PrevedoDemandaEstoqueHTML

Eduardo de Souza Dias

12/18/2019

```
#Projeto 02 - Curso BigDataRAzure da DSA (parte da Formação cientista de dados)
#Prevendo demanda de produtos (Grupo Bimbo)

#Definindo diretório de trabalho
setwd("C:/Cursos/FCD/01-BigDataRAzure/Cap20-ProjetosFeedback/Projeto02-PrevendoDemandaEstoque")
getwd()
```

```
## [1] "C:/Cursos/FCD/01-BigDataRAzure/Cap20-ProjetosFeedback/Projeto02-PrevendoDemandaEstoque"
```

```
library(data.table)
library(ggplot2)
library(dplyr)
```

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

```
## The following objects are masked from 'package:data.table':
##
##     between, first, last
```

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

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

```
library(grid)
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
library(caret)
```

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

```
library(Metrics)
```

```
##
## Attaching package: 'Metrics'
```

```
## The following objects are masked from 'package:caret':
##
##     precision, recall
```

```
#BASE DE DADOS

df <- fread("train.csv")

#Criando um dataset sample para análise exploratória
df <- df[sample.int(nrow(df),100000),]

#Criando uma coluna com a demanda em pesos
df$Demanda_equil <- ifelse((df$Venta_hoy-df$Dev_proxima)>0,df$Venta_hoy-df$Dev_proxima,0)

#Verificando a base de clientes
dfClientes <- fread("cliente_tabla.csv", encoding = "UTF-8")
length(unique(dfClientes$Cliente_ID))
```

```
## [1] 930500
```

```
length(unique(dfClientes$NombreCliente))
```

```
## [1] 311155
```

```
#Como o Cliente_ID é chave forte, ainda que duplicado em alguns casos,
#é melhor identificador que o nome. Com isso, irei trazer o nome apenas para nos ajudar
#na análise exploratória, mas nao o utilizarei no modelo
dfClientes <- dfClientes %>%
  group_by(Cliente_ID) %>%
  summarise(first(NombreCliente))
df <- left_join(df,dfClientes,by="Cliente_ID")
names(df)[names(df) == "first(NombreCliente)"] <- "NombreCliente"
rm(dfClientes)

#Verificando base de produtos
dfProdutos <- fread("producto_tabla.csv", encoding = "UTF-8")
length(unique(dfProdutos$Producto_ID))
```

```
## [1] 2592
```

```
length(unique(dfProdutos$NombreProducto))
```

```
## [1] 2592
```

```
#Não há IDs e nomes iguais. Com isso, irei trazer o nome dos produtos para a tabela,
#apenas para nos ajudar na análise exploratória
df <- left_join(df,dfProdutos,by="Producto_ID")
rm(dfProdutos)

#Verificano base de cidade e estado
dfTownState <- fread("town_state.csv", encoding = "UTF-8")
length(unique(dfTownState$Agencia_ID))
```

```
## [1] 790
```

```
length(unique(dfTownState$Town))
```

```
## [1] 260
```

```
length(unique(dfTownState$State))
```

```
## [1] 33
```

```
#Nao ha IDs e noms iguais. Com isso, irei trazer o nome dos produtos para a tabela,
#apenas para nos ajudar na analise exploratoria
df <- left_join(df,dfTownState,by="Agencia_ID")
rm(dfTownState)

#Procurando por valores NA
qtNA_df <- df[rowSums(is.na(df)) > 0,]
qtNA_df
```

```
##  [1] Semana            Agencia_ID        Canal_ID          Ruta_SAK
##  [5] Cliente_ID        Producto_ID       Venta_uni_hoy     Venta_hoy
##  [9] Dev_uni_proxima   Dev_proxima       Demanda_uni_equil Demanda_equil
## [13] NombreCliente     NombreProducto    Town              State
## <0 rows> (or 0-length row.names)
```

```
rm(qtNA_df)

#Transformando variáveis em categoricas
df$Semana <- as.factor(df$Semana)
df$Agencia_ID <- as.factor(df$Agencia_ID)
df$Canal_ID <- as.factor(df$Canal_ID)
df$Producto_ID <- as.factor(df$Producto_ID)
df$Ruta_SAK <- as.factor(df$Ruta_SAK)
df$Cliente_ID <- as.factor(df$Cliente_ID)

#Análise básica dos dados
summary(df)
```

```
##  Semana      Agencia_ID       Canal_ID        Ruta_SAK       Cliente_ID
##  3:15147   1911   : 1078   1      :91020   1201   :  602   653378 :  157
##  4:14551   1123   :  943   4      : 5038   1202   :  591   653039 :    9
##  5:14485   1220   :  916   11     : 1313   1204   :  537   652850 :    6
##  6:13630   2013   :  844   2      : 1057   1203   :  532   424338 :    5
##  7:14026   1945   :  825   7      :  884   1213   :  532   424478 :    5
##  8:14168   1351   :  806   6      :  394   1205   :  531   16578  :    4
##  9:13993   (Other):94588   (Other):  294   (Other):96675   (Other):99814
##   Producto_ID    Venta_uni_hoy       Venta_hoy        Dev_uni_proxima
##  1240   : 2908   Min.   :   0.00   Min.   :    0.00   Min.   :   0.0000
##  1242   : 2768   1st Qu.:   2.00   1st Qu.:   16.76   1st Qu.:   0.0000
##  2233   : 2709   Median :   3.00   Median :   30.00   Median :   0.0000
##  1250   : 2573   Mean   :   7.19   Mean   :   66.72   Mean   :   0.1382
##  1284   : 2207   3rd Qu.:   6.00   3rd Qu.:   55.90   3rd Qu.:   0.0000
##  1146   : 1970   Max.   :2130.00   Max.   :25328.16   Max.   :2240.0000
##  (Other):84865
##   Dev_proxima       Demanda_uni_equil  Demanda_equil      NombreCliente
##  Min.   :   0.000   Min.   :   0.000   Min.   :    0.00   Length:100000
##  1st Qu.:   0.000   1st Qu.:   2.000   1st Qu.:   16.66   Class :character
##  Median :   0.000   Median :   3.000   Median :   29.64   Mode  :character
##  Mean   :   1.177   Mean   :   7.109   Mean   :   65.91
##  3rd Qu.:   0.000   3rd Qu.:   6.000   3rd Qu.:   55.52
##  Max.   :4032.000   Max.   :2130.000   Max.   :25328.16
##
##  NombreProducto         Town              State
##  Length:100000      Length:100000      Length:100000
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
```

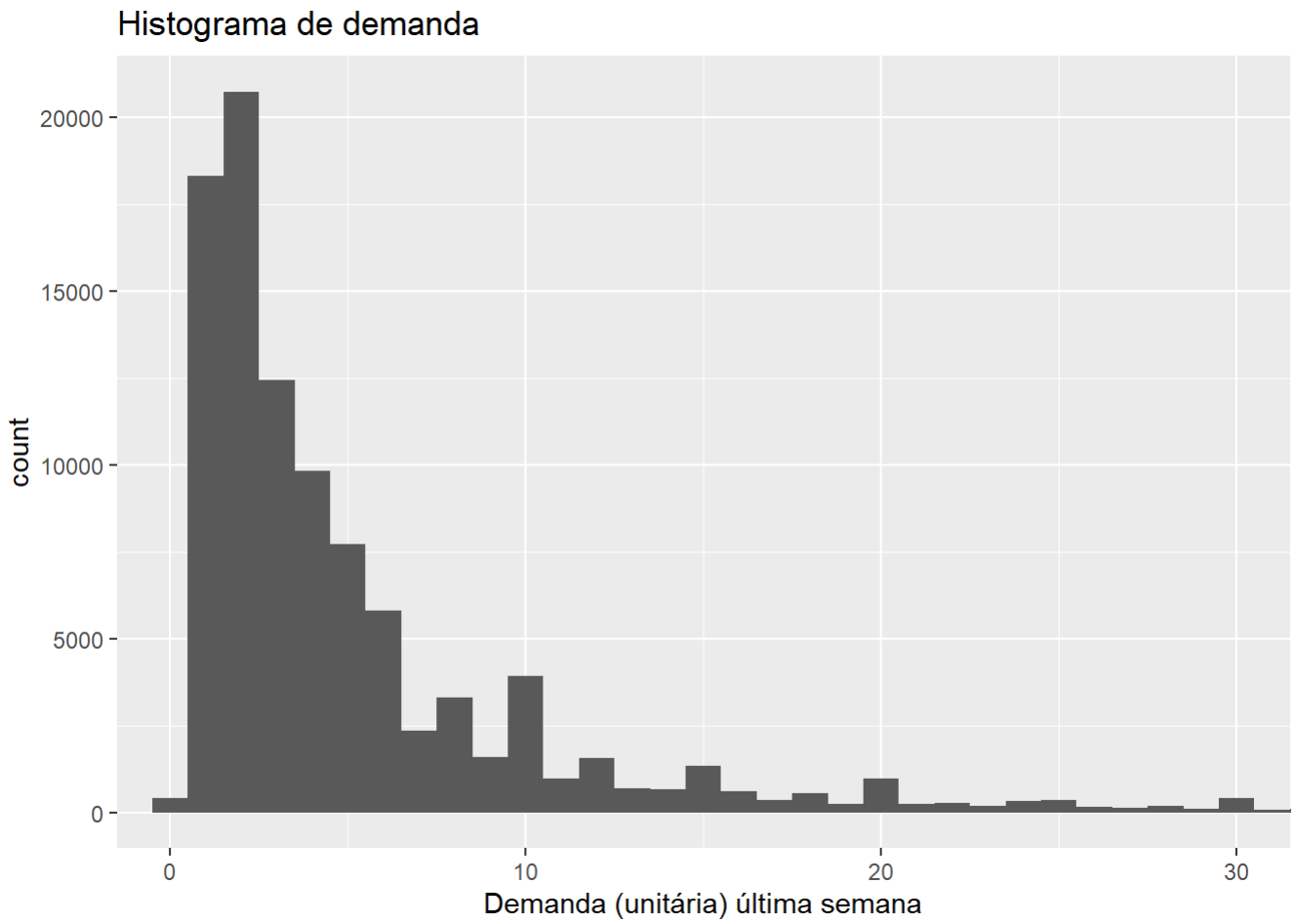
```
str(df)
```

```
## 'data.frame':    100000 obs. of  16 variables:
##  $ Semana           : Factor w/ 7 levels "3","4","5","6",..: 6 3 3 2 6 4 3 5 5 2 ...
##  $ Agencia_ID       : Factor w/ 535 levels "1110","1111",..: 433 363 81 504 499 251 535 365 1
06 370 ...
##  $ Canal_ID         : Factor w/ 8 levels "1","2","4","5",..: 1 1 1 1 3 1 1 1 1 1 ...
##  $ Ruta_SAK         : Factor w/ 1877 levels "1","2","3","4",..: 212 136 747 1012 1491 317 333
959 336 319 ...
##  $ Cliente_ID       : Factor w/ 89398 levels "60","65","465",..: 9673 14977 78640 74208 394 2
8917 53250 86689 40584 27853 ...
##  $ Producto_ID      : Factor w/ 965 levels "72","73","106",..: 146 52 805 807 831 72 692 739
60 72 ...
##  $ Venta_uni_hoy    : int  2 1 15 10 8 18 2 1 1 13 ...
##  $ Venta_hoy        : num  28.5 18.9 79.2 52.8 59.8 ...
##  $ Dev_uni_proxima  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Dev_proxima      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Demanda_uni_equil: int  2 1 15 10 8 18 2 1 1 13 ...
##  $ Demanda_equil    : num  28.5 18.9 79.2 52.8 59.8 ...
##  $ NombreCliente    : chr  "LULU" "FLORENTINO BARCENAS MARTINEZ" "NO IDENTIFICADO" "NO IDENTI
FICADO" ...
##  $ NombreProducto   : chr  "Totopos 280g DH 6469" "Bimbollos 8p 450g BIM 1160" "Gansito 1p 50
g MTB MLA 43285" "Gansito 1p 50g CCharola MTA MLA 43307" ...
##  $ Town             : chr  "2397 SALINA CRUZ" "2161 IRAPUATO GUADALUPE" "2260 GONZALEZ GALLO"
"2368 TAMPICO BIMBO" ...
##  $ State            : chr  "OAXACA" "GUANAJUATO" "JALISCO" "TAMAULIPAS" ...
```

