	1.00
deviance	1.00	1.01
sigma	1.00	1.00
tau	1.00	1.00

Multivariate psrf

1.01

```
heidel.diag(jags1.mcmc)
```

[[1]]

	Stationarity test	start iteration	p-value
a	passed	1	0.9968
b	passed	1	0.7819
deviance	passed	1	0.0782
sigma	passed	1	0.1597
tau	passed	1	0.1750

	Halfwidth test	Mean	Halfwidth
a	passed	2.068	6.52e-03
b	passed	-5.000	2.21e-05
deviance	passed	1432.362	2.14e-01
sigma	passed	1.015	2.41e-03
tau	passed	0.974	4.63e-03

[[2]]

	Stationarity test	start iteration	p-value
a	passed	1	0.226
b	passed	1	0.513
deviance	passed	1	0.315
sigma	passed	1	0.402
tau	passed	1	0.452

		Halfwidth	Mean	Halfwidth
	test			
a	passed	2.074	6.33e-03	
b	passed	-5.000	2.46e-05	
deviance	passed	1432.464	2.10e-01	
sigma	passed	1.014	2.48e-03	
tau	passed	0.975	4.74e-03	

Vale la pena mencionar que la formulación anterior de un modelo en JAGS no es la única forma de hacerlo. Existen variantes que siguen la misma línea aunque con algunas diferencias en la forma en que se ingresan los datos, se define el modelo, entre otras.

Ejercicio 5.3. Ajuste el modelo anterior pero cambie la distribución a priori de τ por una $gamma(\epsilon, \epsilon)$ con $\epsilon = 0,001$.

Ejercicio 5.4. Ajuste el modelo anterior pero use una distribución a priori $U(0,01, 100)$ para σ en lugar de τ , tal como sugiere Gelman (2006).

Ejercicio 5.5. Modifique algunos de los parámetros de MCMC tales como número de iteraciones (`n.iter`), *thinning* (`n.thin`), etc.

Capítulo 6

Modelos Lineales y Modelos Lineales Generalizados Bayesianos

6.1. Modelos Lineales Bayesianos

Asuma que

$$y_i = \beta_0 + \beta_1 x_{1i} + \dots + \beta_p x_{pi} + \varepsilon_i,$$

donde $i = 1, \dots, n$.

En términos de matrices,

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon},$$

donde $\boldsymbol{\varepsilon} \sim N(\mathbf{0}, \sigma^2 \mathbf{I})$.

La verosimilitud para estos datos está dada por

$$L(\boldsymbol{\beta}, \sigma^2 | \mathbf{X}, \mathbf{y}) = (\sqrt{2\pi}\sigma)^n \exp \left[-\frac{1}{2\sigma^2} (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^t (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \right].$$

Los estimadores de ML están dados por

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^t \mathbf{X})^{-1} \mathbf{X}^t \mathbf{y}, \quad \hat{\sigma}^2 = \frac{1}{n} (\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}})^t (\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}).$$

En el contexto Bayesiano, necesitamos especificar distribuciones a priori para $\boldsymbol{\beta}$ y $\tau = 1/\sigma^2$ (precisión): $\pi(\boldsymbol{\beta})$ y $\pi(\tau)$, independientes. Por lo tanto, la distribución a posteriori de los parámetros en el modelo está dada por

$$p(\boldsymbol{\beta}, \tau | \mathbf{X}, \mathbf{y}) \propto L(\boldsymbol{\beta}, \tau | \mathbf{X}, \mathbf{y}) \pi(\boldsymbol{\beta}) \pi(\tau).$$

La expresión anterior tiene forma cerrada únicamente bajo algunas condiciones en las distribuciones a priori.

Una posibilidad para las distribuciones a priori es asumir que

$$\boldsymbol{\beta} \sim N(c_0, \mathbf{V}_0), \quad \tau \sim \text{gamma}(a_0, b_0),$$

donde c_0 y \mathbf{V}_0 son conocidas. Si \mathbf{V}_0 es diagonal entonces se asumen distribuciones previas independientes para $\boldsymbol{\beta}$. Los parámetros de la *gamma* son ambos conocidos (esta parametrización asume que $E(\tau) = a_0/b_0$ y $\text{Var}(\tau) = a_0/b_0^2$).

Otras ideas de distribuciones a priori se pueden encontrar aquí.

6.1.1. Predicción

Supongamos que queremos predecir \tilde{y} usando $\tilde{\mathbf{x}}$. La predicción Bayesiana se hace usando la *distribución predictiva a posteriori*

$$p(\tilde{y}|\mathbf{y}) = \int p(\tilde{y}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{y})\boldsymbol{\theta}.$$

La forma explícita de $p(\tilde{y}|\mathbf{y})$ usualmente no está disponible pero se puede obtener una muestra de esta o una aproximación. Tanto en JAGS como en INLA se agregan las fila de datos nuevos con la respuesta `y[i]<-NA` para hacer las predicciones respectivas.

A manera de ejemplo vamos a predecir una observación cuando $x = 2$ usando los datos simulados similares a los del capítulo anterior.

```
set.seed(1)
# Datos simulados
n <- 100
x <- 1:n
epsilon <- rnorm(n, 0, 1)
y <- 2-5*x + epsilon

example1 <- data.frame(x = x, y = y)
pred <- data.frame(x=2, y=NA)
combined <- rbind(example1, pred)

# Modelo
modelo <- function()
{
  for (i in 1:n){
    y[i] ~ dnorm(y.hat[i], tau)
```

```

      y.hat[i] <- a + b * x[i]
    }
    # Distribuciones a priori
    a ~ dnorm(0, .0001)
    b ~ dnorm(0, .0001)
    tau <- pow(sigma, -2)
    sigma ~ dunif(0, 100)
  }

# Crear base de datos para el modelo jags
sim.dat.jags <-list(
  n = nrow(combined),
  x = combined$x,
  y = combined$y
)

library(rjags)
library(R2jags)

parametros <- c("a","b", "tau", "sigma", "y")
jags1 <- jags(data = sim.dat.jags,
              inits = NULL,
              parametros,
              n.iter = 1000,
              model.file = modelo,
              n.burnin = 200,
              n.chains = 1,
              n.thin = 1)

```

```

Compiling model graph
  Resolving undeclared variables
  Allocating nodes
Graph information:
  Observed stochastic nodes: 100
  Unobserved stochastic nodes: 4
  Total graph size: 415

```

Initializing model

```
str(jags1$BUGSoutput$sims.list)
```

List of 6

```
$ a      : num [1:800, 1] 1.82 2.22 1.85 1.84 2.31 ...
```



```

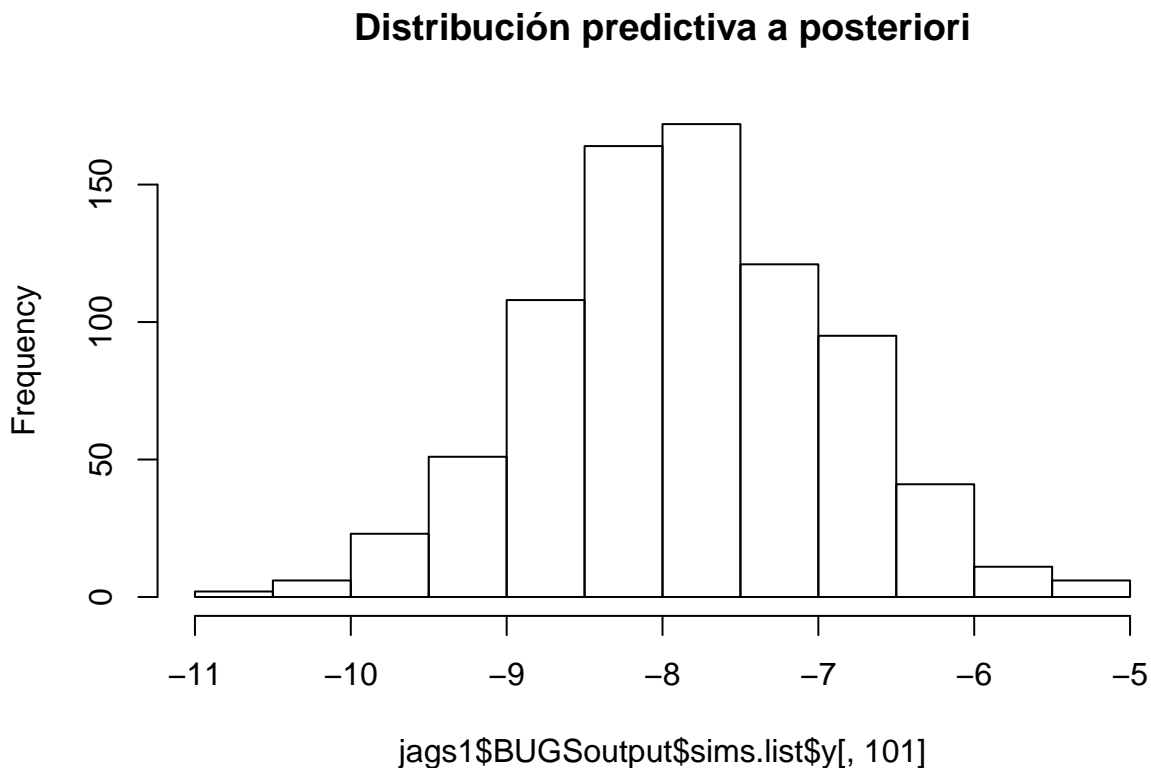
$ b      : num [1:800, 1] -4.99 -5 -4.99 -5 -5 ...
$ deviance: num [1:800, 1] 270 263 267 265 263 ...
$ sigma   : num [1:800, 1] 0.854 0.953 0.888 0.852 0.866 ...
$ tau     : num [1:800, 1] 1.37 1.1 1.27 1.38 1.33 ...
$ y       : num [1:800, 1:101] -3.63 -3.63 -3.63 -3.63 -3.63 ...

```

```
jags1$BUGSoutput$summary["y[101]",]
```

	mean	sd	2.5%	25%	50%	75%
	-7.8506829	0.9275418	-9.6474137	-8.4686465	-7.8604567	-7.2325576
	97.5%					
	-6.0674786					

```
hist(jags1$BUGSoutput$sims.list$y[,101], main='Distribución predictiva a posteriori')
```



6.1.2. Modelo Lineal Mixto

A continuación vamos a ilustrar el ajuste de un modelo lineal mixto (interceptos aleatorios) para los datos de pruebas de aprovechamiento (PIAT) el cual se usó anteriormente. Vamos a ajustar este modelo usando JAGS.

El modelo que vamos a ajustar es el siguiente:

$$y_{ij} = \beta_0 + \beta_1 * agegrp_i + u_i + e_{ij},$$

donde $i = 1, \dots, 89; j = 1, 2, 3$, $u_i \stackrel{i.i.d.}{\sim} N(0, \sigma_u^2)$, $e_{ij} \stackrel{i.i.d.}{\sim} N(0, \sigma_e^2)$, independientes entre sí. Las distribuciones a priori para los parámetros son las siguientes (independientes):

$$\pi(\beta_j) \sim N(0, .0001), \sigma_e \sim U(0, 100), \sigma_u \sim U(0, 100).$$

```
data(reading, package="brinla")
head(reading)

  id agegrp piat
1  1     6.5   18
2  1     8.5   35
3  1    10.5   59
4  2     6.5   18
5  2     8.5   25
6  2    10.5   28

attach(reading)
id_num <- as.numeric(as.character(id))

# Modelo lineal con interceptos aleatorios
mod.mix <- function(){

# Likelihood
  for (i in 1:n){
    piat[i] ~ dnorm(mu[i], tau_e)
    mu[i] <- beta0 + beta1 * agegrp[i] + u[id_num[i]]
  }

# Distribuciones a priori
  beta0 ~ dnorm(0, .0001)
  beta1 ~ dnorm(0, .0001)
  tau_e <- pow(sigma_e, -2)
  sigma_e ~ dunif(0, 100)

  for (j in 1:Ninds){
    u[j] ~ dnorm(0, tau_u) # intercepto aleatorio de individuo
  }
  sigma_u ~ dunif(0, 100) # varianza entre individuos
  tau_u <- 1 / (sigma_u * sigma_u) # precisión
}
```

```

# Crear base de datos para el modelo jags
sim.dat.jags <-list(
  n = nrow(reading),
  id_num = id_num,
  Ninds = length(table(reading$id)),
  agegrp = agegrp,
  piat = piat
)

library(rjags)
library(R2jags)

parametros <- c("beta0","beta1", "tau_e", "tau_u")
jags2 <- jags(data = sim.dat.jags,
  inits = NULL,
  parametros,
  n.iter = 10000,
  model.file = mod.mix,
  n.burnin = 2000,
  n.chains = 1,
  n.thin = 1)

```

```

Compiling model graph
  Resolving undeclared variables
  Allocating nodes
Graph information:
  Observed stochastic nodes: 267
  Unobserved stochastic nodes: 93
  Total graph size: 1359

```

Initializing model

```

# Estadísticas Descriptivas Distribución A Posteriori
jags2$BUGSoutput$summary

```

	mean	sd	2.5%	25%
beta0	-11.55824253	2.257875411	-15.88094973	-13.08783508
beta1	5.03295171	0.252133373	4.53022722	4.86290274
deviance	1776.35123546	17.111339354	1745.58240419	1764.46043643
tau_e	0.02203501	0.002406871	0.01753933	0.02039424
tau_u	0.03457540	0.008925610	0.02110399	0.02843926
	50%	75%	97.5%	

```

beta0      -11.59185380  -10.06891291   -7.07955300
beta1       5.03370690   5.20388381   5.52147536
deviance 1775.51795020 1786.96814569 1812.84679489
tau_e       0.02193137   0.02362521   0.02696339
tau_u       0.03328252   0.03922020   0.05581629

```

```
jags1$BUGSoutput$DIC
```

```
[1] 267.071
```

```
jags1$BUGSoutput$pD
```

```
[1] 2.809354
```

```
# Diagnósticos de convergencia
```

```
library(ggmcmc)
```

```
library(coda)
```

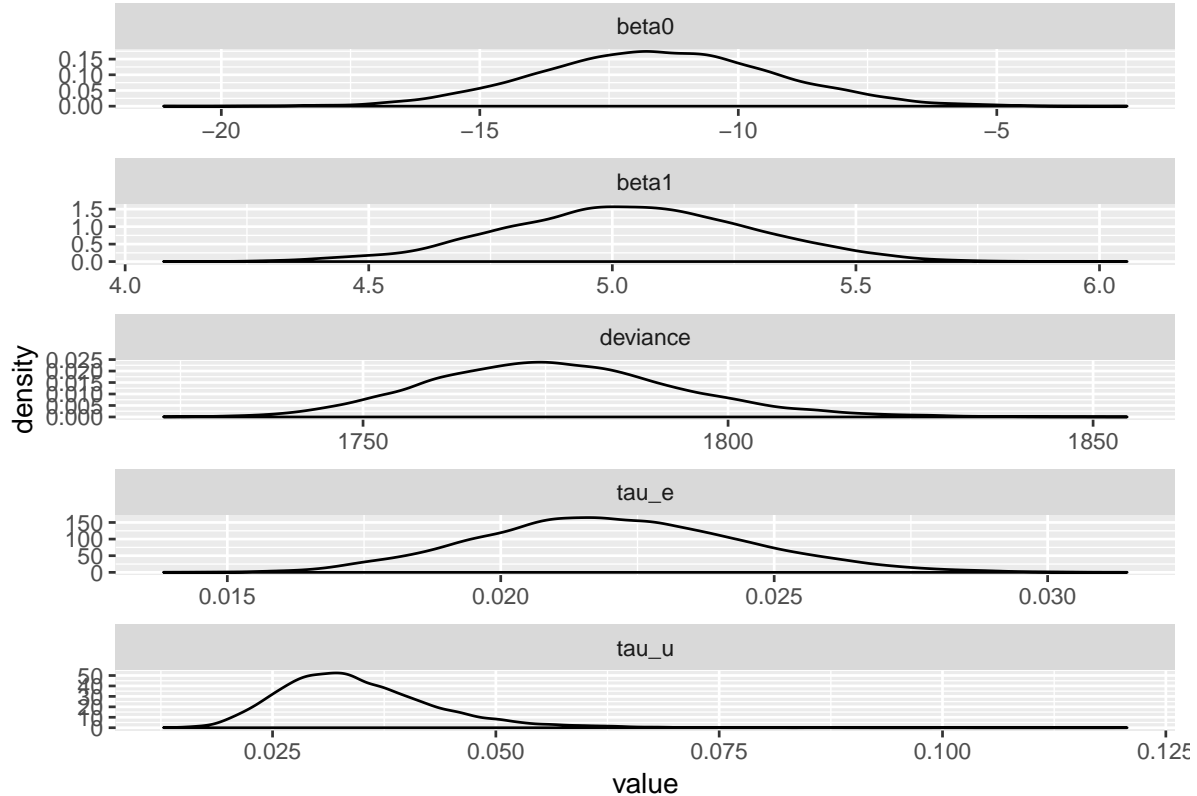
```
jags2.mcmc <- as.mcmc(jags2)
```

```
M <- ggs(jags2.mcmc)
```

