Tarea 1

Maria Carolina Navarro Monge C05513 Tábata Picado Carmona C05961
Jose Pablo Trejos Conejo C07862

Parte I

1.Análisis descriptivo de las variables Cuotas y Salarios con respecto a la variable Sexo.

Primeramente, se cargan las librerías , la base de datos y se fija una semilla para la reproducibilidad de los resultados.

```
library(knitr)
tinytex::install_tinytex(force = TRUE)
library(tidyverse)
library(plotly)
library(DT)
library(univariateML)
library(rriskDistributions) #encuentra que distribuciónn de probabilidades es la
                            #que ajusta mejor con una colección de datos.
                            #que ajusta mejor con una colección de datos.
library(fitdistrplus)
library(metRology)
library(ks)
library(boot)

BD <- read_csv2("BaseSalarios.csv")[-6]
BD <- BD %>% rename("Salario" = "U. Salario", "Cuotas" = "Coutas" )

set.seed(292625)
```

Seguidamente, se realiza el resumen de los datos según el sexo. Se calcula la media, máximo, mínimo y varianza de las cuotas y salarios.

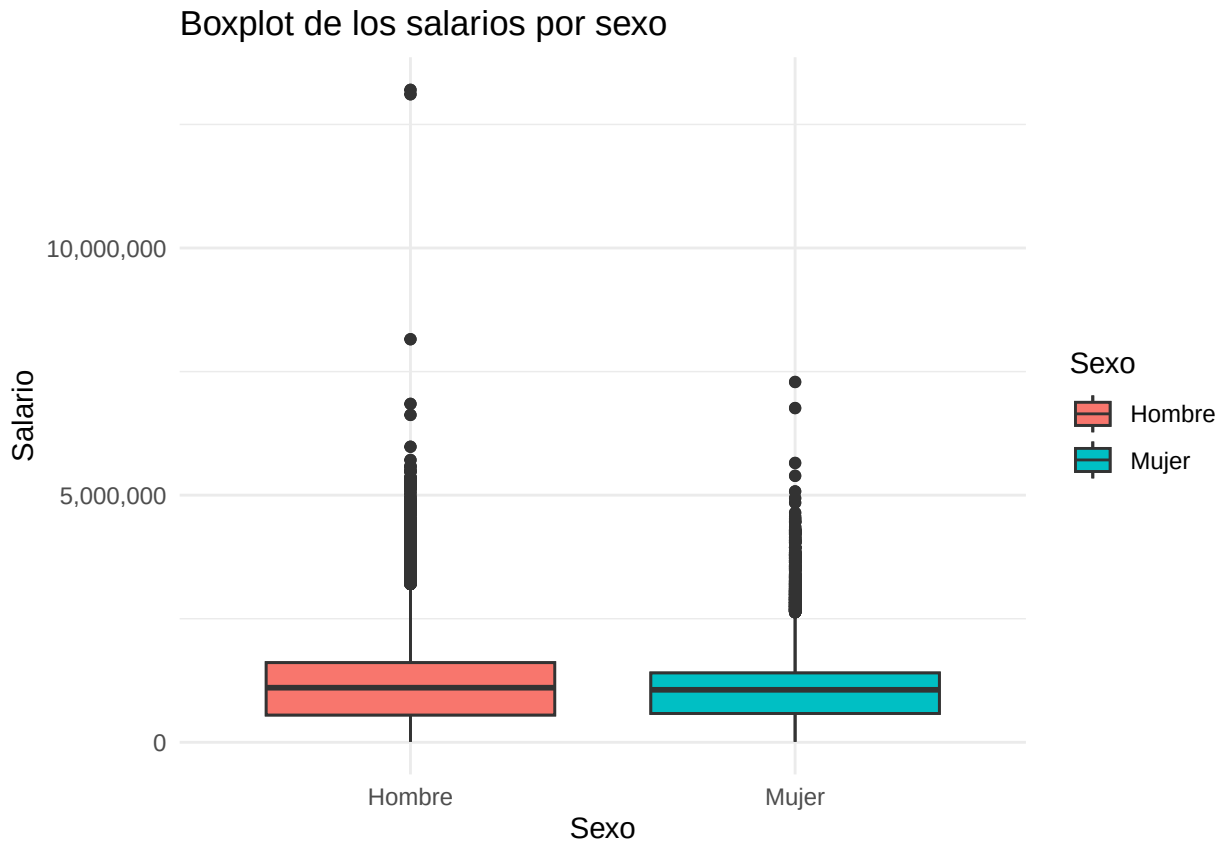
```
resumen<- BD %>% group_by(Sexo) %>% summarise_at(vars(Cuotas,Salario),
                                                 list(Mínimo = min,
                                                 Media = mean,
                                                 Máximo = max,
                                                 Varianza = var))

knitr::kable(resumen, format = "markdown")
```

Sexo	Cuotas_Mínimo	Salario_Mínimo	Cuotas_Media	Salario_Media	Cuotas_Máximo	Salario_Máximo	Cuotas_Varianza	Salario_Varianza
1	1	10880.92	135.1816	1157202	371	13199892	7658.278	509080243513
2	1	10223.99	142.9733	1046661	373	7290150	8187.470	287419606661

Se puede observar en el cuadro anterior que tanto para hombres como para mujeres el mínimo de cuotas es 1 y el máximo solo difiere en 2 cuotas entre ambos sexos mientras que, la media es mayor para las mujeres. Ahora analizando la variable salario, es menor el mínimo de las mujeres que el de los hombres y el mismo comportamiento sucede con el máximo, el cual difiere de gran manera entre un sexo y otro. Además, en promedio los salarios de hombres y mujeres son bastante similares, sin embargo, el salario del sexo masculino sigue por encima del femenino. Por último, se cumple que los salarios de los hombres y las cuotas de las mujeres son los que tienen una mayor variabilidad con respecto a su sexo opuesto.

2. Gráfico boxplot de los salarios según sexo.

```
#Gráfico boxplot
boxplot_salarios<- ggplot(BD, aes(x = factor(Sexo, labels = c("Hombre", "Mujer")), y = Salario, fill= fa
  geom_boxplot() + geom_boxplot() +
  labs(title = "Boxplot de los salarios por sexo",x = "Sexo", y = "Salario",
       fill = "Sexo") +
  scale_y_continuous(labels = scales::comma_format()) +
  theme_minimal()

print(boxplot_salarios)
```



3.¿Qué puede concluir con respecto a los salarios y sexo? ¿Existe alguna diferencia entre los sexos a nivel salarial?

De acuerdo con el gráfico de caja de bigotes, se muestra que los hombres presentan un salario central superior al de las mujeres. También, para el caso de las mujeres, la mediana está más cercana a la parte superior de la caja lo que indica un sesgo a la izquierda, lo que significa que la media es inferior a la mediana. En cuanto a los hombres, prácticamente no se muestra sesgo, ya que, la mediana se muestra muy centrada lo que indica

una mayor simetría. Además, es posible ver que el salario máximo de los hombres es significativamente superior al de las mujeres estando por encima de los diez millones de colones, en cambio, el mayor salario para una mujer es de 7 290 150. Para los hombres, la varianza se muestra casi el doble que el de las mujeres, lo cual, se puede explicar mediante lo descrito anteriormente. Esto indica que existen diferencias según el sexo en cuanto a salarios.

