Tarea 1

Alejandro Brenes (C21319), Santiago Fernández (C22943), Eyeri Méndez (C24765)

2024-09-20

Inicialmente, se cargan todas las librerías necesarias para la tarea.

```
pacman::p_load(readxl,
               dplyr,
               univariateML,
               rriskDistributions,
               ks,
               boot,
               ggplot2,
               cowplot)
```

Se lee la base de datos necesaria.

```
BaseSalarios <- read_excel("BaseSalarios.xlsx")
```

Se corrige el formato de algunas columnas.

```
BaseSalarios$Fec.Nac <- as.Date(BaseSalarios$Fec.Nac)

BaseSalarios <- BaseSalarios %>%
  rename(Cuotas = Coutas)
```

Ahora se procede con los ejercicios.

Parte I

1)

Agregamos la categoría de nivel, en esta, se pondrá un 1 si la observación tiene menos de 150 cuotas y 2 si se tienen más o igual cantidad de cuotas, pues no se especifica qué hacer con las personas que tienen exactamente las 150 cuotas.

```
BaseSalarios <- BaseSalarios %>%
  mutate(Nivel = ifelse(Cuotas < 150, "1", "2"))
```

2)

Generamos una tabla resumen con los requisitos solicitados, en primer lugar, filtrando solo por nivel.

```
BaseSalarios %>%
  group_by(Nivel) %>%
  summarise(prom_sal = mean(U.Salario, na.rm = TRUE),
            var_sal = var(U.Salario, na.rm = TRUE),
            cant_sal = length(U.Salario),
            max_sal = max(U.Salario, na.rm = TRUE),
            min_sal = min(U.Salario, na.rm = TRUE)
            )
```

```
## # A tibble: 2 x 6
##   Nivel prom_sal       var_sal cant_sal  max_sal min_sal
##   <chr>    <dbl>         <dbl>    <int>    <dbl>   <dbl>
## 1 1      792104. 209766412237.     3058 3642975.  10063.
## 2 2     1362840. 325093901051.     2419 6925642. 158431.
```

En la tabla annterior se observan múltiples datos interesantes, con respecto al promedio, las personas con más de 150 cuotas presentan un promedio más elevado, con una diferencia salarial de más de $600.000. Esto se puede unir a la variable de máximo y mínimo, en la primera de estas se observa que el salario máximo de las personas con más de 150 cuotas supera por más de $3.000.000 a los del nivel 1. De forma similar que en el mínimo, en el cual se puede ver que, nuevamente, hay una diferencia abismal entre el nivel 1 y el nivel 2, siendo este último el que tiene un valor más grande.

Con respecto a la cantidad, con los datos mostrados se puede ver que es más común que las personas tengan menos de 150, sin embargo, la cantidad de individuos que presentan más que estas no es tan bajo, pues, en la base de datos, son unos 600 menos.

Pasando a la varianza, se observa una cantidad muy elevada en ambos niveles, esto se puede reforzar con los valores máximos y mínimos de cada nivel, los cuales se encuentran a mucha distancia, esto indica que hay una gran diversidad de datos numéricamente hablando.

En este punto, se procede a hacer una tabla filtrando por sexo.

```
BaseSalarios %>%
  group_by(Sexo) %>%
  summarise(prom_sal = mean(U.Salario, na.rm = TRUE),
            var_sal = var(U.Salario, na.rm = TRUE),
            cant_sal = length(U.Salario),
            max_sal = max(U.Salario, na.rm = TRUE),
            min_sal = min(U.Salario, na.rm = TRUE)
            )
```

```
## # A tibble: 2 x 6
##    Sexo prom_sal       var_sal cant_sal  max_sal min_sal
##   <dbl>    <dbl>         <dbl>    <int>    <dbl>   <dbl>
## 1     1 1156586. 493446452711.     1649 5414449.  16392.
## 2     2  995755. 267642874723.     3828 6925642.  10063.
```

Se logra ver que, en este caso, la cantidad de personas en el sexo 1 es bastante menor que en el sexo 2, lo cual, en primer lugar, explica la elevada varianza de esta primer categoría, además, hace que haya que tomar los datos con precaución, pues la diferencia de observaciones puede llevar a que los datos extremos de los datos se concentren en alguna de las categorías, distorsionando las conclusiones.

Nuevamente, con los datos presentes, el sexo 1 presenta un mayor salario promedio y un salario mínimo mayor, aunque, por otro lado, el sexo 2 presenta un mayor salario máximo, lo cual evidencia que, por lo general, la primer categoría tiene mayor salario, pero algunas observaciones, quizá atípicas, pueden caer en la otra categoría.

Finalmente se realiza una tabla filtrando por sexo y nivel.

```
BaseSalarios %>%
  group_by(Sexo, Nivel) %>%
  summarise(prom_sal = mean(U.Salario, na.rm = TRUE),
            var_sal = var(U.Salario, na.rm = TRUE),
            cant_sal = length(U.Salario),
            max_sal = max(U.Salario, na.rm = TRUE),
            min_sal = min(U.Salario, na.rm = TRUE)
            )
```

```
## # A tibble: 4 x 7
## # Groups:   Sexo [2]
##    Sexo Nivel prom_sal       var_sal cant_sal  max_sal min_sal
##   <dbl> <chr>    <dbl>         <dbl>    <int>    <dbl>   <dbl>
## 1     1 1      864924. 272262960005.      982 3267917.  16392.
## 2     1 2     1585989. 509892012798.      667 5414449. 186703.
## 3     2 1      757658. 176624287110.     2076 3642975.  10063.
## 4     2 2     1277885. 228800963868.     1752 6925642. 158431.
```

De forma similar a las tablas anteriores, se observa que la cantidad de personas en cada grupo influye directamente en la varianza, a pesar de que todas las categorías presentan un valor muy elevado en esta última categoría, se observa que el valor baja cuantas más observaciones tiene el grupo.

Se puede observar que el promedio es más influido por el nivel, pues, el nivel 2 presenta los promedios más altos de los 4 grupos, lo cual se podría intuir de las 2 tablas anteriores, pues la diferencia de promedios entre sexos no era muy grande en comparación con la diferencia de promedios en las categorías de nivel.

Asimismo, se puede ver que el peso del nivel se mantiene en los máximos y mínimos, pues, se observa que en el nivel 2 se encuentran los valores más elevados de ambas variables, dejando claro el peso del nivel sobre el sexo.

