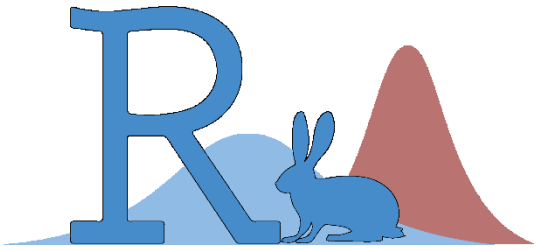
RabbitR: Un programa para resolver modelos lineales usando inferencia Bayesiana



Desarrollado por:
Marina Martínez-Álvaro
mamaral9@upv.es
Cristina Casto-Rebollo
3ccasto@gmail.com

Diseño:
Agustín Blasco
ablasco@dca.upv.es

INTRODUCCIÓN AL PAQUETE RABBITR

Rabbit fue programado originalmente en FORTRAN por Wagdy Mekkawy, Dianelys González-Peña y Agustin Blasco. El programa tuvo un éxito limitado porque sólo admitía un efecto aleatorio, no incluía interacciones y sobre todo porque en FORTAN los nombres y las salidas numéricas tenían que definirse de forma muy poco flexible. Con la versión en R preparada por Marina Martínez-Álvaro y Cristina Casto-Rebollo el programa adquiere la flexibilidad que necesitaba, y la comodidad de uso que era imprescindible para su divulgación. **RabbitR** puede ir mejorando a partir de las sugerencias que nos hagan los usuarios, lo consideramos un "work in progress" y esperamos que ayude a la generalización de los métodos bayesianos en los análisis simples cotidianos.

OBJETIVO DEL PAQUETE RABBITR

RabbitR resuelve modelos lineales sencillos para la comparación de tratamientos como los que pueden resolver los programas GLM o MIXED del SAS. Un ejemplo de modelo que puede resolver **RabbitR** es

$$y = m + R + T + R * T + b_1x_1 + b_2x_2 + c + p + e$$

donde y son los datos a analizar, R y T son efectos fijos (ruido y tratamiento), $R * T$ son las interacciones, x_1 y x_2 son covariables, c y p son efectos aleatorios, y e es el residuo.

RabbitR admite cualquier número de efectos fijos, covariables y efectos aleatorios, pero solo admite interacciones dobles entre efectos fijos.

RabbitR utiliza *inferencia Bayesiana* y ofrece sus resultados siguiendo las propuestas de Blasco (2001, 2005, 2017, 2021) para la inferencia Bayesiana en estos modelos. Estas propuestas son una aproximación novedosa basada en la posibilidad que tiene la inferencia Bayesiana de usar intervalos de confianza definidos por el investigador. Con estos intervalos se puede proponer no sólo que la probabilidad de que la diferencia entre tratamientos sea mayor (o menor) que cero, sino la probabilidad de que esta diferencia sea *relevante*, o bien determinar la probabilidad de que esta diferencia sea irrelevante; esto es, cero a efectos prácticos. Se puede obtener, para una diferencia entre tratamientos, un *valor mínimo garantizado* con una probabilidad determinada. También se pueden usar *cocientes entre tratamientos* en lugar de diferencias, puesto que en ocasiones puede ser más expresivo comparar los tratamientos de forma relativa.

Esta forma de analizar los datos ha ido implantándose, y cada vez es mayor el número de publicaciones en las que se utiliza alguna de las propuestas; el éxito se debe a que son una ayuda considerable para la discusión de los resultados. Una explicación detallada de los procedimientos se puede encontrar en Blasco (2017, cap. 2; 2021, cap. 5).

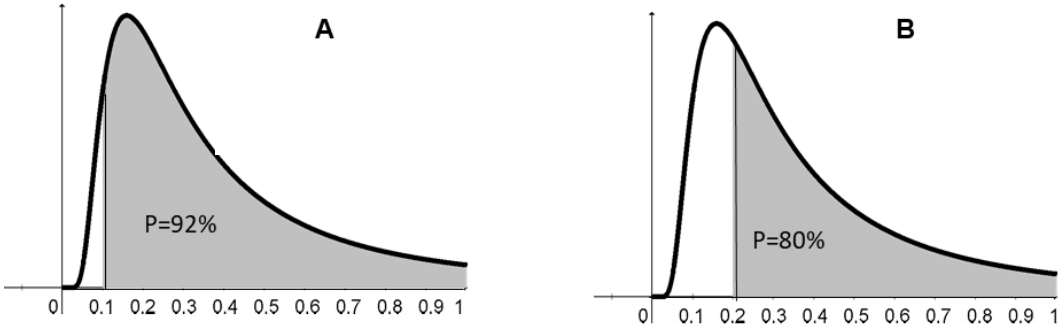
La inferencia Bayesiana se ha usado generalmente para resolver problemas fuera del alcance de la estadística clásica; por ejemplo modelos anidados (modelos dentro de modelos, como la estima de efectos genéticos en curvas de crecimiento o de lactación), modelos donde el número de efectos desconocidos es muy superior al número de datos (genómica, y otras −ómicas), determinación genética y ambiental de la varianza residual para la transformación de problemas multivariantes en univariantes mediante la *marginalización*, etc. (ver Blasco 2017, cap. 8). Nosotros creemos que la inferencia Bayesiana puede ser muy útil también en problemas sencillos, habituales en la investigación, como la comparación de tratamientos y la corrección por efectos de ruido o por covariables.

LA INFERENCIA BAYESIANA

La inferencia Bayesiana se basa en estimar las probabilidades de lo que se desea estimar en función de los datos de que se dispone. Por ejemplo, si tenemos dos piensos y deseamos evaluarlos con arreglo a su índice de conversión (IC), la pregunta no es si difieren o no (nunca van a ser exactamente iguales hasta la última cifra decimal) sino si difieren por encima de un **valor relevante.**

El **valor relevante** es la cantidad por debajo de la cual dos tratamientos *son iguales a efectos prácticos*; o bien, la cantidad por encima de la cual se toma una decisión (sustituir un pienso por otro, por ejemplo). Cuando se diseña un experimento, **el valor relevante** es la cantidad que se usa para definir a partir de cuándo las diferencias entre tratamientos serán significativas, o para definir la longitud deseada de los intervalos de confianza. En la estadística clásica el **valor relevante** sólo se usa para diseñar un experimento para un carácter, por lo que frecuentemente aparecen en otros caracteres significaciones irrelevantes y diferencias n.s. muy relevantes; pero en la estadística Bayesiana podemos calcular las probabilidades de relevancia de cualquier carácter. Si no hay argumentos económicos o biológicos para definir el valor relevante, puede usarse una fracción (p. ej., 1/2 ó 1/3) de la desviación típica del carácter (Blasco 2017, Appendix 1.1; 2021, 5.1.3).

Para calcular las probabilidades, en la inferencia Bayesiana se usa una función auxiliar, la ***función de densidad posterior*** de lo que queremos estimar, dados los datos de nuestro experimento. En estas funciones, *el área entre dos puntos representa la probabilidad de que el parámetro que estimamos esté entre esos dos puntos*. Téngase presente que la probabilidad de un punto es siempre cero, debido a que hay infinitos puntos; es decir, la probabilidad de que la diferencia en índice de conversión entre dos piensos sea exactamente 0.1 (es decir, 0.1000000…) es 1/∞; es decir, cero. Pero con las funciones de densidad posterior podemos crear cualquier intervalo de confianza que pueda sernos útil y calcular el área de ese intervalo.