4.Compare su conclusión con una prueba de hipótesis sobre las medias de las categorías de sexo.

Se aplica la prueba t.test para comparar las medias de los salarios de los hombres y mujeres con el fin de encontrar diferencias o no entre los salarios y corroborar la conclusión del inciso anterior. Para la aplicación de la prueba se considera como hipótesis nula que la diferencia entre las medias es de cero. Además, se emplea un nivel de significancia del 5%. Los resultados obtenidos son los siguientes:

```
salarios_hombres <-split(BD, BD$Sexo)[[1]][[5]]
salarios_mujeres <- split(BD, BD$Sexo)[[2]][[5]]

t.test(salarios_hombres, salarios_mujeres)
```

```
##
##  Welch Two Sample t-test
##
## data:  salarios_hombres and salarios_mujeres
## t = 25.046, df = 49886, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  101889.5 119190.8
## sample estimates:
## mean of x mean of y
##   1157202   1046661
```

Como se indica, el p-valor de la prueba es menor a 2.2e-16, esto implica que la hipótesis nula es rechazada. Por ende, se acepta con un nivel de confianza del 95% que existe diferencias entre los salarios promedios de hombres y mujeres. Por tanto, se obtiene la misma conclusión que el inciso anterior.

Parte II

Utilizando la variable U.Salarios sin filtrar por Sexo, construya:

1. El histograma de los salarios.

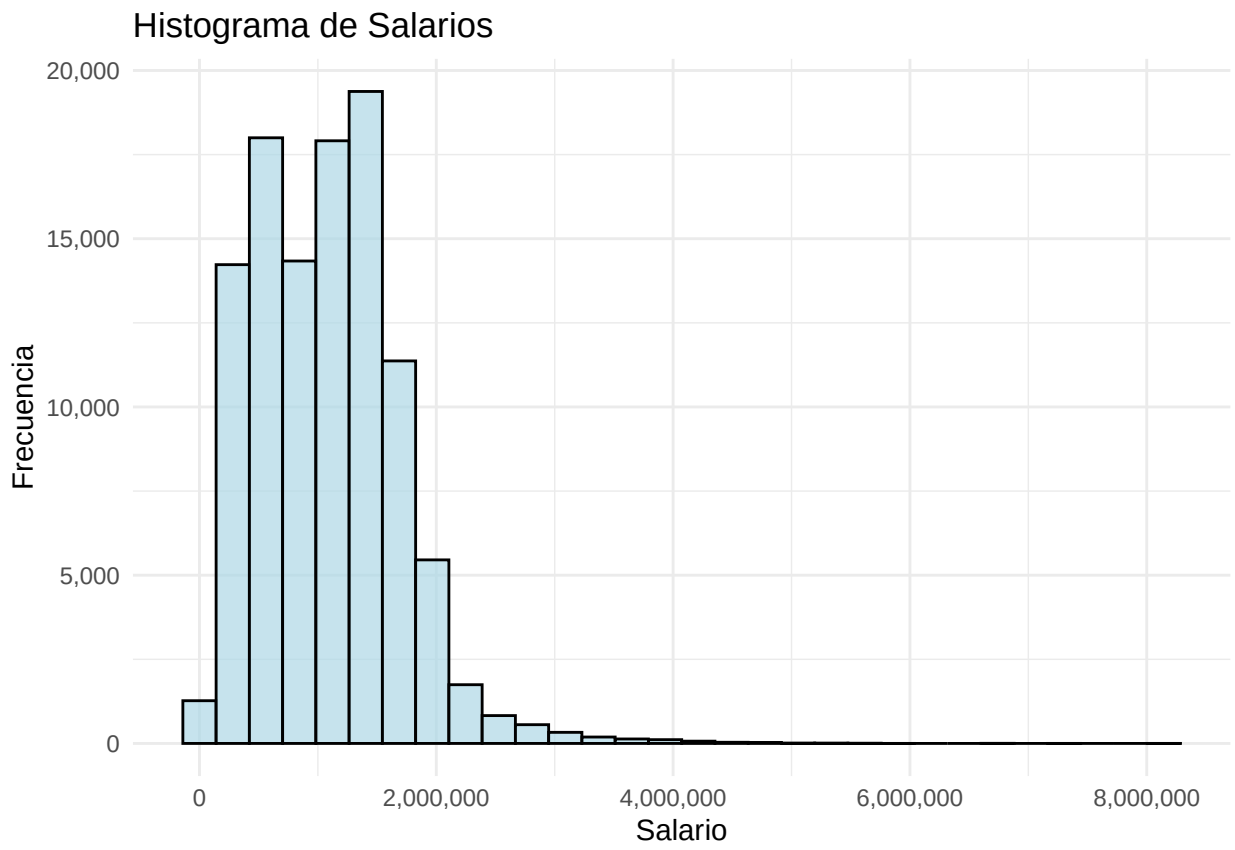
Antes de realizar el histograma se deciden eliminar los outliers más distantes para que esto no afecte a la hora de graficar. Los eliminados son aquellos salarios superiores a los 10 millones de colones, esto debido a que son valores muy altos que se alejan de manera muy significativa del promedio. En total fueron eliminadas dos observaciones.

```
BD <- BD[BD$Salario <= 10000000, ]
```

Después de tener la base de datos adecuada, se realiza el histograma de los salarios.

```
#Histograma de los salarios
hist_salarios <- ggplot(BD, aes(x = Salario)) +
  geom_histogram(fill = "lightblue", color = "black", alpha = 0.7) +
```

```
  labs(title = "Histograma de Salarios", x = "Salario", y = "Frecuencia") +
  scale_x_continuous(labels = scales::comma_format()) +
  scale_y_continuous(labels = scales::comma_format()) +
  theme_minimal()
print(hist_salarios)
```



2.La densidad de los salarios por kernel (no paramétrica) usando como kernel:

a.Biweigth

b.Normal (gaussiana)

c.Epanechnikov

d.Coseno

e.Uniforme (rectangular)

