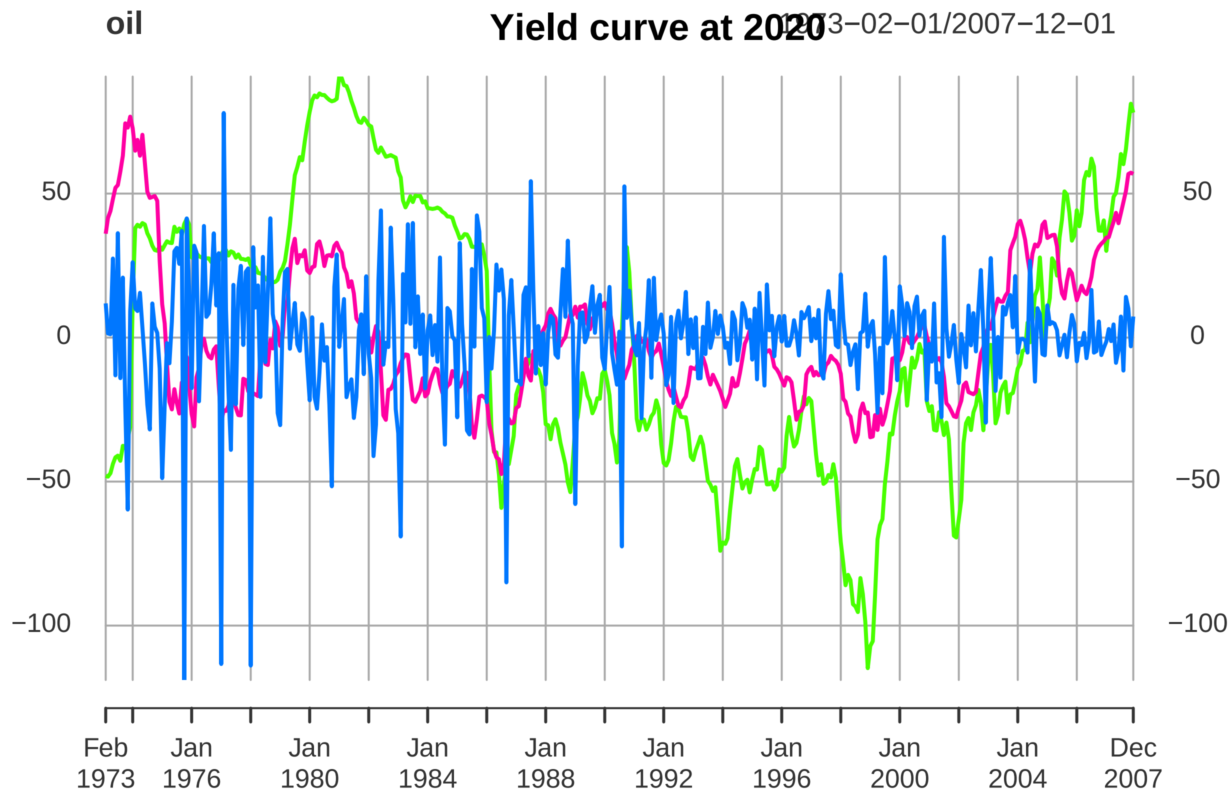
Assignement

Point 1

The time series below represents the monthly time series of:

1. % change in global crude oil production
2. the real price of oil
3. the real economy activity

from 1973:1 to 2007:12.



As we can see from acf its clear signaling the presence of an autocorrelation process. From the partial autocorrelation function we can infer that it's probably first-order autocorrelation since the only significant column is the first one (also the second one, but it has a negative sign).

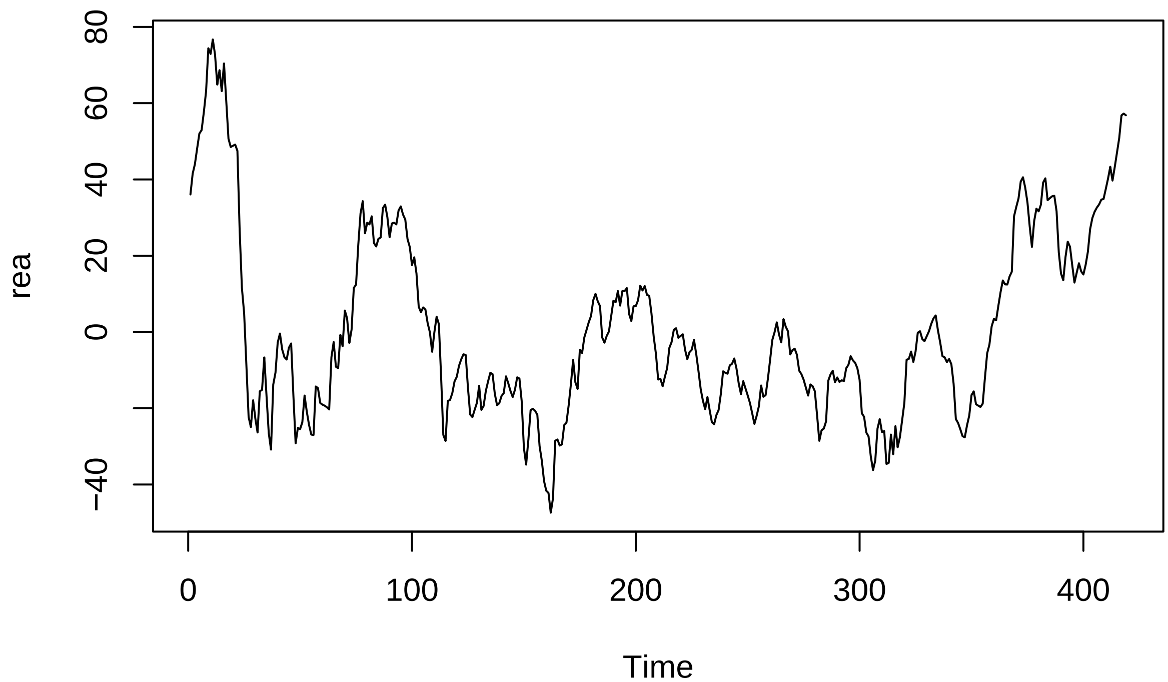
In order to test if the rea is an $I(1)$, we will use an ADF test with a minimun lag =1. We will perform the test specifing four different type of the process:

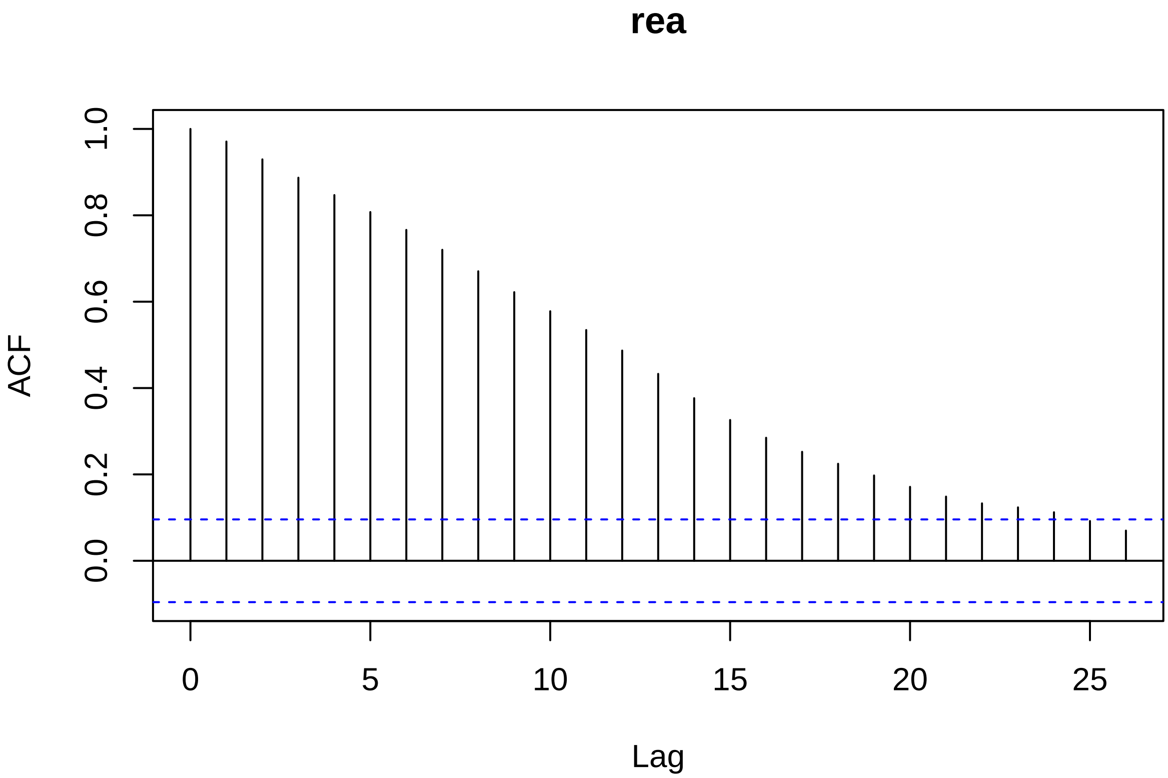
1. No constant, no trend
2. Constant
3. Constant with trend