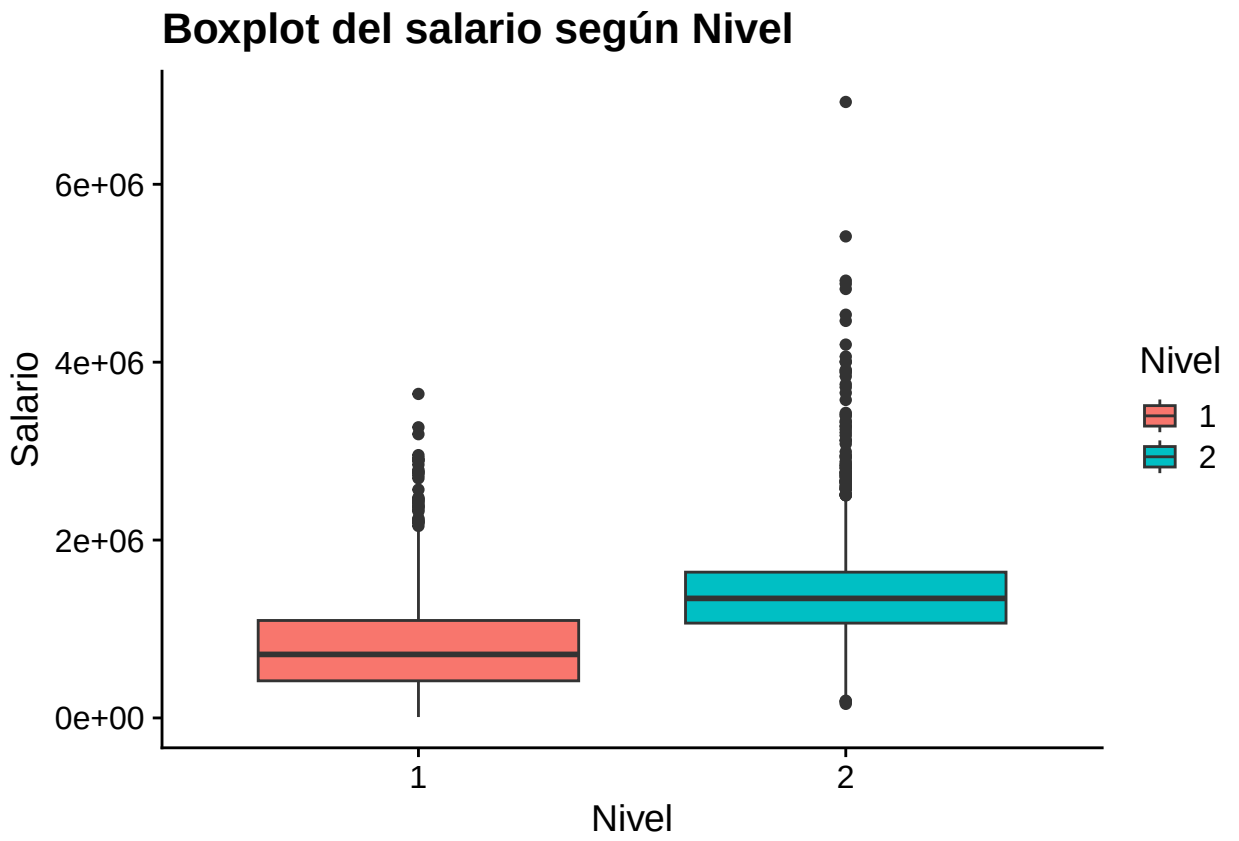
3)

El diagrama de caja, o diagrama de cajas y bigotes, es un gráfico que muestra la distribución de los datos con 5 puntos principales, el máximo, el mínimo y los 3 cuartiles de los datos, es decir, muestra los 4 cuartiles de la muestra de datos y sus extremos. Junto a toda esa información, se logra observar una medida de tendencia central (mediana), medidas de dispersión (el rango y el rango intercuartílico) y la simetría o asimetría de una función, pues los datos pueden estar concentrados en alguno de los extremos de la distribución, en el centro o en un punto medio de estos (Flores, J., & Flores, R. 2018).

4)

Se presenta ahora el gráfico de cajas y bigotes para la variable del salario, filtrando por la categoría Nivel.

```
BaseSalarios %>%
  ggplot(aes(x = Nivel, y = U.Salario, fill = Nivel)) +
  geom_boxplot() +
  labs(title = "Boxplot del salario según Nivel", x = "Nivel", y = "Salario") +
  theme_cowplot()
```



5)

Hay una diferencia clara con respecto a los niveles, las personas que tienen 150 o más cuotas presentan un salario más alto en general, esto se puede comprobar visualmente con los cuartiles de las 2 cajas, pues, el percentil 75% del nivel 1 está casi igual que el percentil 25% del nivel 2, lo cual denota una gran diferencia en cuanto a salarios entre los niveles.

De la mano con lo anterior, se puede ver que el punto en donde más se concentran datos en el nivel 2 (la mediana, representada por la línea del centro de la caja azul), superaría el percentíl 75% del nivel 1, mostrando, una vez más, la gran diferencia entre ambos niveles.

Por otro lado, es interesante la cantidad de valores atípicos, fuera del rango intercuartílico, aunque no se puede determinar una cantidad exacta visualmente, se logra observar que en el nivel 2 hay múltiples valores que superan el máximo del nivel 1, evidenciando que las personas que se encuentran en este último tienen un salario de, a lo sumo, 4.000.000, mientras que en el otro grupo se encuentran varias observaciones por encima de este valor, llegando a superar los 6.000.000.

6)

Usando la prueba de hipótesis, se presenta el siguiente resultado.

```
# Filtramos las observaciones de nivel 1 para salarios
salarios.n1 <- BaseSalarios %>%
  filter(Nivel == "1") %>%
  select(U.Salario)
```

```r
prom.n2 <- BaseSalarios %>%
  filter(Nivel == "2") %>%
  select(U.Salario)

prom.n2 <- mean(prom.n2$U.Salario)

t.test(salarios.n1, mu = prom.n2, alternative = "less", conf.level = 0.95)
```

```
##
##  One Sample t-test
##
## data:  salarios.n1
## t = -68.911, df = 3057, p-value < 2.2e-16
## alternative hypothesis: true mean is less than 1362840
## 95 percent confidence interval:
##      -Inf 805730.9
## sample estimates:
## mean of x
##  792103.6
```

Para esta prueba de hipótesis nos centraremos, inicialmente, en la hipótesis nula, esta dice que la media verdadera (del nivel 1) es mayor o igual a la media del nivel 2. La respuesta a esta pregunta se puede ver con el p-valor resultante de la prueba, el cual es exactamente:

```r
t.test(salarios.n1, mu = prom.n2, alternative = "less", conf.level = 0.95)$p.value
```

