DATA SCIENTIST CHALLENGE

1. Librerias a utilizar

```
library(data.table)
library(ggplot2)
library(rstudioapi)
library(ggpubr)
library(dplyr)
library(party)
library(reshape2)
library(MLmetrics)
library(caret)
library(cluster)
require(caTools)
library(ROCR)
```

2.Carga del dataset

```
#Se lee la data

path <- "E:/Data_Science/Nala_test/nala.csv"

df_datos <- read.csv(path)

#Se transforma la variable de ID_USER por el nombre correcto

df_datos$ID_USER <- df_datos$ï..ID_USER

df_datos$ï..ID_USER <- NULL
```

3. Missings en las variables

```
colSums(is.na(df_datos))
```

```
##          genero           monto           fecha            hora     dispositivo
##               0               0               0               0               0
## establecimiento          ciudad         tipo_tc       linea_tc      interes_tc
##            3410            3910               0               0               0
##      status_txn        is_prime            dcto       cashback          fraude
##               0               0               0               0               0
##         ID_USER
##               0
```

4. Análisis Univariado Cuantitativo

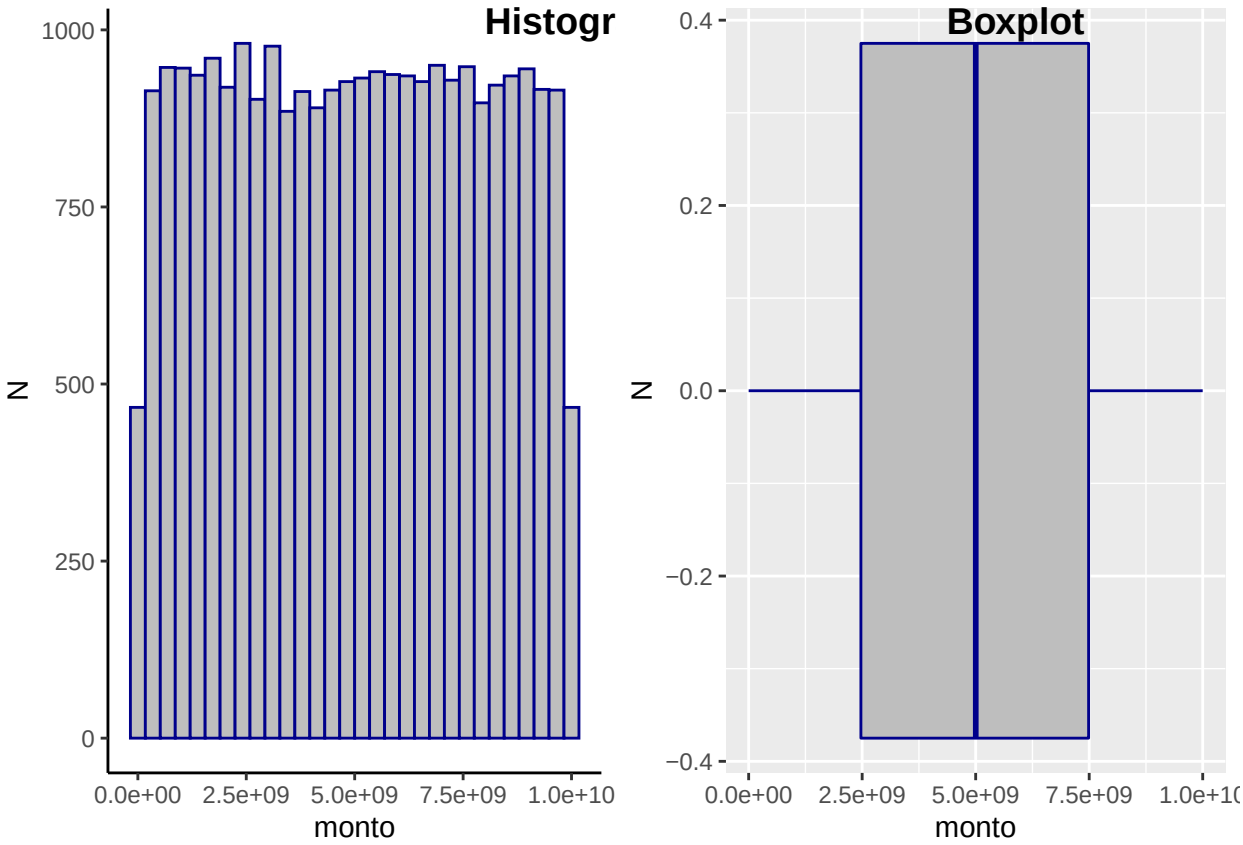
```
# Se define una función para analizar las variables cuantitativas
f_Ana_Uni_Cuan <- function(df,variable){
```

```
  g1 <- ggplot(data = df)+
    geom_histogram(mapping = aes(x = get(variable)),color='darkblue',fill="gray")+
    xlab(variable)+
    ylab("N")+
    theme_classic()

  g2 <- ggplot(data = df)+
    geom_boxplot(mapping = aes(x = get(variable)),color='darkblue',fill="gray")+
    xlab(variable)+
    ylab("N")

  plots <- ggarrange(plotlist = list(g1,g2), labels = c("Histograma","Boxplot"),
                     ncol = 2, hjust = -2.4)

  dt_Est_Descr <- df %>%
    select(variable)
  dt_Est_Descr <- t(as.matrix(summary(dt_Est_Descr)))

  return(list(graficos = plots,
              Est_Desc = copy(dt_Est_Descr)))
}
```

Variable Monto

```
f_Ana_Uni_Cuan(df_datos,"monto")
```

```
## $graficos
```



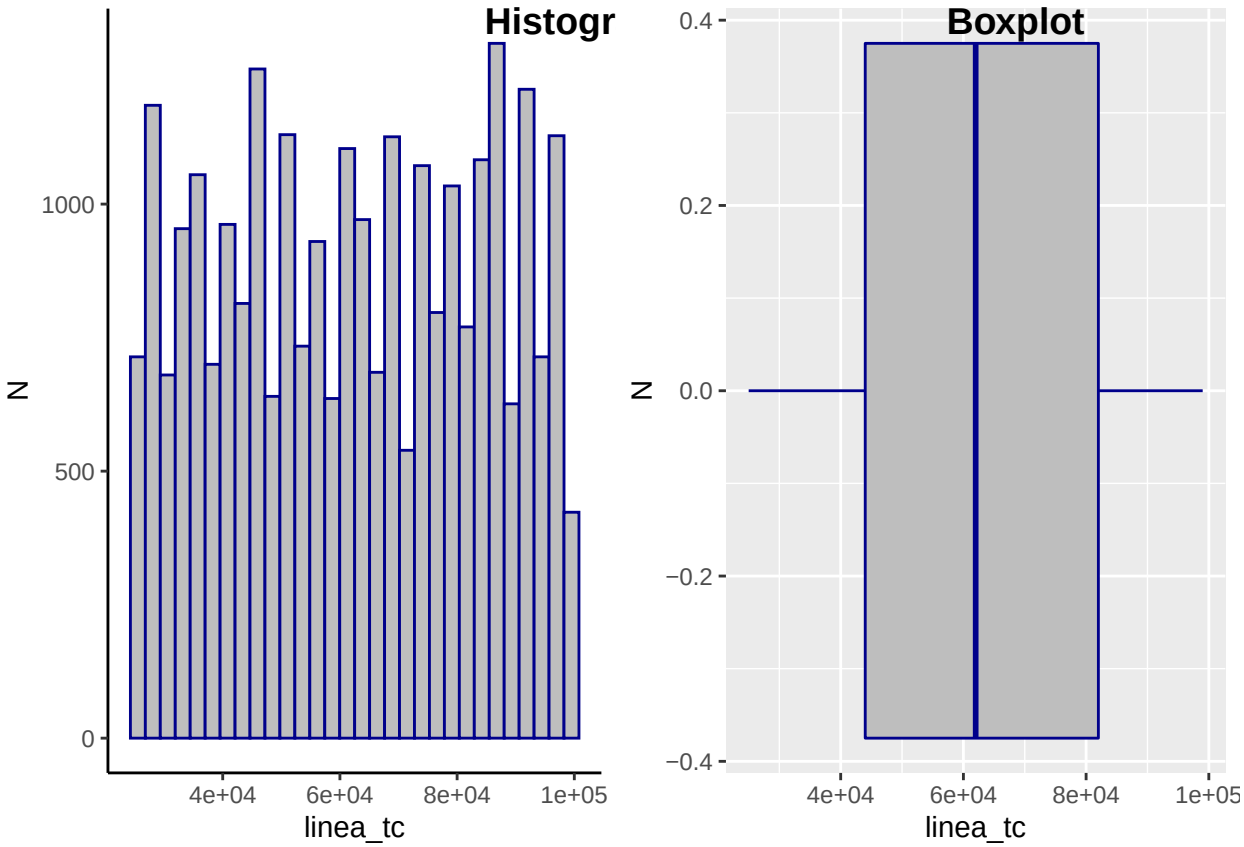

```
##
## $Est_Desc
##
##     monto Min.   :1.384e+05   1st Qu.:2.472e+09   Median :5.002e+09
##
##     monto Mean   :4.989e+09   3rd Qu.:7.487e+09   Max.   :9.999e+09
```

SE VE UNA DISTRIBUCION UNIFORME, SIN PRESENCIA DE OUTLIERS

Variable Linea TC

```
f_Ana_Uni_Cuan(df_datos,"linea_tc")
```

```
## $graficos
```



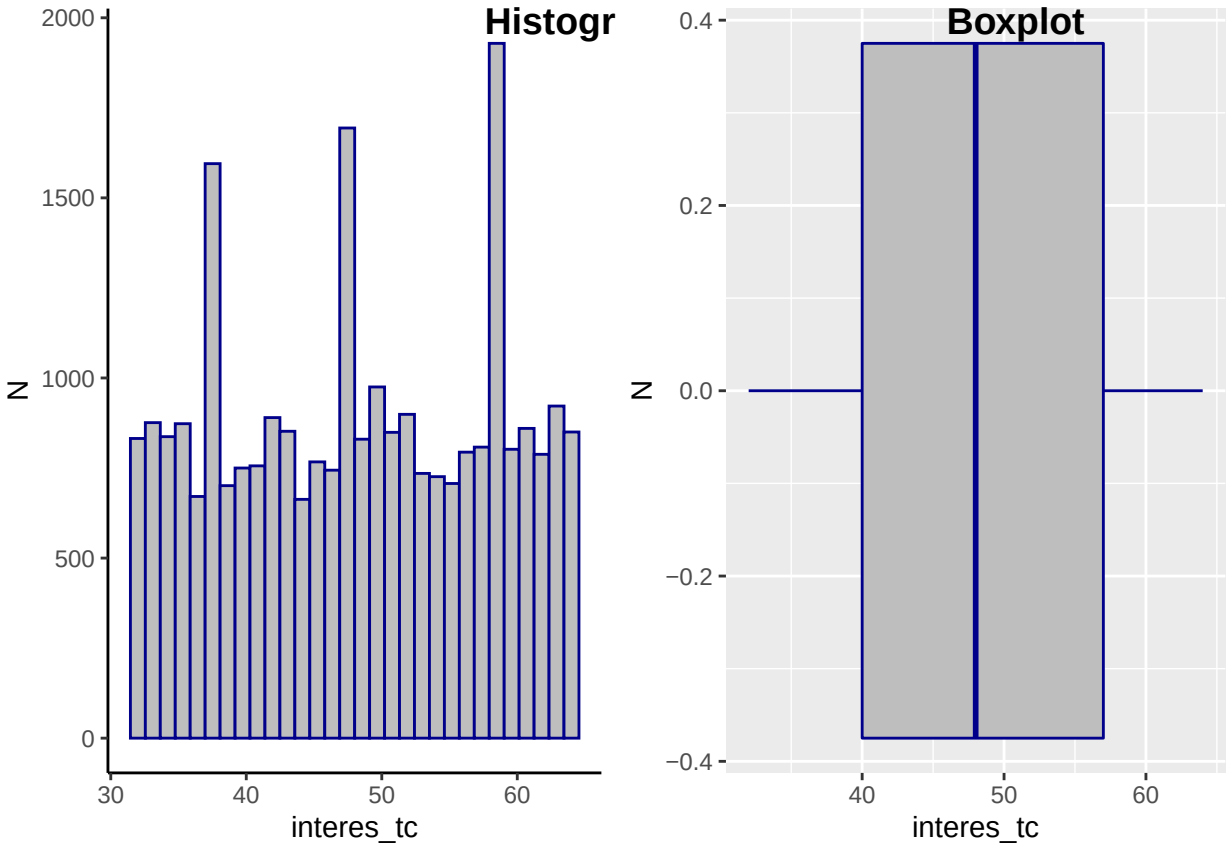

```
##
## $Est_Desc
##
##    linea_tc Min.   :25000   1st Qu.:44000   Median :62000   Mean   :62477
##
##    linea_tc 3rd Qu.:82000   Max.   :99000
```

