SARIMA

gpetrini

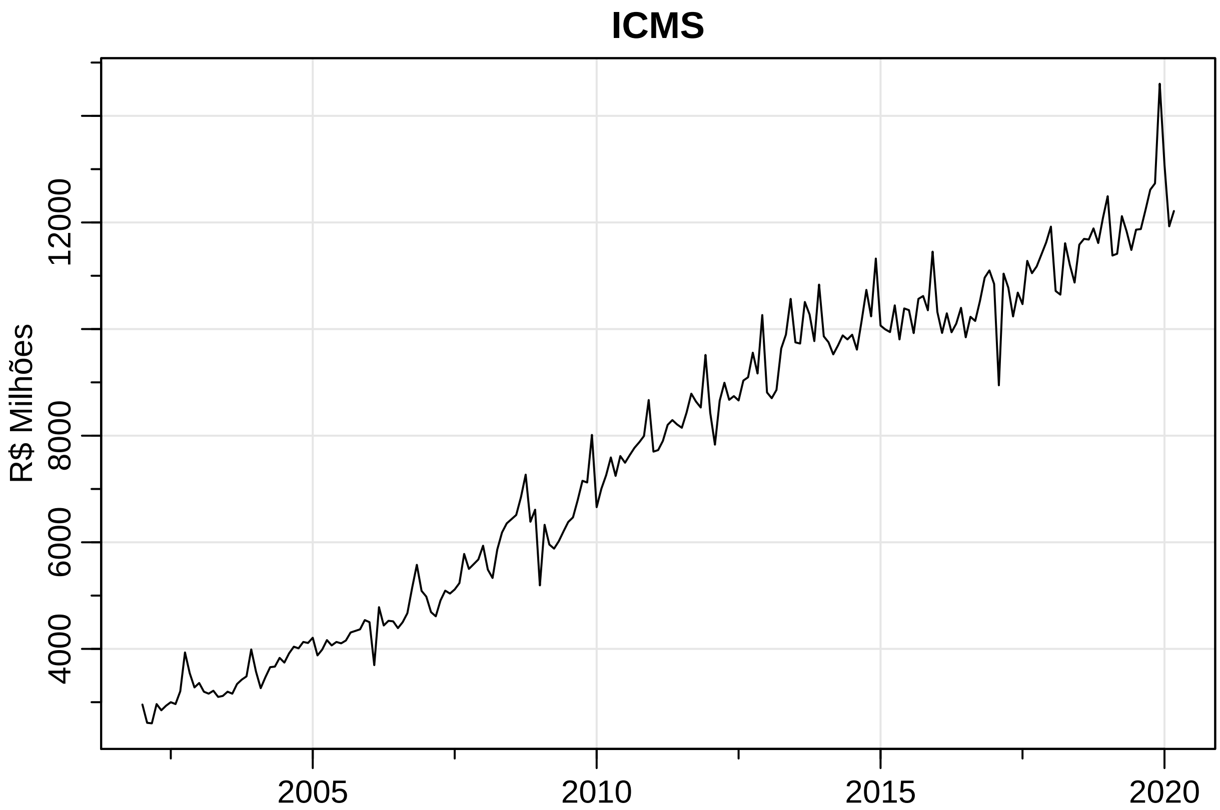
19/05/2020

Pacotes

```
library(tidyverse)
```

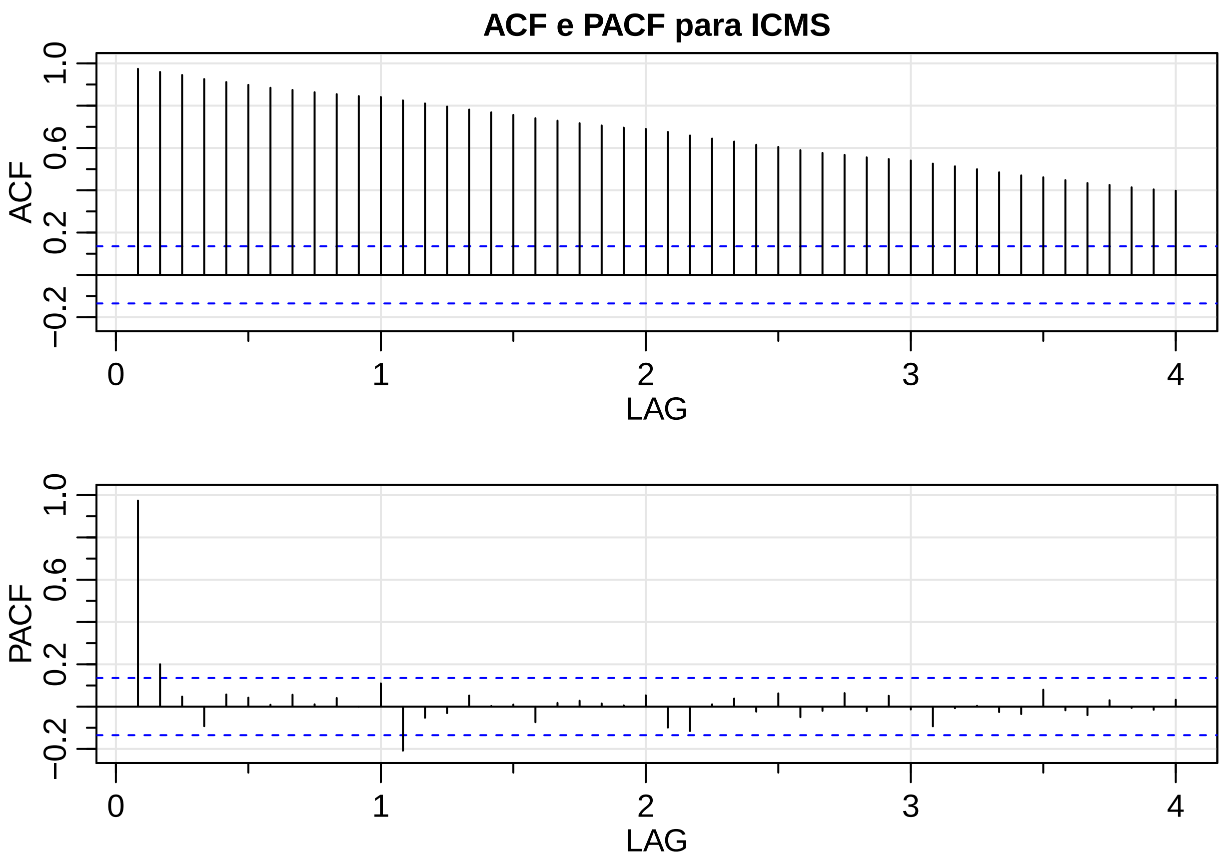
Dados

```
df <- read.csv(
    "../data/raw_data.csv",
    encoding="UTF-8",
    stringsAsFactors=FALSE
    )
df <- df[,c( # Subset das colunas
    "ICMS.Nominal.milhões.de.reais"
    )]
df <- ts(
    data = df,
    start = c(2002,01),
    frequency = 12
    )
df %>% astsa::tsplot(
    main = "ICMS",
    ylab = "R$ Milhões",
    xlab = ""
        )
```



