Solución del taller de problemas inferencia 2023 MAT3 GIN

Khaoula Ikkene

23 de diciembre de 2023

Contenidos

1 Taller INDIVIDUAL Problemas evaluable 22-23: Estadística Inferencial

Cada apartado es 1 punto. Total 18 puntos

Se trata de resolver los siguientes problemas y cuestiones en un fichero Rmd y su salida en un informe en html, word o pdf o escrito manualmente y escaneado.

1.1 Problema 1: Contraste de parámetros de dos muestras. Test AB. (6 puntos)

Se quiere evaluar dos interfaces gráficas para un vídeo juego la tipo A que es la actual y una nueva tipo B. Se selecciona dos muestras de jugadores independientes la primera prueba la interfaz A y la segunda la B. En cada muestra se observa el tiempo utilizado para completar una fase del juego en minutos. Las muestras son de tamaños $n_A = 1000$ y $n_B = 890$.

Los datos están adjuntos a los enunciados, en la carpeta `datasets` en un ficheros `AB.csv` que contien las variables tiempo y muestra que vale A o B.

1. Cargad de datos y calculad estadísticos descriptivos básicos y diagramas de caja e histogramas muestrales, utilizad la función `density`, comparativos de las dos muestras.
2. Estudiad si podemos aceptar que las muestras son normales con el test K-S-L, Ardenson-Darling test, Shapiro-Wilks y Dagostino-Pearson.
3. Calcular el estadístico de contraste del test K-S-L para la muestra A y comprobad el resultado.
4. Comprobad con el test de Fisher de razón de varianzas si las varianza de las dos muestras son iguales contra que son distintas. Tenéis que resolver el test de Fisher con R y de forma manual y el de Flinger de R y decidir utilizando el p-valor.

5. Con la información anterior elegid el contraste adecuado para saber si hay evidencia de que la la nueva interfaz mejora el tiempo de la actual. Resolver manualmente definiendo adecuadamente las hipótesis y decidid según el p-valor.
6. Calculad e interpretar el intervalo de confianza de los estadísticos del los test de medias y el de Fisher de los apartados 4 y 5.

##Solución

Apartado 1 Cargaremos los datos

```
 AB = read_csv("datasets/AB.csv")
```

```
## Rows: 1890 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): muestra
## dbl (1): tiempo
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

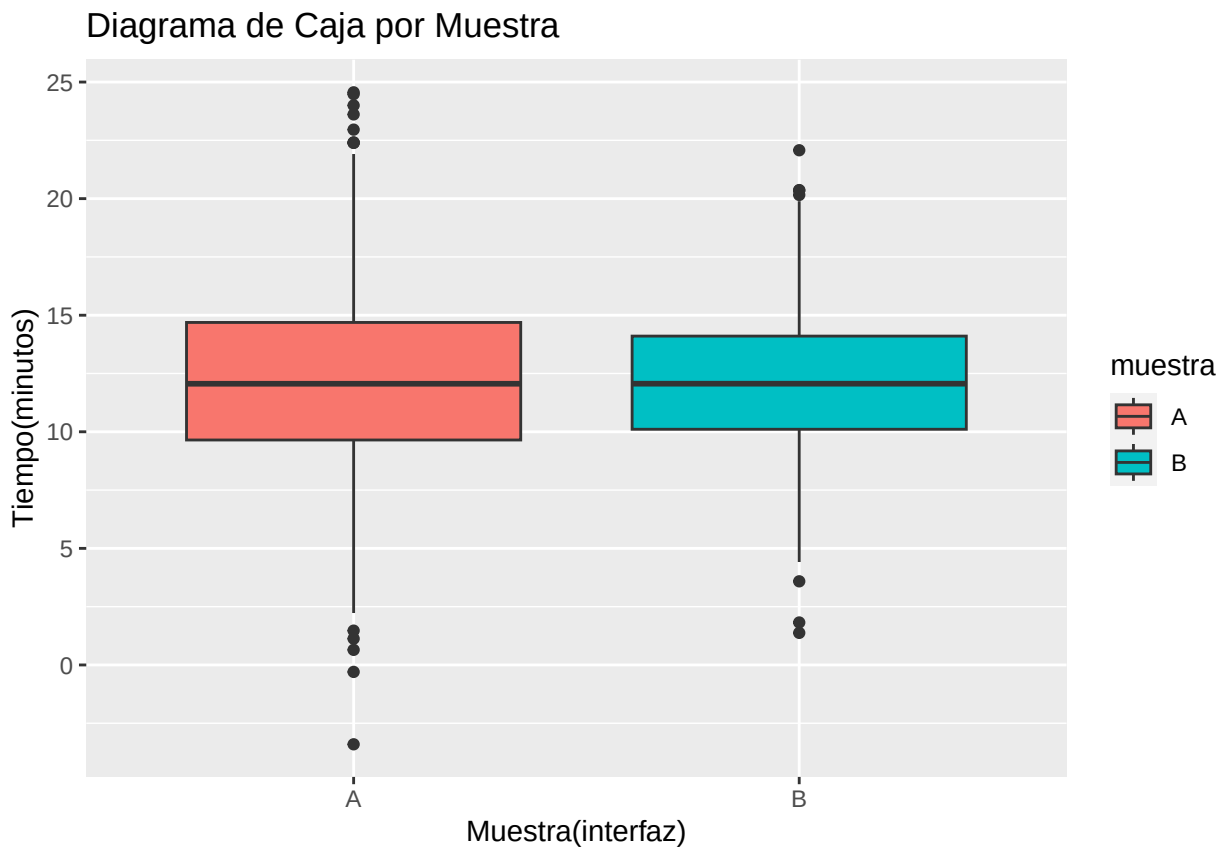
Cálculo de los estadísticos descriptivos