f.Triangular

Para todas use como bw igual al cross-validation insesgado

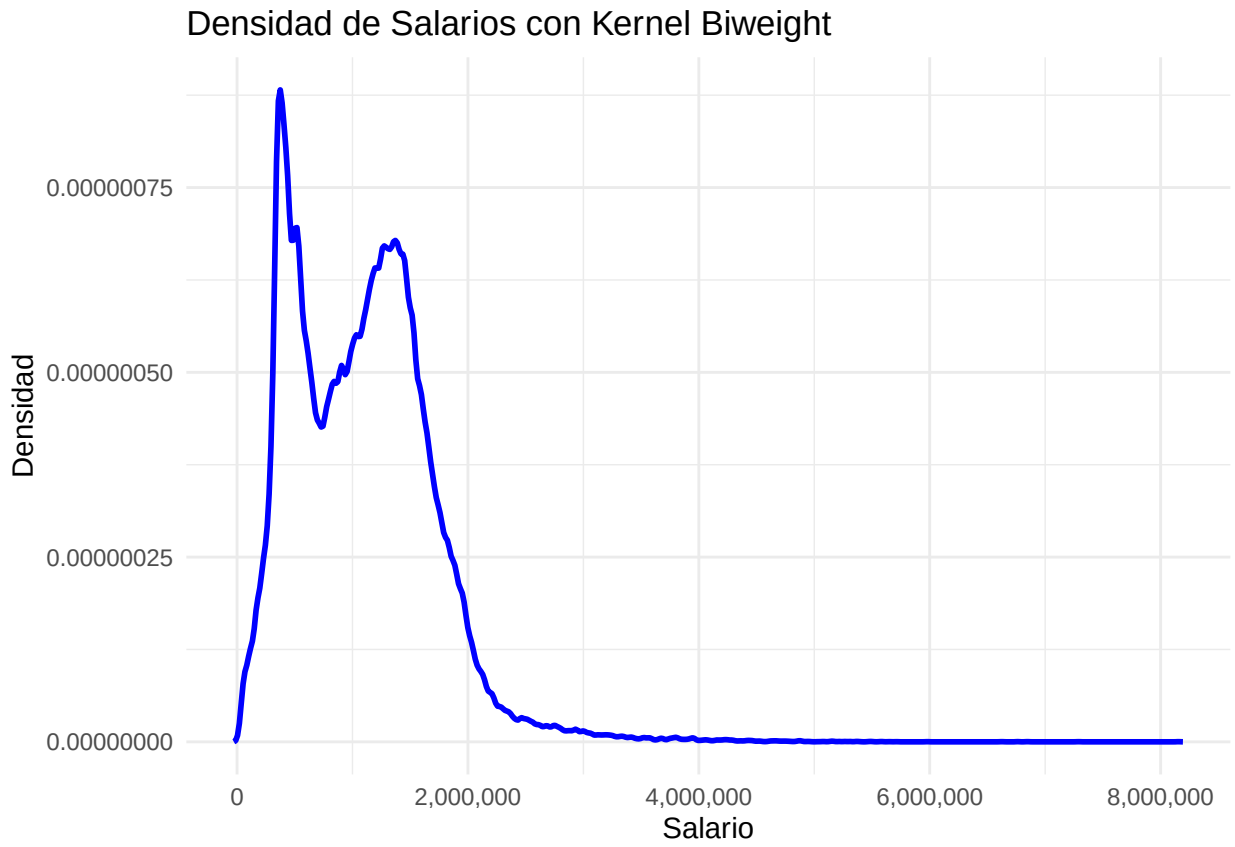
Para mayor facilidad se decidió hacer una función que creara cada gráfico por medio de la función predeterminada density. Además, se ajustan algunos detalles del gráfico para que sea más legible.

```
Kernels <- c("biweight", "gaussian", "epanechnikov", "cosine", "rectangular", "triangular")

densidades <- lapply(Kernels, function(kernel) {
  density(BD$Salario, kernel = kernel, bw = "ucv")
```

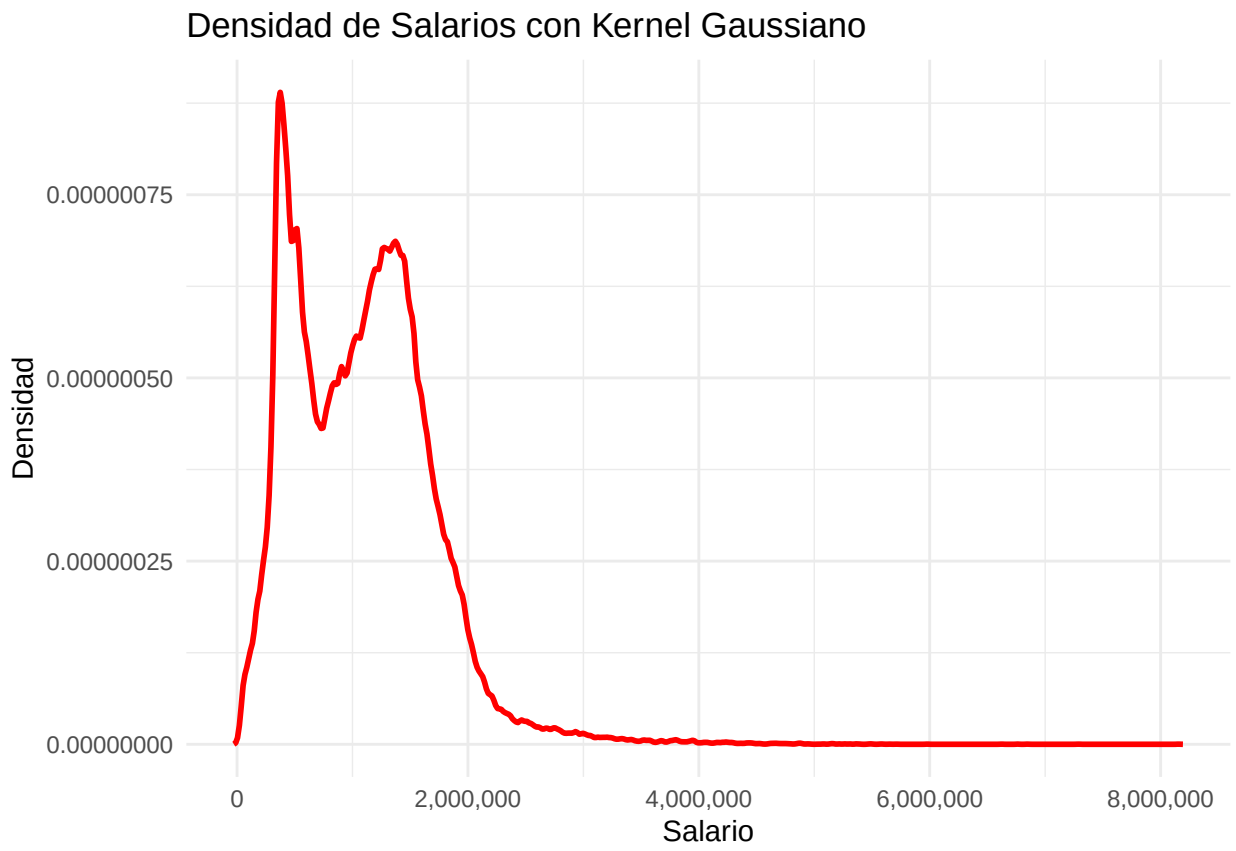
```
})

crear_grafico_kernel <- function(densidad, titulo, col) {
  ggplot(data.frame(x = densidad$x, y = densidad$y), aes(x = x, y = y)) +
    geom_line(color = col, size = 1) +
    labs(title = titulo, y = "Densidad", x = "Salario") +
    scale_x_continuous(labels = scales::comma_format()) +
    scale_y_continuous(labels = scales::comma_format()) +
    theme_minimal()
}

# Límites de los ejes y cuadrícula para todos los gráficos
xlim <- c(0, max(BD$Salario))
ylim <- c(0, max(sapply(densidades, function(d) max(d$y))))
```

A continuación se muestran los gráficos correspondientes a cada Kernel.

