# Modelos lineales mixtos en R

Luis Cayuela

Junio de 2010

EcoLab, Centro Andaluz de Medio Ambiente, Universidad de Granada – Junta de Andalucía, Avenida del Mediterráneo s/n, E-18006, Granada. E-mail: lcayuela@ugr.es.

## Modelos lineales mixtos en R (versión 1.0)

Publicado por: Luis Cayuela




Para cualquier comentario o sugerencia por favor remitirse al autor de la obra.

# Índice

# 1. Introducción

Los modelos mixtos son usados cuando los datos tienen algún tipo de estructura jerárquica o de agrupación como los diseños de medidas repetidas, las series temporales, los diseños anidados o por bloques aleatorizados. Los modelos mixtos permiten tener coeficientes fijos (aquellos cuyos niveles son de interés para el experimentador) y aleatorios (aquellos cuyos niveles son sólo una realización de todos los posibles niveles procedentes de una población) y varios términos de error. Pueden ser una herramienta muy útil, pero son potencialmente difíciles de comprender y aplicar. En esta sesión intentaremos dar una visión aplicada de los modelos lineales mixtos con especial referencia a los distintos tipos de diseños vistos en la sesión anterior.

Podéis encontrar una buena introducción al tema de los modelos mixtos en el capítulo 8 de Zuur *et al.* (2007) y en el capítulo 19 de Crawley (2002). El libro de Zuur *et al.* (2009) es también una buena referencia para profundizar más en el tema de los modelos mixtos sin mucho aditamento matemático. Otras referencias con un fondo más matemático y, en mi opinión, bastante más difíciles de entender, son Pinheiro & Bates (2000) o Faraway (2006).

## 1.1. Los paquetes nlme y lme4

En R hay dos paquetes que nos van a permitir ajustar modelos mixtos. El primero que apareció fue el paquete nlme(), escrito inicialmente por José Pinheiro (Bell Laboratories) y Douglas Bates (University of Wisconsin) y al que luego se han sumado otros autores. El código de este paquete se escribió para que fuera compatible con S y S-Plus (las versiones comerciales de R).

```
> citation("nlme")

To cite package 'nlme' in publications use:

  Jose Pinheiro, Douglas Bates, Saikat DebRoy, Deepayan Sarkar and the
  R Core team (2009). nlme: Linear and Nonlinear Mixed Effects Models.
  R package version 3.1-96.

A BibTeX entry for LaTeX users is

  @Manual{,
    title = {nlme: Linear and Nonlinear Mixed Effects Models},
    author = {Jose Pinheiro and Douglas Bates and Saikat DebRoy and Deepayan Sarkar and th
    year = {2009},
    note = {R package version 3.1-96},
  }
```

El otro paquete, que apareció posteriormente, se llama lme4 y ha sido escrito originalmente por Douglas Bates. El código de este paquete no es compatible con S y S-Plus y la sintaxis, como veremos más adelante, cambia ligeramente con respecto al paquete nlme.

```
> citation("lme4")

To cite package 'lme4' in publications use:

  Douglas Bates and Martin Maechler (2010). lme4: Linear mixed-effects
  models using S4 classes. R package version 0.999375-33.
  http://CRAN.R-project.org/package=lme4

A BibTeX entry for LaTeX users is

  @Manual{,
    title = {lme4: Linear mixed-effects models using S4 classes},
    author = {Douglas Bates and Martin Maechler},
    year = {2010},
    note = {R package version 0.999375-33},
    url = {http://CRAN.R-project.org/package=lme4},
  }

ATTENTION: This citation information has been auto-generated from the
package DESCRIPTION file and may need manual editing, see
'help("citation")' .
```

Ambos paquetes contienen varias funciones relacionadas con los modelos mixtos, no solamente los modelos lineales mixtos, pero las que más nos van a interesar son las funciones lme() y lmer() de los paquetes nlme y lme4 respectivamente.

### 1.1.1. La función lme()

La especificación de los efectos fijos y aleatorios del modelo en la función lme() se realiza con dos argumentos distintos, fixed y random respectivamente. En el caso de que no haya efectos fijos en el modelo (todos los efectos son aleatorios) la componente fija del modelo estimaría sólo la constante o intercept:

fixed = y ~ 1

El argumento fixed no es totalmente necesario y se puede omitir en el caso de que no haya efectos fijos. En este caso basta con especificar una fórmula como en el caso de la función lm() y R entiende que todos los factores explicativos son aleatorios. La formulación entonces sería así:

y ~ a + b + c

dónde a, b y c serían factores aleatorios. En el caso de que haya factores fijos y aleatorios ambos componentes deben ser definidos de manera individual. El componente aleatorio en este caso se especificaría de la siguiente forma:

random = ~ 1 |a/b/c

Un detalle importante es que el nombre de la variable respuesta y no está repetido en la fórmula de los efectos aleatorios: en su lugar se deja un espacio en blanco a la izquierda del símbolo ~. En la mayoría de los modelos mixtos se

asumen que los efectos aleatorios tienen una media de cero y que lo que interesa cuantificar es la variación en la constante (esto es lo que significa el 1, ver sección 1.2) causada por las diferencias entre los niveles del factor de los efectos aleatorios. Después de la constante viene una barra vertical | que significa 'dada la siguiente distribución de las variables aleatorias'. En este ejemplo concreto hay tres efectos aleatorios con 'c anidado dentro de b, que a su vez está anidado dentro de a'. Los factores están separados por la barra oblicua / y las variables se enumeran de izquierda a derecha en orden decreciente siguiendo una jerarquía espacial o temporal. La formulación completa del modelo utilizando la función lme() sería así:

```
lme(fixed = y ~ 1, random = ~ 1 | a/b/c)
```

### 1.1.2. La función lmer()

En la función lmer() la fórmula se especifica en un único argumento incluyendo ambos tipos de efectos. Los efectos fijos se especifican en primer lugar a la derecha del símbolo ~ en la forma habitual. A continuación se especifican los efectos aleatorios precedidos por un + y en paréntesis. R identifica los efectos aleatorios porque llevan la barra vertical |. La especificación de factores anidados se haría con los ':' en vez de con la barra '/'. Sin embargo, al especificar la anidación de unos factores dentro de otros no se está asumiendo los efectos aleatorios de todos los factores incluidos en la fórmula como en el caso de la función lme(). Para el ejemplo anterior el modelo quedaría especificado de la siguiente forma:

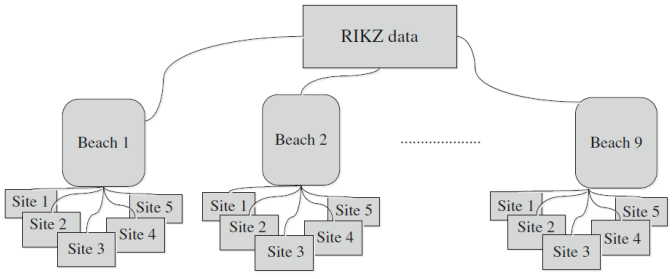
```
lme4(y ~ 1 + (1|a) + (1|a:b) + (1 | a:b:c))
```

A lo largo de las próximas secciones se verán formulaciones más complejas para la componente aleatoria del modelo mixto.

## 1.2. Un ejemplo de modelo mixto

### 1.2.1. Los datos

Zuur *et al.* (2007) utilizan datos de bentos marino procedente de nueve zonas intermareales de la costa holandesa para presentar los modelos mixtos. Los datos fueron recogidos por el instituto holandés RIKZ en el verano de 2002. En cada zona intermareal (playa) se tomaron cinco muestras de la macro-fauna y variables abióticas siguiendo el siguiente esquema de muestreo.



En este ejemplo vamos a ver si existe alguna relación entre el número de especies y el NAP, una variable que indica la altura de cada estación de muestreo con respecto al nivel medio de la marea. Como la riqueza de especies es un conteo, sería más apropiado utilizar un modelo lineal generalizado (GLM) con una distribución de errores de tipo Poisson. Sin embargo, para simplificar las cosas utilizaremos un modelo de regresión lineal con una distribución de errores normal o Gausiana.

Vamos a leer los datos. Éstos vienen en la forma de una matriz de abundancias de cada una de las especies observadas (columnas 2 a la 76). Para calcular la riqueza tenemos que convertir estas abundancias a presencia y sumar todas ellas para cada una de las filas (muestras):

```
> RIKZ <- read.table(file = "http://tinyurl.com/3y8s6tp", header = TRUE)
> RIKZ$Richness <- apply(RIKZ[, 2:76] > 0, 1, sum)
```

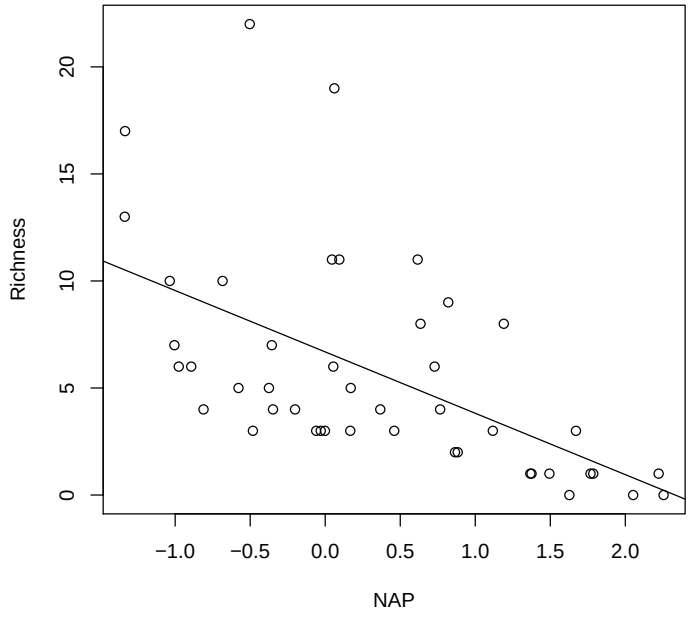
### 1.2.2. Una aproximación al análisis de los datos con modelos lineales

Sin saber de la existencia de los modelos mixtos, tal vez nuestra primera aproximación sería ajustar un modelo lineal asumiendo independencia de las muestras, de acuerdo al siguiente modelo:

$$Riqueza_i = \alpha + \beta \cdot NAP_i + \varepsilon_i \qquad \varepsilon_i \sim N(0, \sigma^2)$$

Dicho modelo originaría la siguiente gráfica:

```
> tmp <- lm(Richness ~ NAP, data = RIKZ)
> plot(RIKZ$NAP, RIKZ$Richness, xlab = "NAP", ylab = "Richness")
> abline(tmp)
```



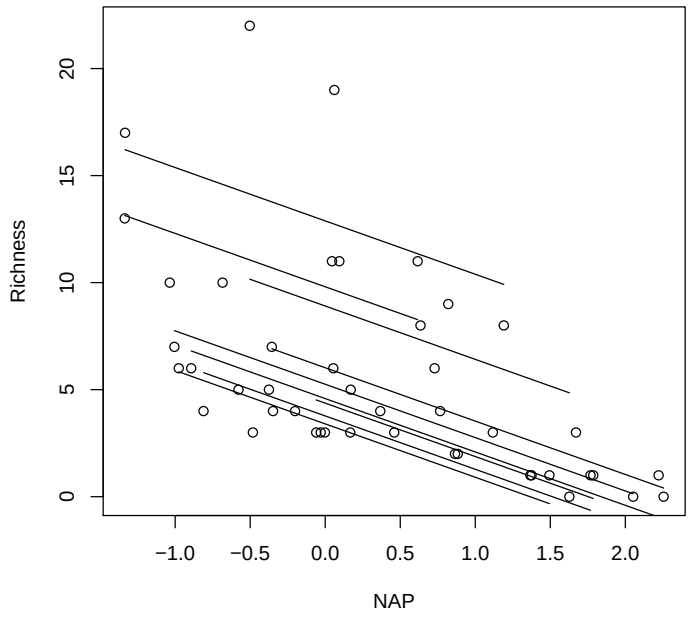

Y contiene tres parámetros desconocidos: los dos parámetros de la regresión (constante y pendiente) y la varianza residual ($\sigma^2$). El modelo asume que la relación entre riqueza y NAP es la misma en todas las playas. Sin embargo podría ocurrir que dicho modelo no es el mismo en todas las playas, en cuyo cayo tendríamos tres opciones: (1) que la pendiente fuera la misma pero la constante (Intercept) no lo fuera; (2) que la constante fuera la misma pero que la pendiente no lo fuera; y (3) que ni la constante ni la pendiente fueran las mismas para todas las playas.