En la figura se representan dos intervalos de confianza de la misma función de densidad posterior para la diferencia entre IC de dos piensos según los datos **y** del experimento. En el caso **A** el intervalo va desde 0.1 hasta infinito; es decir, muestra la probabilidad de que la diferencia en IC entre los dos piensos sea mayor de 0.1. Si este es el **valor relevante r**, la probabilidad de que la diferencia entre piensos sea relevante para IC es del 92%, lo que puede ser suficiente para tomar una decisión de sustituir uno por el otro. Obsérvese que *no hablamos de significación ni de P-valor*, esta es la *probabilidad de la diferencia entre IC sea relevante*, lo que llamamos **probabilidad de relevancia Pr**. Se puede calcular también la probabilidad del intervalo [-r, +r]; es decir, la probabilidad de que la diferencia sea irrelevante o **probabilidad de similitud Ps**. En el caso **B**, deseamos saber con una probabilidad del 80% cuál es *como mínimo* la diferencia entre IC; es decir, cuál es el **valor garantizado k** de la diferencia entre IC con una probabilidad determinada (en el caso de la figura B podemos garantizar que la diferencia en IC es al menos 0.2 con un 80% de probabilidad).

Referencias

Blasco, A. 2001. The Bayesian controversy in animal breeding. J. Anim. Sci. 79:2023–2046.
Blasco, A. 2005. The use of Bayesian statistics in meat quality analyses: A review. Meat Sci. 69:115–122. doi:10.1016/j.meatsci.2004.06.012.
Blasco, A. 2017. A Scope of the Possibilities of Bayesian Inference + MCMC. Bayesian Data Anal. Anim. Sci. 167–192. doi:10.1007/978-3-319-54274-4_8.
Blasco, A. 2021. Mejora Genética Animal. Síntesis.

ANTES DE EMPEZAR

El paquete **RabbitR** está programado en lenguaje R, lo que implica que el usuario debe instalar en su ordenador el lenguaje **R**, y es conveniente instalar un editor (**RStudio**).

Instalación de R en Windows versión 4.3.3

Ir a la página https://cran.rstudio.com/bin/windows/base/old/

- Descargar la versión R 4.3.3 pulsando sobre R 4.3.3
- Ejecutar **R-4.3.3-win** y seguir las instrucciones con los parámetros por defecto.

Es importante instalar una versión R 4.3.2 o superior, porque con otras versiones pueden aparecer problemas si no eres un usuario de R experimentado. En la dirección https://github.com/marinamartinezalvaro/RabbitR se irán colgando las actualizaciones del paquete conforme vayan apareciendo nuevas necesidades.

Instalación de RStudio versión 2023.12.1+402

Ir a la página https://posit.co/download/rstudio-desktop

- RECUADRO AZUL: Download RStudio Desktop for Windows
- Ejecutar **RStudio** y seguir las instrucciones con los parámetros por defecto.

Conviene usar la version 2023.12.1+402 por la misma razón que instalamos R versión >4.3.2

Ejecutar Rabbit por primera vez

Cuando se instale el paquete **RabbitR** por primera vez debes estar conectado a internet. Se descargarán una serie de paquetes de las que **RabbitR** depende. Una vez estos paquetes queden instalados, no habrá que repetir el proceso y cuando se vuelva a ejecutar **RabbitR** el proceso será más rápido.

- Crear un directorio de trabajo (con el Explorador de archivos de Windows), y trasladar allí el **fichero de datos**.
- Seleccionar el directorio de trabajo: Se usa la línea de menú superior al abrir Rstudio
  Abrir **Rstudio**
  Menú Session
  Set working directory
  Choose directory
- Instalar el paquete RabbitR por primera vez. En la ventana Console escribir

```
> install.packages("devtools")
> devtools::install_github("marinamartinezalvaro/RabbitR")
> library(RabbitR)
```

Atención: **R *distingue entre mayúsculas y minúsculas.***

Ya se puede trabajar con **RabbitR**!!

PRÁCTICA 1: LA FUNCIÓN RABBIT

La función Rabbit del paquete RabbitR sirve como función interactiva todo en uno para ejecutar modelos lineales generales, generar distribuciones posteriores y calcular inferencias.

EL FICHERO DE DATOS

Fichero de texto o Excel con alguna de estas extensiones: **.txt .xls .xlsx .csv** (separado por "," o ";"). Al poner el nombre, recordar que **R *distingue entre mayúsculas y minúsculas.* El nombre no puede tener espacios en blanco**, acentos, *ñ* o caracteres especiales.

CONTENIDO:

En columnas los *Efectos fijos, efectos aleatorios, Covariables* y *Caracteres*, no importa en qué orden. El archivo de datos puede contener otras columnas adicionales que no se vayan a usar en al análisis (p.ej. otros caracteres, el identificador de los animales, etc.).

Puede haber *valores ausentes* en los caracteres o las covariables. Un valor ausente puede ser un *espacio en blanco*, o una palabra (p.ej. NA), o un número que se pueda identificar como distinto a cualquier otro dato del fichero de datos (p. ej., 99999). En la función Rabbit(), el valor ausente *debe ser el mismo en toda la base de datos*.

La separación decimal debe ser el punto. ***No se admiten comas.***

La primera fila del archivo de datos es para los nombres de los efectos y los caracteres a analizar. Es recomendable que los nombres no tengan espacios en blanco, acentos o caracteres o símbolos que no sean del alfabeto inglés.

Los niveles de los efectos fijos pueden ser números o caracteres. Los niveles de los efectos aleatorios también, aunque conviene que sean numéricos porque suelen tener muchos niveles.

EJEMPLO

Vamos a analizar la grasa perirrenal (PFat), la grasa intramuscular (IMF) y el pH del lomo de conejos pertenecientes a varias generaciones de selección por grasa intramuscular (LG), y a ambos sexos (Sex), a peso vivo (LW) constante. Los datos han sido tomados en diferentes estaciones (E) y orden de parto (OP) diferentes, y tenemos un efecto aleatorio de camada (c). El tratamiento (en lo que estamos interesados) es Sex (machos M y hembras F) y el ruido E (Verano V, Primavera P, Invierno I) y OP (1, 2). La covariable es LW y el efecto aleatorio c. En este ejemplo, obviaremos LG.

Se aplica el mismo modelo a todos los caracteres

$IMF = m + Sex + E + OP + b \cdot LW + c + e$

$PFat = m + Sex + E + OP + b \cdot LW + c + e$

El fichero de Datos es DataIMF.csv y lo podéis encontrar en Teams

AE	OP	LG	Sex	c	LW	pH	IMF	PFat
1	1	1	1	1	1790	5.64	1.089	9.2
1	1	1	1	2	1795	5.45	1.118	12.4
1	1	1	1	3	1505	99999	0.939	3.1
1	1	1	1	4	1870	5.64	1.229	13.82
1	1	1	1	5	1725	5.51	1.068	7.01
1	1	1	1	6	1570	5.44	1.439	8.2
1	1	1	1	7	1780	5.47	1.359	9.7
1	1	1	1	8	1675	5.45	1.066	7.81
1	1	1	1	9	1870	5.53	1.261	16.51

LA FUNCIÓN Rabbit

PUESTA EN MARCHA

1. Entrar en Rstudio.

2. Seleccionar el directorio de trabajo, como al entrar por primera vez

3. Poner el fichero de datos en el directorio de trabajo

4. Ejecutar **Rabbit**. Como ya instalamos antes el paquete RabbitR y hemos cargado el paquete usando library(), basta hacer:

```
>RESULTADOS<-Rabbit()
```

Puede usarse "RESULTADOS" o cualquier otro nombre. *Recordar que R distingue entre mayúsculas y minúsculas*.

