ECON233 Project

Tho Do, Yang Zhiyao, Soh Zhe Hong

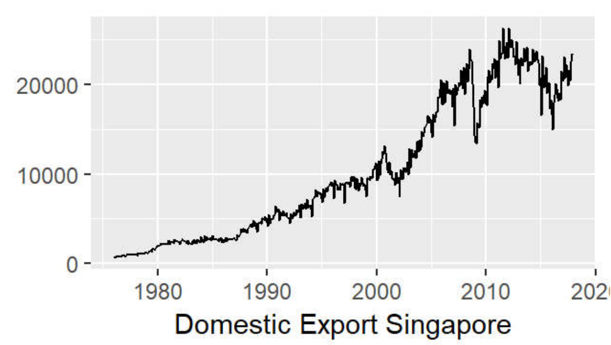
April 1, 2019

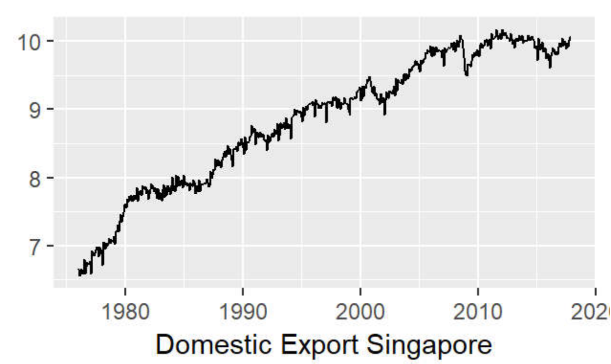
We investigate the components of Singapore Domestic Export time series from Jan 1976 to Dec 2017. Then, we attempt to forecast it for the period of Jan 2018-Dec2018.

1. Data Transformation

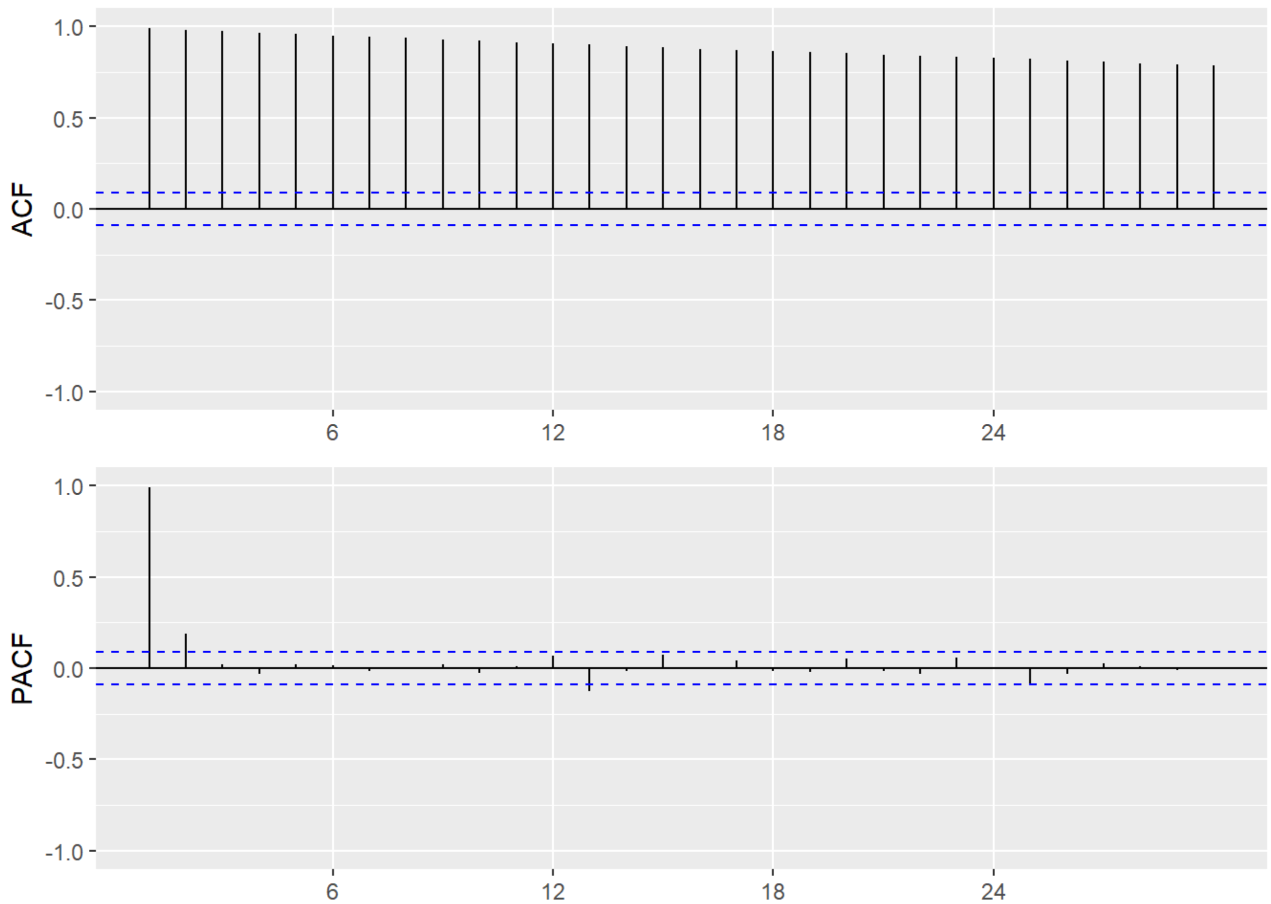
Input data and transform data timeseries object.

```
# transform to log
p1 <- autoplot(dx_ts) + xlab("Domestic Export Singapore") + theme(aspect.ratio=1/2)
y <- ts(log(dx_ts), start = c(year(start_date), month(start_date)),
        end = c(year(end_date), month(end_date)),
        frequency = 12)
p2 <- autoplot(y) + xlab("Domestic Export Singapore") + theme(aspect.ratio=1/2)
grid.arrange(p1,p2, ncol = 2)
```





```
correl(y)
```



Run the ADF and KPSS test to check stationary

```
y %>%  ur.df(type="trend", selectlags="AIC") %>% summary()
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt + z.diff.lag)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.276925 -0.055782  0.002077  0.056677  0.218115
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.3688933  0.1146191   3.218  0.00137 **
## z.lag.1     -0.0483819  0.0158584  -3.051  0.00240 **
## tt           0.0002769  0.0001066   2.597  0.00969 **
## z.diff.lag  -0.4719049  0.0392522 -12.022  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0857 on 498 degrees of freedom
## Multiple R-squared:  0.2612, Adjusted R-squared:  0.2568
## F-statistic:  58.7 on 3 and 498 DF,  p-value: < 2.2e-16
##
##
## Value of test-statistic is: -3.0509 6.2188 5.7033
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -3.96 -3.41 -3.12
## phi2  6.09  4.68  4.03
## phi3  8.27  6.25  5.34
```

```
y %>% ur.kpss(type="tau", lags="short") %>% summary()
```

```
##
## #######################
## # KPSS Unit Root Test #
## #######################
##
## Test is of type: tau with 5 lags.
##
## Value of test-statistic is: 1.2152
##
## Critical value for a significance level of:
##                 10pct  5pct 2.5pct  1pct
## critical values 0.119 0.146  0.176 0.216
```

The series is not stationary from both tests. Hence we proceed to first difference it. We run similar tests on differenced series to ensure that it is now stationary.