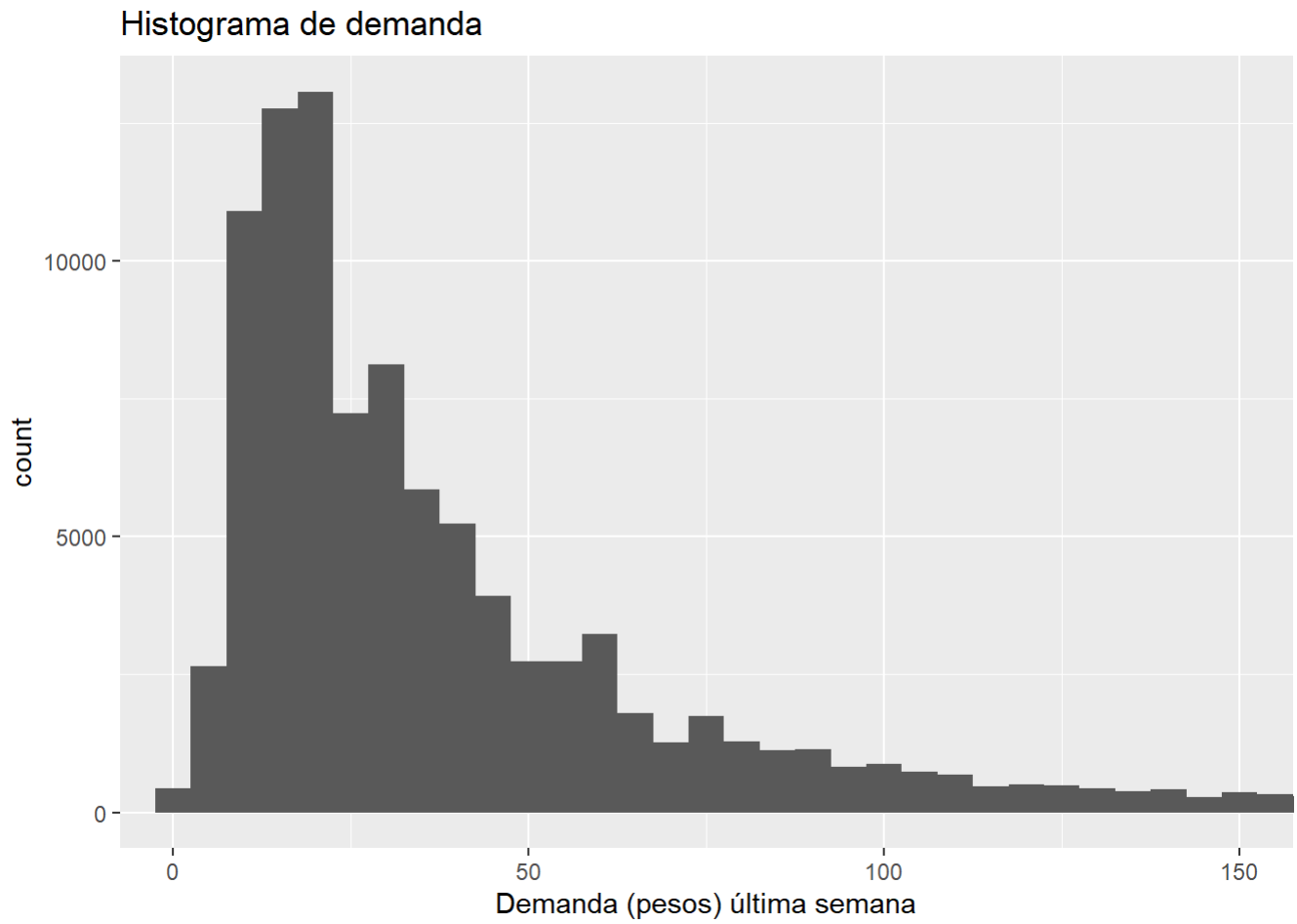
```
head(df)
```

```
##   Semana Agencia_ID Canal_ID Ruta_SAK Cliente_ID Producto_ID Venta_uni_hoy
## 1      8       2242        1     1102     145421        6469             2
## 2      5       2029        1     1024     224781        1160             1
## 3      5       1310        1     2022    4457827       43285            15
## 4      4       4041        1     2857    4306351       43307            10
## 5      8       4017        4     4804      10447       44371             8
## 6      6       1615        1     1212     546848        1284            18
##   Venta_hoy Dev_uni_proxima Dev_proxima Demanda_uni_equil Demanda_equil
## 1     28.52               0           0                2         28.52
## 2     18.86               0           0                1         18.86
## 3     79.20               0           0               15         79.20
## 4     52.80               0           0               10         52.80
## 5     59.84               0           0                8         59.84
## 6     54.36               0           0               18         54.36
##                 NombreCliente                      NombreProducto
## 1                       LULU                 Totopos 280g DH 6469
## 2 FLORENTINO BARCENAS MARTINEZ            Bimbollos 8p 450g BIM 1160
## 3              NO IDENTIFICADO          Gansito 1p 50g MTB MLA 43285
## 4              NO IDENTIFICADO Gansito 1p 50g CCharola MTA MLA 43307
## 5               7 ELEVEN PLAZA       Mantecadas 2p 105g MTB TR 44371
## 6                    EL KIOSKO              Rebanada 2p 55g BIM 1284
##                      Town      State
## 1          2397 SALINA CRUZ     OAXACA
## 2   2161 IRAPUATO GUADALUPE GUANAJUATO
## 3       2260 GONZALEZ GALLO    JALISCO
## 4        2368 TAMPICO BIMBO TAMAULIPAS
## 5      2481 WONDER GUERRERO NUEVO LEÓN
## 6 2358 MARTINEZ DE LA TORRE   VERACRUZ
```

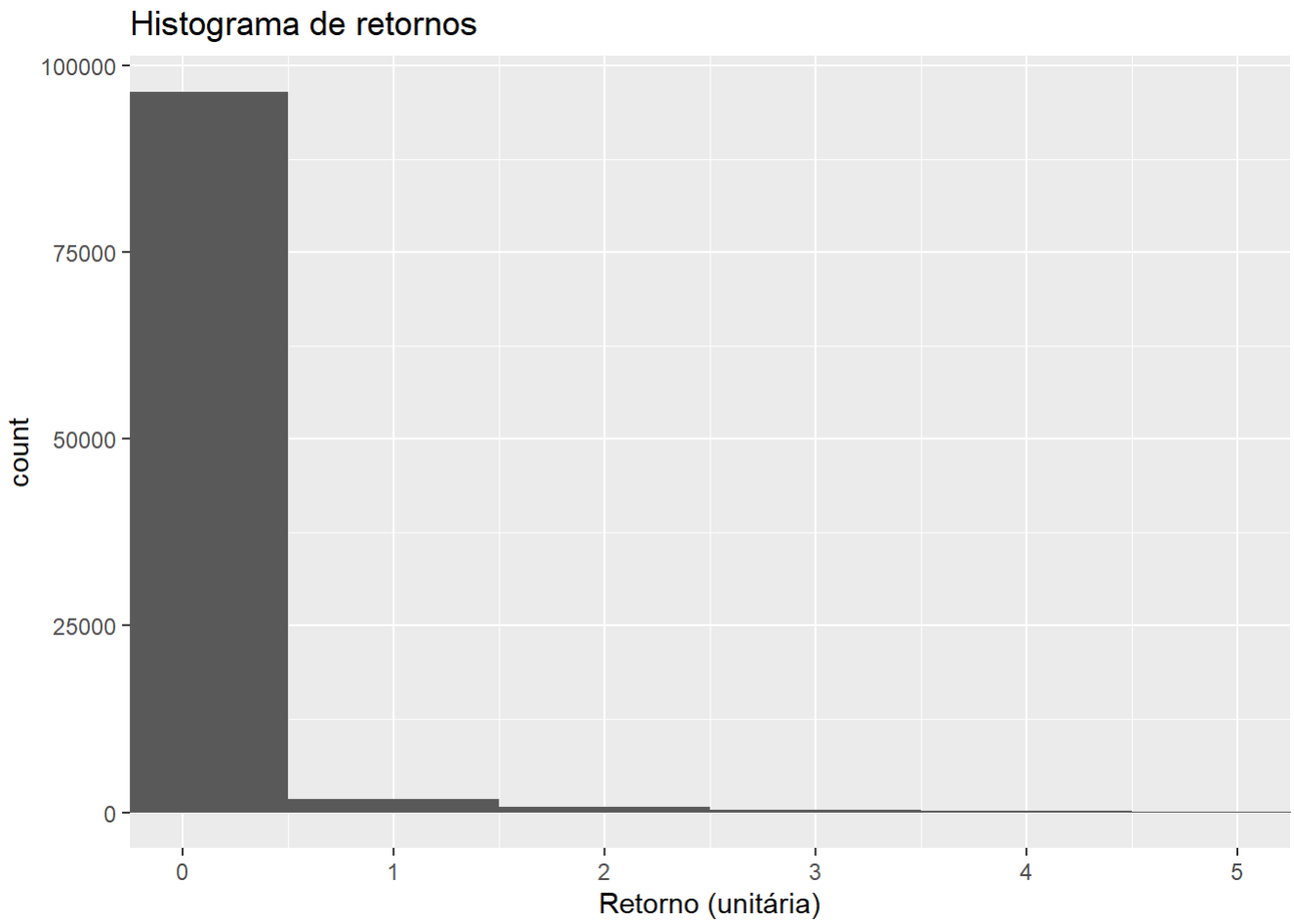
```
View(df)

#ANÁLISE EXPLORATÓRIA

#Distribuição dos dados
g <- ggplot(df)

g + geom_histogram(aes(Venta_uni_hoy), binwidth = 1) + coord_cartesian(xlim = c(0,30)) + labs(ti
tle="Histograma de demanda", x="Demanda (unitária) última semana")
```



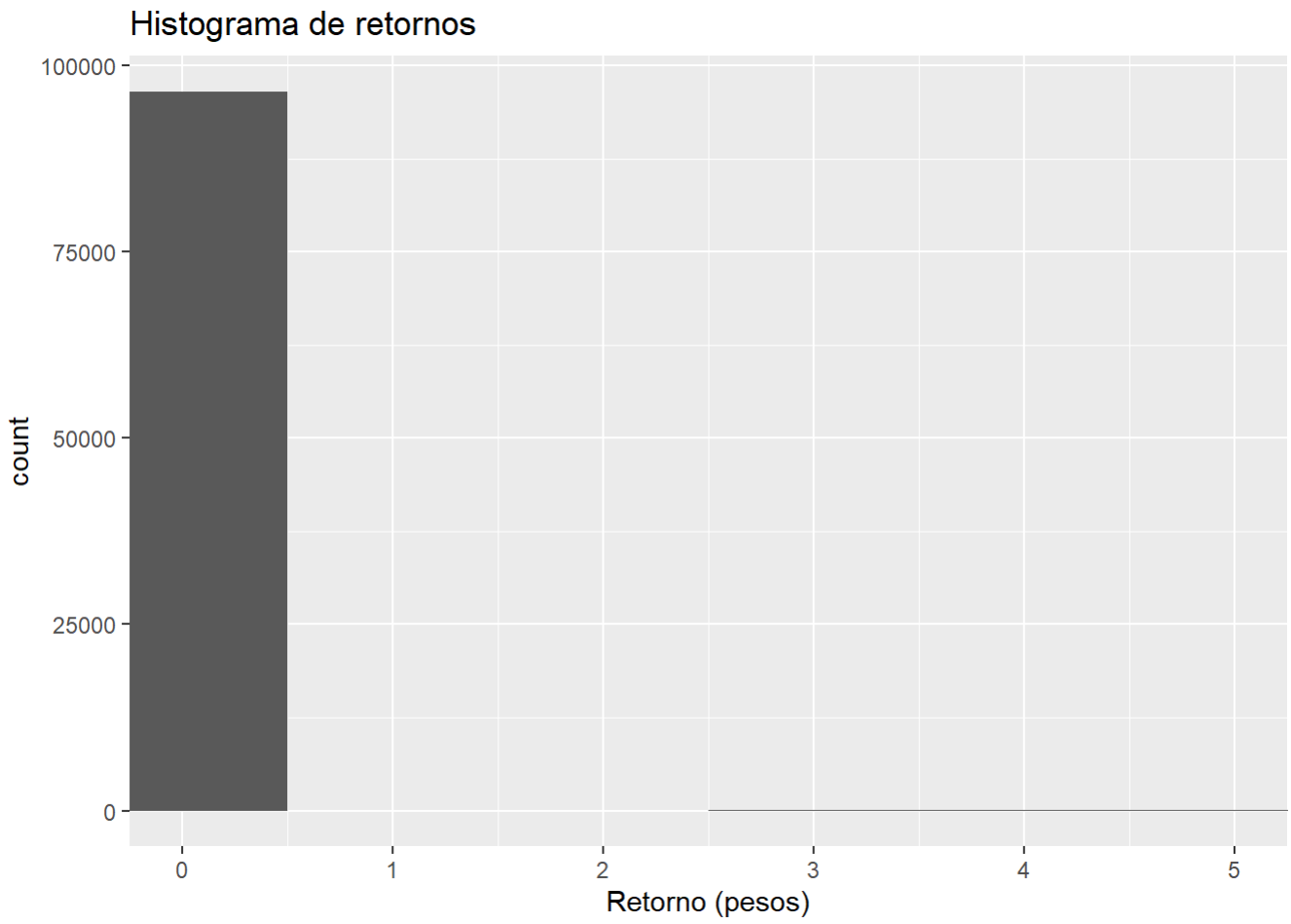

```
g + geom_histogram(aes(Venta_hoy), binwidth = 5) + coord_cartesian(xlim = c(0,150)) + labs(title
="Histograma de demanda", x="Demanda (pesos) última semana")
```



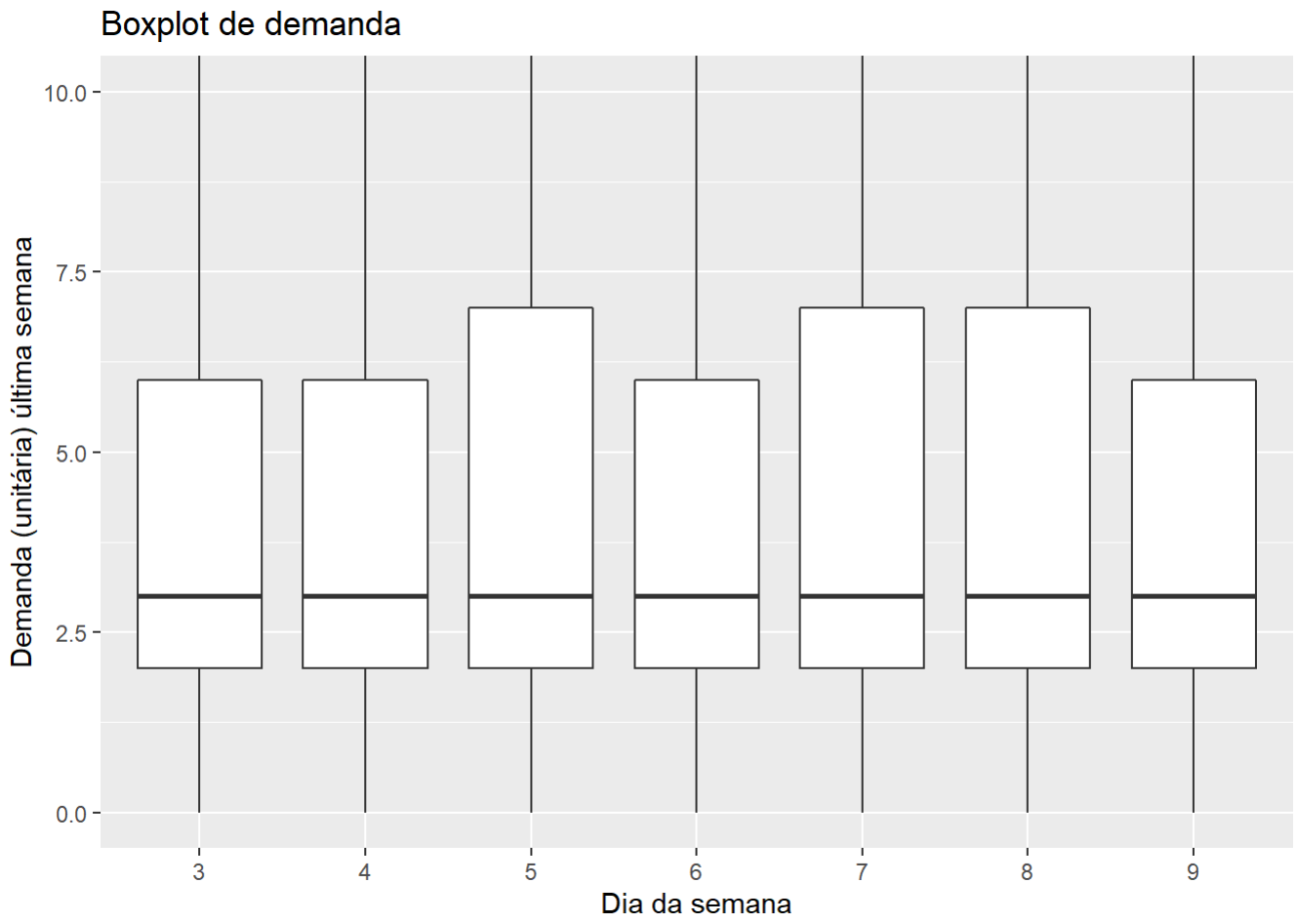

```
g + geom_histogram(aes(Dev_uni_proxima), binwidth = 1) + coord_cartesian(xlim = c(0,5)) + labs(t
itle="Histograma de retornos", x="Retorno (unitária)")
```




```
g + geom_histogram(aes(Dev_proxima), binwidth = 1) + coord_cartesian(xlim = c(0,5)) + labs(title
="Histograma de retornos", x="Retorno (pesos)")
```

