

Case Interagente

Thiago Barral

1/31/2020

Forecast de demanda

Uma distribuidora de energia elétrica recebe diversos chamados durante o dia. Um dos desafios da operação é planejar a capacidade necessária (quantidade de equipes de campo) para as próximas 24 horas. E, para isso, é necessário estimar a demanda (quantidade chamados/OSs). O projeto irá construir um mecanismo de regressão/forecast para prever a demanda que vai ser recebida a cada hora, nas próximas 24 horas. O arquivo .csv enviado contém registros de OSs de 2019 • DataSol – data da solicitação • NumOS – ID do chamado/OS • Regional – ID da Regional

Carregando pacotes necessários para a avaliação

```
library(ggplot2)
library(lubridate)

##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##     date

library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:lubridate':
##
##     intersect, setdiff, union

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

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

library(tidyr)
library(forecast)

## Registered S3 method overwritten by 'xts':
##   method      from
##   as.zoo.xts zoo

## Registered S3 method overwritten by 'quantmod':
```

```
## method          from
## as.zoo.data.frame zoo

## Registered S3 methods overwritten by 'forecast':
## method          from
## fitted.fracdiff  fracdiff
## residuals.fracdiff fracdiff

library(xts)

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##   as.Date, as.Date.numeric

##
## Attaching package: 'xts'

## The following objects are masked from 'package:dplyr':
##
##   first, last

library(sqldf)
```

```
## Loading required package: gsubfn
## Loading required package: proto
## Loading required package: RSQLite

library(xlsx)
```

Including Plots

You can also embed plots, for example:

```
df2 <- read.csv('df2.csv')
str(df2)

## 'data.frame':   77321 obs. of  3 variables:
##  $ NumOS      : num  2.04e+13 2.04e+13 2.04e+13 2.04e+13 2.04e+13 ...
##  $ Regional: Factor w/ 5 levels "CENTRO","LESTE",...: 5 5 5 5 5 5 5 5 1 ...
##  $ DataSol  : Factor w/ 67203 levels "2019-01-01 00:10:00",...: 38845 44812 62681 42972 24465 13482 41...
```

Análise exploratória e manipulação dos dados

- verificar se há valores faltantes
- analisar se os dados estão proporcionalmente divididos entre as regiões

```
any(is.na(df2))

## [1] FALSE

table(df2$Regional)

##
## CENTRO  LESTE  NORTE  OESTE   SUL
## 18153   17498  11442  25865  4363
```

```
df2$NumOS_fator <- as.factor(df2$NumOS)
df2$NumOS <- NULL
df2$NumOS_num <- as.numeric(df2$NumOS_fator)
```

Transformar classe da variável DataSol de Factor para POSIXct

```
df2$DataSol <- as.POSIXct(df2$DataSol, format = "%Y-%m-%d %H:%M:%OS")
df2$NA_ <- is.na(df2$DataSol)
table(df2$NA_)
```

```
##
## FALSE
## 77321
```

```
df2 <- na.omit(df2)
df2$NA_ <- NULL
```

Excluindo observações com mesmo número de OS possível erro

O número da ordem de serviço (NumOS) é único para cada ordem de serviço, ou seja, se há observações com esse número duplicado, pode-se dizer que houve algum erro, e como encontrou-se somente alguns valores duplicados esses podem ser retirados sem alterar a análise.

```
any(duplicated(df2$NumOS_num))
```

```
## [1] TRUE
```

```
table(duplicated(df2$NumOS_num))
```

```
##
## FALSE  TRUE
## 77256    65
```

```
df2$duplicado <- duplicated(df2$NumOS_num)
duplicados <- sqldf("SELECT *
                    FROM df2
                    WHERE duplicado == TRUE")
View(duplicados)
df2 <- sqldf("SELECT *
            FROM df2
            WHERE duplicado == FALSE")
df2$duplicado <- NULL
```

Criando outras variáveis

Foram criadas variáveis para dividir o dataset por região.

```
table(df2$Regional)
```

```
##
## CENTRO  LESTE  NORTE  OESTE   SUL
## 18141   17479  11435  25845  4356
```

```
df2$SUL <- as.factor(df2$Regional == 'SUL')
df2$CENTRO <- as.factor(df2$Regional == 'CENTRO')
df2$LESTE <- as.factor(df2$Regional == 'LESTE')
```

```
df2$OESTE <- as.factor(df2$Regional == 'OESTE')
df2$NORTE <- as.factor(df2$Regional == 'NORTE')
```

Criando as variáveis hora, mês, dia da semana e dia do mês

```
df2$Hora <- hour(df2$DataSol)
df2$DiadaSemana <- wday(df2$DataSol)
df2$DiadoMes <- mday(df2$DataSol)
df2$Mes <- month(df2$DataSol)
df2$Ano <- year(df2$DataSol)
df2 <- unite(df2, Ano, Mes, DiadoMes, col = 'Datadia', sep = '-', remove = FALSE)
```

Separando por região

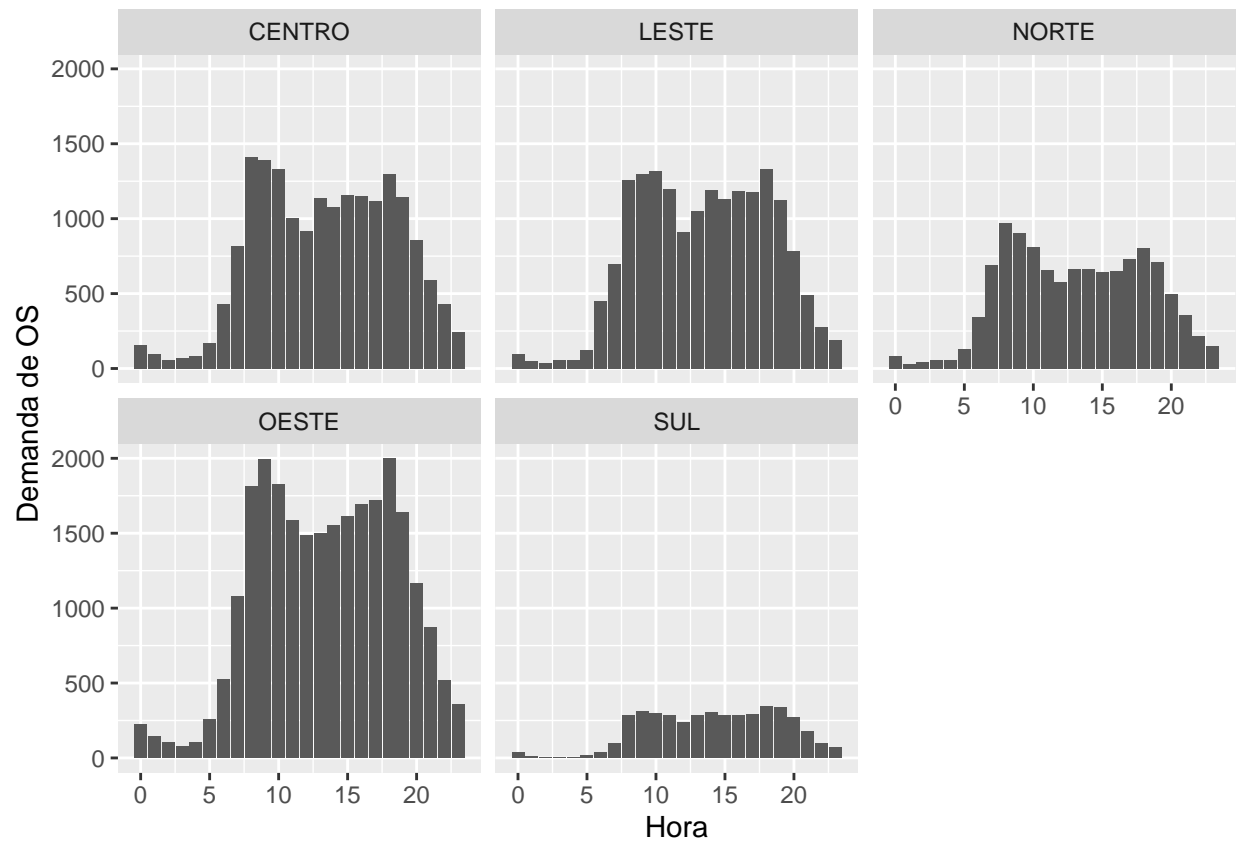
```
reg_sul <- dplyr::filter(df2, df2$SUL == TRUE)
reg_norte <- dplyr::filter(df2, df2$NORTE == TRUE)
reg_leste <- dplyr::filter(df2, df2$LESTE == TRUE)
reg_oeste <- dplyr::filter(df2, df2$OESTE == TRUE)
reg_centro <- dplyr::filter(df2, df2$CENTRO == TRUE)
```

Gráficos

A análise dos dados fica mais clara e intuitiva através dos gráficos abaixo.