SE TIENE UNA DISTRIBUCION SIMETRICA MULTIMODAL SIN PRESENCIA DE OUTLIERS

Variable Interes TC

```
f_Ana_Uni_Cuan(df_datos,"interes_tc")
```

```
## $graficos
```



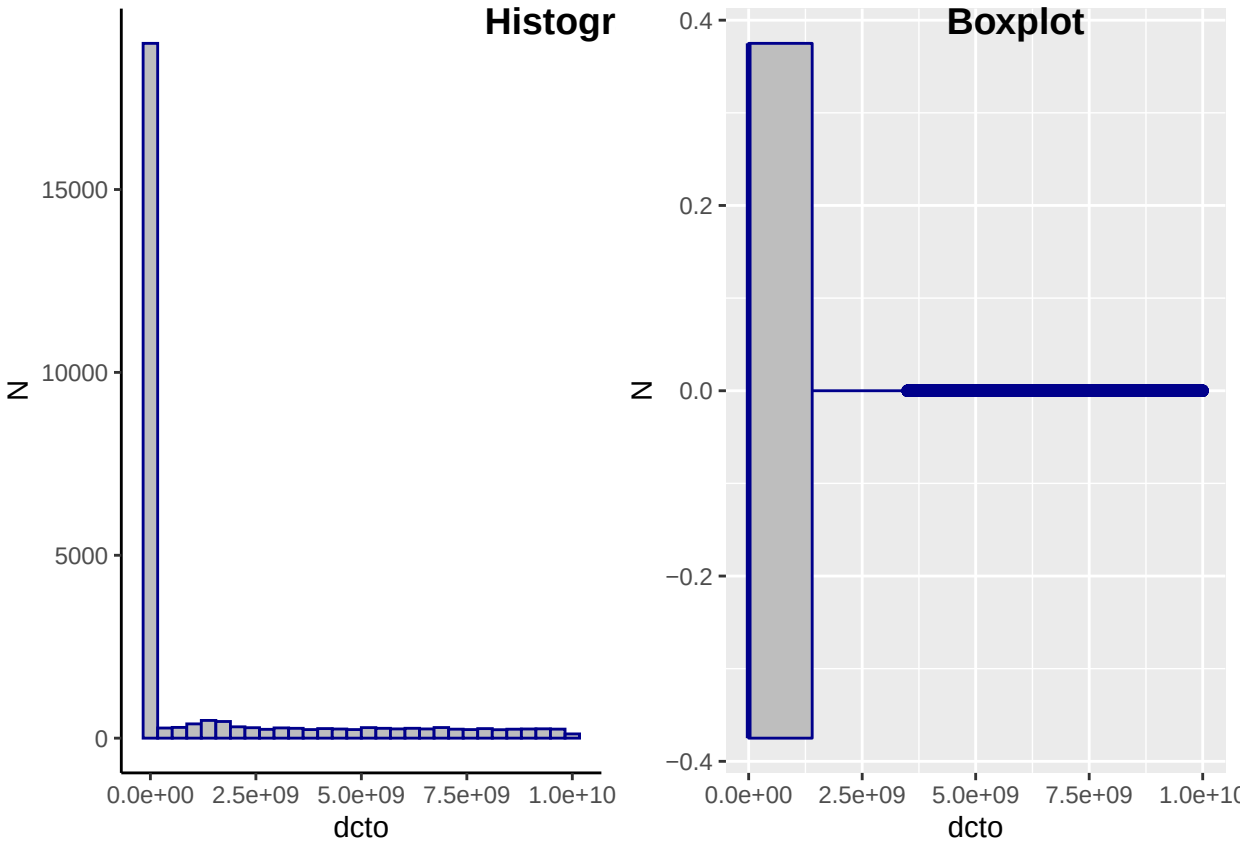

```
##
## $Est_Desc
##
##   interes_tc Min.   :32.00   1st Qu.:40.00   Median :48.00   Mean   :48.22
##
##   interes_tc 3rd Qu.:57.00   Max.   :64.00
```

SE TIENE UNA DISTRIBUCION SIMETRICA MULTIMODAL SIN PRESENCIA DE OUTLIERS

Variable Descuento(dcto)

```
f_Ana_Uni_Cuan(df_datos,"dcto")
```

```
## $graficos
```



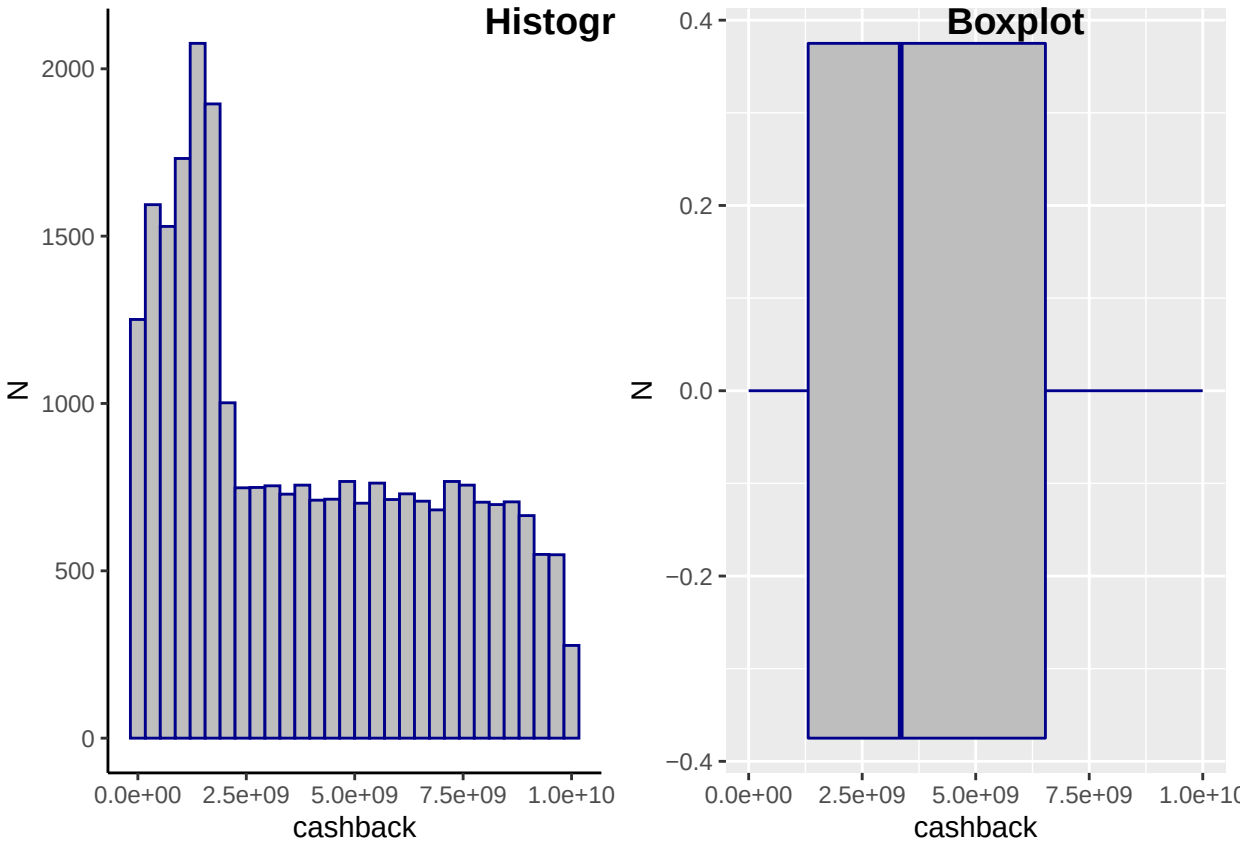

```
##
## $Est_Desc
##
##      dcto Min.   :0.000e+00   1st Qu.:0.000e+00   Median :0.000e+00
##
##      dcto Mean   :1.400e+09   3rd Qu.:1.399e+09   Max.   :9.999e+09
```

SE TIENE UNA DISTRIBUCION ASIMETRICA POSITIVA CON ALTA PRESENCIA DE OUTLIERS

Variable Cashback

```
f_Ana_Uni_Cuan(df_datos,"cashback")
```

```
## $graficos
```



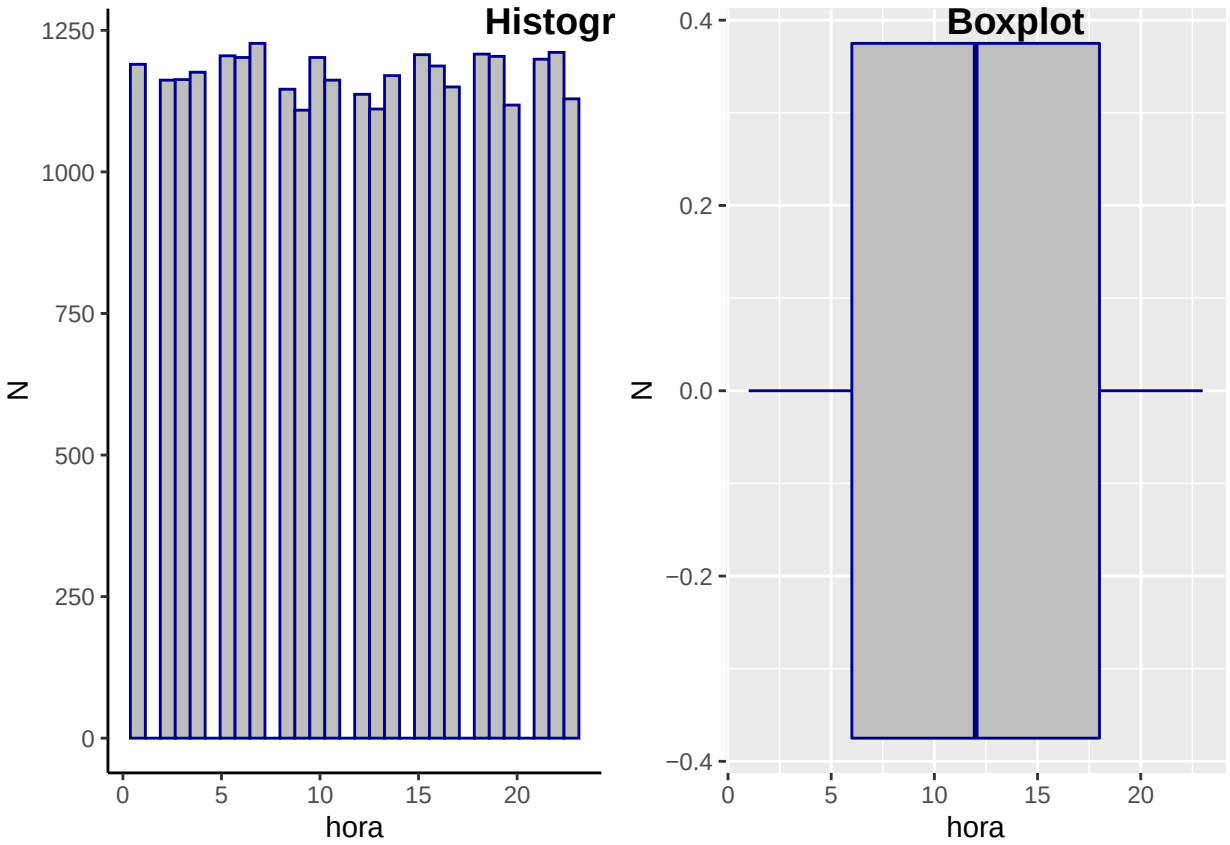

```
##
## $Est_Desc
##
##    cashback Min.   :1.384e+04   1st Qu.:1.311e+09   Median :3.348e+09
##
##    cashback Mean   :3.960e+09   3rd Qu.:6.535e+09   Max.   :9.998e+09
```