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

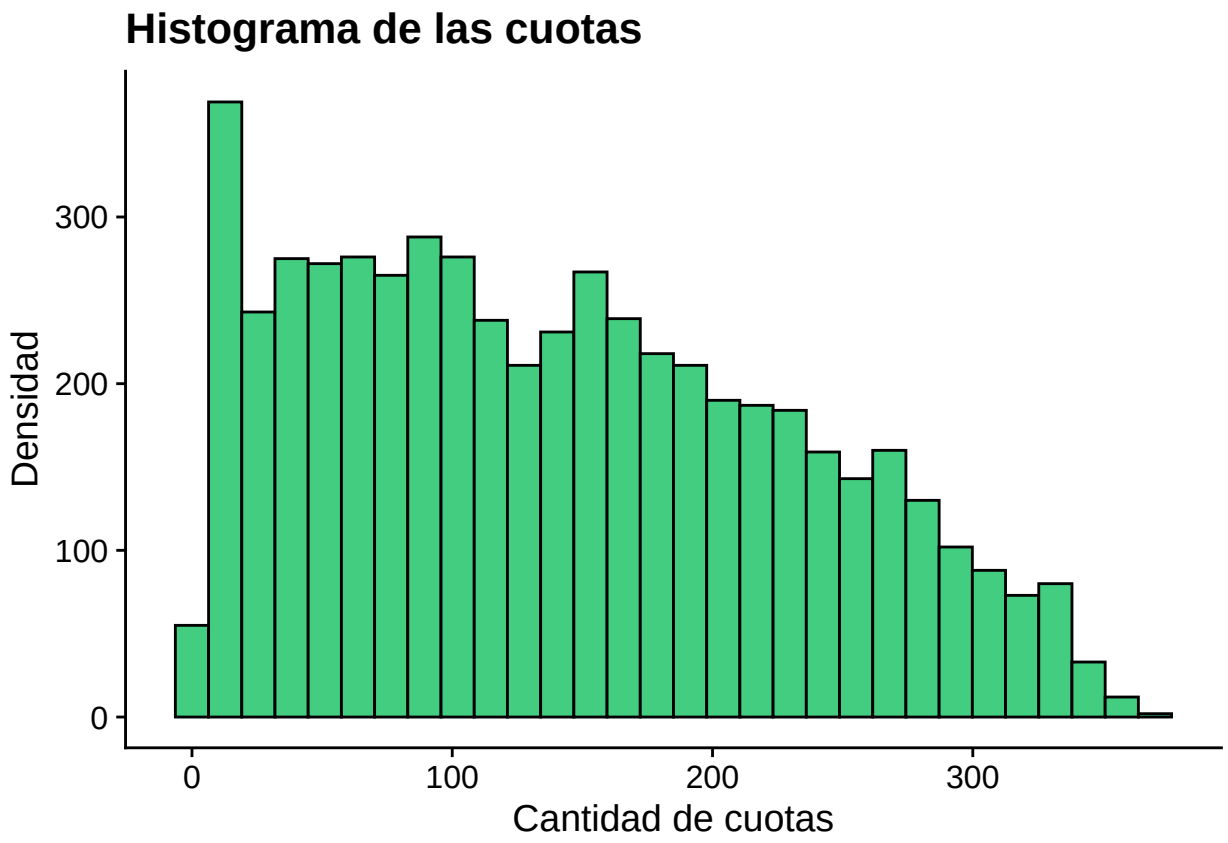
Recordemos que el p-valor indica que si se asume la hipótesis nula cierta, la probabilidad de que sea realmente verdadera. En este caso, hay una probabilidad de exactamente 0 de que la media del nivel 1 sea mayor o igual a la media del nivel 2, lo cual refuerza la conclusión del inciso anterior.

Parte II

1)

El histograma vendría dado por:

```r
BaseSalarios %>%
  ggplot(aes(x = Cuotas)) +
  geom_histogram(fill = "seagreen3", color = "black") +
  labs(title = "Histograma de las cuotas", x = "Cantidad de cuotas", y = "Densidad") +
  theme_cowplot()
```



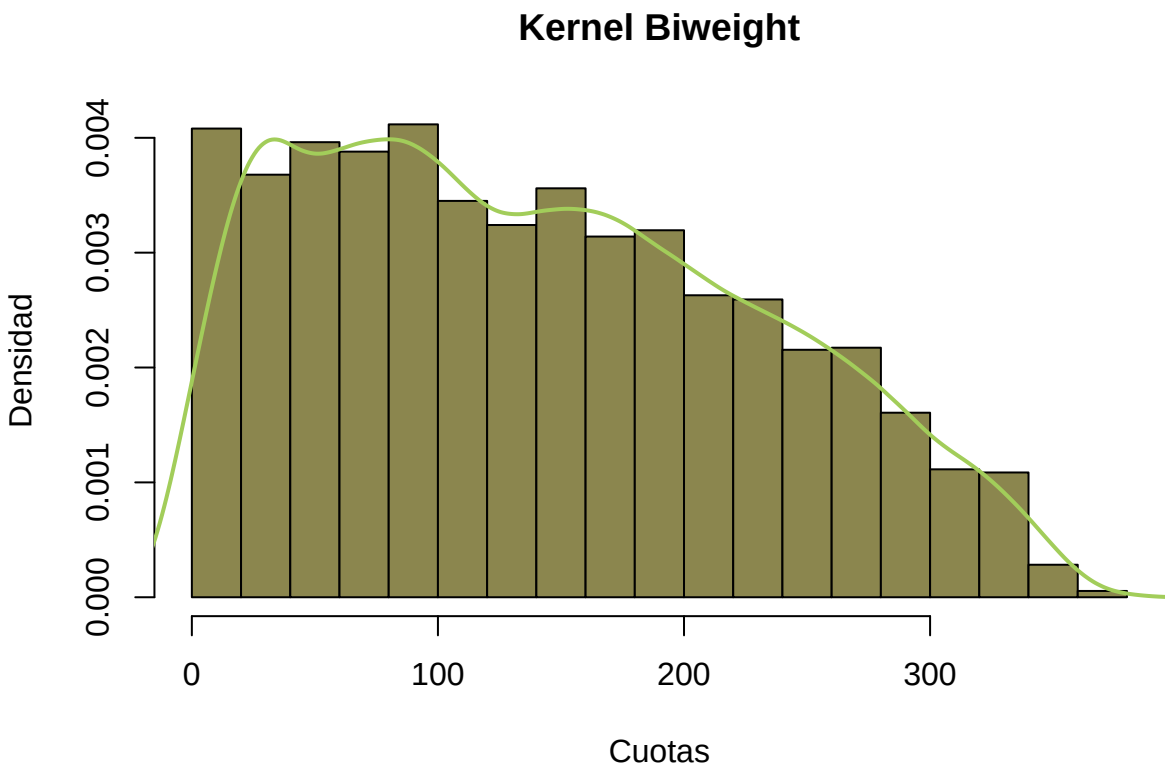
2)

Iniciamos calculando el bandwith pedido:

```
(h <- (0.9 * min(c(sd(BaseSalarios$Cuotas), ((IQR(BaseSalarios$Cuotas)) / (1.35)))) * (nrow(BaseSalarios
```

