FEIR 50: Contrastes no paramétricos

Apuntes del curso FEIR3, curso 2014/15 actualizados. `Última actualización:` lunes 06 febrero 2017, 11:20:10

Álvaro Hernández Vicente

Índice

1. Contrastes paramétricos frente a no paramétricos

Una de las primeras dificultades con las que uno se encuentra al hacer estadística es decidir correctamente qué prueba utilizar en un caso concreto. Esta decisión depende del tipo de datos que estemos manejando (y las suposiciones que podemos hacer) y de lo que queramos contrastar.

Debido a la gran importancia que tienen las suposiciones sobre la distribución (normalidad, simetría, etc.) los contrastes se pueden dividir entre **paramétricos** y **no paramétricos**. La estadística **paramétrica** es aquella en la que se tiene (o se asume) cierta información sobre la distribución de probabilidad de la población y se quiere decidir sobre los parámetros que la definen. Por ejemplo, la prueba t supone que la población es normal y se decide sobre la media.

Los casos en los que no se tenga información sobre la distribución (o no se pueda asumir) son los que trata la estadística **no paramétrica**. También se suele denominar *libre de distribución* (debido a que algunos autores consideran una definición más amplia de estadística no paramétrica).

Se deben emplear pruebas no paramétricas en lugar de paramétricas cuando:

- los datos se puedan ordenar de alguna manera (ordinales) pero no haya normalidad. Por ejemplo, datos provenientes de escalas Likert donde *"1"* significa *"Muy en desacuerdo"* y *"10" "Muy de acuerdo"*;
- los datos son categóricos o nominales, esto es, se pueden distinguir en categorías. Por ejemplo, datos de personas según sexo o raza;
- los datos son numéricos pero no provienen de una normal. En este caso se podría intentar emplear transformaciones de datos para lograr normalidad.

Muchos de los métodos paramétricos, por suerte, funcionan bien cuando la normalidad solo se puede suponer aproximadamente. Por otro lado, los no paramétricos se ha visto que son casi tan capaces de detectar diferencias entre poblaciones como los paramétricos cuando se cumplen todos los supuestos, y cuando no se cumplen a menudo son más potentes en hacerlo. Por esa razón, algunos estadísticos prefieren utilizar contrastes no paramétricos frente a sus análogos paramétricos (ver Wackerly, Mendenhall, & Scheaffer, 2008).

Un resumen de las pruebas que veremos en este tema junto con sus alternativas paramétricas se recogen en la siguiente tabla:

Cuadro 1: Pruebas paramétricas y no paramétricas

Comparar \ Tipo de datos	Paramétricos	Ordinales	Categóricos
Dos grupos independientes	t independiente	Mann-Whitney	Exacto de Fisher
Dos grupos dependientes	t dependiente	Wilcoxon	McNemar
Dos o más grupos independientes	ANOVA de una vía	Kruskal-Wallis	Chi-cuadrado
Dos o más grupos dependientes	ANOVA medidas repetidas	Friedman	Q de Cochran

Nota: En el enlace Motulsky (2012) se puede encontrar más información y una tabla más amplia que la anterior.

2. Datos ordinales

En este apartado nos centramos en las diferentes pruebas que podemos utilizar cuando los datos de los que disponemos se pueden ordenar de alguna manera. Por ejemplo, las ya comentadas escalas Likert.

Las pruebas que veremos en este apartado se basan en el cálculo de los números de orden o rangos (del inglés *ranking*). Los números de orden de un vector son los valores que resultan de ordenarlos de menor a mayor y asignar a cada elemento su posición. Por ejemplo, si tenemos un vector con cinco valores el vector de números de orden asigna al menor de ellos un "1" y al mayor un "5". Veamos un ejemplo sencillo de cómo se calcula el vector de números de orden.

```
vector       <- c( 14, 10, 13, 11, 12, 17, 19 )
rangosVector <- rank( vector ) # números de orden
rbind( vector, rangosVector ) # mostrar los dos vectores por filas
```

```
##              [,1] [,2] [,3] [,4] [,5] [,6] [,7]
## vector         14   10   13   11   12   17   19
## rangosVector    5    1    4    2    3    6    7
```

Como podemos ver, el primer valor del vector es "14". El primer número de orden (correspondiente al "14") indica la posición que le correspondería a ese valor si se ordenara el vector de menor a mayor. El "14" es el quinto valor más pequeño del vector (más pequeños son 10, 11, 12 y 13), por lo que si se ordenara quedaría en la posición "5". Así, el primer número de orden es "5".

¿Y si hubieran dos valores "14"? Al ordenar nos quedarían en las posiciones "5" y "6", ¿qué números de orden se les asignan? La función `rank()` por defecto les asigna a cada uno el valor medio "5.5", pero con el argumento `ties.method` se puede cambiar.

```
vector2       <- c( 14, 10, 13, 11, 12, 14, 19 )
rangosVector2 <- rank( vector2, ties.method = "average" )
rbind( vector2, rangosVector2 )
```

```
##               [,1] [,2] [,3] [,4] [,5] [,6] [,7]
## vector2       14.0   10   13   11   12 14.0   19
## rangosVector2  5.5    1    4    2    3  5.5    7
```

2.1. Comparación entre dos grupos

Pasamos a ver algunas pruebas no paramétricas para comparar las distribuciones de dos grupos.

2.1.1. Prueba *U* de Mann-Whitney (suma de rangos de Wilcoxon)

La **prueba *U* de Mann-Whitney**, conocida simplemente como *U-test* y equivalente a la *prueba de los rangos sumados de Wilcoxon*, es la prueba no paramétrica alternativa al *t*-test para muestras independientes (*unpaired* en inglés).

Se utiliza cuando los datos son ordinales (que tienen un orden) pero no se puede asumir normalidad. Así, cuando tenemos muestras pequeñas se prefiere el *U*-test antes que el *t*-test. Al contrario de lo que mucha gente cree, y se encuentra erróneamente en bastantes artículos, el *U*-test requiere de homogeneidad de varianzas (homocedasticidad). Y otro detalle es que aunque el *U*-test se considera el equivalente no paramétrico al *t*-test, este compara medianas y no medias.

Este test, al utilizar números de orden o rangos, también es preferible cuando existen valores atípicos.

2.1.1.1. ¿Cómo funciona la prueba *U* de Mann-Whitney?

Supongamos que tenemos un par de muestras y queremos contrastar si hay diferencias entre las poblaciones de las que provienen.

```
muestraA <- c( 1.1, 3.4, 4.3, 2.1, 7.0 , 2.5 )
muestraB <- c( 7.0, 8.0, 3.0, 5.0, 6.2 , 4.4 )
```

La idea que hay detrás del *U*-test es juntar las dos muestras y asignar a cada valor un número de orden, esto es, al valor más pequeño se le asigna un "1", al segundo más pequeño un "2", y así hasta el más grande (en este caso, el valor más grande es "8.0" y, por tanto, se le asigna un "12"). Si hubiera valores repetidos, llamadas ligaduras (o *ties* en inglés), se les asigna la media (en este caso, el valor "7.0" está repetido y en lugar de asignarles "10" y "11" se les asigna "10.5" a cada uno).

```
muestraTotal  <- c( muestraA, muestraB ) # unión de las muestras
rangosMuestra <- rank( muestraTotal )
muestraTotal  <- cbind( muestraTotal, rangosMuestra )
muestraTotal
```

```
##       muestraTotal rangosMuestra
##  [1,]          1.1           1.0
##  [2,]          3.4           5.0
##  [3,]          4.3           6.0
##  [4,]          2.1           2.0
##  [5,]          7.0          10.5
##  [6,]          2.5           3.0
##  [7,]          7.0          10.5
##  [8,]          8.0          12.0
##  [9,]          3.0           4.0
## [10,]          5.0           8.0
## [11,]          6.2           9.0
## [12,]          4.4           7.0
```

Después se suman los números de orden para cada muestra (el mínimo de estos dos valores, que llamamos R_1 y R_2, es lo que se conoce como **suma de rangos de Wilcoxon**). El estadístico U estará definido como

$$U_1 = R_1 - \frac{n_1(n_1+1)}{2}$$

$$U_2 = R_2 - \frac{n_2(n_2+1)}{2}$$
$$U = \text{mín}(U_1, U_2)$$

donde n_1 y n_2 son los tamaños de cada muestra.

```
sumaRangosA <- sum( rangosMuestra[ 1:6 ] )
sumaRangosB <- sum( rangosMuestra[ 6:12 ] )
n1 <- length( muestraA )
n2 <- length( muestraB )
U1 <- sumaRangosA - n1 * ( n1 + 1 ) / 2
U2 <- sumaRangosB - n2 * ( n2 + 1 ) / 2
c( U1, U2 )
```

```
## [1]  6.5 32.5
```

En la práctica, cuando aparecen ligaduras (valores repetidos) no se calcula el valor exacto del estadístico U; se calcula una aproximación.