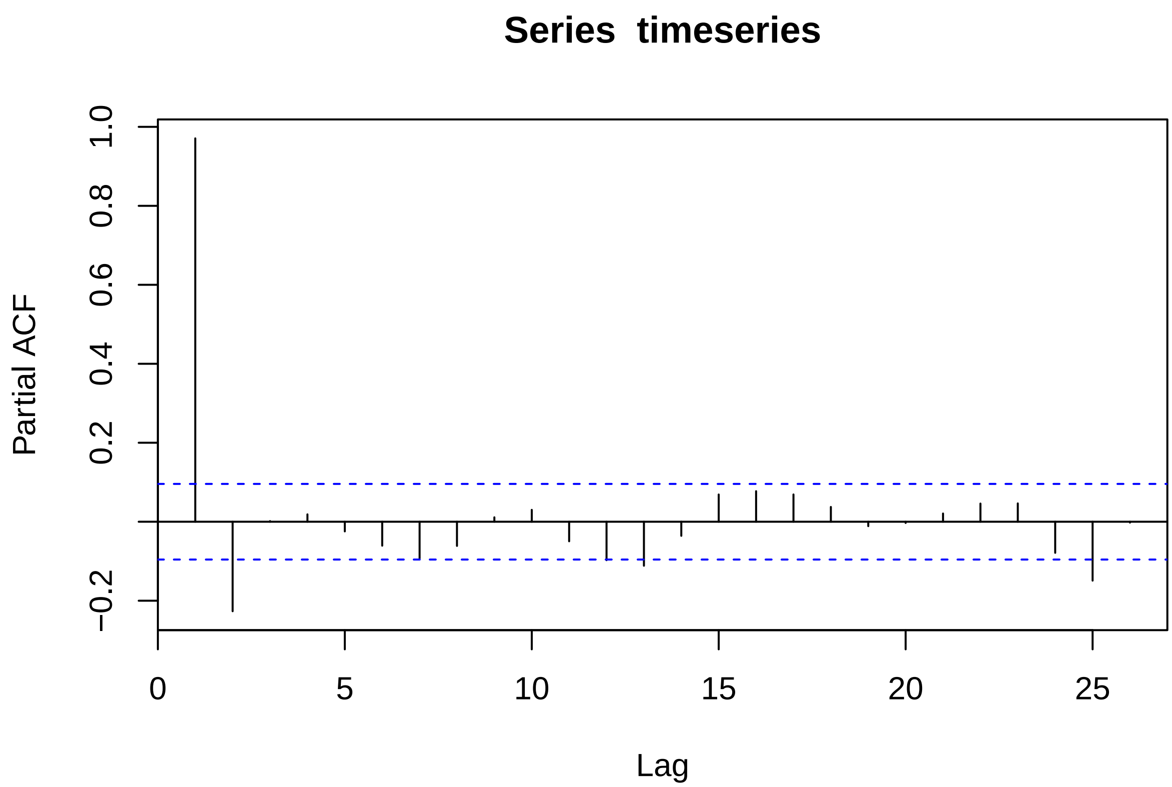
First, we print the first time series graph. We perform the different types of the test with a maximum lag order of 12:

$$rea_t = \alpha + \sigma_1 reat_{t-1} + ... + \sigma_1 2\delta reat_{t-13}$$

The criteria for selection of the lag order is the one which has lower BIC:




rea
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
25
Lag




```
## [1] "Without constant and without time trend"

##
## === Test statistics =========
##              tau1
## statistic -3.0561
##
## === Test critical values ====
##       1pct  5pct 10pct
## tau1 -2.58 -1.95 -1.62
##
## === Combined output =========
## [1] "-3.06 [1]***"

## [1] "Max lag : 12"

## [1] "Lag used: 1"

## [1] "BIC: 2400.42837419751"

## [1] "With constant and without time trend"

##
## === Test statistics =========
##              tau2   phi1
## statistic -3.0642 4.6954
##
## === Test critical values ====
##       1pct  5pct 10pct
```

```
## tau2 -3.44 -2.87 -2.57
## phi1  6.47  4.61  3.79
##
## === Combined output =========
## [1] "-3.06 [1]**"

## [1] "Max lag : 12"

## [1] "Lag used: 1"

## [1] "BIC: 2406.39318640431"

## [1] "With constant and with time trend"

##
## === Test statistics =========
##              tau3   phi2   phi3
## statistic -3.2836 4.5302 6.7945
##
## === Test critical values ====
##       1pct  5pct 10pct
## tau3 -3.98 -3.42 -3.13
## phi2  6.15  4.71  4.05
## phi3  8.34  6.30  5.36
##
## === Combined output =========
## [1] "-3.28 [1]*"

## [1] "Max lag : 12"

## [1] "Lag used: 1"

## [1] "BIC: 2408.28425811698"
```

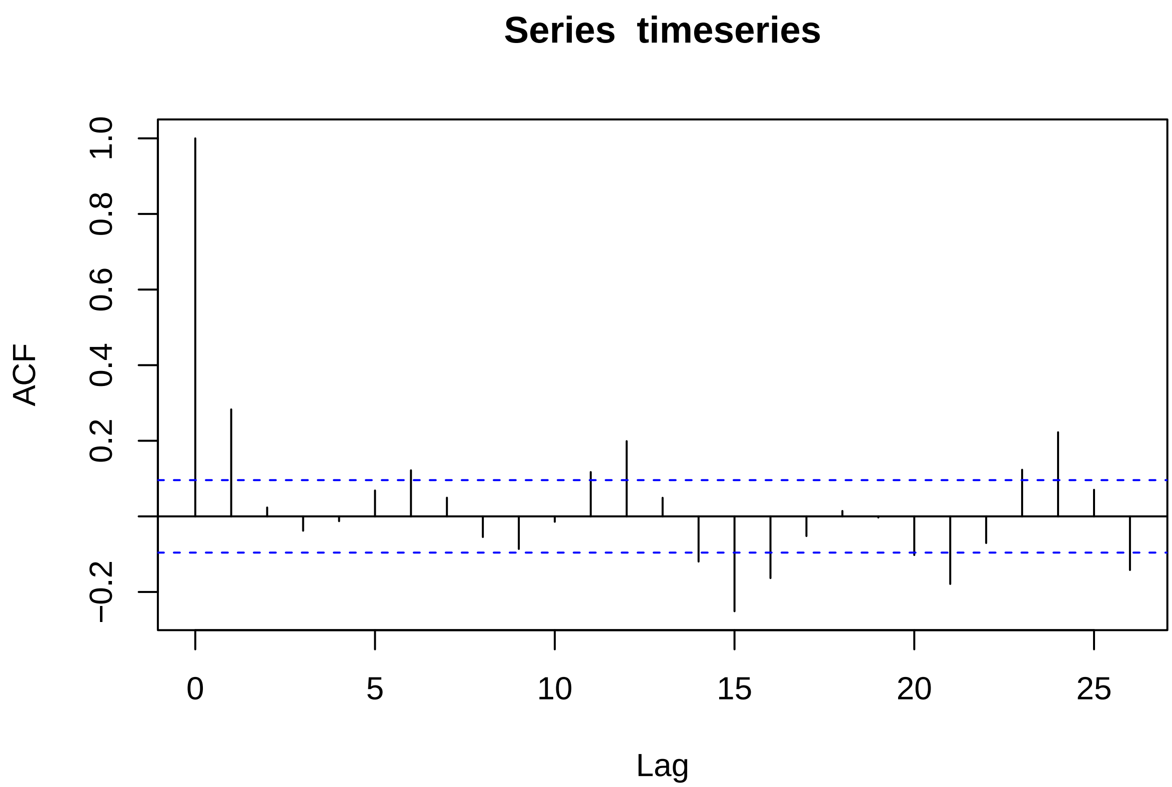
The results of the ADF tests shows that the process is stationary with the simplest specification (without constant and time trend), up to the third significance level (over 1%). However, the other possible specification, which add a constant and then also a time trend present higher p-values, thus the specification we are going to select is the first one. This is consistent with what we should expect, since rea_t is computed as a percentage deviation from the mean (it's basically an indicator of the business cycle).

