

Tablas de contingencia. Asociación. Modelos log-



Introducción.

En esta práctica vamos a explorar algunos métodos para extraer información útil cuando los datos de **los que disponemos son categóricos, es decir, vienen dados en forma de factores** o variables cualitativas o categóricas (**atributos**).

En estos casos, la información suele sintetizarse y presentarse mediante las denominadas “**tablas de contingencia**”. En este tipo de tablas, se muestran las frecuencias conjuntas, es decir, el número de casos que comparten los distintos niveles o categorías de los diferentes factores.

Cuando se trabaja con varios factores, representados en su correspondiente tabla de contingencia, uno de los análisis más interesantes es determinar si existe **asociación entre los factores**. Esto es, si se aprecia algún tipo de relación estadística entre estas variables categóricas o cualitativas, en el sentido de si se puede afirmar que el hecho de que los individuos de la muestra o población tomen ciertos niveles o categorías en unos factores hace que tiendan a tomar ciertos niveles o categorías de otro u otros factores.

El caso más simple es en el que solo se trabaja con dos atributos, variables categóricas o factores, y existen varias medidas para verificar la existencia de asociación entre ellos. Este caso lo veremos en la parte inicial de la práctica.

No obstante, a veces intervienen en el análisis más de dos factores, que además pueden interactuar entre sí. En este caso, una opción es recurrir a la estimación de **modelos logarítmico-lineales**, que permiten trabajar

simultáneamente con varios factores y estudiar sus interacciones. A esto dedicaremos la parte final de la práctica.

Vamos a aplicar los conceptos anteriores al caso de las empresas eólicas españolas, sobre las cuales se ha extraído información cualitativa concretada en los siguientes factores o atributos:

- **DIMENSION:** tamaño del grupo empresarial al que pertenece la empresa en cuestión. Tiene tres niveles: grande, media y reducida.
- **AUTOFINA:** capacidad de autofinanciación de la empresa a medio y largo plazo. Tiene tres niveles: alta, positiva y negativa.
- **FJUR:** forma jurídica. Tiene dos posibles categorías: Sociedad anónima o Sociedad limitada.

Preparando Datos.

Abriremos **R-Studio** y crearemos nuestro **proyecto** siguiendo la instrucción **File → New Project**. Nos preguntará si crea el proyecto en una nueva carpeta o en una ya existente. Vamos a crearlo, por ejemplo, en el disco extraíble D, carpeta R, subcarpeta “**loglineal**”, que ya existe. Aparecerá una ventana para buscar la carpeta y, tras localizarla, pulsaremos **Open** y **Create Project**.

Vamos a ir a la carpeta del proyecto y vamos a guardar en ella los dos archivos de esta práctica: un archivo de **Microsoft® Excel®** llamado “eolica_contingencia.xlsx” y un *script* denominado “loglineal_eolica.R”. Si abrimos el archivo de **Microsoft® Excel®**, comprobaremos que se compone de tres hojas. La primera muestra el criterio de búsqueda de casos en la base de datos **Sabi®**; la segunda recoge la descripción de las variables consideradas, y la tercera (hoja “Datos”) guarda los datos que debemos importar desde **R-Studio**. Estos datos se corresponden con diferentes variables económico-financieras de las empresas productoras de electricidad mediante generación eólica.

Luego vamos a cerrar el archivo de **Microsoft® Excel®** y volveremos a **R-Studio**. Vamos a abrir nuestro *script* “loglineal_eolica.R” con **File → Open**

File... Este script contiene el programa que vamos a ir ejecutando en la práctica.

La primera línea / instrucción en el script es:

```
rm(list = ls())
```

La instrucción tiene como objeto limpiar el *Environment* de objetos de anteriores sesiones de trabajo. Para importar los datos, ejecutaremos el código:

```
## Importando datos

library (readxl)
eolicas <- read_excel("eolica_contingencia.xlsx", sheet = "Datos")
```

Podemos observar cómo, en el *Environment*, ya aparece un objeto. Este objeto es una estructura de datos tipo *data frame*, se llama “eolicas” y contiene 19 columnas, una por cada variable del archivo de Microsoft® Excel®.

R ha considerado la primera columna (NOMBRE) como una variable de tipo cualitativo. En realidad, no es una variable, sino el nombre de los casos (empresas). Para evitar que R tome los nombres de los individuos como una variable, podemos redefinir nuestro *data frame* diciéndole que tome esa primera columna como los nombres de los individuos o casos (filas):

```
eolicas <- data.frame(eolicas, row.names = 1)
```

En la línea anterior hemos asignado al *data frame* “eolicas” los propios datos de “eolicas”; pero indicando que la primera columna de datos no es una variable; sino el nombre de los individuos, casos o filas. Así, vemos que ya no aparece NOMBRE como variable, y en el *Environment* el *data frame* “eolicas” cuenta con 18 variables (una menos). Con *summary()* comprobamos las variables que contiene el *data frame*, y sus características:

```
summary (eolicas)
```

En nuestro análisis solo vamos a considerar los factores o atributos AUTOFINA y DIMENSION. Por ello, crearemos con ellas un nuevo *data frame* llamado, por ejemplo, “originales”:

```
## Seleccionando factores/atributos para el analisis

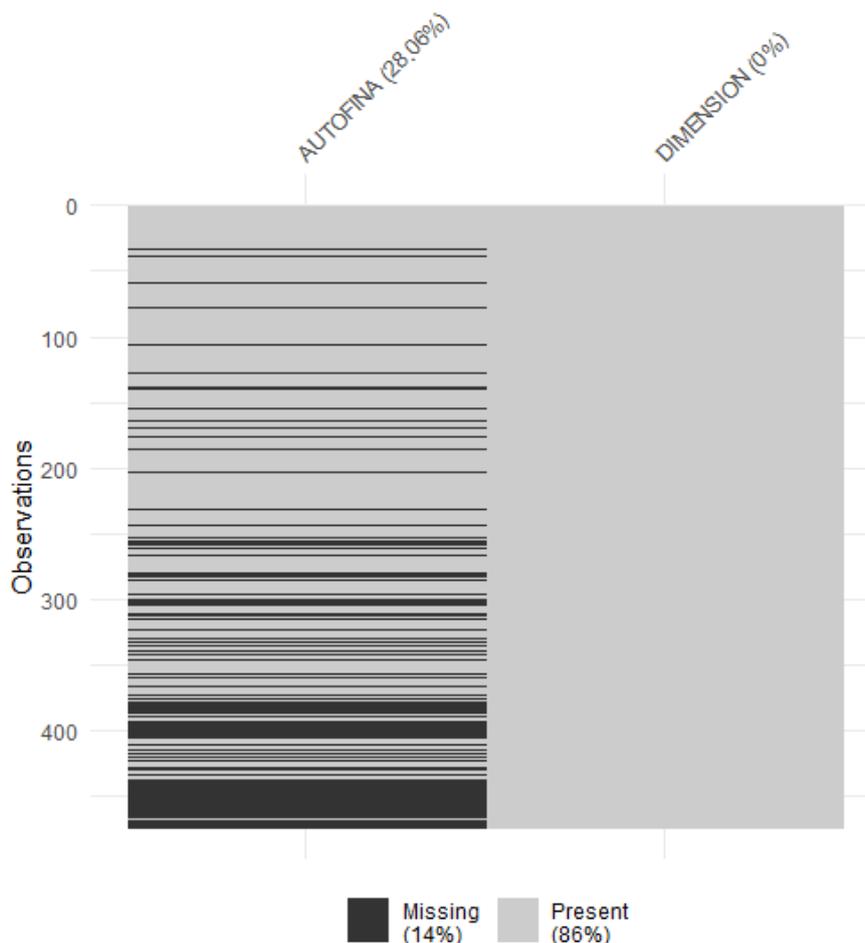
library(dplyr)
originales<-select(eolicas, AUTOFINA, DIMENSION)
summary (originales)
```

