

Valores da produção de motocicletas no País

Rayza Camila dos Santos Silva

2023-01-14

Introdução

O banco de dados que será apresentado refere-se aos valores da produção de motocicletas no País entre janeiro de 1993 à setembro de 2022, com suas variáveis data e valor(R\$). Através desse banco de dados irei realizar uma análise de série temporal e apresentar o melhor modelo para previsão para os próximos passos (meses/anos).

Pacotes necessários

```
library(TSA)

##
## Attaching package: 'TSA'
##
## The following objects are masked from 'package:stats':
##
##   acf, arima
##
## The following object is masked from 'package:utils':
##
##   tar

library(forecast)

## Registered S3 method overwritten by 'quantmod':
##   method      from
##   as.zoo.data.frame zoo

## Registered S3 methods overwritten by 'forecast':
##   method      from
##   fitted.Arima TSA
##   plot.Arima   TSA

library(fpp3)

## -- Attaching packages ----- fpp3 0.4.0 --
## v tidyverse    3.1.8      v tsibble      1.1.3
## v dplyr         1.0.10     v tsibbledata  0.4.1
## v tidyr         1.2.1      v feasts       0.3.0
## v lubridate     1.9.0      v fable        0.3.2
## v ggplot2       3.4.0

## -- Conflicts ----- fpp3_conflicts --
## x lubridate::date() masks base::date()
## x dplyr::filter()   masks stats::filter()
```

```
## x tsibble::intersect() masks base::intersect()
## x tsibble::interval() masks lubridate::interval()
## x dplyr::lag() masks stats::lag()
## x tsibble::setdiff() masks base::setdiff()
## x tsibble::union() masks base::union()
```

```
library(stringr)
library(readr)
```

```
##
## Attaching package: 'readr'

## The following object is masked from 'package:TSA':
##
## spec
```

Entrando com o banco de dados

```
dataset<-read_delim( "bcddata.sgs.1377.csv",
                    delim = ";", escape_double = FALSE, col_types = cols(data = col_date(format = "%d/%"))
head(dataset)
```

```
## # A tibble: 6 x 2
##   data      valor
##   <date>    <dbl>
## 1 1993-01-01 4555
## 2 1993-02-01 5066
## 3 1993-03-01 6815
## 4 1993-04-01 5744
## 5 1993-05-01 6358
## 6 1993-06-01 7091
```

Tratamento do banco de dados

```
dados <- dataset %>%
  mutate( ano_mes = str_sub( data, 1, 7 ) ) %>%
  group_by(ano_mes) %>% summarise( producao = sum(valor) )
dados
```

```
## # A tibble: 357 x 2
##   ano_mes producao
##   <chr>    <dbl>
## 1 1993-01    4555
## 2 1993-02    5066
## 3 1993-03    6815
## 4 1993-04    5744
## 5 1993-05    6358
## 6 1993-06    7091
## 7 1993-07    7518
## 8 1993-08    7970
## 9 1993-09   10242
## 10 1993-10   9049
## # ... with 347 more rows
```

transformando parao formato tsibble

```
serie <- dados %>%  
  mutate(data = yearmonth(ano_mes))%>%  
  as_tsibble(index= data )  
serie
```

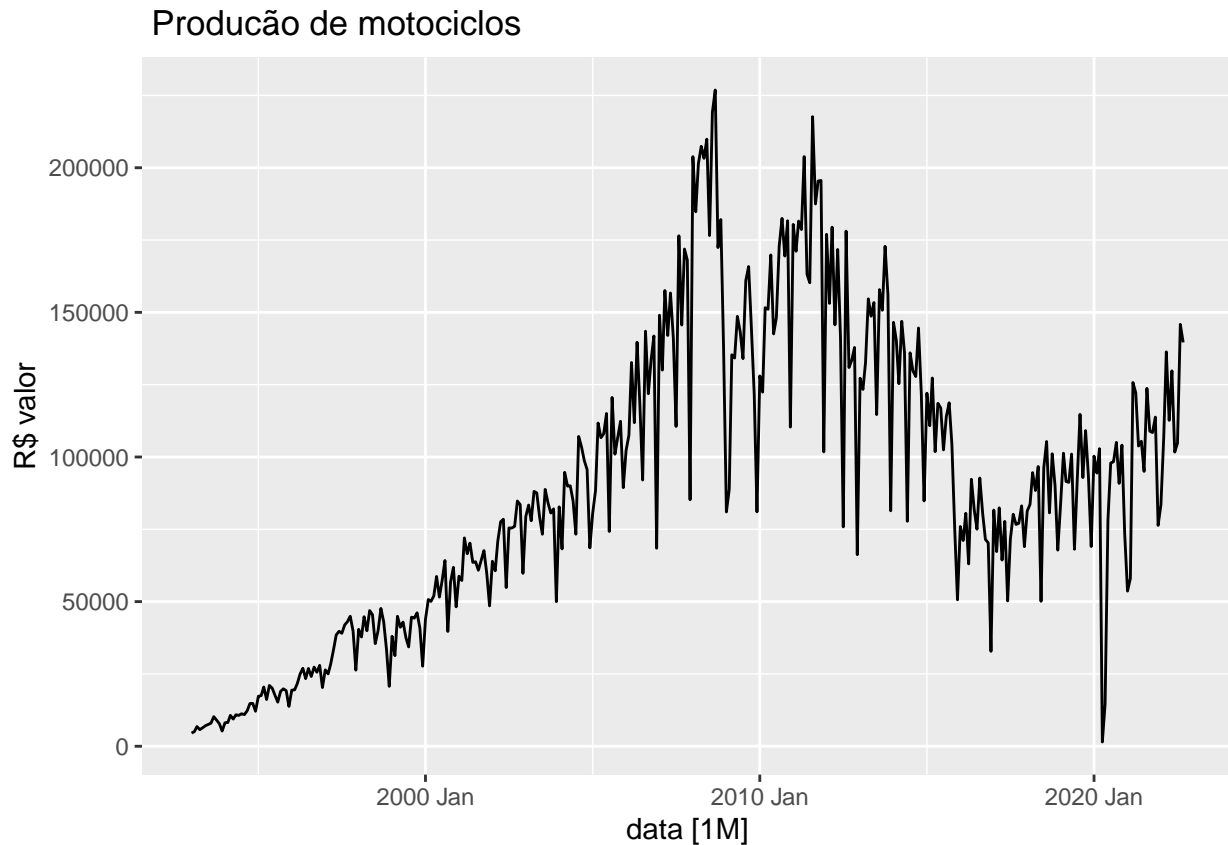
```
## # A tsibble: 357 x 3 [1M]  
##   ano_mes producao      data  
##   <chr>      <dbl>    <mth>  
## 1 1993-01      4555 1993 Jan  
## 2 1993-02      5066 1993 Feb  
## 3 1993-03      6815 1993 Mar  
## 4 1993-04      5744 1993 Apr  
## 5 1993-05      6358 1993 May  
## 6 1993-06      7091 1993 Jun  
## 7 1993-07      7518 1993 Jul  
## 8 1993-08      7970 1993 Aug  
## 9 1993-09     10242 1993 Sep  
## 10 1993-10     9049 1993 Oct  
## # ... with 347 more rows
```

```
tail(serie)
```

```
## # A tsibble: 6 x 3 [1M]  
##   ano_mes producao      data  
##   <chr>      <dbl>    <mth>  
## 1 2022-04    112678 2022 Apr  
## 2 2022-05    129781 2022 May  
## 3 2022-06    101695 2022 Jun  
## 4 2022-07    104776 2022 Jul  
## 5 2022-08    145852 2022 Aug  
## 6 2022-09    139622 2022 Sep
```

Análisisando o comportamento da série

```
plot_serie = serie %>%  
  autoplot(producao) +  
  labs(title=" Produção de motocicletas ",  
        y=" R$ valor ")  
plot_serie
```



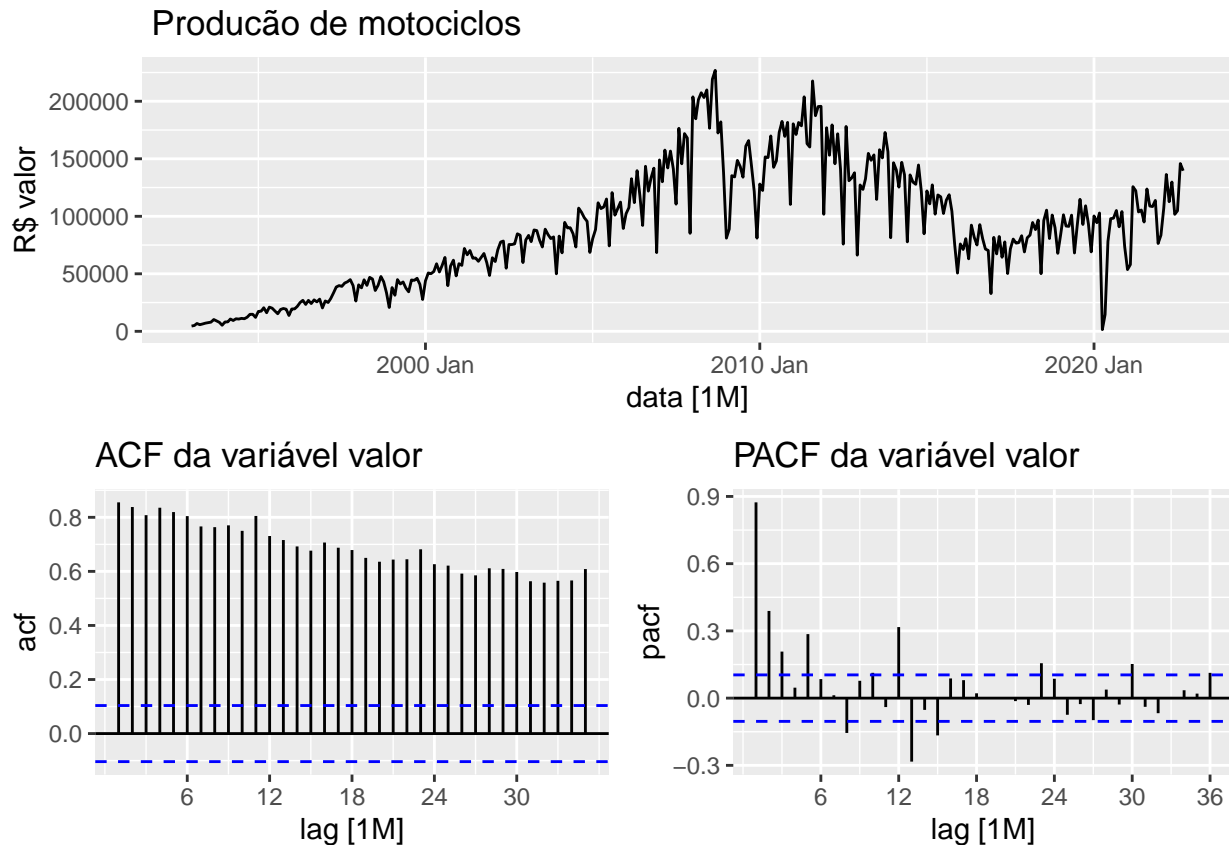
O gráfico apresenta não estacionariedade, pois, pode-se observar que o gráfico possui tendência e não tem sazonalidade.