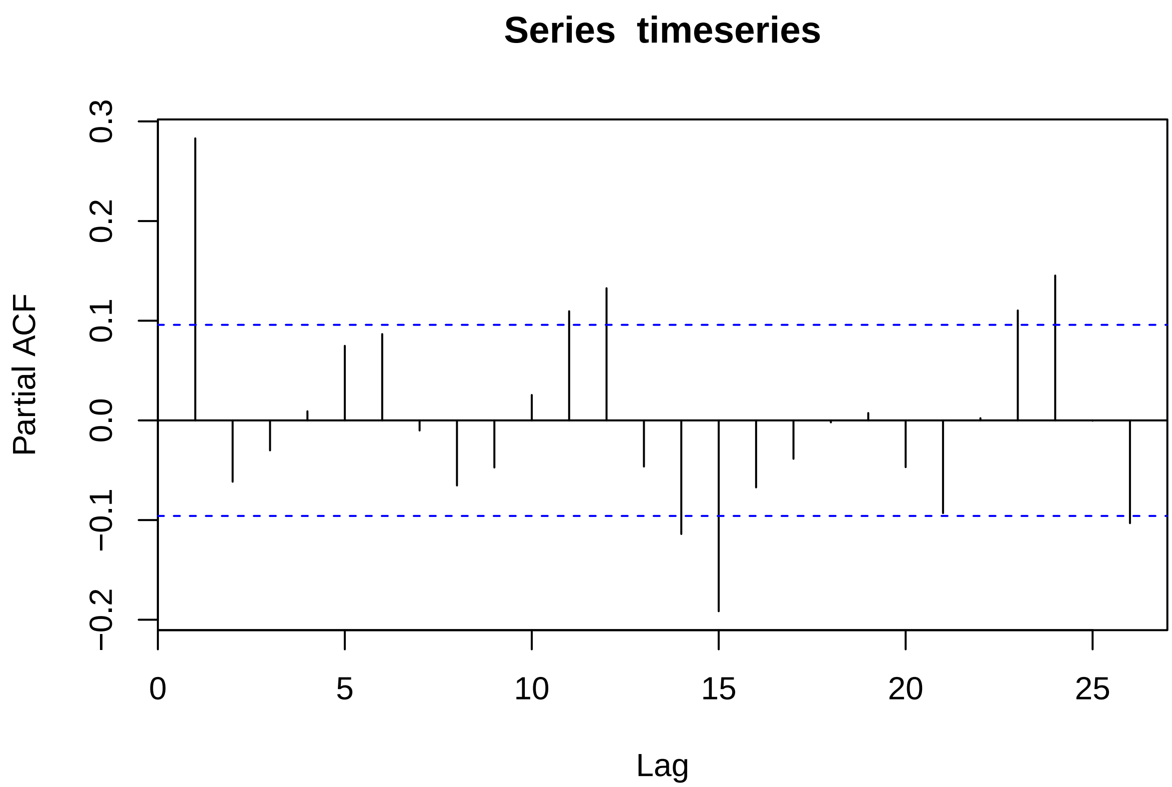
Thus, the specification we select in the end is:

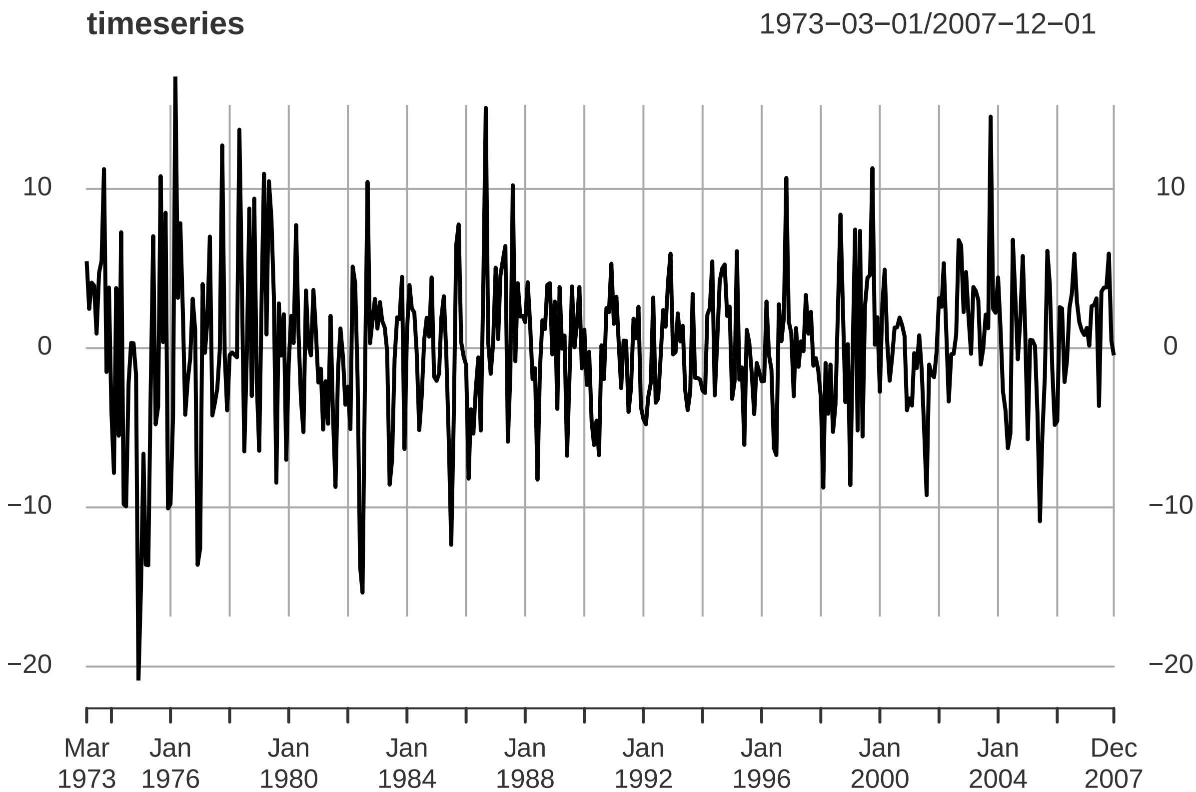
$$\delta rea_t = \delta_1 \Delta reat_{t-1} + ... + \delta_1 2 \Delta reat_{t-13}$$