2.1.1.2. En R

Para hacer todos estos cálculos con R podemos utilizar la función `wilcox.test()`.

```
wilcox.test( muestraA, muestraB )
```

```
## Warning in wilcox.test.default(muestraA, muestraB): cannot compute exact p-
## value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  muestraA and muestraB
## W = 6.5, p-value = 0.07765
## alternative hypothesis: true location shift is not equal to 0
```

Nota: El estadístico U en R se denomina W. Por defecto en la función se tiene `paired = FALSE`. Nos aparece un mensaje de aviso (que no de error) debido a que las ligaduras no permiten calcular el p-valor exacto (se usa una aproximación a la normal).

El estadístico U se aproxima a una distribución normal (cuando los tamaños muestrales son mayores a 20 se considera muy buena) y, por tanto, se puede tipificar a una normal Z (hay fórmulas con las que se pueden calcular la media y la desviación típica). En R podemos emplear la función `wilcox_test()` del paquete `coin` (no preinstalado por defecto) que requerirá un fórmula como argumento (no la pareja de muestras). Es la opción recomendada si hay ligaduras en los datos.

```
## install.packages( "coin" )
library( "coin" )
grupo <- factor( c( rep( "A", length(muestraA) ),
                    rep( "B", length(muestraB) ) ) )
muestraTotal <- c( muestraA, muestraB )
wilcox_test( muestraTotal ~ grupo, distribution = "exact" )
```

```
##
##  Exact Wilcoxon-Mann-Whitney Test
##
## data:  muestraTotal by grupo (A, B)
## Z = -1.8447, p-value = 0.06926
## alternative hypothesis: true mu is not equal to 0
```

2.1.1.3. Tamaño del efecto

El tamaño del efecto para un U-test se calcula a partir de la Z que nos devuelve la función `wilcox_test()` con la fórmula

$$r = \frac{Z}{\sqrt{N}}$$

donde N es el tamaño total de la muestra ($N = n_1 + n_2$). El signo no importa, así que se comunica el valor absoluto de r.

Cuadro 2: Tamaño del efecto

	Pequeño	Mediano	Grande
abs(r)	0.1	0.3	0.5

En nuestro caso anterior

```
N            <- length( muestraA ) + length ( muestraB ) # tamaño muestral
wilcoxTest   <- wilcox_test( muestraTotal ~ grupo, distribution = "exact" )
tamañoEfecto <- statistic( wilcoxTest ) / sqrt( N )
tamañoEfecto
```

```
##          A
## -0.5325195
```

2.1.1.4. Ejemplo

Supongamos que queremos saber las diferencias de opinión entre jóvenes y adultos sobre una cierta ley. Para ello se ha pasado una encuesta en la que se ha puntuado de 1 a 10 el grado de acuerdo (siendo "1" *"Muy en desacuerdo"* y *"10" "Muy de acuerdo"*). ¿La respuesta de los jóvenes es más negativa que la de los adultos?

```
dfEncuesta <- read.table( "files/feir50A-encuesta.csv" , header = TRUE, sep = ";" )
head( dfEncuesta )
```

```
##   grupo opinion
## 1 joven       3
## 2 joven       4
## 3 joven       5
## 4 joven       4
## 5 joven       8
## 6 joven       4
```

```
wilcox.test( dfEncuesta$opinion ~ dfEncuesta$grupo, alternative = "greater" )
```

```
## Warning in wilcox.test.default(x = c(9L, 10L, 3L, 5L, 9L, 4L, 6L, 7L, 5L, :
## cannot compute exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  dfEncuesta$opinion by dfEncuesta$grupo
## W = 78, p-value = 0.01796
## alternative hypothesis: true location shift is greater than 0
```

Como el p-valor es menor a 0.05 rechazamos la hipótesis nula a favor de la alternativa de que los adultos están más de acuerdo con la ley.

```
wilcoxTest <- wilcox_test( dfEncuesta$opinion ~ dfEncuesta$grupo, alternative = "greater",
                           distribution = "exact" )
N          <- nrow( dfEncuesta ) # tamaño muestral
statistic( wilcoxTest ) / sqrt( N ) # tamaño del efecto
```

```
##    adulto
## 0.4776153
```

En valor absoluto obtenemos un tamaño del efecto de 0.4776, cercano a considerarse *grande*.

Nota: En este ejemplo solo hemos visto cómo aplicar la prueba U. Para que fuese completo se debería contrastar primero si cumple el requisito de homogeneidad de varianzas con, por ejemplo, el test de Levene.

2.1.2. Prueba de los rangos con signo de Wilcoxon

La **prueba de los rangos con signo de Wilcoxon** (*Wilcoxon Signed-rank test* en inglés), también conocida como test de Wilcoxon, es la versión no paramétrica del t-test para muestras dependientes. Igual que la prueba U de Mann-Whitney este test compara medianas en lugar de medias.

2.1.2.1. ¿Cómo funciona el test de los rangos con signo de Wilcoxon?

Cuando tenemos muestras dependientes lo primero que se hace habitualmente es calcular sus diferencias. La idea de esta prueba es guardar los signos de esas diferencias y, con los valores absolutos, asignar números de orden como en la prueba de Mann-Whitney (quitando los de diferencia nula).

```
grupoAntes       <- c( 2, 4, 6, 1, 3 )
grupoDespues     <- c( 5, 2, 7, 1, 6 )
grupoDiferencia  <- grupoAntes - grupoDespues
rangosDiferencia <- rank( abs( grupoDiferencia[ grupoDiferencia != 0 ] ) )
## el resultado de los números de orden es c( 3.5, 2.0, 1.0, 3.5 )
## añadimos un 0 a mano para cuadrar la tabla
rangosDiferencia <- c( 3.5, 2.0, 1.0, 0, 3.5 )
dfGrupo <- data.frame( grupoAntes, grupoDespues, sign( grupoDiferencia ),
                       abs( grupoDiferencia ), rangosDiferencia )
names(dfGrupo ) <- c( "antes", "despues", "signo", "diferencia", "rangos" )
dfGrupo
```

```
##   antes despues signo diferencia rangos
## 1     2       5    -1          3    3.5
## 2     4       2     1          2    2.0
## 3     6       7    -1          1    1.0
## 4     1       1     0          0    0.0
## 5     3       6    -1          3    3.5
```

Una vez hecho esto hay que sumar, por un lado, los números de orden que provienen de diferencias positivas y, por otro lado, los de diferencias negativas (los de diferencias nulas no se suman). El mínimo de estos dos valores es el valor del estadístico W de Wilcoxon.

$$W_+ = \sum_{\text{signo}>0} \text{rango}(i)$$

$$W_- = \sum_{\text{signo}<0} \text{rango}(i)$$

$$W = \text{mín}(W_+, W_-)$$

donde rango(i) es el número de orden que le corresponde a la observación i.

```
sumaPositivos <- sum( dfGrupo[ dfGrupo$signo ==  1, ]$rangos )
sumaNegativos <- sum( dfGrupo[ dfGrupo$signo == -1, ]$rangos )
c( sumaPositivos, sumaNegativos )
```

```
## [1] 2 8
```

2.1.2.2. En R

Para hacer estos cálculos con R podemos utilizar la función `wilcox.test()` con el argumento `paired = TRUE`.

```
wilcox.test( grupoAntes, grupoDespues, paired = TRUE )
```

```
## Warning in wilcox.test.default(grupoAntes, grupoDespues, paired = TRUE):
## cannot compute exact p-value with ties
```

```
## Warning in wilcox.test.default(grupoAntes, grupoDespues, paired = TRUE):
## cannot compute exact p-value with zeroes
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  grupoAntes and grupoDespues
## V = 2, p-value = 0.3573
## alternative hypothesis: true location shift is not equal to 0
```

Nota: El estadístico se denomina V en lugar de W. Cuando hay ceros o ligaduras se calcula un p-valor aproximado.

Este estadístico W (denominado V en R) también se puede tipificar a un valor Z (hay fórmulas para la media y la varianza) que será útil para calcular posteriormente el tamaño del efecto. En R se puede utilizar la función `wilcoxsign_test()` del paquete `coin`. Es el recomendado cuando hay ligaduras o ceros.

```
## install.packages( "coin" )
library( "coin" )
wilcoxsign_test( grupoAntes ~ grupoDespues, distribution = "exact" )
```

```
##
##  Exact Wilcoxon-Pratt Signed-Rank Test
##
## data:  y by x (pos, neg)
##   stratified by block
## Z = -1.0937, p-value = 0.375
## alternative hypothesis: true mu is not equal to 0
```

2.1.2.3. Tamaño del efecto

El tamaño del efecto para un test de Wilcoxon se calcula y considera de manera similar al del test de Mann-Whitney

$$r = \frac{Z}{\sqrt{N}}$$

donde N es el tamaño total de la muestra (aunque hayan sido observaciones sobre los mismos sujetos).