Inspeção gráfica

```
astsa::acf2(
    series = df,
    main = "ACF e PACF para ICMS"
    )
```




```
##      [,1] [,2] [,3]  [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
## ACF  0.97 0.96 0.94  0.93 0.91 0.90 0.88 0.87 0.86  0.85  0.85  0.84  0.82
## PACF 0.97 0.20 0.05 -0.09 0.06 0.04 0.01 0.06 0.01  0.04  0.00  0.11 -0.21
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25]
## ACF   0.81  0.80  0.78  0.77  0.76  0.74  0.73  0.72  0.71  0.70  0.69  0.68
## PACF -0.05 -0.03  0.05  0.00  0.01 -0.07  0.02  0.03  0.01  0.01  0.05 -0.10
##      [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37]
## ACF   0.66  0.64  0.63  0.62  0.61  0.59  0.58  0.57  0.56  0.55  0.54  0.53
## PACF -0.12  0.01  0.04 -0.02  0.06 -0.05 -0.02  0.06 -0.02  0.05 -0.01 -0.09
##      [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48]
## ACF   0.51   0.5  0.48  0.47  0.46  0.45  0.43  0.43  0.41  0.40  0.40
## PACF -0.01   0.0 -0.03 -0.04  0.08 -0.02 -0.04  0.03 -0.01 -0.01  0.03
```

Estimação

Auto ARIMA

```
forecast::auto.arima(df)
```

```
## Series: df
## ARIMA(0,1,2)(0,0,2)[12] with drift
##
```

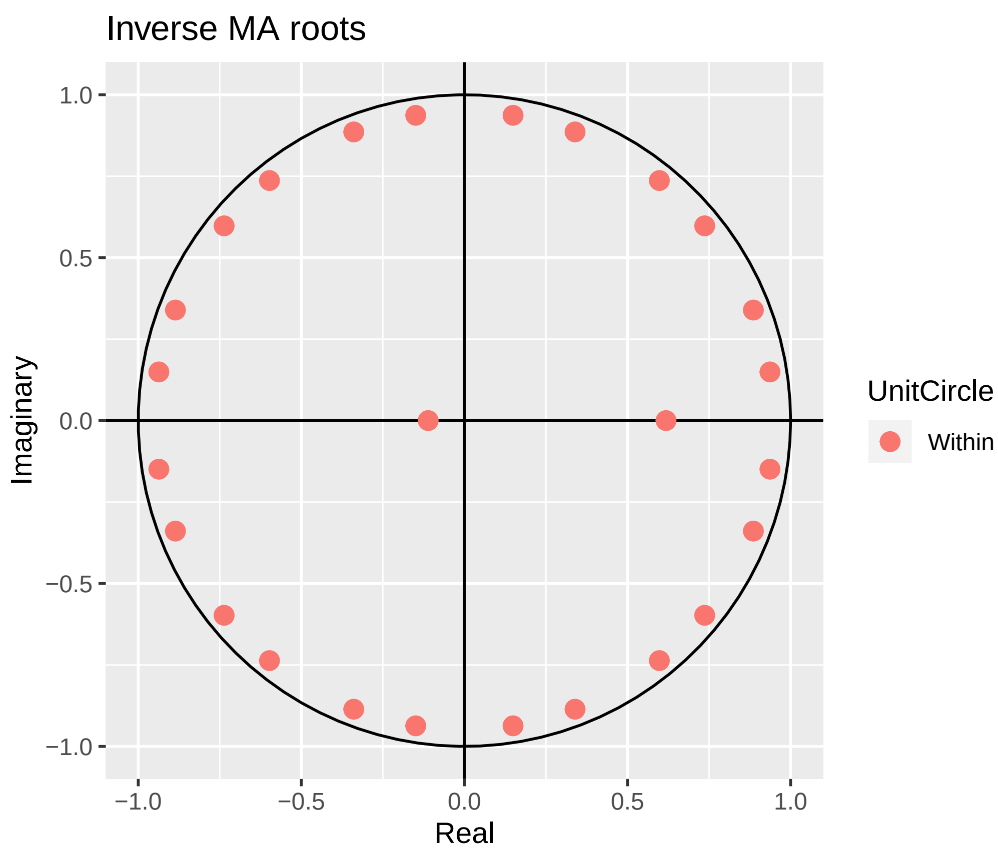
```
## Coefficients:
##           ma1      ma2    sma1    sma2    drift
##       -0.6458  -0.1618  0.3864  0.2924  45.4887
## s.e.   0.0673   0.0732  0.0710  0.0665   8.3972
##
## sigma^2 estimated as 153437:  log likelihood=-1610.43
## AIC=3232.87   AICc=3233.27   BIC=3253.18
```

```r
forecast::Arima(
    y = df,
    order = c(0,1,2),
    seasonal = list(order=c(0,0,2),period=12),
    lambda = 0
) -> model
model %>% summary()
```

```
## Series: df
## ARIMA(0,1,2)(0,0,2)[12]
## Box Cox transformation: lambda= 0
##
## Coefficients:
##          ma1      ma2    sma1    sma2
##       -0.507  -0.0687  0.3385  0.2818
## s.e.   0.069   0.0719  0.0726  0.0655
##
## sigma^2 estimated as 0.003024:  log likelihood=323.43
## AIC=-636.85   AICc=-636.57   BIC=-619.93
##
## Training set error measures:
##                    ME     RMSE     MAE      MPE     MAPE      MASE       ACF1
## Training set 70.12772 402.8316 282.473 0.837341 3.828772 0.4650301 -0.1005522
```