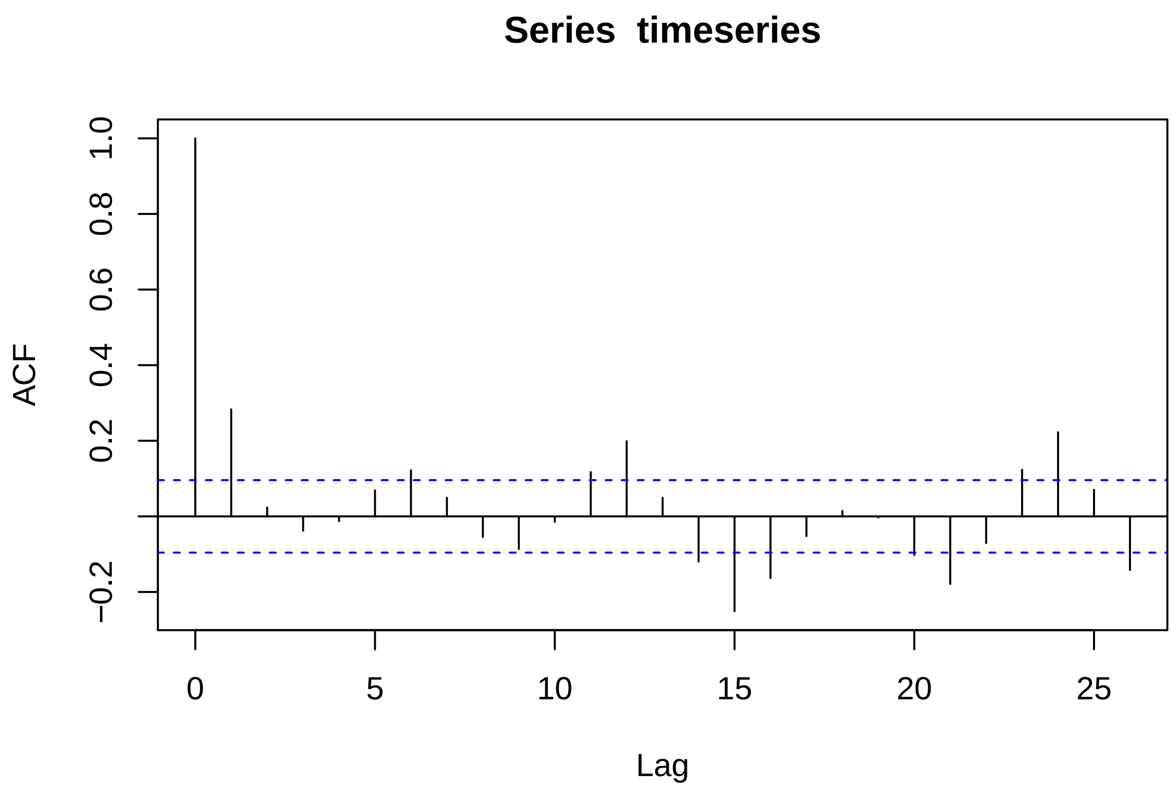
Point 2

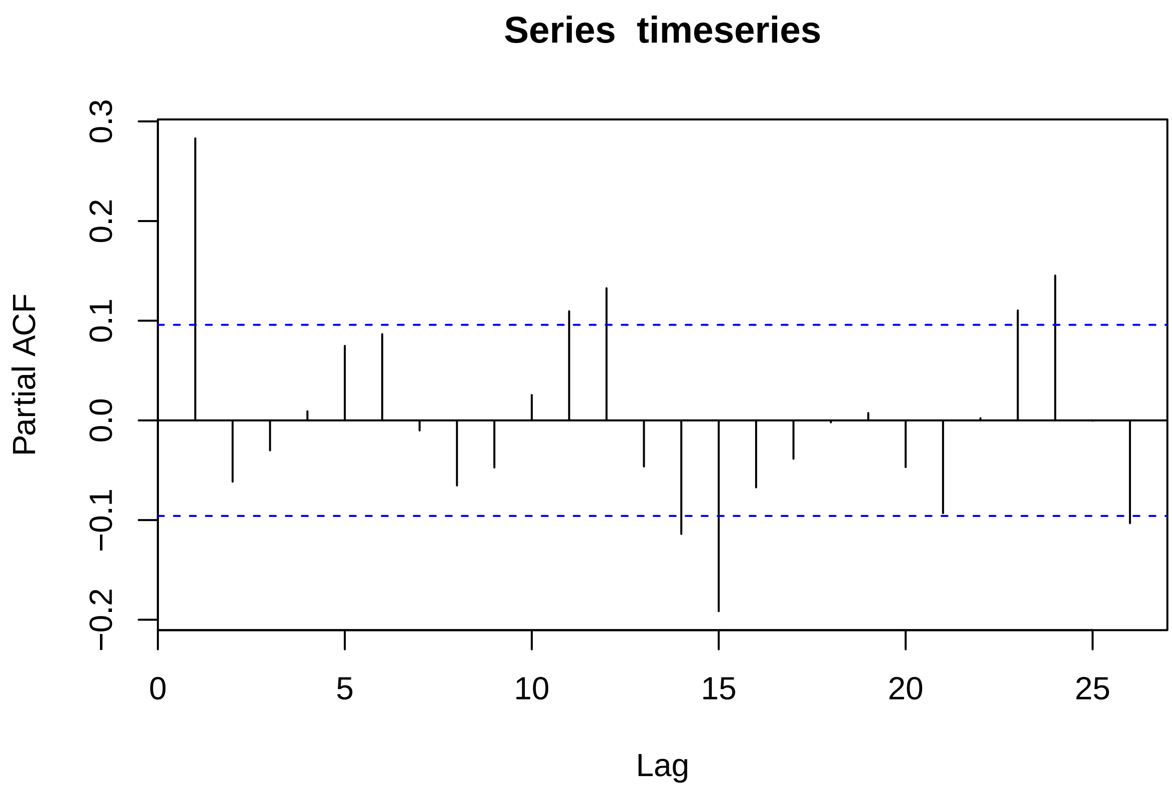
We take the first difference of the timeseries rea and check if it is stationary with an adf test. Before that we print the time series of the first differences, the acf, and the pacf to understand the correct specification for the ADF test.


Series timeseries
ACF
1.0
0.8
0.6
0.4
0.2
−0.2
0
5
10
15
20
25
Lag


Series timeseries
Partial ACF
0.3
0.2
0.1
0.0
−0.1
−0.2
0
5
10
15
20
25
Lag


timeseries
1973−03−01/2007−12−01
10
0
−10
−20
Mar 1973
Jan 1976
Jan 1980
Jan 1984
Jan 1988
Jan 1992
Jan 1996
Jan 2000
Jan 2004
Dec 2007


Series timeseries
ACF
Lag
1.0
0.8
0.6
0.4
0.2
−0.2
0
5
10
15
20
25



The above graphs clearly underline the stationarity of the process, indeed the acf for the $lag > 2$ the partial autcorrelation is not statisticaly different from 0. As for the partial autcorrelation that is statistically different only for some lag>10. From the plot of the time series we can see a mean reverting process, and so I will opt for the specifications with constant and time trend, becouse it is less restricitve. So the test will have the following specifications:

$$\delta rea_t = \sigma_1 \delta reat_{t-1} + ... + \sigma_1 2 \delta reat_{t-13}$$

$$\delta rea_t = \alpha + \sigma_1 \delta reat_{t-1} + ... + \sigma_1 2 \delta reat_{t-13}$$

$$\delta rea_t = \alpha + \beta * t + \sigma_1 \delta reat_{t-1} + ... + \sigma_1 2 \delta reat_{t-13}$$

The test will be performed with all passible four specification, and will be selected the specification with lower adf value.

```
## [1] "Without constant and without time trend"
```

```
##
## === Test statistics =========
##              tau1
## statistic -12.928
##
## === Test critical values ====
##       1pct  5pct 10pct
## tau1 -2.58 -1.95 -1.62
##
## === Combined output =========
## [1] "-12.93 [1]***"
```

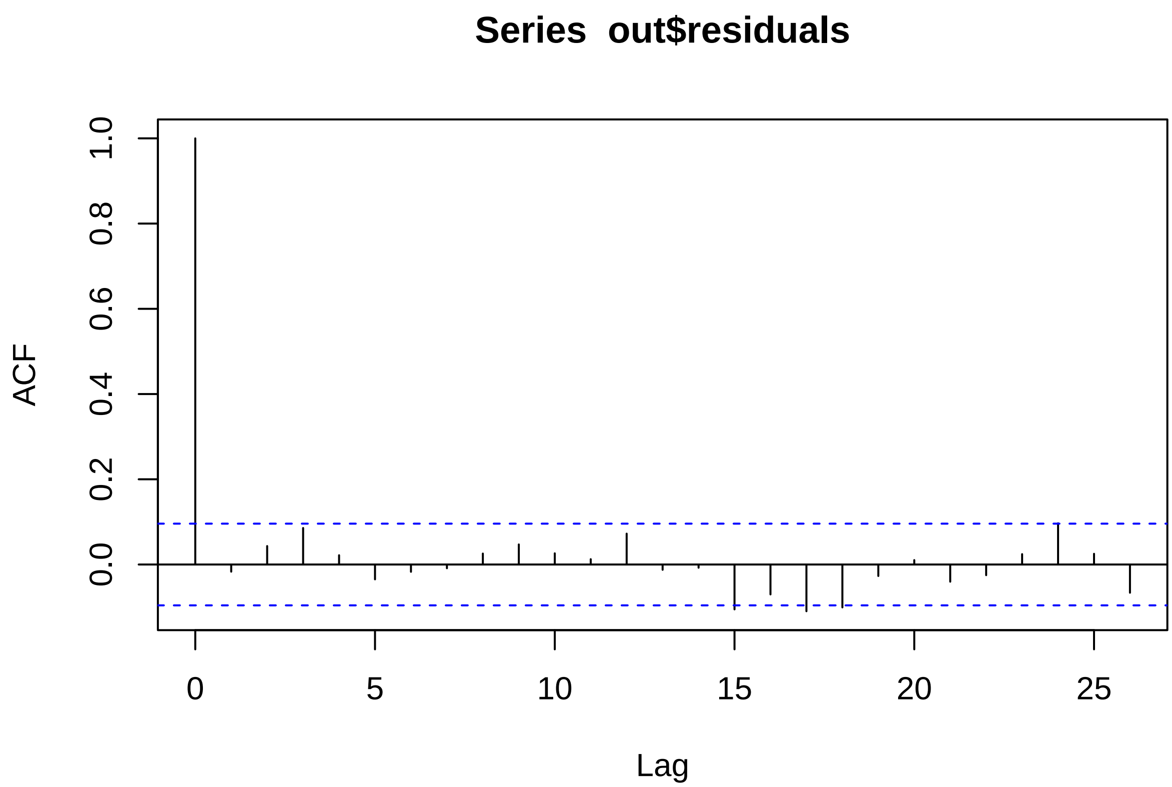
```
## [1] "Max lag : 12"
```

```
## [1] "Lag used: 1"

## [1] "BIC: 2401.03338921915"

## [1] "With constant and without time trend"

##
## === Test statistics =========
##              tau2   phi1
## statistic -12.913 83.372
##
## === Test critical values ====
##       1pct  5pct 10pct
## tau2 -3.44 -2.87 -2.57
## phi1  6.47  4.61  3.79
##
## === Combined output =========
## [1] "-12.91 [1]***"

## [1] "Max lag : 12"

## [1] "Lag used: 1"

## [1] "BIC: 2407.05552268091"

## [1] "With constant and with time trend"

##
## === Test statistics =========
##              tau3   phi2   phi3
## statistic -13.095 57.158 85.736
##
## === Test critical values ====
##       1pct  5pct 10pct
## tau3 -3.98 -3.42 -3.13
## phi2  6.15  4.71  4.05
## phi3  8.34  6.30  5.36
##
## === Combined output =========
## [1] "-13.09 [1]***"

## [1] "Max lag : 12"

## [1] "Lag used: 1"

## [1] "BIC: 2409.43562388553"
```