El siguiente paso será localizar los posibles **missing values**, ya que para realizar el análisis es necesario que todos los casos posean dato para todas las variables originales. Para tener una idea general, se puede utilizar la función `vis_miss()` del paquete `visdat`, que nos localizará gráficamente los **missing values** de las diferentes variables, y calculará el porcentaje de casos que supone, con respecto al total de observaciones:

```
## Identificando missing values.

library(visdat)
vis_miss(originales)
```

El resultado del código anterior es el siguiente gráfico:



Se observa cómo el atributo DIMENSION carece de *missing values*. En cambio, el 28% de los casos carece de dato en el factor AUTOFINA, luego tendrán que ser eliminados. Para localizarlos, podemos filtrar nuestro *data frame* con las herramientas de **dplyr**:

```
originales %>% filter(is.na(AUTOFINA) | is.na(DIMENSION)) %>%  
  select(AUTOFINA, DIMENSION)
```

Los casos con *missing values* que aparecen en pantalla deberían ser analizados para estudiar la posibilidad de obtener los datos faltantes mediante la consulta de otras fuentes o algún proceso de estimación. Si no es posible, como supondremos en el ejemplo, tendrán que ser eliminados, por ejemplo, con el código:

```
originales <- originales %>%  
  filter(! is.na(AUTOFINA) & ! is.na(DIMENSION))
```

Como resultado, el *data frame* “originales” pasa a tener 341 casos (133 casos eliminados).

Al trabajar con atributos o factores, no tiene sentido realizar la detección de *outliers*.

Construcción y descripción de la tabla de contingencia bidimensional.

El primer paso del análisis es **construir la tabla de contingencia** a partir de la información contenida en nuestro *data frame* “originales”. La función para convertir datos del *data frame* en una estructura de almacenamiento de datos especial llamada **table** (que es la tabla de contingencia) es precisamente **table()**:

```
## Construyendo Tabla de contingencia  
tab.originales <- table(originales)
```

En el **Environment** aparecerá el objeto “tab.originales”, de tipo *table*. Para representarla, incluyendo sus frecuencias marginales, podremos utilizar los recursos de los paquetes **knitr** y **kableExtra**, mediante el código:

```
## Construyendo Tabla de contingencia  
  
library(knitr)  
library(kableExtra)  
knitr.table.format = "html"
```

```

addmargins(tab.originales) %>%
  kable(caption="Empresas eólicas") %>%
  kable_styling(full_width = F, bootstrap_options = "striped",
"bordered", "condensed", position = "center", font_size = 12) %>%
  add_header_above(c("AUTOFINANCIACION"= 1, DIMENSION=3, " "=1), bold=T,
line=T) %>%
  row_spec(0, bold= T, align = "c") %>%
  column_spec(1, bold = T)

```

El resultado será:

Empresas eólicas

AUTOFINANCIACION	DIMENSION			Sum
	GRANDE	MEDIA	REDUCIDA	
ALTA	79	27	53	159
NEGATIVA	5	18	9	32
POSITIVA	65	41	44	150
Sum	149	86	106	341

La tabla también se puede representar gráficamente mediante la función **mosaic()** de la librería **vcd**, con lo que se percibirán mejor la magnitud de las frecuencias conjuntas (celdas de la tabla):

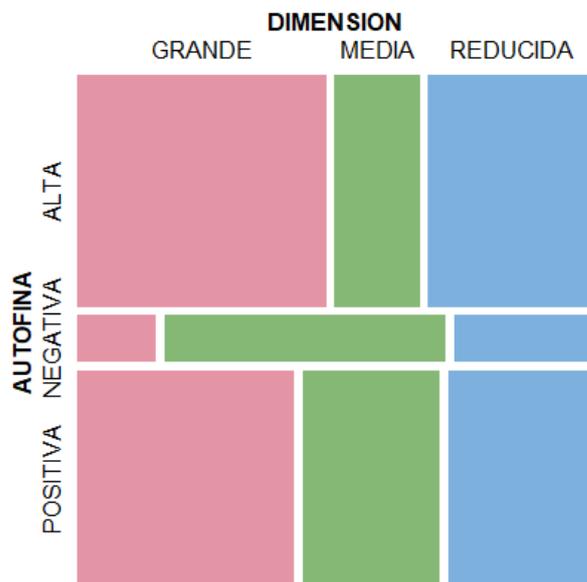
```

library(vcd)
mosaic(tab.originales,
  main="Eólicas: Capac. autofinanciación y Dim. grupo
empresarial.",
  sub = "Tabla de contingencia",
  shade=T,
  gp= shading_Marimekko(tab.originales),
  main_gp = gpar(fontsize = 14), sub_gp = gpar(fontsize = 12))

```

Obteniéndose la figura:

Eólicas: Capac. autofinanciación y Dim. grupo empresarial.



Puede observarse que el mayor número de frecuencias se da para las empresas pertenecientes a grupos empresariales de tamaño grande y con una capacidad de autofinanciación a medio y largo plazo alta. En cambio, el menor número de casos es el de empresas eólicas pertenecientes a grupos empresariales de tamaño grande y capacidad de autofinanciación a medio y largo plazo negativa.