Gráfico ACF e PACF da série

```
acf_producao = serie %>%
  ACF(lag_max = 36) %>%
  autoplot() + labs(title="ACF da variável valor")

pacf_producao = serie %>%
  PACF(lag_max = 36) %>%
  autoplot() + labs(title="PACF da variável valor")

gridExtra::grid.arrange(plot_serie, acf_producao, pacf_producao, layout_matrix=rbind(c(1,1), c(2,3)))
```



As informações pertinentes aos gráficos acima são da autocorrelação dos resíduos com seus números de períodos anteriores (lag), apontando sua acf e sua correlação na pacf (parcial). Identificamos na acf um declínio rápido a zero e significativo para conclusão da presença de tendência nos dados em vários valores positivos, levando a escolha do uso do modelo MA(11) e na pacf AR(3).

Verificando-se a série é estacionária

```
serie %>%
  features(producao, unitroot_kpss)
```

```
## # A tibble: 1 x 2
##   kpss_stat kpss_pvalue
##   <dbl>     <dbl>
## 1      2.95      0.01
```

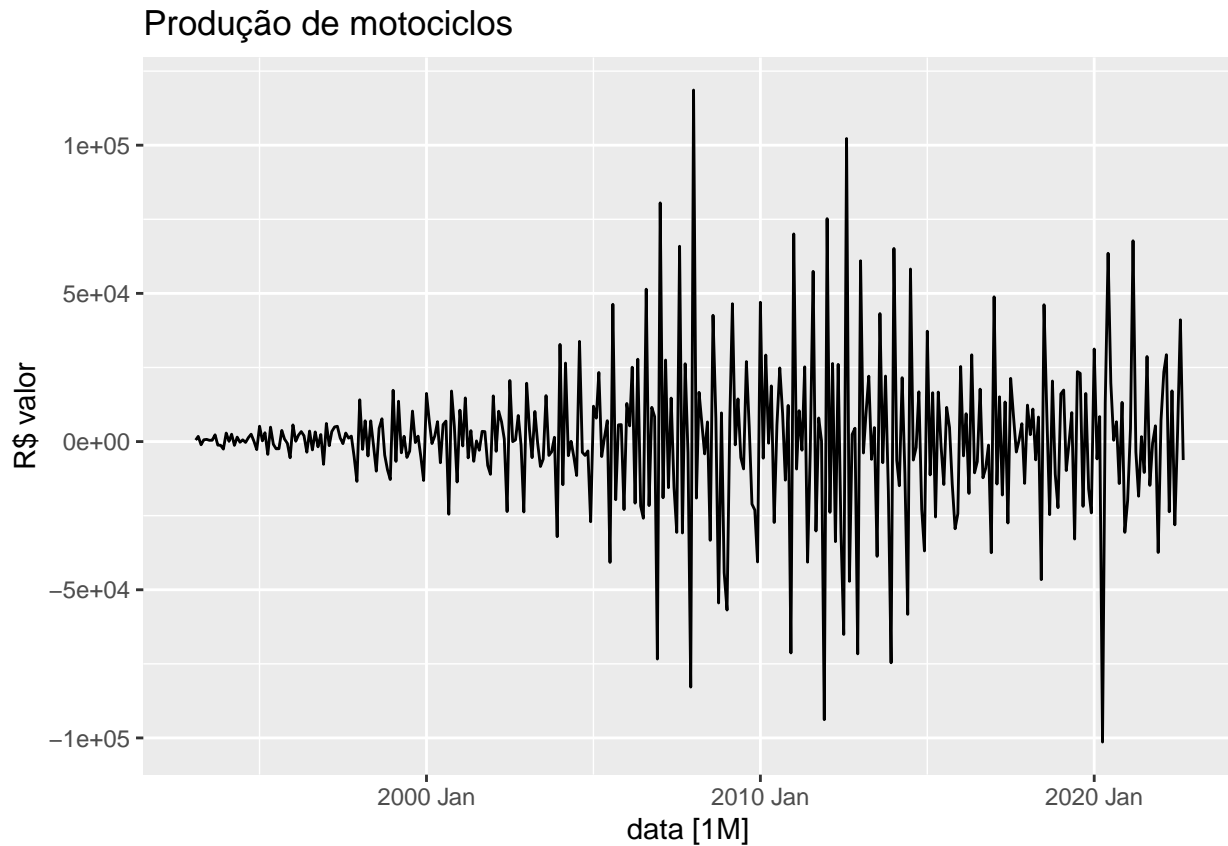
Como $Kpss_value = 0,01 < pvalue = 0,05$, então, rejeitar-se H_0 e concluir que não é estacionária.

Aplicando o operador de diferença para tornar a série estacionária

Gráfico estacionário

```
plot_seried = serie %>%
  autoplot( difference(producao) ) +
  labs(title="Produção de motocicletas ",
       y="R$ valor")
plot_seried
```

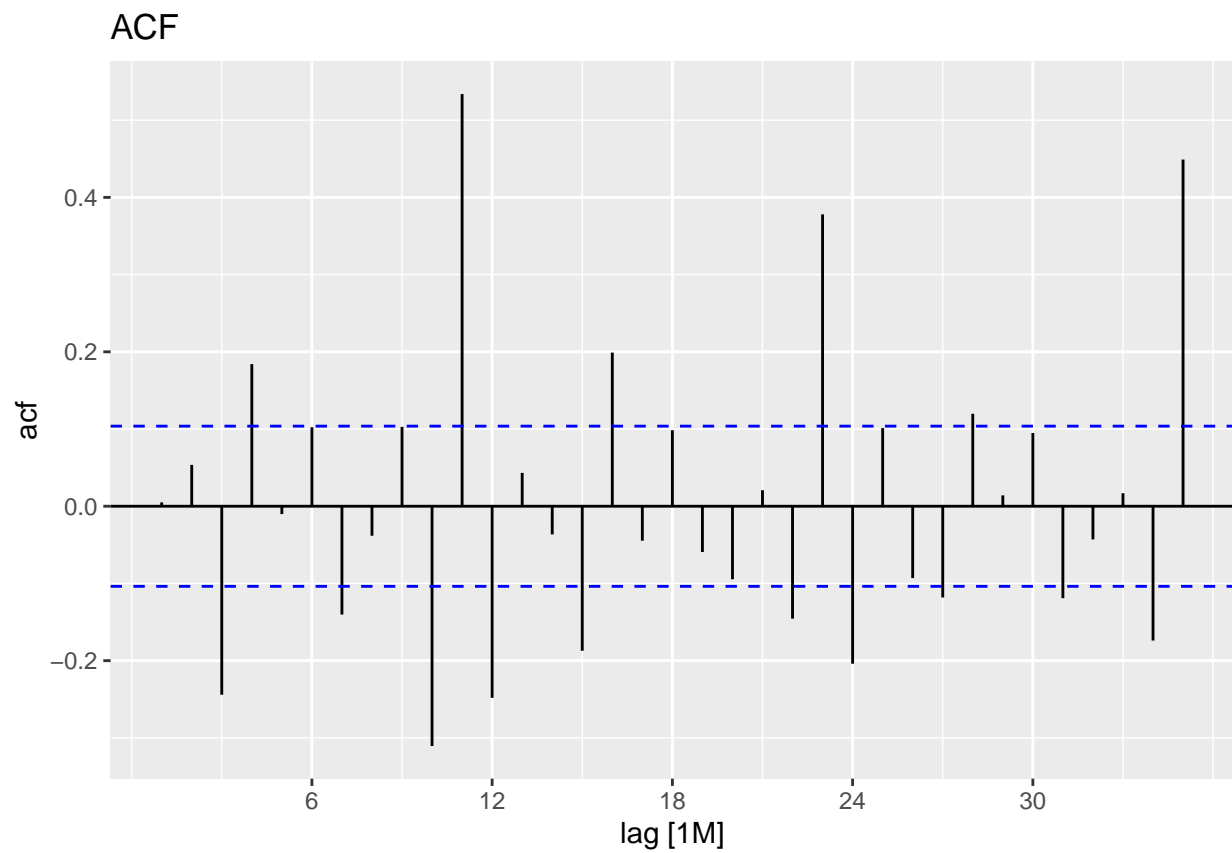
```
## Warning: Removed 1 row containing missing values (`geom_line()`).
```