SE TIENE UNA DISTRIBUCION ASIMETRICA POSITIVA SIN PRESENCIA DE OUTLIERS

Variable Hora

```
f_Ana_Uni_Cuan(df_datos,"hora")
```

```
## $graficos
```




```
##
## $Est_Desc
##
##      hora Min.   : 1.00   1st Qu.: 6.00   Median :12.00   Mean   :11.99
##
##      hora 3rd Qu.:18.00   Max.   :23.00
```

SE TIENE UNA DISTRIBUCION SIMETRICA UNIFORME SIN PRESENCIA DE OUTLIERS

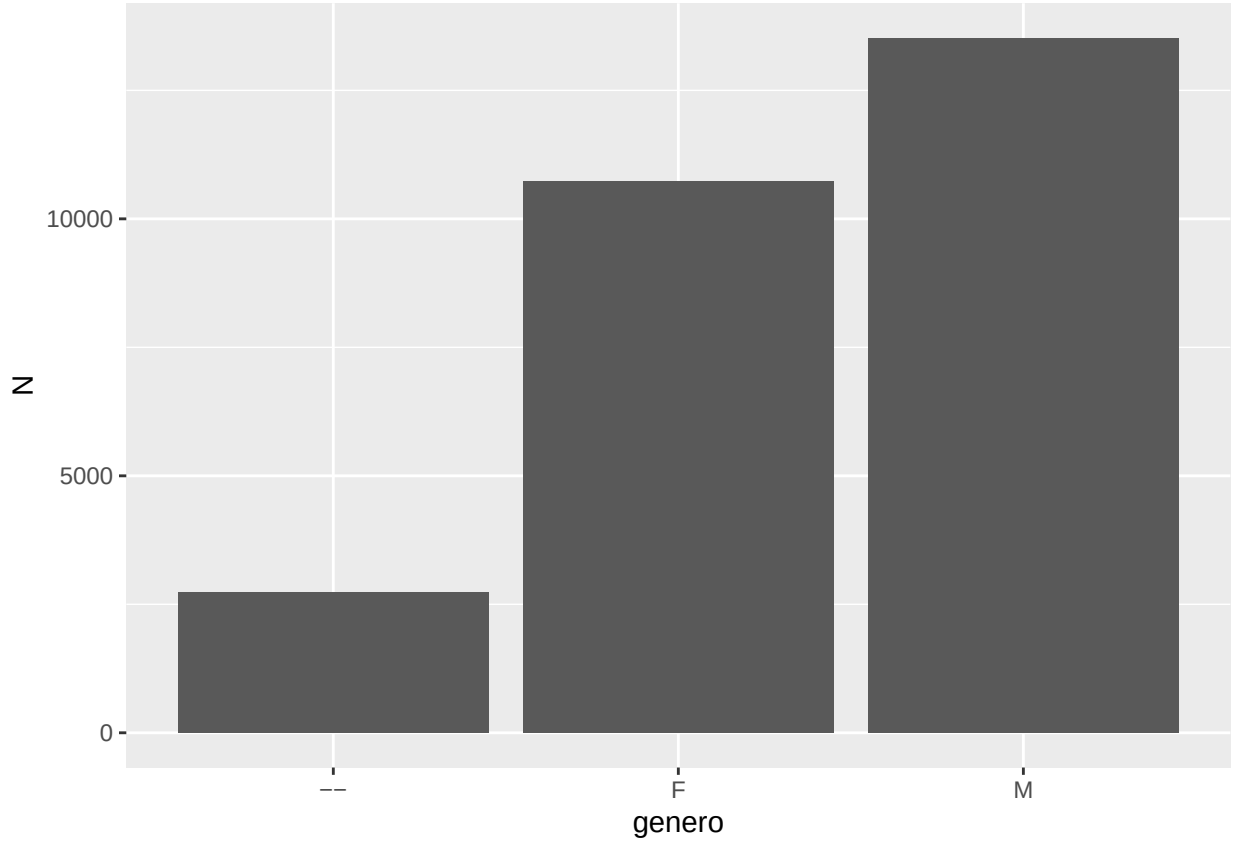
5. Análisis Univariado Cualitativo

```
# Se define una función para analizar las variables cualitativas
f_Ana_Uni_Cual <- function(df,variable){

  summary_var <- df%>%
    group_by(get(variable))%>%
    summarise(porc = round(100*n()/nrow(df),2))

  g <- ggplot(df)
  g <- g + geom_bar(aes(x = as.factor(get(variable))), stat = "count")
  g <- g + xlab(variable)
  g <- g + ylab("N")

  return(list(summary_var,g))
}
```

Variable Genero

```
f_Ana_Uni_Cual(df_datos,"genero")
```

```
## [[1]]
## # A tibble: 3 x 2
##   `get(variable)`  porc
##   <chr>           <dbl>
## 1 --               10.1
## 2 F                39.8
## 3 M                50.1
##
## [[2]]
```



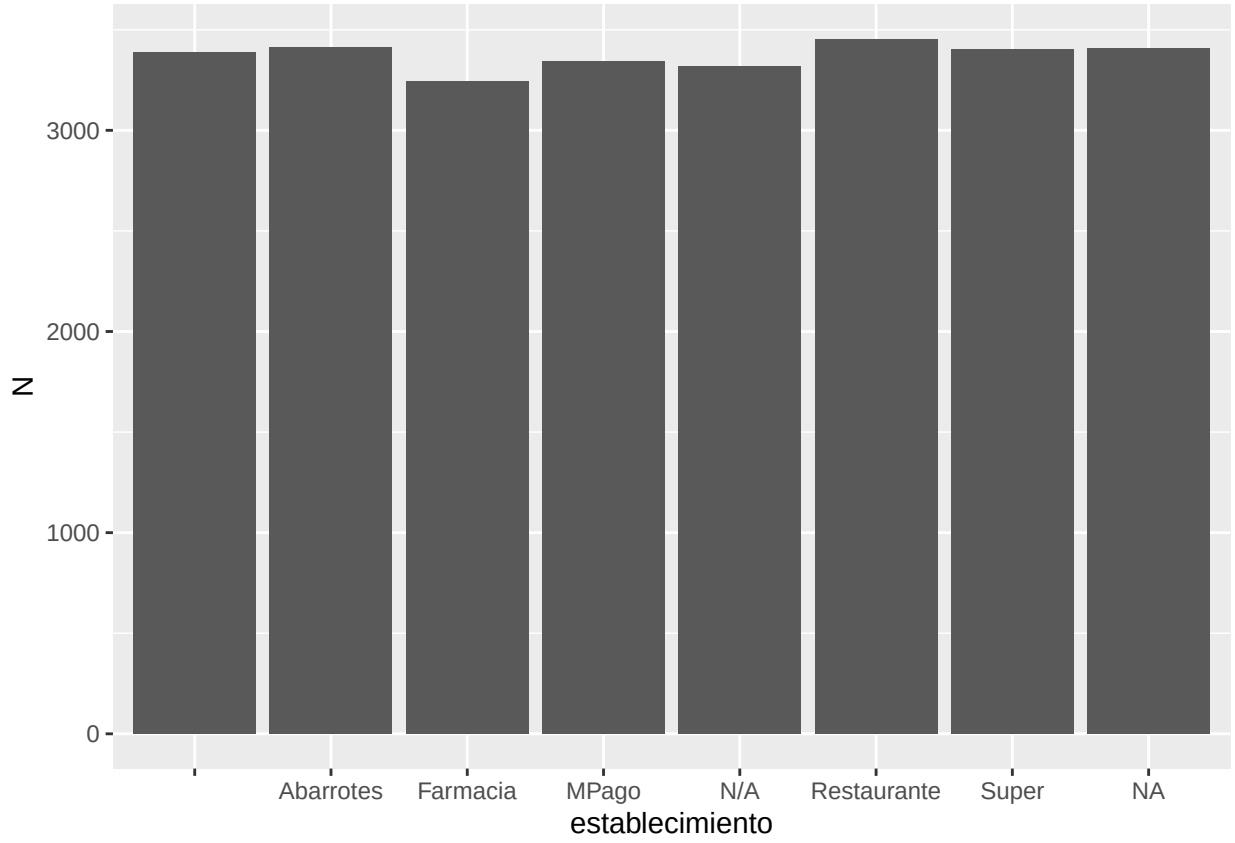
EL 50% DE LAS TRANSACCIONES SON REALIZADAS POR MUJERES

Variable Establecimiento

```
f_Ana_Uni_Cual(df_datos,"establecimiento")
```

```
## [[1]]
## # A tibble: 8 x 2
##   `get(variable)`  porc
##   <chr>           <dbl>
## 1 ""               12.6
## 2 "Abarrotes"      12.7
## 3 "Farmacia"       12.0
```

```
## 4 "MPago"          12.4
## 5 "N/A"            12.3
## 6 "Restaurante"    12.8
## 7 "Super"          12.6
## 8  <NA>            12.6
##
## [[2]]
```



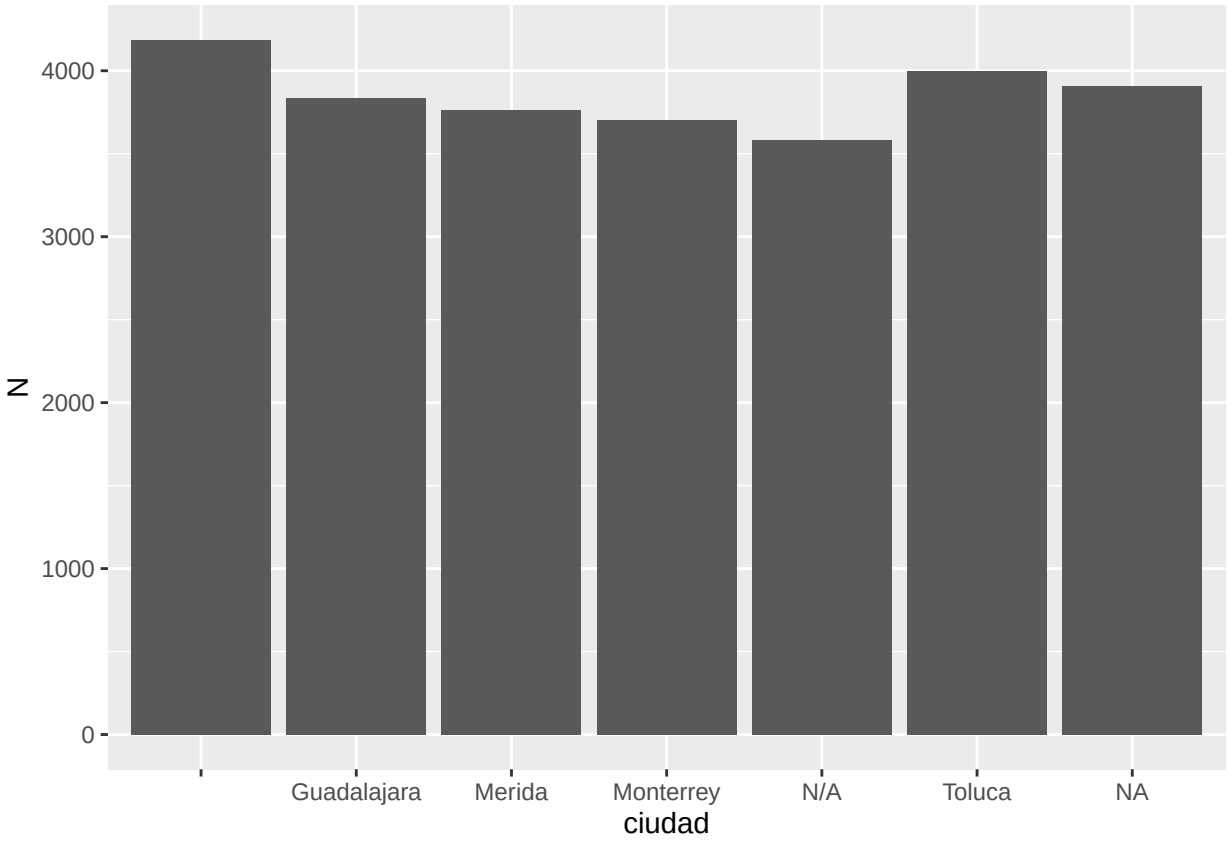
SE TIENE UN 37% QUE NO SE DEFINE EL ESTABLECIMIENTO, EL RESTO SE DISTRIBUYE DE MANERA UNIFORME

