Curso: 2015 / 2016
18 y 19 de febrero de 2016

Profesores: José Francisco Calvo Sendín jfcalvo@um.es
José Antonio Palazón Ferrando palazon@um.es
Paqui Carreño Fructuoso mariafra@um.es

Guion de ejercicios prácticos

Para la realización de los ejercicios del seminario utilizaremos el software estadístico `R`, que puede descargarse gratuitamente desde la página http://cran.r-project.org/. También podemos utilizar el software `RStudio` (http://www.rstudio.com/).

Antes de empezar

Una vez iniciado `R` o `RStudio` debemos cargar el archivo de datos accediendo al servidor con la siguiente función:

```
load(url("http://www.um.es/docencia/emc/DEyAE.RData"))
```

Este archivo contiene todos los datos de ejemplo y una función necesaria para el desarrollo de los ejercicios propuestos. Podemos ver todos los objetos cargados con `ls()`.

Alternativamente, podemos descargar el archivo en nuestro ordenador accediendo con un navegador de internet a http://www.um.es/docencia/emc/DEyAE.RData. A continuación, una vez iniciado `R`, es conveniente cambiar el directorio de trabajo a dicha carpeta (*>Archivo >Cambiar dir…*), de forma que podamos acceder cómodamente al archivo descargado: *>Archivo >Cargar área de trabajo…*

Para el desarrollo de los ejercicios incluidos en este guion también se necesita la instalación adicional en `R` de varias librerías no instaladas por defecto: `lme4`, `car`, `nnet` y `AICcmodavg`. Para ello podemos utilizar, por ejemplo:

```
install.packages("lme4")
```

ANOVA, regresión lineal y ANCOVA

Para análisis de la varianza y la covarianza utilizaremos el objeto `ejemplo` (datos simulados).

```
ejemplo
attach(ejemplo)
plot(cov, y, col=trat)
```

Un ANOVA sencillo:

```
summary(aov(y ~ trat))
tapply(y, trat, mean)
```

Es equivalente a un test de la *t*:

```
t.test(y ~ trat, var.equal=T)
```

Un test no paramétrico:

```
kruskal.test(y ~ trat)
```

Un test de aleatorización con la función *ad hoc* MonteCarlo:

```
MonteCarlo
sample(trat)
MonteCarlo(y, trat)
```

Con una variable cuantitativa haremos una regresión (modelo lineal):

```
summary(lm(y ~ cov))
plot(cov, y, col=trat)
abline(24, 2.5, col="blue")
```

También podemos usar un modelo lineal para hacer un análisis equivalente al ANOVA:

```
summary(lm(y ~ trat))
```

Los modelos lineales nos permiten además combinar variables de diferente naturaleza. Por ejemplo, un ANCOVA clásico:

```
lm(y ~ trat + cov) -> modelo1
summary(modelo1)
points(cov, modelo1$fit, pch=19, col=trat)
abline(33.5, 2.5)
abline(33.5-19, 2.5, col="red")
```

Interpretemos las interacciones:

```
lm(y ~ trat * cov) -> modelo2
summary(modelo2)
points(cov, modelo2$fit, pch=19, col=trat)
abline(41, 0, col="green")
abline(41 - 34, 5, col="gold")
```

También podemos utilizar la función `aov()` para hacer un ANCOVA. Sin embargo, los estadísticos que proporciona `aov()` en este caso no son los apropiados. Hay que utilizarla junto a otra función adicional `drop1()`:

```
drop1(aov(y ~ trat * cov), ~., test="F")
```

Una función muy útil es `anova()` [no confundir con `aov()`], que permite comparar dos modelos:

```
anova(modelo1, modelo2)
```

ANOVA de dos factores

Analizaremos un ejemplo con datos procedentes de un diseño de bloques aleatorios. Se trata de la respuesta de crecimiento de la hierba algodonera (*Eriophorum angustifolium*) a cuatro tratamientos de fertilización en cinco localidades (**bloques**) en la tundra de Alaska. [Fuente: Krebs (1999), pág. 358.]

```
cottongrass
attach(cottongrass)
aov(growth ~ trat + loc) -> modelo
summary(modelo)
```

Las comparaciones múltiples las podemos hacer con el test de Tukey:

```
TukeyHSD(modelo)
```

Modelos lineales generalizados

Utilizaremos una matriz de datos simulados sobre la ocupación y reproducción de una rapaz territorial. [Fuente: OCW Ecología Metodológica y Cuantitativa: http://ocw.um.es/ciencias/ecologia-metodologica-y-cuantitativa]

```
territorios
attach(territorios)
```