En nuestro caso anterior

```
N              <- length( grupoAntes ) + length ( grupoDespues ) # tamaño muestral
wilcoxSignTest <- wilcoxsign_test( grupoAntes ~ grupoDespues, distribution = "exact" )
statistic( wilcoxSignTest ) / sqrt( N ) # tamaño del efecto
```

```
##        pos
## -0.3458702
```

2.1.3. Otros contrastes

2.1.3.1. Test de Kolmogorov-Smirnov para dos muestras

Sirve también para comparar dos muestras independientes, es decir, es una alternativa al U-test. En este caso, sin embargo, se comparan distribuciones de probabilidad en general (posición, forma, etc.); por tanto, se podría dar el caso que diera resultados distintos a la prueba U. Ver `?ks.test()`.

Algunos test del siguiente apartado también se pueden utilizar para comparar dos grupos.

2.2. Comparación entre más de dos grupos

En este apartado pasamos a ver las pruebas no paramétricas para contrastar si hay diferencias entre más de dos grupos. Además, para los casos en los que se obtengan resultados significativos veremos cómo realizar análisis post-hoc controlando la significación.

2.2.1. Prueba *H* de Kruskal-Wallis

La **prueba *H* de Kruskal-Wallis** es la alternativa no paramétrica al modelo ANOVA de una vía para comparar más de dos grupos independientes. Cuando este test nos devuelve resultados significativos quiere decir que, en al menos dos grupos, hay diferencias pero no sabemos en cuáles de ellos (ni cuántas hay). Para saber qué grupos difieren entre sí se utilizan pruebas post-hoc (comparaciones dos a dos controlando la significación).

Al igual que la prueba U, bajo la hipótesis nula se asume que los datos provienen de la misma distribución. Esto quiere decir que es necesaria la homogeneidad de varianzas.

2.2.1.1. ¿Cómo funciona la prueba *H* de Kruskal-Wallis?

Esta prueba, al igual que las dos vistas en el apartado anterior (para comparar dos grupos), utiliza la suma de rangos de Wilcoxon. En este caso, como comparamos más de dos grupos (digamos k grupos) llamaremos R_k a la suma de números de orden de cada grupo.

Este test lo que hace es calcular el estadístico H a partir de la fórmula

$$H = \frac{12}{N(N-1)} \sum_{i=1}^{k} \left(\frac{R_i^2}{n_i} \right) - 3(N+1)$$

donde N es el tamaño total de la muestra y n_i el tamaño de la muestra de cada grupo. O bien, en el caso de que hubieran ligaduras (*ties*, valores repetidos) la fórmula se corrige dividiendo la anterior

$$H = \frac{\frac{12}{N(N-1)} \sum_{i=1}^{k} \left(\frac{R_i^2}{n_i} \right) - 3(N+1)}{1 - \frac{\sum_{i=1}^{g} (t_i^3 - t_i)}{N^3 - N}}$$

donde g denota el número de grupos de ligaduras y t_i el total de números de orden ligados en el i-ésimo grupo.

Si el tamaño muestral es mayor a 5, el estadístico H sigue una distribución χ^2 con $k-1$ grados de libertad.

2.2.1.2. En R

Podemos utilizar la función `kruskal.test()`. Veamos un ejemplo de cómo se usa.

2.2.1.3. Ejemplo

Supongamos que queremos ver si hay diferencias en el número de piropos que recibe el profesorado de matemáticas que da clases en la facultad de educación (E), con el de matemáticas (M) y con el de biología (B). (Datos simulados).

```
piropos   <- c( 6, 1, 7, 4, 8, 10, 5, 0, 2, 1, 2, 4, 4, 1, 4, 2, 3, 1, 5, 9)
facultad  <- factor( c( rep( "E", 7 ), rep( "M", 7 ), rep( "B", 6 ) ) )
dfPiropos <- data.frame( piropos, facultad )
head( dfPiropos )
```

```
##   piropos facultad
## 1       6        E
## 2       1        E
## 3       7        E
## 4       4        E
## 5       8        E
## 6      10        E
```

```
kruskal.test( piropos ~ facultad, data = dfPiropos )
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  piropos by facultad
## Kruskal-Wallis chi-squared = 6.4222, df = 2, p-value = 0.04031
```

Nota: El estadístico H en R se denomina *Kruskal-Wallis chi-squared*.

Al hacer la prueba con R obtenemos un p-valor de 0.0403 (menor a 0.05) luego rechazamos la hipótesis nula de que no hay diferencias en el número de piropos recibidos según la facultad.

2.2.1.4. Post-hoc

En la prueba de Kruskal-Wallis la hipótesis alternativa es que no todos los grupos tienen la misma distribución, esto es, que en al menos dos grupos hay diferencias. Para saber entre qué par de grupos se han encontrado diferencias realizamos un análisis post-hoc.

Se puede abordar de dos formas:

Primera opción: Hacer la prueba de Mann-Whitney sobre cada par de grupos. Como ya sabemos que esto aumenta el error de tipo I, después hacemos alguna corrección de la significación (Bonferroni u otra). Se puede hacer utilizando la función `pairwise.wilcox.test()`.

Recordemos que la corrección de Bonferroni es bastante estricta cuando el número de grupos es grande (más de 6). Para ver las posibles correcciones que permite esta función (y cuándo utilizarlas) podemos ver la ayuda de R con `?p.adjust`.

```
pairwise.wilcox.test( piropos, facultad, p.adjust = "bonferroni", exact = FALSE )
```

```
##
##  Pairwise comparisons using Wilcoxon rank sum test
##
```

```
## data:  piropos and facultad
##
##   B     E
## E 0.753 -
## M 0.503 0.061
##
## P value adjustment method: bonferroni
```

Hacer la prueba de Mann-Whitney sobre todas las parejas dispara el error de tipo I bastante rápido, por eso algunos autores recomiendan seleccionar primero los pares de grupos que más nos interese comparar y, de esta forma, hacer menos comparaciones.

Segunda opción: También podemos comparar cada pareja con Mann-Whitney, pero esta vez utilizando una desigualdad que se obtiene aplicando el test de Tukey (*Tukey's range test* en inglés). La fórmula que se utiliza es

$$|\bar{R}_u - \bar{R}_v| \geq z_{\alpha/k(k-1)} \sqrt{\frac{N(N+1)}{12}\left(\frac{1}{n_u} + \frac{1}{n_v}\right)}$$

donde $\bar{R}_u$ es la media de números de orden del grupo u, n_u el tamaño muestral del grupo u y en lo demás se repite notación. Esta es la alternativa que aparece explicada en A. Field, Miles, & Field (2012) y Gibbons & Chakraborti (2003). Podemos utilizar la función `kruskalmc()` del paquete `pgirmess`.

```
## install.packages( "pgirmess" )
library( "pgirmess" )
kruskalmc( piropos ~ facultad, data = dfPiropos )
```

```
## Multiple comparison test after Kruskal-Wallis
## p.value: 0.05
## Comparisons
##      obs.dif critical.dif difference
## B-E 3.607143     7.879552      FALSE
## B-M 4.321429     7.879552      FALSE
## E-M 7.928571     7.570429       TRUE
```

Esta función nos devuelve los dos valores de la desigualdad para cada par de grupos y una tercera columna indicando si se cumple (cuando lo hace es cuando se han encontrado diferencias).

Nota: La instalación del paquete `pgirmess` en GNU/Linux puede dar algún problema. En este caso habría que instalar las librerías `proj-dev` y `libgdal1`. Ver detalles en Giraudoux (2014).

2.2.1.5. Tamaño del efecto

Cuando se hace una prueba de Kruskal-Wallis no hay una manera general para calcular el tamaño del efecto. Lo que se suele hacer es calcular el tamaño del efecto sobre las parejas con diferencias significativas del análisis post-hoc. Es decir, se calculan como ya vimos en el apartado de la prueba de Mann-Whitney sobre cada pareja.