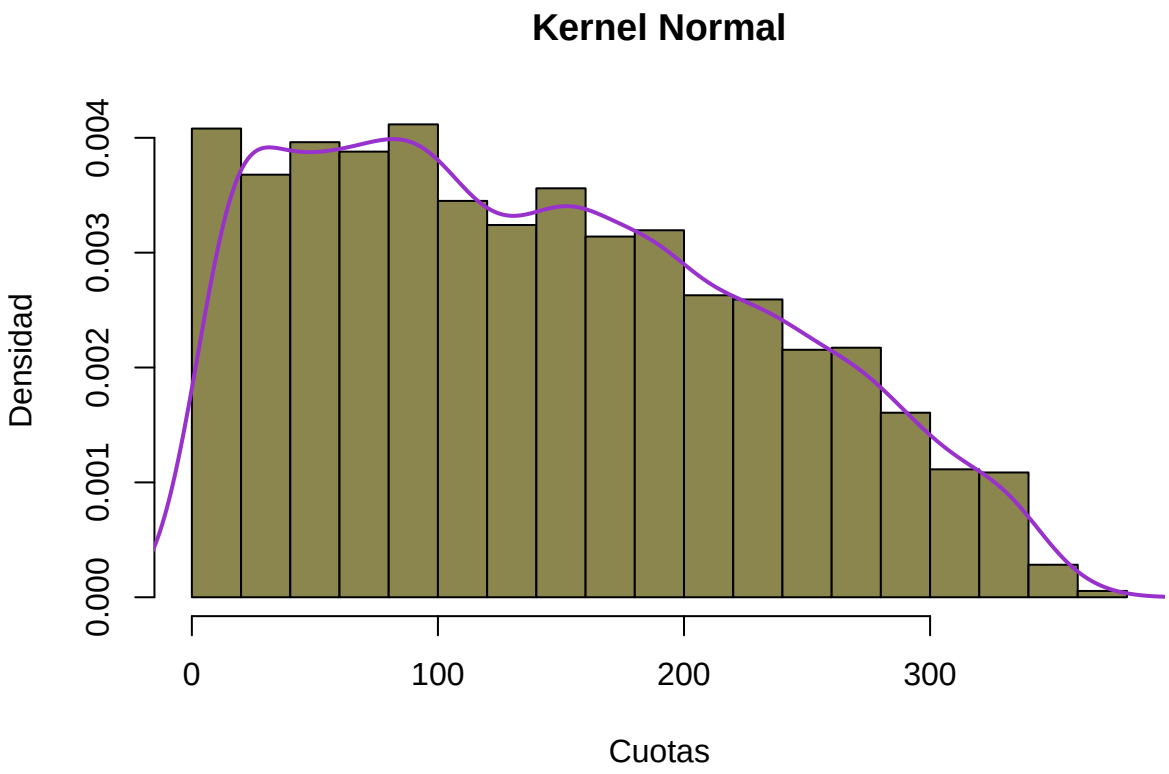
```
## [1] 14.51797
```

```
D <- density(BaseSalarios$Cuotas, kernel = "biweight", bw = h)
hist(
  BaseSalarios$Cuotas,
  main = "Kernel Biweight",
  freq = FALSE,
  col = "khaki4",
  xlab = "Cuotas",
  ylab = "Densidad"
)
lines(D,
      lwd = 2,
      lty = 1,
      col = "darkolivegreen3")
```



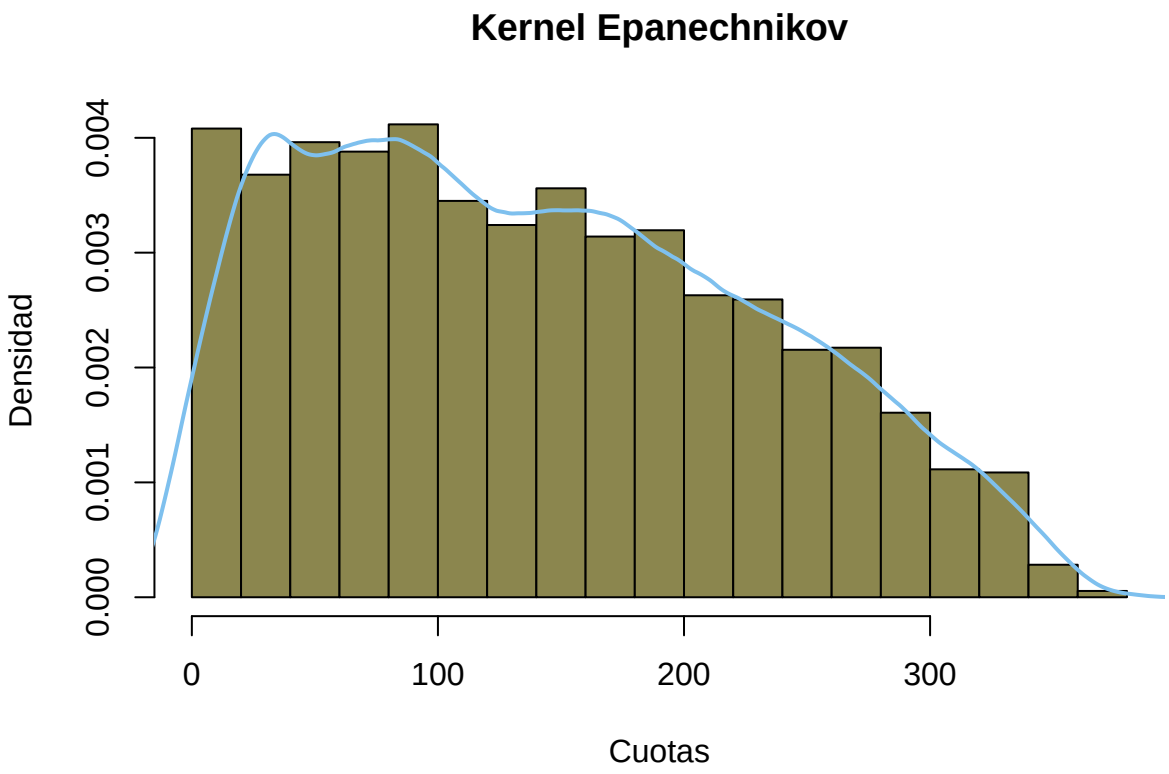
Seguimos con el kernel gaussiano.

```
D <- density(BaseSalarios$Cuotas, kernel = "gaussian", bw = h)
hist(
  BaseSalarios$Cuotas,
  main = "Kernel Normal",
  freq = FALSE,
  col = "khaki4",
  xlab = "Cuotas",
  ylab = "Densidad"
)
lines(D,
      lwd = 2,
      lty = 1,
      col = "darkorchid")
```



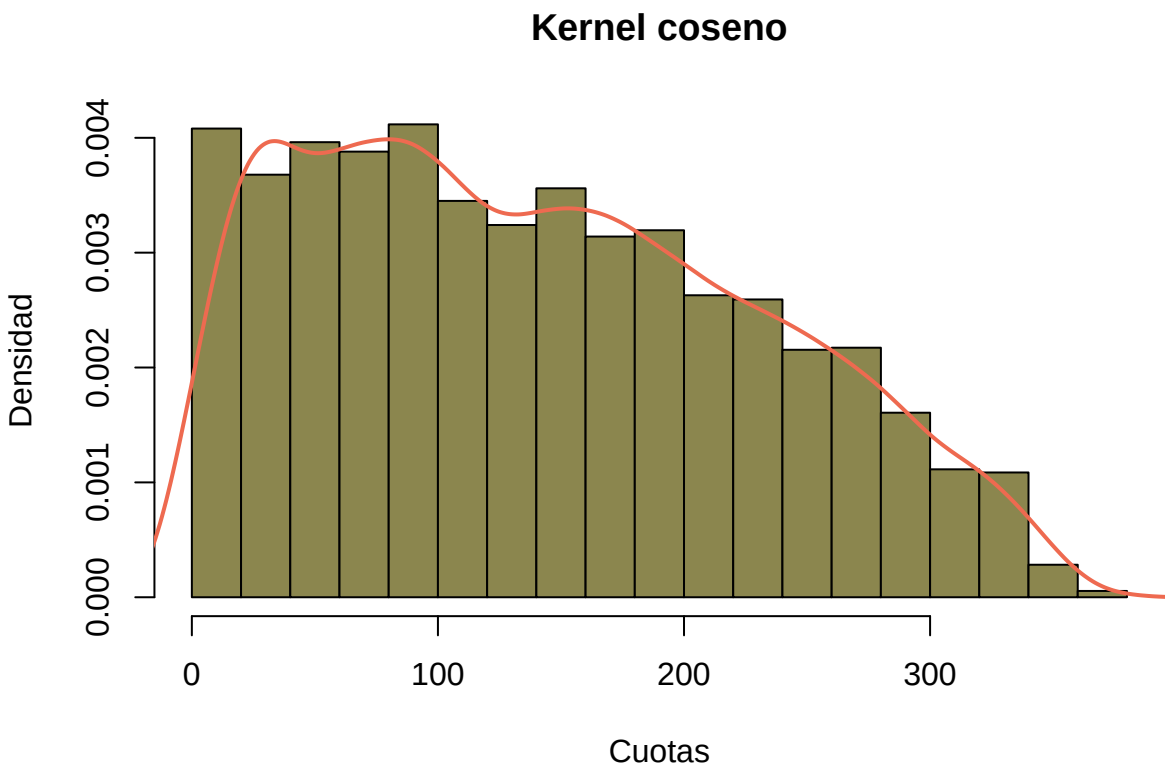
Siguiendo con el de Epanechnikov.

```
D <- density(BaseSalarios$Cuotas, kernel = "epanechnikov", bw = h)
hist(
  BaseSalarios$Cuotas,
  main = "Kernel Epanechnikov",
  freq = FALSE,
  col = "khaki4",
  xlab = "Cuotas",
  ylab = "Densidad"
)
lines(D,
      lwd = 2,
      lty = 1,
      col = "skyblue2")
```



