Examen 2

Guerrero Garcilazo Maria Fernanda, Walls Salcedo Carlos

2025-04-29

Un gerente del banco está preocupado porque cada vez más clientes están abandonando sus servicios de tarjeta de crédito. Este conjunto de datos consta de más de 10,000 clientes que fueron seleccionados a través de un muestreo aleatorio sin reemplazo e incluye información sobre su edad, salario, estado civil, límite de la tarjeta de crédito, categoría de la tarjeta de crédito, entre otros.

Para la solución de este examen, utiliza solo como variables cuantitativas: Customer_Age, Months_on_book, Credit_Limit y Total_Trans_Amt.

```
library(readxl)
library(dplyr)
```

```
##
## Adjuntando el paquete: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(patchwork)
library(moments)
library(e1071)
```

```
##
## Adjuntando el paquete: 'e1071'
```

```
## The following objects are masked from 'package:moments':
##
##     kurtosis, moment, skewness
```

```
library(fitdistrplus)
```

```
## Warning: package 'fitdistrplus' was built under R version 4.4.2
```

```
## Cargando paquete requerido: MASS
```

```
##
## Adjuntando el paquete: 'MASS'
```

```
## The following object is masked from 'package:patchwork':
##
##     area
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
## Cargando paquete requerido: survival
```

```
library(nortest)
```

```
banco <- read_excel("C:/Users/DELL/Downloads/BankChurners.xlsx")
head(banco)
```

```
## # A tibble: 6 × 13
##   CLIENTNUM Attrition_Flag   Customer_Age Gender Dependent_count Education_Level
##       <dbl> <chr>                   <dbl> <chr>            <dbl> <chr>
## 1 768805383 Existing Custom…           45 M                    3 High School
## 2 818770008 Existing Custom…           49 F                    5 Graduate
## 3 713982108 Existing Custom…           51 M                    3 Graduate
## 4 769911858 Existing Custom…           40 F                    4 High School
## 5 709106358 Existing Custom…           40 M                    3 Uneducated
## 6 713061558 Existing Custom…           44 M                    2 Graduate
## # ℹ 7 more variables: Marital_Status <chr>, Income_Category <chr>,
## #   Card_Category <chr>, Months_on_book <dbl>, Total_Relationship_Count <dbl>,
## #   Credit_Limit <dbl>, Total_Trans_Amt <dbl>
```

Consigna 1: Limpieza de datos y manipulación.

Contamos los NA’s de nuestras variables cuantitativas

```
colSums(is.na(banco[c("Customer_Age", "Months_on_book", "Credit_Limit","Total_Trans_Amt")]))
```

```
##    Customer_Age  Months_on_book    Credit_Limit Total_Trans_Amt
##              29              19               0               0
```

Reemplazamos los NA's con la media de cada variable

```
banco$Customer_Age[is.na(banco$Customer_Age)] <- mean(banco$Customer_Age, na.rm =T)
banco$Months_on_book[is.na(banco$Months_on_book)] <- mean(banco$Months_on_book, na.rm =T)
banco$Credit_Limit[is.na(banco$Credit_Limit)] <- mean(banco$Credit_Limit, na.rm =T)
banco$Total_Trans_Amt[is.na(banco$Total_Trans_Amt)] <- mean(banco$Total_Trans_Amt, na.rm =T)
```

Verificamos que ya no existan NA's

```
colSums(is.na(banco[c("Customer_Age", "Months_on_book", "Credit_Limit", "Total_Trans_Amt")]))
```

```
##    Customer_Age  Months_on_book    Credit_Limit Total_Trans_Amt
##               0               0               0               0
```

```
#SUMMARY para ver cuáles son mis variables categóricas
summary(banco)
```

```
##    CLIENTNUM         Attrition_Flag      Customer_Age      Gender
##  Min.   :708082083   Length:10127       Min.   :26.00   Length:10127
##  1st Qu.:713036770   Class :character   1st Qu.:41.00   Class :character
##  Median :717926358   Mode  :character   Median :46.00   Mode  :character
##  Mean   :739177606                      Mean   :46.32
##  3rd Qu.:773143533                      3rd Qu.:52.00
##  Max.   :828343083                      Max.   :73.00
##  Dependent_count Education_Level    Marital_Status     Income_Category
##  Min.   :0.000   Length:10127       Length:10127       Length:10127
##  1st Qu.:1.000   Class :character   Class :character   Class :character
##  Median :2.000   Mode  :character   Mode  :character   Mode  :character
##  Mean   :2.346
##  3rd Qu.:3.000
##  Max.   :5.000
##  Card_Category      Months_on_book  Total_Relationship_Count  Credit_Limit
##  Length:10127       Min.   :13.00   Min.   :1.000            Min.   : 1438
##  Class :character   1st Qu.:32.00   1st Qu.:3.000            1st Qu.: 2555
##  Mode  :character   Median :36.00   Median :4.000            Median : 4549
##                     Mean   :35.93   Mean   :3.813            Mean   : 8632
##                     3rd Qu.:40.00   3rd Qu.:5.000            3rd Qu.:11068
##                     Max.   :56.00   Max.   :6.000            Max.   :34516
##  Total_Trans_Amt
##  Min.   :   55
##  1st Qu.: 2318
##  Median : 3798
##  Mean   : 4426
##  3rd Qu.: 5906
##  Max.   :24574
```

Usamos Table para obtener las clasificaciones

```
table(banco$Gender)
```

```
##
##    F    M
## 5358 4769
```

```
table(banco$Attrition_Flag)
```

```
##
## Attrited Customer Existing Customer
##              1627              8500
```

```
table(banco$Education_Level)
```

```
##
##       College     Doctorate      Graduate   High School Post-Graduate
##          1013           451          3128          2013           516
##    Uneducated       Unknown
##          1487          1519
```

```
table(banco$Marital_Status)
```

```
##
## Divorced  Married  MArried   Single  Unknown
##      748     4684        3     3943      749
```

```
table(banco$Income_Category)
```

```
##
##        $120K +    $40K - $60K    $60K - $80K   $80K - $120K Less than $40K
##            727           1790           1402           1535           3561
##        Unknown
##           1112
```

```
table(banco$Card_Category)
```

```
##
##     Blue     BLue     BLUE     Gold Platinum   Silver
##     9409        1        4      116       20      564
```

Corregimos según los tables y volvemos a mostrarlos

```
banco$Marital_Status[banco$Marital_Status=="MArried"]<-"Married"
banco$Card_Category[banco$Card_Category=="BLue"]<-"Blue"
banco$Card_Category[banco$Card_Category=="BLUE"]<-"Blue"


table(banco$Gender)
```

```
##
##    F    M
## 5358 4769
```

```
table(banco$Attrition_Flag)
```

```
##
## Attrited Customer Existing Customer
##              1627              8500
```

```
table(banco$Education_Level)
```

```
##
##       College     Doctorate      Graduate   High School Post-Graduate
##          1013           451          3128          2013           516
##    Uneducated       Unknown
##          1487          1519
```

```
table(banco$Marital_Status)
```

```
##
## Divorced  Married   Single  Unknown
##      748     4687     3943      749
```

```
table(banco$Income_Category)
```

```
##
##        $120K +    $40K - $60K    $60K - $80K   $80K - $120K Less than $40K
##            727           1790           1402           1535           3561
##        Unknown
##           1112
```

```
table(banco$Card_Category)
```

```
##
##     Blue     Gold Platinum   Silver
##     9414      116       20      564
```

Vemos si las categóricas tienen NA's

```
colSums(is.na(banco[c("Gender","Attrition_Flag","Marital_Status","Income_Category","Card_Category")]))
```

```
##          Gender  Attrition_Flag  Marital_Status Income_Category   Card_Category
##               0               0               0               0              13
```

Reemplazamos NA con la moda en Card_Category, según el table de Card_Category, la categoría que más se repite es "Blue"

```
banco$Card_Category[is.na(banco$Card_Category)] <- "Blue"
```

Volvemos a colocar un colSums

```
colSums(is.na(banco[c("Gender","Attrition_Flag","Marital_Status","Income_Category","Card_Category")]))
```

```
##          Gender  Attrition_Flag  Marital_Status Income_Category   Card_Category
##               0               0               0               0               0
```