```
tabla_estadisticos = AB %>% group_by(muestra) %>% summarise(N=n(),Mean_muestra =mean(tiempo,na.rm = TRU
  knitr :: kable(tabla_estadisticos)
```

muestra	N	Mean_muestra	Desv_muestra	max_muestra	min_muestra
A	1000	12.17585	3.925452	24.55896	-3.403282
B	890	12.11932	2.990599	22.07549	1.376467

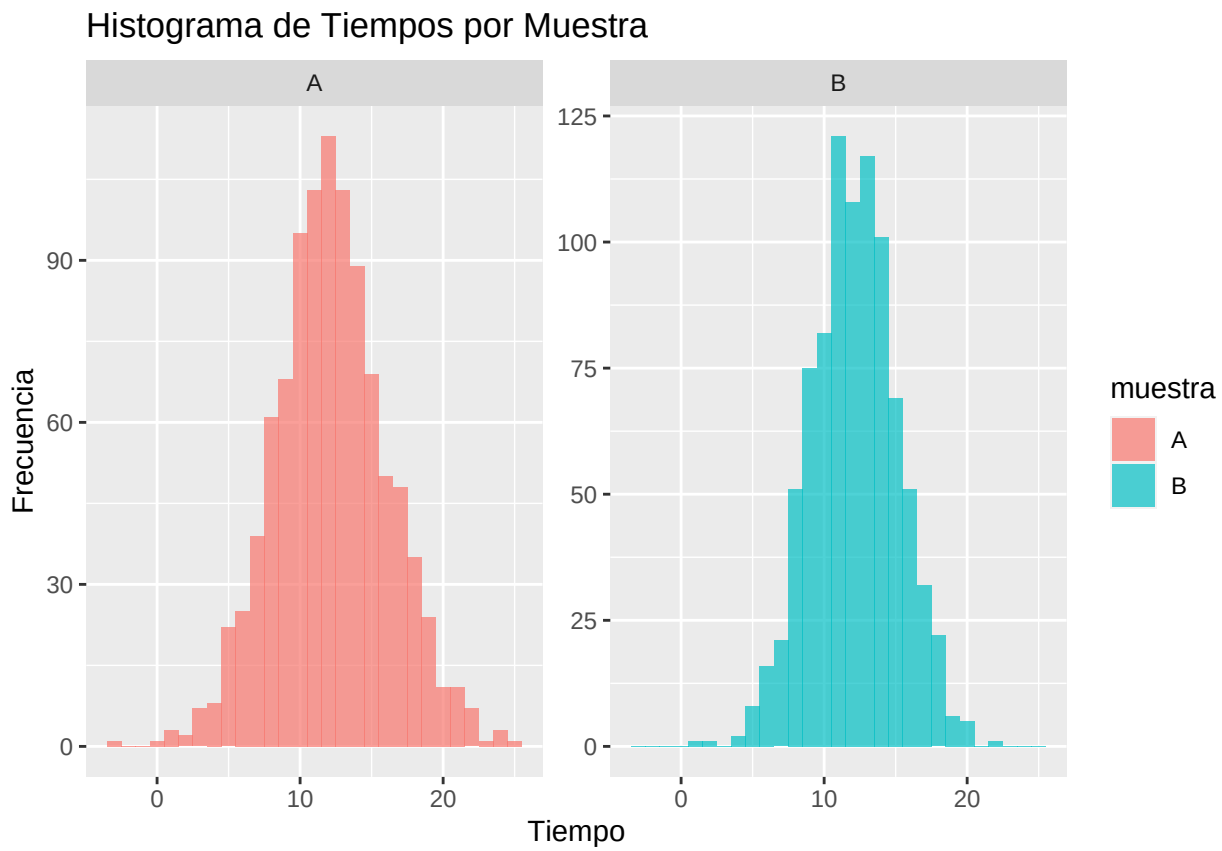
Diagramas de caja

```
ggplot(AB, aes(x = muestra, y = tiempo, fill = muestra)) +
  geom_boxplot() +
  labs(title = "Diagrama de Caja por Muestra",
       x = "Muestra(interfaz)",
       y = "Tiempo(minutos)")
```



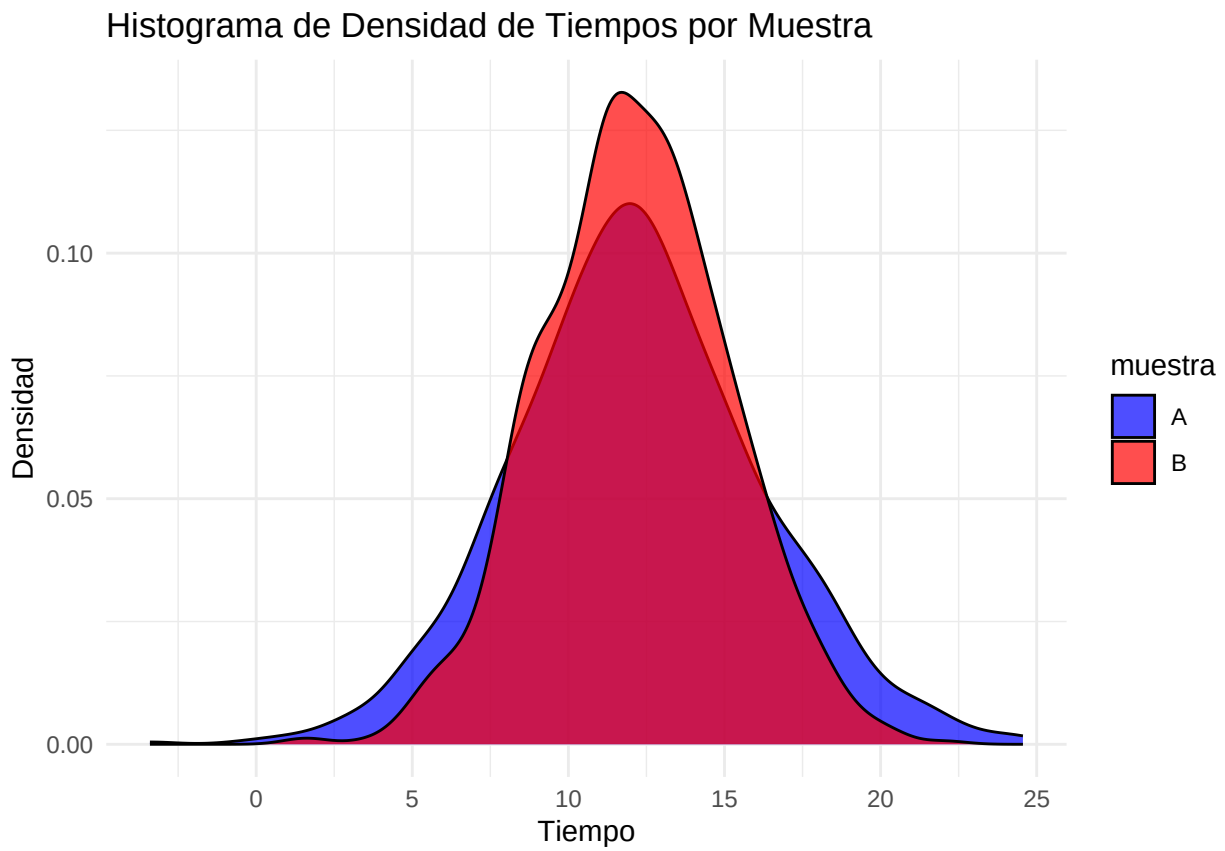
Histogramas muestrales