En el primer caso, el modelo vendría dado por la siguiente ecuación:

$$Riqueza_{ij} = \alpha_j + \beta \cdot NAP_{ij} + \varepsilon_{ij} \qquad \varepsilon_{ij} \sim N(0, \sigma^2)$$

Y su gráfica correspondiente, en dónde tendríamos nueva rectas de regresión (una por cada playa), todas con la misma pendiente pero distinta constante (*Intercept*):

```
> tmp1 <- lm(Richness ~ NAP + factor(Beach), data = RIKZ)
> plot(RIKZ$NAP, RIKZ$Richness, xlab = "NAP", ylab = "Richness")
> for (i in 1:9) {
+     J <- RIKZ$Beach == i
+     x1 <- RIKZ$NAP[J]
+     y1 <- tmp1$fitted[J]
+     Ord <- order(x1)
+     lines(x1[Ord], y1[Ord])
+ }
```

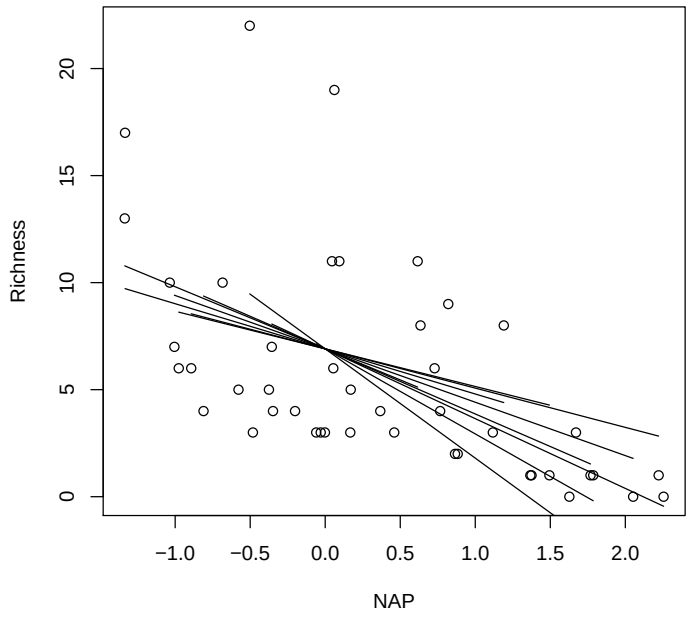



En el segundo caso, el modelo vendría dado por la siguiente ecuación:

$$Riqueza_{ij} = \alpha + \beta_j \cdot NAP_{ij} + \varepsilon_{ij} \qquad \varepsilon_{ij} \sim N(0, \sigma^2)$$

Y su gráfica correspondiente, en dónde tendríamos nueva rectas de regresión (una por cada playa), todas con la misma constante pero distinta pendiente:

```
> tmp2 <- lm(Richness ~ NAP + factor(Beach):NAP, data = RIKZ)
> plot(RIKZ$NAP, RIKZ$Richness, xlab = "NAP", ylab = "Richness")
> for (i in 1:9) {
+     J <- RIKZ$Beach == i
+     x1 <- RIKZ$NAP[J]
+     y1 <- tmp2$fitted[J]
+     Ord <- order(x1)
+     lines(x1[Ord], y1[Ord])
+ }
```

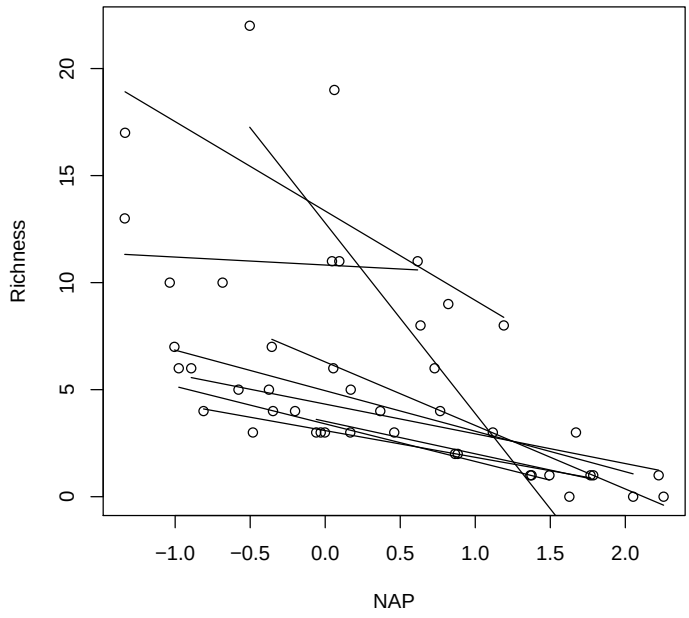



En el tercer caso, el modelo vendría dado por la siguiente ecuación:

$$Riqueza_{ij} = \alpha_j + \beta_j \cdot NAP_{ij} + \varepsilon_{ij} \qquad \varepsilon_{ij} \sim N(0, \sigma^2)$$

Y su gráfica correspondiente, en dónde tendríamos nueva rectas de regresión (una por cada playa), todas con distinta constante y pendiente:

```
> tmp3 <- lm(Richness ~ NAP * factor(Beach), data = RIKZ)
> plot(RIKZ$NAP, RIKZ$Richness, xlab = "NAP", ylab = "Richness")
> for (i in 1:9) {
+     J <- RIKZ$Beach == i
+     x1 <- RIKZ$NAP[J]
+     y1 <- tmp3$fitted[J]
+     Ord <- order(x1)
+     lines(x1[Ord], y1[Ord])
+ }
```




El modelo con una única recta de regresión es sin duda el más sencillo, y el modelo con nueve rectas de regresión con diferentes constantes y pendientes para cada una de las nueve playas el más complejo. El número de parámetros en el primer modelo es de 3, en los dos modelos intermedios de $1 + 9 + 1 = 11$, y en el último modelo, el más complejo, de $9 + 9 + 1 = 19$.

Comparemos ahora los distintos modelos utilizando el estadístico F con la función anova().

```
> anova(tmp, tmp1, test = "F")

Analysis of Variance Table

Model 1: Richness ~ NAP
Model 2: Richness ~ NAP + factor(Beach)
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
```

```
1     43 744.12
2     35 327.74  8    416.37 5.5581 0.0001441 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

> anova(tmp, tmp2, test = "F")

Analysis of Variance Table

Model 1: Richness ~ NAP
Model 2: Richness ~ NAP + factor(Beach):NAP
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1     43 744.12
2     35 699.28  8    44.831 0.2805 0.9681

> anova(tmp, tmp3, test = "F")

Analysis of Variance Table

Model 1: Richness ~ NAP
Model 2: Richness ~ NAP * factor(Beach)
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1     43 744.12
2     27 165.92 16    578.19 5.8804 2.993e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

> anova(tmp1, tmp3, test = "F")

Analysis of Variance Table

Model 1: Richness ~ NAP + factor(Beach)
Model 2: Richness ~ NAP * factor(Beach)
  Res.Df    RSS Df Sum of Sq      F   Pr(>F)
1     35 327.74
2     27 165.92  8    161.82 3.2915 0.009434 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Claramente el modelo más complejo es el mejor modelo y el que explica mayor cantidad de variabilidad (se minimiza la suma de cuadrados residual). Ahora bien, estamos interesados en la relación general entre riqueza y NAP y no nos importa tanto el efecto ocasionado por las características particulares de cada playa. Pero si eliminamos la playa del análisis entonces las diferencias en la riqueza de especies que son atribuibles a las características particulares de cada playa pasaría a formar parte del término de la varianza residual. No tener esto en consideración podría afectar los errores estándar y los $p$-valores de los efectos fijos. Esto podría producir como resultado, por ejemplo, la no detección

de una relación significativa entre la riqueza y el NAP. Por tanto hace falta incorporar el efecto de la playa en el modelo. ¡Pero esto implica incorporar 16 parámetros más en el modelo, lo que conlleva la pérdida de 16 grados de libertad! Una alternativa posible para evitar ésto es utilizar los modelos mixtos. Pero el uso de los modelos mixtos ofrece, además, otras ventajas. Como vimos en la sesión anterior, si consideramos la playa como un factor fijo, nuestras conclusiones sólo pueden referirse a esas playas concretas. Sin embargo, si tratamos la playa como un factor aleatorio, entonces nuestras conclusiones se referirán a todas las playas, de las cuáles, estas nueve son sólo una muestra del total de la población de playas. Por tanto los resultados del análisis al considerar la playa como un factor aleatorio permitiría predecir la relación entre la riqueza y el NAP en un sentido más general, sin tener que limitarnos a estas nueve playas en concreto.

### 1.2.3. Re-analizando los datos con modelos lineales mixtos: El paquete nlme

Vamos a comenzar re-analizando el primero de los tres modelos anteriores, que consideraba la existencia de diferencias en la constante de la recta de regresión entre la riqueza y el NAP para las distintas playas. Si consideramos la playa como un factor aleatorio, el modelo quedaría formulado de la siguiente forma:

$$Riqueza_{ij} = \alpha + \beta \cdot NAP_{ij} + a_j + \varepsilon_{ij}$$

$$a_j \sim N(0, \sigma_a^2) \qquad \varepsilon_{ij} \sim N(0, \sigma^2)$$

El índice *j* (representando a las playas) toma valores de 1 a 9, e *i* (representando las muestras dentro de cada playa) toma valores de 1 a 5. El el modelo lineal anterior, acabábamos con nueve líneas de regresión. La pendiente estimada, los nueve puntos de corte y sus errores estándares nos decían cómo era de diferente la relación entre la riqueza y el NAP para las distintas playas (sin que cambiase la pendiente). En este modelo, asumimos que sólo hay una línea de regresión con una única constante y una única pendiente. La constante $\alpha$ y la pendiente $\beta$ son los parámetros fijos del modelo. Además, hay una constante aleatoria, $a_j$, que añade cierta cantidad de variación a la constante del modelo general en cada una de las playas. Se asume que esta constante aleatoria sigue una distribución normal de media 0 y varianza $\sigma_a^2$. Por lo tanto, los parámetros estimados en el modelo son cuatro, $\alpha$, $\beta$, la varianza residual $\sigma^2$, y la varianza de la constante $\sigma_a^2$. De esta forma tenemos un modelo equivalente al modelo lineal, pero en dónde sólo es necesario estimar cuatro parámetros y no once, ya que en vez de tener nueve estimaciones de las constantes en cada una de las rectas de regresión de las nueve playas, tenemos nueve valores no estimados $\sigma_1, \ldots, \sigma_9$ que asumimos siguen una distribución normal. Lo que estimamos en este modelo es la varianza de esta distribución.

Vamos a ajustar el modelo lineal mixto correspondiente con la función lme():

```
> library(nlme)
> lme5 <- lme(Richness ~ NAP, data = RIKZ, random = ~1 | factor(Beach))
> summary(lme5)
```

```
Linear mixed-effects model fit by REML
 Data: RIKZ
       AIC      BIC    logLik
  247.4802 254.5250 -119.7401

Random effects:
 Formula: ~1 | factor(Beach)
        (Intercept) Residual
StdDev:    2.944065  3.05977

Fixed effects: Richness ~ NAP
                Value Std.Error DF   t-value p-value
(Intercept)  6.581893 1.0957618 35  6.006682       0
NAP         -2.568400 0.4947246 35 -5.191574       0
 Correlation:
    (Intr)
NAP -0.157

Standardized Within-Group Residuals:
       Min         Q1        Med         Q3        Max
-1.4227495 -0.4848006 -0.1576462  0.2518966  3.9793918

Number of Observations: 45
Number of Groups: 9
```

```
> anova(lme5)
```

```
            numDF denDF  F-value p-value
(Intercept)     1    35 27.63494  <.0001
NAP             1    35 26.95244  <.0001
```

La parte de los resultados que se refiere a los efectos aleatorios nos muestra que la varianza residual es $\sigma^2 = 3{,}06^2$ y la varianza de la constante $\sigma_a^2 = 2{,}944^2$. Para la parte de los efectos fijos del modelo, $\alpha + \beta \cdot NAP_{ij}$, la constante se estima en $\alpha = 6{,}582$ y la pendiente en $\beta = -2{,}568$. Ambos parámetros son significativamente distintos de 0. La correlación entre la constante y la pendiente es pequeña. La tabla ANOVA también muestra la significación de la pendiente $\beta$.

En resumen, el modelo estima una respuesta general de la forma $6{,}582 - 2{,}568 \cdot NAP_{ij}$. Estos parámetros estimados por el modelo son significativamente distintos de 0. Para cada playa, la constante aumenta o disminuye por un valor aleatorio. Este valor aleatorio sigue una distribución normal con media esperada 0 y varianza $\sigma_a^2 = 2{,}944^2$. La varianza residual, es decir, el error que podemos añadir a cada una de nuestras predicciones, viene dada por $\sigma^2 = 3{,}06$.

### 1.2.4. Re-analizando los datos con modelos lineales mixtos: El paquete lme4

El ajuste del modelo con la función lmer() del paquete lme4 es prácticamente idéntico a cómo lo hemos hecho utilizando la función lme() del paquete nlme. La principal diferencia es que no hace falta especificar el argumento random y los efectos aleatorios se especifican como parte de la fórmula, de la siguiente forma:

```
> library(lme4)
> lmer5 <- lmer(Richness ~ NAP + (1 | Beach), data = RIKZ)
> summary(lmer5)