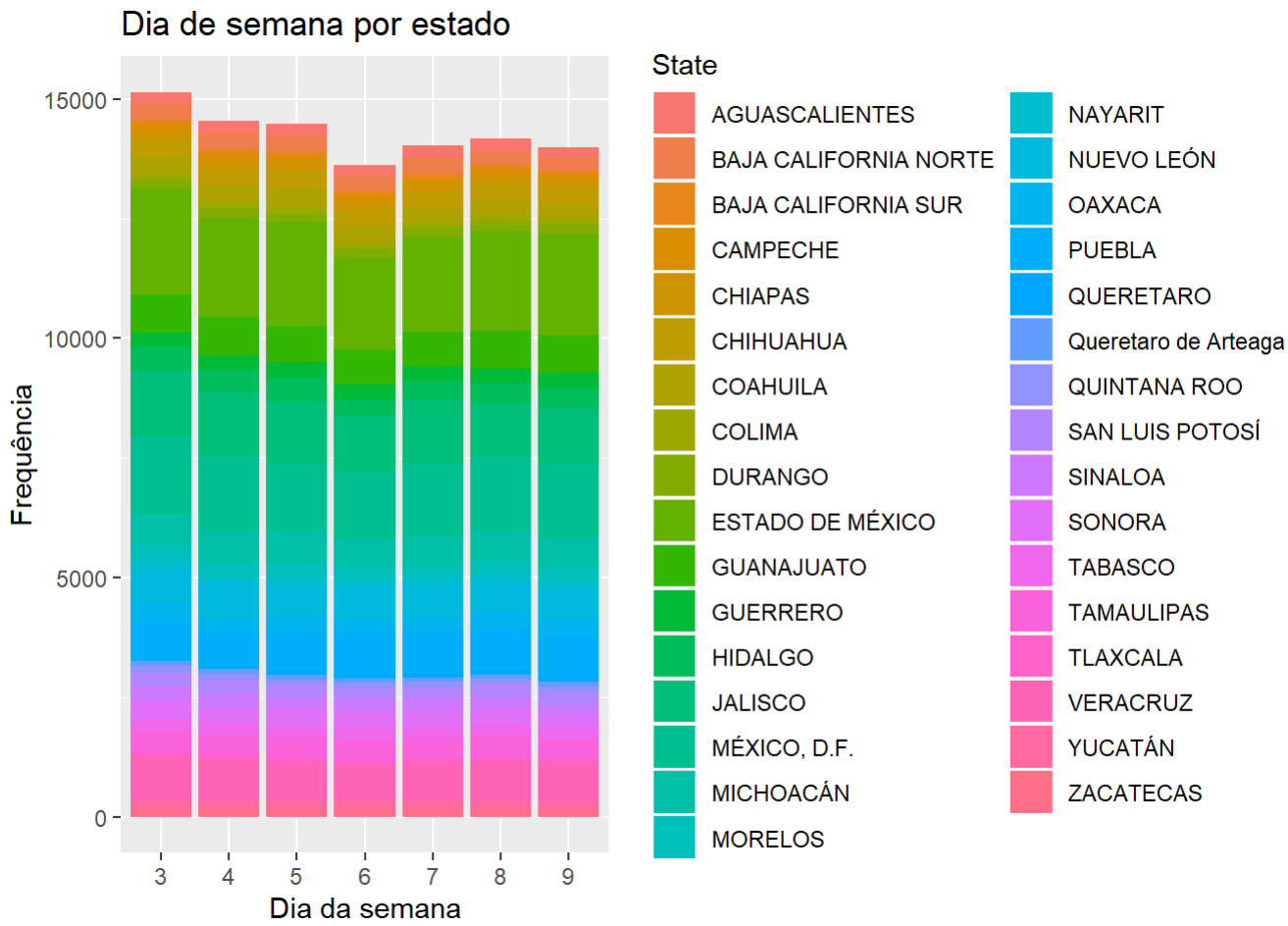



```
#Demanda muito concentrada (parecendo uma distribuição exponencial). Os retornos também
#concentrados, com muitos registros sem retorno. A base possui muitos outlyers, o que dificulta
#a visualizacao dos gráficos (precisa limitar os eixos)

#Dias da semana
g + geom_boxplot(aes(x=Semana, y=Venta_uni_hoy)) + coord_cartesian(ylim = c(0,10)) + labs(title=
"Boxplot de demanda",x="Dia da semana", y="Demanda (unitária) última semana")
```

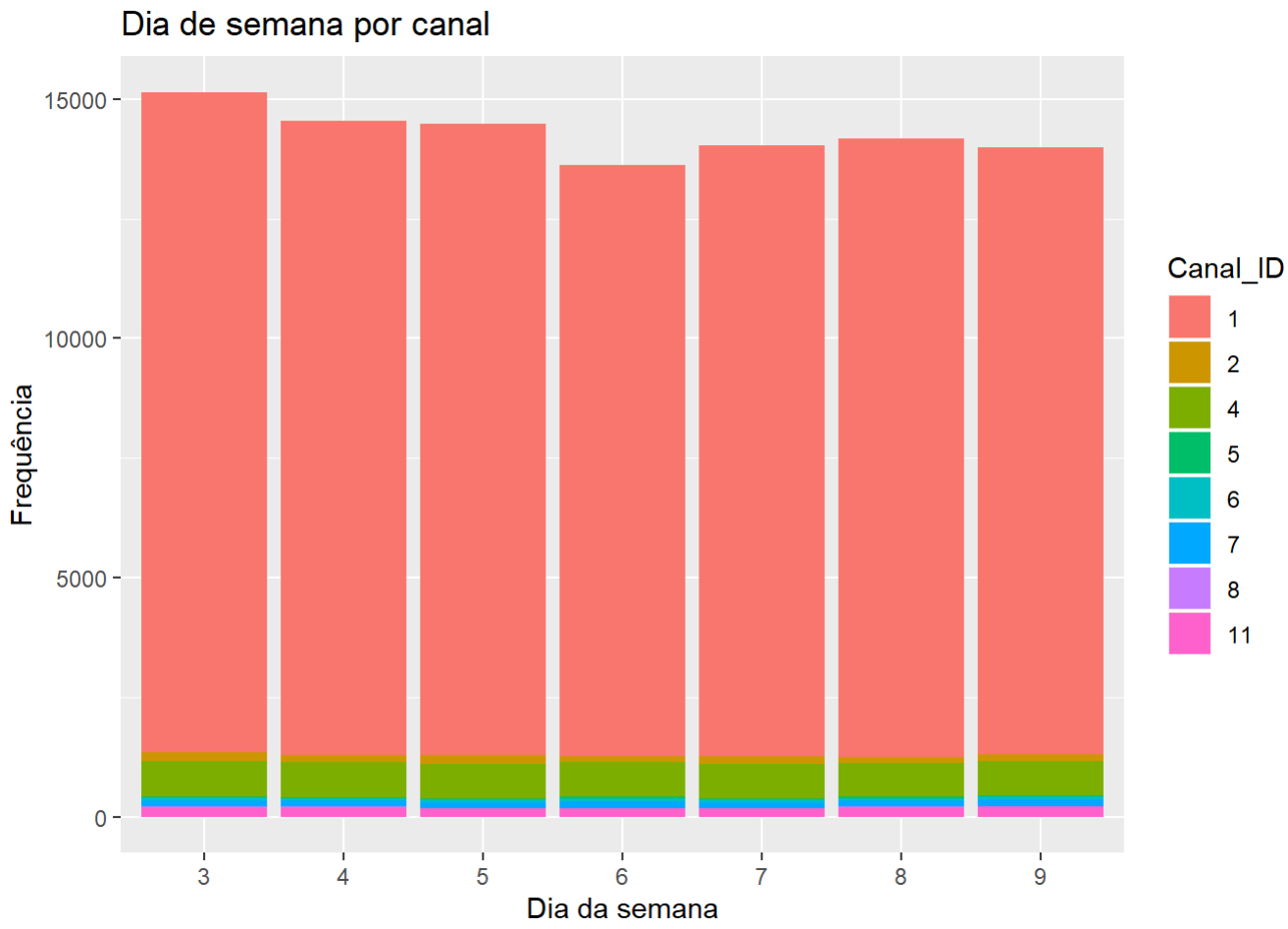



```
g + geom_boxplot(aes(x=Semana, y=Venta_hoy)) + coord_cartesian(ylim = c(0,75)) + labs(title="Box
plot de demanda", x="Dia da semana", y="Demanda (pesos) última semana")
```

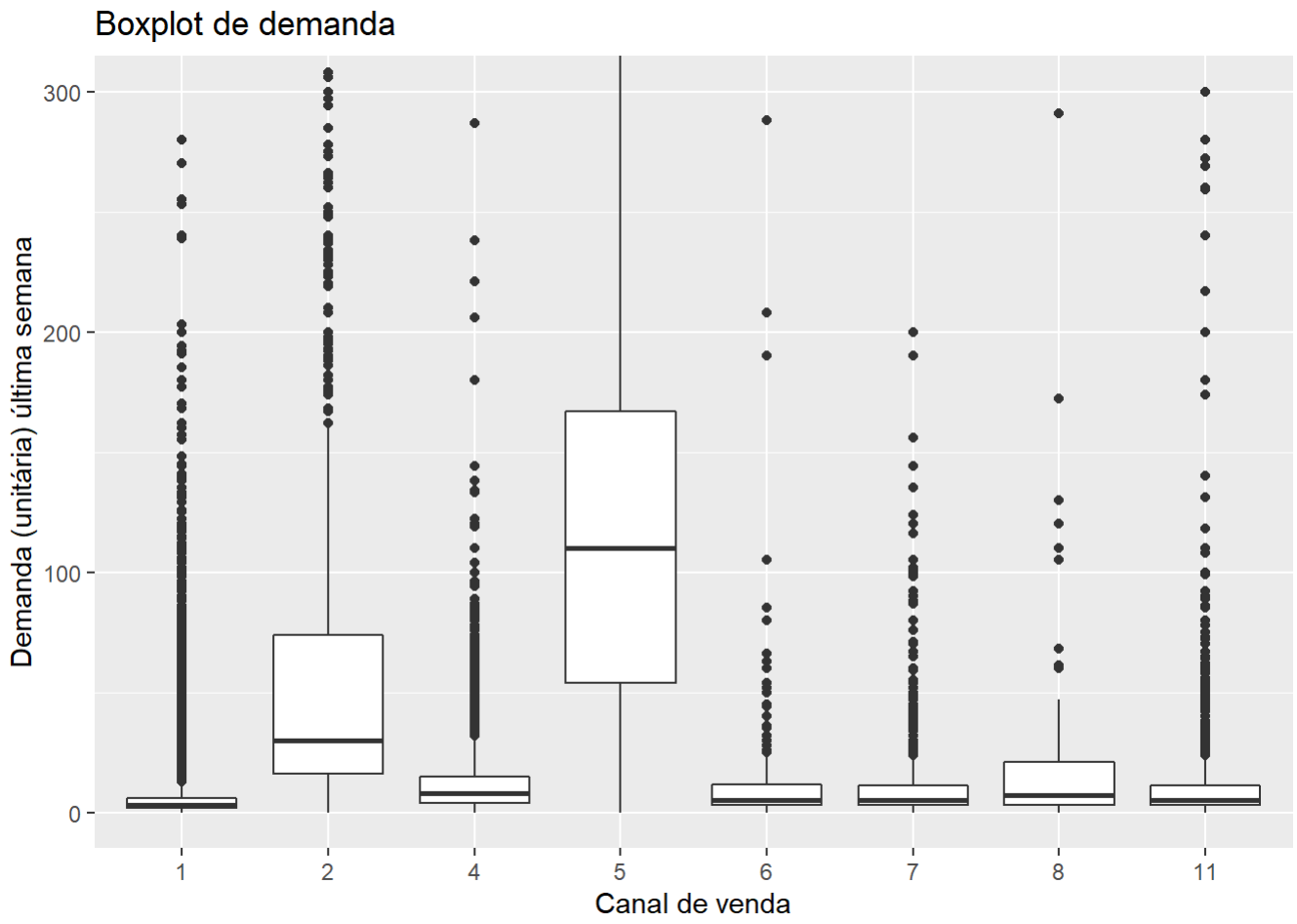

```
g + geom_bar(aes(Semana, fill=State)) + labs(title="Dia de semana por estado", x="Dia da semana"
, y="Frequência")
```



