TFG Codigo

Diego Brito

Librerias

```
library(MASS)
library(tidyverse)
library(readr)
library(psych)
library(ggplot2)
library(dplyr)
library(corrplot)
library(RColorBrewer)
library(gridExtra)
library(caret)
library(pROC)
library(car)

# library(MXM)
# library(parallel)
# library(doParallel)
```

Base de datos

```
setwd("C:\\Users\\diego\\OneDrive\\Escritorio\\UCM\\Cuarto\\Segundo Cuatri")
datos <- read.csv(file = "application_data.csv")
```

Depuracion de datos

primero vemos cuantas observaciones faltantes hay por columna

```r
data.frame(sort(colSums(is.na(datos))))
```

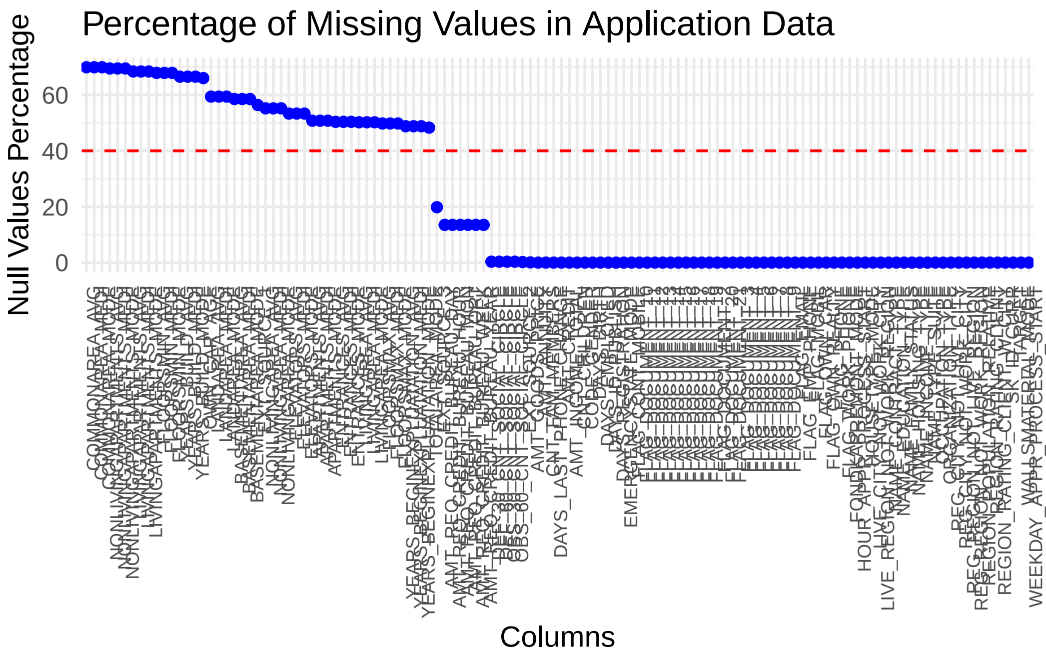
```
                              sort.colSums.is.na.datos...
SK_ID_CURR                                              0
TARGET                                                  0
NAME_CONTRACT_TYPE                                      0
CODE_GENDER                                             0
FLAG_OWN_CAR                                            0
FLAG_OWN_REALTY                                         0
CNT_CHILDREN                                            0
AMT_INCOME_TOTAL                                        0
AMT_CREDIT                                              0
NAME_TYPE_SUITE                                         0
NAME_INCOME_TYPE                                        0
NAME_EDUCATION_TYPE                                     0
NAME_FAMILY_STATUS                                      0
NAME_HOUSING_TYPE                                       0
REGION_POPULATION_RELATIVE                              0
DAYS_BIRTH                                              0
DAYS_EMPLOYED                                           0
DAYS_REGISTRATION                                       0
DAYS_ID_PUBLISH                                         0
FLAG_MOBIL                                              0
FLAG_EMP_PHONE                                          0
FLAG_WORK_PHONE                                         0
FLAG_CONT_MOBILE                                        0
FLAG_PHONE                                              0
FLAG_EMAIL                                              0
OCCUPATION_TYPE                                         0
REGION_RATING_CLIENT                                    0
REGION_RATING_CLIENT_W_CITY                             0
WEEKDAY_APPR_PROCESS_START                              0
HOUR_APPR_PROCESS_START                                 0
REG_REGION_NOT_LIVE_REGION                              0
REG_REGION_NOT_WORK_REGION                              0
LIVE_REGION_NOT_WORK_REGION                             0
REG_CITY_NOT_LIVE_CITY                                  0
REG_CITY_NOT_WORK_CITY                                  0
LIVE_CITY_NOT_WORK_CITY                                 0
ORGANIZATION_TYPE                                       0
FONDKAPREMONT_MODE                                      0
HOUSETYPE_MODE                                          0
```

WALLSMATERIAL_MODE	0
EMERGENCYSTATE_MODE	0
FLAG_DOCUMENT_2	0
FLAG_DOCUMENT_3	0
FLAG_DOCUMENT_4	0
FLAG_DOCUMENT_5	0
FLAG_DOCUMENT_6	0
FLAG_DOCUMENT_7	0
FLAG_DOCUMENT_8	0
FLAG_DOCUMENT_9	0
FLAG_DOCUMENT_10	0
FLAG_DOCUMENT_11	0
FLAG_DOCUMENT_12	0
FLAG_DOCUMENT_13	0
FLAG_DOCUMENT_14	0
FLAG_DOCUMENT_15	0
FLAG_DOCUMENT_16	0
FLAG_DOCUMENT_17	0
FLAG_DOCUMENT_18	0
FLAG_DOCUMENT_19	0
FLAG_DOCUMENT_20	0
FLAG_DOCUMENT_21	0
DAYS_LAST_PHONE_CHANGE	1
CNT_FAM_MEMBERS	2
AMT_ANNUITY	12
AMT_GOODS_PRICE	278
EXT_SOURCE_2	660
OBS_30_CNT_SOCIAL_CIRCLE	1021
DEF_30_CNT_SOCIAL_CIRCLE	1021
OBS_60_CNT_SOCIAL_CIRCLE	1021
DEF_60_CNT_SOCIAL_CIRCLE	1021
AMT_REQ_CREDIT_BUREAU_HOUR	41519
AMT_REQ_CREDIT_BUREAU_DAY	41519
AMT_REQ_CREDIT_BUREAU_WEEK	41519
AMT_REQ_CREDIT_BUREAU_MON	41519
AMT_REQ_CREDIT_BUREAU_QRT	41519
AMT_REQ_CREDIT_BUREAU_YEAR	41519
EXT_SOURCE_3	60965
TOTALAREA_MODE	148431
YEARS_BEGINEXPLUATATION_AVG	150007
YEARS_BEGINEXPLUATATION_MODE	150007
YEARS_BEGINEXPLUATATION_MEDI	150007
FLOORSMAX_AVG	153020

FLOORSMAX_MODE	153020
FLOORSMAX_MEDI	153020
LIVINGAREA_AVG	154350
LIVINGAREA_MODE	154350
LIVINGAREA_MEDI	154350
ENTRANCES_AVG	154828
ENTRANCES_MODE	154828
ENTRANCES_MEDI	154828
APARTMENTS_AVG	156061
APARTMENTS_MODE	156061
APARTMENTS_MEDI	156061
ELEVATORS_AVG	163891
ELEVATORS_MODE	163891
ELEVATORS_MEDI	163891
NONLIVINGAREA_AVG	169682
NONLIVINGAREA_MODE	169682
NONLIVINGAREA_MEDI	169682
EXT_SOURCE_1	173378
BASEMENTAREA_AVG	179943
BASEMENTAREA_MODE	179943
BASEMENTAREA_MEDI	179943
LANDAREA_AVG	182590
LANDAREA_MODE	182590
LANDAREA_MEDI	182590
OWN_CAR_AGE	202929
YEARS_BUILD_AVG	204488
YEARS_BUILD_MODE	204488
YEARS_BUILD_MEDI	204488
FLOORSMIN_AVG	208642
FLOORSMIN_MODE	208642
FLOORSMIN_MEDI	208642
LIVINGAPARTMENTS_AVG	210199
LIVINGAPARTMENTS_MODE	210199
LIVINGAPARTMENTS_MEDI	210199
NONLIVINGAPARTMENTS_AVG	213514
NONLIVINGAPARTMENTS_MODE	213514
NONLIVINGAPARTMENTS_MEDI	213514
COMMONAREA_AVG	214865
COMMONAREA_MODE	214865
COMMONAREA_MEDI	214865

ahora tenemos que ver que hacemos con esas observaciones, hay 2 opciones, eliminar aquellas observaciones o sistituir los valores aplicando reglas sustitutivas

```r
# Calcular el porcentaje de valores nulos por columna
null_datos_df <- datos |>
  summarise(across(everything(), ~ sum(is.na(.)) * 100 / n()))  |> # control + shift + m
  pivot_longer(cols = everything(), names_to = "Column_Name", values_to = "Null_Values_Percen

# Crear el gráfico de puntos
ggplot(null_datos_df, aes(x = reorder(Column_Name, -Null_Values_Percentage), y = Null_Values_
  geom_point(color = "blue") +
  geom_hline(yintercept = 40, linetype = "dashed", color = "red") +  # Línea de referencia al
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, size = 7)) +
  labs(title = "Percentage of Missing Values in Application Data",
       x = "Columns",
       y = "Null Values Percentage")
```



Variables con mas de un 40 % de datos faltantes

```r
# que columnas tienen mas del 40 % de sus datos missing o NA
# Filtrar columnas con 40% o más de valores nulos
# ponemos como limite un 40 % de datos faltantes, porque sistituir mas de un 40 - 50 % de dat
# con la mediana o media no es buena idea teniendo tanto % de datos faltantes

nullcol_40_application <- null_datos_df  |>
```

```r
  filter(Null_Values_Percentage >= 40)

# Mostrar el resultado
print(nullcol_40_application)
```

```
# A tibble: 45 x 2
   Column_Name                  Null_Values_Percentage
   <chr>                                         <dbl>
 1 OWN_CAR_AGE                                    66.0
 2 EXT_SOURCE_1                                   56.4
 3 APARTMENTS_AVG                                 50.7
 4 BASEMENTAREA_AVG                               58.5
 5 YEARS_BEGINEXPLUATATION_AVG                    48.8
 6 YEARS_BUILD_AVG                                66.5
 7 COMMONAREA_AVG                                 69.9
 8 ELEVATORS_AVG                                  53.3
 9 ENTRANCES_AVG                                  50.3
10 FLOORSMAX_AVG                                  49.8
# i 35 more rows
```

Datos faltantes

cuantos datos faltantes tenemos por columna

```r
categorical_columns <- c('NAME_CONTRACT_TYPE', 'CODE_GENDER', 'NAME_TYPE_SUITE', 'NAME_INCOM

contact_col <- c("FLAG_MOBIL", "FLAG_EMP_PHONE", "FLAG_WORK_PHONE",
                 "FLAG_CONT_MOBILE", "FLAG_PHONE", "FLAG_EMAIL")

col_Doc <- c("FLAG_DOCUMENT_2", "FLAG_DOCUMENT_3", "FLAG_DOCUMENT_4", "FLAG_DOCUMENT_5", "FLA
             "FLAG_DOCUMENT_7", "FLAG_DOCUMENT_8", "FLAG_DOCUMENT_9", "FLAG_DOCUMENT_10", "FL
             "FLAG_DOCUMENT_12", "FLAG_DOCUMENT_13", "FLAG_DOCUMENT_14", "FLAG_DOCUMENT_15",
             "FLAG_DOCUMENT_17", "FLAG_DOCUMENT_18", "FLAG_DOCUMENT_19", "FLAG_DOCUMENT_20",

ext <- c("EXT_SOURCE_1", "EXT_SOURCE_2", "EXT_SOURCE_3")
data.frame(sort(colSums(is.na(datos))))
```

```
                             sort.colSums.is.na.datos...
SK_ID_CURR                                             0
TARGET                                                 0
```

NAME_CONTRACT_TYPE	0
CODE_GENDER	0
FLAG_OWN_CAR	0
FLAG_OWN_REALTY	0
CNT_CHILDREN	0
AMT_INCOME_TOTAL	0
AMT_CREDIT	0
NAME_TYPE_SUITE	0
NAME_INCOME_TYPE	0
NAME_EDUCATION_TYPE	0
NAME_FAMILY_STATUS	0
NAME_HOUSING_TYPE	0
REGION_POPULATION_RELATIVE	0
DAYS_BIRTH	0
DAYS_EMPLOYED	0
DAYS_REGISTRATION	0
DAYS_ID_PUBLISH	0
FLAG_MOBIL	0
FLAG_EMP_PHONE	0
FLAG_WORK_PHONE	0
FLAG_CONT_MOBILE	0
FLAG_PHONE	0
FLAG_EMAIL	0
OCCUPATION_TYPE	0
REGION_RATING_CLIENT	0
REGION_RATING_CLIENT_W_CITY	0
WEEKDAY_APPR_PROCESS_START	0
HOUR_APPR_PROCESS_START	0
REG_REGION_NOT_LIVE_REGION	0
REG_REGION_NOT_WORK_REGION	0
LIVE_REGION_NOT_WORK_REGION	0
REG_CITY_NOT_LIVE_CITY	0
REG_CITY_NOT_WORK_CITY	0
LIVE_CITY_NOT_WORK_CITY	0
ORGANIZATION_TYPE	0
FONDKAPREMONT_MODE	0
HOUSETYPE_MODE	0
WALLSMATERIAL_MODE	0
EMERGENCYSTATE_MODE	0
FLAG_DOCUMENT_2	0
FLAG_DOCUMENT_3	0
FLAG_DOCUMENT_4	0
FLAG_DOCUMENT_5	0

FLAG_DOCUMENT_6	0
FLAG_DOCUMENT_7	0
FLAG_DOCUMENT_8	0
FLAG_DOCUMENT_9	0
FLAG_DOCUMENT_10	0
FLAG_DOCUMENT_11	0
FLAG_DOCUMENT_12	0
FLAG_DOCUMENT_13	0
FLAG_DOCUMENT_14	0
FLAG_DOCUMENT_15	0
FLAG_DOCUMENT_16	0
FLAG_DOCUMENT_17	0
FLAG_DOCUMENT_18	0
FLAG_DOCUMENT_19	0
FLAG_DOCUMENT_20	0
FLAG_DOCUMENT_21	0
DAYS_LAST_PHONE_CHANGE	1
CNT_FAM_MEMBERS	2
AMT_ANNUITY	12
AMT_GOODS_PRICE	278
EXT_SOURCE_2	660
OBS_30_CNT_SOCIAL_CIRCLE	1021
DEF_30_CNT_SOCIAL_CIRCLE	1021
OBS_60_CNT_SOCIAL_CIRCLE	1021
DEF_60_CNT_SOCIAL_CIRCLE	1021
AMT_REQ_CREDIT_BUREAU_HOUR	41519
AMT_REQ_CREDIT_BUREAU_DAY	41519
AMT_REQ_CREDIT_BUREAU_WEEK	41519
AMT_REQ_CREDIT_BUREAU_MON	41519
AMT_REQ_CREDIT_BUREAU_QRT	41519
AMT_REQ_CREDIT_BUREAU_YEAR	41519
EXT_SOURCE_3	60965
TOTALAREA_MODE	148431
YEARS_BEGINEXPLUATATION_AVG	150007
YEARS_BEGINEXPLUATATION_MODE	150007
YEARS_BEGINEXPLUATATION_MEDI	150007
FLOORSMAX_AVG	153020
FLOORSMAX_MODE	153020
FLOORSMAX_MEDI	153020
LIVINGAREA_AVG	154350
LIVINGAREA_MODE	154350
LIVINGAREA_MEDI	154350
ENTRANCES_AVG	154828

```
ENTRANCES_MODE                                    154828
ENTRANCES_MEDI                                    154828
APARTMENTS_AVG                                    156061
APARTMENTS_MODE                                   156061
APARTMENTS_MEDI                                   156061
ELEVATORS_AVG                                     163891
ELEVATORS_MODE                                    163891
ELEVATORS_MEDI                                    163891
NONLIVINGAREA_AVG                                 169682
NONLIVINGAREA_MODE                                169682
NONLIVINGAREA_MEDI                                169682
EXT_SOURCE_1                                      173378
BASEMENTAREA_AVG                                  179943
BASEMENTAREA_MODE                                 179943
BASEMENTAREA_MEDI                                 179943
LANDAREA_AVG                                      182590
LANDAREA_MODE                                     182590
LANDAREA_MEDI                                     182590
OWN_CAR_AGE                                       202929
YEARS_BUILD_AVG                                   204488
YEARS_BUILD_MODE                                  204488
YEARS_BUILD_MEDI                                  204488
FLOORSMIN_AVG                                     208642
FLOORSMIN_MODE                                    208642
FLOORSMIN_MEDI                                    208642
LIVINGAPARTMENTS_AVG                              210199
LIVINGAPARTMENTS_MODE                             210199
LIVINGAPARTMENTS_MEDI                             210199
NONLIVINGAPARTMENTS_AVG                           213514
NONLIVINGAPARTMENTS_MODE                          213514
NONLIVINGAPARTMENTS_MEDI                          213514
COMMONAREA_AVG                                    214865
COMMONAREA_MODE                                   214865
COMMONAREA_MEDI                                   214865
```

```
# Convertir las columnas a factor (categóricas)
datos[categorical_columns] <- lapply(datos[categorical_columns], as.factor)
```

Factorizamos las variables contacto y otras que sean necesarias

```
datos <- datos %>%
  mutate(across(all_of(contact_col), as.factor)) %>%
  mutate(across(all_of(col_Doc), as.factor))
```

variables categoricas

con pocos datos faltantes (moda)

```
# Función para imputar valores faltantes con la moda
 imputar_moda <- function(x) {
   if (is.factor(x) | is.character(x)) {  # Verifica si es categórica
     moda <- names(sort(table(x), decreasing = TRUE))[1]  # Encuentra la moda
     x[is.na(x)] <- moda  # Reemplaza los NA con la moda
   }
   return(x)
 }
```

```
#categorical_columns <- c(categorical_columns,"AMT_INCOME_RANGE")
# Aplicar la función a todas las columnas categóricas
 datos[categorical_columns] <- lapply(datos[categorical_columns], imputar_moda)
```

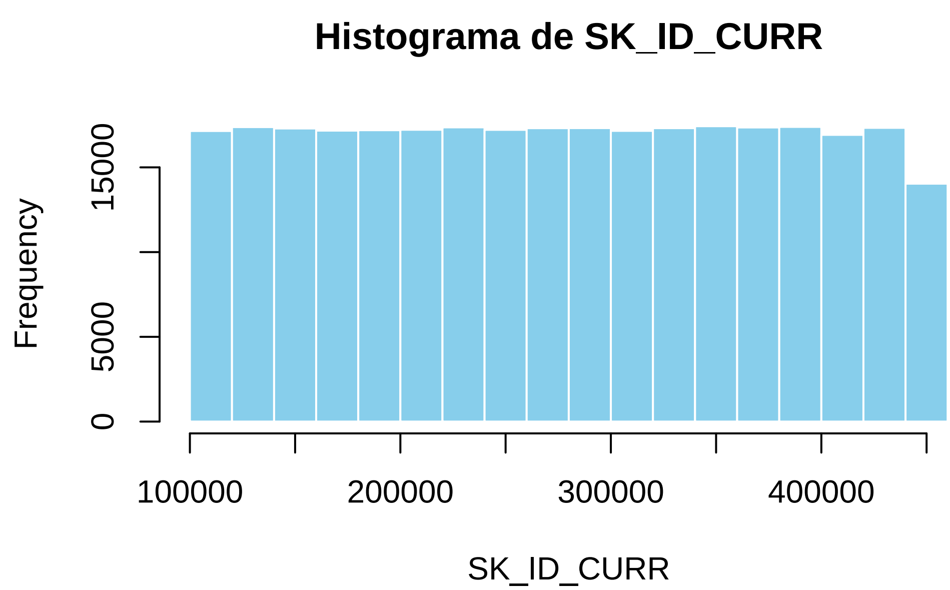
variables numericas

para sustituir aquellas variables que son numericas y tienen una observacion faltante, haremos uso de la media.

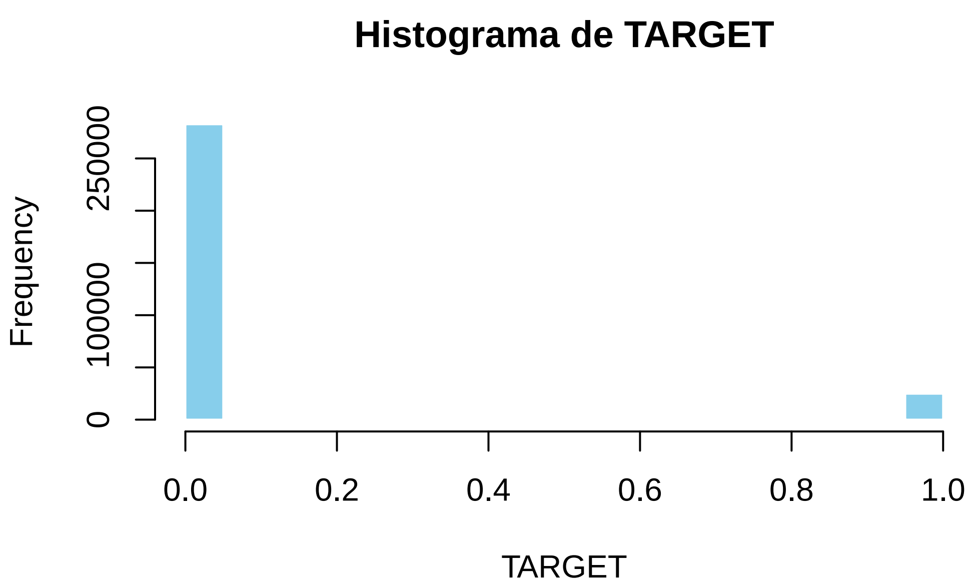
```
distribucion_variables_numericas <- function(datos) {
  numeric_columns <- datos |> select_if(is.numeric) |> names()  # Selecciona las variables nu

  for (col in numeric_columns) {
    cat("\n--------------------------------------------------\n")
    cat("Distribución de la variable:", col, "\n")
    cat("--------------------------------------------------\n")

    print(summary(datos[[col]]))  # Resumen estadístico
    hist(datos[[col]], main = paste("Histograma de", col), col = "skyblue", border = "white"

    # Test de Kolmogorov-Smirnov para normalidad
    ks_test <- ks.test(datos[[col]], "pnorm", mean(datos[[col]], na.rm = TRUE), sd(datos[[co

    cat("\nTest de Kolmogorov-Smirnov para la normalidad:\n")
```

```r
    print(ks_test)

    if (ks_test$p.value < 0.05) {
      cat("  La variable", col, "NO sigue una distribución normal (p <", ks_test$p.value, ")\
    } else {
      cat("  La variable", col, "SIGUE una distribución normal (p =", ks_test$p.value, ")\n")
    }
  }
}

# Llamada a la función
distribucion_variables_numericas(datos)
```

```
--------------------------------------------------
Distribución de la variable: SK_ID_CURR
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 100002  189146  278202  278181  367143  456255
```

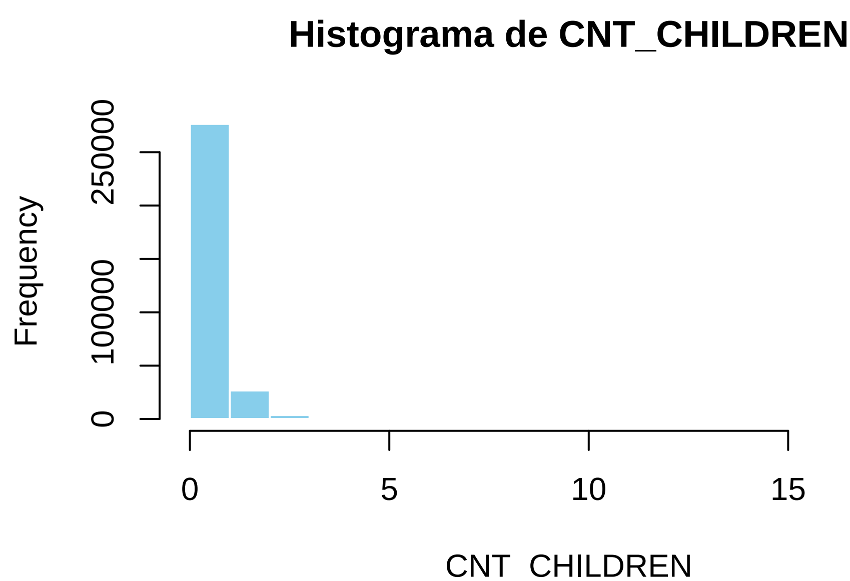


```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.057265, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable SK_ID_CURR NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: TARGET
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
0.00000 0.00000 0.00000 0.08073 0.00000 1.00000

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



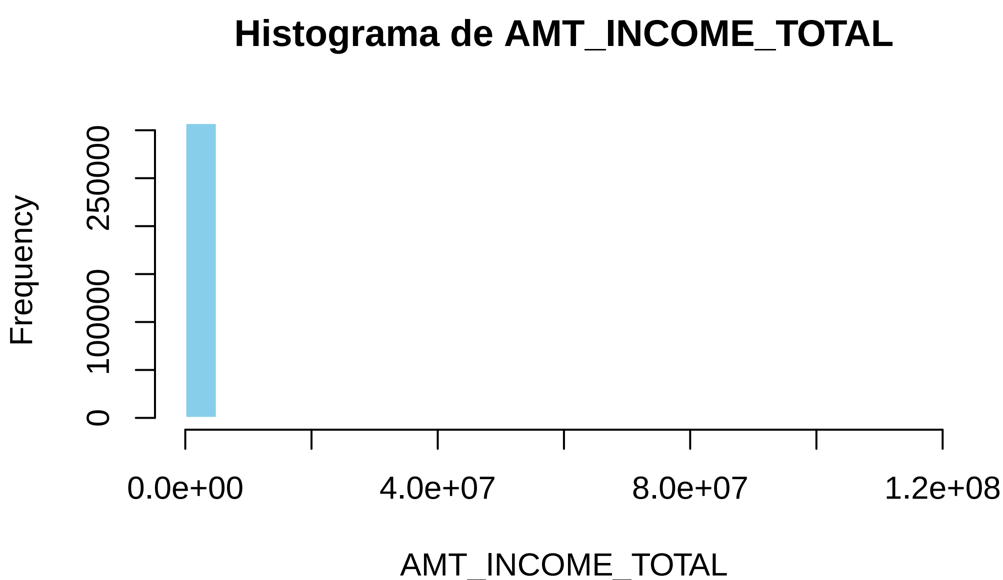

```
Test de Kolmogorov-Smirnov para la normalidad:
```

```
    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.53579, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable TARGET NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: CNT_CHILDREN
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0000  0.0000  0.0000  0.4171  1.0000 19.0000

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



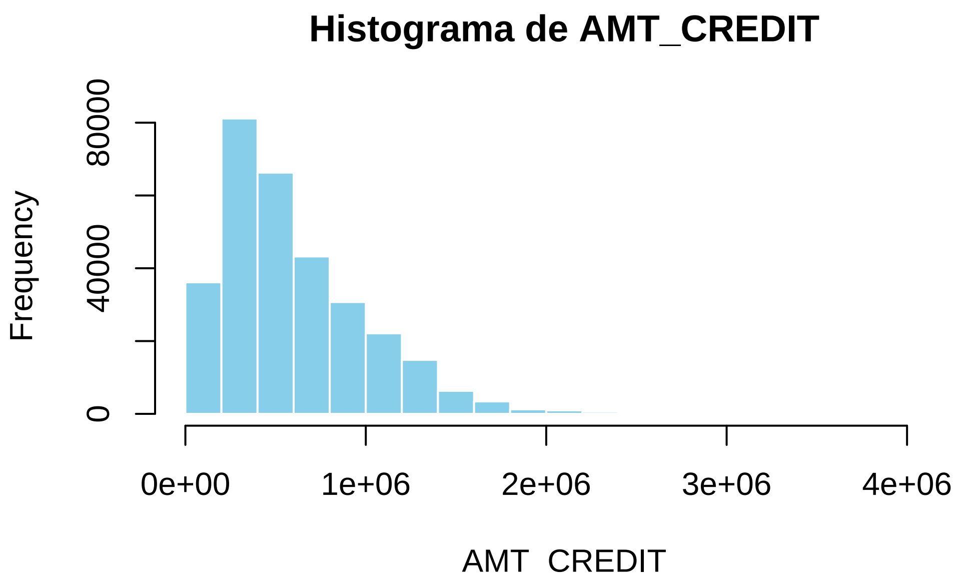

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test
```

```
data:  datos[[col]]
D = 0.41858, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable CNT_CHILDREN NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: AMT_INCOME_TOTAL
-------------------------------------------------
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
    25650    112500    147150    168798    202500 117000000

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



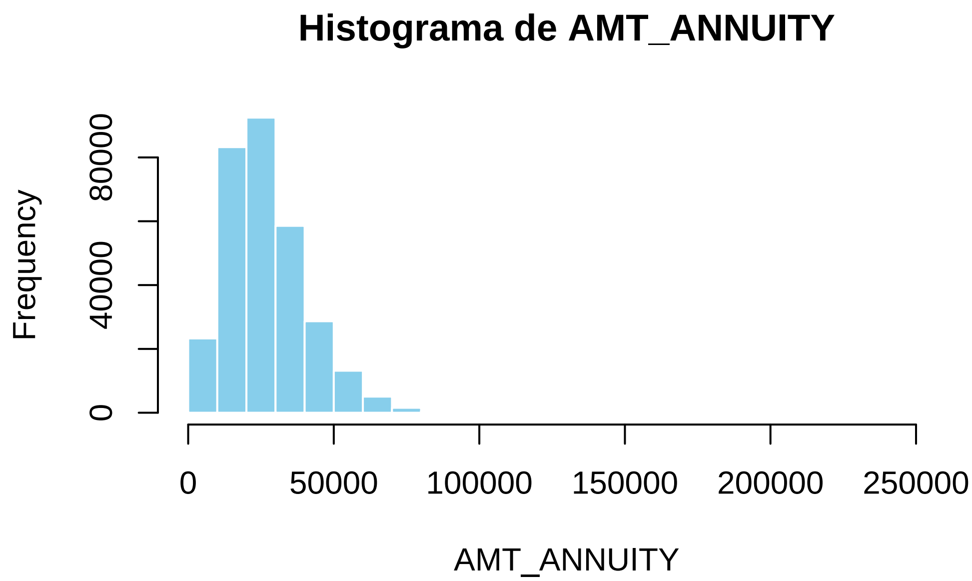

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.30171, p-value < 2.2e-16
```

```
alternative hypothesis: two-sided

 La variable AMT_INCOME_TOTAL NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: AMT_CREDIT
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  45000  270000  513531  599026  808650 4050000 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



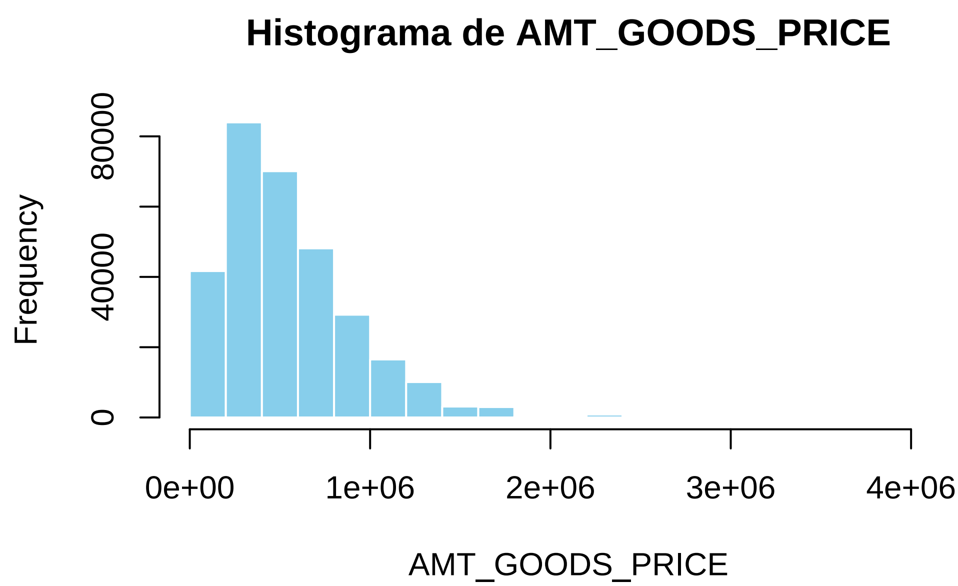

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.11015, p-value < 2.2e-16
alternative hypothesis: two-sided
```