Resumen de las variables cuantitativas agrupados por tipo de tarjeta

```
resumen_Customer_Age <- banco %>%
  group_by(Card_Category) %>%
  summarise(
    Num_Obs = n(),
    Min = min(Customer_Age, na.rm = TRUE),
    Big_inf = boxplot.stats(Customer_Age)$stats[1],
    Q1 = quantile(Customer_Age, 0.25, na.rm = TRUE),
    Mediana = median(Customer_Age, na.rm = TRUE),
    Media = mean(Customer_Age, na.rm = TRUE),
    Q3 = quantile(Customer_Age, 0.75, na.rm = TRUE),
    Big_sup = boxplot.stats(Customer_Age)$stats[5],
    Max = max(Customer_Age, na.rm = TRUE)
  )

resumen_Months_on_book <- banco %>%
  group_by(Card_Category) %>%
  summarise(
    Num_Obs = n(),
    Min = min(Months_on_book, na.rm = TRUE),
    Big_inf = boxplot.stats(Months_on_book)$stats[1],
    Q1 = quantile(Months_on_book, 0.25, na.rm = TRUE),
    Mediana = median(Months_on_book, na.rm = TRUE),
    Media = mean(Months_on_book, na.rm = TRUE),
    Q3 = quantile(Months_on_book, 0.75, na.rm = TRUE),
    Big_sup = boxplot.stats(Months_on_book)$stats[5],
    Max = max(Months_on_book, na.rm = TRUE)
  )

resumen_Credit_Limit<- banco %>%
  group_by(Card_Category) %>%
  summarise(
    Num_Obs = n(),
    Min = min(Credit_Limit, na.rm = TRUE),
    Big_inf = boxplot.stats(Credit_Limit)$stats[1],
    Q1 = quantile(Credit_Limit, 0.25, na.rm = TRUE),
    Mediana = median(Credit_Limit, na.rm = TRUE),
    Media = mean(Credit_Limit, na.rm = TRUE),
    Q3 = quantile(Credit_Limit, 0.75, na.rm = TRUE),
    Big_sup = boxplot.stats(Credit_Limit)$stats[5],
    Max = max(Credit_Limit, na.rm = TRUE))


resumen_Total_Trans_Amt<- banco %>%
  group_by(Card_Category) %>%
  summarise(
    Num_Obs = n(),
    Min = min(Total_Trans_Amt, na.rm = TRUE),
    Big_inf = boxplot.stats(Total_Trans_Amt)$stats[1],
```

```
    Q1 = quantile(Total_Trans_Amt, 0.25, na.rm = TRUE),
    Mediana = median(Total_Trans_Amt, na.rm = TRUE),
    Media = mean(Total_Trans_Amt, na.rm = TRUE),
    Q3 = quantile(Total_Trans_Amt, 0.75, na.rm = TRUE),
    Big_sup = boxplot.stats(Total_Trans_Amt)$stats[5],
    Max = max(Total_Trans_Amt, na.rm = TRUE))


print(resumen_Customer_Age)
```

```
## # A tibble: 4 × 10
##   Card_Category Num_Obs   Min Big_inf    Q1 Mediana Media    Q3 Big_sup   Max
##   <chr>           <int> <dbl>   <dbl> <dbl>   <dbl> <dbl> <dbl>   <dbl> <dbl>
## 1 Blue             9427    26      26  41        46  46.4    52      68    73
## 2 Gold              116    29      29  41        46  45.4    49      61    63
## 3 Platinum           20    39      39  43.8      48  47.5    51      56    56
## 4 Silver            564    26      28  41        45  45.6    50      63    65
```

```
print(resumen_Months_on_book)
```

```
## # A tibble: 4 × 10
##   Card_Category Num_Obs   Min Big_inf    Q1 Mediana Media    Q3 Big_sup   Max
##   <chr>           <int> <dbl>   <dbl> <dbl>   <dbl> <dbl> <dbl>   <dbl> <dbl>
## 1 Blue             9427    13      18  31        36  36.0  40        53    56
## 2 Gold              116    18      27  33        36  35.4  37.2      44    55
## 3 Platinum           20    23      23  33.5      36  36.2  41.2      46    46
## 4 Silver            564    13      22  32        36  35.4  39        49    56
```

```
print(resumen_Credit_Limit)
```

```
## # A tibble: 4 × 10
##   Card_Category Num_Obs    Min Big_inf     Q1 Mediana  Media    Q3 Big_sup   Max
##   <chr>           <int>  <dbl>   <dbl>  <dbl>   <dbl>  <dbl> <dbl>   <dbl> <dbl>
## 1 Blue             9427  1438.   1438.  2474.    4098  7357.  9065   18927 34516
## 2 Gold              116  3735    6224  22724.   34516 28416. 34516   34516 34516
## 3 Platinum           20 15987   23981  31882.   34516 30283. 34516   34516 34516
## 4 Silver            564  2899    2899  15126.   29306 25110. 34516   34516 34516
```

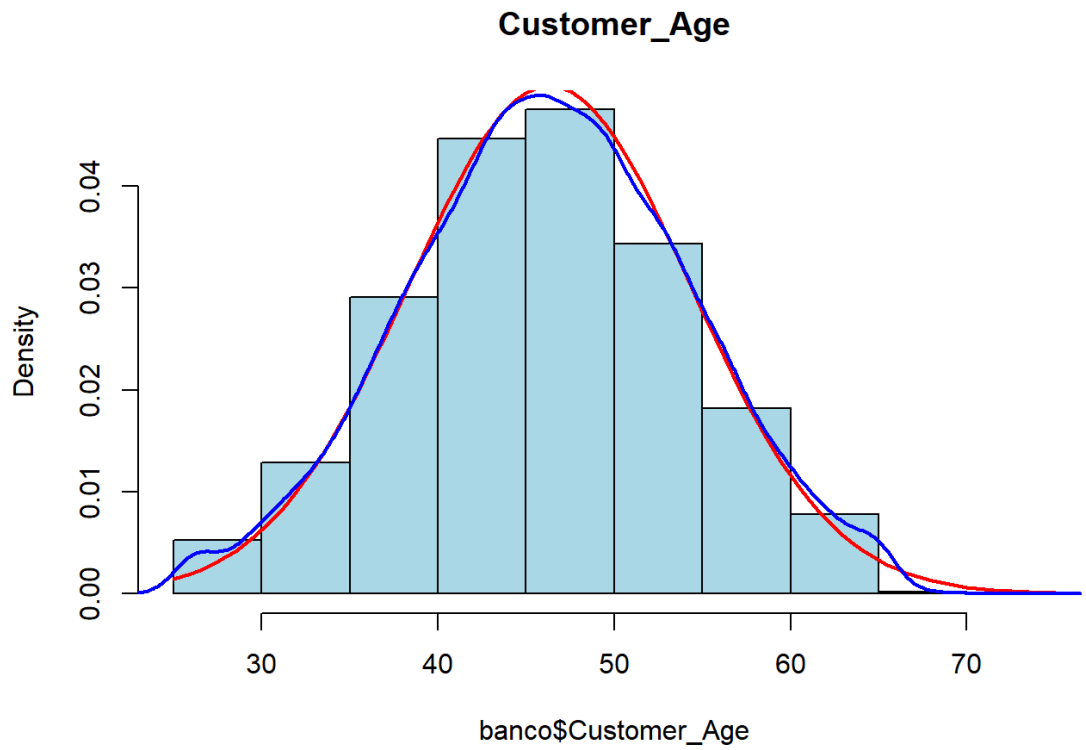
```
print(resumen_Total_Trans_Amt)
```

```
## # A tibble: 4 × 10
##   Card_Category Num_Obs   Min Big_inf    Q1 Mediana Media    Q3 Big_sup   Max
##   <chr>           <int> <dbl>   <dbl> <dbl>   <dbl> <dbl> <dbl>   <dbl> <dbl>
## 1 Blue             9427    55      55 2306.   3775  4414. 5887    11248 24574
## 2 Gold              116   530     530 2313.   4030  4351. 6048.   10524 13098
## 3 Platinum           20   659     659 2748.   4106. 4511. 5337     9117 10765
## 4 Silver            564   169     169 2502.   4149  4629. 6188.   11544 18853
```

Consigna 2: Histogramas de Frecuencias con ggplot2

Primero verificamos si nuestras variables son normales o no con histogramas sencillos

```
#Para Customer_Ager
hist(banco$Customer_Age, probability = TRUE, col = "lightblue", main = "Customer_Age")
curve(dnorm(x, mean = mean(banco$Customer_Age, na.rm=TRUE),
            sd = sd(banco$Customer_Age, na.rm=TRUE)),
      col = "red", lwd = 2, add = TRUE)
lines(density(banco$Customer_Age, na.rm=TRUE), col = "blue", lwd = 2)
```

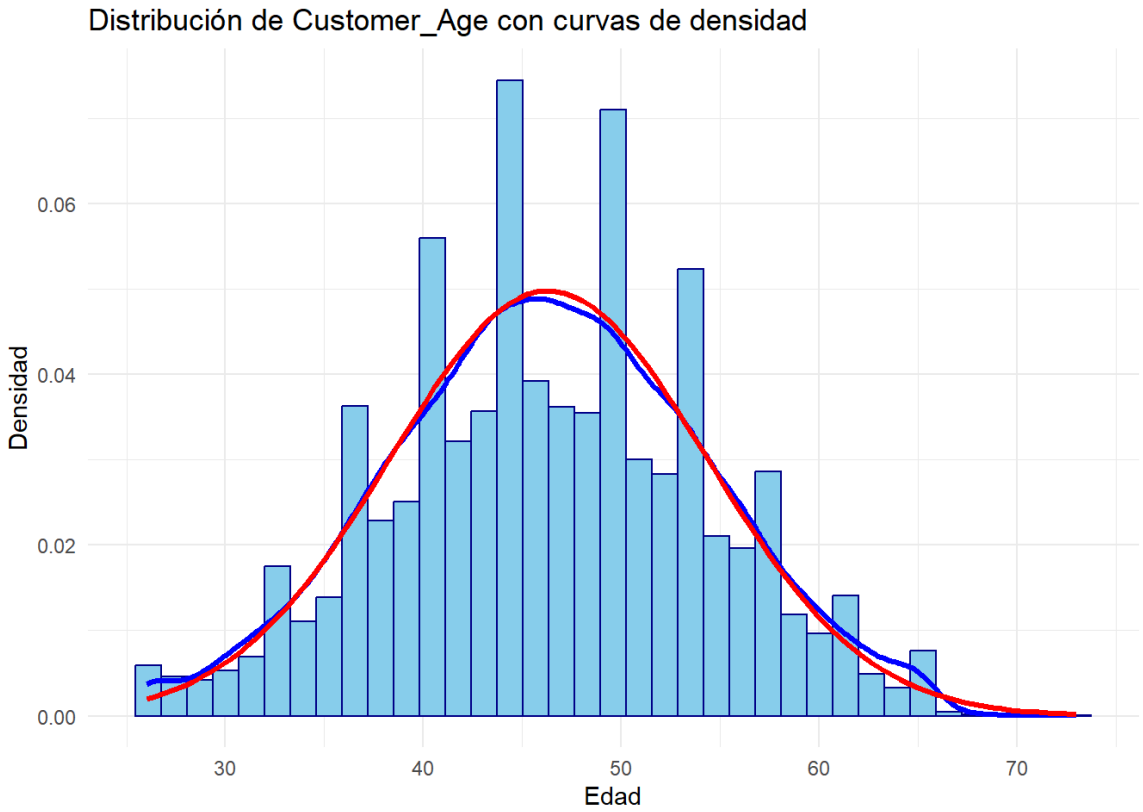


Según nuestra gráfica, es posible que los datos se distribuyan normalmente por lo que usaremos k_scott