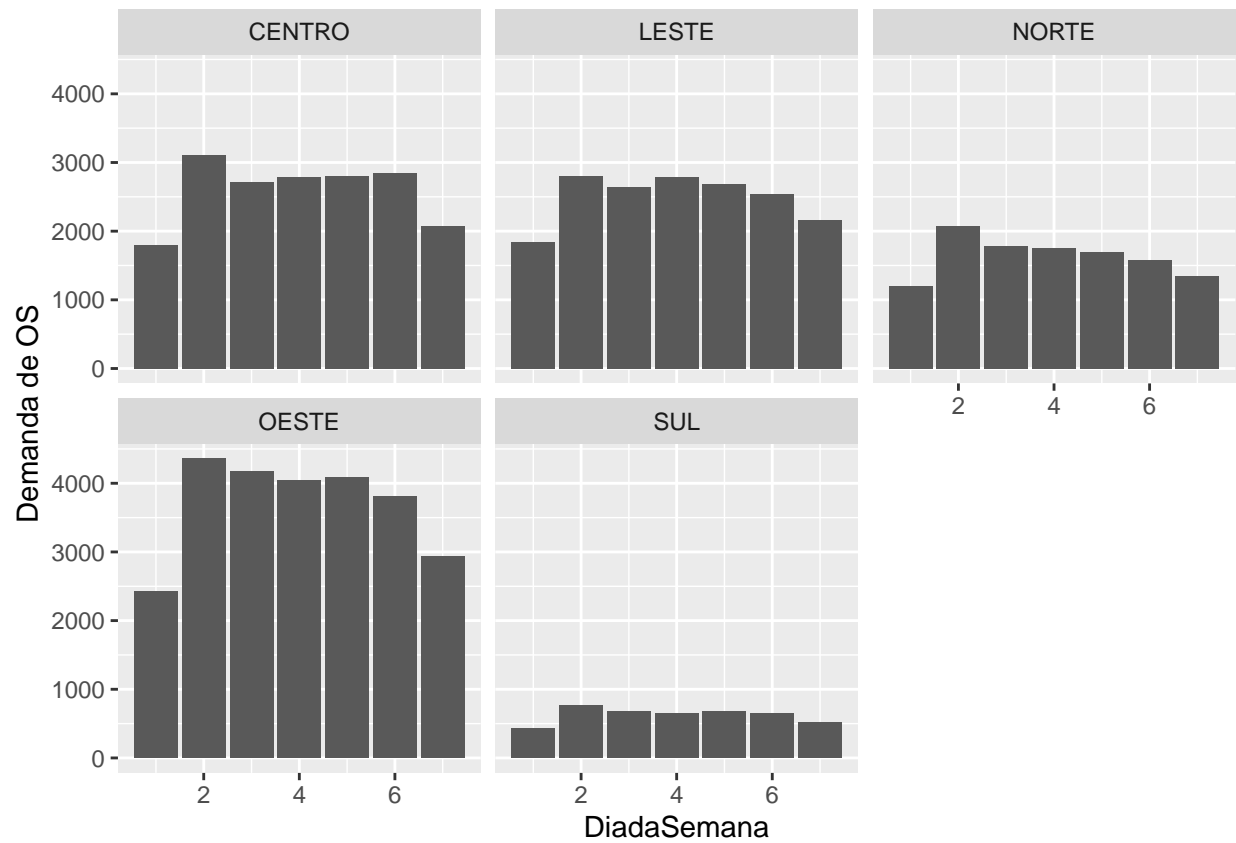
```
df2_freq_reg_hora <- df2 %>%
  group_by(Regional, Hora) %>%
  count()

ggplot(df2_freq_reg_hora, aes(x = Hora, y = n))+
  geom_bar(stat = 'identity') +
  ylab('Demanda de OS') +
  facet_wrap(~ Regional)
```



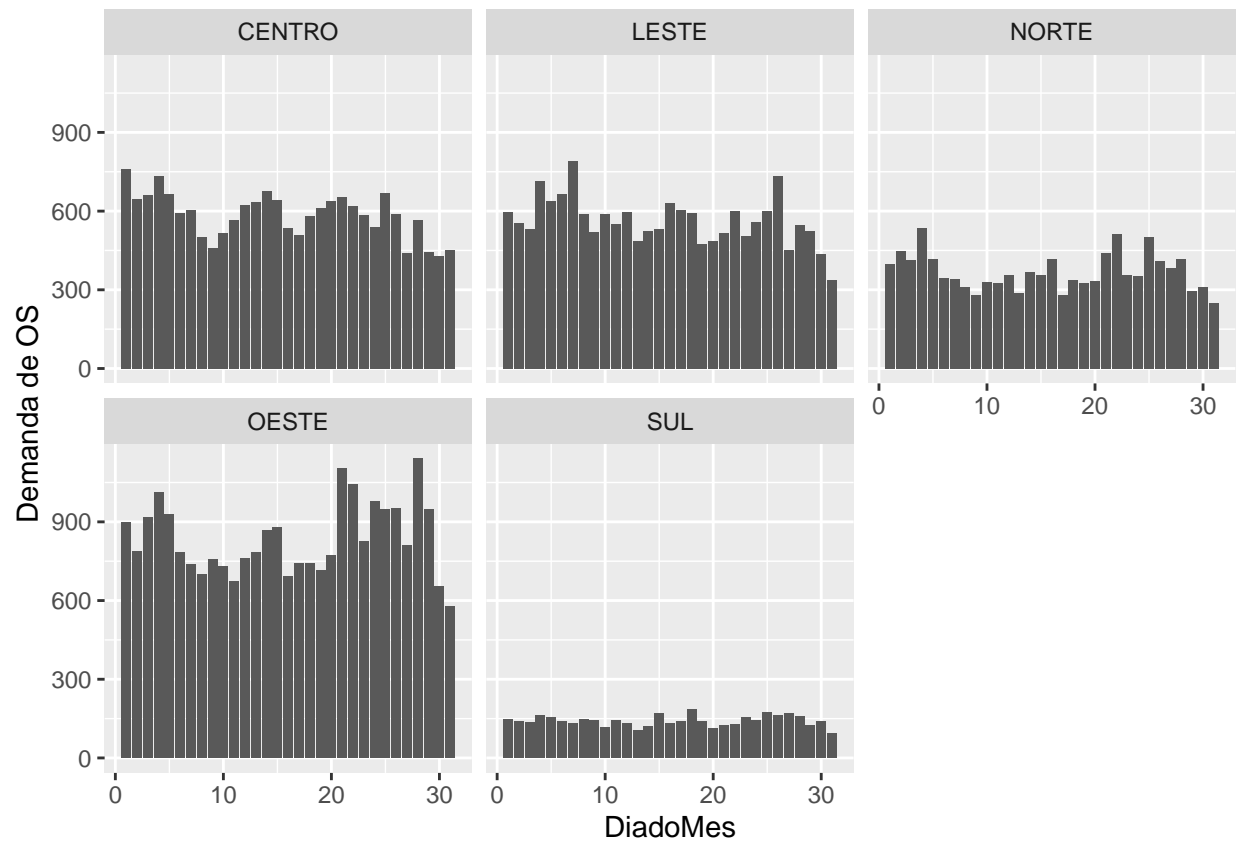
```
df2_freq_reg_diadasemana <- df2 %>%
  group_by(Regional, DiadaSemana) %>%
  count()

ggplot(df2_freq_reg_diadasemana, aes(x = DiadaSemana, y = n))+
  geom_bar(stat = 'identity') +
  ylab('Demanda de OS') +
  facet_wrap(~ Regional)
```



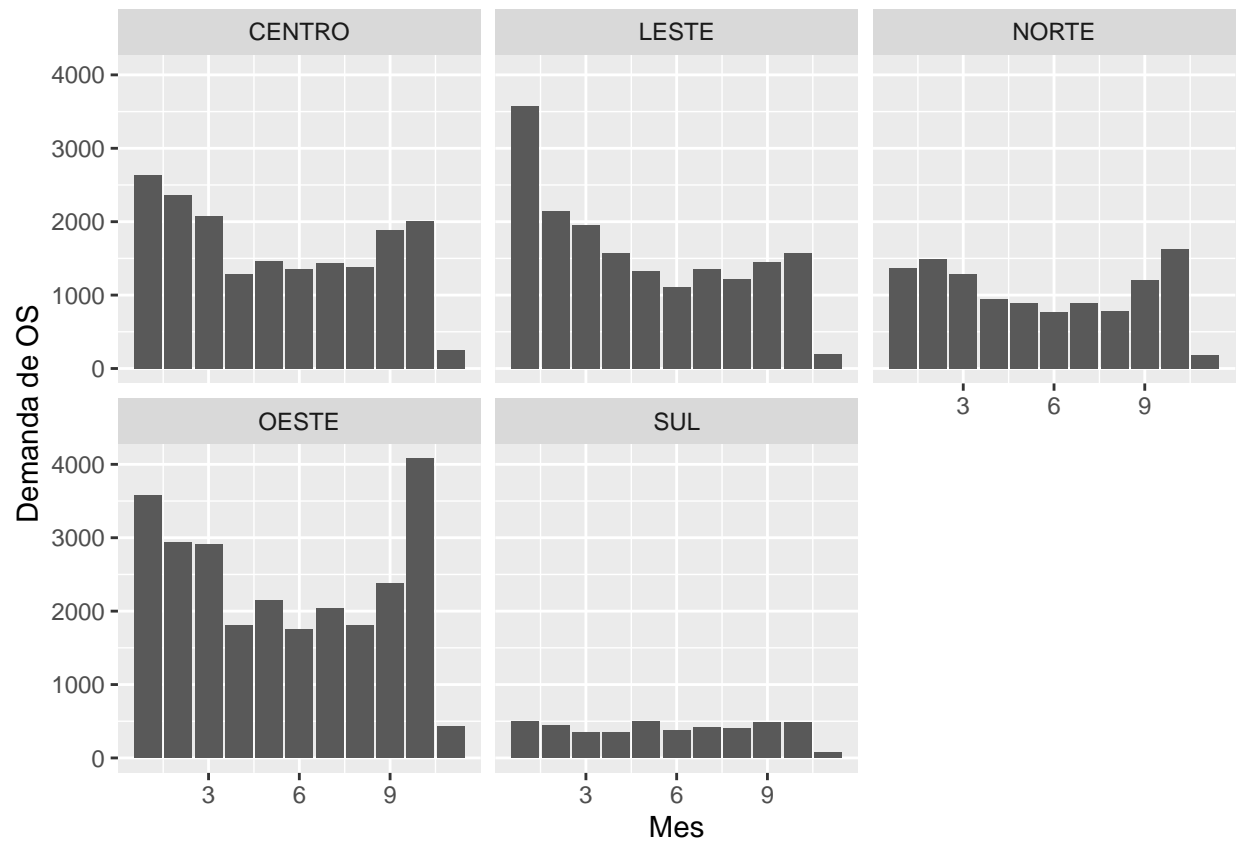
```
df2_freq_reg_diadomes <- df2 %>%
  group_by(Regional, DiadoMes) %>%
  count()

ggplot(df2_freq_reg_diadomes, aes(x = DiadoMes, y = n)) +
  geom_bar(stat = 'identity') +
  ylab('Demanda de OS') +
  facet_wrap(~ Regional)
```



```
df2_freq_reg_mes <- df2 %>%
  group_by(Regional, Mes) %>%
  count()

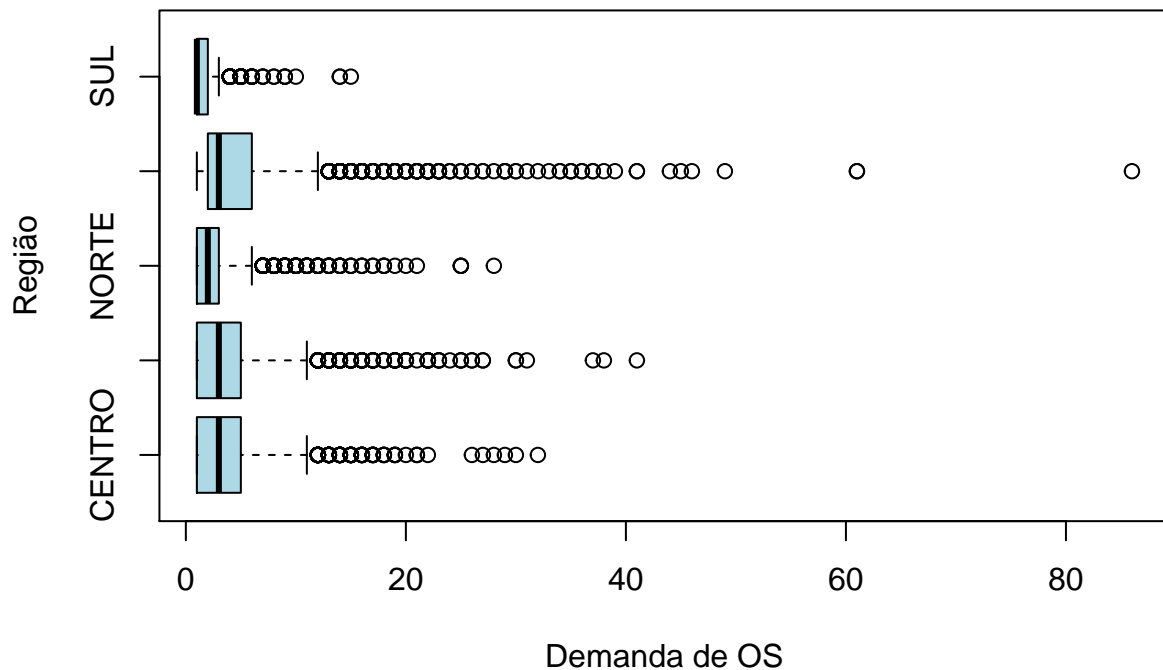
ggplot(df2_freq_reg_mes, aes(x = Mes, y = n)) +
  geom_bar(stat = 'identity') +
  ylab('Demanda de OS') +
  facet_wrap(~ Regional)
```



```
df2_freq_reg_dia <- df2 %>%
  group_by(Regional, Datadia) %>%
  count()

df2_freq_reg_datasol <- df2 %>%
  group_by(Regional, Mes, DiadoMes, Hora) %>%
  count()

boxplot(data = df2_freq_reg_datasol, n ~ Regional, horizontal = TRUE, xlab = 'Demanda de OS', ylab = 'R
```