```
 La variable AMT_CREDIT NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: AMT_ANNUITY
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   1616   16524   24903   27109   34596  258026      12 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



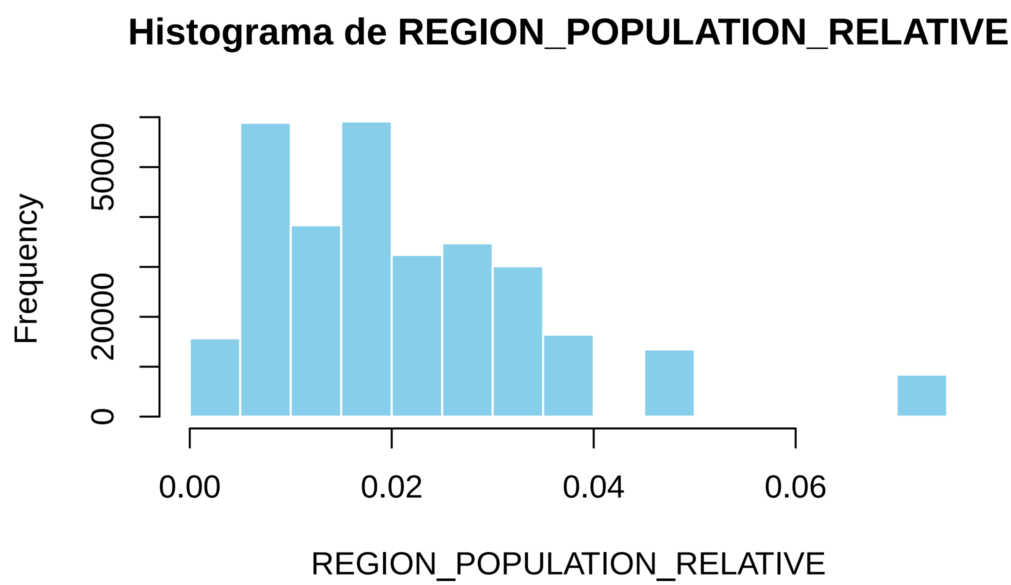

```
Test de Kolmogorov-Smirnov para la normalidad:

	Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.0789, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable AMT_ANNUITY NO sigue una distribución normal (p < 0 )
```

```
--------------------------------------------------
Distribución de la variable: AMT_GOODS_PRICE
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
  40500  238500  450000  538396  679500 4050000     278 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



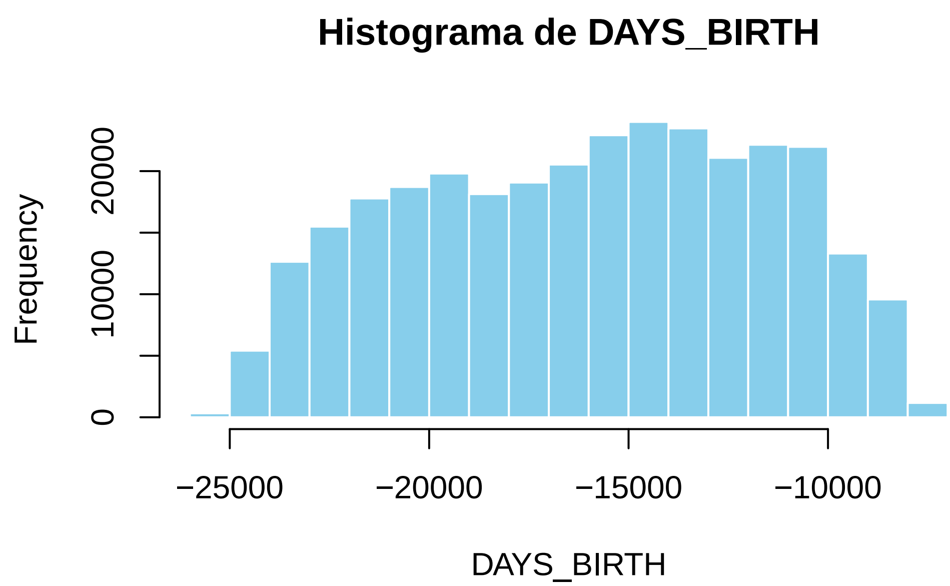

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.14269, p-value < 2.2e-16
alternative hypothesis: two-sided

  La variable AMT_GOODS_PRICE NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: REGION_POPULATION_RELATIVE
```

```
------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
0.00029 0.01001 0.01885 0.02087 0.02866 0.07251 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```

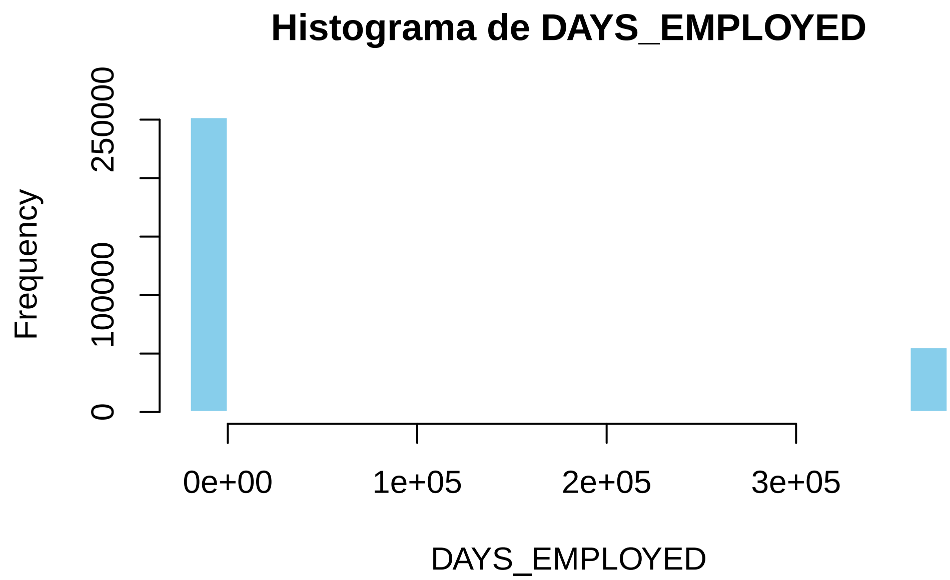



```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.11345, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable REGION_POPULATION_RELATIVE NO sigue una distribución normal (p < 0 )

------------------------------------------------
Distribución de la variable: DAYS_BIRTH
------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
 -25229  -19682  -15750  -16037  -12413   -7489 
```

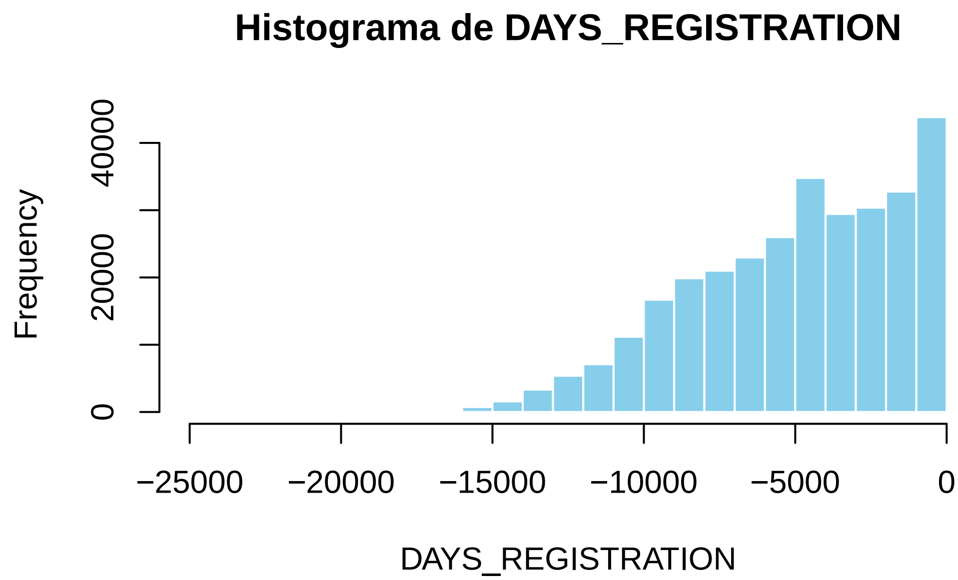
```
Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



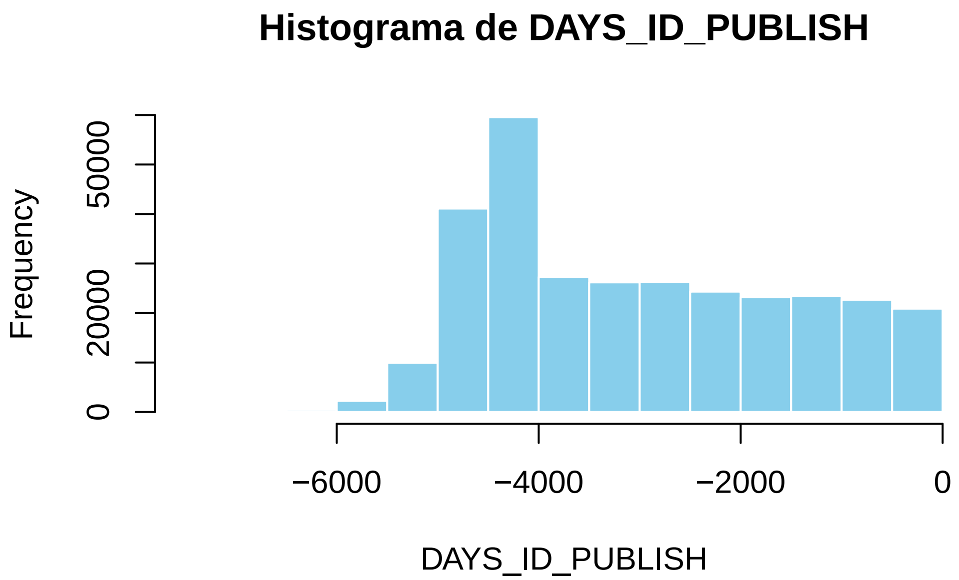

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.048582, p-value < 2.2e-16
alternative hypothesis: two-sided

  La variable DAYS_BIRTH NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: DAYS_EMPLOYED
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
 -17912   -2760   -1213   63815    -289  365243 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



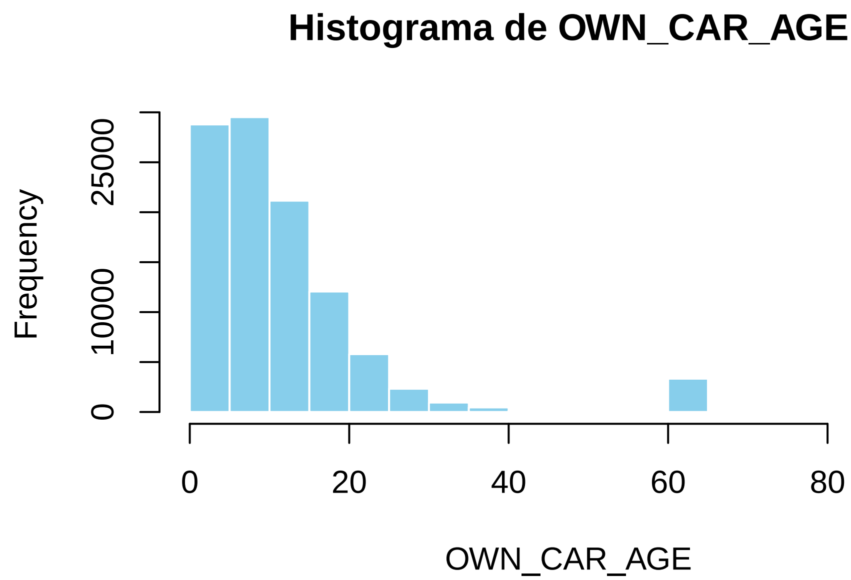

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.49419, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable DAYS_EMPLOYED NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: DAYS_REGISTRATION
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 -24672   -7480   -4504   -4986   -2010       0

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



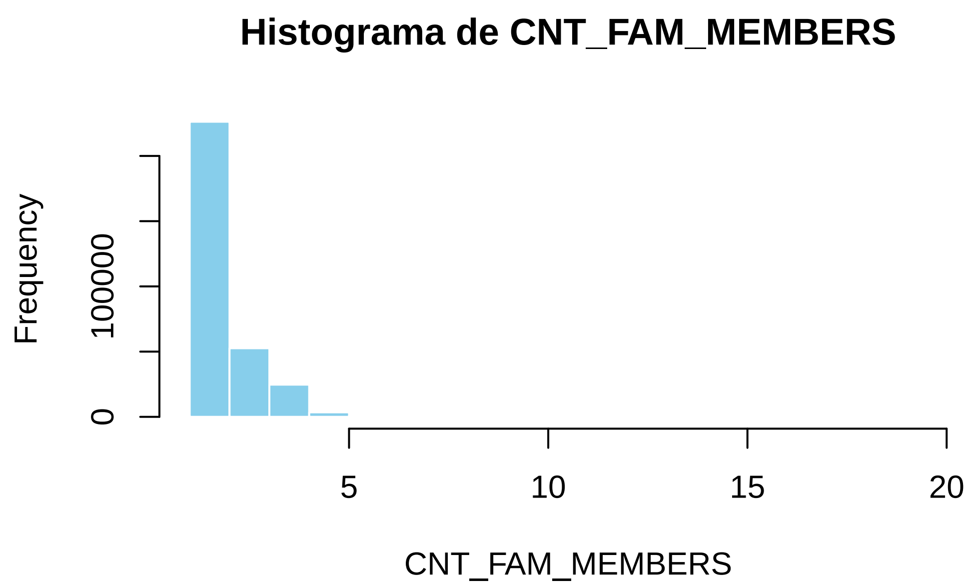

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.078483, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable DAYS_REGISTRATION NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: DAYS_ID_PUBLISH
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  -7197   -4299   -3254   -2994   -1720       0

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



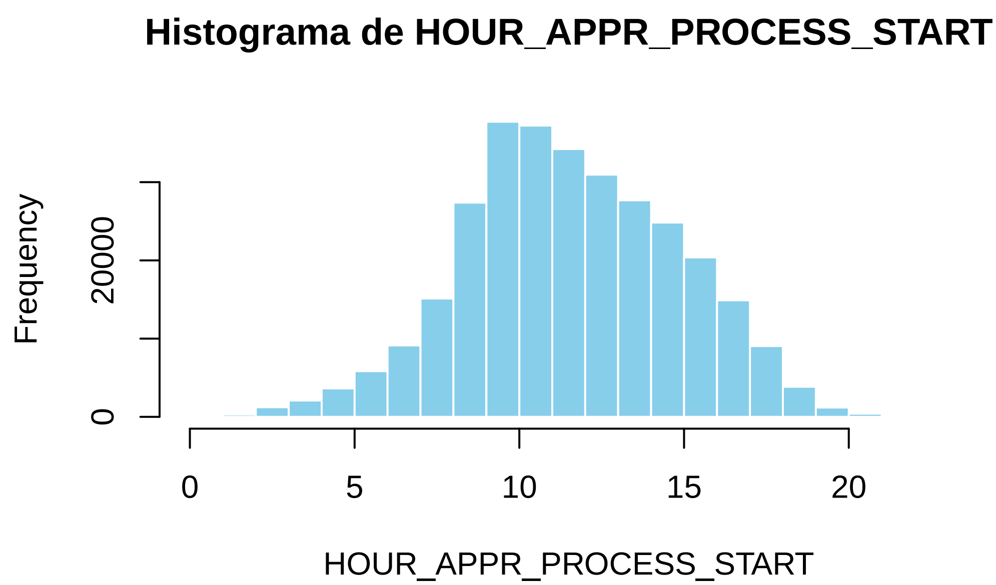

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.12221, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable DAYS_ID_PUBLISH NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: OWN_CAR_AGE
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   0.00    5.00    9.00   12.06   15.00   91.00  202929 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



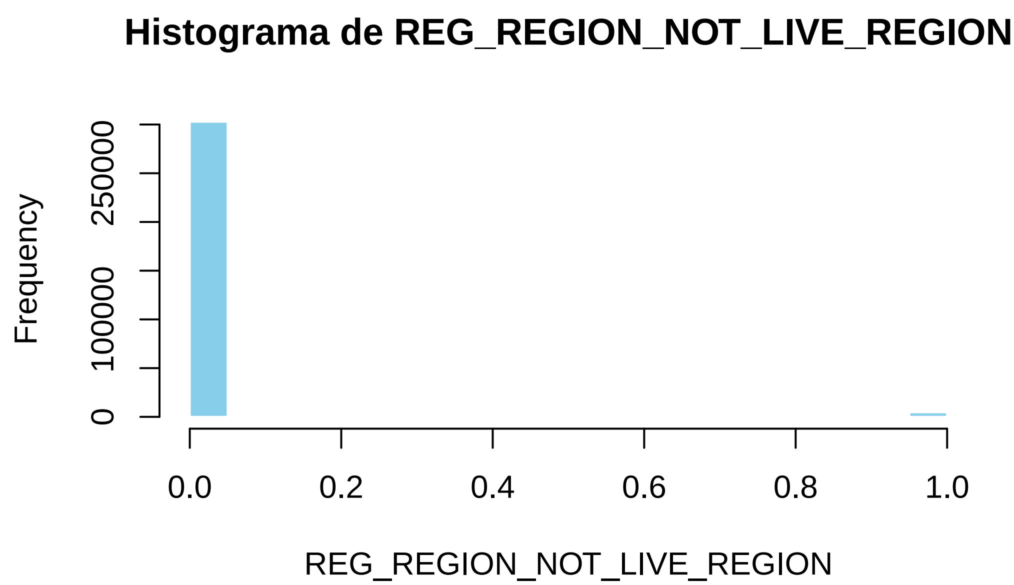

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.16271, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable OWN_CAR_AGE NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: CNT_FAM_MEMBERS
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
  1.000   2.000   2.000   2.153   3.000  20.000       2

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



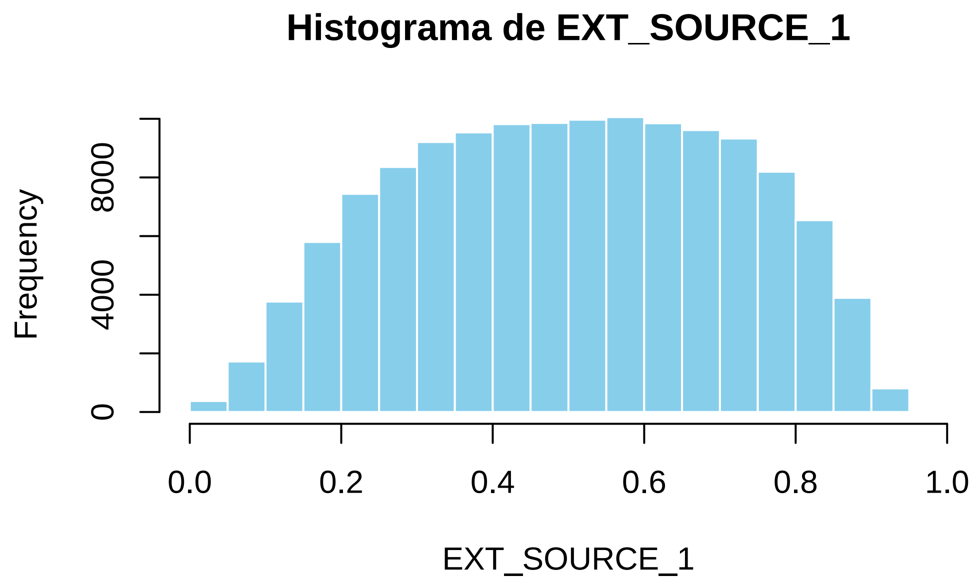

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.30217, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable CNT_FAM_MEMBERS NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: HOUR_APPR_PROCESS_START
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00   10.00   12.00   12.06   14.00   23.00

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



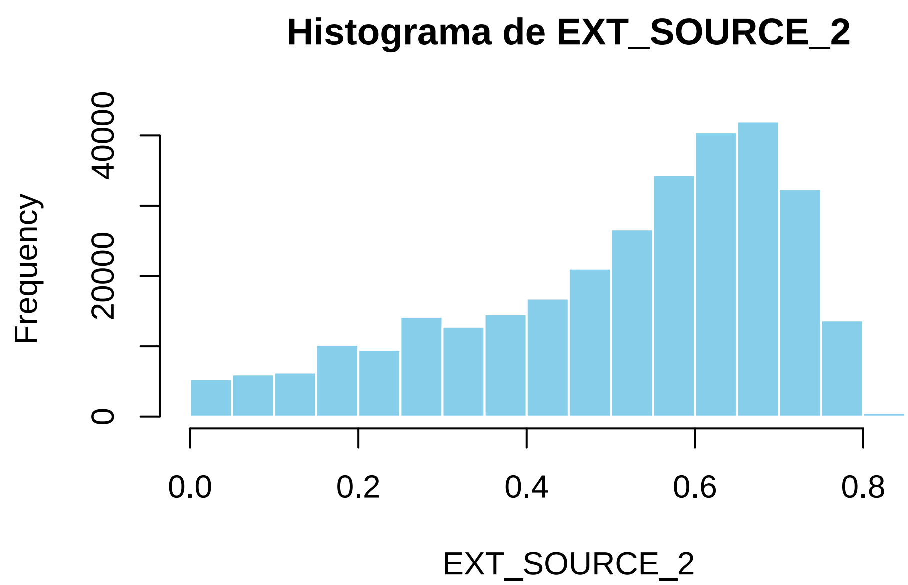

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.08234, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable HOUR_APPR_PROCESS_START NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: REG_REGION_NOT_LIVE_REGION
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
0.00000 0.00000 0.00000 0.01514 0.00000 1.00000

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



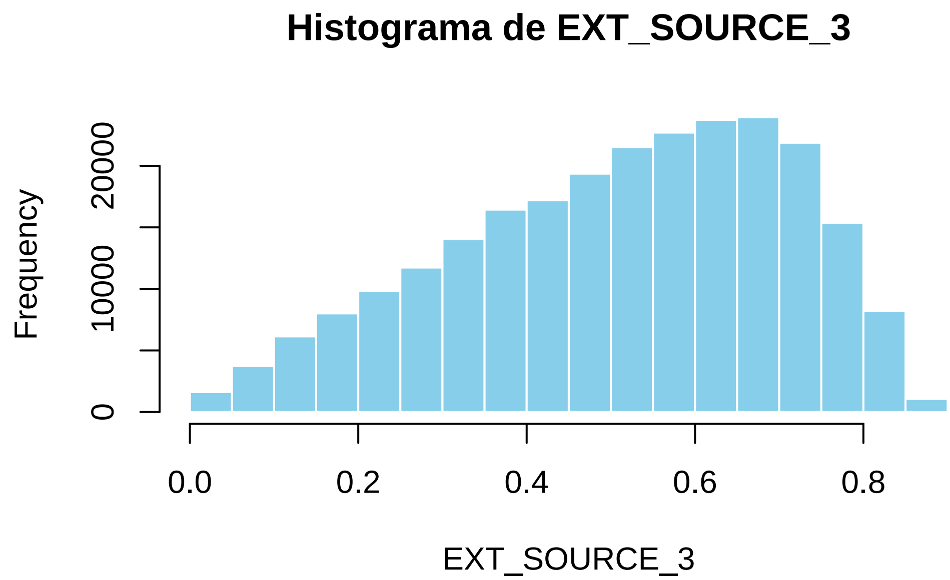

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.5342, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable REG_REGION_NOT_LIVE_REGION NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: EXT_SOURCE_1
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.01    0.33    0.51    0.50    0.68    0.96  173378

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



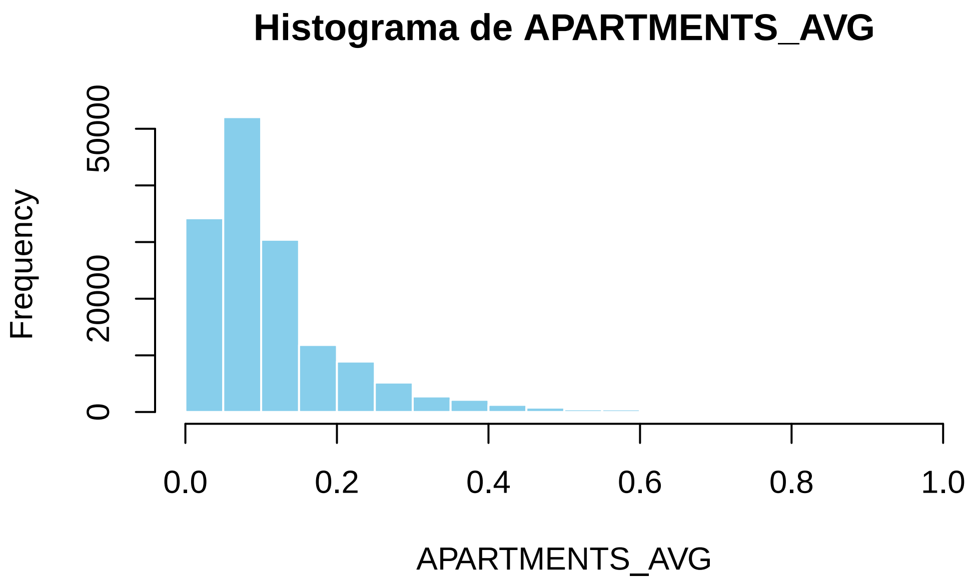

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.044677, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable EXT_SOURCE_1 NO sigue una distribución normal (p < 5.58411e-233 )

-------------------------------------------------
Distribución de la variable: EXT_SOURCE_2
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
 0.0000  0.3925  0.5660  0.5144  0.6636  0.8550     660

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



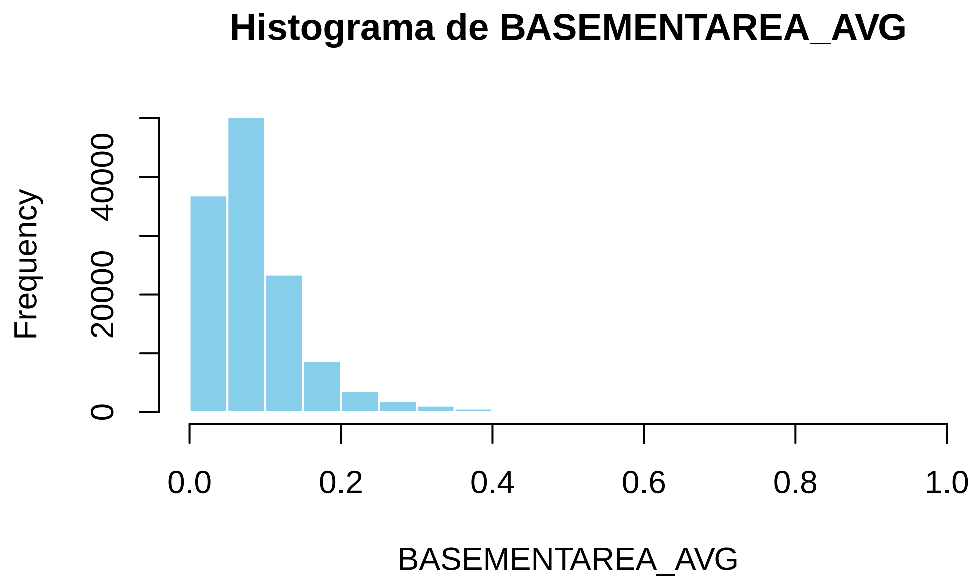

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.10691, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable EXT_SOURCE_2 NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: EXT_SOURCE_3
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   0.00    0.37    0.54    0.51    0.67    0.90   60965 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



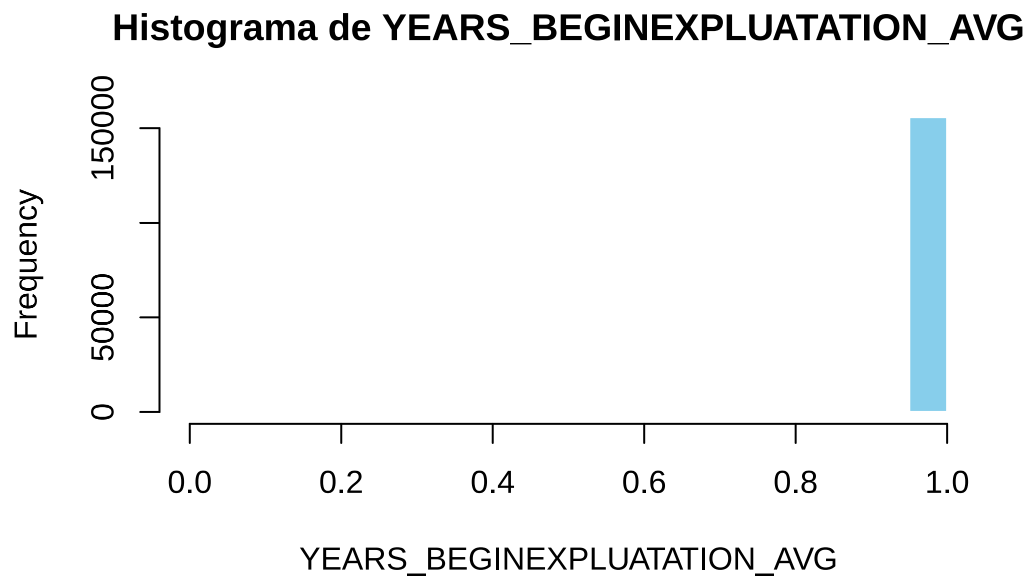

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.061755, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable EXT_SOURCE_3 NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: APARTMENTS_AVG
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   0.00    0.06    0.09    0.12    0.15    1.00  156061 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



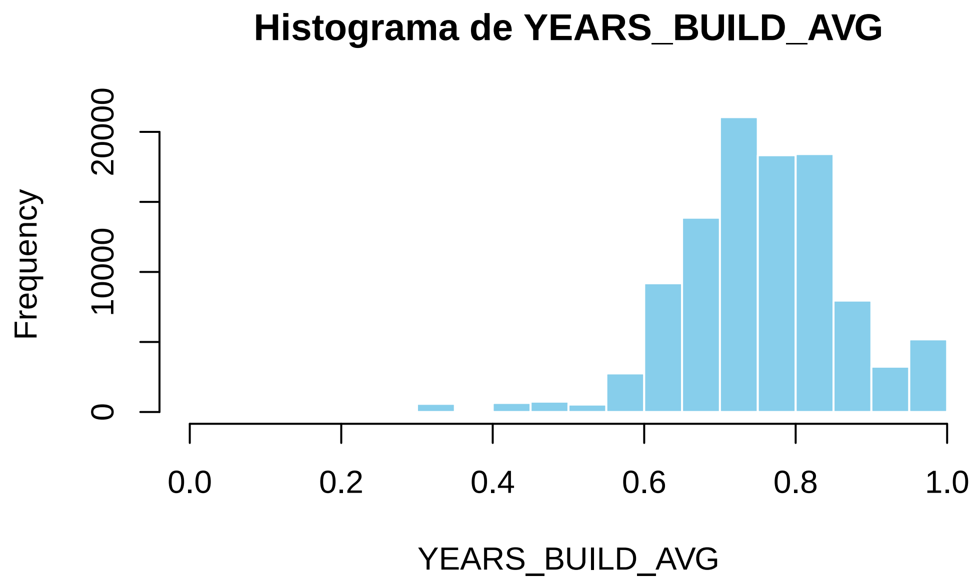

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.1668, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable APARTMENTS_AVG NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: BASEMENTAREA_AVG
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.04    0.08    0.09    0.11    1.00  179943