Linear mixed model fit by REML
Formula: Richness ~ NAP + (1 | Beach)
   Data: RIKZ
   AIC   BIC logLik deviance REMLdev
 247.5 254.7 -119.7    241.9   239.5
Random effects:
 Groups   Name        Variance Std.Dev.
 Beach    (Intercept) 8.6675   2.9441
 Residual             9.3622   3.0598
Number of obs: 45, groups: Beach, 9

Fixed effects:
            Estimate Std. Error t value
(Intercept)   6.5819     1.0957   6.007
NAP          -2.5684     0.4947  -5.192

Correlation of Fixed Effects:
    (Intr)
NAP -0.157

> anova(lmer5)

Analysis of Variance Table
    Df Sum Sq Mean Sq F value
NAP  1 252.33  252.33  26.952
```

Como vemos, los valores de los parámetros aleatorios y fijos estimados por el modelo son exactamente iguales utilizando cualquiera de las dos funciones. En el caso de la función lmer(), y al contrario de lo que ocurre con la función lme(), no se dan los valores de significación de los coeficientes fijos estimados por el modelo. Buscando en la lista de distribución de R, uno acabará encontrándose con un debate casi filosófico sobre la capacidad de estos modelos de realizar test de significación utilizando el estadístico F. En concreto, Douglas Bates, autor principal de este paquete, cree que no es posible estimar con fiabilidad los grados de libertad de los parámeros fijos y,

por tanto, proponer una distribución del estadístico F lo suficientemente fiable como realizar un test de significación. En este contexto la significación de los parámetros fijos en el modelo puede hacerse comparando modelos más complejos frente a modelos menos complejos utilizando un criterio de información (AIC, BIC), siempre y cuando los efectos aleatorios no cambien. En este ejemplo, el test de significación vendría dado de la siguiente forma:

```
> lmer5.0 <- lmer(Richness ~ NAP + (1 | Beach), data = RIKZ, REML = F)
> lmer5.1 <- lmer(Richness ~ 1 + (1 | Beach), data = RIKZ, REML = F)
> anova(lmer5.0, lmer5.1)

Data: RIKZ
Models:
lmer5.1: Richness ~ 1 + (1 | Beach)
lmer5.0: Richness ~ NAP + (1 | Beach)
        Df    AIC    BIC  logLik  Chisq Chi Df Pr(>Chisq)
lmer5.1  3 269.30 274.72 -131.65
lmer5.0  4 249.83 257.06 -120.92 21.474      1  3.586e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

La razón por la que se utiliza el argumento REML = F se explica en la sección 1.3. Como vemos, el modelo completo (lmer5.0) produce un AIC que es inferior al AIC del modelo anidado (que no contempla la variación en las pendientes), y además el test Chi-cuadrado realizado sobre el parámetro de verosimilitud (logLik) es significativo, lo que indica que que el modelo completo es más parsimonioso. Todo ello apunta a que existe un efecto general y significativo del NAP sobre la riqueza de especies, resultado coincidente con el obtenido anteriormente con la función lme().

¿Y cómo comprobar la significación de las variables aleatorias? En el caso de las funciones lme() y lmer(), no se calculan p-valores para los términos aleatorios. La idea sería básicamente la misma que para el caso de los efectos fijos con la función lmer(): habría que comparar modelos alternativos con una estructura fija idéntica pero que variasen en sus efectos aleatorios. Sin embargo la estimación de los parámetros de estos modelos ha de hacerse con el estimador de máxima verosimilitud restringida (REML) como se explica más adelante.

### 1.2.5. Otros efectos aleatorios en modelos lineales mixtos

Siguiendo con el ejemplo anterior, es posible que contemplemos la posibilidad de que haya, además de distintos puntos de corte, distintas pendientes para cada una de las playas. En este caso el modelo quedaría formulado de las siguiente forma:

$$Riqueza_{ij} = \alpha + a_j + \beta \cdot NAP_{ij} + b_j \cdot NAP_{ij} + \varepsilon_{ij}$$

$$\varepsilon_{ij} \sim N(0, \sigma^2) \qquad a_j \sim N(0, \sigma_a^2) \qquad b_j \sim N(0, \sigma_b^2)$$

El modelo es exactamente igual que el modelo anterior excepto por el término $b_j \cdot NAP_{ij}$. Este nuevo término permite la variación aleatoria de la pendiente en cada una de las playas. El modelo es muy parecido al modelo lineal completo con interacción que ajustamos en la sección 1.2.2, pero con muchos menos parámetros: dos para los efectos fijos $\alpha$ y $\beta$, y tres para las varianzas de los términos aleatorios $\sigma^2$, $\sigma_a^2$ y $\sigma_b^2$. En general, se permite que haya una correlación alta entre las varianzas estimadas $\sigma_a^2$ y $\sigma_b^2$.

Vamos a ajustar el modelo lineal mixto correspondiente (los dos modelos lme6 son equivalentes):

```
> library(nlme)
> lme6 <- lme(Richness ~ NAP, data = RIKZ, random = ~NAP | factor(Beach))
> lme6 <- lme(Richness ~ NAP, data = RIKZ, random = ~1 + NAP |
+     factor(Beach))
> summary(lme6)

Linear mixed-effects model fit by REML
 Data: RIKZ
       AIC      BIC    logLik
  244.3839 254.9511 -116.1919

Random effects:
 Formula: ~1 + NAP | factor(Beach)
 Structure: General positive-definite, Log-Cholesky parametrization
            StdDev   Corr
(Intercept) 3.549074 (Intr)
NAP         1.714959 -0.99
Residual    2.702822

Fixed effects: Richness ~ NAP
                Value Std.Error DF   t-value p-value
(Intercept)  6.588704 1.2647639 35  5.209434   0e+00
NAP         -2.830026 0.7229388 35 -3.914614   4e-04
 Correlation:
    (Intr)
NAP -0.819

Standardized Within-Group Residuals:
       Min         Q1        Med         Q3        Max
-1.8213293 -0.3411050 -0.1674617  0.1921276  3.0397129

Number of Observations: 45
Number of Groups: 9

> anova(lme6)
```

```
            numDF denDF  F-value p-value
(Intercept)     1    35 12.19506  0.0013
NAP             1    35 15.32420  0.0004
```

La parte de los resultados que se refiere a los efectos aleatorios nos muestra que la varianza residual es $\sigma^2 = 2{,}70^2$, la varianza de la constante $\sigma_a^2 = 3{,}55^2$ y la varianza de la pendiente $\sigma_b^2 = 1{,}71^2$. Vemos que disminuye la varianza residual (lo que no explica el modelo) con respecto al modelo anterior. Para la parte de los efectos fijos del modelo, $\alpha + \beta \cdot NAP_{ij}$, la constante estimada es $\alpha = 6{,}588$ y la pendiente $\beta = -2{,}83$. Ambos parámetros son significativamente distintos de 0 y muy similares a los coeficientes estimados en el modelo anterior.

## 1.3. Selección de modelos

Tenemos dos modelos mixtos: uno en dónde la constante varía aleatoriamente (lme5) y otro en dónde tanto la constante y la pendiente varían aleatoriamente (lme6). ¿Qué modelo es mejor o más apropiado? Hay que notar que ambos modelos son idénticos en cuanto a sus efectos fijos y sólo varían en sus efectos aleatorios. Para comparar los dos modelos con los mismos efectos fijos pero con distintos efectos aleatorios, podemos usar un test de verosimilitud.

```
> anova(lme5, lme6)
```

```
     Model df      AIC      BIC    logLik   Test  L.Ratio p-value
lme5     1  4 247.4802 254.5250 -119.7401
lme6     2  6 244.3839 254.9511 -116.1919 1 vs 2 7.096378  0.0288
```

El AIC sugiere que el modelo lme6 es más apropiado, pero el BIC sugiere que el modelo lme5 es mejor. El p-valor del test de verosimilitud indica que el modelo más complejo (lme6) es el más parsimonioso y por tanto elegiríamos éste. Sin embargo, hay un problema con el test de verosimilitud. En modelos lineales se puede utilizar un test F para comparar modelos anidados, en dónde la suma de cuadrados residual de los dos modelos obtenida por mínimos cuadrados es utilizada para calcular el estadístico de contraste F. En este caso, podemos comprobar hipótesis como $H_0 : \beta_0 = 0$ frente a $H_1 : \beta_0 \neq 0$. En los modelos lineales mixtos el estimador utilizado no es mínimos cuadrados sino máxima verosimilitud (ML). El criterio de verosimilitud del modelo completo $L_0$ y el modelo anidado $L_1$ son utilizados para comprobar hipótesis referentes a la significación de los parámetros en el modelo. Tomando el logaritmo o, más comúnmente, $-2 \cdot log$, se calcula un estadístico de la forma $L = -2(logL_0 - logL_1)$. Se puede demostrar que bajo la hipótesis nula, este estadístico sigue aproximadamente una distribución Chi-cuadrado con $\nu$ grados de libertad, en dónde $\nu$ es la diferencia entre el número de parámetros de ambos modelos. ¿Cuál sería la hipótesis nula en este caso? La única diferencia entre el modelo lme6 y el modelo lme5 es el componente aleatorio $b_j \cdot NAP_{ij}$, dónde $b_j \sim N(0, \sigma_b^2)$. Si comparamos ambos modelos con un test de verosimilitud, estaríamos comprobando la hipótesis nula de que $H_0 : \sigma_b^2 = 0$ frente a $H_1 : \sigma_b^2 > 0$. La hipótesis alternativa contiene un '>' porque se supone

que los componentes de la varianza no pueden tomar valores negativos. Esto es lo que se conoce como el **problema del límite** (*boundary problem*); estamos comprobando si la varianza es cero, pero este valor está en el límite de todos los posibles valores que puede tomar la varianza. El problema es que la teoría subyacente al test de verosimilitud, que es la que genera un p-valor, asume que no hay un límite en el espacio del parámetro estimado. En resumen, hay que tener mucho cuidado cuando se interpreta el p-valor derivado de un test de verosimilitud cuando estamos comprobando la hipótesis nula en el límite de la distribución del parámetro en cuestión. Si el p-valor es muy grande o muy pequeño, no suele haber problema en interpretarlo, pero cuando encontramos un p-valor en el borde de la significación hay que tener cuidado con cómo lo interpretamos. Pinheiro & Bates (2000) y Faraway (2006) argumentan que, en este sentido, las técnicas de bootstraping son más confiables a la hora de estimar los p-valores si estamos comprobando una hipótesis en el límite de su distribución. Otra alternativa es hacer la estimación utilizando un estimador de máxima verosimilitud restringida (REML), que es lo que por defecto hacen las funciones lme() y lmer(), que permite corregir en parte el problema de la estimación de la varianza en el límite de su distribución.

De lo visto hasta el momento concluimos que hay dos formas de estimar los parámetros de los modelos, mediante máxima verosimilitud (ML) y mediante máxima verosimilitud restringida (REML). Es importante mencionar que no es posible comparar un modelo estimado mediante ML con otro modelo estimado mediante REML.

En los modelos lineales mixtos, tenemos dos tipos de efectos, los efectos fijos y los aleatorios. Aunque generalmente vamos a estar más interesados en los efectos fijos, si tenemos una estructura de efectos aleatorios mal definida es posible que eso afecte a la estimación de los efectos fijos. Por ello es importante seleccionar la mejor estructura para cada uno de estas dos componentes. Para ello se recomienda lo siguiente (Zuur *et al.* 2009):

1. Empieza ajustando un modelo en dónde la componente fija contenga todas las variables explicativas e interacciones posibles. Esto es lo que se conoce como modelo más allá del óptimo (*beyond optimal model*). Si no es posible ajustar este modelo porque hay demasiados parámetros, hay que intentar seleccionar las variables e interacciones que creamos influyen más sobre la variable respuesta.

2. Utilizando el modelo más allá del óptimo, hay que encontrar la estructura de la componente aleatoria óptima. Para ello propondremos distintos modelos alternativos con la misma estructura en la componente fija pero que varían en su componente aleatoria. Porque en la componente aleatoria estamos estimando varianzas nos encontramos con el problema de la estimación en el límite de la distribución de los parámetros estimados mencionado anteriormente. Por ello estos modelos tienen que estar estimados utilizando REML. La comparación entre distintos modelos podemos hacerla utilizando la función `anova()`.

3. Una vez que hemos definido la estructura óptima de la componente aleatoria, tenemos que buscar la estructura óptima de la componente fija

del modelo. Para ello podemos utilizar el estadístico F o el estadístico t obtenido mediante el estimador REML con la función lme()[1] o comparar modelos anidados. Para comparar modelos que tienen la misma estructura en la componente aleatoria pero difieren en la componente fija se debe de utilizar un estimador LM y no un estimador de REML.

4. Cuando se ha seleccionado la estructura de la componente fija, se presenta el modelo final utilizando un estimador REML.