Para percibir mejor las frecuencias marginales de cada nivel o categoría de ambos atributos o factores, pueden representarse estas mediante gráficos de barras. Para conseguirlo, vamos a utilizar el paquete gráfico `ggplot2`, y el paquete `patchwork`, que simplemente permite maquetar varios gráficos generados con `ggplot2` para que aparezcan juntos, en una determinada disposición. El código es el siguiente:

```
## Representando frecuencias de categorias en factores

library (ggplot2)
library (patchwork)

g1 <- ggplot(originales, mapping= aes(x= DIMENSION, fill = DIMENSION))
+
  geom_bar() +
  ggtitle("Tamaño del grupo empresarial", subtitle = "Empresas eólicas")
+
  ylab("Frecuencias") +
  xlab("Dimensión")

g2 <- ggplot(originales, mapping= aes(x= AUTOFINA, fill = AUTOFINA)) +
```

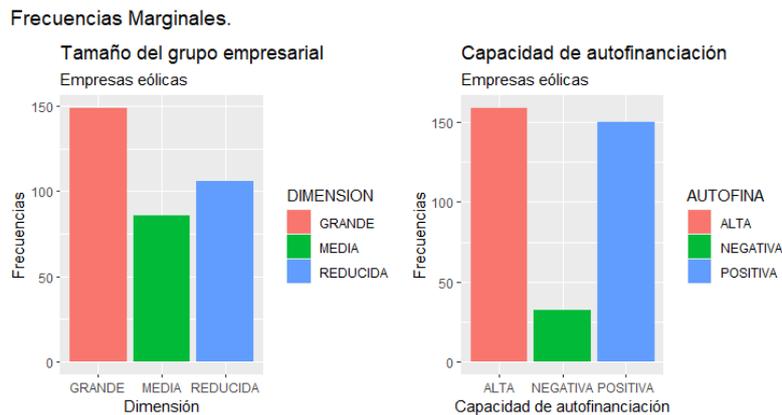
```

geom_bar() +
ggtitle("Capacidad de autofinanciación", subtitle = "Empresas
eólicas") +
ylab("Frecuencias") +
xlab("Capacidad de autofinanciación")

(g1 + g2) + plot_annotation(title = "Frecuencias Marginales.",
                             theme = theme(plot.title =
element_text(size = 14)))

```

El resultado obtenido es:



Puede observarse cómo, en cuanto al tamaño del grupo empresarial de pertenencia (DIMENSION), son mayoritarias las empresas pertenecientes a grupos de gran volumen. En cuanto a la capacidad de autofinanciación a largo y medio plazo (AUTOFINA), las de capacidad alta o positiva presentan frecuencias similares; y son muy minoritarias las empresas con capacidad negativa.

Asociación.

Uno de los aspectos más interesantes, a la hora de analizar una tabla de contingencia bidimensional, es estudiar si existe asociación entre ambos factores o atributos, es decir, si existe cierta tendencia a acumularse frecuencias en ciertas intersecciones entre categorías y niveles de uno y otro.

Para saber si existe asociación o no entre ambos factores, la idea es comparar mediante una medida estadística lo “cercana” o “lejana” que está nuestra tabla de contingencia con respecto a la tabla teórica que tendría que darse (con la misma frecuencia total o número de casos) en caso de independencia entre tales factores o atributos. Este es el cometido de la

prueba de independencia Chi-cuadrado entre las categorías de ambas variables cualitativas. La función que realiza dicha prueba es `chisq.test()`.

```
#TEST DE INDEPENDENCIA / ASOCIACION  
chisq.test(tab.originales)
```

El resultado de la prueba es:

```
Pearson's Chi-squared test  
  
data:  tab.originales  
X-squared = 24.34, df = 4, p-value = 6.828e-05
```

El p-valor de la prueba es menor que 0.05, por lo que rechazamos la hipótesis nula de independencia: hay cierta **relación estadística** entre las categorías de los atributos (asociación).

La utilización del coeficiente *Chi-cuadrado* tiene una debilidad a la hora de “medir” el grado de asociación entre los dos factores: no está acotado entre 0 y 1. Una medida alternativa que sí está acotada entre 0 y 1 es el *coeficiente V de Cramer*, que puede obtenerse con la función `CramerV()` del paquete **DescTools**:

```
library (DescTools)  
CramerV(tab.originales)
```

El valor del coeficiente es:

```
[1] 0.1889148
```

El valor anterior indica que, aunque hay asociación según la prueba de independencia *Chi-cuadrado*, esta asociación es moderada.

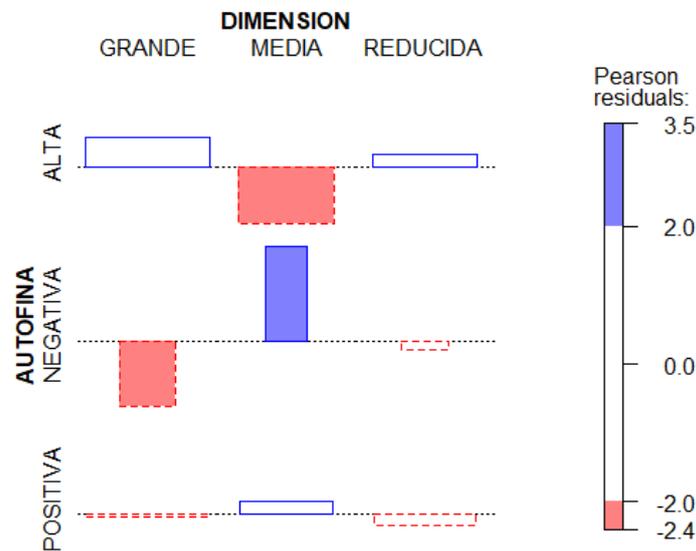
Precisamente, se pueden mostrar gráficamente las desviaciones o diferencias (residuos tipificados) entre las frecuencias de la tabla observada con respecto a las frecuencias que deberían existir en el caso teórico de independencia entre tales atributos o factores. Para ello, cabe utilizar la función `assoc()` del paquete **vcd**:

```
assoc(tab.originales,  
      main="Eólicas: Capac. autofinanciación y Dim. grupo empresarial.",  
      sub = "Asociación",  
      compress= F,  
      gp= shading_Friendly(tab.originales),  
      main_gp = gpar(fontsize = 14), sub_gp = gpar(fontsize = 12),
```

legend=TRUE)

Obteniéndose:

Eólicas: Capac. autofinanciación y Dim. grupo empresarial.



Asociación

Se observa cómo en la combinación “DIMENSION media” con “AUTOFINA negativa”, los errores son elevados y con signo positivo (relleno azul), lo que indica que, en este caso, la frecuencia observada es sensiblemente superior a la que cabría esperar en caso de independencia. En cambio, en las combinaciones “DIMENSION media” con “AUTOFINA alta”, y “DIMENSION grande” con “AUTOFINA negativa”; los errores son apreciablemente altos, pero con signo negativo (relleno anaranjado); lo que se interpreta como que las frecuencias observadas son sensiblemente inferiores a las que debería haber en caso de independencia. En definitiva, cuantos más bloques “coloreados” hay en el gráfico de asociación, más intensa será la asociación entre ambos factores o atributos; y más lejos se estará de la situación de independencia estadística.

Modelos logarítmico-lineales.

La asociación entre los dos factores se puede estudiar también mediante **modelos log-lineales**, aunque la verdadera fortaleza de estos modelos se halla en el análisis de asociación para más de dos factores. Así pues, vamos

a introducir en el análisis un factor o atributo más, el FJUR, correspondiente a la forma jurídica de la empresa.