Por motivos biológicos, económicos o por ser 1/3 de la desviación típica del carácter, supondremos que un **valor relevante** para una diferencia en grasa intramuscular se produce a partir de 0.07, y para grasa perirrenal a partir de 1 g. Por lo tanto, *nuestro valor relevante para la diferencia entre tratamientos en IMF es* 0.07 y en *PFat* es 1. En el caso de que hiciéramos cocientes entre tratamientos, el valor relevante podría ser, por ejemplo, un 5% para ambos caracteres; esto es, sería relevante que un tratamiento fuera un 5% superior a otro, y como valor relevante usaríamos 1.05. En este ejemplo no tendremos en cuenta las interacciones.

Hay que ir respondiendo a las preguntas de la función *Rabbit*:

EJEMPLO

```
Enter the name of the datafile with its extension .txt, .csv., .xls or .xlsx
        DataIMF.csv
Has the data missing values (Enter Yes=Y or No=N)  ?
        Y
Please, enter the missing value. If it is a blank, enter a space
        99999
The number of rows in the data file is 502
Enter the total number of traits
        2
Enter the name of the trait 1
        IMF
Enter the name of the trait 2
        PFat
Table: Summary Statistics of Traits

             Mean          SD     Min   1st Qu..25%   Median   3rd Qu..75%     Max         CV   Missing Values
-----  ----------  ----------  ------  ------------  -------  ------------  ------  ---------  ---------------
IMF       1.180445   0.1640249   0.796         1.061   1.1655       1.28975    1.72   13.89517                8
PFat     10.620199   4.5220638   2.510         7.480   9.8000      12.91500   44.06   42.57984                0
Let's define the model. Remember, all traits will be analyzed with the same model!
Enter the number of treatments
        1
```

Enter the name of Treatment 1

Sex

The number of levels read in Treatment Sex are: 2

Comparisons between treatment levels: Enter DIFFERENCE=D or RATIO=R

D

Enter the number of noise effects

2

Enter the name of Noise effect 1

AE

The number of levels read in Noise AE are: 2

Enter the name of Noise effect 2

OP

The number of levels read in Noise OP are: 3

```
Contingency Tables across effects
[1] "Sex vs AE"

       AE1 AE2
  Sex1 192  56
  Sex2 208  46

Contingency Tables across effects
[1] "Sex vs OP"

       OP1 OP2 OP3
  Sex1 219  27   2
  Sex2 223  29   2

Contingency Tables across effects
[1] "AE vs OP"

      OP1 OP2 OP3
  AE1 348  48   4
  AE2  94   8   0
```

Enter the number of covariates (0,1,2,…)

1

Enter the name of Covariate 1

LW

See below the summary statistics of covariates: LW

Table: Summary Statistics of Covariates

	Mean	SD	Min	1st Qu..25%	Median	3rd Qu..75%	Max	CV	Missing Values
LW	1757.058	177.1815	1380	1630	1745	1880	2595	10.08399	0

Do you want to consider any interactions of order 2 (Enter Yes=Y or No=N) ?

N (La versión 1.0 de RabbitR no incluye interacciones)

Enter the number of random effects

1

Enter the name of the random effect 1

c

Model equation for all Traits is : y = mean + Sex + AE + OP + b* LW + Random(c)

Do you want to establish the MCMC characteristics (Enter Yes=Y or No=N)?

N (Las características MCMC por defecto son suficientes)

Your parameter file its ready!

Bunny Starting ...

```
Analysis for Trait IMF in progress
-----------------------------------------------------
Features of posterior samples:
-----------------------------------------------------
Number of iterations = 30000
Burn-in = 5000
Lag = 10
Number of samples stored = 2500
-----------------------------------------------------
Model Evaluation Criterion:
-----------------------------------------------------
DIC (Deviance Information Criterion) = -432.9782
-----------------------------------------------------
Geweke's Convergence Diagnostics Z-Scores Summary:
-----------------------------------------------------
Effects Z-Scores:
  (Intercept): -0.561166539739003
  Sex2: -0.30050448897673
  AE2: -0.350541294668947
  OP2: -0.124256419093273
  OP3: 1.68771800985571
  LW: 0.745928264477138
VarianceComponents Z-Scores:
  c: 2.27617910150364 (Potential issue with convergence)
  Ve: -1.46806176493024

Interpretation of Geweke's Z-Scores:
Z-scores within the range of -2 to 2 generally indicate that the chain has converged to the target distributi
on.Z-scores outside this range may suggest issues with convergence, warranting further investigation.
-----------------------------------------------------
Computing Means...
 Means computed!
-----------------------------------------------------
Computing Contrasts between levels of Treatment effects...
-----------------------------------------------------
Contrasts computed!  Covariates computed!  Variances of Random Effects computed!

Analysis for Trait PFat in progress
-----------------------------------------------------
Features of posterior samples:
-----------------------------------------------------
Number of iterations = 30000
Burn-in = 5000
Lag = 10
Number of samples stored = 2500
-----------------------------------------------------
Model Evaluation Criterion:
-----------------------------------------------------
DIC (Deviance Information Criterion) = 2474.962
-----------------------------------------------------
Geweke's Convergence Diagnostics Z-Scores Summary:
-----------------------------------------------------
Effects Z-Scores:
  (Intercept): 0.56295387862287
  Sex2: 0.429486454068053
  AE2: -0.241830592772411
  OP2: -1.51989438189441
  OP3: -0.419190627509812
  LW: -0.536467927600128
VarianceComponents Z-Scores:
  c: -1.077895798628
  Ve: 1.26867284064471

Interpretation of Geweke's Z-Scores:
Z-scores within the range of -2 to 2 generally indicate that the chain has converged to the target distributi
on.
Z-scores outside this range may suggest issues with convergence, warranting further investigation.
-----------------------------------------------------
Computing Means...

 Means computed!
-----------------------------------------------------

Computing Contrasts between levels of Treatment effects...
-----------------------------------------------------
 Contrasts computed!  Covariates computed!  Variances of Random Effects computed!
```

iBayes Starting ...

Enter the probability for the HPD interval, for example 0.95

0.95

Do you want to calculate a guaranteed value (K) with a given probability? (Enter Yes=Y or No=N)

Y

Enter the probability for the guaranteed value K, for example 0.80

0.80

Do you want to calculate probability (PR) of contrasts being greater than a relevant value (R)? (Enter Yes=Y or No=N)

Y

Do you want to calculate probability of similarity of contrasts (PS)? (Enter Yes=Y or No=N)

Y

Enter the R values, separated by commas if multiple traits (e.g., 0.1,0.2)

0.07,1 (En caso de ratios, pondríamos 1.05 si R=5%)

Do you want to save the inferences into a table? (Enter Yes=Y or No=N):

Y

Do you want to generate and save plots of contrasts? (Enter Yes=Y or No=N):