Variable Ciudad

```
f_Ana_Uni_Cual(df_datos,"ciudad")
```

```
## [[1]]
## # A tibble: 7 x 2
##   `get(variable)`  porc
##   <chr>           <dbl>
## 1 ""               15.5
## 2 "Guadalajara"    14.2
## 3 "Merida"         13.9
## 4 "Monterrey"      13.7
## 5 "N/A"            13.3
## 6 "Toluca"         14.8
## 7  <NA>            14.5
##
```

```
## [[2]]
```

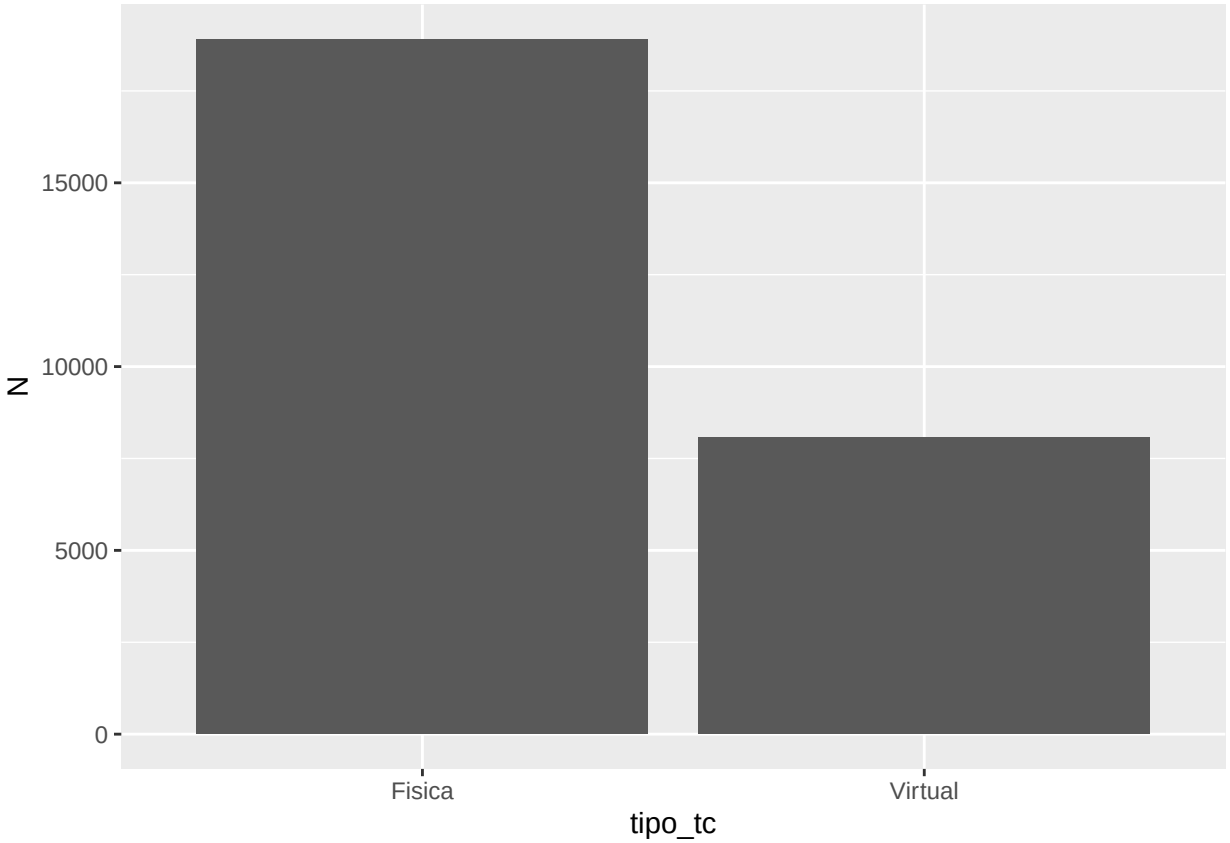


SE TIENE UN 37% QUE NO SE DEFINE CIUDAD, EL RESTO SE DISTRIBUYE DE MANERA UNIFORME

Variable Tipo TC

```
f_Ana_Uni_Cual(df_datos,"tipo_tc")
```

```
## [[1]]
## # A tibble: 2 x 2
##   `get(variable)`  porc
##   <chr>           <dbl>
## 1 Fisica           70.1
## 2 Virtual          29.9
##
## [[2]]
```

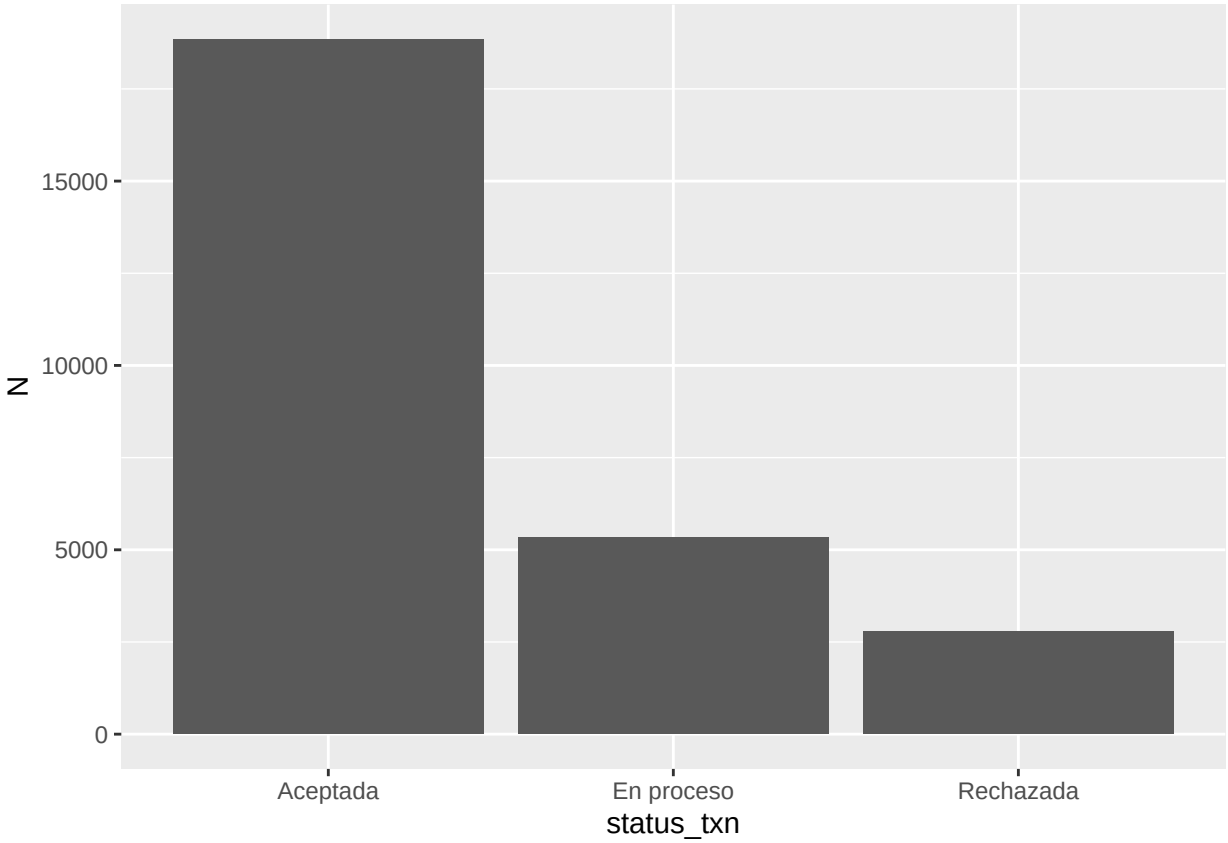


EL 70% SON TARJETAS FISICAS

Variable Status TxN

```
f_Ana_Uni_Cual(df_datos,"status_txn")
```

```
## [[1]]
## # A tibble: 3 x 2
##   `get(variable)`  porc
##   <chr>           <dbl>
## 1 Aceptada         69.9
## 2 En proceso       19.8
## 3 Rechazada        10.3
##
## [[2]]
```

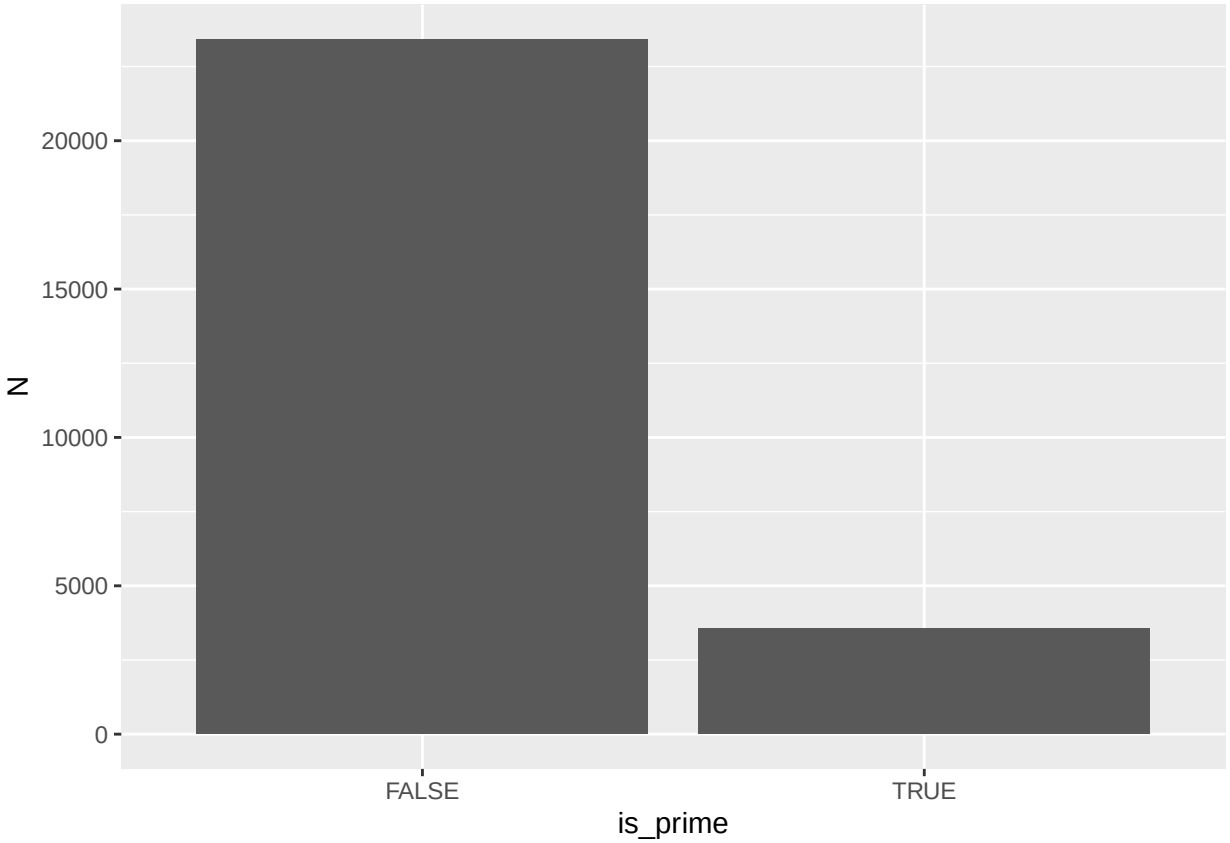


EL 10% DE LAS TRANSACCIONES REALIZADAS HAN SIDO RECHAZADAS

Variable Primera vez de uso(is_prime)

```
f_Ana_Uni_Cual(df_datos,"is_prime")
```

```
## [[1]]
## # A tibble: 2 x 2
##   `get(variable)`  porc
##   <lgl>           <dbl>
## 1 FALSE            86.8
## 2 TRUE             13.2
##
## [[2]]
```

