Tarea sobre los Temas 3 a 5

Grado en Estadística. Universidad de Sevilla

Marta Venegas Pardo

Contents

1 Pregunta 1 (datasets::precip)

En el dataset datasets::precip se recoge la cantidad media de precipitaciones de 70 ciudades de Estados Unidos (unidad = inches).

Extraigo los datos

```
precip<-datasets::precip
# View(precip)
```

```
#dotchart(precip[order(precip)], main = "precip data")
#title(sub = "Average annual precipitation (in.)")
```

```
precip_data<-as.data.frame(precip)
names<-names(precip)
precipdf<-cbind.data.frame(names,precip_data)
colnames(precipdf)<-c("City","Precip")



cbind.data.frame(precipdf[1:35,],precipdf[36:70,])%>%
  kable(booktabs=TRUE,longtable=TRUE) %>%
  kable_styling(latex_options = c("striped","scale_down")) %>%
  footnote(general = "Annual Precipitation in US Cities")
```

```
## Warning in styling_latex_scale_down(out, table_info): Longtable cannot be
## resized.
```

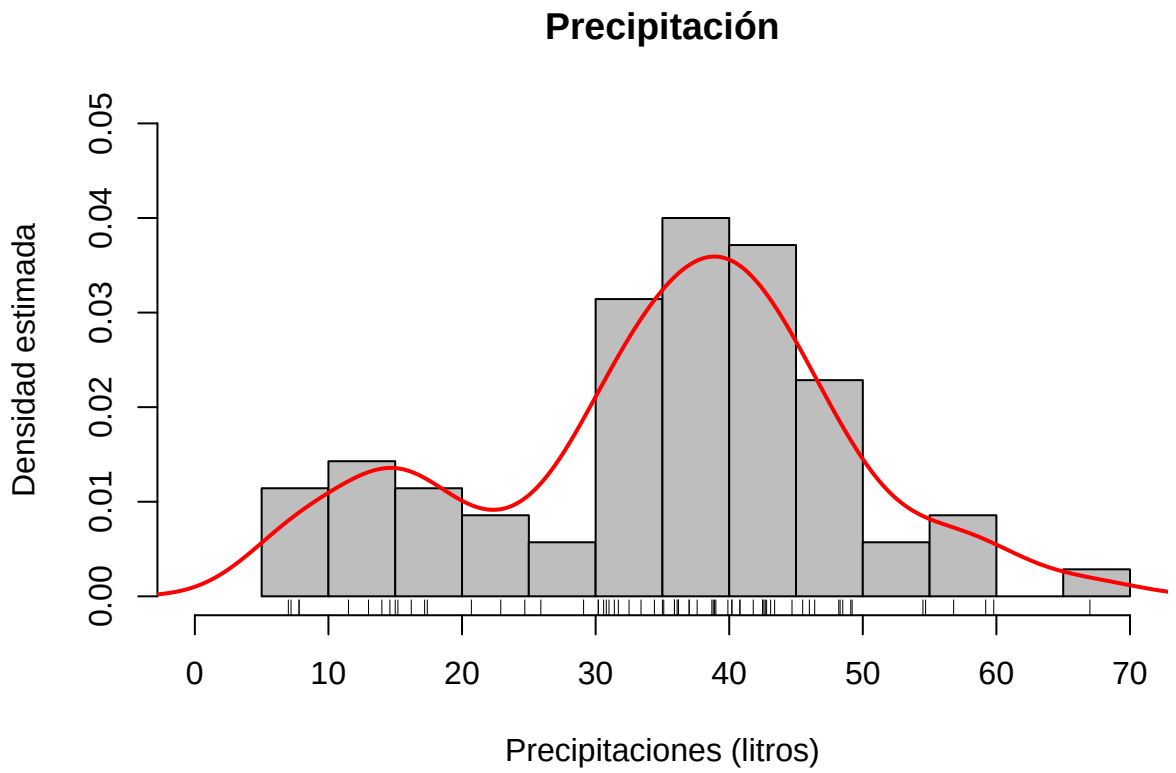
City	Precip	City	Precip
Mobile	67.0	Reno	7.2
Juneau	54.7	Concord	36.2
Phoenix	7.0	Atlantic City	45.5
Little Rock	48.5	Albuquerque	7.8
Los Angeles	14.0	Albany	33.4
Sacramento	17.2	Buffalo	36.1
San Francisco	20.7	New York	40.2
Denver	13.0	Charlotte	42.7
Hartford	43.4	Raleigh	42.5
Wilmington	40.2	Bismark	16.2
Washington	38.9	Cincinnati	39.0
Jacksonville	54.5	Cleveland	35.0
Miami	59.8	Columbus	37.0
Atlanta	48.3	Oklahoma City	31.4
Honolulu	22.9	Portland	37.6
Boise	11.5	Philadelphia	39.9
Chicago	34.4	Pittsburg	36.2
Peoria	35.1	Providence	42.8
Indianapolis	38.7	Columbia	46.4
Des Moines	30.8	Sioux Falls	24.7
Wichita	30.6	Memphis	49.1
Louisville	43.1	Nashville	46.0
New Orleans	56.8	Dallas	35.9
Portland	40.8	El Paso	7.8
Baltimore	41.8	Houston	48.2
Boston	42.5	Salt Lake City	15.2
Detroit	31.0	Burlington	32.5
Sault Ste. Marie	31.7	Norfolk	44.7
Duluth	30.2	Richmond	42.6
Minneapolis/St Paul	25.9	Seattle Tacoma	38.8
Jackson	49.2	Spokane	17.4
Kansas City	37.0	Charleston	40.8
St Louis	35.9	Milwaukee	29.1
Great Falls	15.0	Cheyenne	14.6
Omaha	30.2	San Juan	59.2