Raízes características

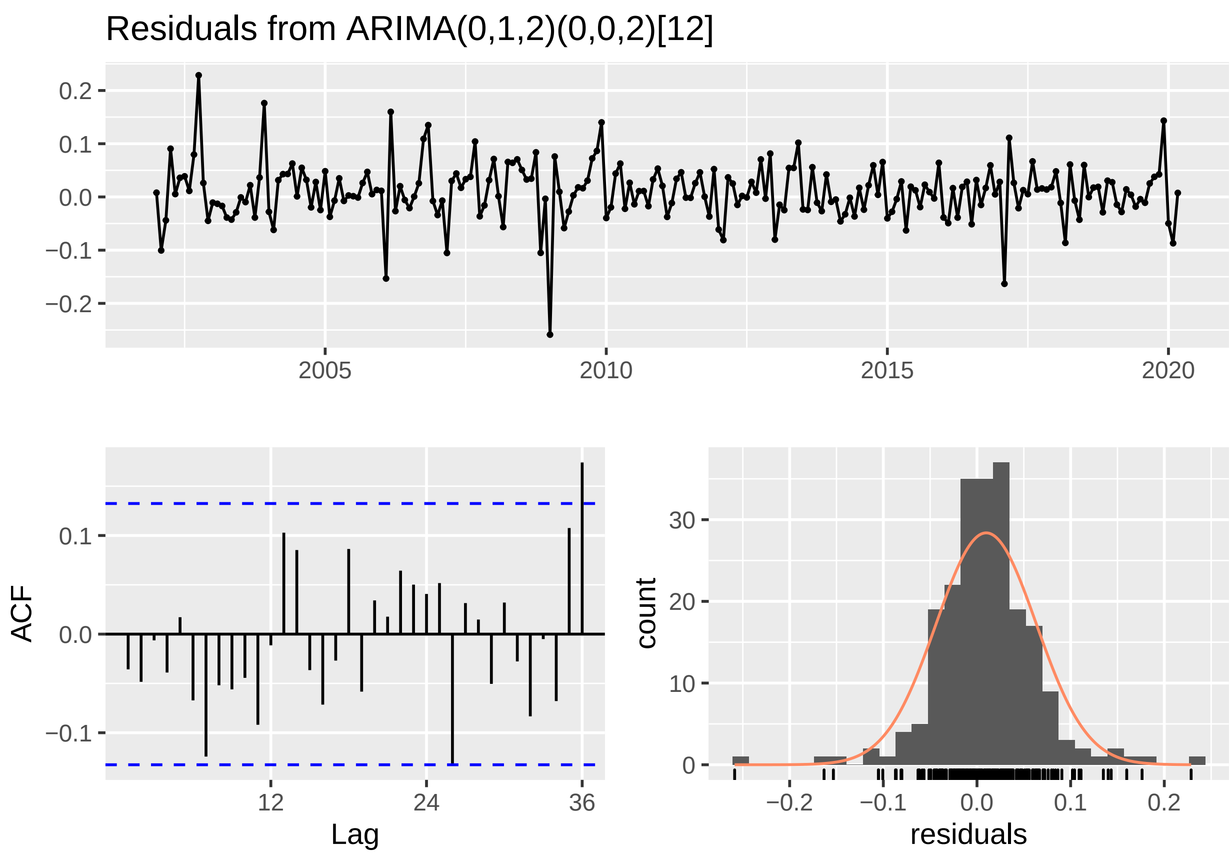
```r
forecast::autoplot(model)
```



Pós-Estimação

Resíduos

```
forecast::checkresiduals(model)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(0,1,2)(0,0,2)[12]
## Q* = 20.492, df = 20, p-value = 0.4275
##
## Model df: 4.   Total lags used: 24
```

Observação: Últimas observações são tão distoantes que são tratados como *outliers*.

```r
for (outlier in forecast::tsoutliers(df)$index) {
    print(paste0("Outlier em ", zoo::index(df)[outlier]))
}
```

```
## [1] "Outlier em 2017.08333333333"
## [1] "Outlier em 2019.91666666667"
```

Box-Pierce

```r
Box.test(
    x = df,
    lag = 15,
    type = "Box-Pierce"
)
```

```
##
##  Box-Pierce test
##
## data:  df
## X-squared = 2556.7, df = 15, p-value < 2.2e-16
```