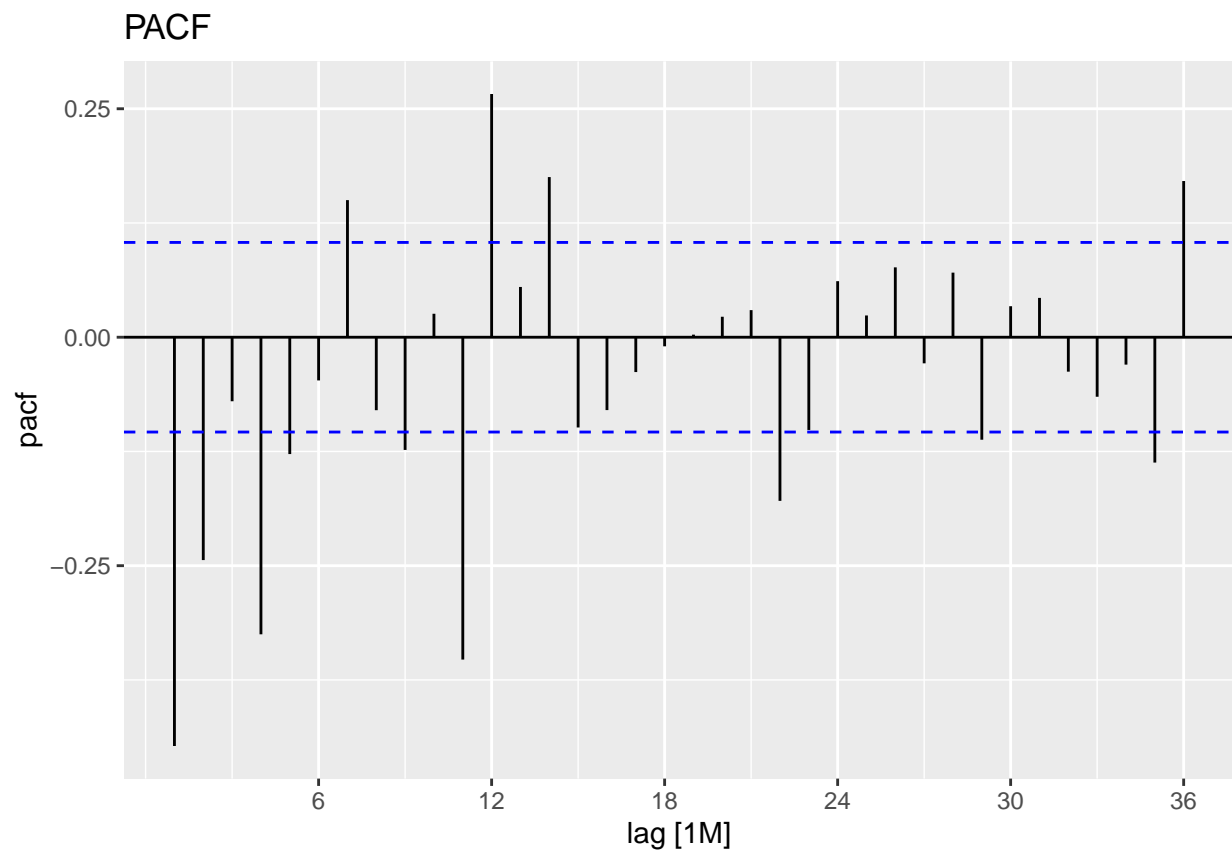
O gráfico está estacionário, isto é, ao longo do tempo os valores da produção de motocicletas estão aleatoriamente próximos de uma média constante.

Gráfico ACF e PACF da serie

```
acf_D = serie %>%  
  ACF(difference(producao), lag_max = 36) %>%  
  autoplot() + labs(title="ACF")  
acf_D
```

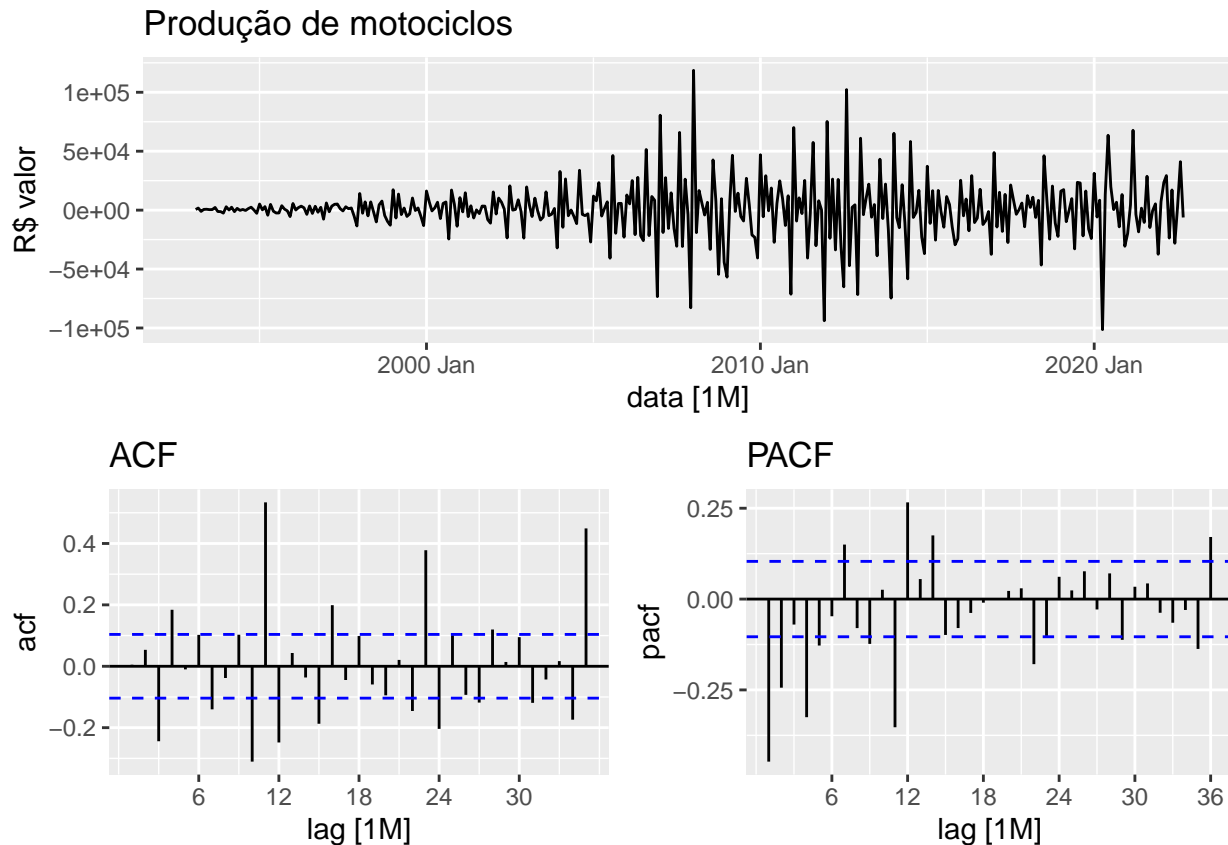


```
pacf_D = serie %>%
  PACF(difference(producao), lag_max = 36) %>%
  autoplot() + labs(title="PACF")
pacf_D
```



```
gridExtra::grid.arrange(plot_seriesD, acf_D, pacf_D, layout_matrix=rbind(c(1,1),c(2,3) ))
```

```
## Warning: Removed 1 row containing missing values (`geom_line()`).
```

Verificando-se a serie está estacionária

```
serie%>%
  features(difference(producao), unitroot_kpss)
```

```
## # A tibble: 1 x 2
##   kpss_stat kpss_pvalue
##   <dbl>      <dbl>
## 1    0.0456        0.1
```

Com $Kpss_value = 0,1 > pvalor = 0,05$, não rejeitar-se H_0 e conclui que é estacionária.

Definindo a base de dados em treinamento e teste.

```
train <- serie %>%
  filter_index("1993 Jan" ~ "2021 dec")
tail(train)
```

```
## # A tsibble: 6 x 3 [1M]
##   ano_mes producao    data
##   <chr>    <dbl>    <mth>
## 1 2021-07   95025 2021 Jul
## 2 2021-08  123722 2021 Aug
## 3 2021-09  108948 2021 Sep
## 4 2021-10  108456 2021 Oct
## 5 2021-11  113776 2021 Nov
```

```
## 6 2021-12      76359 2021 Dec
```

```
head(train)
```

```
## # A tsibble: 6 x 3 [1M]
##   ano_mes producao      data
##   <chr>      <dbl>    <mth>
## 1 1993-01      4555 1993 Jan
## 2 1993-02      5066 1993 Feb
## 3 1993-03      6815 1993 Mar
## 4 1993-04      5744 1993 Apr
## 5 1993-05      6358 1993 May
## 6 1993-06      7091 1993 Jun
```

