GrupoBimbo-InventoryDemanda

Hiago W. Petris

7/17/2021

Carregando dados e pacotes

```
library(readr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

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

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

```
library(caret)
```

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

```
library(ggplot2)
library(mltools)
library(e1071)
```

```
##
## Attaching package: 'e1071'
```

```
## The following object is masked from 'package:mltools':
##
##     skewness
```

```
library(stringr)

# Lendo os arquivos
df <- read_csv('C:/Users/hiago/Downloads/datasets/train.csv')
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Semana = col_double(),
##   Agencia_ID = col_double(),
##   Canal_ID = col_double(),
##   Ruta_SAK = col_double(),
##   Cliente_ID = col_double(),
```

```
##   Producto_ID = col_double(),
##   Venta_uni_hoy = col_double(),
##   Venta_hoy = col_double(),
##   Dev_uni_proxima = col_double(),
##   Dev_proxima = col_double(),
##   Demanda_uni_equil = col_double()
## )
```

```
df_cliente <- read_csv('C:/Users/hiago/Downloads/datasets/cliente_tabla.csv')
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Cliente_ID = col_double(),
##   NombreCliente = col_character()
## )
```

```
df_producto <- read_csv('C:/Users/hiago/Downloads/datasets/producto_tabla.csv')
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Producto_ID = col_double(),
##   NombreProducto = col_character()
## )
```

```
df_town <- read_csv('C:/Users/hiago/Downloads/datasets/town_state.csv')
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Agencia_ID = col_double(),
##   Town = col_character(),
##   State = col_character()
## )
```

```
df_test <- read_csv('C:/Users/hiago/Downloads/datasets/test.csv')
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   id = col_double(),
##   Semana = col_double(),
##   Agencia_ID = col_double(),
##   Canal_ID = col_double(),
##   Ruta_SAK = col_double(),
##   Cliente_ID = col_double(),
##   Producto_ID = col_double()
## )
```

Pré processamento

```
head(df)
```

```
## # A tibble: 6 x 11
##   Semana Agencia_ID Canal_ID Ruta_SAK Cliente_ID Producto_ID Venta_uni_hoy
##    <dbl>      <dbl>    <dbl>    <dbl>      <dbl>       <dbl>         <dbl>
```

```
## 1      3       1110        7     3301      15766        1212            3
## 2      3       1110        7     3301      15766        1216            4
## 3      3       1110        7     3301      15766        1238            4
## 4      3       1110        7     3301      15766        1240            4
## 5      3       1110        7     3301      15766        1242            3
## 6      3       1110        7     3301      15766        1250            5
## # ... with 4 more variables: Venta_hoy <dbl>, Dev_uni_proxima <dbl>,
## #   Dev_proxima <dbl>, Demanda_uni_equil <dbl>
```

```
str(df)
```

```
## spec_tbl_df[,11] [74,180,464 x 11] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Semana           : num [1:74180464] 3 3 3 3 3 3 3 3 3 3 ...
##  $ Agencia_ID       : num [1:74180464] 1110 1110 1110 1110 1110 1110 1110 1110 1110 1110 ...
##  $ Canal_ID         : num [1:74180464] 7 7 7 7 7 7 7 7 7 7 ...
##  $ Ruta_SAK         : num [1:74180464] 3301 3301 3301 3301 3301 ...
##  $ Cliente_ID       : num [1:74180464] 15766 15766 15766 15766 15766 ...
##  $ Producto_ID      : num [1:74180464] 1212 1216 1238 1240 1242 ...
##  $ Venta_uni_hoy    : num [1:74180464] 3 4 4 4 3 5 3 6 4 6 ...
##  $ Venta_hoy        : num [1:74180464] 25.1 33.5 39.3 33.5 22.9 ...
##  $ Dev_uni_proxima  : num [1:74180464] 0 0 0 0 0 0 0 0 0 0 ...
##  $ Dev_proxima      : num [1:74180464] 0 0 0 0 0 0 0 0 0 0 ...
##  $ Demanda_uni_equil: num [1:74180464] 3 4 4 4 3 5 3 6 4 6 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Semana = col_double(),
##   ..   Agencia_ID = col_double(),
##   ..   Canal_ID = col_double(),
##   ..   Ruta_SAK = col_double(),
##   ..   Cliente_ID = col_double(),
##   ..   Producto_ID = col_double(),
##   ..   Venta_uni_hoy = col_double(),
##   ..   Venta_hoy = col_double(),
##   ..   Dev_uni_proxima = col_double(),
##   ..   Dev_proxima = col_double(),
##   ..   Demanda_uni_equil = col_double()
##   .. )
```

```
head(df_cliente)
```

```
## # A tibble: 6 x 2
##   Cliente_ID NombreCliente
##        <dbl> <chr>
## 1          0 SIN NOMBRE
## 2          1 OXXO XINANTECATL
## 3          2 SIN NOMBRE
## 4          3 EL MORENO
## 5          4 SDN SER  DE ALIM  CUERPO SA CIA  DE INT
## 6          4 SDN SER DE ALIM CUERPO SA CIA DE INT
```

```
str(df_cliente)
```

```
## spec_tbl_df[,2] [935,362 x 2] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Cliente_ID   : num [1:935362] 0 1 2 3 4 4 5 6 7 8 ...
##  $ NombreCliente: chr [1:935362] "SIN NOMBRE" "OXXO XINANTECATL" "SIN NOMBRE" "EL MORENO" ...
##  - attr(*, "spec")=
```

```
##   .. cols(
##   ..   Cliente_ID = col_double(),
##   ..   NombreCliente = col_character()
##   .. )
```

```
head(df_producto)
```

```
## # A tibble: 6 x 2
##   Producto_ID NombreProducto
##         <dbl> <chr>
## 1           0 NO IDENTIFICADO 0
## 2           9 Capuccino Moka 750g NES 9
## 3          41 Bimbollos Ext sAjonjoli 6p 480g BIM 41
## 4          53 Burritos Sincro 170g CU LON 53
## 5          72 Div Tira Mini Doradita 4p 45g TR 72
## 6          73 Pan Multigrano Linaza 540g BIM 73
```

```
str(df_producto)
```

```
## spec_tbl_df[,2] [2,592 x 2] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Producto_ID   : num [1:2592] 0 9 41 53 72 73 98 99 100 106 ...
##  $ NombreProducto: chr [1:2592] "NO IDENTIFICADO 0" "Capuccino Moka 750g NES 9" "Bimbollos Ext sAjon
##  - attr(*, "spec")=
##   .. cols(
##   ..   Producto_ID = col_double(),
##   ..   NombreProducto = col_character()
##   .. )
```

```
head(df_town)
```

```
## # A tibble: 6 x 3
##   Agencia_ID Town                 State
##        <dbl> <chr>                <chr>
## 1       1110 2008 AG. LAGO FILT   MÉXICO, D.F.
## 2       1111 2002 AG. AZCAPOTZALCO MÉXICO, D.F.
## 3       1112 2004 AG. CUAUTITLAN  ESTADO DE MÉXICO
## 4       1113 2008 AG. LAGO FILT   MÉXICO, D.F.
## 5       1114 2029 AG.IZTAPALAPA 2 MÉXICO, D.F.
## 6       1116 2011 AG. SAN ANTONIO MÉXICO, D.F.
```