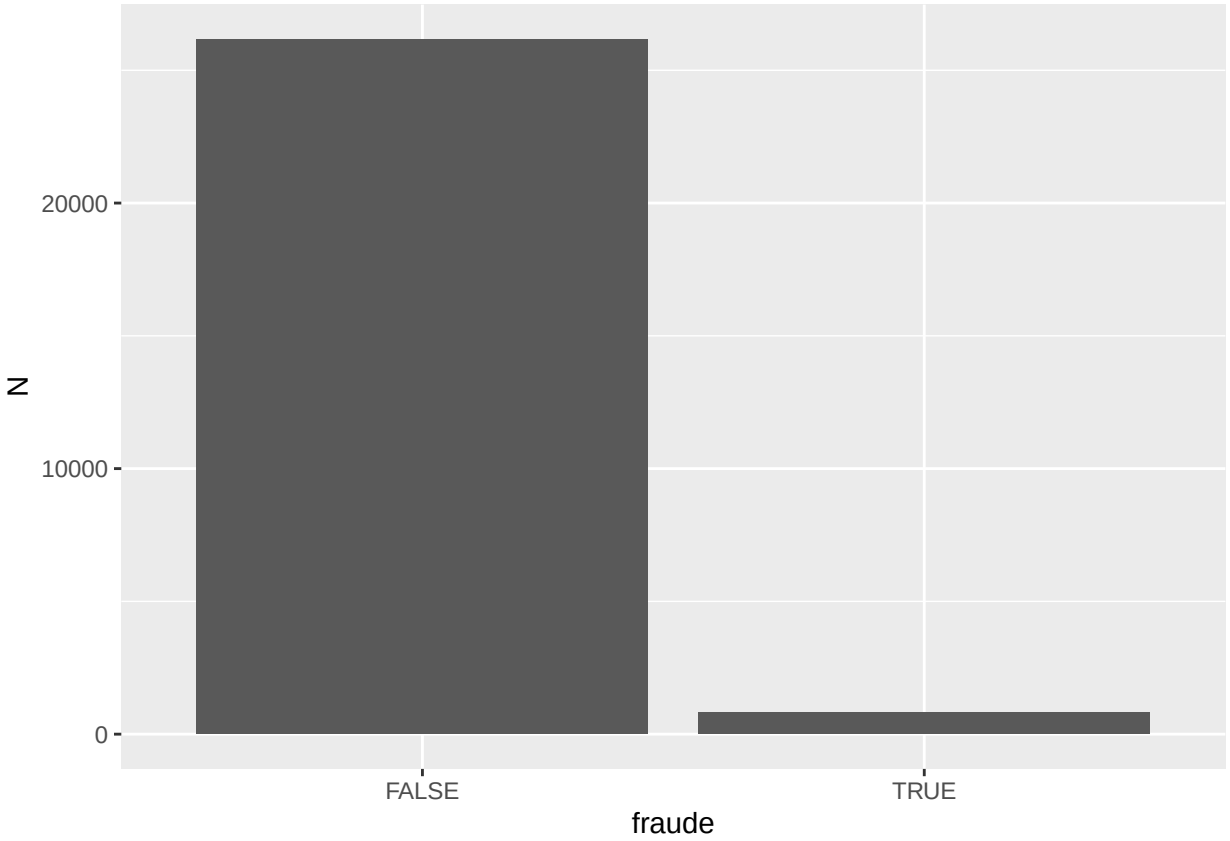


EL 13% DE LAS OPERACIONES FUERON EL PRIMER USO DE LA TARJETA

Variable Fraude

```
f_Ana_Uni_Cual(df_datos,"fraude")
```

```
## [[1]]
## # A tibble: 2 x 2
##   `get(variable)`  porc
##   <lgl>           <dbl>
## 1 FALSE              97
## 2 TRUE                3
##
## [[2]]
```



EL 3% DE LAS OPERACIONES FUERON VICTIMAS DE FRAUDES

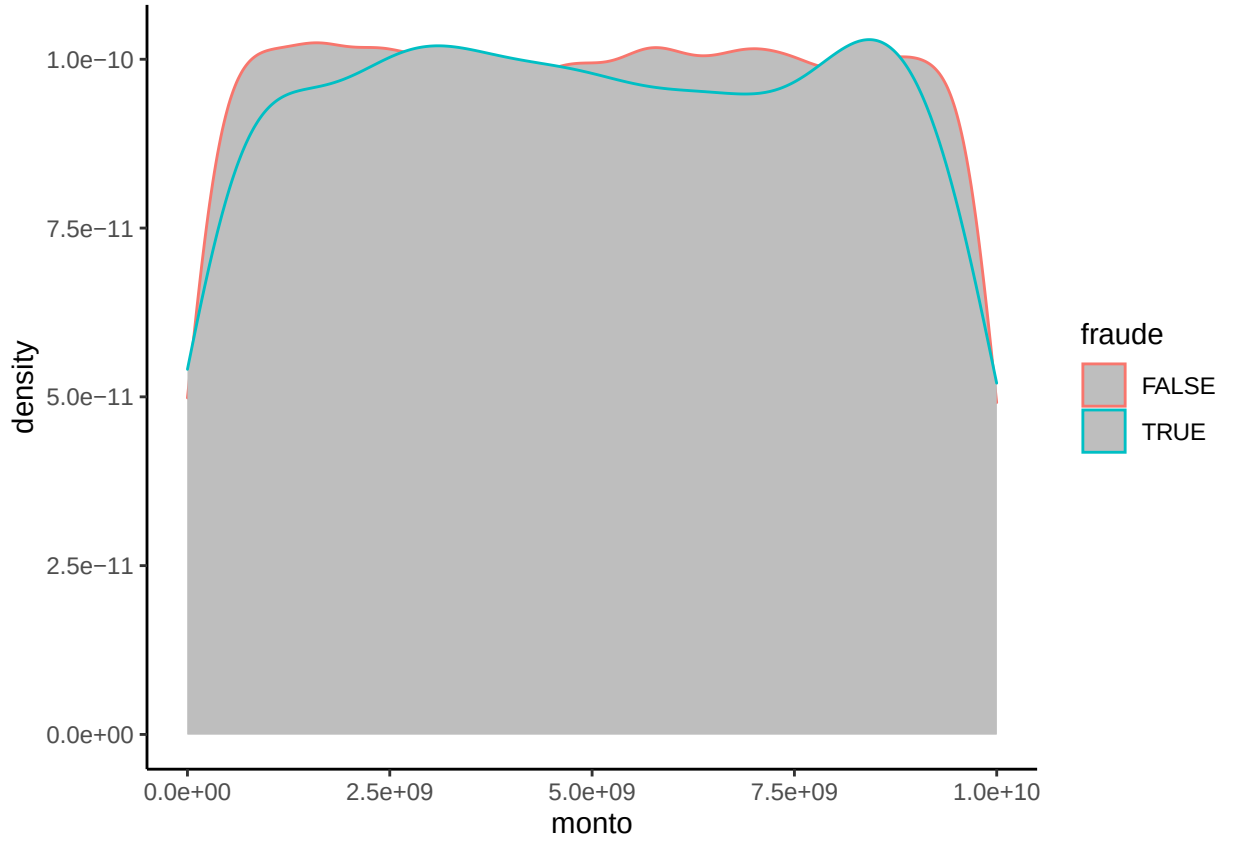
```
#Se separan las variables cualitativas y cuantitativas

variables_cuant <- c("monto","linea_tc","interes_tc","dcto","cashback")
variables_cual <- c("genero","establecimiento","ciudad","tipo_tc","fraude")
```

6. Análisis Bivariado

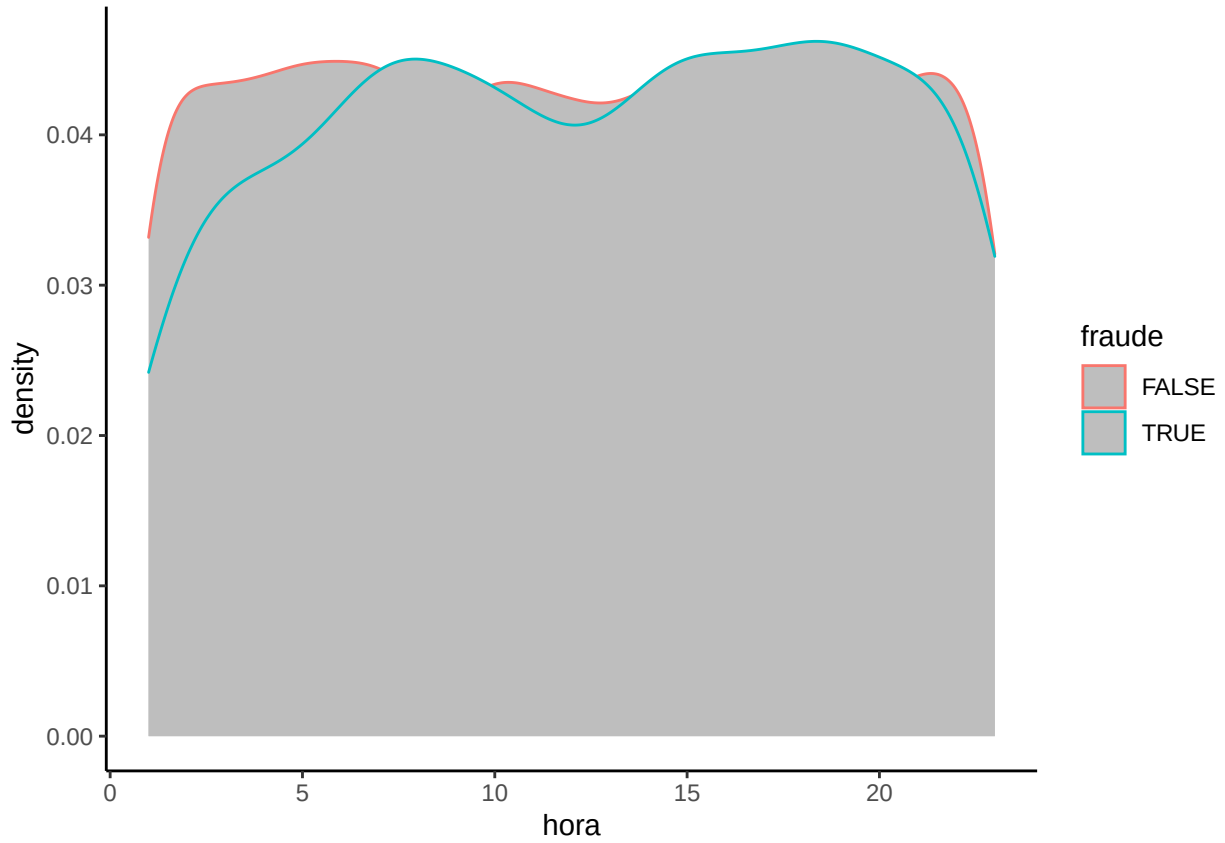
```
#Funcion para análisis bivariado con variables cuantitativas

biv_cuan_variables <- function(df,target,variable_cuant){

  g1 <- ggplot(data = df)+
    geom_density(mapping = aes(x = get(variable_cuant), colour = fraude),fill="gray")+
    xlab(variable_cuant)+
    ylab("density")+
    theme_classic()

  return(g1)
}

#Funcion para análisis bivariado con variables cualitativas

biv_cual_variables <- function(df,target,variable_cual){
```

```
  g1 <- ggplot(data = df)+
    geom_bar(mapping = aes(x = get(variable_cual), fill = fraude),position = position_fill())+
    scale_y_continuous(labels = scales::percent_format())+
    xlab(variable_cual)+
    ylab("density")+
    theme_classic()

  tab <- table(df[,c(variable_cual,"fraude")])

  print(tab)
  return(g1)
}
```