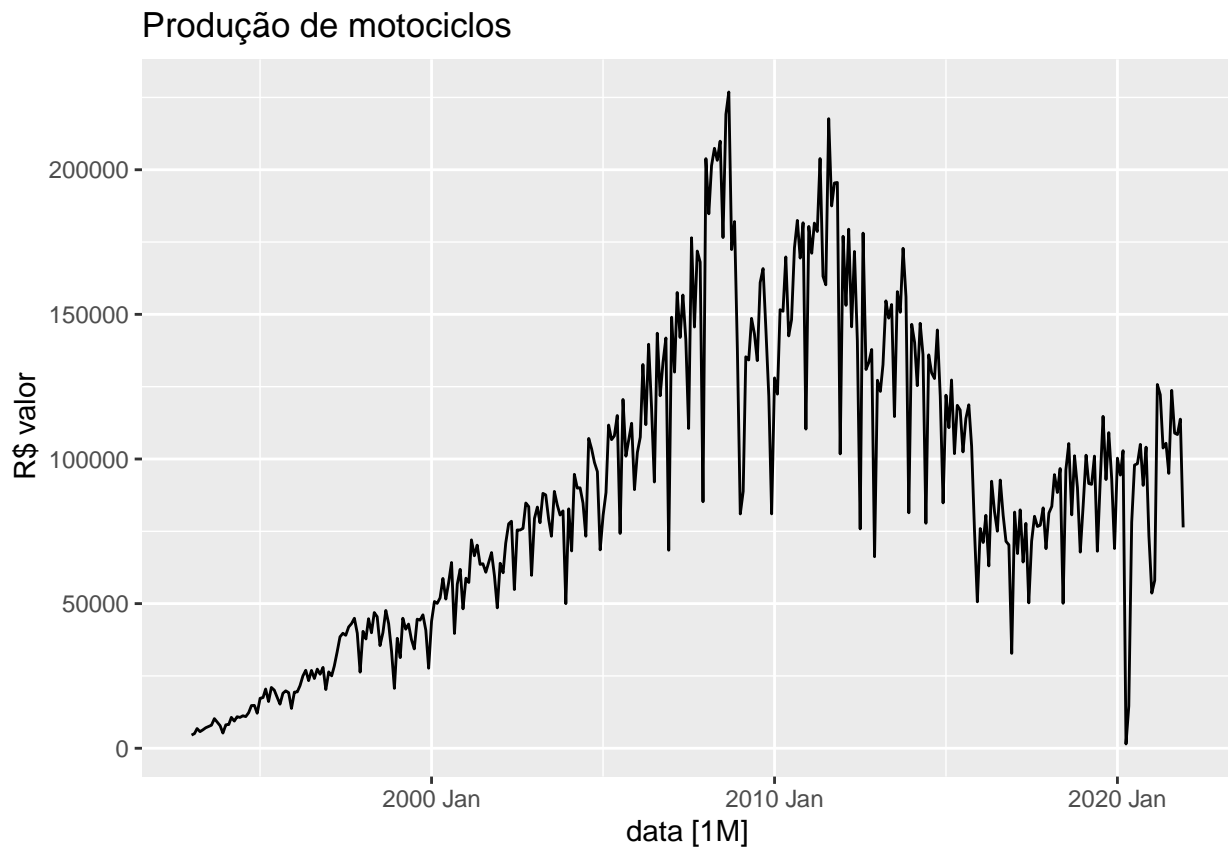
```
test <- serie %>%
  filter_index("2022" ~ . );test
```

```
## Warning: `yearmonth()` may yield unexpected results.
## i Please use arg `format` to supply formats.
```

```
## # A tsibble: 9 x 3 [1M]
##   ano_mes producao      data
##   <chr>      <dbl>    <mth>
## 1 2022-01      83543 2022 Jan
## 2 2022-02     107046 2022 Feb
## 3 2022-03     136350 2022 Mar
## 4 2022-04     112678 2022 Apr
## 5 2022-05     129781 2022 May
## 6 2022-06     101695 2022 Jun
## 7 2022-07     104776 2022 Jul
## 8 2022-08     145852 2022 Aug
## 9 2022-09     139622 2022 Sep
```

Gráfico do Comportamento da serie

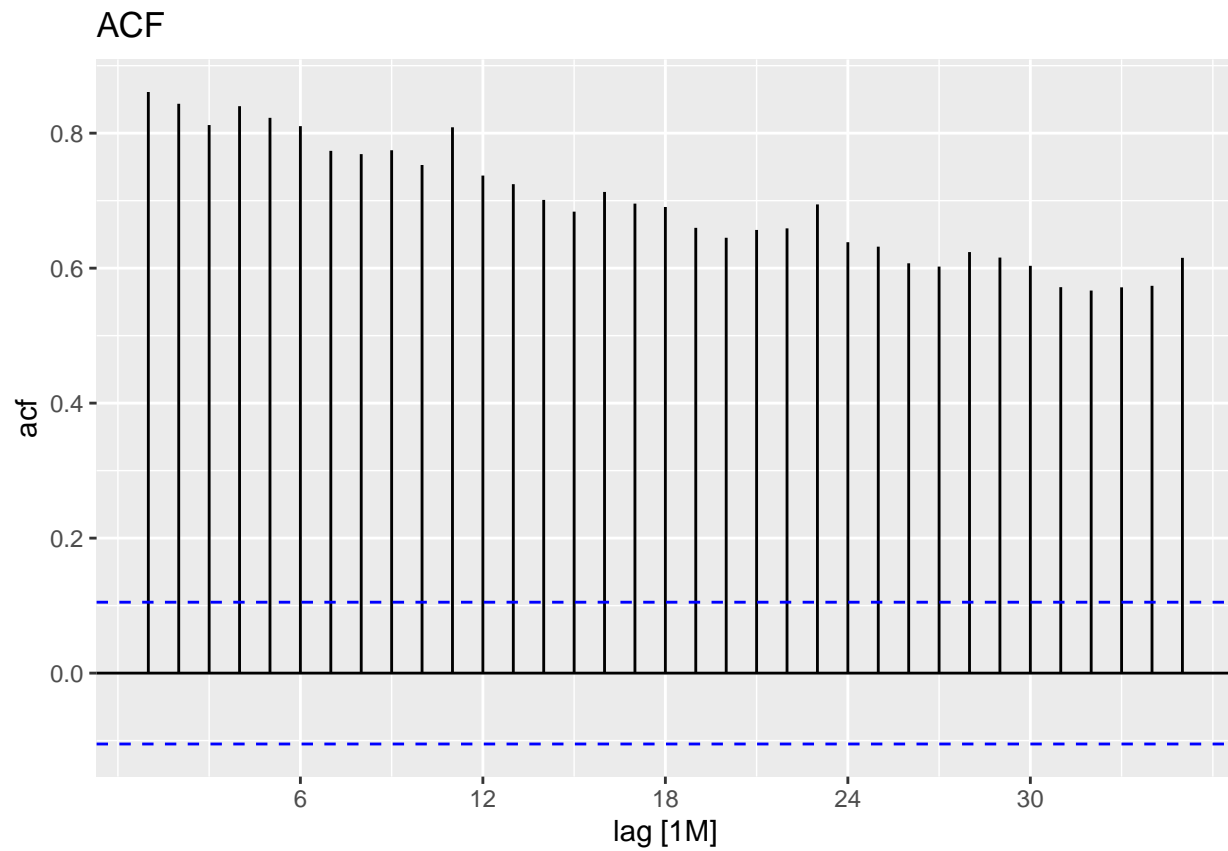
```
plot_serie = train %>%
  autoplot(producao) +
  labs(title="Produção de motocicletas ",
        y="R$ valor ")
plot_serie
```



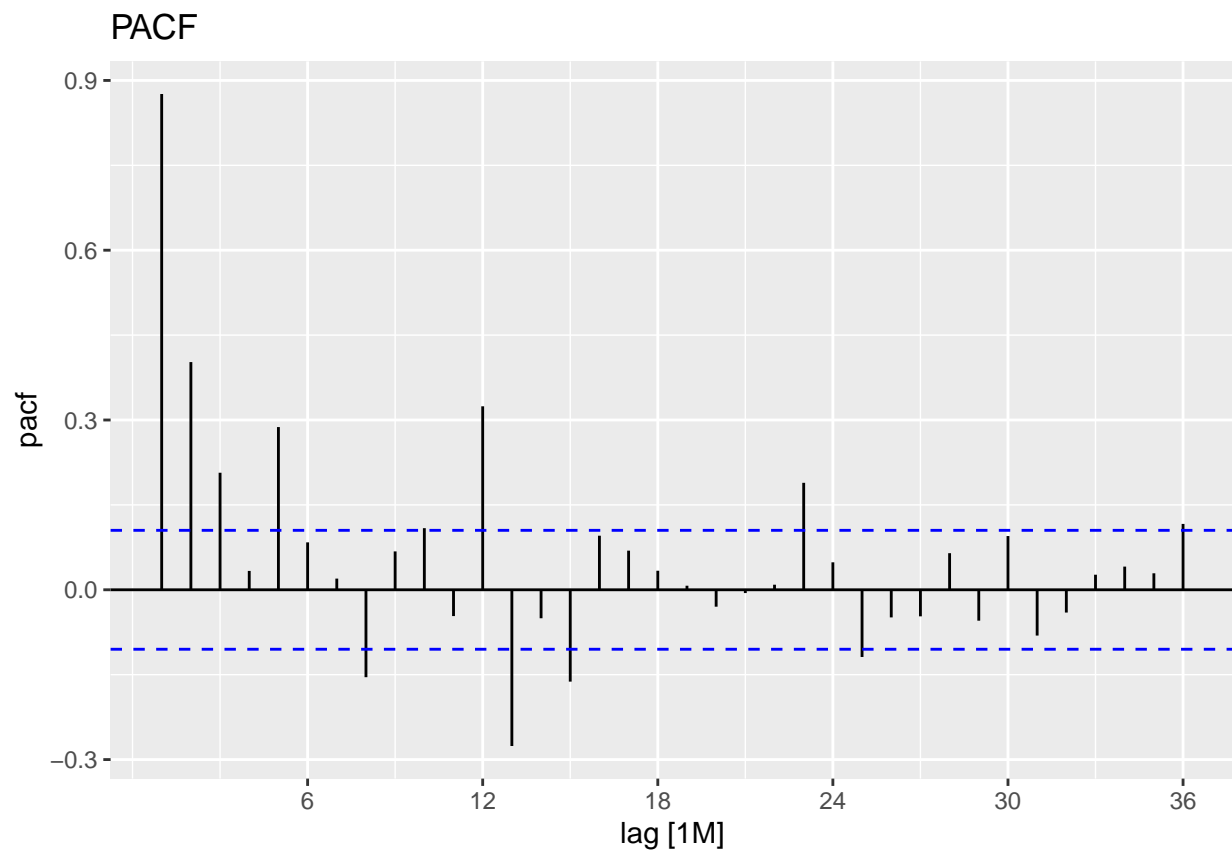
O gráfico apresenta não estacionariedade, pois, pode-se observar que o gráfico possui tendência e não tem sazonalidade.