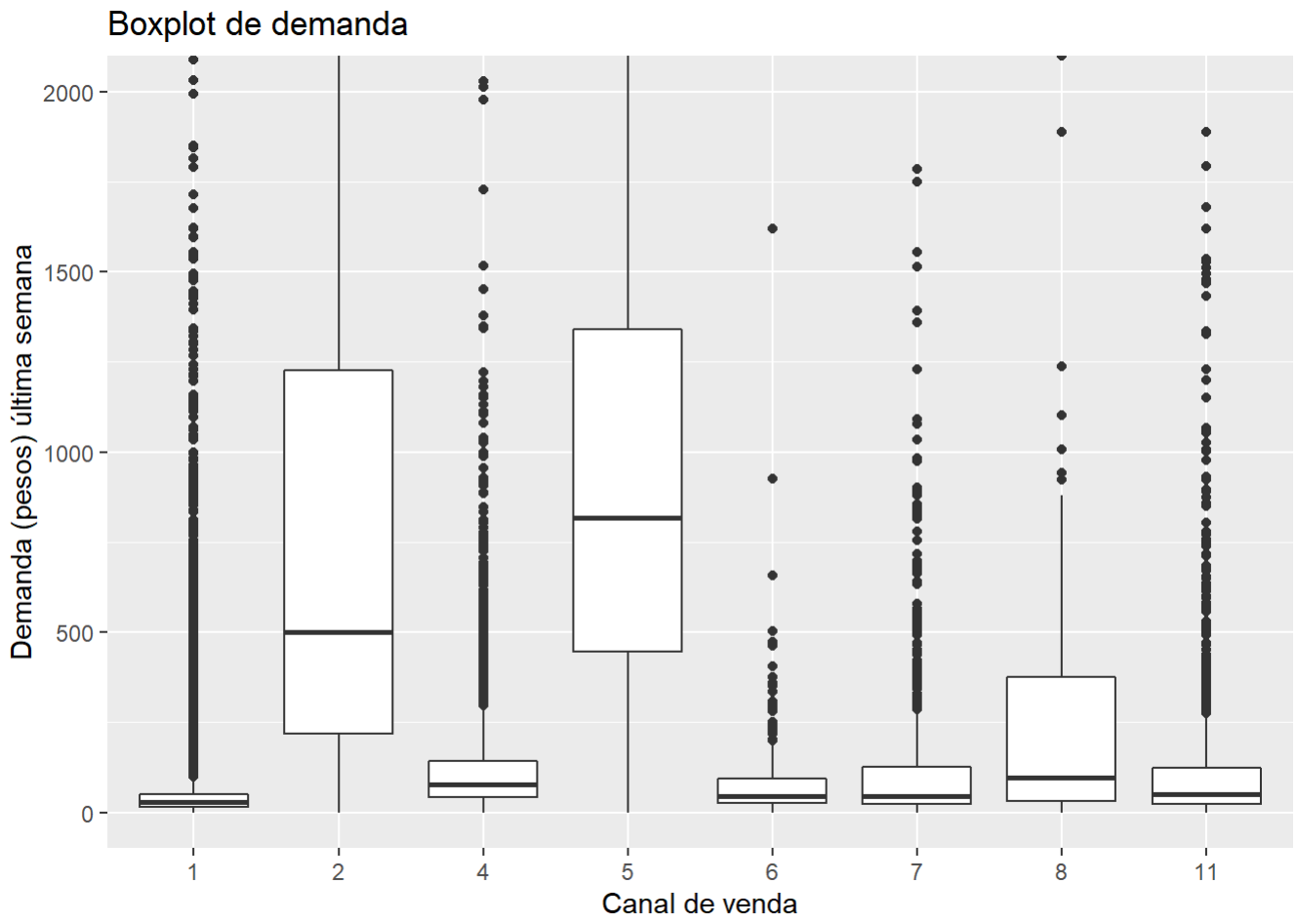

```
g + geom_bar(aes(Semana, fill=Canal_ID)) + labs(title="Dia de semana por canal", x="Dia da seman
a", y="Frequência")
```



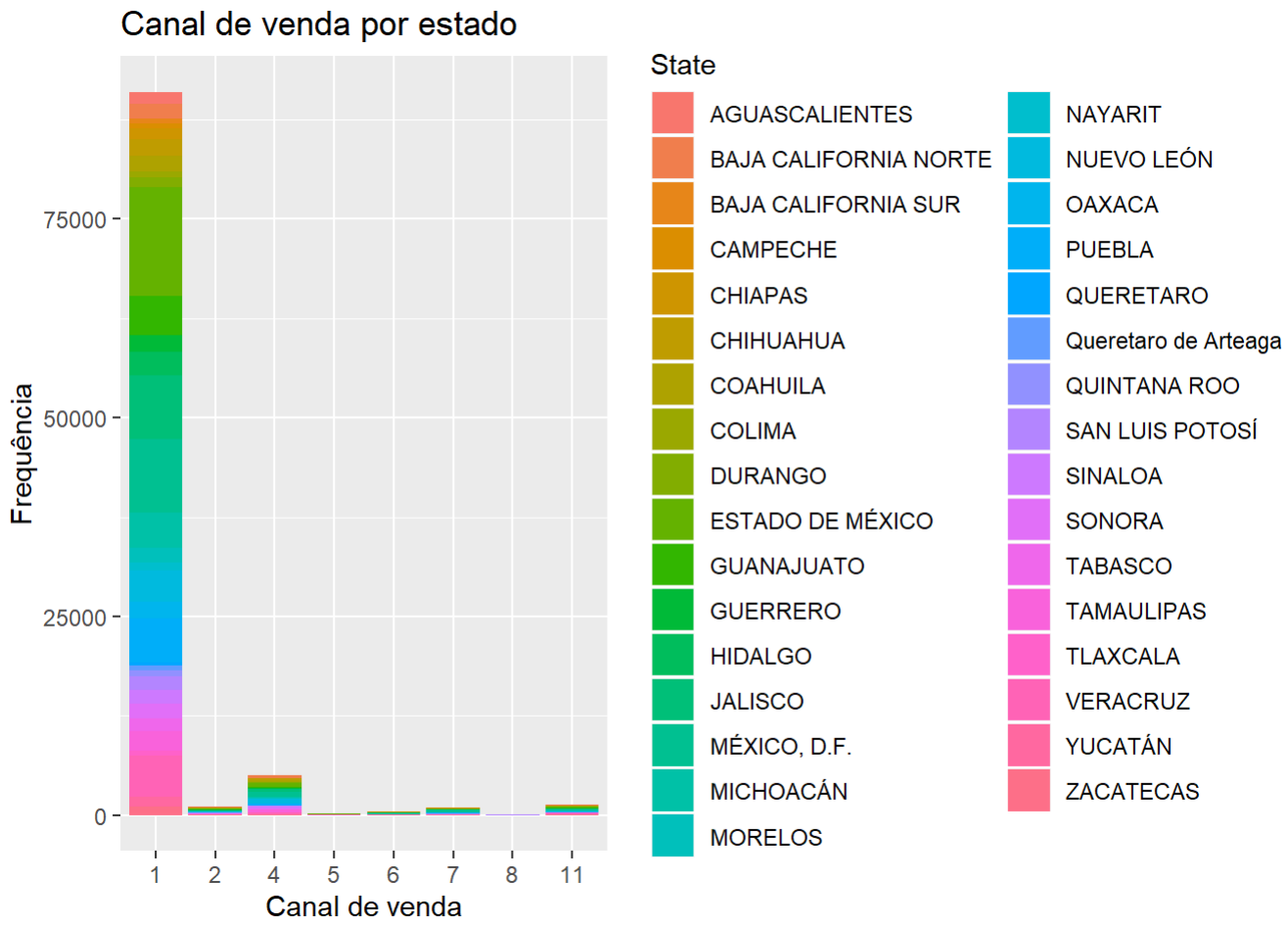

```
#Demanda parece uniforme ao longo dos dias, assim como o número de aparições de cada dia e sua
#Localização e canal de venda (concentrada no ID 1).

#Canal de venda
g + geom_boxplot(aes(x=Canal_ID, y=Venta_uni_hoy)) + coord_cartesian(ylim = c(0,300)) + labs(tit
le="Boxplot de demanda",x="Canal de venda", y="Demanda (unitária) última semana")
```




```
g + geom_boxplot(aes(x=Canal_ID, y=Venta_hoy)) + coord_cartesian(ylim = c(0,2000))+ labs(title=
"Boxplot de demanda", x="Canal de venda", y="Demanda (pesos) última semana")
```

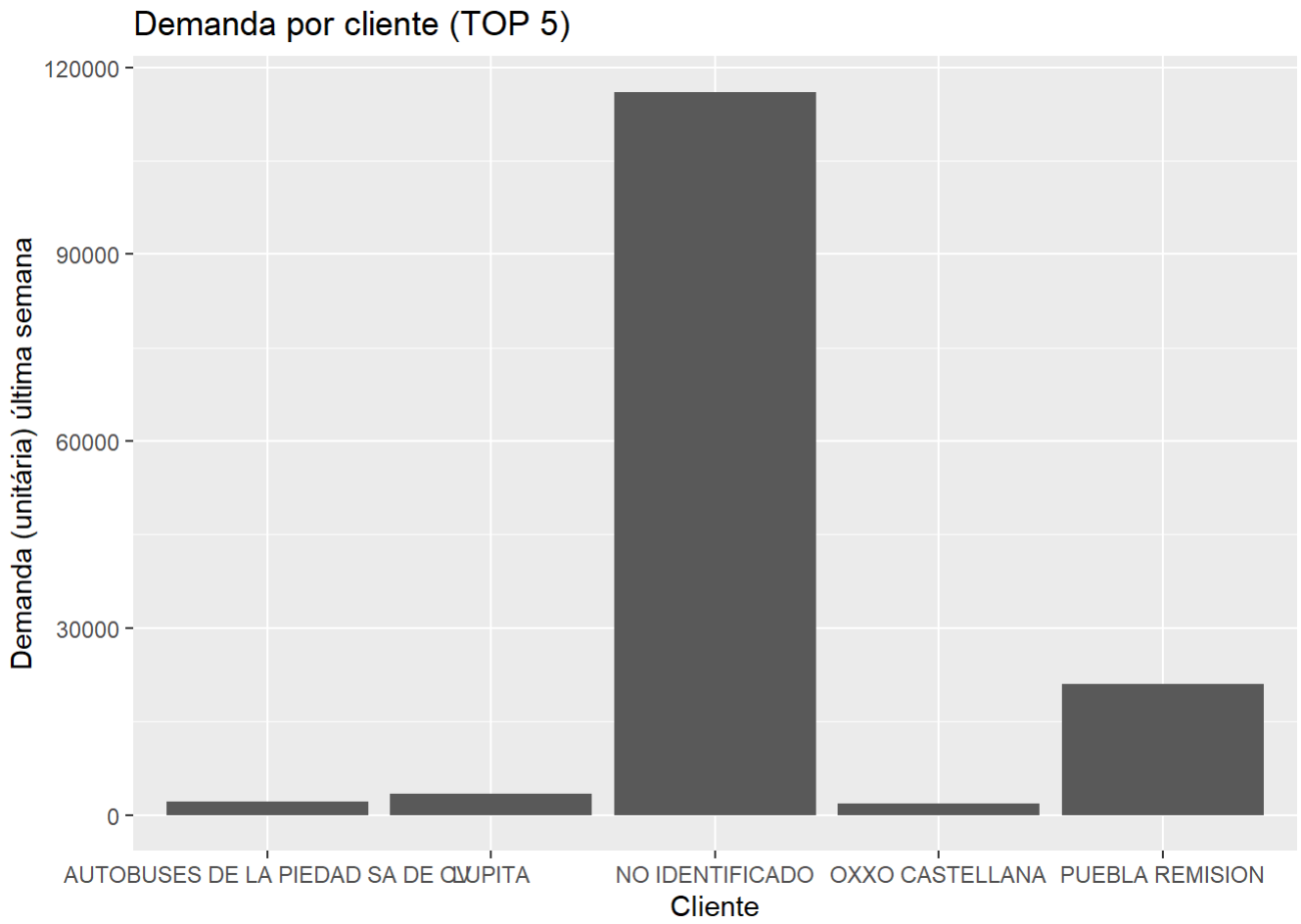



```
g + geom_bar(aes(Canal_ID, fill=State)) + labs(title="Canal de venda por estado", x="Canal de ve
nda", y="Frequência")
```

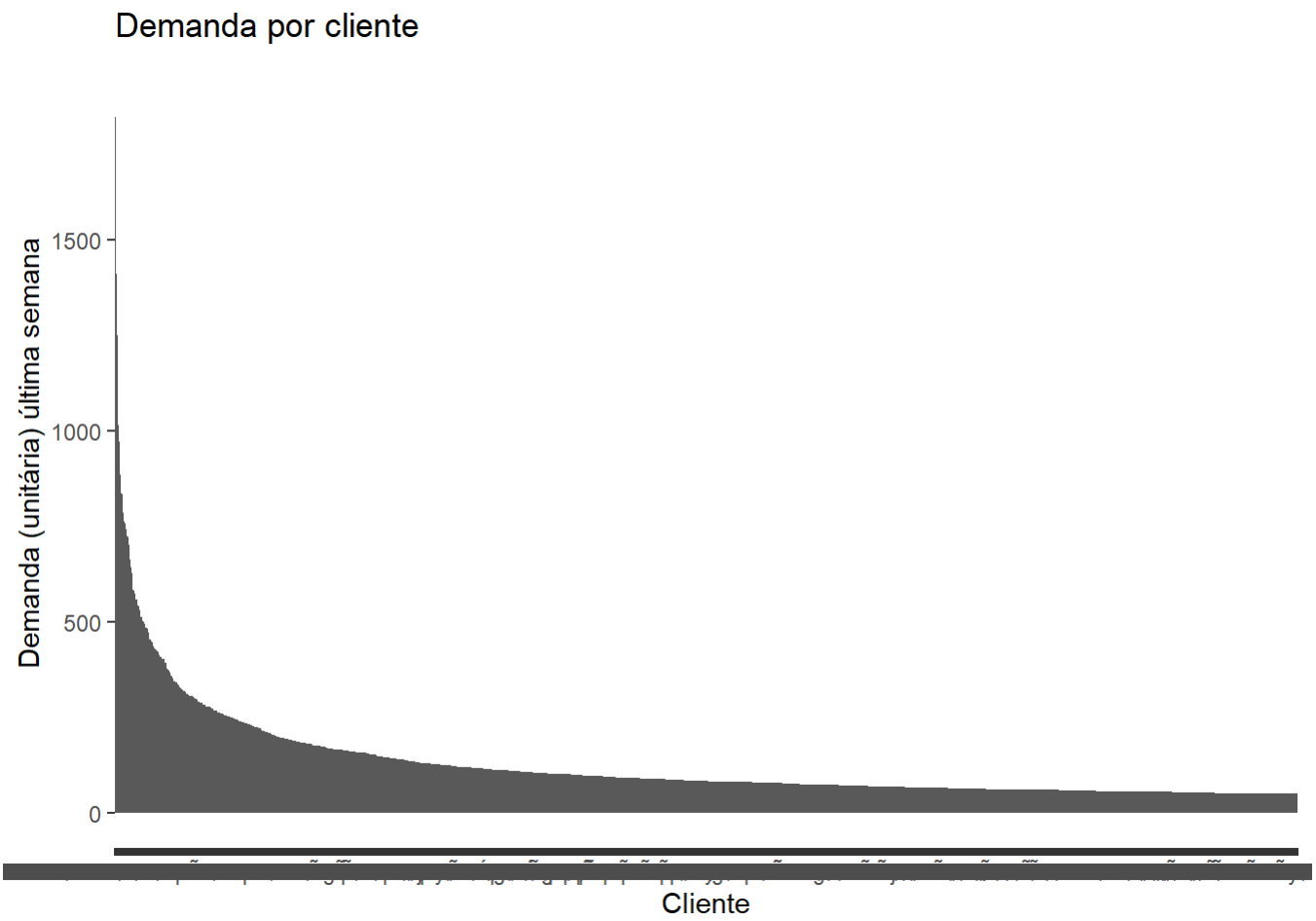



```
#Apesar do ID 1 concentrar a maior parte dos registros na base e possuir pouca variabilidade
#na demanda, outros canais como 2 e 5 apresentam muita variabilidade.
#Já é possível também perceber concentração de alguns estados na base (cores em tom de verde)

#Cliente
df %>%
  group_by(NombreCliente) %>%
  summarize(Venta_uni_hoy = sum(Venta_uni_hoy)) %>%
  arrange(desc(Venta_uni_hoy)) %>%
  top_n(5) %>%
  ggplot + geom_bar(aes(x=NombreCliente, y=Venta_uni_hoy),stat="identity")  + labs(title="Demand
a por cliente (TOP 5)",x="Cliente", y="Demanda (unitária) última semana")
```