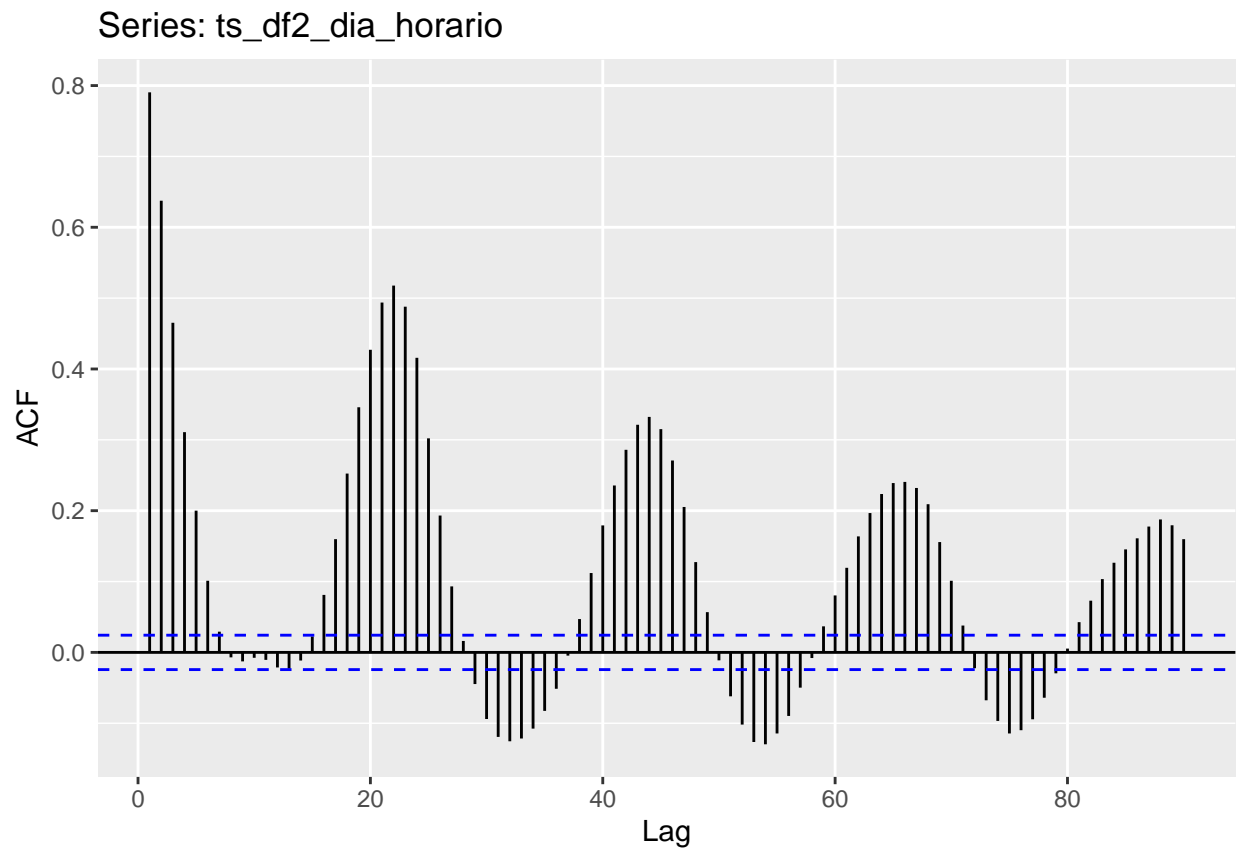
ACF e PACF

Ao se observar o gráfico ACF para série temporal de todas as regiões juntas observa-se que a série não é estacionária pois os valores de ACF decaem de forma devagar com o aumento dos lags. Pode-se afirmar também que há uma tendência de aumento na série temporal pois há uma maior autocorrelação para os primeiros lags. E também que há sazonalidade pois observa-se a autocorrelação aumentando durante os períodos constantes.

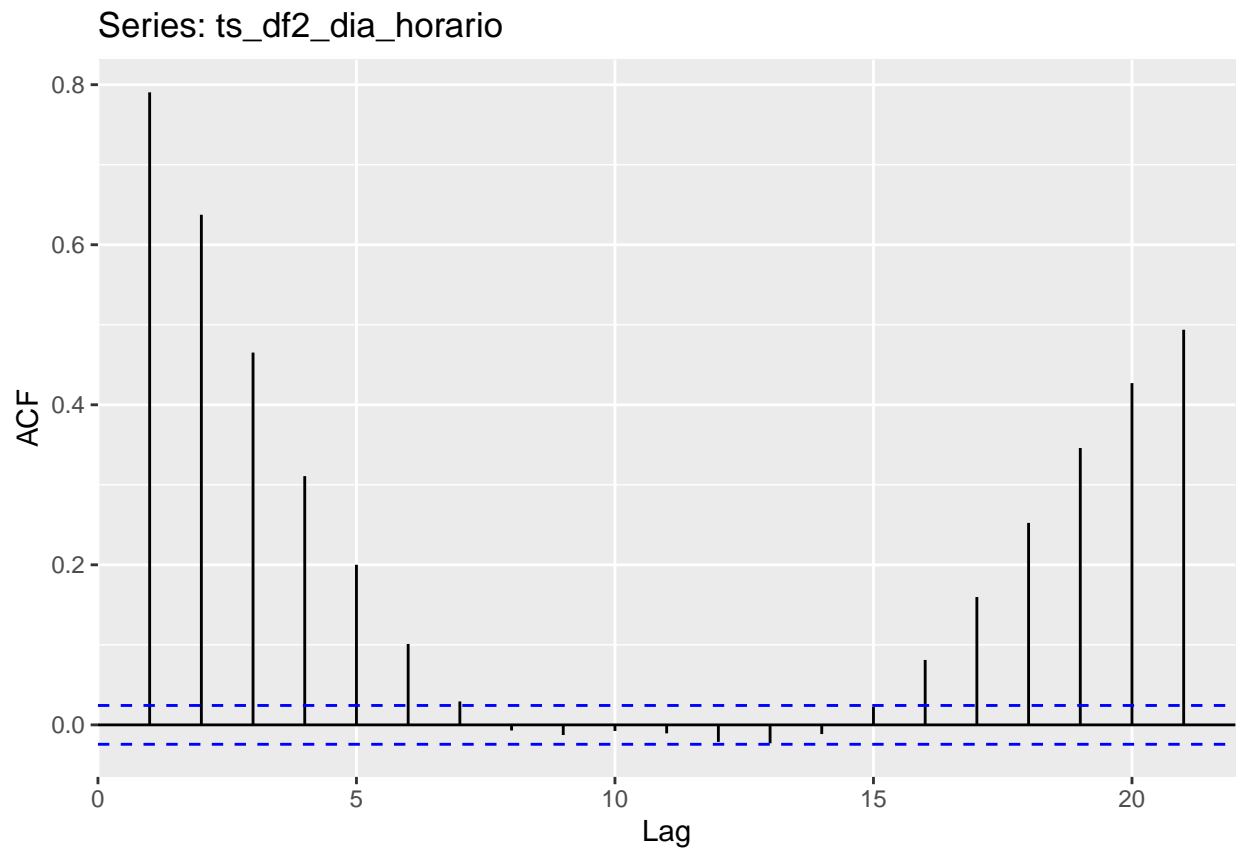
```
df2_freq_dia_horario <- df2 %>%
  group_by(Datadia, Hora) %>%
  count()

df2_freq_dia_horario$Horario <- stringr::str_c(as.character(df2_freq_dia_horario$Datadia), ' ', as.character(df2_freq_dia_horario$Hora))
df2_freq_dia_horario$Horario <- ymd_h(df2_freq_dia_horario$Horario)

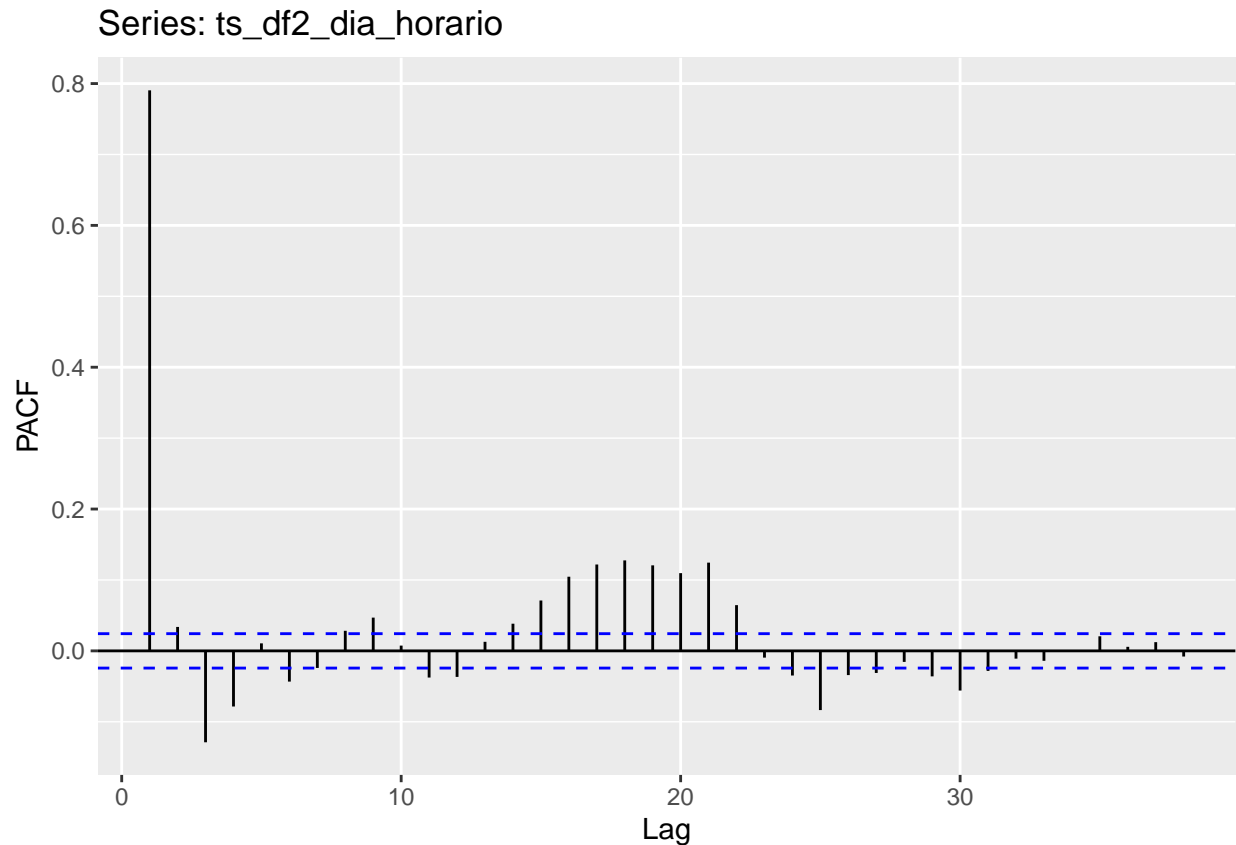
xts_df2_dia_horario <- xts(df2_freq_dia_horario$n, df2_freq_dia_horario$Horario)
ts_df2_dia_horario <- ts(xts_df2_dia_horario)
ggAcf(ts_df2_dia_horario, lag.max = 90, type = 'correlation')
```



```
ggAcf(ts_df2_dia_horario, lag.max = 21, type = 'correlation')
```



```
ggPacf(ts_df2_dia_horario)
```