```
str(df_town)
```

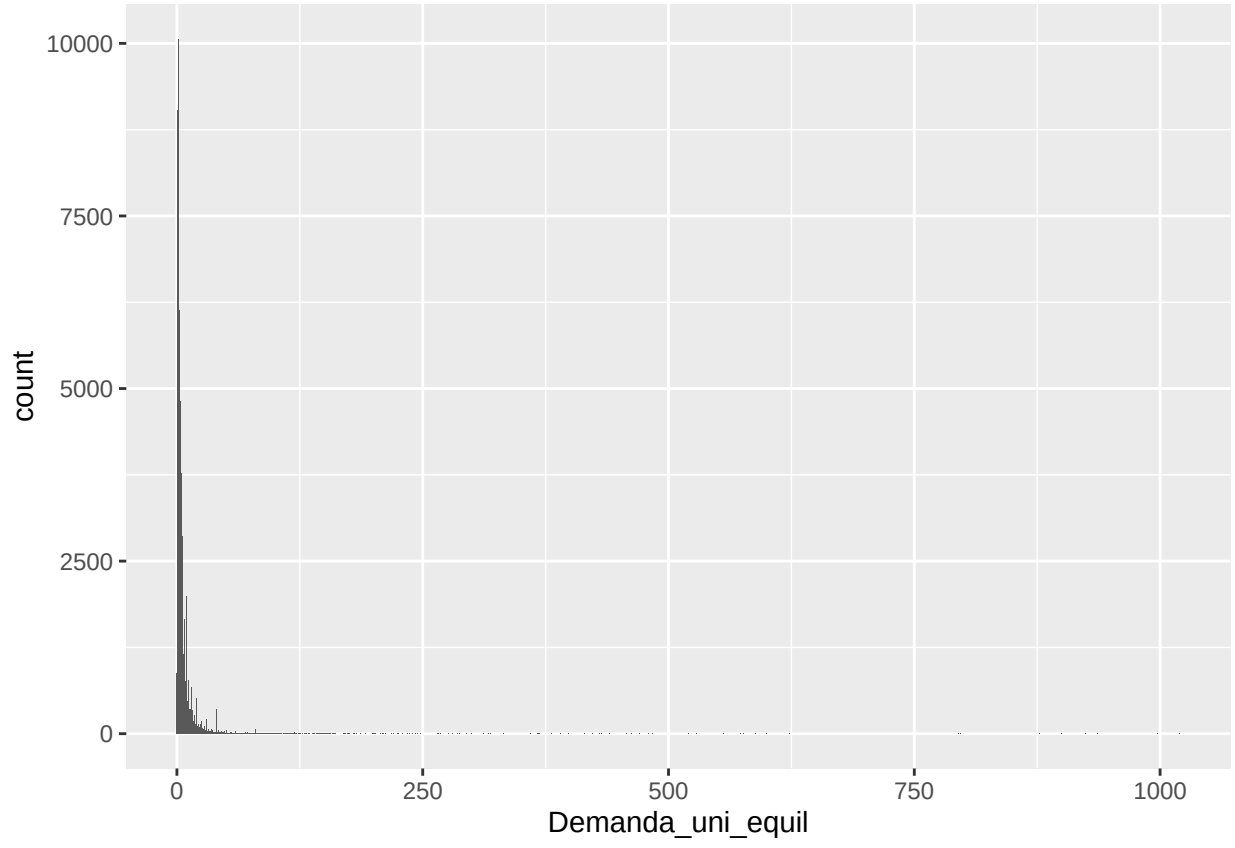
```
## spec_tbl_df[,3] [790 x 3] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Agencia_ID: num [1:790] 1110 1111 1112 1113 1114 ...
##  $ Town      : chr [1:790] "2008 AG. LAGO FILT" "2002 AG. AZCAPOTZALCO" "2004 AG. CUAUTITLAN" "2008 A
##  $ State     : chr [1:790] "MÉXICO, D.F." "MÉXICO, D.F." "ESTADO DE MÉXICO" "MÉXICO, D.F." ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Agencia_ID = col_double(),
##   ..   Town = col_character(),
##   ..   State = col_character()
##   .. )
```

```
# Os datasets cliente, producto e town não serão relevantes para o processo de predição
# pois contém apenas descrições de Ids referenciados no DF principal.
# Porém podem ser úteis no processo de análise
```

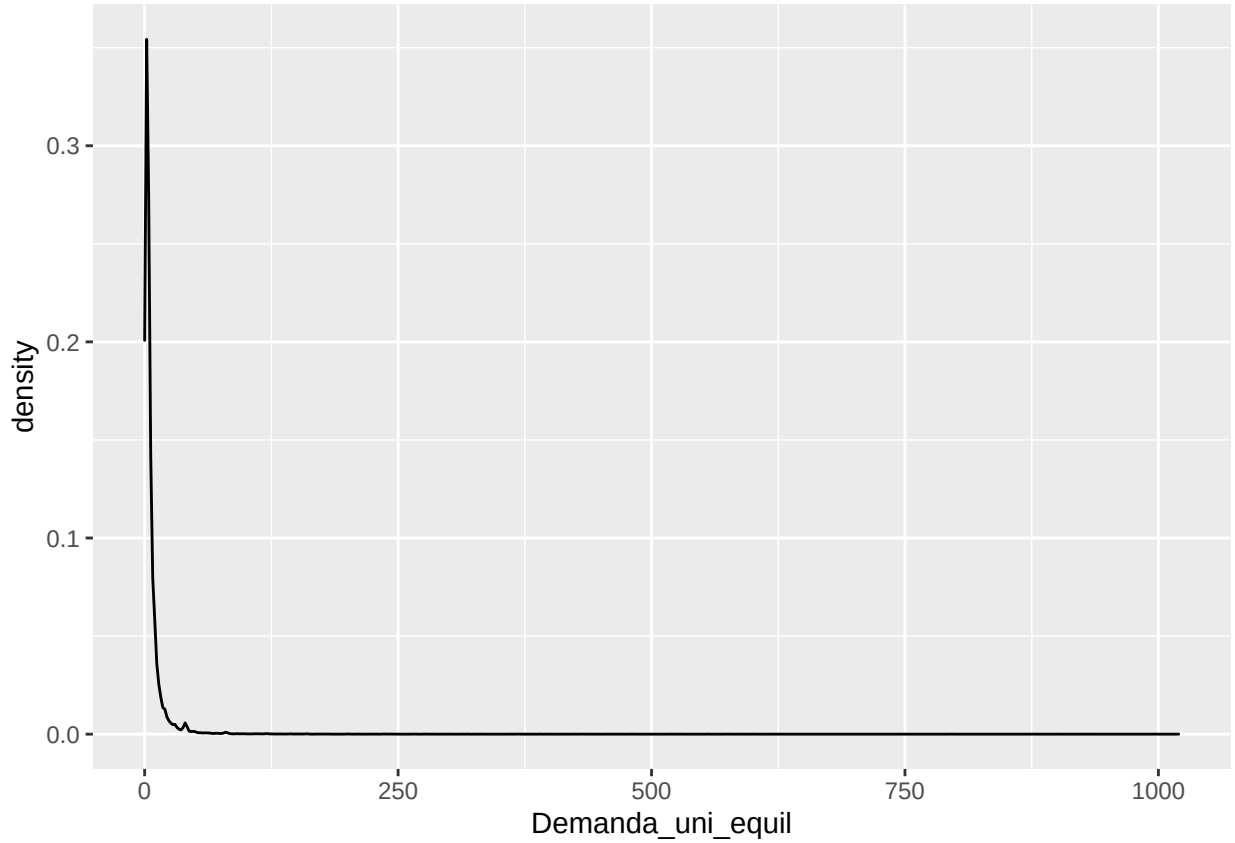
```
# Selecionado apenas varáveis preditoras + target
# As variáveis Venta_uni_hot, venta_hoy, dev_uni_proxima e dev_proxima não estão no
# df de teste, pois a variável target é resultado de cálculos a partir dessas variáveis

df <- df %>% select(Semana, Agencia_ID, Canal_ID, Ruta_SAK, Cliente_ID, Producto_ID, Demanda_uni_equil)