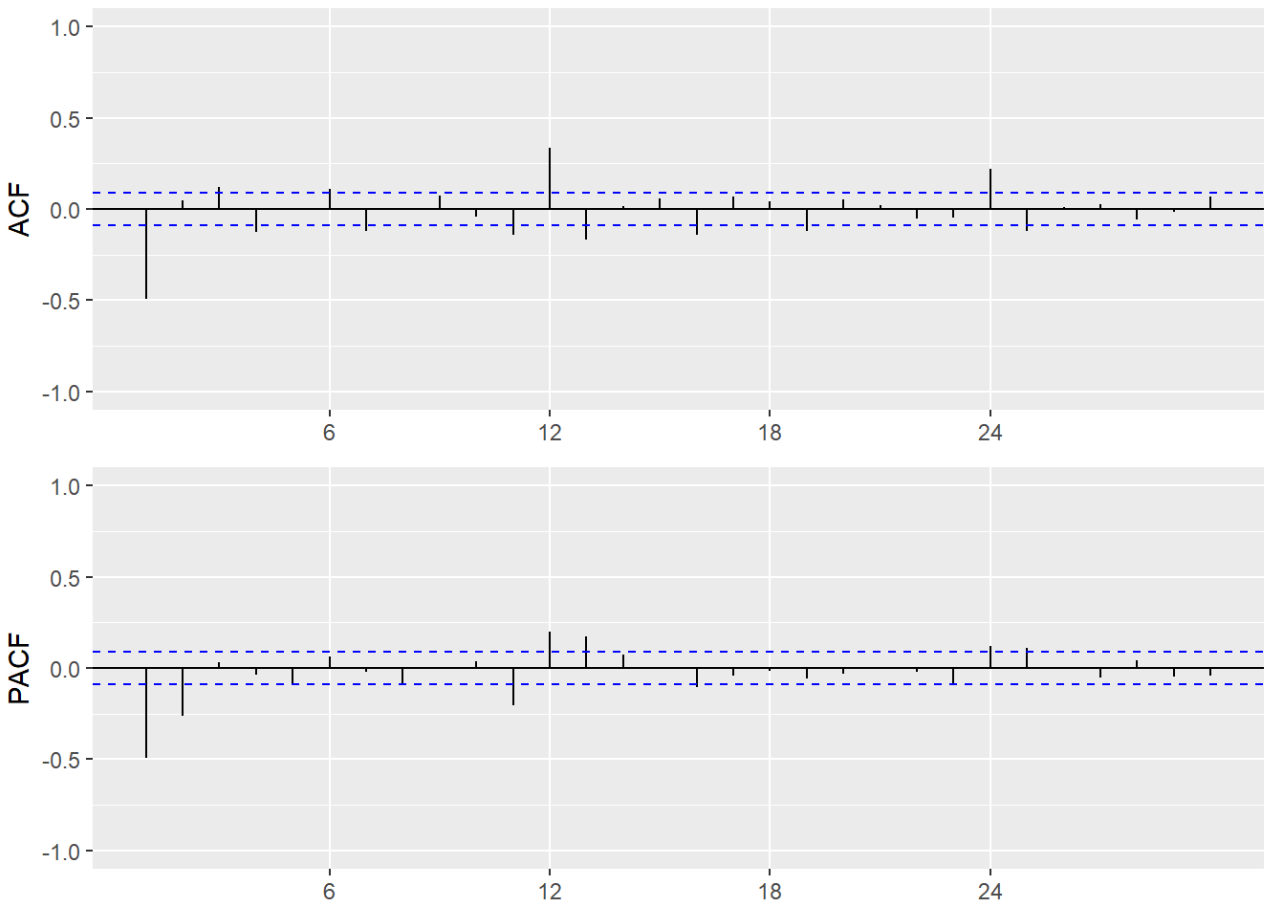
```
diff(y) %>% ur.df(type = "drift", selectlags = "AIC") %>% summary()
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression drift
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + z.diff.lag)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.33159 -0.04911  0.00036  0.05256  0.21638
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.013074   0.003772   3.467 0.000573 ***
## z.lag.1     -1.884868   0.074702 -25.232  < 2e-16 ***
## z.diff.lag   0.261043   0.043184   6.045 2.93e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08364 on 498 degrees of freedom
## Multiple R-squared:  0.7646, Adjusted R-squared:  0.7636
## F-statistic: 808.7 on 2 and 498 DF,  p-value: < 2.2e-16
##
##
## Value of test-statistic is: -25.2318 318.3236
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau2 -3.43 -2.86 -2.57
## phi1  6.43  4.59  3.78
```

```
diff(y) %>% ur.kpss(type ="mu", lags = "short") %>% summary()
```

```
##
## #######################
## # KPSS Unit Root Test #
## #######################
##
## Test is of type: mu with 5 lags.
##
## Value of test-statistic is: 0.2257
##
## Critical value for a significance level of:
##                 10pct  5pct 2.5pct  1pct
## critical values 0.347 0.463  0.574 0.739
```