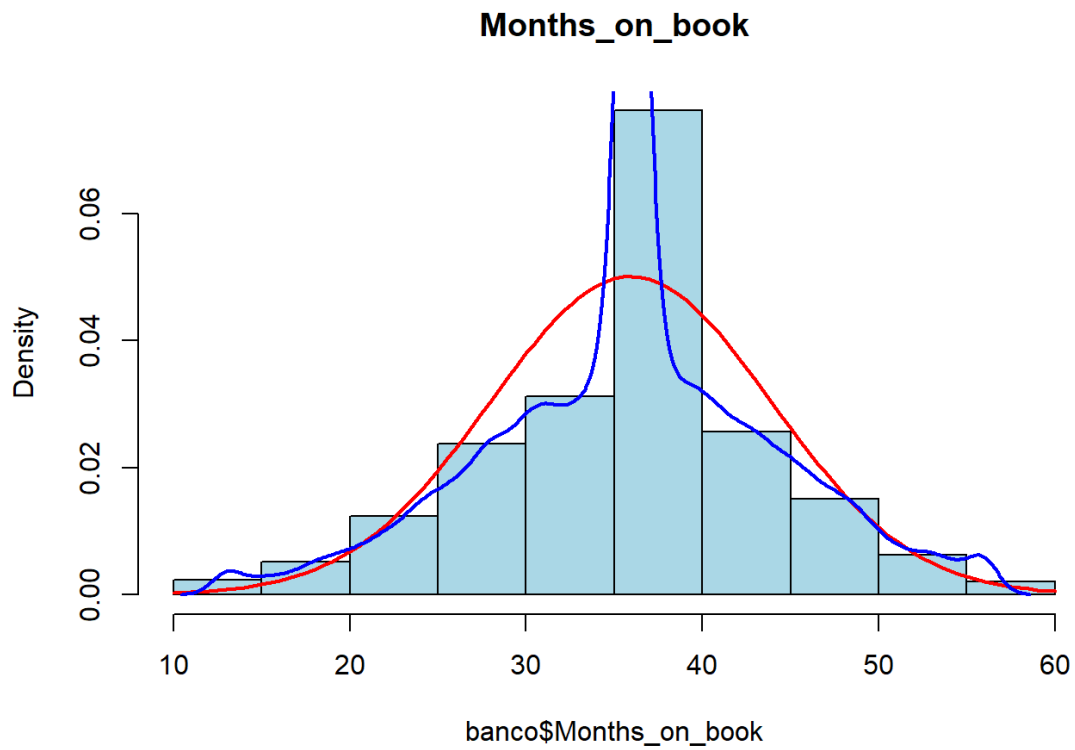
```
k_scott1 <- nclass.scott(banco$Customer_Age)

# Media y desviación estándar
media <- mean(banco$Customer_Age, na.rm = TRUE)
sd <- sd(banco$Customer_Age, na.rm = TRUE)


ggplot(banco, aes(x = Customer_Age)) +
  geom_histogram(aes(y = ..density..), bins = k_scott1, fill = "skyblue", color = "darkblue") +
  geom_density(color = "blue", size = 1.2) +  # Curva empírica
  stat_function(fun = dnorm, args = list(mean = media, sd = sd), color = "red", size = 1.2) +
  labs(title = "Distribución de Customer_Age con curvas de densidad",
       x = "Edad", y = "Densidad") +
  theme_minimal()
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## ℹ Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
## ℹ Please use `after_stat(density)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

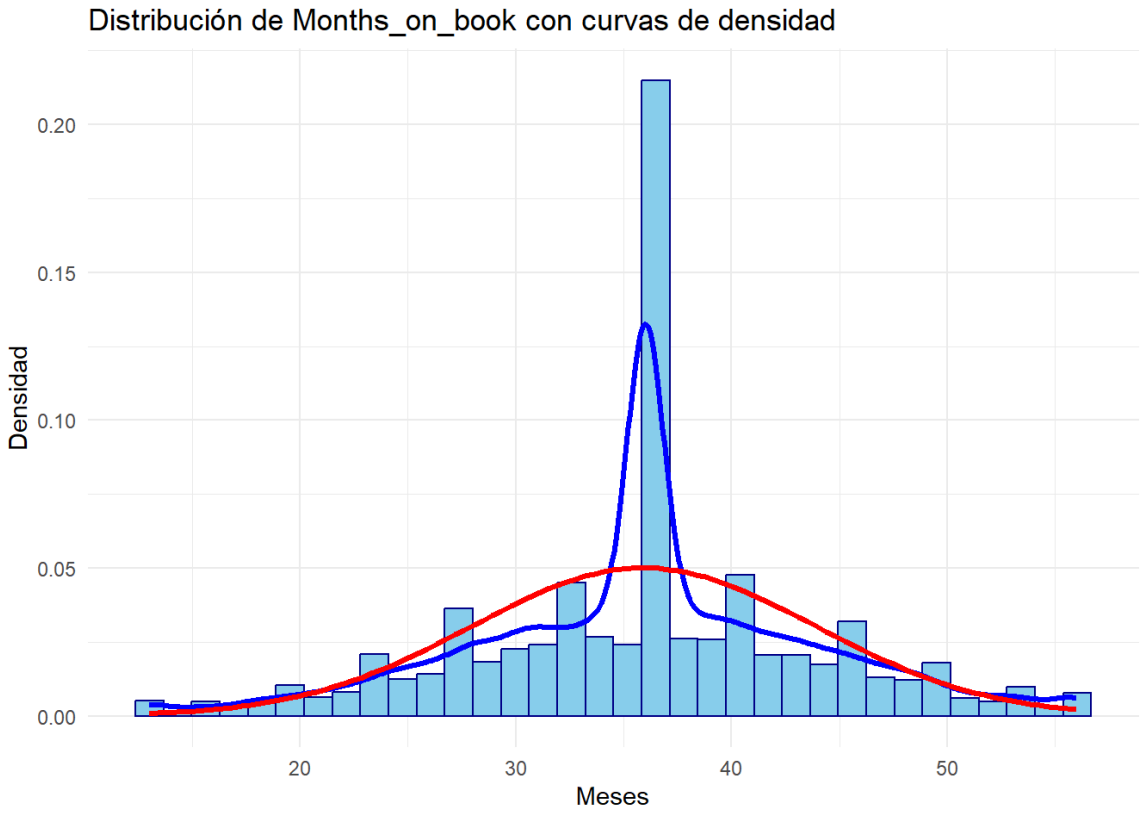



```
hist(banco$Months_on_book, probability = TRUE, col = "lightblue", main = "Months_on_book")
curve(dnorm(x, mean = mean(banco$Months_on_book, na.rm=TRUE),
            sd = sd(banco$Months_on_book, na.rm=TRUE)),
      col = "red", lwd = 2, add = TRUE)
lines(density(banco$Months_on_book, na.rm=TRUE), col = "blue", lwd = 2)
```

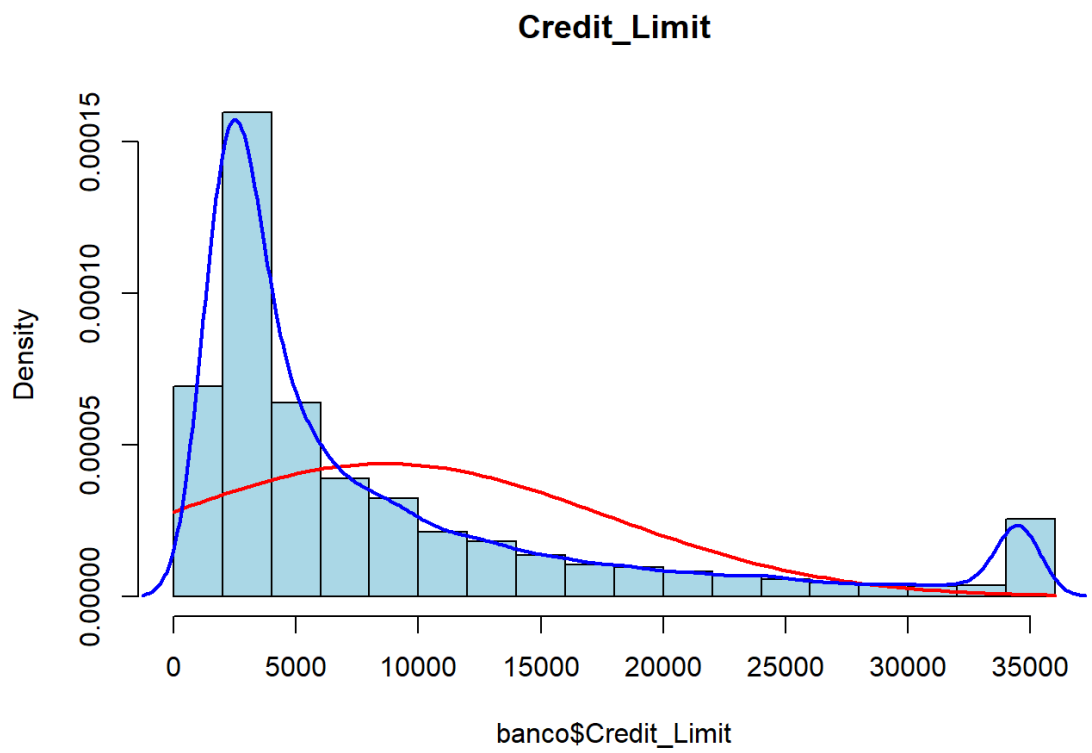


Según nuestra gráfica, los datos tienen forma de campana, usaremos scott

