Taller 2 entrega problema en grupo. MAT3 (estadística) GIN2 2020-2021 - Estadística inferencial mayo 2021.

nombre1, apellido1_1 apellido1_22; nombre2, apellido2_1 apellido2_2;...

Contenidos

1 Taller 2 evaluable. Entrega de problemas

Taller en grupo entregad las soluciones en .Rmd y .html o .pdf. o escribirlas de forma manual y escanear el resultado, en un solo fichero. Cada apartado vale 1 punto en total hay 15 puntos y se pondera la 10 puntos.

1.1 Problema 1

a. Consideremos la siguiente muestra aleatoria simple de una v.a. continua X: $-3, -2, -1, 0, 0, 1, 2, 3, 4$ de tamaño $n = 9$. Calcular, en esta muestra, el error estándar de estadístico media aritmética de la muestra.
b. Consideremos la siguiente muestra aleatoria simple de tamaño $n = 10$ de una v.a. X con distribución $Ber(p)$: $1, 0, 1, 0, 1, 1, 1, 1, 1, 0$ Calcular, en esta muestra, el estadístico proporción muestral y su error estándar.
c. Suponiendo que la población es normal calcular un intervalo de confianza del 95% para μ_X.
d. Suponiendo que la población es normal calcular un intervalo de confianza del 95% para σ_X^2.

Ayuda de R, acabad vosotros los cálculos

```
muestra1=c(-3,-2,-1,0,0,1,2,3,4)
mean(muestra1)
```

```
## [1] 0.4444444
```

```
sum(muestra1)
```

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

```
sum(muestra1^2)
```

```
## [1] 44
```

```
n=length(muestra1)
n
```

```
## [1] 9
```

```
muestra2=c(1,0,1,0,1,1,1,1,1,0)
table(muestra2)
```

```
## muestra2
## 0 1
## 3 7
```

```
length(muestra2)
```

```
## [1] 10
```

1.1.1 Solución

1.2 Problema 2

Queremos comparar los rendimientos medidos en consumo de CPU de dos configuraciones (C1 y C2) de un servidor de datos tienen una media similar,de hecho queremos tener evidencia contra que el rendimiento medio del servidor C1 es superior al del servidor C2. No conocemos σ_1 y σ_2. Disponemos de dos muestras independientes de consumo por hora realizados para cada configuración C1 y C2, de tamaños $n_1 = n_2 = 100$, respectivamente.

Para bajarlos utilizad la dirección del los ficheros `raw` que se muestran en el siguiente código

```
C1=read.csv(
  "https://raw.githubusercontent.com/joanby/estadistica-inferencial/master/datasets/C1.csv",
            header=TRUE)$time
C2=read.csv(
  "https://raw.githubusercontent.com/joanby/estadistica-inferencial/master/datasets/C2.csv",
  header=TRUE)$time

n1=length(na.omit(C1))
n1
```

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

```
n2=length(na.omit(C2))
n2
```

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

```
media.muestra1=mean(C1,na.rm=TRUE)
media.muestra1
```

```
## [1] 38.5841
```

```
media.muestra2=mean(C2,na.rm=TRUE)
media.muestra2
```

```
## [1] 33.7953
```

```
desv.tip.muestra1=sd(C1,na.rm=TRUE)
desv.tip.muestra1
```

```
## [1] 3.014567
```

```
desv.tip.muestra2=sd(C2,na.rm=TRUE)
desv.tip.muestra2
```

```
## [1] 6.727062
```

Calculamos las medias y las desviaciones típicas muestrales de los tiempos empleados para cada muestra. Los datos obtenidos se resumen en la siguiente tabla:

$$
\begin{aligned}
n_1 &= 100, & n_2 &= 100\\
\overline{x}_1 &= 38.5841, & \overline{x}_2 &= 33.7953\\
\tilde{s}_1 &= 3.014567, & \tilde{s}_2 &= 6.7270621
\end{aligned}
$$