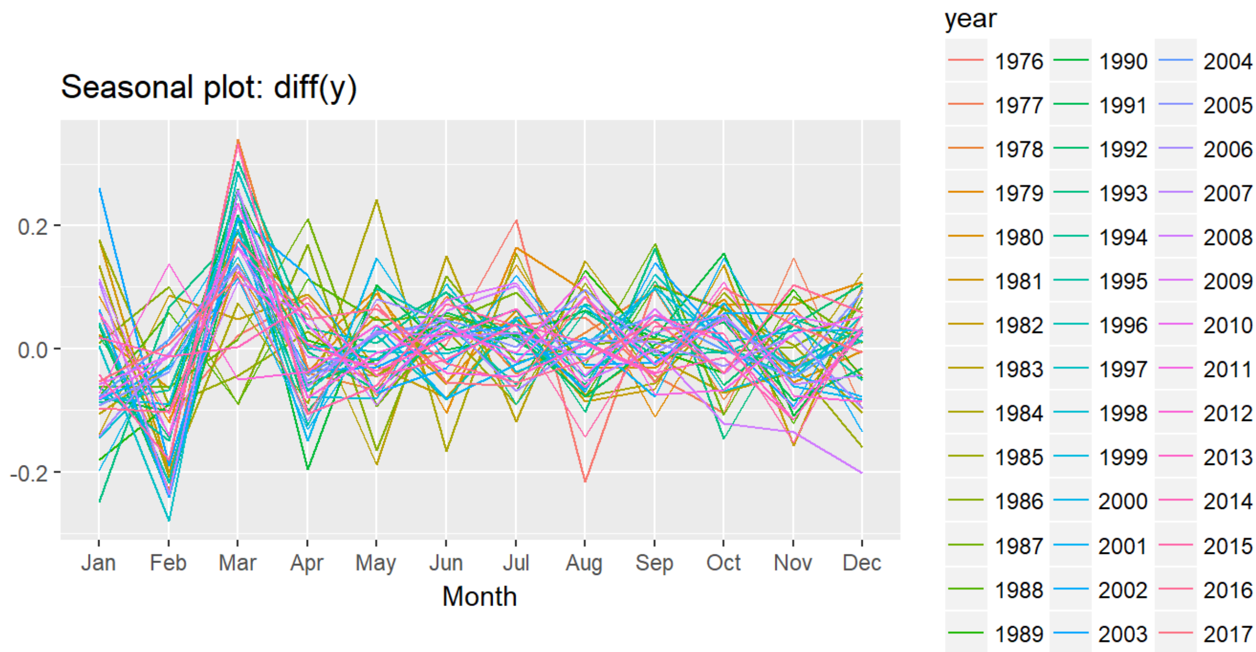
```
correl(diff(y))
```



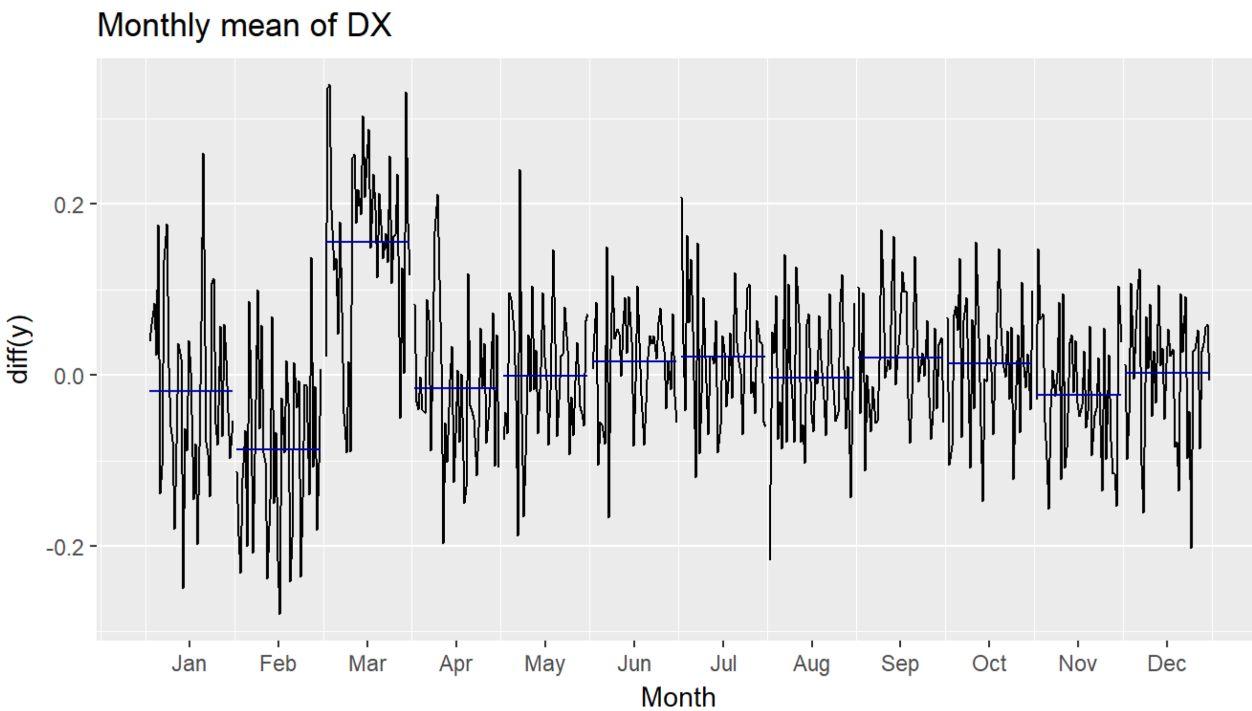
PACF, we see 2 significant spikes at lag 1 and 2. Hence, we will fit an AR 2 component in our model.

We investigate seasonality to see whether it exists

```
ggseasonplot(diff(y)) + theme(aspect.ratio = 1/2)
```




```
ggsubseriesplot(diff(y)) + ggtitle("Monthly mean of D ") + theme(aspect.ratio = 1/2) #
a seasonality component on March
```



There are seasonality at March every year. This suggests us to try either SARIMA or ARIMA with seasonal dummies.