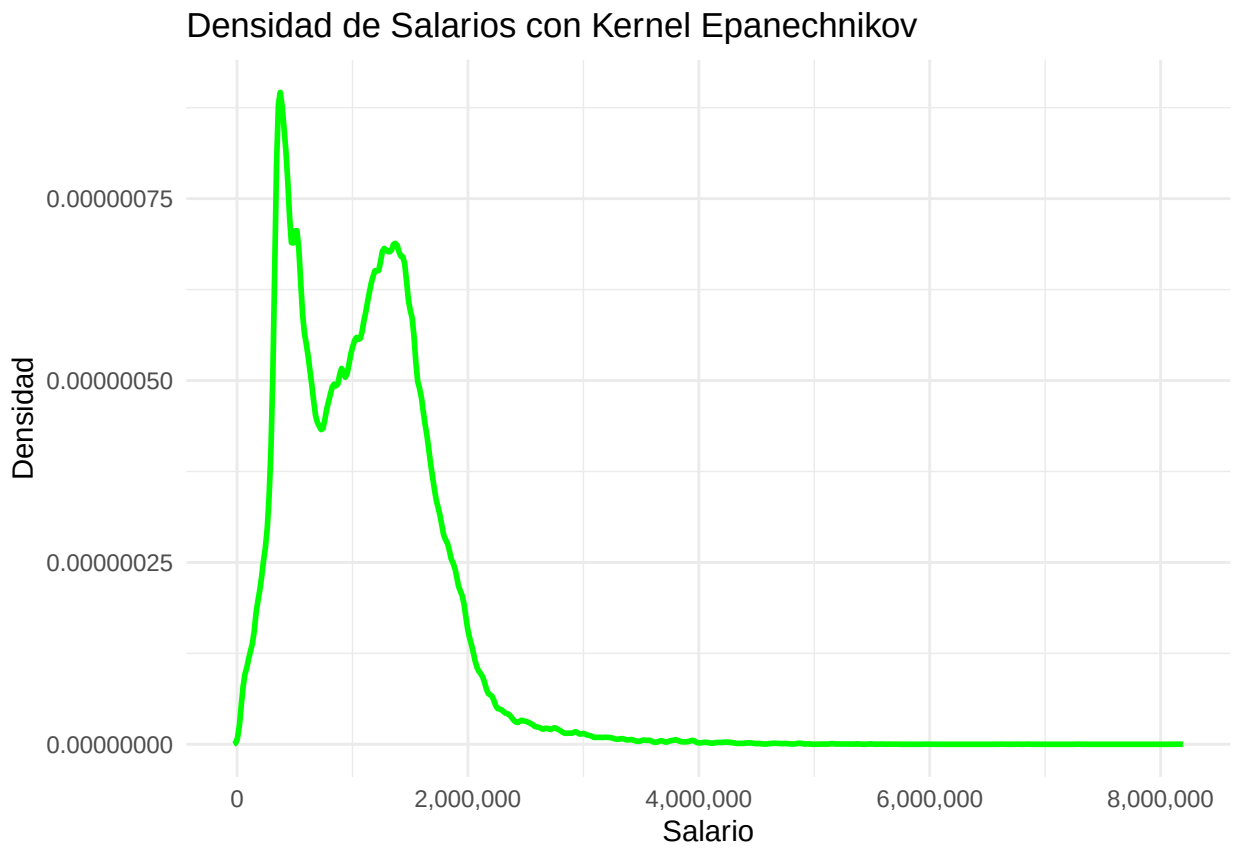
```
biweight <- crear_grafico_kernel(densidades[[1]],
                                 "Densidad de Salarios con Kernel Biweight", "blue")
print(biweight)
```



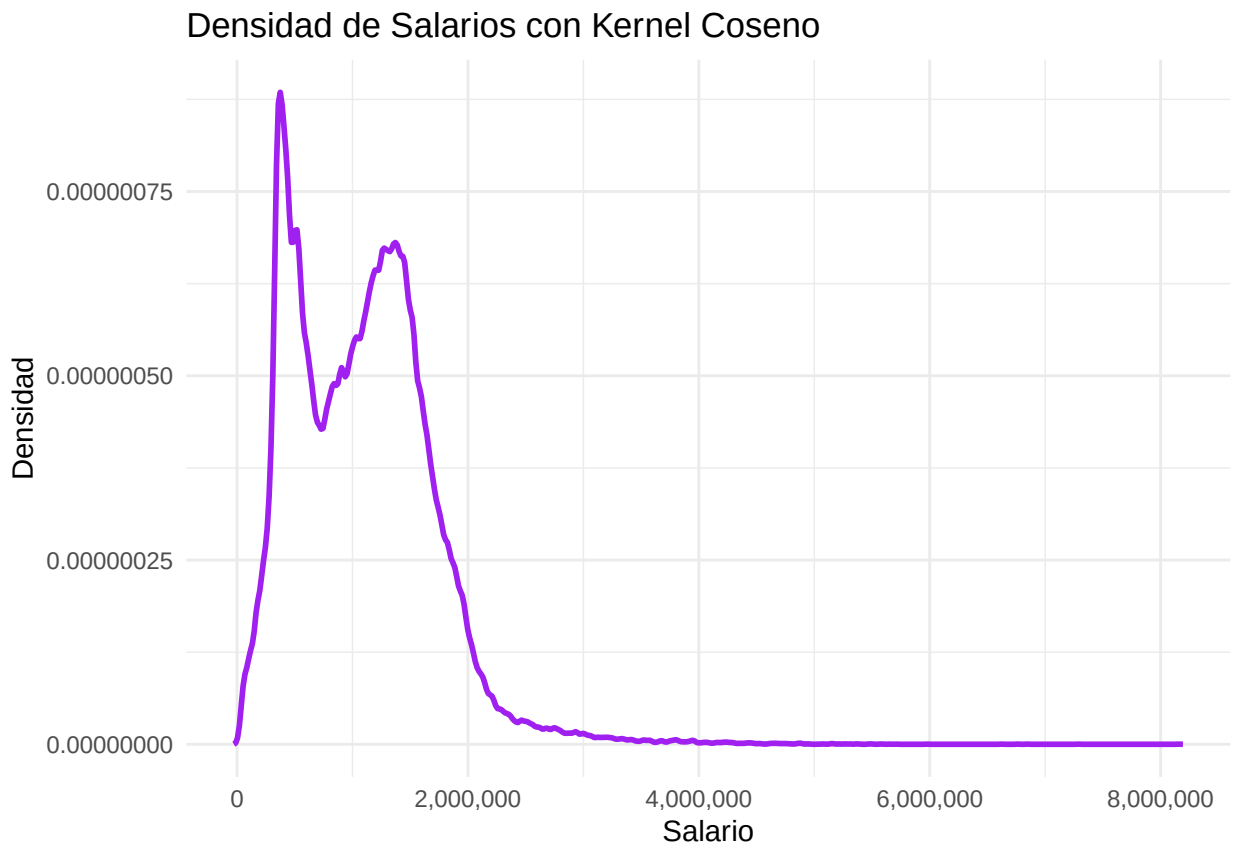

```
gaussian <- crear_grafico_kernel(densidades[[2]],
                                 "Densidad de Salarios con Kernel Gaussiano", "red")
print(gaussian)
```