En este caso, solo se han encontrado diferencias entre las facultades de educación (E) y matemáticas (M).

```
library( "coin" )
wilcoxTestEM <- wilcox_test( piropos ~ factor( facultad ), distribution = "exact",
                             data = dfPiropos[ facultad == "E" | facultad=="M", ] )
N <- nrow( dfPiropos[ facultad == "E" | facultad == "M", ] ) # tamaño muestral
statistic( wilcoxTestEM ) / sqrt( N ) # tamaño del efecto
```

```
##         E
## 0.6381062
```

2.2.2. Prueba de Friedman

La **prueba de Friedman** (también conocida como **ANOVA de Friedman**) es la alternativa no paramétrica al modelo ANOVA de un vía con medidas repetidas para comparar más de dos grupos dependientes. Igual que en la prueba de Kruskal-Wallis para saber qué grupos difieren (en el caso de resultados significativos del test) también se utilizan pruebas post-hoc.

2.2.2.1. ¿Cómo funciona la prueba de Friedman?

Se vuelven a utilizar los rangos de Wilcoxon, pero de una manera un poco diferente. Supongamos que en lugar de tener un par de muestras "antes" y "después" (como en la prueba de rangos con signo de Wilcoxon) tenemos más de dos (por ejemplo, algún seguimiento más prolongado sobre el tiempo) y queremos saber si hay diferencias entre algunas de ellas.

Lo que se hace es calcular los números de orden sobre cada individuo, por ejemplo, si hubieran tres grupos (`ayer`, `hoy` y `mañana`) se les asignarían los valores del uno al tres. Así, después se sumarían estos números de orden sobre cada grupo obteniéndose los R_i (se estarían sumando los unos, doses y treses de cada individuo).

Una vez que la suma de números de orden está calculada sobre cada grupo se calcula el estadístico F_r con la fórmula

$$F_r = \left(\frac{12}{Nk(k+1)} \sum_{i=1}^{k} R_i^2 \right) - 3N(k+1)$$

donde N es el número de individuos (no de tamaño de la muestra, ya que, los valores son apareados) y k es el número de grupos. Cuando hay ligaduras se utiliza la fórmula con corrección

$$F_r = \frac{12 \sum_{i=1}^{k} R_i^2 - 3N^2 k(k+1)^2}{Nk(k^2-1) - \sum \sum t(t^2-1)}$$

donde la doble suma indica que se sumas las ligaduras repetidas t sobre cada uno de los individuos.

Cuando el número de individuos (N) es mayor a 10, igual que en la prueba H, el estadístico F_r sigue una distribución χ^2 con $k-1$ grados de libertad.

2.2.2.2. En R

Podemos utilizar la función `friedman.test()`. Veamos un ejemplo de cómo se usa.

2.2.2.3. Ejemplo

Ahora se desea probar si hay diferencias en el número de piropos recibidos por el profesorado de didáctica de las matemáticas en la facultad de educación según el día en que se recogen los datos: el de la presentación (*t1*), el anterior al examen (*t2*) y después de la evaluación (*t3*). (Datos simulados).

```
piroposEd   <- c( 9, 5, 2, 6, 3, 1, 5, 5, 7, 11, 5, 1, 8, 4, 3, 10, 4, 1, 7, 3, 4 )
dia         <- factor( rep( c( "t1", "t2", "t3" ), 7 ) )
profesor    <- factor( rep( 1:7, each = 3 ) )
dfPiroposEd <- data.frame( piroposEd, dia, profesor )
head( dfPiroposEd )
```

```
##   piroposEd dia profesor
## 1         9  t1        1
## 2         5  t2        1
## 3         2  t3        1
```

```
## 4         6  t1        2
## 5         3  t2        2
## 6         1  t3        2
```

```
## friedman.test ( piroposEd ~ dia | profesor )
friedman.test( piroposEd, dia, profesor )
```

```
##
##  Friedman rank sum test
##
## data:  piroposEd, dia and profesor
## Friedman chi-squared = 7.1852, df = 2, p-value = 0.02753
```

Nota: El estadístico F_r en R se denomina *Friedman chi-squared*.

Obtenemos un p-valor de 0.02753 (menor a 0.05) luego rechazamos la hipótesis nula de que no hay diferencias en el número de piropos recibidos según el día.

2.2.2.4. Post-hoc

Igual que con la prueba de Kruskal-Wallis para saber sobre qué grupos se han encontrado diferencias hacemos un análisis post-hoc. Y de nuevo, se puede abordar de dos formas (las análogas a Kruskal-Wallis):

Primera opción: Hacer la prueba de los rangos con signo de Wilcoxon sobre cada pareja. Se puede utilizar la función `pairwise.wilcox.test()` con el argumento `paired = TRUE` y la corrección que se quiera.

```
pairwise.wilcox.test( piroposEd, dia, p.adjust = "bonferroni", exact = FALSE,
                      paired = TRUE )
```

```
##
##  Pairwise comparisons using Wilcoxon signed rank test
##
## data:  piroposEd and dia
##
##    t1   t2
## t2 0.10 -
## t3 0.10 0.45
##
## P value adjustment method: bonferroni
```

Segunda opción: Igual que con Kruskal-Wallis se utiliza una desigualdad que se obtiene a partir del test de Tukey. En este caso

$$|\bar{R}_u - \bar{R}_v| \geq z_{\alpha/k(k-1)} \sqrt{\frac{Nk(k+1)}{6}}$$

donde se usa la misma notación que anteriormente. Una función para hacer esto con R es `friedmanmc()` del paquete `pgirmess`.

```
library( "pgirmess" )
friedmanmc( piroposEd, dia, profesor )
```

```
## Multiple comparisons between groups after Friedman test
## p.value: 0.05
## Comparisons
##       obs.dif critical.dif difference
## t1-t2     7.0     8.957452      FALSE
## t1-t3     9.5     8.957452       TRUE
## t2-t3     2.5     8.957452      FALSE
```

2.2.2.5. Tamaño del efecto

Ocurre lo mismo que con el tamaño del efecto para la prueba de Kruskal-Wallis. En este caso, se calculan de igual manera que para la prueba de rangos con signo de Wilcoxon sobre cada pareja.

En nuestro ejemplo solo se han encontrado diferencias entre el día de la presentación (*t1*) y el de la evaluación (*t3*).

```
dfPiroposEd13  <- dfPiroposEd[ dia == "t1" | dia == "t3", ]
wSignTest <- wilcoxsign_test( piroposEd ~ factor( dia ) | profesor,
                              distribution = "exact",
                              data = dfPiroposEd13 )
N     <- nrow( dfPiroposEd13 ) # tamaño muestral de t1 y t3
statistic( wSignTest ) / sqrt( N )
```

```
##       pos
## 0.5883317
```

2.2.3. Otros

2.2.3.1. Prueba de Jonckheere-Terpstra

Se utiliza en lugar de la de Kruskal-Wallis si existe una ordenación natural del factor (un orden ascendente o descendente que tenga más sentido). En ese caso, esta prueba es más potente que la de Kruskal-Wallis.

Por ejemplo, supongamos que queremos comparar el número de piropos recibidos por los profesores según la cantidad de asignaturas que imparten (de 1 a 3). En este caso, una hipótesis alternativa puede ser que cuantas más asignaturas dan los profesores más piropos reciben.

En R se puede utilizar la función `jonckheere.test()` del paquete `clinfun`. Con el argumento `alternative =` se puede modificar la hipótesis alternativa.

```
## install.packages( "clinfun" )
library( "clinfun" )
piropos2    <- c( 3, 1, 1, 4, 8, 10, 5, 0, 2, 4, 1, 2, 4, 4, 10, 12, 14, 7, 11, 9)
asignaturas <- c( rep( 1, 7 ), rep( 2, 7 ), rep( 3, 6 ) )
head( cbind( piropos2, asignaturas ) ) # mostramos los datos
```

```
##      piropos2 asignaturas
## [1,]        3           1
## [2,]        1           1
## [3,]        1           1
## [4,]        4           1
## [5,]        8           1
## [6,]       10           1
```

```
jonckheere.test( piropos2, asignaturas, alternative = "increasing" )
```

```
## Warning in jonckheere.test(piropos2, asignaturas, alternative = "increasing"): Sample size > 100 or c
##  p-value based on normal approximation. Specify nperm for permutation p-value
```

```
##
##  Jonckheere-Terpstra test
##
## data:
## JT = 96, p-value = 0.02047
## alternative hypothesis: increasing
```

Un texto recomendable sobre esta prueba se puede encontrar en A. Field et al. (2012).

2.2.3.2. Prueba Q de Cochran

Es una alternativa a la prueba de Friedman cuando las variables son dicotómicas. Se estudiará en el apartado siguiente *Datos categóricos*.

2.2.3.3. Nota

Las fórmulas que se utilizan en los contrastes anteriores tienen su interpretación y su significado. Si alguien tiene interés en ver cómo se obtienen puede ver Gibbons & Chakraborti (2003).

3. Datos nominales o categóricos

En el apartado anterior hemos visto métodos para comparar grupos cuando nuestros datos son de tipo ordinal (que tienen un orden) como, por ejemplo, las escalas Likert. En este apartado vamos a ver qué pruebas se pueden hacer cuando los datos son nominales (también conocidos como categóricos).