Criando variáveis Horário e n

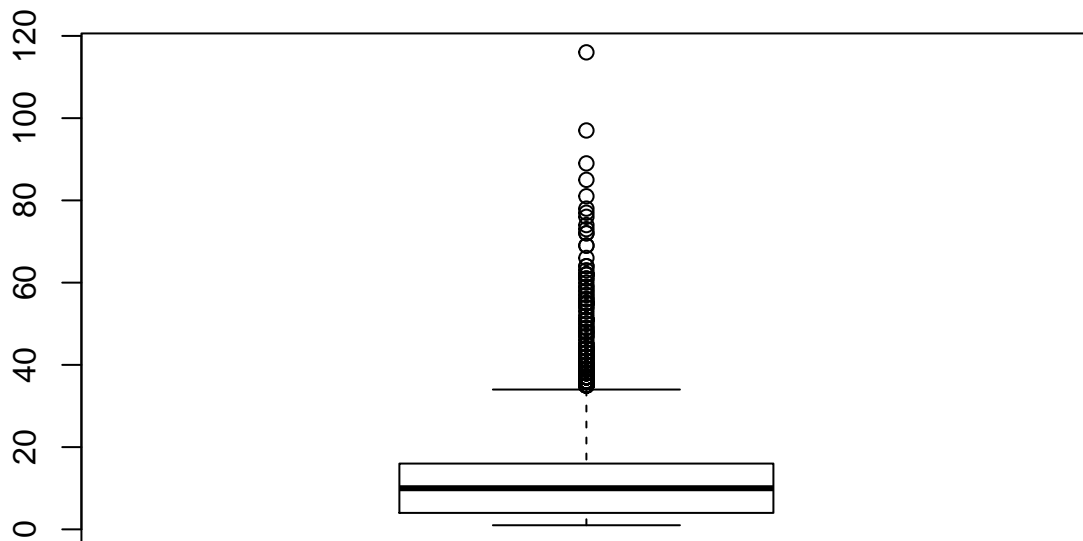
- através da variável 'Datasol' criou-se a variável 'Horário' que contém a data e a hora da solicitação
- e a variável 'n' representa o número de 'OS' recebidos para cada hora.

```
df2_freq_dia_horario <- df2 %>%
  group_by(Datadia, Hora) %>%
  count()
```

```
df2_freq_dia_horario$Horario <- stringr::str_c(as.character(df2_freq_dia_horario$Datadia), ' ', as.character(df2_freq_dia_horario$Hora))
df2_freq_dia_horario$Horario <- ymd_h(df2_freq_dia_horario$Horario)
```

Tratando os outliers

```
boxplot(df2_freq_dia_horario$n)
```



```
df2_freq_dia_horario <- sqldf("SELECT Horario, n
    FROM df2_freq_dia_horario
    WHERE n < 35")
xts_df2_dia_horario <- xts(df2_freq_dia_horario$n, df2_freq_dia_horario$Horario)
ts_df2_dia_horario <- ts(xts_df2_dia_horario)
```

Análise da região central

Alteração das variáveis para fator

```
reg_centro$Hora <- as.factor(reg_centro$Hora)
reg_centro$DiadaSemana <- as.factor(reg_centro$DiadaSemana)
reg_centro$DiadoMes <- as.factor(reg_centro$DiadoMes)
reg_centro$Mes <- as.factor(reg_centro$Mes)
reg_centro$Ano <- as.factor(reg_centro$Ano)
```

Análise exploratória

```
teste_centro <- sqldf("SELECT DataSol, NumOS_num, Hora, DiadaSemana, DiadoMes, Mes, Ano
    FROM reg_centro
    ORDER BY DataSol")

summary(teste_centro)
```

```
##      DataSol      NumOS_num      Hora      DiadaSemana
## Min. :2019-01-01 00:14:00 Min. : 2 8 : 1414 1:1796
## 1st Qu.:2019-02-25 11:24:00 1st Qu.:19023 9 : 1389 2:3108
## Median :2019-05-16 08:01:00 Median :38414 10 : 1332 3:2713
## Mean :2019-05-24 03:53:35 Mean :38246 18 : 1296 4:2791
## 3rd Qu.:2019-08-23 15:09:00 3rd Qu.:57511 15 : 1161 5:2809
## Max. :2019-11-05 16:03:00 Max. :77255 16 : 1151 6:2849
## (Other):10398 7:2075
##      DiadoMes      Mes      Ano
## 1 : 759 1 :2630 2019:18141
## 4 : 732 2 :2361
## 14 : 677 3 :2083
## 25 : 669 10 :2005
## 5 : 664 9 :1887
## 3 : 660 5 :1469
## (Other):13980 (Other):5706

table(teste_centro$Hora)

##
## 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
## 157 97 55 69 85 171 430 818 1414 1389 1332 1006 918 1136 1075 1161
## 16 17 18 19 20 21 22 23
## 1151 1119 1296 1146 855 591 428 242

table(teste_centro$DiadaSemana)

##
## 1 2 3 4 5 6 7
## 1796 3108 2713 2791 2809 2849 2075

table(teste_centro$DiadoMes)

##
## 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
## 759 645 660 732 664 591 603 503 461 516 568 623 635 677 642 536 508 581 613 640
## 21 22 23 24 25 26 27 28 29 30 31
## 652 620 585 538 669 589 442 565 444 430 450

table(teste_centro$Mes)

##
## 1 2 3 4 5 6 7 8 9 10 11
## 2630 2361 2083 1287 1469 1356 1432 1378 1887 2005 253

any(is.na(teste_centro))

## [1] FALSE
```

Criando variáveis Horário e n

```
teste_centro <- teste_centro %>%
  group_by(Ano, Mes, DiadoMes, Hora) %>%
  count()

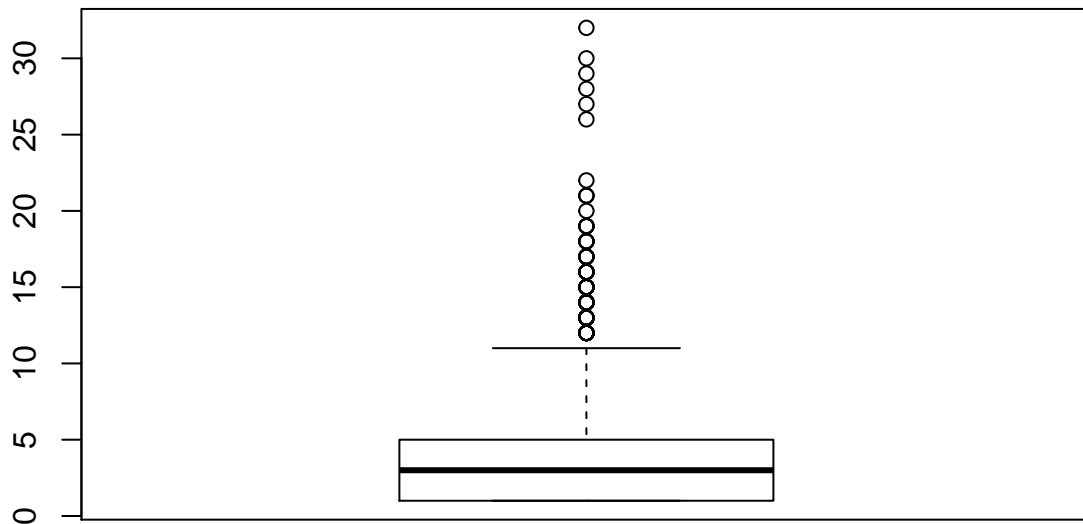
teste_centro <- unite(teste_centro, Ano, Mes, DiadoMes, col = 'Horario', sep = '-')