```
ggplot(AB, aes(x = tiempo, fill = muestra)) +
  geom_histogram(binwidth = 1, position = "identity", alpha = 0.7) +
  labs(title = "Histograma de Tiempos por Muestra",
       x = "Tiempo",
       y = "Frecuencia") +
  facet_wrap(~muestra, scales = "free_y")
```



Histograma de densidad

```
ggplot(AB, aes(x = tiempo, fill = muestra)) +
  geom_density(alpha = 0.7, position = "identity") +
  labs(title = "Histograma de Densidad de Tiempos por Muestra",
       x = "Tiempo",
       y = "Densidad") +
  scale_fill_manual(values = c("A" = "blue", "B" = "red")) +
  theme_minimal()
```



Apartado 2 Miramos si podemos acceptar que las muestras A y B son normales usando :

Obtenemos primero las muetras A y B por separado

```
muestra_A=AB$tiempo[AB$muestra == "A"]
muestra_B = AB$tiempo[AB$muestra == "B"]
```

Nuestro contraste es la muestra A es:

H_0 :La muestra de A sigue una distribución normal
H_1 :La muestra de A sigue cualquier distribución

-Test K-S-L

```
library(nortest)
lillie.test(muestra_A)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  muestra_A
## D = 0.026624, p-value = 0.09182
```

-Test de Ardenson-Darling

```
ad.test(muestra_A)
```

```
##
##  Anderson-Darling normality test
##
## data:  muestra_A
## A = 0.74028, p-value = 0.05364
```

-Test Shapiro-Wilks(S-W)

```
shapiro.test(muestra_A)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  muestra_A
## W = 0.99759, p-value = 0.1496
```

Test de Dagostino-Pearson, que para ello tenemos que usar la libreria moments de R

```
library(moments)
agostino.test(muestra_A)
```

```
##
##  D'Agostino skewness test
##
## data:  muestra_A
## skew = 0.069448, z = 0.902119, p-value = 0.367
## alternative hypothesis: data have a skewness
```

En todos los tests usados el p-valor era mayor que 0.05 y por lo tanto no podemos rechazar la hipótesis nula.

Para la muestra B repetimos los mismos cálculos:

-Test K-S-L

```
lillie.test(muestra_B)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  muestra_B
## D = 0.016554, p-value = 0.8021
```

-Test de Ardenson-Darling

```
ad.test(muestra_B)
```

```
##
##  Anderson-Darling normality test
##
## data:  muestra_B
## A = 0.18654, p-value = 0.9048
```

-Test Shapiro-Wilks(S-W)

```
shapiro.test(muestra_B)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  muestra_B
## W = 0.99905, p-value = 0.9375
```

```
agostino.test(muestra_B)
```

```
##
##  D'Agostino skewness test
##
## data:  muestra_B
## skew = -0.018649, z = -0.228993, p-value = 0.8189
## alternative hypothesis: data have a skewness
```

En los cuatro tests el p-valor era mayor que 0.05. Por ello no tenemos suficientes evidencias para rechazar la hipótesis nula.

En conclusión, probando con los cuatro tests, podemos afirmar que las muestras A y B son normales.

Apartado 3

Calcularemos manualmente el estadístico de contraste del test K-S-L para la muestra A usando la seguiente formula:

$$D_n(x_i) = \max\left\{\left|F_X(x_i) - \frac{i-1}{n}\right|, \left|F_X(x_i) - \frac{i}{n}\right|\right\}$$

Ordenamos la muestra

```
muestra_A_ordenada <- sort(muestra_A)
```

Calculamos la función de distribución empírica (ECDF)

```
ecdf_A <- ecdf(muestra_A_ordenada)

mu <- mean(muestra_A)
sigma <- sd(muestra_A)
cdf_teorica <- pnorm(muestra_A_ordenada, mean = mu, sd = sigma)
diferencias <- abs(ecdf_A(muestra_A_ordenada) - cdf_teorica)
D_n <- max(diferencias)

D_n
```

```
## [1] 0.02662392
```

Para comprobar nuestro cálculo obtenemos el mismo contraste con R:

```
lillie.test(muestra_A)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  muestra_A
## D = 0.026624, p-value = 0.09182
```

Efectivamente, el resultado es identico.Y por lo tanto podemos afirmar que no se puede rechazar la hipótesis nula, ya que el p-valor es mayor que 0.05

Apartado 4

Constraste de hipótesis:

$$H_0 : \sigma_A^2 = \sigma_B^2$$
$$H_1 : \sigma_A^2 \neq \sigma_B^2$$

El estadístico que usaremos:

$$f_0 = \frac{\tilde{S_1}^2}{\tilde{S_2}^2}$$