Note:
Annual Precipitation in US Cities

1.1 Realizar una estimación no paramétrica de la función de densidad por el método del núcleo.

```
x=precipdf[,2]
hist(precipdf[,2], br=10,prob=TRUE, main="Precipitación",
     col="gray",
     xlab="Precipitaciones (litros)", ylab="Densidad estimada",
     ylim = c(0,0.05),xlim = c(0,70))
```

```
lines(density(x, bw="SJ"),lwd=2,col="red")
rug(x)
```



Hemos elegidio el método SJ, que implementa la propuesta de Sheather y Jones, basado en la estimación del núcleo de f''

1.2 Realizar una estimación no paramétrica de la función de densidad por el método de los logsplines.

Estima el logaritmo de la función de densidad mediante un spline cúbico.

```
library(logspline)
ajuste <- logspline(precip_data)
ajuste # 7nudos, criterio BIC
```

```
##  knots A(1)/D(2)  loglik    AIC minimum penalty maximum penalty
##      4         2 -282.85 578.45           15.02             Inf
##      5         2 -275.34 567.68            0.51           15.02
##      6         2 -275.29 571.82              NA              NA
##      7         2 -274.83 575.15            0.41            0.51
##      8         2 -274.79 579.31              NA              NA
##      9         2 -274.41 582.82            0.02            0.41
##     10         1 -274.40 587.04            0.00            0.02
## the present optimal number of knots is  5
## penalty(AIC) was the default: BIC=log(samplesize): log( 70 )= 4.25
```

Vamos a comprobar los valores AIC

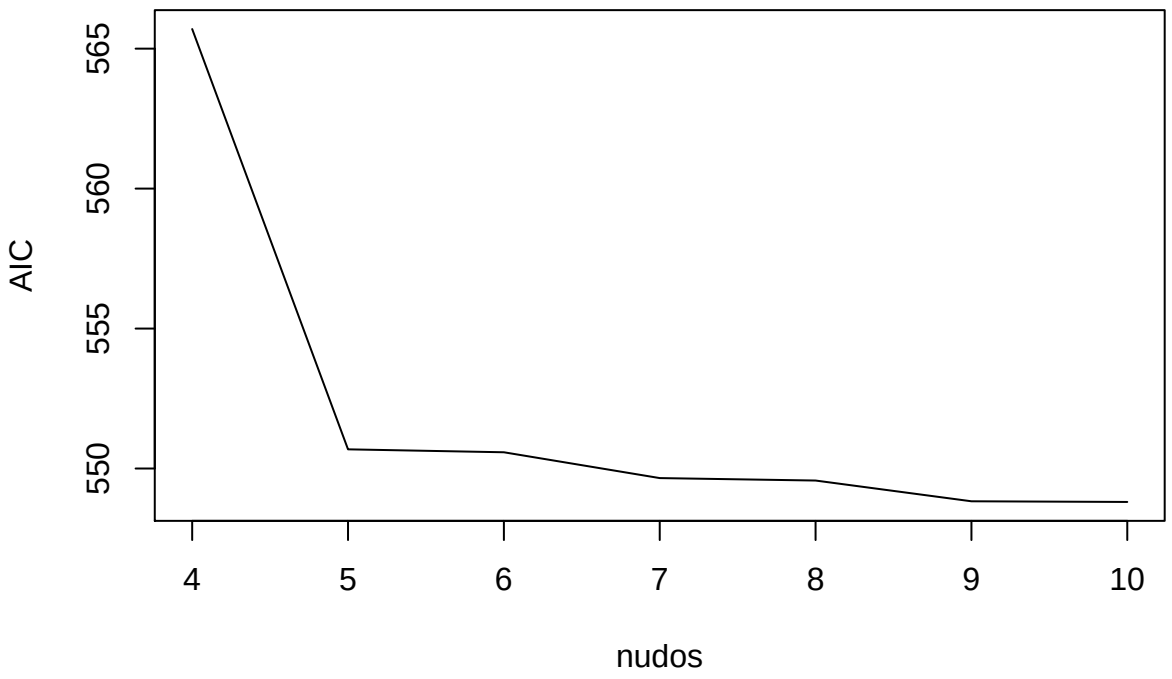
```
resul<- ajuste$logl
nudos<- resul[,1]
logL<- resul[,3]
```

```
AIC<- -2*logL+log(length(precip_data))*(nudos-1)
AIC
```

```
## [1] 565.7001 550.6848 550.5803 549.6576 549.5704 548.8295 548.8057
```