## 1.4. Análisis de los residuos

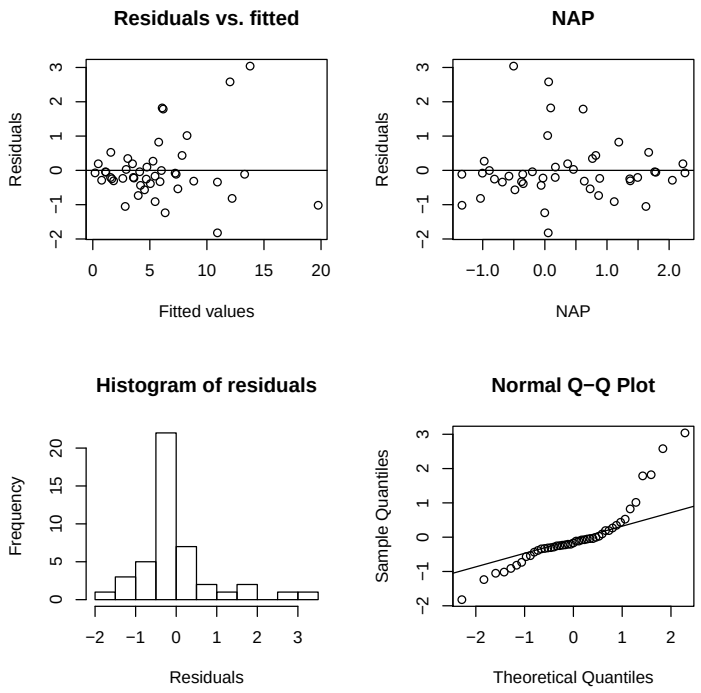
Para saber si el modelo es adecuado debemos mirar los residuos normalizados del modelo estimado a partir de REML. Al contrario que en el caso de los modelos lineales (lm) o modelos lineales generalizados (glm) no tenemos una función que nos genera todos los gráficos de los residuos, así que tendremos que generarlos nosotros individualmente. Interesa dibujar los residuos frente a los valores predichos para ver si hay homocedasticidad. Si no hay homogeneidad de varianzas entonces se puede: (i) aplicar una transformación; (ii) intentar averiguar si el incremento de la varianza para determinados valores es debido a alguna covariable; (iii) utilizar otro tipo de modelos como los modelos lineales generalizados mixtos con una distribución de errores diferente (por ejemplo Poisson si la variable respuesta es un conteo). Es interesante dibujar los gráficos de los residuos frente a las variables explicativas. De nuevo, no tenemos que observar ningún patrón en estas gráficas. Si la variable explicativa es un factor, la varianza ha de ser homogénea entre los distintos niveles del factor.

También se puede sacar un histograma de los residuos para ver si hay normalidad. Dicha normalidad también se puede comprobar con tests estadísticos como el test de Shapiro-Wilk.

Finalmente podemos obtener el qqplot de los residuos con la función qqnorm().

[1] Recordemos que la función lmer() del paquete lme4 no hace estos contrastes de hipótesis.

```
> Res <- residuals(lme6, type = "normalized")
> Fit <- fitted(lme6)
> par(mfrow = c(2, 2))
> plot(Res ~ Fit, xlab = "Fitted values", ylab = "Residuals", main = "Residuals vs. fitted
> abline(h = 0)
> plot(Res ~ RIKZ$NAP, xlab = "NAP", ylab = "Residuals", main = "NAP")
> abline(h = 0)
> hist(Res, main = "Histogram of residuals", xlab = "Residuals")
> qqnorm(Res)
> qqline(Res)
```




## 2. Diseño por bloques

Los datos del RIKZ son un claro ejemplo de diseño por bloques. En este caso no tenemos un diseño por bloques aleatorizados puesto que dentro de los bloques no investigamos la respuesta de una variable (riqueza) frente a un factor, sino frente a una covariable (NAP). En este caso tendríamos por tanto un bloque que es consecuencia de la agrupación de las unidades muestrales dentro de un factor de agrupación que son las playas (recordemos que se tomaron cinco muestras por playa). Por tanto, no es del interés del investigador ver si la playa tiene un efecto sobre la riqueza de especies, pero no se puede obviar este factor puesto que tiene una influencia sobre la varianza de la variable respuesta y, por tanto, va a tener un efecto probable en la estimación de los parámetros de interés en el modelo (NAP).

Vamos a utilizar ahora otro ejemplo distinto. En este estudio un nutricionista investiga el efecto de seis tipos de dieta distintos (a,..., f) en la ganancia de peso de conejos domésticos. Para ello selecciona 3 conejos más o menos uniformes en cuanto a tamaño y peso procedentes de 10 camadas distintas. En total hay por tanto 30 muestras. Como hay seis tratamientos y sólo 3 conejos por camada, no se puede aplicar todos los tratamientos a todos los conejos de cada camada. El diseño es, por tanto, un diseño factorial por bloques aleatorizados incompletos.

Vamos a leer los datos:

```
> library(faraway)
> data(rabbit)
> xtabs(gain ~ treat + block, data = rabbit)

     block
treat   b1  b10   b2   b3   b4   b5   b6   b7   b8   b9
    a  0.0 37.3 40.1  0.0 44.9  0.0  0.0 45.2 44.0  0.0
    b 32.6  0.0 38.1  0.0  0.0  0.0 37.3 40.6  0.0 30.6
    c 35.2  0.0 40.9 34.6 43.9 40.9  0.0  0.0  0.0  0.0
    d  0.0 42.3  0.0 37.5  0.0 37.3  0.0 37.9  0.0 27.5
    e  0.0  0.0  0.0  0.0 40.8 32.0 40.5  0.0 38.5 20.6
    f 42.2 41.7  0.0 34.3  0.0  0.0 42.8  0.0 51.9  0.0
```

## 2.1. Paso 1: Aplicación de un modelo lineal

Como ya hemos visto, una forma de analizar el efecto de un bloque es considerando éste como un efecto fijo por medio de un modelo lineal. Es importante tener en cuenta que a la hora de formular el modelo el bloque tiene que ir en primer lugar. Esto es porque los modelos lineales en R se ajustan utilizando una suma de cuadrados de tipo I, que tiene en cuenta el orden de entrada de las variables en el modelo, y nos interesa analizar el efecto del factor (dieta) sobre la respuesta (ganancia de peso) una vez que se haya tenido en cuenta la variabilidad atribuible al bloque (camada). También es importante observar que el diseño no es cruzado, es decir que no todos los niveles del factor están representados dentro de cada uno de los niveles del bloque. Por tanto, plantear una interacción en este modelo no es posible.

```
> lm.rabbit <- lm(gain ~ block + treat, data = rabbit)
> anova(lm.rabbit)

Analysis of Variance Table

Response: gain
          Df Sum Sq Mean Sq F value    Pr(>F)
block      9 730.39  81.154  8.0738 0.0002454 ***
treat      5 158.73  31.745  3.1583 0.0381655 *
Residuals 15 150.77  10.052
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Observamos que tanto el bloque como el tratamiento son significativos.

## 2.2. Paso 2: Ajuste de un modelo lineal mixto

Aunque el modelo lineal no es necesariamente inadecuado, la inclusión del bloque como un factor fijo se hace a expensas de 9 grados de libertad. En este sentido el modelo mixto equivalente sería mucho más parsimonioso en término del número de parámetros.

```
> library(nlme)
> lme.rabbit1 <- lme(gain ~ treat, random = ~1 | block, data = rabbit)
> anova(lme.rabbit1)

            numDF denDF  F-value p-value
(Intercept)     1    15 590.5297  <.0001
treat           5    15   3.2818  0.0336

> summary(lme.rabbit1)

Linear mixed-effects model fit by REML
 Data: rabbit
       AIC      BIC    logLik
  166.3569 175.7813 -75.17844

Random effects:
 Formula: ~1 | block
        (Intercept) Residual
StdDev:    4.657853 3.175519

Fixed effects: gain ~ treat
               Value Std.Error DF   t-value p-value
(Intercept) 39.53540  2.130338 15 18.558278  0.0000
treatb      -2.50718  2.208700 15 -1.135139  0.2741
treatc      -0.18407  2.208700 15 -0.083340  0.9347
treatd      -0.88516  2.208700 15 -0.400759  0.6942
treate      -5.64602  2.208700 15 -2.556263  0.0219
treatf       2.81003  2.208700 15  1.272254  0.2227
 Correlation:
       (Intr) treatb treatc treatd treate
treatb -0.518
treatc -0.518  0.500
treatd -0.518  0.500  0.500
treate -0.518  0.500  0.500  0.500
treatf -0.518  0.500  0.500  0.500  0.500

Standardized Within-Group Residuals:
       Min         Q1        Med         Q3        Max
-1.3794203 -0.5213453 -0.1701517  0.6456859  1.4148990
```

```
Number of Observations: 30
Number of Groups: 10
```

## 2.3. Paso 3: Elección de la estructura de los efectos aleatorios

Recordemos que para elegir la estructura de los efectos aleatorios es necesario incluir todos los posibles términos fijos y sus interacciones en el modelo (más allá del óptimo). Luego se comparan distintos modelos que varían en sus efectos aleatorios pero que mantienen la misma estructura fija por medio de REML. Los modelos que podemos plantear en este caso son: (1) un modelo sin efecto aleatorio (equivalente a un modelo lineal pero utilizando un estimador REML). Este modelo lo calcularemos con la función gls(); (2) un modelo que incluya la varianza aleatoria de la constante (gran media) dentro de cada bloque.

```
> lme.rabbit0 <- gls(gain ~ treat, data = rabbit)
> lme.rabbit1 <- lme(gain ~ treat, data = rabbit, random = ~1 |
+     block)
> anova(lme.rabbit0, lme.rabbit1)

            Model df      AIC      BIC    logLik   Test  L.Ratio p-value
lme.rabbit0     1  7 174.2621 182.5085 -80.13107
lme.rabbit1     2  8 166.3569 175.7813 -75.17844 1 vs 2 9.905255  0.0016
```

Vemos que el modelo con el efecto aleatorio bloque es más parsimonioso que el modelo sin el efecto aleatorio.

## 2.4. Paso 4: Elección de la estructura de los efectos fijos

Ya tenemos la estructura de los efectos aleatorios. Podemos utilizar el test t o el test F para comprobar la significación de la(s) variable(s) fija(s). Pero tal vez es mejor opción comparar modelos anidados como en el caso de los efectos aleatorios. Recordemos que, para hacer esto, debemos estimar los parámetros de los modelos utilizando un estimador de ML.

```
> lme.rabbit2 <- lme(gain ~ 1, data = rabbit, random = ~1 | block,
+     method = "ML")
> lme.rabbit3 <- lme(gain ~ treat, data = rabbit, random = ~1 |
+     block, method = "ML")
> anova(lme.rabbit2, lme.rabbit3)

            Model df      AIC      BIC    logLik   Test  L.Ratio p-value
lme.rabbit2     1  3 188.8307 193.0343 -91.41536
lme.rabbit3     2  8 183.7730 194.9825 -83.88648 1 vs 2 15.05776  0.0101
```

Y, de nuevo, vemos que el modelo con el efecto fijo del tratamiento es más parsimonioso que el modelo que contiene únicamente el efecto de la constante.

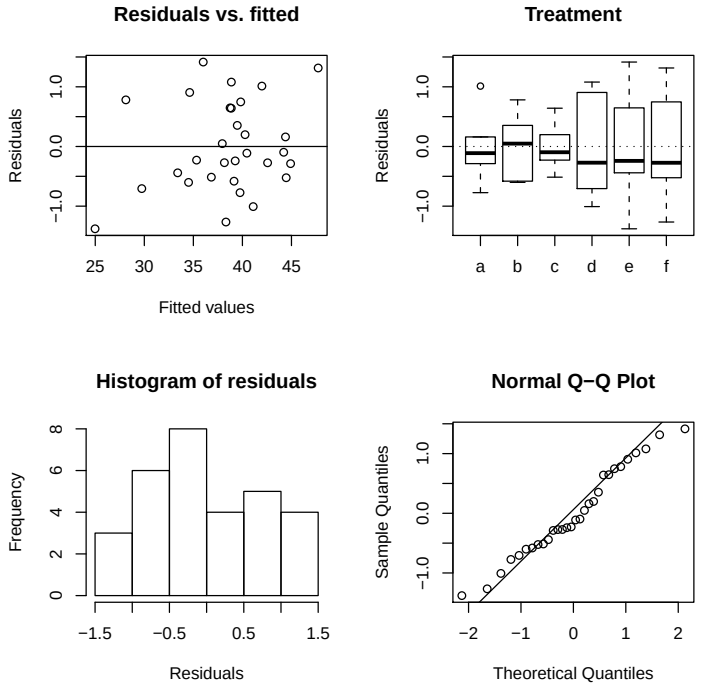
## 2.5. Paso 5: Presentación del modelo final con REML

Finalmente, presentamos el modelo final utilizando REML. Es necesario que veamos cómo es de adecuado el modelo de acuerdo con los supuestos del modelo, fundamentalmente normalidad y homocedasticidad.