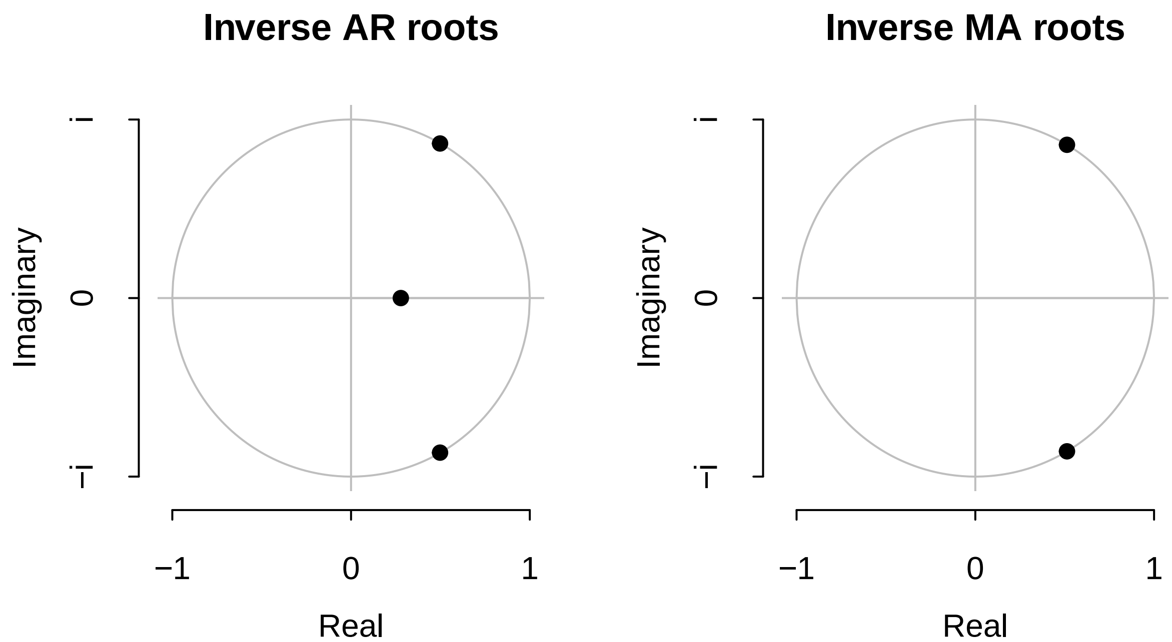
The test above shows the stationarity of the process with an $\alpha >= 1$, (indipendetemente dalla specificazione) Thus, the order of integration of the rea is the second one, because the series is an $I(1)$.