Lo dibujamos

```
plot(nudos,AIC,type="l")
```



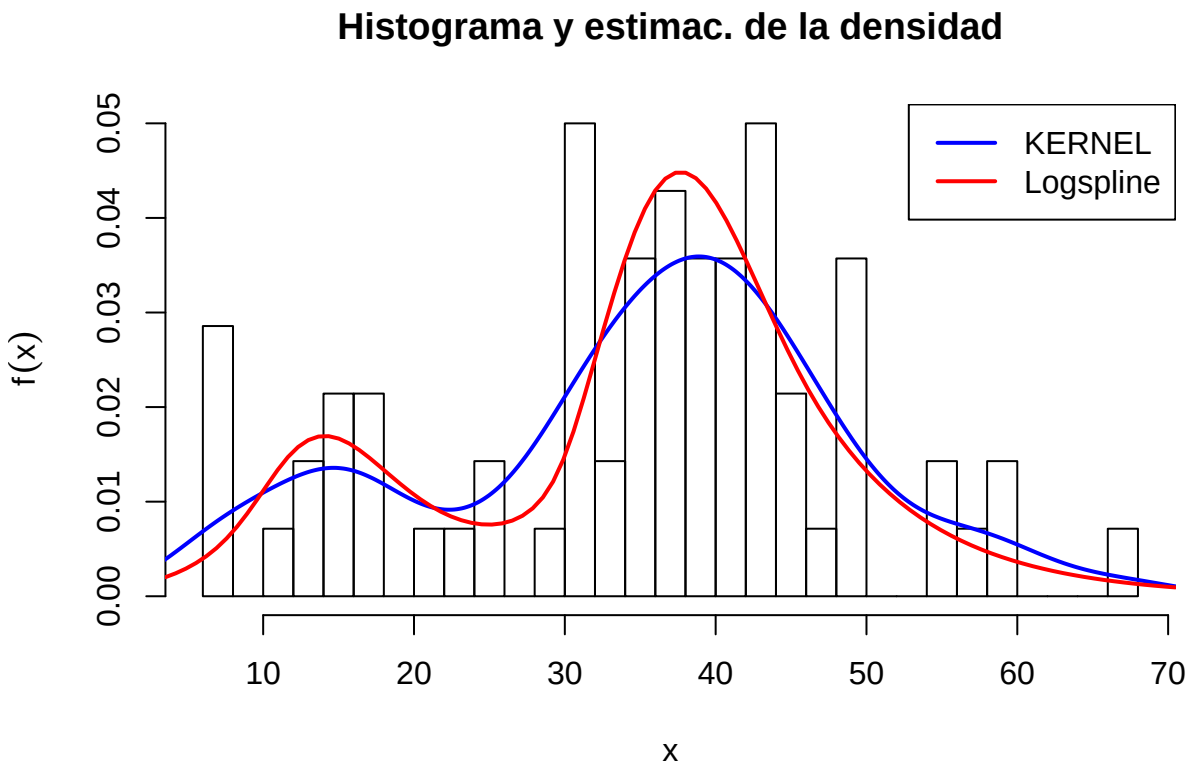
Ahora el histograma

```
hist(x,br=30, prob=TRUE,
     main="Histograma y estimac. de la densidad",
     ylab = expression(hat(f)(x)),
     xlab="x")

lines(density(precipdf[,2],bw="SJ"),col="blue",lwd=2)

plot(ajuste,col="red",lwd=2,add=TRUE)

legend("topright",
       col=c("blue","red"),
       lwd=2,
       legend=c("KERNEL","Logspline"))
```