Regresión discreta o de Poisson:

```
glm(pollos ~ alt, family=poisson) -> modelo
summary(modelo)
plot(alt, pollos)
points(alt[is.na(pollos)!=TRUE], modelo$fit, pch=19, col="red")
```

Comprueba cómo se obtiene la curva ajustada utilizando la ecuación del modelo de Poisson:

$$\ln(\mathrm{E}y) = b_0 + b_1 x \quad \rightarrow \quad \mathrm{E}y = e^{b_0 + b_1 x}$$

```
exp(0.8613412 -0.0008336 * (700:1100)) -> Ey
lines(700:1100, Ey, col="red")
```

Regresión logística con datos binarios (1/0):

```
glm(ocup ~ alt, family=binomial) -> modelo
summary(modelo)
plot(alt, ocup)
points(alt, modelo$fit, pch=19, col="red")
```

Comprueba cómo se obtiene la curva ajustada utilizando la ecuación del modelo logístico:

$$\ln\left(\frac{p}{1-p}\right) = b_0 + b_1 x \quad \rightarrow \quad p = \frac{e^{b_0 + b_1 x}}{1 + e^{b_0 + b_1 x}}$$

```
exp(-8.869028 + 0.010527 * (700:1100)) -> Ey
lines(700:1100, Ey/(1 + Ey))
```

Interpretación de modelos con variables cualitativas:

```
glm(pollos ~ orient, family=poisson) -> modelo
summary(modelo)
```

Utilizaremos también la función `anova()` para testar cada variable/factor:

```
anova(modelo, test="Chisq")
```

Las diferencias entre los grupos de la variable cualitativa pueden analizarse modificando la modalidad de referencia:

```
glm(pollos ~ I(orient=="Sur") + I(orient=="Este"), family=poisson) -> modelo
summary(modelo)
```

Análisis "completo" de la ocupación y la reproducción:

```
glm(ocup ~ alt * orient, family=binomial) -> modelo
anova(modelo, test="Chisq")
glm(pollos ~ alt * orient, family=poisson) -> modelo
anova(modelo, test="Chisq")
```

Modelo "clásico" de regresión logística

Usaremos los datos de un experimento de germinación de semillas de *Halocnemum strobilaceum*. Se trata de réplicas de 20 semillas sometidas a diferentes tratamientos de presión osmótica. [Fuente: OCW Ecología Metodológica y Cuantitativa: http://ocw.um.es/ciencias/ecologia-metodologica-y-cuantitativa]

```
halocnemum
attach(halocnemum)
glm(cbind(germ, nogerm) ~ pres, family=binomial) -> modelo
summary(modelo)
plot(pres, modelo$fit)
```

Sobredispersión: mayor varianza de lo esperado [(desvianza residual / grados de libertad) > 1]. Uso de la familia *quasibinomial*: cambian los estadísticos, no los coeficientes. [Para regresiones discretas existe también la familia *quasipoisson*.]

```
glm(cbind(germ, nogerm) ~ pres, family=quasibinomial) -> modelo
summary(modelo)
```

Más sobre regresión

Diagnósticos: función `influence.measures()`. [Por ejemplo `influence.measures(modelo)`.]

Regresión no lineal: función `nls()`.

Ajustaremos la ecuación de Michaelis-Menten a los datos del objeto `puromycin`, correspondiente a un estudio sobre la velocidad de una reacción enzimática en células tratadas con puromicina (fuente: librería de `R` "nlstools"):

```
puromycin
attach(puromycin)
plot(conc, rate)

nls(rate ~ Vmax * conc / (Km + conc), start=c(Vmax=200, Km=0.05)) -> modelo
summary(modelo)
```

Modelos mixtos (`lmm` y `glmm`): medidas repetidas

Usaremos otro conjunto de datos simulados (objeto `ejemplo2`). Necesitaremos las librerías `nlme` y `lme4`.

```
library(nlme)
ejemplo2
attach(ejemplo2)
plot(groupedData(y ~ cov | ind))
x11()

lme(y ~ cov, random = ~1 | ind) -> modelo1
anova(modelo1)
modelo1$fit
plot(cov, modelo1$fit[,2], col=ind)
points(cov, modelo1$fit[,1], pch=19, col="grey50")
```

Modelo con correlación entre constante y pendiente:

```
lme(y ~ cov, random = ~1 + cov | ind) -> modelo2
anova(modelo2)
plot(cov, modelo2$fit[,2], col=ind)
points(cov, modelo2$fit[,1], pch=19, col="grey50")
```

Con la función `anova()` podemos comprobar que el segundo modelo mejora considerablemente el ajuste:

```
anova(modelo1, modelo2)
```

Para el caso de modelos lineales mixtos generalizados:

```
library(lme4)
territorios
attach(territorios)
glmer(ocup ~ alt * orient + (1 | terr), family=binomial) -> modelo
summary(modelo)
```