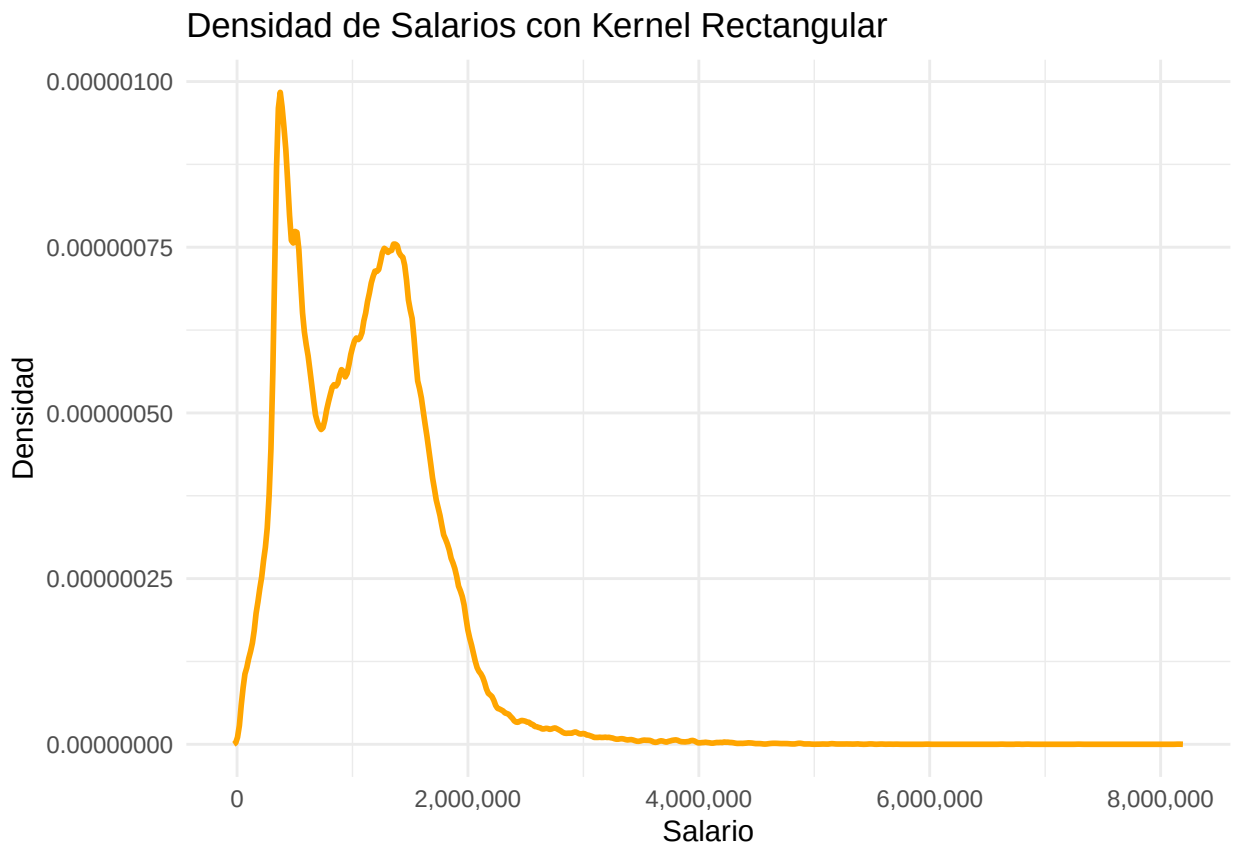

```
epanechnikov <- crear_grafico_kernel(densidades[[3]],
                                     "Densidad de Salarios con Kernel Epanechnikov","green")
print(epanechnikov)
```



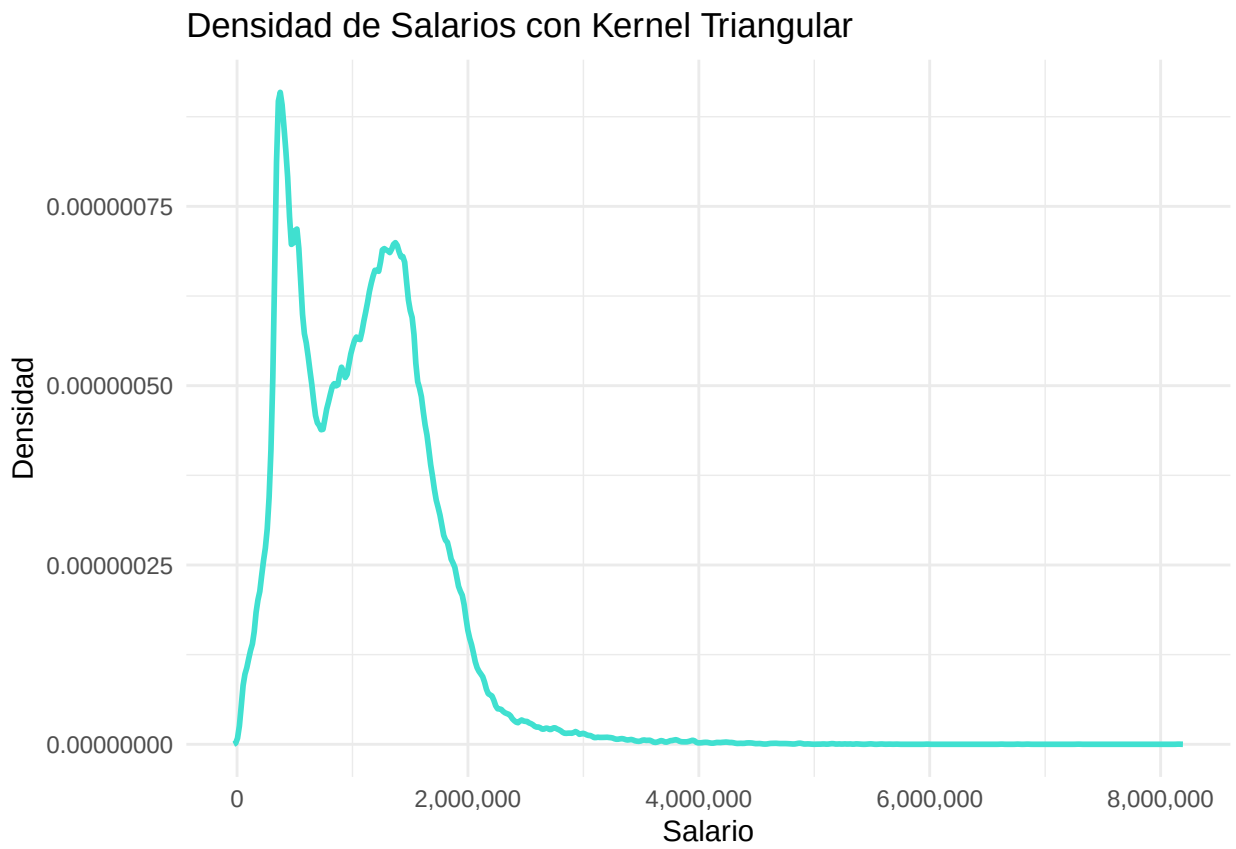

```
cosine <- crear_grafico_kernel(densidades[[4]],
                               "Densidad de Salarios con Kernel Coseno", "purple")
print(cosine)
```




```
rectangular <- crear_grafico_kernel(densidades[[5]],
                                    "Densidad de Salarios con Kernel Rectangular", "orange")
print(rectangular)
```




```
triangular <- crear_grafico_kernel(densidades[[6]],
                                   "Densidad de Salarios con Kernel Triangular", "turquoise")
print(triangular)
```