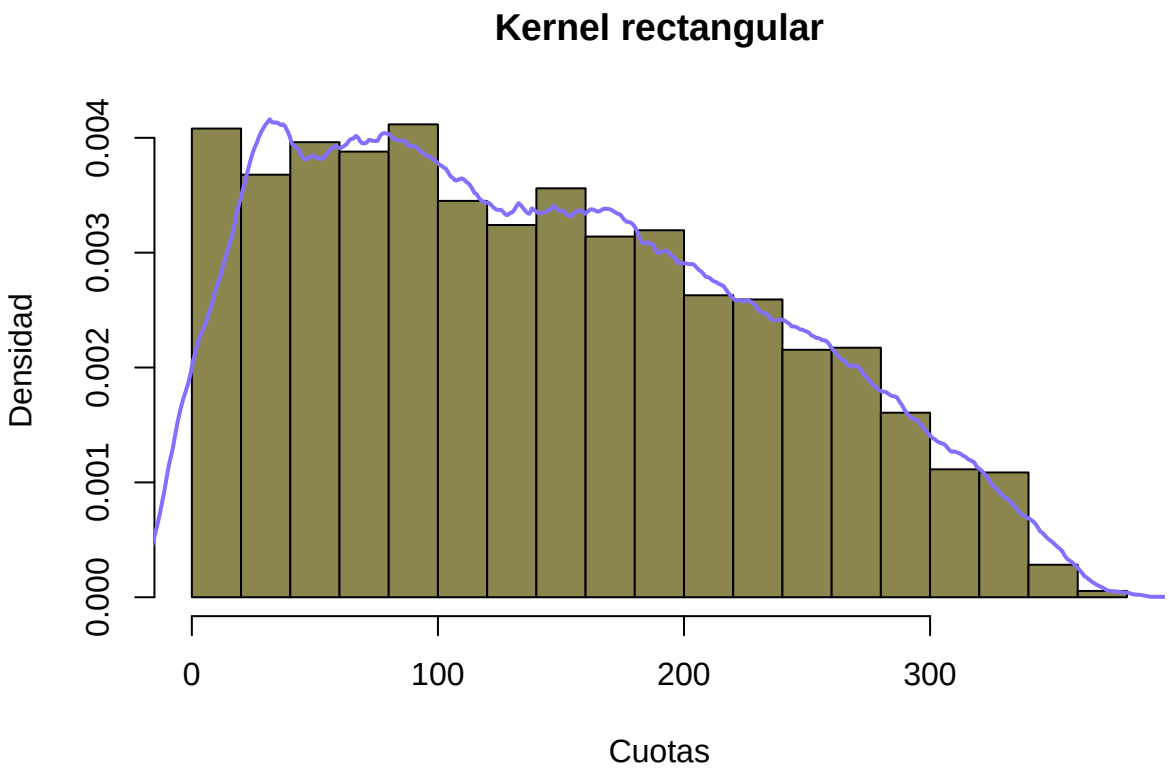
Procedemos con el de coseno.

```
D <- density(BaseSalarios$Cuotas, kernel = "cosine", bw = h)
hist(
  BaseSalarios$Cuotas,
  main = "Kernel coseno",
  freq = FALSE,
  col = "khaki4",
  xlab = "Cuotas",
  ylab = "Densidad"
)
lines(D,
      lwd = 2,
      lty = 1,
      col = "coral2")
```



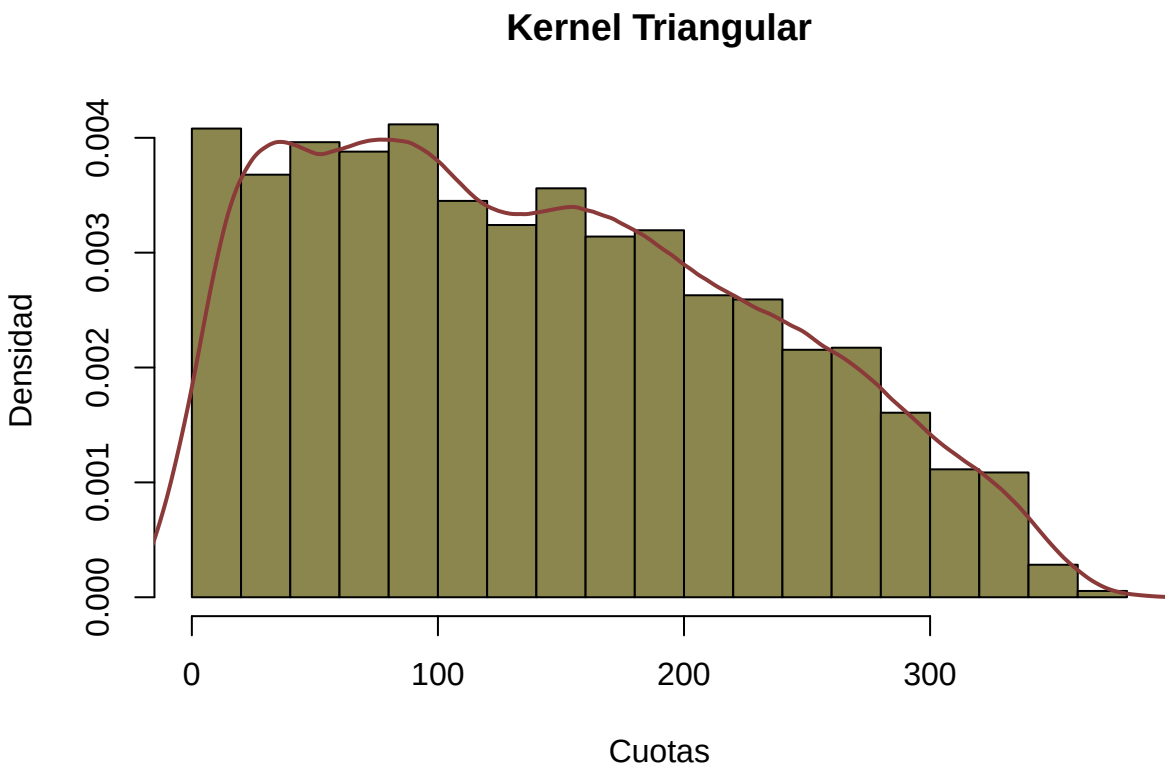
Procedemos con el rectangular.

```
D <- density(BaseSalarios$Cuotas, kernel = "rectangular", bw = h)
hist(
  BaseSalarios$Cuotas,
  main = "Kernel rectangular",
  freq = FALSE,
  col = "khaki4",
  xlab = "Cuotas",
  ylab = "Densidad"
)
lines(D,
      lwd = 2,
      lty = 1,
      col = "lightslateblue")
```