En este caso la varianza entre territorios es muy baja, por lo que el modelo mixto apenas difiere del "clásico":

```
glm(ocup ~ alt * orient, family=binomial) -> modelo
summary(modelo)
```

En el caso de modelos obtenidos con la función `glmer()`, también puede usarse la función `anova()` para comparar dos o más modelos entre sí, pero aplicada sobre un único modelo no proporciona valores de *P* de las variables. En su lugar, puede usarse la función `Anova()` de la librería `car`:

```
library(car)
Anova(modelo)
```

Modelos anidados

Utilizaremos los datos del objeto `glycogen`, consistentes en medidas de glucógeno en plasma de ratas sometidas a diferentes tratamientos: dos medidas independientes en 3 preparaciones por rata (6 ratas) y 3 tratamientos. [Fuente: Sokal y Rohlf (1995), pág. 289). Es un ANOVA anidado (Modelo I: tratamiento fijo y niveles subordinados aleatorios). Es importante señalar que las variables deben considerarse como factores, y por tanto es necesario usar `factor()`.

```
glycogen
attach(glycogen)
aov(gly ~ factor(trat) + Error(factor(rat)/factor(prep))) -> modelo
summary(modelo)
```

El único test de interés es el del tratamiento. No obstante, si queremos otros tests, podemos calcular las *F* dividiendo la varianza (`Mean Sq`) de cada factor por la de su subordinado.

Ratas dentro de tratamientos:

```
265.9 / 49.5
1 - pf(265.9 / 49.5, 3, 12)
```

Preparaciones dentro de ratas:

```
49.5 / 21.17
1 - pf(49.5 / 21.17, 12, 18)
```

Con efectos aleatorios interesa conocer los **componentes de varianza**. Se pueden calcular con los valores del ANOVA anterior, pero es más sencillo utilizar un modelo de regresión mixto:

```
lme(gly ~ factor(trat), random = ~1 | rat/prep) -> modelo
anova(modelo)
VarCorr(modelo)
```

Para expresar los componentes de varianza como porcentajes sobre el total de varianza:

```
c(36.06482, 14.16667, 21.16667)/sum(36.06482, 14.16667, 21.16667)
```

Análisis del ejemplo del *cottongrass* (bloques) con un modelo mixto:

```
attach(cottongrass)
lme(growth ~ trat, random=~1 | loc ) -> modelo
anova(modelo)
```

Comprobemos que, para el factor "tratamiento", proporciona el mismo resultado que el ANOVA realizado anteriormente:

```
summary(aov(growth ~ trat + loc))
```

Análisis split-plot

Analizaremos las cosechas de tres variedades de avena (expresadas como 1/4 lb por *sub-plot*, cada uno de 1/80 acre). Hay seis bloques (I-VI) y cuatro tratamientos de fertilización con nitrógeno. El diseño *split-plot* es un caso especial de diseño anidado: corresponde a una estructura de *plots* (tres parcelas de cultivo, una con cada variedad) anidados en bloques (distintas fincas), y *sub-plots* (los cuatro tratamientos de fertilización) anidados en *plots*. [Fuente: Venables y Ripley (2002), pág. 282.]

```
avena
attach(avena)
lme(yield ~ variety * factor(nitro), random= ~1 | block/variety) -> modelo
anova(modelo)
```

El mismo análisis con la función `aov()`:

```
summary(aov(yield ~ variety * factor(nitro) + Error(block/variety)))
```

Análisis de datos pareados

Utilizaremos los datos contenidos en el objeto `sewage`. Son datos de un estudio en el que se analiza la densidad (transformada logarítmicamente) de coliformes por ml en aguas sometidas a dos métodos de cloración a lo largo de 8 días de tratamiento. [Fuente: librería de `R` "PairedData"].

```
sewage
attach(sewage)
```

El método clásico de análisis en estos casos es un test de la t (especificando la opción paired=T), pero también pueden analizarse con un modelo mixto:

```
t.test(coliform ~ method, paired=T)
lme(coliform ~ method, random= ~1 | day) -> modelo
anova(modelo)
```

Tablas de contingencia, chi-cuadrado y análisis log-lineal

Usaremos los datos de ocupación territorial anteriormente analizados, pero presentados en esta ocasión de manera distinta en el objeto `ocupa`. Analizaremos los datos utilizando varios métodos alternativos:

```
ocupa
table(ocupa) -> ocupat
ocupat
as.data.frame.table(ocupat) -> ocupaf
```

```
ocupaf
```

Análisis de χ^2 clásico (es el único que proporciona un valor de *P* ligeramente diferente):

```
chisq.test(ocupat)
```