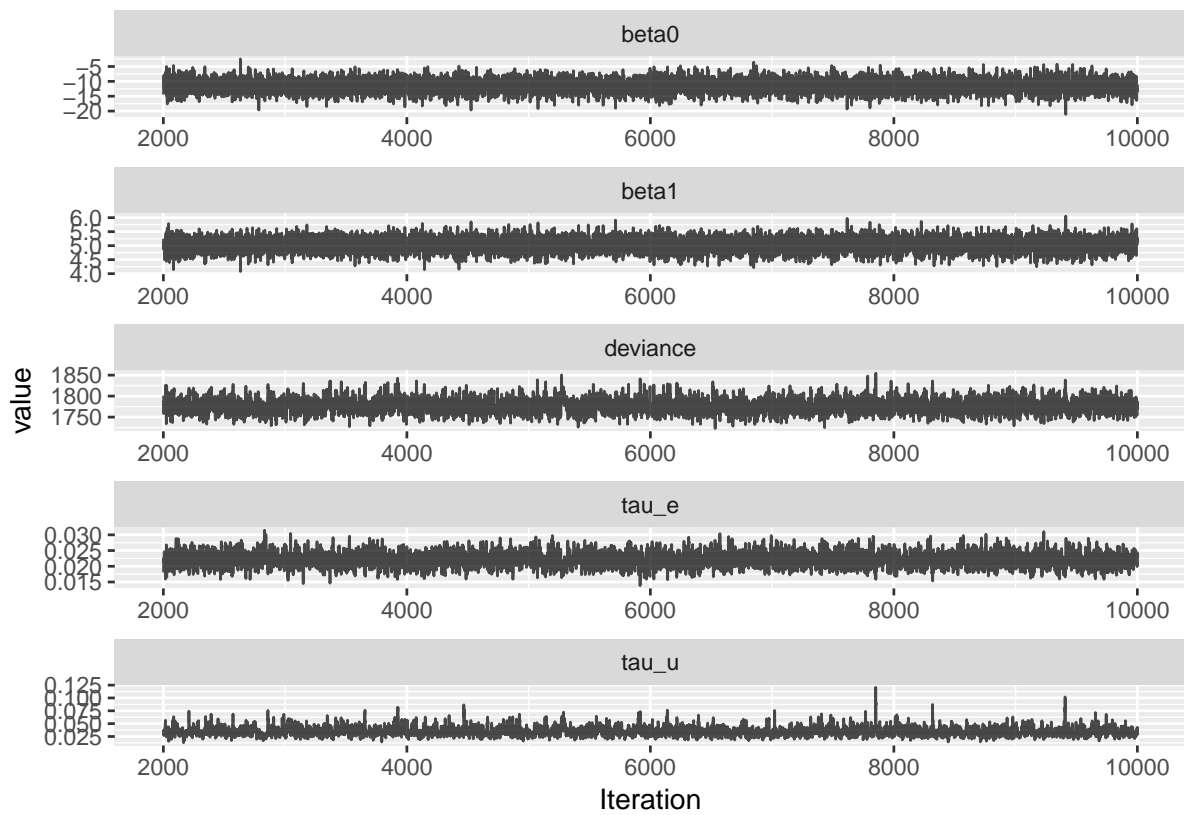
```
# Diagnósticos graficos
```

```
density <- ggs_density(M)
```

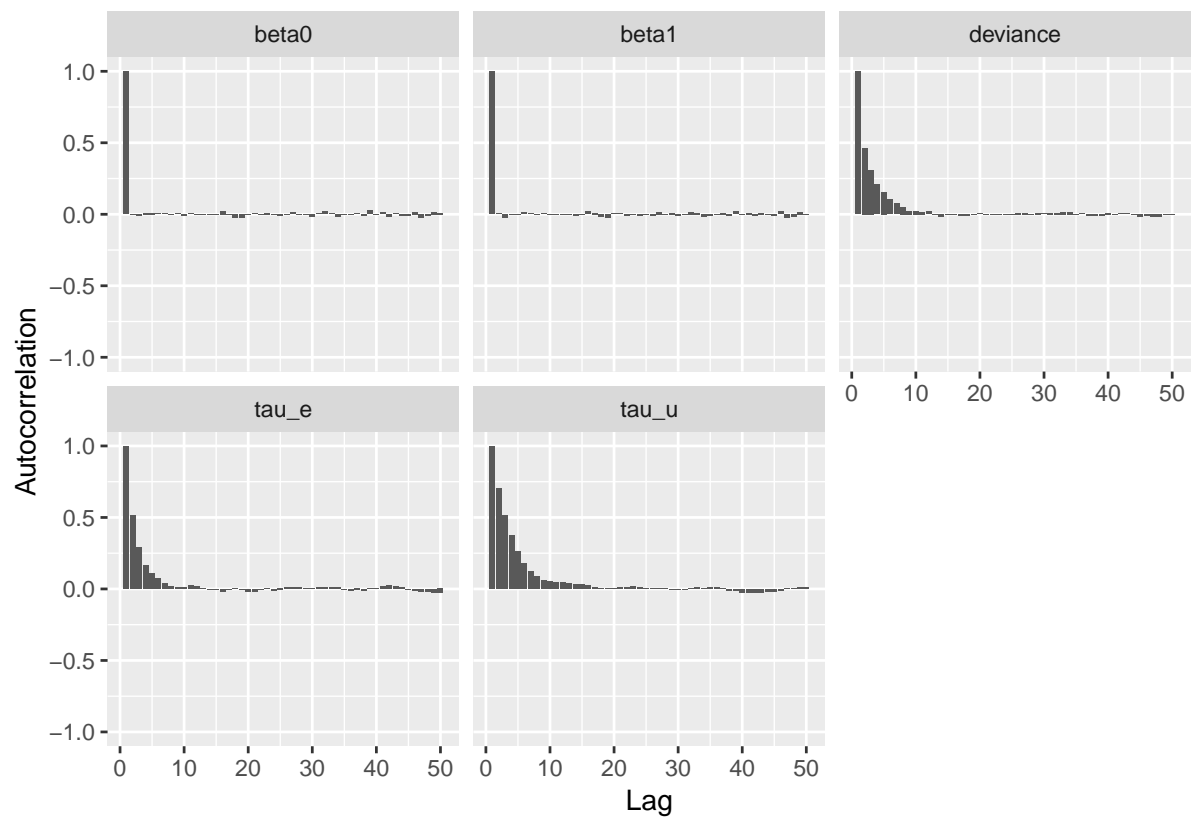
```
density
```



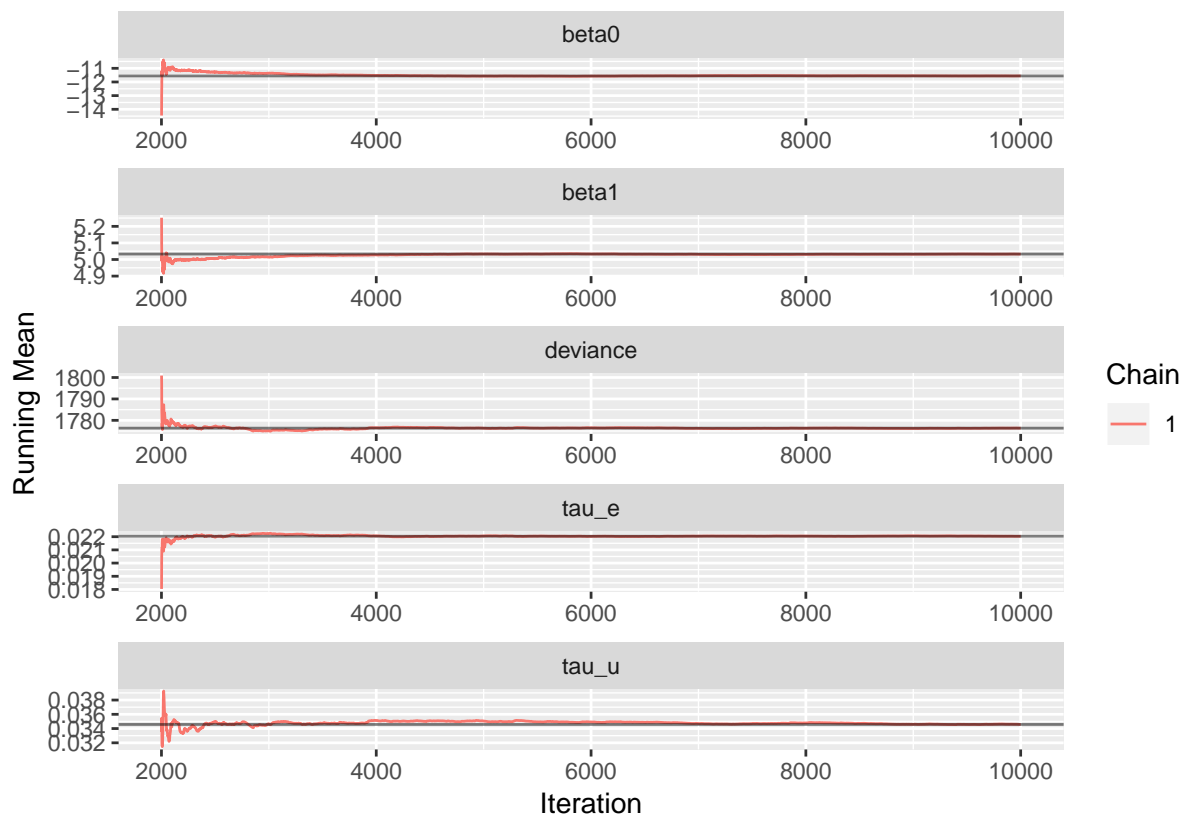
```
trace <- ggs_traceplot(M)
trace
```



```
auto <- ggs_autocorrelation(M)
auto
```



```
run <- ggs_running(M)
run
```



Ejercicio 6.1. Calcule un intervalo de credibilidad del 95 % para la expresión $\sigma_u^2/(\sigma_u^2 + \sigma^2)$.

```
# Código para la práctica
data(reading, package="brinla")
head(reading)
attach(reading)
id_num <- as.numeric(as.character(id))
cagegrp <- agegrp - 8.5 # edad centrada

# Modelo lineal con interceptos y pendientes aleatorias correlacionadas
mod.mix3 <- function(){

# Likelihood
for (i in 1:n){
  piat[i] ~ dnorm(mu[i], tau_e)
  mu[i] <- beta0[id_num[i]] + beta1[id_num[i]]*cagegrp[i]
}

# Priors
mu_int ~ dnorm(0, 0.0001) # media interceptos aleatorios
```

```

mu_slope ~ dnorm(0, 0.0001) # media pendientes aleatorias
sigma_int ~ dunif(0, 100) # desv. est. interceptos
sigma_slope ~ dunif(0, 100) # desv. est. pendientes
rho ~ dunif(-1, 1) # correlación entre interceptos y pendientes
Sigma_B[1, 1] <- pow(sigma_int, 2) # matriz de covarianza efectos aleatorios
Sigma_B[2, 2] <- pow(sigma_slope, 2)
Sigma_B[1, 2] <- rho*sigma_int*sigma_slope
Sigma_B[2, 1] <- Sigma_B[1, 2]
covariance <- Sigma_B[1, 2]
Tau_B[1:2, 1:2] <- inverse(Sigma_B[,])
for (i in 1:Ninds) {
  B_hat[i, 1] <- mu_int
  B_hat[i, 2] <- mu_slope
  B[i, 1:2] ~ dmnorm(B_hat[i, ], Tau_B[,]) # efectos aleatorios bivariados
  beta0[i] <- B[i, 1] # intercepto aleatorio
  beta1[i] <- B[i, 2] # pendiente aleatoria
}
sigma_e ~ dunif(0, 100) # Desv. est. residual
tau_e <- 1/(sigma_e*sigma_e)
}

library(rjags)
library(R2jags)

# Crear base de datos para el modelo jags
sim.dat.jags <- list(
  n = nrow(reading),
  id_num = id_num,
  Ninds = length(table(reading$id)),
  cagegrp = cagegrp,
  piat = piat
)

parametros <- c("beta0", "beta1", "tau_e", "mu_int", "sigma_int",
               "mu_slope", "sigma_slope", "rho", "covariance", "sigma_e")
jags2 <- jags(data = sim.dat.jags,
              inits = NULL,
              parametros,
              n.iter = 10000,
              model.file = mod.mix3,
              n.burnin = 2000,
              n.chains = 1,

```



```
n.thin = 1)
```

6.2. Modelos Lineales Generalizados (MLG) Bayesianos

Los modelos lineales generalizados y generalizados mixtos siguen una formulación similar a la presentada en el Capítulo 1. En la inferencia Bayesiana, sin embargo, se debe especificar las distribuciones a priori para los parámetros en el modelo. En lugar de definir nuevamente estos modelos, vamos a ilustrar el uso de JAGS para ajustar un MLG. En el caso de un MLG mixto se siguen ideas similares a aquellas usadas en los modelos lineales Bayesianos.

6.2.1. Ejemplo: Bajo peso en recién nacidos

Este conjunto de datos corresponde a 189 nacimientos en una clínica obstétrica (Hosmer y Lemeshow, 2004). La variable respuesta es la indicadora LOW para bajo peso del recién nacido (menos de 2,500 gramos). Las variables explicativas son AGE, RACE (1=white, 2=black, 3=other), SMOKE (0=no, 1=si), HT (hipertensión 0/1), UI (irritabilidad uterina, 0/1), FTV (número de visitas al médico durante el primer trimestre) (Wang et al., 2018).

```
data(lowbwt, package="brinla")
head(lowbwt)
```

	LOW	AGE	LWT	RACE	SMOKE	HT	UI	FTV
1	1	28	120	3	1	0	1	0
2	1	29	130	1	0	0	1	2
3	1	34	187	2	1	1	0	0
4	1	25	105	3	0	1	0	0
5	1	25	85	3	0	0	1	0
6	1	27	150	3	0	0	0	0

El modelo que vamos a ajustar es el siguiente:

$$\text{logit}(p_i) = \beta_0 + \beta_1 * AGE_i + \dots + \beta_8 * FTV_i,$$

donde $p_i = E(Y_i)$ es la probabilidad de que un recién nacido tenga bajo peso. La variable respuesta $Y_i \sim \text{Bernoulli}(p_i)$.