Previo a la aplicación del modelo, hemos de preparar los datos, para eliminar los *missing values*, ya que ahora se ha incluido un tercer factor o atributo:

```
## MODELOS LOG-LINEALES

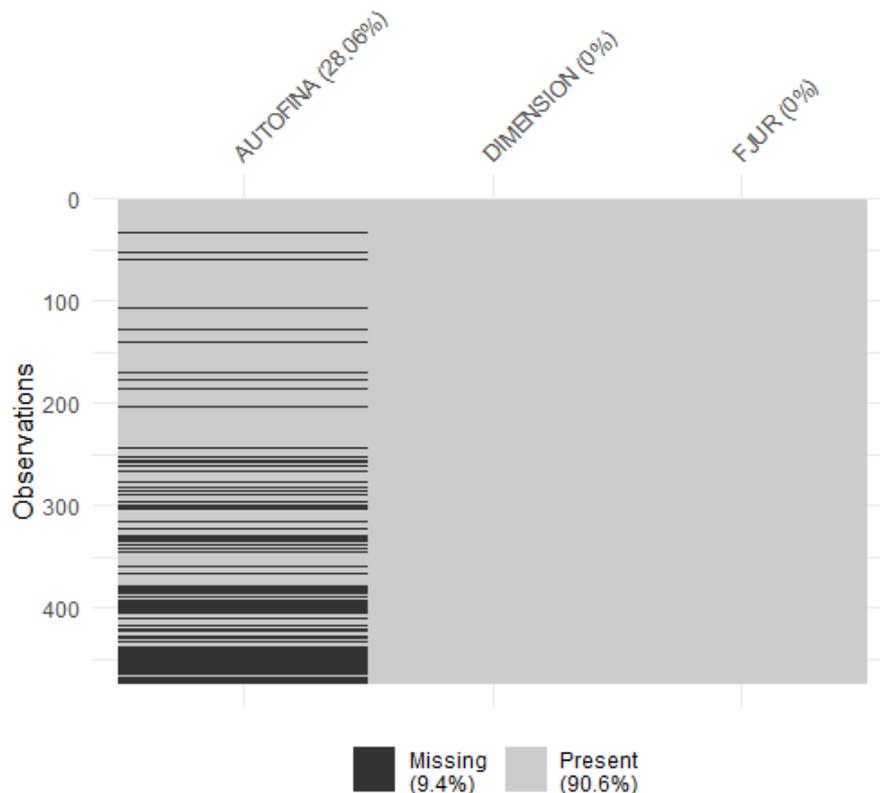
## Seleccionando factores/atributos para el analisis

originales2<-select(eolicas, AUTOFINA, DIMENSION, FJUR)
summary (originales2)

## Identificando missing values.

vis_miss(originales2)
originales2 %>% filter(is.na(AUTOFINA) | is.na(DIMENSION) | is.na(FJUR))
%>%
  select(AUTOFINA, DIMENSION, FJUR)
originales2 <- originales2 %>%
  filter(! is.na(AUTOFINA) & ! is.na(DIMENSION) & ! is.na(FJUR))
```

Se ha creado el *data frame* “originales2”, con el que se construirá la tabla de contingencia correspondiente. El gráfico proporcionado por `vis_miss()` es:



Puede apreciarse cómo, de nuevo, los *missing values* solo se concentran en el factor o atributo AUTOFINA, por lo que se eliminarán los mismos datos que en el caso bidimensional anterior.

Ahora, se construirá la tabla de contingencia, que denominaremos "tab.originales2". La presentaremos en formato tabla, y la visualizaremos gráficamente con `mosaic()`:

```
## Construyendo Tabla de contingencia

tab.originales2 <- table(originales2)
kable(tab.originales2, caption = "Empresas eólicas",
      col.names = c("AUTOFINANCIACION", "DIMENSION", "F. JURIDICA",
                    "Frecuencias")) %>%
  kable_styling(full_width = F, bootstrap_options = "striped",
                "bordered", "condensed", position = "center", font_size = 12) %>%
  row_spec(0, bold= T, align = "c")

mosaic(tab.originales2,
       main="Eólicas: Capac. autofinanciación, Dim. grupo empresarial,
       forma jurídica.",
       shade=T,
       gp= shading_Marimekko(tab.originales),
       main_gp = gpar(fontsize = 11), sub_gp = gpar(fontsize = 9))
```

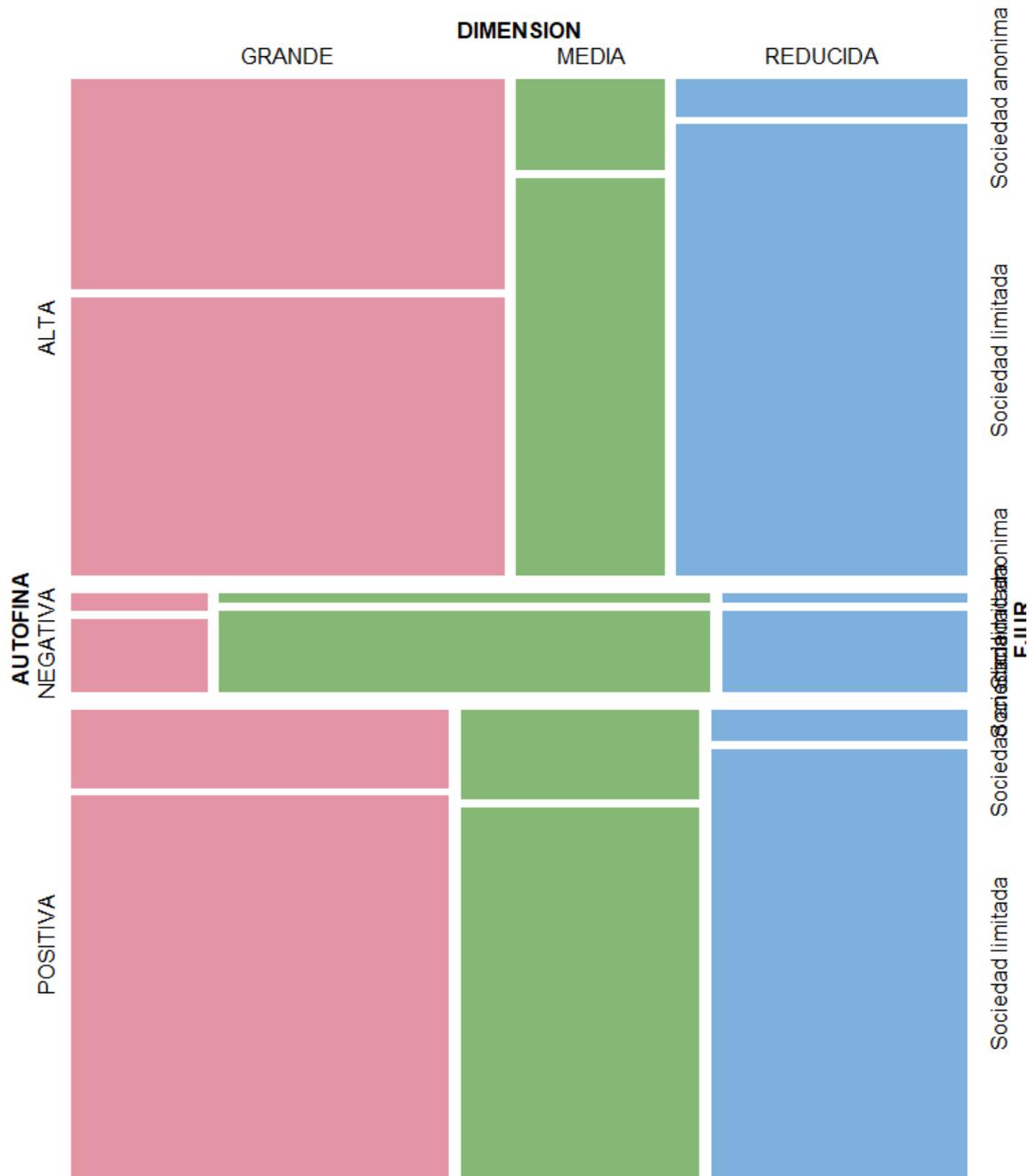
La tabla obtenida es:

Empresas eólicas

AUTOFINANCIACION	DIMENSION	F. JURIDICA	Frecuencias
ALTA	GRANDE	Sociedad anonima	34
NEGATIVA	GRANDE	Sociedad anonima	1
POSITIVA	GRANDE	Sociedad anonima	11
ALTA	MEDIA	Sociedad anonima	5
NEGATIVA	MEDIA	Sociedad anonima	2
POSITIVA	MEDIA	Sociedad anonima	8
ALTA	REDUCIDA	Sociedad anonima	4
NEGATIVA	REDUCIDA	Sociedad anonima	1
POSITIVA	REDUCIDA	Sociedad anonima	3
ALTA	GRANDE	Sociedad limitada	45
NEGATIVA	GRANDE	Sociedad limitada	4
POSITIVA	GRANDE	Sociedad limitada	54
ALTA	MEDIA	Sociedad limitada	22
NEGATIVA	MEDIA	Sociedad limitada	16
POSITIVA	MEDIA	Sociedad limitada	33
ALTA	REDUCIDA	Sociedad limitada	49
NEGATIVA	REDUCIDA	Sociedad limitada	8
POSITIVA	REDUCIDA	Sociedad limitada	41

Y el gráfico de mosaico:

Eólicas: Capac. autofinanciación, Dim. grupo empresarial, forma jurídica.



En cuanto a las frecuencias marginales de cada nivel o categoría de los tres atributos o factores, pueden representarse estas mediante gráficos de barras. El código es el siguiente:

```
g1b <- ggplot(originales2, mapping= aes(x= DIMENSION, fill = DIMENSION))
+
  geom_bar() +
  ggtitle("Tamaño del grupo empresarial", subtitle = "Empresas
eólicas") +
  ylab("Frecuencias") +
  xlab("Dimensión")
```

```

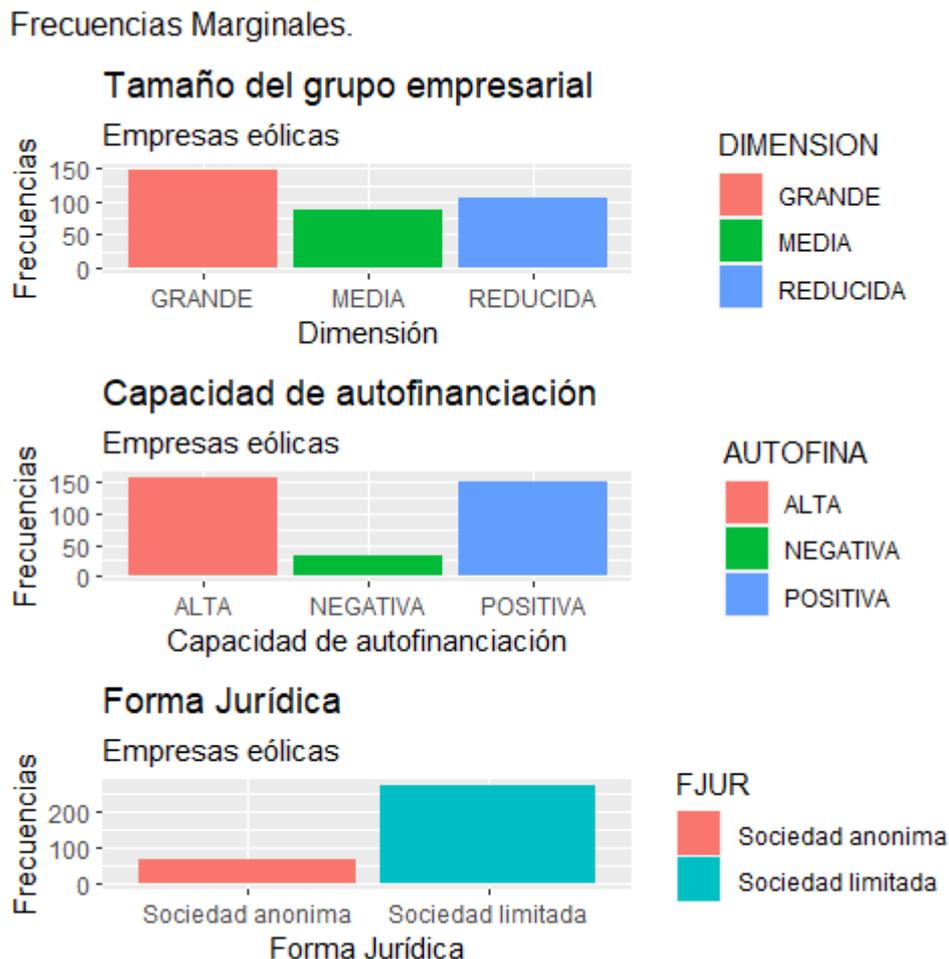
g2b <- ggplot(originales2, mapping= aes(x= AUTOFINA, fill = AUTOFINA))
+
  geom_bar() +
  ggtitle("Capacidad de autofinanciación", subtitle = "Empresas
eólicas") +
  ylab("Frecuencias") +
  xlab("Capacidad de autofinanciación")

g3b <- ggplot(originales2, mapping= aes(x= FJUR, fill = FJUR)) +
  geom_bar() +
  ggtitle("Forma Jurídica", subtitle = "Empresas eólicas") +
  ylab("Frecuencias") +
  xlab("Forma Jurídica")

(g1b / g2b / g3b) + plot_annotation(title = "Frecuencias Marginales.",
                                   theme = theme(plot.title =
element_text(size = 12)))

```

El resultado obtenido es:



Los modelos log-lineales se especifican y estiman mediante la función `loglm()` de la librería `MASS` (el paquete `MASS` lo hemos activado junto a la

propia función `loglm()` ya que, de hacerlo con la función `library()`, se crea un conflicto con la función `select()` del paquete `dplyr`):

```
## Modelo de independencia

simplemod <- MASS::loglm(~ AUTOFINA + DIMENSION + FJUR, data=
tab.originales2)
simplemod
```

La primera línea estima el modelo y lo guarda en el objeto “simplemod”. La segunda línea muestra los resultados de la estimación:

```
Call:
MASS::loglm(formula = ~AUTOFINA + DIMENSION + FJUR, data =
tab.originales2)
```

```
Statistics:
                X^2 df      P(> X^2)
Likelihood Ratio 59.48973 12 2.796012e-08
Pearson          66.55540 12 1.404316e-09
```

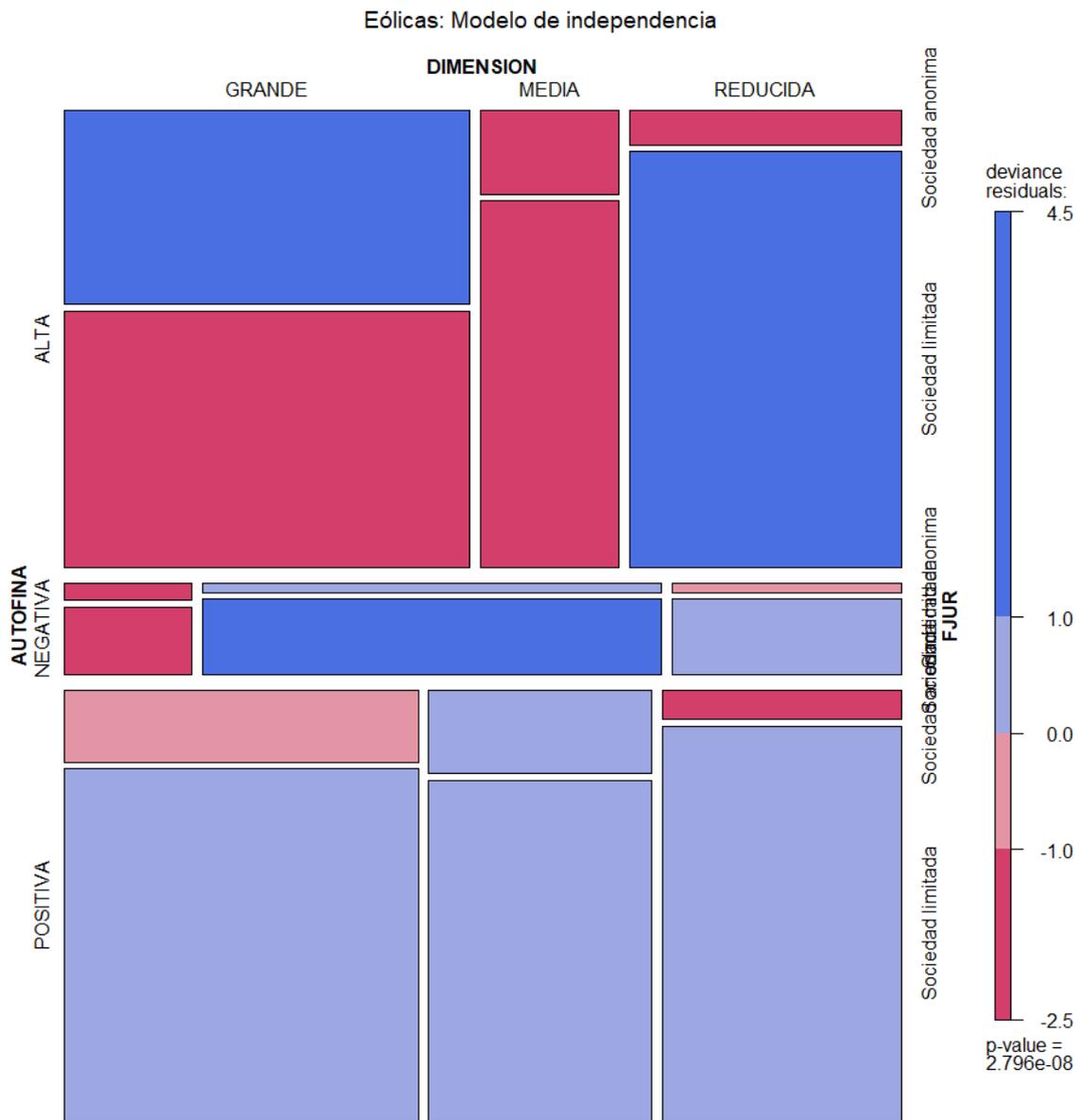
En esta información se muestran dos pruebas sobre la idoneidad del modelo estimado que, al ser este el modelo de independencia, sirven a su vez como contrastes de independencia. Aunque los resultados son similares, nos vamos a fijar en el contraste basado en el valor de la **ratio de verosimilitud** (*Likelihood Ratio*). Para que la especificación sea correcta (es decir, que pueda admitirse independencia entre los atributos o factores), se requiere una ratio pequeña (relativamente próxima a 0), lo que implica el no-rechazo de la hipótesis nula de especificación adecuada, lo que implica a su vez un ***p-valor superior a 0,05***. En este caso, en cambio, **se rechaza la hipótesis nula** (*p-valor inferior a 0,05*) y no se podría admitir independencia entre los factores o atributos, lo que lleva a sugerir la **existencia de asociación**, al menos entre algunos de los factores o atributos que entran en el modelo.

Además, se puede hacer un gráfico de mosaico con la siguiente línea de código. Cuanto más intensos son los colores, mayores serán los residuos (en valor absoluto) y, por lo tanto, peor será el ajuste obtenido, lo que se deberá a que se ha considerado (erróneamente) que no hay interacción entre los factores (no asociación o independencia):

```
plot(simplemod, panel = mosaic,
     main="Eólicas: Modelo de independencia",
     residuals_type = c("deviance"),
     gp = shading_hcl,
     gp_args = list(interpolate = c(0, 1)),
     main_gp = gpar(fontsize = 14),
```

```
sub_gp = gpar(fontsize = 9)
```

El gráfico obtenido es:



Por otro lado, el elemento "simplemod\$param" contiene los parámetros estimados del modelo:

```
simplemod$param
```

Siendo estos:

```
$` (Intercept) `
[1] 2.478525
```

```
$AUTOFINA
      ALTA  NEGATIVA  POSITIVA
```

```
0.5538124 -1.0493559 0.4955435
```

```
$DIMENSION
```

```
GRANDE MEDIA REDUCIDA  
0.29670207 -0.25289694 -0.04380514
```

```
$FJUR
```

```
Sociedad anonima Sociedad limitada  
-0.6858478 0.6858478
```