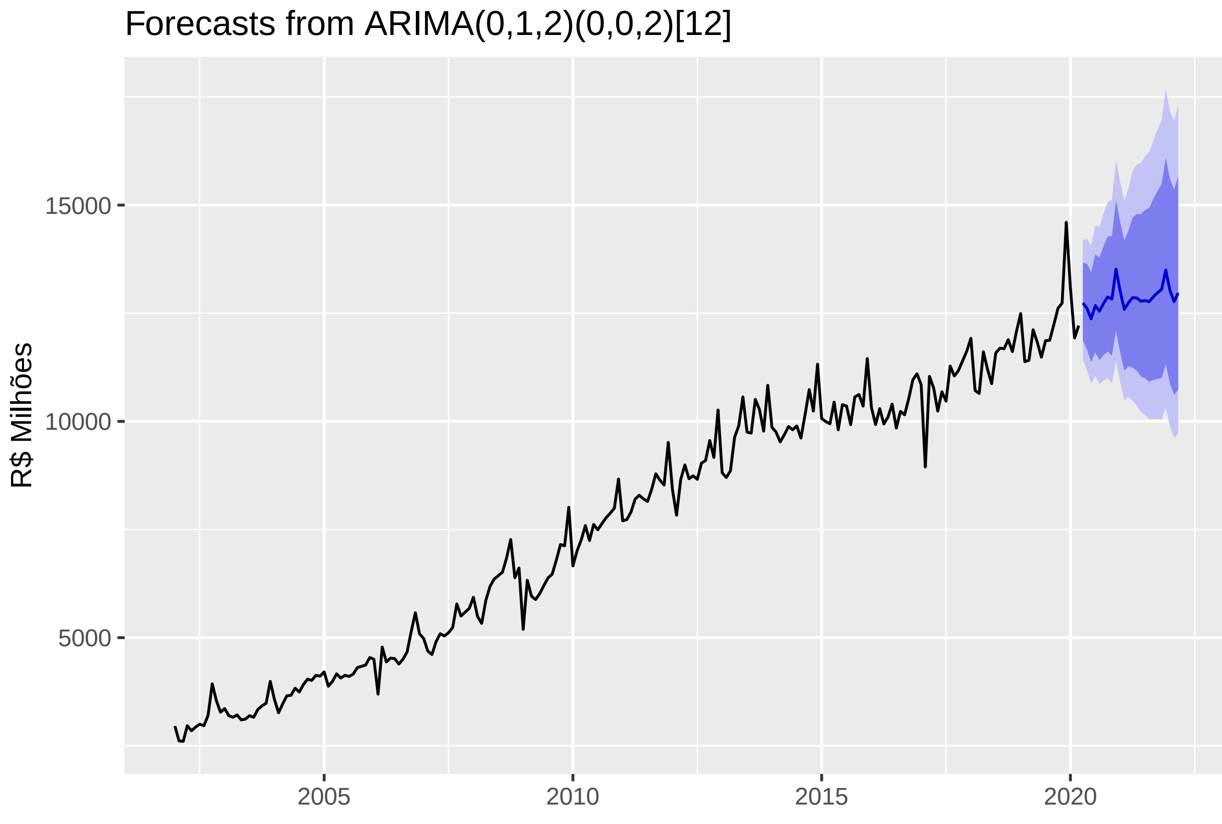
Ljung-Box

```
Box.test(
    x = df,
    lag = 15,
    type = "Ljung-Box"
)
```

```
##
##  Box-Ljung test
##
## data:  df
## X-squared = 2672.5, df = 15, p-value < 2.2e-16
```

Previsão

```
forecast::autoplot(
    forecast::forecast(model),
    ylab="R$ Milhões",
    xlab=""
    )
```



Acurácia

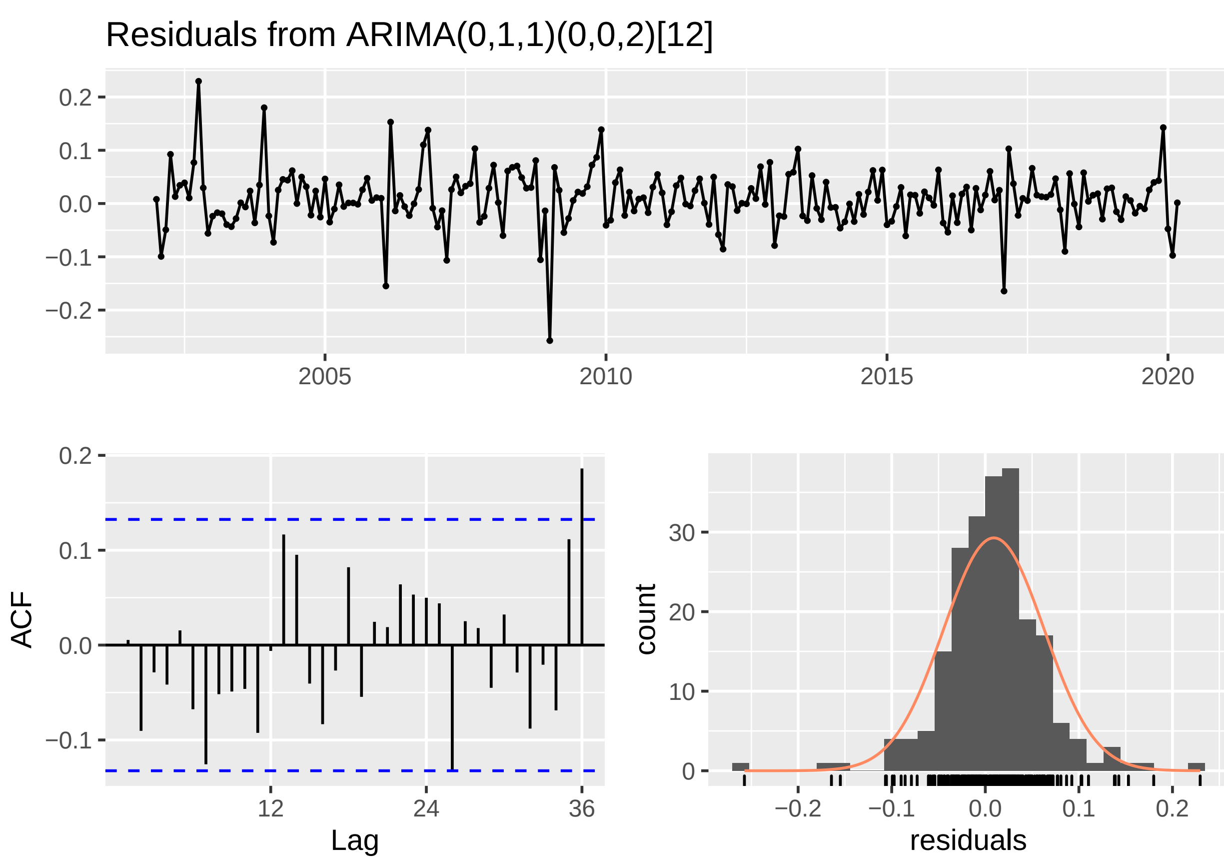
```
forecast::accuracy(model) %>% knitr::kable()
```