```
var_muestra_A = var(muestra_A)
var_muestra_B = var(muestra_B)
media_muestra_A = mean(muestra_A)
media_muestra_B=mean(muestra_B)
desv_muestra_A=sd(muestra_A)
desv_muestra_B=sd(muestra_B)
f_0 = var_muestra_A/var_muestra_B
f_0
```

```
## [1] 1.722912
```

Calculamos ahora el p-valor empleando la seguiente formula:

$$\text{p-valor: } \min\left\{2 \cdot P\left(F_{n_1-1,n_2-1} \leq f_0\right), 2 \cdot P\left(F_{n_1-1,n_2-1} \geq f_0\right)\right\}.$$

```
n_A=1000
n_B = 890
p_valor_manual= min(2*pf(f_0,n_A-1,n_B-1),2*pf(f_0,n_A-1,n_B-1,lower.tail = FALSE))
p_valor_manual
```

```
## [1] 0.00000000000000001632253
```

Test de Fisher usando R

```
var.test(muestra_A,muestra_B,alternative ="two.sided" )
```

```
##
##  F test to compare two variances
##
## data:  muestra_A and muestra_B
```

```
## F = 1.7229, num df = 999, denom df = 889, p-value < 0.00000000000000022
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  1.515729 1.957468
## sample estimates:
## ratio of variances
##           1.722912
```

Como que el p-valor es demasiando pequeño podemos rechazar la hipótesis nula.

Ahora usando el test Fligner

```
fligner.test(list(muestra_A,muestra_B))
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  list(muestra_A, muestra_B)
## Fligner-Killeen:med chi-squared = 43.044, df = 1, p-value =
## 0.00000000005353
```

Con el test de Flinger obtenemos también un p-valor muy pequeño que nos permite rechazar la hipótesis nula.

Apartado 5

Constraste de hipótesis:

$$H_0 : \mu_A = \mu_B$$
$$H_1 : \mu_A > \mu_B$$

calcularemos el seguiente estadístico

$$: T = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{\tilde{S}_1^2}{n_1} + \frac{\tilde{S}_2^2}{n_2}}}$$

```
estadistico_T2 = (media_muestra_A-media_muestra_B)/(sqrt((desv_muestra_A^2/n_A)+(desv_muestra_B^2/n_B))
estadistico_T2
```

```
## [1] 0.354286
```

Calcularemos el p-valor usando la seguiente fórumula

$$p = P(Z \geq z_0)$$

```
p_valor_2 = pt(estadistico_T2,df = n_A+n_B-2, lower.tail = FALSE)
p_valor_2
```

```
## [1] 0.3615821
```

```
t.test(muestra_A,muestra_B,alternative="greater")
```

```
##
##  Welch Two Sample t-test
##
## data:  muestra_A and muestra_B
## t = 0.35429, df = 1845.1, p-value = 0.3616
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  -0.2060502        Inf
## sample estimates:
## mean of x mean of y
##  12.17585  12.11932
```

Efectivamente, el p-valor calculado manualmente y el que da el test de t Student son idénticos. Como que el p-valor es mayor que el valor de significancia no podemos rechazar al hipótesis nula. Dicho de otra forma, no tenemos evidencias suficientes para afirmar que la nueva interfaz (B) mejora el tiempo que la actual.

Apartado 6 Intervalos de confianza:

El test Fisher para el apartado 4

```
var.test(muestra_A,muestra_B,alternative ="two.sided" )$conf.int
```

```
## [1] 1.515729 1.957468
## attr(,"conf.level")
## [1] 0.95
```

-Manualamente

```
Intervalo_confinaza3=c(f_0*qf(0.05/2,n_A-1,n_B-1),f_0*qf(0.975,n_A-1,n_B-1))
Intervalo_confinaza3
```

```
## [1] 1.516461 1.958413
```

Para el apartado 5 test de medias:

```
Intervalo_confinaza4=c((media_muestra_A-media_muestra_B)-qt((1-0.05),n_A+n_B-2)* sqrt((var_muestra_A/n_A
Intervalo_confinaza4
```

```
## [1] -0.2060472        Inf
```

con R

```
t.test(muestra_A,muestra_B,alternative="greater")$conf.int
```

```
## [1] -0.2060502        Inf
## attr(,"conf.level")
## [1] 0.95
```

En conclusión el intervalo de confianza para el test de las medias incluye el valor 1 por lo tanto no se puede rechazar la hipótesis nula. En cuanto al intervalo de confianza de las varianzas como que no incluye el 0 se puede rechazar la hipótesis nula.

1.2 Problema 2: Bondad de ajuste. La ley de Benford. (4 puntos)

La ley de Benford es una distribución discreta que siguen las frecuencias de los primero dígitos significativos (de 1 a 9) de algunas series de datos curiosas.

Sea una v.a. X con dominio $D_X = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ diremos que sigue una ley de Benford si

$$P(X = x) = \log_{10}\left(1 + \frac{1}{x}\right) \text{ para } x \in \{1, 2, 3, 4, 5, 6, 7, 8, 9\}.$$

Concretamente las porobabilidades son