Y finalmente con el triangular.

```
D <- density(BaseSalarios$Cuotas, kernel = "triangular", bw = h)
hist(
  BaseSalarios$Cuotas,
  main = "Kernel Triangular",
  freq = FALSE,
  col = "khaki4",
  xlab = "Cuotas",
  ylab = "Densidad"
)
lines(D,
      lwd = 2,
      lty = 1,
      col = "indianred4")
```

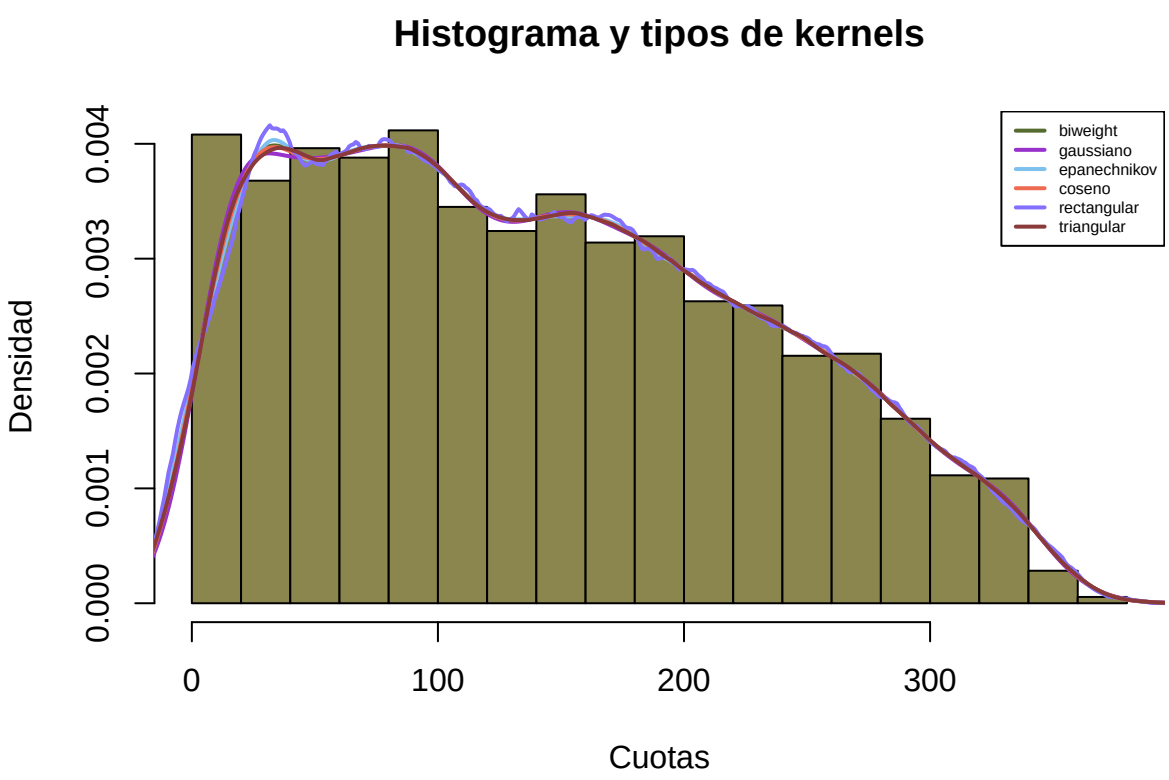


3)

Ahora juntamos todo lo anterior en un solo gráfico.

```
kernels <-
  c("biweight",
    "gaussiano",
    "epanechnikov",
    "coseno",
    "rectangular",
    "triangular")

D1 <- density(BaseSalarios$Cuotas, kernel = "biweight", bw = h)
D2 <- density(BaseSalarios$Cuotas, kernel = "gaussian", bw = h)
D3 <- density(BaseSalarios$Cuotas, kernel = "epanechnikov", bw = h)
D4 <- density(BaseSalarios$Cuotas, kernel = "cosine", bw = h)
D5 <- density(BaseSalarios$Cuotas, kernel = "rectangular", bw = h)
D6 <- density(BaseSalarios$Cuotas, kernel = "triangular", bw = h)

hist(
  BaseSalarios$Cuotas,
  main = "Histograma y tipos de kernels",
  freq = FALSE,
  col = "khaki4",
  xlab = "Cuotas",
```

```
  ylab = "Densidad"
)

lines(D1,
      lwd = 2,
      lty = 1,
      col = "darkolivegreen")
lines(D2,
      lwd = 2,
      lty = 1,
      col = "darkorchid")
lines(D3,
      lwd = 2,
      lty = 1,
      col = "skyblue2")
lines(D4,
      lwd = 2,
      lty = 1,
      col = "coral2")
lines(D5,
      lwd = 2,
      lty = 1,
      col = "lightslateblue")
lines(D6,
      lwd = 2,
      lty = 1,
      col = "indianred4")

legend(
  "topright",
  legend = kernels,
  col = c(
    "darkolivegreen",
    "darkorchid",
    "skyblue2",
    "coral2",
    "lightslateblue",
    "indianred4"
  ),
  lty = 1,
  lwd = 2,
  cex = 0.5
)
```



Parte III

1)

El Criterio de Información de Akaike (AIC) es una herramienta objetiva que cuantifica la idoneidad de un modelo específico en comparación con un conjunto limitado de modelos. Ofrece un método sencillo y objetivo para seleccionar el modelo más apropiado para describir los datos observados ().

Por lo tanto, el Criterio de Información de Akaike es una medida utilizada para comparar y seleccionar modelos estadísticos, especialmente en contextos donde se emplea la máxima verosimilitud. El AIC se basa en la idea de evaluar tanto la calidad del ajuste del modelo como su complejidad, y su fórmula está dada por:

AIC = $2k - 2ln(\hat{L})$,

donde k es el número de parámetros libres del modelo y $\hat{L}$ es la función de máxima verosimilitud.