head(df)
```

```
## # A tibble: 6 x 7
##   Semana Agencia_ID Canal_ID Ruta_SAK Cliente_ID Producto_ID Demanda_uni_equil
##    <dbl>      <dbl>    <dbl>    <dbl>      <dbl>       <dbl>              <dbl>
## 1      3       1110        7     3301      15766        1212                  3
## 2      3       1110        7     3301      15766        1216                  4
## 3      3       1110        7     3301      15766        1238                  4
## 4      3       1110        7     3301      15766        1240                  4
## 5      3       1110        7     3301      15766        1242                  3
## 6      3       1110        7     3301      15766        1250                  5
```

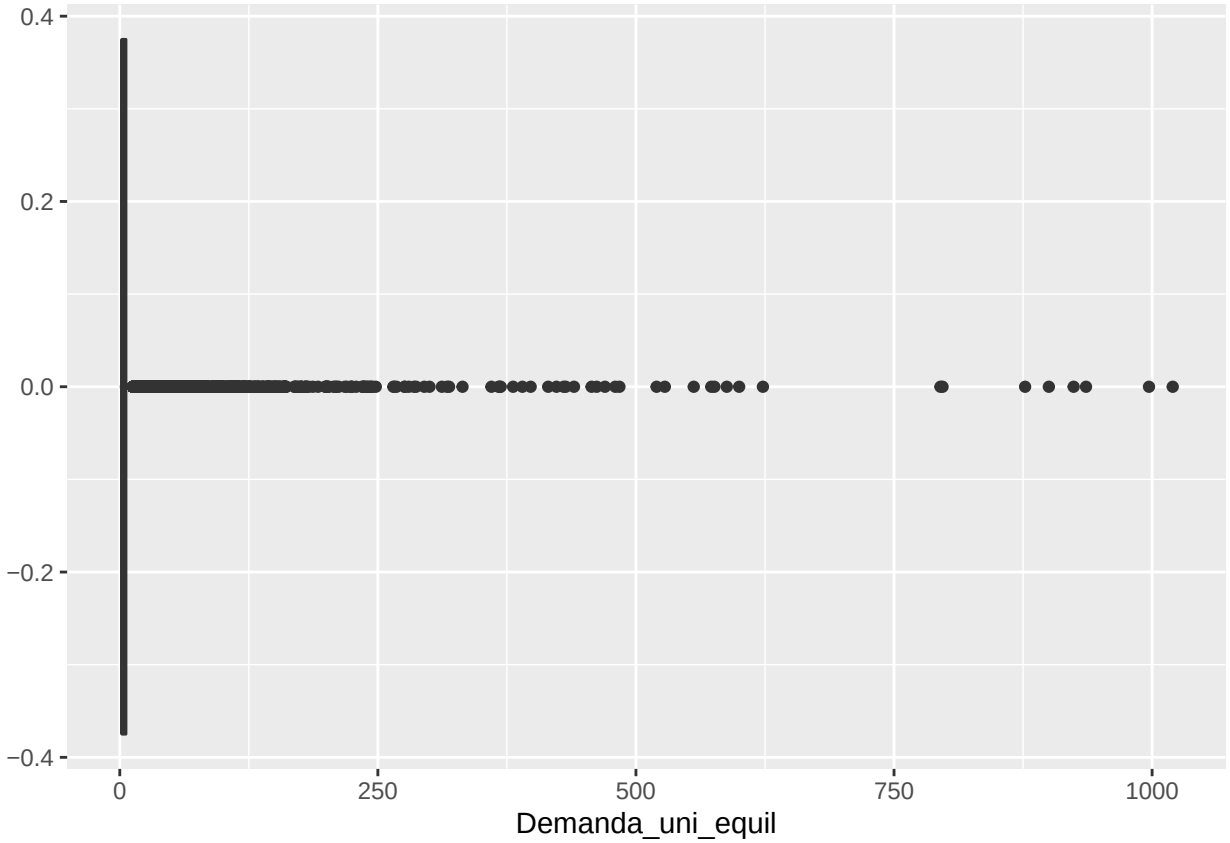
```
# Sub sampling
df_sample <- df %>% sample_n(50000)

# Analisando distribuição da variável target
ggplot(df_sample,aes(x=Demanda_uni_equil)) +
  geom_bar()
```



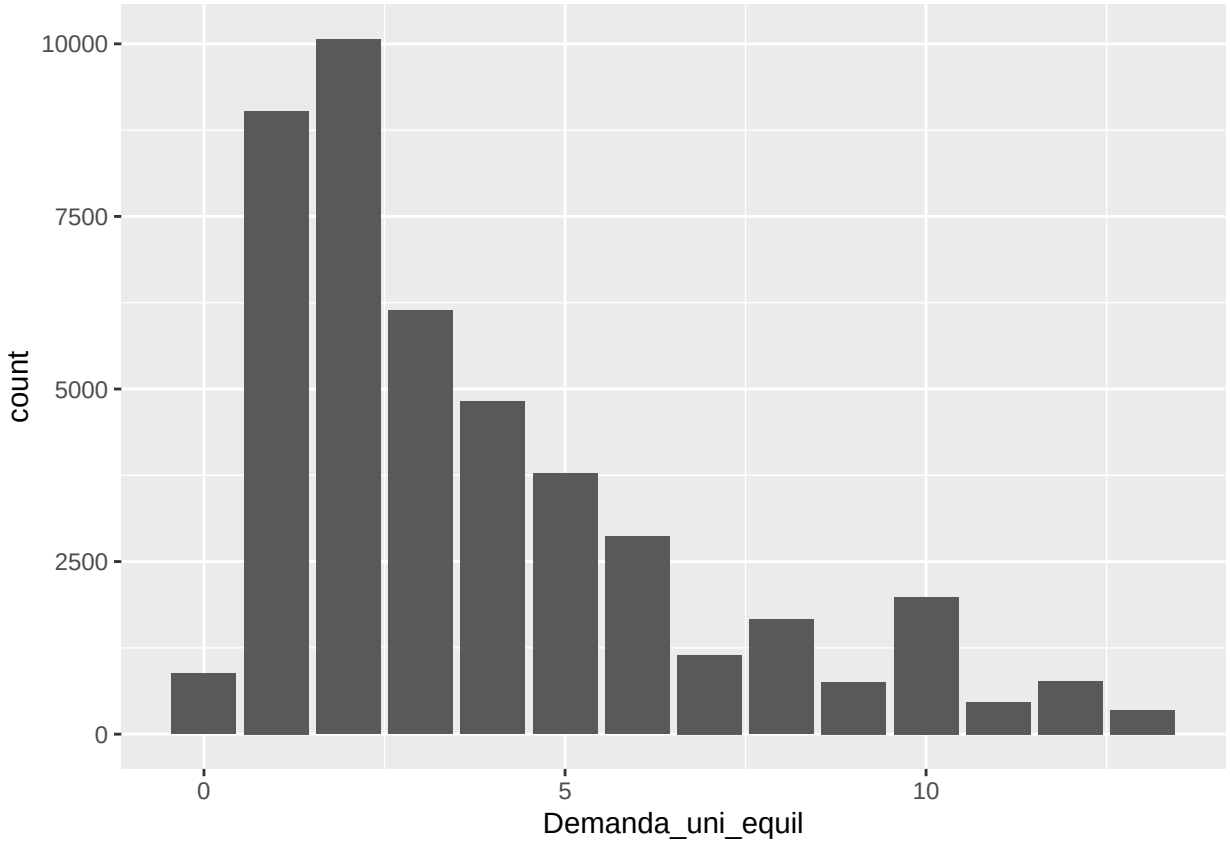

```
ggplot(df_sample,aes(x=Demanda_uni_equil)) +
  geom_density()
```



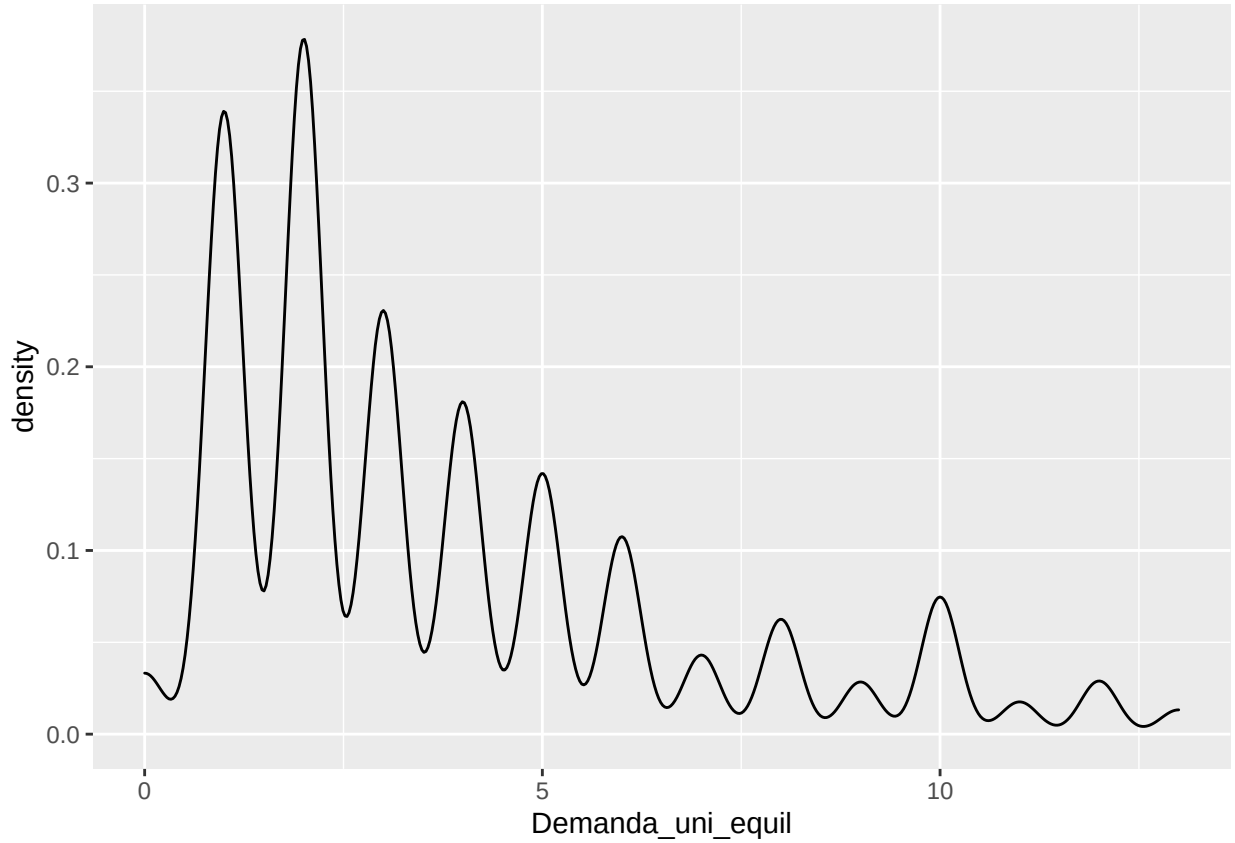