```
## [1] 0.30103000 0.17609126 0.12493874 0.09691001 0.07918125 0.06694679 0.05799195
## [8] 0.05115252 0.04575749
```

	Díg. 1	Díg. 2	Díg. 3	Díg. 4	Díg. 5	Díg. 6	Díg. 7	Díg. 8	Díg. 9
prob	0.30103	0.1760913	0.1249387	0.09691	0.0791812	0.0669468	0.0579919	0.0511525	0.0457575

En general esta distribución se suele encontrar en tablas de datos de resultados de observaciones de funciones científicas, contabilidades, cocientes de algunas distribuciones …

1. Contrastar con un test χ^2 si el primer dígito significativo de las cubos de los números naturales del 1 al 1000 sigue esa distribución.
2. Contrastar con un test χ^2 si que el segundo dígito significativo de los cubos los números naturales del 1 al 1000 sigue una uniforme discreta de los diez dígitos del 0 al 9.
3. Calcular manualmente el estadístico y el p-valor del los dos contrates anteriores.
4. Dibujad con `R` para los apartados 1 y 2 los diagramas de frecuencias esperados y observados. Comentad estos gráficos.

##Solución

Apartado 1 El constraste e estudiar:

$$\begin{cases} H_0: & \text{El primer dígito de los cubos de los primeros 1000 números naturales sigue una distribución Benford,} \\ H_1: & \text{sigue cualquier otra distribución.} \end{cases}$$

-Generamos primero los cubos de los numeros de 1 al 1000

```
cubos <- str_sub(as.character(c(1:1000)^3),1,2)
len = length(cubos)
```

Extracción del primer dígito significativo de cada cubo

```
primeros_digitos <- as.numeric(substr(as.character(cubos), 1, 1))
```

Tabla de frecuencias observadas/empíricas

```r
frec_emp_primero = table(primeros_digitos)
frec_emp_primero
```

```
## primeros_digitos
##   1   2   3   4   5   6   7   8   9
## 226 159 124 106  94  83  74  71  63
```

Cálculo de frecuencias esperadas según la ley de Benford

```r
frec_esp_primero = (1000 * prob)
frec_esp_primero
```

```
## [1] 301.03000 176.09126 124.93874  96.91001  79.18125  66.94679  57.99195
## [8]  51.15252  45.75749
```

Contraste de ^{2}

```r
chisq.test(frec_emp_primero, p = prob)
```

```
##
##  Chi-squared test for given probabilities
##
## data:  frec_emp_primero
## X-squared = 46.459, df = 8, p-value = 0.0000001944
```

Como que el p-value es bastante pequeño podemos rechazar la hipótesis nula. Por lo tanto el primer dígito de los primeros 1000 cubos no sigue la distribución de Benford.

Apartado 2

Constraste:

$$\begin{cases} H_0: & \text{El segundp dígito de los cubos de los primeros 1000 números naturales sigue una distribución Benford,} \\ H_1: & \text{sigue cualquier otra distribución.} \end{cases}$$

Extracción del segundo dígito significativo de cada cubo

```r
segundos_digitos <- as.numeric(substr(as.character(cubos), 2, 2))
```

Tabla de frecuencias observadas

```r
frec_emp_segundo <- table(segundos_digitos)
frec_emp_segundo
```

```
## segundos_digitos
##   0   1   2   3   4   5   6   7   8   9
## 115 109 106  98 104  97  91  99  89  90
```

Cálculo de frecuencias esperadas (uniforme discreta)

```r
frec_exp_segundo <- rep(100, 10)
frec_exp_segundo
```

```
##  [1] 100 100 100 100 100 100 100 100 100 100
```

Contraste de ^{2}

```r
chisq.test(frec_emp_segundo, p = rep(1/10, 10))
```

```
##
##  Chi-squared test for given probabilities
##
## data:  frec_emp_segundo
## X-squared = 6.7495, df = 9, p-value = 0.6632
```

Como que el p-valor>0.05 no tenemos suficientes evidencias para rechazar la hipótesis nula. Por lo tanto el segundo dígito de los cubos de los 1000 primeros numeros naturales sigue una distribución uniforme.

Apartado 3 Vamos a calcular manualemente el estadístico y el p-valor de los apartados 1 y 2.

El estadístico a calcular es el seguiente :

$$\chi^2 = \sum_{i=1}^{k} \frac{(\text{ frec. empíricas }_i - \text{ frec. teóricas }_i)^2}{\text{frec. teóricas }_i} = \sum_{i=1}^{k} \frac{(n_i - e_i)^2}{e_i}$$

```r
chi2_esperado = sum((frec_emp_primero-frec_esp_primero)^2/frec_esp_primero)
chi2_esperado
```

```
## [1] 46.45925
```

Y nuestro p-valor se calcula usando la seguiente formula:

$$p = P\left(\chi^2_{k-1} > \chi_0\right)$$

```r
pchisq(chi2_esperado,8,lower.tail = FALSE)
```

```
## [1] 0.0000001943865
```

Para el segundo apartado:

Repetimos los mismos cálculos