3.1. Introducción: tablas de contingencia

Supongamos que tenemos una muestra de personas en la que se indica para cada una si tiene problemas del corazón y si hace deporte.

```
deporte   <- c( "Sí", "No", "No", "Sí", "No" )
enfermo   <- c( "No", "No", "Sí", "No", "No" )
dfCorazon <- data.frame( deporte, enfermo )
dfCorazon
```

```
##   deporte enfermo
## 1      Sí      No
## 2      No      No
## 3      No      Sí
## 4      Sí      No
## 5      No      No
```

En estos casos, en lugar de trabajar con un `data.frame` con un par de columnas de "Sí" o "No" lo que se hace es trabajar con tablas de contingencia (que significa posibilidad de que algo suceda) o frecuencias donde se indica el número de casos según ambas categorías. Se pueden utilizar las funciones `xtabs()` o `table()`.

```
tCorazon <- table( deporte, enfermo )
## tCorazon <- xtabs( ~ deporte + enfermo, data = dfCorazon)
tCorazon
```

```
##        enfermo
## deporte No Sí
##      No  2  1
##      Sí  2  0
```

Como vemos, con esta tabla podemos observar, por ejemplo, que hay dos personas de la muestra que hacen deporte y no están enfermos del corazón. Con la función `addmargins()` podemos completar la tabla con las sumas por filas y columnas.

```
addmargins( tCorazon )
```

```
##        enfermo
## deporte No Sí Sum
##     No   2  1   3
##     Sí   2  0   2
```

```
##     Sum  4  1   5
```

Ahora podemos identificar fácilmente, por ejemplo, que de las cuatro personas que no están enfermas del corazón dos hacen deporte y dos no lo hacen.

Si queremos ver (en lugar del número de casos absolutos) la tabla de proporciones utilizamos la función `prop.table()`

```
tCorazonFrec <- prop.table( tCorazon )
tCorazonFrec
```

```
##        enfermo
## deporte  No  Sí
##      No 0.4 0.2
##      Sí 0.4 0.0
```

Y otra vez se pueden añadir las sumas por filas y columnas

```
addmargins( prop.table ( tCorazonFrec ) )
```

```
##        enfermo
## deporte  No  Sí Sum
##     No  0.4 0.2 0.6
##     Sí  0.4 0.0 0.4
##     Sum 0.8 0.2 1.0
```

Para que nos aparezcan porcentajes (si es que nos gustan más) bastaría con multiplicar por cien

```
addmargins( prop.table ( tCorazonFrec ) ) * 100
```

```
##        enfermo
## deporte  No  Sí Sum
##     No   40  20  60
##     Sí   40   0  40
##     Sum  80  20 100
```

Para ver más detalles sobre estas (y otras) funciones se puede consultar el libro de texto Kabacoff (2011).

Si hubieran más de dos categorías (por ejemplo: hacer deporte, enfermo del corazón y fumar) se utiliza la función `ftable()`.

```
fuma      <- c( "Sí", "Sí", "Sí", "No", "Sí")
dfCorazon <- data.frame ( deporte, enfermo, fuma )
## ftable ( xtabs( ~ fuma + deporte + enfermo , data = dfCorazon ) )
ftable ( table( fuma, deporte, enfermo ) )
```

```
##              enfermo No Sí
## fuma deporte
## No   No               0  0
##      Sí               1  0
## Sí   No               2  1
##      Sí               1  0
```

Nota: La función `table()` ignora los valores perdidos (NAs) por defecto. Para incluir NA como una categoría válida en el conteo de frecuencias hay que incluir el argumento `useNA = ''ifany''`.

3.2. Comparación entre dos o más proporciones/frecuencias

En este apartado vamos a ver las diferentes pruebas que podemos utilizar para comparar dos (o más) variables categóricas. Si no obtuviéramos resultados significativos diríamos que las variables no están relacionadas entre sí (hipótesis nula).

3.2.1. Prueba χ^2 de Pearson

La **prueba χ^2 de Pearson** (pronunciada **ji-cuadrado** y a veces como **chi-cuadrado**) nos permite contrastar si existe relación entre dos variables categóricas mediante la tabla de contingencia cuando tenemos datos no pareados.

3.2.1.1. ¿Cómo funciona la prueba χ^2 de Pearson?

La idea se basa en medir la desviación de los datos observados en la tabla de frecuencias con respecto a los datos esperados por azar. Esto se hace aplicando la fórmula

$$\chi^2 = \sum_{i,j} \frac{(\text{observado}_{ij} - \text{esperado}_{ij})^2}{\text{esperado}_{ij}}$$

donde i, j recorren las filas y las columnas de la tabla de contingencia. Los valores observados son los que aparecen en las tablas de contingencia y los valores esperados se calculan a partir de las sumas de columnas y filas como pasamos a ver.

Supongamos, por ejemplo, que tenemos una muestra de 260 personas de un determinado país en la que se muestra su género y si la persona va en bicicleta al trabajo. Queremos ver si están relacionadas las variables `genero` con la de `bicicleta`.

```
personas  <- c( 71, 48, 65, 76 )
genero    <- c( "hombre", "hombre", "mujer", "mujer" )
bicicleta <- c( "Sí", "No", "Sí", "No" )
dfBici    <- data.frame( personas, bicicleta, genero )
addmargins( xtabs( dfBici ) )
```

```
##           genero
## bicicleta hombre mujer Sum
##       No      48    76 124
##       Sí      71    65 136
##       Sum    119   141 260
```

Fijémonos en los hombres que sí se desplazan en bicicleta ($i = 2$ y $j = 1$). El valor que aparece en la tabla es 71, luego $\text{observado}_{21} = 71$. El valor esperado se calcula multiplicando el número de hombres que hay (119) por el número de personas que van en bicicleta (136) entre el número total de personas, esto es

$$\text{esperado}_{21} = \frac{\text{totalFila2} \times \text{totalColumna1}}{\text{totalMuestra}} = \frac{136 \times 119}{260} \approx 62{,}246$$

Por tanto, para la entrada de la fila 2 columna 1 el valor que se suma al estadístico es

$$\frac{(71 - 62{,}246)^2}{62{,}246} \approx 1{,}23$$

y, repitiendo el proceso en toda la tabla de contingencia y sumando se calcula el estadístico χ^2.

Como es obvio por el nombre, el estadístico sigue aproximadamente una distribución χ^2 con $(f-1)(c-1)$ grados de libertad, donde f indica el número total de filas y c el de columnas.

El problema de esta prueba surge de que utiliza una aproximación a la χ^2, es decir, surge cuando tenemos tablas de contingencia pequeñas. Así, cuando tenemos tablas 2×2 Yates propuso una corrección a la fórmula (ver A. Field et al., 2012 para más información) que es la que utilizan algunas funciones de R.

Las únicas suposiciones que deben cumplirse para utilizar la prueba χ^2 de Pearson son:

- que los datos no estén apareados (dependientes). En esa situación se utilizaría la prueba de McNemar que veremos más adelante;
- y que las frecuencias esperadas no sean menores a 5. En tablas grandes es suficiente que el 80 % lo cumplan (nunca siendo menores a 1). Si no se da esta situación también se opta por usar el test exacto de Fisher. Estas condiciones también se conocen como *"condiciones de Cochran"*.

3.2.1.2. En R

Utilizamos la función `chisq.test()` indicando la tabla de contingencia

```
chisq.test( xtabs( dfBici ) )
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  xtabs(dfBici)
## X-squared = 4.2316, df = 1, p-value = 0.03968
```

En este caso, como el p-valor es menor a 0.05 se rechaza la hipótesis nula de que las variables no guardan relación entre sí. Al ser una tabla 2×2 vemos que utiliza la corrección de Yates; para no hacerla basta poner el argumento `correct = FALSE`.

Nota: En R el estadístico se denomina *X-squared*.

3.2.1.3. Fuerza de asociación (tamaño del efecto)

Como en esta prueba estamos contrastando si las variables están relacionadas (esto es, asociadas) se suele hablar de **fuerza de asociación** en lugar de tamaño del efecto.