El **modelo saturado** (que, en el caso de dos factores, es el que incluye el término de interacción entre estos), en la práctica, no es relevante porque, aunque explica al 100% las frecuencias observadas, no indica cuál de los niveles de los factores o atributos y sus interacciones son los más relevantes (modelo *redundante*). Para su estimación, simplemente se sustituyen los signos “+” del modelo anterior por los signos “*”. Guardando la estimación en el objeto “satmod”, las instrucciones son:

```
satmod <- MASS::loglm(~ AUTOFINA * DIMENSION * FJUR,  
data=tab.originales2)  
satmod  
plot(satmod, panel = mosaic,  
main="Eólicas: Modelo saturado",  
residuals_type = c("deviance"),  
gp = shading_hcl,  
gp_args = list(interpolate = c(0, 1)),  
main_gp = gpar(fontsize = 14),  
sub_gp = gpar(fontsize = 9))
```

Los resultados, en esta ocasión, son:

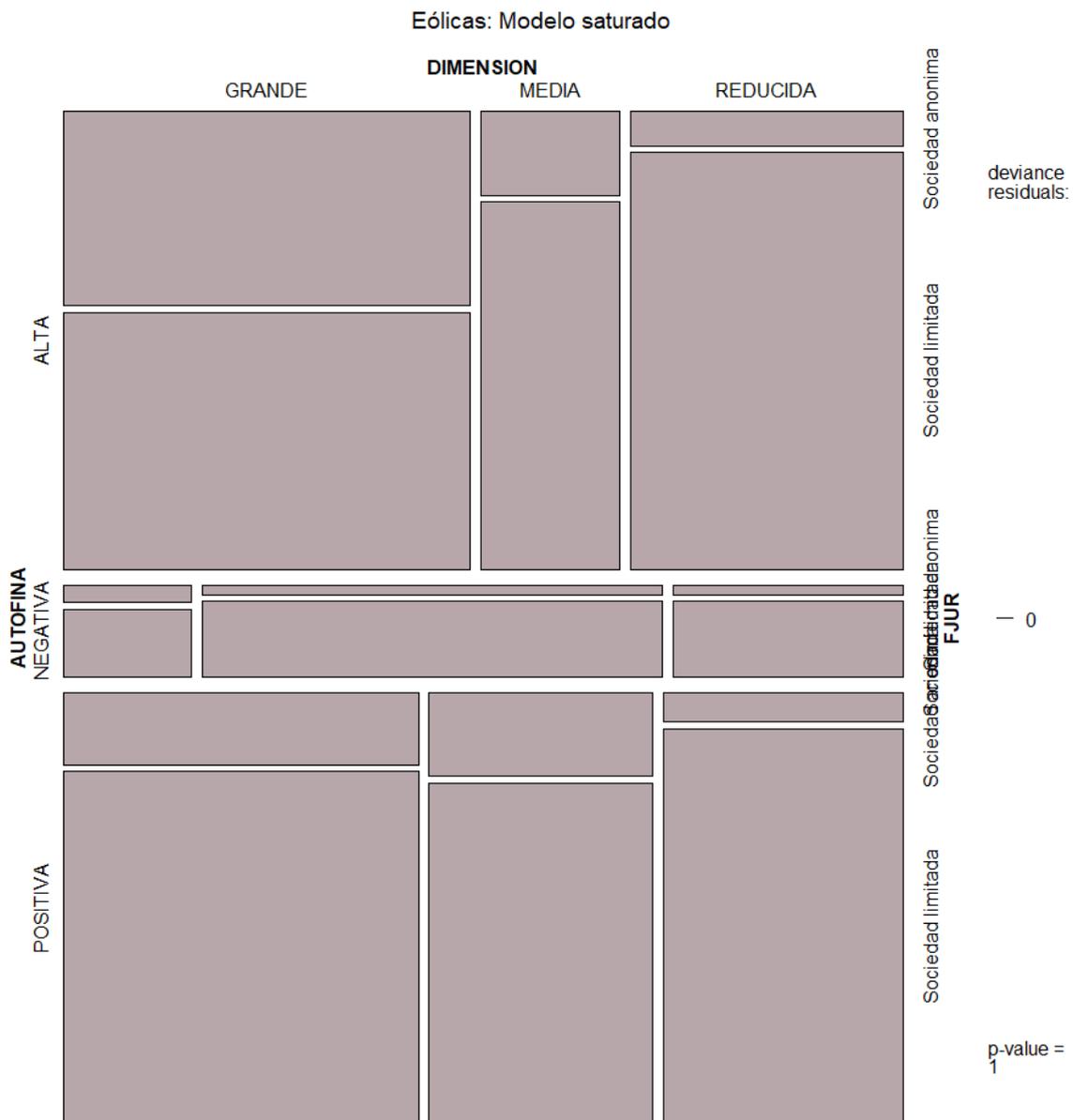
```
Call:  
MASS::loglm(formula = ~AUTOFINA * DIMENSION * FJUR, data =  
tab.originales2)
```

```
Statistics:
```

	X ²	df	P(> X ²)
Likelihood Ratio	0	0	1
Pearson	0	0	1

Obviamente, los *p-valores* de ambas pruebas de idoneidad de la especificación toman valor 1, ya que el modelo saturado ajusta perfectamente las frecuencias de la tabla multidimensional de la muestra. A pesar de ello, como se ha señalado, **este modelo no es útil** a efectos prácticos, porque se requiere un modelo que solamente recoja los efectos más importantes.

En cuanto al gráfico de mosaico de los residuos, estos son, para todas las frecuencias conjuntas 0, dado que el ajuste es perfecto:



Elección del modelo final.

Queda por encontrar la **especificación que proporcione el modelo que mejor representa la realidad recogiendo solo las interacciones más importantes** (es decir, evitando la saturación del modelo).

Una opción es emplear métodos automatizados para que, una vez se tiene el modelo saturado estimado, se obtenga una especificación más sencilla (Principio de Parsimonia) sin una pérdida grande de validez del modelo. Por ejemplo, un método es el **step / backward**, que, en función del *Criterio de Información de Akaike (AIC)*, irá probando a estimar **especificaciones más**

simples que disminuyan el AIC (lo que implica una mejor especificación).
En nuestro caso, se aplicará con el código:

```
## Elección del modelo final.  
  
defmod <- step(satmod, scale = 0,  
              direction = c("backward"),  
              trace = 1, steps = 1000)  
  
defmod
```