```
k_fd1 <- nclass.scott(banco$Months_on_book)
media1 <- mean(banco$Months_on_book, na.rm = TRUE)
sd1 <- sd(banco$Months_on_book, na.rm = TRUE)

ggplot(banco, aes(x = Months_on_book)) +
  geom_histogram(aes(y = ..density..), bins = k_fd1, fill = "skyblue", color = "darkblue") +
  geom_density(color = "blue", size = 1.2) +
  stat_function(fun = dnorm, args = list(mean = media1, sd = sd1), color = "red", size = 1.2) +
  labs(title = "Distribución de Months_on_book con curvas de densidad",
       x = "Meses", y = "Densidad") +
  theme_minimal()
```

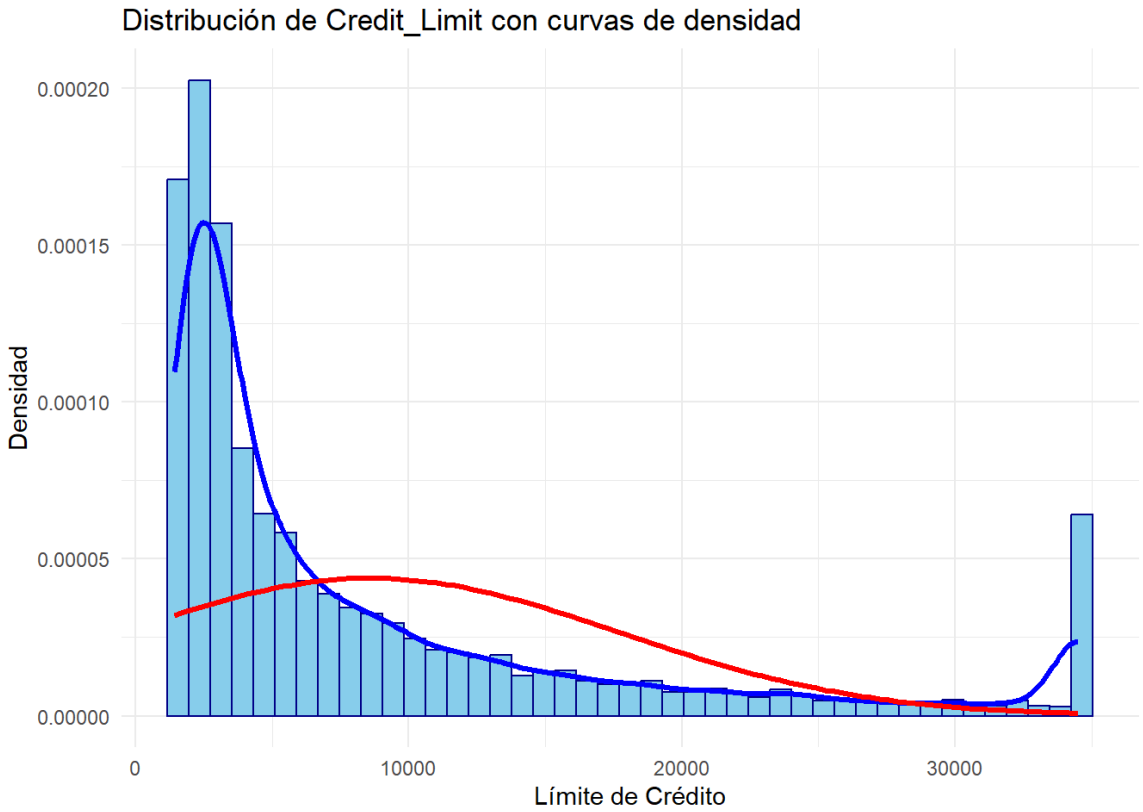



```
hist(banco$Credit_Limit, probability = TRUE, col = "lightblue", main = "Credit_Limit")
curve(dnorm(x, mean = mean(banco$Credit_Limit, na.rm=TRUE),
            sd = sd(banco$Credit_Limit, na.rm=TRUE)),
      col = "red", lwd = 2, add = TRUE)
lines(density(banco$Credit_Limit, na.rm=TRUE), col = "blue", lwd = 2)
```

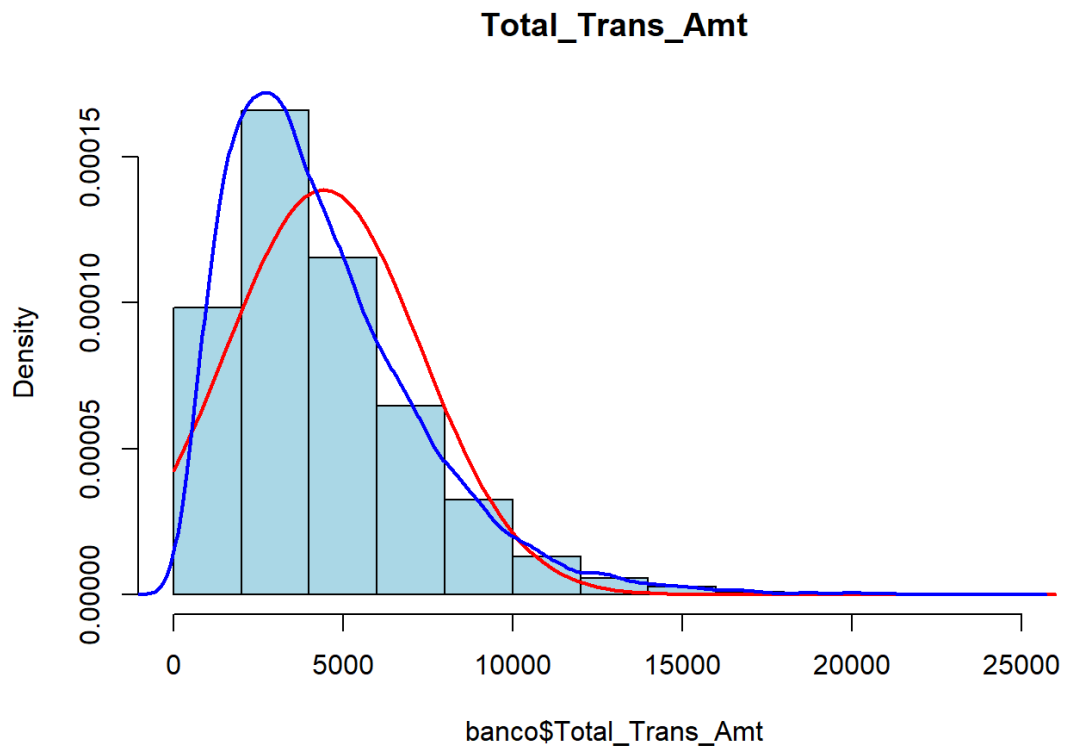


Según nuestra gráfia, los datos están sesgados por lo que usaremos Freedman

```
k_fd2 <- nclass.FD(banco$Credit_Limit)
media2 <- mean(banco$Credit_Limit, na.rm = TRUE)
sd2 <- sd(banco$Credit_Limit, na.rm = TRUE)

ggplot(banco, aes(x = Credit_Limit)) +
  geom_histogram(aes(y = ..density..), bins = k_fd2, fill = "skyblue", color = "darkblue") +
  geom_density(color = "blue", size = 1.2) +
  stat_function(fun = dnorm, args = list(mean = media2, sd = sd2), color = "red", size = 1.2) +
  labs(title = "Distribución de Credit_Limit con curvas de densidad",
       x = "Límite de Crédito", y = "Densidad") +
  theme_minimal()
```

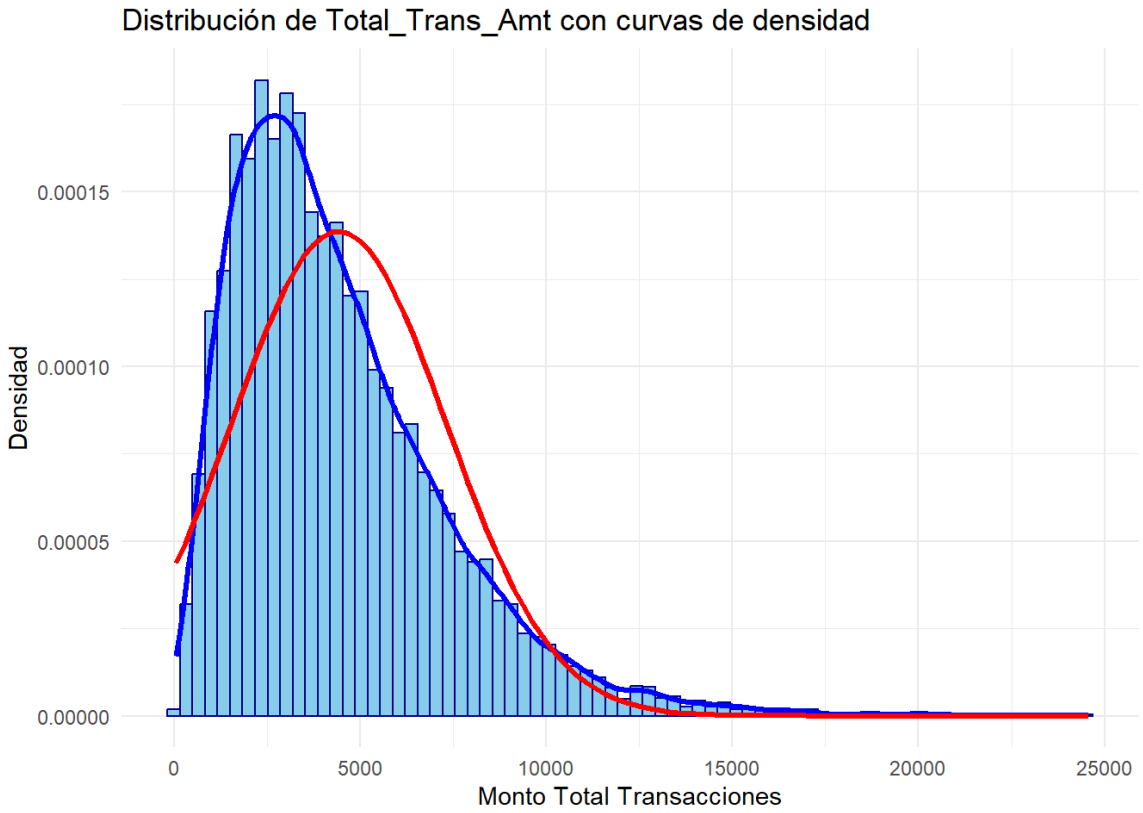