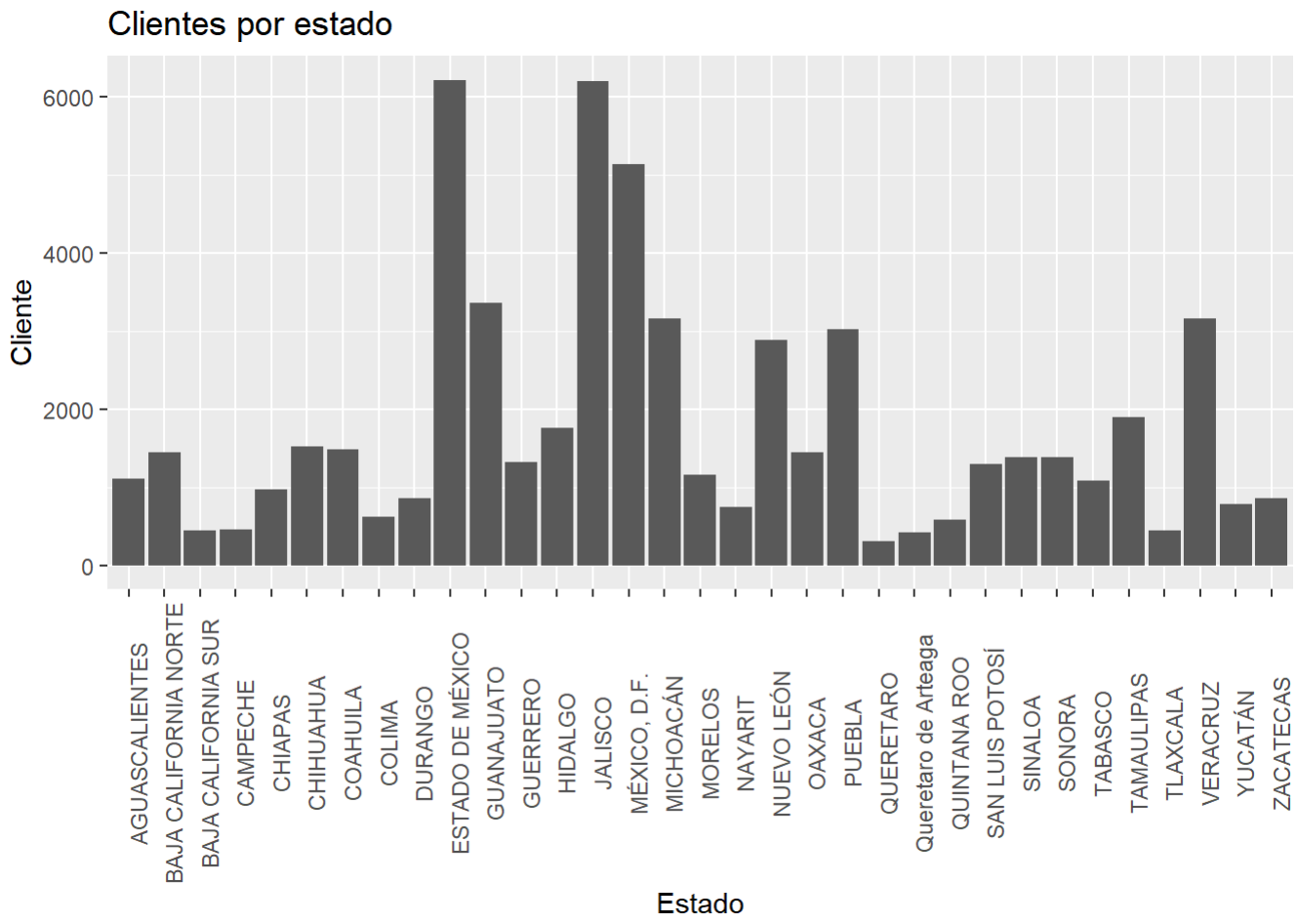
```
## Selecting by Venta_uni_hoy
```



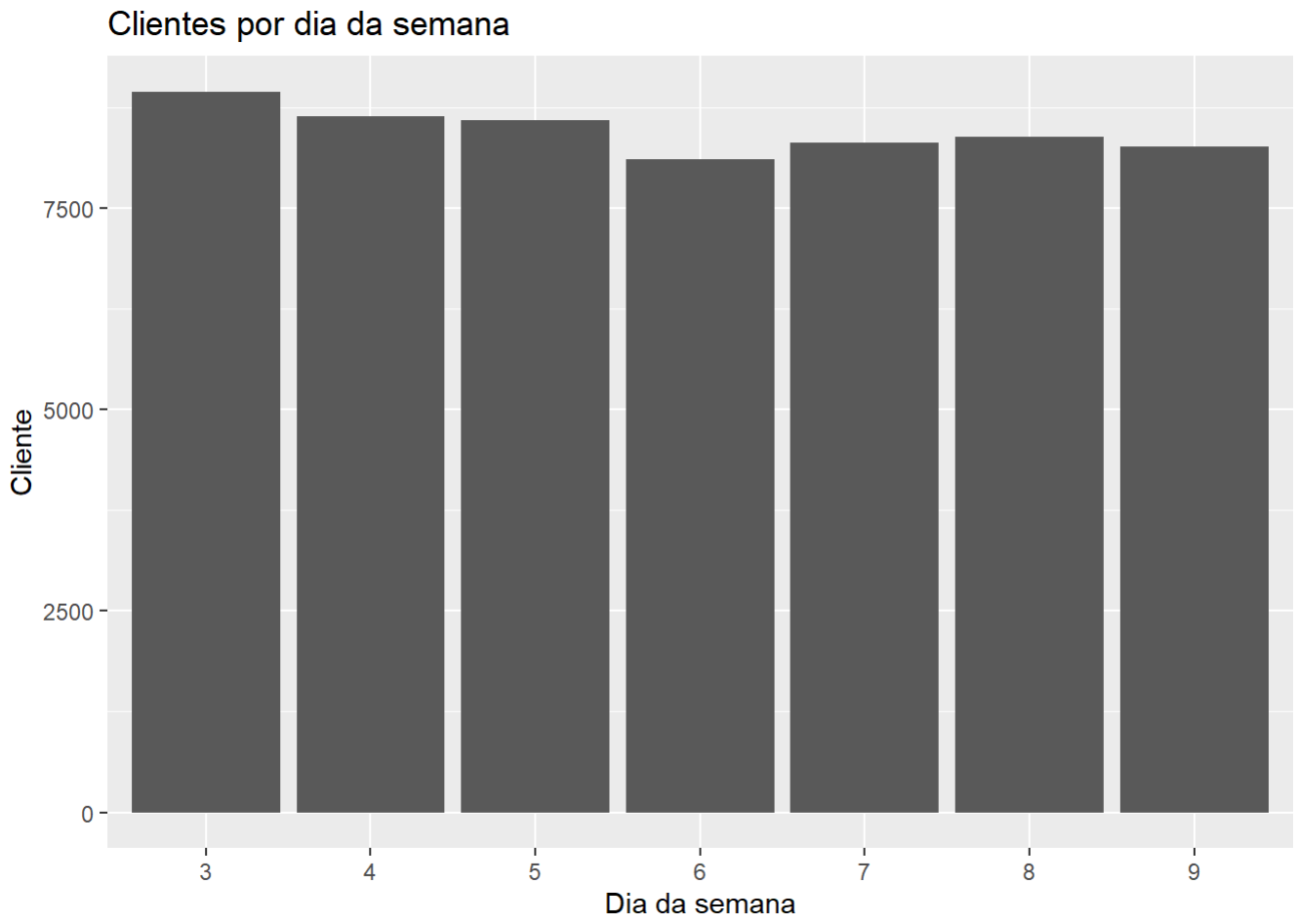

```
df %>%
  group_by(NombreCliente) %>%
  summarize(Venta_uni_hoy = sum(Venta_uni_hoy)) %>%
  arrange(desc(Venta_uni_hoy)) %>%
  slice(5:2000) %>%
  ggplot + geom_bar(aes(x=reorder(NombreCliente,-Venta_uni_hoy), y=Venta_uni_hoy),stat="identit
y")  + labs(title="Demanda por cliente",x="Cliente", y="Demanda (unitária) última semana")
```



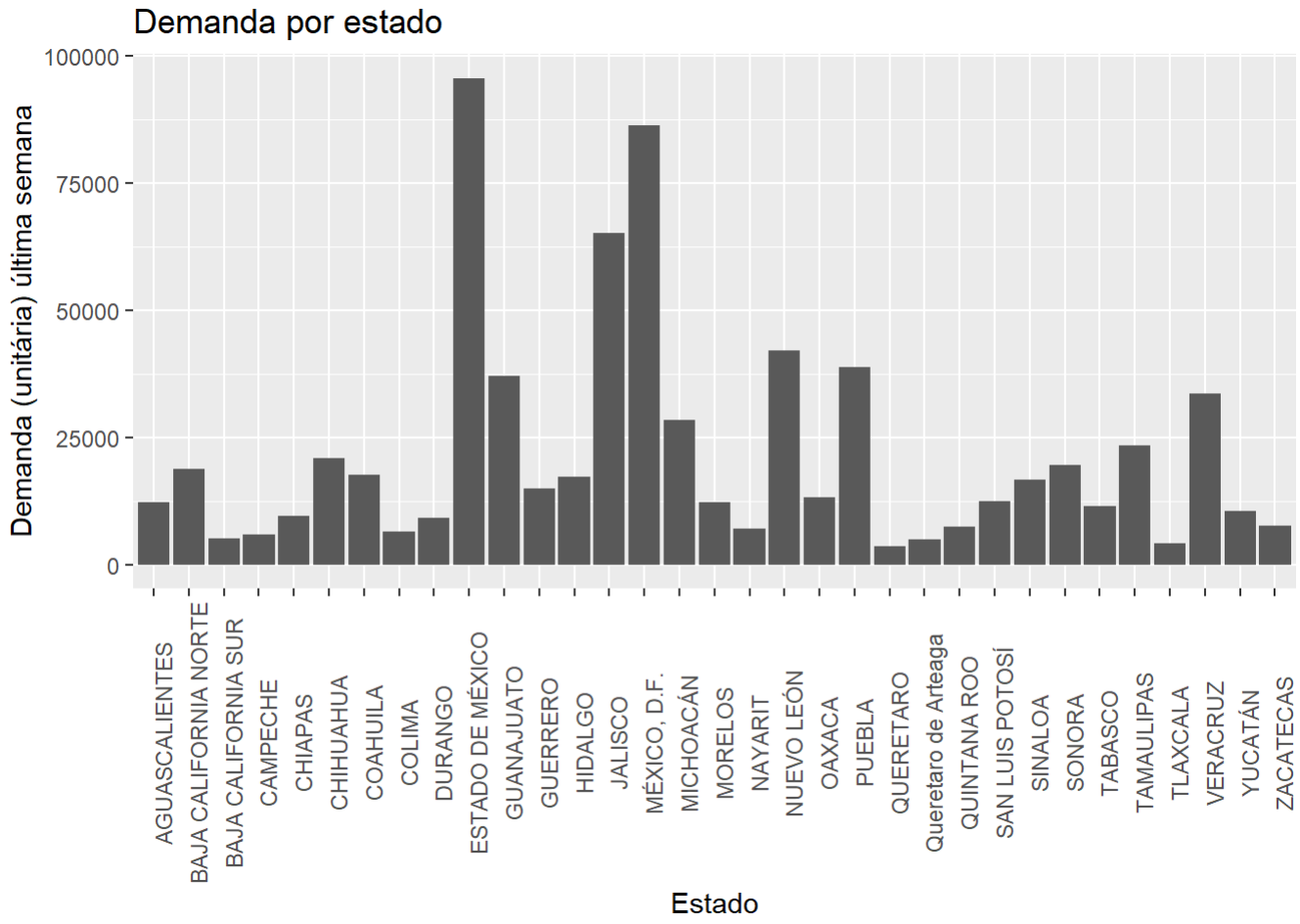

```
df %>%
  group_by(State) %>%
  summarize(Clientes = n_distinct(NombreCliente)) %>%
  ggplot + geom_bar(aes(x=State, y=Clientes),stat="identity") + theme(axis.text.x = element_text
(angle = 90)) + labs(title="Clientes por estado",x="Estado", y="Cliente")
```