```r
chi2_esperado2 = sum((frec_emp_segundo-frec_exp_segundo)^2/frec_exp_segundo)
chi2_esperado2
```

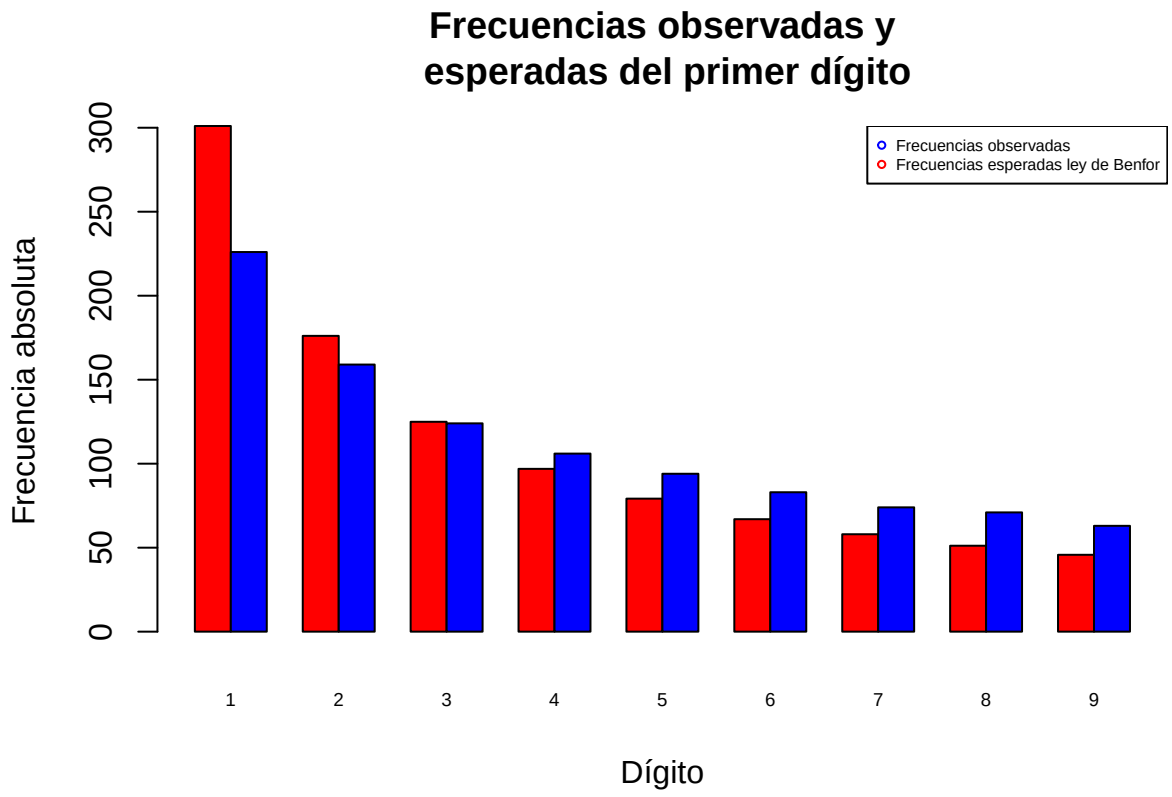
```
## [1] 6.74
```

```r
pchisq(chi2_esperado2,9,lower.tail = FALSE)
```

```
## [1] 0.6641684
```

Apartado 4 Diagrama de barras para el primer dígito significativo.

```r
barplot(rbind(frec_esp_primero,frec_emp_primero),
beside=TRUE,col=c("red","blue"),
main="Frecuencias observadas y\n esperadas del primer dígito",
cex.names =0.6,xlab="Dígito",ylab="Frecuencia absoluta")
legend("topright",legen=c("Frecuencias observadas",
"Frecuencias esperadas ley de Benfor"),pch=1,col=c("blue","red"),
cex=0.5)
```

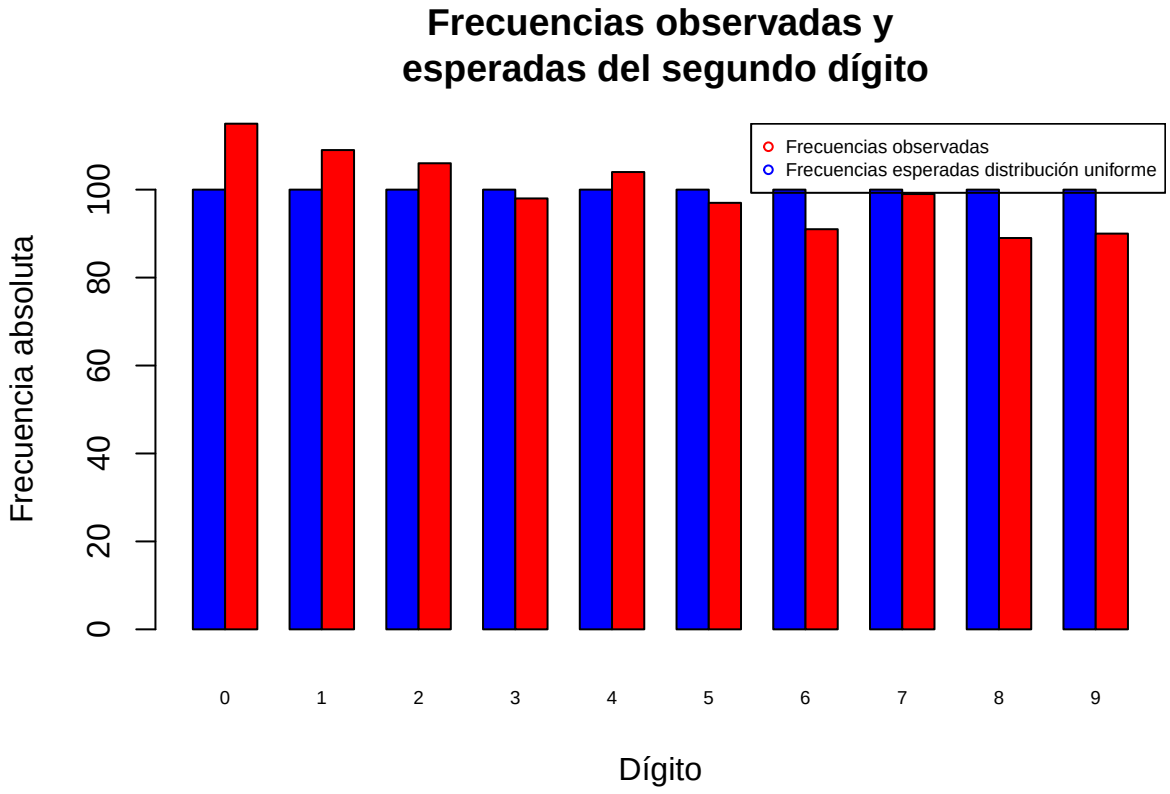


Como se puede ver en el diagrama la diferencia entre las frecuencias esperas y teorias del primero apartado, llevando para el dígito 3, es bastante grandey esta es la razón prinicipal por la cúal se rechaza la hipótesis nula.

Diagrama de barras para el segundo dígito significativo

```r
barplot(rbind(frec_exp_segundo,frec_emp_segundo),
beside=TRUE,col=c("blue","red"),
main="Frecuencias observadas y\n esperadas del segundo dígito",
cex.names =0.6,xlab="Dígito",ylab="Frecuencia absoluta")
legend("topright",legen=c("Frecuencias observadas",
```