```
hist(banco$Total_Trans_Amt, probability = TRUE, col = "lightblue", main = "Total_Trans_Amt")
curve(dnorm(x, mean = mean(banco$Total_Trans_Amt, na.rm=TRUE),
            sd = sd(banco$Total_Trans_Amt, na.rm=TRUE)),
      col = "red", lwd = 2, add = TRUE)
lines(density(banco$Total_Trans_Amt, na.rm=TRUE), col = "blue", lwd = 2)
```



Según nuestra gráfica nuestros datos están sesgados por lo que usaremos freedman

```
k_fd3 <- nclass.FD(banco$Total_Trans_Amt)
media3 <- mean(banco$Total_Trans_Amt, na.rm = TRUE)
sd3 <- sd(banco$Total_Trans_Amt, na.rm = TRUE)

ggplot(banco, aes(x = Total_Trans_Amt)) +
  geom_histogram(aes(y = ..density..), bins = k_fd3, fill = "skyblue", color = "darkblue") +
  geom_density(color = "blue", size = 1.2) +
  stat_function(fun = dnorm, args = list(mean = media3, sd = sd3), color = "red", size = 1.2) +
  labs(title = "Distribución de Total_Trans_Amt con curvas de densidad",
       x = "Monto Total Transacciones", y = "Densidad") +
  theme_minimal()
```



Cálculo de asimetría con skewness

```
# Customer_Age
skewness(banco$Customer_Age, na.rm = TRUE)
```

```
## [1] -0.03193719
```

```
# Months_on_book
skewness(banco$Months_on_book, na.rm = TRUE)
```

```
## [1] -0.1048376
```

```
# Credit_Limit
skewness(banco$Credit_Limit, na.rm = TRUE)
```

```
## [1] 1.666232
```

```
# Total_Trans_Amt
skewness(banco$Total_Trans_Amt, na.rm = TRUE)
```

```
## [1] 1.343119
```

1. El coeficiente de asimetría para la **edad del cliente** es prácticamente cero, lo que indica que la distribución de la edad de los clientes es aproximadamente simétrica, muy poco sesgada a la izquierda. El banco no tiene un sesgo significativo hacia clientes más jóvenes o mayores.
2. En los **meses**, la asimetría negativa leve podría sugerir un poco de sesgo a la izquierda, probablemente hay una acumulación de clientes que han estado poco tiempo con la cuenta.
3. Para el **crédito límite** tenemos una asimetría positiva lo que indica sesgo a la derecha y que la mayoría de los clientes tienen un límite de crédito relativamente bajo.
4. La asimetría positiva para el **monto total de transacciones** sugiere que la mayoría de los clientes realiza transacciones de bajo valor, existe un sesgo a la derecha.

Calculamos curtosis

```
kurtosis_age <- kurtosis(banco$Customer_Age, type = 2)
kurtosis_months <- kurtosis(banco$Months_on_book, type = 2)
kurtosis_credit <- kurtosis(banco$Credit_Limit, type = 2)
kurtosis_trans <- kurtosis(banco$Total_Trans_Amt, type = 2)

kurtosis_age
```

```
## [1] -0.2811209
```

```
kurtosis_months
```

```
## [1] 0.409556
```

```
kurtosis_credit
```

```
## [1] 1.808989
```

```
kurtosis_trans
```

```
## [1] 2.774147
```

1. Para la **edad** la curtosis es platicúrtica (curtosis negativa), los datos están más dispersa y aplanada que una distribución normal.

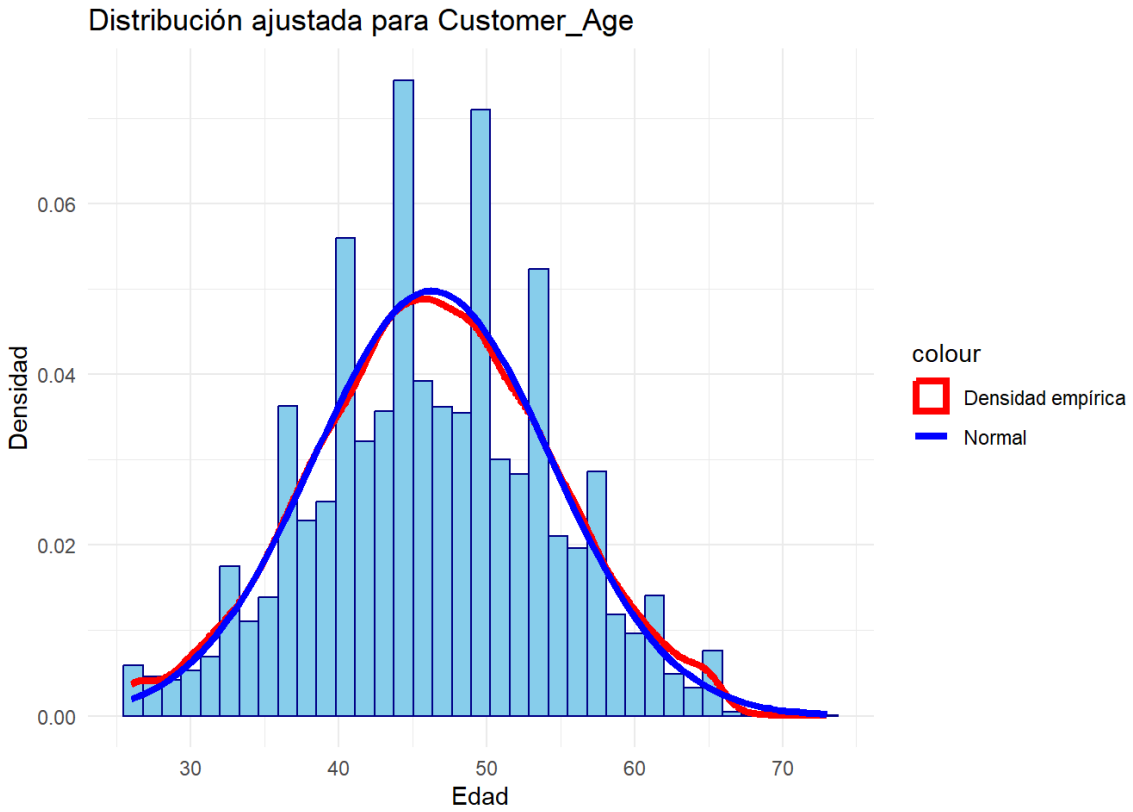
2. Para los **meses** la curtosis es leptocúrtica (ligeramente positiva), están distribuidos de forma cercana a lo normal, pero con una ligera tendencia a tener más casos en el centro y colas algo más pesadas. Quizá los clientes tienen una antigüedad parecida.
3. Para el **crédito** es leptocúrtica (curtosis positiva), tienen una distribución muy concentrada en el centro, pero también con muchos valores extremos. Problablemnte muchos clientes tienen límites similares pero hay algunos con límites muy largos.
4. Para las **transacciones** la curtosis es leptocúrtica. Es decir,está fuertemente concentrado en un valor central, pero con mucha presencia de valores extremos.

Proponemos distribución normal para la edad con base en la asimetría y curtosis obtenida anteriormente

```
k_scott <- nclass.scott(banco$Customer_Age)

ajuste_normal <- fitdist(banco$Customer_Age, "norm", method = "mle")

ajuste_normal
```

```
## Fitting of the distribution ' norm ' by maximum likelihood
## Parameters:
##       estimate Std. Error
## mean 46.322341 0.07957952
## sd    8.008326 0.05627122
```