Y

```
Processing Trait: IMF
--------------------------------------------------
 Model Mean
Median:  1.181934
Mean:  1.181953
SD:  0.007018359
HPD Lower:  1.168617
HPD Upper:  1.195995
--------------------------------------------------
--------------------------------------------------
 Residual Variance
Median:  0.02388016
Mean:  0.02394869
SD:  0.001625195
HPD Lower:  0.02069276
HPD Upper:  0.02712196
--------------------------------------------------
Inferences of posterior chains for treatMeans
--------------------------------------------------
Sex 1
Median:  1.166706
Mean:  1.165928
SD:  0.02839652
HPD Lower:  1.112862
HPD Upper:  1.219871
P0:  1
Guaranteed Value with prob 0.8:  1.140984
--------------------------------------------------
Sex 2
Median:  1.153149
Mean:  1.152455
SD:  0.02846924
HPD Lower:  1.098314
HPD Upper:  1.209316
P0:  1
Guaranteed Value with prob 0.8:  1.128632
--------------------------------------------------
Inferences of posterior chains for Compare
--------------------------------------------------
Sex 1-2
Median:  0.0136849
Mean:  0.01347306
SD:  0.01402777
HPD Lower:  -0.01532127
HPD Upper:  0.03937308
P0:  0.8304
Guaranteed Value with prob 0.8:  0.001608101
PR with R 0.07:  0
PS with R 0.07:  1
--------------------------------------------------
Inferences of posterior chains for Cov
--------------------------------------------------
Cov LW
Median:  0.00027447
Mean:  0.0002752855
SD:  4.578658e-05
HPD Lower:  0.0001899744
HPD Upper:  0.0003662265
P0:  1
Guaranteed Value with prob 0.8:  0.0002376927
```

```
--------------------------------------------------
Inferences of posterior chains for RandomVariances
--------------------------------------------------
RandomVariances c
Median:  3.132024e-07
Mean:  0.0001499061
SD:  0.0004423472
HPD Lower:  1.42194e-12
HPD Upper:  0.001045848
--------------------------------------------------


Processing Trait: PFat
--------------------------------------------------
 Model Mean
Median:  10.62626
Mean:  10.62411
SD:  0.1370603
HPD Lower:  10.34926
HPD Upper:  10.87476
--------------------------------------------------
--------------------------------------------------
 Residual Variance
Median:  7.075414
Mean:  7.140453
SD:  0.7012371
HPD Lower:  5.848523
HPD Upper:  8.510283
--------------------------------------------------
Inferences of posterior chains for treatMeans
--------------------------------------------------
Sex 1
Median:  9.034583
Mean:  9.033289
SD:  0.5269318
HPD Lower:  7.950903
HPD Upper:  10.0159
P0:  1
Guaranteed Value with prob 0.8:  8.594745
--------------------------------------------------
Sex 2
Median:  10.63319
Mean:  10.62284
SD:  0.5247854
HPD Lower:  9.555619
HPD Upper:  11.61131
P0:  1
Guaranteed Value with prob 0.8:  10.19254
--------------------------------------------------
Inferences of posterior chains for Compare
--------------------------------------------------
Sex 1-2
Median:  -1.592421
Mean:  -1.589552
SD:  0.2357411
HPD Lower:  -2.059659
HPD Upper:  -1.151798
P0:  1
Guaranteed Value with prob 0.8:  -1.392096
PR with R 1:  0.9932
PS with R 1:  0.0068
--------------------------------------------------
Inferences of posterior chains for Cov
--------------------------------------------------
Cov LW
Median:  0.01818749
Mean:  0.01821006
SD:  0.0008301467
HPD Lower:  0.01670523
HPD Upper:  0.01995836
P0:  1
Guaranteed Value with prob 0.8:  0.01750453
--------------------------------------------------
Inferences of posterior chains for RandomVariances
--------------------------------------------------
RandomVariances c
Median:  1.167787
Mean:  1.137753
SD:  0.5990442
HPD Lower:  0.008924544
HPD Upper:  2.094267
--------------------------------------------------
```

LOS RESULTADOS

Para cada carácter el modelo es

y = m + Sex + E + OP+ b·LW + c + e

Rabbit ofrece los resultados de los tratamientos, pero no de los efectos de ruido, dado que no son interesantes (son ruido). **Rabbit** ofrece los resultados en el fichero:

Results.xlsx

Es un fichero Excel en el que se ofrecen los resultados para todos los caracteres. Por ejemplo, para IMF los resultados se ofrecen en las siguientes líneas:

- La media **m** del modelo para el carácter (IMF_ModelMean)
- La varianza del residuo σ_e^2 (IMF_Resvariance)
- Los **efectos** de cada tratamiento sumándoles la media **m** (IMF_treatMeans_Sex_F) (IMF_treatMeans_Sex_M).
- Las **diferencias entre los niveles** de cada tratamiento (IMF_Compare_Sex_F-M).
- Los **coeficientes de regresión de las covariables** (IMF_Cov_LW).
- La varianza del efecto aleatorio (IMF_RandomVariances_c)

Resultados de la inferencia Bayesiana

Los resultados de la inferencia Bayesiana provienen de la *función de densidad posterior*. Para cada una de las estimaciones anteriores, en el fichero **Results.xlsx** se ofrece:

- La mediana de la distribución posterior, Median
- La media de la distribución posterior, Mean
- La **desviación típica SD** de la distribución posterior. *Cuando esta distribución es simétrica*, como es el caso de las estimas de diferencias entre tratamientos, el valor verdadero está entre los límites marcados por **± 2·SD** con un 95% de probabilidad. Es frecuente ofrecer **SD** por su *similitud aparente con el* s.e., pero aquí su significado es completamente diferente; aquí indica que el valor verdadero está entre esos límites con una probabilidad del 95%, *cosa que no ocurre con el intervalo* ± 2·s.e. En realidad, la s.d. confunde y no debiera ofrecerse, pero es frecuente encontrarla en artículos de inferencia Bayesiana.
- **El intervalo de confianza de máxima densidad [HPD_lower, HPD_upper]** Nos indica los límites del intervalo más estrecho posible que encierra la densidad deseada (p. ej. 95%). El verdadero valor está entre los límites de este intervalo con la probabilidad definida por el usuario.
- En el caso de efectos, covariables y contrastes, se ofrece **la probabilidad P0** de que el efecto, covariable o el contraste (la diferencia entre tratamientos) sea mayor que cero si es positivo o menor que cero si es negativo (la probabilidad de que sea exactamente igual a cero 0.0000… es cero).
- En el caso de efectos, covariables y contrastes, si el usuario ha definido el **valor garantizado**, se ofrece este valor (**K**) *con la probabilidad definida por el usuario*. En ocasiones el valor de K no está definido y no se ofrece (por ejemplo, si la mediana el contraste es positivo y el valor de K es negativo, el valor de k no garantiza nada en ese caso).
- Para los efectos y contrastes, si el usuario ha definido el **valor relevante R**, se ofrece la **probabilidad de relevancia PR** del contraste; esto es, la probabilidad de que sea

mayor que **R** si el contraste es positivo o menor que **R** si es negativo. Si **PR** es elevada (por ejemplo, más del 80%), entonces *la diferencia entre tratamientos es relevante*.

- Para los efectos y contrastes, si el usuario lo ha demandado, se ofrece la **probabilidad de similitud PS**; esto es, la probabilidad de que el verdadero valor sea irrelevante (cero a efectos prácticos) y se encuentre entre **-R** y **– R** (1/R y R si es un ratio). Si **PS** es elevada, *la diferencia entre tratamientos es irrelevante, cero a efectos prácticos*. Si **PS** es muy baja, hay que recurrir a **PR** para ver si la diferencia entre tratamientos es relevante. Si **PS** es intermedia (por ejemplo, 40%), entonces no sabemos si la diferencia entre tratamientos es irrelevante (nula a efectos prácticos) o no, no hay suficientes datos para tomar una decisión. Es el equivalente a *no significativo* en estadística clásica, no se sabe si hay diferencia relevante entre tratamientos; n.s. = no sé, no quiere decir que no hay diferencias relevantes entre tratamientos, sino que no lo sabemos por falta de datos suficientes.