```
> lme.rabbit <- lme(gain ~ treat, data = rabbit, random = ~1 |
+     block)
> anova(lme.rabbit)

            numDF denDF  F-value p-value
(Intercept)     1    15 590.5297  <.0001
treat           5    15   3.2818  0.0336

> summary(lme.rabbit)

Linear mixed-effects model fit by REML
 Data: rabbit
       AIC      BIC    logLik
  166.3569 175.7813 -75.17844

Random effects:
 Formula: ~1 | block
        (Intercept) Residual
StdDev:    4.657853 3.175519

Fixed effects: gain ~ treat
               Value Std.Error DF   t-value p-value
(Intercept) 39.53540  2.130338 15 18.558278  0.0000
treatb      -2.50718  2.208700 15 -1.135139  0.2741
treatc      -0.18407  2.208700 15 -0.083340  0.9347
treatd      -0.88516  2.208700 15 -0.400759  0.6942
treate      -5.64602  2.208700 15 -2.556263  0.0219
treatf       2.81003  2.208700 15  1.272254  0.2227
 Correlation:
       (Intr) treatb treatc treatd treate
treatb -0.518
treatc -0.518  0.500
treatd -0.518  0.500  0.500
treate -0.518  0.500  0.500  0.500
treatf -0.518  0.500  0.500  0.500  0.500

Standardized Within-Group Residuals:
       Min         Q1        Med         Q3        Max
-1.3794203 -0.5213453 -0.1701517  0.6456859  1.4148990

Number of Observations: 30
Number of Groups: 10
```

El modelo final sería del tipo $gain = 39{,}535 - 5{,}646 \cdot Trat_e$ (ya que los coeficientes para el resto de los tratamientos no son significativamente distintos de cero) con una varianza residual de $\sigma^2 = 3{,}175^2$ y una varianza para la constante de $\sigma_a^2 = 4{,}658^2$ (este es el efecto estimado del bloque en nuestro modelo).

```
> Res <- residuals(lme.rabbit, type = "normalized")
> Fit <- fitted(lme.rabbit)
> par(mfrow = c(2, 2))
> plot(Res ~ Fit, xlab = "Fitted values", ylab = "Residuals", main = "Residuals vs. fitted
> abline(h = 0)
> boxplot(Res ~ rabbit$treat, ylab = "Residuals", main = "Treatment")
> abline(h = 0, lty = 3)
> hist(Res, main = "Histogram of residuals", xlab = "Residuals")
> qqnorm(Res)
> qqline(Res)
```



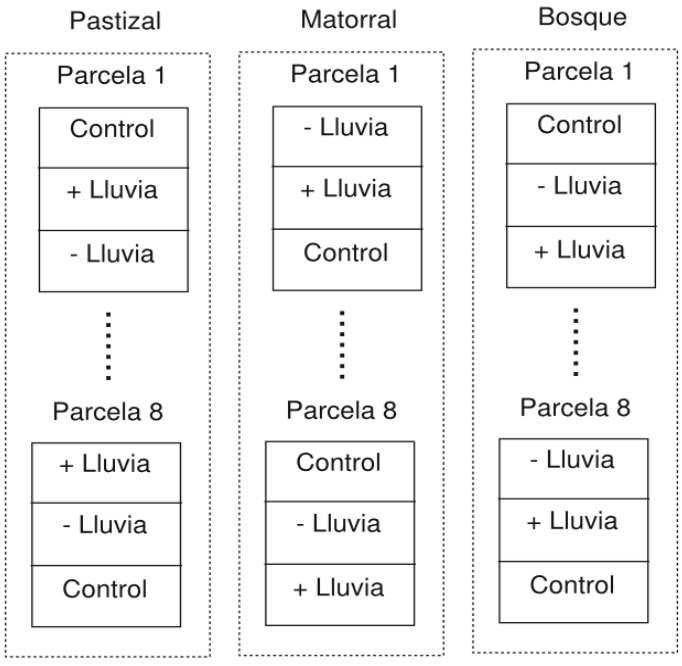

Parece que hay algo de heterocedasticidad debido a que no hay igualdad de varianzas entre los grupos. Además, podría haber también problemas de normalidad. No parece que haya problemas de linealidad. ¿Soluciones al problema? Ninguna necesariamente óptima: probar otro tipo de modelos como los modelos lineales generalizados con una distribución de errores diferente (por ejemplo, gamma), los modelos aditivos generalizados o modelos lineales mixtos no lineales. También podríamos utilizar alguna técnica no paramétrica como los árboles de regresión o, simplemente, quedarnos con este modelo a

pesar de sus limitaciones.

## 3. Diseños de medidas repetidas y split-plot

En el ejemplo anterior, tendríamos un diseño de medidas repetidas si en vez de tres conejos de cada camada, tomamos un único conejo y le aplicamos distintos tratamientos a cada uno de ellos en distintos momentos. No obstante, el análisis de los datos mediante el uso de modelos lineales mixtos hubiera sido exactamente igual. En el diseño por bloques asumimos que hay un efecto de la camada que afecta a la relación entre la ganancia de peso y el tratamiento. En el diseño de medidas repetidas asumimos que hay un efecto del individuo que afecta a la relación entre la ganancia de peso y el tratamiento.

Vamos a usar otro ejemplo ahora para ilustrar el diseño de medidas repetidas y su extensión incorporando factores inter-sujetos, el split-plot. El objetivo de este experimento es investigar si distintos escenarios de cambio climático (incluyendo más y menos lluvia estival) afectan a distintos parámetros ecofisiológicos de varias especies leñosas típicas del bosque mediterráneo, y si este efecto varía dependiendo del tipo de hábitat (pastizal, matorral, bosque)[2]. En este ejercicio nos centraremos en una única especie (*Quercus ilex*) y una única variable respuesta (biomasa). El diseño viene ilustrado de la siguiente forma:




[2]Los datos han sido cedidos por Luis Matías (Departamento de Ecología, Universidad de Granada). No se permite el uso de estos datos excepto para fines docentes sin el permiso del autor (lmatias@ugr.es).

Pero ¡cuidado! La parcela 1 de pastizal no tiene absolutamente nada que ver con la parcela 1 de matorral o bosque. Preparamos los datos:

```
> sn <- read.table("http://tinyurl.com/sn-mati", header = T)
> xtabs(Total.Biomass ~ Plot + Scenario + Habitat, data = sn)

, , Habitat = Forest

    Scenario
Plot Control Dry Summer Wet Summer
   1   1.389      1.084      0.985
   2   1.422      1.667      1.793
   3   1.189      1.356      1.636
   4   1.953      1.371      1.490
   5   0.849      1.369      1.143
   6   0.950      1.351      1.951
   7   1.410      2.913      2.303
   8   1.203      1.538      1.853

, , Habitat = Grassland

    Scenario
Plot Control Dry Summer Wet Summer
   1   0.000      2.331      4.908
   2   2.596      3.454      3.611
   3   1.534      2.765      3.517
   4   6.777      1.143      2.741
   5   1.627      2.849      2.860
   6   1.662      2.233      2.724
   7   0.000      2.005      1.841
   8   1.238      1.689      4.084

, , Habitat = Shrubland

    Scenario
Plot Control Dry Summer Wet Summer
   1   1.991      2.150      2.357
   2   2.056      1.845      2.354
   3   1.533      2.151      1.781
   4   1.881      1.558      2.218
   5   1.718      1.655      2.086
   6   1.117      2.323      1.865
   7   1.641      1.475      1.391
   8   0.986      2.003      1.322

> sn$Plot <- rep(1:24, each = 3)
> sn <- sn[!sn$Total.Biomass == 0, ]
```

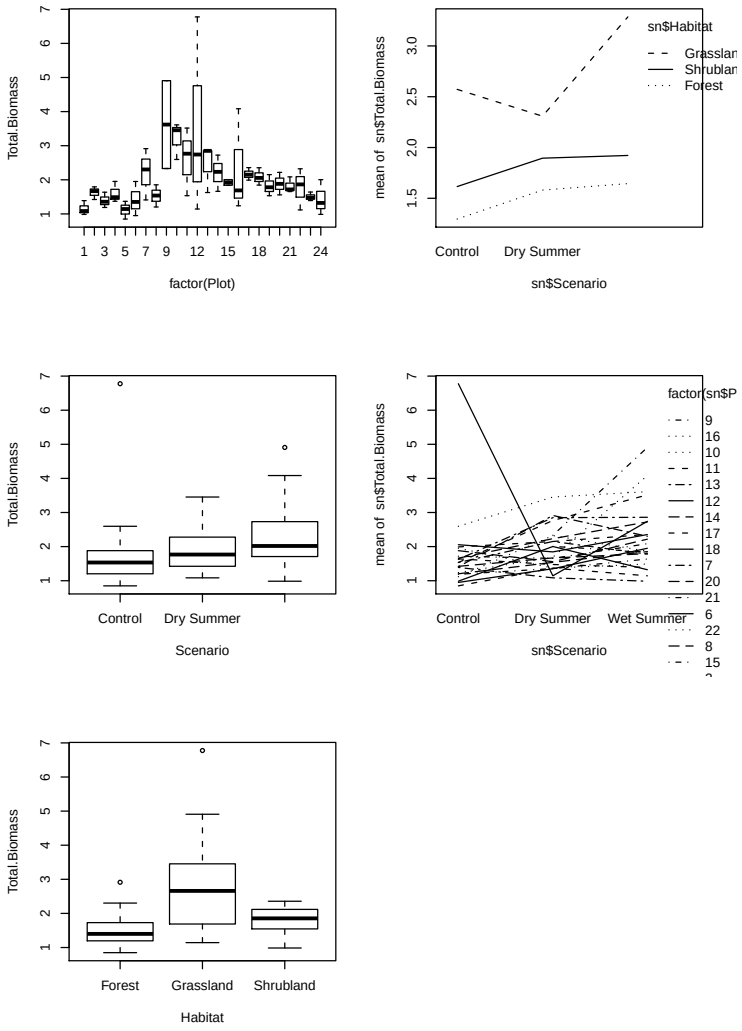
Eliminamos los valores de biomasa que valen 0 porque las semillas de estas muestras no han germinado y el problema aquí sería otro. Si el sujeto muestral

es la parcela, tenemos un factor de medidas repetidas (escenario) con tres niveles y un factor inter-sujetos (habitat) también con tres niveles. Si sólamente tuviéramos en cuenta el factor escenario tendríamos un diseño de medidas repetidas, pero como tenemos los dos tipos de factores (intra- e inter-sujetos) entonces el diseño se denomina de tipo split-plot.

## 3.1. Paso 1: Aplicación de un modelo lineal

Antes de ajustar un modelo lineal vamos a obtener algunos gráficos exploratorios. Como podemos ver, hay diferencias en la biomasa total dentro de cada parcela (pero recordemos que en cada parcela se experimentan los tres niveles del factor escenario, así que es más fácil ver este efecto en el término interacción porque aquí estamos promediando los valores de biomasa de plantas sometidas a sequía, lluvia estival y al control). También parece que hay un efecto positivo de la lluvia estival en la producción de biomasa leñosa, y que se produce más biomasa en el pastizal que en los otros dos tipos de hábitat. Los gráficos de interacción parecen indicar una interacción entre el hábitat y el escenario. La respuesta de la biomasa al escenario parece ser homogéneo dentro de cada parcela excepto por una parcela en dónde detectamos una respuesta muy distinta. Habría que ver qué pasa con estos datos. Tal vez haya algo anormal o algún error de medición o de muestreo, en cuyo caso deberíamos desestimar este dato en el análisis. Por el momento lo dejaremos. No tiene sentido plantear una interacción entre la parcela y el hábitat porque no son factores cruzados sino anidados (es decir, no todos los niveles del factor parcela están representados dentro del factor hábitat. De hecho las parcelas 1 a 8 se encuentran en un tipo de hábitat, las parcelas 9 a 16 en otro y las parcelas 17 a 24 en otro, así que la interacción no es posible).

```
> par(mfcol = c(3, 2))
> plot(Total.Biomass ~ factor(Plot), data = sn)
> plot(Total.Biomass ~ Scenario, data = sn)
> plot(Total.Biomass ~ Habitat, data = sn)
> interaction.plot(sn$Scenario, sn$Habitat, sn$Total.Biomass)
> interaction.plot(sn$Scenario, factor(sn$Plot), sn$Total.Biomass)
```




Vamos a analizar los datos con un modelo lineal. Como sólo hay un dato por cada parcela y escenario no es posible calcular la significación del término interacción entre el factor y el bloque (no hay réplicas). Por tanto el modelo que plantearemos será el siguiente:

```
> lm.sn <- lm(Total.Biomass ~ factor(Plot) + Scenario * Habitat,
+     data = sn)
```

```
> anova(lm.sn)

Analysis of Variance Table

Response: Total.Biomass
                 Df  Sum Sq Mean Sq F value  Pr(>F)
factor(Plot)     23 29.5922 1.28662  1.8075 0.04932 *
Scenario          2  2.7137 1.35683  1.9061 0.16192
Scenario:Habitat  4  2.3363 0.58408  0.8205 0.51981
Residuals        40 28.4733 0.71183
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

El bloque es significativo pero ni el escenario ni el hábitat lo son. Con este modelo estamos gastando 23 grados de libertad con el bloque (parcela) cuando podríamos utilizar sólamente uno estimando el efecto de la parcela como la varianza de todos los posibles coeficientes de los niveles de la parcela en su efecto sobre la constante (Intercept).