Gráfico ACF e PACF do treino

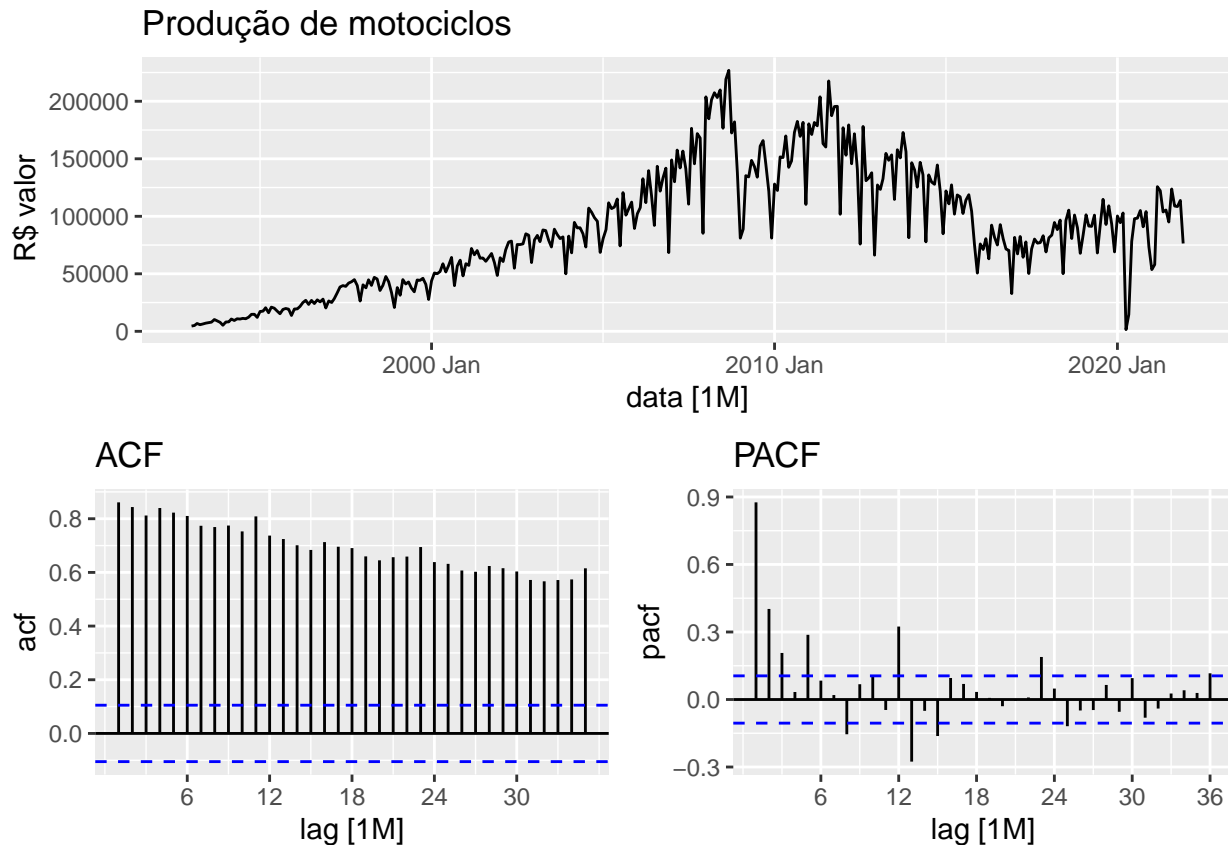
```
acf = train %>%  
  ACF(producao, lag_max = 36) %>%  
  autoplot() + labs(title="ACF")  
acf
```



```
pacf = train %>%  
  PACF(producao, lag_max = 36) %>%  
  autoplot() + labs(title="PACF")  
pacf
```



```
gridExtra::grid.arrange(plot_serie,acf,pacf,layout_matrix=rbind(c(1,1),c(2,3) ))
```



Identificamos na acf um declínio rápido a zero e significativo para conclusão da presença de tendência nos dados em vários valores positivos, levando a escolha do uso do modelo MA(11) e na pacf AR(3).

Verificando-se o treinamento é estacionária

```
train %>%
  features(producao, unitroot_kpss)
```

```
## # A tibble: 1 x 2
##   kpss_stat kpss_pvalue
##   <dbl>     <dbl>
## 1      2.93         0.01
```

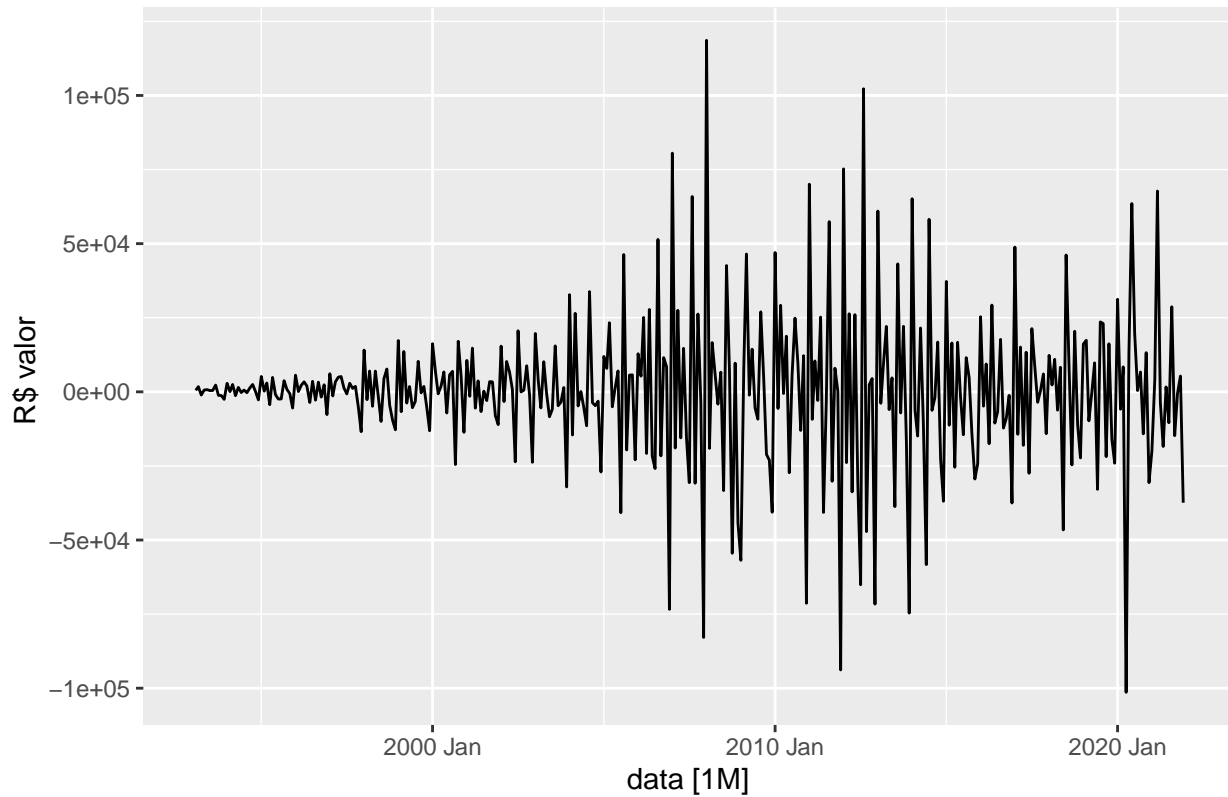
Como $Kpss_value = 0,01 < pvalor = 0,05$, então, rejeitar-se H_0 e conclui que não é estacionária.

Aplicando o operador de diferença no treinamento

Gráfico do operador de diferença no treinamento

```
plot_serieD = train %>%
  autoplot( difference(producao) ) +
  labs(title=" ",
       y="R$ valor ")
plot_serieD
```

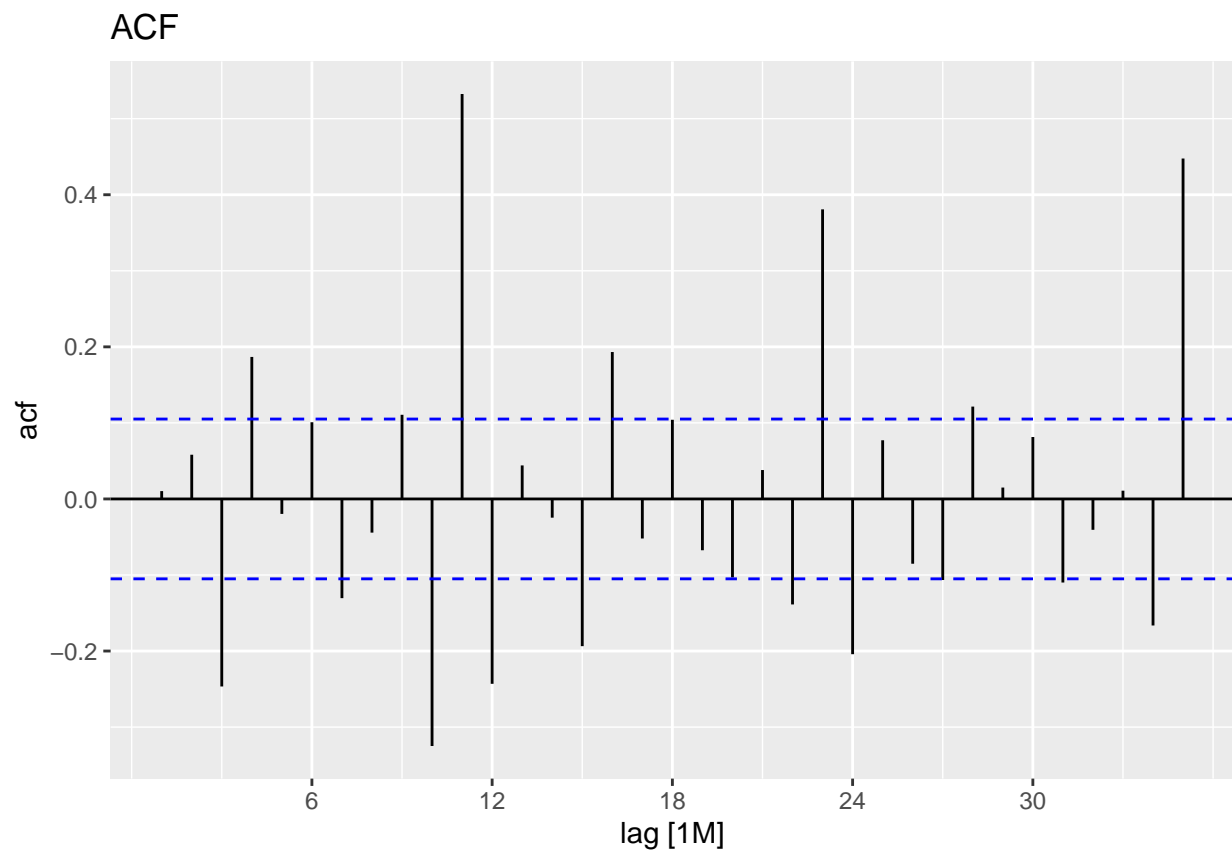
```
## Warning: Removed 1 row containing missing values (`geom_line()`).
```