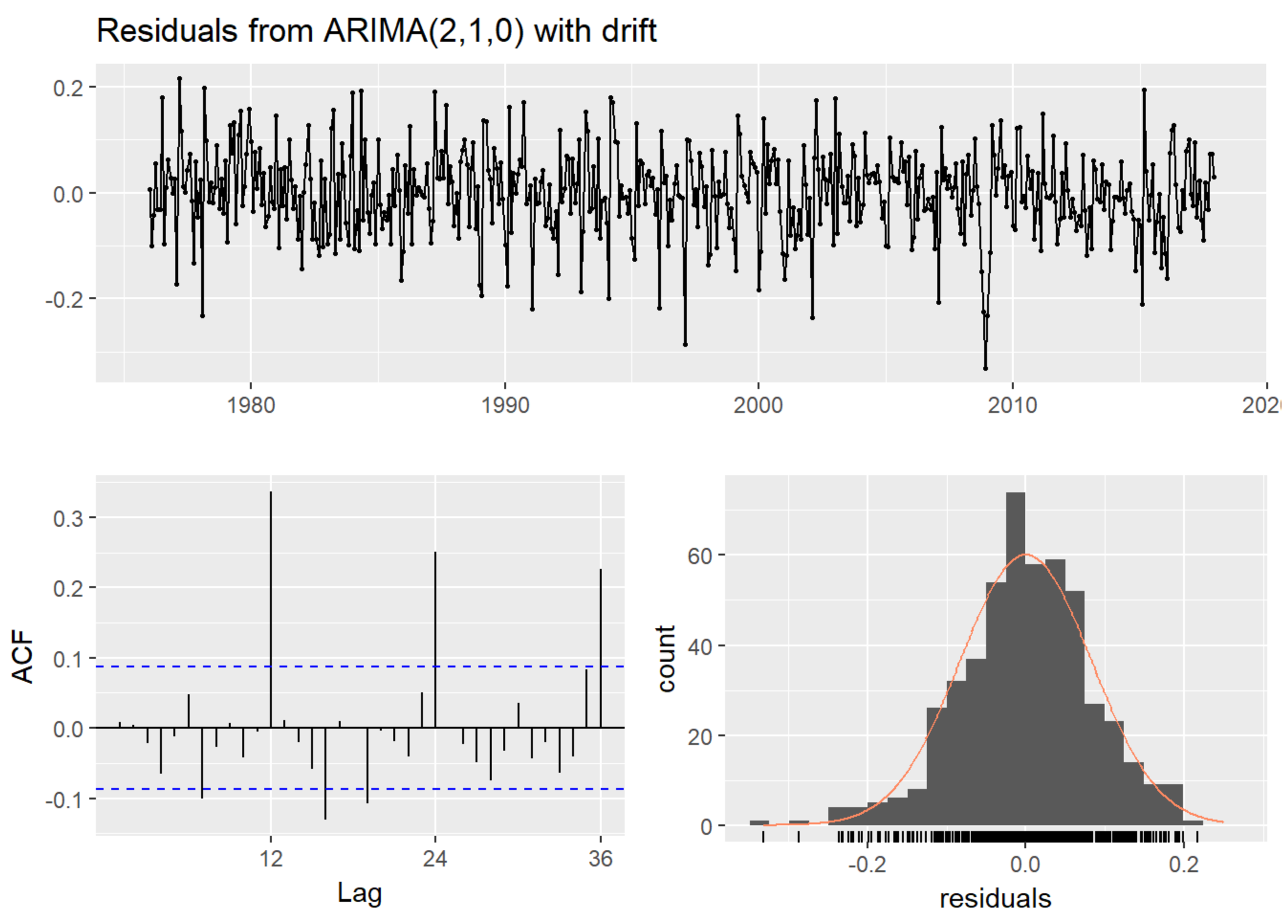
2. Model Selection

We run different models and pick the one with smallest BIC.

```
fit1 <- Arima(y, order=c(2,1,0),
              include.constant = T,
              lambda= U  ) # box-cox transformation, lambda=0 is log-transformatio
summary(fit1)
```

```
## Series: y
## ARIMA(2,1,0) with drift
##
## Coefficients:
##           ar1      ar2   drift
##       -0.6233  -0.2609  0.0069
## s.e.   0.0431   0.0430  0.0020
##
## sigma 2 estimated as 0.006991:  log likelihood=535.8
## AIC=-1063.6   AICc=-1063.52    IC=-1046.72
##
## Training set error measures:
##                         ME       RMSE         MAE           MPE      MAPE
## Training set -0.0001155082 0.08327948 0.06440026 0.0007738744 0.7423076
##                   MASE        ACF1
## Training set 0.4594836 0.008790464
```

```
checkresiduals(fit1)
```

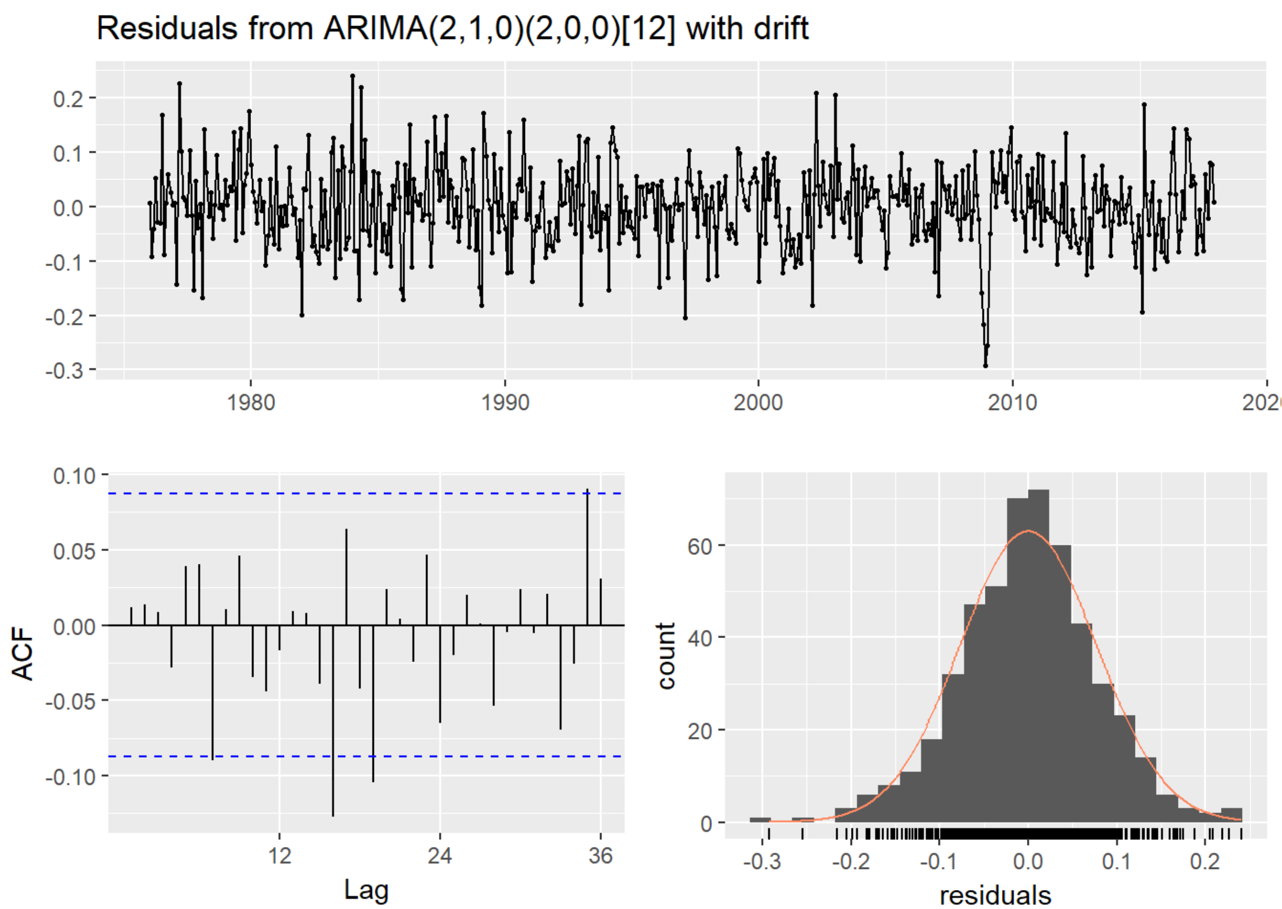


```
##
##   jung- ox test
##
## data:  Residuals from ARIMA(2,1,0) with drift
## Q* = 122, df = 21, p-value = 3.331e-16
##
## Model df: 3.   Total lags used: 24
```

```
fit2 <- Arima(y, order=c(2,1,0),
              seasonal=c(2,0,0),
              include.constant = T,
              lambda= U  ) # box-cox transformation, lambda=0 is log-transformation
summary(fit2)
```

```
## Series: y
## ARIMA(2,1,0)(2,0,0) 12  with drift
##
## Coefficients:
##           ar1      ar2    sar1    sar2   drift
##       -0.6336  -0.2474  0.2818  0.1647  0.0070
## s.e.   0.0434   0.0433  0.0440  0.0453  0.0032
##
## sigma 2 estimated as 0.006035:  log likelihood=572.75
## AIC=-1133.49   AICc=-1133.32    IC=-1108.17
##
## Training set error measures:
##                          ME       RMSE        MAE          MPE     MAPE
## Training set -0.0002196658 0.07721942 0.05950216 -0.002752762 0.685622
##                    MASE       ACF1
## Training set 0.4245367 0.01154626
```

```
checkresiduals(fit2)
```