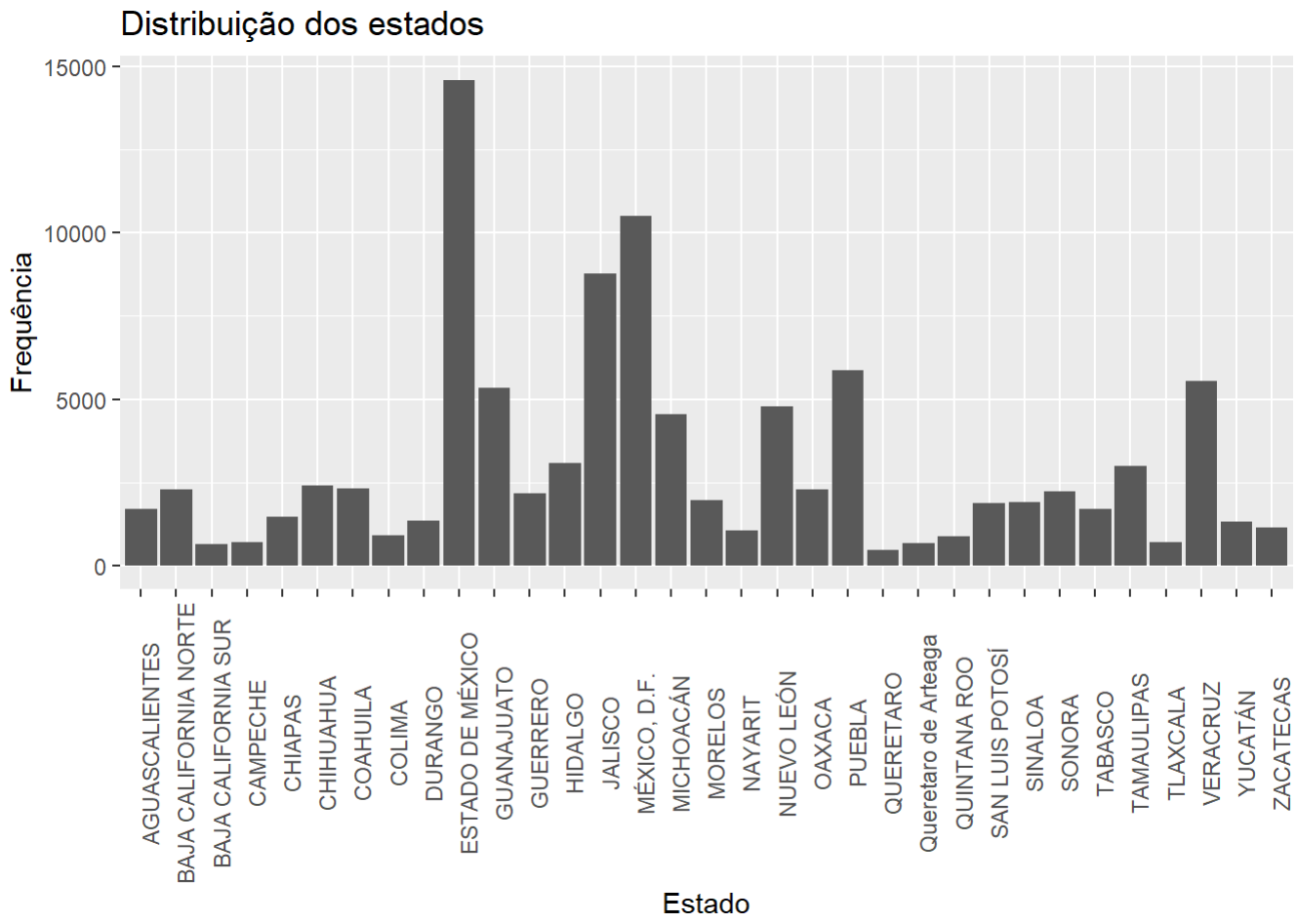

```
df %>%
  group_by(Semana) %>%
  summarize(Clientes = n_distinct(NombreCliente)) %>%
  ggplot + geom_bar(aes(x=Semana, y=Clientes),stat="identity")+ labs(title="Clientes por dia da
 semana",x="Dia da semana", y="Cliente")
```



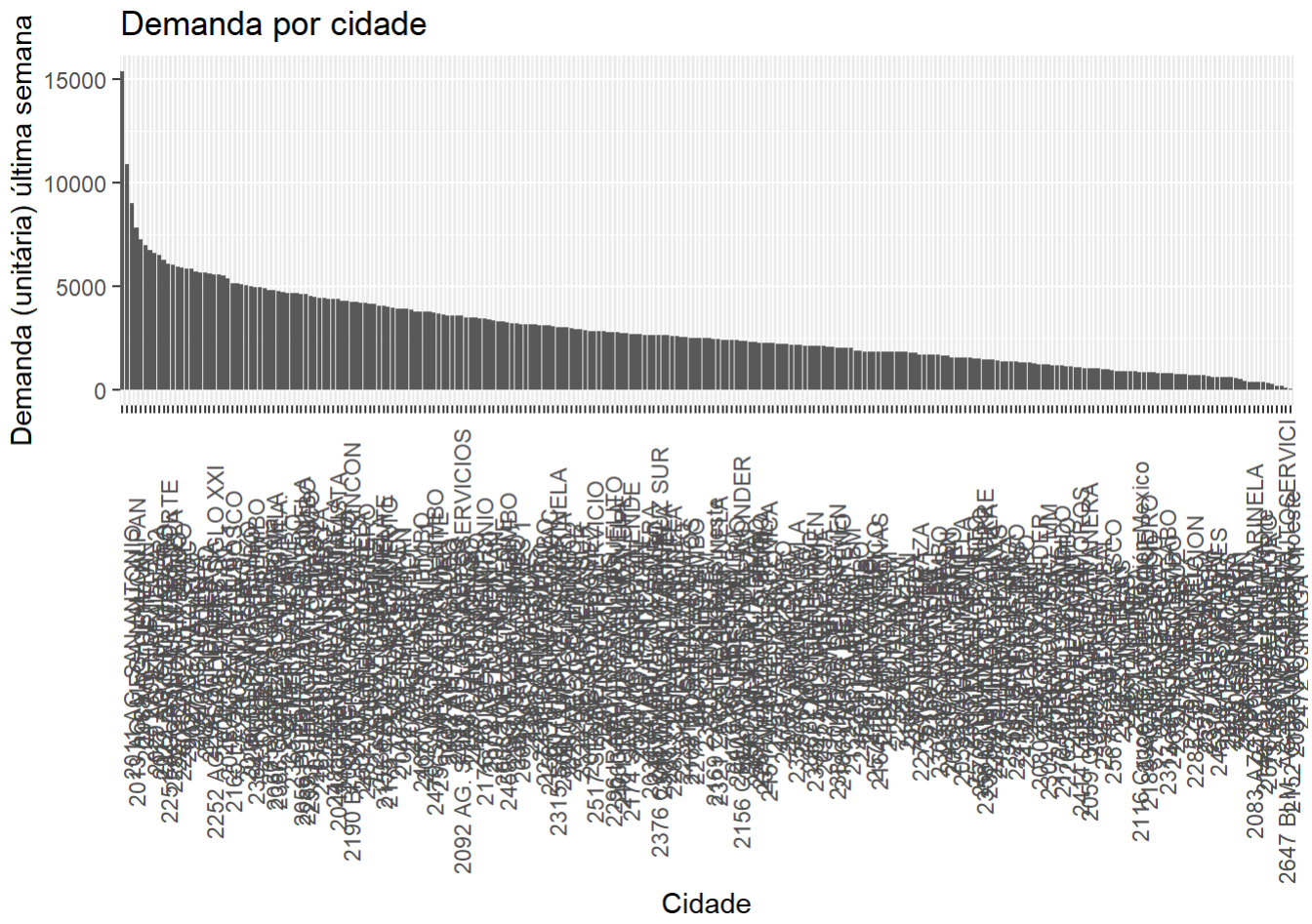

```
#Grande parte da demanda está em clientes não identificados, como é possível ver no top 5 client
es.
#O restante apresenta um decaimento que parece exponencial. Alguns estados possuem muito mais
#clientes que outro. Já a divisão por dia da semana parece uniforme

#Estado e cidade
df %>%
  group_by(State) %>%
  summarize(Venta_uni_hoy = sum(Venta_uni_hoy)) %>%
  arrange(desc(Venta_uni_hoy)) %>%
  ggplot + geom_bar(aes(x=State, y=Venta_uni_hoy),stat="identity") + theme(axis.text.x = element
_text(angle = 90))  + labs(title="Demanda por estado",x="Estado", y="Demanda (unitária) última s
emana")
```



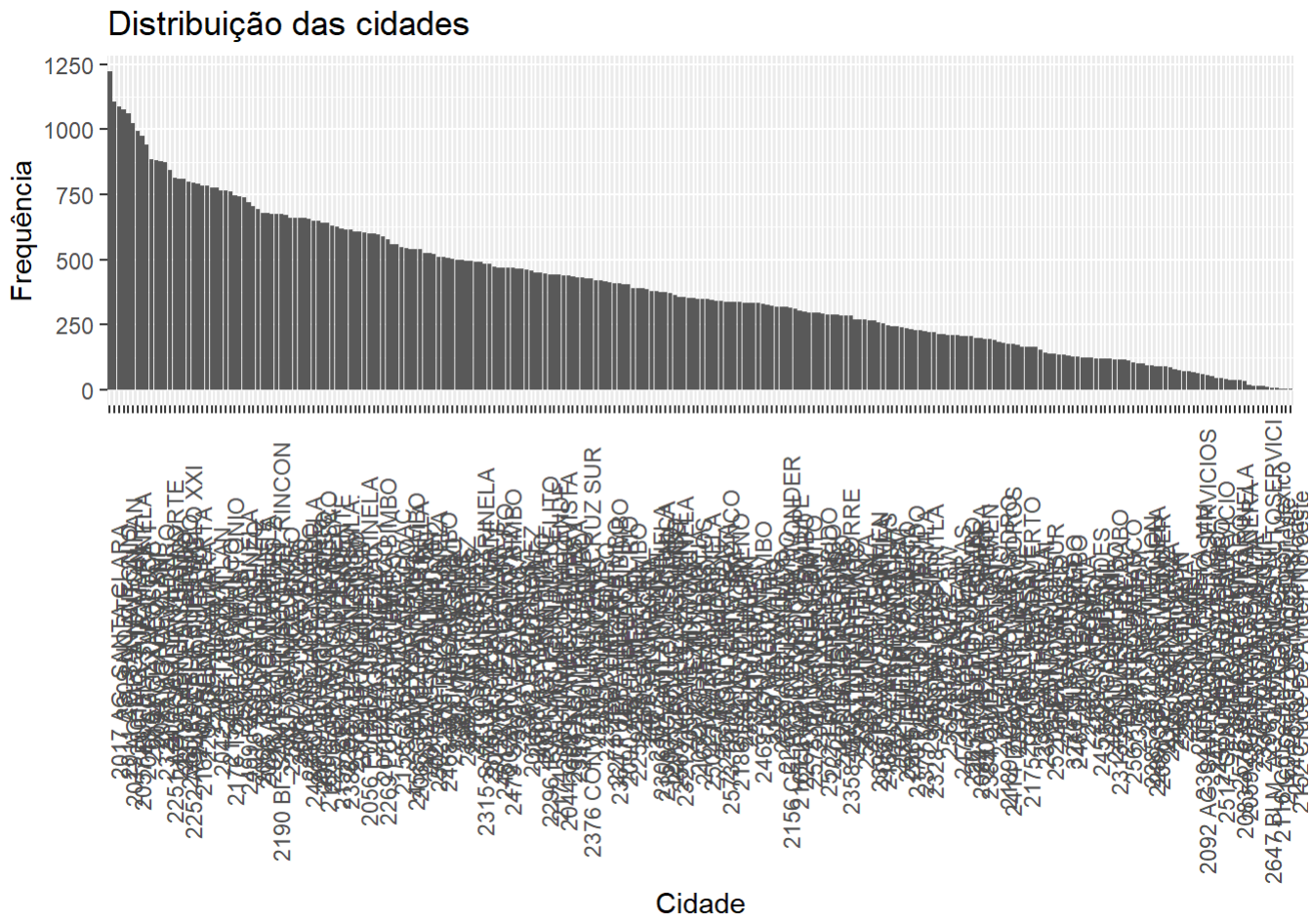

```
g + geom_bar(aes(State))+ theme(axis.text.x = element_text(angle = 90)) + theme(axis.text.x = el
ement_text(angle = 90))  + labs(title="Distribuição dos estados",x="Estado", y="Frequência")
```



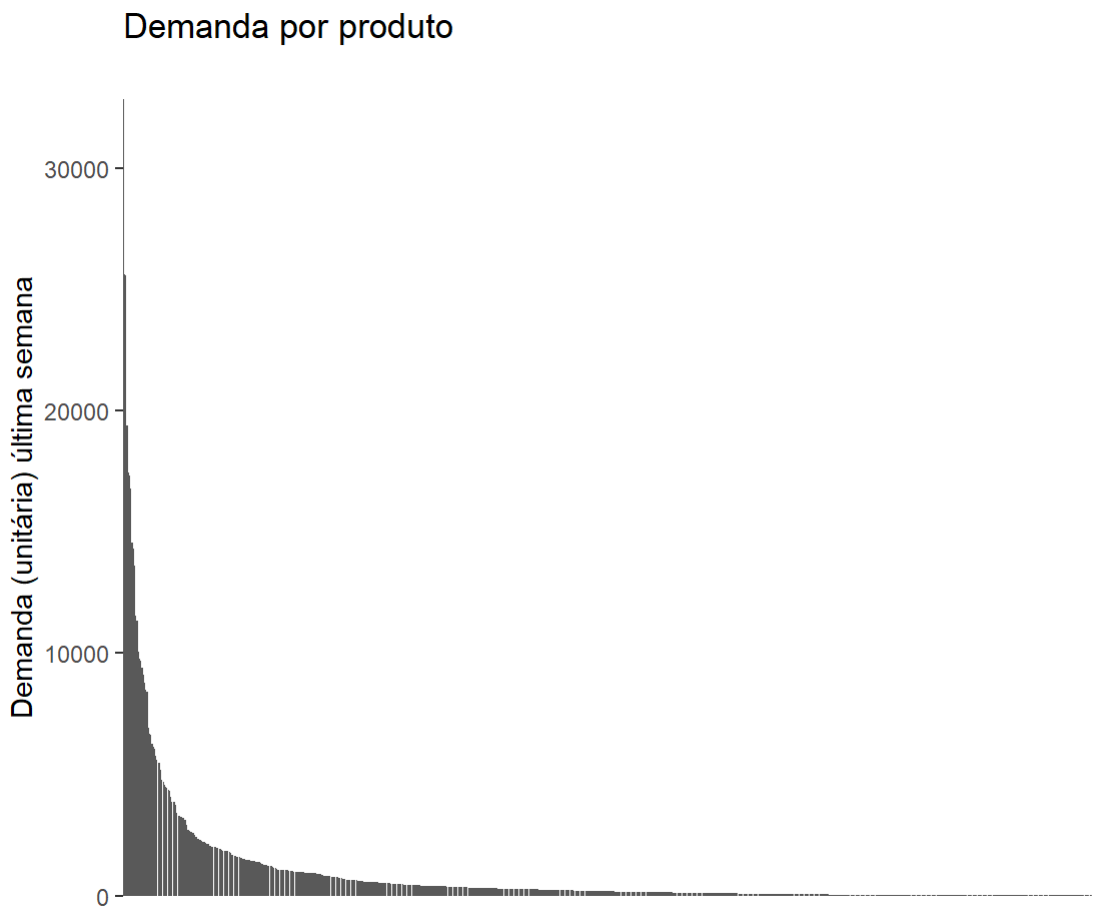

```
df %>%
  group_by(Town) %>%
  summarize(Venta_uni_hoy = sum(Venta_uni_hoy)) %>%
  arrange(desc(Venta_uni_hoy)) %>%
  ggplot + geom_bar(aes(x=reorder(Town,-Venta_uni_hoy), y=Venta_uni_hoy),stat="identity") + them
e(axis.text.x = element_text(angle = 90)) + theme(axis.text.x = element_text(angle = 90))  + lab
s(title="Demanda por cidade",x="Cidade", y="Demanda (unitária) última semana")
```