```
ggplot(df_sample,aes(x=Demanda_uni_equil)) +
  geom_boxplot()
```



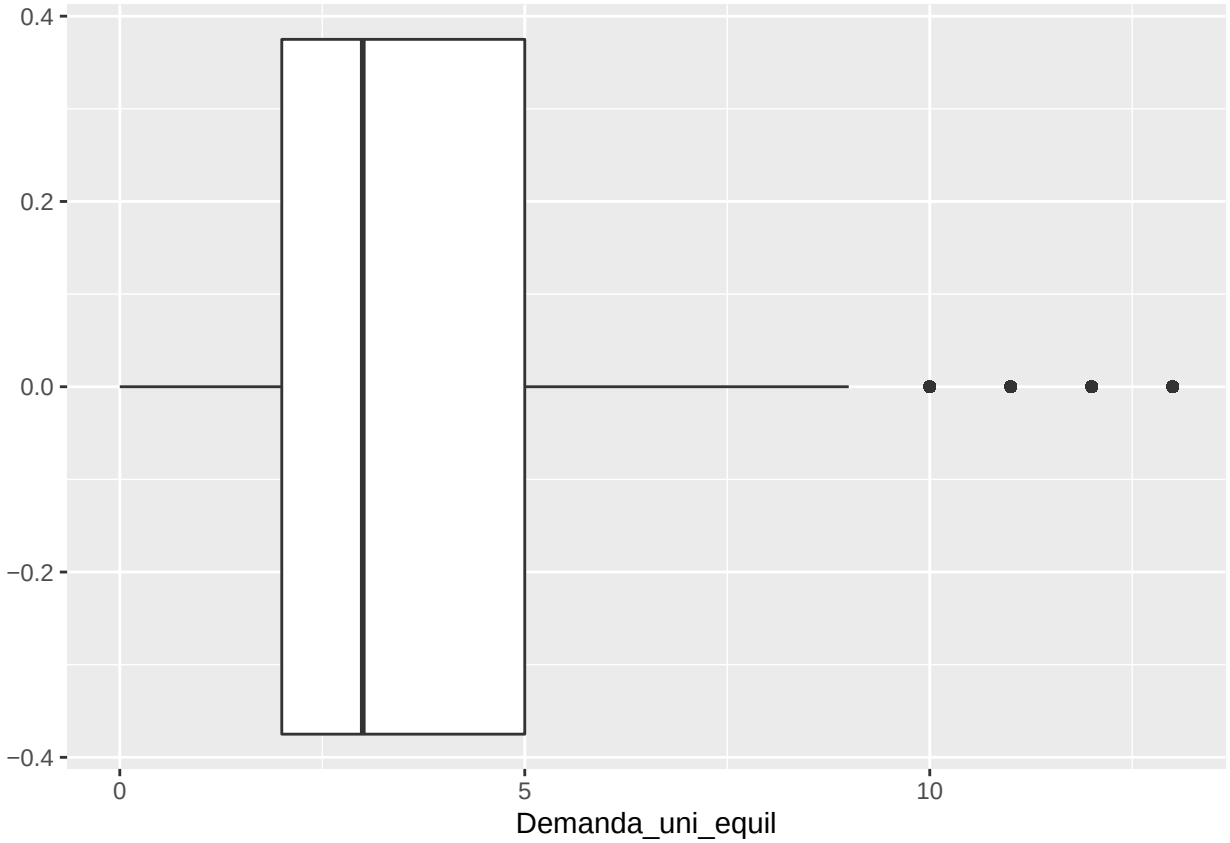

```
# Seleciona apenas os dados até o percentil 90
df_2 <- df_sample %>% filter(Demanda_uni_equil < quantile(df_sample$Demanda_uni_equil,0.9))

# Analisando novamente a distribuição da variável
ggplot(df_2,aes(x=Demanda_uni_equil)) +
  geom_bar()
```



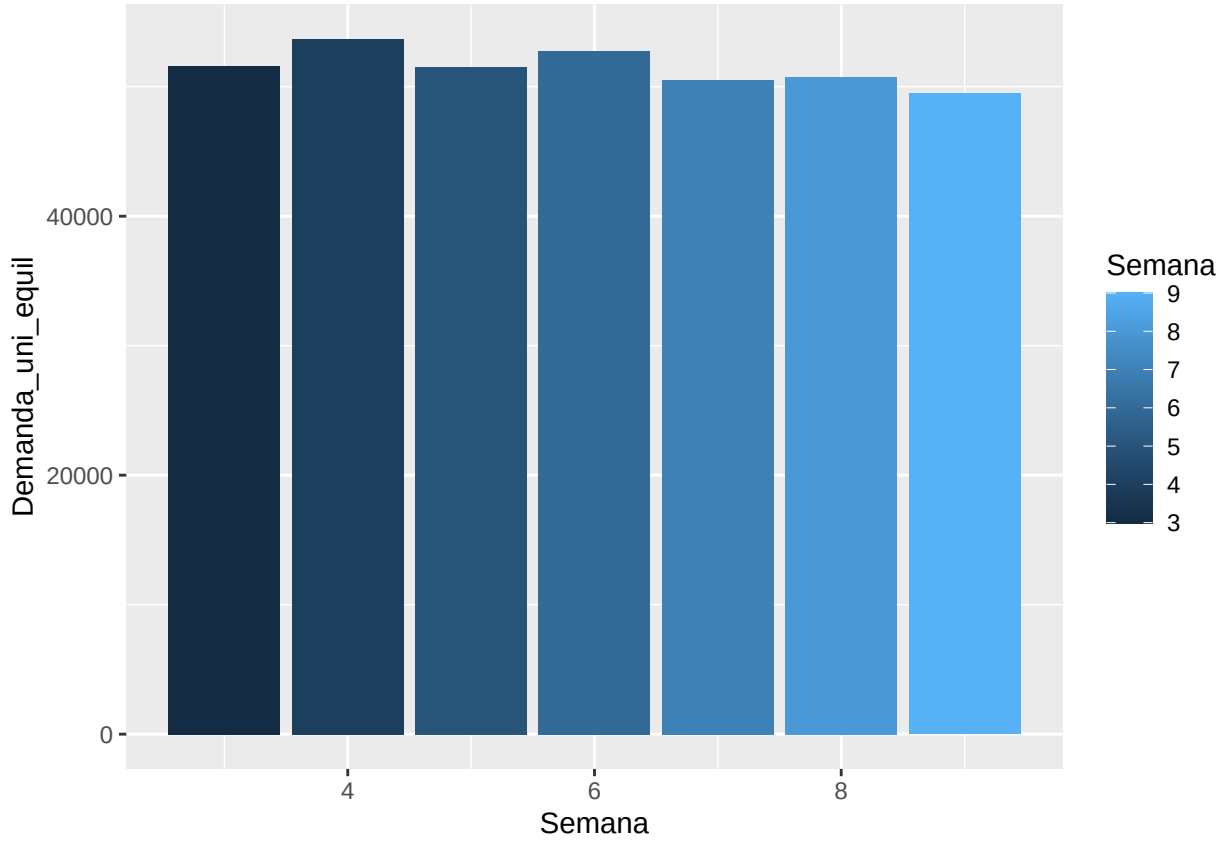