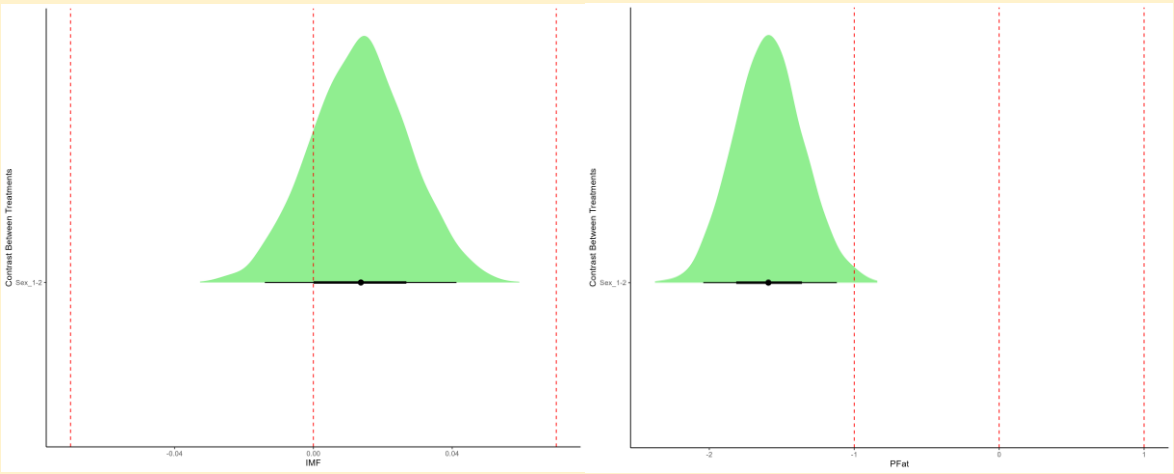
Fichero de Contrasts_posterior_plot_IMF.tiff con el nombre definido por el usuario: Gráficos de las densidades posteriores de los contrastes, con líneas en 0 y ± el valor relevante.

EJEMPLO

Results.xlsx

	Median	Mean	SD	HPD_lower	HPD_upper	P0	K	PR	PS
IMF_ModelMean	1.182	1.182	0.007	1.169	1.196				
IMF_ResVariance	0.024	0.024	0.002	0.021	0.027				
IMF_treatMeans_Sex_1	1.167	1.166	0.028	1.113	1.220	1.000	1.141		
IMF_treatMeans_Sex_2	1.153	1.152	0.028	1.098	1.209	1.000	1.129		
IMF_Compare_Sex_1-2	0.014	0.013	0.014	-0.015	0.039	0.830	0.002	0.000	1.000
IMF_Cov_LW	0.000	0.000	0.000	0.000	0.000	1.000	0.000		
IMF_RandomVariances_c	0.000	0.000	0.000	0.000	0.001		0.000		
PFat_ModelMean	10.626	10.624	0.137	10.349	10.875				
PFat_ResVariance	7.075	7.140	0.701	5.849	8.510				
PFat_treatMeans_Sex_1	9.035	9.033	0.527	7.951	10.016	1.000	8.595		
PFat_treatMeans_Sex_2	10.633	10.623	0.525	9.556	11.611	1.000	10.193		
PFat_Compare_Sex_1-2	-1.592	-1.590	0.236	-2.060	-1.152	1.000	-1.392	0.993	0.007
PFat_Cov_LW	0.018	0.018	0.001	0.017	0.020	1.000	0.018		
PFat_RandomVariances_c	1.168	1.138	0.599	0.009	2.094		0.624		

Contrasts_posterior_plot_IMF and Pfat.tiff



Ahora, prueba a usar la función Rabbit cambiando los tratamientos (por ejemplo,ahora incluye OP, y quítalo del ruido) y usando el ratio para comparar niveles. Recuerda que deberás ajustar el valor relevante de acuerdo a un ratio.

PRÁCTICA 2: HANDS ON CHAINS (BUNNY Y BAYES)

Ayer usamos la función *Rabbit* para realizar los análisis y el programa nos calculó para cada una de las estimaciones, varias inferencias de la distribución marginal posterior. Hoy, obtendremos las distribuciones posteriores y haremos nosotros mismos las inferencias en Excel. Para ello, utilizaremos las funciones *iCreateParam(), Bunny()* y *iBayes()* por separado, que son las tres funciones que componen la función *Rabbit()*. Las funciones *iCreateParam* e *iBayes* tienen su versión no interactiva que puede resultar más práctica una vez el usuario este familiarizo con el uso de R.

Utilizaremos los mismos datos de ayer (DataIMF.csv) y el siguiente modelo (esta vez, incluyendo LG como tratamiento).

$$IMF = m + Sex + LG + E + OP + b \cdot LW + c + e$$

Por simplicidad, hoy trabajaremos solamente con el carácter IMF. Empezaremos creando nuestro archivo de parámetros con la función *iCreateParam*(), contestando a las mismas preguntas de la función *Rabbit()*.

EJEMPLO

```
param_list<-iCreateParam()
```

Alternativamente, podemos usar la función *CreateParam*() con los siguientes argumentos:

```
param_list<-CreateParam(
        file.name = "DataIMF.csv",
        na.codes=c("99999"),
        hTrait = c("IMF"),
        hTreatment = c("Sex", "LG"),
        askCompare="D",
        hNoise = c("AE", "OP"),
        hCov = "LW",
        hRand = "c")
```

Donde:

file.name: Cadena entre comas " " que especifica el nombre o el nombre y la ruta del fichero de datos (si no está en el directorio de trabajo), que debe estar en formato '.csv','.xlsx' o '.txt'. Si el archivo de datos está precargado en el entorno, el nombre de los datos puede introducirse como una cadena "" sin extensión.

na.codes: Vector de caracteres que especifica cómo se codifican los valores omitidos en el archivo de datos. Se permiten diferentes valores en el mismo archivo. Los códigos por defecto son c("", "NA", "NULL").

hTrait: Vector que especifica los nombres de los rasgos. Los usuarios pueden especificar los rasgos utilizando hTrait, pTrait o ambos argumentos.

pTrait: Vector que especifica las posiciones de columna en el archivo de datos de los rasgos. Los usuarios pueden especificar los rasgos utilizando hTrait, pTrait o ambos argumentos.

hTreatment: Vector que especifica los nombres de los efectos de tratamiento. Los usuarios pueden especificar los efectos del tratamiento utilizando hTreatment, pTreatment o ambos argumentos.

pTreatment: Vector que especifica las posiciones de columna en el archivo de datos de los efectos de tratamiento. Los usuarios pueden especificar los efectos de tratamiento utilizando hTreatment, pTreatment o ambos argumentos.

askCompare: Carácter que especifica cómo comparar los niveles de tratamiento: '"D"'para diferencia, '"R"' para ratio, o '"NA"' si no procede. Por defecto es '"D"'.

hNoise: Vector que especifica los nombres de los efectos de ruido. Los usuarios pueden especificar los efectos de ruido utilizando hNoise, pNoise o ambos argumentos.

pNoise: Vector que especifica las posiciones de columna en el archivo de datos de los efectos de ruido. Los usuarios pueden especificar los efectos de ruido utilizando hNoise, pNoise o ambos argumentos.

hCov: Vector que especifica los nombres de las covariables. Los usuarios pueden especificar las covariables utilizando hCov, pCov o ambos argumentos.

pCov: Vector que especifica las posiciones de columna en el archivo de datos de las covariables. Los usuarios pueden especificar las covariables utilizando hCov, pCov o ambos argumentos.

hRand: Vector que especifica los nombres de los efectos aleatorios. Los usuarios pueden especificar los efectos aleatorios utilizando hRand, pRand o ambos argumentos.

pRand: Vector que especifica las posiciones de columna en el archivo de datos de los efectos aleatorios. Los usuarios pueden especificar los efectos aleatorios utilizando hRand, pRand o ambos argumentos.

Seed: número entero utilizado como semilla aleatoria para el muestreo MCMC con el fin de garantizar la reproducibilidad.