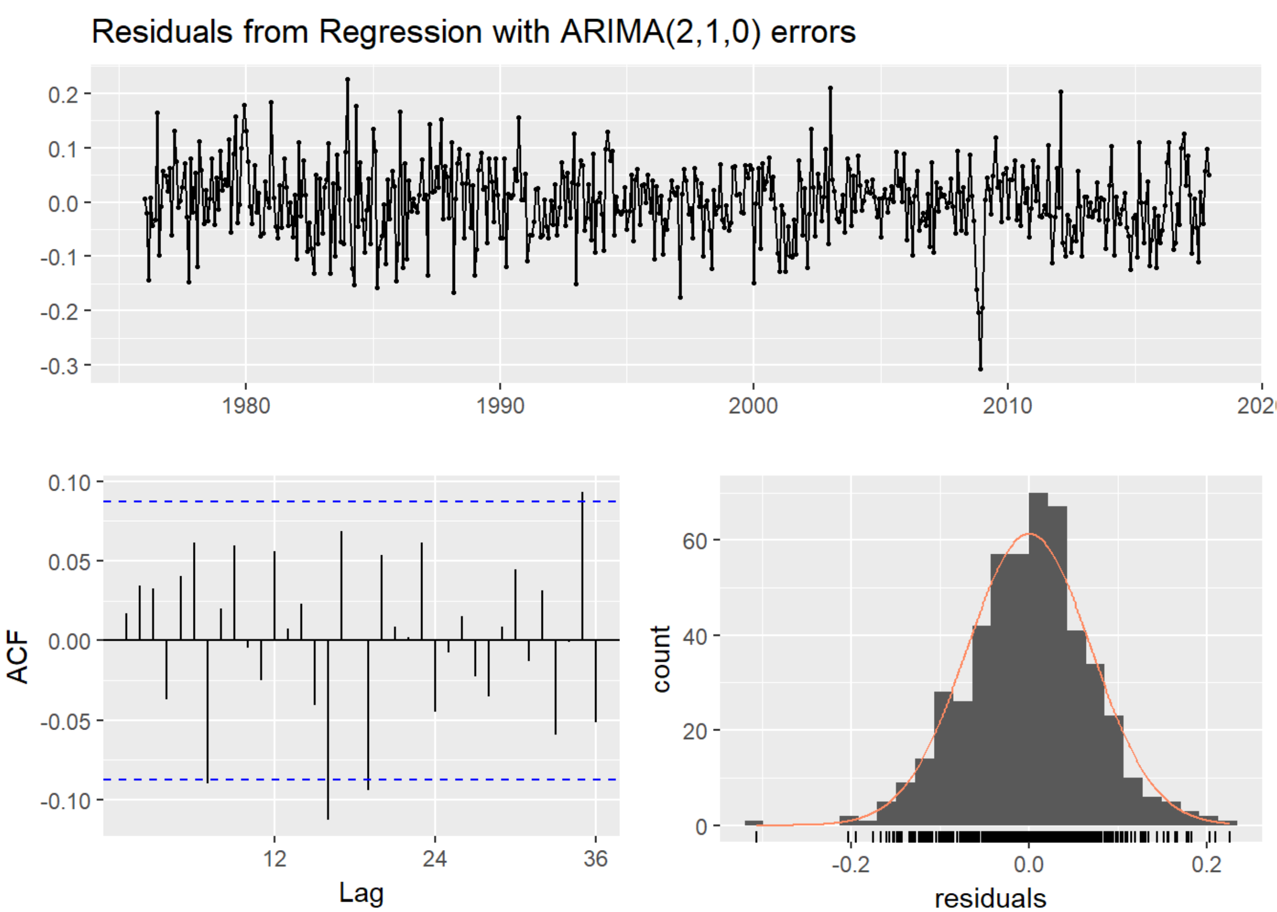



```
##
##   jung- ox test
##
## data:  Residuals from ARIMA(2,1,0)(2,0,0) 12  with drift
## Q* = 31.478, df = 19, p-value = 0.03575
##
## Model df: 5.   Total lags used: 24
```

```
fit3 <- Arima(y, order=c(2,1,0),
              xreg=seasonalvars,
              include.constant = T,
              lambda= U  ) # box-cox transformation, lambda=0 is log-transformation
summary(fit3)
```

```
## Series: y
## Regression with ARIMA(2,1,0) errors
##
## Coefficients:
##           ar1      ar2   drift       an      Feb     Mar     Apr      May
##       -0.6343  -0.2338  0.0067  -0.0248  -0.1188  0.0301  0.0077  -0.0001
## s.e.   0.0434   0.0435  0.0017   0.0129   0.0124  0.0129  0.0139   0.0139
##           un      ul     Aug     Sep      ct      ov
##       0.0093  0.0241  0.0144  0.0277  0.0345  0.0052
## s.e.  0.0140  0.0139  0.0138  0.0128  0.0123  0.0128
##
## sigma 2 estimated as 0.00504:  log likelihood=623.69
## AIC=-1217.38   AICc=-1216.4    IC=-1154.07
##
## Training set error measures:
##                         ME       RMSE        MAE         MPE      MAPE
## Training set -6.788121e-05 0.06992842 0.05434589 0.002762279 0.6291447
##                   MASE       ACF1
## Training set 0.3877477 0.01711578
```

```
checkresiduals(fit3)
```

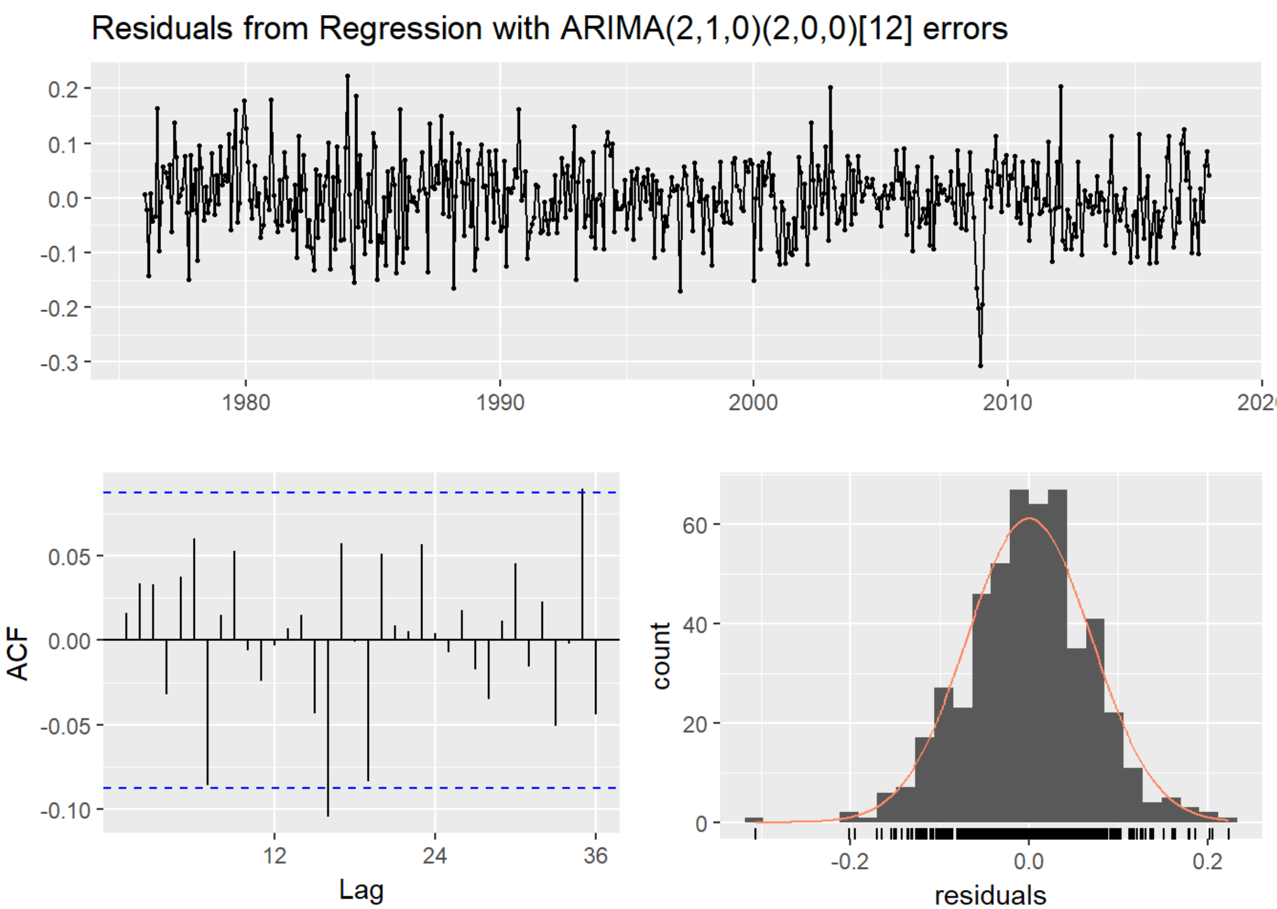


```
##
##   jung- ox test
##
## data:  Residuals from Regression with ARIMA(2,1,0) errors
## Q* = 32.377, df = 10, p-value = 0.0003465
##
## Model df: 14.   Total lags used: 24
```

```
fit4 <- Arima(y, order=c(2,1,0),
              seasonal=c(2,0,0),
              xreg=seasonalvars,
              include.constant = T,
              lambda= U  ) # box-cox transformation, lambda=0 is log-transformation
summary(fit4)
```

```
## Series: y
## Regression with ARIMA(2,1,0)(2,0,0) 12  errors
##
## Coefficients:
##           ar1      ar2    sar1     sar2   drift       an      Feb     Mar
##       -0.6292  -0.2295  0.0614  -0.0521  0.0067  -0.0247  -0.1186  0.0301
## s.e.   0.0436   0.0436  0.0452   0.0460  0.0017   0.0130   0.0125  0.0130
##          Apr      May      un      ul     Aug     Sep      ct      ov
##       0.0077  -0.0002  0.0092  0.0241  0.0144  0.0277  0.0348  0.0051
## s.e.  0.0140   0.0141  0.0142  0.0140  0.0139  0.0129  0.0124  0.0129
##
## sigma 2 estimated as 0.00503:  log likelihood=625.16
## AIC=-1216.31   AICc=-1215.05    IC=-1144.56
##
## Training set error measures:
##                         ME       RMSE        MAE         MPE      MAPE
## Training set -5.624588e-05 0.06971833 0.05400397 0.002835932 0.6249928
##                   MASE       ACF1
## Training set 0.3853081 0.01586549
```

```
checkresiduals(fit4)
```




```
##
##   jung- ox test
##
## data:  Residuals from Regression with ARIMA(2,1,0)(2,0,0) 12  errors
## Q* = 25.288, df = 8, p-value = 0.001389
##
## Model df: 16.   Total lags used: 24
```