```
ggplot(df_2,aes(x=Demanda_uni_equil)) +
  geom_density()
```




```
ggplot(df_2,aes(x=Demanda_uni_equil)) +
  geom_boxplot()
```




```
# ----------------------- Análise exploratória -----------------------

# Demandas por semana
ggplot(df_sample,aes(fill=Semana,x=Semana, y=Demanda_uni_equil)) +
  geom_bar(stat='identity')
```




```
# Clientes com mais demanda

df_demanda_por_cliente <- df_2 %>%
  group_by(Cliente_ID) %>% summarise(Total = sum(Demanda_uni_equil)) %>%
  arrange(desc(Total)) %>% top_n(10)
```

```
## Selecting by Total
```

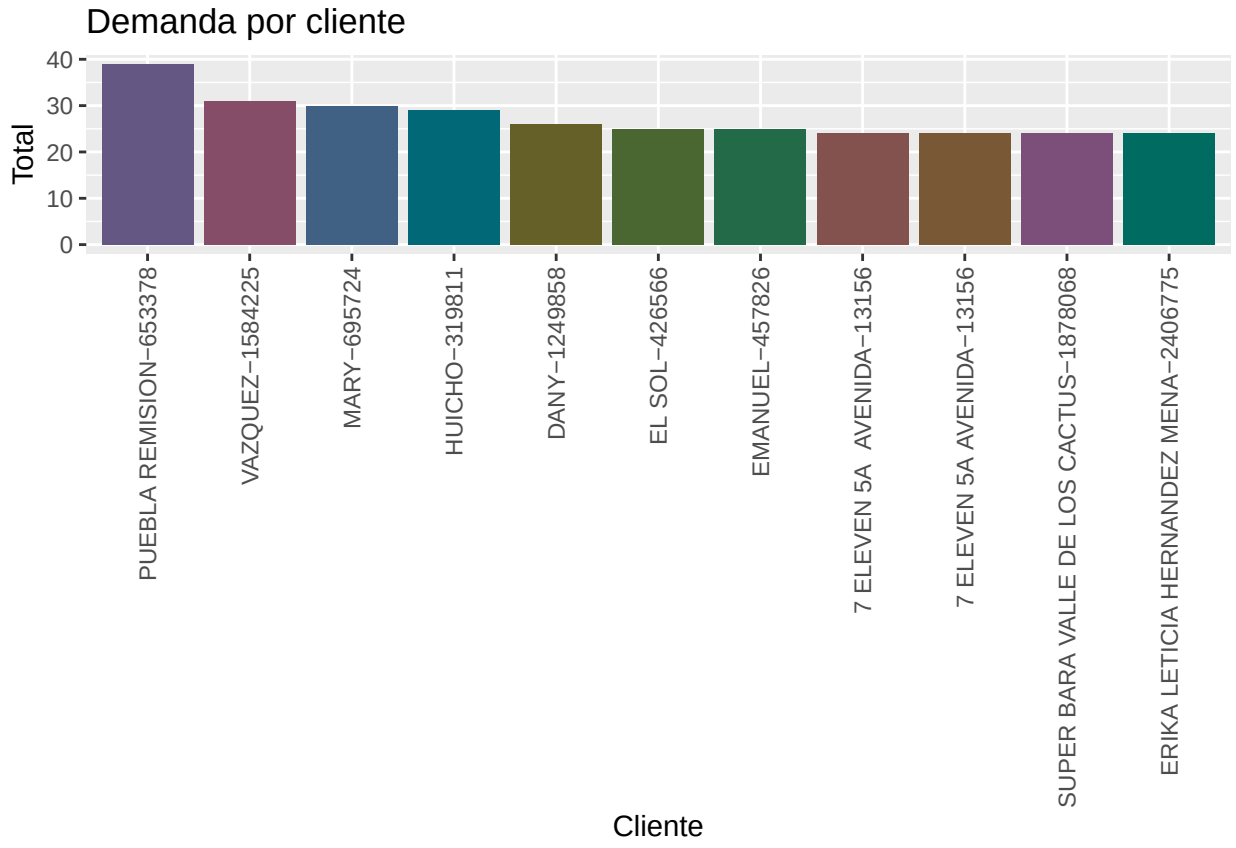
```
head(df_demanda_por_cliente)
```

```
## # A tibble: 6 x 2
##   Cliente_ID Total
##        <dbl> <dbl>
## 1     653378    39
## 2    1584225    31
## 3     695724    30
## 4     319811    29
## 5    1249858    26
## 6     426566    25
```

```
df_demanda_por_cliente <- merge(df_demanda_por_cliente, df_cliente, by='Cliente_ID')

df_demanda_por_cliente <- df_demanda_por_cliente %>% arrange(desc(Total))

df_demanda_por_cliente$NombraclienteId <- str_c(df_demanda_por_cliente$NombreCliente,'-',df_demanda_por

head(df_demanda_por_cliente)
```

```
##   Cliente_ID Total   NombreCliente        NombraclienteId
## 1     653378    39 PUEBLA REMISION PUEBLA REMISION-653378
## 2    1584225    31         VAZQUEZ        VAZQUEZ-1584225
## 3     695724    30            MARY           MARY-695724
## 4     319811    29          HUICHO         HUICHO-319811
## 5    1249858    26            DANY          DANY-1249858
## 6     426566    25          EL SOL         EL SOL-426566
```

```
df_demanda_por_cliente$NombraclienteId <- factor(df_demanda_por_cliente$NombraclienteId, levels=df_dema

ggplot(df_demanda_por_cliente,aes(fill=NombreCliente,x=NombraclienteId, y=Total)) +
  geom_bar(stat='identity') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1), legend.position = "none") +
  scale_fill_hue(l=40, c=35) +
  ggtitle("Demanda por cliente") +
  xlab("Cliente")
```




```
# Produtos com mais demanda
df_demanda_por_produto <- df_2 %>% select(Producto_ID, Demanda_uni_equil) %>%
  group_by(Producto_ID) %>% summarise(Total = sum(Demanda_uni_equil)) %>%
  arrange(desc(Total)) %>% top_n(10)
```