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



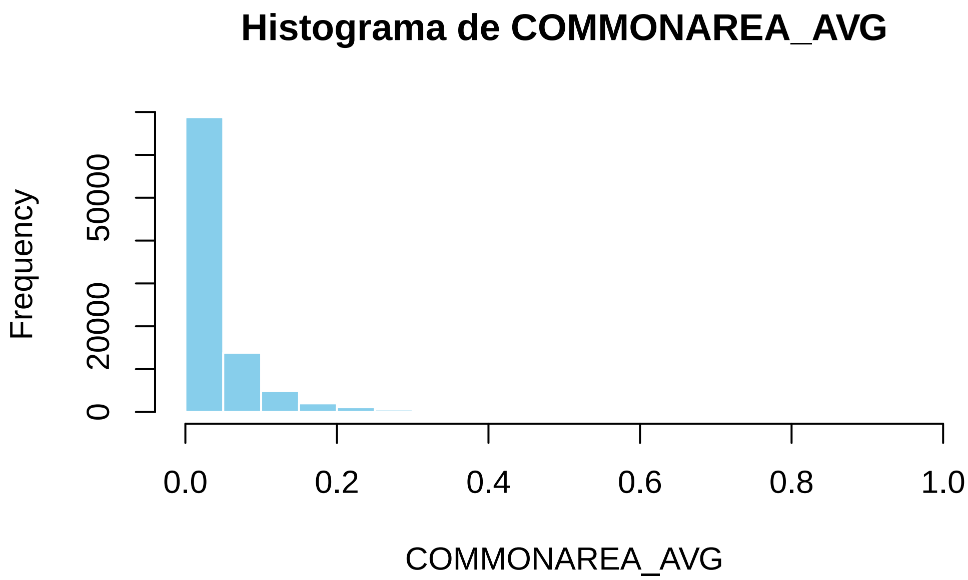

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.14167, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable BASEMENTAREA_AVG NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: YEARS_BEGINEXPLUATATION_AVG
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   0.00    0.98    0.98    0.98    0.99    1.00  150007 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



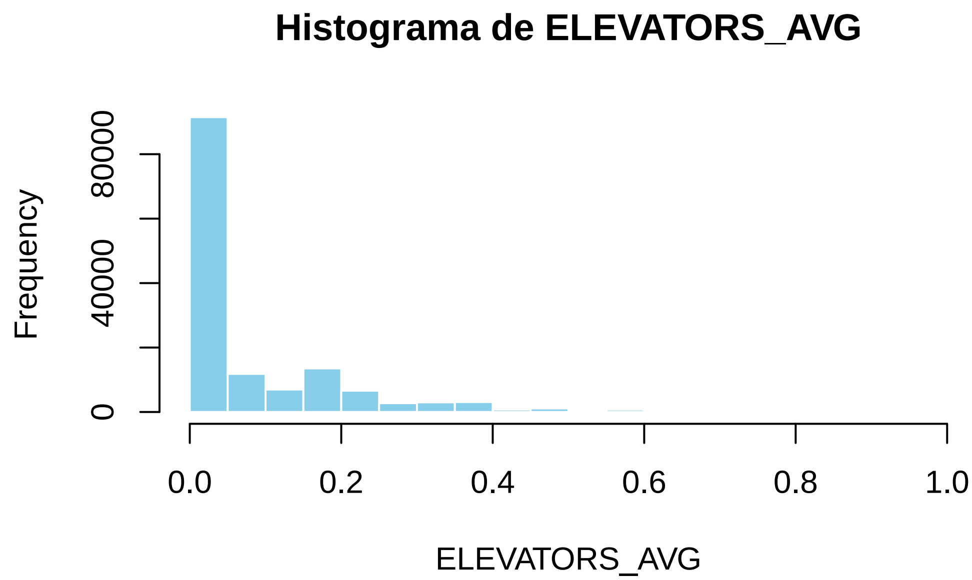

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.39064, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable YEARS_BEGINEXPLUATATION_AVG NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: YEARS_BUILD_AVG
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.69    0.76    0.75    0.82    1.00  204488

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



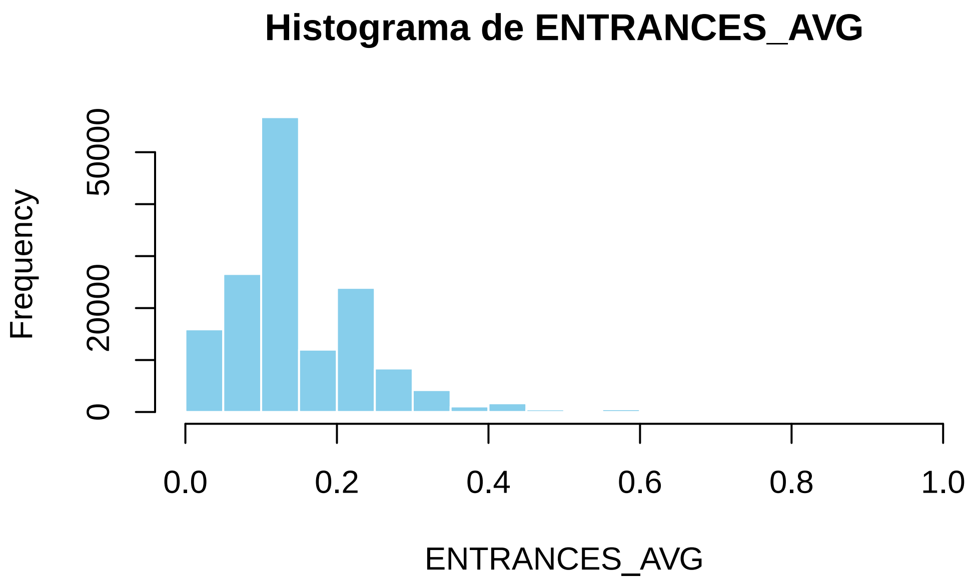

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.051642, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable YEARS_BUILD_AVG NO sigue una distribución normal (p < 4.560853e-239 )

--------------------------------------------------
Distribución de la variable: COMMONAREA_AVG
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.01    0.02    0.04    0.05    1.00  214865

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



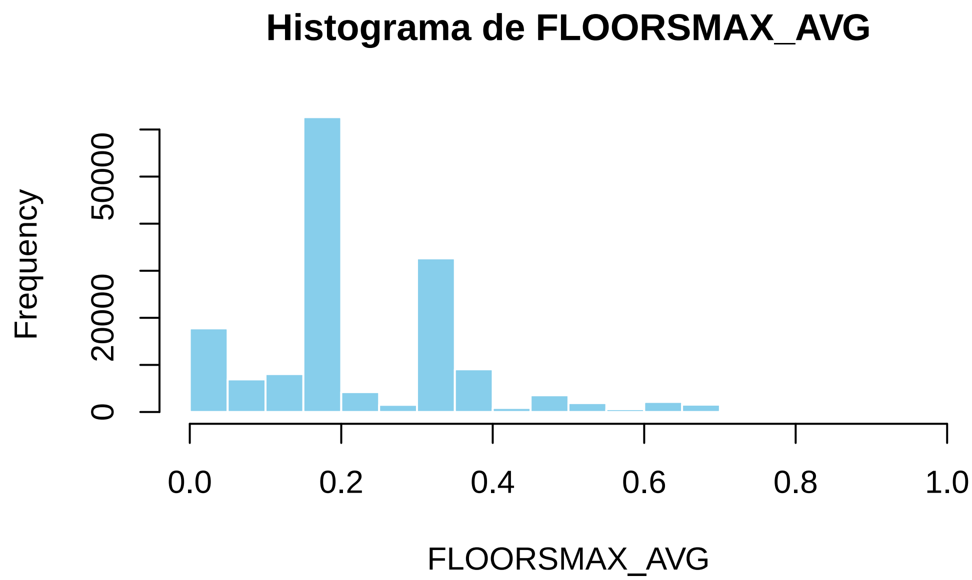

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.27866, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable COMMONAREA_AVG NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: ELEVATORS_AVG
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.00    0.00    0.08    0.12    1.00  163891

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



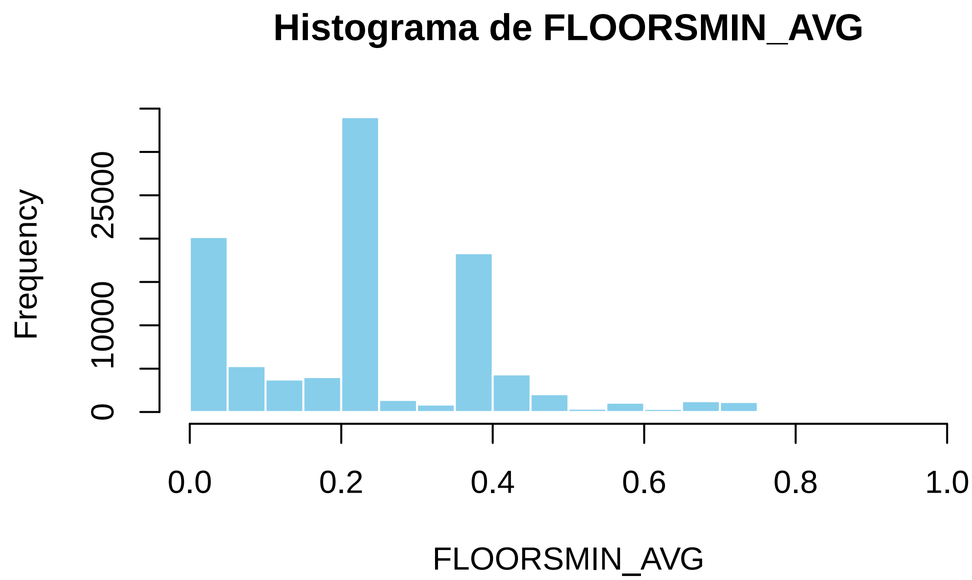

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.3181, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable ELEVATORS_AVG NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: ENTRANCES_AVG
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.07    0.14    0.15    0.21    1.00  154828

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



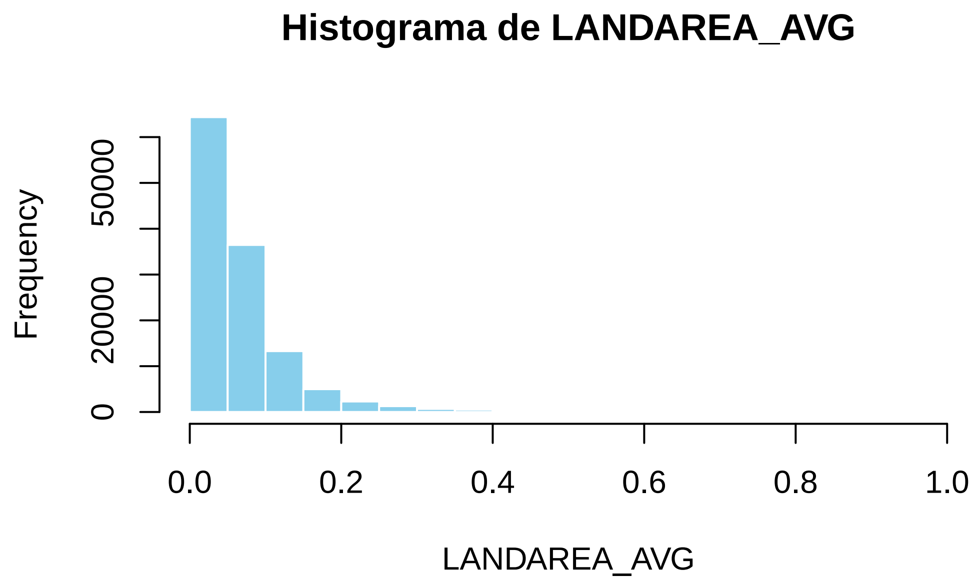

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.19338, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable ENTRANCES_AVG NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: FLOORSMAX_AVG
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.17    0.17    0.23    0.33    1.00  153020

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



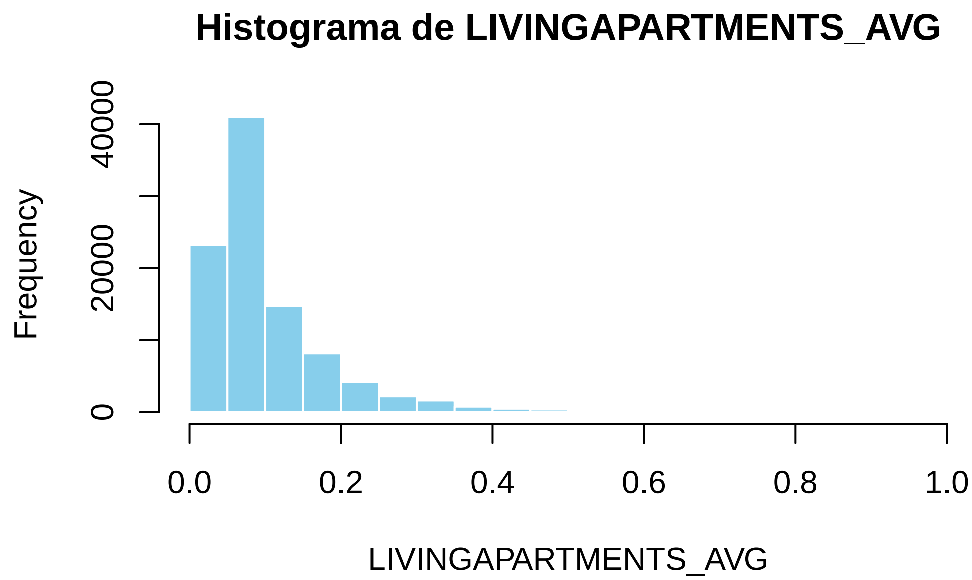

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.27317, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable FLOORSMAX_AVG NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: FLOORSMIN_AVG
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.08    0.21    0.23    0.38    1.00  208642

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



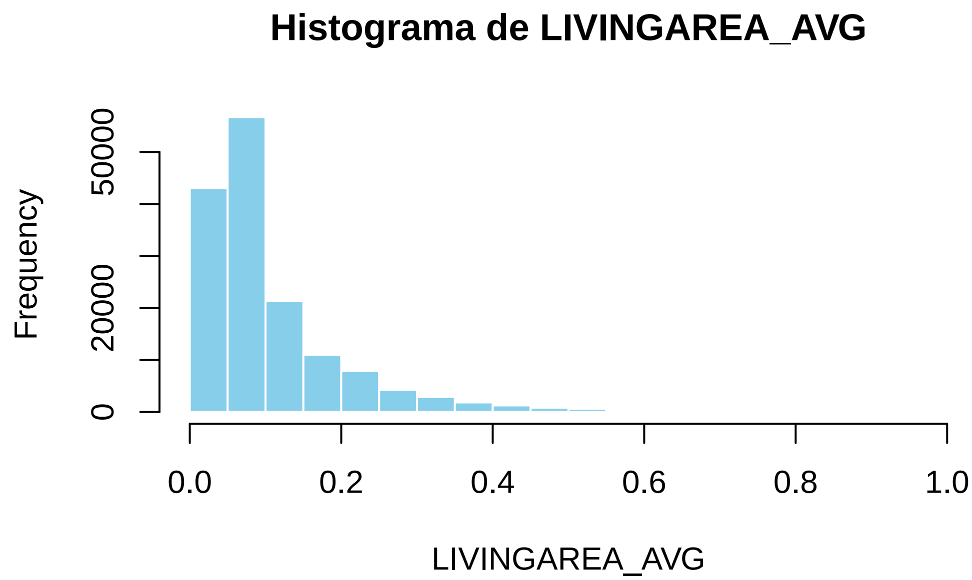

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.22705, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable FLOORSMIN_AVG NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: LANDAREA_AVG
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.02    0.05    0.07    0.09    1.00  182590

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



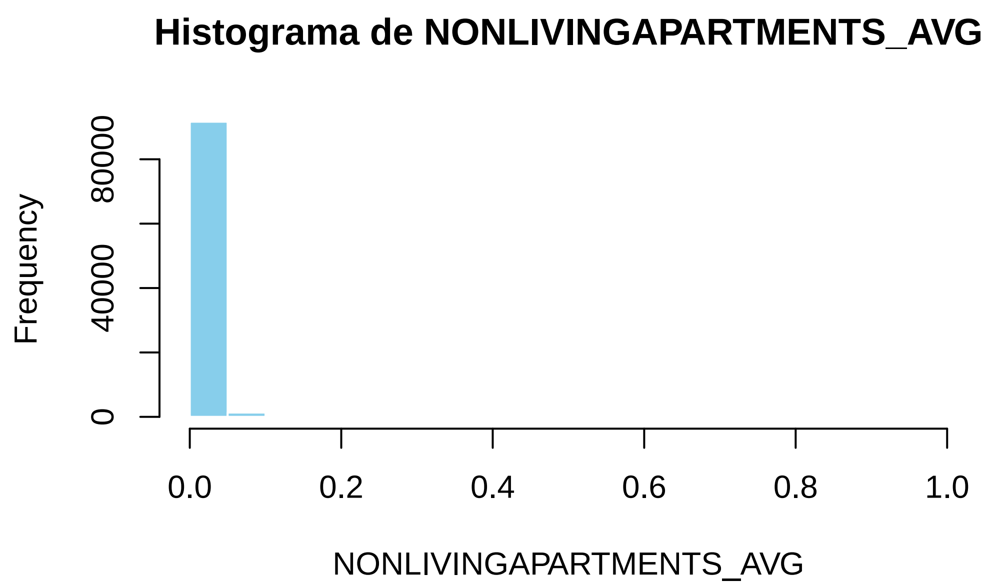

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.20694, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable LANDAREA_AVG NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: LIVINGAPARTMENTS_AVG
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.05    0.08    0.10    0.12    1.00  210199

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



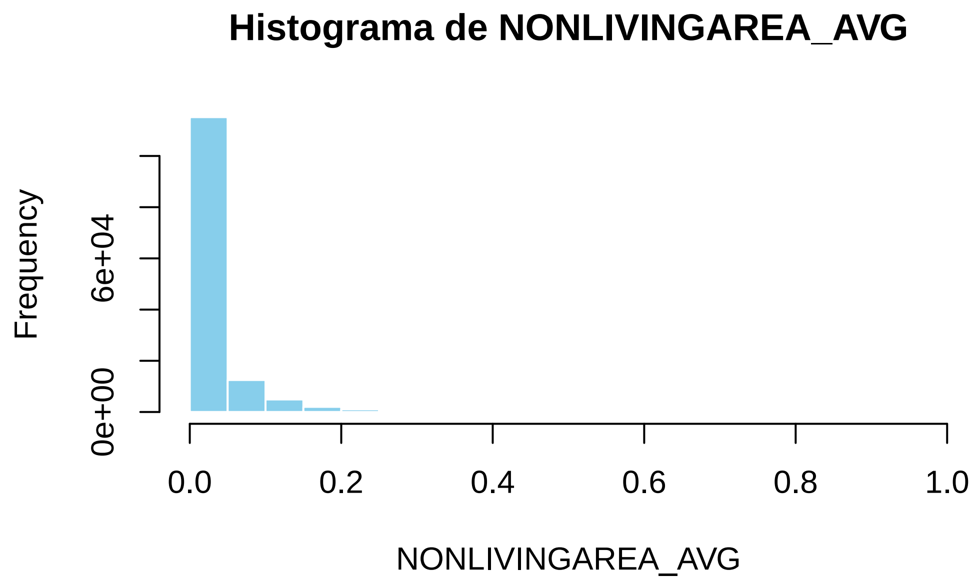

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.17467, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable LIVINGAPARTMENTS_AVG NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: LIVINGAREA_AVG
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.05    0.07    0.11    0.13    1.00  154350

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



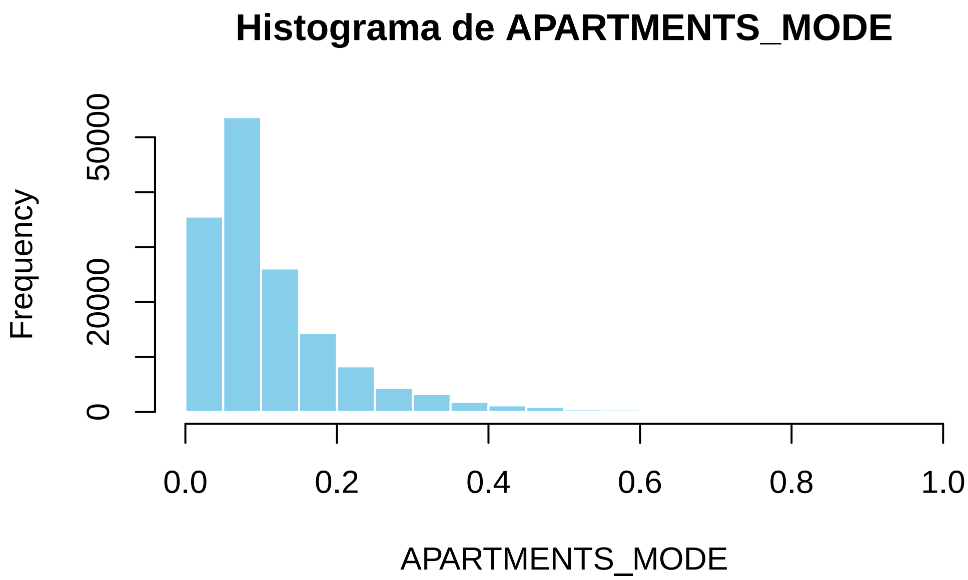

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.18232, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable LIVINGAREA_AVG NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: NONLIVINGAPARTMENTS_AVG
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   0.00    0.00    0.00    0.01    0.00    1.00  213514 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



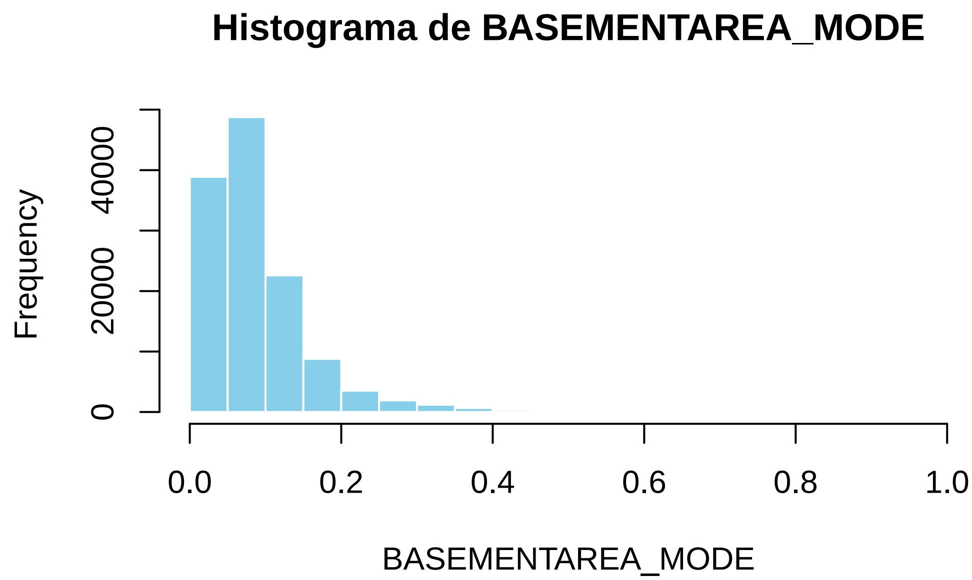

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.42679, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable NONLIVINGAPARTMENTS_AVG NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: NONLIVINGAREA_AVG
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   0.00    0.00    0.00    0.03    0.03    1.00  169682 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



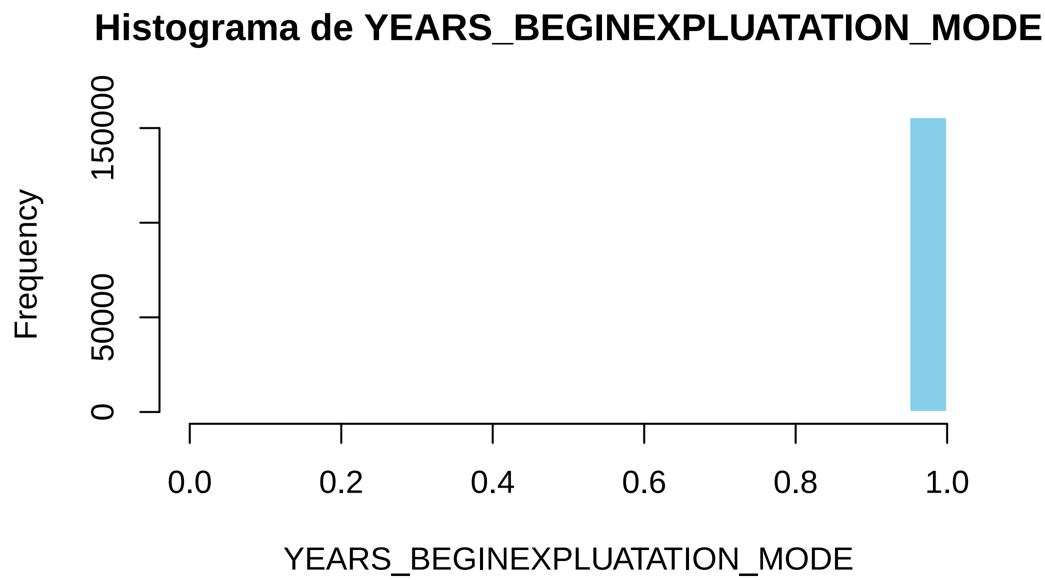

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.34168, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable NONLIVINGAREA_AVG NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: APARTMENTS_MODE
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.05    0.08    0.11    0.14    1.00  156061

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



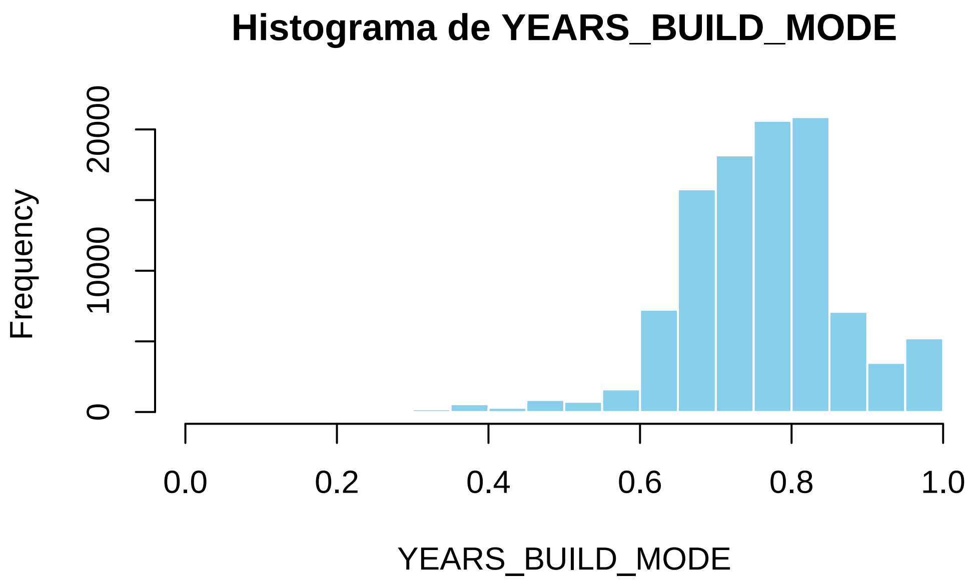

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.17123, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable APARTMENTS_MODE NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: BASEMENTAREA_MODE
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.04    0.07    0.09    0.11    1.00  179943

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



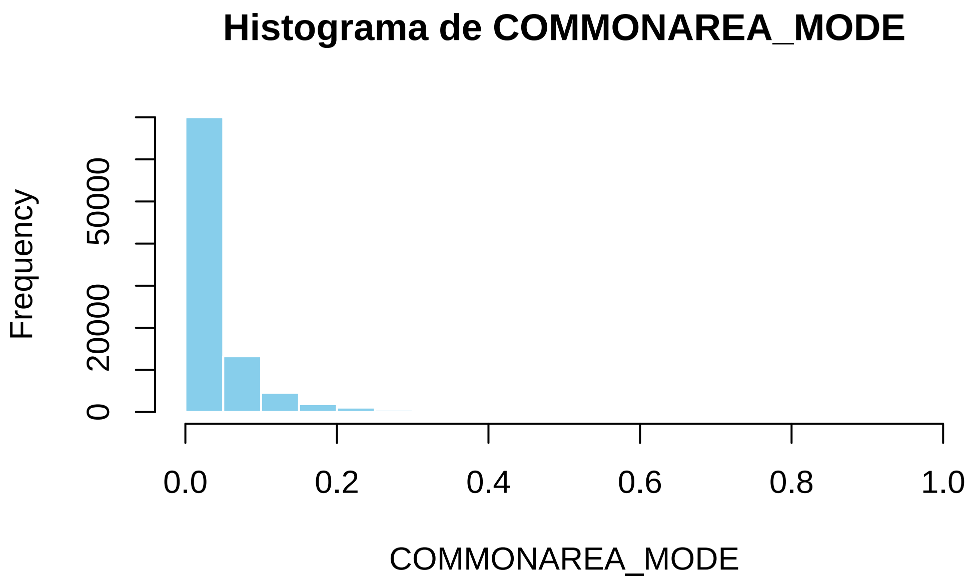

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.14955, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable BASEMENTAREA_MODE NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: YEARS_BEGINEXPLUATATION_MODE
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.98    0.98    0.98    0.99    1.00  150007

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



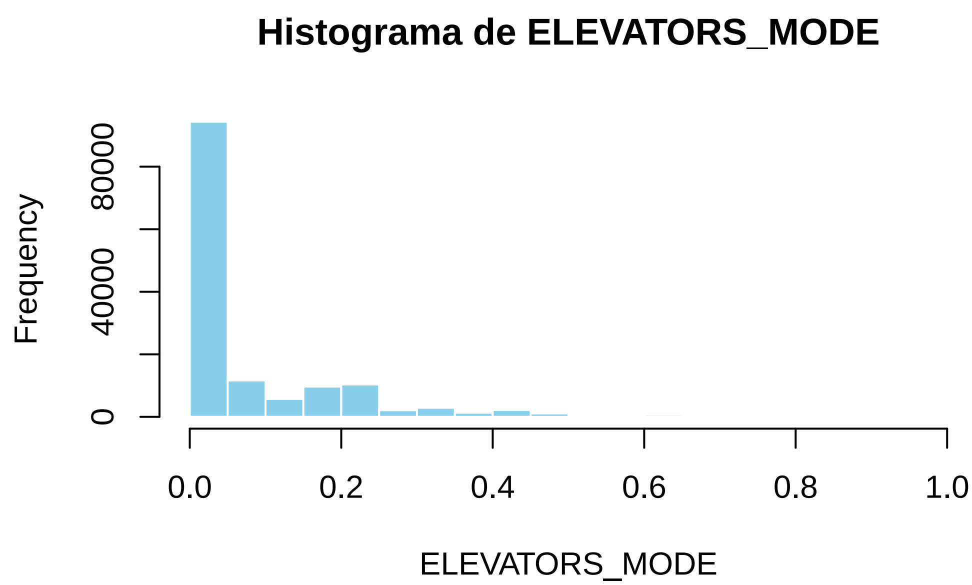

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.39761, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable YEARS_BEGINEXPLUATATION_MODE NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: YEARS_BUILD_MODE
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.70    0.76    0.76    0.82    1.00  204488