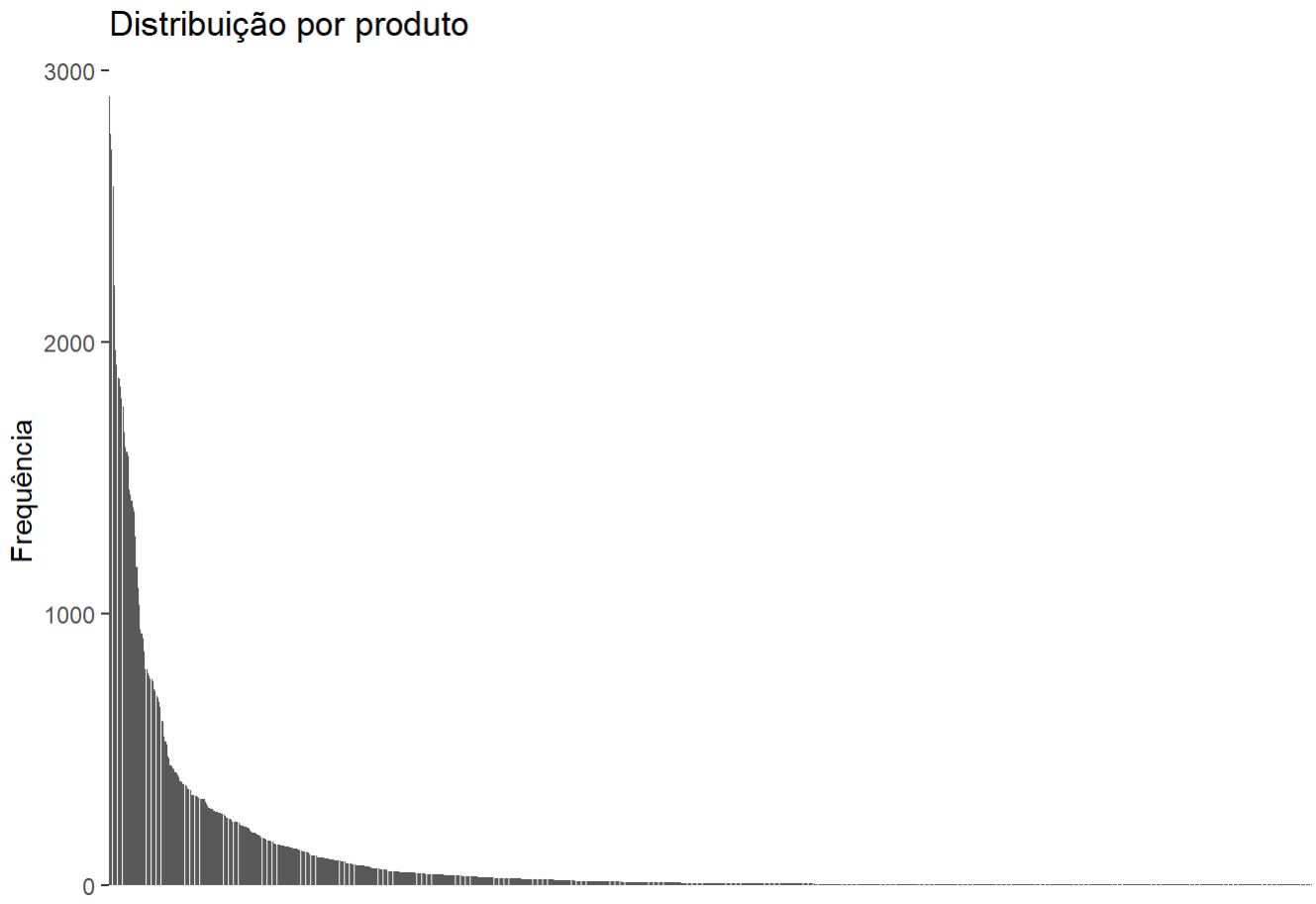

```
g + geom_bar(aes(x=reorder(Town,-table(Town)[Town]))) + theme(axis.text.x = element_text(angle = 90))  + labs(title="Distribuição das cidades",x="Cidade", y="Frequência")
```



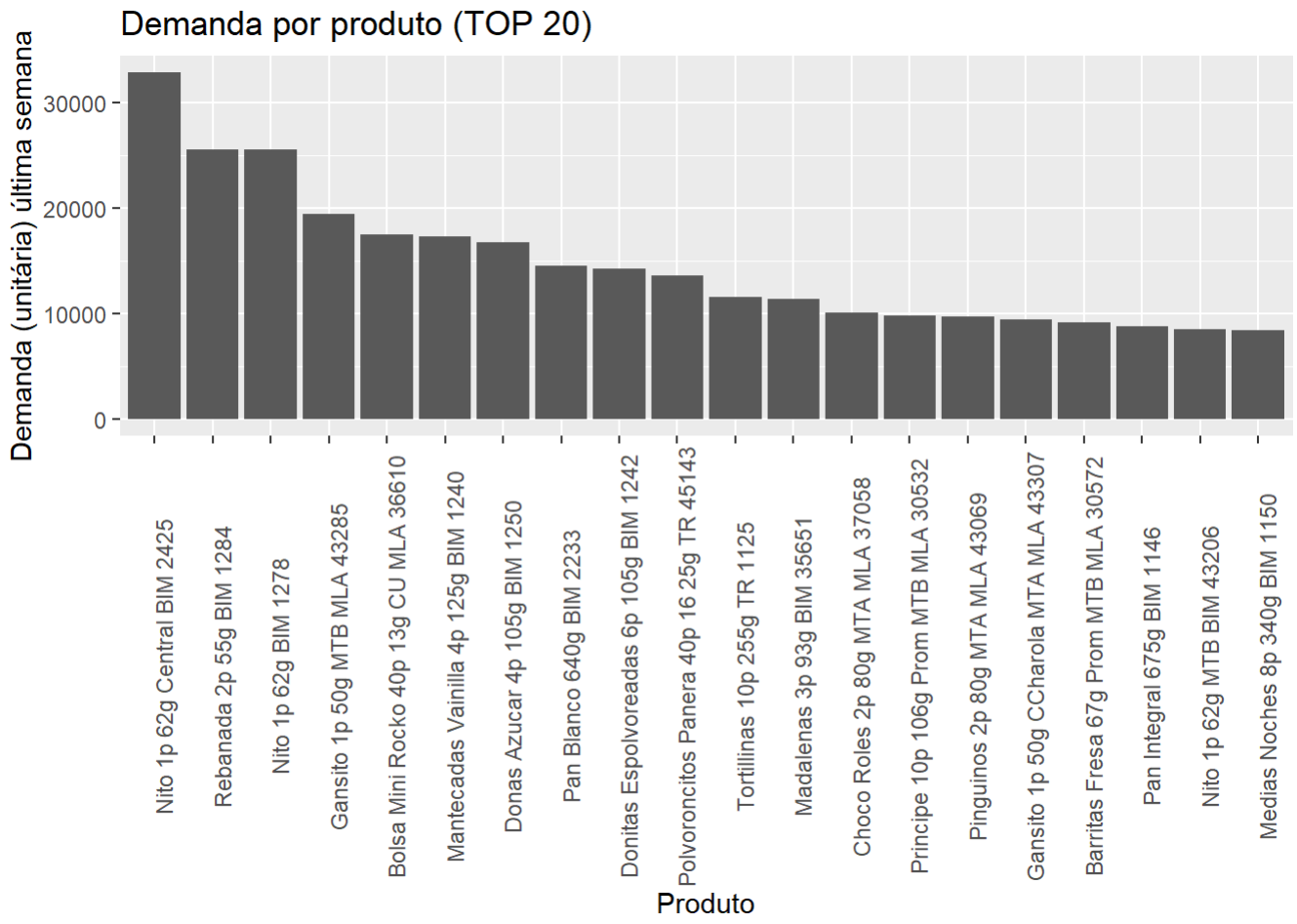

```
#A demanda segue a aparicão na base das cidades e estados. É possível verificar que poucas cidad
es
#e estados dominam a demanda. Três estados (Estado do Mexico, Jalisco e Mexico DF) representam
#as maiores demandas

#Produto
df %>%
  group_by(NombreProducto) %>%
  summarize(Venta_uni_hoy = sum(Venta_uni_hoy)) %>%
  arrange(desc(Venta_uni_hoy)) %>%
  ggplot + geom_bar(aes(x=reorder(NombreProducto,-Venta_uni_hoy), y=Venta_uni_hoy),stat="identit
y") + theme(axis.title.x=element_blank(),

axis.text.x=element_blank(),

axis.ticks.x=element_blank())  + labs(title="Demanda por produto", y="Demanda (unitária) última
 semana")
```



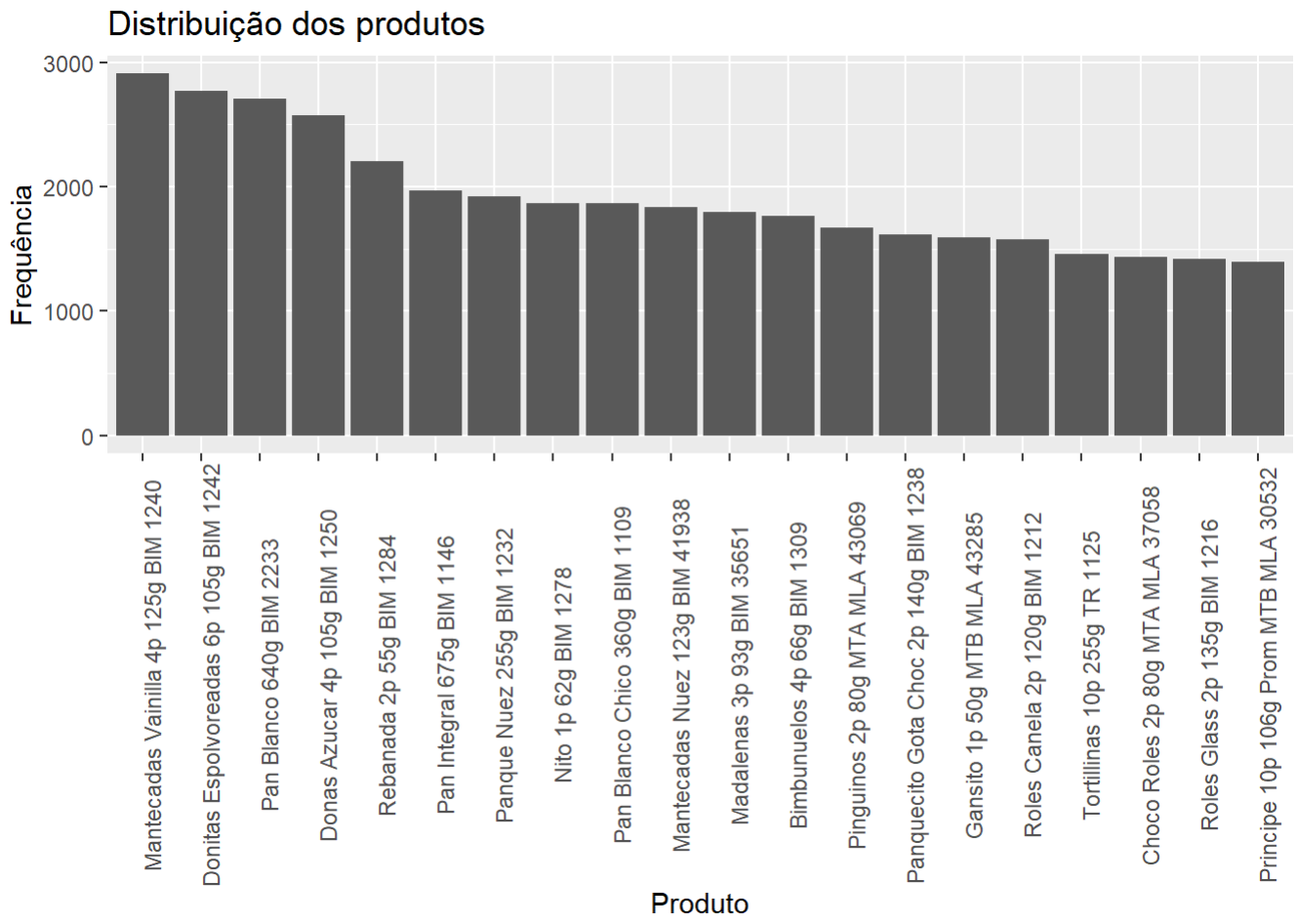

```
g + geom_bar(aes(x=reorder(NombreProducto,-table(NombreProducto)[NombreProducto])))+ theme(axis.
title.x=element_blank(),

axis.text.x=element_blank(),

axis.ticks.x=element_blank())  + labs(title="Distribuição por produto", y="Frequência")
```