```

```
teste_centro <- unite(teste_centro, Horário, Hora, col = 'Horário', sep = ' ')
teste_centro$Horário <- ymd_h(teste_centro$Horário)
```

Tratando os outliers

```
boxplot(teste_centro$n)
```



```
table(teste_centro$n)
```

```
##
##      1      2      3      4      5      6      7      8      9     10     11     12     13     14     15     16
## 1325 1030  853  643  438  289  175  122   78   58   31   32   22   16   15   10
##      17     18     19     20     21     22     26     27     28     29     30     32
##       8      5      5      1      3      1      1      1      1      1      1      1
```

```
summary(teste_centro$n)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.000   1.000   3.000   3.512   5.000   32.000
```

```
centro_freq_hora <- sqldf("SELECT *
                           FROM teste_centro
                           WHERE n < 11")
str(centro_freq_hora)
```

```
## 'data.frame':   5011 obs. of  2 variables:
```

```
## $ Horário: POSIXct, format: "2018-12-31 22:00:00" "2019-01-01 05:00:00" ...
## $ n      : int  2 1 2 4 3 2 2 1 1 1 ...

xts_centro_freq_hora <- xts(centro_freq_hora$n, centro_freq_hora$Horario)
ts_centro_freq_hora <- ts(xts_centro_freq_hora)
```

Análise da região norte

Alteração das variáveis para fator

```
reg_norte$Hora <- as.factor(reg_norte$Hora)
reg_norte$DiadaSemana <- as.factor(reg_norte$DiadaSemana)
reg_norte$DiadoMes <- as.factor(reg_norte$DiadoMes)
reg_norte$Mes <- as.factor(reg_norte$Mes)
reg_norte$Ano <- as.factor(reg_norte$Ano)
```

Análise exploratória

```
teste_norte <- sqldf("SELECT DataSol, NumOS_num, Hora, DiadaSemana, DiadoMes, Mes, Ano
FROM reg_norte
ORDER BY DataSol")
```

```
summary(teste_norte)
```

```
##      DataSol      NumOS_num      Hora      DiadaSemana
## Min.   :2019-01-01 03:48:00 Min.   :  11  8      : 969  1:1205
## 1st Qu.:2019-03-01 04:33:00 1st Qu.:21028 9      : 904  2:2075
## Median :2019-05-23 16:29:00 Median :39843 10     : 811  3:1782
## Mean   :2019-05-31 18:30:07 Mean   :40109 18     : 804  4:1755
## 3rd Qu.:2019-09-02 19:29:30 3rd Qu.:59884 17     : 731  5:1695
## Max.   :2019-11-05 15:34:00 Max.   :77247 19     : 711  6:1574
##                                     (Other):6505  7:1349
##      DiadoMes      Mes      Ano
## 4      : 534  10      :1623  2019:11435
## 22     : 514  2       :1485
## 25     : 503  1       :1373
## 2      : 448  3       :1291
## 21     : 440  9       :1206
## 16     : 418  4       : 941
## (Other):8578  (Other):3516
```

```
table(teste_norte$Hora)
```

```
##
##  0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19
## 85 31 43 54 57 132 342 693 969 904 811 658 576 665 662 643 650 731 804 711
## 20 21 22 23
## 498 355 214 147
```

```
table(teste_norte$DiadaSemana)
```

```
##
##  1  2  3  4  5  6  7
## 1205 2075 1782 1755 1695 1574 1349
```



```
table(teste_norte$DiadoMes)
```

```
##  
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  
## 397 448 415 534 417 346 341 311 280 328 327 356 287 367 358 418 279 337 325 334  
##  21  22  23  24  25  26  27  28  29  30  31  
## 440 514 356 352 503 409 382 418 296 311 249
```

```
table(teste_norte$Mes)
```

```
##  
##   1   2   3   4   5   6   7   8   9  10  11  
## 1373 1485 1291 941 894 765 891 779 1206 1623 187
```

```
any(is.na(teste_norte))
```

```
## [1] FALSE
```

Criando variáveis Horário e n

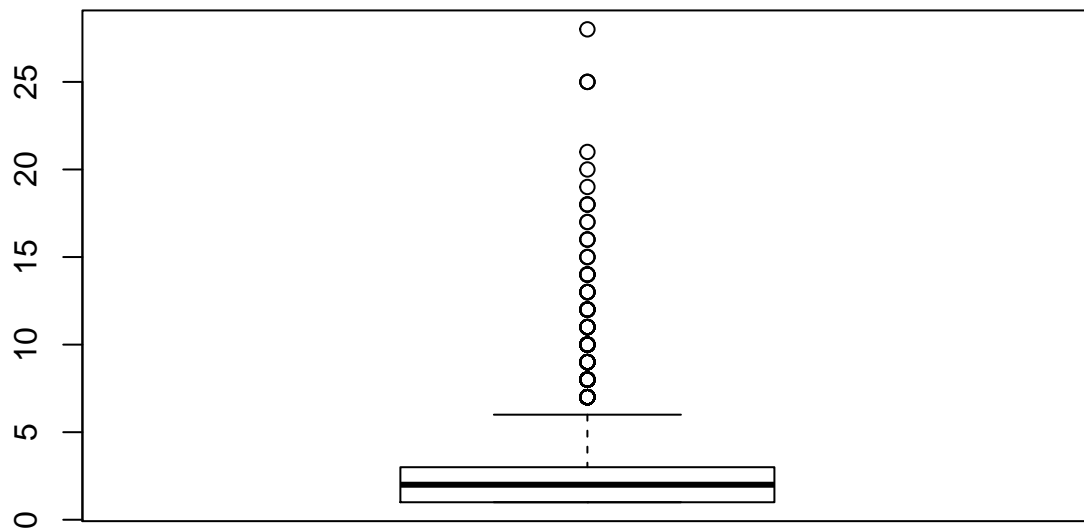
```
teste_norte <- teste_norte %>%  
  group_by(Ano, Mes, DiadoMes, Hora) %>%  
  count()
```

```
teste_norte <- unite(teste_norte, Ano, Mes, DiadoMes, col = 'Horario', sep = '-')
```

```
teste_norte <- unite(teste_norte, Horario, Hora, col = 'Horario', sep = ' ')  
teste_norte$Horario <- ymd_h(teste_norte$Horario)
```

Tratando os outliers

```
boxplot(teste_norte$n)
```



```
table(teste_norte$n)
```

```
##
##      1      2      3      4      5      6      7      8      9     10     11     12     13     14     15     16
## 1606 1148  695  375  241  119   65   37   28   21   15   11    7    6    3    3
##    17    18    19    20    21    25    28
##     2     3     1     1     1     3     1
```

```
summary(teste_norte$n)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.000  1.000   2.000   2.604  3.000   28.000
```

```
treino_norte_freq_hora <- sqldf("SELECT *
    FROM teste_norte
    WHERE n < 7")
str(teste_norte)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':  4392 obs. of  2 variables:
## $ Horário: POSIXct, format: "2019-01-01 03:00:00" "2019-01-01 07:00:00" ...
## $ n      : int  1 2 1 1 1 3 1 2 1 3 ...
```

```
xts_treino_norte_freq_hora <- xts(treino_norte_freq_hora$n, treino_norte_freq_hora$Horario)
ts_treino_norte_freq_hora <- ts(xts_treino_norte_freq_hora)
```

Análise da região sul

Alteração das variáveis para fator

```
reg_sul$Hora <- as.factor(reg_sul$Hora)
reg_sul$DiadaSemana <- as.factor(reg_sul$DiadaSemana)
reg_sul$DiadoMes <- as.factor(reg_sul$DiadoMes)
reg_sul$Mes <- as.factor(reg_sul$Mes)
reg_sul$Ano <- as.factor(reg_sul$Ano)
```

Análise exploratória

```
teste_sul <- sqldf("SELECT DataSol, NumOS_num, Hora, DiadaSemana, DiadoMes, Mes, Ano
FROM reg_sul
ORDER BY DataSol")
```

```
summary(teste_sul)
```

```
##      DataSol      NumOS_num      Hora      DiadaSemana
## Min.   :2019-01-01 14:42:00 Min.   :  60  18   : 347  1:429
## 1st Qu.:2019-03-14 19:19:30 1st Qu.:25196 19   : 336  2:759
## Median :2019-06-05 15:59:30 Median :43083 9    : 310  3:673
## Mean   :2019-06-06 11:36:08 Mean   :41380 14   : 307  4:651
## 3rd Qu.:2019-08-27 09:00:00 3rd Qu.:58067 10   : 295  5:680
## Max.   :2019-11-05 14:54:00 Max.   :77237 17   : 291  6:650
##                                     (Other):2470  7:514
##      DiadoMes      Mes      Ano
## 18      : 184      1      : 495  2019:4356
## 25      : 172      5      : 492
## 27      : 170      9      : 486
## 15      : 169     10      : 486
## 4       : 161      2      : 440
## 26      : 161      7      : 417
## (Other):3339  (Other):1540
```

```
table(teste_sul$Hora)
```

```
##
##  0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19
## 35  8  5  4  5 14 39 99 282 310 295 281 240 282 307 283 286 291 347 336
## 20 21 22 23
## 268 174 96 69
```

```
table(teste_sul$DiadaSemana)
```

```
##
##  1  2  3  4  5  6  7
## 429 759 673 651 680 650 514
```

```
table(teste_sul$DiadoMes)
```

```
##
##  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
## 146 140 135 161 156 138 131 148 143 117 142 133 106 120 169 130 138 184 139 111
## 21 22 23 24 25 26 27 28 29 30 31
```

```
## 124 128 156 142 172 161 170 158 125 140 93
table(teste_sul$Mes)

##
## 1 2 3 4 5 6 7 8 9 10 11
## 495 440 344 348 492 379 417 398 486 486 71
any(is.na(teste_sul))

## [1] FALSE
```