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



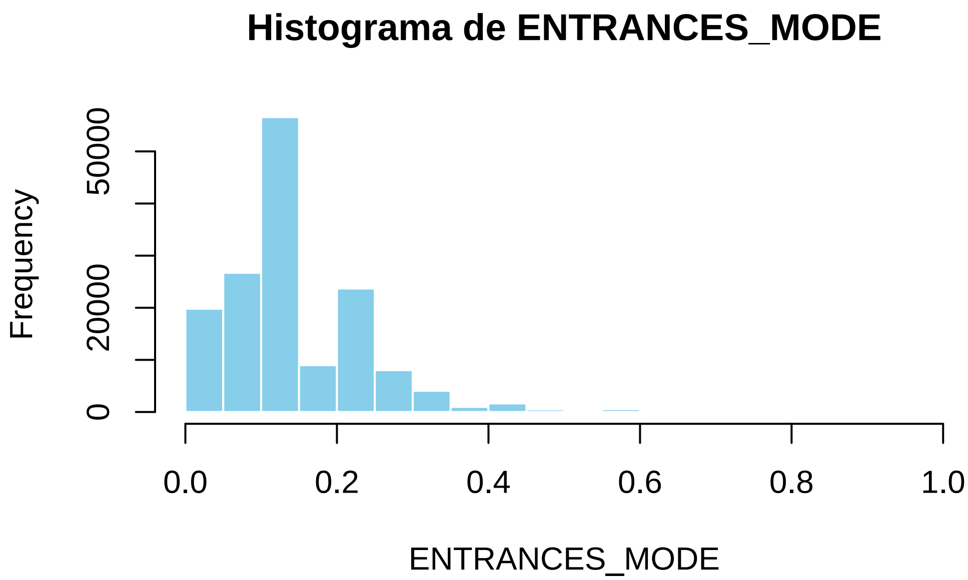

```
Test de Kolmogorov-Smirnov para la normalidad:

	Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.054756, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable YEARS_BUILD_MODE NO sigue una distribución normal (p < 1.021391e-268 )

--------------------------------------------------
Distribución de la variable: COMMONAREA_MODE
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   0.00    0.01    0.02    0.04    0.05    1.00  214865 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



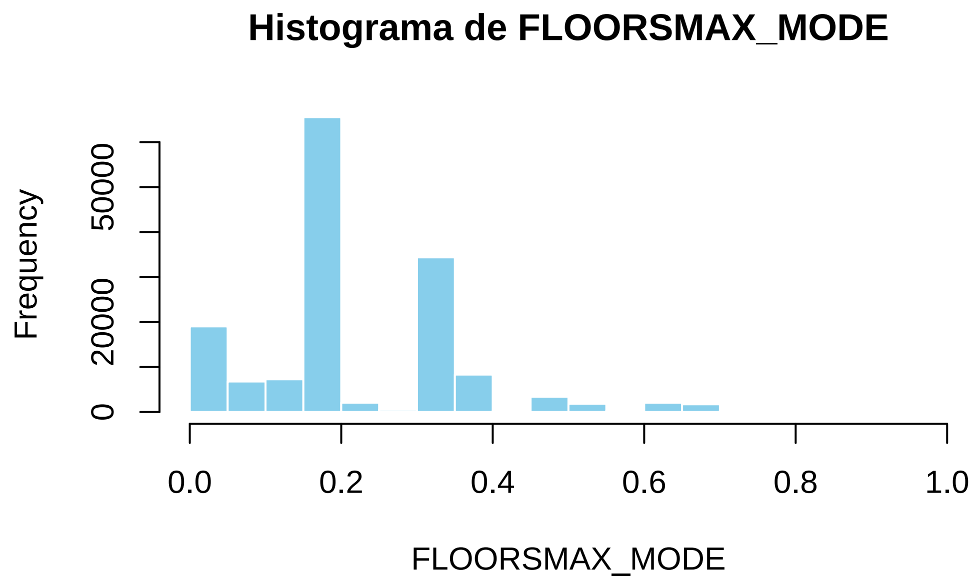

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.28379, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable COMMONAREA_MODE NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: ELEVATORS_MODE
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   0.00    0.00    0.00    0.07    0.12    1.00  163891 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



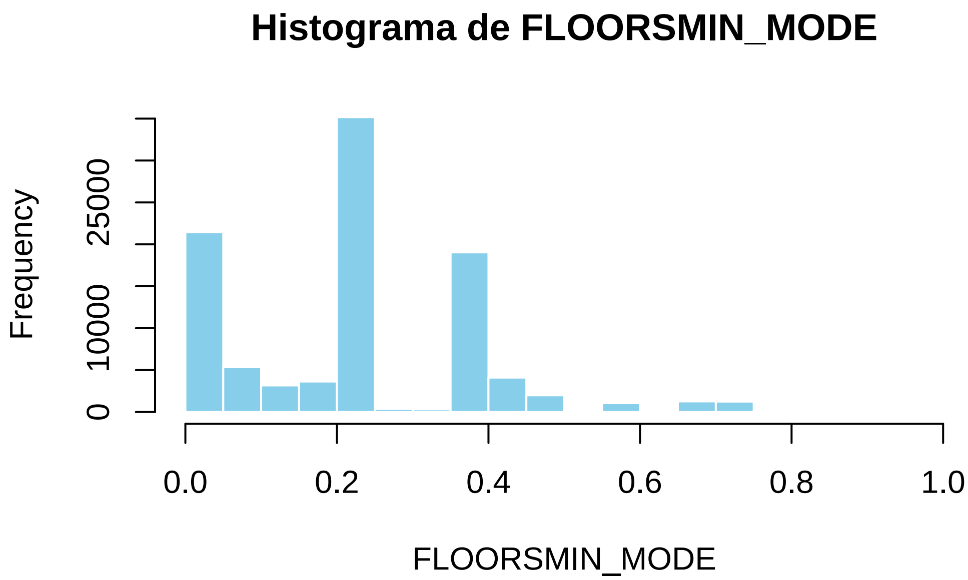

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.33652, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable ELEVATORS_MODE NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: ENTRANCES_MODE
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.07    0.14    0.15    0.21    1.00  154828

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



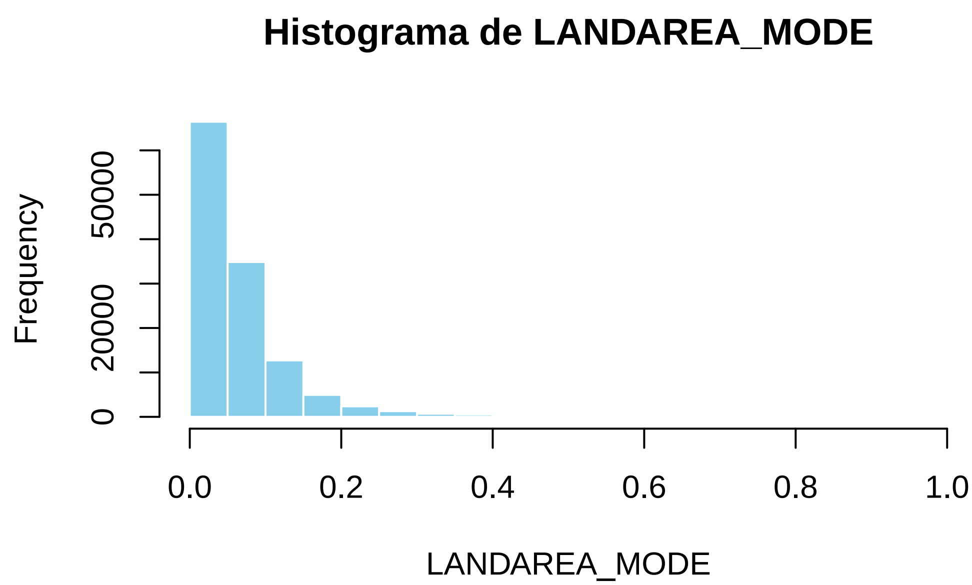

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.204, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable ENTRANCES_MODE NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: FLOORSMAX_MODE
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.17    0.17    0.22    0.33    1.00  153020

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



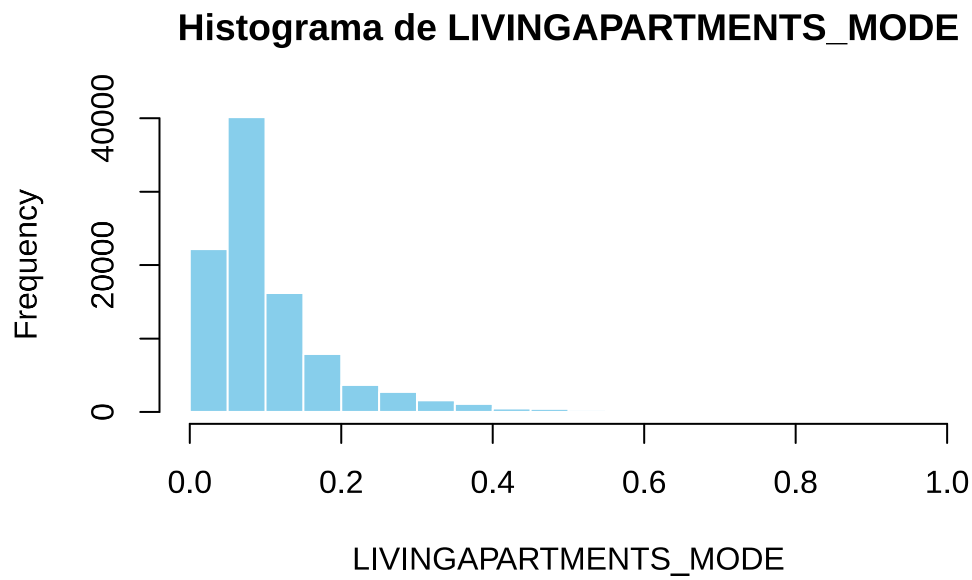

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.28906, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable FLOORSMAX_MODE NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: FLOORSMIN_MODE
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.08    0.21    0.23    0.38    1.00  208642

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



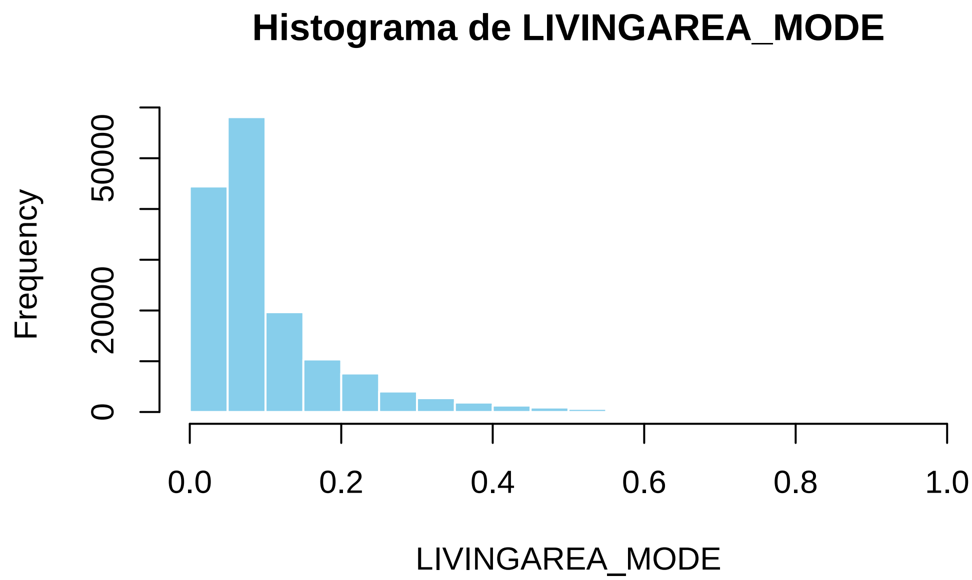

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.23649, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable FLOORSMIN_MODE NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: LANDAREA_MODE
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.02    0.05    0.06    0.08    1.00  182590

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



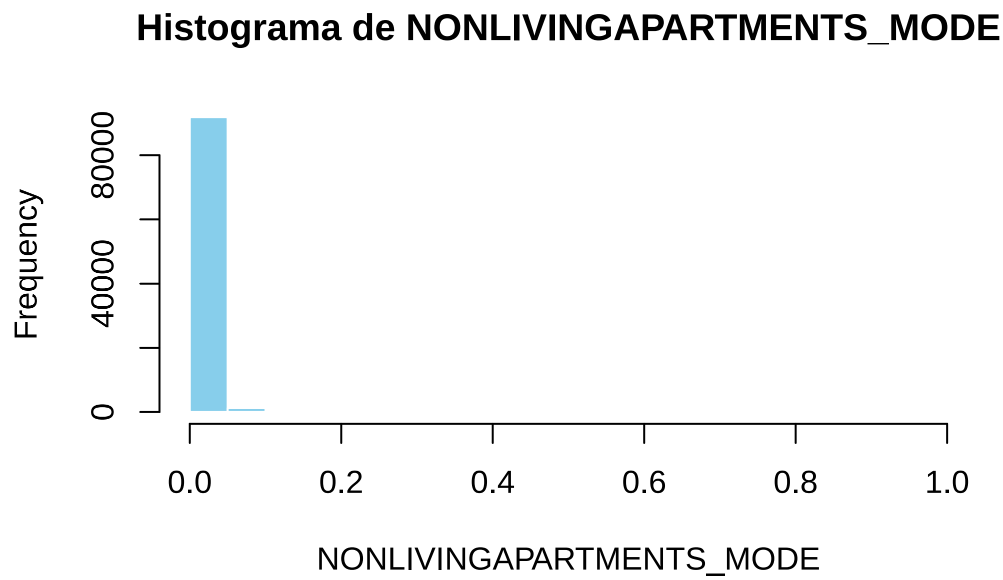

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.21343, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable LANDAREA_MODE NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: LIVINGAPARTMENTS_MODE
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   0.00    0.05    0.08    0.11    0.13    1.00  210199 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



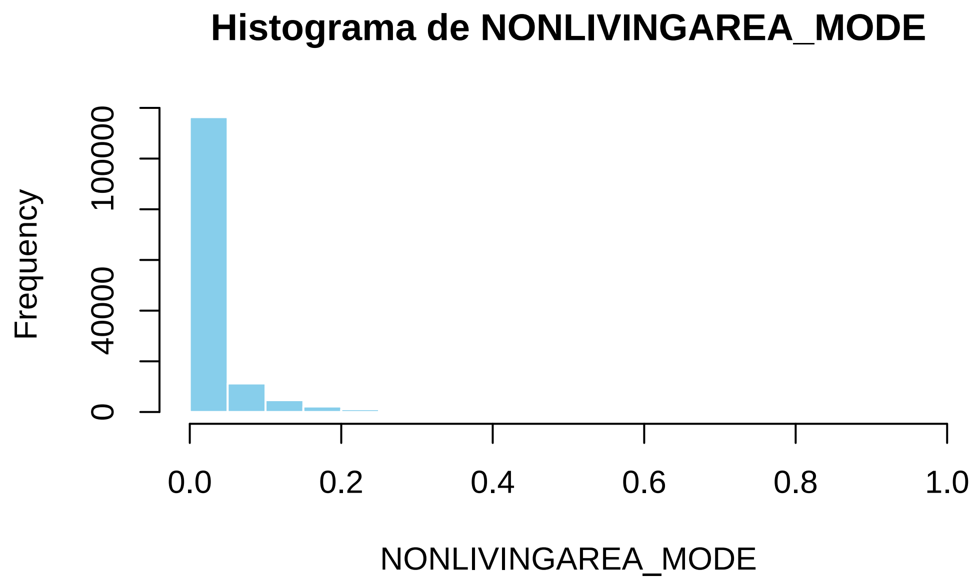

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.17894, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable LIVINGAPARTMENTS_MODE NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: LIVINGAREA_MODE
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.04    0.07    0.11    0.13    1.00  154350

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



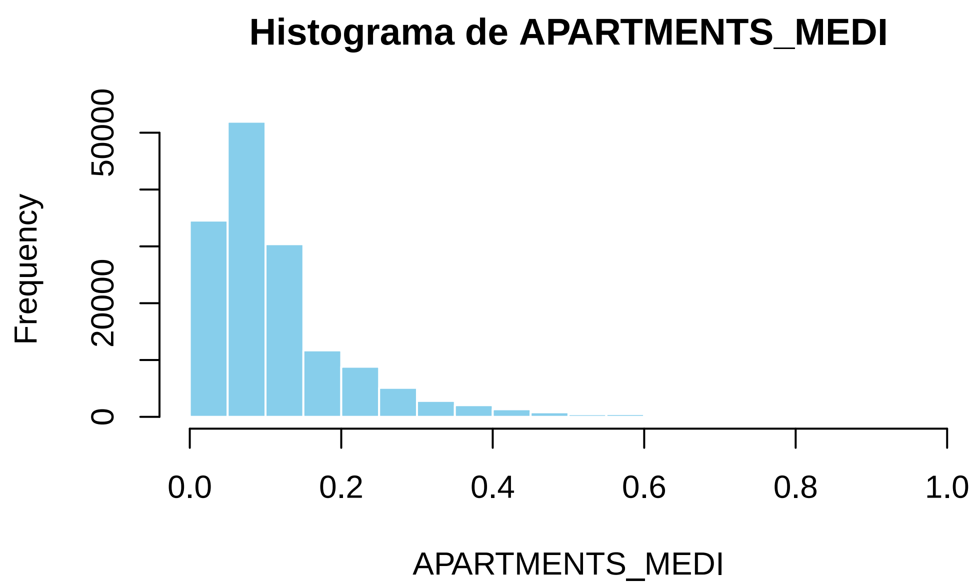

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.19075, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable LIVINGAREA_MODE NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: NONLIVINGAPARTMENTS_MODE
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   0.00    0.00    0.00    0.01    0.00    1.00  213514 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



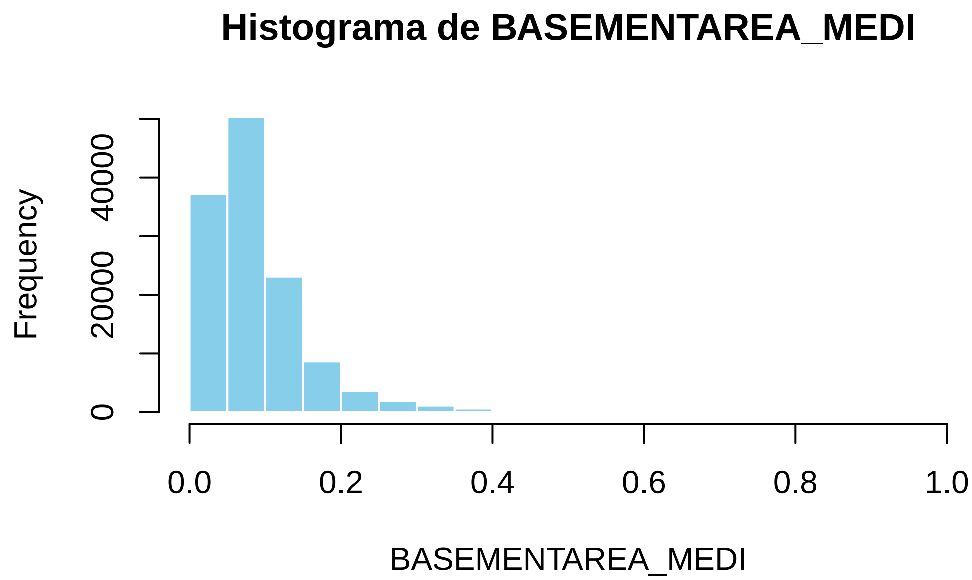

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.43073, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable NONLIVINGAPARTMENTS_MODE NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: NONLIVINGAREA_MODE
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.00    0.00    0.03    0.02    1.00  169682

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



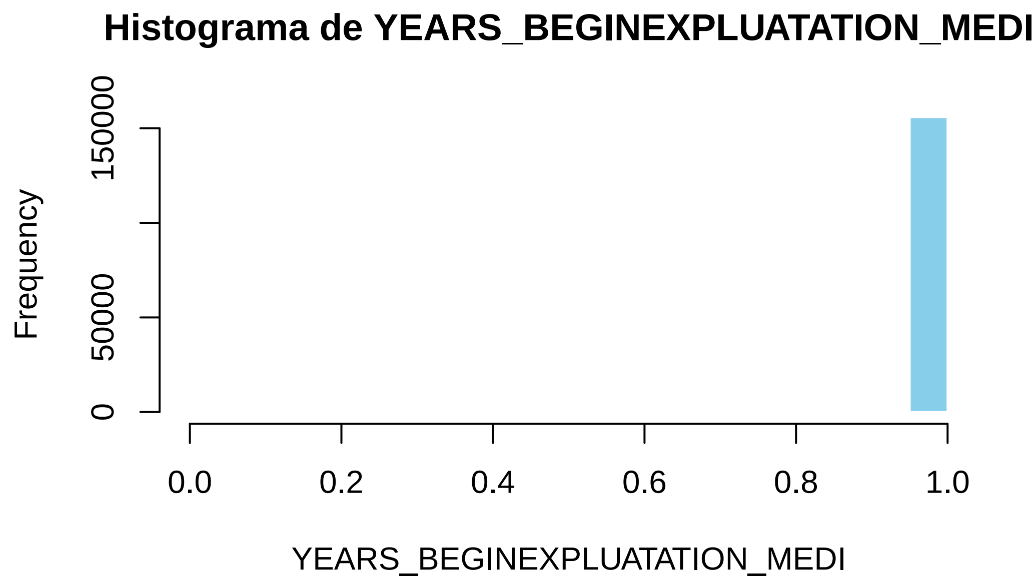

```
Test de Kolmogorov-Smirnov para la normalidad:

	Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.35025, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable NONLIVINGAREA_MODE NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: APARTMENTS_MEDI
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   0.00    0.06    0.09    0.12    0.15    1.00  156061 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



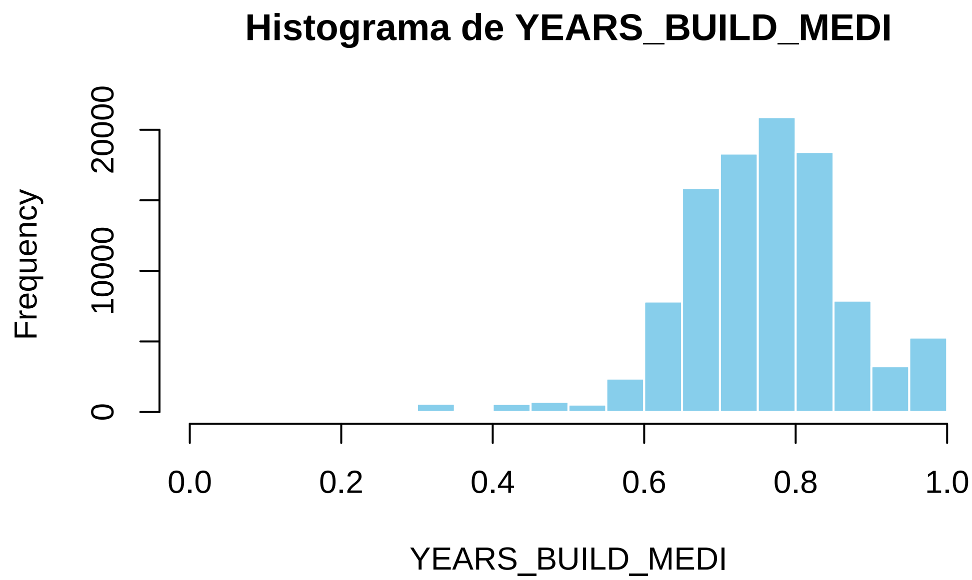

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.16968, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable APARTMENTS_MEDI NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: BASEMENTAREA_MEDI
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.04    0.08    0.09    0.11    1.00  179943