Iter: Número entero que especifica el número de iteraciones MCMC. Por defecto es '30000'.

burnin: número entero que especifica el número de iteraciones MCMC iniciales que deben descartarse.

Lag: Entero que especifica el intervalo de adelgazamiento para el muestreo MCMC. Por defecto es 10.

Una vez que el archivo de parámetros está listo, podemos proceder a ajustar nuestro modelo con la función *Bunny()* y especificaremos *Chain=TRUE* para que nos guarde las muestras de las distribuciones marginales posteriores de cada estimación. Hoy trabajaremos con estas muestras, calculando nosotros a mano las inferencias.

EJEMPLO

```
bunny_results <- Bunny(params = param_list, Chain = TRUE)
```

params: Un archivo de parámetros generado por 'iCreateParam' o 'CreateParam'

Chain: Valor lógico (por defecto=FALSE). lógico (por defecto='FALSE'). Si se establece en 'TRUE', la función escribe un archivo Excel con todas las muestras de las distribuciones posteriores de los parámetros estimados para cada rasgo para el análisis post-hoc.

LOS RESULTADOS

PosteriorChains_IMF.xlsx

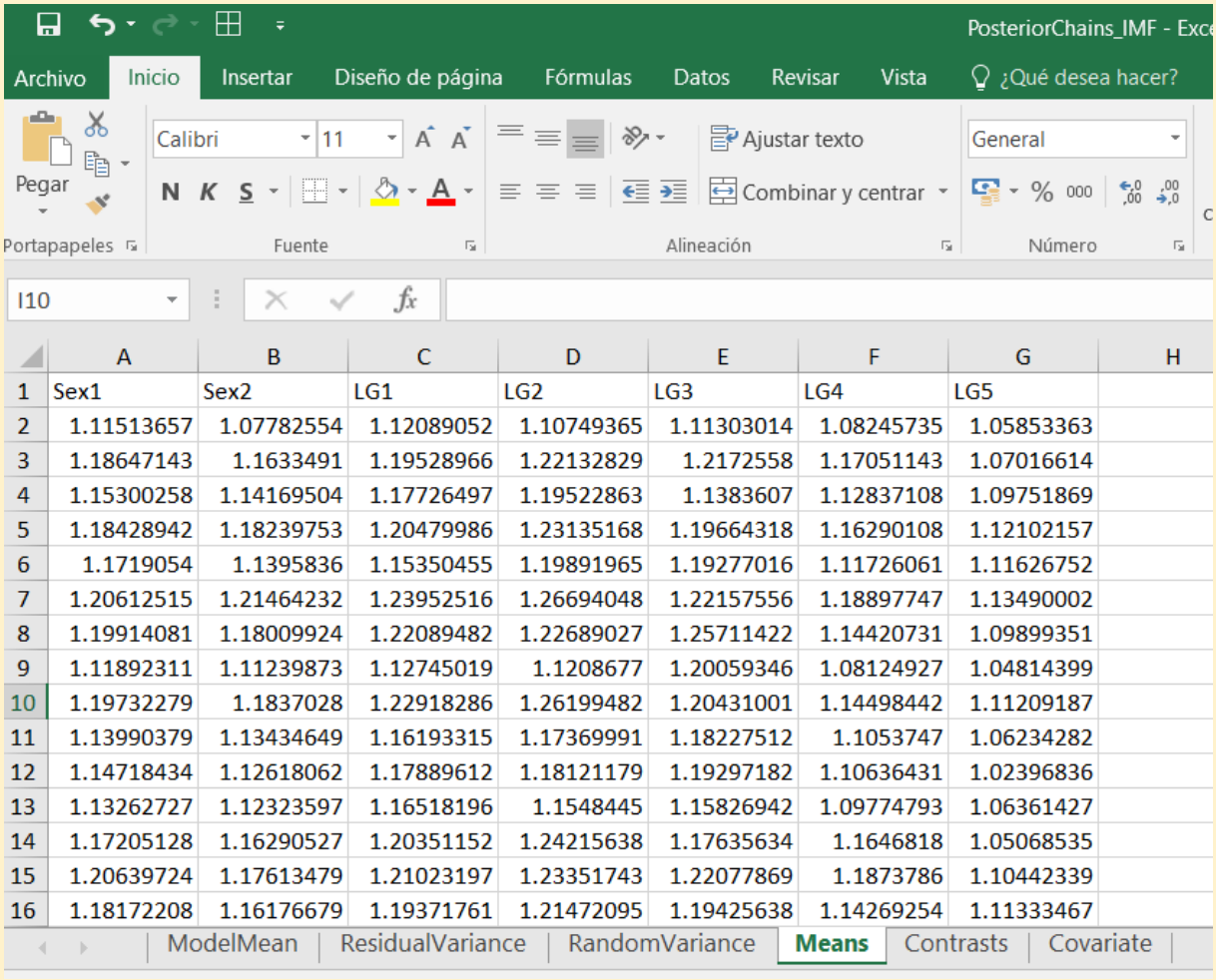
Es un fichero Excel con varias hojas que ofrecen las muestras aleatorias de la densidad posterior que se han utilizado para calcular las probabilidades de cada parámetro:

- ModelMean: La media m del modelo para el carácter
- ResidualVariance: La varianza del residuo σ_e^2
- RandomVariance: La varianza del efecto aleatorio
- Means: Los **efectos** de cada tratamiento sumándoles la media **m**
- Contrasts: Los contrastes **entre los niveles** de cada tratamiento
- Covariate: Los **coeficientes de regresión de las covariables**

Para calcular las áreas de probabilidad hay que hacer integrales. Como esto puede ser complicado, se usa un método de cálculo numérico llamado MCMC, que provee muestras aleatorias (*cadenas*) de la función de densidad posterior para calcular las probabilidades.

EJEMPLO

PosteriorChains_IMF.xlsx



	A	B	C	D	E	F	G	H
1	Sex1	Sex2	LG1	LG2	LG3	LG4	LG5	
2	1.11513657	1.07782554	1.12089052	1.10749365	1.11303014	1.08245735	1.05853363	
3	1.18647143	1.1633491	1.19528966	1.22132829	1.2172558	1.17051143	1.07016614	
4	1.15300258	1.14169504	1.17726497	1.19522863	1.1383607	1.12837108	1.09751869	
5	1.18428942	1.18239753	1.20479986	1.23135168	1.19664318	1.16290108	1.12102157	
6	1.1719054	1.1395836	1.15350455	1.19891965	1.19277016	1.11726061	1.11626752	
7	1.20612515	1.21464232	1.23952516	1.26694048	1.22157556	1.18897747	1.13490002	
8	1.19914081	1.18009924	1.22089482	1.22689027	1.25711422	1.14420731	1.09899351	
9	1.11892311	1.11239873	1.12745019	1.1208677	1.20059346	1.08124927	1.04814399	
10	1.19732279	1.1837028	1.22918286	1.26199482	1.20431001	1.14498442	1.11209187	
11	1.13990379	1.13434649	1.16193315	1.17369991	1.18227512	1.1053747	1.06234282	
12	1.14718434	1.12618062	1.17889612	1.18121179	1.19297182	1.10636431	1.02396836	
13	1.13262727	1.12323597	1.16518196	1.1548445	1.15826942	1.09774793	1.06361427	
14	1.17205128	1.16290527	1.20351152	1.24215638	1.17635634	1.1646818	1.05068535	
15	1.20639724	1.17613479	1.21023197	1.23351743	1.22077869	1.1873786	1.10442339	
16	1.18172208	1.16176679	1.19371761	1.21472095	1.19425638	1.14269254	1.11333467	

Ahora, usa el archivo de salida *PosteriorChains_IMF.xlsx* para resolver los ejercicios a continuación.