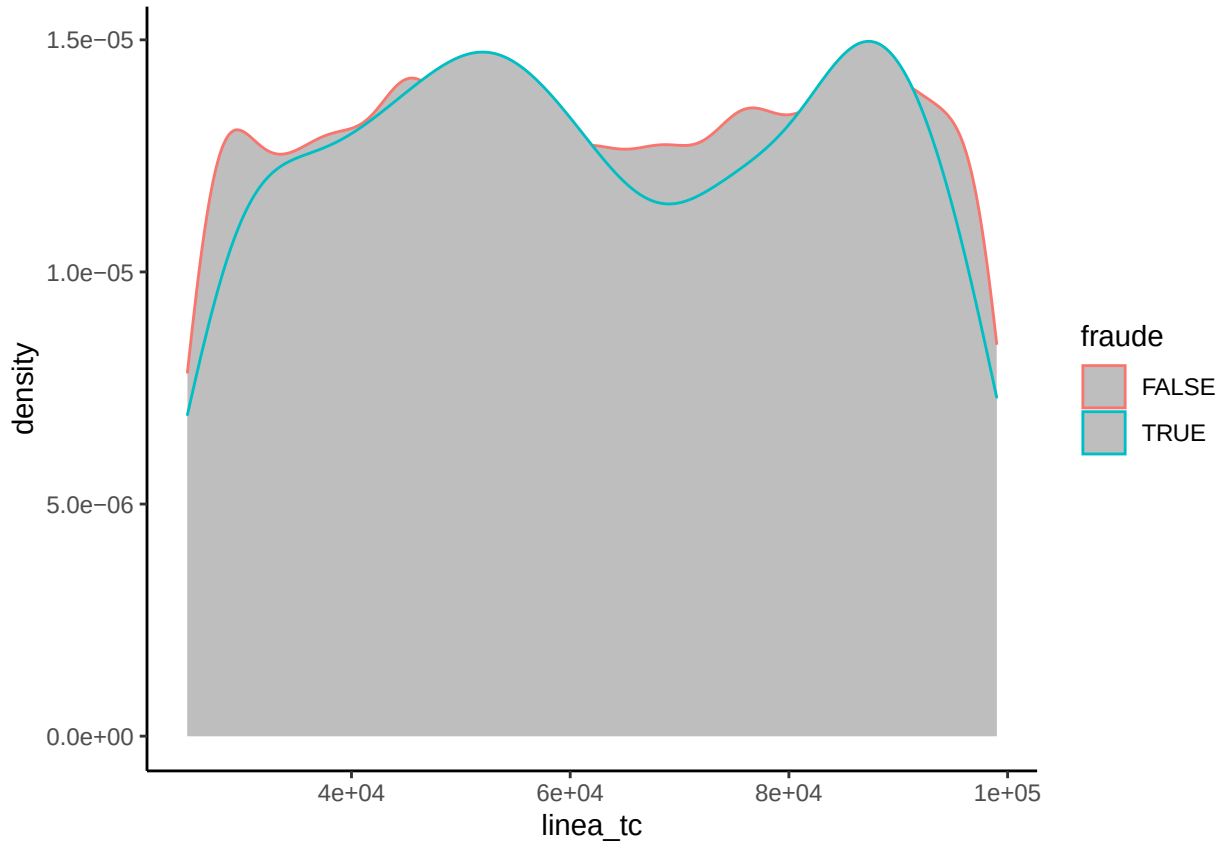
```
biv_cuan_variables(df_datos,"fraude","monto")
```



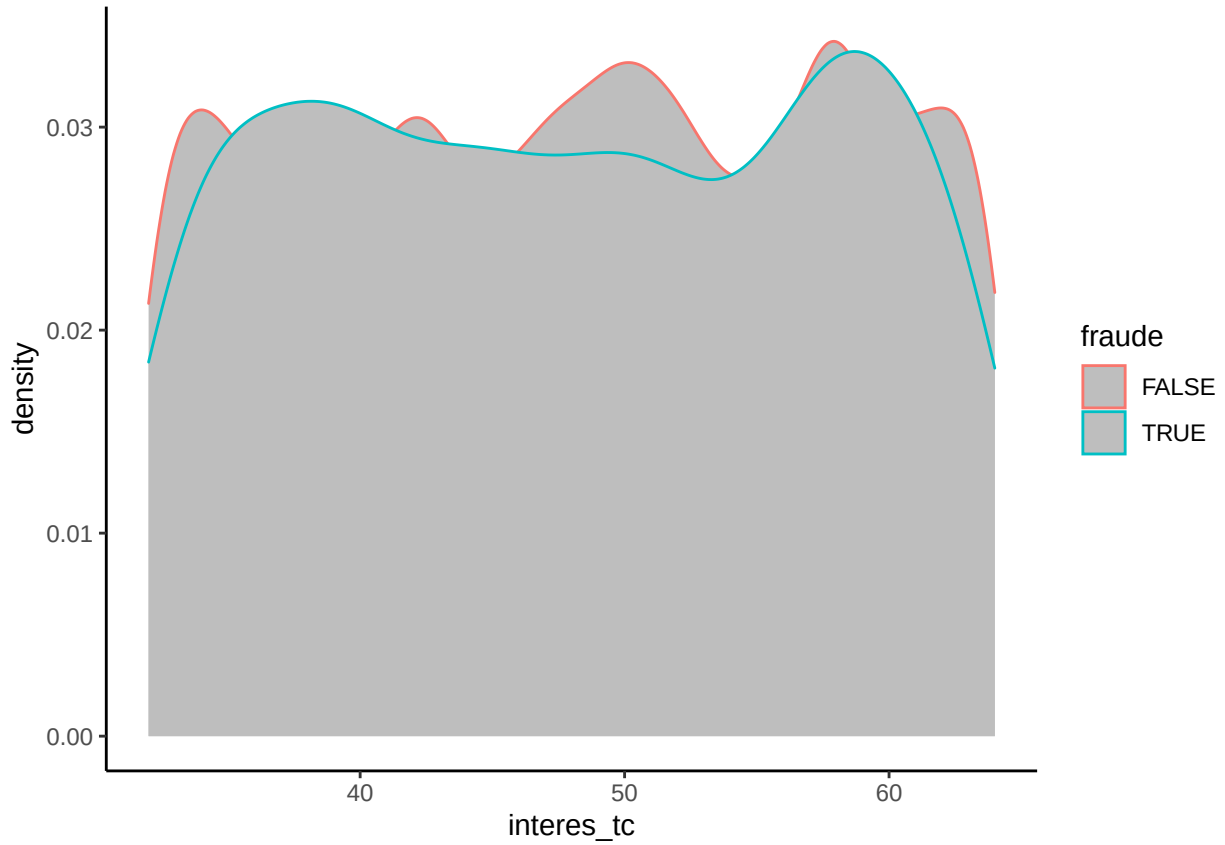

```
biv_cuan_variables(df_datos,"fraude","hora")
```



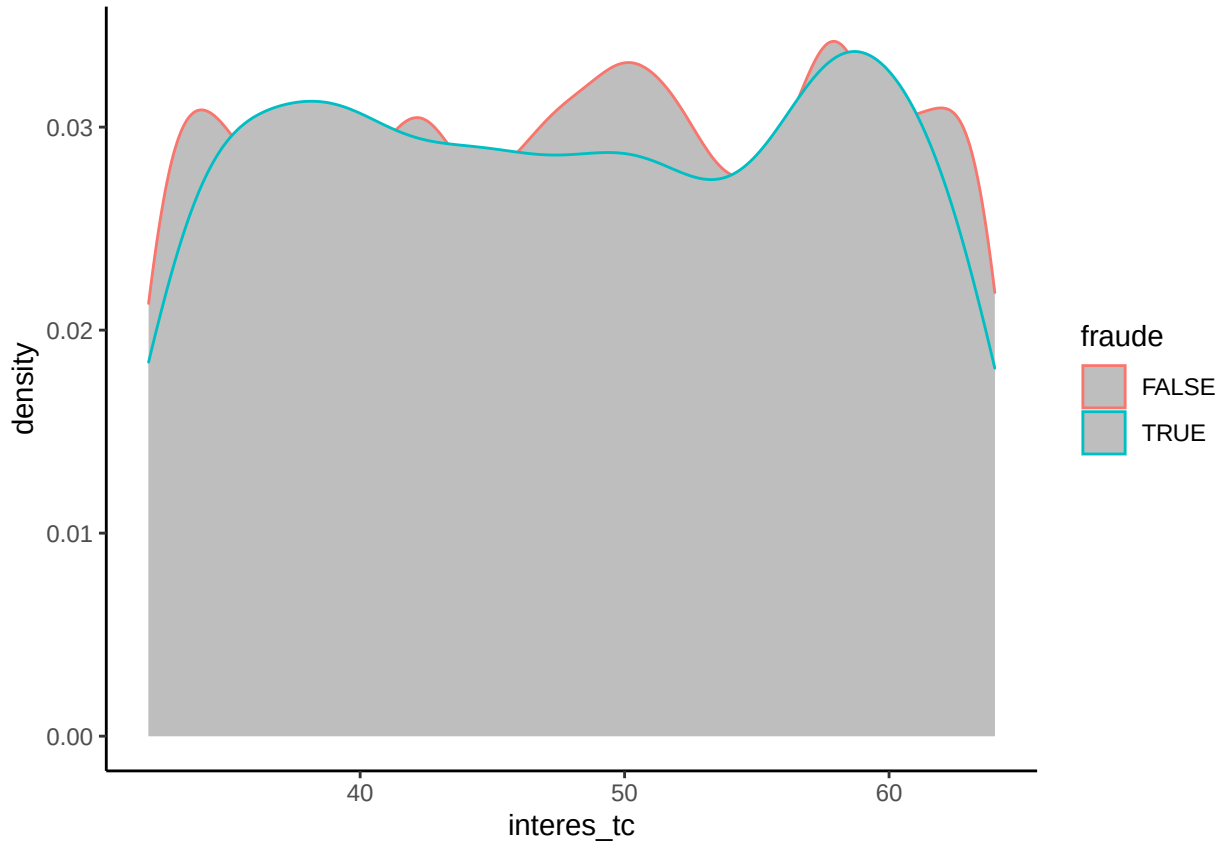

```
biv_cuan_variables(df_datos,"fraude","linea_tc")
```




```
biv_cuan_variables(df_datos,"fraude","interes_tc")
```

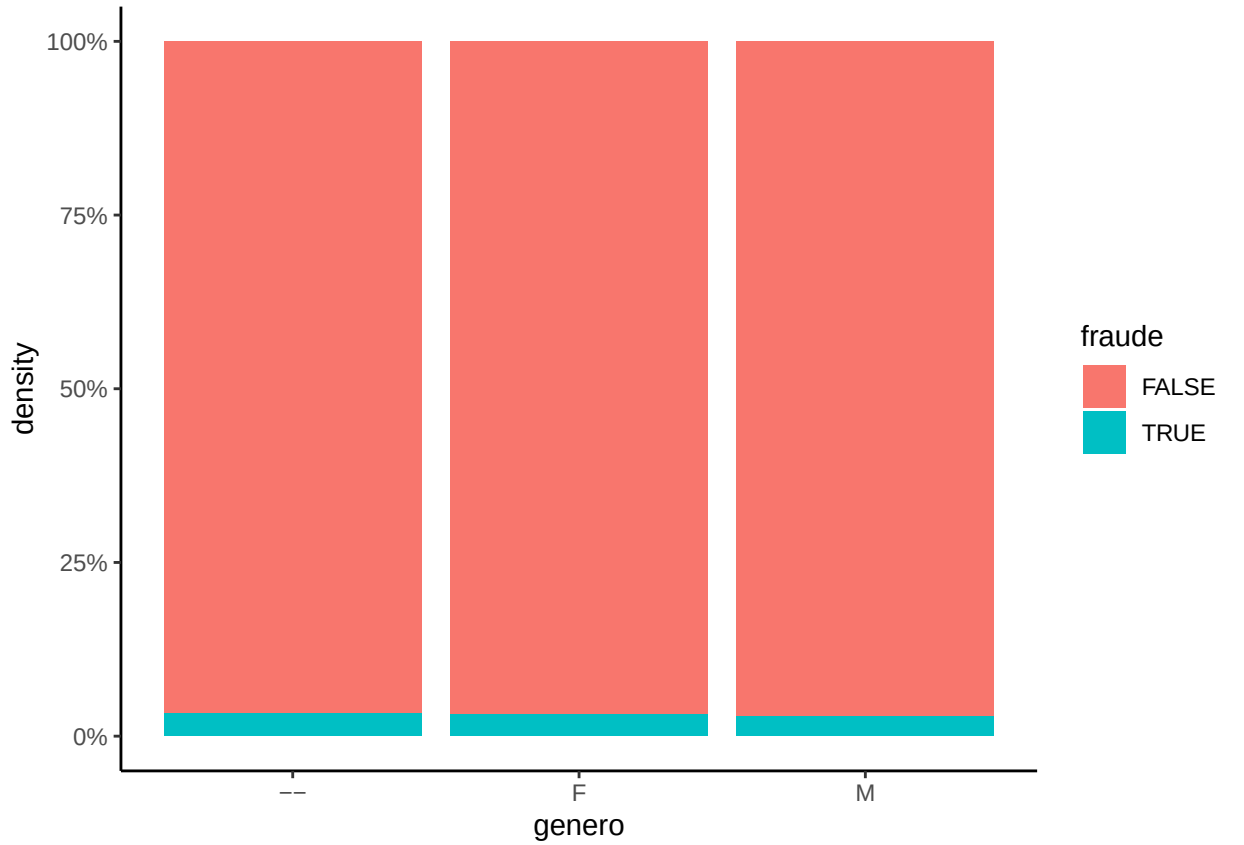



```
biv_cuan_variables(df_datos,"fraude","interes_tc")
```