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



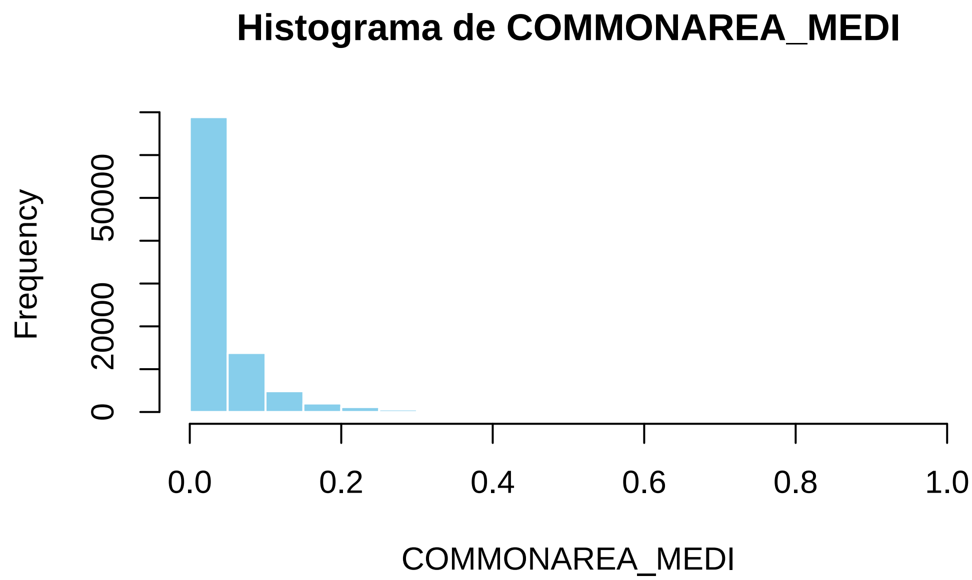

```
Test de Kolmogorov-Smirnov para la normalidad:

	Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.14225, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable BASEMENTAREA_MEDI NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: YEARS_BEGINEXPLUATATION_MEDI
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   0.00    0.98    0.98    0.98    0.99    1.00  150007 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



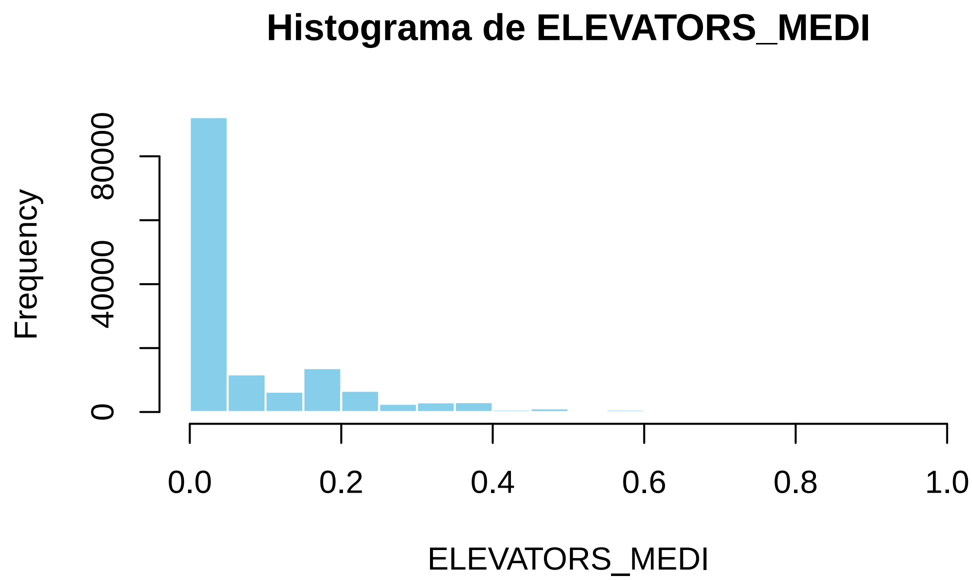

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.39156, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable YEARS_BEGINEXPLUATATION_MEDI NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: YEARS_BUILD_MEDI
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.69    0.76    0.76    0.83    1.00  204488

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



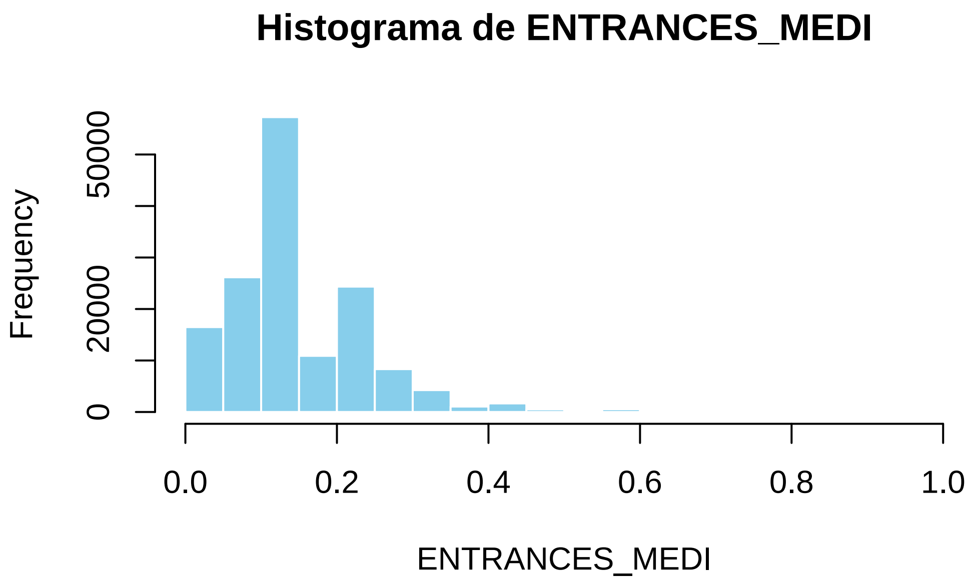

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.051814, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable YEARS_BUILD_MEDI NO sigue una distribución normal (p < 1.165368e-240 )

-------------------------------------------------
Distribución de la variable: COMMONAREA_MEDI
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.01    0.02    0.04    0.05    1.00  214865

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



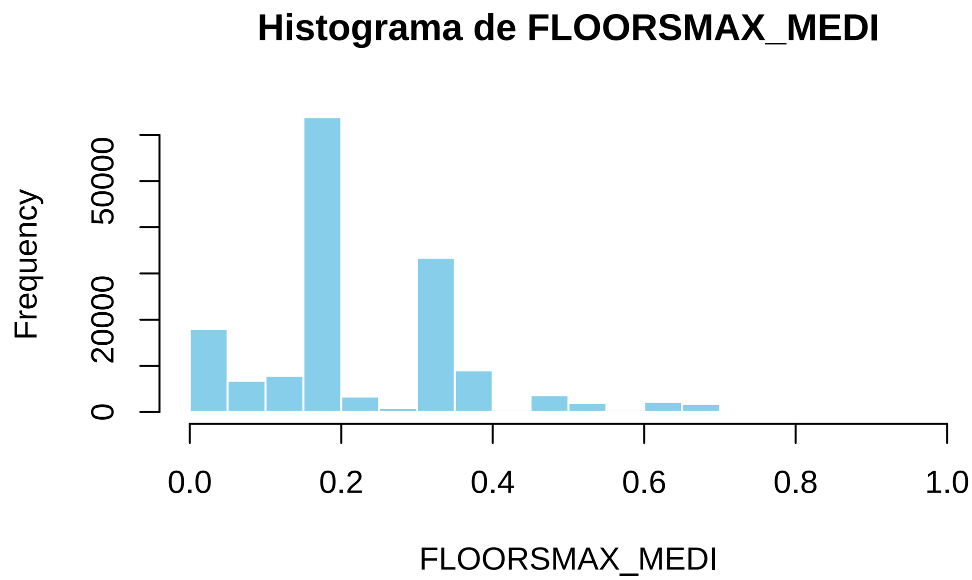

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.27905, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable COMMONAREA_MEDI NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: ELEVATORS_MEDI
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.00    0.00    0.08    0.12    1.00  163891

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



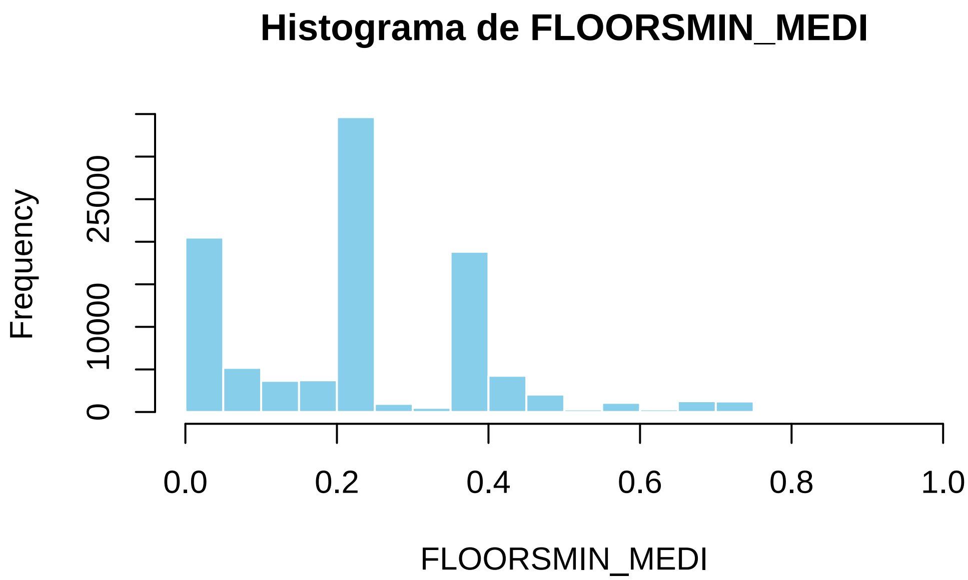

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.32521, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable ELEVATORS_MEDI NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: ENTRANCES_MEDI
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.07    0.14    0.15    0.21    1.00  154828

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



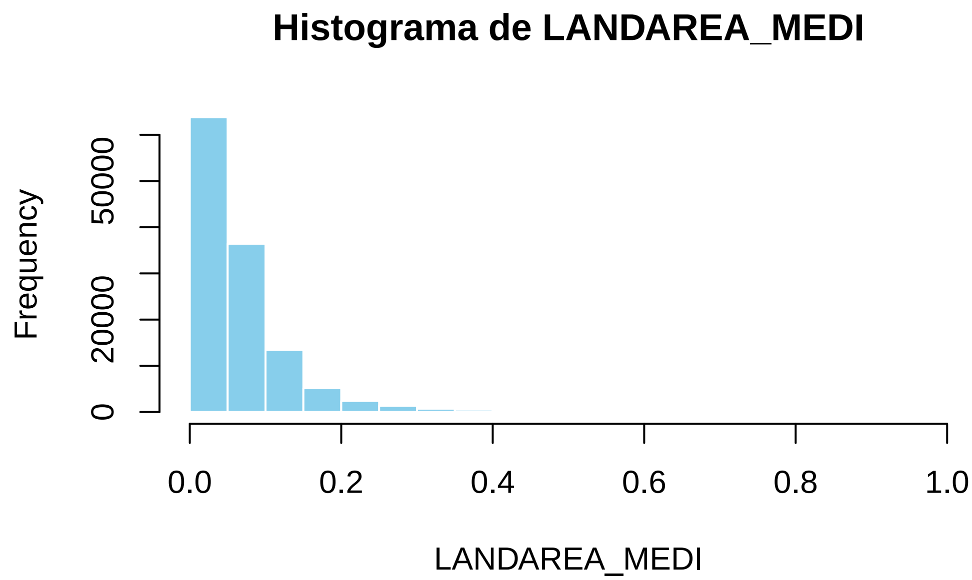

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.19915, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable ENTRANCES_MEDI NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: FLOORSMAX_MEDI
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.17    0.17    0.23    0.33    1.00  153020

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



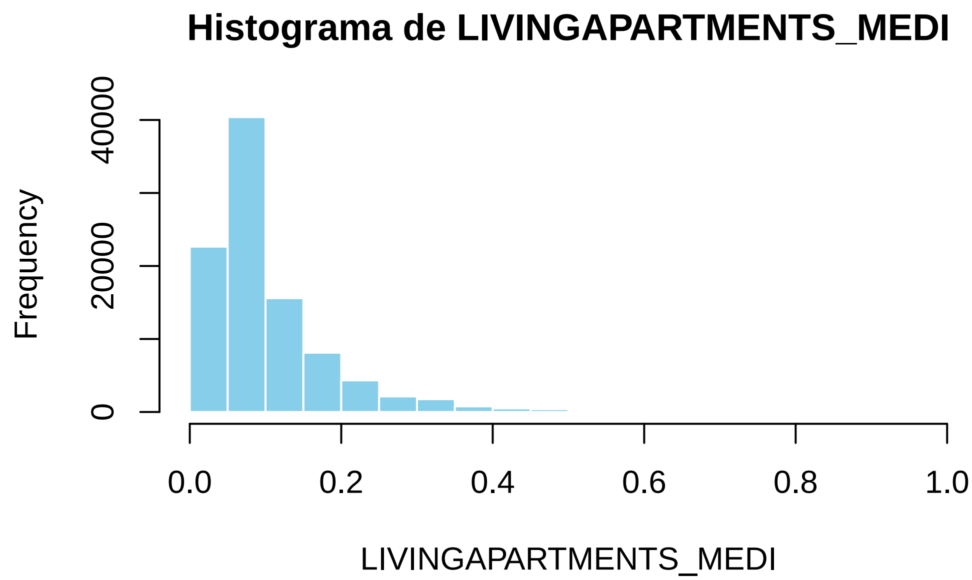

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.28113, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable FLOORSMAX_MEDI NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: FLOORSMIN_MEDI
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   0.00    0.08    0.21    0.23    0.38    1.00  208642 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



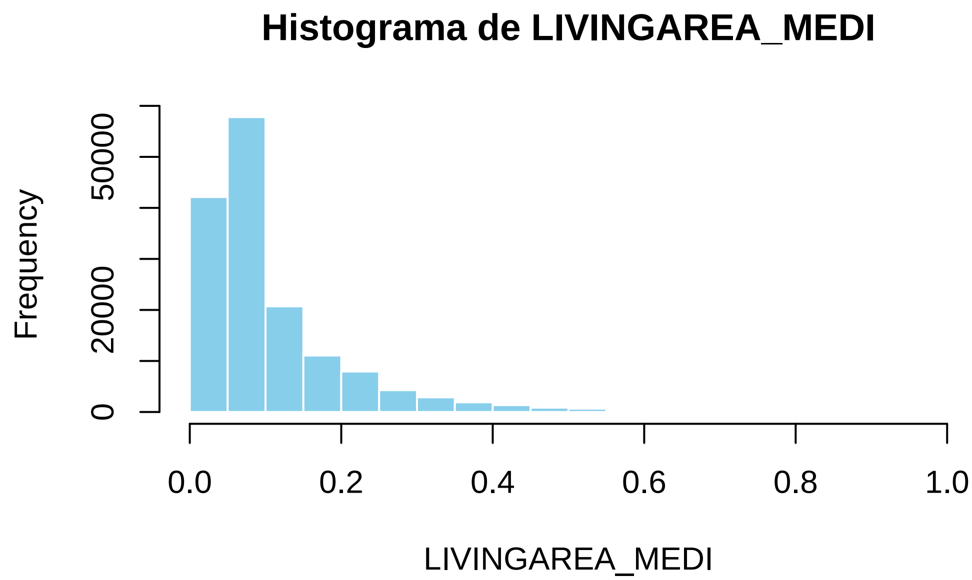

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.23289, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable FLOORSMIN_MEDI NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: LANDAREA_MEDI
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.02    0.05    0.07    0.09    1.00  182590

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



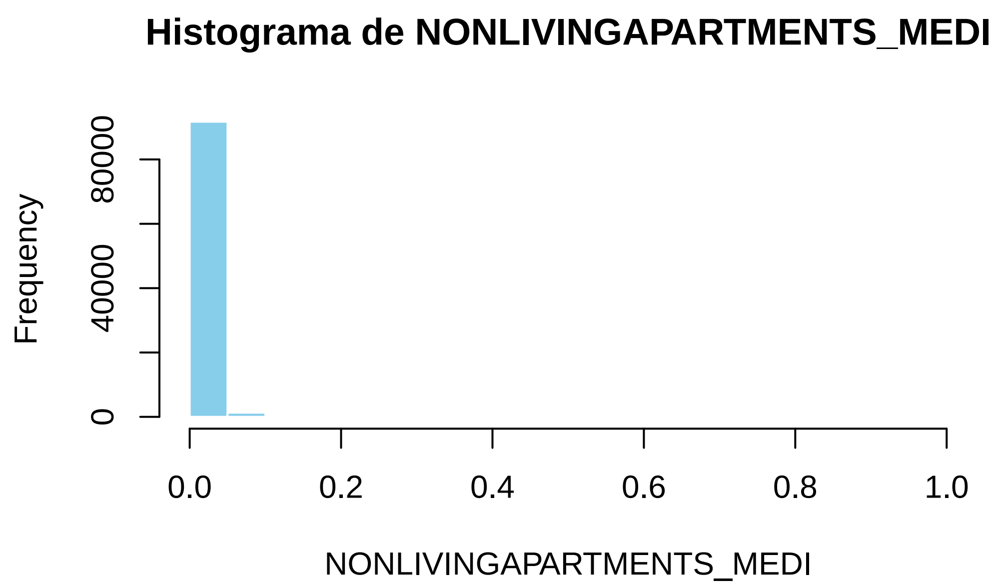

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.20683, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable LANDAREA_MEDI NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: LIVINGAPARTMENTS_MEDI
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.05    0.08    0.10    0.12    1.00  210199

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



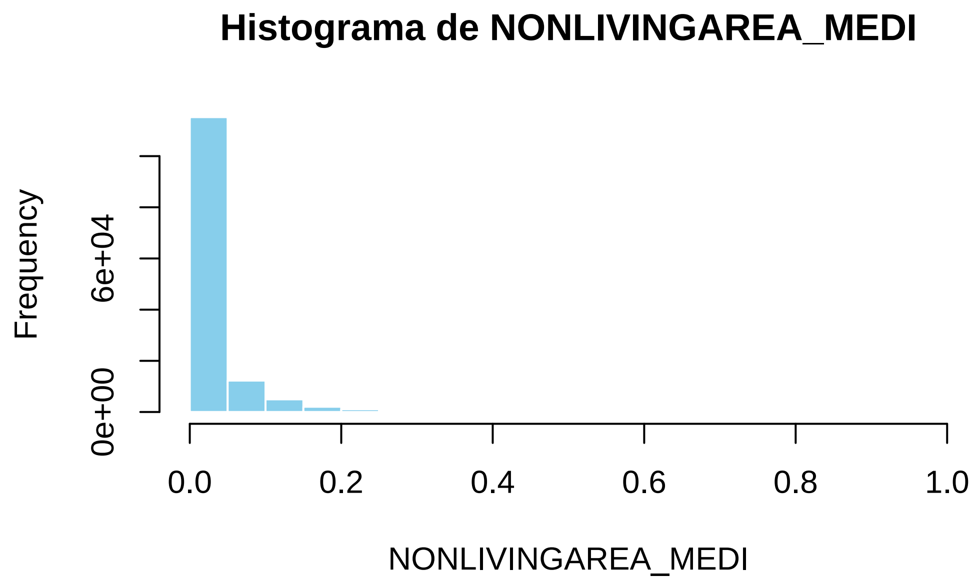

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.17714, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable LIVINGAPARTMENTS_MEDI NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: LIVINGAREA_MEDI
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.05    0.07    0.11    0.13    1.00  154350

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



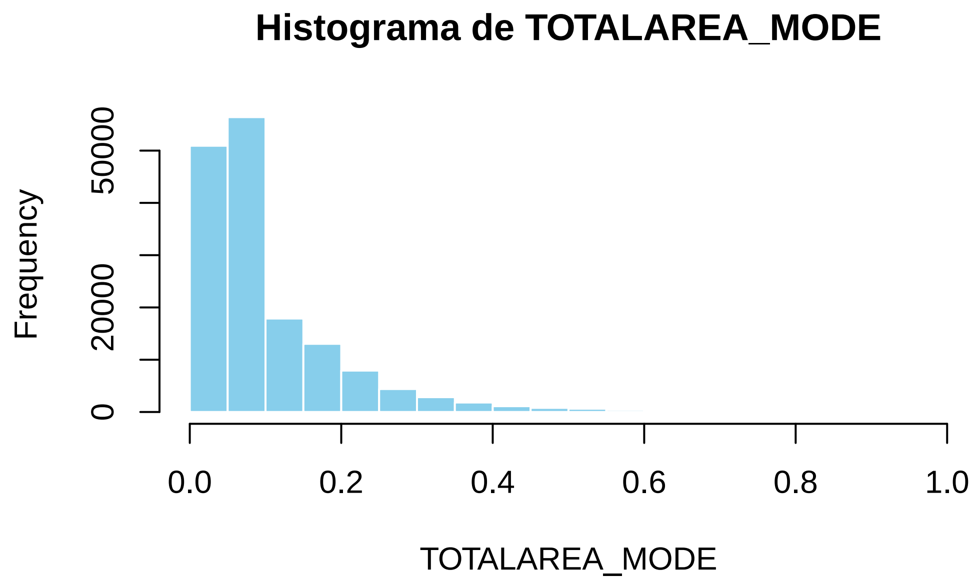

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.18396, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable LIVINGAREA_MEDI NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: NONLIVINGAPARTMENTS_MEDI
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.00    0.00    0.01    0.00    1.00  213514

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



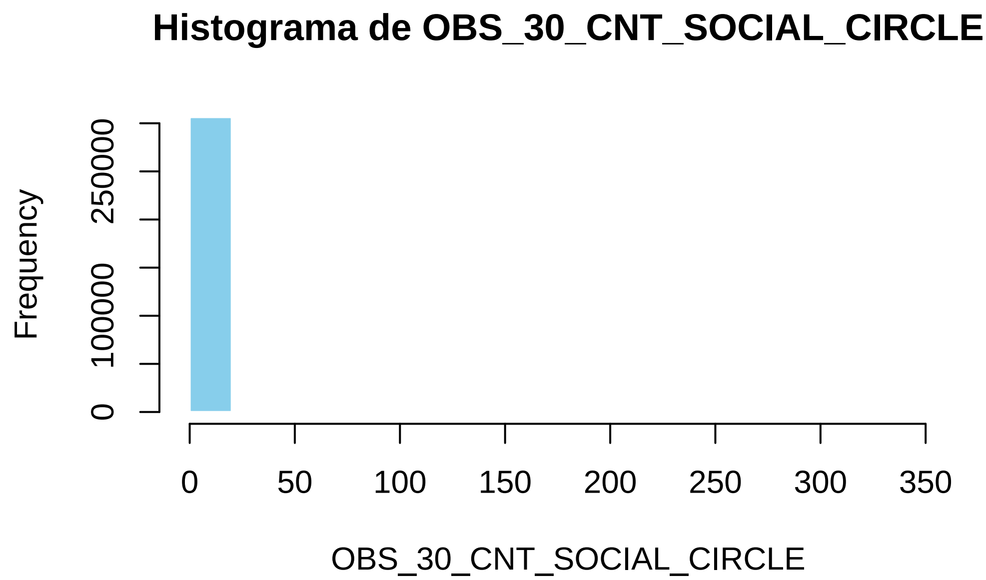

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.42761, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable NONLIVINGAPARTMENTS_MEDI NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: NONLIVINGAREA_MEDI
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.00    0.00    0.03    0.03    1.00  169682

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



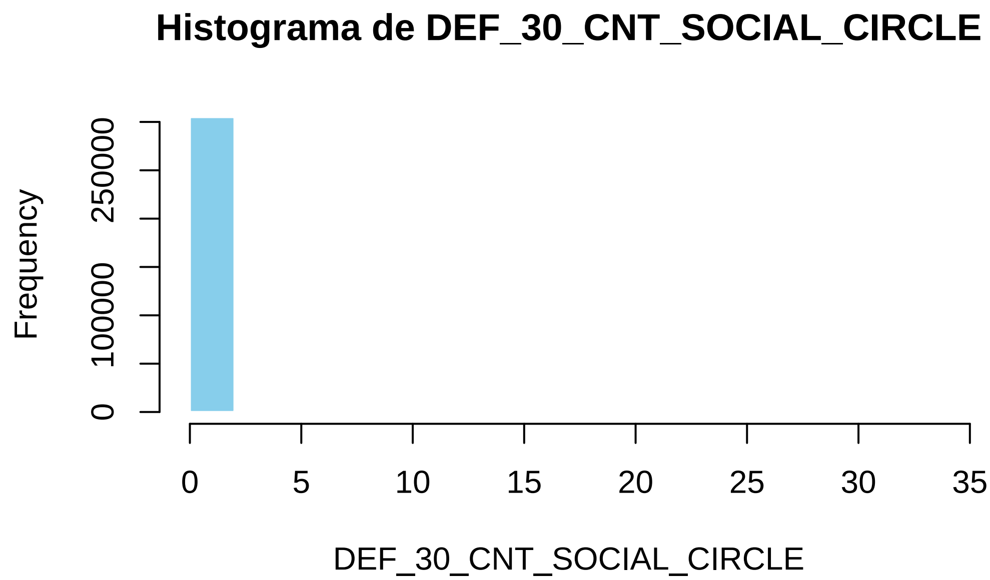

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.34369, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable NONLIVINGAREA_MEDI NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: TOTALAREA_MODE
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.04    0.07    0.10    0.13    1.00  148431

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



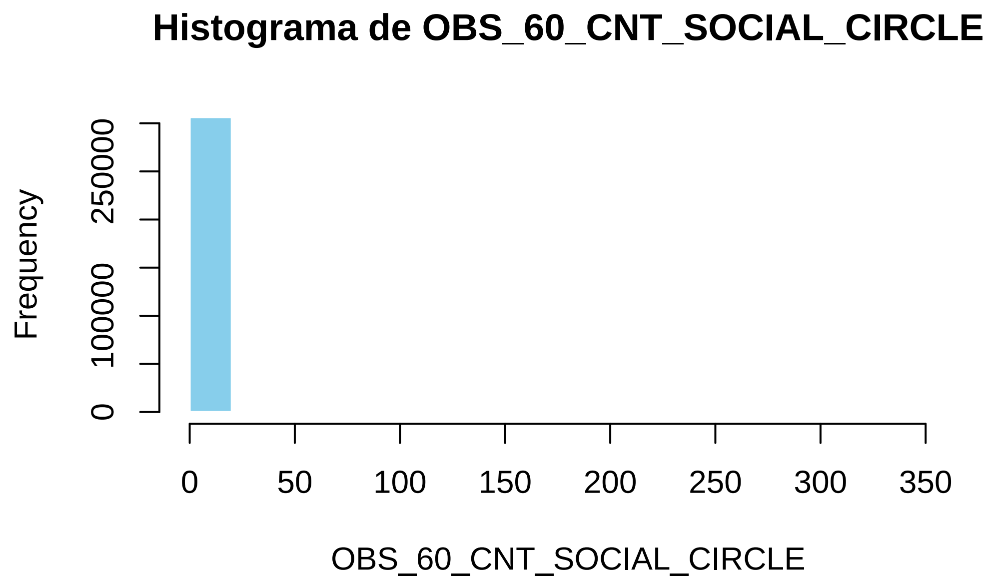

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.18429, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable TOTALAREA_MODE NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: OBS_30_CNT_SOCIAL_CIRCLE
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
  0.000   0.000   0.000   1.422   2.000 348.000    1021

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



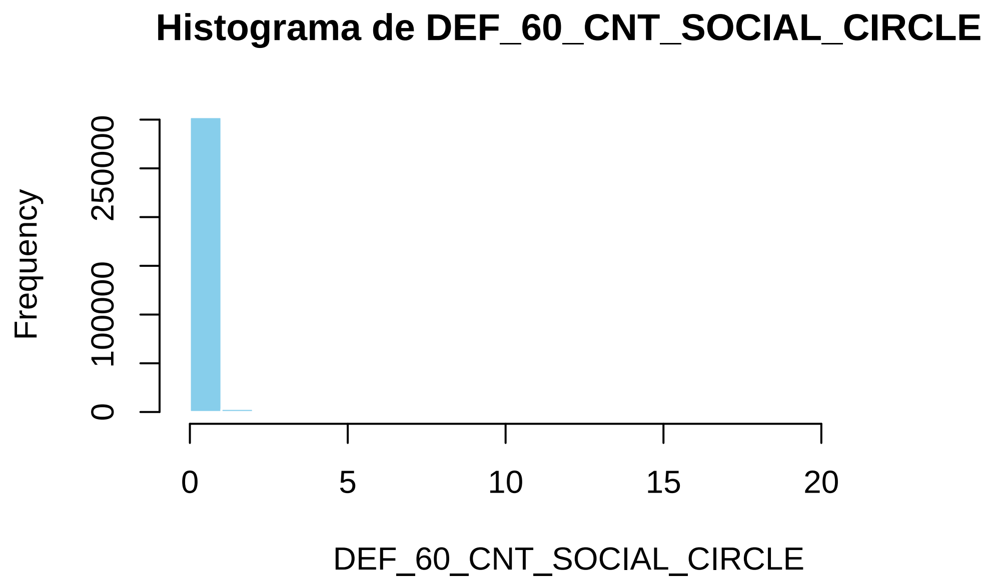

```
Test de Kolmogorov-Smirnov para la normalidad:

	Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.27681, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable OBS_30_CNT_SOCIAL_CIRCLE NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: DEF_30_CNT_SOCIAL_CIRCLE
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
 0.0000  0.0000  0.0000  0.1434  0.0000 34.0000    1021 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



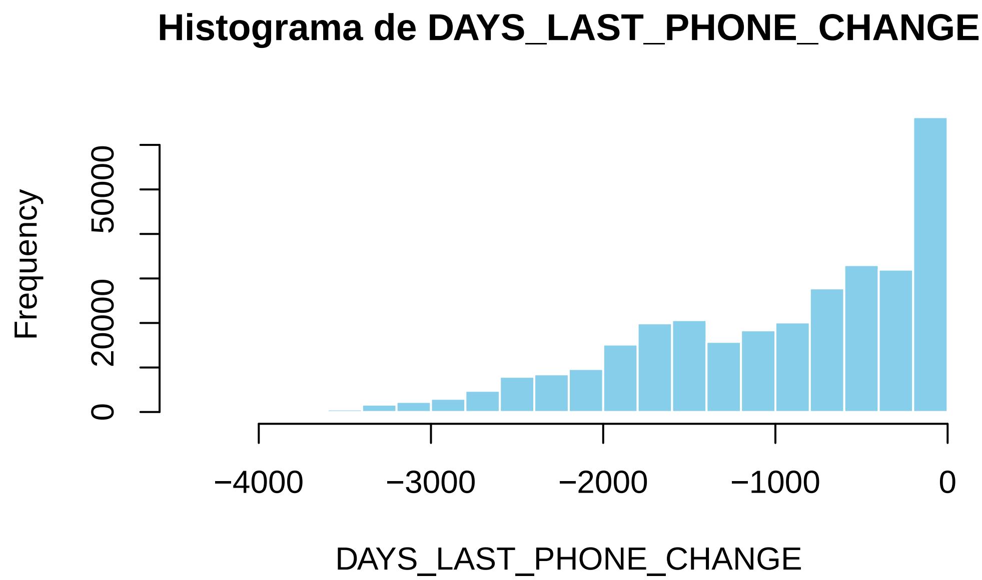

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.51118, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable DEF_30_CNT_SOCIAL_CIRCLE NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: OBS_60_CNT_SOCIAL_CIRCLE
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
  0.000   0.000   0.000   1.405   2.000 344.000    1021

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



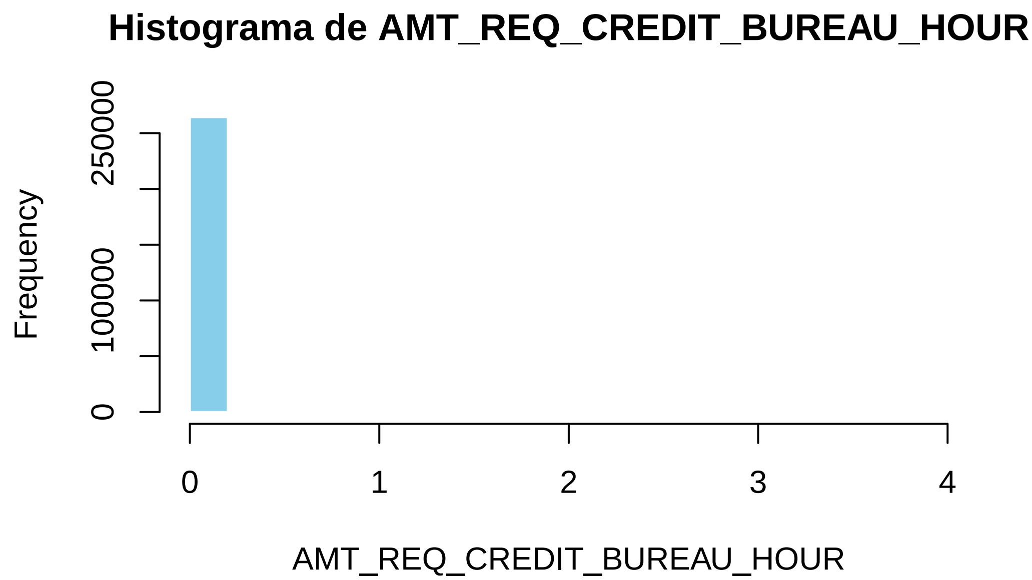

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.27743, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable OBS_60_CNT_SOCIAL_CIRCLE NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: DEF_60_CNT_SOCIAL_CIRCLE
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
    0.0     0.0     0.0     0.1     0.0    24.0    1021

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



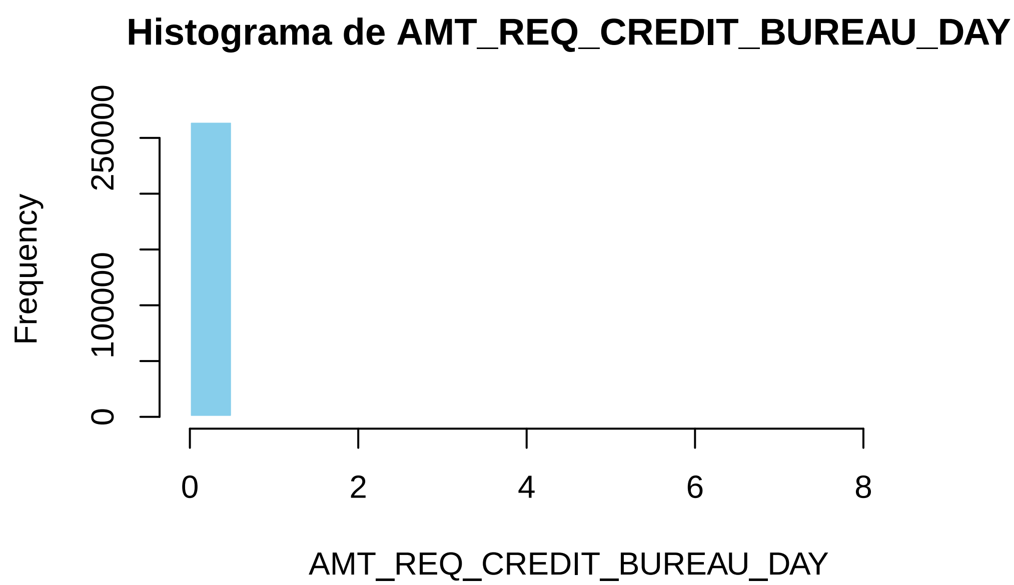

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.52471, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable DEF_60_CNT_SOCIAL_CIRCLE NO sigue una distribución normal (p < 0 )

------------------------------------------------
Distribución de la variable: DAYS_LAST_PHONE_CHANGE
------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
-4292.0 -1570.0  -757.0  -962.9  -274.0     0.0       1

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



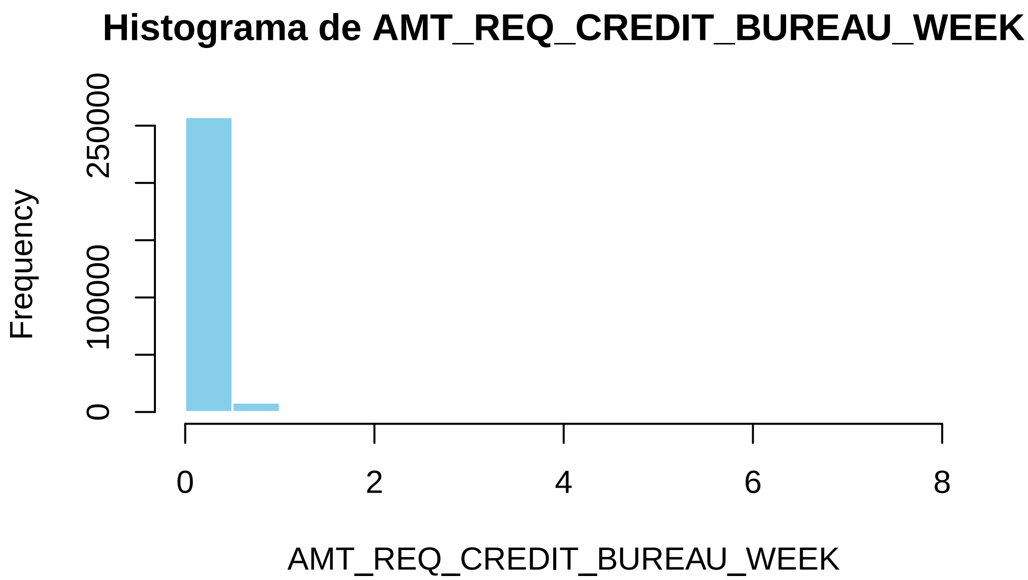

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.1221, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable DAYS_LAST_PHONE_CHANGE NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: AMT_REQ_CREDIT_BUREAU_HOUR
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   0.00    0.00    0.00    0.01    0.00    4.00   41519 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



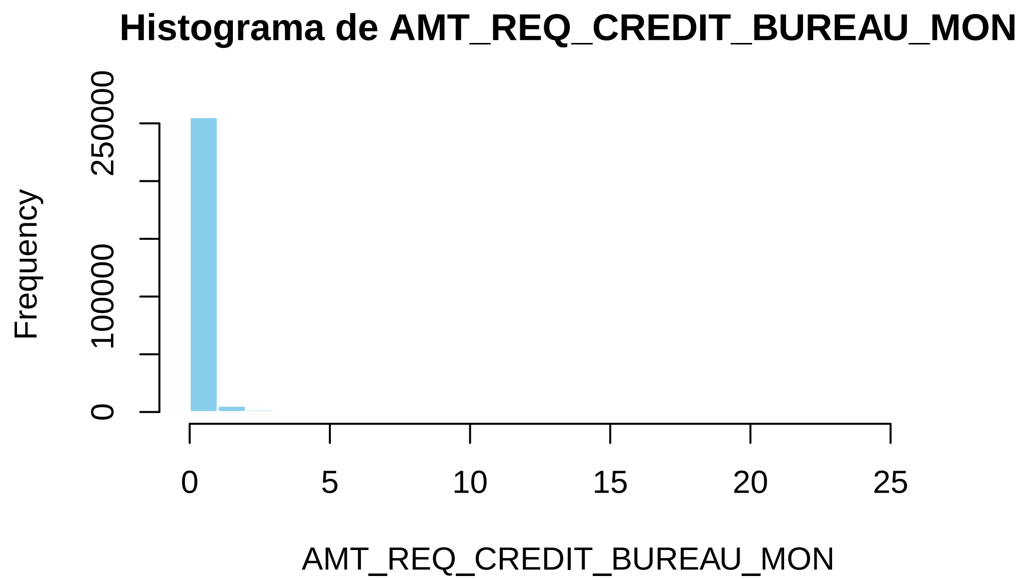

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.52432, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable AMT_REQ_CREDIT_BUREAU_HOUR NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: AMT_REQ_CREDIT_BUREAU_DAY
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.00    0.00    0.01    0.00    9.00   41519

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



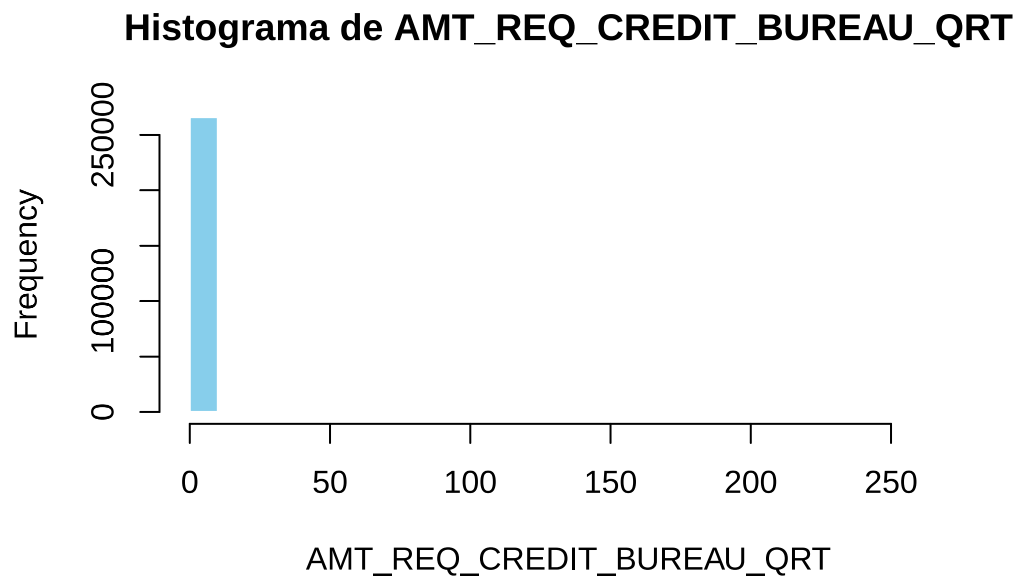

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.5196, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable AMT_REQ_CREDIT_BUREAU_DAY NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: AMT_REQ_CREDIT_BUREAU_WEEK
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.00    0.00    0.03    0.00    8.00   41519