EJERCICIOS

1. Abre el archivo **PosteriorChains_IMF.xlsx**
2. En una nueva hoja de Excel, estima la respuesta a la selección en la generación 2. Esto es, la diferencia entre la Media de la línea Alta (LG4) y Baja (LG5) de la generación 2 (Pestaña *Means*). Para esto, crea una nueva columna que sea la diferencia entre las columnas LG4-LG5.
   Nota: debe coincidir con la cadena LG_4-5 en la pestaña *Contrasts*
3. Ordena la columna (LG4-LG5) del valor más bajo al más alto.
4. Estima la respuesta a la selección. Calcula la media y la mediana de esta nueva columna (LG4-LG5). La mediana es el valor que cae en el medio de la cadena MCMC; por ejemplo, si tienes 250 iteraciones, la mediana es el valor en la iteración 125.
5. Calcula la probabilidad de la respuesta de ser mayor que 0. Cuenta cuantas muestras MCMC son positivas y divide entre el número total de muestras.
6. Calcula el valor garantizado k con una probabilidad del 80%. Coge el valor situado en la posición que corresponde al 20% de todas las iteraciones. Por ejemplo, si tienes 5000 iteraciones, las primeras 20% van de 1 a 1000; coge como k el valor de la iteración 1000.
7. Usa un valor relevante del 0.07 para IMF. Calcula la probabilidad de la respuesta a la selección de ser superior a este valor relevante. Comprueba cuantas muestras MCMC son superiores a 0.07 y divide entre el número total de muestras.
8. Calcula la Probabilidad de similitud. Comprueba cuantas muestras están entre ± el valor relevante; es decir, cuantas muestras están entre -0.07 y 0.07. Divide entre el número total de muestras.
9. Ahora repito los análisis [crea de nuevo el archivo de parámetros con *iCreateParam()* o *CreateParam()* y vuelve a correr el modelo con *Bunny()*], esta vez usando RATIOS en lugar de DIFERENCIAS, y fíjate en el contraste entre niveles del tratamiento Sexo. Es decir, en una nueva hoja de Excel, crea una nueva columna que sea Sex1/Sex2.
10. Ordena la columna (Sex1/Sex2) del valor más bajo a más alto.
11. Calcula la Media y la Mediana de esta columna.
12. Calcula la probabilidad del ratio de ser mayor a 1
13. Calcula el valor garantizado con una probabilidad del 77%
14. Considera como valor relevante tener más del 10% de IMF. Calcula la probabilidad de Sex1/Sex2 de ser mayor que este valor relevante, es decir 1.10 (o 1/1.10 si el ratio es mayor a 1)
15. Calcula la probabilidad de similitud. Es decir, la proporción de muestras que están entre 0.90 and 1.10.

Cuando acabes, puedes utilizar la función *iBayes* (o *Bayes*) para comprobar tus respuestas.

EJEMPLO

```
inferences <- iBayes(params=my_param_file, bunny=bunny_results)
```

Alternativamente, podemos usar la función *Bayes*() con los siguientes argumentos:

```
inferences <- Bayes(
  params = param_list,
  bunny = bunny_results,
  HPD = 0.95,
  K = TRUE,
  probK = 0.80,
  PR = TRUE,
  R = 0.07,
  PS = TRUE,
  SaveTable = TRUE,
  plot = TRUE)
```

Donde:

Params: Un archivo de parámetros generado por `iCreateParam` o `CreateParam`.

Bunny: Una lista producida por la función *Bunny*, incluyendo todas las especificaciones del modelo y muestras posteriores de inferencias para diferentes caracteres.

HPD: Un valor escalar entre 0 y 1 que especifica la probabilidad deseada para calcular el intervalo de densidad posterior más alta (HPD). El valor predeterminado es 0.95. El intervalo HPD se calcula para la media del modelo, los tratamientos, las covariables, los contrastes y los componentes de varianza.

P0: Un valor lógico (por defecto es `TRUE`). Si es `TRUE`, calcula la probabilidad de que la distribución posterior sea mayor que cero si la mediana es positiva, o menor que cero si la mediana es negativa. Esto se aplica a tratamientos, contrastes y covariables. Para contrastes entre tratamientos expresados como ratios, calcula probabilidades de ser mayor o menor que 1.

K: Un valor lógico(por defecto es `FALSE`). Si es `TRUE`, calcula un valor "garantizado" de la distribución posterior con una probabilidad especificada (`probK`). Este valor sólo se muestra si su signo coincide con el signo de la mediana de la distribución posterior. Se aplica a tratamientos, contrastes, covariables y componentes de varianza.

probK: Un valor numérico entre 0 y 1 que especifica el umbral de probabilidad para calcular el valor garantizado cuando `K` es `TRUE`. Si `K` es `TRUE` y no se especifica `probK`, se utiliza un valor por defecto de 0.80.

PR: Un valor lógico (por defecto es `FALSE`). Si es `TRUE`, calcula la probabilidad de que la distribución posterior sea mayor que un valor relevante (`R`) si la mediana es positiva, o menor que `R` si la mediana es negativa. Sólo se aplica a los contrastes.

R: Un vector numérico que contiene un valor relevante para cada carácter. Si `PR` es `TRUE`, este argumento es obligatorio. Para contrastes entre tratamientos expresados como ratios, `R` debe considerarse en porcentaje (por ejemplo, 1,1 para un incremento del 10%). Los valores sugeridos son un tercio de la desviación estándar del rasgo para las diferencias, y el 10 por ciento para los ratios.

PS: Un valor lógico (por defecto es `FALSE`).Si es `TRUE`, calcula la probabilidad de similitud, es decir, la probabilidad de que la distribución posterior se encuentre entre `-R` y `R` (o entre `1/R` y `R` si es un ratio). Sólo se aplica a los contrastes.

SaveTable: Un valor lógico (por defecto es `TRUE`).Si es `TRUE`, las inferencias de las distribuciones posteriores impresas en la consola también se guardan en un archivo CSV.

Plot: Un valor lógico (por defecto es `FALSE`).Si es `TRUE`, genera y guarda gráficos de distribuciones posteriores para contrastes en formato .tiff, resaltando los valores `P0` y `PR` si es aplicable.

Podéis encontrar información sobre cada función usando ??RabbitR

APÉNDICE: LOS MODELOS ESTADÍSTICOS

Los efectos aleatorios

Son efectos en cuyos valores no estamos interesados. En estadística clásica se supone que si se repite el experimento los valores de estos efectos cambian de forma aleatoria. Por ejemplo, si estamos interesados en el efecto de un tratamiento sobre el carácter tamaño de camada y se dispone de varias camadas por hembra, no estamos interesados en el valor de cada hembra concreta; se supone que si se repite el experimento las hembras serán distintas y los efectos de cada una de ellas también. Lo mismo ocurre con el efecto del tamaño de camada común; si medimos la grasa perirrenal en dos hermanos de la misma camada, el efecto de la hembra sobre el valor del carácter Grasa perirrenal en cada hermano es aleatorio, porque repitiendo el experimento cambiarían las hembras.

Los efectos de ruido

Son efectos en cuyos valores no estamos interesados. Nos hubiera gustado que todos los animales hubieran sido medidos en la misma estación y el mismo orden de parto, pero no ha sido posible, por lo que se corrige por estos efectos para que los datos vengan *como si* pertenecieran todos al mismo parto y la misma estación. Obviamente a cada animal la estación y el orden del parto le afecta de una forma diferente, por lo que cuando se corrige por un efecto de ruido se corrige por un *efecto medio* de verano o de invierno y por un *efecto medio* de primer o segundo parto, no por el efecto real que ha sufrido cada animal concreto. Conviene que haya un número suficiente de individuos dentro de cada nivel de esos efectos para que la corrección media esté bien estimada.