Este método nos da una aproximación de la distancia entre el modelo y el verdadero proceso que genera los datos observados, el cual es desconocido y a veces hasta difícil de definir. Dado que la estimación está basada en los datos observados, esta distancia es siempre relativa y depende del conjunto de datos que se utilizó. Por lo tanto, un valor de AIC no tiene un significado por sí mismo, sino que es interpretable al compararse con otros valores de AIC utilizando los mismos datos observados ().

En este sentido, el término $2k$ penaliza la complejidad del modelo, previniendo que este sea demasiado complejo, lo cual podría llevar a un sobreajuste, mientras que el término $-2ln(\hat{L})$ representa el ajuste del modelo a los datos. Un valor más bajo indicaría un mejor ajuste.

Por lo tanto, el AIC equilibra el ajuste del modelo con su simplicidad, penalizando la adición de parámetros que no mejoran significativamente el ajuste del modelo. De esta manera, cuando se comparan dos o más modelos, el modelo

preferido es el que tiene el AIC más bajo, ya que significa que el modelo logra un buen equilibrio entre simplicidad y ajuste a los datos.

2)

```
comparacion.univariate <- model_select(BaseSalarios$Cuotas,
                                       models = c("exp", "gamma", "lnorm",
                                                  "weibull", "lgamma", "unif"),
                                       criterion = "aic",
                                       na.rm = TRUE)
comparacion.univariate
```

```
## Maximum likelihood estimates for the Weibull model
##   shape    scale
##   1.504  155.510
```

3)

```
comparacion.rrisk <- fit.cont(BaseSalarios$Cuotas)
```

```
##                   logL       AIC       BIC Chisq(value) Chisq(p) AD(value)
## Normal       -32430.73  64865.45  64878.67      1216.35        0     56.49
## Cauchy       -33822.33  67648.66  67661.88      3531.13        0    143.65
## Logistic     -32618.92  65241.83  65255.05      1500.51        0     59.21
## Exponential  -32584.89  65171.78  65178.39      1395.05        0    210.33
## Chi-square  -124028.73 248059.46 248066.06          Inf        0       Inf
## Uniform           NULL      NULL      NULL          Inf        0       Inf
## Gamma        -32160.53  64325.05  64338.27       803.46        0     58.07
## Lognormal    -32794.97  65593.94  65607.15      2189.18        0    149.71
## Weibull      -32021.27  64046.54  64059.75       593.93        0     40.13
## F            -40588.69  81181.37  81194.59     30957.52        0   2200.72
## Student      -44604.57  89211.15  89217.75     69809.16        0   4756.44
## Gompertz     -31811.61  63627.22  63640.43       200.06        0      9.10
##                 H(AD) KS(value)    H(KS)
## Normal      rejected      0.07 rejected
## Cauchy      rejected      0.15 rejected
## Logistic    rejected      0.08 rejected
## Exponential rejected      0.12 rejected
## Chi-square      NULL      0.47 rejected
## Uniform         NULL      0.06 rejected
## Gamma       rejected      0.07 rejected
## Lognormal   rejected      0.11 rejected
## Weibull     rejected      0.05 rejected
## F               NULL      0.50 rejected
## Student         NULL      0.72 rejected
## Gompertz        NULL      0.03 rejected
```

```
##
## Chosen continuous distribution is: Weibull (weibull)
```

```
## Fitted parameters are:
##      shape      scale
##   1.503597 155.486535
```

4)

Como se puede observar en el dataframe del punto anterior, bajo el criterio del AIC, la distribución Weibull es la que más se aproxima a la variable Cuotas, el cual es el mismo resultado que en el punto 2 de esta parte. Además, los parámetros de dicha distribución obtenidos en los puntos 2 y 3 son prácticamente los mismos, por lo que se decide seleccionar la distribución Weibull como la más idónea para los datos de la variable Cuotas.

5)

Con la distribución seleccionada, se procede a construir un intervalo de confianza para la media y la desviación estándar de la variable Cuotas, usando las fórmulas para la media y la varianza de una distribución Weibull.

```
cuotas.weibull <- mlweibull(BaseSalarios$Cuotas)

ic.media <- bootstrapml(
  cuotas.weibull,
  map = function(x)
    x[2] * gamma(1 + 1 / x[1]),
  probs = c(0.05, 0.95)
)

ic.desv.est <- bootstrapml(
  cuotas.weibull,
  map = function(x)
    sqrt(x[2] ^ 2 * (gamma(1 + 2 / x[1]) - (gamma(
      1 + 1 / x[1]
    ) ^ 2))),
  probs = c(0.05, 0.95)
)

ic.media
```

```
##       5%      95%
## 138.2853 142.4577
```

```
ic.desv.est
```

```
##       5%      95%
## 93.16399 96.93924
```

Parte IV

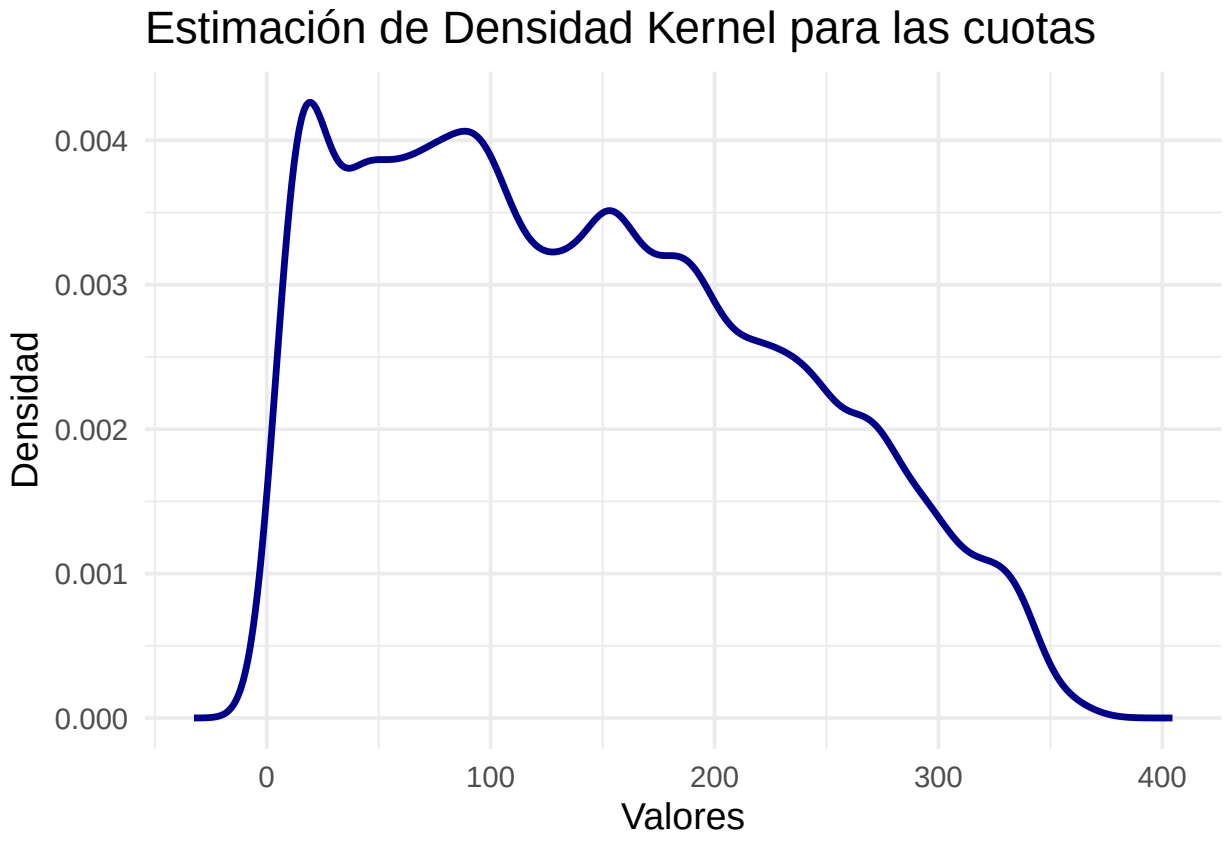
1)