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training set	70.12772	402.8316	282.473	0.837341	3.828772	0.4650301	-0.1005522

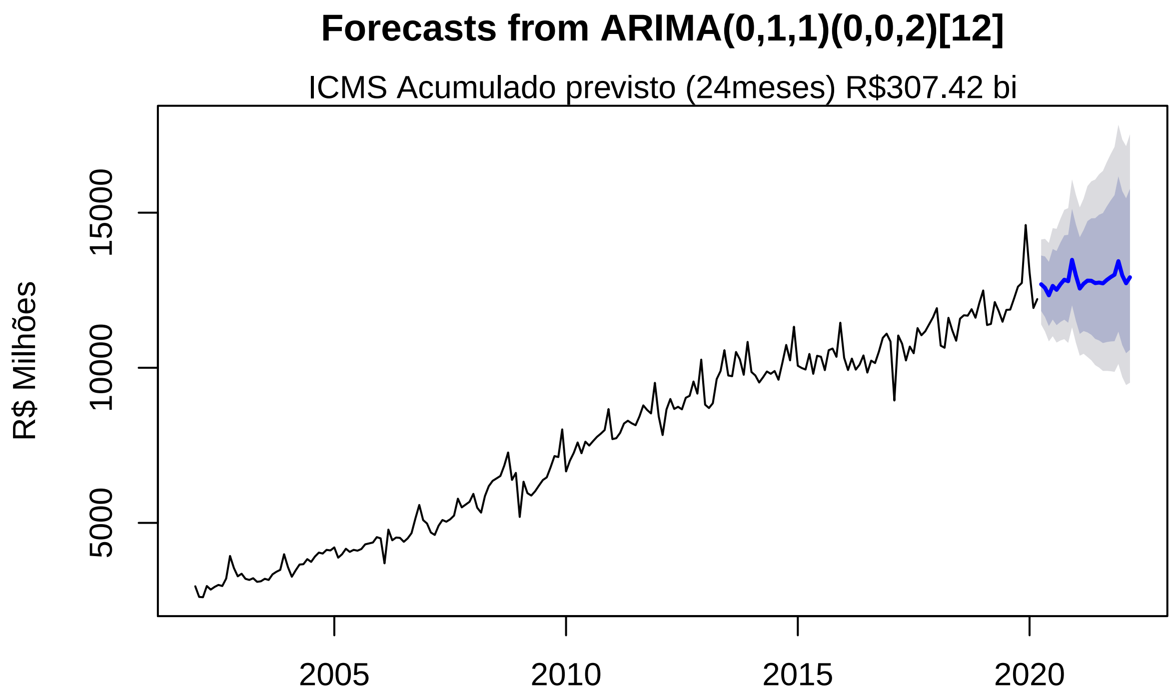
Comparação entre modelos

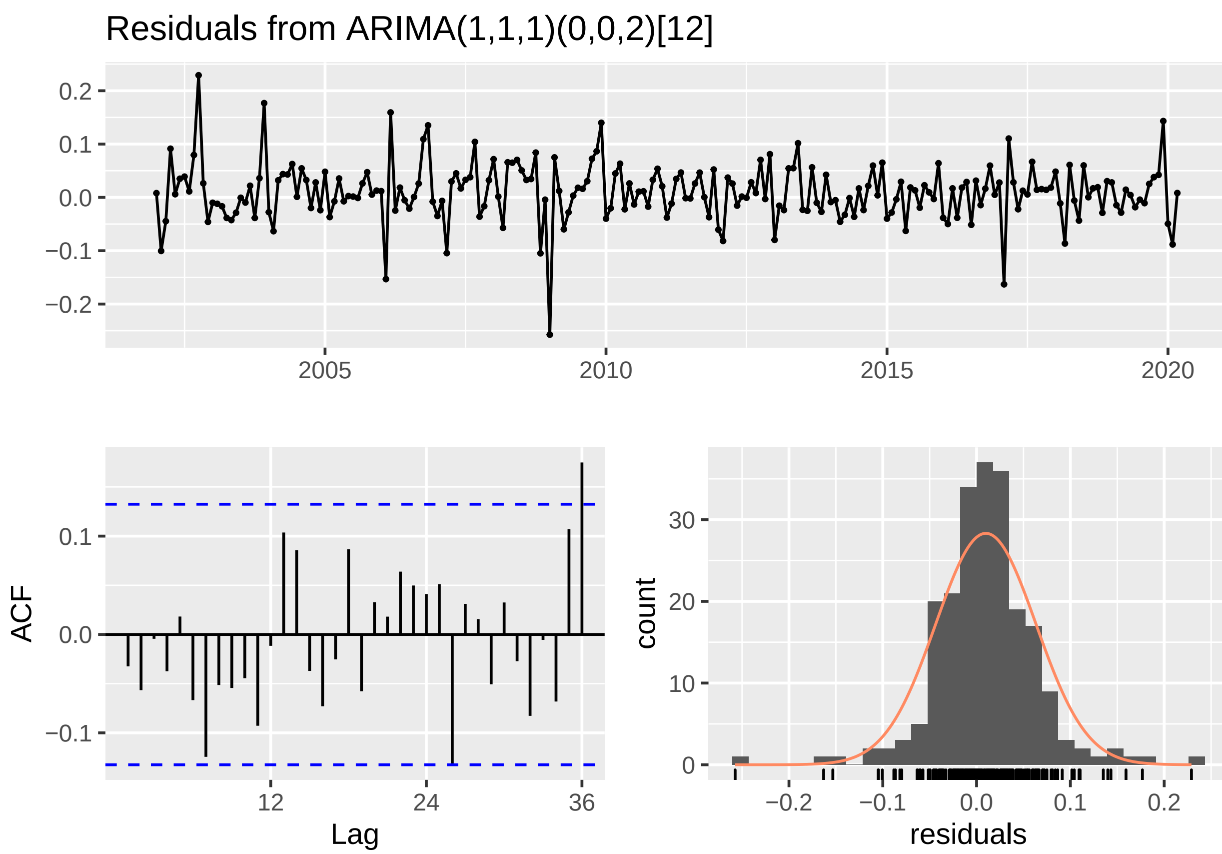
```
candidatos <- list( # Mantendo a ordem sazonal
    c(0,1,1),
    #c(0,1,2), # Default
    c(1,1,1),
    c(1,1,2),
    c(2,1,2)
)

comparacao <- forecast::accuracy(model)
rownames(comparacao) <- c("Modelo Base")
```

```
for (ordem in candidatos) {
    texto <- paste0(
        "SARIMA(",
        ordem[1],",",
        ordem[2],",",
        ordem[3],
        ")(0,0,2)[12]")
    fit <- forecast::Arima(
    y = df,
    order = ordem,
    seasonal = list(order=c(0,0,2),period=12),
    lambda = 0)
    forecast::checkresiduals(
        fit,
        main =texto)
    forecast::forecast(fit) %>%
       plot(ylab="R$ Milhões", xlab="")
    duracao <- forecast::forecast(fit)[4]$mean %>% length()
    previsto <- forecast::forecast(fit)[4] %>% as.data.frame() %>% sum()
    mtext(paste0("ICMS Acumulado previsto (", duracao, "meses) R$", (previsto/1000) %>% round(digits = 
    comparacao <- rbind(comparacao,forecast::accuracy(fit))
    rownames(comparacao)[dim(comparacao)[1]] <- texto

}
```