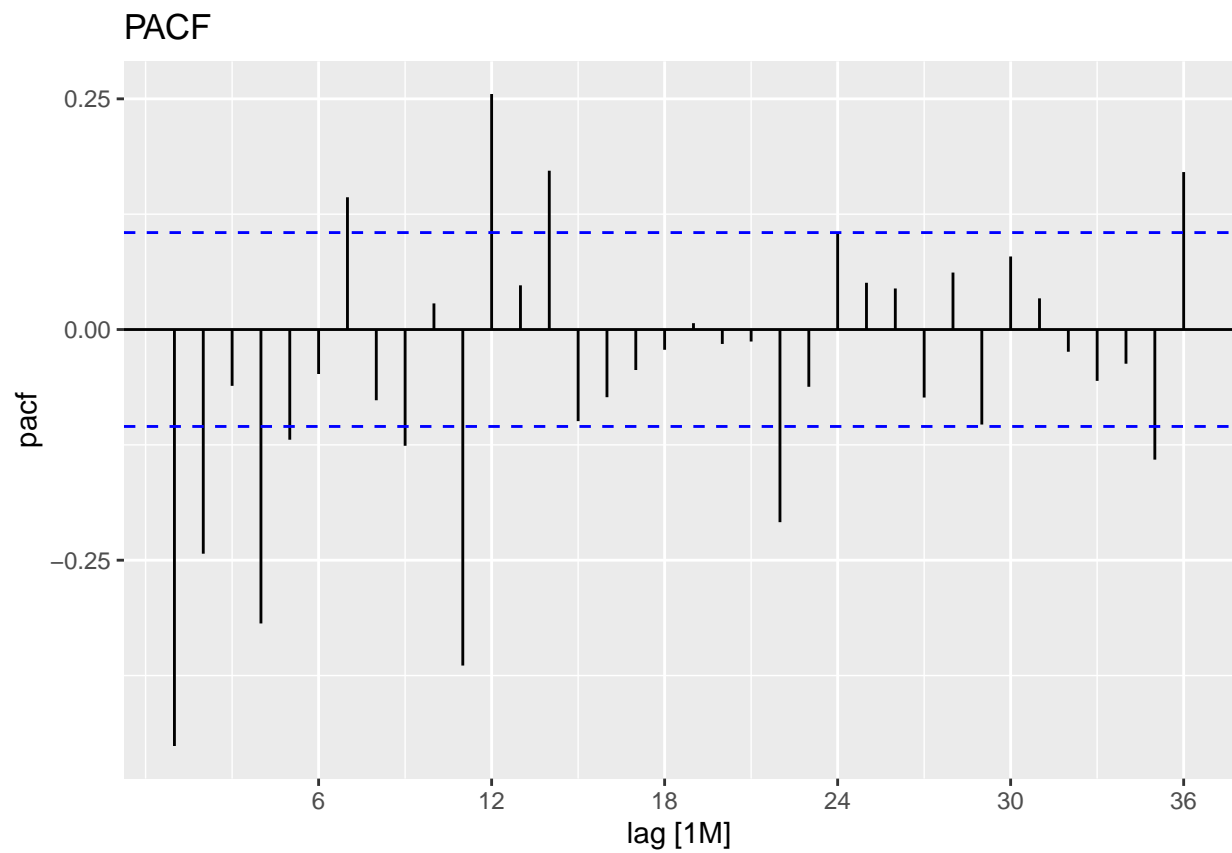
Pode-se observar que o gráfico está estacionário, isto é, ao longo do tempo os valores da produção de motocicletas estão aleatoriamente próximo de uma média constante. E o gráfico apresenta tendência e sazonalidade.

Gráfico ACF e PACF do operador diferença no treinamento

```
acf_D = train %>%
  ACF(difference(producao), lag_max = 36) %>%
  autoplot() + labs(title="ACF")
acf_D
```

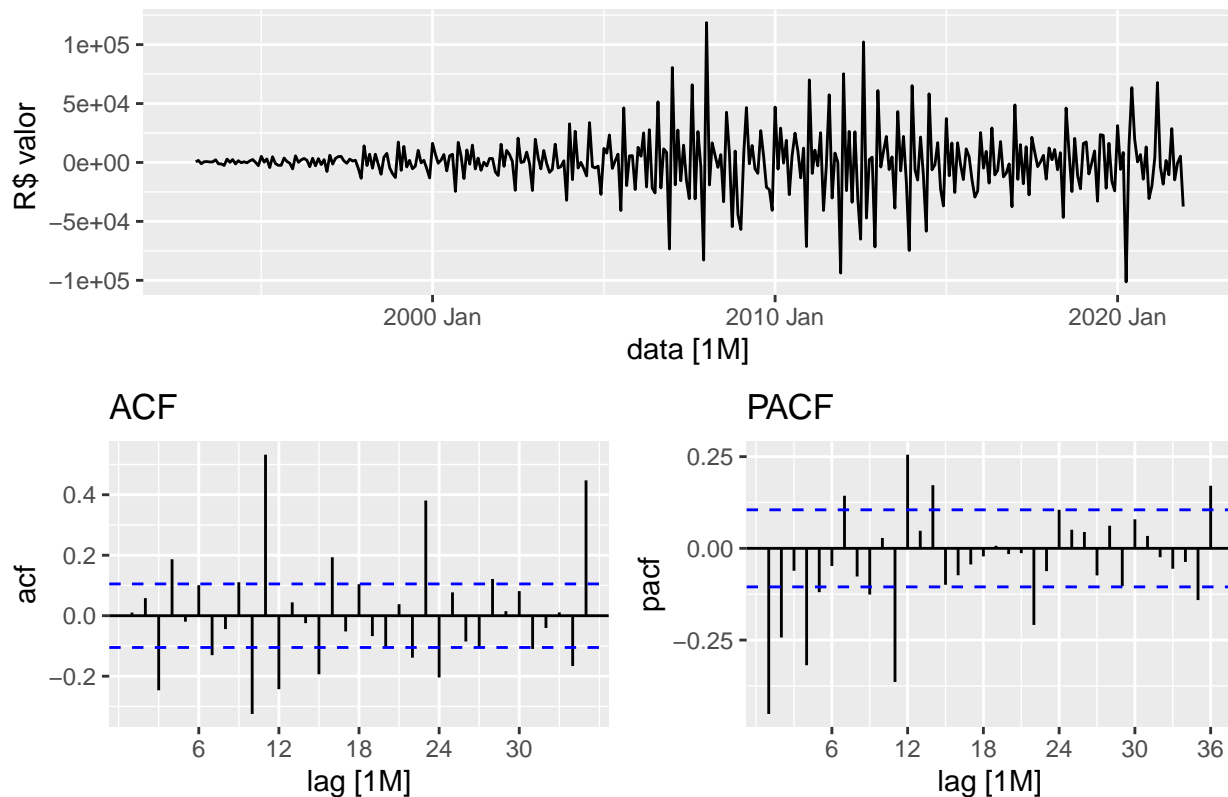


```
pacf_D = train %>%
  PACF(difference(producao), lag_max = 36) %>%
  autoplot() + labs(title="PACF")
pacf_D
```

```
gridExtra::grid.arrange(plot_seriesD, acf_D, pacf_D, layout_matrix=rbind(c(1,1),c(2,3) ))
```

```
## Warning: Removed 1 row containing missing values (`geom_line()`).
```



Observar-se que o gráfico apresenta tendência e sazonalidade a cada ano, ou seja, a partir disso será feito uma análise no modelo SARIMA. Identificamos na acf um modelo MA(3) e na pacf o modelo AR(2). Sabendo que a serie é mensal, o período $m=12$ corresponde a quantidade de ano. A partir dessa análise vamos indetcticar o melhor medelo para prever os próximos passos.