```
## Selecting by Total
```

```
head(df_demanda_por_produto)
```

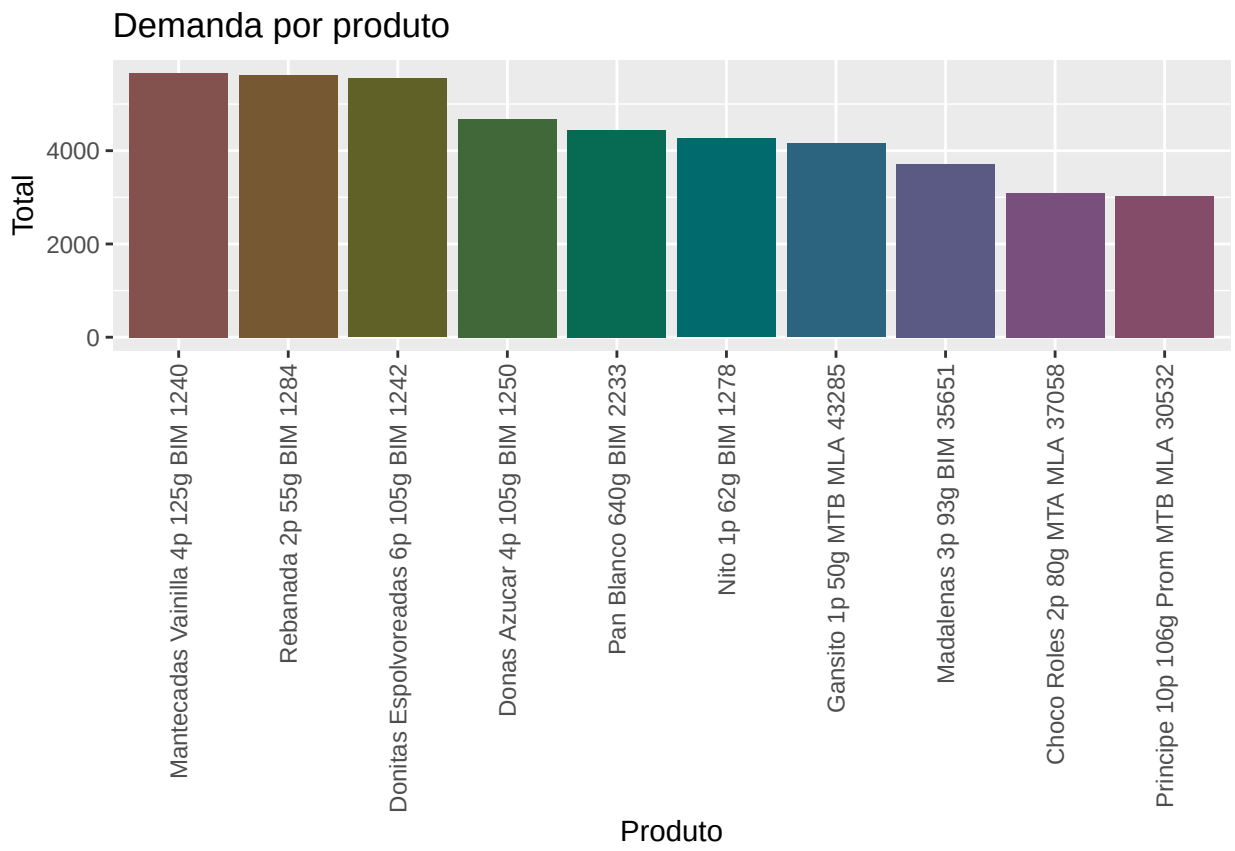
```
## # A tibble: 6 x 2
##   Producto_ID Total
```

```
##         <dbl> <dbl>
## 1        1240  5667
## 2        1284  5619
## 3        1242  5553
## 4        1250  4678
## 5        2233  4448
## 6        1278  4259
```

```
df_demanda_por_produto <- merge(df_demanda_por_produto, df_producto, by='Producto_ID')

df_demanda_por_produto <- df_demanda_por_produto %>% arrange(desc(Total))

head(df_demanda_por_produto)
```

```
##   Producto_ID Total                         NombreProducto
## 1        1240  5667   Mantecadas Vainilla 4p 125g BIM 1240
## 2        1284  5619               Rebanada 2p 55g BIM 1284
## 3        1242  5553 Donitas Espolvoreadas 6p 105g BIM 1242
## 4        1250  4678          Donas Azucar 4p 105g BIM 1250
## 5        2233  4448               Pan Blanco 640g BIM 2233
## 6        1278  4259                   Nito 1p 62g BIM 1278
```

```
df_demanda_por_produto$NombreProducto <- factor(df_demanda_por_produto$NombreProducto, levels=df_demanda

ggplot(df_demanda_por_produto,aes(fill=NombreProducto,x=NombreProducto, y=Total)) +
  geom_bar(stat='identity') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1), legend.position = "none") +
  scale_fill_hue(l=40, c=35) +
  ggtitle("Demanda por produto") +
  xlab("Produto")
```




```
# Cidades com mais demanda
df_demanda_por_cidade <- df_2 %>% select(Agencia_ID, Demanda_uni_equil) %>%
  group_by(Agencia_ID) %>% summarise(Total = sum(Demanda_uni_equil)) %>%
  arrange(desc(Total)) %>% top_n(10)
```

```
## Selecting by Total
```

```
head(df_demanda_por_cidade)
```

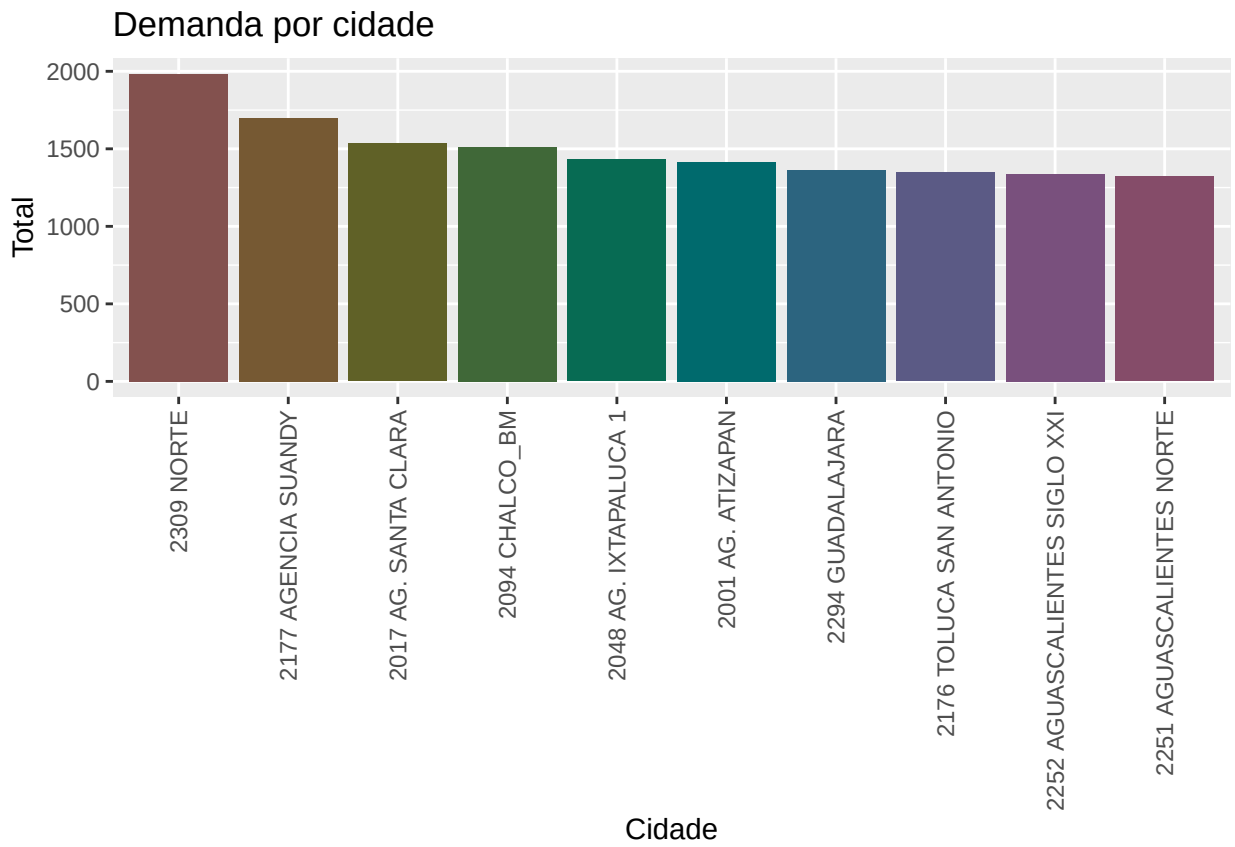
```
## # A tibble: 6 x 2
##   Agencia_ID Total
##        <dbl> <dbl>
## 1       1911  1985
## 2       2013  1701
## 3       1126  1535
## 4       1123  1512
## 5       1220  1431
## 6       1117  1414
```

```
df_demanda_por_cidade <- merge(df_demanda_por_cidade, df_town, by='Agencia_ID')

df_demanda_por_cidade <- df_demanda_por_cidade %>% arrange(desc(Total))

head(df_demanda_por_cidade)
```

```
##   Agencia_ID Total                 Town           State
## 1       1911  1985            2309 NORTE          JALISCO
## 2       2013  1701   2177 AGENCIA SUANDY ESTADO DE MÉXICO
```

```
## 3       1126  1535  2017 AG. SANTA CLARA ESTADO DE MÉXICO
## 4       1123  1512        2094 CHALCO_BM ESTADO DE MÉXICO
## 5       1220  1431 2048 AG. IXTAPALUCA 1 ESTADO DE MÉXICO
## 6       1117  1414     2001 AG. ATIZAPAN ESTADO DE MÉXICO
```