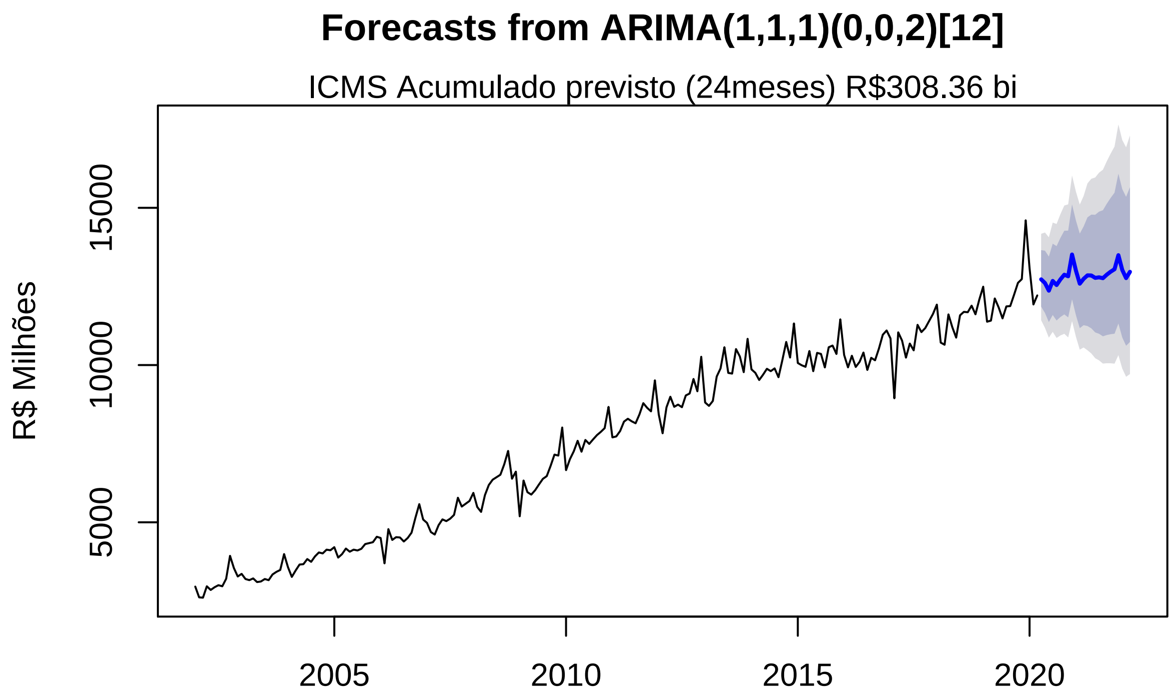


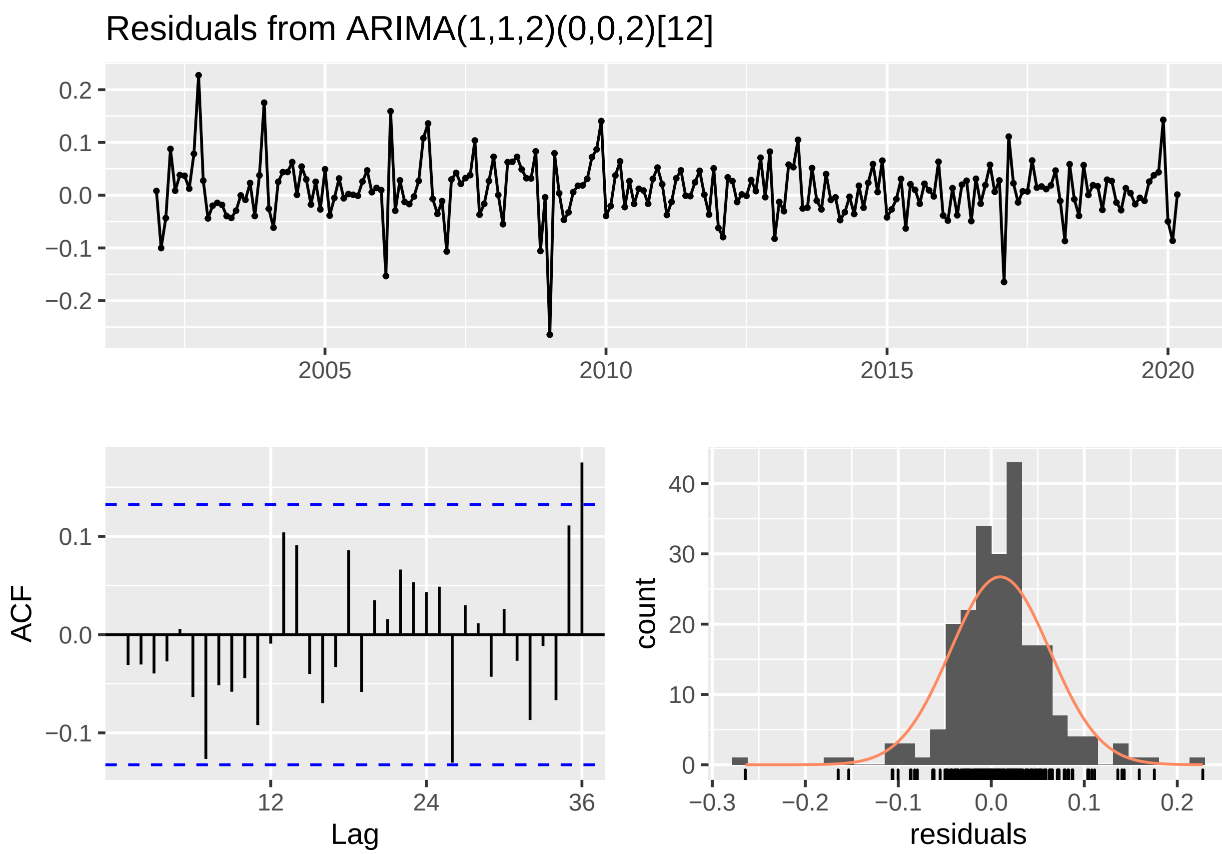
```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(0,1,1)(0,0,2)[12]
## Q* = 23.205, df = 21, p-value = 0.3331
##
## Model df: 3.   Total lags used: 24
```