In conclusion, we pick ARIMA 2,1,0 with seasonal dummies as our chosen model since it has lowest BIC compared to other.

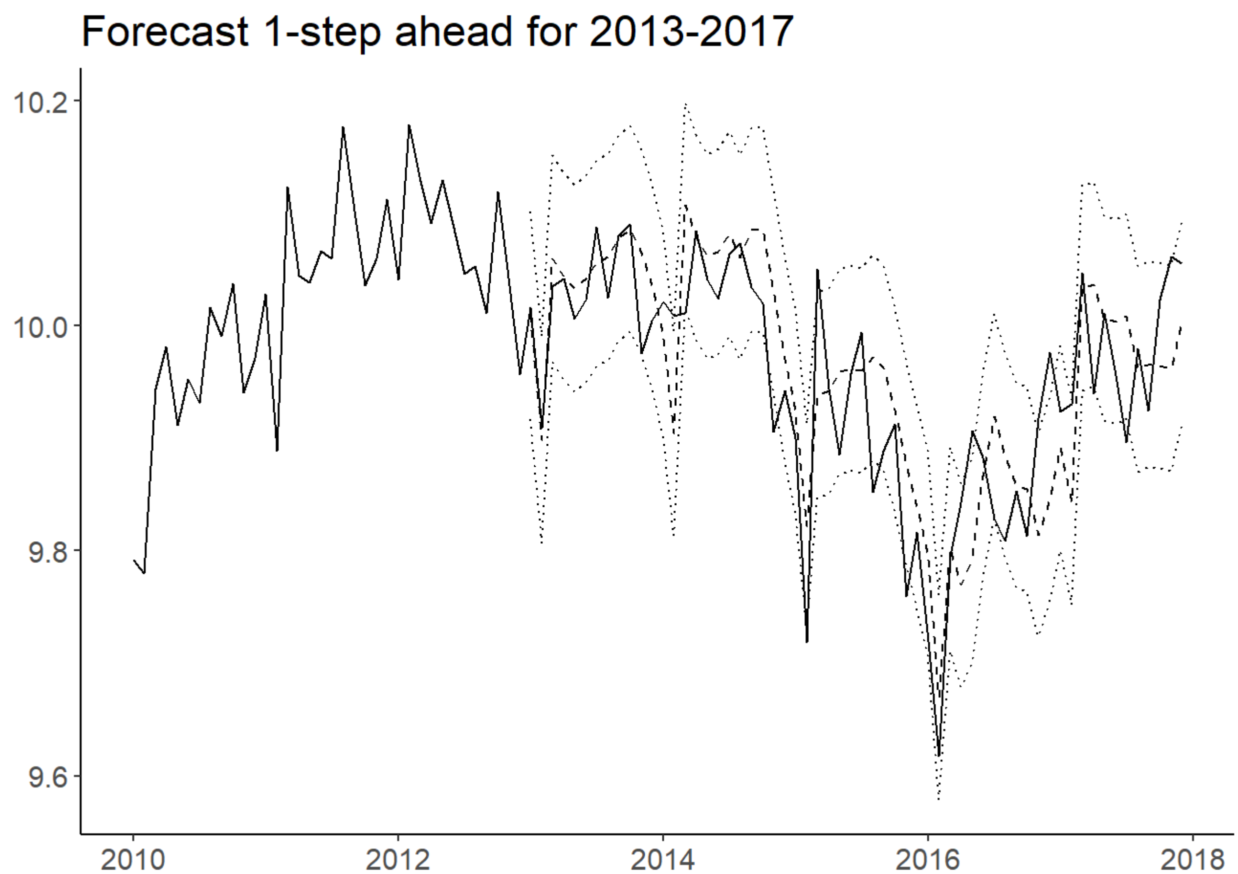
```
fit_chosen <- Arima(y, order=c(2,1,0),
              xreg=seasonalvars,
              include.constant = T,
              lambda= U  ) # box-cox transformation, lambda=0 is log-transformation
summary(fit_chosen)
```

```
## Series: y
## Regression with ARIMA(2,1,0) errors
##
## Coefficients:
##           ar1      ar2   drift       an      Feb     Mar     Apr      May
##       -0.6343  -0.2338  0.0067  -0.0248  -0.1188  0.0301  0.0077  -0.0001
## s.e.   0.0434   0.0435  0.0017   0.0129   0.0124  0.0129  0.0139   0.0139
##           un      ul     Aug     Sep      ct      ov
##       0.0093  0.0241  0.0144  0.0277  0.0345  0.0052
## s.e.  0.0140  0.0139  0.0138  0.0128  0.0123  0.0128
##
## sigma 2 estimated as 0.00504:  log likelihood=623.69
## AIC=-1217.38   AICc=-1216.4    IC=-1154.07
##
## Training set error measures:
##                         ME       RMSE        MAE         MPE      MAPE
## Training set -6.788121e-05 0.06992842 0.05434589 0.002762279 0.6291447
##                    MASE       ACF1
## Training set 0.3877477 0.01711578
```