En este modelo se asumen distribuciones a priori independientes para los coeficientes del modelo:

$$\pi(\beta_j) \sim N(0, .0001).$$

```
attach(lowbwt)
X <- as.matrix(cbind(rep(1, nrow(lowbwt)), lowbwt[, -1]) ) # matriz diseño

# Modelo lineal con interceptos y pendientes aleatorias correlacionadas
mod.logistic <- function(){

  # Likelihood
  for (i in 1:n){
    LOW[i] ~ dbern(p[i])
    p[i] <- 1/( 1+exp( -inprod(X[i, ], beta[])) )
  }

  # Priors
  for (j in 1:ncols) {
    beta[j] ~ dnorm(0, 0.0001)
  }
}

library(rjags)
library(R2jags)

# Crear base de datos para el modelo jags
sim.dat.jags <- list(
  n = nrow(lowbwt),
  X = X,
  ncols = ncol(X),
  LOW = LOW
)

parametros <- c("beta")
jags3 <- jags(data = sim.dat.jags,
              inits = NULL,
              parametros,
              n.iter = 5000,
              model.file = mod.logistic,
              n.burnin = 1000,
              n.chains = 1,
              n.thin = 1)
```

Compiling model graph
 Resolving undeclared variables

```

Allocating nodes
Graph information:
  Observed stochastic nodes: 189
  Unobserved stochastic nodes: 8
  Total graph size: 3205

```

```

Initializing model

```

```

jags3$BUGSoutput$summary

```

	mean	sd	2.5%	25%	50%
beta[1]	-0.07630662	1.465693046	-2.56109132	-1.05865548	-0.25225513
beta[2]	-0.03008741	0.033660096	-0.09493473	-0.05291370	-0.02879487
beta[3]	-0.01461342	0.007447793	-0.02979207	-0.01975837	-0.01398274
beta[4]	0.51687858	0.241648174	0.05530470	0.34773334	0.51704373
beta[5]	1.12742665	0.431088584	0.29771148	0.84021732	1.11468273
beta[6]	1.97167443	0.747703551	0.59314602	1.45766560	1.93497064
beta[7]	0.90734707	0.481285116	-0.04892484	0.58711942	0.90611140
beta[8]	0.04428503	0.174797156	-0.30943667	-0.06886706	0.04719766
deviance	215.42429359	4.466670478	209.04712713	212.14997902	214.61683564

	75%	97.5%
beta[1]	0.734940212	3.35566009
beta[2]	-0.006894093	0.03339383
beta[3]	-0.009199546	-0.00143641
beta[4]	0.677869874	0.99955526
beta[5]	1.410528725	1.98854518
beta[6]	2.441415674	3.50646170
beta[7]	1.231898870	1.82219300
beta[8]	0.158874704	0.38191830
deviance	217.914383309	225.95384634

```

# Diagnósticos de convergencia

```

```

library(ggmcmc)

```

```

library(coda)

```

```

jags3.mcmc <- as.mcmc(jags3)

```

```

M <- ggs(jags3.mcmc)

```

```

# Diagnósticos graficos

```

```

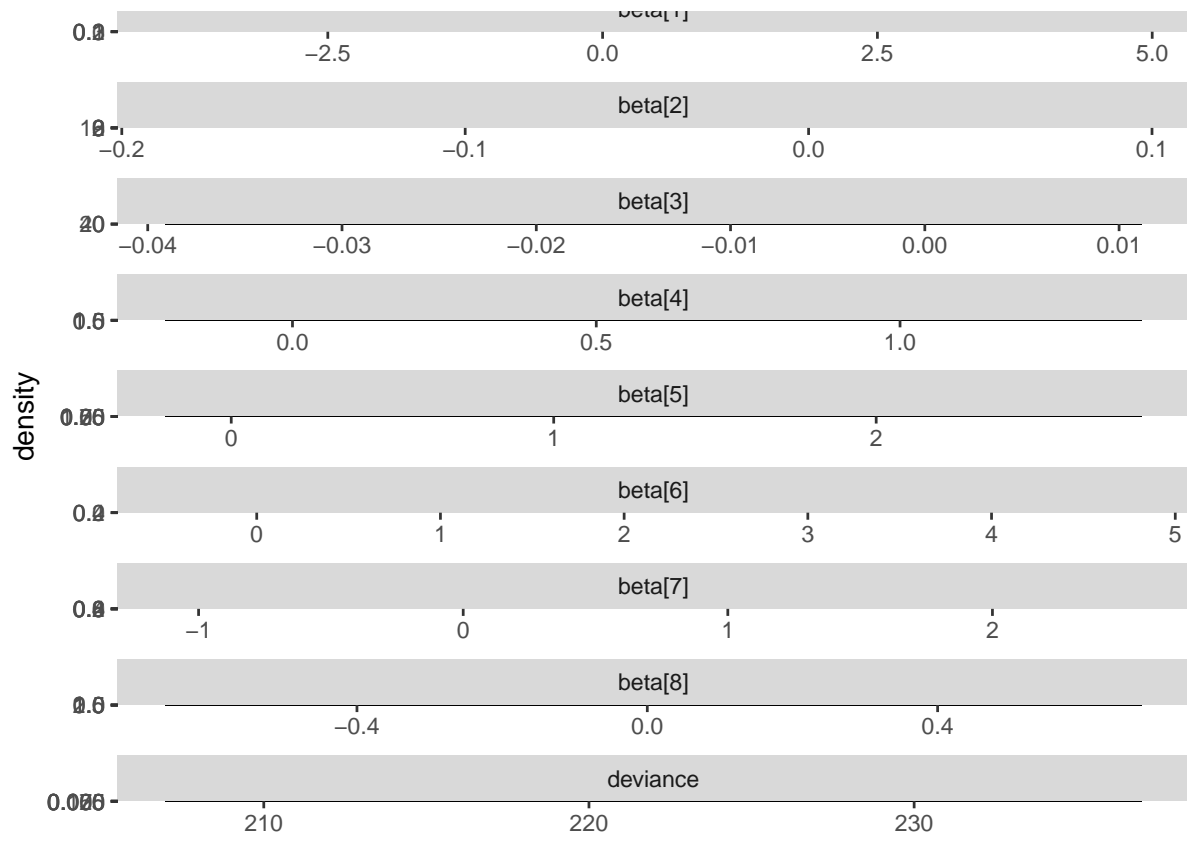
density <- ggs_density(M)

```

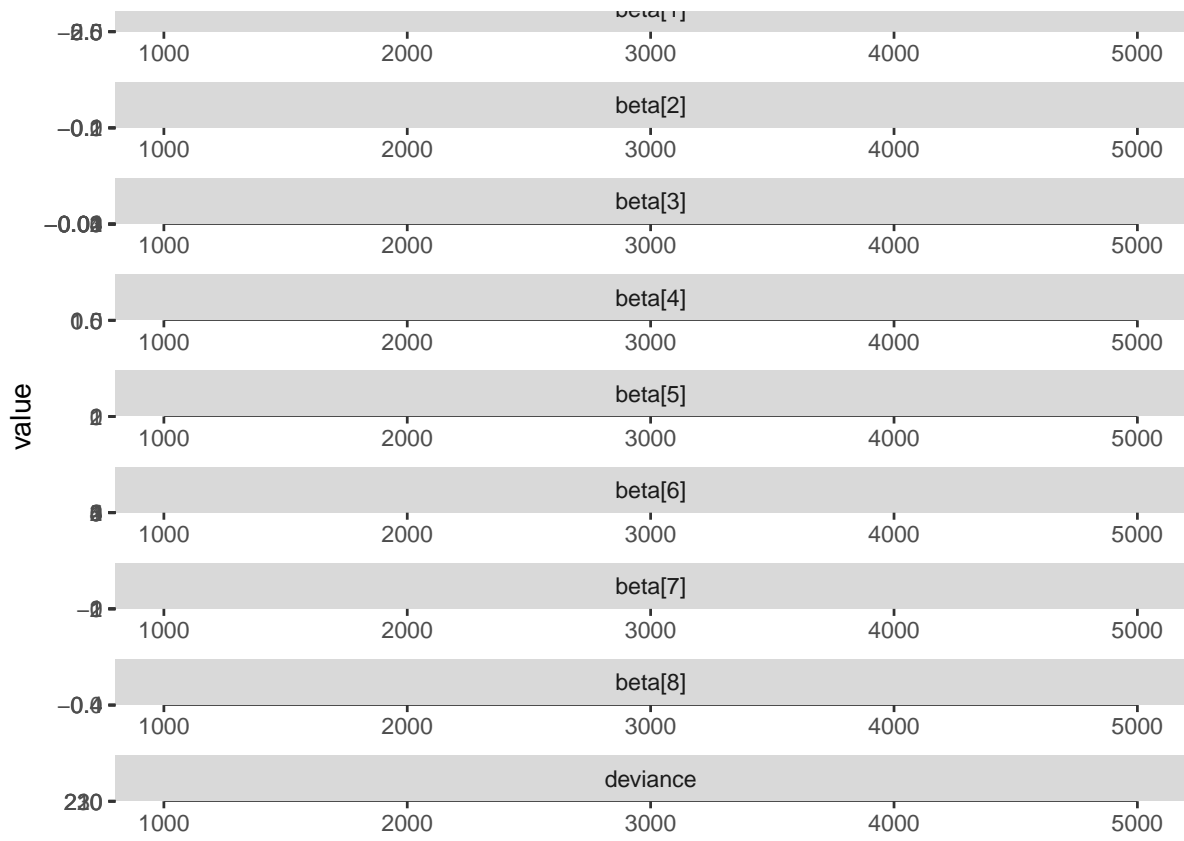
```

density

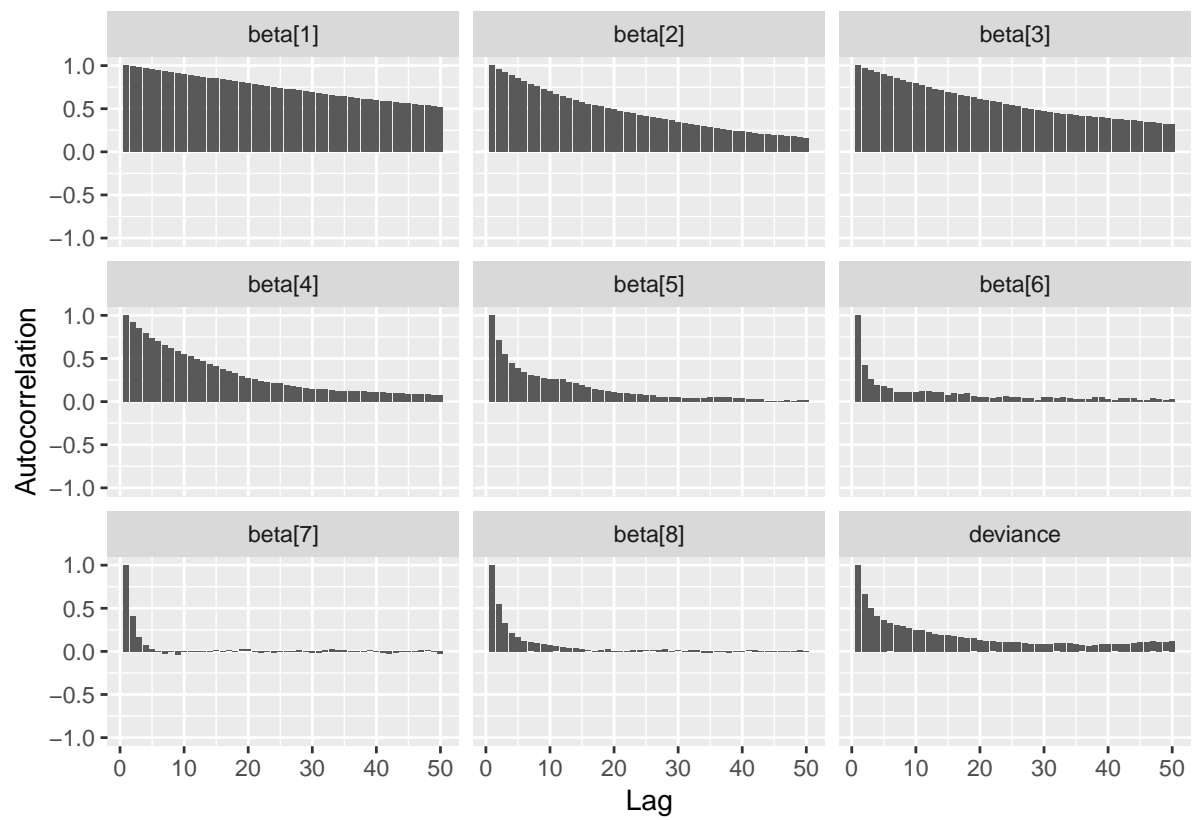
```



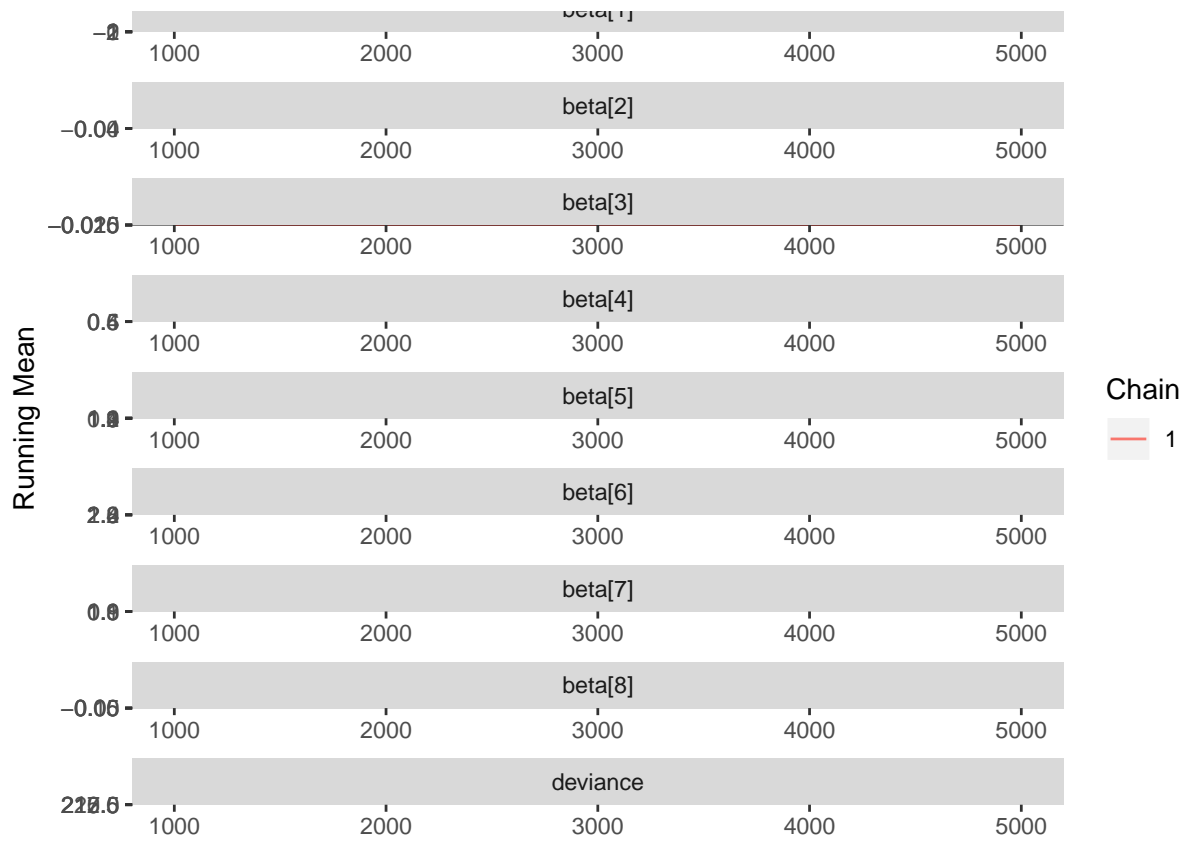
```
trace <- ggs_traceplot(M)
trace
```



```
auto <- ggs_autocorrelation(M)
auto
```



```
run <- ggs_running(M)
run
```



Capítulo 7

Introducción a INLA (Integrated Nested Laplace Approximation)

INLA es un método relativamente reciente para ajustar modelos Bayesianos. Una de las motivaciones de INLA es resolver la dificultad computacional de MCMC en problemas con bastantes datos. En muchas aplicaciones, el proceso de muestreo de la distribución posterior usando MCMC puede tardar demasiado y en muchos casos no es viable con los recursos computacionales existentes.