```
ggplot(banco, aes(x = Customer_Age)) +
  geom_histogram(aes(y = after_stat(density)), bins = k_scott, fill = "skyblue", color = "darkblue") +
  geom_density(aes(color = "Densidad empírica"), lwd = 1.5) +
  stat_function(fun = dnorm,
                args = list(mean = ajuste_normal$estimate["mean"],
                            sd = ajuste_normal$estimate["sd"]),
                aes(color = "Normal"), lwd = 1.5) +
  labs(title = "Distribución ajustada para Customer_Age",
       x = "Edad", y = "Densidad") +
  scale_color_manual(values = c("Densidad empírica" = "red", "Normal" = "blue")) +
  theme_minimal()
```



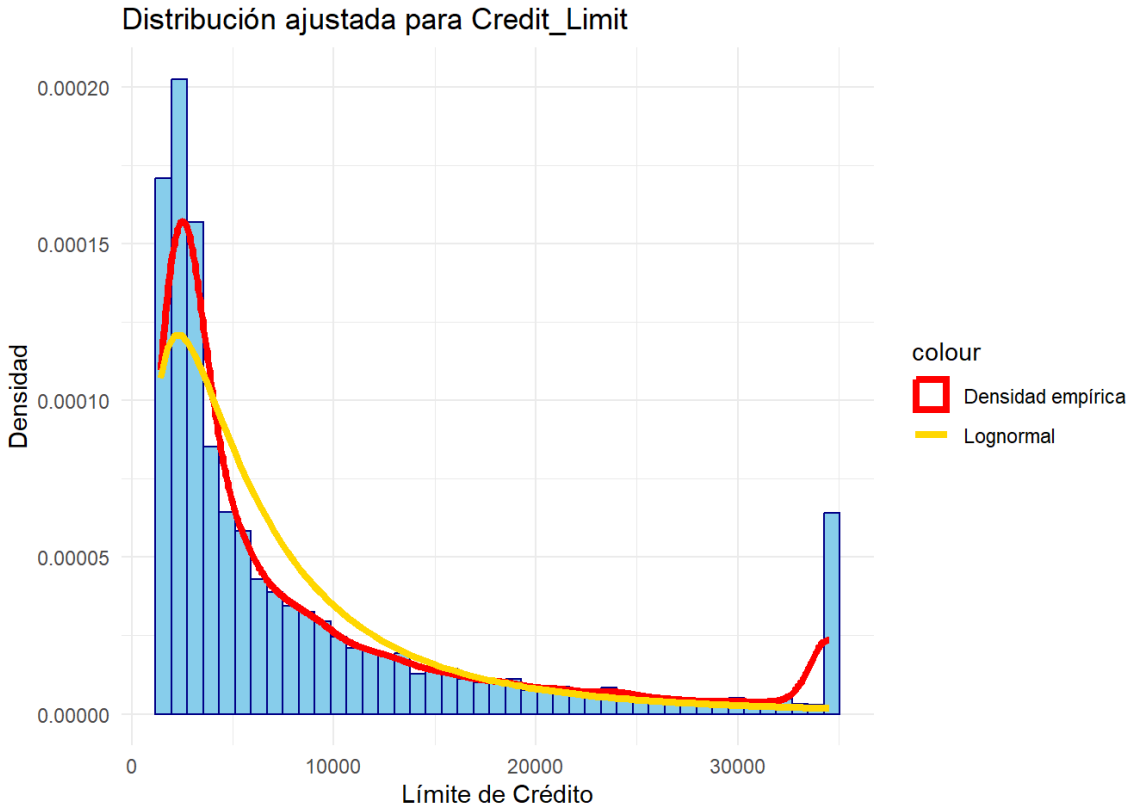
Conclusión La distribución normal podría ser la adecuada para modelar la edad de los clientes, aunque debe tenerse en cuenta que podrían existir pequeños sesgos al considerar ciertas edades más frecuentes.

Proponemos distribución log-normal para el crédito con base en la asimetría y curtosis obtenida anteriormente

```
k_freedman <- nclass.FD(banco$Credit_Limit)

ajuste_lognorm_credit <- fitdist(banco$Credit_Limit, "lnorm", method = "mle")
ajuste_lognorm_credit
```

```
## Fitting of the distribution ' lnorm ' by maximum likelihood
## Parameters:
##          estimate  Std. Error
## meanlog 8.6034121 0.009278498
## sdlog   0.9337231 0.006560855
```

```
ggplot(banco, aes(x = Credit_Limit)) +
  geom_histogram(aes(y = after_stat(density)), bins = k_freedman, fill = "skyblue", color = "darkblue") +
  geom_density(aes(color = "Densidad empírica"), lwd = 1.5) +
  stat_function(fun = dlnorm,
                args = list(meanlog = ajuste_lognorm_credit$estimate["meanlog"],
                            sdlog = ajuste_lognorm_credit$estimate["sdlog"]),
                aes(color = "Lognormal"), lwd = 1.5) +
  labs(title = "Distribución ajustada para Credit_Limit",
       x = "Límite de Crédito", y = "Densidad") +
  scale_color_manual(values = c("Densidad empírica" = "red", "Lognormal" = "gold")) +
  theme_minimal()
```



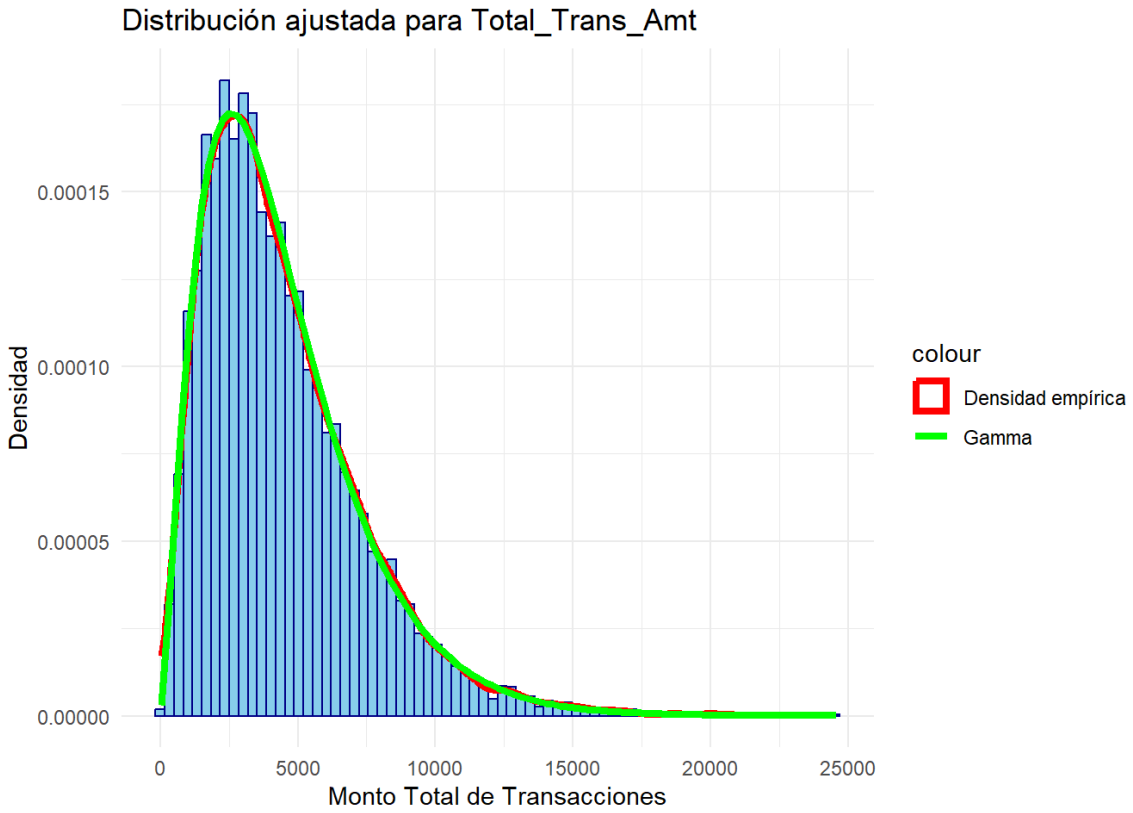
Conclusión La distribución lognormal puede modelar bien nuestro límite de crédito por la distribución asimétrica de los límites de crédito de los clientes en lo valores bajos y algunos pocos valores muy altos.

Para las transacciones proponemos Gamma con base en curtosisy asimetría

```r
k_freedman2 <- nclass.FD(banco$Total_Trans_Amt)

ajuste_gamma_trans <- fitdist(banco$Total_Trans_Amt, "gamma", method = "mle")


ajuste_gamma_trans
```

```
## Fitting of the distribution ' gamma ' by maximum likelihood
## Parameters:
##           estimate  Std. Error
## shape 2.4071190407 1.38699e-02
## rate  0.0005440034 4.31590e-08
```

```r
ggplot(banco, aes(x = Total_Trans_Amt)) +
  geom_histogram(aes(y = after_stat(density)), bins = k_freedman2, fill = "skyblue", color = "darkblue") +
  geom_density(aes(color = "Densidad empírica"), lwd = 1.5) +
  stat_function(fun = dgamma,
                args = list(shape = ajuste_gamma_trans$estimate["shape"],
                            rate = ajuste_gamma_trans$estimate["rate"]),
                aes(color = "Gamma"), lwd = 1.5) +
  labs(title = "Distribución ajustada para Total_Trans_Amt",
       x = "Monto Total de Transacciones", y = "Densidad") +
  scale_color_manual(values = c("Densidad empírica" = "red", "Gamma" = "green")) +
  theme_minimal()
```



Conclusión: La distribución gamma se ajusta bastante bien al monto total de transacciones. Por lo general la gamma des útil para representar el comportamiento de transacciones financieras donde la mayoría de los montos son relativamente pequeños.

Ejercicio extra:

```
library(LaplacesDemon)  # usamos esta libreria para la densidad dlaplace
```

```
## Warning: package 'LaplacesDemon' was built under R version 4.4.3
```

```
ajuste_laplace <- fitdist(banco$Months_on_book, "laplace", start = list(location = median(banco$Months_on_book), scale = 1))
```

```
## Warning in fitdist(banco$Months_on_book, "laplace", start = list(location =
## median(banco$Months_on_book), : The dlaplace function should return a
## zero-length vector when input has length zero
```