Estimar modelos especificados

```
caf_fit <- train %>% model(
  ETS1 = ETS( producao ~ error("A") + trend("N") + season("N")),
  ETS2 = ETS( producao ~ error("A") + trend("A") + season("N")),
  ETS3 = ETS( producao ~ error("A") + trend("M") + season("N")),
  ETS4 = ETS( producao ~ error("A") + trend( "N" ) + season("A")),
  ETS5 = ETS( producao ~ error("A") + trend( "N" ) + season("M")),
  ARIMA011002 = ARIMA(producao ~ pdq(0,1,1) + PDQ(0,0,2)),
  ARIMA012022 = ARIMA(producao ~ pdq(0,1,2) + PDQ(0,0,2)),
  ARIMA012013 = ARIMA(producao ~ pdq(0,1,2) + PDQ(0,1,3)),
  auto = ARIMA(producao),
  auto1 = ETS(producao))
caf_fit
```

```
## # A mable: 1 x 10
##           ETS1           ETS2           ETS3           ETS4           ETS5
##           <model>         <model>         <model>         <model>         <model>
## 1 <ETS(A,N,N)> <ETS(A,A,N)> <ETS(A,M,N)> <ETS(A,N,A)> <ETS(A,N,M)>
## # ... with 5 more variables: ARIMA011002 <model>, ARIMA012022 <model>,
## #   ARIMA012013 <model>, auto <model>, auto1 <model>
```

Desempenho dos modelos

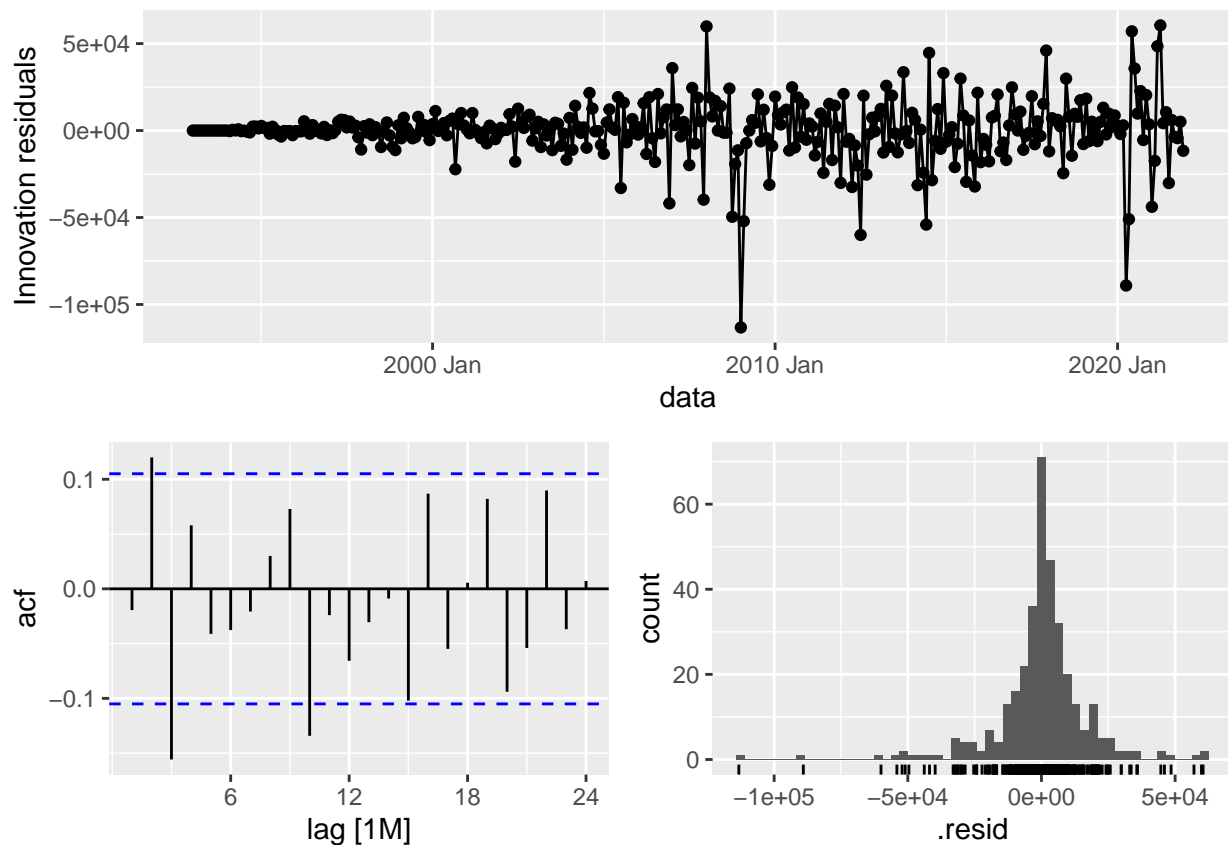
```
glance(caf_fit)
```

```
## # A tibble: 10 x 11
##   .model      sigma2 log_lik   AIC   AICc   BIC      MSE    AMSE      MAE ar_ro~1
##   <chr>      <dbl>   <dbl> <dbl> <dbl> <dbl>   <dbl>   <dbl>   <dbl> <list>
## 1 ETS1      4.55e+8 -4486. 8978. 8978. 8990.   4.52e8  5.05e8  1.38e+4 <NULL>
## 2 ETS2      4.58e+8 -4486. 8982. 8982. 9001.   4.52e8  5.05e8  1.34e+4 <NULL>
## 3 ETS3      4.58e+8 -4486. 8982. 8982. 9002.   4.52e8  5.05e8  1.38e+4 <NULL>
## 4 ETS4      3.13e+8 -4415. 8860. 8862. 8918.   3.01e8  3.46e8  1.14e+4 <NULL>
## 5 ETS5      2.58e+8 -4381. 8792. 8793. 8850.   2.47e8  2.88e8  9.66e+3 <NULL>
## 6 ARIMA0110~ 3.73e+8 -3917. 7841. 7842. 7857.    NA      NA      NA      <cpl>
## 7 ARIMA0120~ 3.73e+8 -3916. 7842. 7843. 7862.    NA      NA      NA      <cpl>
## 8 ARIMA0120~ 3.10e+8 -3753. 7518. 7519. 7541.    NA      NA      NA      <cpl>
## 9 auto      3.73e+8 -3917. 7841. 7842. 7857.    NA      NA      NA      <cpl>
## 10 auto1     2.71e-2 -4262. 8558. 8560. 8623.   2.69e8  3.08e8  1.11e-1 <NULL>
## # ... with 1 more variable: ma_roots <list>, and abbreviated variable name
## #   1: ar_roots
```

O melhor modelo para prever os próximos passos será o ARIMA012013, dado que, possui o menor AIC,AICc e BIC.

Comportamento dos resíduos

```
caf_fit %>%
  select(ARIMA012013) %>%
  gg_tsresiduals()
```



Gerando uma previsão $h = 10$ (passos)

```
caf_fc <- caf_fit %>% forecast(h = 10)
caf_fc
```

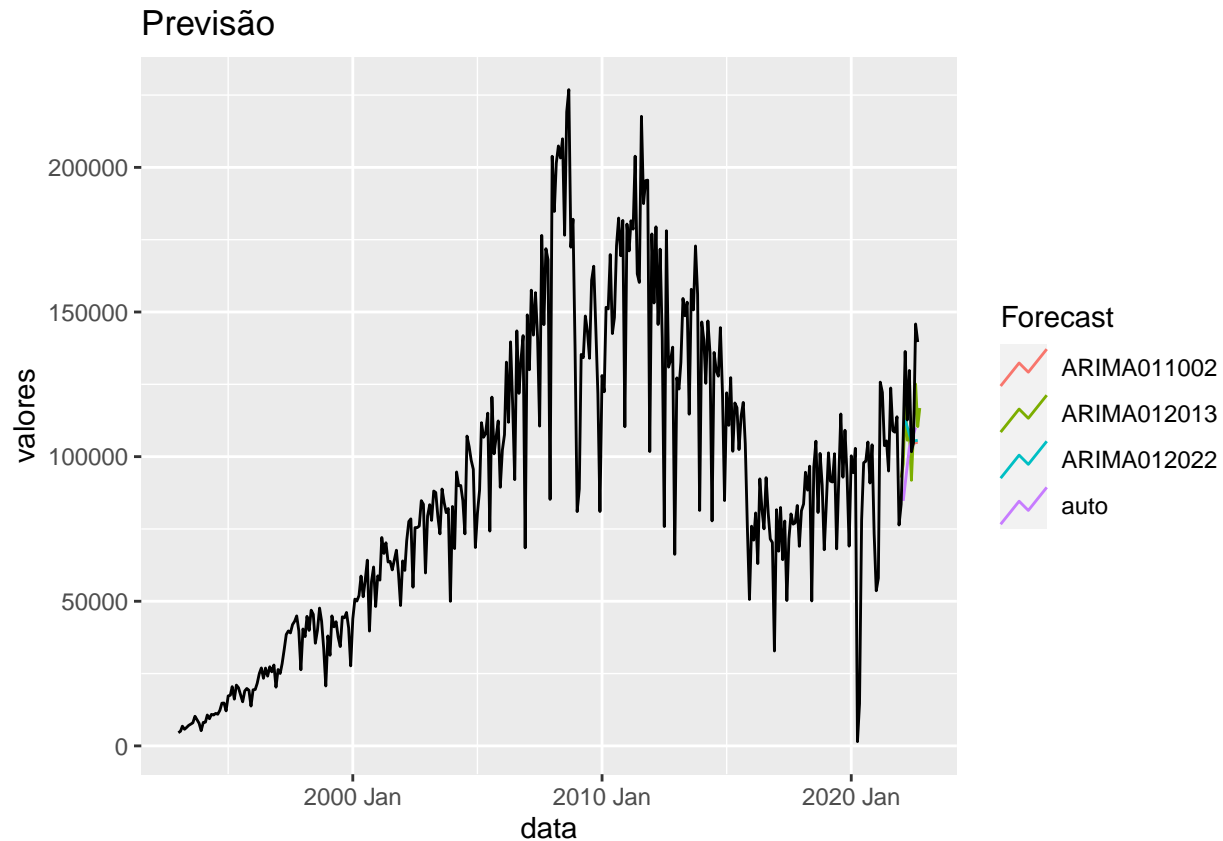
```
## # A tibble: 100 x 4 [1M]
## # Key:   .model [10]
##   .model    data      producao .mean
##   <chr>    <mth>          <dist> <dbl>
## 1 ETS1    2022 Jan N(99503, 4.5e+08) 99503.
## 2 ETS1    2022 Feb  N(99503, 5e+08)  99503.
## 3 ETS1    2022 Mar  N(99503, 5.4e+08) 99503.
## 4 ETS1    2022 Apr  N(99503, 5.8e+08) 99503.
## 5 ETS1    2022 May  N(99503, 6.2e+08) 99503.
## 6 ETS1    2022 Jun  N(99503, 6.7e+08) 99503.
## 7 ETS1    2022 Jul   N(99503, 7.1e+08) 99503.
## 8 ETS1    2022 Aug   N(99503, 7.5e+08) 99503.
## 9 ETS1    2022 Sep   N(99503, 7.9e+08) 99503.
## 10 ETS1   2022 Oct   N(99503, 8.4e+08) 99503.
## # ... with 90 more rows
```