El método INLA está implementado en la función `inla()` de la librería INLA de R. Para más detalles y ejemplos de cómo usar esta librería visite su página de ayuda.

7.1. Modelos Gaussianos Latentes (MGL)

En esta parte definimos los Modelos Gaussianos Latentes (MGL) que incluyen los modelos anteriores y muchos más existentes en la literatura. Esta definición caracteriza los modelos con ciertos componentes en común y son relevantes dado que el uso de INLA está restringido a este tipo de modelos.

Una formulación más general de un MGL en un modelo jerárquico con tres niveles es el siguiente (Rue and Martino, 2009). El primer nivel asume independencia condicional con la densidad de y_i perteneciente a la familia exponencial (Blangiardo and Cameletti, 2015)

$$\mathbf{y}|\boldsymbol{\theta}, \boldsymbol{\psi} \sim \prod_{i=1}^n p(y_i|\boldsymbol{\theta}, \boldsymbol{\psi}_2), \quad (\text{modelo datos})$$

donde $\boldsymbol{\psi} = (\boldsymbol{\psi}_1, \boldsymbol{\psi}_2)^t$, $\boldsymbol{\psi}_1$ son los hiperparámetros (*hyperparameters*) y $\boldsymbol{\psi}_2$ son los parámetros de ruido (*nuisance*).

En el segundo nivel se asume que $\boldsymbol{\theta}$ dado $\boldsymbol{\psi}_1$ es un *Campo Aleatorio de Markov Gaussiano (GMRF)* (definido más adelante)

$$\boldsymbol{\theta}|\boldsymbol{\psi}_1 \sim N(\mathbf{0}, \boldsymbol{\Sigma}(\boldsymbol{\psi}_1)) \quad (\text{A priori GMRF})$$

En el tercer nivel se asume que $\boldsymbol{\psi} = (\boldsymbol{\psi}_1, \boldsymbol{\psi}_2)^t \sim \pi(\boldsymbol{\psi})$ (Hyperprior).

La distribución a posteriori está dada por

$$p(\boldsymbol{\theta}, \boldsymbol{\psi}|\mathbf{y}) \propto \left\{ \prod_{i=1}^n p(y_i|\boldsymbol{\theta}, \boldsymbol{\psi}_2) \right\} \pi(\boldsymbol{\theta}|\boldsymbol{\psi}_1)\pi(\boldsymbol{\psi})$$

En general, un MGL se puede definir en términos de un predictor lineal con componentes aditivos

$$\eta_i = \alpha + \sum_{m=1}^M \beta_m x_{mi} + \sum_{l=1}^L f_l(z_{li}).$$

Aquí se asume que $\boldsymbol{\theta} = (\alpha, \boldsymbol{\beta}, \mathbf{f}) \sim \text{GMRF}(\boldsymbol{\psi})$.

Note que la regresión lineal clásica, modelos mixtos, modelos jerárquicos y modelos espaciales y espacio-temporales, entre otros, se pueden acomodar acomodando los términos en la definición del componente lineal η_i . Por ejemplo, para la regresión lineal clásica

$$\eta_i = \alpha + \sum_{m=1}^M \beta_m x_{mi},$$

mientras que para el caso de modelos jerárquicos, $f_l(\cdot) \sim N(0, \sigma_f^2)$ con $\sigma_f^2|\boldsymbol{\psi} \sim \pi(\cdot)$.

Ejercicio 7.1. Asuma un modelo lineal Bayesiano $y_i = \beta_0 + \beta_1 x_{1i} + e_i$, donde $e_i \stackrel{i.i.d.}{\sim} N(0, \sigma_e^2)$, y los coeficientes β siguen distribuciones a priori Gaussianas como al comienzo de la sección. $\tau = 1/\sigma_e^2$ sigue una distribución $\text{gamma}(a, b)$, ambos valores conocidos.

Identifique $\boldsymbol{\theta}$, $\boldsymbol{\psi}_1$ y $\boldsymbol{\psi}_2$ en el modelo anterior.

Ejercicio 7.2. Asuma un modelo lineal mixto Bayesiano con interceptos aleatorios $y_i = \beta_0 + \beta_1 x_{1i} + b_{i0} + e_i$, donde $e_i \stackrel{i.i.d.}{\sim} N(0, \sigma_e^2)$, $b_{i0} \stackrel{i.i.d.}{\sim} N(0, \sigma_b^2)$, e_i y b_{i0} independientes, y los coeficientes β siguen distribuciones a priori Gaussianas como al comienzo de la sección. $\tau_e = 1/\sigma_e^2$ y $\tau_b = 1/\sigma_b^2$ siguen una distribución $\text{gamma}(a, b)$, ambos valores conocidos.

Identifique $\boldsymbol{\theta}$, $\boldsymbol{\psi}_1$ y $\boldsymbol{\psi}_2$ en el modelo anterior.

Antes de hablar de INLA como un método para llevar a cabo inferencia Bayesiana, se debe tener en cuenta lo siguiente para que las aproximaciones sean precisas y el método sea eficiente (Wang et al., 2018):

- El número de hiperparámetros $\boldsymbol{\psi}$ debe ser pequeño, típicamente entre 2 y 5, pero no más de 20.
- Cuando el número de observaciones n es grande $\sim 10^4$, $\boldsymbol{\theta}$ debe ser un campo aleatorio de Markov Gaussiano (GMRF en inglés).
 - *Campo Aleatorio de Markov Gaussiano (GMRF)*: $\boldsymbol{\eta}$ es un GMRF si tiene una densidad normal multivariada y además cumple la independencia condicional (*Propiedad de Markov*). La ventaja de tener un GMRF se debe a las matrices de precisión son dispersas lo cual redundo en algoritmos computacionales eficientes. Por ejemplo, si se asume que $\boldsymbol{\eta}$ sigue un proceso autoregresivo de primer orden (AR(1)) la matriz de precisión es una matriz con una banda:

$$\mathbf{Q} = \sigma_{\eta}^{-2} \begin{bmatrix} 1 & -\rho & \cdots & \cdots & \cdots \\ -\rho & 1 + \rho^2 & -\rho & \cdots & \cdots \\ \cdots & -\rho & 1 + \rho^2 & -\rho & \cdots \\ \vdots & \vdots & \vdots & \vdots & -\rho \\ \cdots & \cdots & \cdots & -\rho & 1 \end{bmatrix}$$

- Cada y_i depende de un componente η_i .

7.1.1. Aproximación de Laplace

La idea general es aproximar $\log g(x)$ usando una expansión de Taylor de grado 2 alrededor de la moda \hat{x} (Baio, 2013). Esto es,

$$\log g(x) \approx \log g(\hat{x}) - \frac{1}{2\hat{\sigma}^2}(x - \hat{x})^2,$$

donde $\hat{\sigma}^2 = -1/(\partial^2 \log g(x)/\partial x^2)|_{x=\hat{x}}$.

De esta manera,

$$\int g(x) dx = \int e^{\log g(x)} dx \approx c \int \exp \left[-\frac{(x - \hat{x})^2}{2\hat{\sigma}^2} \right] dx.$$

Por lo tanto, bajo la aproximación de Laplace

$$g(x) \approx N(\hat{x}, \hat{\sigma}^2).$$

Recuerde que la aproximación de Laplace también se mencionó en la aproximación de las integrales involucradas en la verosimilitud de los modelos no lineales mixtos.

7.1.1.1. Ejemplo de aproximación de Laplace (Baio, 2013)

Considere la función de densidad de una distribución χ_k^2 :

$$p(x) = \frac{g(x)}{c} = \frac{x^{k/2-1}e^{-x/2}}{c}.$$

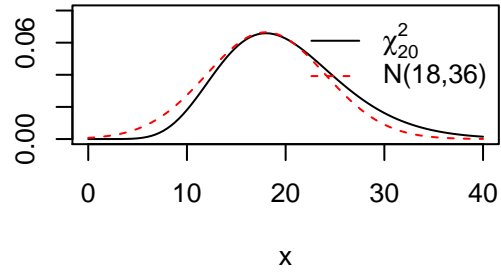
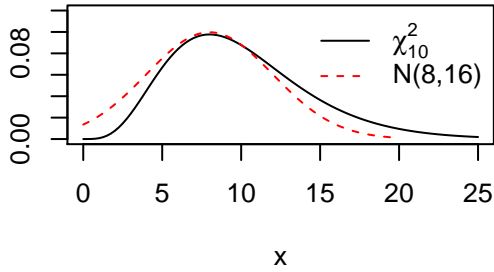
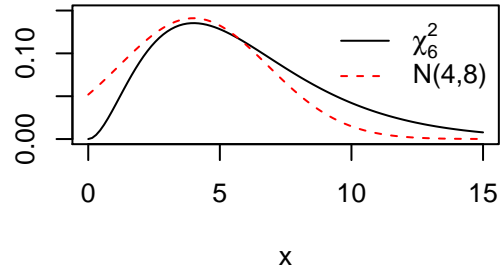
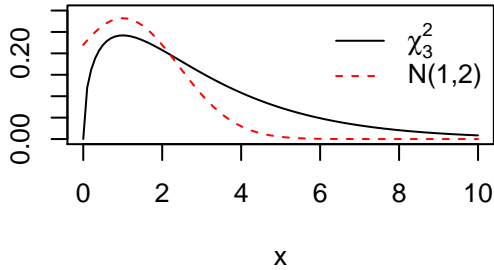
- $l(x) = \log g(x) = \left(\frac{k}{2} - 1\right) \log x - \frac{x}{2}.$
- $l'(x) = \left(\frac{k}{2} - 1\right) x^{-1} - \frac{1}{2}.$
- $l''(x) = -\left(\frac{k}{2} - 1\right) x^{-2}.$

Resolviendo las respectivas ecuaciones se tiene que

$$\hat{x} = k - 2, \quad \hat{\sigma}^2 = 2(k - 2).$$

Por lo tanto,

$$p(x) \approx \tilde{p}(x) = \text{Normal}(k - 2, 2(k - 2)).$$



7.2. *Integrated Nested Laplace Approximation (INLA)*

La idea en la inferencia Bayesiana es obtener una aproximación de las distribuciones a posteriori (Rue and Martino, 2009)

$$p(\theta_j|\mathbf{y}) = \int p(\theta_j, \boldsymbol{\psi}|\mathbf{y}) d\boldsymbol{\psi} = \int p(\boldsymbol{\psi}|\mathbf{y})p(\theta_j|\boldsymbol{\psi}, \mathbf{y}) d\boldsymbol{\psi}.$$

$$p(\psi_k|\mathbf{y}) = \int p(\boldsymbol{\psi}|\mathbf{y}) d\boldsymbol{\psi}_{-k}.$$

Así, el problema se reduce a estimar $p(\boldsymbol{\psi}|\mathbf{y})$ y $p(\theta_j|\boldsymbol{\psi}, \mathbf{y})$. Esto se puede llevar a cabo usando aproximaciones de Laplace (Blangiardo and Cameletti, 2015)

$$p(\boldsymbol{\psi}|\mathbf{y}) \approx \frac{p(\boldsymbol{\psi})p(\boldsymbol{\theta}|\boldsymbol{\psi})p(\mathbf{y}|\boldsymbol{\theta})}{\tilde{p}(\boldsymbol{\theta}|\boldsymbol{\psi}, \mathbf{y})}\Big|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}(\boldsymbol{\psi})} = \tilde{p}(\boldsymbol{\psi}|\mathbf{y}),$$

donde $\hat{\boldsymbol{\theta}}(\boldsymbol{\psi})$ es la moda de $\tilde{p}(\boldsymbol{\theta}|\boldsymbol{\psi}, \mathbf{y})$.

$p(\theta_j|\boldsymbol{\psi}, \mathbf{y})$ se aproxima usando tres métodos (Wang et al., 2018), todos disponibles en R-INLA:

- *Gaussiano*: La aproximación no siempre es buena.
- *Laplace*: Usa aproximación de Laplace para $p(\boldsymbol{\theta}_{-j}|\theta_j, \boldsymbol{\psi}, \mathbf{y})$. Generalmente funciona bien pero es costosa computacionalmente. Recomendada si se requiere más precisión. Una forma de especificar aproximaciones con mayor precisión en la función `inla()` es la siguiente:

```
inla(..., control=inla=list(strategy="laplace",
  int.strategy="grid", dz=0.1, diff.logdens=20),
  num.threads=1)
```

- *Laplace simplificada*: Usa aproximación de Taylor de tercer grado para el numerador y denominador de $\tilde{p}(\theta_j|\boldsymbol{\psi}, \mathbf{y})$. Este es el método por defecto en R-INLA.

Finalmente, la distribución marginal a posteriori de un parámetro se aproxima por integración numérica

$$\tilde{p}(\theta_j|\mathbf{y}) \approx \sum_{h=1}^H \tilde{p}(\theta_j|\psi_h^*, \mathbf{y})\tilde{p}(\psi_h^*|\mathbf{y})\Delta_h.$$

Las aproximaciones de Laplace anidadas para estimar $p(\boldsymbol{\psi}|\mathbf{y})$ y $p(\theta_j|\boldsymbol{\psi}, \mathbf{y})$ junto con esta última aproximación numérica a la integral es lo que le da el nombre de **INLA**.

Finalmente, en los casos en los que se pueden ajustar modelos frecuentistas se recomienda comparar estas estimaciones frecuentistas con las obtenidas usando INLA. De otro lado, cuando MCMC es viable, estas estimaciones también podrían servir como un punto de comparación. Otro recurso para verificar si las estimaciones obtenidas con INLA son razonables es llevar a cabo simulaciones del modelo que quiere ajustar.