```
"Frecuencias esperadas distribución uniforme"),pch=1,col=c("red","blue"),
cex=0.6)
```



1.3 Problema 3: Homegeneidad e independencia. (3 puntos).

Queremos analiza los resultados de aprendizaje con tres tecnologías. Para ello se seleccionan grupos de 4 Grados (Grado1, Grado2, Grado3, y Grado4) de 50 estudiantes y se les somete a evaluación después de un curso que se encuentran en los datos adjuntos `datasets/tecnologias_4_grados.csv`.

Se pide

1. Discutid si hacemos un contraste de independencia o de homogeneidad de las distribuciones de las notas por tecnología. Escribid las hipótesis del contraste.
2. Interpretad la función `chisq.test` y resolved el contraste.
3. Calculad las frecuencias teóricas como producto de los vectores marginales y calculad el estadístico de contraste y el p-valor.

##Solución **Apartado 1** Primero definimos que es cada constraste.

En un contraste de independencia se toma una muestra transversal de la población, es decir, se selecciona al azar una cierta cantidad de individuos de la población, se observan las dos variables sobre cada uno de ellos, y se contrasta si las probabilidades conjuntas son iguales al producto de las probabilidades marginales de cada variable.

Mientras que en un contraste de homogeneidad se escoge una de las variables y para cada uno de sus posibles valores se toma una muestra aleatoria, de tamaño prefijado, de individuos con ese valor para esa variable.

En mi caso usaré un constastre de homogneidad ya que tenemos una variable (nota) y para cada uno de sus posibles valores se toma una muestra de tamaño prefijado(50), de individuos con ese valor para esa variable (grupos de estudiantes).

Escribimos ahora el constraste planteado:

$$\begin{cases} H_0: & \text{La distribución de la variable condicional nota es la misma para cualquier tecnología} \\ H_1: & \text{La distribución de la variable condicional nota no es la misma para cualquier tecnología} \end{cases}$$

Apartado 2 Para usar chisp.tst necesitamos la tabla de contigencia que se obtiene de la seguiente forma: Primero cargamos el documento de datos y usamos la función table para crear nuestra tabla de contingencia.

```
datos = read.csv("datasets/tecnologias_4_grados.csv")
fre_abs <- table(datos$tecnologia, datos$nota)
fre_abs
```

```
##
##           A  E  N  S
##   Python 21 14 10  5
##   R      18 11 17  4
##   Stata  19 15 13  3
```

```
chisq.test(fre_abs)
```

```
## Warning in chisq.test(fre_abs): Chi-squared approximation may be incorrect
```

```
##
##  Pearson's Chi-squared test
##
## data:  fre_abs
## X-squared = 3.2414, df = 6, p-value = 0.778
```

Como que el p-valor es bastante mayor concluimos que no tenemos evidencias suficientes para rechazar la hipótesis nula.

Apartado 3 Cálculo de las frecuencias teoricas

```
suma_filas = rowSums(fre_abs)
suma_cols =colSums(fre_abs)

N = sum(fre_abs)

fre_teoricas <-suma_filas %*%t(suma_cols)/N
fre_teoricas
```

```
##             A        E        N S
## [1,] 19.33333 13.33333 13.33333 4
## [2,] 19.33333 13.33333 13.33333 4
## [3,] 19.33333 13.33333 13.33333 4
```

El estadístico de constraste es :

```r
valor_chi2 = sum((fre_abs-fre_teoricas)^2 /fre_teoricas)
valor_chi2
```

```
## [1] 3.241379
```

```r
dim(fre_abs)
```

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

```r
pvalor <- pchisq(valor_chi2,df=(3-1)*(4-1),lower.tail=FALSE)
pvalor
```

```
## [1] 0.7779982
```

Como que el p-valor es mayor que 0.05 concluimos que no tenemos evidencias suficientes para rechazar la hipótesis nula (igual que en el aprtado 2).

1.4 Problema 4: Contraste de proporciones de dos muestras independientes. (3. puntos)

Queremos comparar las proporciones de aciertos de dos redes neuronales que detectan si una foto con un móvil de una avispa es una avispa velutina o asiática o si es una avispa común. Esta avispa en una especie invasora y peligrosa por el veneno de su picadura. Para ello disponemos de una muestra de 1000 imágenes de insectos etiquetadas como avispa velutina y no velutina.

Aquí tenéis el acceso a los datos. Cada uno está en fichero selecciona 500 fotos de de forma independiente para el algoritmo 1 y el 2. Los aciertos están codificados con 1 y los fallos con 0.

Se pide:

1. Cargad los datos desde el servidos y calcular el tamaño de las muestras y la proporción de aciertos de cada muestra.
2. Contrastad si hay evidencia de que las las proporciones de aciertos del algoritmo 1 son mayores que las del algoritmo 2. Definid bien las hipótesis y las condiciones del contraste. Tenéis que hacer el contraste con funciones de `R` y resolver el contrate con el p-valor.
3. Calculad el intervalo de confianza para la diferencia de proporciones **pág 187 tema 4: CH** que vimos de forma manual en teoría.

##Solución 4 **Apartado 1**

Cargamos los datos

```r
algoritmo1= read.table("https://bioinfo.uib.es/~recerca/MATIIIGINF/velutina/algoritmo1.csv")
algoritmo2 = read.table("https://bioinfo.uib.es/~recerca/MATIIIGINF/velutina/algoritmo2.csv")
```

Tamaño de las muestras y proporción de aciertos

Muestra 1:

```r
n1 = length(algoritmo1$V1)
n1
```

```
## [1] 500
```

```r
P1 = prop.table(table(algoritmo1))["1"]
P1
```

```
##     1
## 0.792
```

```r
aciertos1 = P1*n1
aciertos1
```

```
##   1
## 396
```

Muestra 2:

```r
n2 = length(algoritmo2$V1)
n2
```

```
## [1] 500
```

```r
P2 = prop.table(table(algoritmo2))["1"]
P2
```

```
##     1
## 0.874
```

```r
aciertos2 = P2*n2
aciertos2
```

```
##   1
## 437
```

Apartado 2

Sean p1 y p2 las proporciones de aciertos de los algoritmos 1 y 2 respectivamente, el constraste que se pide es el siguiente:

$$\begin{cases} H_0: & p_1 = p_2 \\ H_1: & p_1 > p_2 \end{cases}$$

Para ello obtenemos primero la matriz de aciertos de las dos muestras

```r
X = matrix(c(aciertos1,aciertos2, n1-aciertos1,n2-aciertos2),nrow=2,byrow = TRUE)
X
```

```
##      [,1] [,2]
## [1,]  396  437
## [2,]  104   63
```

Como que nuestras muestras son relativamente grandes el más apropiado es usar el prop.test

```
prop.test(c(aciertos1,aciertos2),c(n1,n2),alternative="greater",conf.level = 0.95)
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  c(aciertos1, aciertos2) out of c(n1, n2)
## X-squared = 11.502, df = 1, p-value = 0.9997
## alternative hypothesis: greater
## 95 percent confidence interval:
##  -0.1225654  1.0000000
## sample estimates:
## prop 1 prop 2
##  0.792  0.874
```

Dado que el p-value es bastante grande no podemos rechazar la hipótesis nula. Por lo tanto no tenemos suficientes evidencias para concluir que la proporción de aciertos del algoritmo 1 es mayor que la del algoritmo2.

Apartado 3 Vamos a calcular el intervalo de confianza para p1-p2 usando la seguiente formúla:

$$\left(\hat{p}_1 - \hat{p}_2 - z_{1-\frac{\alpha}{2}}\sqrt{\left(\frac{n_1\hat{p}_1 + n_2\hat{p}_2}{n_1+n_2}\right)\left(1-\frac{n_1\hat{p}_1 + n_2\hat{p}_2}{n_1+n_2}\right)\left(\frac{1}{n_1}+\frac{1}{n_2}\right)}, \hat{p}_1 - \hat{p}_2 + z_{1-\frac{\alpha}{2}}\sqrt{\left(\frac{n_1\hat{p}_1 + n_2\hat{p}_2}{n_1+n_2}\right)\left(1-\frac{n_1\hat{p}_1 + n_2\hat{p}_2}{n_1+n_2}\right)\left(\frac{1}{n_1}+\frac{1}{n_2}\right)}\right)$$

Pero para un contraste unilateral que es el que se da en nuestro caso, calcularemos el seguiente intervalo:

$$\left(\hat{p}1 - \hat{p}_2 - z1 - \frac{\alpha}{2}\sqrt{\left(\frac{n_1\hat{p}_1 + n_2\hat{p}_2}{n_1+n_2}\right)\left(1-\frac{n_1\hat{p}_1 + n_2\hat{p}_2}{n_1+n_2}\right)\left(\frac{1}{n_1}+\frac{1}{n_2}\right)}, \infty)$$

Calcularemos primero el factor dentro de la raíz para simplificar el cálculo

```
factor1 = (aciertos1+ aciertos2)/(n1+n2)
element = factor1*(1-factor1)*((1/n1)+(1/n2))
```

```
intervalo_conf=c(P1-P2-(qnorm((1-(0.05/2))) *sqrt(element)),Inf)

intervalo_conf
```

```
##          1
## -0.1282337        Inf
```

1.5 Problema 5 : Contraste de proporciones de dos muestras emparejadas. (2. puntos)

En el problema anterior hemos decidido quedarnos con el mejor de los algoritmos y mejorarlo. Pasamos las mismas 1000 imágenes a la version_beta del algoritmo y a la version_alpha. Aquí tenéis el acceso a los datos en el mismo orden para las 1000 imágenes. Cada uno está en fichero los aciertos están codificados con 1 y los fallos con 0.

1. Cargad los datos desde el servidos y calcular el tamaño de las muestras y la proporción de aciertos de cada muestra.
2. Contrastad si hay evidencia de que las proporciones de aciertos del algoritmo alfa son iguales que las del algoritmo beta. Definid bien las hipótesis y las condiciones del contraste. De forma manual como se explicó en **teoría pág 246 tema 4: CH** y resolver con el p-valor.

##Solución 5

Apartadp 1 Cargamos los datos usando la fución read.table de R

```
algoritmo_alpha= read.table("https://bioinfo.uib.es/~recerca/MATIIIGINF/velutina2/algoritmo_alpha.csv")
algoritmo_beta = read.table("https://bioinfo.uib.es/~recerca/MATIIIGINF/velutina2/algoritmo_beta.csv")
```

Calcularemos el tamaño de las muestras y proporción de aciertos para : Muestra Alpha:

```
n_alpha = length(algoritmo_alpha$V1)
n_alpha
```

```
## [1] 1000
```

```
PA_alpha = prop.table(table(algoritmo_alpha))["1"]
PA_alpha
```

```
##     1
## 0.875
```

Muestra Beta:

```
n_beta = length(algoritmo_beta$V1)
n_beta
```

```
## [1] 1000
```

```
PA_beta = prop.table(table(algoritmo_beta))["1"]
PA_beta
```

```
##     1
## 0.897
```

Apartado 2 Se nos pide contrastar la seguiente hipótesis:

$$\begin{cases} H_0: & p_\alpha = p_\beta \\ H_1: & p_\alpha \neq p_\beta \end{cases}$$

Creamos primero la matriz y luego usaremos el test de mcnemar

```r
matriz = table(algoritmo_alpha$V1, algoritmo_beta$V1)
mcnemar.test(matriz)
```

```
##
##  McNemar's Chi-squared test with continuity correction
##
## data:  matriz
## McNemar's chi-squared = 2.2273, df = 1, p-value = 0.1356
```

Como que p-value es mayor que 0.05 no podemos rechazar la hipótesis nula.Es decir no podemos rechazar la igualdad de p_alpha p_beta.