Criando variáveis Horário e n

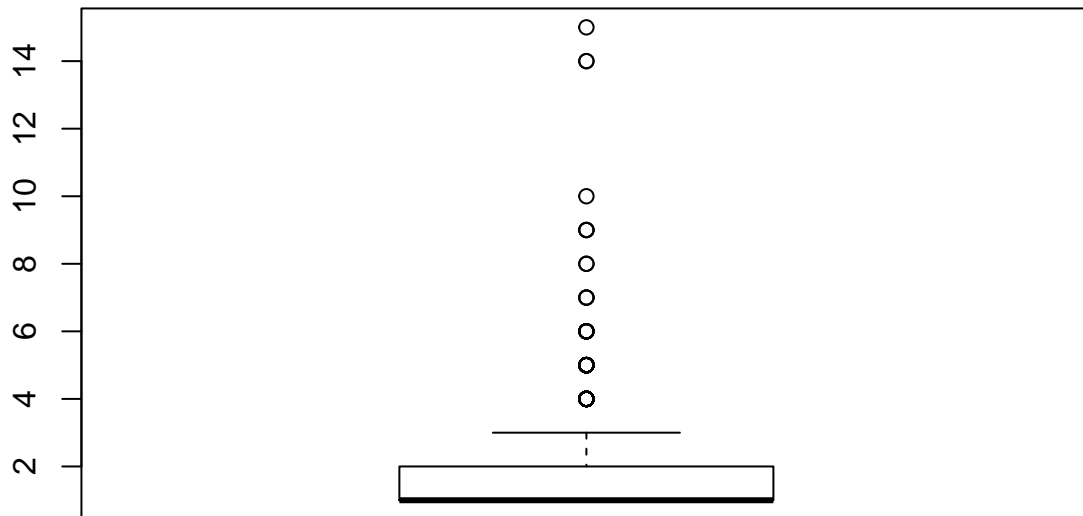
```
teste_sul <- teste_sul %>%
  group_by(Ano, Mes, DiadoMes, Hora) %>%
  count()

teste_sul <- unite(teste_sul, Ano, Mes, DiadoMes, col = 'Horario', sep = '-')

teste_sul <- unite(teste_sul, Horario, Hora, col = 'Horario', sep = ' ')
teste_sul$Horario <- ymd_h(teste_sul$Horario)
```

Tratando os outliers

```
boxplot(teste_sul$n)
```



```
table(teste_sul$n)
```

```
##
##      1      2      3      4      5      6      7      8      9     10     14     15
## 1680   693   242    56    35     8     3     2     3     1     2     1
```

```
summary(teste_sul)
```

```
##      Horario              n
##  Min.   :2019-01-01 14:00:00  Min.   : 1.000
## 1st Qu.:2019-03-14 13:15:00  1st Qu.: 1.000
##  Median :2019-06-02 18:30:00  Median : 1.000
##   Mean   :2019-06-04 11:55:51  Mean    : 1.598
## 3rd Qu.:2019-08-22 11:30:00  3rd Qu.: 2.000
##   Max.   :2019-11-05 14:00:00  Max.    :15.000
```

```
sul_freq_hora <- sqldf("SELECT *
                        FROM teste_sul
                        WHERE n < 4")
```

```
xts_sul_freq_hora <- xts(sul_freq_hora$n, sul_freq_hora$Horario)
ts_sul_freq_hora <- ts(xts_sul_freq_hora)
str(sul_freq_hora)
```

```
## 'data.frame':   2615 obs. of  2 variables:
## $ Horario: POSIXct, format: "2019-01-01 12:00:00" "2019-01-01 19:00:00" ...
## $ n      : int  1 1 1 1 1 1 1 1 1 2 ...
```

Análise da região leste

Alteração das variáveis para fator

```
reg_leste$Hora <- as.factor(reg_leste$Hora)
reg_leste$DiadaSemana <- as.factor(reg_leste$DiadaSemana)
reg_leste$DiadoMes <- as.factor(reg_leste$DiadoMes)
reg_leste$Mes <- as.factor(reg_leste$Mes)
reg_leste$Ano <- as.factor(reg_leste$Ano)
```

Análise exploratória

```
teste_leste <- sqldf("SELECT DataSol, NumOS_num, Hora, DiadaSemana, DiadoMes, Mes, Ano
FROM reg_leste
ORDER BY DataSol")
```

```
summary(teste_leste)
```

```
##      DataSol              NumOS_num      Hora      DiadaSemana
## Min.   :2019-01-01 00:10:00   Min.    :    1   18      :1329    1:1839
## 1st Qu.:2019-02-11 11:00:30   1st Qu.:14784   10      :1321    2:2810
## Median :2019-04-20 17:41:07   Median :33730    9      :1296    3:2645
## Mean   :2019-05-10 02:58:37   Mean    :34925    8      :1261    4:2791
## 3rd Qu.:2019-08-02 18:54:00   3rd Qu.:53719   11      :1198    5:2690
## Max.   :2019-11-05 15:52:00   Max.     :77254   14      :1194    6:2547
##                                     (Other):9880    7:2157
##      DiadoMes      Mes      Ano
## 7      : 791    1      :3570   2019:17479
## 26     : 733    2      :2146
## 4      : 715    3      :1958
## 6      : 664   10     :1574
## 5      : 637    4      :1571
## 16     : 630    9      :1450
## (Other):13309   (Other):5210
```

```
table(teste_leste$Hora)
```

```
##
##  0   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15
## 94  51  37  54  56 121 452 695 1261 1296 1321 1198  910 1051 1194 1132
## 16  17  18  19  20  21  22  23
## 1183 1175 1329 1128 783 488 279 191
```

```
table(teste_leste$DiadaSemana)
```

```
##
##  1   2   3   4   5   6   7
## 1839 2810 2645 2791 2690 2547 2157
```

```
table(teste_leste$DiadoMes)
```

```
##
##  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20
## 597 555 533 715 637 664 791 589 522 588 550 596 486 524 532 630 605 592 473 488
## 21  22  23  24  25  26  27  28  29  30  31
```

```
## 516 602 505 557 601 733 453 546 525 437 337
table(teste_leste$Mes)

##
##      1      2      3      4      5      6      7      8      9     10     11
## 3570 2146 1958 1571 1333 1107 1355 1215 1450 1574  200
any(is.na(teste_leste))

## [1] FALSE
```

Criando variáveis Horário e n

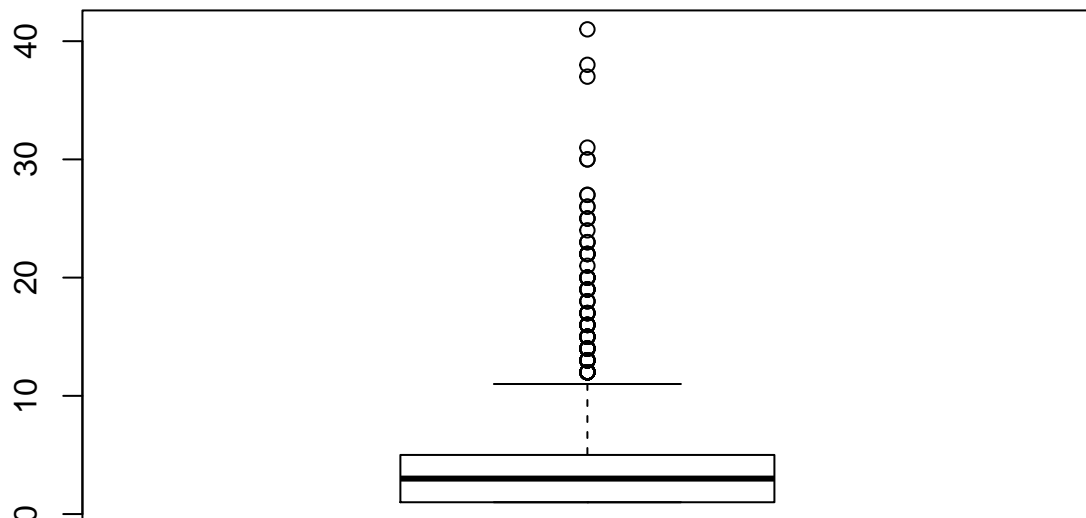
```
teste_leste <- teste_leste %>%
  group_by(Ano, Mes, DiadoMes, Hora) %>%
  count()

teste_leste <- unite(teste_leste, Ano, Mes, DiadoMes, col = 'Horario', sep = '-')

teste_leste <- unite(teste_leste, Horario, Hora, col = 'Horario', sep = ' ')
teste_leste$Horario <- ymd_h(teste_leste$Horario)
```

Tratando os outliers

```
boxplot(teste_leste$n)
```



```
table(teste_leste$n)
```

```
##
##      1      2      3      4      5      6      7      8      9     10     11     12     13     14     15     16
## 1247 1025  803  543  453  277  178  112   62   41   42   30   15   16    9    9
##   17   18   19   20   21   22   23   24   25   26   27   30   31   37   38   41
##    6    3    6    7    1    6    3    1    3    2    2    2    1    1    1    1
```

```
summary(teste_leste$n)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   1.000   3.000   3.561   5.000   41.000
```

```
leste_freq_hora <- sqldf("SELECT *
FROM teste_leste
WHERE n < 12")
```

```
xts_leste_freq_hora <- xts(leste_freq_hora$n, leste_freq_hora$Horario)
ts_leste_freq_hora <- ts(xts_leste_freq_hora)
```

Análise da região oeste

Alteração das variáveis para fator

```
reg_oeste$Hora <- as.factor(reg_oeste$Hora)
reg_oeste$DiadaSemana <- as.factor(reg_oeste$DiadaSemana)
reg_oeste$DiadoMes <- as.factor(reg_oeste$DiadoMes)
```



```
reg_oeste$Mes <- as.factor(reg_oeste$Mes)
reg_oeste$Ano <- as.factor(reg_oeste$Ano)
```