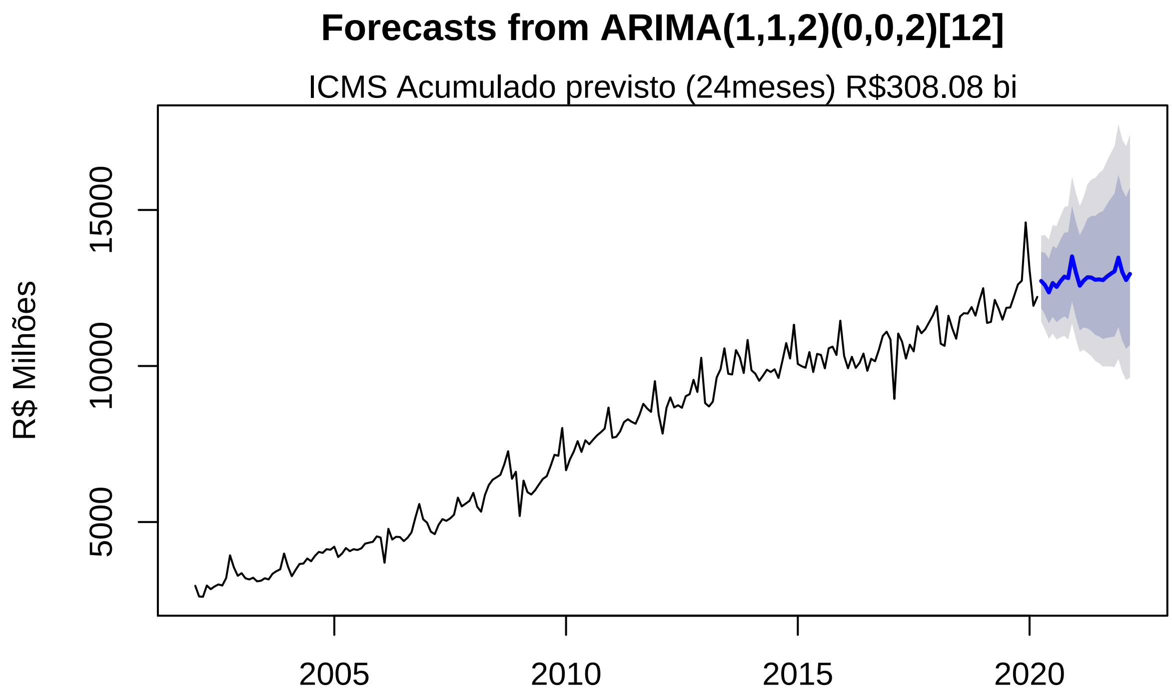



```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(1,1,1)(0,0,2)[12]
## Q* = 20.663, df = 20, p-value = 0.4172
##
## Model df: 4.   Total lags used: 24
```