A manera de ejemplo, considere el siguiente caso donde INLA podría subestimar un parámetro de interés (Wang et al., 2018). Suponga que $y_i \sim \text{Bernoulli}(p_i)$, donde $p_i = e^{\eta_i} / (1 + e^{\eta_i})$, y $\eta_i \sim N(\mu, \sigma^2)$, independientes, con $\mu = \sigma^2 = 1$. Note que la media a posteriori de σ es 0.012, un valor mucho más pequeño que el verdadero valor 1. Sin embargo, el intercepto se estima bien.

```
# Datos simulados
set.seed(1)
n <- 100
u <- rnorm(n, mean=0, sd=1)
eta <- 1+u
p <- exp(eta)/(1+exp(eta))
y <- rbinom(n, size=1, prob=p)
data <- data.frame(y=y, x=1:n)
library(INLA)
library(brinla)
result <- inla(y~1 + f(x, model="iid"), data=data,
               family="binomial",
               Ntrials=1)
round(bri.hyperpar.summary(result), 4)
```

	mean	sd	q0.025	q0.5	q0.975	mode
SD for x	0.012	0.0124	0.0036	0.0084	0.0431	0.0057

```
round(result$summary.fixed, 4)
```

	mean	sd	0.025quant	0.5quant	0.975quant	mode	kld
(Intercept)	0.9947	0.2253	0.565	0.9903	1.45	0.9813	0

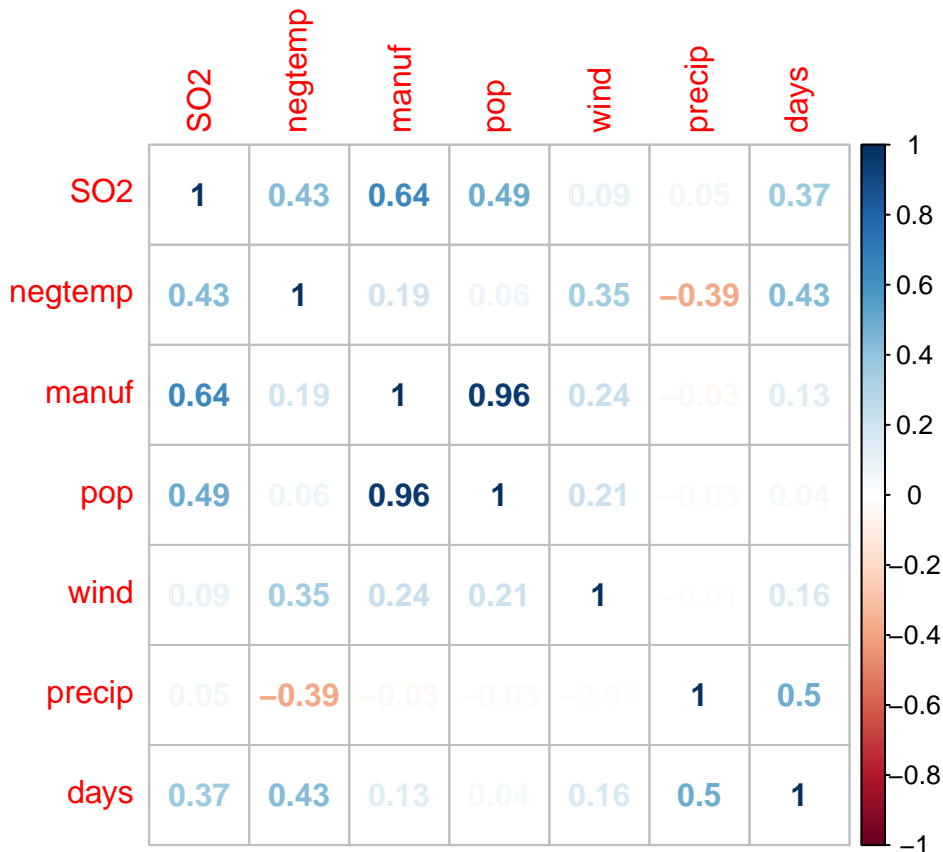
7.3. Modelos Lineales en R-INLA

El siguiente ejemplo ilustra el uso de R-INLA para ajustar modelos lineales (Wang et al., 2018). Vamos a usar la misma base de datos del Capítulo 4. La función de R para ajustar modelos es `inla()`.

La mayoría de ejemplos de estas notas se basan en el libro *Bayesian Regression Modeling with INLA* (Wang et al., 2018). Las bases de datos y códigos de R acompañando este libro se pueden bajar aquí.

Primero, vamos a visualizar las correlaciones entre las variables.

```
# Ejemplo: Regresión -- US air pollution data
data(usair, package = "brinla")
library(corrplot)
M <- cor(usair)
corrplot(M, method="number")
```



Ahora, vamos a ajustar un modelo lineal usando inferencia frecuentista para compararlo con la salida de `inla()`.

```
# Frecuentista
usair.formula1 <- SO2 ~ negtemp + manu + wind + precip + days
usair.lm1 <- lm(usair.formula1, data = usair)
round(coef(summary(usair.lm1)), 4)
```

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	135.7714	50.0610	2.7121	0.0103
negtemp	1.7714	0.6366	2.7824	0.0086
manuf	0.0256	0.0046	5.5544	0.0000
wind	-3.7379	1.9444	-1.9224	0.0627
precip	0.6259	0.3885	1.6111	0.1161

```
days          -0.0571      0.1748 -0.3265    0.7460
```

```
# Error estándar residual
round(summary(usair.lm1)$sigma, 4)
```

```
[1] 15.79
```

A continuación se ajusta el modelo Bayesiano usando INLA. Las distribuciones a priori por defecto para los coeficientes y la precisión son:

$$\beta_j \sim N(0, 10^6), \quad \log(\tau) \sim \log \text{Gamma}(1, 10^{-5}).$$

```
# Bayesiana con INLA
usair.inla1 <- inla(usair.formula1, data = usair,
                   control.compute = list(dic = TRUE, waic=TRUE,
                                           cpo = TRUE))
# summary(usair.inla1)
round(usair.inla1$summary.fixed, 4)
```

	mean	sd	0.025quant	0.5quant	0.975quant	mode	kld
(Intercept)	135.4886	50.1125	36.5084	135.4952	234.2816	135.5117	0
negtemp	1.7690	0.6376	0.5097	1.7690	3.0260	1.7692	0
manuf	0.0256	0.0046	0.0165	0.0256	0.0347	0.0256	0
wind	-3.7229	1.9443	-7.5610	-3.7234	0.1121	-3.7241	0
precip	0.6248	0.3892	-0.1437	0.6249	1.3921	0.6249	0
days	-0.0567	0.1751	-0.4024	-0.0567	0.2886	-0.0567	0

```
round(usair.inla1$summary.hyperpar, 4)
```

	mean	sd	0.025quant	0.5quant
Precision for the Gaussian observations	0.0042	0.001	0.0025	0.0042
	0.975quant	mode		
Precision for the Gaussian observations	0.0064	0.004		

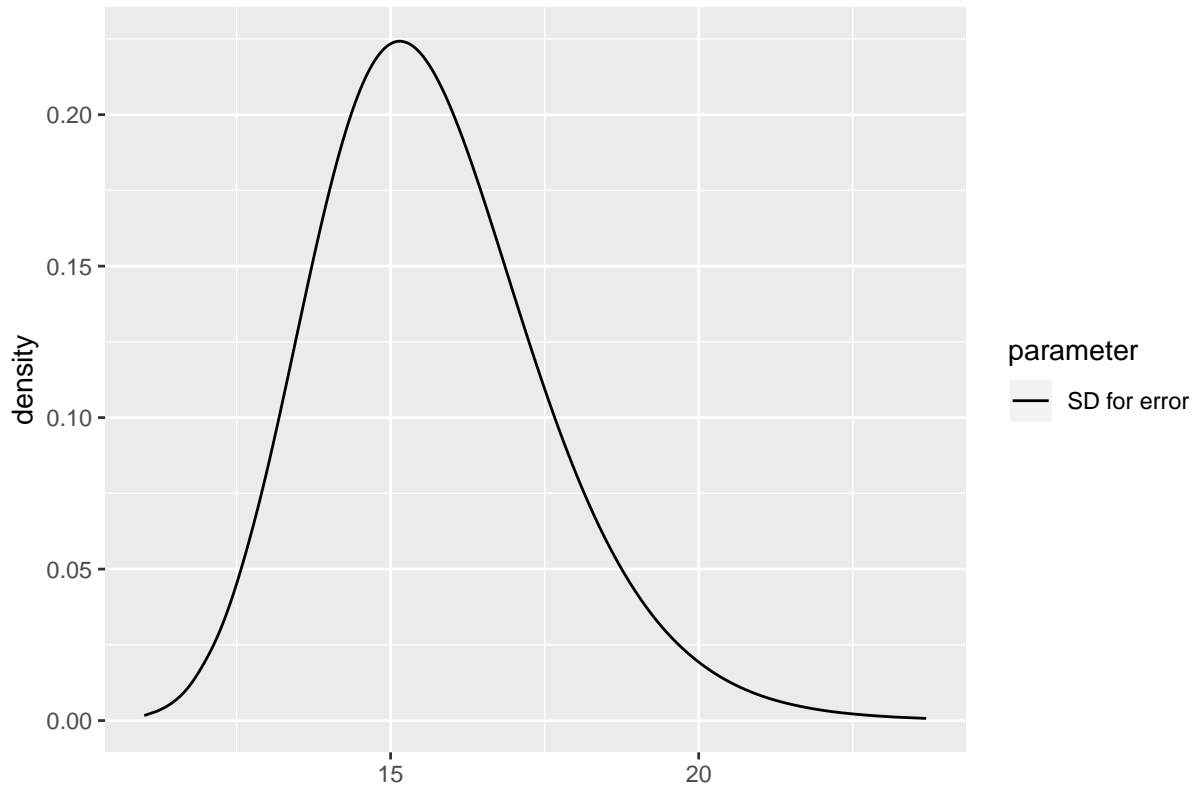
```
# calcula la estimación a posteriori de sigma
inla.emarginal(fun = function(x) 1/sqrt(x),
               marg = usair.inla1$marginals.hyperpar$`Precision for the Gaussian obser
```

```
[1] 15.67484
```

```
round(bri.hyperpar.summary(usair.inla1), 4)
```

	mean	sd	q0.025	q0.5	q0.975
SD for the Gaussian observations	15.6733	1.862	12.5262	15.4929	19.831
	mode				
SD for the Gaussian observations	15.1434				

```
# densidad a posteriori de sigma
bri.hyperpar.plot(usair.inla1)
```



```
# Modificar las distribuciones a priori
usair.inla2 <- inla(usair.formula1, data = usair,
  control.compute = list(dic = TRUE, cpo = TRUE),
  control.fixed = list(mean.intercept = 100,
    prec.intercept = 10^(-2), mean =
      list(negtemp = 2, wind = -3,
        default = 0), prec = 1),
  control.family = list(hyper = list(prec =
    list(prior = "gaussian",
      param = c(0, 1))))))
summary(usair.inla2)
```

Call:

```
c("inla(formula = usair.formula1, data = usair, control.compute = list(dic = TRUE, ",
```

Time used:

Pre-processing	Running inla	Post-processing	Total
1.6099	0.2633	1.1377	3.0110

Fixed effects:

	mean	sd	0.025quant	0.5quant	0.975quant	mode	kld
(Intercept)	102.3903	9.5635	83.6079	102.3920	121.1461	102.3963	0
negtemp	1.3844	0.2127	0.9657	1.3846	1.8018	1.3850	0
manuf	0.0255	0.0040	0.0176	0.0255	0.0334	0.0255	0
wind	-2.9520	0.8034	-4.5299	-2.9521	-1.3749	-2.9523	0
precip	0.4408	0.2469	-0.0468	0.4412	0.9254	0.4420	0
days	0.0406	0.0900	-0.1360	0.0404	0.2181	0.0400	0

The model has no random effects

Model hyperparameters:

	mean	sd	0.025quant	0.5quant
Precision for the Gaussian observations	0.0053	0.0011	0.0035	0.0053
	0.975quant	mode		
Precision for the Gaussian observations	0.0076	0.0051		

Expected number of effective parameters(std dev): 4.327(0.0729)

Number of equivalent replicates : 9.476

Deviance Information Criterion (DIC) ...: 347.51

Effective number of parameters: 5.236

Marginal log-Likelihood: -197.51

CPO and PIT are computed

Posterior marginals for linear predictor and fitted values computed

```
## Predicción en lm
new.data <- data.frame(negtemp = c(-50, -60, -40),
                      manuf = c(150, 100, 400),
                      pop = c(200, 100, 300),
                      wind = c(6, 7, 8),
                      precip = c(10, 30, 20),
                      days = c(20, 100, 40))
predict(usair.lm1, new.data, se.fit = TRUE)
```

\$fit

	1	2	3
	33.72743	18.94993	55.47696

\$se.fit

	1	2	3
--	---	---	---

```
14.936928 5.329492 17.639438
```

```
$df  
[1] 35
```

```
$residual.scale  
[1] 15.78998
```

```
## Prediction in inla  
usair.combined <- rbind(usair, data.frame(SO2 = c(NA, NA, NA), new.data))  
usair.link <- c(rep(NA, nrow(usair)), rep(1, nrow(new.data)))  
usair.inla1.pred <- inla(usair.formula1, data = usair.combined, control.predictor = li  
usair.inla1.pred$summary.fitted.values[(nrow(usair)+1):nrow(usair.combined),]
```

	mean	sd	0.025quant	0.5quant	0.975quant
fitted.Predictor.42	33.65311	14.949170	4.172477	33.65537	63.12941
fitted.Predictor.43	18.92731	5.333553	8.409718	18.92800	29.44414
fitted.Predictor.44	55.40395	17.657441	20.583797	55.40620	90.22169

	mode
fitted.Predictor.42	33.65962
fitted.Predictor.43	18.92931
fitted.Predictor.44	55.41043

```
# variable selection with AIC  
library(MASS)  
usair.step <- stepAIC(usair.lm1, trace = FALSE)  
usair.step$anova
```

Stepwise Model Path
Analysis of Deviance Table

Initial Model:
SO2 ~ negtemp + manu + wind + precip + days

Final Model:
SO2 ~ negtemp + manu + wind + precip

	Step	Df	Deviance	Resid. Df	Resid. Dev	AIC
1				35	8726.322	231.7816
2	- days	1	26.57448	36	8752.897	229.9063

```
# Final multiple regression model  
usair.formula2 <- SO2 ~ negtemp + manu + wind + precip  
usair.lm2 <- lm(usair.formula2, data = usair)  
round(coef(summary(usair.lm2)), 4)
```

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	123.1183	31.2907	3.9347	0.0004
negtemp	1.6114	0.4014	4.0148	0.0003
manuf	0.0255	0.0045	5.6150	0.0000
wind	-3.6302	1.8923	-1.9184	0.0630
precip	0.5242	0.2294	2.2852	0.0283

Final best model: the smallest DIC (=341.26) in all subsets regression

```
usair.inla3 <- inla(usair.formula2,
  data = usair,
  control.compute = list(dic = TRUE, cpo = TRUE))
round(usair.inla3$summary.fixed, 4)
```

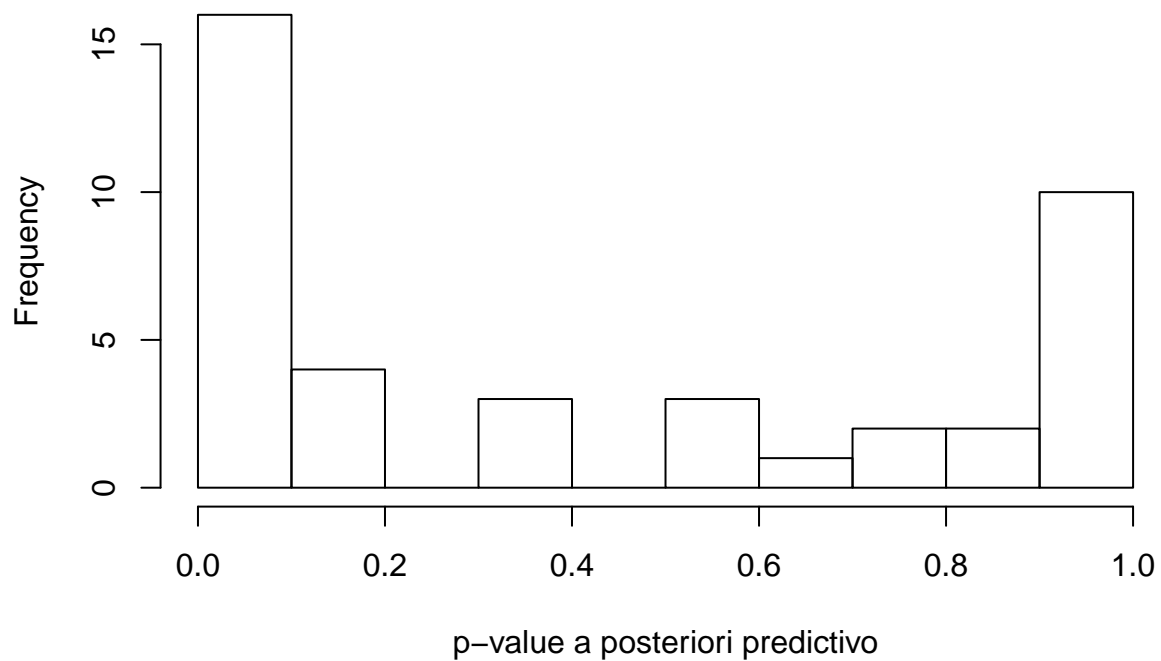
	mean	sd	0.025quant	0.5quant	0.975quant	mode	kld
(Intercept)	122.9364	31.3062	61.1112	122.9405	184.6455	122.9509	0
negtemp	1.6102	0.4019	0.8166	1.6102	2.4025	1.6103	0
manuf	0.0255	0.0045	0.0165	0.0255	0.0344	0.0255	0
wind	-3.6168	1.8919	-7.3510	-3.6172	0.1141	-3.6179	0
precip	0.5239	0.2298	0.0702	0.5239	0.9769	0.5240	0

```
c(usair.inla1$dic$dic, usair.inla3$dic$dic)
```

```
[1] 350.7161 348.6329
```