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```



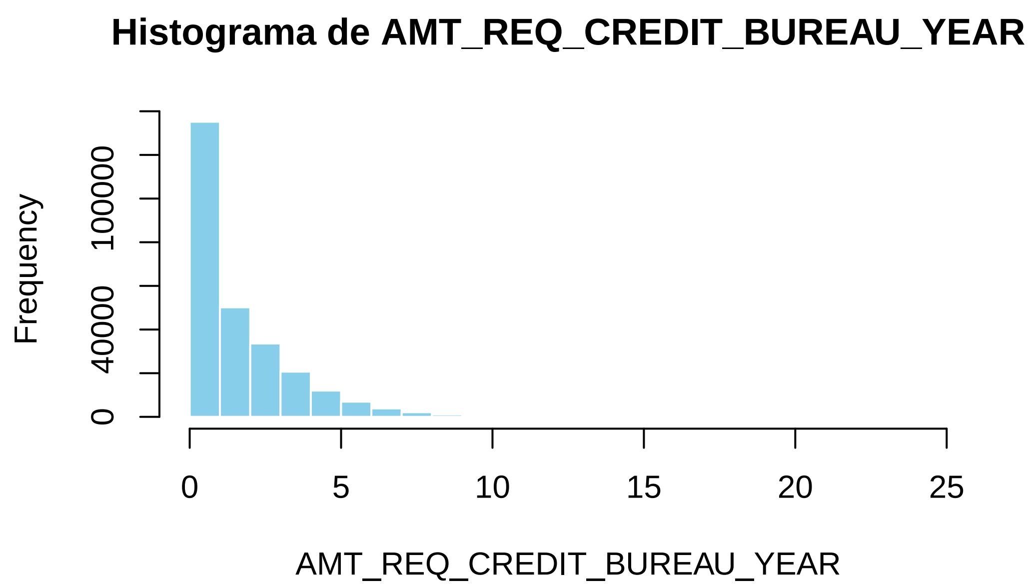

```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.53457, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable AMT_REQ_CREDIT_BUREAU_WEEK NO sigue una distribución normal (p < 0 )

-------------------------------------------------
Distribución de la variable: AMT_REQ_CREDIT_BUREAU_MON
-------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.00    0.00    0.27    0.00   27.00   41519

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```




```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.45031, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable AMT_REQ_CREDIT_BUREAU_MON NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: AMT_REQ_CREDIT_BUREAU_QRT
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   0.00    0.00    0.00    0.27    0.00  261.00   41519 

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```




```
Test de Kolmogorov-Smirnov para la normalidad:

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.4408, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable AMT_REQ_CREDIT_BUREAU_QRT NO sigue una distribución normal (p < 0 )

--------------------------------------------------
Distribución de la variable: AMT_REQ_CREDIT_BUREAU_YEAR
--------------------------------------------------
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
    0.0     0.0     1.0     1.9     3.0    25.0   41519

Warning in ks.test.default(datos[[col]], "pnorm", mean(datos[[col]], na.rm =
TRUE), : ties should not be present for the one-sample Kolmogorov-Smirnov test
```




```
Test de Kolmogorov-Smirnov para la normalidad:

	Asymptotic one-sample Kolmogorov-Smirnov test

data:  datos[[col]]
D = 0.19321, p-value < 2.2e-16
alternative hypothesis: two-sided

 La variable AMT_REQ_CREDIT_BUREAU_YEAR NO sigue una distribución normal (p < 0 )
```

```r
# Función para imputar valores faltantes con la media
imputar_mediana <- function(x) {
  if (is.numeric(x)) {  # Verifica si es numérica
    x[is.na(x)] <- median(x, na.rm = TRUE)  # Calcula y reemplaza con la media
  }
  return(x)
}
```

```r
numeric_columns <- datos |> select_if(is.numeric) |> names()

 # Aplicar la función a todas las columnas numéricas
datos[numeric_columns] <- lapply(datos[numeric_columns], imputar_mediana)
```

```r
data.frame(sort(colSums(is.na(datos))))
```

```
                            sort.colSums.is.na.datos...
SK_ID_CURR                                            0
TARGET                                                0
NAME_CONTRACT_TYPE                                    0
CODE_GENDER                                           0
FLAG_OWN_CAR                                          0
FLAG_OWN_REALTY                                       0
CNT_CHILDREN                                          0
AMT_INCOME_TOTAL                                      0
AMT_CREDIT                                            0
AMT_ANNUITY                                           0
AMT_GOODS_PRICE                                       0
NAME_TYPE_SUITE                                       0
NAME_INCOME_TYPE                                      0
NAME_EDUCATION_TYPE                                   0
NAME_FAMILY_STATUS                                    0
NAME_HOUSING_TYPE                                     0
REGION_POPULATION_RELATIVE                            0
DAYS_BIRTH                                            0
DAYS_EMPLOYED                                         0
DAYS_REGISTRATION                                     0
DAYS_ID_PUBLISH                                       0
OWN_CAR_AGE                                           0
FLAG_MOBIL                                            0
FLAG_EMP_PHONE                                        0
FLAG_WORK_PHONE                                       0
FLAG_CONT_MOBILE                                      0
FLAG_PHONE                                            0
FLAG_EMAIL                                            0
OCCUPATION_TYPE                                       0
CNT_FAM_MEMBERS                                       0
REGION_RATING_CLIENT                                  0
REGION_RATING_CLIENT_W_CITY                           0
WEEKDAY_APPR_PROCESS_START                            0
HOUR_APPR_PROCESS_START                               0
REG_REGION_NOT_LIVE_REGION                            0
REG_REGION_NOT_WORK_REGION                            0
LIVE_REGION_NOT_WORK_REGION                           0
```

REG_CITY_NOT_LIVE_CITY	0
REG_CITY_NOT_WORK_CITY	0
LIVE_CITY_NOT_WORK_CITY	0
ORGANIZATION_TYPE	0
EXT_SOURCE_1	0
EXT_SOURCE_2	0
EXT_SOURCE_3	0
APARTMENTS_AVG	0
BASEMENTAREA_AVG	0
YEARS_BEGINEXPLUATATION_AVG	0
YEARS_BUILD_AVG	0
COMMONAREA_AVG	0
ELEVATORS_AVG	0
ENTRANCES_AVG	0
FLOORSMAX_AVG	0
FLOORSMIN_AVG	0
LANDAREA_AVG	0
LIVINGAPARTMENTS_AVG	0
LIVINGAREA_AVG	0
NONLIVINGAPARTMENTS_AVG	0
NONLIVINGAREA_AVG	0
APARTMENTS_MODE	0
BASEMENTAREA_MODE	0
YEARS_BEGINEXPLUATATION_MODE	0
YEARS_BUILD_MODE	0
COMMONAREA_MODE	0
ELEVATORS_MODE	0
ENTRANCES_MODE	0
FLOORSMAX_MODE	0
FLOORSMIN_MODE	0
LANDAREA_MODE	0
LIVINGAPARTMENTS_MODE	0
LIVINGAREA_MODE	0
NONLIVINGAPARTMENTS_MODE	0
NONLIVINGAREA_MODE	0
APARTMENTS_MEDI	0
BASEMENTAREA_MEDI	0
YEARS_BEGINEXPLUATATION_MEDI	0
YEARS_BUILD_MEDI	0
COMMONAREA_MEDI	0
ELEVATORS_MEDI	0
ENTRANCES_MEDI	0
FLOORSMAX_MEDI	0

FLOORSMIN_MEDI	0
LANDAREA_MEDI	0
LIVINGAPARTMENTS_MEDI	0
LIVINGAREA_MEDI	0
NONLIVINGAPARTMENTS_MEDI	0
NONLIVINGAREA_MEDI	0
FONDKAPREMONT_MODE	0
HOUSETYPE_MODE	0
TOTALAREA_MODE	0
WALLSMATERIAL_MODE	0
EMERGENCYSTATE_MODE	0
OBS_30_CNT_SOCIAL_CIRCLE	0
DEF_30_CNT_SOCIAL_CIRCLE	0
OBS_60_CNT_SOCIAL_CIRCLE	0
DEF_60_CNT_SOCIAL_CIRCLE	0
DAYS_LAST_PHONE_CHANGE	0
FLAG_DOCUMENT_2	0
FLAG_DOCUMENT_3	0
FLAG_DOCUMENT_4	0
FLAG_DOCUMENT_5	0
FLAG_DOCUMENT_6	0
FLAG_DOCUMENT_7	0
FLAG_DOCUMENT_8	0
FLAG_DOCUMENT_9	0
FLAG_DOCUMENT_10	0
FLAG_DOCUMENT_11	0
FLAG_DOCUMENT_12	0
FLAG_DOCUMENT_13	0
FLAG_DOCUMENT_14	0
FLAG_DOCUMENT_15	0
FLAG_DOCUMENT_16	0
FLAG_DOCUMENT_17	0
FLAG_DOCUMENT_18	0
FLAG_DOCUMENT_19	0
FLAG_DOCUMENT_20	0
FLAG_DOCUMENT_21	0
AMT_REQ_CREDIT_BUREAU_HOUR	0
AMT_REQ_CREDIT_BUREAU_DAY	0
AMT_REQ_CREDIT_BUREAU_WEEK	0
AMT_REQ_CREDIT_BUREAU_MON	0
AMT_REQ_CREDIT_BUREAU_QRT	0
AMT_REQ_CREDIT_BUREAU_YEAR	0

Estandarizar valores

Primero pasamos las columnas con dias negativos a positivos

```
# Lista de columnas con días negativos
date_col <- c("DAYS_BIRTH", "DAYS_EMPLOYED", "DAYS_REGISTRATION", "DAYS_ID_PUBLISH")

# Convertir valores negativos a positivos en todas las columnas de la lista
datos[date_col] <- abs(datos[date_col])
```

Ahora vamos a organizar a las personas segun su nivel de ingresos (Dicotomizamos)

```
# Dividir AMT_INCOME_TOTAL por 100,000
datos$AMT_INCOME_TOTAL <- datos$AMT_INCOME_TOTAL / 100000

# Definir los límites de los bins
bins <- c(0,1,2,3,4,5,6,7,8,9,10,11)

# Definir las etiquetas para los rangos de ingresos
slot <- c('0-100K','100K-200K', '200K-300K','300K-400K','400K-500K',
          '500K-600K','600K-700K','700K-800K','800K-900K','900K-1M', '1M Above')

# Crear la nueva variable categórica usando cut()
datos$AMT_INCOME_RANGE <- cut(datos$AMT_INCOME_TOTAL, breaks = bins, labels = slot, include.l

# Calcular la frecuencia relativa (%) de cada categoría en AMT_INCOME_RANGE
prop.table(table(datos$AMT_INCOME_RANGE)) * 100
```

```
      0-100K    100K-200K    200K-300K    300K-400K    400K-500K    500K-600K 
20.729695163 50.734999788 21.210691261  4.776115517  1.744668526  0.356353672 
   600K-700K    700K-800K    800K-900K      900K-1M     1M Above 
 0.282804878  0.052720817  0.096980269  0.009112240  0.005857869 
```

Relaizamos lo mismo para la cantida de credito, la edad y las horas trabajadas para facilitar las comparaciones en el futuro

```
# Dividir AMT_CREDIT por 100,000
datos$AMT_CREDIT <- datos$AMT_CREDIT / 100000

# Definir los límites de los bins
```

```r
bins <- c(0,1,2,3,4,5,6,7,8,9,10,100)

# Definir las etiquetas para los rangos de crédito
slots <- c('0-100K','100K-200K', '200K-300K','300K-400K','400K-500K',
           '500K-600K','600K-700K','700K-800K','800K-900K','900K-1M', '1M Above')

# Crear la nueva variable categórica
datos$AMT_CREDIT_RANGE <- cut(datos$AMT_CREDIT, breaks = bins, labels = slots, include.lowes

# Calcular la frecuencia relativa (%) de cada categoría en AMT_CREDIT_RANGE
prop.table(table(datos$AMT_CREDIT_RANGE)) * 100
```

```
   0-100K 100K-200K 200K-300K 300K-400K 400K-500K 500K-600K 600K-700K 700K-800K
 1.952450  9.801275 17.824728  8.564897 10.418489 11.131960  7.820533  6.241403
800K-900K   900K-1M  1M Above
 7.086576  2.902986 16.254703
```

```r
# Crear la variable AGE a partir de DAYS_BIRTH
datos$AGE <- floor(abs(datos$DAYS_BIRTH) / 365)

# Definir los límites de los bins
bins <- c(0, 20, 30, 40, 50, 100)

# Definir las etiquetas para los grupos de edad
slots <- c('0-20', '20-30', '30-40', '40-50', '50 above')

# Crear la nueva variable categórica
datos$AGE_GROUP <- cut(datos$AGE, breaks = bins, labels = slots, include.lowest = TRUE)

# Calcular la frecuencia relativa (%) de cada categoría en AGE_GROUP
prop.table(table(datos$AGE_GROUP)) * 100
```

```
        0-20        20-30        30-40        40-50     50 above
3.251916e-04 1.717174e+01 2.702895e+01 2.419458e+01 3.160440e+01
```

```r
datos$AGE <- floor(abs(datos$DAYS_BIRTH) / 365)
```

```
# Crear la variable YEARS_EMPLOYED a partir de DAYS_EMPLOYED
datos$YEARS_EMPLOYED <- floor(abs(datos$DAYS_EMPLOYED) / 365)

# Definir los límites de los bins
bins <- c(0, 5, 10, 20, 30, 40, 50, 60, 150)

# Definir las etiquetas para los grupos de años de empleo
slots <- c('0-5', '5-10', '10-20', '20-30', '30-40', '40-50', '50-60', '60 above')

# Crear la nueva variable categórica
datos$EMPLOYMENT_YEAR <- cut(datos$YEARS_EMPLOYED, breaks = bins, labels = slots, include.low

# Calcular la frecuencia relativa (%) de cada categoría en EMPLOYMENT_YEAR
prop.table(table(datos$EMPLOYMENT_YEAR)) * 100
```

```
        0-5        5-10       10-20       20-30       30-40       40-50 
60.49806256 22.20340529 12.95248218  3.33509164  0.94155162  0.06940671 
      50-60    60 above 
 0.00000000  0.00000000 
```

Se lleva a cabo esto para poder facilitar la comparacion entre observaciones y la clasificacion de modelos. Viendo la diferencia entre los distintos grupos

L1 PENALTY PARA LA REGRESION USAR apuntaría brevemente en cada caso, que puedes hacer para seguir

Factorial de variables

Variables economicas

```
economic_vars <- datos[, c("AMT_INCOME_TOTAL", "AMT_CREDIT", "AMT_ANNUITY", "AMT_GOODS_PRICE"
#"CNT_FAM_MEMBERS" "CNT_CHILDREN"

economic_vars_scaled <- scale(economic_vars)
factor_analysis <- factanal(economic_vars_scaled, factors = 2, rotation = "varimax")

print(factor_analysis, digits = 3, cutoff = 0.3, sort = TRUE)
```

```
Call:
factanal(x = economic_vars_scaled, factors = 2, rotation = "varimax")

Uniquenesses:
AMT_INCOME_TOTAL       AMT_CREDIT      AMT_ANNUITY  AMT_GOODS_PRICE
           0.908            0.020            0.328            0.006
     OWN_CAR_AGE    DAYS_EMPLOYED
           0.999            0.953

Loadings:
                 Factor1 Factor2
AMT_CREDIT        0.973
AMT_ANNUITY       0.717   0.398
AMT_GOODS_PRICE   0.980
AMT_INCOME_TOTAL
OWN_CAR_AGE
DAYS_EMPLOYED

               Factor1 Factor2
SS loadings      2.436   0.351
Proportion Var   0.406   0.059
Cumulative Var   0.406   0.464

Test of the hypothesis that 2 factors are sufficient.
The chi square statistic is 671.06 on 4 degrees of freedom.
The p-value is 6.43e-144
```

```
print(factor_analysis$loadings)
```

```
Loadings:
                 Factor1 Factor2
AMT_INCOME_TOTAL  0.110   0.283
AMT_CREDIT        0.973   0.182
AMT_ANNUITY       0.717   0.398
AMT_GOODS_PRICE   0.980   0.181
OWN_CAR_AGE
DAYS_EMPLOYED            -0.216

               Factor1 Factor2
SS loadings      2.436   0.351
Proportion Var   0.406   0.059
```

```
Cumulative Var   0.406   0.464
```

```r
print("------------------------- KMO -----------------------------------")
```

```
[1] "------------------------- KMO -----------------------------------"
```

```r
KMO(economic_vars_scaled)  # Índice de adecuación muestral
```

```
Kaiser-Meyer-Olkin factor adequacy
Call: KMO(r = economic_vars_scaled)
Overall MSA =  0.7
MSA for each item =
AMT_INCOME_TOTAL       AMT_CREDIT      AMT_ANNUITY  AMT_GOODS_PRICE
            0.87             0.63             0.97             0.63
     OWN_CAR_AGE    DAYS_EMPLOYED
            0.61             0.70
```

```r
cortest.bartlett(economic_vars_scaled)  # Prueba de esfericidad de Bartlett
```

```
R was not square, finding R from data

$chisq
[1] 1417942

$p.value
[1] 0

$df
[1] 15
```

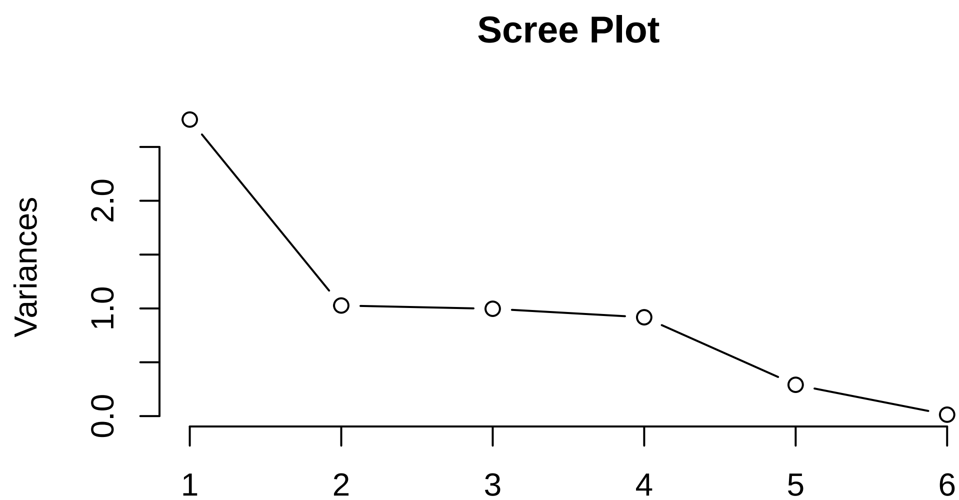
```r
print("------------------------ loadings ------------------------------------")
```

```
[1] "------------------------ loadings ------------------------------------"
```

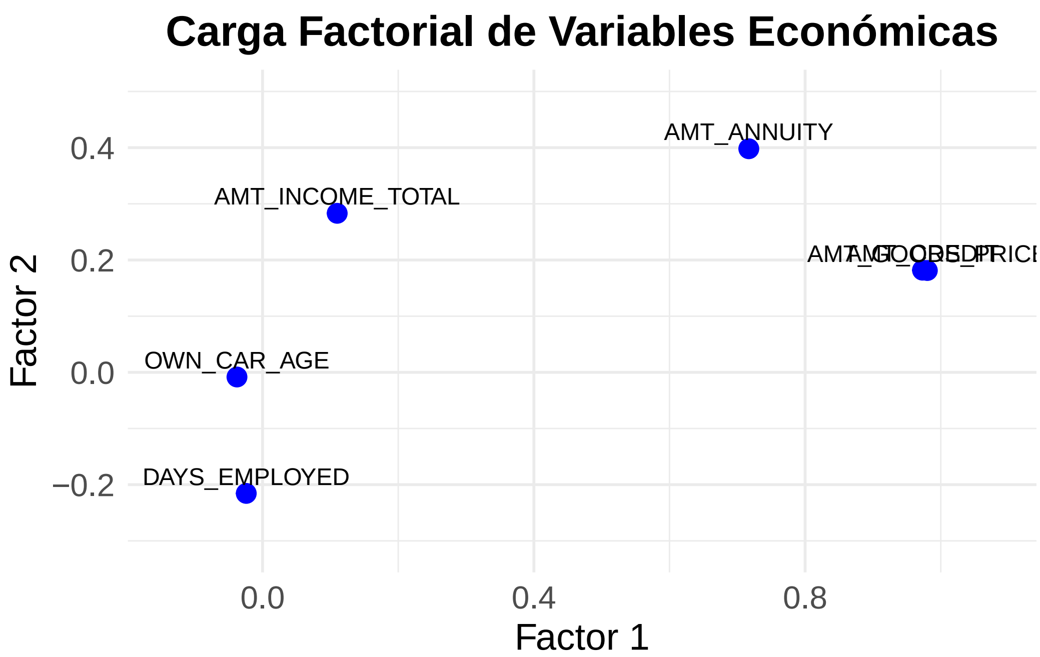
```r
loadings <- as.data.frame(factor_analysis$loadings[,1:2])
loadings$Variable <- rownames(loadings)
print("-------------------------- ggplot ----------------------------------")
```

```
[1] "-------------------------- ggplot ----------------------------------"
```

```r
pca_result <- prcomp(economic_vars_scaled, scale = TRUE)
screeplot(pca_result, type = "lines", main = "Scree Plot")
```



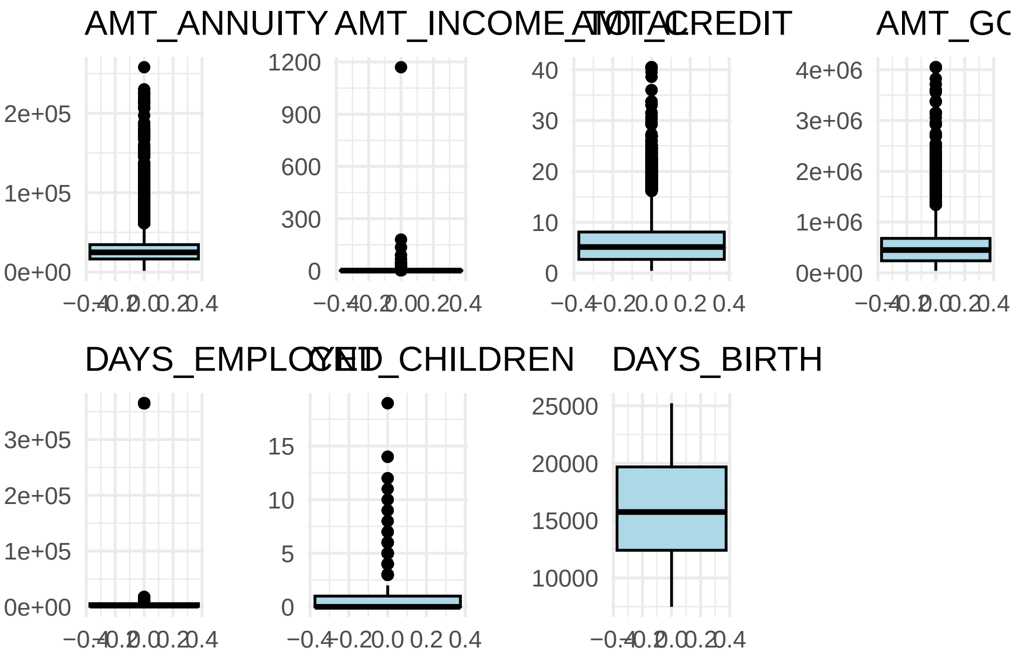

```r
ggplot(loadings, aes(x = Factor1, y = Factor2, label = Variable)) +
  geom_point(color = "blue", size = 3) +  # Agrega puntos
  geom_text(vjust = -0.5, hjust = 0.5, size = 3) +  # Reduce tamaño de texto
  theme_minimal() +
  ggtitle("Carga Factorial de Variables Económicas") +
  xlab("Factor 1") +
  ylab("Factor 2") +
  theme(
    plot.title = element_text(hjust = 0.5, size = 16, face = "bold"),
    axis.title = element_text(size = 14),
    axis.text = element_text(size = 12)
  ) +
  xlim(c(min(loadings$Factor1) - 0.1, max(loadings$Factor1) + 0.1)) +
  ylim(c(min(loadings$Factor2) - 0.1, max(loadings$Factor2) + 0.1))
```



Valores atipicos

```
# Definir las variables para analizar outliers
app_outlier_col_1 <- c('AMT_ANNUITY', 'AMT_INCOME_TOTAL', 'AMT_CREDIT', 'AMT_GOODS_PRICE', 'I
app_outlier_col_2 <- c('CNT_CHILDREN', 'DAYS_BIRTH')

# Crear boxplots para app_outlier_col_1
plots1 <- lapply(app_outlier_col_1, function(var) {
  ggplot(datos, aes(y = .data[[var]])) +
    geom_boxplot(fill = "lightblue", color = "black") +
    labs(title = var, y = "") +
    theme_minimal()
})

# Crear boxplots para app_outlier_col_2
plots2 <- lapply(app_outlier_col_2, function(var) {
  ggplot(datos, aes(y = .data[[var]])) +
    geom_boxplot(fill = "lightblue", color = "black") +
    labs(title = var, y = "") +
    theme_minimal()
})
```

```r
# Mostrar todos los gráficos en una sola figura
grid.arrange(grobs = c(plots1, plots2), ncol = 4)
```




```r
# eliminamos la categoria unknown de NAME_FAMILY_STATUS al no tener ninguna observacion
datos <- datos |> filter(NAME_FAMILY_STATUS != "Unknown")
datos$NAME_FAMILY_STATUS <- droplevels(datos$NAME_FAMILY_STATUS)
#eliminamos la categoria de "60 above" y "50-60" para YEARS_EMPLOYED
datos <- datos[!datos$EMPLOYMENT_YEAR %in% c("50-60", "60 above"), ]
# eliminamos la categoria XNA que tiene 0 observaciones
datos <- datos[datos$CODE_GENDER != "XNA", ]
datos$CODE_GENDER <- droplevels(datos$CODE_GENDER)
# hemos tenido problemas con las personas que estan desempleadas, hay que asignarlas un valo
datos$EMPLOYMENT_YEAR <- ifelse(
  datos$NAME_INCOME_TYPE == "Unemployed", "0", as.character(datos$EMPLOYMENT_YEAR))
datos$EMPLOYMENT_YEAR <- as.factor(datos$EMPLOYMENT_YEAR)
# aquellas observaciones que ya no se han podido sustituir ya sea por valores atipicos o cau
datos <- na.omit(datos)
```

Tablas de contingencia

```
tb_conting <- function(df, x, vec){
  for(i in seq_along(vec)){
    cat("\nTabla de Contingencia para:", vec[i], "\n")

    # Crear tabla de contingencia con nombres de filas y columnas
    tab <- table(df[[x]], df[[vec[i]]])
    dimnames(tab) <- list(TARGET = levels(factor(df[[x]])), Variable = levels(factor(df[[vec

    print(tab)

    cat("\nTest de Chi-Cuadrado:\n")
    chi_test <- chisq.test(tab)
    print(chi_test)

    cat("\n--------------------------\n")
  }
}


# Llamada a la función, suponiendo que df es tu base de datos
tb_conting(datos, "TARGET", contact_col)  # Puedes probar con col_Doc o ext también
```

```
Tabla de Contingencia para: FLAG_MOBIL
      Variable
TARGET      0      1
     0      1 230098
     1      0  21832

Test de Chi-Cuadrado:


Warning in chisq.test(tab): Chi-squared approximation may be incorrect



    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 2.7239e-22, df = 1, p-value = 1
```

```
--------------------------

Tabla de Contingencia para: FLAG_EMP_PHONE
      Variable
TARGET      0      1
     0     25 230074
     1      9  21823

Test de Chi-Cuadrado:

Warning in chisq.test(tab): Chi-squared approximation may be incorrect


    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 11.463, df = 1, p-value = 0.0007101


--------------------------

Tabla de Contingencia para: FLAG_WORK_PHONE
      Variable
TARGET      0      1
     0 174752  55347
     1  15931   5901

Test de Chi-Cuadrado:

    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 95.784, df = 1, p-value < 2.2e-16


--------------------------

Tabla de Contingencia para: FLAG_CONT_MOBILE
      Variable
TARGET      0      1
```

```
     0    490 229609
     1     43  21789

Test de Chi-Cuadrado:

	Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 0.17177, df = 1, p-value = 0.6785


--------------------------

Tabla de Contingencia para: FLAG_PHONE
      Variable
TARGET      0      1
     0 165455  64644
     1  16534   5298

Test de Chi-Cuadrado:

	Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 145.42, df = 1, p-value < 2.2e-16


--------------------------

Tabla de Contingencia para: FLAG_EMAIL
      Variable
TARGET      0      1
     0 215396  14703
     1  20550   1282

Test de Chi-Cuadrado:

	Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 8.9079, df = 1, p-value = 0.002839
```

```
--------------------------
```

```
tb_conting(datos, "TARGET", col_Doc)  # Puedes probar con col_Doc o ext también
```

```
Tabla de Contingencia para: FLAG_DOCUMENT_2
      Variable
TARGET      0      1
     0 230090      9
     1  21828      4

Test de Chi-Cuadrado:


Warning in chisq.test(tab): Chi-squared approximation may be incorrect



    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 5.4751, df = 1, p-value = 0.01929


--------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_3
      Variable
TARGET      0      1
     0  55752 174347
     1   3938  17894

Test de Chi-Cuadrado:

    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 422.5, df = 1, p-value < 2.2e-16


--------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_4
```

```
      Variable
TARGET      0      1
     0 230079     20
     1  21832      0

Test de Chi-Cuadrado:

Warning in chisq.test(tab): Chi-squared approximation may be incorrect


    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 0.96074, df = 1, p-value = 0.327


--------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_5
      Variable
TARGET      0      1
     0 226355   3744
     1  21483    349

Test de Chi-Cuadrado:

    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 0.084646, df = 1, p-value = 0.7711


--------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_6
      Variable
TARGET      0      1
     0 228048   2051
     1  21698    134

Test de Chi-Cuadrado:
```

```
    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 17.548, df = 1, p-value = 2.802e-05


--------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_7
      Variable
TARGET      0      1
     0 230052     47
     1  21829      3

Test de Chi-Cuadrado:

Warning in chisq.test(tab): Chi-squared approximation may be incorrect


    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 0.17534, df = 1, p-value = 0.6754


--------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_8
      Variable
TARGET      0      1
     0 207499  22600
     1  20016   1816

Test de Chi-Cuadrado:

    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 51.349, df = 1, p-value = 7.732e-13


--------------------------
```

```
Tabla de Contingencia para: FLAG_DOCUMENT_9
      Variable
TARGET      0      1
     0 229016   1083
     1  21759     73

Test de Chi-Cuadrado:

    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 7.8141, df = 1, p-value = 0.005184


--------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_10
      Variable
TARGET      0      1
     0 230093      6
     1  21832      0

Test de Chi-Cuadrado:

Warning in chisq.test(tab): Chi-squared approximation may be incorrect


    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 0.00083827, df = 1, p-value = 0.9769


--------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_11
      Variable
TARGET      0      1
     0 228973   1126
     1  21757     75
```

```
Test de Chi-Cuadrado:

    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 8.6322, df = 1, p-value = 0.003303


--------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_12
      Variable
TARGET      0      1
     0 230097      2
     1  21832      0

Test de Chi-Cuadrado:

Warning in chisq.test(tab): Chi-squared approximation may be incorrect


    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 5.4479e-22, df = 1, p-value = 1


--------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_13
      Variable
TARGET      0      1
     0 229063   1036
     1  21803     29

Test de Chi-Cuadrado:

    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 46.973, df = 1, p-value = 7.198e-12
```

```
-------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_14
      Variable
TARGET      0      1
     0 229244    855
     1  21802     30

Test de Chi-Cuadrado:

    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 30.57, df = 1, p-value = 3.221e-08


-------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_15
      Variable
TARGET      0      1
     0 229748    351
     1  21821     11

Test de Chi-Cuadrado:

    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 13.8, df = 1, p-value = 0.0002033


-------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_16
      Variable
TARGET      0      1
     0 227248   2851
     1  21682    150

Test de Chi-Cuadrado:
```

```
    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 51.147, df = 1, p-value = 8.571e-13


--------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_17
      Variable
TARGET      0      1
     0 230020     79
     1  21830      2

Test de Chi-Cuadrado:

    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 3.1869, df = 1, p-value = 0.07423


--------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_18
      Variable
TARGET      0      1
     0 227768   2331
     1  21690    142

Test de Chi-Cuadrado:

    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 26.604, df = 1, p-value = 2.497e-07


--------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_19
      Variable
TARGET      0      1
```

```
     0 229932    167
     1  21820     12

Test de Chi-Cuadrado:

    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 0.64075, df = 1, p-value = 0.4234


--------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_20
      Variable
TARGET      0      1
     0 229957    142
     1  21819     13

Test de Chi-Cuadrado:

    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 2.9034e-28, df = 1, p-value = 1


--------------------------

Tabla de Contingencia para: FLAG_DOCUMENT_21
      Variable
TARGET      0      1
     0 230010     89
     1  21818     14

Test de Chi-Cuadrado:

    Pearson's Chi-squared test with Yates' continuity correction

data:  tab
X-squared = 2.5675, df = 1, p-value = 0.1091
```