Análise exploratória

```
teste_oeste <- sqldf("SELECT DataSol, NumOS_num, Hora, DiadaSemana, DiadoMes, Mes, Ano
FROM reg_oeste
ORDER BY DataSol")
```

```
summary(teste_oeste)
```

```
##      DataSol      NumOS_num      Hora      DiadaSemana
## Min.   :2019-01-01 00:44:00 Min.    :  4  18    : 1996  1:2430
## 1st Qu.:2019-02-28 14:47:00 1st Qu.:20878 9     : 1992  2:4355
## Median :2019-05-27 15:49:00 Median :40679 10    : 1825  3:4178
## Mean   :2019-06-01 19:51:46 Mean   :40283 8     : 1814  4:4047
## 3rd Qu.:2019-09-05 09:13:00 3rd Qu.:60402 17    : 1722  5:4082
## Max.   :2019-11-05 16:03:00 Max.   :77256 16    : 1694  6:3812
##                                     (Other):14802  7:2941
##      DiadoMes      Mes      Ano
## 28      : 1140    10      :4074  2019:25845
## 21      : 1102     1      :3573
## 22      : 1044     2      :2941
## 4       : 1013     3      :2908
## 24      :  979     9      :2374
## 26      :  952     5      :2141
## (Other):19615  (Other):7834
```

```
table(teste_oeste$Hora)
```

```
##
##  0   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15
## 224 146 102  75 101 260 522 1081 1814 1992 1825 1587 1487 1501 1555 1610
##  16  17  18  19  20  21  22  23
## 1694 1722 1996 1638 1165  873  515  360
```

```
table(teste_oeste$DiadaSemana)
```

```
##
##  1   2   3   4   5   6   7
## 2430 4355 4178 4047 4082 3812 2941
```

```
table(teste_oeste$DiadoMes)
```

```
##
##  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16
## 899 786 915 1013 930 783 738 698 757 729 672 761 784 867 877 692
##  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31
## 741 741 713 772 1102 1044 825 979 947 952 810 1140 946 654 578
```

```
table(teste_oeste$Mes)
```

```
##
##  1   2   3   4   5   6   7   8   9  10  11
## 3573 2941 2908 1810 2141 1757 2033 1808 2374 4074 426
```

```
any(is.na(teste_oeste))
```

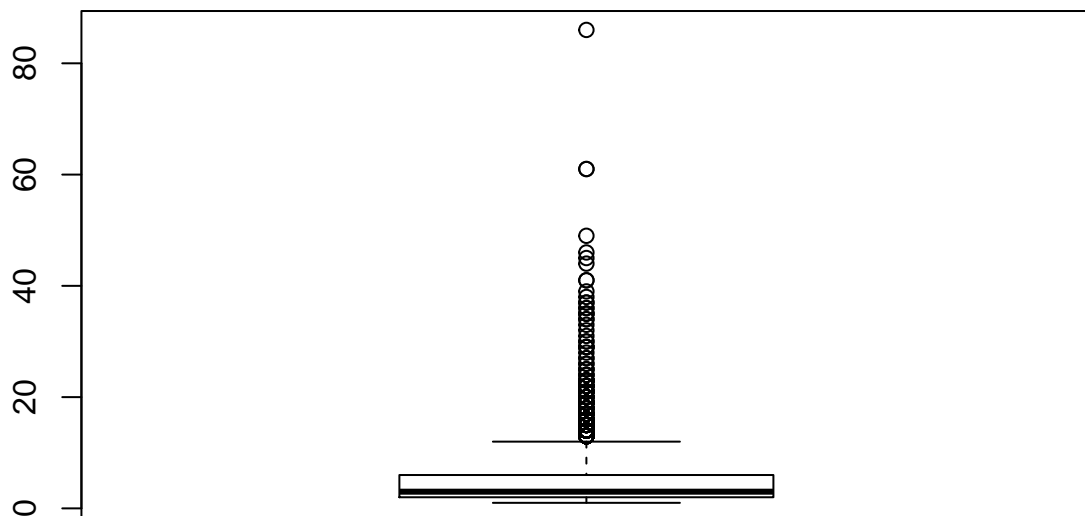
```
## [1] FALSE
```

Criando variáveis Horário e n

```
teste_oeste <- teste_oeste %>%  
  group_by(Ano, Mes, DiadoMes, Hora) %>%  
  count()  
  
teste_oeste <- unite(teste_oeste, Ano, Mes, DiadoMes, col = 'Horario', sep = '-')  
  
teste_oeste <- unite(teste_oeste, Horario, Hora, col = 'Horario', sep = ' ')  
teste_oeste$Horario <- ymd_h(teste_oeste$Horario)
```

Tratando os outliers

```
boxplot(teste_oeste$n)
```



```
table(teste_oeste$n)
```

```
##  
##    1     2     3     4     5     6     7     8     9    10    11    12    13    14    15    16  
## 1144  897  821  645  503  425  330  216  160  124   86   52   47   39   23   21
```

```
## 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32
## 10 8 13 7 9 8 8 3 3 2 2 1 4 2 1 1
## 33 34 35 36 37 38 39 41 44 45 46 49 61 86
## 1 2 3 2 2 1 1 2 1 1 1 1 2 1
```

```
summary(teste_oeste$n)
```

```
## Min. 1st Qu. Median Mean 3rd Qu. Max.
## 1.000 2.000 3.000 4.586 6.000 86.000
```

```
oeste_freq_hora <- sqldf("SELECT *
FROM teste_oeste
WHERE n < 13")
```

```
xts_oeste_freq_hora <- xts(oeste_freq_hora$n, oeste_freq_hora$Horario)
ts_oeste_freq_hora <- ts(xts_oeste_freq_hora)
```

Análise dos modelos para previsão de demanda de ordem de serviço

Naive Forecast

- utilizou-se o modelo de previsão Naive como base para as demais previsões.

```
df2_freq_horario_0 <- read.xlsx(file = 'df2_freq_horario_0.xlsx', sheetName = 'df2_freq_horario_0', head = FALSE)
df2_freq_horario_0$contagem <- c(1:7413)
treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 7390, select = c(Horario, n))
ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 7389, select = c(Horario, n))

naive_df2_horario <- naive(c(ts_treino_df2_freq_dia_horario), h = 24)
df_naive_df2_horario <- as.data.frame(naive_df2_horario)
accuracy(f = df_naive_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)
```

```
## ME RMSE MAE MPE MAPE
## Test set 8.5 13.82932 8.5 100 100
```

Modelo de Suavização Exponencial com Tendência Linear - Holt

SES-Smoothing exponential simple SES é um modelo preditivo que trabalha com as médias ponderadas dos valores do passado

Holt linear tren method Holt aumenta o poder do SES permitindo a previsão de dados com tendência.

```
previsao_df2_holt_horario = holt(treino_df2_freq_dia_horario$n, h = 24)
df_prev_df2_holt_horario <- as.data.frame(previsao_df2_holt_horario)

accuracy(f = df_prev_df2_holt_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)
```

```
## ME RMSE MAE MPE MAPE
## Test set 5.685089 12.301 7.91735 -Inf Inf
```

SARIMA

```
fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario <- as.data.frame(sarima_prev_df2_horario)
```

```
accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)
```

```
##                ME      RMSE      MAE  MPE MAPE
## Test set -0.641598 10.73355 8.851115 -Inf  Inf
```

A partir da análise do erro, principalmente pelo RMSE, observa-se que o modelo SARIMA apresentou a melhor previsão.

O modelo SARIMA foi o que obteve a MELHOR PREVISÃO

Walk Forward Validation para Sarima

O mês de outubro foi utilizado para a realização de walfoward validation dia a dia, ou seja, para cada dia de outubro utilizou-se o modelo SARIMA para a previsão das 24 seguintes e em seguida foi analisado o erro.

```
treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6553, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6552 & contagem < 6577, select =
```

```
ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario
```

```
fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_1.10 <- as.data.frame(sarima_prev_df2_horario)
```

```
erro_sarima_1.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)
```

```
accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)
```

```
##                ME      RMSE      MAE  MPE MAPE
## Test set -0.766598 4.467057 3.68358 -Inf  Inf
```

```
treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6577, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6576 & contagem < 6601, select =
```

```
ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario
```

```
fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_2.10 <- as.data.frame(sarima_prev_df2_horario)
```

```
erro_sarima_2.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)
```

```
accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)
```

```
##                ME      RMSE      MAE  MPE MAPE
## Test set 0.1917353 5.608523 4.655448 -Inf  Inf
```

```
treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6601, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6600 & contagem < 6625, select =
```

```

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_3.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_3.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME    RMSE      MAE  MPE MAPE
## Test set  3.441735 7.5463 5.639984 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6625, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6624 & contagem < 6649, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_4.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_4.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME    RMSE      MAE  MPE MAPE
## Test set -0.6832647 5.529378 4.754206 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6649, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6648 & contagem < 6673, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_5.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_5.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME    RMSE      MAE  MPE MAPE
## Test set -1.641598 4.6572 4.090298 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6673, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6672 & contagem < 6697, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))

```

```

sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_6.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_6.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME      RMSE      MAE  MPE MAPE
## Test set -3.308265 5.364982 4.595243 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6697, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6696 & contagem < 6721, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario$Horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_7.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_7.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME      RMSE      MAE  MPE MAPE
## Test set 0.8167353 6.729775 5.054245 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6721, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6720 & contagem < 6745, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario$Horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_8.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_8.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME      RMSE      MAE  MPE MAPE
## Test set -0.5582647 4.549406 3.920081 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6745, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6744 & contagem < 6769, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario$Horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_9.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_9.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

```

```

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME      RMSE      MAE  MPE MAPE
## Test set -0.1832647 5.85374 4.651183 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6769, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6768 & contagem < 6793, select =

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horar

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_10.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_10.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME      RMSE      MAE  MPE MAPE
## Test set -0.5999313 4.832656 3.87931 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6793, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6792 & contagem < 6817, select =

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horar

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_11.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_11.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME      RMSE      MAE  MPE MAPE
## Test set -1.266598 4.919395 4.269487 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6817, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6816 & contagem < 6841, select =

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horar

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_12.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_12.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME      RMSE      MAE  MPE MAPE
## Test set -3.558265 5.83672 5.139487 -Inf  Inf

```

```

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6841, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6840 & contagem < 6865, select =

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_13.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_13.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME        RMSE        MAE  MPE  MAPE
## Test set -1.349931 6.222407 5.374088 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6865, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6864 & contagem < 6889, select =

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_14.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_14.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME        RMSE        MAE  MPE  MAPE
## Test set 4.983402 11.05238 8.904121 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6889, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6888 & contagem < 6913, select =

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_15.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_15.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME        RMSE        MAE        MPE        MAPE
## Test set 0.4417353 5.776514 4.480231 -90.94049 121.3307

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6913, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6912 & contagem < 6937, select =

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)