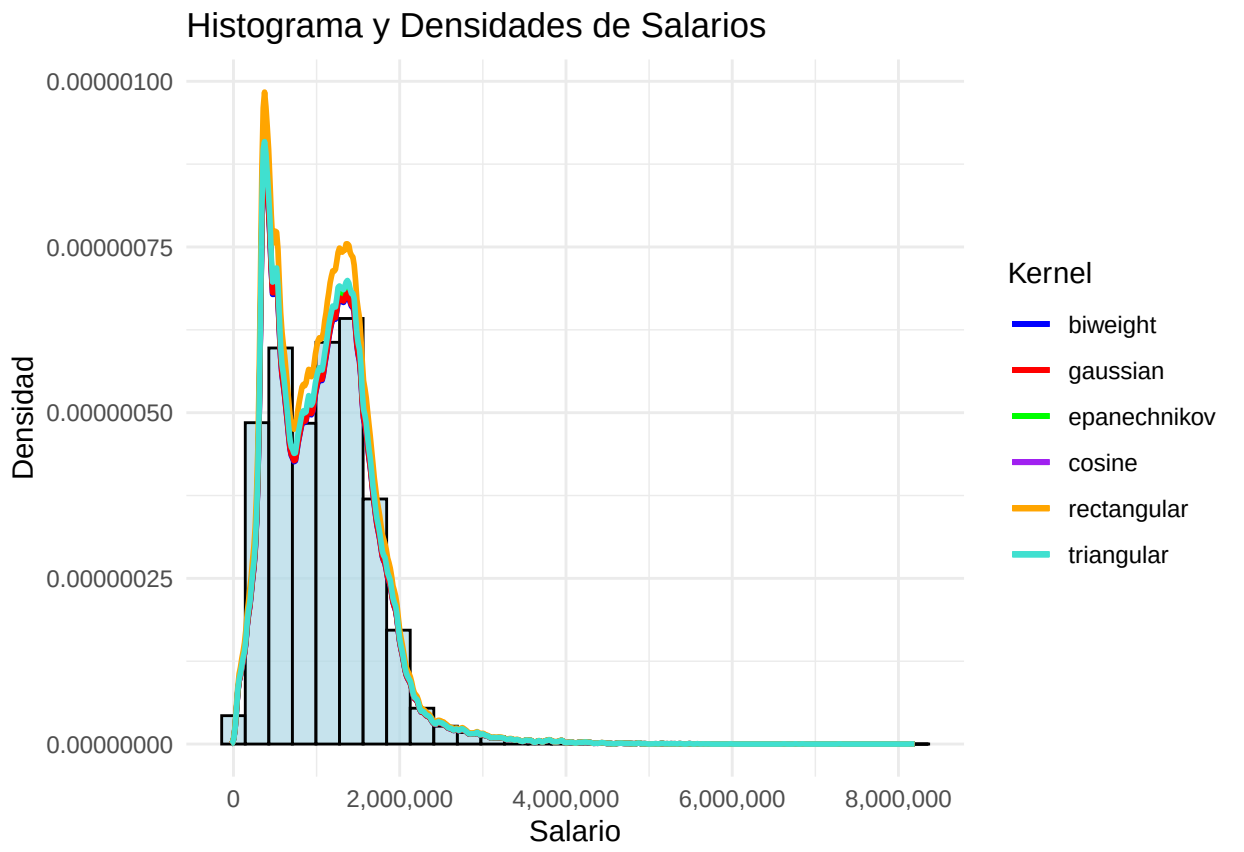


3.En un solo gráfico muestre los resultados de las densidades con el histograma.

Para este punto se toma el histograma del punto 1, pero se le hace un ajuste para que no coloque la frecuencia en el eje y si no la densidad puesto que, es lo que interesa en este caso y hace que se logre ajustar con las curvas de densidad por kernel graficadas anteriormente.

```
hist_salarios_densidad <- ggplot(BD, aes(x = Salario, y = after_stat(density))) +
  geom_histogram(fill = "lightblue", color = "black", alpha = 0.7) +
  labs(title = "Histograma y Densidades de Salarios", x = "Salario", y = "Densidad") +
  scale_x_continuous(labels = scales::comma_format()) +
  scale_y_continuous(labels = scales::comma_format()) +
  theme_minimal()

juntos <- hist_salarios_densidad +
  geom_line(data = data.frame(x = densidades[[1]]$x, y = densidades[[1]]$y, color = Kernels[1]),
            aes(x = x, y = y, color = Kernels[1]), size = 1) +
  geom_line(data = data.frame(x = densidades[[2]]$x, y = densidades[[2]]$y, color = Kernels[2]),
            aes(x = x, y = y, color = Kernels[2]), size = 1) +
  geom_line(data = data.frame(x = densidades[[3]]$x, y = densidades[[3]]$y, color = Kernels[3]),
            aes(x = x, y = y, color = Kernels[3]), size = 1) +
  geom_line(data = data.frame(x = densidades[[4]]$x, y = densidades[[4]]$y, color = Kernels[4]),
            aes(x = x, y = y, color = Kernels[4]), size = 1) +
  geom_line(data = data.frame(x = densidades[[5]]$x, y = densidades[[5]]$y, color = Kernels[5]),
            aes(x = x, y = y, color = Kernels[5]), size = 1) +
  geom_line(data = data.frame(x = densidades[[6]]$x, y = densidades[[6]]$y, color = Kernels[6]),
            aes(x = x, y = y, color = Kernels[6]), size = 1) +
  scale_color_manual(values = c("blue", "red", "green", "purple", "orange",
```

```
                                "turquoise"), labels = Kernels) +
  labs(color = "Kernel")
print(juntos)
```



Parte III

1. Explique en que consiste el criterio de información de Akaike (AIC) y cuál es el criterio de selección entre dos modelos.

El criterio de información de Akaike (AIC) es una medida de la calidad de un modelo dentro de un conjunto de modelos. Brinda información sobre la complejidad de un modelo y su exactitud, pues describe el sesgo y la varianza presentes en el modelo estadístico en estudio. Con el AIC se permite determinar cuál modelo es el más apropiado entre los modelos estadísticos propuestos. Como criterio de selección, se considera como mejor modelo aquel cuyo valor de AIC es menor. Se debe tener presente que el AIC no es prueba de hipótesis, por lo que no indica si un modelo es de calidad, es decir, todos los modelos pueden ser erróneos y el AIC solo indica entre ellos cúal es el que mejor se ajusta.