1.3 Estimar P[precip>42] y el cuantil 0.90.

```
1-plogspline(42, ajuste)
```

```
## [1] 0.2865462
```

```
qlogspline(0.9, ajuste)
```

```
## [1] 50.17582
```

2 Pregunta 2

Cuando cierta empresa recibe una invitación para optar a un contrato, la oferta no se puede completar hasta que sea revisada por cuatro departamentos: Ingeniería, Personal, Legal y Contabilidad. Los departamentos empiezan a trabajar al mismo tiempo, pero lo hacen de forma independiente. El tiempo en semanas que emplean en completar la revisión es una variable aleatoria con las siguientes distribuciones:

- Ingeniería: Exponencial con media 3 semanas
- Personal: Normal con media 4 y desviación típica 1;
- Legal: 2 o 4 semanas, siendo ambos valores equiprobables
- Contabilidad: Uniforme continua en el intervalo (1,5).

Se trata de simular el tiempo W que tarda la empresa en preparar una oferta. Para ello se pueden implementar los siguientes pasos:

2.1 Fijar M (número de ofertas, se recomienda al menos 1000).

```
M=1000 # número de ofertas
```

2.2 Definir una matriz Mx4 donde se irán almacenando los valores generados.

```
matriz=matrix(NA,M,4) # Posteriormente almacenaremos los datos
dim(matriz)
```

```
## [1] 1000    4
```

2.3 Repetir M veces:

2.3.1 Generar de forma independiente los cuatro tiempos según las cuatro distribuciones.

```
W = rep(NA,M)
for (i in 1:M) { # Guardar esos cuatro tiempos en una fila de la matriz
  vIng = rexp(1,1/3)
  VPer = rnorm(1,4,1)
  x=c(2,4)
  vLeg = sample(x,1,replace = T)
  vCon = runif(1,1,5)


# Para cada i genero un valor de W
  W[i] = max(vIng,VPer,vLeg,vCon) #Calcular W como el máximo de los cuatro tiempos.
}
# Al final tendrás una m.a.s. de valores W de tamaño M: W_1, ..., W_M
```

Creo la matriz

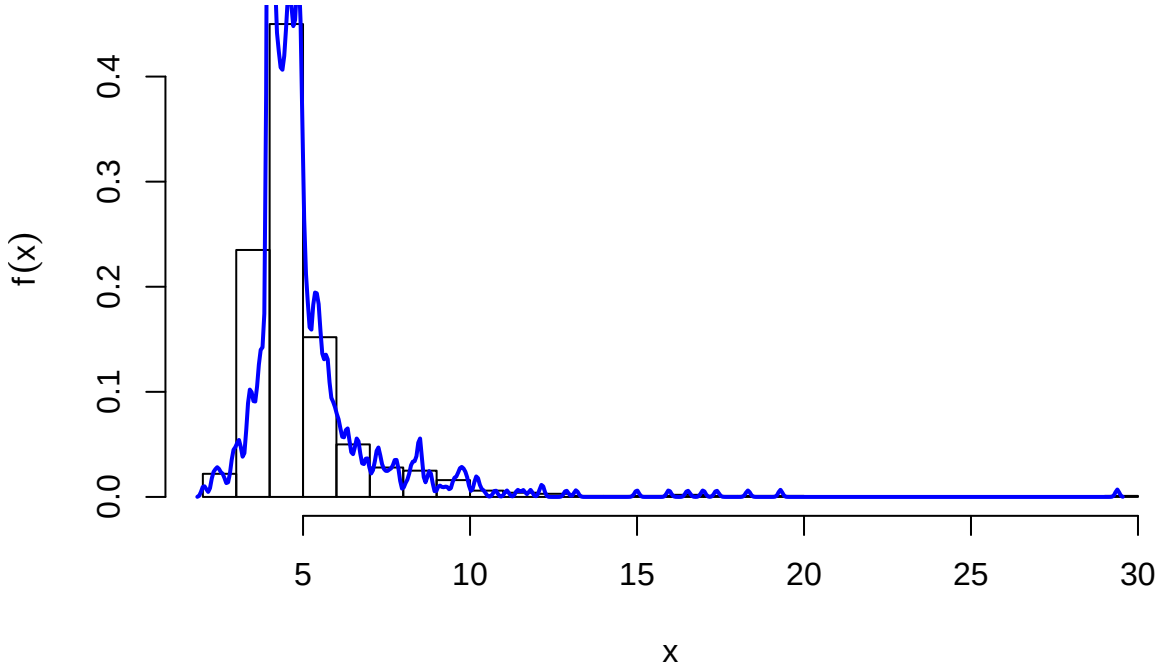
```
m= matrix(W,ncol = 4)
colnames(m)<- c("Ingeniería","Personal","Legal","Contabilidad")
head(m)
```

```
##      Ingeniería Personal    Legal Contabilidad
## [1,]   5.107585 4.203989 2.995548     5.853414
## [2,]   8.296528 4.702960 4.365256     4.079787
## [3,]   6.583318 4.493815 5.062158    12.121483
## [4,]   4.612778 4.000000 3.809379     4.850915
## [5,]   4.771417 4.000000 4.635431     4.184819
## [6,]   4.507948 5.089022 4.635927     3.556583
```