Para medir la fuerza de asociación con R podemos utilizar la función `assocstats()` del paquete `vcd`. Esta función nos devuelve el valor de las tres medidas de asociación más frecuentes:

- El **coeficiente *phi***: cuando tenemos una tabla 2×2 varía entre 0 y 1 y tiene una interpretación similar a la r de Pearson (más correlación cuanto más cercano a 1). En tablas mayores el límite superior sobrepasa el 1 y es más difícil de interpretar.
- El **coeficiente de contingencia**: siempre varía entre 0 y 1 y se interpreta como la *phi*. El problema es que pocas veces alcanza el valor 1. Por esa razón Cramer propuso posteriormente la V.
- La ***V* de Cramer**: para tablas 2×2 coincide con la *phi*. Siempre varía entre 0 y 1, y en tablas mayores de 2×2 no le pasa lo que al coeficiente de contingencia, es decir, puede alcanzar su valor máximo 1.

```
## install.packages( "vcd" )
library( "vcd" )
```

```
## Loading required package: grid
```

```
assocstats( xtabs( dfBici ) )
```

```
##                     X^2 df P(> X^2)
## Likelihood Ratio 4.7785  1 0.028818
## Pearson          4.7598  1 0.029131
##
## Phi-Coefficient   : 0.135
## Contingency Coeff.: 0.134
## Cramer's V        : 0.135
```

Al ejecutar la función con el ejemplo anterior podemos observar que al tratarse de una tabla 2×2 las tres medidas ofrecen valores muy similares.

Otra medida que se suele utilizar además de estas es la **razón de oportunidades**, más conocida por el inglés *odds ratio*.

3.2.1.4. Post-hoc

Cuando hay más de de tres categorías en una variable (por ejemplo, una tabla 2×3) y se han obtenido resultados significativos nos puede interesar saber en qué niveles de la variable se han encontrado las relaciones.

Primera opción: ver los valores residuales de Pearson. Los valores residuales son las diferencias entre los valores observados y esperados, y si estos se dividen por la raíz de los esperados se obtienen los residuales de Pearson. Estos valores se estandarizan y se pueden comparar con la normal (por ejemplo, si supera 1.96 es significativo a un nivel de 0.05)

$$\text{residual Pearson}_{i,j} = \frac{\text{observado}_{ij} - \text{esperado}_{ij}}{\sqrt{\text{esperado}_{ij}}}$$

En R se calculan cuando hacemos nuestro test con `chisq.test()` pero no se muestran. Para verlos hacemos `chisq.test()$residuals` o `chisq.test()$stdres` para los estandarizados.

```
testChi <- chisq.test( xtabs( dfBici ) )
testChi$residuals
```

```
##          genero
## bicicleta    hombre     mujer
##        No -1.161987  1.067493
##        Sí  1.109539 -1.019311
```

```
testChi$stdres
```

```
##          genero
## bicicleta    hombre     mujer
##        No -2.181703  2.181703
##        Sí  2.181703 -2.181703
```

Nota: Aquí solo vemos cómo utilizar la función, por eso utilizamos la anterior tabla 2×2.

Si uno se fija en la fórmula puede ver que el estadístico χ^2 es la suma de los cuadrados de los residuales de Pearson. Así, lo bueno (y lo malo) de esta opción es que es una manera visual (y no muy exacta) de ver qué valores tienen más peso en el estadístico.

Segunda opción: partir la tabla en varias tablas 2×2 y a cada una hacerle una prueba χ^2 de Pearson corrigiendo la significación. En el ejemplo siguiente vamos a ver cómo partir una tabla 2×4 escogiendo cada pareja de columnas sin más.

Para corregir la significación podemos utilizar Bonferroni dividiendo el nivel de significación α entre el número k de comparaciones que realicemos. Si hacemos todas las comparaciones posibles, k se calcula como

$$k = \frac{r!}{2!(r-2)!} \cdot \frac{c!}{2!(c-2)!} = \frac{r\,(r-1)\,c\,(c-1)}{4}.$$

En muchos libros y apuntes (Agresti, 2002; Anderson, 2014; University, 2014) se recomienda una partición diferente que mantiene el valor del estadístico (los estadísticos de las particiones suman el de la tabla original). En este caso, se utiliza un valor de k menos restrictivo, esto es, un valor que no reduzca tanto el nivel de significación (ver DeVries, 2007)

$$k = \frac{r!}{2!(r-1)!} \cdot \frac{c!}{2!(c-1)!} = \frac{r \cdot c}{4}$$

3.2.1.5. Ejemplo

El conjunto de datos `HairEyeColor` de R contiene la distribución del color del pelo, el color los ojos y el género de 592 estudiantes de la Universidad de Delaware en 1974. ¿Están relacionados el género con el color del pelo?

Primero nos quedamos solo con el color del pelo y el género. Después le aplicamos la prueba χ^2

```
## trabajamos solo con el género y el color del pelo
tColorPelo <- margin.table( HairEyeColor, c( 3, 1 ) )
tColorPelo
```

```
##         Hair
## Sex      Black Brown Red Blond
##   Male      56   143  34    46
##   Female    52   143  37    81
```

```
chisq.test( tColorPelo )
```

```
##
##  Pearson's Chi-squared test
##
## data:  tColorPelo
## X-squared = 7.9942, df = 3, p-value = 0.04613
```

Obtenemos un p-valor de 0.04613 (menor a 0.05) luego rechazamos la hipótesis nula de que no existe relación. El siguiente paso es ir escogiendo parejas de columnas y repitiendo la prueba χ^2 a cada una. En este caso, al haber cuatro columnas (los colores del pelo: moreno , castaño, rojo y rubio) todas las posibles parejas serían: 1-2, 1-3, 1-4, 2-3, 2-4 y 3-4. Si quisiéramos hacer comparaciones dos a dos sobre todas las parejas utilizaríamos $\alpha = \frac{0{,}05}{6} = 0{,}0083$.

Hagamos, por ejemplo, los casos 1-2 (moreno-castaño) y el 2-4 (castaño-rubio) para ver la dinámica. En este caso utilizamos $\alpha = \frac{0{,}05}{2} = 0{,}025$ por hacer solo dos comparaciones.

```
## seleccionamos las columnas 1 = moreno y 2 = castaño
t12ColorPelo <- tColorPelo[ , c( 1, 2 ) ]
t12ColorPelo
```

```
##         Hair
## Sex      Black Brown
##   Male      56   143
##   Female    52   143
```

```
chisq.test( t12ColorPelo )
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  t12ColorPelo
## X-squared = 0.046225, df = 1, p-value = 0.8298
```

En este caso, como el p-valor sale mayor a 0.025 aceptamos la hipótesis nula de que no existe relación entre el género y el color del pelo moreno o castaño.

```
## seleccionamos las columnas 2 = castaño y 4 = rubio
t24ColorPelo <- tColorPelo[ , c( 2, 4 ) ]
t24ColorPelo
```

```
##         Hair
## Sex      Brown Blond
##   Male     143    46
```

```
##   Female   143    81
```

```
chisq.test( t24ColorPelo )
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  t24ColorPelo
## X-squared = 6.1842, df = 1, p-value = 0.01289
```

Ahora el p-valor sale significativo (menor a 0.025), luego rechazamos la hipótesis nula de que no existe relación entre el género y el color del pelo castaño y rubio.

3.2.2. Test exacto de Fisher

El **test exacto de Fisher** aunque se denomina test es más bien un método de cálculo de las probabilidades de la χ^2 cuando el tamaño muestral es pequeño.

Uno de los problemas con la prueba χ^2 de Pearson es que utiliza una aproximación a la distribución χ^2 (de ahí el comentario de los valores mayores a cinco) y cuanto mayor es el tamaño muestral mejor se aproxima. En estos casos en los que se tiene poca muestra es donde se utiliza el método ideado por Fisher para calcular de manera exacta la probabilidad del estadístico de la χ^2.

Generalmente se utiliza con tablas de contingencia 2×2 y con poca muestra, aunque también se puede utilizar con tablas mayores (eso sí, tardaría más tiempo en calcular y en esos casos se podría usar la prueba χ^2 de Pearson). También suele utilizarse cuando no se cumplen las *condiciones de Cochran* y no podemos aplicar la prueba χ^2 de Pearson.

3.2.2.1. ¿Cómo funciona el test exacto de Fisher?

Dada una tabla de contingencia con los valores marginales (los de los márgenes, las sumas de filas y columnas) se calcula la probabilidad de que, con esos valores marginales, se hayan dado por azar los valores de la tabla o más raros. Veamos un ejemplo.

Supongamos que tenemos una tabla de contingencia cualquiera

Cuadro 3: Tabla de contingencia. Prueba exacta de Fisher.

	Categoría1	Categoría2	Total
Categoría3	a	b	a+b
Categoría4	c	d	c+d
Total	a+c	b+d	n=a+b+c+d

Suponiendo la hipótesis nula de independencia los valores marginales están fijos y solo hay un grado de libertad (esto es, sabiendo el valor de una celda de la tabla se pueden calcular los demás). La probabilidad de que en la posición de "a" aparezca el valor "a" por azar (y por tanto, que salgan "b", "c" y "d" en las demás) se calcula con la fórmula de la distribución hipergeométrica

$$p = \frac{\binom{a+b}{a}\binom{c+d}{c}}{\binom{n}{a+c}} = \frac{(a+b)!\,(c+d)!\,(a+c)!\,(b+d)!}{a!\,b!\,c!\,d!\,n!}$$

Si se calculan todas las probabilidades de todas las posibles tablas, y después se suman las que tengan probabilidad menor o igual que la nuestra (esto depende de la hipótesis alternativa) obtenemos el p-valor.