```
## Warning in fitdist(banco$Months_on_book, "laplace", start = list(location =
## median(banco$Months_on_book), : The plaplace function should return a
## zero-length vector when input has length zero
```

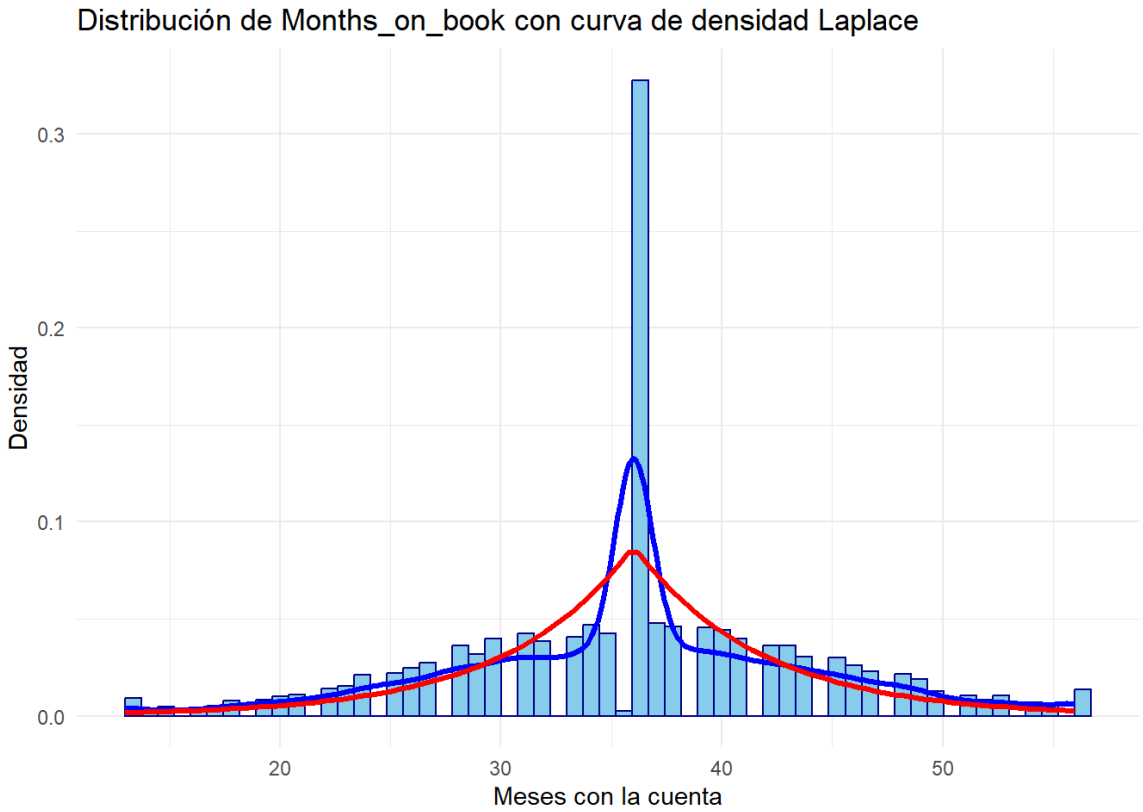
```r
# Extraer parámetros
loc <- ajuste_laplace$estimate["location"]
scale <- ajuste_laplace$estimate["scale"]

loc
```

```
## location
## 35.99999
```

```r
scale
```

```
##    scale
## 5.710328
```

```r
k_freedman <- nclass.FD(banco$Months_on_book)


ggplot(banco, aes(x = Months_on_book)) +
  geom_histogram(aes(y = after_stat(density)), bins = k_freedman, fill = "skyblue", color = "darkblue") +
  geom_density(color = "blue", size = 1.2) +
  stat_function(fun = dlaplace, args = list(location = loc, scale = scale), color = "red", size = 1.2) +  # Teórica Laplace
  labs(title = "Distribución de Months_on_book con curva de densidad Laplace",
       x = "Meses con la cuenta", y = "Densidad") +
  theme_minimal()
```



Conclusión La distribución Laplace puede ser la adecuada para modelar esta variable ya que la colas pesadas se ajusta coonsiderando la concentración en el centro como la presencia de valores extremos en las colas.

Consigna 3: Prueba de hipótesis Parte 1

El banco afirma que al menos el 50% de los clientes del género masculino son casados. Realiza una prueba de hipótesis para verificar esta afirmación con un nivel de significancia del 5%.

Realizaremos una prueba de hipótesis para proporción

Supuestos: 1 Asumimos que los datos prvienen de una muestra aleatoria 2 La variable de estado civil sera éxito si es casado y fracaso si es desconocido o soltero, (Cada observación es independiente) 3 Tamaño de la muestra lo suficientemente grande para aplicar la normal

```r
# Filtrar solo los clientes masculinos
df_m <- subset(banco, Gender == "M")

# Crear una nueva variable que sea 1 si es casado, 0 en otro caso
df_m$casado <- ifelse(df_m$Marital_Status == "Married", 1, 0)

# Verificaos el tamaño de la muestra

n <- nrow(df_m)
p0 <- 0.5

# Número de éxitos
x <- sum(df_m$casado)

# Condiciones para la aproximación normal
n * p0  # Debe ser ≥ 5
```

```
## [1] 2384.5
```

```r
n * (1 - p0)  # Debe ser ≥ 5
```

```
## [1] 2384.5
```

Ya que asumimos que los datos provienen de una Muestra Aleatoria, que son independientes y verificamos la condición para la normal (2384.5 ≥ 5), se cumplen los supuestos y podemos continuar

Establecemos la hipótesis, siendo p la proporción de hombres casados. Son mutuamente excluyentes

$$H_0 : p \geq 0.5$$
$$H_1 : p < 0.5$$

```r
# Comenzamos con la prueb obteniendo proporciones
x <- sum(df_m$Marital_Status == "Married", na.rm = TRUE)
n_prop <- length(df_m$Marital_Status)

# Proporción muestral
p_hat <- x / n_prop

# Error estándar bajo la hipótesis nula
sd <- sqrt((p0 * (1 - p0)) / n_prop)

# Estadístico de prueba Z
z_score <- (p_hat - p0) / sd

# Valor p para una prueba de una cola inferior
p_value <- pnorm(z_score, lower.tail = T)

# Nivel de significancia
alpha <- 0.05

# Mostrar los resultados
cat("Tamaño de la muestra (n):", n_prop, "\n")
```

```
## Tamaño de la muestra (n): 4769
```

```r
cat("Número de hombres casados (x):", x, "\n")
```

```
## Número de hombres casados (x): 2236
```

```r
cat("Proporción muestral (p_hat):", p_hat, "\n")
```

```
## Proporción muestral (p_hat): 0.4688614
```

```r
cat("Error estándar (sd):", sd, "\n")
```

```
## Error estándar (sd): 0.007240296
```

```r
cat("Estadístico de prueba Z:", z_score, "\n")
```

```
## Estadístico de prueba Z: -4.300736
```

```
cat("Valor p:", p_value, "\n")
```

```
## Valor p: 8.511587e-06
```

```
cat("Nivel de significancia (alpha):", alpha, "\n")
```

```
## Nivel de significancia (alpha): 0.05
```

```
# Tomar una decisión basada en el valor p
if (p_value < alpha) {
  cat("\nRechazamos la hipótesis nula.\n")
  cat("Hay evidencia estadística significativa para afirmar que la proporción de clientes masculinos casados es mayor al 50% con un nivel de significancia del 5%.\n")
} else {
  cat("\nNo rechazamos la hipótesis nula.\n")
  cat("No hay suficiente evidencia estadística para afirmar que la proporción de clientes masculinos casados es mayor al 50% con un nivel de significancia del 5%.\n")
}
```

```
##
## Rechazamos la hipótesis nula.
## Hay evidencia estadística significativa para afirmar que la proporción de clientes masculinos casados es mayor al 50% con un nivel de significancia del 5%.
```

```
# Opción alternativa para verificar esultados de la prueba anterior
prop.test(x = x, n = n_prop, p = 0.5, alternative = "less", correct = FALSE)
```

```
##
##  1-sample proportions test without continuity correction
##
## data:  x out of n_prop, null probability 0.5
## X-squared = 18.496, df = 1, p-value = 8.512e-06
## alternative hypothesis: true p is less than 0.5
## 95 percent confidence interval:
##  0.0000000 0.4807618
## sample estimates:
##         p
## 0.4688614
```

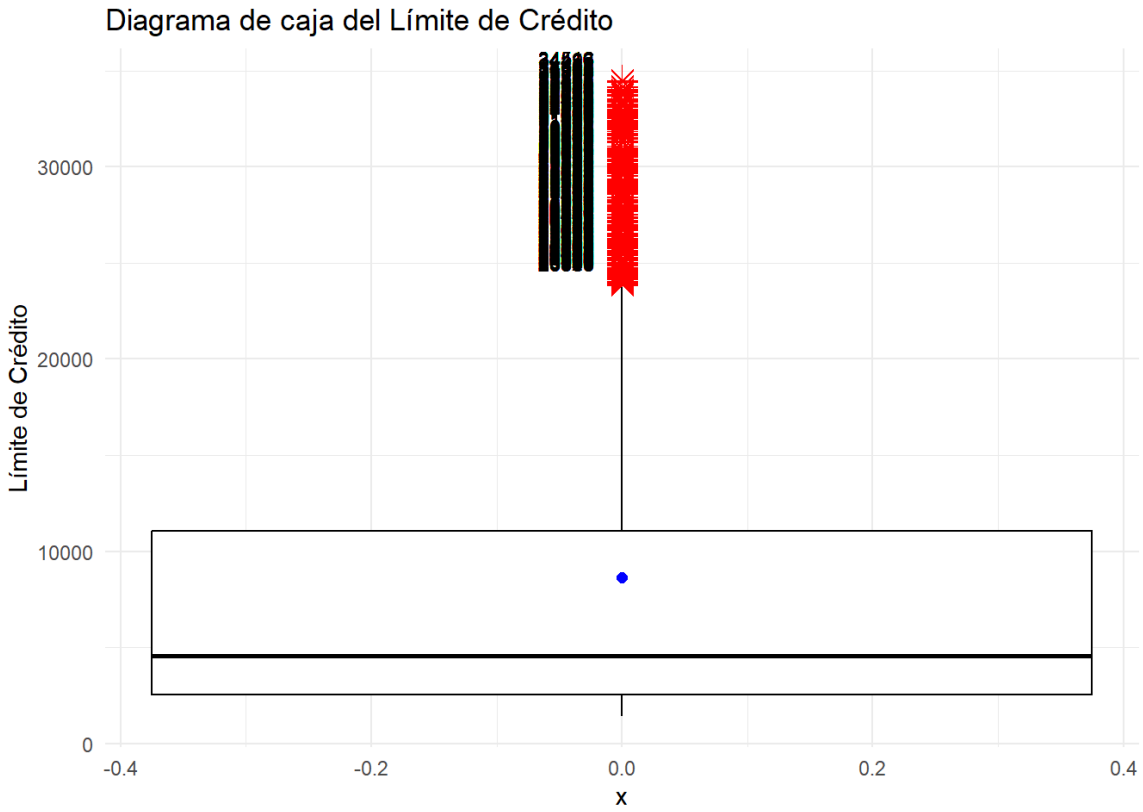
Conclusión para el banco después de comprara ambas pruebas La proporción muestral de hombres casados es de aproximadamente 0.4689 (46.89%), y el intervalo de confianza del 95% para la proporción verdadera se encuentra entre 0 y 0.4808. Dado el valor p, tenemos evidencia estadística significativa para concluir que la proporción de clientes del género masculino que están casados es menor al 50%, con un nivel de significancia del 5%. La afirmación del banco de que al menos el 50% de los clientes masculinos son casados no está respaldada por estos datos.

Consigna 4: Prueba de hipótesis Parte 2

Se sospecha que el límite de crédito promedio de los clientes es mayor a $8,500. Realiza una prueba de hipótesis para verificar esta afirmación con un nivel de significancia del 5%.

Realizamos un ggplot