```
df_demanda_por_cidade$Town <- factor(df_demanda_por_cidade$Town, levels=df_demanda_por_cidade$Town)

ggplot(df_demanda_por_cidade,aes(fill=Town,x=Town, y=Total)) +
  geom_bar(stat='identity') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1), legend.position = "none") +
  scale_fill_hue(l=40, c=35) +
  ggtitle("Demanda por cidade") +
  xlab("Cidade")
```




```
# Estados com mais demanda
df_demanda_por_estado <- df_2 %>% select(Agencia_ID, Demanda_uni_equil)

head(df_demanda_por_estado)
```

```
## # A tibble: 6 x 2
##   Agencia_ID Demanda_uni_equil
##        <dbl>             <dbl>
## 1      25699                 1
## 2       1238                 2
## 3       1614                 3
## 4       1955                 1
## 5       1122                 3
```

```
## 6       1235                 1
```

```
df_demanda_por_estado <- merge(df_demanda_por_estado, df_town, by='Agencia_ID')

head(df_demanda_por_estado)
```

```
##   Agencia_ID Demanda_uni_equil               Town        State
## 1       1110                 0 2008 AG. LAGO FILT MÉXICO, D.F.
## 2       1110                13 2008 AG. LAGO FILT MÉXICO, D.F.
## 3       1110                12 2008 AG. LAGO FILT MÉXICO, D.F.
## 4       1110                 2 2008 AG. LAGO FILT MÉXICO, D.F.
## 5       1110                 2 2008 AG. LAGO FILT MÉXICO, D.F.
## 6       1110                 8 2008 AG. LAGO FILT MÉXICO, D.F.
```

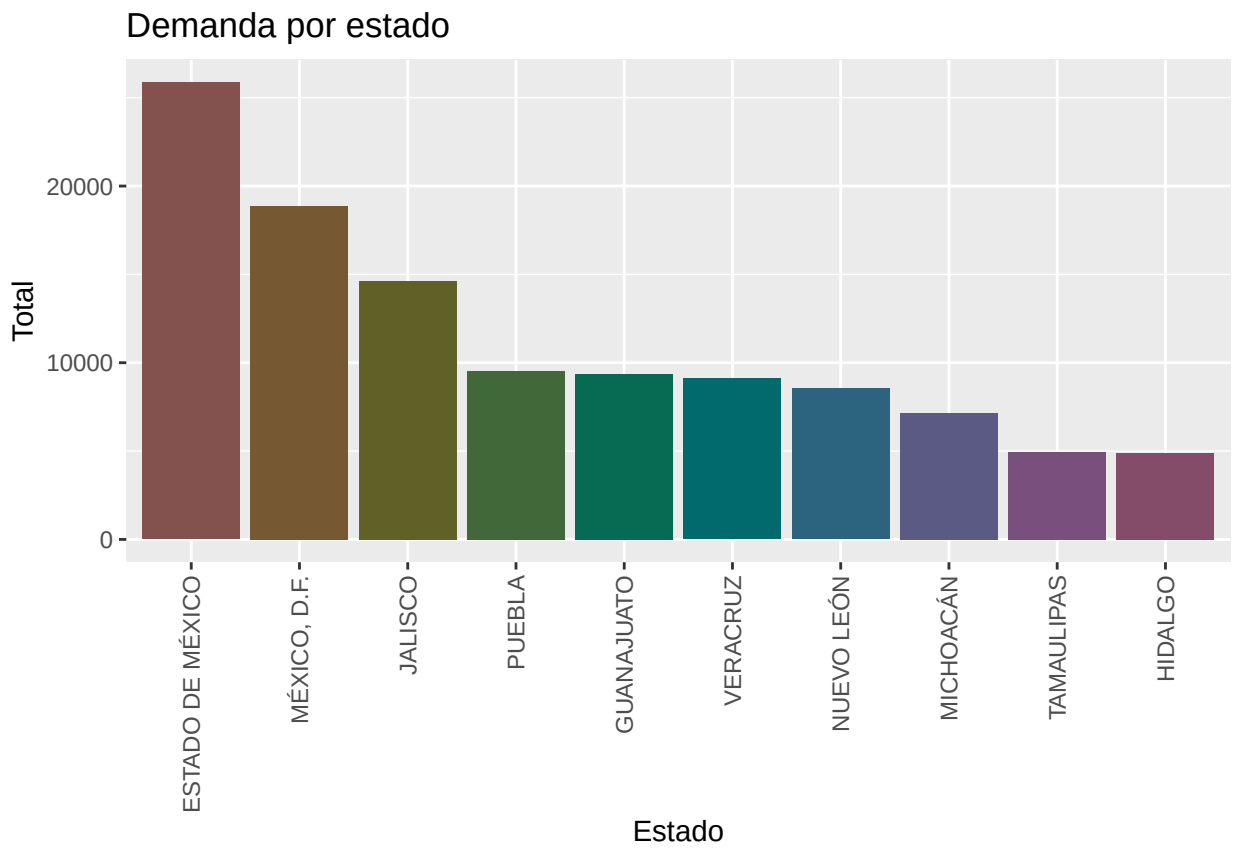
```
df_demanda_por_estado <- df_demanda_por_estado %>%
  group_by(State) %>% summarise(Total = sum(Demanda_uni_equil)) %>%
  arrange(desc(Total)) %>% top_n(10)
```

```
## Selecting by Total
```

```
head(df_demanda_por_estado)
```

```
## # A tibble: 6 x 2
##   State            Total
##   <chr>            <dbl>
## 1 ESTADO DE MÉXICO 25856
## 2 MÉXICO, D.F.     18894
## 3 JALISCO          14645
## 4 PUEBLA            9546
## 5 GUANAJUATO        9341
## 6 VERACRUZ          9102
```

```
df_demanda_por_estado$State <- factor(df_demanda_por_estado$State, levels=df_demanda_por_estado$State)

ggplot(df_demanda_por_estado,aes(fill=State,x=State, y=Total)) +
  geom_bar(stat='identity') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1), legend.position = "none") +
  scale_fill_hue(l=40, c=35) +
  ggtitle("Demanda por estado") +
  xlab("Estado")
```




```
# -------------------- Dividindo em dados de treino e de teste ----------------------
amostra <- createDataPartition(df_2$Demanda_uni_equil,p=0.7,list=F)
testData <- df_2[amostra,]
trainData <- df_2[-amostra,]
```

Modelos

```
# Criando um modelo LM com todas variáveis
modelo_lm_v1 <- lm(Demanda_uni_equil ~ .,data=trainData)

summary(modelo_lm_v1)
```

```
##
## Call:
## lm(formula = Demanda_uni_equil ~ ., data = trainData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.9138 -2.0684 -0.8317  1.3804  9.7430
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.319e+00  9.314e-02  35.637  < 2e-16 ***
## Semana      -9.865e-03  1.245e-02  -0.793    0.428
## Agencia_ID   6.624e-06  6.031e-06   1.098    0.272
## Canal_ID     1.854e-01  2.184e-02   8.488  < 2e-16 ***
```

```
## Ruta_SAK     1.319e-04  2.065e-05   6.386 1.76e-10 ***
## Cliente_ID  -6.983e-08  1.375e-08  -5.077 3.88e-07 ***
## Producto_ID  5.843e-06  1.410e-06   4.144 3.43e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.888 on 13403 degrees of freedom
## Multiple R-squared:  0.01984,    Adjusted R-squared:  0.0194
## F-statistic: 45.22 on 6 and 13403 DF,  p-value: < 2.2e-16
```

```
predictions <- predict(modelo_lm_v1,testData)

# rmsle: Root Mean Square Logarithmic Error
# 0.599
rmsle(predictions, testData$Demanda_uni_equil)
```

```
## [1] 0.5977337
```

```
# Modelo 2 com variáveis mais significativas
modelo_lm_v2 <- lm(Demanda_uni_equil ~ Canal_ID + Ruta_SAK + Cliente_ID + Producto_ID, data=trainData)

summary(modelo_lm_v2)
```

```
##
## Call:
## lm(formula = Demanda_uni_equil ~ Canal_ID + Ruta_SAK + Cliente_ID +
##     Producto_ID, data = trainData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.9416 -2.0646 -0.8283  1.3845  9.7061
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.278e+00  5.317e-02  61.652  < 2e-16 ***
## Canal_ID     1.844e-01  2.183e-02   8.447  < 2e-16 ***
## Ruta_SAK     1.322e-04  2.065e-05   6.401 1.60e-10 ***
## Cliente_ID  -6.976e-08  1.375e-08  -5.072 3.98e-07 ***
## Producto_ID  5.852e-06  1.410e-06   4.150 3.34e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.888 on 13405 degrees of freedom
## Multiple R-squared:  0.01971,    Adjusted R-squared:  0.01942
## F-statistic: 67.37 on 4 and 13405 DF,  p-value: < 2.2e-16
```