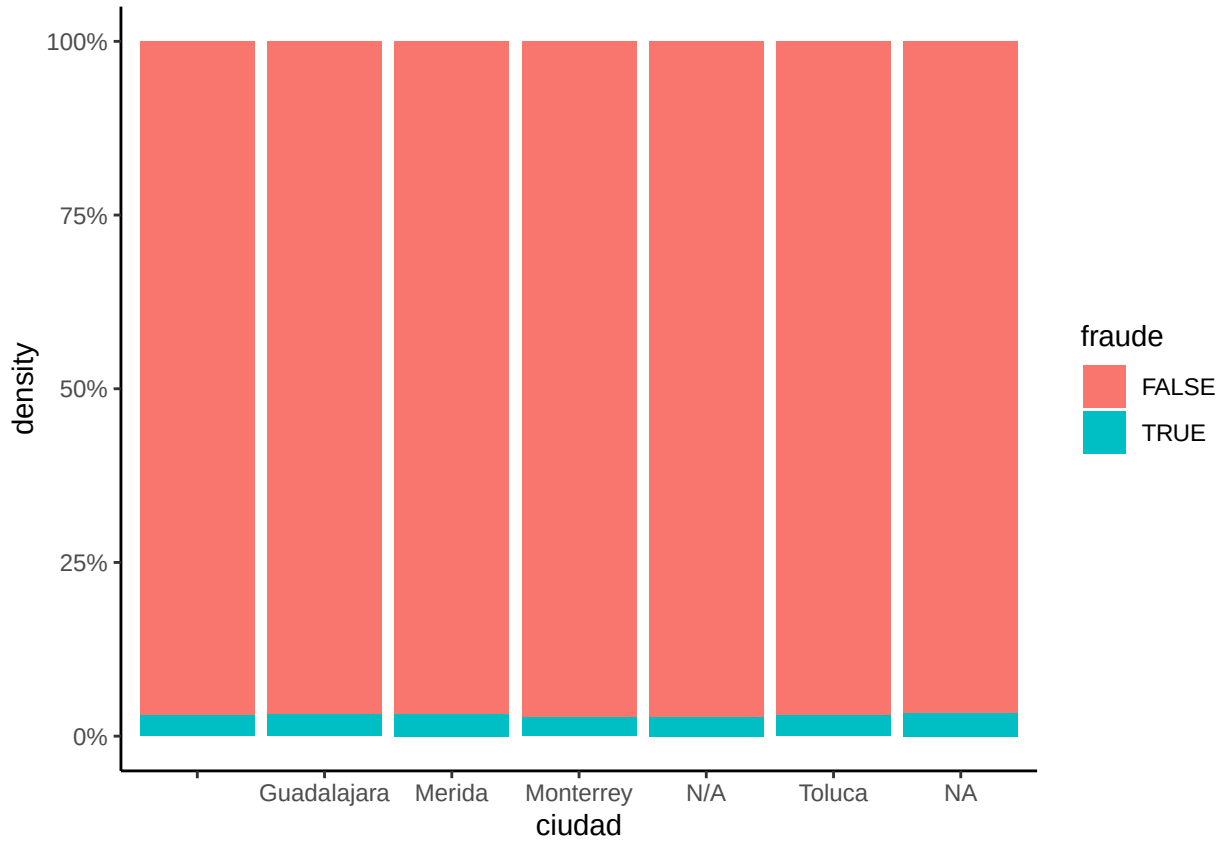

```
biv_cual_variables(df_datos,"fraude","genero")
```

```
##       fraude
## genero FALSE  TRUE
##     --  2642    88
##     F  10392   334
##     M  13131   388
```



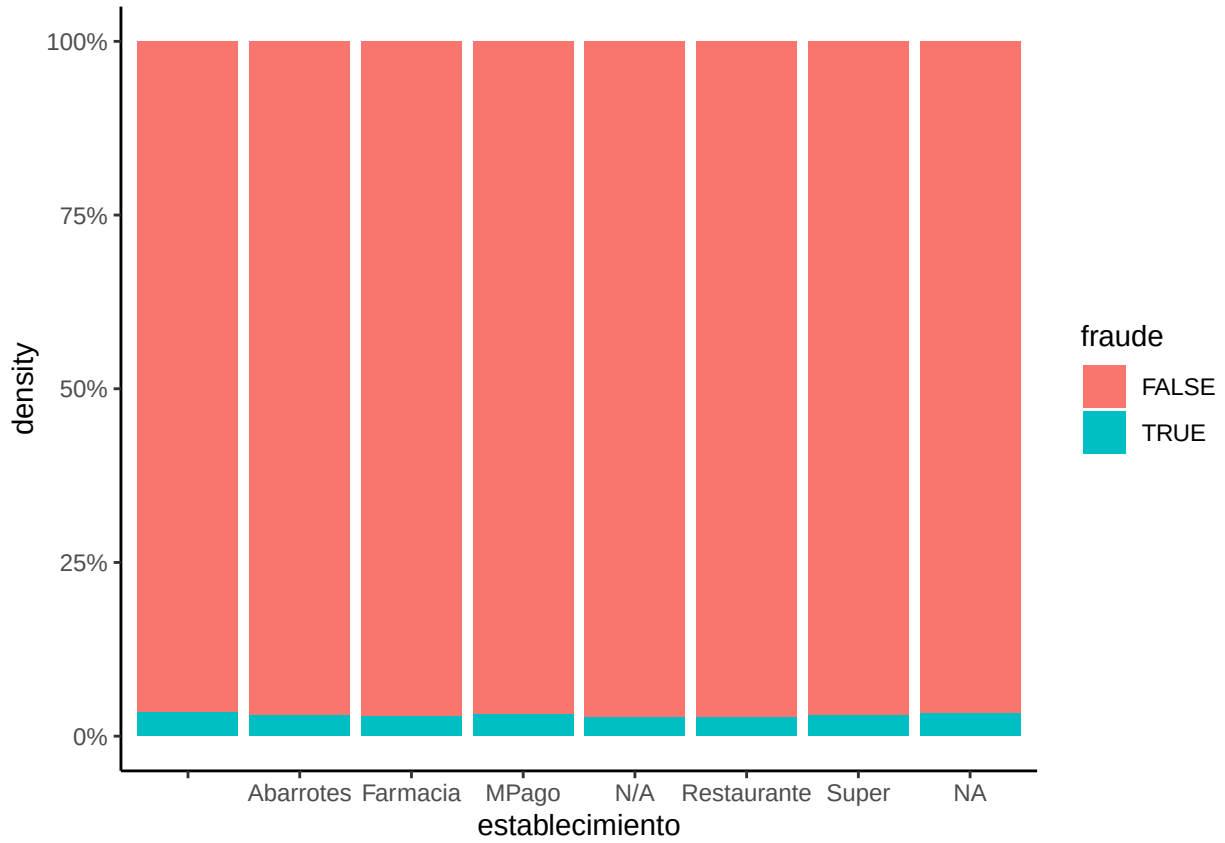

```
biv_cual_variables(df_datos,"fraude","ciudad")
```

```
##              fraude
## ciudad        FALSE TRUE
##                4063  124
##   Guadalajara  3715  118
##   Merida       3641  120
##   Monterrey    3606  100
##   N/A          3482   99
##   Toluca       3879  118
```



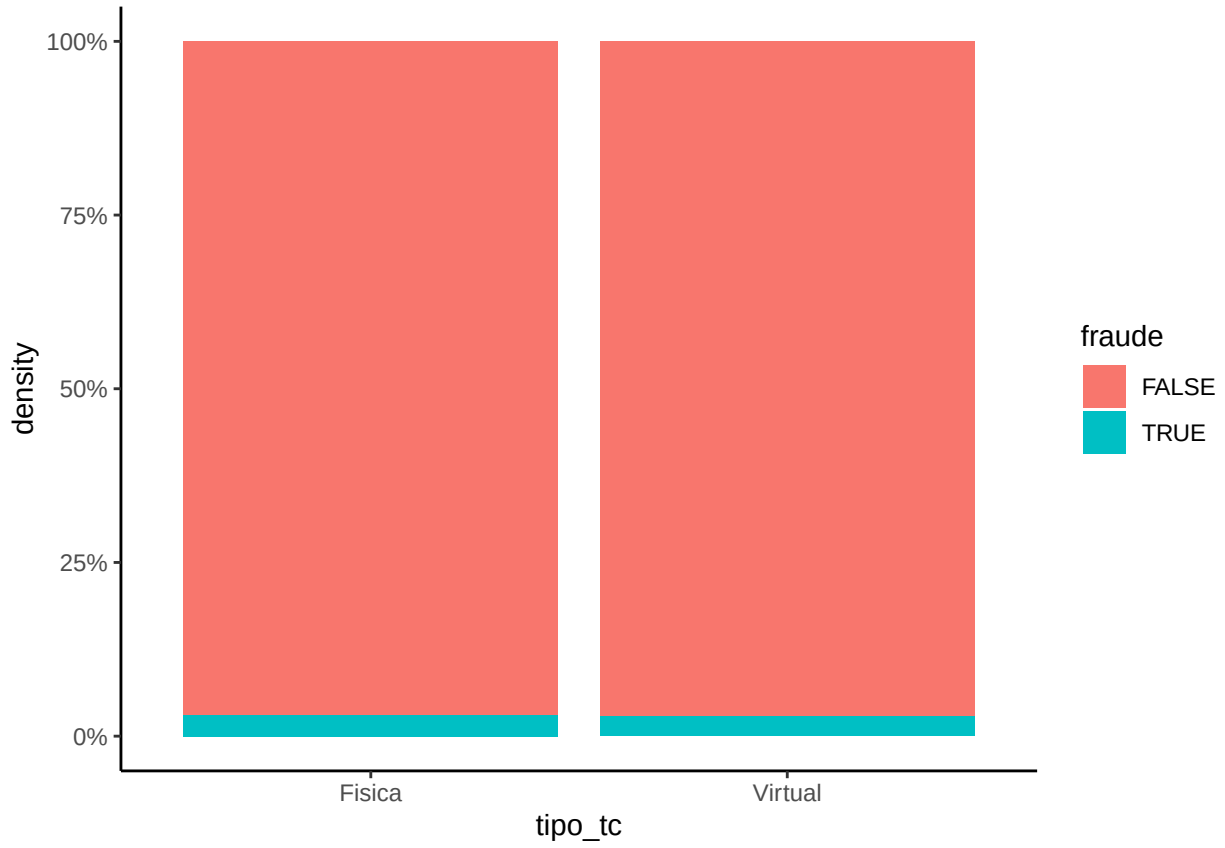

```
biv_cual_variables(df_datos,"fraude","establecimiento")
```

```
##                fraude
## establecimiento FALSE TRUE
##                  3274  115
##     Abarrotes    3313  102
##     Farmacia     3150   92
##     MPago        3239  104
##     N/A          3230   90
##     Restaurante  3361   93
##     Super        3300  102
```




```
biv_cual_variables(df_datos,"fraude","tipo_tc")
```

```
##          fraude
## tipo_tc   FALSE  TRUE
##   Fisica  18324   579
##   Virtual  7841   231
```