Se pide:

a. Comentad brevemente el código de R explicando que hace cada instrucción.
b. Contrastad si hay evidencia de que los rendimientos medios son distintas entre los dos grupos. En dos casos considerando las varianzas desconocidas pero iguales o desconocidas pero distintas. Tenéis que hacer el contraste de forma manual y con funciones de `R` y resolver el contrate con el p-valor.
c. Calculad e interpretar los intervalos de confianza BILATERALES al nivel de confianza del 95% para la diferencia de medias de los rendimientos en los casos anteriores.
d. Comprobad con el test de Fisher y el de Levene 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 test de Levene con `R` y decidir utilizando el p-valor.

1.2.1 Solución

1.3 Problema 3

Se prueba la misma implementación de una algoritmo para reconocer caras de la base de datos de una empresa con dos diferente tipos de cámaras.

Para ello $n = 100$ trabajadores pasan por cada una de las cámaras 1 vez.

Los resultados se pueden cargar con el siguiente código.

```
caras=read.csv(
  "https://raw.githubusercontent.com/joanby/estadistica-inferencial/master/datasets/caras.csv",
            header=TRUE)
str(caras)
```

```
## 'data.frame':    100 obs. of  3 variables:
##  $ empleado: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ aciertoA: int  0 1 1 1 1 1 1 1 1 1 ...
##  $ aciertoB: int  1 1 1 1 1 1 1 1 1 1 ...
```

```
table(caras$aciertoA,caras$aciertoB)
```

```
##
##      0  1
##   0  0 12
##   1  1 87
```

Donde `empleadop` es la variable el identificador del empleado y `aciertoA` y `aciertoB` valen 1 si se acierta la identidad y 0 si se falla para el mismo empleado en cada una de las cámaras.

Se pide:

a. Cargad los datos desde el servidos y calcular el tamaño de las muestras y la proporción de aciertos de cada muestra.
b. Contrastad si hay evidencia de que las las proporciones de aciertos con la cámara A son iguales que las del algoritmo con la cámara . Definid bien las hipótesis y las condiciones del contraste. Resolver el contraste de forma manual utilizando `R` solo como calculadora y resolver el contraste con el p-valor (calculado con R).
c. Resolver el contraste con funciones de R.
d. Calcular un intervalo de confianza bilateral para la diferencia de la proporciones al nivel de confianza del 95% con R y de forma manual utilizando R como calculadora y para calcular los cuantiles.

1.3.1 Solución

1.4 Problema 4

El encargado de calidad piensa que X = número de quejas de clientes por día en las oficinas de atención al cliente de una determinada zona de una ciudad sigue una ley $Po(\lambda = 5)$. Para comprobarlo toma una muestra de $n = 100$ días:

```
quejas=read.csv(
  "https://raw.githubusercontent.com/joanby/estadistica-inferencial/master/datasets/quejas.csv",
           header=TRUE)
str(quejas)
```

```
## 'data.frame':    100 obs. of  1 variable:
##  $ Num_quejas: int  4 6 4 2 6 2 7 10 7 4 ...
```

```
ni=c(0,table(quejas))
names(ni)[1]="0"
ni
```

```
##  0  1  2  3  4  5  6  7  8  9 10 11
##  0  1  8 11 16 16 14 14 11  4  4  1
```

```
n=sum(ni)
n
```

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

```
pi=c(dpois(0:10,lambda=5),1-sum(dpois(0:10,lambda=5)))
names(pi)=c(paste0("Prob(X=",0:10,")"),"Prob(X>=11)")
pi
```

```
##   Prob(X=0)   Prob(X=1)   Prob(X=2)   Prob(X=3)   Prob(X=4)   Prob(X=5)
## 0.006737947 0.033689735 0.084224337 0.140373896 0.175467370 0.175467370
##   Prob(X=6)   Prob(X=7)   Prob(X=8)   Prob(X=9)  Prob(X=10) Prob(X>=11)
## 0.146222808 0.104444863 0.065278039 0.036265577 0.018132789 0.013695269
```

```
sum(pi)
```