3. Forecasting 2013- 2017 1 step ahead in sample

We test this model on a somewhat validation set to see its performance. Note that this is overfitting since the chosen model is built on this data.

```
no_fcst_mths <- length(window(y, start = c(2013,1), end = c(2017,12)))
no_remain_months <- length(y) - no_fcst_mths
fcst1step <-ts(matrix(rep( A,no_fcst_mths*3),ncol=3), start=c(2013,1), frequency=12) #
to store forecasts
colnames(fcst1step) <- c("mean", "lower", "upper")

for (i in 1:no_fcst_mths)

  temp_mdl <- Arima(y 1:(no_remain_months+i-1) ,
                order = c(2,1,0),
                xreg = seasonalvars 1:(no_remain_months+i-1), ,
                include.constant = T,
                biasadj = T,
                lambda =  U  )


  temp <- forecast(temp_mdl, h=1, xreg=matrix(seasonalvars (no_remain_months+i):(no_re
main_months+i), ,nrow = 1))
  fcst1step i, <-cbind(temp mean, temp lower ,"80%" , temp upper ,"80%" )



ts.plot <- ts.union(Actual=window(y, start = c(2010,1)), fcst1step)
autoplot(ts.plot) +
  scale_color_manual(values=rep("black", 4)) +
  ylab("") + xlab("") +
  aes(linetype=series) +
  scale_size_manual(values = c(0.5, 2, 0.75)) +
  scale_linetype_manual(values=c("solid", "dashed", rep("dotted",2))) +
  ts_thm + theme(legend.position="none") + ggtitle("Forecast 1-step ahead for 2013-201
7")
```



Evaluate R-s uared and RMSE on this period

```
f1err <- ts.union(Actual=window(y,start=c(2013,1)),Fcst=fcst1step ,1 )
sse <- sum((f1err ,"Actual" -f1err ,"Fcst" ) 2)
sst <- sum((f1err ,"Actual" -mean(f1err ,"Actual" )) 2)
  SR2 <- 1-sse/sst
print(paste0(" ut-of-sample RMSE is ",as.character(round(sqrt(sse/length(f1err)),2))))
```

```
##  1  " ut-of-sample RMSE is 0.05"
```

```
print(paste0(" ut-of-sample R-sqr is ",as.character(round(  SR2,2))))
```

```
##  1  " ut-of-sample R-sqr is 0.61"
```

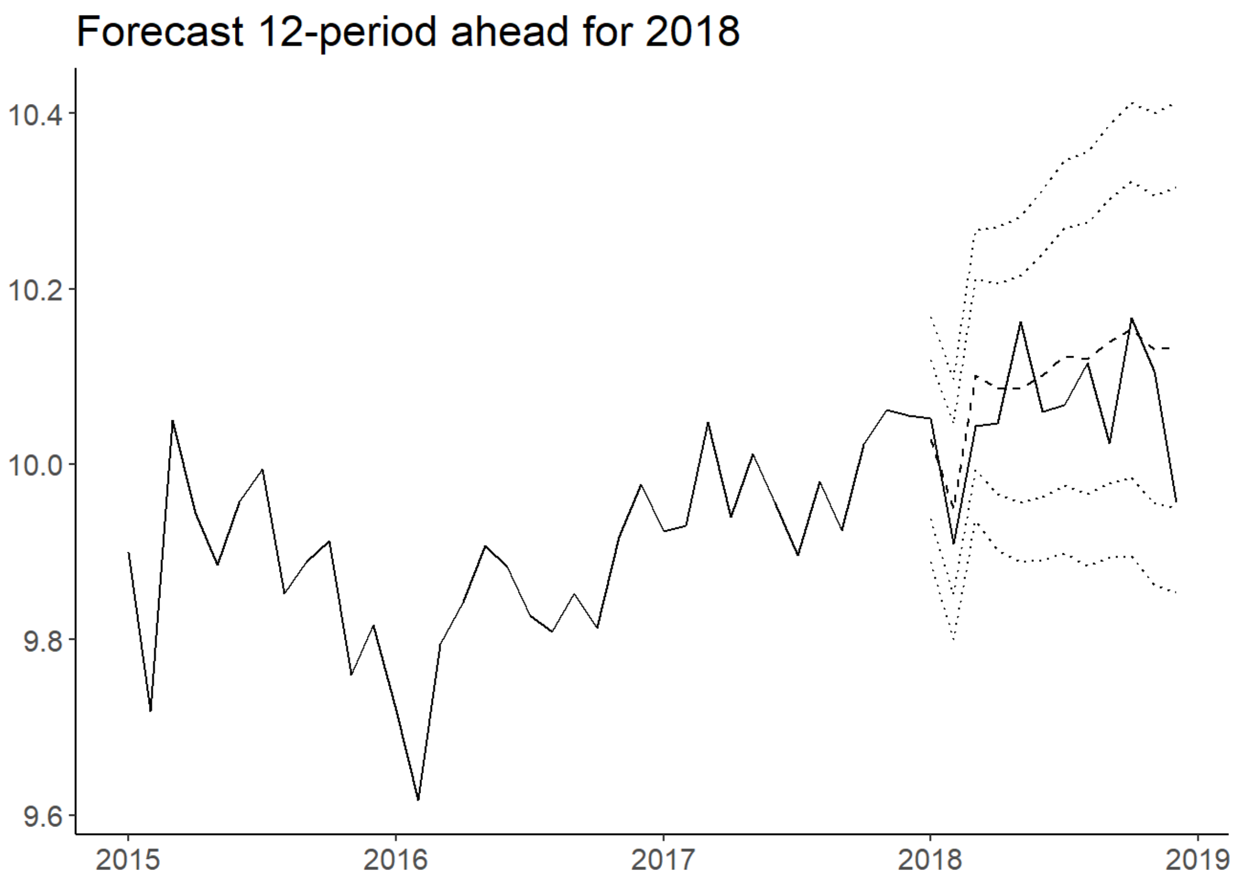
It is not surprising that the forecast is uite good.

. Forecasting 2018 out of sample

Forecast 12 period ahead

```
fcst <- forecast(fit_chosen, h=12, xreg = seasonalvars) # forecast 1 to 28 steps (1 ye
ars x 12 months) ahead
actual_data <- read_csv(paste(dropbox_path,"  EC  233  ","dx_2018.csv",sep=""), col_na
mes = FA SE)
colnames(actual_data) <- c("Date", "D ")
actual_data D  <- log(actual_data D )
dx_actual_ts <- ts(actual_data D , start = c(1976,1), frequency = 12)
dx_os_ts <- window(dx_actual_ts, start = c(2018, 1), end = c(2018,12), frequecy = 12)

autoplot(ts.union(window(dx_actual_ts, start = c(2015,1), end = c(2018,12)),
                  window(fcst mean, start = c(2015,1), end = c(2018,12)),
                  window(fcst lower, start = c(2015,1), end = c(2018,12)),
                  window(fcst upper, start = c(2015,1), end = c(2018,12)))) +
  scale_color_manual(values=rep("black", 6)) +
  ylab("") + xlab("") +
  aes(linetype=series) +
  scale_size_manual(values = c(0.5, 2, 0.75)) +
  scale_linetype_manual(values=c("solid", "dashed", rep("dotted",4))) +
  ts_thm + theme(legend.position="none") + ggtitle("Forecast 12-period ahead for 201
8")
```

```
##  arning in window.default(x, ...): 'start' value not changed

##  arning in window.default(x, ...): 'start' value not changed

##  arning in window.default(x, ...): 'start' value not changed
```