Se pide:

- Estudiar gráficamente la distribución de la variable aleatoria W "tiempo que transcurre hasta completar la oferta".

```
hist(W,
     br=30,
     prob=TRUE,
     main="Histograma y estimación de la densidad",
     ylab=expression(hat(f)(x)),xlab="x")
lines(density(W,bw="SJ"),col="blue",lwd=2)
```



Estimar su media y su mediana.

```
apply(X=m,FUN=median,MARGIN = 2) # Mediana por columnas
```

```
##   Ingeniería     Personal        Legal Contabilidad
##     4.565951     4.511617     4.608508     4.595797
```

```
apply(X=m,FUN=mean,MARGIN = 2) # Media por columnas
```

```
##   Ingeniería     Personal        Legal Contabilidad
##     5.196131     4.954124     4.961743     4.955806
```

Para los datos totales, sin hacerlo por departamento:

Estimación del tiempo medio

```
mean(W)
```

```
## [1] 5.016951
```

Estimación de la mediana:

```
median(W)
```

```
## [1] 4.565055
```

2.3.2 Estimar la probabilidad de que W supere las 6 semanas.

$$P[W > 6]$$

```
ajuste<- logspline(m)
ajuste
```

```
##  knots A(1)/D(2)   loglik     AIC minimum penalty maximum penalty
##      6         2 -1438.34 2911.23           63.03             Inf
```

```
##      7         2 -1406.83 2855.10           30.04           63.03
##      8         2 -1391.80 2831.96            4.69           30.04
##      9         2 -1391.72 2838.71              NA              NA
##     10         2 -1387.12 2836.40            2.70            4.69
##     11         2 -1386.45 2841.97              NA              NA
##     12         2 -1384.48 2844.95              NA              NA
##     13         2 -1384.39 2851.67              NA              NA
##     14         2 -1381.72 2853.23            0.02            2.70
##     15         2 -1381.71 2860.12            0.01            0.02
##     16         2 -1381.70 2867.02            0.00            0.01
##     17         1 -1381.70 2873.93            0.00            0.00
## the present optimal number of knots is  8
## penalty(AIC) was the default: BIC=log(samplesize): log( 1000 )= 6.91
```

Luego, la probabilidad será:

```
1-plogspline(6, ajuste)
```

```
## [1] 0.1355751
```

2.3.3 ¿Cuál es el departamento que suele tardar más en completar la revisión?

```
m %>%
  colSums()/M
```

```
##   Ingeniería     Personal        Legal Contabilidad
##     1.299033     1.238531     1.240436     1.238952
```

El tiempo medio de cada departamento, por tanto, vemos que el departamento de personal es el que es más lento a la hora de hacer una revisión completa.

2.3.4 ¿Cuál es la ordenación más frecuente de los cuatro tiempos?

```
library(modeest)
```

```
## Registered S3 method overwritten by 'rmutil':
##   method         from
##   print.response httr
```

```
apply(X=m,FUN=mfv,MARGIN = 2)
```

```
##   Ingeniería     Personal        Legal Contabilidad
##            4            4            4            4
```

```
mfv(W)
```

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

4 unidades de tiempo (horas)

3 Pregunta 3

Generar aleatoriamente un conjunto de datos donde tengo sentido construir un modelo de clasificación o de predicción. Ajustar el modelo y estimar su capacidad de generalización mediante Jackknife y mediante Validación Cruzada (K=10).