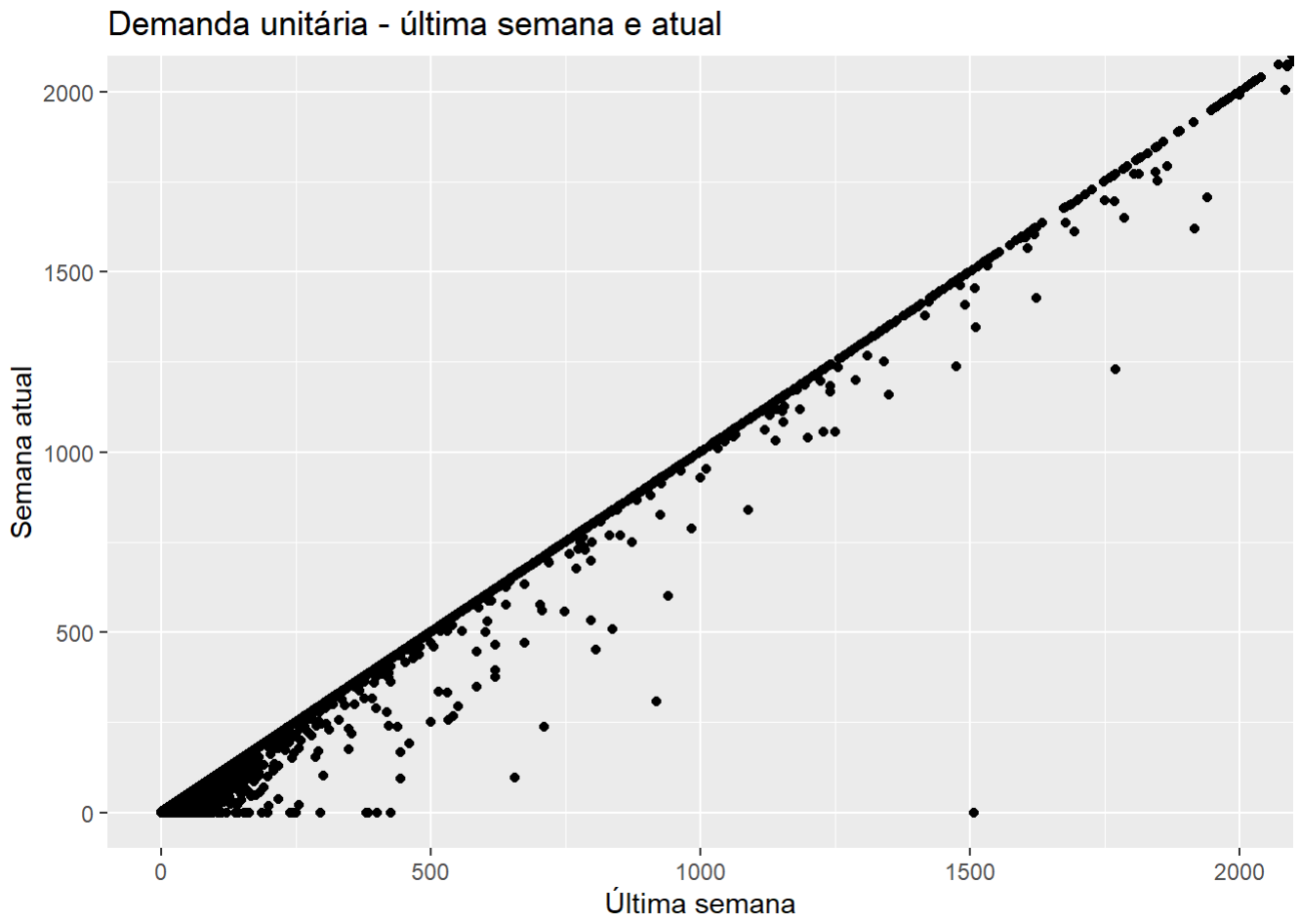

```
df %>%
  group_by(NombreProducto) %>%
  summarize(Venta_uni_hoy = sum(Venta_uni_hoy)) %>%
  arrange(desc(Venta_uni_hoy)) %>%
  slice(1:20) %>%
  ggplot + geom_bar(aes(x=reorder(NombreProducto,-Venta_uni_hoy), y=Venta_uni_hoy),stat="identit
y") + theme(axis.text.x = element_text(angle = 90))  + labs(title="Demanda por produto (TOP 20)"
,x="Produto", y="Demanda (unitária) última semana")
```




```
df %>%
  group_by(NombreProducto) %>%
  summarize(contar = n()) %>%
  arrange(desc(contar)) %>%
  slice(1:20) %>%
  ggplot + geom_bar(aes(x=reorder(NombreProducto,-contar), y=contar),stat="identity") + theme(ax
is.text.x = element_text(angle = 90))  + labs(title="Distribuição dos produtos",x="Produto", y=
"Frequência")
```



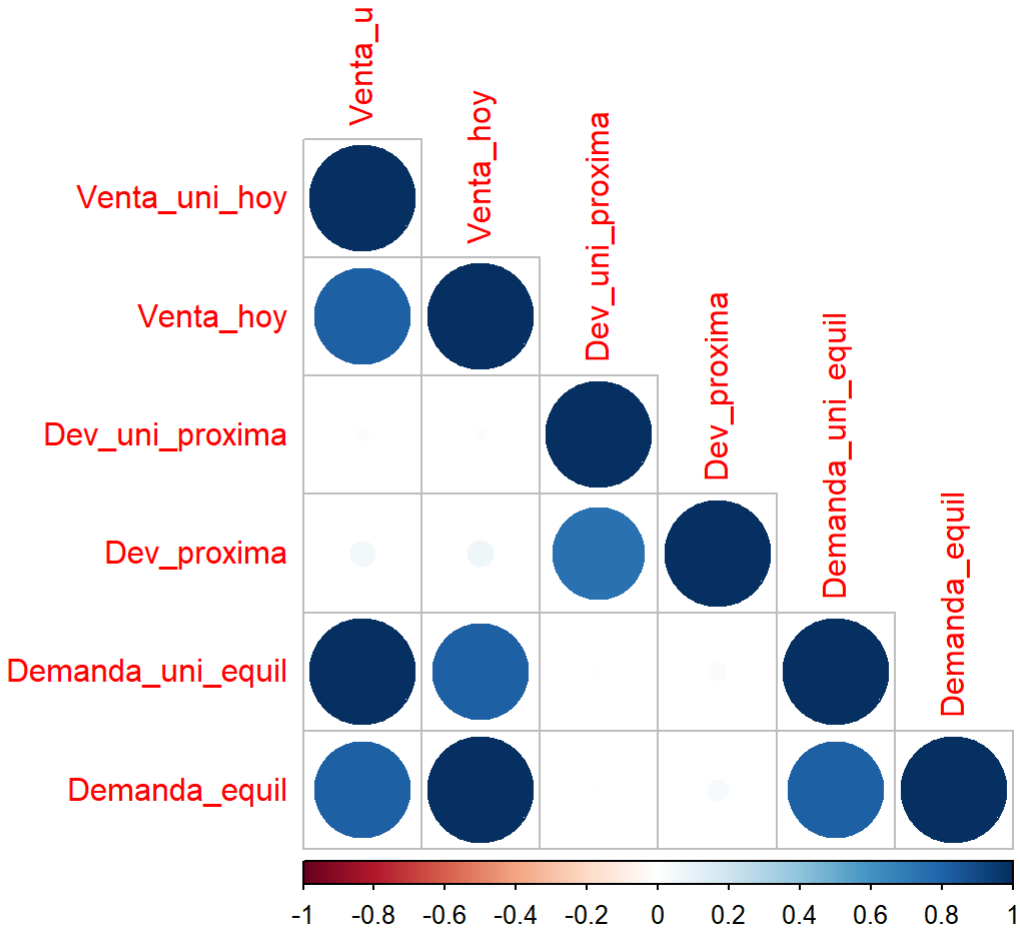

```
#Apesar de alguns produtos concentrarem grande parte da demanda, estes não são o que aparecem ma
is
#na base. Isto é, alguns produtos quando demandados o fazem em grande quantidade, mesmo que em
#pouca frequência, como pode se visto nos gráficos dos 20 produtos mais demandados em comparação
#aos 20 que mais aparecem na base

#Demanda anterior com demanda atual
g + geom_point(aes(x=Venta_hoy,y=Demanda_equil)) + coord_cartesian(xlim = c(0,2000),ylim = c(0,2
000))  + labs(title="Demanda unitária - última semana e atual",x="Última semana", y="Semana atua
l")
```




```
#As variáveis possuem uma correlacao muito alta, indicando que a demanda da semana anterior
#é fundamental para previsão da demanda atual. Isso ja era esperado dada a grande quantidade
#de semanas onde não há retornos, sendo a demanda atual igual a anterior
cor(df$Venta_hoy,df$Demanda_equil,method="pearson")
```

```
## [1] 0.999195
```

```
correlacoes = cor(df[c("Venta_uni_hoy","Venta_hoy","Dev_uni_proxima","Dev_proxima","Demanda_uni_
equil","Demanda_equil")],method="pearson")
corrplot(correlacoes, type="lower")
```


Venta_u
Venta_hoy
Dev_uni_proxima
Dev_proxima
Demanda_uni_equil
Demanda_equil
Venta_uni_hoy
Venta_hoy
Dev_uni_proxima
Dev_proxima
Demanda_uni_equil
Demanda_equil
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1

```
#Como a correlação entre a demanda unitária e por pesos é muito grande, utilizarei
#apenas a demanda por unidade, afim de evitar multicolinearidade

#MODELO PREDITIVO

#Trazendo novamente o dataset total e adicionando apenas a variável State
df <- fread("train.csv")

#Como a base é muito grande, vou criar uma base menor (1 milhão de linhas) para o projeto
#considerando apenas fins didáticos e a capacidade da minha máquina
df <- df[sample.int(nrow(df),1000000),]

#Adicionando estado
dfTownState <- fread("town_state.csv", encoding = "UTF-8")
df <- left_join(df,dfTownState,by="Agencia_ID")
rm(dfTownState)

#Transformando variáveis em categóricas
df$Semana <- as.factor(df$Semana)
df$Canal_ID <- as.factor(df$Canal_ID)

#Separação treino e teste
colunas <- c("Semana","State","Canal_ID","Venta_uni_hoy","Dev_uni_proxima","Demanda_uni_equil")
trainIndex <- createDataPartition(df$Demanda_uni_equil, p = .7, list = FALSE, times = 1)
trainSet <- df[trainIndex,colunas]
testSet  <- df[-trainIndex,colunas]
rm(df)

#Criando o modelo
modelo <- train(Demanda_uni_equil ~ ., data=trainSet, method="lm")
```

```
## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading
```

```
summary(modelo)
```

```
##
## Call:
## lm(formula = .outcome ~ ., data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -461.61   -0.01    0.00    0.03  330.15
##
## Coefficients:
##                               Estimate Std. Error   t value Pr(>|t|)
## (Intercept)                 -1.154e-02  9.937e-03    -1.161 0.245666
## Semana4                     -5.411e-03  4.580e-03    -1.181 0.237429
## Semana5                     -7.081e-04  4.621e-03    -0.153 0.878217
## Semana6                     -3.788e-03  4.677e-03    -0.810 0.418061
## Semana7                     -5.020e-03  4.651e-03    -1.079 0.280424
## Semana8                     -4.238e-04  4.649e-03    -0.091 0.927375
## Semana9                     -3.074e-03  4.642e-03    -0.662 0.507869
## `StateBAJA CALIFORNIA NORTE`  4.318e-02  1.251e-02     3.452 0.000557 ***
## `StateBAJA CALIFORNIA SUR`    3.907e-02  1.823e-02     2.143 0.032128 *
## StateCAMPECHE                 2.258e-02  1.790e-02     1.262 0.207103
## StateCHIAPAS                  4.839e-02  1.403e-02     3.450 0.000561 ***
## StateCHIHUAHUA                2.860e-02  1.231e-02     2.324 0.020108 *
## StateCOAHUILA                 2.267e-02  1.252e-02     1.811 0.070169 .
## StateCOLIMA                   3.266e-02  1.628e-02     2.007 0.044744 *
## StateDURANGO                  4.496e-02  1.449e-02     3.102 0.001920 **
## `StateESTADO DE MÉXICO`       4.193e-02  1.003e-02     4.182 2.89e-05 ***
## StateGUANAJUATO               3.514e-02  1.094e-02     3.213 0.001314 **
## StateGUERRERO                 7.682e-02  1.274e-02     6.031 1.63e-09 ***
## StateHIDALGO                  3.285e-02  1.184e-02     2.775 0.005516 **
## StateJALISCO                  3.987e-02  1.038e-02     3.839 0.000123 ***
## `StateMÉXICO, D.F.`           3.713e-02  1.025e-02     3.623 0.000291 ***
## StateMICHOACÁN                5.007e-02  1.115e-02     4.491 7.11e-06 ***
## StateMORELOS                  2.909e-02  1.312e-02     2.217 0.026598 *
## StateNAYARIT                  3.721e-02  1.532e-02     2.429 0.015129 *
## `StateNUEVO LEÓN`             3.342e-02  1.108e-02     3.017 0.002555 **
## StateOAXACA                   4.048e-02  1.268e-02     3.192 0.001414 **
## StatePUEBLA                   2.845e-02  1.080e-02     2.634 0.008439 **
## StateQUERETARO                6.781e-02  2.110e-02     3.215 0.001307 **
## `StateQueretaro de Arteaga`   5.301e-02  1.824e-02     2.906 0.003657 **
## `StateQUINTANA ROO`           4.391e-02  1.649e-02     2.664 0.007732 **
## `StateSAN LUIS POTOSÍ`        4.026e-02  1.330e-02     3.026 0.002474 **
## StateSINALOA                  3.817e-02  1.302e-02     2.933 0.003362 **
## StateSONORA                   4.119e-02  1.273e-02     3.236 0.001214 **
## StateTABASCO                  5.309e-02  1.339e-02     3.966 7.31e-05 ***
## StateTAMAULIPAS               1.973e-02  1.193e-02     1.654 0.098174 .
## StateTLAXCALA                 1.464e-02  1.759e-02     0.832 0.405139
## StateVERACRUZ                 3.362e-02  1.086e-02     3.096 0.001961 **
## StateYUCATÁN                  2.755e-02  1.439e-02     1.914 0.055584 .
## StateZACATECAS                3.173e-02  1.494e-02     2.124 0.033701 *
## Canal_ID2                   -1.483e-01  1.230e-02   -12.056  < 2e-16 ***
## Canal_ID4                    5.289e-02  5.793e-03     9.130  < 2e-16 ***
## Canal_ID5                   -2.039e+00  2.969e-02   -68.688  < 2e-16 ***
## Canal_ID6                    9.808e-02  2.017e-02     4.862 1.16e-06 ***
```

```
## Canal_ID7                    1.121e-01  1.347e-02     8.321  < 2e-16 ***
## Canal_ID8                    6.443e-01  4.272e-02    15.082  < 2e-16 ***
## Canal_ID9                    3.742e+00  1.047e+00     3.575 0.000351 ***
## Canal_ID11                   8.175e-02  1.108e-02     7.379 1.60e-13 ***
## Venta_uni_hoy                9.922e-01  6.234e-05 15916.726  < 2e-16 ***
## Dev_uni_proxima             -4.203e-01  5.275e-04  -796.864  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.046 on 699952 degrees of freedom
## Multiple R-squared:  0.9977, Adjusted R-squared:  0.9977
## F-statistic: 6.246e+06 on 48 and 699952 DF,  p-value: < 2.2e-16
```

```
#Pela ANOVA, podemos ver que as variáveis numéricas Venta_uni_hoy e Dev_uni_proxima
#possuem p-values muito pequenos, o que significa que são variáveis importantes
#para explicar a variável de saída. Já quando olhamos as variáveis categóricas,
#o resultado é menos expressivo, porém alguns dias da semana, canais e estados
#possuem p-values bem pequenos (nível de significância de 5%).

#Verificando a acurácia do modelo
scores <- data.frame(actual = testSet$Demanda_uni_equil,
                     prediction = predict(modelo, newdata = testSet))

#Calculando o erro quadrático logarítmico médio (RMSLE)
scores$prediction <- ifelse((scores$prediction)>0,scores$prediction,0)
rmsle(scores$actual, scores$prediction)
```

```
## [1] 0.092573
```

```
#O rmsle é baixo, mostrando que o modelo conseguiu uma boa acurácia.
```