Regresión logística con la tabla de datos original:

```
attach(ocupa)
glm(ocup ~ orient, family=binomial) -> modelo1
anova(modelo1, test="Chisq")
```

Regresión logística con la tabla modificada (estructura "éxitos/fracasos"):

```
glm(ocupat ~ c("E","N","S"), family=binomial) -> modelo2
anova(modelo2, test="Chisq")
```

Regresión logística con la tabla de frecuencias usando la opción de ponderación (`weights=Freq`):

```
attach(ocupaf)
glm(ocup ~ orient, weights=Freq, family=binomial) -> modelo3
anova(modelo3, test="Chisq")
```

Modelo **log-lineal**. Se utiliza una regresión de Poisson con la tabla de frecuencias, donde `Freq` es la variable dependiente para obtener el denominado "modelo saturado":

```
glm(Freq ~ ocup * orient, family=poisson) -> modelo4
anova(modelo4, test="Chisq")
```

Cálculo de probabilidades de cada celda. El modelo saturado (`modelo4`) predice exactamente la frecuencia de cada celda de la tabla `ocupat`:

```
summary(modelo4)
predict(modelo4, type="response")
```

Utilizando los coeficientes del `modelo4` podemos calcularlos. Por ejemplo, Este y vacío:

```
exp(2.4849 -0.4055)
```

Sur y vacío:

```
exp(2.4849 -0.4055 + 0.2877  -0.5754)
```

Otra alternativa es utilizar un **modelo multinomial**. Los modelos multinomiales son una generalización de los modelos binomiales que se utilizan cuando la variable de respuesta tiene más de dos modalidades. Necesitaremos la librería `nnet`.

```
library(nnet)
multinom(ocup ~ orient, weights=Freq) -> modelo5
summary(modelo5)
```

La función `multinom()` no proporciona valores de *P*, por lo que necesitaremos comparar el modelo de interés con el modelo nulo, o bien usar directamente la función `Anova()` de la librería "car":

```
multinom(ocup ~ 1, weights=Freq) -> modelo0
anova(modelo0, modelo5)
Anova(modelo5)
```

Tablas de contingencia con más de un factor

Usaremos los datos del objeto `polymorphism` que contiene las frecuencias de alelos del polimorfismo PPARg Pro12Ala en poblaciones humanas de Polonia y Estados Unidos. Los "casos" son enfermos de diabetes. [Fuente: Ardlie et al. (2002).]

```
polymorphism
attach(polymorphism)
xtabs(freq ~ diabetes + allele + population)

glm(freq ~ diabetes * allele * population, family=poisson) -> modelo
anova(modelo, test="Chisq")
```

Solo interesan los tests correspondientes a los términos de interacción con la variable dependiente (`diabetes`).

Selección de modelos e inferencia multimodelo

En un procedimiento de selección de modelos utilizaremos la verosimilitud y el AIC:

```
logLik(modelo)
AIC(modelo)
```

Necesitaremos la librería `AICcmodavg`. Usaremos datos de concentraciones de calcio (mg/100 ml) en plasma sanguíneo de aves de ambos sexos sometidas a tres tratamientos hormonales. [Fuente: Zar (2010), pág. 251.]

```
library(AICcmodavg)
birds
attach(birds)
```

Utilizaremos cinco modelos distintos, incluido el modelo nulo:

```
lm(Ca ~ 1) -> m0
lm(Ca ~ hormone) -> m1
lm(Ca ~ sex) -> m2
lm(Ca ~ hormone + sex) -> m3
lm(Ca ~ hormone * sex) -> m4
```

Crearemos la lista de "candidatos", asignando un nombre a cada uno, y obtendremos la tabla:

```
candidatos <- list(m0, m1, m2, m3, m4)
nombres <- c("nulo", "hormone", "sex", "hormone+sex", "hormone*sex")
aictab(candidatos, nombres)
```

Con la función de inferencia multimodelo estimaremos, por ejemplo, el coeficiente del tratamiento T3:

```
modavg(candidatos, "hormoneT3", nombres, exclude=list(""))
```

Referencias

Ardlie KG, Lunetta KL, Seielstad M. 2002. Testing for population subdivision and association in four case-control studies. *American Journal of Human Genetics*, 71: 304-311.

Krebs CJ. 1999. *Ecological Methodology*. 2ª ed. Benjamin/Cummings, Menlo Park, CA.

Sokal RR, Rohlf FJ. 1995. *Biometry*, 3ª ed. Freeman, New York.

Venables WN, Ripley BD. 2002. Modern Applied Statistics with S. 4ª ed. Springer, New York.

Zar JH. 2010. *Biostatistical Analysis*. 5ª ed. Prentice Hall, New Jersey.