Posterior Predictive Checking

```
usair.inla3.pred <- inla(usair.formula2,
  data = usair,
  control.predictor = list(link = 1, compute =
    TRUE))
post.predicted.pval <- vector(mode = "numeric", length = nrow(usair))
for(i in (1:nrow(usair))) {
  post.predicted.pval[i] <- inla.pmarginal(q=usair$S02[i],
    marginal =
      usair.inla3.pred$marginals.fitted.values[[i]])
}
hist(post.predicted.pval, main="",
  breaks = 10, xlab="p-value a posteriori predictivo")
```



```
# Validación cruzada
```

```
sum(usair.inla3$cpo$failure)
```

```
[1] 0
```

```
par(mfrow = c(1, 2))
```

```
hist(usair.inla3$cpo$pit, main="", breaks = 10, xlab = "PIT")
```

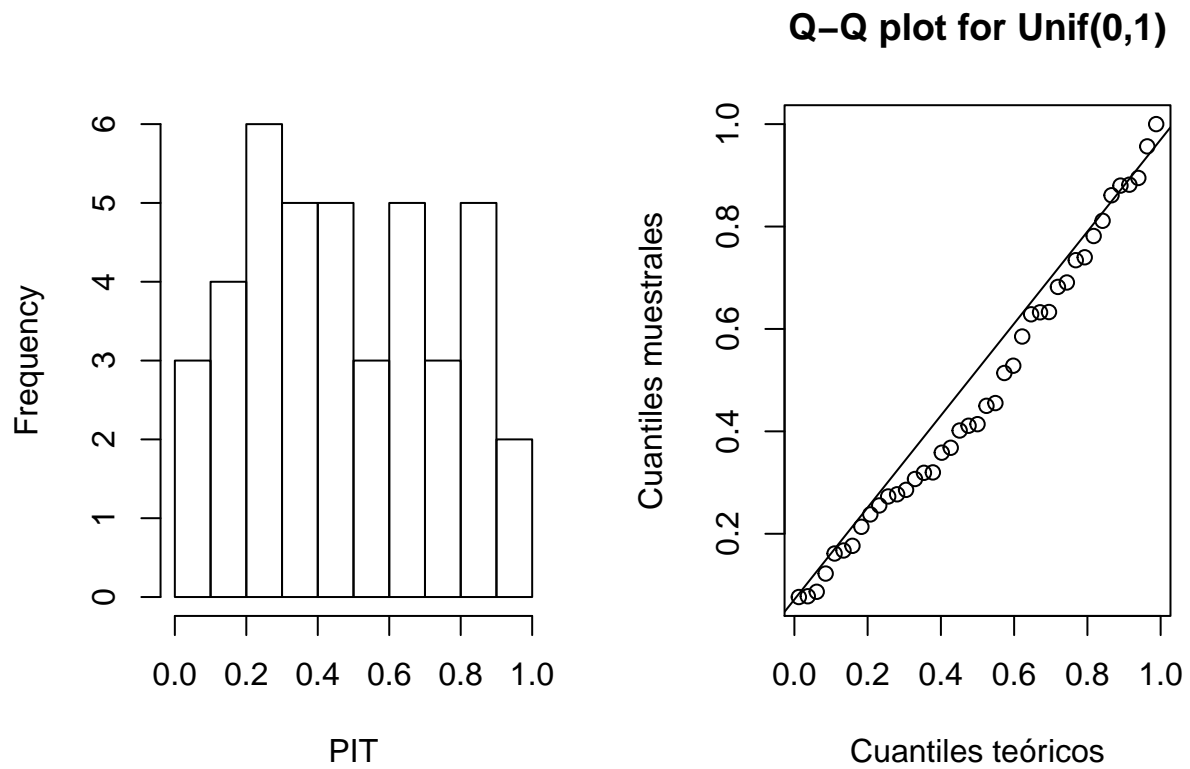
```
qqplot(qunif(ppoints(length(usair.inla3$cpo$pit))),
```

```
      usair.inla3$cpo$pit, main = "Q-Q plot for Unif(0,1)",
```

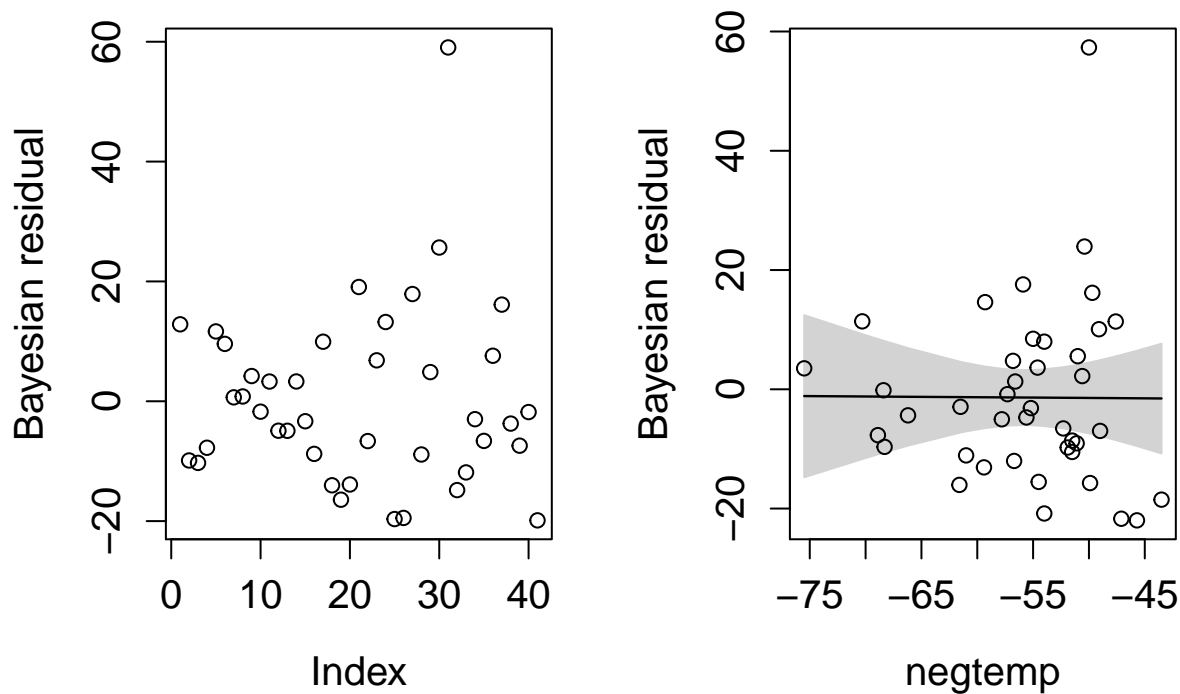
```
      xlab = "Cuantiles teóricos", ylab = "Cuantiles muestrales")
```

```
qqline(usair.inla3$cpo$pit, distribution = function(p) qunif(p),
```

```
      prob = c(0.1, 0.9))
```



```
## Residuales Bayesianos
par(mfrow=c(1,2))
bri.lmresid.plot(usair.inla3)
bri.lmresid.plot(usair.inla3, usair$negtemp,
                  xlab = "negtemp", smooth =TRUE)
```



7.4. Modelos Lineales Mixtos en R-INLA

Para este ejemplo vamos usar los datos de contenido de nitrógeno de pastos (*reeds*) en diferentes localizaciones de Inglaterra (Wang et al., 2018). El modelo que se ajusta es el siguiente:

$$y_{ij} = \alpha + u_i + e_{ij},$$

donde $i = 1, 2, 3; j = 1, \dots, 5$, u_i y e_{ij} se distribuyen normalmente con media 0 y varianzas σ_u^2 y σ_e^2 , respectivamente. σ_u^2 y σ_e^2 son hiperparámetros.

```
data(reeds, package="brinla")
summary(reeds)
```

```
site    nitrogen
A:5   Min.    :2.350
B:5   1st Qu.:2.620
C:5   Median  :3.060
      Mean    :3.029
      3rd Qu.:3.265
      Max.    :3.930
```

```
library(lme4)
mmod <- lmer(nitrogen ~ 1+(1|site), reeds)
summary(mmod)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: nitrogen ~ 1 + (1 | site)
Data: reeds
```

```
REML criterion at convergence: 13
```

```
Scaled residuals:
```

Min	1Q	Median	3Q	Max
-1.2208	-0.7542	-0.2632	0.9136	1.6116

```
Random effects:
```

Groups	Name	Variance	Std.Dev.
site	(Intercept)	0.18724	0.4327
Residual		0.08549	0.2924

```
Number of obs: 15, groups: site, 3
```

```
Fixed effects:
```

	Estimate	Std. Error	t value
(Intercept)	3.029	0.261	11.61

```
library(INLA); library(brinla)
formula <- nitrogen ~ 1 + f(site, model="iid")
imod <- inla(formula,family="gaussian", data = reeds)
summary(imod)
```

```
Call:
```

```
"inla(formula = formula, family = \"gaussian\", data = reeds)"
```

```
Time used:
```

Pre-processing	Running inla	Post-processing	Total
1.4592	0.3035	0.2424	2.0051

```
Fixed effects:
```

	mean	sd	0.025quant	0.5quant	0.975quant	mode	kld
(Intercept)	3.0293	0.1402	2.7587	3.0293	3.3004	3.0293	1e-04

```
Random effects:
```

Name	Model
------	-------

```
site IID model
```

Model hyperparameters:

	mean	sd	0.025quant	0.5quant
Precision for the Gaussian observations	13.25	5.203	5.471	12.49
Precision for site	20.35	25.119	1.934	12.81
	0.975quant	mode		
Precision for the Gaussian observations	25.59	10.940		
Precision for site	84.75	5.094		

Expected number of effective parameters(std dev): 1.243(0.5604)

Number of equivalent replicates : 12.07

Marginal log-Likelihood: -20.88

Note que la opción `model="iid"` define el tipo de modelo/estructura para las variables latentes. Una lista completa con la descripción de otras opciones se puede encontrar aquí. La función `inla()` permite acomodar diferentes modelos, varianzas de efectos aleatorios diferentes o varianza residual por un factor, efectos aleatorios con una matriz \mathbf{Z} , efectos espaciales, campos Gaussianos, entre otros escenarios. El enlace anterior y la página de INLA tiene herramientas para acomodar diferentes modelos.

```
# Estadísticas descriptivas a posteriori
inla.contrib.sd(imod)$hyper
```

	mean	sd	2.5%	97.5%
sd for the Gaussian observations	0.2899021	0.05779684	0.1980458	0.4246160
sd for site	0.3138384	0.16176737	0.1095412	0.6977975

```
bri.hyperpar.summary(imod)
```

	mean	sd	q0.025	q0.5
SD for the Gaussian observations	0.2907869	0.05832998	0.1981704	0.2828316
SD for site	0.3131306	0.15739695	0.1089296	0.2790943
	q0.975	mode		
SD for the Gaussian observations	0.4260810	0.2664690		
SD for site	0.7136871	0.2217604		

```
# Media y mode incorrectas (no invariantes a transformación)
invsqrt <- function(x) 1/sqrt(x)
sdt <- invsqrt(imod$summary.hyperpar[,-2])
row.names(sdt) <- c("SD of epsilon", "SD of site")
```



```
sdt
```

```
              mean 0.025quant  0.5quant 0.975quant      mode
SD of epsilon 0.2747081  0.4275265 0.2829383  0.1976649 0.3023336
SD of site    0.2216854  0.7191411 0.2794259  0.1086254 0.4430831
```

```
# Media y moda correctas
```

```
prec.site <- imod$marginals.hyperpar$"Precision for site"
prec.epsilon <- imod$marginals.hyperpar$"Precision for the Gaussian observations"
c(epsilon=inla.emarginal(invsqrt,prec.epsilon),
  site=inla.emarginal(invsqrt,prec.site))
```

```
epsilon      site
0.2908564 0.3135427
```

```
sigma.site <- inla.tmarginal(invsqrt, prec.site)
sigma.epsilon <- inla.tmarginal(invsqrt, prec.epsilon)
c(epsilon=inla.mmarginal(sigma.epsilon),
  site=inla.mmarginal(sigma.site))
```

```
epsilon      site
0.2664770 0.2217529
```

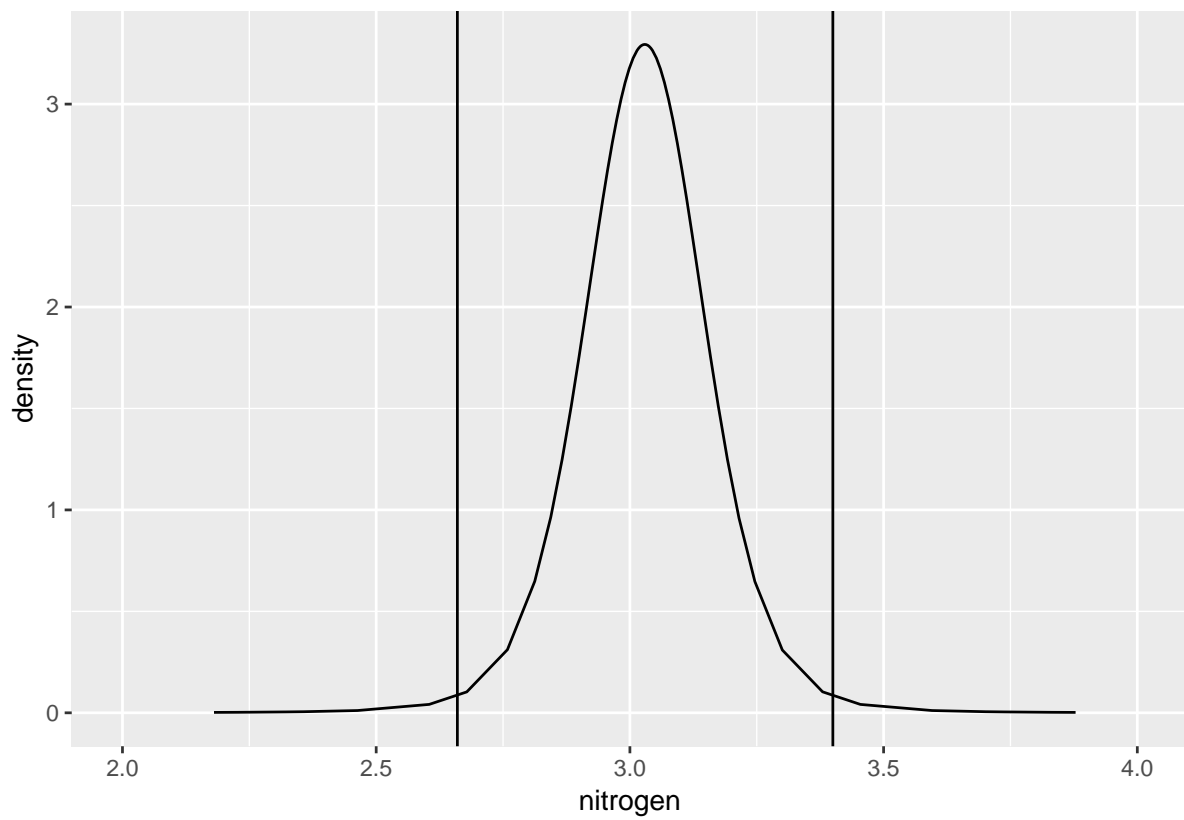
```
# Intervalo de credibilidad para el coef. de correlación intraclase
```

```
sampvars <- 1/inla.hyperpar.sample(1000,imod)
sampicc <- sampvars[,2]/(rowSums(sampvars))
quantile(sampicc, c(0.025,0.5,0.975))
```