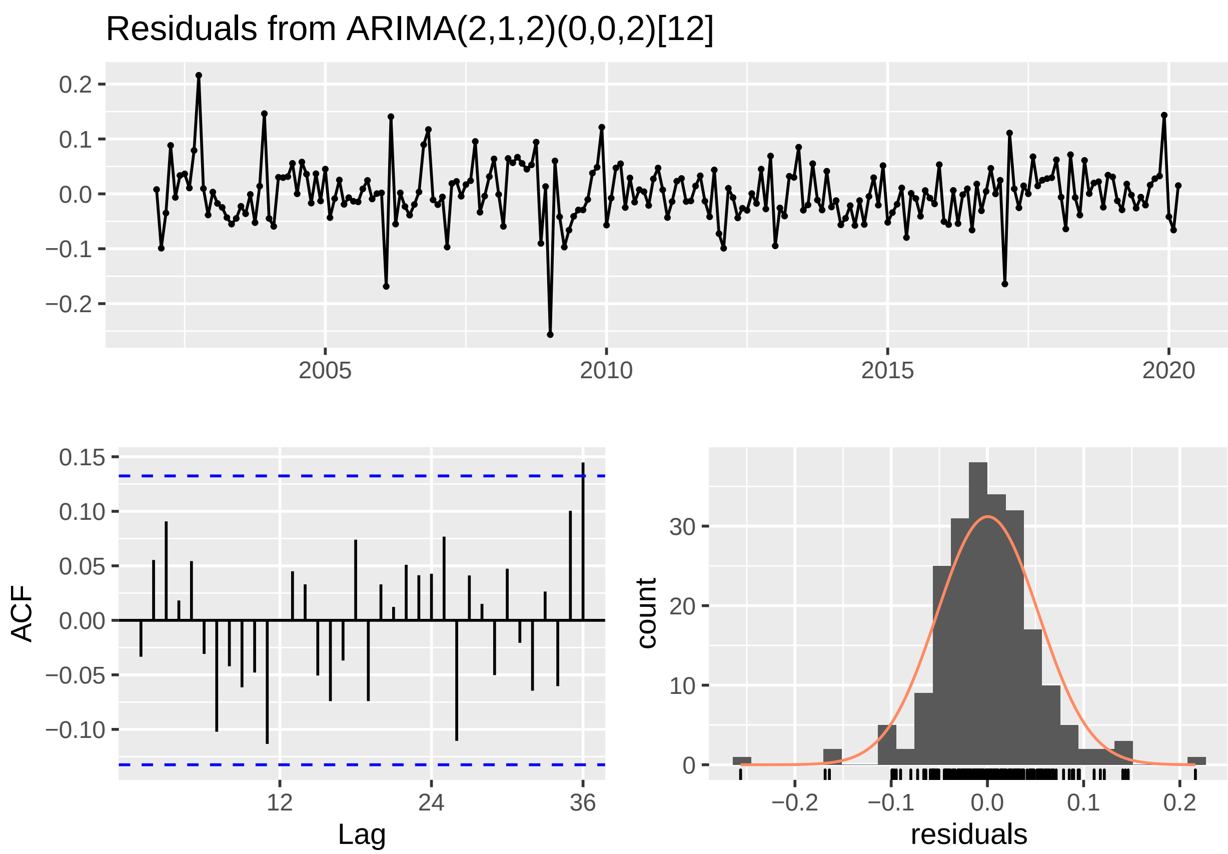

Forecasts from ARIMA(1,1,1)(0,0,2)[12]
ICMS Acumulado previsto (24meses) R$308.36 bi
R$ Milhões
5000
10000
15000
2005
2010
2015
2020



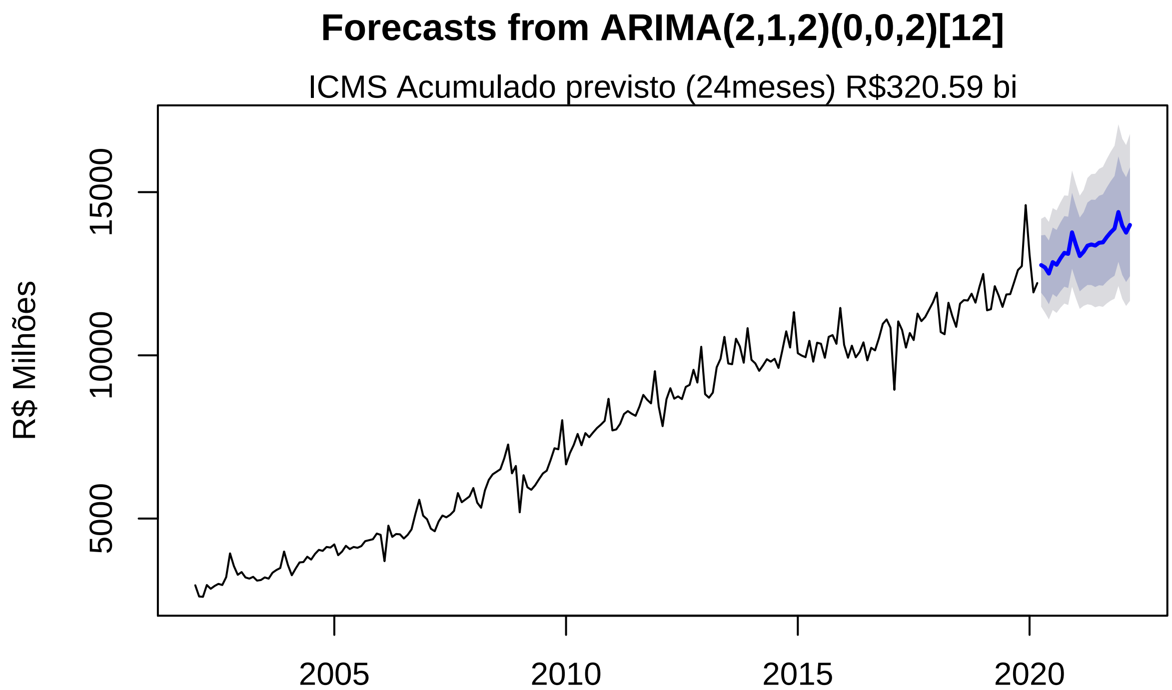

```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(1,1,2)(0,0,2)[12]
## Q* = 20.845, df = 19, p-value = 0.3454
##
## Model df: 5.   Total lags used: 24
```



Forecasts from ARIMA(1,1,2)(0,0,2)[12]
ICMS Acumulado previsto (24meses) R$308.08 bi
R$ Milhões
5000
10000
15000
2005
2010
2015
2020




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(2,1,2)(0,0,2)[12]
## Q* = 18.344, df = 18, p-value = 0.4332
##
## Model df: 6.   Total lags used: 24
```




```
comparacao %>% knitr::kable()
```

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Modelo Base	70.127717	402.8316	282.4730	0.8373410	3.828772	0.4650301	-0.1005522
SARIMA(0,1,1)(0,0,2)[12]	65.139053	403.4973	284.0200	0.7646784	3.862414	0.4675768	-0.0569902
SARIMA(1,1,1)(0,0,2)[12]	70.322638	402.9299	282.8368	0.8404188	3.832961	0.4656289	-0.0971518
SARIMA(1,1,2)(0,0,2)[12]	67.937311	401.9003	280.5545	0.8057799	3.814876	0.4618716	-0.0940527
SARIMA(2,1,2)(0,0,2)[12]	1.810129	395.7659	281.4144	-0.0969885	3.816794	0.4632873	-0.0885626