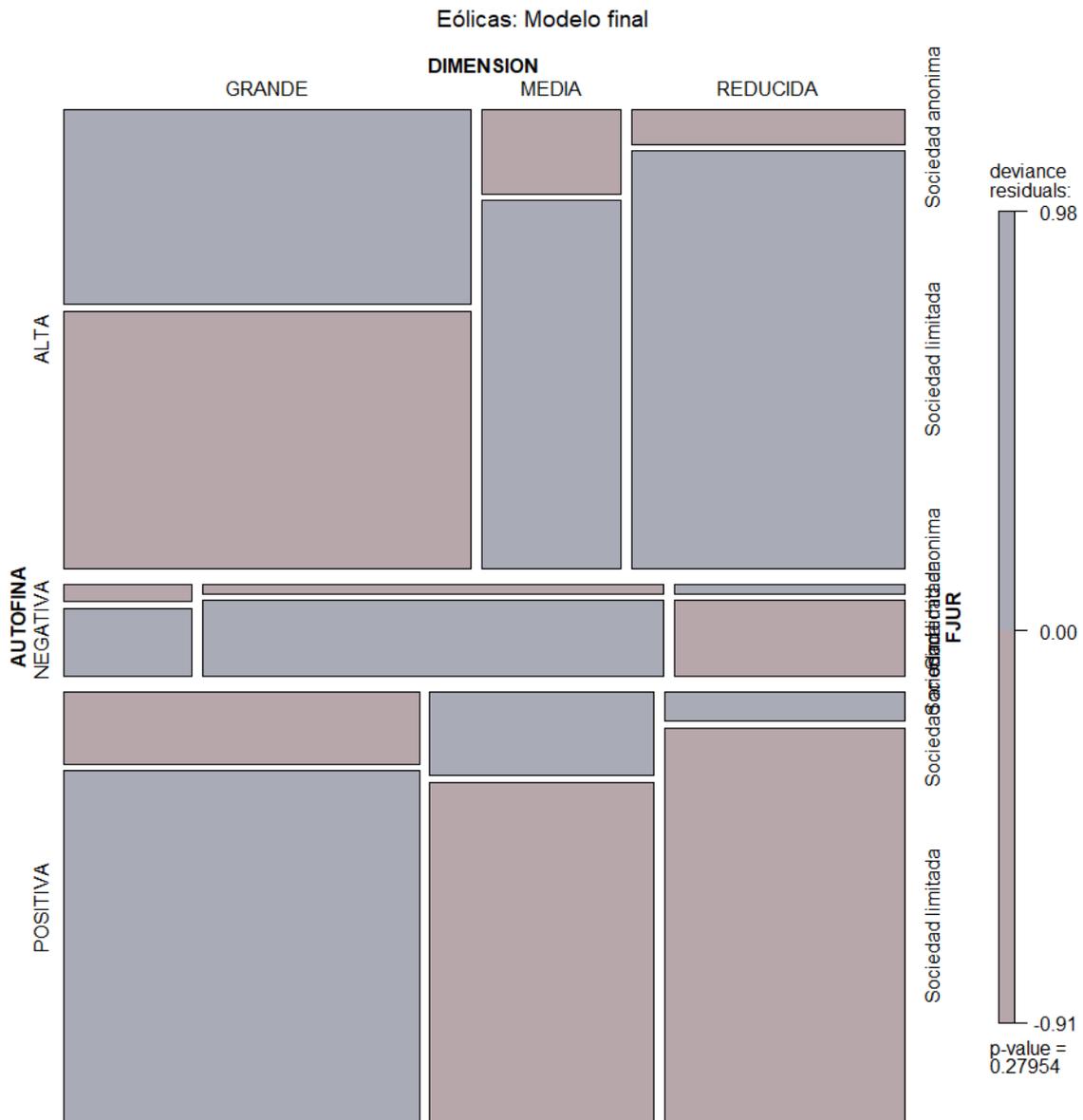
La solución final se almacena en el objeto “defmod”, y es:

```
Call:  
MASS::loglm(formula = ~AUTOFINA + DIMENSION + FJUR + AUTOFINA:DIMENSION  
+  
  AUTOFINA:FJUR + DIMENSION:FJUR, data = tab.originales2, evaluate =  
FALSE)  
  
Statistics:  
  
              X^2 df  P(> X^2)  
Likelihood Ratio 5.076507  4 0.2795371  
Pearson          5.277757  4 0.2599664
```

Se aprecia cómo esta solución contiene todos los efectos directos y las interacciones de dos en dos. Solo se diferencia del modelo saturado en que se suprime la interacción entre los tres atributos, de modo simultáneo. De este modo, **se confirma también la existencia de asociación** entre los tres atributos, considerados dos a dos.

Por otro lado, en las pruebas de validez de la especificación, se comprueba como se obtienen p-valores superiores a 0,05, lo que implica que el modelo es adecuado para representar la realidad. Así, trazando el gráfico de mosaico de los residuos en este caso, mediante el siguiente código:

```
plot(defmod, panel = mosaic,  
     main="Eólicas: Modelo final",  
     residuals_type = c("deviance"),  
     gp = shading_hcl,  
     gp_args = list(interpolate = c(0, 1)),  
     main_gp = gpar(fontsize = 14),  
     sub_gp = gpar(fontsize = 9))
```



Se observan frecuencias conjuntas con residuos o errores relativamente pequeños, lo que hace pensar en un buen ajuste del modelo.

Por último, los parámetros estimados del modelo se obtendrán mostrando el elemento "defmod\$param":

```
defmod$param
```

Y son:

```
$` (Intercept) `
[1] 2.254709
```

```
$AUTOFINA
      ALTA  NEGATIVA  POSITIVA
0.7111356 -1.1833867  0.4722512
```

```
$DIMENSION
      GRANDE      MEDIA  REDUCIDA
0.30340847  0.05598182 -0.35939029
```

```
$FJUR
Sociedad anonima Sociedad limitada
      -0.8804463      0.8804463
```

```
$AUTOFINA.DIMENSION
      DIMENSION
AUTOFINA      GRANDE      MEDIA  REDUCIDA
ALTA      0.3808516 -0.5510703  0.17021873
NEGATIVA -0.6439607  0.6931189 -0.04915817
POSITIVA  0.2631091 -0.1420485 -0.12106055
```

```
$AUTOFINA.FJUR
      FJUR
AUTOFINA  Sociedad anonima Sociedad limitada
ALTA      0.2669399      -0.2669399
NEGATIVA -0.1316088      0.1316088
POSITIVA -0.1353310      0.1353310
```

```
$DIMENSION.FJUR
      FJUR
DIMENSION  Sociedad anonima Sociedad limitada
GRANDE      0.38364100      -0.38364100
MEDIA      0.08851442      -0.08851442
REDUCIDA -0.47215542      0.47215542
```

Como ejemplo de interpretación, en los parámetros de la interacción, puede apreciarse como existe cierta asociación positiva entre el hecho de ser una sociedad anónima y pertenecer a un grupo empresarial mediano o grande; y de ser una sociedad limitada y tener una dimensión del grupo empresarial reducida. También, como ya se comentó, la capacidad de autofinanciación viene ligada a pertenecer a un grupo empresarial de dimensión grande, y a ser una Sociedad Anónima.

This work © 2022 by [Miguel Ángel Tarancón](#) and [Consolación Quintana](#) is licensed under [Attribution-NonCommercial-NoDerivatives 4.0 International](#) 

Updated: 13/12/2022