Evaluate forecast on real data for 2018

```
f2err <- ts.union(Actual=window(dx_os_ts,start=c(2018,1)),
                  Fcst=window(fcst mean, start = c(2018,1), end = c(2018,12)))
sse <- sum((f2err ,"Actual" -f2err ,"Fcst" ) 2)
sst <- sum((f2err ,"Actual" -mean(f2err ,"Actual" )) 2)
  SR2 <- 1-sse/sst
print(paste0(" ut-of-sample RMSE is ",as.character(round(sqrt(sse/12),2))))
```

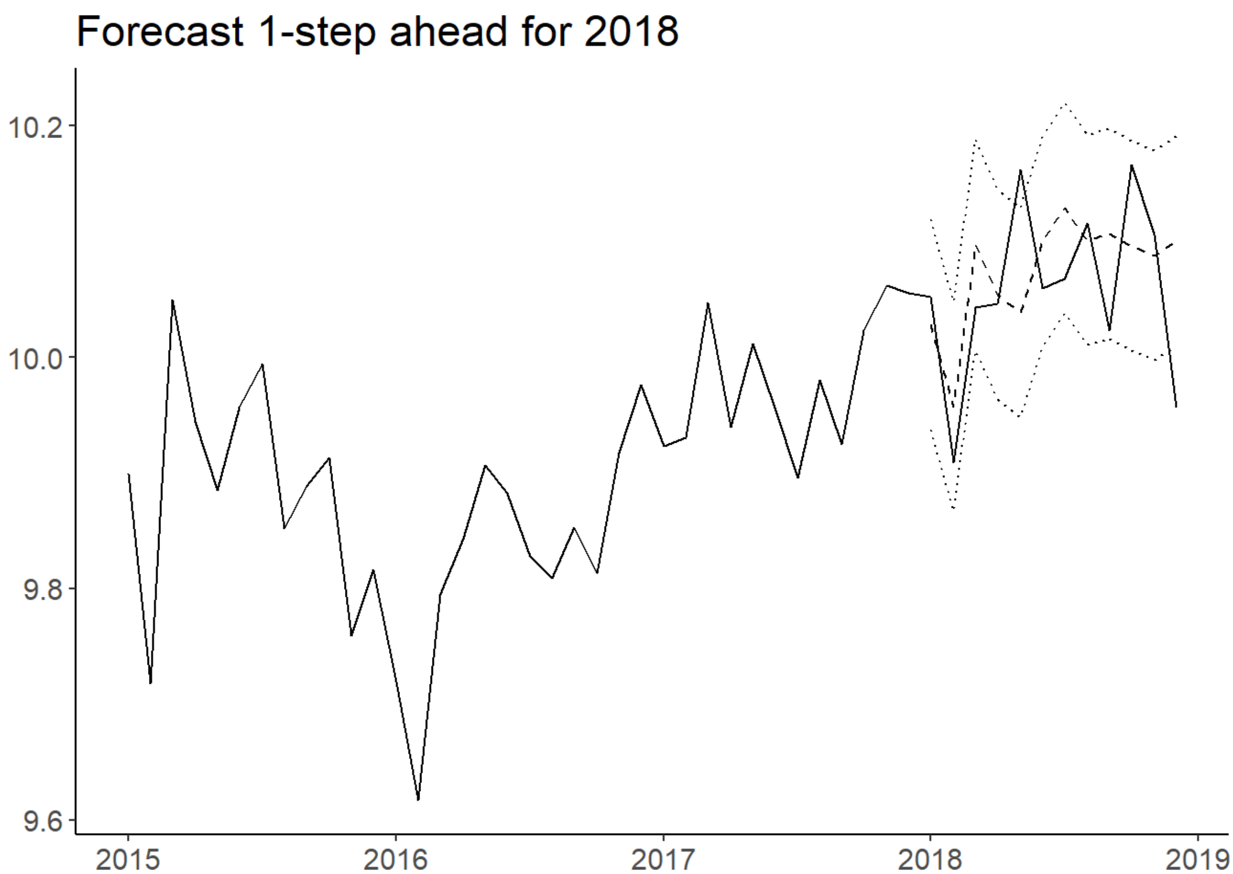
```
##  1  " ut-of-sample RMSE is 0.07"
```

```
print(paste0(" ut-of-sample R-sqr is ",as.character(round(  SR2,2))))
```

```
##  1  " ut-of-sample R-sqr is -0.01"
```

Forecast 1-step ahead for the model during 2018.

```
seasonalvars2 <- seasonaldummy(dx_actual_ts)
no_fcst_mths <- length(window(dx_actual_ts, start = c(2018,1), end = c(2018,12)))
no_remain_months <- length(dx_actual_ts) - no_fcst_mths # remove 2 months of jan and f
eb 2019
fcst1step_3 <-ts(matrix(rep( A,no_fcst_mths*3),ncol=3), start=c(2018,1), frequency=12)
# to store forecasts
colnames(fcst1step_3) <- c("mean", "lower", "upper")

for (i in 1:no_fcst_mths)

  temp_mdl <- Arima(dx_actual_ts 1:(no_remain_months+i-1) ,
                     order = c(2,1,0),
                     xreg = seasonalvars2 1:(no_remain_months+i-1), ,
                     include.constant = T,
                     biasadj = T,
                     lambda =  U  )



  temp <- forecast(temp_mdl, h=1, xreg=matrix(seasonalvars2 (no_remain_months+i):(no_r
emain_months+i), ,nrow = 1),
                   biasadj = T, lambda =  U  )
  fcst1step_3 i, <-cbind(temp mean, temp lower ,"80%" , temp upper ,"80%" )



ts.plot <- ts.union(Actual=window(dx_actual_ts, start = c(2015,1)), fcst1step_3)
autoplot(ts.plot) +
  scale_color_manual(values=rep("black", 4)) +
  ylab("") + xlab("") +
  aes(linetype=series) +
  scale_size_manual(values = c(0.5, 2, 0.75)) +
  scale_linetype_manual(values=c("solid", "dashed", rep("dotted",2))) +
  ts_thm + theme(legend.position="none") + ggtitle("Forecast 1-step ahead for 2018")
```



Evaluate the accuracy and R-s uared for this period

```
f1err_2 <- ts.union(Actual=window(dx_actual_ts,start=c(2018,1)),Fcst=fcst1step_3 ,1 )
sse <- sum((f1err_2 ,"Actual" -f1err_2 ,"Fcst" ) 2)
sst <- sum((f1err_2 ,"Actual" -mean(f1err_2 ,"Actual" )) 2)
  SR2 <- 1-sse/sst
print(paste0(" ut-of-sample RMSE is ",as.character(round(sqrt(sse/length(f1err_2)),
2))))
```

```
##  1  " ut-of-sample RMSE is 0.05"
```

```
print(paste0(" ut-of-sample R-sqr is ",as.character(round(  SR2,2))))
```

```
##  1  " ut-of-sample R-sqr is 0.04"
```