De los gráficos bivariados se pueden obtener las siguientes conclusiones

- Las variables cuantitativas presentan una distribución
- Las variables cualitativas no presentan mucha discriminación respecto a la variable fraude(target)

7. Clusterización de Clientes(Segmentación)

```
#Obtenemos la matriz de Correlacion para determinar qué variables se incluirán

cor(df_datos[,variables_cuant])
```

```
##                 monto    linea_tc  interes_tc       dcto    cashback
## monto       1.00000000 -0.00727318  0.01087976 0.24774809  0.46590039
## linea_tc   -0.00727318  1.00000000 -0.03993189 0.00545173 -0.01183356
## interes_tc  0.01087976 -0.03993189  1.00000000 0.01047473  0.01090385
## dcto        0.24774809  0.00545173  0.01047473 1.00000000  0.03924366
## cashback    0.46590039 -0.01183356  0.01090385 0.03924366  1.00000000
```

De la matriz se observa que ninguna supera en valor absoluto el 0.5, entonces no se descarta ninguna

```
#SE NORMALIZA LAS VARIABLES NUMERICAS

preproc1 <- preProcess(df_datos[,variables_cuant], method=c("center", "scale"))

norm1 <- predict(preproc1, df_datos[,variables_cuant])

#SE REALIZA MERGE CON LA VARIABLE DE TIPO_TC QUE TAMBIÉN SE UTILIZARÁ PARA CLUSTERIZAR
```

```
norm1$tipo_tc <- ifelse(df_datos$tipo_tc=="Virtual",1,0)

#CLUSTERS MEDIANTE K MEANS
set.seed(567)

km.res <- kmeans(norm1, 4, nstart = 25)

#SE VISUALIZA LOS RESULTOS Y LOS CENTROS PARA OBTENER DESCRIPTIVOS

print(km.res$centers)
```

```
##        monto    linea_tc   interes_tc       dcto   cashback   tipo_tc
## 1 -0.4871762 -0.90965057  0.095100485 -0.3369570 -0.5523052 0.3639284
## 2  0.8313389 -0.04163615  0.006070923 -0.4992974  1.2799264 0.1214320
## 3 -0.4730789  0.88810076 -0.111204955 -0.3371467 -0.5288251 0.3632192
## 4  0.6665516  0.01971975  0.033860583  2.1680994  0.2102429 0.3141089
```

Determinamos 4 segmentos de clientes, los cuales son:

- Perfil 1:Clientes con baja linea de TC(riesgosos), bajo monto de consumo y alta tasa de interes
- Perfil 2:Clientes con baja linea de TC(riesgosos), alto monto de consumo y con tarjeta fisica
- Perfil 3:Clientes con alta linea de TC(riesgosos), bajo monto de consumo y con tarjeta fisica
- Perfil 4:Clientes con linea de TC(riesgosos) media, alto monto de consumo y mayores descuentos

Analizando los distntos clusters se puede determinar que el perfil correspondiente a cada uno sería:

- Clientes conservadores linea baja
- Clientes con Mora potencial
- Clientes Afluentes - Se tendría que determinar por qué no consumen más y presentar más ofertas.
- Clientes connservadores linea alta

8. Modelo de prevención de fraude

```
# Con la data normalizada y considerando las variables que presentan mayor discriminación respecto a la

norm1$fraude <- ifelse(df_datos$fraude==FALSE,0,1)

#Se divide la data en train y test

sample <-  sample.split(norm1$fraude, SplitRatio = .80)
train <-  subset(norm1, sample == TRUE)
test  <-  subset(norm1, sample == FALSE)


model <- glm(fraude ~ monto+linea_tc+interes_tc+dcto+cashback+tipo_tc,
             data = train, family = binomial)

summary(model)$coef
```

```
##                 Estimate Std. Error      z value  Pr(>|z|)
## (Intercept) -3.459843383 0.04773183 -72.48503369 0.0000000
## monto       -0.002770749 0.04713331  -0.05878536 0.9531231
```

```
## linea_tc    -0.009634197 0.03993448  -0.24125006 0.8093613
## interes_tc  -0.036121166 0.03988733  -0.90558001 0.3651582
## dcto        -0.005531963 0.04192775  -0.13194037 0.8950315
## cashback    -0.033978382 0.04705393  -0.72211566 0.4702234
## tipo_tc     -0.055766977 0.09027354  -0.61775551 0.5367365
```

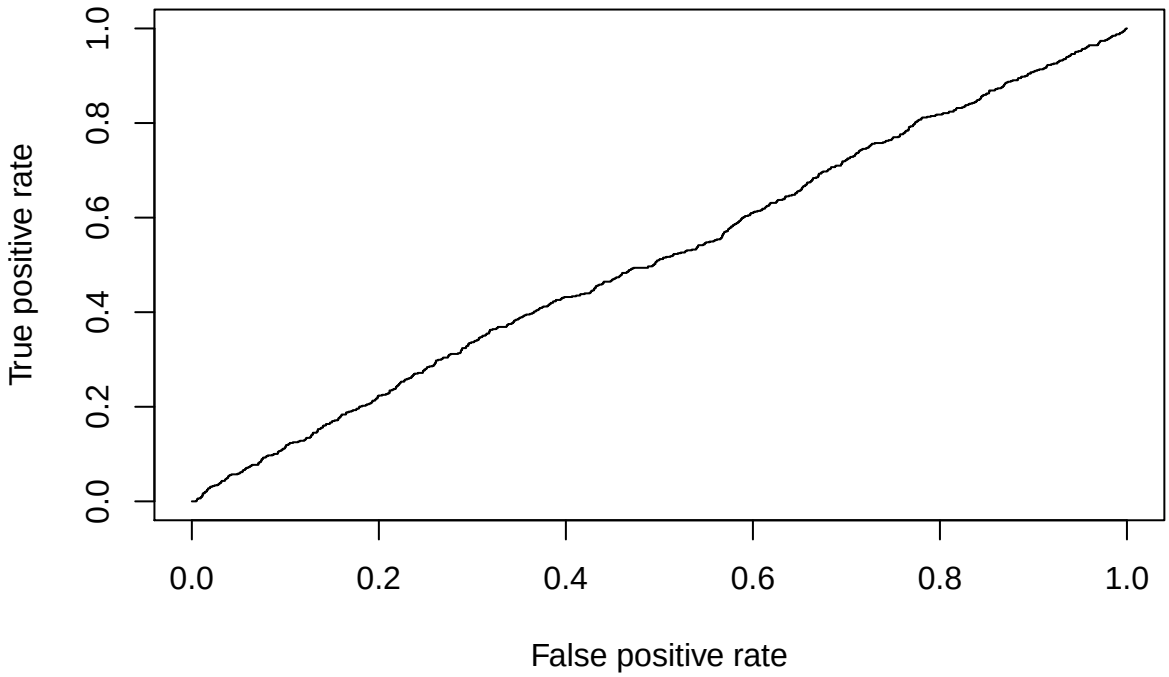
Todas las variables ingresadas en el modelo tienen p-value significativo y no se descarta ninguna

```
# Se obtiene la importancia de las variables
varImp(model)
```

```
##               Overall
## monto      0.05878536
## linea_tc   0.24125006
## interes_tc 0.90558001
## dcto       0.13194037
## cashback   0.72211566
## tipo_tc    0.61775551
```

Dentro de las variables más importantes en el modelo están:
* TIPO DE TARJETA * CANTIDAD DE CASHBACK * INTERÉS DE TC

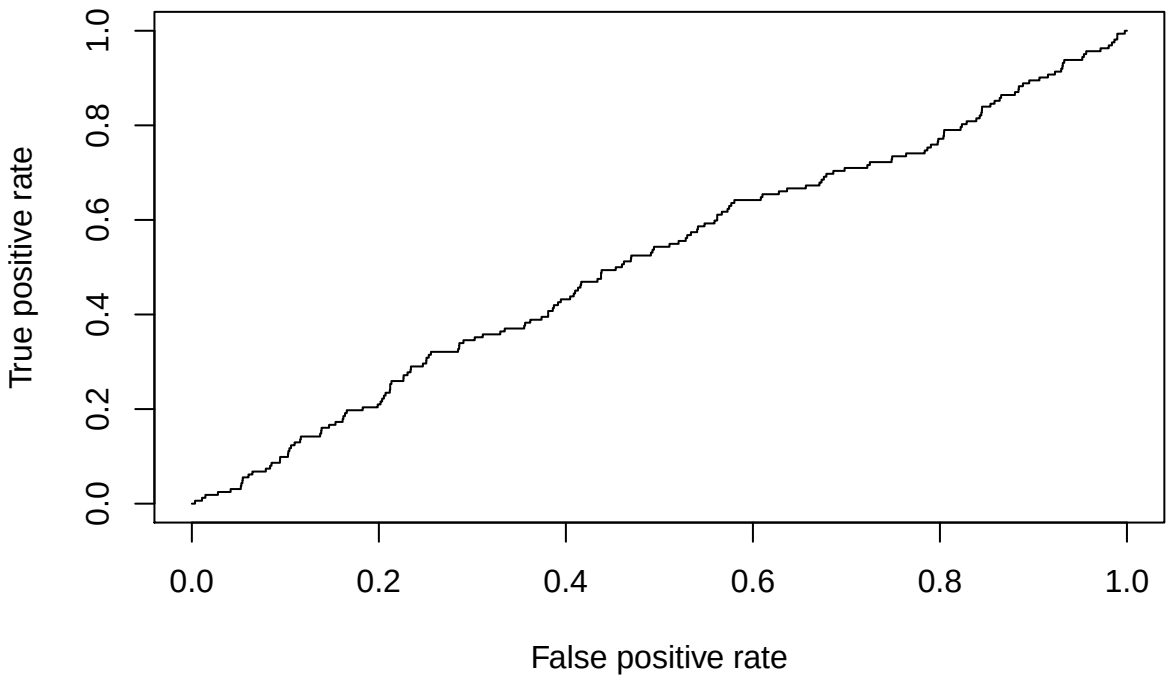
```
# Se calcula el AUC para la data de train
prob <- predict(model, newdata=train, type="response")
pred <- prediction(prob, train$fraude)
perf <- performance(pred, measure = "tpr", x.measure = "fpr")
plot(perf)
```



```
auc <- performance(pred, measure = "auc")
auc <- auc@y.values[[1]]
auc
```

```
## [1] 0.515879
```

```
# Se calcula el AUC para la data de test
prob <- predict(model, newdata=test, type="response")
pred <- prediction(prob, test$fraude)
perf <- performance(pred, measure = "tpr", x.measure = "fpr")
plot(perf)
```




```
auc <- performance(pred, measure = "auc")
auc <- auc@y.values[[1]]
auc
```

```
## [1] 0.5150399
```

De los valores de AUC, tanto de train y test, se puede ver que no hay un buen poder de discriminación. Esto se puedo ver en el análisis de las variables, en las cuales se ve que las distribuciones de los que sufrieron fraude y los que no son similares.

Esto sin embargo, podría ser mejorado con técnicas más complejas de machine learning.

9. Trade-Off del Modelo

- Al usar esta técnica econométrica para modelar nos da una fácil interpretabilidad de las variables y su aporte.

- El modelo nos permite gestionar y tomar mayor acción en las variables de mayor importancia y ver cuánto aumenta la probabilidad de ser fraude.
- No requiere muchos recursos para ejecutar y es rápido.
- La data no es linealmente separable, por lo que el modelo no tiene mucho poder predictivo.
- Rápido para poder implementar para nuevas observaciones

10. INSIGHTS RELEVANTES

- Las variables en el dataset no ayuda a discriminar los fraudes.
- Las variables que mayor aportan son las cualitativas.
- Con modelos como RNN se podría lograr mayor AUC pero se perdería interpretabilidad.
- En el segmento de afluentes se tendría que priorizar para que tengan un ticket promedio mayor, debido a que tiene mayor capacidad(linea) y se podría generar mayores beneficios.
- Se debería buscar más variables como demográficas o históricas para poder evaluarlas e ingresarlas en un nuevo modelo.