2. Análisis AIC de los salarios sin filtrar por Sexo para determinar que densidad paramétrica es la que más se le aproxima.

A continuación, se presenta la implementación del criterio AIC mediante la función model_select del paquete univariateML

```
#-----Análisis AIC-----|
```

```r
#Comparación densidades paramétricas por el criterio AIC

salarios <- BD$Salario
model_select(salarios, models = univariateML_models, criterion = "aic",
             na.rm = FALSE)
```

```
## Maximum likelihood estimates for the Skew Student-t model
##      mean         sd         nu         xi
## 1.064e+06  6.180e+05  5.387e+01  2.401e+00
```

De acuerdo al resultado proporcionado, la densidad paramétrica que más se ajusta a los datos de los salarios es la t-student.

3. Utilizando el paquete rriskDistributions, y la función fit.cont, bajo el criterio AIC, replique el punto 2 de esta parte.

Empleando la función sugerida se obtiene lo siguiente:

```r
fit.cont(salarios)
```

	logL	AIC	BIC	Chisq(value)	Chisq(p)	AD(value)	H(AD)	KS(value)	H(KS)
Cauchy	-1579406.45	3158816.91	3158836.05	64966.97	0	2262.66	rejected	0.12	rejected
Uniform	NULL	NULL	NULL	Inf	0	Inf	NULL	0.08	rejected
Lognormal	-1559929.72	3119863.45	3119882.59	35838.95	0	1826.63	rejected	0.10	rejected
Weibull	-1552303.26	3104610.52	3104629.67	11129.72	0	458.63	rejected	0.05	rejected
F	-1852800.9	3705605.8	3705624.94	2312033.080		47820.62	NULL	0.60	rejected
Student	-1925112.91	3850227.82	3850237.39	4676102.550		94832.02	NULL	0.79	rejected

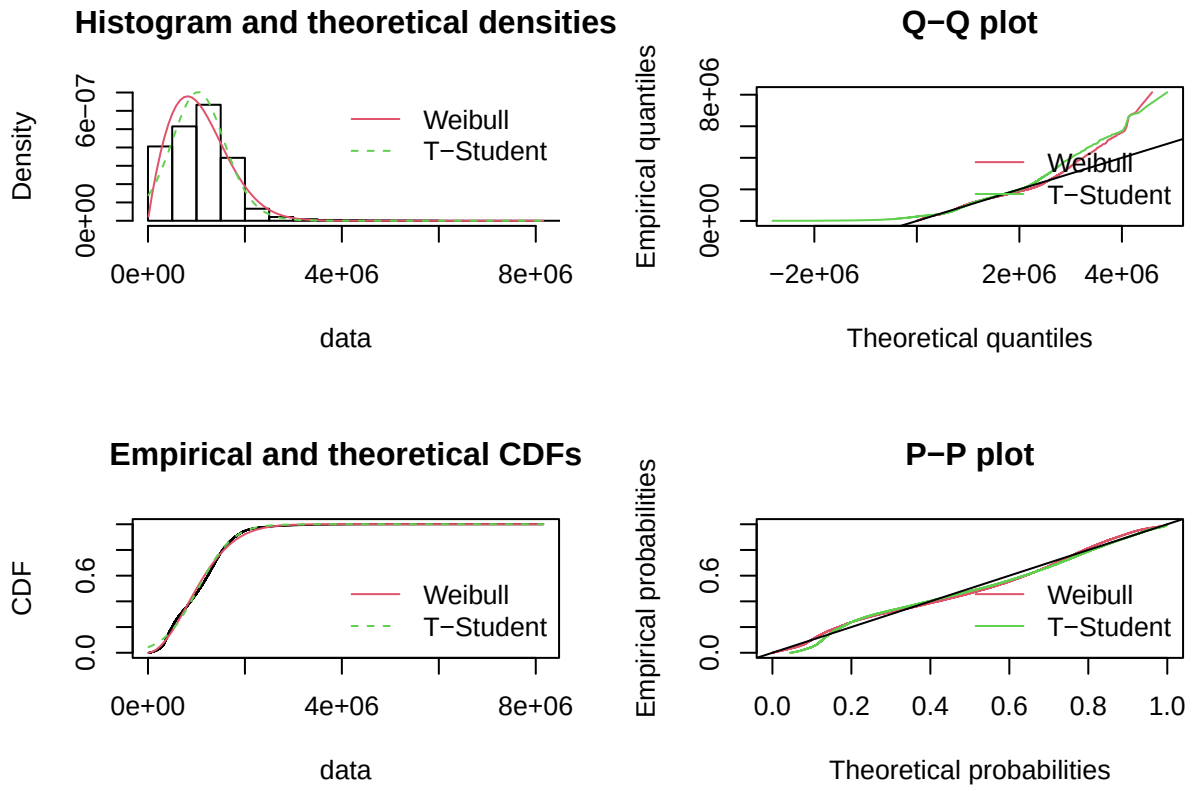
Usando el criterio AIC la densidad Weibull es el que presenta menor valor de esta medida, por ende, se toma como la mejor distribución para los salarios.

4.Comente y compare los resultados de los puntos 2 y 3, seleccione una distribución

Los resultados obtenidos en los puntos 2 y 3 son distintos, con el primer método se obtuvo que la densidad que mejor se aproxima es la t-Student. Sin embargo, con la segunda forma se tiene que la mejor es la densidad de Weibull. Para determinar cúal de estas distribuciones se ajusta mejor, se procede a analizar los siguientes gráficos comparativos:

```r
#----Elegir distribución---|

#Comparación gráfica entre la distribución weibull y la t-student

fw <- fitdist(salarios, "weibull")
ft <- fitdist(salarios, "t.scaled", start=list(df=3,mean=mean(salarios),sd= sd(salarios)))

par(mfrow = c(2,2))
leyenda <-c("Weibull", "T-Student")
```

```
denscomp(list(fw, ft), legendtext = leyenda)
qqcomp(list(fw, ft), legendtext = leyenda)
cdfcomp(list(fw, ft), legendtext = leyenda)
ppcomp(list(fw, ft), legendtext = leyenda)
```



A partir del gráfico del histograma con las densidades teóricas, es posible observar que la densidad representada por la línea roja es la que mejor se ajusta al histograma de los datos de los salarios. Entonces, para este caso, la distribución Weibull es la que mejor se aproxima.

Con el Q-Q plot se permiten ver las diferencias entre la distribución teórica de los datos y la empírica. La distribución t-student y weibull se muestran similarmente ajustados aproximadamente entre los cuántiles teóricos 0 y 2e+06. Pero, la distribución t-student representado por el gráfico verde, se encuentra más alejado de la línea negra al inicio y al final de esta. Por ende, la distribución Weibull se ajusta mejor.