## 3.2. Paso 2: Ajuste de un modelo lineal mixto

Vamos ahora a ajustar el modelo mixto equivalente al modelo lineal anterior.

```
> library(nlme)
> lme.sn <- lme(Total.Biomass ~ Scenario * Habitat, random = ~1 |
+     factor(Plot), data = sn)
> anova(lme.sn)

                 numDF denDF  F-value p-value
(Intercept)          1    40 431.7217  <.0001
Scenario             2    40   2.5693  0.0892
Habitat              2    21  13.9081  0.0001
Scenario:Habitat     4    40   0.9049  0.4703

> summary(lme.sn)

Linear mixed-effects model fit by REML
 Data: sn
       AIC      BIC    logLik
  186.9631 210.1827 -82.48156

Random effects:
 Formula: ~1 | factor(Plot)
         (Intercept) Residual
StdDev: 2.953030e-05 0.804266

Fixed effects: Total.Biomass ~ Scenario * Habitat
```

```
                                        Value Std.Error DF   t-value p-value
(Intercept)                          1.2956250 0.2843510 40  4.556429  0.0000
ScenarioDry Summer                   0.2855000 0.4021330 40  0.709964  0.4818
ScenarioWet Summer                   0.3486250 0.4021330 40  0.866940  0.3911
HabitatGrassland                     1.2767083 0.4343533 21  2.939332  0.0078
HabitatShrubland                     0.3197500 0.4021330 21  0.795135  0.4354
ScenarioDry Summer:HabitatGrassland -0.5492083 0.5919237 40 -0.927836  0.3591
ScenarioWet Summer:HabitatGrassland  0.3647917 0.5919237 40  0.616282  0.5412
ScenarioDry Summer:HabitatShrubland -0.0058750 0.5687019 40 -0.010331  0.9918
ScenarioWet Summer:HabitatShrubland -0.0422500 0.5687019 40 -0.074292  0.9411
 Correlation:
                                    (Intr) ScnrDS ScnrWS HbttGr HbttSh SDS:HG
ScenarioDry Summer                  -0.707
ScenarioWet Summer                  -0.707  0.500
HabitatGrassland                    -0.655  0.463  0.463
HabitatShrubland                    -0.707  0.500  0.500  0.463
ScenarioDry Summer:HabitatGrassland  0.480 -0.679 -0.340 -0.734 -0.340
ScenarioWet Summer:HabitatGrassland  0.480 -0.340 -0.679 -0.734 -0.340  0.538
ScenarioDry Summer:HabitatShrubland  0.500 -0.707 -0.354 -0.327 -0.707  0.480
ScenarioWet Summer:HabitatShrubland  0.500 -0.354 -0.707 -0.327 -0.707  0.240
                                    SWS:HG SDS:HS
ScenarioDry Summer
ScenarioWet Summer
HabitatGrassland
HabitatShrubland
ScenarioDry Summer:HabitatGrassland
ScenarioWet Summer:HabitatGrassland
ScenarioDry Summer:HabitatShrubland  0.240
ScenarioWet Summer:HabitatShrubland  0.480  0.500

Standardized Within-Group Residuals:
        Min          Q1         Med          Q3         Max
-1.79635849 -0.52757734 -0.05789441  0.32727824  5.22795549

Number of Observations: 70
Number of Groups: 24
```

Aunque este no es todavía el modelo definitivo es curioso observar que ahora tanto el hábitat como el escenario (con un $\alpha = 0{,}10$) son significativos (la interacción entre ambos no lo es). En cuanto a su componente fija, este sería el modelo más allá del óptimo (el modelo completo). Por lo tanto podemos utilizar esta estructura de los efectos fijos para elegir la estructura de los efectos aleatorios más adecuada.

## 3.3. Paso 3: Elección de la estructura de los efectos aleatorios

Los modelos que podemos plantear en este caso son: (1) un modelo sin efecto aleatorio (equivalente a un modelo lineal pero utilizando un estimador REML).

Este modelo lo calcularemos con la función gls(); (2) un modelo que incluya la varianza aleatoria de la constante (gran media) dentro de cada bloque; (3) un modelo que incluya la varianza de la constante y las varianzas aleatorias para cada uno de los niveles del factor habitat menos uno. Este último modelo asumiría que la respuesta (biomasa) a los distintos niveles del factor escenario no es igual en todas las parcelas, sino que varía de manera aleatoria dentro de cada una.

```
> lme.sn0 <- gls(Total.Biomass ~ Scenario * Habitat, data = sn)
> lme.sn1 <- lme(Total.Biomass ~ Scenario * Habitat, random = ~1 |
+     factor(Plot), data = sn)
> lme.sn2 <- lme(Total.Biomass ~ Scenario * Habitat, random = ~Scenario |
+     factor(Plot), data = sn)
> anova(lme.sn0, lme.sn1, lme.sn2)

        Model df      AIC      BIC    logLik   Test  L.Ratio p-value
lme.sn0     1 10 184.9631 206.0719 -82.48156
lme.sn1     2 11 186.9631 210.1827 -82.48156 1 vs 2  0.00000  0.9999
lme.sn2     3 16 179.8083 213.5822 -73.90413 2 vs 3 17.15485  0.0042
```

Vemos que el modelo lme.sn2, que incluye la varianza de la constante y las varianzas aleatorias para cada uno de los niveles del factor habitat es el más parsimonioso. Esto es como asumir que hay una interacción entre la parcela y el factor escenario. Con el modelo lineal no teníamos réplicas como para calcular un coeficinte del efecto de cada parcela en cada nivel del factor escenario (esto serían (24-1 * 3-1 coeficientes = 69 coeficientes más en el modelo). Pero como aquí lo que estamos calculando son solamente dos coeficientes (la varianza aleatoria para dos de los tres niveles del factor, que indicaría como varían estos coeficientes de manera aleatoria dentro de cada parcela), entonces sí que tenemos capacidad de estimación.

## 3.4. Paso 4: Elección de la estructura de los efectos fijos

Ya tenemos la estructura de los efectos aleatorios. Podemos utilizar el test t o el test F para comprobar la significación de la(s) variable(s) fija(s). Pero tal vez es mejor opción comparar modelos anidados como en el caso de los efectos aleatorios. Recordemos que, para hacer esto, debemos estimar los parámetros de los modelos utilizando un estimador de ML.

```
> lme.sn3 <- lme(Total.Biomass ~ 1, random = ~Scenario | factor(Plot),
+     data = sn, method = "ML")
> lme.sn4 <- lme(Total.Biomass ~ Scenario, random = ~Scenario |
+     factor(Plot), data = sn, method = "ML")
> lme.sn5 <- lme(Total.Biomass ~ Scenario + Habitat, random = ~Scenario |
+     factor(Plot), data = sn, method = "ML")
> lme.sn6 <- lme(Total.Biomass ~ Scenario * Habitat, random = ~Scenario |
+     factor(Plot), data = sn, method = "ML")
> anova(lme.sn3, lme.sn4, lme.sn5, lme.sn6)
```

```
        Model df      AIC      BIC    logLik   Test   L.Ratio p-value
lme.sn3     1  8 188.0546 206.0425 -86.02728
lme.sn4     2 10 185.6914 208.1763 -82.84568 1 vs 2  6.363196  0.0415
lme.sn5     3 12 171.5988 198.5807 -73.79939 2 vs 3 18.092591  0.0001
lme.sn6     4 16 171.0547 207.0307 -69.52736 3 vs 4  8.544054  0.0736
```

El modelo factorial sin interacción sería el más óptimo de acuerdo a los criterios de información AIC y BIC y al test de verosimilitud. Esto indicaría un efecto significativo de los dos factores sobre la biomasa, pero no de la interacción entre ambos, si bien esta es marginalmente significativa (p-valor=0.0736).

## 3.5. Paso 5: Presentación del modelo final con REML

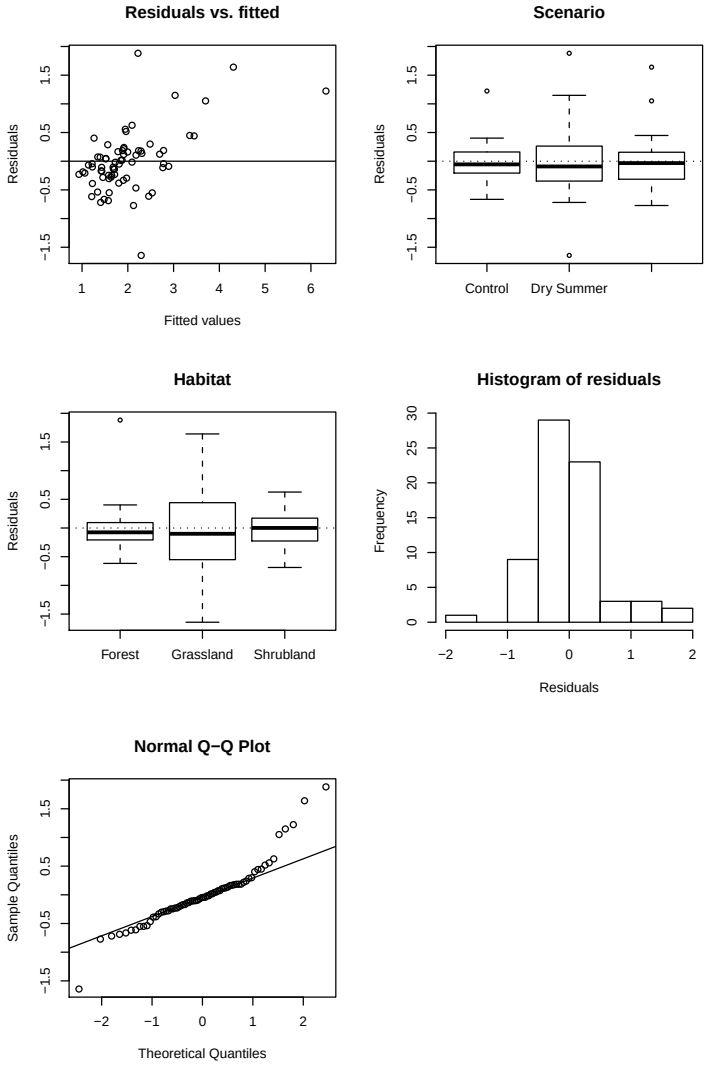
Finalmente, presentamos el modelo final utilizando REML.

```
> lme.sn <- lme(Total.Biomass ~ Scenario + Habitat, random = ~Scenario |
+     factor(Plot), data = sn)
> anova(lme.sn)

            numDF denDF  F-value p-value
(Intercept)     1    44 687.5112  <.0001
Scenario        2    44   3.4570  0.0403
Habitat         2    21  17.7142  <.0001

> summary(lme.sn)

Linear mixed-effects model fit by REML
 Data: sn
       AIC      BIC    logLik
  181.0709 207.1635 -78.53544

Random effects:
 Formula: ~Scenario | factor(Plot)
 Structure: General positive-definite, Log-Cholesky parametrization
                   StdDev    Corr
(Intercept)        1.0137255 (Intr) ScnrDS
ScenarioDry Summer 1.3007944 -0.958
ScenarioWet Summer 1.2176057 -0.882  0.875
Residual           0.3670853

Fixed effects: Total.Biomass ~ Scenario + Habitat
                       Value Std.Error DF  t-value p-value
(Intercept)        1.3759974 0.2485840 44 5.535341  0.0000
ScenarioDry Summer 0.0883965 0.2928952 44 0.301803  0.7642
ScenarioWet Summer 0.4440632 0.2775939 44 1.599687  0.1168
HabitatGrassland   1.0851431 0.1867884 21 5.809479  0.0000
HabitatShrubland   0.3064250 0.1814293 21 1.688950  0.1060
```

```
 Correlation:
                   (Intr) ScnrDS ScnrWS HbttGr
ScenarioDry Summer -0.843
ScenarioWet Summer -0.783  0.829
HabitatGrassland   -0.308 -0.052 -0.055
HabitatShrubland   -0.365  0.000  0.000  0.486

Standardized Within-Group Residuals:
        Min          Q1         Med          Q3         Max
-1.64241140 -0.26996602 -0.04685996  0.18271663  1.88193822

Number of Observations: 70
Number of Groups: 24
```

Tanto el escenario y el hábitat son significativos y también hay un efecto claro de la parcela que afecta a todos los valores de biomasa dentro de cada parcela (constante) y a cada valor de biomasa dentro de cada nivel del factor escenario (coeficiente de los niveles del factor escenario). Por último, es necesario que veamos cómo es de adecuado el modelo de acuerdo con los supuestos del modelo, fundamentalmente normalidad y homocedasticidad.