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

```
ei=n*pi
ei
```

```
##   Prob(X=0)   Prob(X=1)   Prob(X=2)   Prob(X=3)   Prob(X=4)   Prob(X=5)
##   0.6737947   3.3689735   8.4224337  14.0373896  17.5467370  17.5467370
##   Prob(X=6)   Prob(X=7)   Prob(X=8)   Prob(X=9)  Prob(X=10) Prob(X>=11)
##  14.6222808  10.4444863   6.5278039   3.6265577   1.8132789   1.3695269
```

```
ei>5
```

```
##   Prob(X=0)   Prob(X=1)   Prob(X=2)   Prob(X=3)   Prob(X=4)   Prob(X=5)
##       FALSE       FALSE        TRUE        TRUE        TRUE        TRUE
##   Prob(X=6)   Prob(X=7)   Prob(X=8)   Prob(X=9)  Prob(X=10) Prob(X>=11)
##        TRUE        TRUE        TRUE       FALSE       FALSE       FALSE
```

```
# no se cumple la condición para el test chi^2
#hay que agrupar los 3 primeros y los 3 últimos
# test chi^2 sin agrupar...
chi0=sum((ei-ni)^2/ei)
chi0
```

```
## [1] 10.36668
```

```r
k=12# clases   dce 0 a mayor o igual 11
k=12# clases de 0 a 11
pchisq(chi0,df=k-1,lower.tail=FALSE)
```

```
## [1] 0.4977365
```

Se pide:

a. Plantead un contraste de bondad de ajuste χ^2 H_0: los datos siguen una distribución $Po(\lambda = 5)$. Calculas las probabilidades y frecuencias esperadas utilizando los datos del código anterior.
b. Reagrupar los datos y resolver el test manualmente pero usando R para el cálculo del p-valor. Resolver el contraste
c. Resolver el contraste con la función adecuada de R.

1.4.1 Solución

```r
ni
```

```
##  0  1  2  3  4  5  6  7  8  9 10 11
##  0  1  8 11 16 16 14 14 11  4  4  1
```

```r
pi
```

```
##   Prob(X=0)   Prob(X=1)   Prob(X=2)   Prob(X=3)   Prob(X=4)   Prob(X=5)
## 0.006737947 0.033689735 0.084224337 0.140373896 0.175467370 0.175467370
##   Prob(X=6)   Prob(X=7)   Prob(X=8)   Prob(X=9)  Prob(X=10) Prob(X>=11)
## 0.146222808 0.104444863 0.065278039 0.036265577 0.018132789 0.013695269
```

```r
chisq.test(ni,p=pi)
```

```
## Warning in chisq.test(ni, p = pi): Chi-squared approximation may be incorrect
```

```
##
##  Chi-squared test for given probabilities
##
## data:  ni
## X-squared = 10.367, df = 11, p-value = 0.4977
```

```r
ni_agrupado=c(sum(ni[1:3]),ni[4:9],sum(ni[10:12]))
pi_agrupado=c(sum(pi[1:3]),pi[4:9],sum(pi[10:12]))
sum(pi_agrupado)
```

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

```r
n=sum(ni)
n
```

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

```r
n*pi_agrupado>=5
```

```
##           Prob(X=3) Prob(X=4) Prob(X=5) Prob(X=6) Prob(X=7) Prob(X=8)
##      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE
```

```r
chisq.test(ni_agrupado,p = pi_agrupado)
```

```
##
##  Chi-squared test for given probabilities
##
```

```
## data:  ni_agrupado
## X-squared = 6.8987, df = 7, p-value = 0.4395
```

```r
ei_agrupado=n*pi_agrupado
sum((ni_agrupado-ei_agrupado)^2/ei_agrupado)
```

```
## [1] 6.898705
```