Los tratamientos

Son los efectos en cuyos niveles estamos interesados. En nuestro caso deseamos averiguar los valores de los caracteres en cada generación y también deseamos saber si hay diferencias entre sexos (generalmente los tratamientos figuran en el título del trabajo mientras que los efectos de ruido no: un título como “Efecto de la generación y el sexo sobre la grasa intramuscular” implica que ambos son tratamientos). **Rabbit** calcula las medias de los niveles de todos los tratamientos (lo que en estadística clásica equivaldría a LSmeans o medias por mínimos cuadrados) y hace inferencias bayesianas sobre las medias y sobre las diferencias o ratios entre niveles de cada tratamiento. Como en el caso de los efectos de ruido, los efectos de tratamiento son *efectos medios*.

Las covariables

Las covariables son efectos que actúan de forma lineal. Suelen utilizarse como los efectos de ruido, pero en ocasiones tienen interés por ellas mismas. Las covariables pueden ser

discontinuas (como el tamaño de camada, por ejemplo), y por el contrario, ciertos caracteres continuos como el peso pueden transformarse en efectos clasificándolos por niveles (en grupos de 100 en 100g, por ejemplo); por tanto lo que distingue a la covariable no es ser continua o no sino si su relación con el carácter es lineal.

¿Cuándo debo poner un efecto de ruido en un modelo?

Cuando sea razonable. El investigador debe conocer por su conocimiento del problema cuándo debe añadirse un efecto; la estadística es una herramienta, no sustituye al conocimiento. ***La única forma en la que NO se debe poner o quitar un efecto de ruido es atendiendo a si es significativo o no.*** Significativo sólo quiere decir "muestra grande" y n.s. sólo quiere decir "muestra pequeña", no que el efecto sea o no importante; si hay muchos datos, todos los efectos son significativos, y si hay pocos datos no lo son. Es más, si un efecto no existe realmente (por ejemplo, el orden de parto) y lo ponemos, los resultados apenas van a cambiar, y si algo cambian se debe a que por azar la muestra que analizamos tiene valores superiores para un nivel u otro, algo que no deseamos y que estamos de acuerdo en corregir. La precisión mejorará debido a que habrá una reducción de la varianza del error y la pérdida de grados de libertad será mínima. La mayor parte de experimentos de laboratorio tiene un reducido número de efectos de ruido. Los datos de campo suelen tener más efectos de ruido, pero también más datos para estimarlos, por lo que la inclusión de efectos poco importantes no tiene prácticamente consecuencias. En caso de duda puede hacerse un análisis exploratorio con estadística clásica para ver las consecuencias de incluir o no un efecto, pero es raro que su inclusión presente problemas.

¿Cuándo debo incluir un efecto aleatorio?

Los resultados con o sin efecto aleatorio suelen ser bastante similares, aunque hay algunas excepciones. La incertidumbre ("errores estándar", en estadística clásica) se estima mejor incluyendo un efecto aleatorio cuando éste existe. Un efecto lo consideramos aleatorio cuando es difícil de corregir como efecto fijo porque hay muy pocos datos por nivel. Por ejemplo:

- Datos repetidos en un individuo: varias lactaciones o varias camadas. En ese caso el individuo es un efecto aleatorio para la producción de leche o el tamaño de camada.
- Datos que han recibido un ambiente común. Por ejemplo, si tomamos datos de hermanos, el efecto de camada común sería un efecto aleatorio.

Si hay pocos datos por nivel, *la corrección del efecto aleatorio es ligera*, y si hay muchos datos la corrección es prácticamente la misma que la que produce un efecto fijo. Si un efecto fijo tiene pocos datos por nivel, la estima de ese efecto no es buena y puede conducir a errores, por eso en esos casos se prefiere tomar el efecto como aleatorio. Por ejemplo, el efecto de rebaño-año-estación en vacuno de leche podría considerarse como aleatorio si tuviéramos pocos datos por rebaño-año-estación. Hay otras soluciones, como la de agrupar niveles de efectos hasta que haya un número suficiente de datos en el nivel y considerarlo como efecto fijo, pero en ese caso la corrección como *efecto medio* es peor, porque corrige con un solo número un periodo que puede ser muy grande; no hay una solución estándar para el problema y hay que hacer pruebas.

¿Cuándo debo incluir interacciones?

Las interacciones se producen cuando la media de una combinación de efectos (por ejemplo, Invierno y primer parto) no se explica simplemente por la suma de los efectos; es decir, si los individuos en invierno tienen 0.9 lechones menos que la media, y los de primer parto 1.1 lechones menos que la media, los de primer parto en invierno tienen 3.2 lechones menos que la media en lugar de 0.9+1.1=2 lechones menos.

Si se estiman todas las interacciones, muchas de ellas son frecuentemente difíciles de explicar y sus valores dependen más de un efecto de muestreo que de que exista realmente una interacción. Por ejemplo, si tenemos cuatro estaciones y cuatro órdenes de parto, hay que estimar 16 interacciones, por lo que es frecuente que alguna de ellas, por ejemplo, la de Otoño con el tercer parto, tenga un valor elevado simplemente por el efecto de muestreo que se produce cuando se estiman muchos efectos con muchos niveles y muchas interacciones; en estadística clásica salen algunas interacciones significativas por azar.

Se recomienda en general no incluir las interacciones, salvo que haya motivos suficientes como para hacerlo. En general no se suele estar interesado en estimar todas las interacciones, por lo que es preferible en muchas ocasiones estimar solamente interacciones concretas; por ejemplo, la de Invierno con primer parto. Una forma cómoda de hacerlo es calcular un modelo en el que haya un solo efecto EstaciónParto; en nuestro ejemplo el fichero de datos tendría un primer nivel InviernoPrimerparto y un segundo nivel con el resto de las estaciones y partos, por lo que sólo se compararían dos niveles, que son precisamente los niveles en los que estamos interesados.

¿Debo preocuparme si mis datos no siguen una distribución normal?

Rabbit supone que la distribución de los datos es Normal; pero si no fuera así, tampoco es causa de especial preocupación, salvando el caso de distribuciones anómalas o de muestras muy pequeñas. Como Ronald Fisher (uno de los padres de la estadística clásica) indicó en muchas ocasiones, aunque los datos no sigan una distribución Normal, las *medias muestrales*, que es lo que estoy comparando, sí que siguen una distribución Normal si la muestra tiene un número suficiente de datos. El carácter número de muertos en una camada de conejos, por ejemplo, sigue una distribución escalonada en donde la mayor parte de camadas tienen valor 0, algunas 1, unas pocas 2, y ya muy pocas 3 o más. Esto obviamente no es una distribución Normal. Pero si tomamos 30 camadas para estimar la media de camadas de una raza determinada, el valor medio será 0.8; si repetimos el experimento, será 0.9, si lo repetimos, 0.7, y así hasta que muchas muestras se distribuirán en torno al valor verdadero de forma Normal. Claro que, si en vez de 30 camadas por muestra tomamos una sola camada por muestra, repetiremos la distribución escalonada y no tenderemos a la Normal, por eso hace falta que las muestras sean lo suficientemente grandes. Lo mismo ocurre con la estadística bayesiana; sea cual sea la distribución original, las distribuciones posteriores tienden a la Normal si la muestra es lo suficientemente grande, (debe ser más grande si los datos se alejan mucho de la Normal). En general las comparaciones de tratamientos son muy robustas a la falta de normalidad de los datos y no suele hace falta hacer transformaciones, siempre difíciles de interpretar.