Analisis de Datos

En un principio me interesa saber cuales son las variables mas importantes a la hora de predecir si alguien va a devovler el pago o no, por tanto realizamos un modelo con todas las variables y hacemos el ANOVA para ver cuales son las mas significativas

```
#anova(lm(TARGET~.,data=datos))
anova_results <- anova(lm(TARGET ~ ., data = datos))

# Ordenar por la suma de cuadrados (Sum Sq) en orden descendente
(anova_sorted <- anova_results[order(-anova_results$`Sum Sq`), ])
```

```
Analysis of Variance Table

Response: TARGET
                                Df   Sum Sq Mean Sq   F value    Pr(>F)
Residuals                   251666 18566.3    0.07
EXT_SOURCE_3                     1   324.4  324.44 4397.8180 < 2.2e-16 ***
EXT_SOURCE_2                     1   320.4  320.40 4343.0108 < 2.2e-16 ***
DAYS_BIRTH                       1    61.4   61.44  832.7706 < 2.2e-16 ***
AMT_GOODS_PRICE                  1    57.1   57.05  773.3386 < 2.2e-16 ***
FLAG_OWN_CAR                     1    51.8   51.78  701.8813 < 2.2e-16 ***
EXT_SOURCE_1                     1    49.1   49.13  665.9867 < 2.2e-16 ***
CODE_GENDER                      1    47.0   47.01  637.2217 < 2.2e-16 ***
DAYS_EMPLOYED                    1    42.2   42.17  571.6829 < 2.2e-16 ***
REGION_RATING_CLIENT             2    41.8   20.90  283.3041 < 2.2e-16 ***
NAME_EDUCATION_TYPE              4    39.6    9.89  134.0456 < 2.2e-16 ***
AMT_INCOME_TOTAL                 1    29.3   29.31  397.2466 < 2.2e-16 ***
NAME_INCOME_TYPE                 7    28.2    4.03   54.6094 < 2.2e-16 ***
AMT_CREDIT_RANGE                10    26.1    2.61   35.3973 < 2.2e-16 ***
NAME_CONTRACT_TYPE               1    26.0   25.97  352.0502 < 2.2e-16 ***
NAME_FAMILY_STATUS               4    23.3    5.81   78.8051 < 2.2e-16 ***
AMT_CREDIT                       1    21.6   21.62  293.1172 < 2.2e-16 ***
ORGANIZATION_TYPE               56    20.9    0.37    5.0688 < 2.2e-16 ***
DAYS_ID_PUBLISH                  1    18.4   18.44  250.0106 < 2.2e-16 ***
OCCUPATION_TYPE                 18    17.3    0.96   12.9905 < 2.2e-16 ***
REGION_POPULATION_RELATIVE       1    14.8   14.80  200.6091 < 2.2e-16 ***
NAME_HOUSING_TYPE                5    11.8    2.36   32.0557 < 2.2e-16 ***
FLAG_WORK_PHONE                  1    10.0    9.99  135.4386 < 2.2e-16 ***
```

DEF_30_CNT_SOCIAL_CIRCLE	1	9.9	9.88	133.9102	< 2.2e-16	***
REG_CITY_NOT_LIVE_CITY	1	8.0	8.05	109.0890	< 2.2e-16	***
DAYS_REGISTRATION	1	6.9	6.93	93.8974	< 2.2e-16	***
REGION_RATING_CLIENT_W_CITY	2	6.7	3.36	45.5136	< 2.2e-16	***
FLAG_DOCUMENT_3	1	5.3	5.32	72.1344	< 2.2e-16	***
AGE_GROUP	4	4.8	1.20	16.3134	2.285e-13	***
AMT_ANNUITY	1	4.7	4.71	63.8411	1.354e-15	***
EMPLOYMENT_YEAR	5	4.2	0.85	11.4648	4.349e-11	***
FLAG_PHONE	1	3.6	3.58	48.5626	3.207e-12	***
OWN_CAR_AGE	1	2.9	2.91	39.3902	3.476e-10	***
CNT_CHILDREN	1	2.7	2.70	36.5430	1.495e-09	***
DAYS_LAST_PHONE_CHANGE	1	2.5	2.55	34.5516	4.156e-09	***
NAME_TYPE_SUITE	7	2.5	0.35	4.7570	2.331e-05	***
FLAG_DOCUMENT_18	1	2.2	2.19	29.7456	4.931e-08	***
FLAG_DOCUMENT_16	1	2.0	2.03	27.5072	1.566e-07	***
WEEKDAY_APPR_PROCESS_START	6	1.7	0.28	3.7868	0.0008958	***
REG_CITY_NOT_WORK_CITY	1	1.6	1.59	21.5392	3.468e-06	***
WALLSMATERIAL_MODE	7	1.5	0.22	2.9613	0.0041933	**
HOUR_APPR_PROCESS_START	1	1.2	1.21	16.4369	5.031e-05	***
AMT_REQ_CREDIT_BUREAU_QRT	1	1.1	1.11	15.0008	0.0001075	***
APARTMENTS_AVG	1	1.0	1.04	14.0331	0.0001797	***
FLOORSMAX_AVG	1	1.0	0.97	13.1753	0.0002837	***
FLAG_DOCUMENT_5	1	0.9	0.93	12.6546	0.0003747	***
FLAG_DOCUMENT_2	1	0.9	0.92	12.5059	0.0004058	***
FONDKAPREMONT_MODE	4	0.9	0.23	3.0499	0.0159292	*
AMT_INCOME_RANGE	10	0.9	0.09	1.1889	0.2925494	
OBS_30_CNT_SOCIAL_CIRCLE	1	0.8	0.80	10.8412	0.0009928	***
YEARS_EMPLOYED	1	0.6	0.57	7.7489	0.0053749	**
AMT_REQ_CREDIT_BUREAU_WEEK	1	0.5	0.52	6.9830	0.0082291	**
YEARS_BUILD_AVG	1	0.5	0.48	6.4516	0.0110859	*
FLAG_DOCUMENT_14	1	0.5	0.47	6.4222	0.0112710	*
FLAG_EMAIL	1	0.5	0.45	6.1266	0.0133167	*
EMERGENCYSTATE_MODE	2	0.4	0.22	3.0362	0.0480175	*
FLAG_DOCUMENT_13	1	0.4	0.43	5.8133	0.0159061	*
FLAG_DOCUMENT_8	1	0.4	0.43	5.7729	0.0162760	*
FLAG_CONT_MOBILE	1	0.4	0.42	5.6940	0.0170236	*
YEARS_BEGINEXPLUATATION_AVG	1	0.4	0.36	4.8943	0.0269458	*
NONLIVINGAREA_MODE	1	0.3	0.26	3.5766	0.0585992	.
FLAG_DOCUMENT_15	1	0.2	0.23	3.1349	0.0766321	.
AMT_REQ_CREDIT_BUREAU_MON	1	0.2	0.23	3.1288	0.0769197	.
HOUSETYPE_MODE	3	0.2	0.07	0.9618	0.4096356	
COMMONAREA_AVG	1	0.2	0.19	2.5410	0.1109255	
FLAG_DOCUMENT_6	1	0.2	0.18	2.4017	0.1212067	

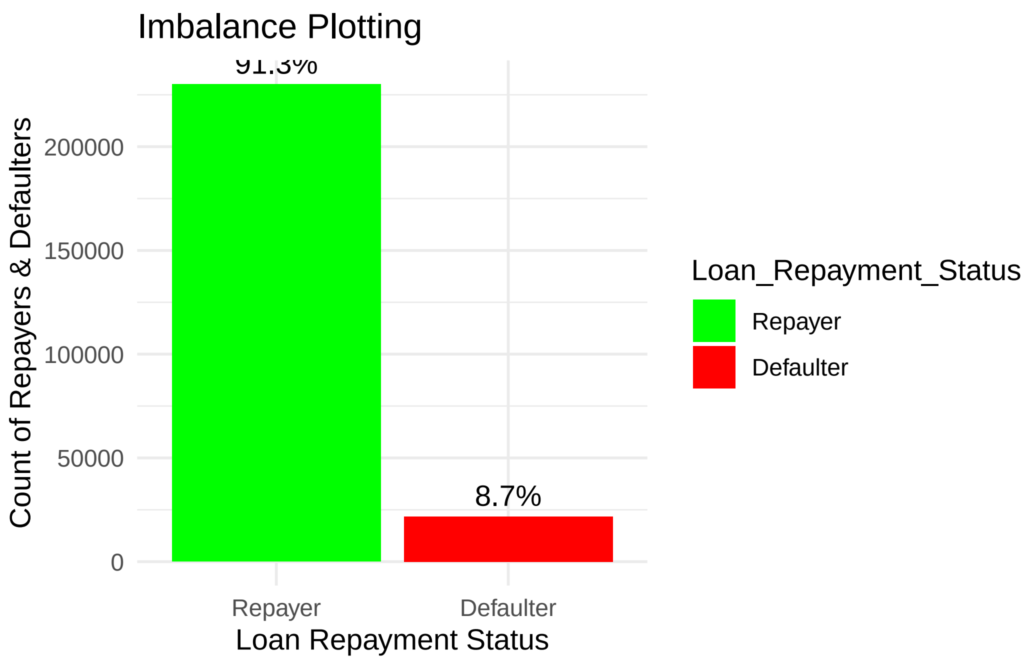
FLAG_OWN_REALTY	1	0.2	0.16	2.2253	0.1357670
FLAG_DOCUMENT_9	1	0.2	0.16	2.1759	0.1401869
AGE	1	0.1	0.13	1.8259	0.1766090
ELEVATORS_AVG	1	0.1	0.13	1.8056	0.1790337
DEF_60_CNT_SOCIAL_CIRCLE	1	0.1	0.13	1.7604	0.1845784
FLAG_DOCUMENT_17	1	0.1	0.13	1.7157	0.1902462
BASEMENTAREA_AVG	1	0.1	0.12	1.6208	0.2029832
LIVINGAPARTMENTS_MODE	1	0.1	0.11	1.4872	0.2226486
LIVE_REGION_NOT_WORK_REGION	1	0.1	0.10	1.4037	0.2361015
NONLIVINGAPARTMENTS_MODE	1	0.1	0.10	1.3907	0.2382909
COMMONAREA_MEDI	1	0.1	0.10	1.3692	0.2419439
ENTRANCES_AVG	1	0.1	0.10	1.3629	0.2430302
LIVINGAPARTMENTS_MEDI	1	0.1	0.10	1.3367	0.2476193
LIVE_CITY_NOT_WORK_CITY	1	0.1	0.09	1.2457	0.2643816
LANDAREA_MODE	1	0.1	0.09	1.2198	0.2694062
LANDAREA_MEDI	1	0.1	0.08	1.0380	0.3082776
YEARS_BEGINEXPLUATATION_MEDI	1	0.1	0.08	1.0232	0.3117562
LANDAREA_AVG	1	0.1	0.07	0.9787	0.3225119
OBS_60_CNT_SOCIAL_CIRCLE	1	0.1	0.07	0.9415	0.3318868
FLAG_DOCUMENT_11	1	0.1	0.06	0.8306	0.3620880
ENTRANCES_MODE	1	0.1	0.06	0.8116	0.3676632
BASEMENTAREA_MEDI	1	0.1	0.06	0.7902	0.3740473
FLAG_DOCUMENT_19	1	0.1	0.05	0.7154	0.3976610
FLAG_DOCUMENT_10	1	0.0	0.04	0.5863	0.4438568
LIVINGAREA_MEDI	1	0.0	0.04	0.5707	0.4499968
ELEVATORS_MODE	1	0.0	0.04	0.5581	0.4550200
SK_ID_CURR	1	0.0	0.04	0.5548	0.4563665
YEARS_BUILD_MEDI	1	0.0	0.03	0.4680	0.4939106
FLAG_DOCUMENT_4	1	0.0	0.03	0.4223	0.5158133
NONLIVINGAREA_AVG	1	0.0	0.02	0.3273	0.5672391
FLAG_DOCUMENT_20	1	0.0	0.02	0.3222	0.5702855
LIVINGAREA_AVG	1	0.0	0.02	0.3196	0.5718629
NONLIVINGAPARTMENTS_MEDI	1	0.0	0.02	0.2963	0.5861934
APARTMENTS_MODE	1	0.0	0.02	0.2934	0.5880228
FLOORSMAX_MODE	1	0.0	0.02	0.2931	0.5882637
FLAG_MOBIL	1	0.0	0.02	0.2601	0.6100336
ENTRANCES_MEDI	1	0.0	0.01	0.2030	0.6522739
FLOORSMAX_MEDI	1	0.0	0.01	0.2024	0.6527951
FLAG_DOCUMENT_7	1	0.0	0.01	0.1791	0.6721436
YEARS_BUILD_MODE	1	0.0	0.01	0.1750	0.6757463
AMT_REQ_CREDIT_BUREAU_YEAR	1	0.0	0.01	0.1715	0.6787674
FLAG_DOCUMENT_21	1	0.0	0.01	0.1643	0.6851913
FLOORSMIN_AVG	1	0.0	0.01	0.1481	0.7003682

```
LIVINGAREA_MODE                   1      0.0     0.01     0.1242 0.7245029
TOTALAREA_MODE                    1      0.0     0.01     0.0983 0.7538283
FLAG_DOCUMENT_12                  1      0.0     0.01     0.0856 0.7698666
FLAG_EMP_PHONE                    1      0.0     0.01     0.0801 0.7770995
YEARS_BEGINEXPLUATATION_MODE      1      0.0     0.01     0.0787 0.7790962
ELEVATORS_MEDI                    1      0.0     0.00     0.0570 0.8112549
NONLIVINGAREA_MEDI                1      0.0     0.00     0.0412 0.8391866
FLOORSMIN_MODE                    1      0.0     0.00     0.0403 0.8408954
APARTMENTS_MEDI                   1      0.0     0.00     0.0268 0.8698642
REG_REGION_NOT_LIVE_REGION        1      0.0     0.00     0.0207 0.8857061
LIVINGAPARTMENTS_AVG              1      0.0     0.00     0.0157 0.9003191
AMT_REQ_CREDIT_BUREAU_HOUR        1      0.0     0.00     0.0137 0.9066605
FLOORSMIN_MEDI                    1      0.0     0.00     0.0099 0.9207980
COMMONAREA_MODE                   1      0.0     0.00     0.0069 0.9340282
BASEMENTAREA_MODE                 1      0.0     0.00     0.0039 0.9504972
AMT_REQ_CREDIT_BUREAU_DAY         1      0.0     0.00     0.0004 0.9844991
REG_REGION_NOT_WORK_REGION        1      0.0     0.00     0.0001 0.9924951
NONLIVINGAPARTMENTS_AVG           1      0.0     0.00     0.0001 0.9926282
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

EXT_SOURCE_3 AMT_GOODS_PRICE FLAG_OWN_CAR EXT_SOURCE_1 CODE_GENDER DAYS_BIRTH NAME_EDUCATION_TYPE DAYS_EMPLOYED AMT_CREDIT NAME_INCOME_TYPE EXT_SOURCE_2 NAME_CONTRACT_TYPE OCCUPATION_TYPE NAME_FAMILY_STATUS AMT_CREDIT_RANGE

```r
# Contar la frecuencia de cada categoría en la variable TARGET
Imbalance <- as.data.frame(table(datos$TARGET))
colnames(Imbalance) <- c("Loan_Repayment_Status", "Count")

# Reemplazar valores 0 y 1 con etiquetas significativas
Imbalance$Loan_Repayment_Status <- factor(Imbalance$Loan_Repayment_Status,
                                          levels = c(0,1),
                                          labels = c("Repayer", "Defaulter"))

# Calcular el porcentaje y crear la etiqueta
Imbalance$Percent <- Imbalance$Count / sum(Imbalance$Count) * 100
Imbalance$Label <- paste0(round(Imbalance$Percent, 1), "%")

# Crear el gráfico de barras con etiquetas de porcentaje
ggplot(Imbalance, aes(x = Loan_Repayment_Status, y = Count, fill = Loan_Repayment_Status)) +
  geom_bar(stat = "identity") +
```

```
  geom_text(aes(label = Label), vjust = -0.5) +  # Añadir etiquetas encima de las barras
  scale_fill_manual(values = c("green", "red")) +
  labs(title = "Imbalance Plotting",
       x = "Loan Repayment Status",
       y = "Count of Repayers & Defaulters") +
  theme_minimal()
```



definimos una funcion que dado una variable nos de un histograma con los pagos devueltos y no devueltos segun la variable

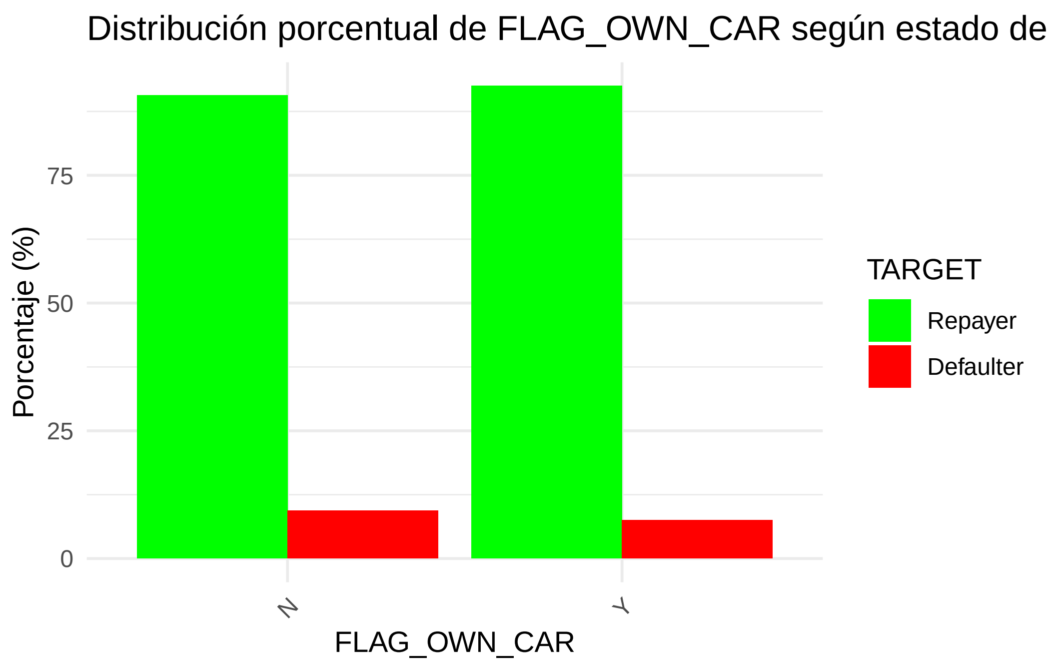
```
# Definir la función
plot_loan_repayment <- function(df, variable) {
  # Verificar que la variable existe
  if (!(variable %in% colnames(df))) {
    stop("La variable especificada no existe en el dataframe.")
  }

  # Crear dataframe de trabajo
  df_plot <- df[, c(variable, "TARGET")]

  # Convertir TARGET a factor con etiquetas
  df_plot$TARGET <- factor(df_plot$TARGET, levels = c(0, 1), labels = c("Repayer", "Defaulte
```

```
  # Calcular proporciones por categoría
  df_prop <- df_plot %>%
    group_by(.data[[variable]], TARGET) %>%
    summarise(n = n(), .groups = "drop") %>%
    group_by(.data[[variable]]) %>%
    mutate(pct = n / sum(n) * 100)

  # Graficar con porcentajes
  ggplot(df_prop, aes_string(x = variable, y = "pct", fill = "TARGET")) +
    geom_bar(stat = "identity", position = "dodge") +
    labs(
      title = paste("Distribución porcentual de", variable, "según estado de pago"),
      x = variable, y = "Porcentaje (%)"
    ) +
    scale_fill_manual(values = c("green", "red")) +
    scale_x_discrete(guide = guide_axis(angle = 45)) +
    theme_minimal()
}
```

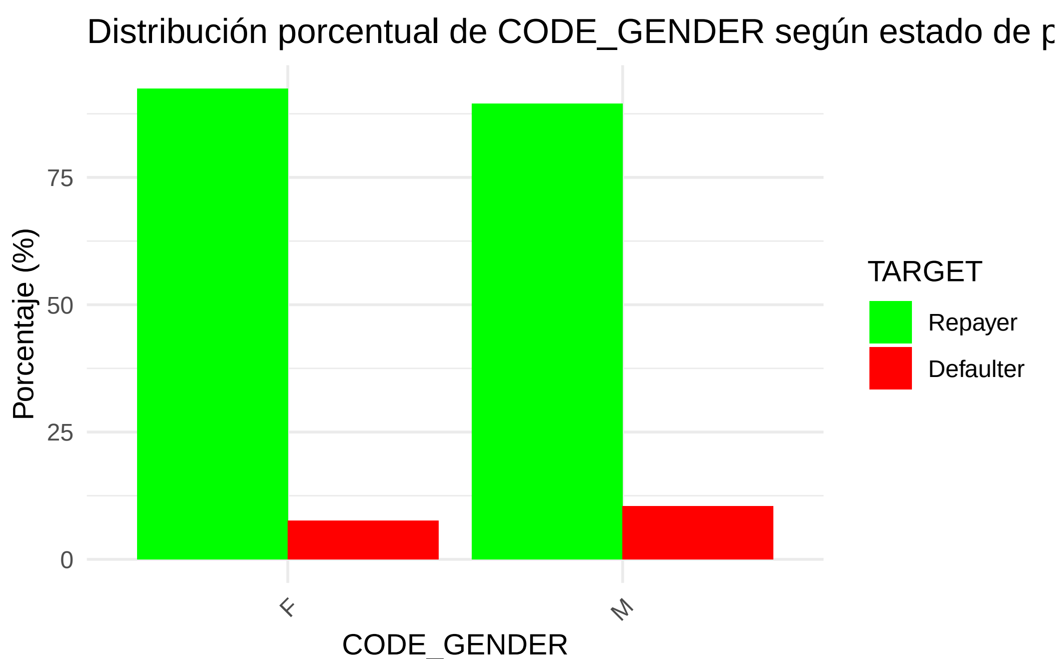
Graficar variables categoricas

```
# Ejemplo de uso con la variable FLAG_OWN_CAR
plot_loan_repayment(datos, "FLAG_OWN_CAR")
```

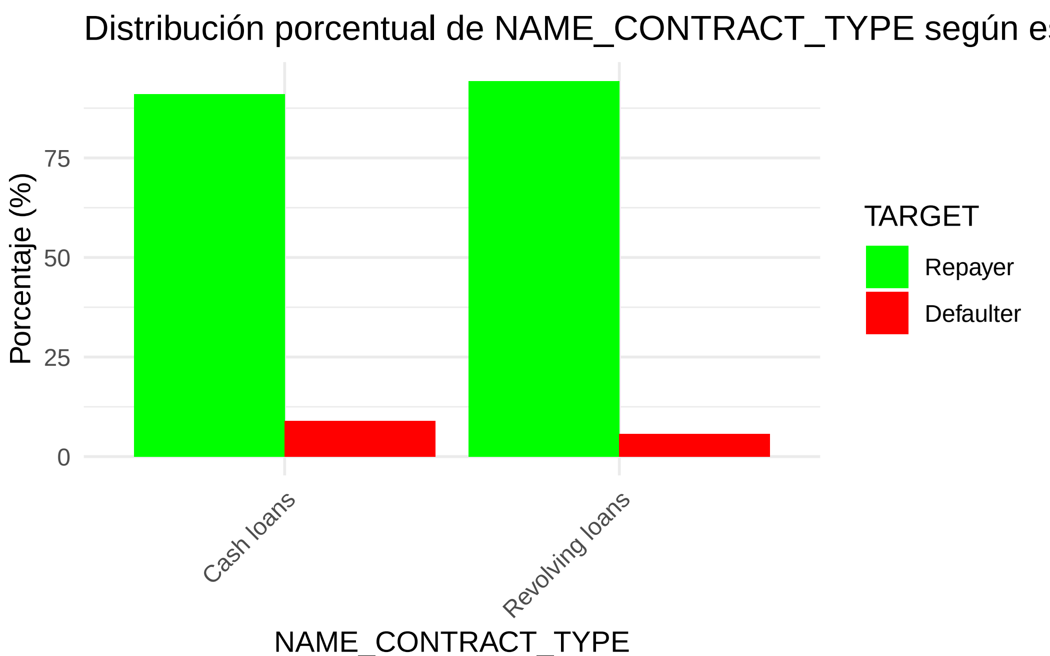
```
Warning: `aes_string()` was deprecated in ggplot2 3.0.0.
i Please use tidy evaluation idioms with `aes()`.
i See also `vignette("ggplot2-in-packages")` for more information.
```



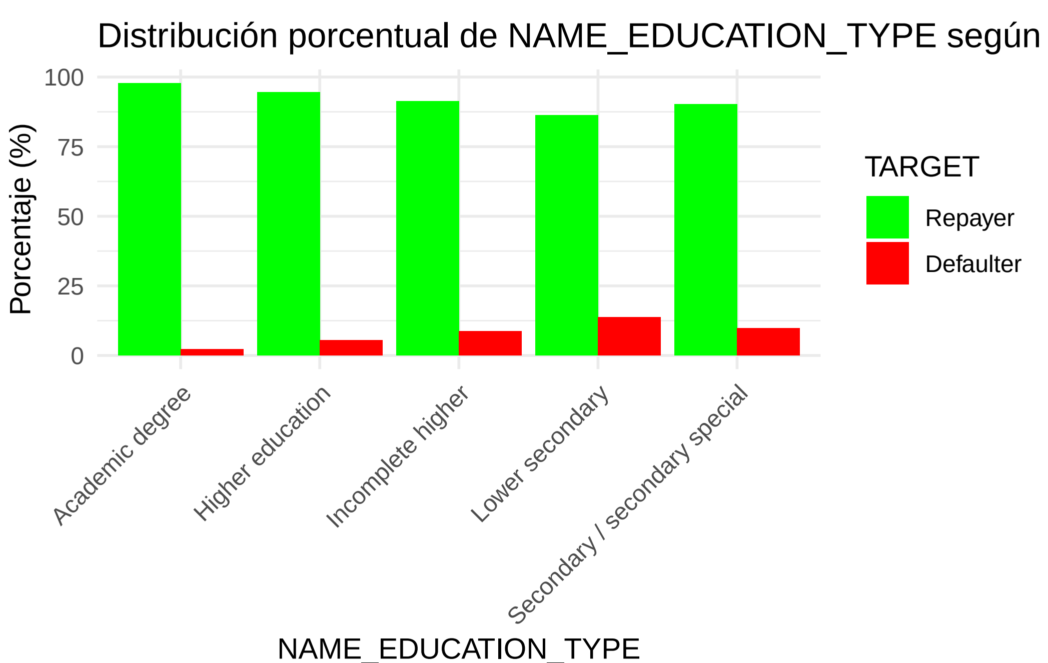

```
plot_loan_repayment(datos, "CODE_GENDER")
```



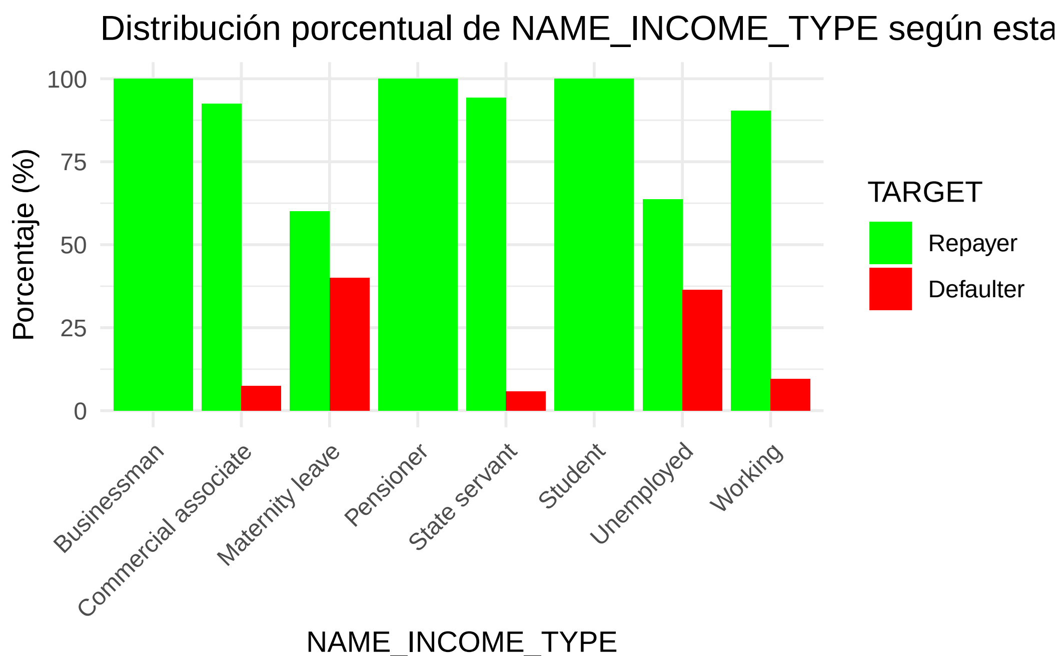
```r
plot_loan_repayment(datos, "NAME_CONTRACT_TYPE")
```



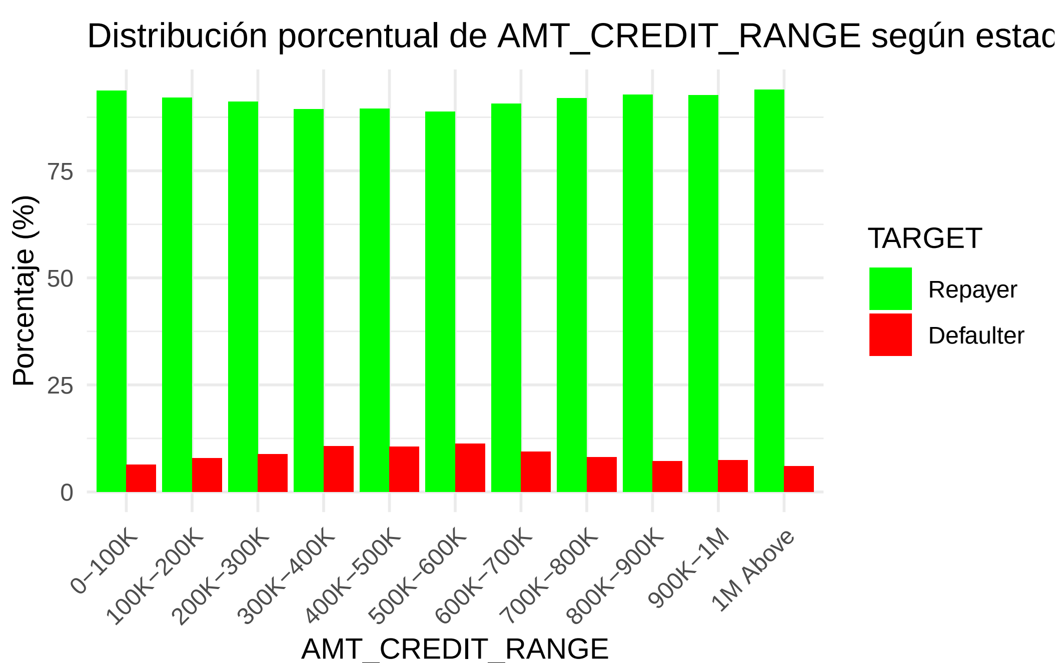