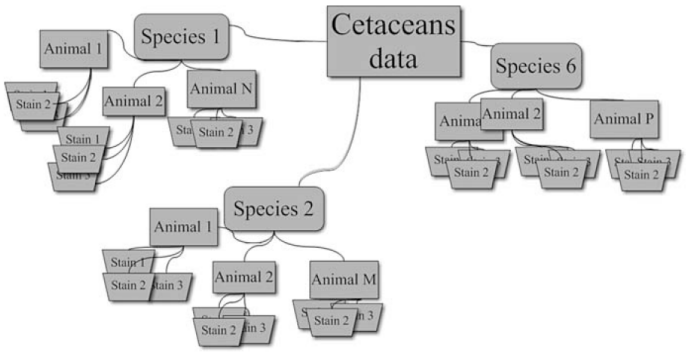
```
> Res <- residuals(lme.sn, type = "normalized")
> Fit <- fitted(lme.sn)
> par(mfrow = c(3, 2))
> plot(Res ~ Fit, xlab = "Fitted values", ylab = "Residuals", main = "Residuals vs. fitted
> abline(h = 0)
> boxplot(Res ~ sn$Scenario, ylab = "Residuals", main = "Scenario")
> abline(h = 0, lty = 3)
> boxplot(Res ~ sn$Habitat, ylab = "Residuals", main = "Habitat")
> abline(h = 0, lty = 3)
> hist(Res, main = "Histogram of residuals", xlab = "Residuals")
> qqnorm(Res)
> qqline(Res)
```



Parece que hay algo de heterocedasticidad debido a que no hay igualdad de varianzas entre los grupos. Además, podría haber también problemas de normalidad y linealidad. Si incluimos la interacción en el modelo (que era marginalmente significativa en nuestro test de verosimilitud) el modelo se ajusta mejor a los supuestos de homocedasticidad y normalidad. Esto ocurre a veces cuando hay alguna variable que guarda relación con la estructura de los residuos, tal podría ser el caso de la interacción entre la parcela y el escenario. Por tanto, una opción viable sería modificar el modelo e incluir dicha interacción en los efectos fijos.

# 4. Diseños factoriales anidados o jerarquizados

En este caso de estudio vamos a tomar un ejemplo recogido en el capítulo 20 de Zuur et al. (2009) que hace referencia a la tesis doctoral de Patricia Lastra Luque (2008)[3]. La estimación de la edad en los odontocetos se basa en el cálculo de los grupos de capas de crecimiento que se depositan en estructuras de registro como los dientes. Generalmente, las secciones dentales se obtienen utilizando un microtomo criostato. Sin embargo, algunos investigadores prefieren obtener secciones finas utilizando un microtomo de parafina tradicional. Hay escasa información disponible acerca de la aplicación de esta técnica sobre los dientes de los delfines. El objetivo de este estudio era investigar si la técnica de parafina podía verse afectada por el método de tinción, controlando los efectos de la especie, el sexo y la región de procedencia. Para ello se tomaron muestras de dientes de varios individuos de diferentes especies de odontocetos[4] en España y Escocia. El factor de interés es el método de tinción (Mayer, Elrich, Toluidine) y la variable respuesta la edad estimada del cetáceo. Otras variables explicativas son el sexo o la región de procedencia del cetáceo (España, Escocia). El esquema de la estructura de los datos sería la siguiente:




Leemos los datos.

[3]Luque, P.L. (2008) Age determination and interpretation of mineralization anomalies in teeth of small cetaceans. Tesis doctoral, Universidad de Vigo, España.

[4]Cetáceos con dientes como los delfines, las marsopas, los cachalotes o las belugas.

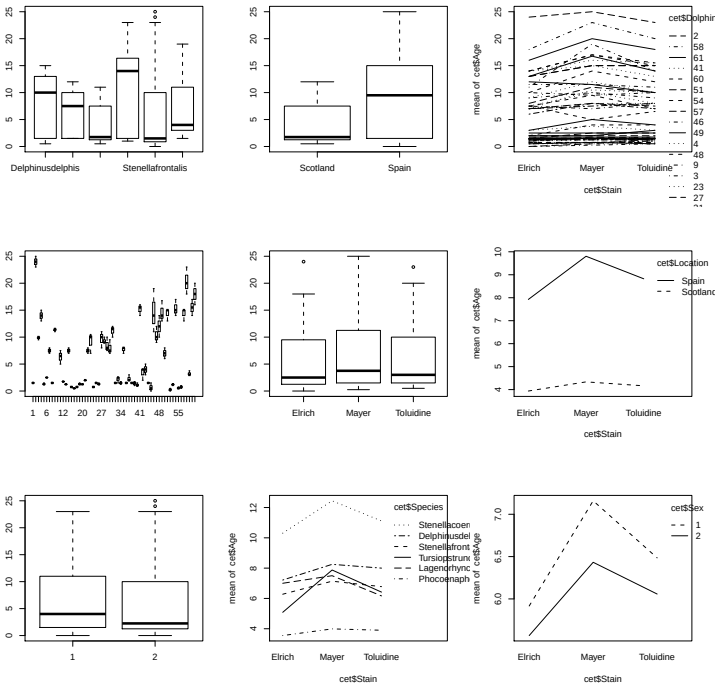
```
> cet <- read.table("http://tinyurl.com/cetaceos", header = T)
> cet$DolphinID <- factor(cet$DolphinID)
> str(cet)

'data.frame':        180 obs. of  6 variables:
 $ DolphinID: Factor w/ 60 levels "1","2","3","4",..: 9 9 9 24 24 24 36 36 36 46 ...
 $ Species  : Factor w/ 6 levels "Delphinusdelphis",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Age      : num  11.5 11.5 11 1.5 1.5 1.5 2 2 3 12 ...
 $ Sex      : int  1 1 1 1 1 1 1 1 1 2 ...
 $ Stain    : Factor w/ 3 levels "Elrich","Mayer",..: 2 1 3 2 1 3 2 1 3 2 ...
 $ Location : Factor w/ 2 levels "Scotland","Spain": 1 1 1 1 1 1 1 1 1 2 ...
```

Si exploramos más de cerca los datos vemos que hay algunos 0 en la variable Sex. Esto es, el sexo de algunos individuos no ha podido ser determinado correctamente. Tenemos que eliminar estos datos antes de continuar con el análisis.

```
> I <- cet$Sex == 0
> cet <- cet[!I, ]
> cet$Sex <- factor(cet$Sex)
```

Antes de ajustar un modelo lineal vamos a obtener algunos gráficos exploratorios. Estas gráficas muestran que hay mucha variación en cuanto a la edad estimada para los distintos especímenes. Hay que resaltar que hay tres muestras por especimen. La misma gráfica para las especies muestra que hay mucha menor variación entre especies. Esto no es raro ya que cada animal va a tener una unica edad estimada tres veces, pero es muy posible que dentro de cada especie tengamos muestras de especímenes que cubran gran parte del rango de edades presente en la población. Incluso aunque alguna especie sea más longeva que otra, va a haber mucha superposición en las edades presentes en las muestras de los distintos especímenes. También parece haber mayor variación en las edades registradas en Escocia que en España por lo que puede que sea necesario especificar distintas varianzas en las distintas regiones. Por otro lado, no parece, a simple vista, que haya una respuesta diferencial de la edad estimada frente al método de tinción en función del nivel de ninguno de los otros factores (especie, especimen, sexo, región).

```
> par(mfcol = c(3, 3))
> boxplot(Age ~ Species, data = cet)
> boxplot(Age ~ DolphinID, data = cet)
> boxplot(Age ~ Sex, data = cet)
> boxplot(Age ~ Location, data = cet)
> boxplot(Age ~ Stain, data = cet)
> interaction.plot(cet$Stain, cet$Species, cet$Age)
> interaction.plot(cet$Stain, cet$DolphinID, cet$Age)
> interaction.plot(cet$Stain, cet$Location, cet$Age)
> interaction.plot(cet$Stain, cet$Sex, cet$Age)
```




## 4.1. Paso 1: Aplicación de un modelo lineal

Vamos a analizar los datos con un modelo lineal. En este modelo vamos a calcular el efecto del método de tinción, la región y el sexo, sin considerar la especia ni el individuo. También vamos a contemplar todas las posibles interacciones entre los factores dos a dos y la interacción entre los tres factores.

```
> f1 <- formula(Age ~ Sex * Stain * Location)
> lm.cet <- lm(f1, data = cet)
> anova(lm.cet)

Analysis of Variance Table

Response: Age
```

```
                   Df Sum Sq Mean Sq F value    Pr(>F)
Sex                 1   10.9   10.95  0.3354    0.5633
Stain               2   34.2   17.09  0.5234    0.5935
Location            1 1024.1 1024.15 31.3711 8.741e-08 ***
Sex:Stain           2    1.2    0.58  0.0177    0.9825
Sex:Location        1   31.7   31.75  0.9724    0.3255
Stain:Location      2   17.7    8.85  0.2710    0.7629
Sex:Stain:Location  2    3.5    1.76  0.0539    0.9476
Residuals         165 5386.6   32.65
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

## 4.2. Paso 2: Ajuste de un modelo lineal mixto

Vamos ahora a ajustar el modelo mixto equivalente al modelo lineal anterior, pero considerando los efectos aleatorios, que en este caso vienen dados por el especimen y la especie. El especimen sería un factor de medidas repetidas ya que dentro de cada especimen comprobamos todos los niveles del factor método de teñido. Pero los especímenes están anidados dentro del factor especie. Esto es porque no todos los niveles del factor especimen están recogidos en todos los niveles del factor especie. La formulación del modelo sería así:

```
> library(nlme)
> lme.cet <- lme(f1, random = ~1 | Species/DolphinID, data = cet)
> anova(lme.cet)

                  numDF denDF  F-value p-value
(Intercept)           1   110 60.19087  <.0001
Sex                   1    50  0.05354  0.8180
Stain                 2   110 24.86562  <.0001
Location              1    50  8.18786  0.0061
Sex:Stain             2   110  0.83863  0.4350
Sex:Location          1    50  0.28262  0.5973
Stain:Location        2   110 12.87702  <.0001
Sex:Stain:Location    2   110  2.56000  0.0819

> summary(lme.cet)

Linear mixed-effects model fit by REML
 Data: cet
       AIC      BIC    logLik
  740.3277 786.9168 -355.1638

Random effects:
 Formula: ~1 | Species
        (Intercept)
StdDev:    0.898073
```

```
 Formula: ~1 | DolphinID %in% Species
        (Intercept) Residual
StdDev:    5.615181 0.828939

Fixed effects: list(f1)
                                    Value Std.Error  DF   t-value p-value
(Intercept)                      4.142943 1.4148661 110  2.928152  0.0041
Sex2                            -0.378897 2.0844159  50 -0.181776  0.8565
StainMayer                       0.375000 0.2621335 110  1.430569  0.1554
StainToluidine                   0.162500 0.2621335 110  0.619913  0.5366
LocationSpain                    4.480572 2.1746964  50  2.060321  0.0446
Sex2:StainMayer                  0.062500 0.4280622 110  0.146007  0.8842
Sex2:StainToluidine              0.170833 0.4280622 110  0.399085  0.6906
Sex2:LocationSpain              -0.902499 3.0639805  50 -0.294551  0.7696
StainMayer:LocationSpain         2.201923 0.4176455 110  5.272229  0.0000
StainToluidine:LocationSpain     1.029808 0.4176455 110  2.465746  0.0152
Sex2:StainMayer:LocationSpain   -1.407280 0.6221848 110 -2.261836  0.0257
Sex2:StainToluidine:LocationSpain -0.738141 0.6221848 110 -1.186369  0.2380
 Correlation:
                                (Intr) Sex2   StnMyr StnTld LctnSp Sx2:SM
Sex2                            -0.517
StainMayer                      -0.093  0.063
StainToluidine                  -0.093  0.063  0.500
LocationSpain                   -0.642  0.332  0.060  0.060
Sex2:StainMayer                  0.057 -0.103 -0.612 -0.306 -0.037
Sex2:StainToluidine              0.057 -0.103 -0.306 -0.612 -0.037  0.500
Sex2:LocationSpain               0.356 -0.682 -0.043 -0.043 -0.622  0.070
StainMayer:LocationSpain         0.058 -0.039 -0.628 -0.314 -0.096  0.384
StainToluidine:LocationSpain     0.058 -0.039 -0.314 -0.628 -0.096  0.192
Sex2:StainMayer:LocationSpain   -0.039  0.071  0.421  0.211  0.064 -0.688
Sex2:StainToluidine:LocationSpain -0.039  0.071  0.211  0.421  0.064 -0.344
                                Sx2:ST Sx2:LS StM:LS StT:LS S2:SM:
Sex2
StainMayer
StainToluidine
LocationSpain
Sex2:StainMayer
Sex2:StainToluidine
Sex2:LocationSpain               0.070
StainMayer:LocationSpain         0.192  0.068
StainToluidine:LocationSpain     0.384  0.068  0.500
Sex2:StainMayer:LocationSpain   -0.344 -0.102 -0.671 -0.336
Sex2:StainToluidine:LocationSpain -0.688 -0.102 -0.336 -0.671  0.500

Standardized Within-Group Residuals:
        Min          Q1         Med          Q3         Max
-2.86316217 -0.35443083 -0.00631148  0.29365611  3.67902864

Number of Observations: 177
Number of Groups:
```

```
Species DolphinID %in% Species
      6                    59
```

## 4.3. Paso 3: Elección de la estructura de los efectos aleatorios

Los modelos que podemos plantear en este caso son: (1) un modelo sin efecto aleatorio (equivalente a un modelo lineal pero utilizando un estimador REML). Este modelo lo calcularemos con la función gls(); (2) un modelo que incluya la varianza aleatoria de la constante (gran media) dentro de cada especimen (anidado dentro de especie) y de cada especie (por tanto serían dos coeficientes de varianza).

En los gráficos de perfil no hemos visto aparentemente ninguna interacción entre el efecto de la tinción y ninguno de los otros factores sobre la edad. En un modelo lineal mixto la interacción entre el método de tinción y los factores aleatorios se estimaría por medio de las varianzas aleatorias estimadas para cada uno de los niveles del factor (método de teñido) menos uno. Pero como a priori no parece haber una interacción, podríamos obviar este modelo más complejo o, simplemente incluirlo y ver si es mejor o peor que los otros dos modelos comparándolos mediante el test de verosimilitud.