```
predictions_2 <- predict(modelo_lm_v2,testData)

# RMSLE = 0.599 -> igual anterior
rmsle(predictions_2, testData$Demanda_uni_equil)
```

```
## [1] 0.5977544
```

```
# KNN
modelo_knn_v1 <- knnreg(Demanda_uni_equil ~ ., data=trainData)

summary(modelo_knn_v1)
```

```
##         Length Class  Mode
## learn   2      -none- list
## k       1      -none- numeric
## terms   3      terms  call
## xlevels 0      -none- list
## theDots 0      -none- list
```

```
predictions_3 <- predict(modelo_knn_v1, testData)

# 0.6526 -> Pior que os outros
rmsle(predictions_3, testData$Demanda_uni_equil)
```

```
## [1] 0.6424282
```

```
# SVR
modelo_svr_v1 <- svm(Demanda_uni_equil ~ ., data=trainData)

summary(modelo_svr_v1)
```

```
##
## Call:
## svm(formula = Demanda_uni_equil ~ ., data = trainData)
##
##
## Parameters:
##    SVM-Type:  eps-regression
##  SVM-Kernel:  radial
##        cost:  1
##       gamma:  0.1666667
##     epsilon:  0.1
##
##
## Number of Support Vectors:  12496
```

```
predictions_4 <- predict(modelo_svr_v1, testData)

# 0.577 -> Melhor resultado até o momento
rmsle(predictions_4, testData$Demanda_uni_equil)
```

```
## [1] 0.5774893
```

```
# SVR - v2
modelo_svr_v2 <- svm(Demanda_uni_equil ~ Canal_ID + Ruta_SAK + Cliente_ID + Producto_ID, data=trainData

summary(modelo_svr_v2)
```

```
##
## Call:
## svm(formula = Demanda_uni_equil ~ Canal_ID + Ruta_SAK + Cliente_ID +
##     Producto_ID, data = trainData)
##
##
## Parameters:
##    SVM-Type:  eps-regression
##  SVM-Kernel:  radial
```

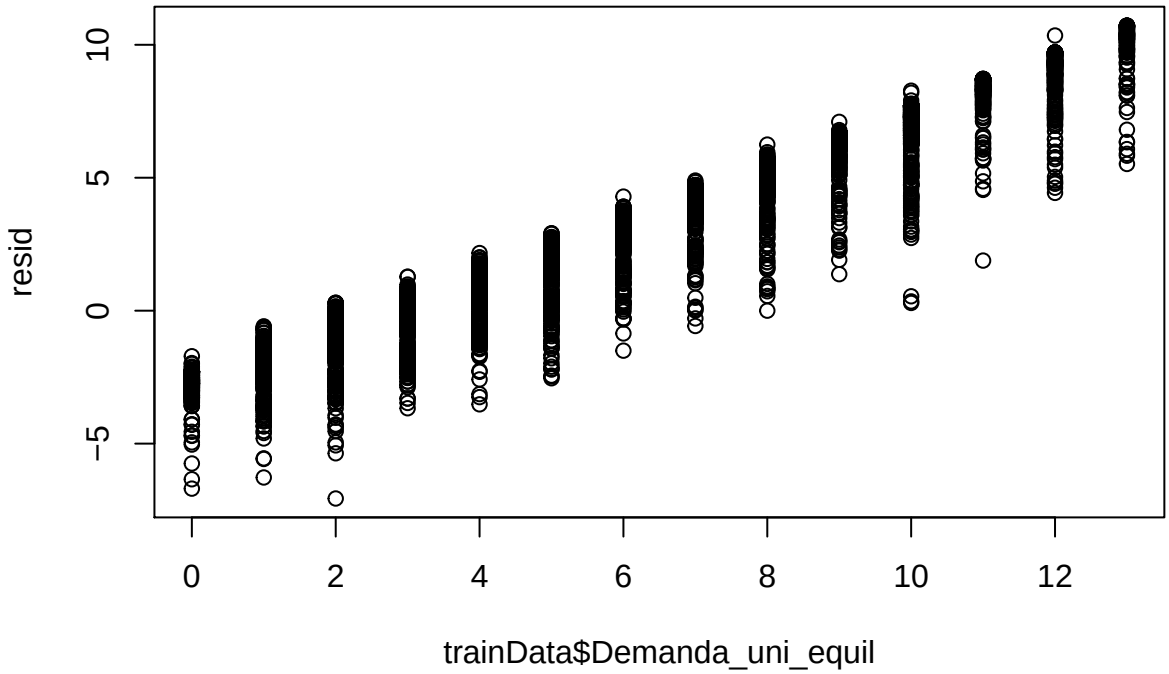
```
##        cost:  1
##       gamma:  0.25
##     epsilon:  0.1
##
##
## Number of Support Vectors:  12526
```

```
predictions_5 <- predict(modelo_svr_v2, testData)

# 0.576 -> Um pouco melhor que o anterior
rmsle(predictions_5, testData$Demanda_uni_equil)
```

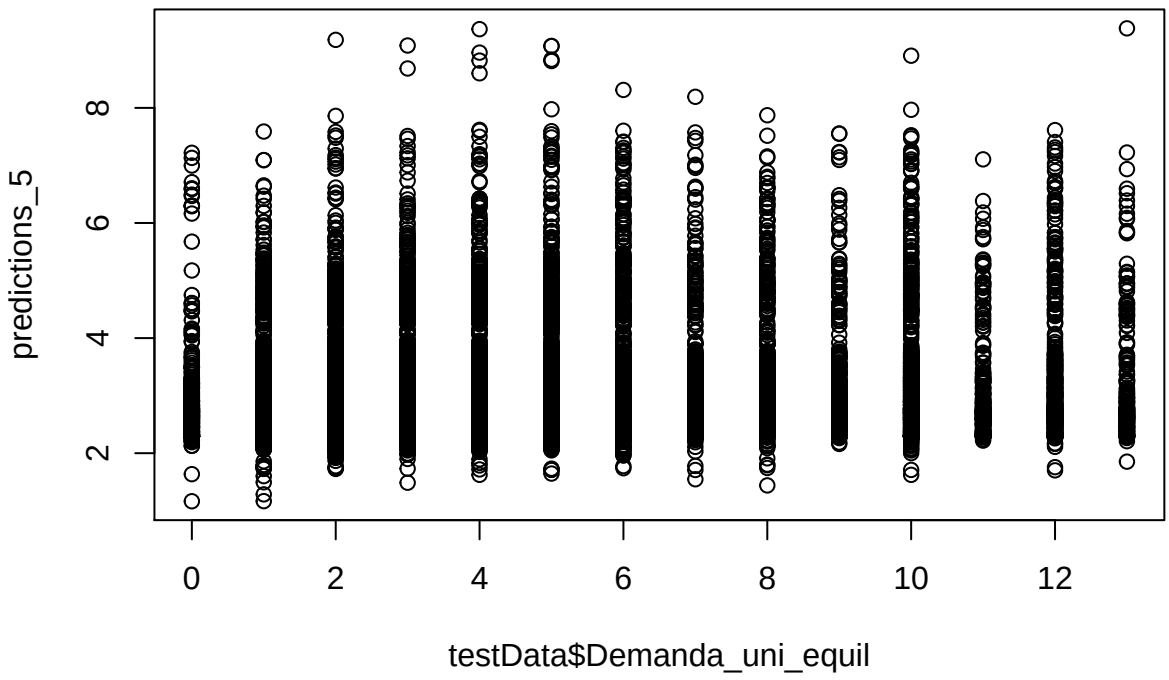
```
## [1] 0.57646
```

Avaliação

```
# Plot Residuals
resid <- resid(modelo_svr_v2)
plot(trainData$Demanda_uni_equil, resid)
```




```
plot(y=predictions_5, x=testData$Demanda_uni_equil)
```


predictions_5
2
4
6
8
0
2
4
6
8
10
12
testData$Demanda_uni_equil