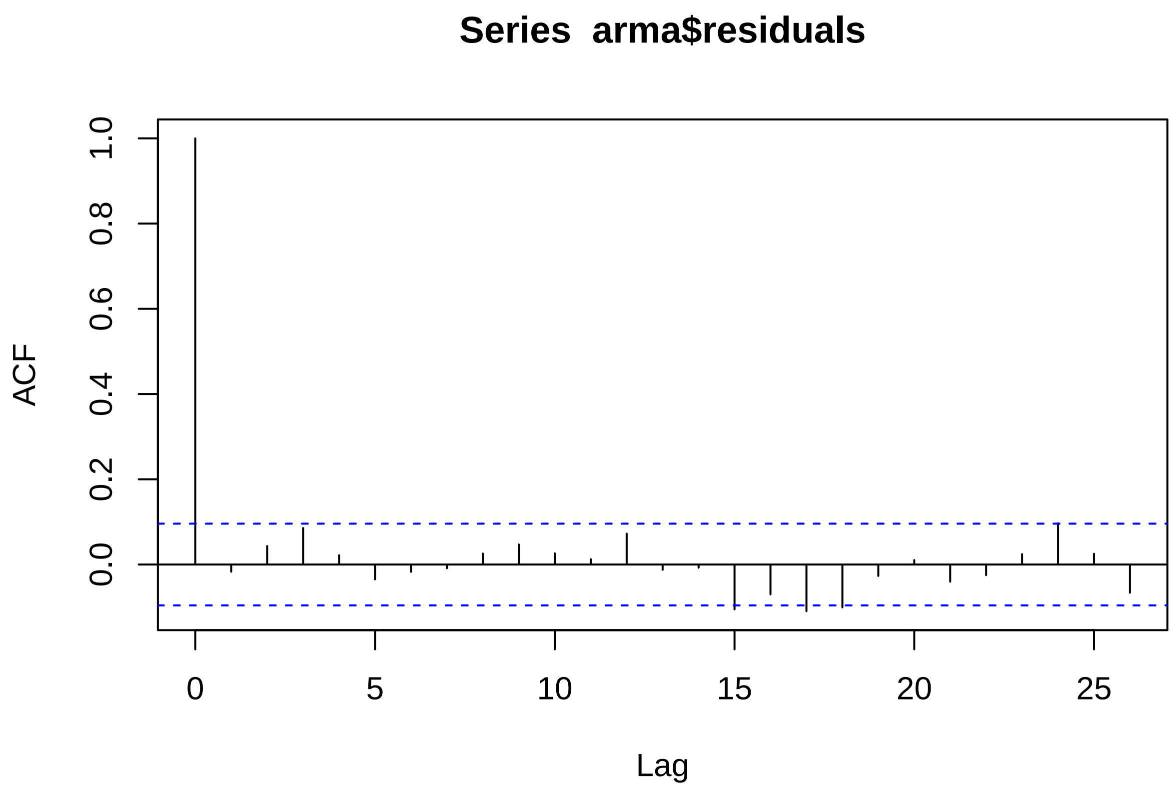
Point 3

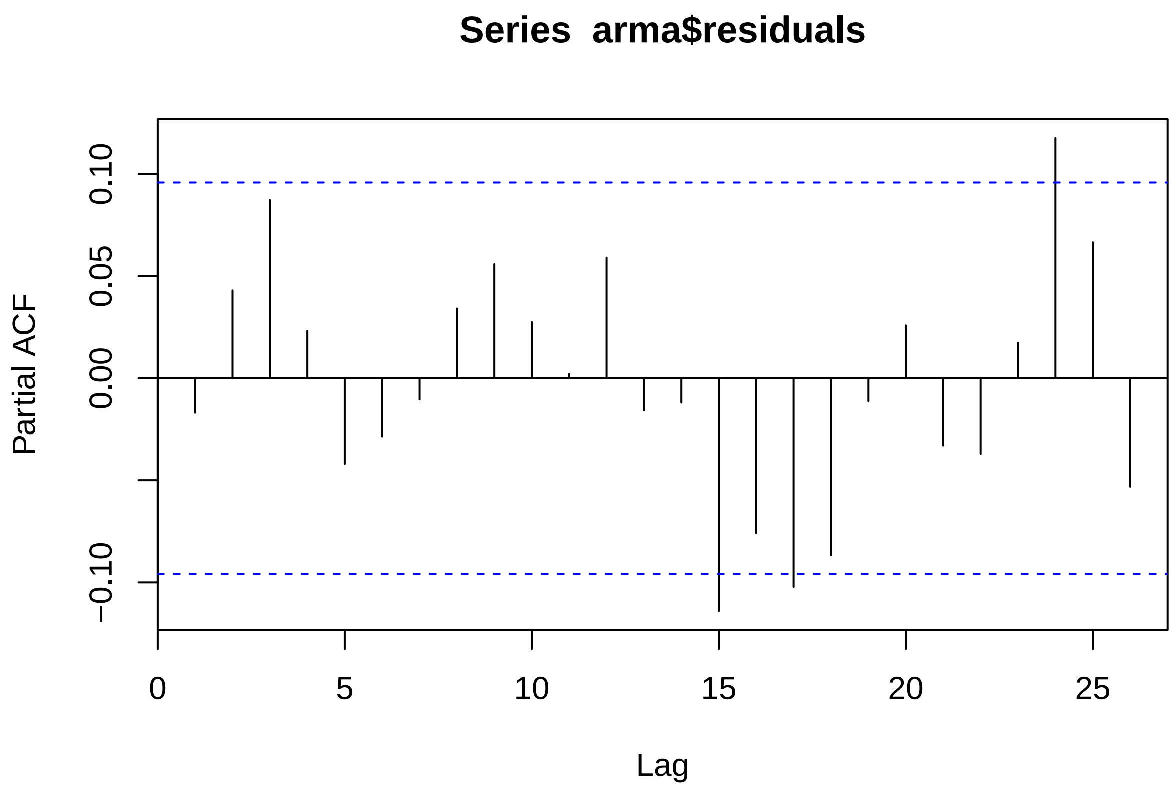
We select the best arma model setting the iper-paramenters (p,q), using the BIC criteria:




```
## [1] "BIC: 2488.27968170354"

##
## Call:
## arima(x = timeseries, order = best_arima, method = "ML")
##
## Coefficients:
##          ar1     ar2    ar3     ma1    ma2  intercept
##        1.274  -1.275  0.278  -1.026  1.000      0.046
## s.e.  0.048   0.048  0.048   0.014  0.021      0.284
##
## sigma^2 estimated as 18.6:  log likelihood = -1207.9,  aic = 2429.8
##
## Training set error measures:
##                     ME   RMSE    MAE    MPE  MAPE    MASE      ACF1
## Training set 0.0049331 4.3126 3.2828 65.447 204.8 0.76975 -0.016833
```


Inverse AR roots
Inverse MA roots
Imaginary
i
0
-i
−1
0
1
Real


Series arma$residuals
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
25
Lag



The autcorrelation function of the residuals it is not statistically different from 0, it looks like white noise. So the arma model adopted is one the fit perfectly the time series:

$$y_t = \theta_1 y_{t-1} + \theta_2 y_{t-2} + \theta_3 y_{t-3} + \beta_1 \epsilon_{t-1} + \beta_2 \epsilon_{t-2} + \epsilon_t$$

The issue regarding this model is an overfitting one, since all the point in the timeseries has been used to fit the model, as opposite to the usual practice. But the aim of this model is not to provide a prediction for the series, but instead the understading of the process in the specific time span of the series.

Point 4

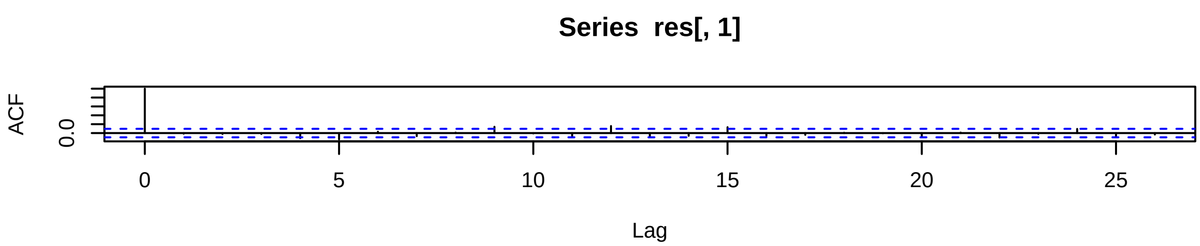
```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag  ADF p.value
## [1,]   0 22.3    0.99
## [2,]   1 32.7    0.99
## [3,]   2 41.8    0.99
## [4,]   3 50.3    0.99
## [5,]   4 59.7    0.99
## [6,]   5 66.9    0.99
## Type 2: with drift no trend
##      lag  ADF p.value
## [1,]   0 22.3    0.99
## [2,]   1 32.7    0.99
```

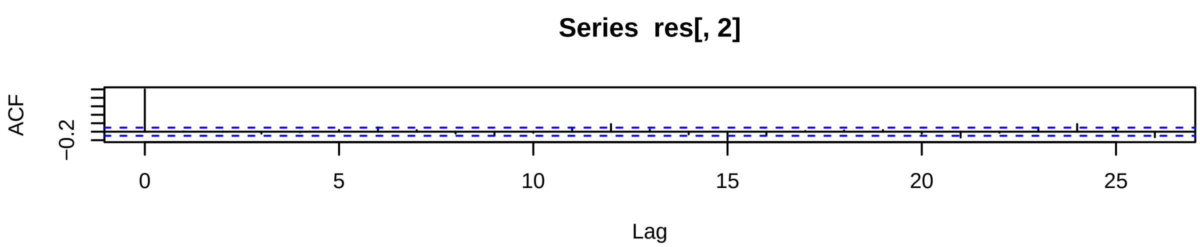
```
## [3,]   2 41.9    0.99
## [4,]   3 50.5    0.99
## [5,]   4 60.1    0.99
## [6,]   5 67.4    0.99
## Type 3: with drift and trend
##      lag  ADF p.value
## [1,]   0 22.3    0.99
## [2,]   1 32.7    0.99
## [3,]   2 41.9    0.99
## [4,]   3 50.5    0.99
## [5,]   4 60.1    0.99
## [6,]   5 67.5    0.99
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

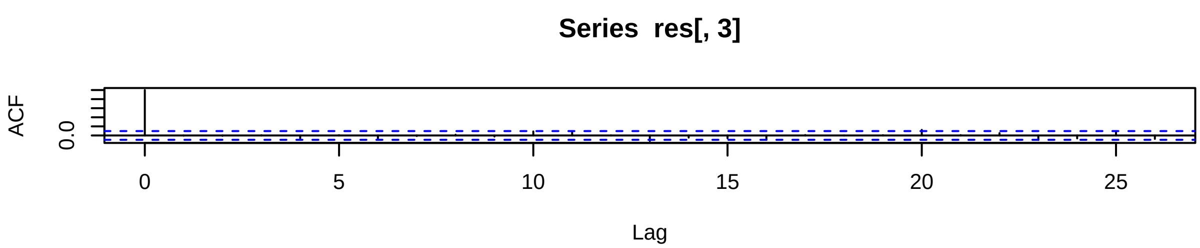
```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag  ADF p.value
## [1,]   0 2.47   0.990
## [2,]   1 1.42   0.960
## [3,]   2 1.60   0.973
## [4,]   3 1.63   0.975
## [5,]   4 1.56   0.970
## [6,]   5 1.35   0.955
## Type 2: with drift no trend
##      lag  ADF p.value
## [1,]   0 2.46    0.99
## [2,]   1 1.41    0.99
## [3,]   2 1.60    0.99
## [4,]   3 1.63    0.99
## [5,]   4 1.56    0.99
## [6,]   5 1.35    0.99
## Type 3: with drift and trend
##      lag  ADF p.value
## [1,]   0 2.47    0.99
## [2,]   1 1.42    0.99
## [3,]   2 1.61    0.99
## [4,]   3 1.64    0.99
## [5,]   4 1.57    0.99
## [6,]   5 1.36    0.99
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

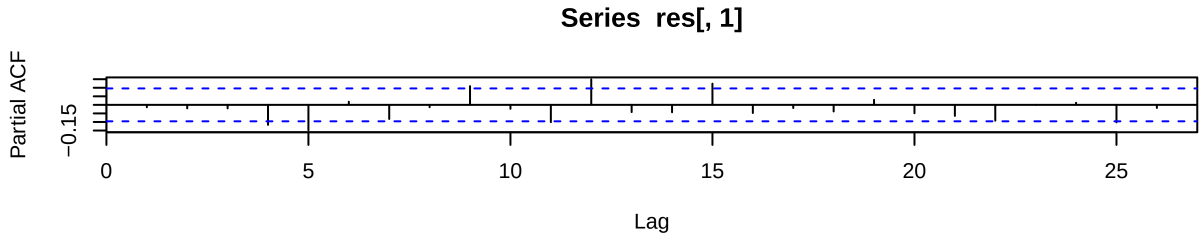
```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag   ADF p.value
## [1,]   0 1.852   0.984
## [2,]   1 0.579   0.811
## [3,]   2 0.886   0.899
## [4,]   3 0.933   0.906
## [5,]   4 1.072   0.923
```