Para el gráfico de la distribución acumulada teórica y la empírica, las diferencias en los ajustes entre ambas distribuciones es menos evidente. Lo mismo para el p-p plot, por lo que, a partir de estos gráficos no se puede deducir cuál es el mejor.

Por tanto, si consideramos los primeros dos criterios, la distribución Weibull se ajusta mejor a los datos de los salarios en estudio.

También, la función fitdist brinda información sobre la medida AIC:

```
print(summary(fw))
```

```
## Fitting of the distribution ' weibull ' by maximum likelihood
## Parameters :
```

```
##           estimate Std. Error
## shape 1.889217e+00          0
## scale 1.218014e+06        NaN
## Loglikelihood:  -1552303   AIC:  3104611   BIC:  3104630
## Correlation matrix:
##       shape scale
## shape     1   NaN
## scale   NaN     1
```

```
print(summary(ft))
```

```
## Fitting of the distribution ' t.scaled ' by maximum likelihood
## Parameters :
##          estimate Std. Error
## df   1.329545e+01      0.000
## mean 1.034088e+06        NaN
## sd   5.569787e+05    131.072
## Loglikelihood:  -1558287   AIC:  3116581   BIC:  3116610
## Correlation matrix:
##        df mean   sd
## df      1  NaN -Inf
## mean  NaN    1  NaN
## sd   -Inf  NaN    1
```

Donde Weibull presenta el menor valor AIC. Con lo cual, se confirma que la distribución Weibull es la que mejor se ajusta.

5.Intervalo de confianza para la media y la desviación estándar usando la función bootstrapml

Para obtener los intervalos de confianza se ejecuta el siguiente código:

```
densidad_salario <- mlweibull(salarios)

#intervalo confianza media
bootstrapml(densidad_salario, map = function(x) x[2]*gamma(1+(1/x[1])))
```

```
##    2.5%   97.5%
## 1077264 1084414
```

```
#intervalo confianza desviación estándar
bootstrapml(densidad_salario, map = function(x) sqrt(x[2]^2*(gamma(1+2/x[1])-(gamma(1+1/x[1]))^2)))
```

```
##     2.5%    97.5%
## 592271.6 597561.0
```

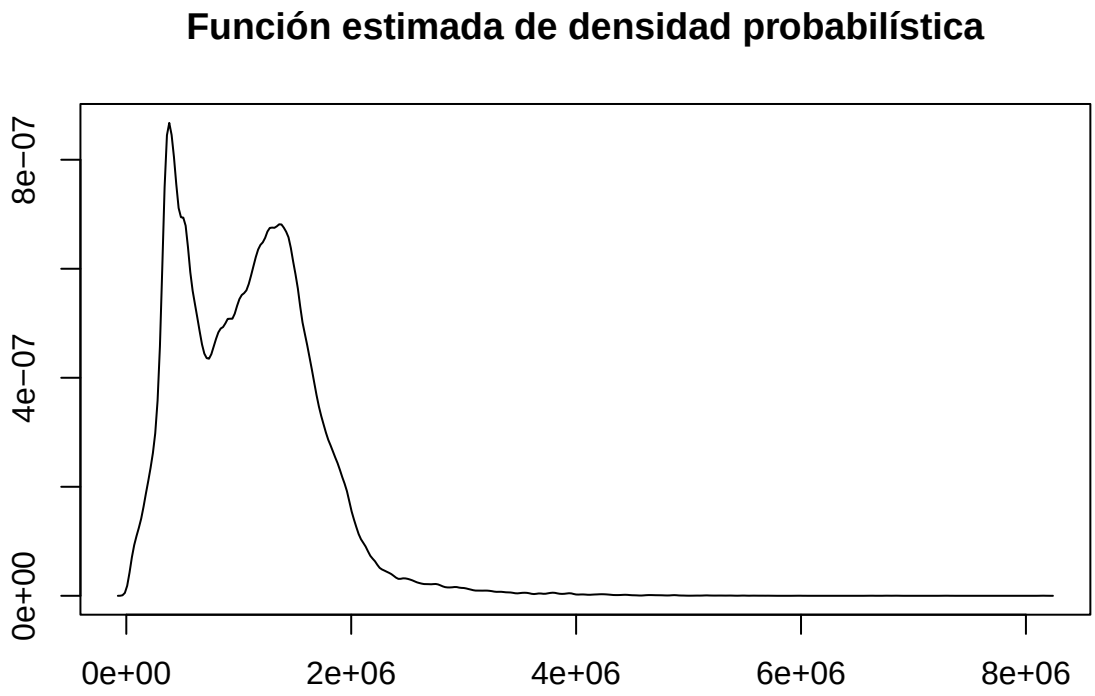
Del cual, se obtiene que el intervalo de confianza al 97.5% para la media es [1077264, 1084414] y para la desviación estándar es [592271.6, 597561.0].

Parte IV

1.Del paquete ks de R, describa brevemente que realiza la función kde y realice como ejemplo un gráfico de la densidad con U. Salarios sin filtrar por sexo.

Inicialmente el paquete *ks* de R es un paquete orientado a suavizar datos univariados o multivariados mediante kernels. Dentro de este paquete se encuentra la función *Kernel density estimate (kde)* la cual estima la función de densidad de probabilidad para datos de 1 a 6 dimensiones de forma no paramétrica.

```
KDE <- kde(BD$Salario)
plot(KDE, main = "Función estimada de densidad probabilística", ylab = "", xlab = "")
```



2.Del paquete boot de R, describa brevemente la función boot.ci

Este paquete contiene funciones y bases de datos para bootstrapping del libro *Bootstrap Methods and Their Application*. La función boot.ci genera 5 tipos diferentes de intervalos de confianza no paramétricos de dos colas iguales: Intervalos de Aproximación normal, intervalos básicos bootstrap, intervalos bootstrap studentnizados, intervalos percentil bootstrap e intervalos ajustados percentil bootstrap, de los cuales se puede hacer una selección según preferencia.

```
resultado_boot <- boot(data=BD$Salario,
                statistic=function(y,indices) mean(y[indices]),
                R=1000)
```

3.Realice un Bootstrap de la media de los U. Salarios sin filtrar por sexo, (para esto deben crear una función que contenga índices) utilizando la función boot.

a.Estime la media Bootstrap (ver la variable t) y compárela con la media de los datos originales (ver la variable t0).

```r
media_real <- resultado_boot$t0
media_boot <- mean(resultado_boot$t)
medias_y_diff <- data.frame(media_real, media_boot, media_real-media_boot)
colnames(medias_y_diff) <- c("Media real", "Media boot", "Diferencia")

print(medias_y_diff)
```

```
##   Media real Media boot Diferencia
## 1    1080558    1080569  -11.21717
```

b.Realice un histograma con los resultados del Bootstrap. Resulta evidente que, aunque existe una diferencia en los resultados, esta no es de gran magnitud, mostrándose así que la media obtenida mediante el método bootstrap es una aproximación muy certera para el caso en cuestión.

```r
hist(resultado_boot$t, main = "Histograma de medias bootstrap", xlab = "Medias", ylab = "Frecuencia")
```