

Práctica 11. Ejercicios

Jesús Martín Fernández

Selecciona el directorio de trabajo y carga la base de datos `df_iam3`, que puedes obtener de la Carpeta de la Práctica 11 en el Aula Virtual:

```
#setwd()  
getwd ()  
  
df_iam3<-read.csv ("df_iam3.csv")
```

Recuerda que, en esta base se recogen una serie de características de 984 sujetos. En un momento en el tiempo se recogió qué sujetos habían tenido un evento tipo infarto de miocardio (`iam`). Posteriormente se siguió a los sujetos hasta otro punto en el tiempo. Tras ese punto `t`, sólo se recogió si el sujeto seguía fumando tras el primer infarto (`fum_p`), la cifra de colesterol (`colesterol_p`), su nueva percepción del estado de salud (`salud_p`) y la ocurrencia de un nuevo reinfarto (`iam2`)

Obten la variable edad (en años cumplidos), suponiendo que la fecha final de seguimiento es el 30/6/2024 a partir de la variable `fech_nac`

```
#fech_nac viene definida como character, cambiamos a formato fecha  
df_iam3$fech_nac <- as.Date(df_iam3$fech_nac)  
fecha_fin <- as.Date("2024-06-30")  
df_iam3$edad <- (fecha_fin-df_iam3$fech_nac)/365.25  
df_iam3$edad <- as.numeric (round (df_iam3$edad,0))
```

Seguidamente etiquetas, como se hizo en la práctica correspondiente, las variables `sex` (etiquetas: 0="mujer; 1="Varón"), `hta`, `fum`, `fum_p`, `DM`, `iam1` e `iam2`, con las etiquetas (0="No", 1="Sí"), la variable `clas_soc` con las etiquetas (0= "baja"; 1= "Media", 2= Alta") y `salud` y `salud_p` con las etiquetas (0="Muy mala"; 1="Mala"; 2= "Regular"; 3= "Buena"; 4= "Muy buena")

```

df_iam3$sex <- factor(df_iam3$sex, levels = c(0, 1),
                     labels = c("Mujer", "Varón"))
df_iam3$hta <- factor(df_iam3$hta, levels = c(0, 1), labels = c("No", "Sí"))
df_iam3$fum <- factor(df_iam3$fum, levels = c(0, 1), labels = c("No", "Sí"))
df_iam3$fum_p <- factor(df_iam3$fum_p,
                       levels = c(0, 1), labels = c("No", "Sí"))
df_iam3$DM <- factor(df_iam3$DM, levels = c(0, 1), labels = c("No", "Sí"))
df_iam3$iam1 <- factor(df_iam3$iam1, levels = c(0, 1), labels = c("No", "Sí"))
df_iam3$iam2 <- factor(df_iam3$iam2, levels = c(0, 1), labels = c("No", "Sí"))
df_iam3$clas_soc <- factor(df_iam3$clas_soc,
                           levels = c(0, 1, 2),
                           labels = c("Baja", "Media", "Alta"))
df_iam3$salud <- factor(df_iam3$salud,
                       levels = c(0,1, 2,3,4),
                       labels = c("Muy mala", "Mala",
                                   "Regular", "Buena", "Muy buena"))
df_iam3$salud_p <- factor(df_iam3$salud_p,
                          levels = c(0,1, 2,3,4),
                          labels = c("Muy mala", "Mala",
                                      "Regular", "Buena", "Muy buena"))

```

Comprueba que ha funcionado todo bien

```
head (df_iam3)
```

1. Comparación de proporciones, tablas de n x m

Vamos a ver si las proporciones de diabéticos son diferentes en las diferentes categorías de la variable `clas_soc`

```

tabla_DM_class <- table(df_iam3$clas_soc, df_iam3$DM)

tabla_DM_class

chi_test_3 <- chisq.test(tabla_DM_class, correct=FALSE)
chi_test_3

# Frecuencias esperadas

print(chi_test_3$expected)

```

Parece que las proporciones son muy diferentes de las esperadas por azar (p-value muy pequeño). Además las frecuencias esperadas son altas siempre por lo que no haría falta la corrección de Yates.

2. Test de comparación de proporciones

Vamos a comparar las proporciones de DM en la clase social “Baja” frente a “Media” y a comparar las proporciones en “Media” frente a “Alta”

```
# Crear tabla de frecuencias
tabla_DM_class <- table(df_iam3$clas_soc, df_iam3$DM)

prop_DM_class <- prop.table(tabla_DM_class, margin = 1)

prop_DM_class

# Calcular las diferencias de proporciones
# Comparar "Baja" vs "Media"
n1_baja <- sum(tabla_DM_class["Baja", ]) # Total en clase Baja
n1_media <- sum(tabla_DM_class["Media", ]) # Total en clase Media
p1_baja <- prop_DM_class["Baja", "Sí"] # Proporción con DM en clase Baja
p1_media <- prop_DM_class["Media", "Sí"] # Proporción con DM en clase Media

dif_prop_baja_media <- prop.test(x =
  c(tabla_DM_class["Baja", "Sí"], tabla_DM_class["Media", "Sí"]),
  n = c(n1_baja, n1_media))

dif_prop_baja_media

# Comparar "Media" vs "Alta"
n1_alta <- sum(tabla_DM_class["Alta", ]) # Total en clase Alta
p1_alta <- prop_DM_class["Alta", "Sí"] # Proporción con DM en clase Alta

dif_prop_media_alta <- prop.test(x =
  c(tabla_DM_class["Media", "Sí"], tabla_DM_class["Alta", "Sí"]),
  n = c(n1_media, n1_alta))

dif_prop_media_alta
```

Si nos fijamos parece haber diferencias claras en las proporciones de diabéticos entre las clases “Baja” y “Media” , con un IC 95% para la diferencia de proporciones entre:

0.09216273 - 0.19564214

Sin embargo las diferencias de proporciones de DM entre las clases “Media” y “Alta”, no son tan importantes, siendo el IC del 95%:

0.002246696 - 0.112646921

En este segundo caso, nos encontramos con una situación curiosa, el extremo superior está muy cerca del 0 (0,0022), pero no lo incluye. Sin embargo el p-value es de 0,1146, lejos del umbral de 0,05. Esta discrepancia ocurre porque el p-value se obtiene de una distribución Chi cuadrado y los IC 95% se construyen bajo el supuesto de una distribución normal.

4. Test de tendencia lineal

En el caso anterior, la pregunta de más interés sería saber si hay una tendencia a que las proporciones de DM fuesen menores según aumenta la clase social. Esta asociación podría estar explicada en el contexto de conocimiento que tenemos porque los estilos de vida puedan ser más saludables en clases sociales altas.

Para testar esta idea haz un test de tendencia lineal entre las proporciones de la variable DM en los grupos (ordenados) de `clas_soc`

```
tabla_DM_class <- table(df_iam3$DM, df_iam3$clas_soc)
tabla_DM_class

# Calcular las proporciones de la tabla de contingencia
proporciones <- prop.table(tabla_DM_class, margin = 2)
proporciones
```

Y ahora hay que disponer los datos para poder usar la función `prop.trend.test`

```
# Definir el número de DM (x) y el tamaño de cada grupo (n)
x <- c(100, 48, 4)
n <- c(310 + 100, 432 + 48, 90 + 4)

# Realizar el test de tendencia
```

```
tendencia_resultado <- prop.trend.test(x, n, score = seq_along(x))  
tendencia_resultado
```

Parece clara una tendencia decreciente y se puede rechazar la hipótesis nula.