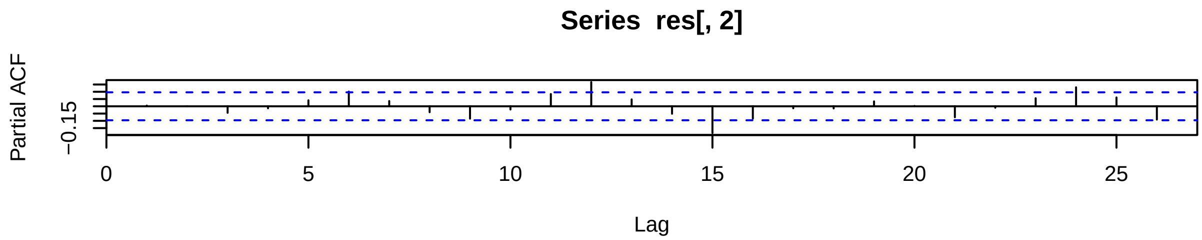
```
## [6,]   5 1.081   0.924
## Type 2: with drift no trend
##      lag   ADF p.value
## [1,]   0 1.847   0.990
## [2,]   1 0.579   0.989
## [3,]   2 0.886   0.990
## [4,]   3 0.933   0.990
## [5,]   4 1.071   0.990
## [6,]   5 1.081   0.990
## Type 3: with drift and trend
##      lag   ADF p.value
## [1,]   0 2.137    0.99
## [2,]   1 0.714    0.99
## [3,]   2 1.070    0.99
## [4,]   3 1.145    0.99
## [5,]   4 1.313    0.99
## [6,]   5 1.332    0.99
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

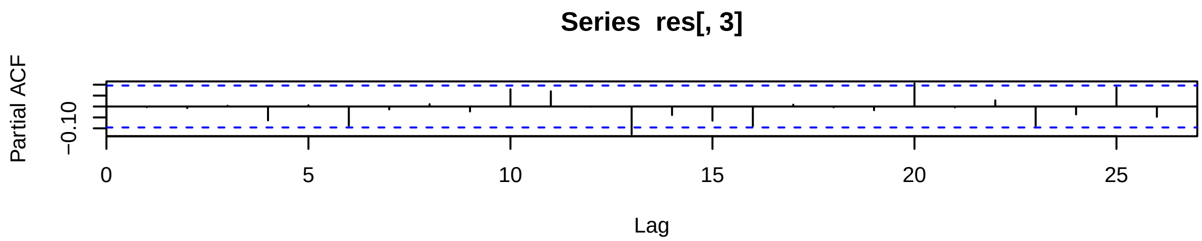












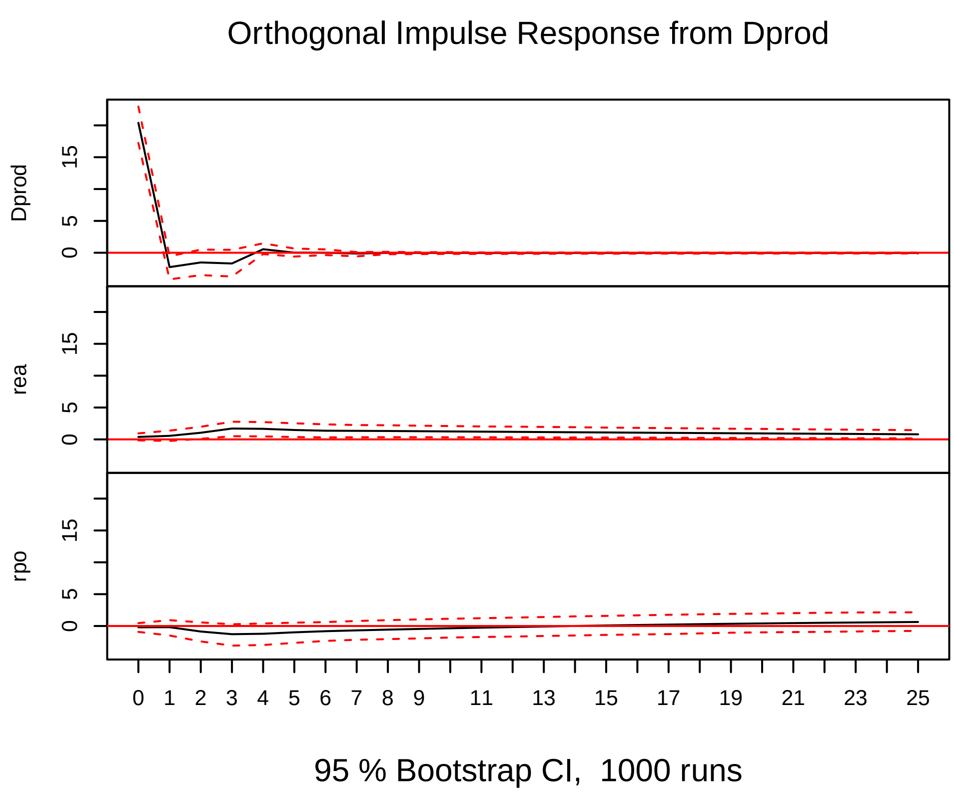

```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: Dprod, rea, rpo
## Deterministic variables: const
## Sample size: 416
## Log Likelihood: -4392.894
## Roots of the characteristic polynomial:
## 0.97 0.97 0.47 0.463 0.463 0.459 0.459 0.292 0.292
## Call:
## VAR(y = oil, type = "const", lag.max = 3, ic = "AIC")
##
##
## Estimation results for equation Dprod:
## ======================================
## Dprod = Dprod.l1 + rea.l1 + rpo.l1 + Dprod.l2 + rea.l2 + rpo.l2 + Dprod.l3 + rea.l3 + rpo.l3 + const
##
##          Estimate Std. Error t value Pr(>|t|)
## Dprod.l1 -0.11675    0.04935   -2.37    0.018 *
## rea.l1    0.26554    0.22167    1.20    0.232
## rpo.l1   -0.11809    0.16268   -0.73    0.468
## Dprod.l2 -0.08582    0.04937   -1.74    0.083 .
## rea.l2   -0.32947    0.34888   -0.94    0.346
## rpo.l2    0.38902    0.26957    1.44    0.150
## Dprod.l3 -0.10959    0.04962   -2.21    0.028 *
## rea.l3    0.00797    0.22101    0.04    0.971
```

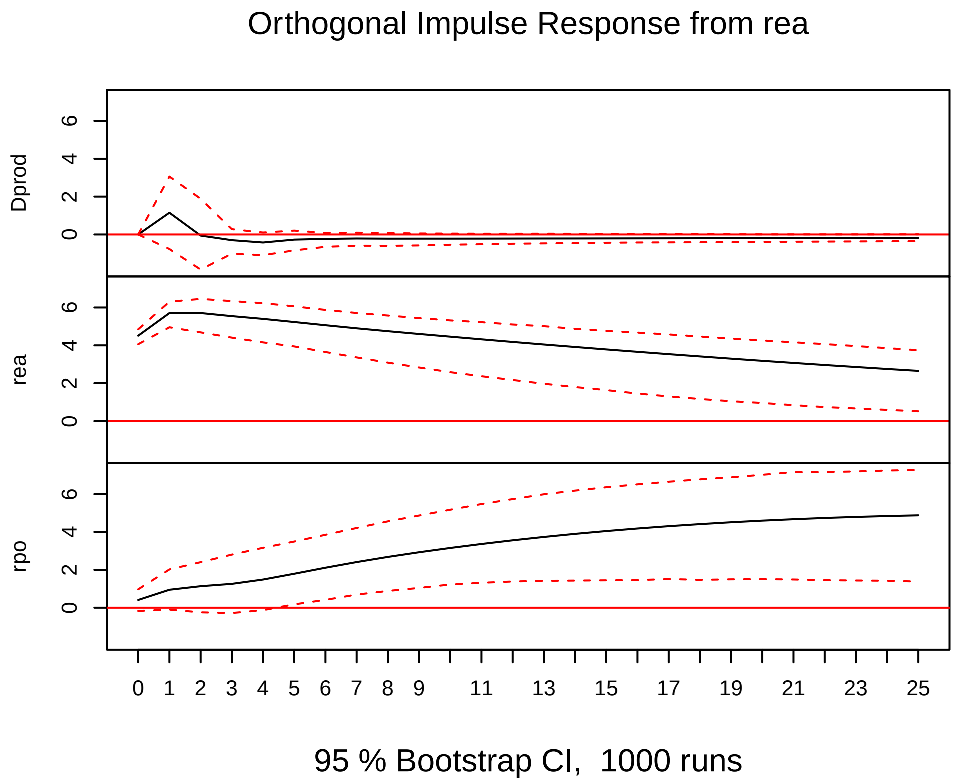
```
## rpo.l3   -0.28326    0.16175   -1.75    0.081 .
## const     1.06061    1.00498    1.06    0.292
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 20.4 on 406 degrees of freedom
## Multiple R-Squared: 0.0397,  Adjusted R-squared: 0.0184
## F-statistic: 1.86 on 9 and 406 DF,  p-value: 0.0558
##
##
## Estimation results for equation rea:
## ====================================
## rea = Dprod.l1 + rea.l1 + rpo.l1 + Dprod.l2 + rea.l2 + rpo.l2 + Dprod.l3 + rea.l3 + rpo.l3 + const
##
##          Estimate Std. Error t value Pr(>|t|)
## Dprod.l1  0.00436    0.01095    0.40   0.6906
## rea.l1    1.25795    0.04918   25.58  < 2e-16 ***
## rpo.l1    0.08174    0.03609    2.26   0.0241 *
## Dprod.l2  0.02371    0.01095    2.16   0.0310 *
## rea.l2   -0.33868    0.07740   -4.38  1.5e-05 ***
## rpo.l2   -0.06405    0.05981   -1.07   0.2848
## Dprod.l3  0.03322    0.01101    3.02   0.0027 **
## rea.l3    0.05412    0.04903    1.10   0.2703
## rpo.l3   -0.01712    0.03589   -0.48   0.6336
## const    -0.06388    0.22296   -0.29   0.7746
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 4.53 on 406 degrees of freedom
## Multiple R-Squared: 0.965,   Adjusted R-squared: 0.964
## F-statistic: 1.25e+03 on 9 and 406 DF,  p-value: <2e-16
##
##
## Estimation results for equation rpo:
## ====================================
## rpo = Dprod.l1 + rea.l1 + rpo.l1 + Dprod.l2 + rea.l2 + rpo.l2 + Dprod.l3 + rea.l3 + rpo.l3 + const
##
##          Estimate Std. Error t value Pr(>|t|)
## Dprod.l1  0.00359    0.01494    0.24   0.8104
## rea.l1    0.08024    0.06710    1.20   0.2325
## rpo.l1    1.43514    0.04925   29.14  < 2e-16 ***
## Dprod.l2 -0.03519    0.01495   -2.35   0.0190 *
## rea.l2   -0.09882    0.10561   -0.94   0.3500
## rpo.l2   -0.58996    0.08160   -7.23  2.4e-12 ***
## Dprod.l3 -0.01240    0.01502   -0.83   0.4094
## rea.l3    0.06237    0.06690    0.93   0.3517
## rpo.l3    0.13258    0.04896    2.71   0.0071 **
## const     0.24641    0.30422    0.81   0.4184
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
```