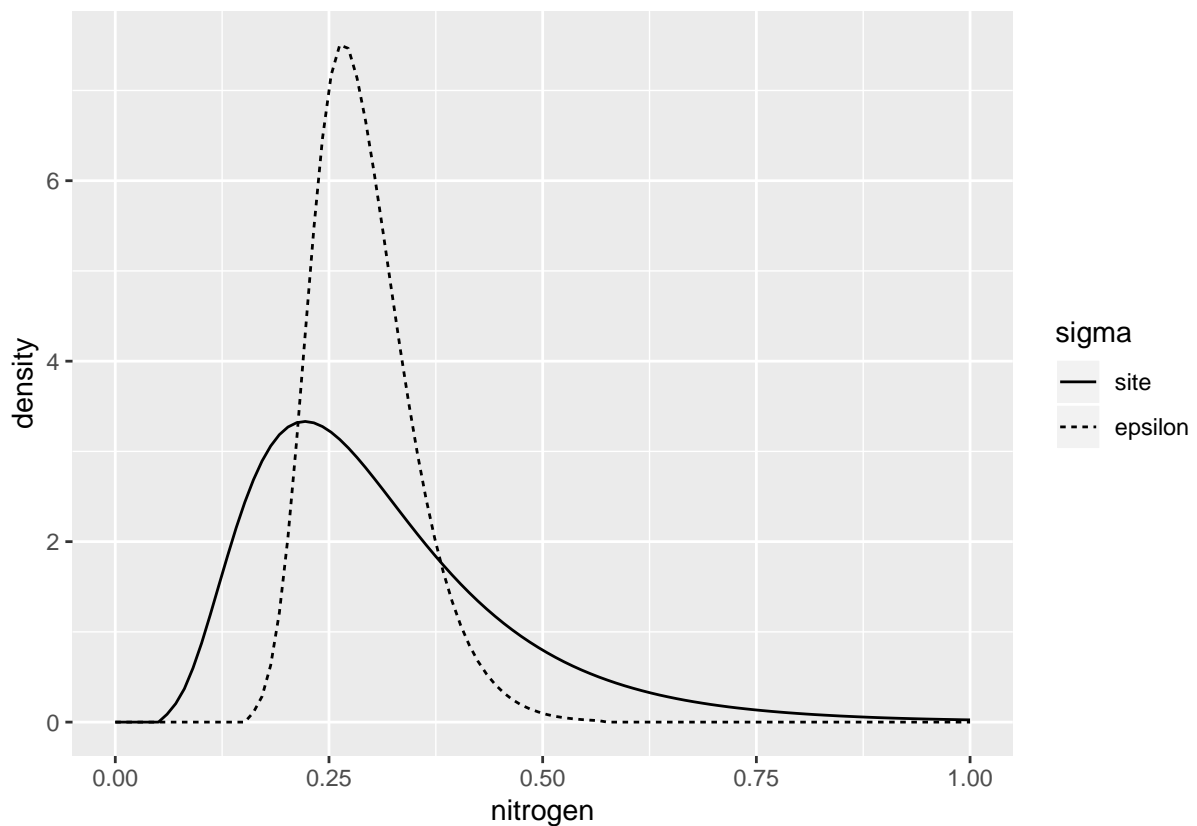
```
      2.5%      50%      97.5%
0.09293244 0.49476617 0.88444498
```

```
# Gráficas de distribuciones a posteriori
```

```
alpha <- data.frame(imod$marginals.fixed[[1]])
library(ggplot2)
ggplot(alpha, aes(x,y)) + geom_line() +
  geom_vline(xintercept = c(2.66, 3.40)) +
  xlim(2,4)+xlab("nitrogen")+ylab("density")
```



```
x <- seq(0,1,len=100)
d1 <- inla.dmarginal(x, sigma.site)
d2 <- inla.dmarginal(x, sigma.epsilon)
rdf <- data.frame(nitrogen=c(x,x),sigma=gl(2,100,
      labels=c("site","epsilon")),density=c(d1,d2))
ggplot(rdf, aes(x=nitrogen, y=density, linetype=sigma))+
  geom_line()
```

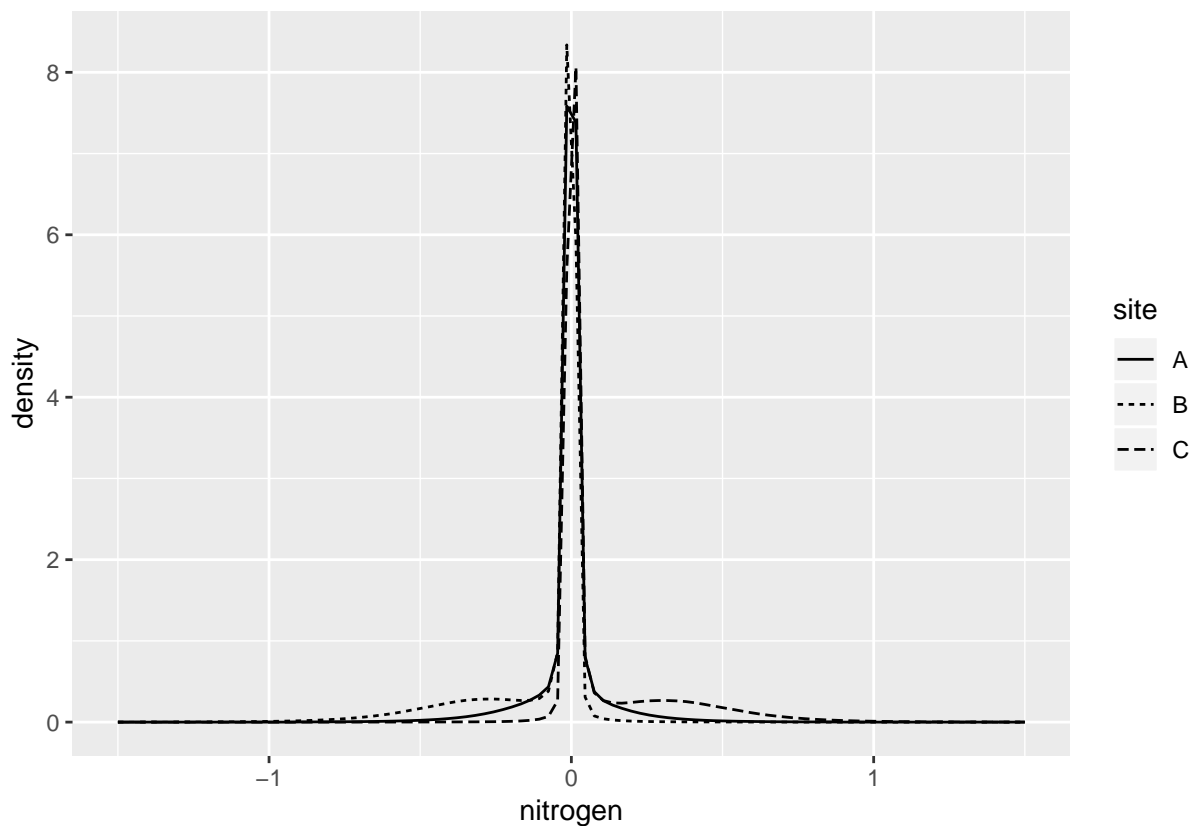


```
# Más fácil!
bri.hyperpar.plot(imod)
```

Es posible especificar otra distribución a priori no incluída en la función `inla()`.

Algo que podemos hacer es extraer las distribuciones a posteriori de los sitios.

```
reff <- imod$marginals.random
x <- seq(-1.5,1.5,len=100)
d1 <- inla.dmarginal(x, reff$site[[1]])
d2 <- inla.dmarginal(x, reff$site[[2]])
d3 <- inla.dmarginal(x, reff$site[[3]])
rdf <- data.frame(nitrogen=x,density=c(d1,d2,d3),site=gl(3,100,
  labels=LETTERS[1:4]))
ggplot(rdf, aes(x=nitrogen, y=density, linetype=site))+geom_line()
```



Las gráficas anteriores se obtienen de manera más directa con el siguiente comando.

```
bri.random.plot(imod)
```

Finalmente, el investigador quizás esté interesado en estimar la probabilidad de que el sitio C tenga más nitrógeno que el sitio A.

```
imod <- inla(formula,family="gaussian", data = reeds,
             control.compute=list(config=TRUE))
psamp <- inla.posterior.sample(n=1000, imod)
psamp[[1]]
```

\$hyperpar

Precision for the Gaussian observations

14.025895

Precision for site

6.710109

\$latent

sample1

Predictor:01 3.04653099

Predictor:02 3.05311933

Predictor:03 3.04712953

```
Predictor:04 3.05195355
Predictor:05 3.04900951
Predictor:06 2.57970293
Predictor:07 2.58257202
Predictor:08 2.57875901
Predictor:09 2.57994773
Predictor:10 2.57947470
Predictor:11 3.48815524
Predictor:12 3.49618084
Predictor:13 3.49392141
Predictor:14 3.49484076
Predictor:15 3.48838775
site:A      0.02043627
site:B     -0.44974551
site:C      0.46297369
(Intercept) 3.03027911
```

```
$logdens
```

```
$logdens$hyperpar
```

```
[1] -6.203985
```

```
$logdens$latent
```

```
[1] 71.19063
```

```
$logdens$joint
```

```
[1] 64.98664
```

```
lvsamp <- t(sapply(psamp, function(x) x$logdens))
colnames(lvsamp) <- row.names(psamp[[1]]$latent)
mean(lvsamp[, 'site:C'] > lvsamp[, 'site:A'])
```

```
[1] 0.622
```

Ejercicio 7.3. Compare las estimaciones frecuentistas con las Bayesianas obtenidas anteriormente con INLA.

Ejercicio 7.4. Use los comandos a continuación para ajustar el modelo lineal con interceptos aleatorios para los datos de nitrógeno usando una distribución a priori débilmente informativa (*penalized complexity prior*) (Wang et al., 2018). Compare estos resultados (estimaciones y gráficas) con los obtenidos con INLA usando las distribuciones previas por defecto.

```

sdres <- sd(reeds$nitrogen)
pcprior <- list(prec = list(prior="pc.prec",
                           param = c(3*sdres, 0.01)))
formula <- nitrogen ~ f(site, model="iid",
                        hyper = pcprior)
pmod <- inla(formula, family="gaussian",
             data=reeds)
pmod$summary.fixed
bri.hyperpar.summary(pmod)
bri.hyperpar.plot(pmod)

```

7.4.1. Modelo lineal con interceptos y pendientes aleatorias

A continuación se presentan los comandos de R para ajustar un modelo lineal con interceptos y pendientes aleatorias correlacionadas usando la base de datos `reading`. Esta parte se usará en una de las prácticas por lo que no se presenta la salida aquí.

```

nid <- length(levels(reading$id))
reading$numid <- as.numeric(reading$id)
reading$slopeid <- reading$numid + nid
formula <- piat ~ cagegrp + f(numid, model="iid2d", n = 2*nid) +
          f(slopeid, cagegrp, copy="numid")
imod <- inla(formula, family="gaussian", data=reading)
imod$summary.fixed
bri.hyperpar.summary(imod)
postmean <- matrix(imod$summary.random$numid[,2],nid,2)
postmean <- sweep(postmean,2,imod$summary.fixed$mean,"+")
p <- ggplot(reading, aes(cagegrp, piat, group=id)) +
     geom_line(col=gray(0.95)) + xlab("centered age")
p+geom_abline(data=postmean,intercept=postmean[,1],slope=postmean[,2])

```

Si el modelo asume que los dos efectos aleatorios no están correlacionadas entonces se omite la parte `copy="numid"` de la fórmula.

7.5. Modelos Espaciales: Cáncer

El siguiente ejemplo ilustra el uso de INLA para ajustar modelos lineales generalizados con efectos espaciales. Los datos corresponden a los casos de cáncer de cerebro (`OBSERVED`), esperados (`EXPECTED`), y `SMR` (*standardized morbidity ratio*= O/E) entre 1988-1994 para

40 distritos en en Navarra, España (Blangiardo and Cameletti, 2015). El 32.5 % de los datos son 0's. La base de datos se puede obtener en este enlace.

```
load("~/Google Drive/12-Spring2018/Curso Argentina/CursoBuenosAires2018/Navarre.RData")
names(brainnav)
```

```
[1] "ZBS"      "NAME"      "OBSERVED"  "EXPECTED"  "RISK"      "RISKLL"
[7] "RISKUL"   "SMR"       "x"         "y"
```

```
navarra.graph <- inla.read.graph("~/Google Drive/12-Spring2018/Curso Argentina/CursoBuenosAires2018/Navarre.RData")
```

```
library(summarytools)
summarytools::freq(brainnav$OBSERVED, order = "freq")
```

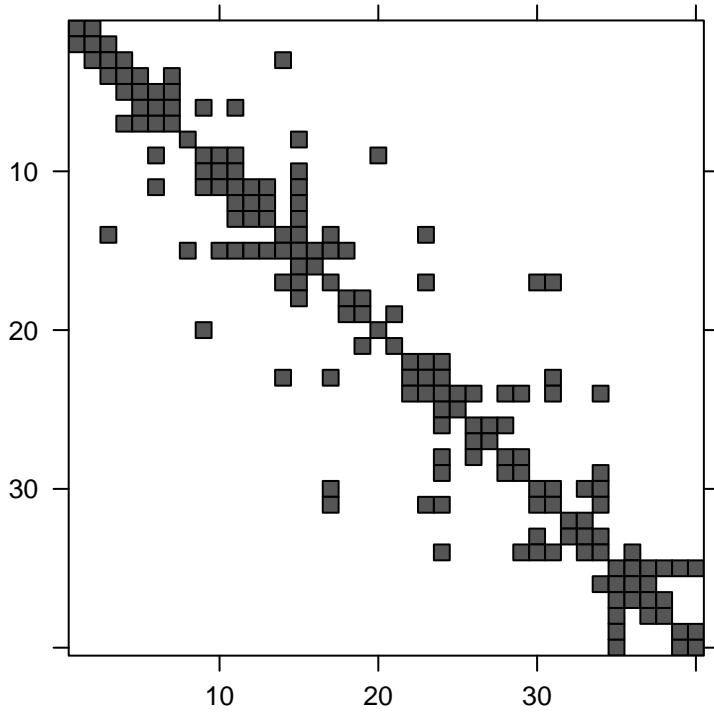
Frequencies

Type: Numeric

	Freq	% Valid	% Valid Cum.	% Total	% Total Cum.
0	13	32.50	32.50	32.50	32.50
3	7	17.50	50.00	17.50	50.00
1	6	15.00	65.00	15.00	65.00
4	6	15.00	80.00	15.00	80.00
2	4	10.00	90.00	10.00	90.00
5	3	7.50	97.50	7.50	97.50
55	1	2.50	100.00	2.50	100.00
<NA>	0			0.00	100.00
Total	40	100.00	100.00	100.00	100.00

Antes de definir el modelo es necesario tener un archivo adicional con la información de los vecinos de cada distrito. Este archivo debe tener un formato especial para ser leído en `inla()`. Una imagen de este archivo se muestra a continuación. Cada punto indica los vecinos de cada distrito. Para más detalles sobre la estructura de este archivo leer (Blangiardo and Cameletti, 2015).

```
image(inla.graph2matrix(navarra.graph),xlab="",ylab="")
```



Dimensions: 40 x 40

Ahora, vamos a especificar el modelo que se ajustará. Sea y_i el número de casos de cáncer de cerebro observados en el distrito $i = 1, \dots, n = 40$. Se asume que Y_i proviene de una distribución cuya densidad es

$$p(y_i | \lambda_i, \pi_0) = \pi_0 I(y_i = 0) + (1 - \pi_0) \frac{e^{-\lambda_i} \lambda_i^{y_i}}{y_i!}.$$

Además, se asume que

$$\log(\lambda_i) = b_0 + u_i + v_i + \log(E_i),$$

donde E_i es el número esperado de casos, $u_i \sim iCAR$ y $v_i \stackrel{i.i.d.}{\sim} Normal(0, \sigma_v^2)$ (*exchangeable*), independientes entre sí. Esto es,

$$u_i | \mathbf{u}_{-i} \sim Normal \left(\frac{1}{N_i} \sum_{j=1}^n a_{ij} u_j, \sigma_i^2 \right),$$

donde a_{ij} es la variable indicadora de los vecinos y N_i es el número de vecinos del distrito i (proviene del archivo `navarra.graph`) y $s_i^2 = \sigma_u^2 / N_i$. Esta formulación ($u_i + v_i$) se conoce como el modelo de **BYM** (**Besag-York-Mollié**).