El propósito de la función kde() es poder realizar una estimación de densidad de núcleo (kernel density estimation) en uno o varios puntos. No se asume una forma específica de la distribución para estimar la densidad de probabilidad del conjunto de datos observados. Es compatible con diferentes tipos de kernel. Devuelve un objeto del tipo 'kde' y los

resultados más importantes que incluye son “eval.points” que corresponden a los datos para el eje x y “estimate” que corresponde a la densidad asignada.

A continuación un ejemplo con la columna ‘Cuotas’:

```
cuotas.kde <- kde(x = BaseSalarios$Cuotas)

cuotas.kde.df <- data.frame(x = cuotas.kde$eval.points,
                            y = cuotas.kde$estimate)

ggplot(cuotas.kde.df, aes(x = x, y = y)) +
  geom_line(color = "darkblue", size = 1.2) +
  labs(title = "Estimación de Densidad Kernel para las cuotas",
       x = "Valores",
       y = "Densidad") +
  theme_minimal(base_size = 14)
```



2)

La función boot.ci() se utiliza con el propósito de calcular intervalos de confianza para estimaciones obtenidas a través de un proceso de Bootstrap. Además, proporciona diferentes tipos de intervalos de confianza. Esta función toma como entrada un objeto de la clase boot, generado a partir de la función boot(), que contiene los resultados del procedimiento bootstrap.

3)

Ahora, el próposito es estimar la media μ para las cuotas. Dado que para la prueba Bootstrap se escogieron 1000 muestras, el estimador seria:

$$\hat{\theta} = \frac{1}{1000} \sum_{i=1}^{1000} \bar{x}_i$$

A continuacion, la prueba Bootstrap usando boot():

```
resultados.boot <- boot(data = BaseSalarios$Cuotas,
                        statistic = function(data, indices)
                        mean(data[indices]),
                        R = 1000)
print(resultados.boot)
```

```
##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
## boot(data = BaseSalarios$Cuotas, statistic = function(data, indices) mean(data[indices]),
##     R = 1000)
##
##
## Bootstrap Statistics :
##     original     bias    std. error
## t1* 141.0909 0.01336882    1.181447
```

Teniendo los resultados, procedemos con la comparacion:

```
media.bootstrap <- mean(resultados.boot$t)
media.original <- resultados.boot$t0

print(paste("Media Bootstrap:", media.bootstrap))
```

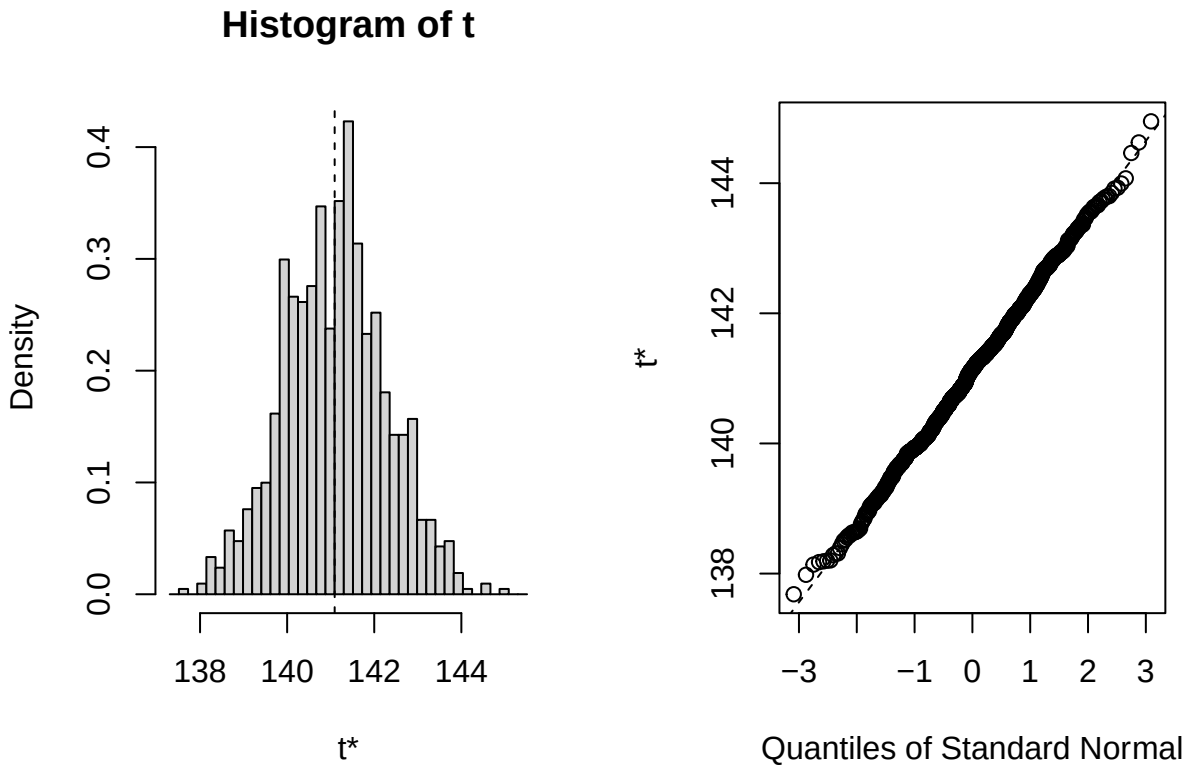
```
## [1] "Media Bootstrap: 141.104294504291"
```

```
print(paste("Media Original:", media.original))
```

```
## [1] "Media Original: 141.090925689246"
```

Finalmente, se realiza el histograma de la prueba Bootstrap:

```
plot(resultados.boot)
```



Referencias

Martínez, D. R., Albín, J. L., Cabaleiro, J. C., Pena, T. F., Rivera, F. F., & Blanco, V. (2009, septiembre). El criterio de información de Akaike en la obtención de modelos estadísticos de rendimiento. XX Jornadas de Paralelismo, A Coruña, España

Flores, J., & Flores, R. (2018). La enseñanza del diagrama de caja y bigotes para mejorar su interpretación. Revista Bases de la Ciencia. e-ISSN 2588-0764, 3(1), 69-75.

kde function - RDocumentation. (2017). Rdocumentation.org. https://www.rdocumentation.org/packages/ks/versions/1.10.7/topics/kde

boot.ci function - RDocumentation. (2024). Rdocumentation.org. https://www.rdocumentation.org/packages/boot/versions/1.3-31/topics/boot.ci