3.1 Generamos los datos con la función sample(aleatoriamente).

Generaremos un dataset con 120 filas y 6 variables, por ejemplo.

```
V1=sample(c(1900:2021), size = 120,replace = TRUE)
V2=runif(120,5,13)
V3=sample(c(14:70), size = 120,replace = TRUE)
V4=runif(120,5,13)
V5=runif(120,0,1)
V6=sample(c(2:220), size = 120,replace = TRUE)

variables=c("A","B","C","D","E","F")
dataSet=cbind.data.frame(V1,V2,V3,V4,V5,V6)

colnames(dataSet)<-variables
head(dataSet)
```

```
##      A         B  C         D          E   F
## 1 2007 10.299073 38  8.925548 0.43793629  86
## 2 1994  7.749463 38  6.624760 0.58161307  49
## 3 1988  8.362668 40 11.481266 0.60620246  35
## 4 2006 11.312834 45  7.645830 0.75878409 211
## 5 1921  6.344973 48 12.648784 0.09451172 165
## 6 1973  8.330727 44  9.059287 0.55158214 142
```

```
summary(dataSet)
```

```
##        A               B                C               D        
##  Min.   :1900   Min.   : 5.042   Min.   :14.00   Min.   : 5.029  
##  1st Qu.:1934   1st Qu.: 7.499   1st Qu.:28.00   1st Qu.: 7.429  
##  Median :1958   Median : 9.454   Median :44.00   Median : 9.278  
##  Mean   :1962   Mean   : 9.329   Mean   :42.88   Mean   : 9.324  
##  3rd Qu.:1994   3rd Qu.:11.385   3rd Qu.:57.25   3rd Qu.:11.222  
##  Max.   :2021   Max.   :12.950   Max.   :70.00   Max.   :12.981  
##        E                   F         
##  Min.   :0.0003458   Min.   :  3.0  
##  1st Qu.:0.2731895   1st Qu.: 44.5  
##  Median :0.5413551   Median : 93.5  
##  Mean   :0.5185485   Mean   :101.4  
##  3rd Qu.:0.7667750   3rd Qu.:161.0  
##  Max.   :0.9948177   Max.   :218.0
```

3.2 Modelo

Elegimos como variable respuesta la variable C

```
modelo=lm(C~.,data=dataSet)
summary(modelo)
```

```
##
## Call:
## lm(formula = C ~ ., data = dataSet)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -30.1340 -14.5370   0.5352  14.9832  27.6322
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 17.95760   87.51798   0.205    0.838
## A            0.01360    0.04432   0.307    0.760
## B           -0.62057    0.70042  -0.886    0.377
## D           -0.25740    0.67221  -0.383    0.702
## E            4.67194    5.41869   0.862    0.390
## F            0.03951    0.02450   1.613    0.110
##
## Residual standard error: 16.7 on 114 degrees of freedom
## Multiple R-squared:  0.03501,    Adjusted R-squared:  -0.00731
## F-statistic: 0.8273 on 5 and 114 DF,  p-value: 0.5328
```

3.3 Estimaciones JACKKNIFE

```
n=nrow(dataSet)
prediJ = numeric(n)
for(i in 1:n){
modelo.i = lm(C~.,data=dataSet[-i,])
prediJ[i]<-predict(modelo.i,dataSet[i,])
}

resi_J=dataSet$C - prediJ
```

```
(RMSE_J<-sqrt( mean(resi_J^2) ))
```

```
## [1] 17.11187
```

4 Pregunta 4

Bootstrap. Implementar una función que calcule el estadístico de Fisher de comparación de coeficientes de correlación lineal:

$$T = \frac{Z_1 - Z_2}{\sqrt{\frac{1}{n_1 - 3} + \frac{1}{n_2 - 3}}}$$

Donde:

- Z1 y Z2 representan la transformación de Fisher de los respectivos coeficientes de correlación lineal para dos grupos
- n1 y n2 son las frecuencias absolutas respectivas de los grupos.

Sobre un conjunto de datos apropiado (que puede ser generado), utilizando el anterior estadístico, realizar e interpretar un test bootstrap bilateral de comparación de los coeficientes de correlación lineal (B=1999).