La variable ZBS contiene el identificador del distrito. La distribución a priori para $\text{logit}^{-1}(\pi_0) \sim N(-2, 1)$.


```
data.navarra <- data.frame(ZBS=brainnav$ZBS,Y=brainnav$OBSERVED,E=brainnav$EXPECTED)

formula.zip <- Y ~ 1 + f(ZBS, model="bym", graph=navarra.graph,
  hyper=list(prec.unstruct=list(prior="gaussian",param=c(0,1)),
    prec.spatial=list(prior="gaussian",param=c(0,1))))

mod.zip1 <- inla(formula.zip,family="zeroinflatedpoisson1",
  data=data.navarra, offset = log(E),
  control.predictor=list(compute=TRUE))

round(mod.zip1$summary.hyperpar, 3)
```

	mean	sd
zero-probability parameter for zero-inflated poisson_1	0.066	0.053
Precision for ZBS (iid component)	4.458	2.282
Precision for ZBS (spatial component)	4.180	2.338
	0.025quant	0.5quant
zero-probability parameter for zero-inflated poisson_1	0.004	0.053
Precision for ZBS (iid component)	1.196	4.093
Precision for ZBS (spatial component)	1.065	3.733
	0.975quant	mode
zero-probability parameter for zero-inflated poisson_1	0.202	0.012
Precision for ZBS (iid component)	9.943	3.119
Precision for ZBS (spatial component)	10.013	2.721

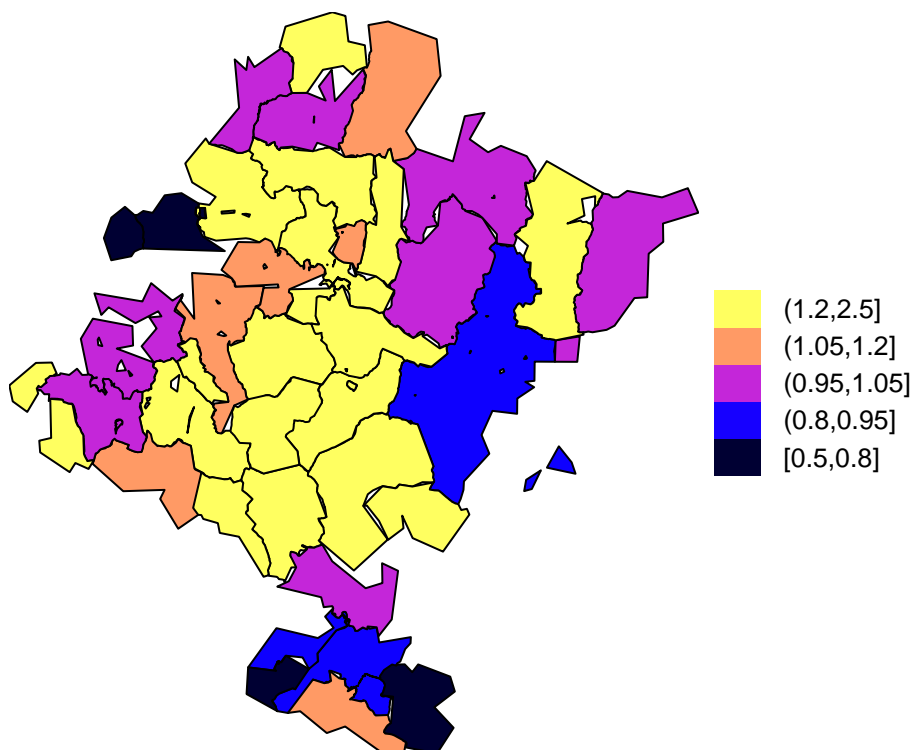
```
Nareas <- nrow(data.navarra)
# Extract the random effects
zeta.navarra1 <- data.frame(zeta=unlist(lapply(mod.zip1$marginals.random$ZBS[1:Nareas,],
  function(x)inla.emarginal(exp,x))))

# Create factor variables
RR.cutoff<- c(0.5, 0.8, 0.95, 1.05, 1.2, 2.5)
RR.navarra1 <- cut(zeta.navarra1$zeta,breaks=RR.cutoff,include.lowest=TRUE)

results <- data.frame(ZBS=data.navarra$ZBS, RR.navarra1)
data.navarra.shp <- attr(brainnav, "data")
attr(brainnav, "data") <- merge(data.navarra.shp, results, by="ZBS")

# Mapa de riesgos relativos a posteriori
library(lattice)
trellis.par.set(axis.line=list(col=NA))
spplot(obj=brainnav, zcol="RR.navarra1",
  main="Mapa de RR a posteriori")
```

Mapa de RR a posteriori



Capítulo 8

Temas Especiales

8.1. Selección de Variables usando LASSO Bayesiano

En algunos problemas de regresión donde hay un número grande de variables relativo al número de observaciones ($p \gg n$), una forma de llevar a cabo selección de variables y estimación al mismo tiempo es usando regresión penalizada o regularizada. Uno de los métodos más usados se denomina LASSO (*Least Absolute Shrinkage and Selection Operator*).

El estimador LASSO de los coeficientes de regresión es el resultado de solucionar el problema

$$\hat{\beta}_{LASSO} = \underset{\beta \in R^p}{\operatorname{argmin}} ||\mathbf{y} - \mathbf{X}\beta||^2 + \lambda \sum_{j=1}^p |\beta_j|.$$

Note que el primer término es la *SSE* de la regresión mientras que el segundo término agrega una penalidad a los coeficientes. El parámetro λ se conoce como *tuning parameter*. Cuando $\lambda \rightarrow 0$ entonces $\hat{\beta}$ tiende a la solución de mínimos cuadrados. Pero cuando $\lambda \rightarrow \infty$ entonces $\hat{\beta}$ tiende a 0. La idea es encontrar un valor óptimo de λ que minimice la varianza del estimador a pesar de que el sesgo aumente (*trade-off*). A diferencia de las estimaciones de mínimos cuadrados, las estimaciones LASSO se encogen hacia cero. Por lo tanto, las variables que no son estadísticamente significativas terminan con coeficientes iguales a 0 (selección de variables+estimación). En la regresión *Ridge*, los coeficientes se encogen pero nunca terminan iguales a cero.

En el contexto Bayesiano, la solución al problema de minimización anterior es equivalente a la moda de la distribución a posterior al asignar distribuciones a priori

doble-exponenciales o de Laplace para los β 's. Por lo tanto, la implementación de LASSO Bayesiano en JAGS se puede llevar a cabo especificando las respectivas distribuciones a priori. La densidad de Laplace toma la siguiente forma:

$$\pi(\beta|\tau) = \tau \exp(-\tau|\beta|/2), \quad \tau > 0.$$

Vamos a ilustrar el uso de LASSO Bayesiano usando el conjunto de datos `usair`. A pesar de que este conjunto de datos es pequeño, la extensión a problemas de mayor dimensión es relativamente fácil.

```
data(usair, package="brinla")
head(usair)
```

	S02	negtemp	manuf	pop	wind	precip	days
Phoenix	10	-70.3	213	582	6.0	7.05	36
Little Rock	13	-61.0	91	132	8.2	48.52	100
San Francisco	12	-56.7	453	716	8.7	20.66	67
Denver	17	-51.9	454	515	9.0	12.95	86
Hartford	56	-49.1	412	158	9.0	43.37	127
Wilmington	36	-54.0	80	80	9.0	40.25	114

```
# Variables estandarizadas
X <- scale(usair[, -1])

# Modelo
modelo <- function()
{
  for (i in 1:n){
    S02[i] ~ dnorm(mu[i], tau)
    mu[i] <- alpha + inprod(X[i, ], beta[])
  }
  # Distribuciones a priori
  alpha ~ dnorm(0, 0.001)
  tau <- pow(sigma, -2)
  sigma ~ dunif(0, 100)

  # Regularización L1 == Laplace (double exponential)
  for (j in 1:ncols) {
    beta[j] ~ ddexp(0, lambda)
  }
  #lambda ~ dunif(0.001, 10)
  lambda ~ dgamma(.001, .001)
}
```

```

# Crear base de datos para el modelo jags
sim.dat.jags <-list(
  n = nrow(X),
  ncols = ncol(X),
  X = X,
  S02 = usair$S02
)

library(rjags)
library(R2jags)

parametros <- c("alpha", "beta", "tau", "lambda")
jags.lasso <- jags(data = sim.dat.jags,
  inits = NULL,
  parametros,
  n.iter = 10000,
  model.file = modelo,
  n.burnin = 2000,
  n.chains = 1,
  n.thin = 1)

```

```

Compiling model graph
  Resolving undeclared variables
  Allocating nodes
Graph information:
  Observed stochastic nodes: 41
  Unobserved stochastic nodes: 9
  Total graph size: 437

```

Initializing model

```
jags.lasso$BUGSoutput$summary[, 1]
```

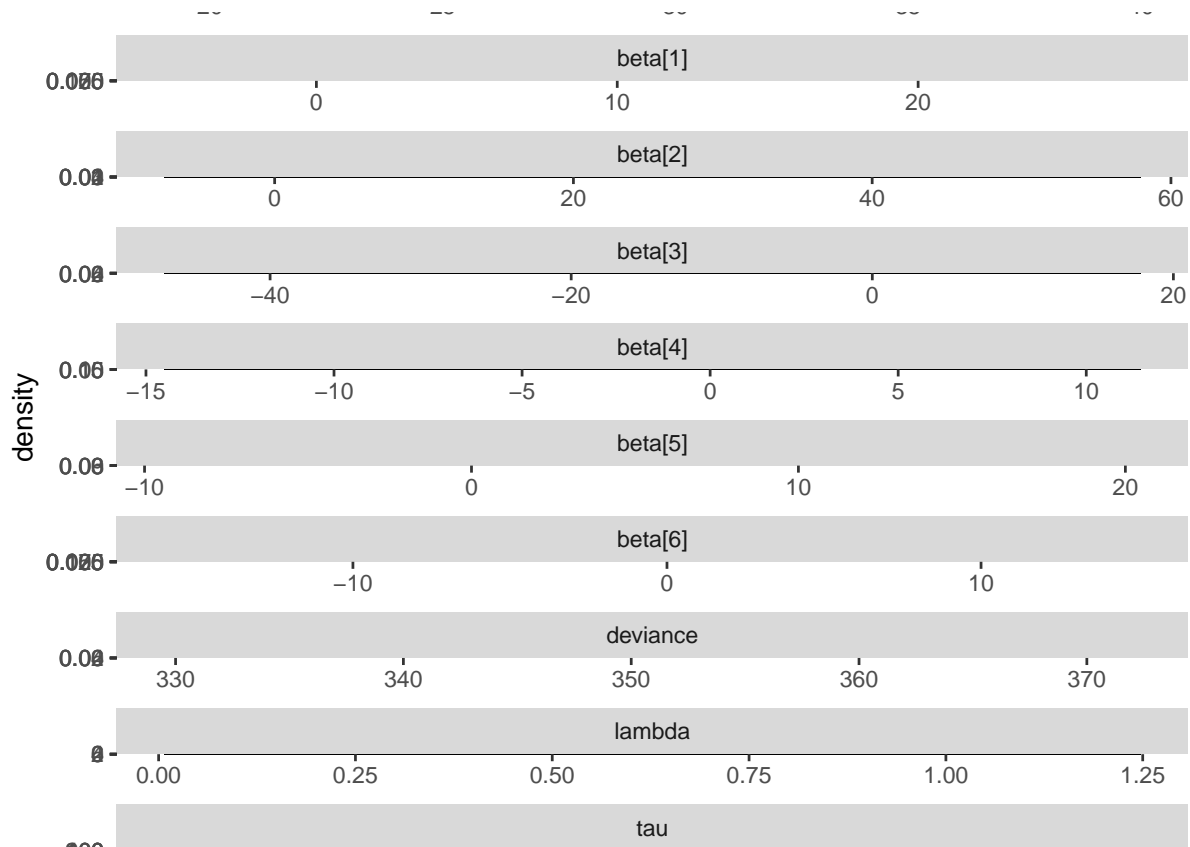
alpha	beta[1]	beta[2]	beta[3]	beta[4]
29.89743758	7.63011524	19.95465148	-6.91804040	-3.19401491
beta[5]	beta[6]	deviance	lambda	tau
3.32790741	1.44102807	342.08572695	0.15267404	0.00408131

```

# Diagn3sticos de convergencia
library(ggmcmc)
library(coda)
jags.lasso.mcmc <- as.mcmc(jags.lasso)
M <- ggs(jags.lasso.mcmc)

```

```
# Diagnósticos graficos
density <- ggs_density(M)
density
```



Compare las estimaciones LASSO anteriores con las obtenidas en el modelo frecuentista. Si el intervalo de credibilidad del 95 % incluye el 0 entonces podemos eliminar esa variable del modelo.

```
lm(usair$S02 ~ X)
```

Call:

```
lm(formula = usair$S02 ~ X)
```

Coefficients:

(Intercept)	Xnegtemp	Xmanuf	Xpop	Xwind
30.049	9.164	36.580	-22.746	-4.545
Xprecip	Xdays			
6.031	-1.380			

En la literatura existen otros métodos Bayesianos de selección de variables que se pueden implementar en JAGS. Aquí podrá encontrar algunos ejemplos sencillos

adicionales.

8.2. Suavizamiento en R-INLA

8.3. Datos Geoespaciales en R-INLA: Malaria en Gambia

Bibliografía

- Agresti, A. (2015). *Foundations of Linear and Generalized Linear Models*. Wiley series in probability and statistics, 1 edition.
- Blangiardo, M. and Cameletti, M. (2015). *Spatial and Spatio-temporal Bayesian Models with R - INLA*. Wiley, 1 edition.
- Carlin, B. P. and Louis, T. A. (2009). *Bayesian Methods for Data Analysis*. CRC Press, 1 edition.
- Lee, C. Y. Y. and Wand, M. P. (2015). Variational methods for fitting complex bayesian mixed effects models to health data. *Statistics in Medicine*.
- Ntzoufras, I. (2009). *Bayesian Modeling Using WinBUGS*. Wiley, 1 edition.
- Plummer, M. (2003). Jags: A program for analysis of bayesian graphical models using gibbs sampling.
- R Core Team (2016). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.
- Rue, H. and Martino, S. (2009). Approximate bayesian inference for latent gaussian models by using integrated nested laplace approximations. *J. R. Statist. Soc. B*, pages 319–392.
- Wang, X., Yue, Y. R., and Faraway, J. J. (2018). *Bayesian Regression Modeling with INLA*. CRC Press, 1 edition.
- Xie, Y. (2018). *knitr: A General-Purpose Package for Dynamic Report Generation in R*. R package version 1.19.