```r
plot_loan_repayment(datos, "NAME_EDUCATION_TYPE")
```



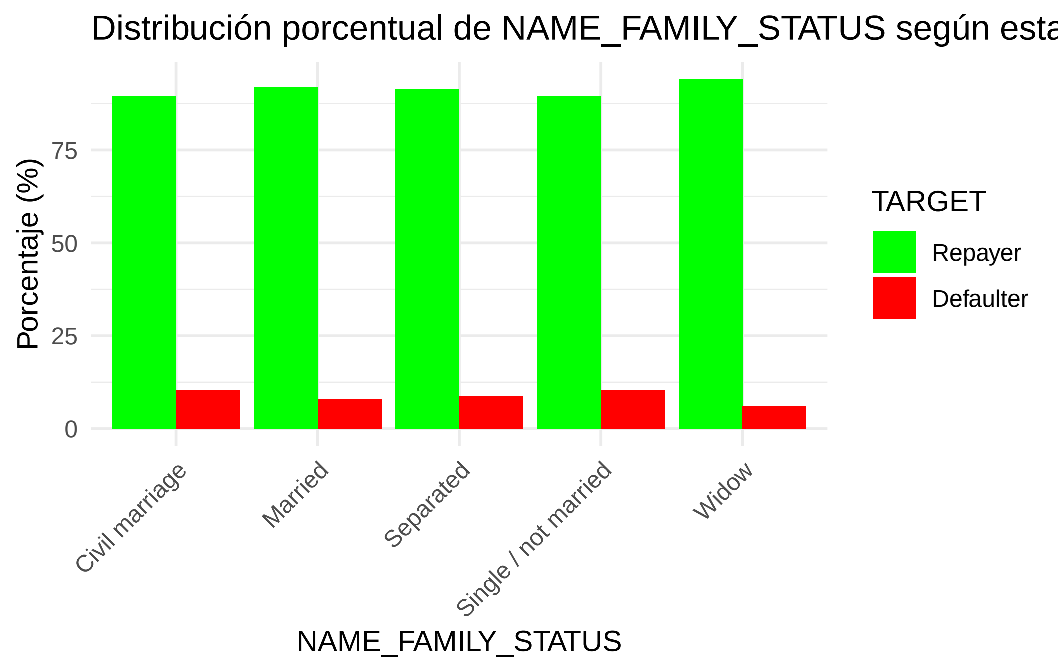
```
plot_loan_repayment(datos, "NAME_INCOME_TYPE")
```



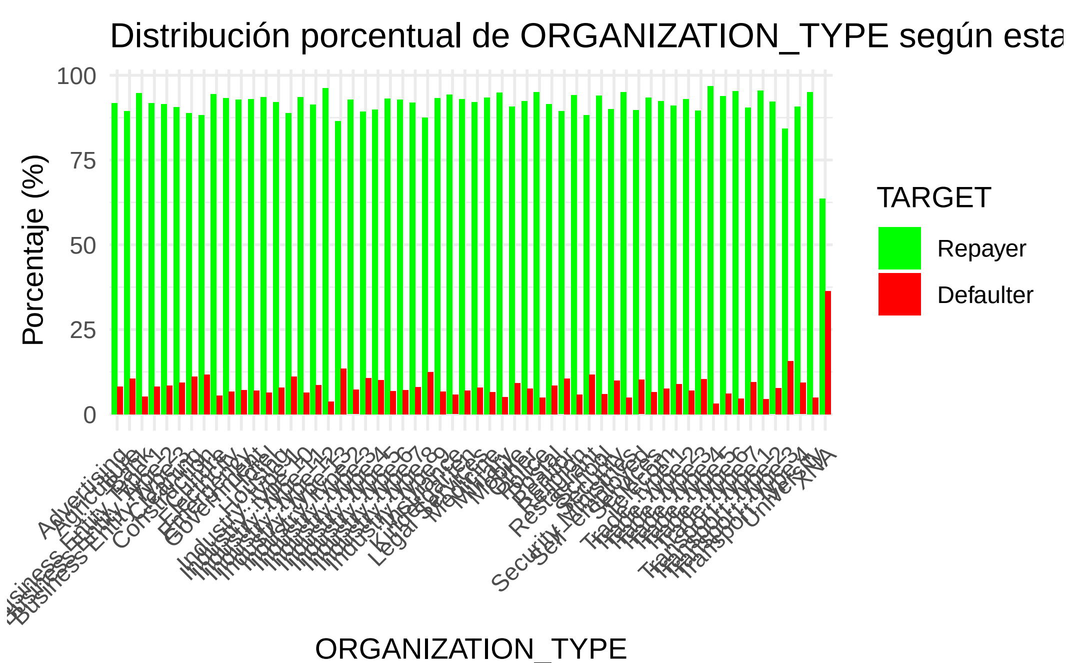

```
plot_loan_repayment(datos, "AMT_CREDIT_RANGE")
```



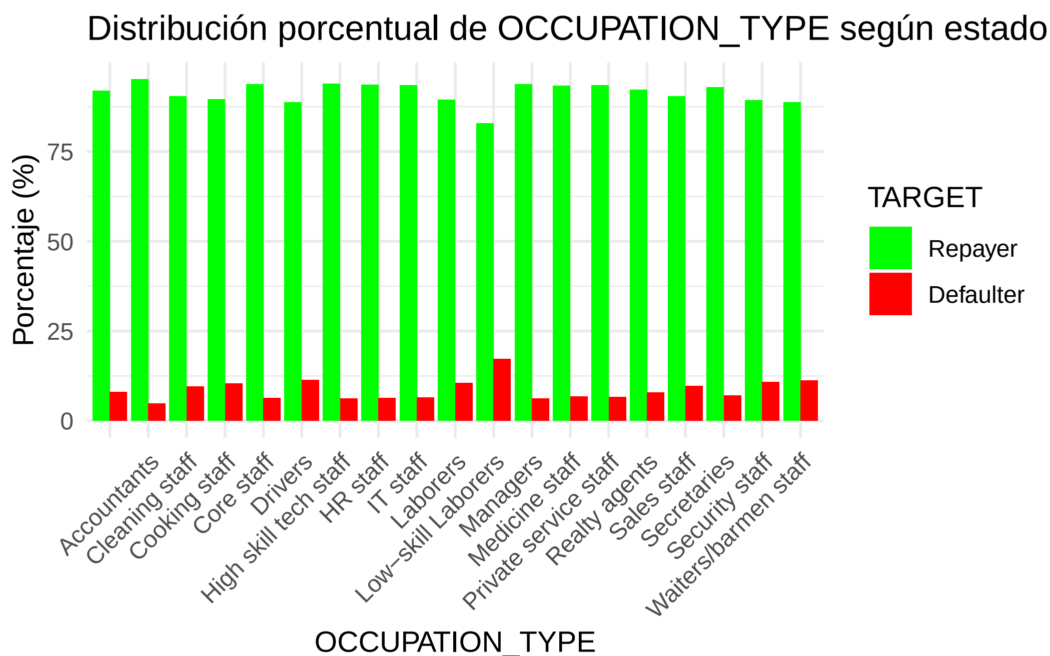
```r
plot_loan_repayment(datos, "NAME_FAMILY_STATUS")
```



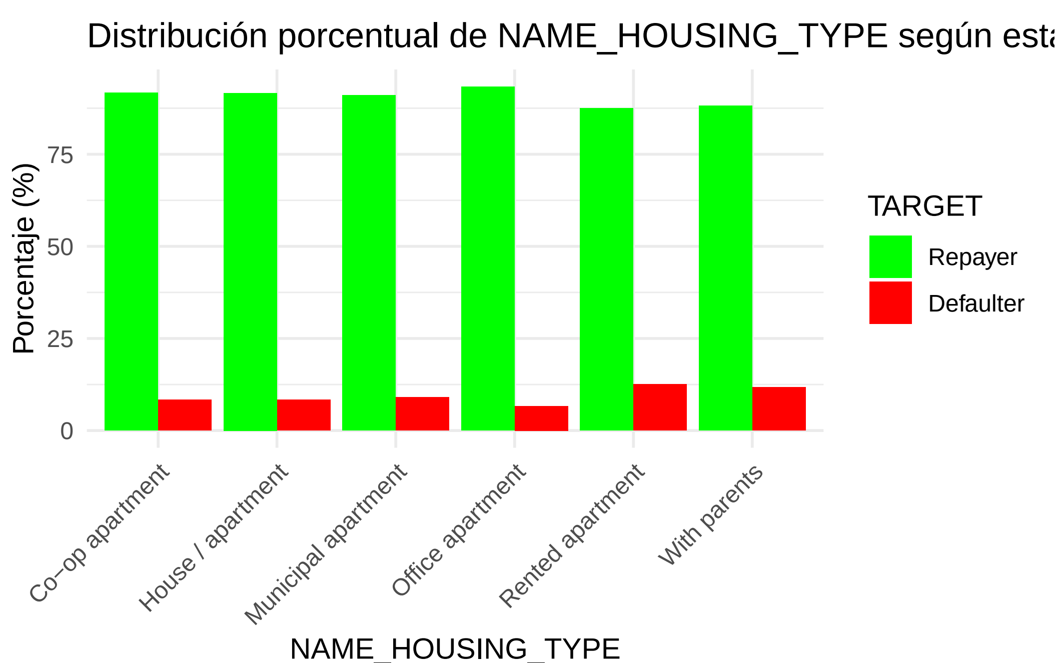

```r
plot_loan_repayment(datos, "ORGANIZATION_TYPE")
```



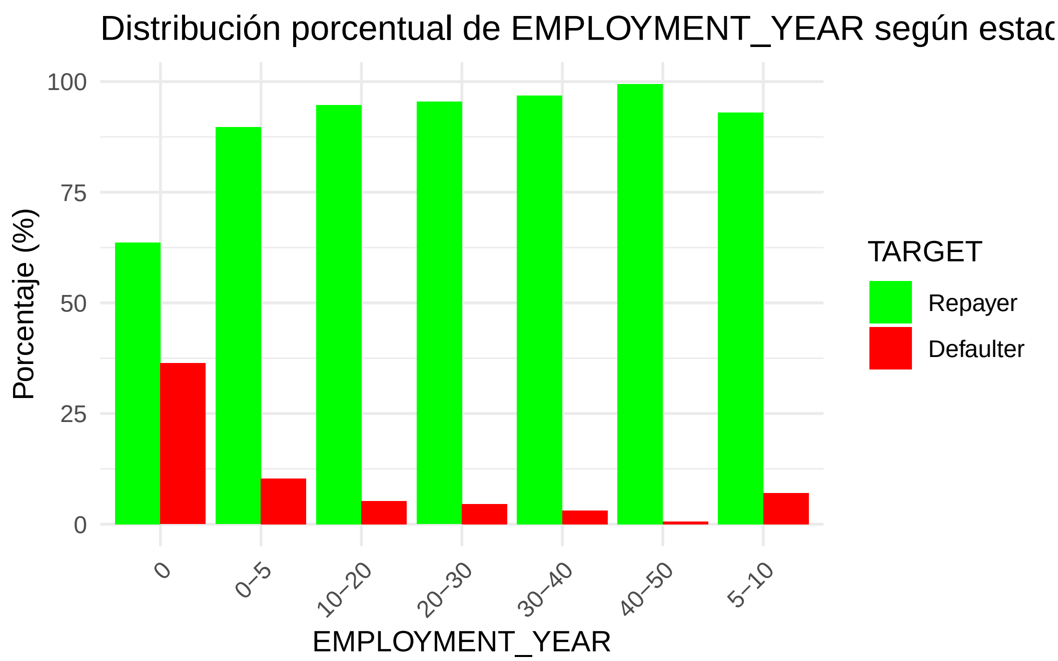
```r
plot_loan_repayment(datos, "OCCUPATION_TYPE")
```



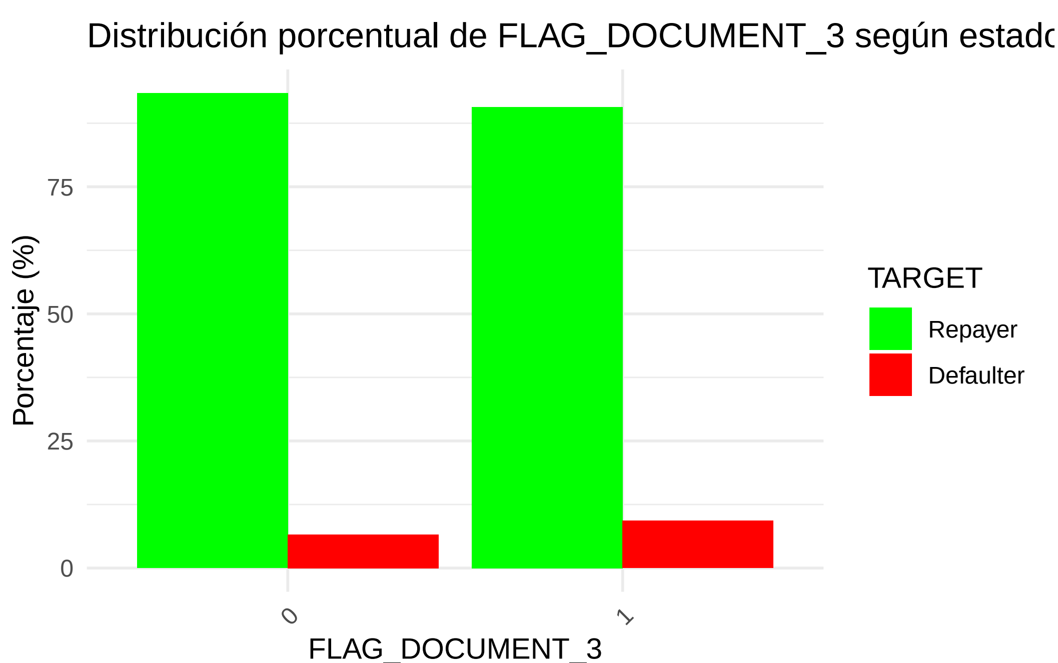

```r
plot_loan_repayment(datos, "NAME_HOUSING_TYPE")
```



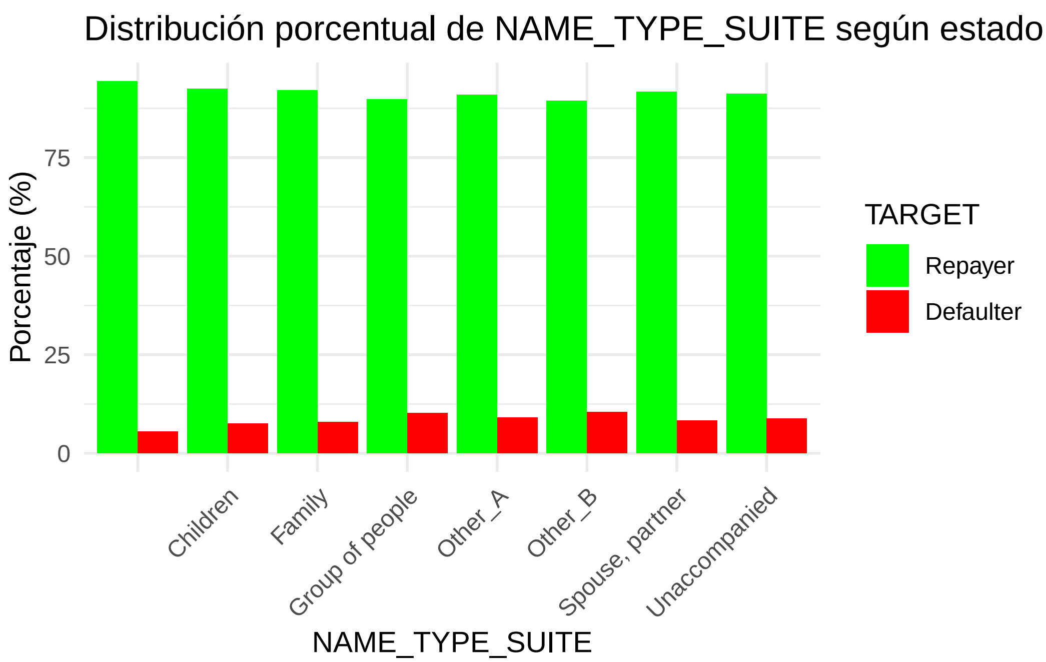
```
plot_loan_repayment(datos, "EMPLOYMENT_YEAR")
```




```
plot_loan_repayment(datos, "FLAG_DOCUMENT_3")
```



```r
plot_loan_repayment(datos, "NAME_TYPE_SUITE")
```



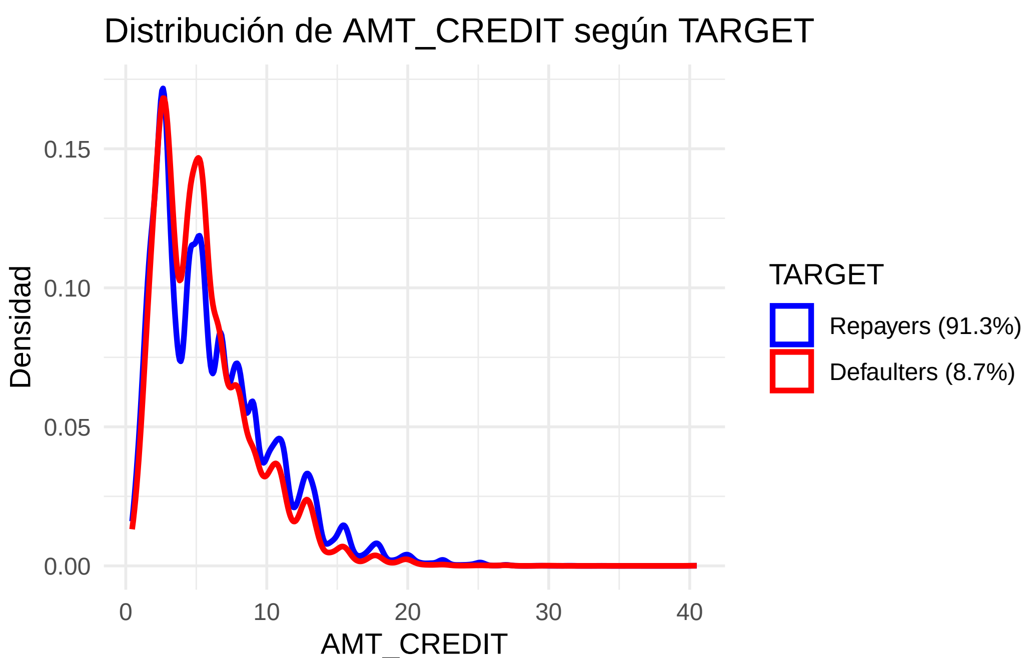
Graficar variables continuas

```r
graficar_variable <- function(data, variable) {
  # Calcular los porcentajes por clase
  porcentajes <- data %>%
    group_by(TARGET) %>%
    summarise(n = n()) %>%
    mutate(porc = paste0(round(100 * n / sum(n), 1), "%"))

  # Crear etiquetas personalizadas
  levels_target <- sort(unique(data$TARGET))
  etiquetas <- paste0(
    ifelse(levels_target == 0, "Repayers", "Defaulters"),
    " (", porcentajes$porc, ")"
  )

  # Graficar con los porcentajes en la leyenda
  ggplot(data, aes(x = .data[[variable]], color = as.factor(TARGET))) +
```

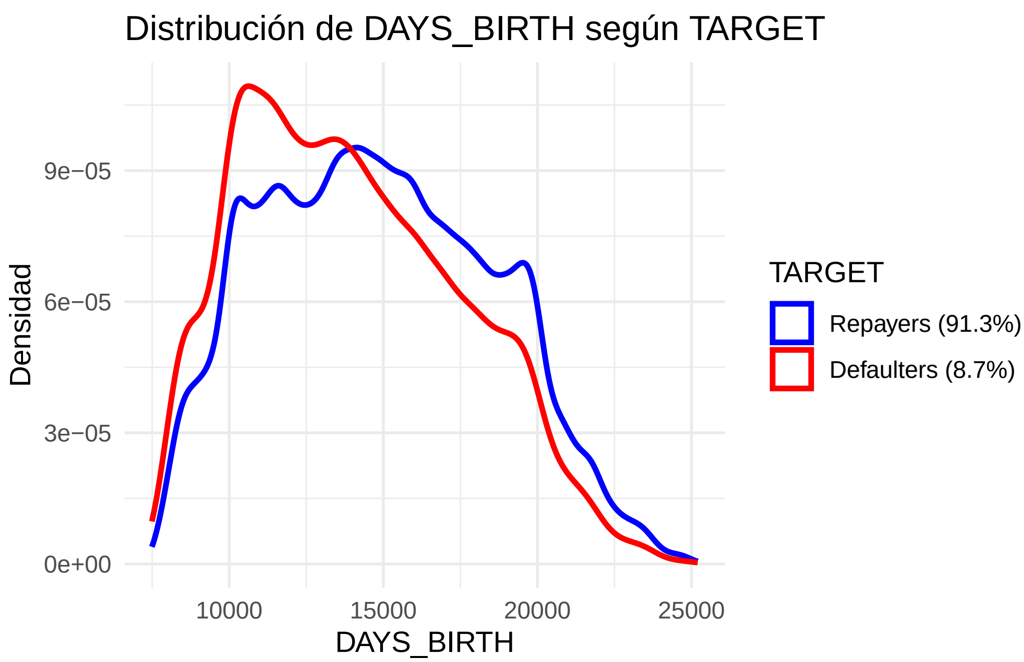
```
    geom_density(size = 1) +
    labs(x = variable, y = "Densidad", title = paste("Distribución de", variable, "según TARG
    scale_color_manual(
      values = c("blue", "red"),
      labels = etiquetas,
      name = "TARGET"
    ) +
    theme_minimal()
}
```

```
# Ejemplo de uso con la variable "AMT_CREDIT"
graficar_variable(datos, "AMT_CREDIT")
```

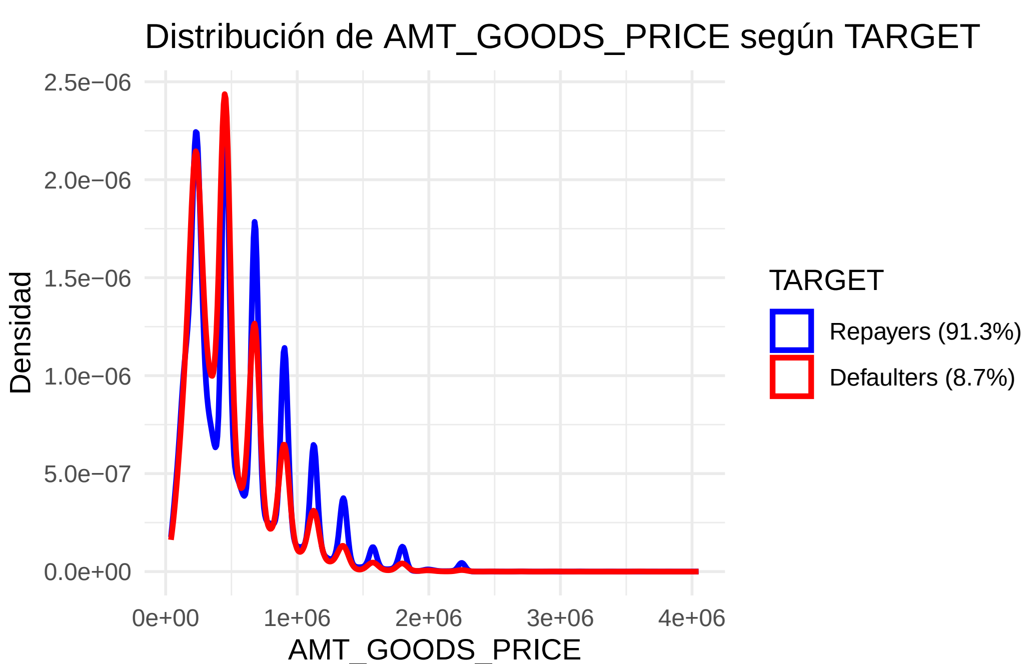
```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead.
```



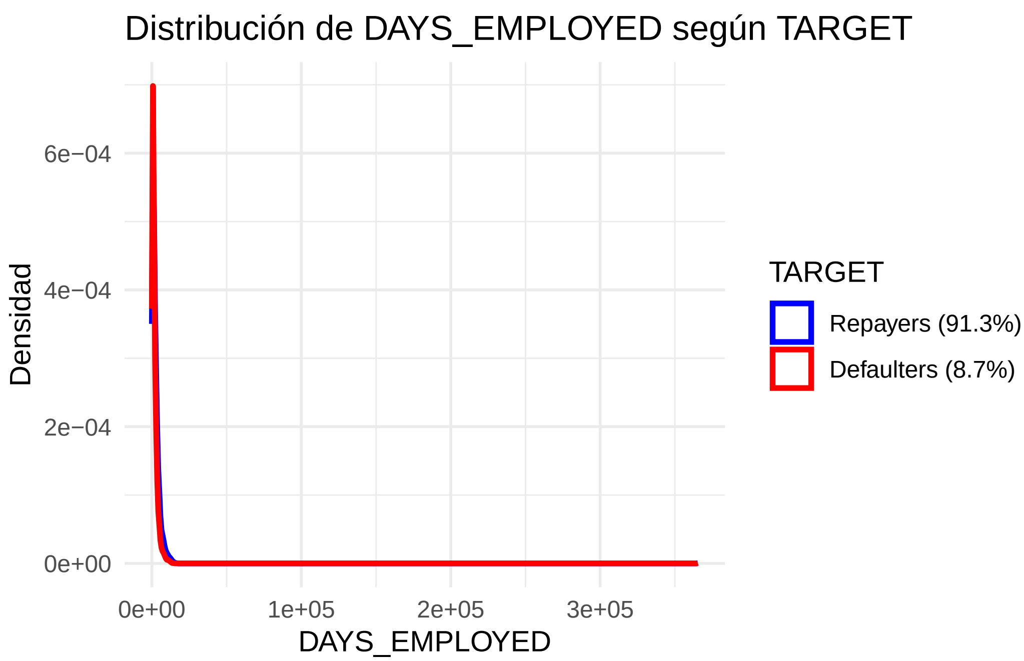

```
# Ejemplo de uso con la variable "AMT_CREDIT"
graficar_variable(datos, "DAYS_BIRTH")
```



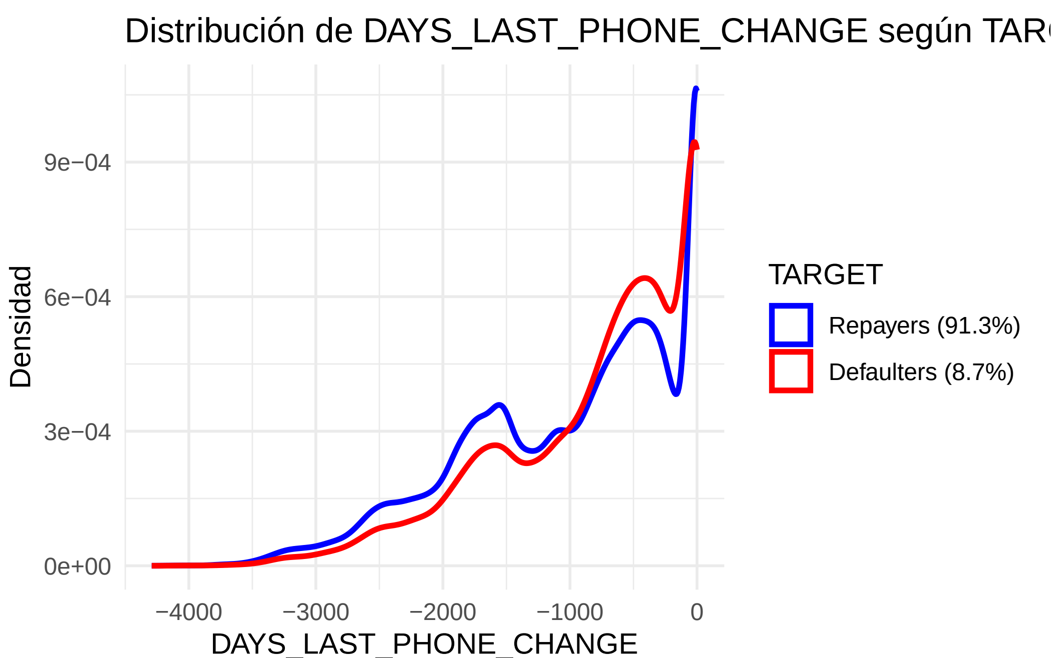

```r
graficar_variable(datos, "AMT_GOODS_PRICE")
```



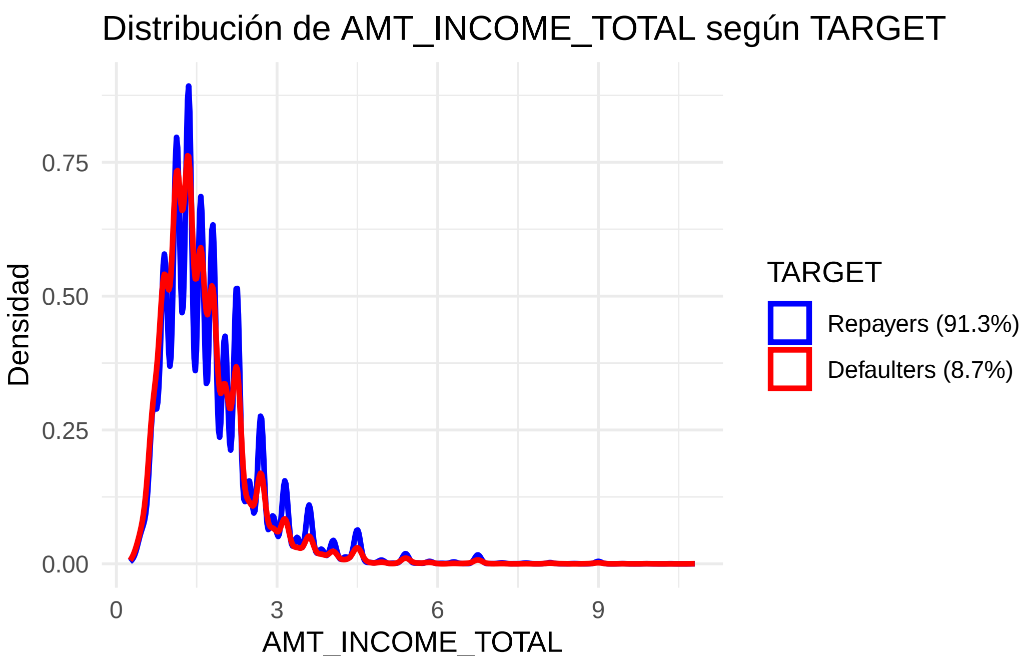
```r
graficar_variable(datos, "DAYS_EMPLOYED")
```



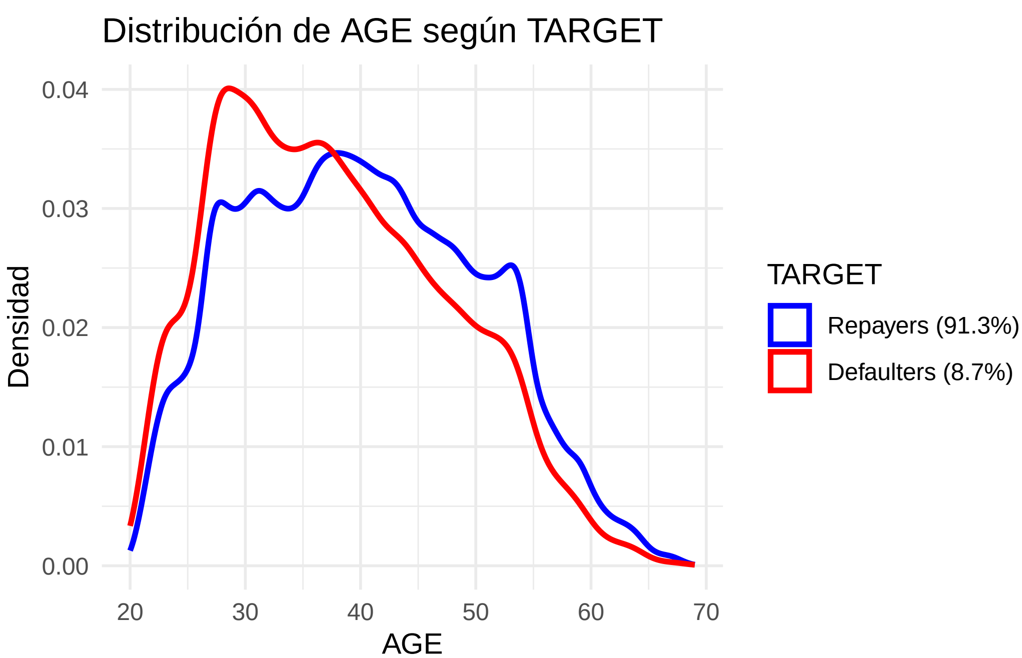

```r
graficar_variable(datos, "DAYS_LAST_PHONE_CHANGE")
```



```r
graficar_variable(datos, "AMT_INCOME_TOTAL")
```




```r
graficar_variable(datos, "AGE")
```



Guardar base de datos depurada para modelos

primero eliminamos las variables menos significativas, y nos quedamos con las mas significativas

```
variables_significativas <- c("EXT_SOURCE_3", "EXT_SOURCE_2", "DAYS_BIRTH", "AMT_GOODS_PRICE
datos<- datos[,variables_significativas]
# eliminamos los NA faltantes, estos se deben a valores atipicos que dan problemas
#guardamos en una base de datos los datos, asi podemos seguir con el TFG sin saturar el PC
save(datos,file="DatosDepurados.RDa")
```