```



```

ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_16.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_16.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME      RMSE      MAE  MPE MAPE
## Test set -1.433265 5.029408 4.301672 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6937, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6936 & contagem < 6961, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_17.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_17.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME      RMSE      MAE  MPE MAPE
## Test set -1.933265 4.526737 4.157769 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6961, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6960 & contagem < 6985, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_18.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_18.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME      RMSE      MAE  MPE MAPE
## Test set 2.358402 7.586795 6.148382 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 6985, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 6984 & contagem < 7009, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)

```

```

df_sarima_prev_df2_horario_19.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_19.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##           ME      RMSE      MAE      MPE      MAPE
## Test set 2.483402 8.201477 6.171861 -37.18337 84.47457

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 7009, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 7008 & contagem < 7033, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario$Horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_20.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_20.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##           ME      RMSE      MAE      MPE      MAPE
## Test set -3.183265 5.934184 5.232845 -Inf   Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 7033, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 7032 & contagem < 7057, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario$Horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_21.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_21.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##           ME      RMSE      MAE      MPE      MAPE
## Test set 0.2750687 11.89729 9.99463 -Inf   Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 7057, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 7056 & contagem < 7081, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario$Horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_22.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_22.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

```

```

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME      RMSE      MAE  MPE MAPE
## Test set -3.891598 10.50319 9.276819 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 7081, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 7080 & contagem < 7105, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_23.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_23.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME      RMSE      MAE  MPE MAPE
## Test set 3.983402 12.15582 9.693505 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 7105, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 7104 & contagem < 7129, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_24.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_24.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME      RMSE      MAE  MPE MAPE
## Test set 3.816735 10.47355 8.543075 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 7129, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 7128 & contagem < 7153, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_25.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_25.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##                ME      RMSE      MAE  MPE MAPE
## Test set 3.483402 8.60089 6.976998 -Inf  Inf

```

```

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 7153, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 7152 & contagem < 7177, select =

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_26.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_26.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##           ME      RMSE      MAE  MPE MAPE
## Test set 2.275069 7.063452 5.604913 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 7177, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 7176 & contagem < 7201, select =

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_27.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_27.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##           ME      RMSE      MAE  MPE MAPE
## Test set 1.608402 7.767693 5.984032 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 7201, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 7200 & contagem < 7225, select =

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_28.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_28.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##           ME      RMSE      MAE  MPE MAPE
## Test set -3.808265 9.413545 8.230199 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 7225, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 7224 & contagem < 7249, select =

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)

```

```

ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_29.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_29.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##           ME      RMSE      MAE  MPE MAPE
## Test set 0.4417353 9.797804 8.332627 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 7249, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 7248 & contagem < 7273, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_30.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_30.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##           ME      RMSE      MAE  MPE MAPE
## Test set 4.858402 9.46165 7.519703 -Inf  Inf

treino_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem < 7273, select = c(Horario, n))
comparar_df2_freq_dia_horario <- subset(df2_freq_horario_0, contagem > 7272 & contagem < 7297, select = c(Horario, n))

ts_treino_df2_freq_dia_horario <- ts(treino_df2_freq_dia_horario)
ts_comparar_df2_freq_dia_horario <- ts(xts(comparar_df2_freq_dia_horario$n, comparar_df2_freq_dia_horario))

fit_sarima_df2_horario <- Arima(treino_df2_freq_dia_horario$n, order = c(10,1,2), seasonal = c(0,0,1))
sarima_prev_df2_horario <- forecast(object = fit_sarima_df2_horario, h = 24, level = 0.95)
df_sarima_prev_df2_horario_31.10 <- as.data.frame(sarima_prev_df2_horario)

erro_sarima_31.10 <- (df_sarima_prev_df2_horario$`Point Forecast` - comparar_df2_freq_dia_horario$n)

accuracy(f = df_sarima_prev_df2_horario$`Point Forecast`, x = comparar_df2_freq_dia_horario$n)

##           ME      RMSE      MAE  MPE MAPE
## Test set 6.275069 10.47615 8.098242 -Inf  Inf

df_sarima_prev_df2_horario_out <- full_join(df_sarima_prev_df2_horario_1.10, df_sarima_prev_df2_horario_29.10, by = c("Point Forecast", "Lo 95", "Hi 95"))

df_sarima_prev_df2_horario_out <- full_join(df_sarima_prev_df2_horario_out, df_sarima_prev_df2_horario_30.10, by = c("Point Forecast", "Lo 95", "Hi 95"))

df_sarima_prev_df2_horario_out <- full_join(df_sarima_prev_df2_horario_out, df_sarima_prev_df2_horario_31.10, by = c("Point Forecast", "Lo 95", "Hi 95"))

```

[illegible]


```

df_sarima_prev_df2_horario_out <- full_join(df_sarima_prev_df2_horario_out, df_sarima_prev_df2_horario_out)

## Joining, by = c("Point Forecast", "Lo 95", "Hi 95")
df_sarima_prev_df2_horario_out <- full_join(df_sarima_prev_df2_horario_out, df_sarima_prev_df2_horario_out)

## Joining, by = c("Point Forecast", "Lo 95", "Hi 95")
df_sarima_prev_df2_horario_out <- full_join(df_sarima_prev_df2_horario_out, df_sarima_prev_df2_horario_out)

## Joining, by = c("Point Forecast", "Lo 95", "Hi 95")
df_sarima_prev_df2_horario_out <- full_join(df_sarima_prev_df2_horario_out, df_sarima_prev_df2_horario_out)

## Joining, by = c("Point Forecast", "Lo 95", "Hi 95")
df_sarima_prev_df2_horario_out <- full_join(df_sarima_prev_df2_horario_out, df_sarima_prev_df2_horario_out)

## Joining, by = c("Point Forecast", "Lo 95", "Hi 95")
df_sarima_prev_df2_horario_out <- full_join(df_sarima_prev_df2_horario_out, df_sarima_prev_df2_horario_out)

## Joining, by = c("Point Forecast", "Lo 95", "Hi 95")
df_sarima_prev_df2_horario_out <- full_join(df_sarima_prev_df2_horario_out, df_sarima_prev_df2_horario_out)

## Joining, by = c("Point Forecast", "Lo 95", "Hi 95")
df_sarima_prev_df2_horario_out <- full_join(df_sarima_prev_df2_horario_out, df_sarima_prev_df2_horario_out)

## Joining, by = c("Point Forecast", "Lo 95", "Hi 95")
df_sarima_prev_df2_horario_out <- full_join(df_sarima_prev_df2_horario_out, df_sarima_prev_df2_horario_out)

## Joining, by = c("Point Forecast", "Lo 95", "Hi 95")
df_sarima_prev_df2_horario_out <- full_join(df_sarima_prev_df2_horario_out, df_sarima_prev_df2_horario_out)

df_sarima_prev_df2_horario_out$`Lo 95` <- NULL
df_sarima_prev_df2_horario_out$`Hi 95` <- NULL
df_sarima_prev_df2_horario_out$`Point Forecast`[df_sarima_prev_df2_horario_out$`Point Forecast` < 0] <- 0

```

Análise dos erros

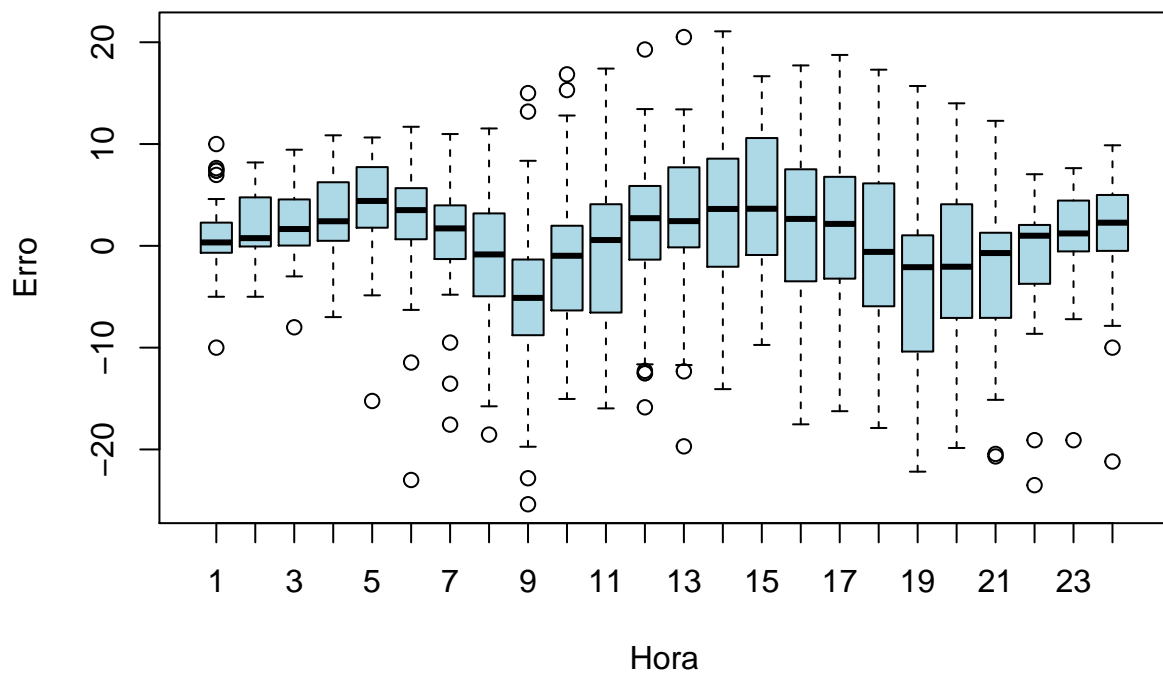
```

df_sarima_prev_df2_horario_out$Hora <- c(1:24)
df_sarima_prev_df2_horario_out$prev_trunc <- round(df_sarima_prev_df2_horario_out$`Point Forecast`, digits = 2)

actuals_out <- df2_freq_horario_0[df2_freq_horario_0$contagem > 6552 & df2_freq_horario_0$contagem < 6552]
df_sarima_prev_df2_horario_out <- cbind(df_sarima_prev_df2_horario_out, actuals_out)
df_sarima_prev_df2_horario_out$erro <- df_sarima_prev_df2_horario_out$`Point Forecast` - df_sarima_prev_df2_horario_out$actuals_out

boxplot(data = df_sarima_prev_df2_horario_out, erro ~ Hora, xlab = 'Hora', ylab = 'Erro', col = 'lightblue')

```



Como observado no boxplot do erro por hora observa-se que o modelo ainda não foi capaz de prever a sazonalidade diária apresentada pelo número de OS, sendo assim o modelo atual apesar de apresentar um erro pequeno ainda pode ser melhorado futuramente.