3.2.2.2. En R

Podemos utilizar la función `fisher.test()` indicando la tabla de contingencia. Repitiendo el ejemplo de la prueba χ^2 de Pearson (aunque en este caso no haría falta utilizar la de Fisher)

```
fisher.test( xtabs( dfBici ) )
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  xtabs(dfBici)
## p-value = 0.03427
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.342271 0.975485
## sample estimates:
## odds ratio
##  0.5794287
```

En este caso, volvemos a obtener un p-valor menor a 0.05 (muy cercano al que devuelve la prueba χ^2 de Pearson) y se rechaza la hipótesis nula de que las variables no están relacionadas.

3.2.2.3. Fuerza de asociación (tamaño del efecto)

Se calcula de la misma manera que para la prueba χ^2 de Pearson.

```
library( "vcd" )
assocstats( xtabs( dfBici ) )
```

```
##                     X^2 df P(> X^2)
## Likelihood Ratio 4.7785  1 0.028818
## Pearson          4.7598  1 0.029131
##
## Phi-Coefficient   : 0.135
## Contingency Coeff.: 0.134
## Cramer's V        : 0.135
```

3.2.3. Prueba de McNemar

La **prueba de McNemar** es la alternativa a la prueba χ^2 de Pearson para muestras dependientes o pareadas. Se aplica en tablas 2×2 cuando tenemos variables dicotómicas (dos valores).

3.2.3.1. ¿Cómo funciona la prueba de McNemar?

Supongamos que tenemos dos variables dependientes, por ejemplo, un test que se pasa antes y después de aplicar cierto tratamiento y que toma valores dicotómicos (puede dar positivo o negativo). El objetivo de la prueba de McNemar es contrastar si el tratamiento es efectivo, esto es, si hace cambiar los test de positivo a negativo (o viceversa).

Cuadro 4: Tabla de contingencia. Prueba de McNemar.

	Después Positivo	Después Negativo	Total
Antes Positivo	a	b	a+b
Antes Negativo	c	d	c+d
Total	a+c	b+d	n=a+b+c+d

Si el tratamiento no tuviera efecto se esperaría que las proporciones de mejorar (pasar de negativo a positivo) y de empeorar (pasar de positivo a negativo) fueran iguales respectivamente, esto es, que $p_a + p_b = p_a + p_c$ y $p_c + p_d = p_b + p_d$; lo que queda como $p_b = p_c$ (hipótesis nula). El estadístico que se utiliza en este caso es

$$\chi^2 = \frac{(b-(b+c)/2)^2}{(b+c)/2} + \frac{(c-(b+c)/2)^2}{(b+c)/2} = \frac{(b-c)^2}{b+c}$$

que se aproxima a una distribución χ^2 con 1 grado de libertad.

Al usar una aproximación, hay que tener cuidado cuando b o c son demasiado pequeños ($b + c < 25$). En este caso se utiliza la distribución binomial para calcular el valor exacto del estadístico (similar a lo que se hace con la prueba χ^2 de Pearson y el test exacto de Fisher).

3.2.3.2. En R

Se puede realizar con la función `mcnemar.test()`. Cuando la suma $b + c$ es pequeña se puede utilizar la función `binom.test()`.

3.2.3.3. Fuerza de asociación (tamaño del efecto)

De nuevo, se calcula de la misma manera que para la prueba χ^2 de Pearson y el test exacto de Fisher. Veámoslo es el siguiente ejemplo.

3.2.3.4. Ejemplo

Supongamos que le preguntamos a un grupo de turistas chinos si se comprarían en España el iPhone6, ya que, sale unos 150 euros más barato por culpa del cambio entre el yuan y el euro. Después les explicamos que en Apple China no valen las garantías internacionales y ni siquiera gestionan las reparaciones de smartphones no comprados en China. Finalmente le volvemos a formular la pregunta de si comprarían un iPhone6 en España. ¿Ha supuesto la información dada un cambio en la intención de compra de los turistas? (Datos simulados).

```
dfMovil <- read.table( "files/feir50A-mcnemar.csv" , header = TRUE, sep = ";" )
dfMovil <- dfMovil[ , -1 ] # limpiamos los datos
head( dfMovil )
```

```
##   antes despues
## 1    Sí      No
## 2    Sí      No
## 3    Sí      Sí
## 4    No      No
## 5    No      No
## 6    Sí      Sí
```

```
tMovil <- table( dfMovil )
tMovil
```

```
##      despues
## antes No Sí
##    No  7  1
##    Sí 10  2
```

```
mcnemar.test( tMovil )
```

```
##
##  McNemar's Chi-squared test with continuity correction
##
## data:  tMovil
```

```
## McNemar's chi-squared = 5.8182, df = 1, p-value = 0.01586
```

El test nos devuelve un p-valor de 0.01586 (menor a 0.05), luego rechazamos la hipótesis nula de que no hay un cambio en la intención de compra de los turistas.

Para calcular la fuerza de asociación recordemos que podemos utilizar la función `assocstats()` del paquete `vcd`.

```
library( "vcd" )
assocstats( tMovil )
```

```
##                     X^2 df P(> X^2)
## Likelihood Ratio 0.066572  1  0.79640
## Pearson          0.065359  1  0.79822
##
## Phi-Coefficient   : 0.057
## Contingency Coeff.: 0.057
## Cramer's V        : 0.057
```

Nota: Aunque la función `mcnemar.test()` por defecto hace corrección por continuidad (se puede modificar con `correct = FALSE`) la suma $b + c$ es pequeña. En este caso se suele utilizar el test binomial exacto con `binom.test()` al que se le pasa la suma de los casos que han cambiado de opinión ($b + c$) y uno de esos dos valores (b o c).

```
## en el ejemplo anterior: b + c = 11
binom.test( x = 10, n = 11, p = 0.5  )
```

```
##
##  Exact binomial test
##
## data:  10 and 11
## number of successes = 10, number of trials = 11, p-value = 0.01172
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
##  0.5872201 0.9977010
## sample estimates:
## probability of success
##              0.9090909
```

Observamos que sale un p-valor de 0.01172, menor a 0.05 como antes.

3.2.4. Prueba Q de Cochran

La **prueba Q de Cochran** es equivalente a la prueba de McNemar para más de dos grupos, esto es, para contrastar la independencia entre varias muestras apareadas. Se utiliza como alternativa a la prueba de Friedman cuando se tienen variables dicotómicas.

3.2.4.1. ¿Cómo funciona la prueba Q?

Supongamos que preguntamos a un grupo de 17 personas si compraría ropa de una cierta marca (`inicial`). Luego le ponemos publicidad de esa marca y le volvemos a hacer la misma pregunta (`publicidad`). Finalmente, le enseñamos los comentarios y opiniones de gente en internet sobre esa marca para repetirles la misma pregunta por última vez (`internet`). Nos preguntamos si la publicidad y los comentarios y opiniones en internet cambia la intención de la gente de comprarla (para bien o para mal).

Cuadro 5: Intención de compra según información dada.

	inicial	publicidad	internet
1	1	1	1
2	0	1	1
3	1	1	1
⋮	⋮	⋮	⋮
16	0	0	1
17	0	1	1

Para saber si hay diferencias significativas en la intención de compra según la información que se le proporcione al cliente se calcula un estadístico Q a partir de la tabla anterior

$$Q = (k-1)\frac{k\sum_{j=1}^{k} G_j^2 - \left(\sum_{j=1}^{k} G_j\right)^2}{k\sum_{i=1}^{b} L_i - \sum_{i=1}^{b} L_i^2}$$

donde k es el número de columnas, b el de filas, G_j la suma de la columna j y L_i la suma de la fila i.

Este estadístico se aproxima a una χ^2 con $k-1$ grados de libertad. Cuando mayor sea el tamaño muestral (b en la fórmula) mejor será la aproximación.

En este ejemplo hemos visto cómo la prueba Q nos puede servir para poder contrastar diferencias de una misma variable (la intención de compra) en diferentes momentos (antes y después de ofrecer cierta información). Otro caso que puede parecer distinto y en el que se puede utilizar también la prueba Q es cuando queremos comparar varias variables (dicotómicas) generalmente en un mismo tiempo. Por ejemplo, si queremos ver si a la hora de comprar un smartphone en un primer momento la gente consideraría hacerse con un iPhone, con un Samsung y con otro modelo. La hipótesis nula es que las variables tienen a coincidir para cada persona, que no hay una preferencia antes de comparar los móviles.