```
ggplot(banco, aes(y = Credit_Limit)) +
  geom_boxplot(
    color = "black",
    outlier.color = "red",
    outlier.shape = 8,
    outlier.size = 4
  ) +
  labs(
    title = "Diagrama de caja del Límite de Crédito",
    y = "Límite de Crédito"
  ) +
  stat_summary(
    fun = mean,
    geom = "point",
    shape = 20,
    size = 3,
    color = "blue",
    aes(x = 0)
  ) +
  geom_text(
    data = banco %>%
      filter(Credit_Limit > quantile(Credit_Limit, 0.75) + 1.5 * IQR(Credit_Limit) |
             Credit_Limit < quantile(Credit_Limit, 0.25) - 1.5 * IQR(Credit_Limit)),
    aes(x = 0, label = round(Credit_Limit, 0)),
    hjust = 1.5, vjust = -0.9,
    color = "black", size = 3
  ) +
  theme_minimal()
```



CONCLUSIONES

1.La mediana está cerca de la parte baja del gráfico, así que la mayoría de los clientes no tiene un límite de crédito tan alto.

2.Se ven muchos outliers arriba del gráfico, lo que nos muestra que hay personas con límites de crédito muy altos comparados con los demás.

3.El promedio está más alto que la mediana eso significa que hay varios con créditos tan altos que suben el promedio, aunque no sean la mayoría.

¿Cuántos valores atípicos (outliers) hay en la variable ‘límite de crédito’?

```
#Calculamos cuantiles e IQR
Q1 <- quantile(banco$Credit_Limit, 0.25)
Q3 <- quantile(banco$Credit_Limit, 0.75)
IQR <- Q3 - Q1


limite_superior <- Q3 + 1.5 * IQR
limite_inferior <- Q1 - 1.5 * IQR


outliers <- banco %>%
  filter(Credit_Limit > limite_superior | Credit_Limit < limite_inferior)

count(outliers)
```

```
## # A tibble: 1 × 1
##       n
##   <int>
## 1   984
```

¿Cuáles son los valores de los 10 primeros valores atípicos?

```
head(outliers$Credit_Limit, 10)
```

```
##  [1] 34516 29081 30367 32426 34516 34516 23957 34516 25300 34516
```

¿A qué identificadores de cliente (CLIENTNUM) corresponden los 10 primeros valores atípicos?

```
head(outliers$CLIENTNUM, 10)
```

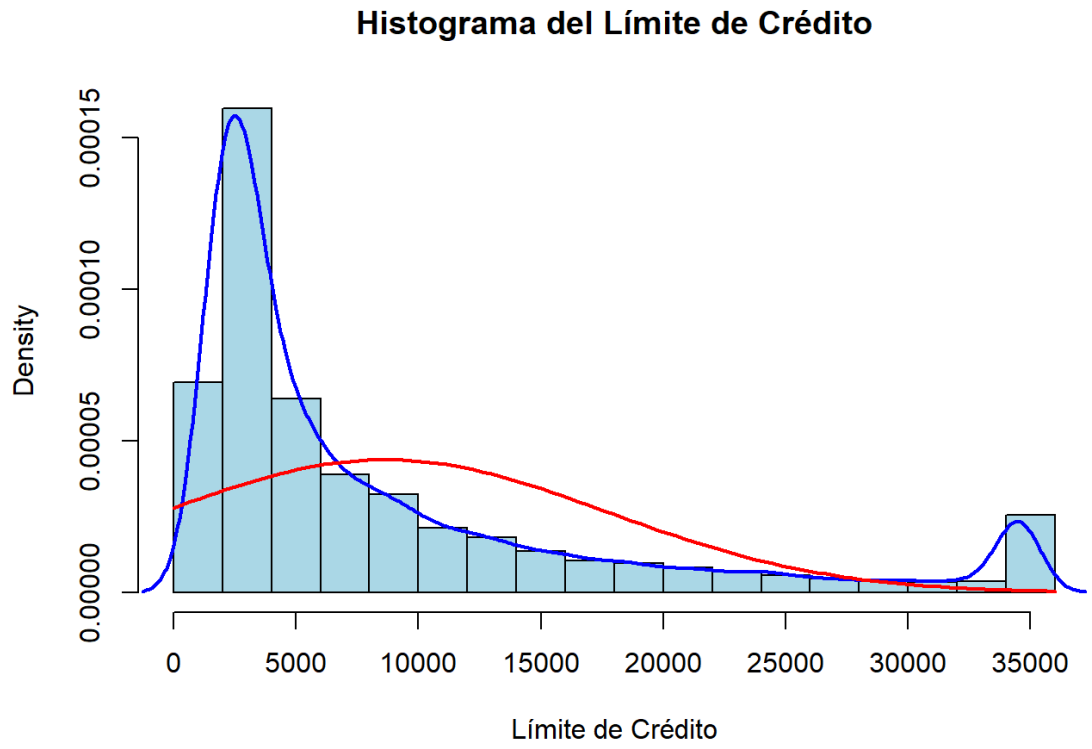
```
##  [1] 810347208 818906208 709967358 827111283 712661433 712030833 710082708
##  [8] 788979258 717975333 715971108
```

Verificamos supuestos:

1.- El texto menciona que los datos fueron obtenidos por una muestra aleatoria 2.- El valor de sigma poblacional no es conocido, ocuparemos t.test 3.- Realizamos pruebas de normalidad

Hacemos un histograma sencillo

```
hist(banco$Credit_Limit,
     probability = TRUE,
     col = "lightblue",
     main = "Histograma del Límite de Crédito",
     xlab = "Límite de Crédito")
lines(density(banco$Credit_Limit, na.rm = TRUE),
      col = "blue",
      lwd = 2)
curve(dnorm(x,
            mean = mean(banco$Credit_Limit, na.rm = TRUE),
            sd = sd(banco$Credit_Limit, na.rm = TRUE)),
      col = "red",
      lwd = 2,
      add = TRUE)
```



Observamos colas pesados por lo que ocuparemos la prueba Anderson-Darling

```
ad.test(banco$Credit_Limit)
```

```
##
##  Anderson-Darling normality test
##
## data:  banco$Credit_Limit
## A = 943.36, p-value < 2.2e-16
```

Como el p-value = 0.6417 < 0.05 se rechaza la hipótesis nula de que los datos se distribuyan normalmente.

Establecemos prueba de hipótesis donde mu es la media del valor de los limites crediticios

$$H_0 : \mu \leq 8500$$
$$H_1 : \mu > 8500$$

Aplicamos un t.test

```
t.test(x = banco$Credit_Limit,
       alternative = "greater",
       mu = 8500,
       conf.level = 0.95)
```

```
##
##  One Sample t-test
##
## data:  banco$Credit_Limit
## t = 1.461, df = 10126, p-value = 0.07202
## alternative hypothesis: true mean is greater than 8500
## 95 percent confidence interval:
##  8483.383      Inf
## sample estimates:
## mean of x
##  8631.954
```

1. Dado que el valor p = 0.07202 > 0.05, no se rechaza la hipótesis nula. Esto significa que no hay evidencia suficiente para afirmar que el límite de crédito promedio es mayor a $8,500.

```
v_critico <- qt(p = 1 - 0.05, df = 10126)
v_critico
```

```
## [1] 1.645004
```

2.Dado que 1.461 no supera a 1.645, entonces se rechaza la hipótesis nula. No se cuenta con evidencia estadísticamente suficiente para afirmar que el promedio del límite de crédito de los clientes sea superior a $8,500.

3. El intervalo de confianza del 95% para la media del límite de crédito comienza en 8483.38, lo cual incluye valores menores a $8,500. Como el valor hipotético de 8,500 no queda fuera del intervalo, no se puede afirmar que la media verdadera sea mayor a $8,500.

Conclusión Con base en la prueba de hipótesis realizada, no se encontró evidencia significativa para rechazar la hipótesis nula, por lo tanto no podemos afirmar que el límite de crédito promedio de los clientes sea mayor a $8,500.