```
> lme.cet0 <- gls(f1, data = cet)
> lme.cet1 <- lme(f1, random = ~1 | Species/DolphinID, data = cet)
```

Vemos que esto da un error para el modelo lme.cet2. Lo que ocurre es que hacen falta más iteraciones para poder estimar los coeficientes aleatorios por medio del estimador REML. Para modificar el número de iteraciones en el proceso de estimación de los coeficientes aleatorios del modelo podemos utilizar la función lmeControl().

```
> lmc <- lmeControl(niterEM = 5200, msMaxIter = 5200)
> lme.cet2 <- lme(f1, random = ~Stain | Species/DolphinID, data = cet,
+     control = lmc)
> anova(lme.cet0, lme.cet1, lme.cet2)

         Model df       AIC       BIC    logLik   Test  L.Ratio p-value
lme.cet0     1 13 1101.4488 1141.8261 -537.7244
lme.cet1     2 15  740.3277  786.9168 -355.1638 1 vs 2 365.1212  <.0001
lme.cet2     3 25  704.9049  782.5536 -327.4525 2 vs 3  55.4227  <.0001
```

Vemos que el modelo lme.cet1, que contiene dos términos aleatorios para la constante (uno para el especimen y otro para la especie) es mucho más adecuado que el modelo sin efectos aleatorios (el AIC disminuye de 1101.45 a 740.33). Cuando incorporamos además el efecto de la varianza sobre los coeficientes de los niveles del factor principal (método de teñido), el modelo mejora un poco más (lme.cet2), pero el cambio más sustancial se produce al incluir la varianza atribuible al efecto de los dos factores aleatorios (uno anidado dentro del otro) sobre la constante del modelo. Como la estimación de los coeficientes de este modelo es costosa y su interpretabilidad es menor, vamos a quedarnos con el modelo lme.cet1.

## 4.4. Paso 4: Elección de la estructura de los efectos fijos

Ya tenemos la estructura de los efectos aleatorios. Vamos ahora a comparar modelos anidados como en el caso de los efectos aleatorios. Recordemos que, para hacer esto, debemos estimar los parámetros de los modelos utilizando un estimador de ML.

```
> lme.cet3 <- lme(Age ~ 1, random = ~1 | Species/DolphinID, data = cet,
+     method = "ML")
> lme.cet4 <- lme(Age ~ Sex, random = ~1 | Species/DolphinID, data = cet,
+     method = "ML")
> lme.cet5 <- lme(Age ~ Stain, random = ~1 | Species/DolphinID,
+     data = cet, method = "ML")
> lme.cet6 <- lme(Age ~ Stain + Location, random = ~1 | Species/DolphinID,
+     data = cet, method = "ML")
> lme.cet7 <- lme(Age ~ Stain * Location, random = ~1 | Species/DolphinID,
+     data = cet, method = "ML")
> lme.cet8 <- lme(Age ~ Stain * Location + Sex:Stain, random = ~1 |
+     Species/DolphinID, data = cet, method = "ML")
> lme.cet9 <- lme(Age ~ Stain * Location + Sex:Location, random = ~1 |
+     Species/DolphinID, data = cet, method = "ML")
> lme.cet10 <- lme(Age ~ Stain * Location + Sex:Location:Stain,
+     random = ~1 | Species/DolphinID, data = cet, method = "ML")
> anova(lme.cet3, lme.cet4, lme.cet5, lme.cet6, lme.cet7, lme.cet8,
+     lme.cet9, lme.cet10)
```

```
          Model df      AIC      BIC    logLik   Test  L.Ratio p-value
lme.cet3      1  4 796.8521 809.5567 -394.4260
lme.cet4      2  5 798.8503 814.7311 -394.4252 1 vs 2  0.00176  0.9665
lme.cet5      3  6 765.5389 784.5958 -376.7695 2 vs 3 35.31138  <.0001
lme.cet6      4  7 760.8669 783.0999 -373.4334 3 vs 4  6.67206  0.0098
lme.cet7      5  9 743.6632 772.2486 -362.8316 4 vs 5 21.20366  <.0001
lme.cet8      6 12 744.9647 783.0785 -360.4824 5 vs 6  4.69846  0.1953
lme.cet9      7 11 746.6293 781.5669 -362.3146 6 vs 7  3.66451  0.0556
lme.cet10     8 15 745.2448 792.8870 -357.6224 7 vs 8  9.38450  0.0522
```

El modelo más parsimonioso sería lme.cet7, con el método de teñido, la región y la interacción entre ambos factores como efectos fijos significativos.

## 4.5. Paso 5: Presentación del modelo final con REML

Finalmente, presentamos el modelo final utilizando REML.

```
> lme.cet <- lme(Age ~ Stain * Location, random = ~1 | Species/DolphinID,
+     data = cet)
> anova(lme.cet)
```

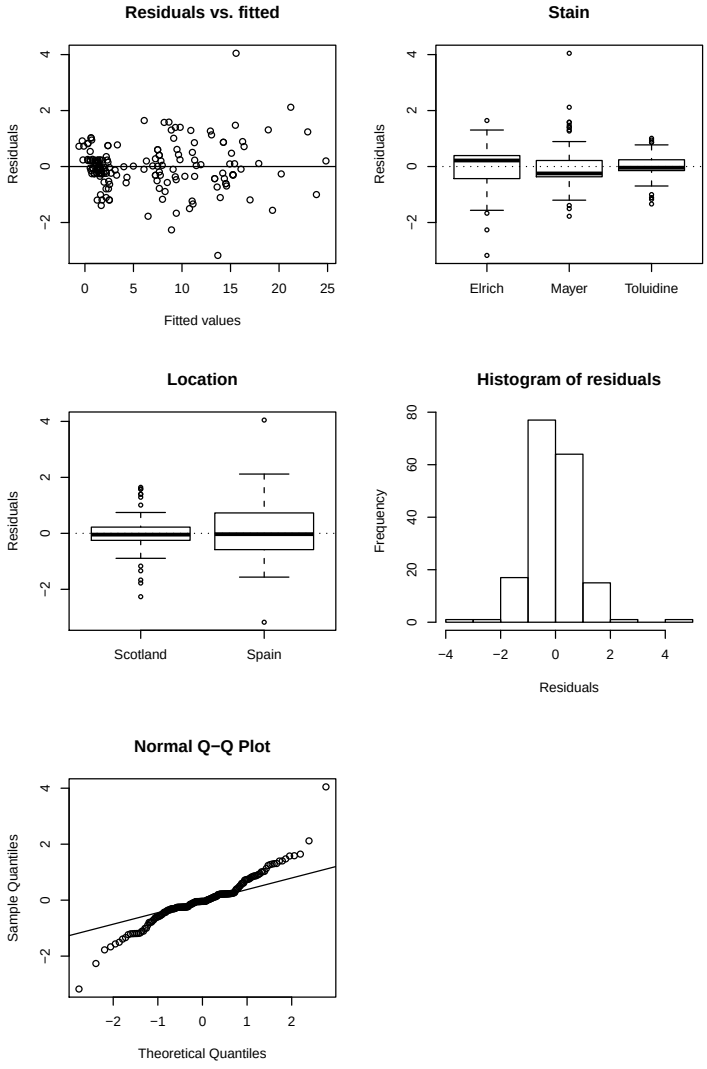
```
               numDF denDF  F-value p-value
(Intercept)        1   114 52.17741  <.0001
Stain              2   114 23.79970  <.0001
Location           1    52  6.64358  0.0128
Stain:Location     2   114 11.22036  <.0001

> summary(lme.cet)

Linear mixed-effects model fit by REML
 Data: cet
       AIC      BIC    logLik
  744.9385 773.2135 -363.4693

Random effects:
 Formula: ~1 | Species
        (Intercept)
StdDev:    1.287754

 Formula: ~1 | DolphinID %in% Species
        (Intercept)  Residual
StdDev:    5.515003 0.8472985

Fixed effects: Age ~ Stain * Location
                               Value Std.Error  DF  t-value p-value
(Intercept)                4.049757 1.3567566 114 2.984881  0.0035
StainMayer                 0.398438 0.2118246 114 1.880978  0.0625
StainToluidine             0.226563 0.2118246 114 1.069576  0.2871
LocationSpain              3.928177 1.8114660  52 2.168507  0.0347
StainMayer:LocationSpain   1.481192 0.3131271 114 4.730323  0.0000
StainToluidine:LocationSpain 0.671586 0.3131271 114 2.144770  0.0341
 Correlation:
                           (Intr) StnMyr StnTld LctnSp StM:LS
StainMayer                 -0.078
StainToluidine             -0.078  0.500
LocationSpain              -0.720  0.058  0.058
StainMayer:LocationSpain    0.053 -0.676 -0.338 -0.086
StainToluidine:LocationSpain  0.053 -0.338 -0.676 -0.086  0.500

Standardized Within-Group Residuals:
        Min          Q1         Med          Q3         Max
-3.17606501 -0.31200647 -0.04457005  0.24460497  4.04732803

Number of Observations: 177
Number of Groups:
               Species DolphinID %in% Species
                     6                    59
```

El método de teñido puede afectar la estimación que hagamos de la edad, con el método Mayer aumentando de manera (ligeramente) significativa la

estimación de la edad del cetáceo con respecto a los otros dos métodos. También la estimación de la edad está condicionada por la región geográfica, con una edad estimada mayor en España que en Escocia. Si vemos el gráfico de interacción de estos dos factores podemos observar que el efecto más destacado del método de teñido Mayer sobre la estimación de la edad se produce en las muestras de cetáceos tomadas en España mientras que en Escocia este efecto no es tan marcado y apenas se perciben diferencias entre los tres métodos. Esto puede ser debido a que en España los individuos muestreados son, en promedio, más mayores y podría ser que la estimación del método Mayer se vea afectada por la edad del especimen.

Debemos también comprobar la idoneidad del modelo desde el punto de vista de los supuestos de normalidad, homocedasticidad y linealidad.

```
> Res <- residuals(lme.cet, type = "normalized")
> Fit <- fitted(lme.cet)
> par(mfrow = c(3, 2))
> plot(Res ~ Fit, xlab = "Fitted values", ylab = "Residuals", main = "Residuals vs. fitted
> abline(h = 0)
> boxplot(Res ~ cet$Stain, ylab = "Residuals", main = "Stain")
> abline(h = 0, lty = 3)
> boxplot(Res ~ cet$Location, ylab = "Residuals", main = "Location")
> abline(h = 0, lty = 3)
> hist(Res, main = "Histogram of residuals", xlab = "Residuals")
> qqnorm(Res)
> qqline(Res)
```



Parece que hay heterocedasticidad en los residuos de las distintas regiones geográficas y también en los métodos de teñido. En el conjunto de los residuos no se detecta este problema. Por otro lado, el histograma de los residuos tiene una forma un poco elongada (posibles problemas de leptokurtosis) y el qq-plot indica una falta de linealidad. Para corregir la falta de homocedasticidad se podría incluir una componente en el modelo que indicase diferencia de varianzas entre diferentes regiones geográficas con el argumento weights de la función lme() (ver Zuur *et al.* 2009, pp. 464), si bien esto no lo vamos a ver aquí. También mejoraría el modelo si incorporamos la estructura de efectos aleatorios más compleja que vimos en el modelo lme.cet2. Ahora bien, si decidimos quedarnos con esta estructura de efectos aleatorios deberíamos repetir los pasos 4 y 5 para seleccionar de nuevo la estructura más adecuada para los efectos fijos.

# 5. Más ejemplos

Se pueden encontrar más ejemplos resueltos en
http://curso-r-uah2010.wikispaces.com/.