Un texto muy recomendable es Nebraska–Lincoln (2007).

3.2.4.2. En R

Podemos utilizar la función `symmetry_test()` del paquete `coin` (ver Yatani, 2014)

Otra alternativa es utilizar la función `cochran.qtest()` del paquete `RVAideMemoire` (puede tardar un poco en instalarse).

3.2.4.3. Ejemplo

Vamos a desarrollar el ejemplo anterior con R. Recordemos que tenemos un grupo de 17 personas a los que se les ha preguntado si comprarían una cierta marca de ropa.

Para llevar a cabo una prueba Q de Cochran en R primero debemos pasar la tabla a vectores como se muestra a continuación.

```
dfCompra <- read.table( "files/feir50A-cochranQ.csv", header = TRUE, sep = ";" )
## preparamos los datos
informacion <- rep( colnames( dfCompra )[], 17 )
head( informacion )
```

```
## [1] "inicial"    "publicidad" "internet"   "inicial"    "publicidad"
## [6] "internet"
```

```
persona <- rep( 1:17, each = 3 )
head( persona, 10 )
```

```
##  [1] 1 1 1 2 2 2 3 3 3 4
```

```
intencion <- as.vector( t( dfCompra ) )
head( intencion, 15 )
```

```
##  [1] 1 1 1 0 1 1 1 1 1 0 0 1 0 0 1
```

```
library( "coin" )
symmetry_test( intencion ~ factor( informacion ) | factor( persona ), teststat = "quad" )
```

```
##
##  Asymptotic General Symmetry Test
##
## data:  intencion by
##   factor(informacion) (inicial, internet, publicidad)
##   stratified by factor(persona)
## chi-squared = 15.2, df = 2, p-value = 0.0005005
```

Obtenemos un p-valor menor de 0.05 luego rechazamos la hipótesis nula de que la información no provoca ningún cambio en la intención de compra.

3.2.4.4. Post-hoc

Para saber qué información ha hecho cambiar a la gente de opinión sobre la marca hacemos un análisis post-hoc. Lo más habitual es hacer comparaciones dos a dos con la prueba de McNemar y corregir la significación (Bonferroni u otra).

En el ejemplo anterior podemos hacer la prueba de McNemar sobre las parejas: `inicial-publicidad`, `inicial-internet` y `publicidad-internet`. Si las vamos guardando para luego poder obtener el p-valor añadiendo `$p.value` al final podemos corregir la significación con la función `p.adjust()`.

Hacemos la prueba de McNemar entre las categorías `inicial` y `publicidad`

```
mcnemarTest12 <- mcnemar.test( table( dfCompra[ , c( 1, 2 ) ] ) )
mcnemarTest12
```

```
##
##  McNemar's Chi-squared test with continuity correction
##
## data:  table(dfCompra[, c(1, 2)])
## McNemar's chi-squared = 2.25, df = 1, p-value = 0.1336
```

ahora entre `inicial` e `internet`

```
mcnemarTest13 <- mcnemar.test( table( dfCompra[ , c( 1, 3 ) ] ) )
mcnemarTest13
```

```
##
##  McNemar's Chi-squared test with continuity correction
##
## data:  table(dfCompra[, c(1, 3)])
## McNemar's chi-squared = 8.1, df = 1, p-value = 0.004427
```

y, por último, entre `publicidad` e `internet`

```
mcnemarTest23 <- mcnemar.test( table( dfCompra[ , c( 2, 3 ) ] ) )
mcnemarTest23
```

```
##
##  McNemar's Chi-squared test with continuity correction
##
## data:  table(dfCompra[, c(2, 3)])
## McNemar's chi-squared = 4.1667, df = 1, p-value = 0.04123
```

Como hemos dicho, al guardarlas en variables `mcnemarTestXY` podemos obtener el p-valor y corregir la significación con la función `p.adjust()`

```
p.adjust( c( mcnemarTest12$p.value, mcnemarTest13$p.value, mcnemarTest23$p.value ),
            method = "bonferroni" )
```

```
## [1] 0.40084321 0.01327958 0.12368050
```

Obtenemos un p-valor menor a 0.05 en el segundo caso, correspondiente a las categorías `inicial` e `internet`. Por tanto, hemos encontrado diferencias significativas en la intención de compra del grupo de personas entre el momento inicial y después de enseñarle los comentarios y opiniones de internet.

3.2.4.5. Fuerza de asociación (tamaño del efecto)

Igual que ocurre con la prueba de Kruskal-Wallis no hay una manera general para calcular la fuerza de asociación (tamaño del efecto). Podemos calcular la fuerza de asociación sobre las parejas con resultados significativos del análisis post-hoc. Esto es, con lo visto en la prueba de McNemar.

En el ejemplo anterior hemos encontrado diferencias entre las categorías `inicial` e `internet`, por tanto, podemos calcular la fuerza de asociación entre ellas

```
library( "vcd" )
assocstats( table( dfCompra[ , c( 1, 3 ) ] ) )
```

```
##                     X^2 df P(> X^2)
## Likelihood Ratio 0.90442  1  0.34160
## Pearson          0.57955  1  0.44649
##
## Phi-Coefficient   : 0.185
## Contingency Coeff.: 0.182
## Cramer's V        : 0.185
```

3.2.5. Otros

3.2.5.1. Prueba de la razón de verosimilitud

Es una alternativa a la prueba χ^2 de Pearson. También se conoce como **prueba *G*** o, en inglés, *likelihood-ratio test*. Un texto recomendable sobre esta prueba es A. Field et al. (2012) o Wikipedia (2014).

3.2.5.2. Prueba de Cochran-Mantel-Haenszel

Se utiliza normalmente cuando queremos comparar tablas de frecuencias 2×2 en diferentes tiempos o situaciones. La hipótesis nula es que no existen diferencias entre las frecuencias de las tablas de contingencia.

Por ejemplo, supongamos que queremos comparar si cierto tratamiento provoca igual mejoría para hombres y mujeres. En este caso, para cada género (masculino y femenino) tendríamos una tabla de frecuencias de tratamiento frente a mejoría.

En R podemos utilizar la función `mantelhaen.test()`. Más ejemplos en los que se puede aplicar esta prueba se pueden encontrar en la ayuda de esta función `?mantelhaen.test()`. Un texto recomendable de esta prueba es McDonald (2014).

Volver al índice del curso

Referencias y bibliografía

Agresti, A. (2002). *Categorical data analysis* (2nd ed.). Wiley-Interscience.

Anderson, C. J. (2014). Applied categorical data analysis. Retrieved October 15, 2014, from http://courses.education.illinois.edu/EdPsy589/

DeVries, J. (2007). About chi squares. Retrieved October 15, 2014, from http://hdl.handle.net/10214/1863

Field, A., Miles, J., & Field, Z. (2012). *Discovering statistics using r* (1st edition.). Sage Publications Ltd.

Gibbons, J. D., & Chakraborti, S. (2003). *Nonparametric statistical inference* (4th ed.). Marcel Dekker.

Giraudoux, P. (2014). Pgirmess & pgirbric: Miscellaneous functions for data handling and analysis in ecology. Retrieved October 15, 2014, from http://giraudoux.pagesperso-orange.fr/

Kabacoff, R. (2011). *R in action* (1st ed.). Shelter Island, NY: Manning Publications.

McDonald, J. H. (2014). Handbook of biological statistics. Sparky House Publishing. Retrieved October 15, 2014, from http://www.biostathandbook.com/cmh.html

Motulsky, H. (2012). Choosing a statistical test (material de intuitive biostatistics). Retrieved October 15, 2014, from http://www.graphpad.com/support/faqid/1790/

Nebraska–Lincoln, U. of. (2007). Bivariate statistics hand-computation cache: Cochran's q test. Retrieved October 15, 2014, from http://psych.unl.edu/psycrs/handcomp/hccochran.PDF

University, P. (2014). STAT 504 - analysis of discrete data. Retrieved October 15, 2014, from https://onlinecourses.science.psu.edu/stat504/node/87

Wackerly, D. D., Mendenhall, W., & Scheaffer, R. L. (2008). *Mathematical statistics with applications* (7th ed.). Cengage Learning.

Wikipedia. (2014). G-test — wikipedia, the free encyclopedia. Retrieved October 15, 2014, from http://en.wikipedia.org/w/index.php?title=G-test&oldid=627245215

Yatani, K. (2014). Statistical methods for hci research. Retrieved October 16, 2014, from http://yatani.jp/teaching/doku.php?id=hcistats:start