Gráfico da serie com as previsões

```
caf_fc %>% filter( (.model == 'ARIMA012013') | (.model == c('auto', "ARIMA011002", "ARIMA012022"))) %>%
  autoplot(serie, level = NULL) +
```

```
labs(
  y = "valores",
  title = "Previsão"
) +
guides(colour = guide_legend(title = "Forecast"))
```

```
## Warning in .model == c("auto", "ARIMA011002", "ARIMA012022"): longer object
## length is not a multiple of shorter object length
```



Accuracia do teste do modelo

```
accuracy(caf_fc, test)
```

```
## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 1 observation is missing at 2022 Oct
```

```
## # A tibble: 10 x 10
##   .model      .type      ME    RMSE    MAE    MPE    MAPE    MASE  RMSSE    ACF1
##   <chr>      <chr>    <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
## 1 ARIMA011002 Test  16381. 21913. 17067. 12.4   13.1   NaN   NaN  -0.0855
## 2 ARIMA012013 Test  11312. 16271. 14301.  8.24  11.6   NaN   NaN  -0.0119
## 3 ARIMA012022 Test  15591. 21208. 16425. 11.8   12.6   NaN   NaN  -0.0746
## 4 auto       Test  16381. 21913. 17067. 12.4   13.1   NaN   NaN  -0.0855
## 5 auto1      Test   6197. 14580. 12292.  3.39  10.3   NaN   NaN   0.0381
## 6 ETS1       Test  18424. 26977. 21971. 13.1   17.3   NaN   NaN   0.0785
## 7 ETS2       Test  14691. 23996. 19071.  9.94  15.1   NaN   NaN   0.0483
## 8 ETS3       Test  18779. 27357. 22334. 13.4   17.6   NaN   NaN   0.0835
```

```
## 9 ETS4      Test  9959. 17343. 14687.  6.52 12.1  NaN  NaN  0.0917
## 10 ETS5     Test  8615. 15627. 13316.  5.56 11.1  NaN  NaN  0.103
```

Já na accuracy o melhor modelo que deve ser utilizado é o auto1 que é referente ao ETS, dado que, possui o menor ME, RMSE, MAE, MPE, MAPE entre todos os modelos. Para isso, devemos refazer a geração de passos no modelo indicado.

Comportamento dos resíduos

```
caf_fit %>%
  select(auto1) %>%
  gg_tsresiduals()
```

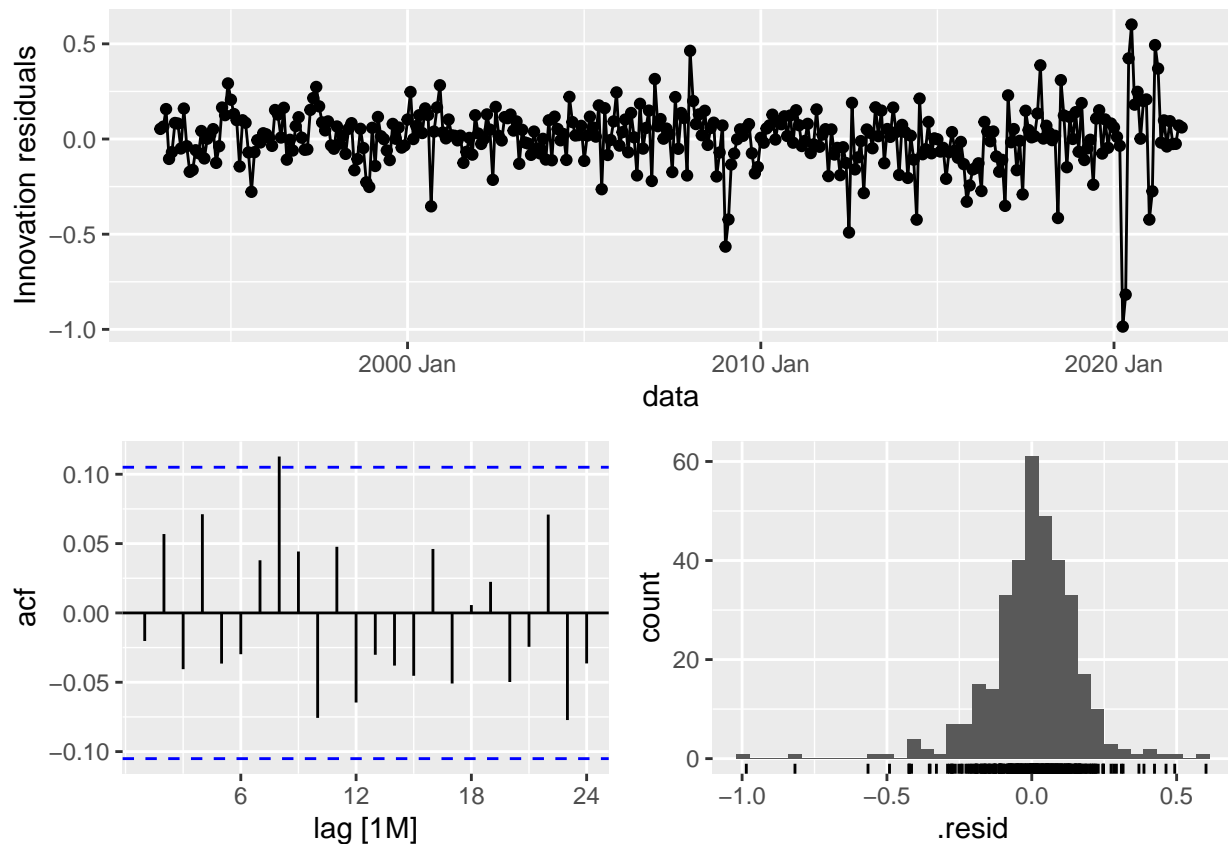
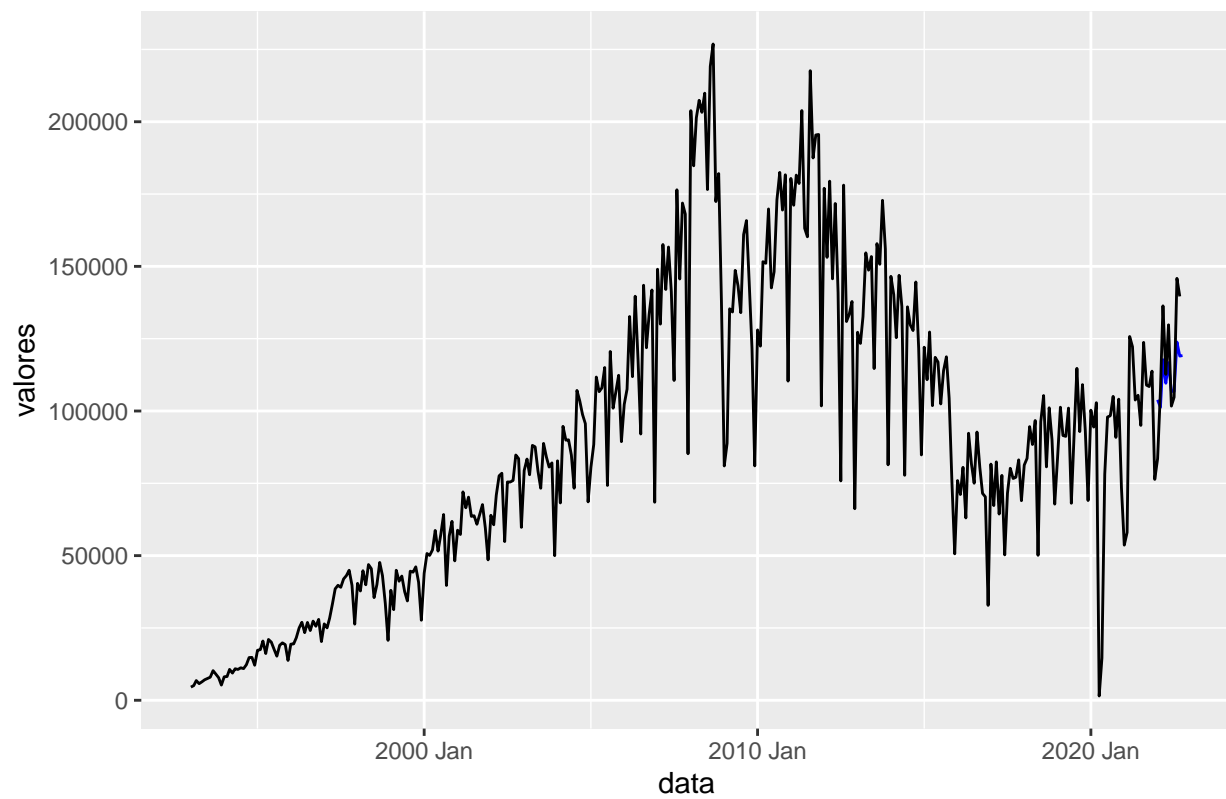


Gráfico da serie com as previsões

```
caf_fc %>% filter( (.model == 'auto1')) %>%
  autoplot(serie, level = NULL) +
  labs(
    y = "valores",
    title = "Previsão"
  ) +
  guides(colour = guide_legend(title = "Forecast"))
```

Previsão



Comportamento dos resíduos

```
caf_fit %>%  
  select(auto1) %>%  
  gg_tsresiduals()
```