```
## Residual standard error: 6.18 on 406 degrees of freedom
## Multiple R-Squared: 0.982,   Adjusted R-squared: 0.982
## F-statistic: 2.47e+03 on 9 and 406 DF,  p-value: <2e-16
##
##
##
## Covariance matrix of residuals:
##        Dprod   rea   rpo
## Dprod 416.15  7.82 -4.10
## rea     7.82 20.48  1.77
## rpo    -4.10  1.77 38.13
##
## Correlation matrix of residuals:
##         Dprod    rea     rpo
## Dprod  1.0000 0.0848 -0.0325
## rea    0.0848 1.0000  0.0632
## rpo   -0.0325 0.0632  1.0000
```

```
##          Dprod     rea     rpo
## Dprod 416.1453  7.8250 -4.0996
## rea     7.8250 20.4834  1.7659
## rpo    -4.0996  1.7659 38.1323
```

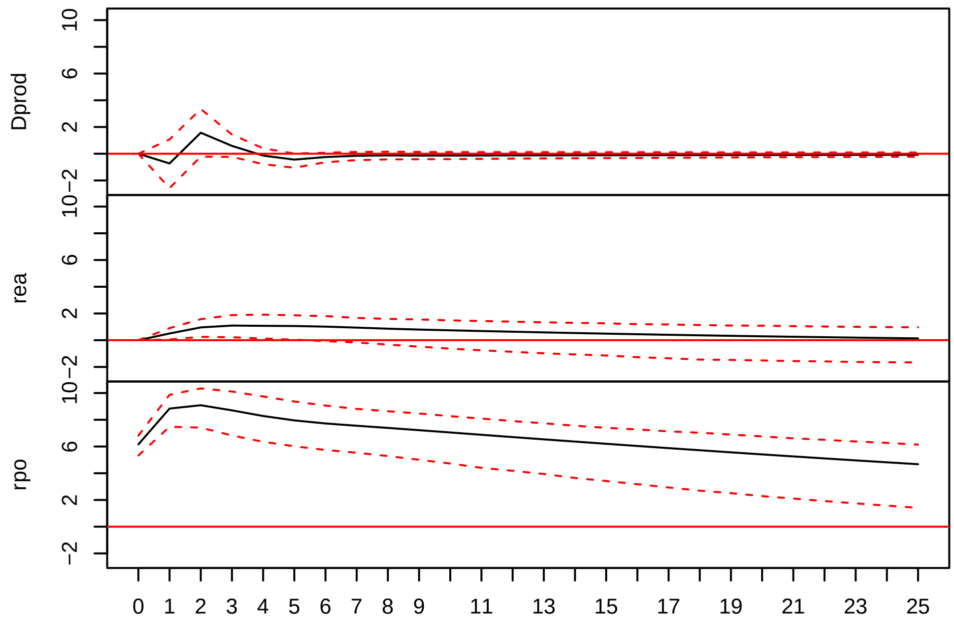
```
##           Dprod      rea       rpo
## Dprod  1.000000 0.084754 -0.032544
## rea    0.084754 1.000000  0.063185
## rpo   -0.032544 0.063185  1.000000
```





Orthogonal Impulse Response from rea
Dprod
rea
rpo
0
2
4
6
0 1 2 3 4 5 6 7 8 9 11 13 15 17 19 21 23 25
95 % Bootstrap CI, 1000 runs

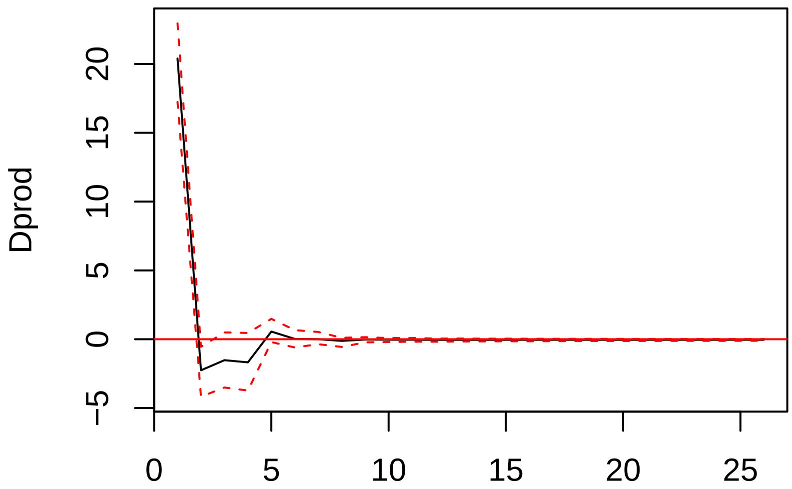
Orthogonal Impulse Response from rpo



95 % Bootstrap CI, 1000 runs

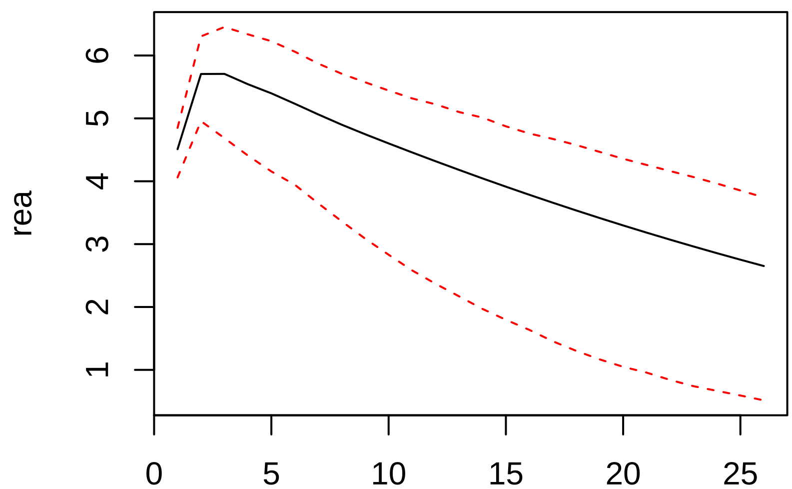
Point 5

Orthogonal Impulse Response from Dprod

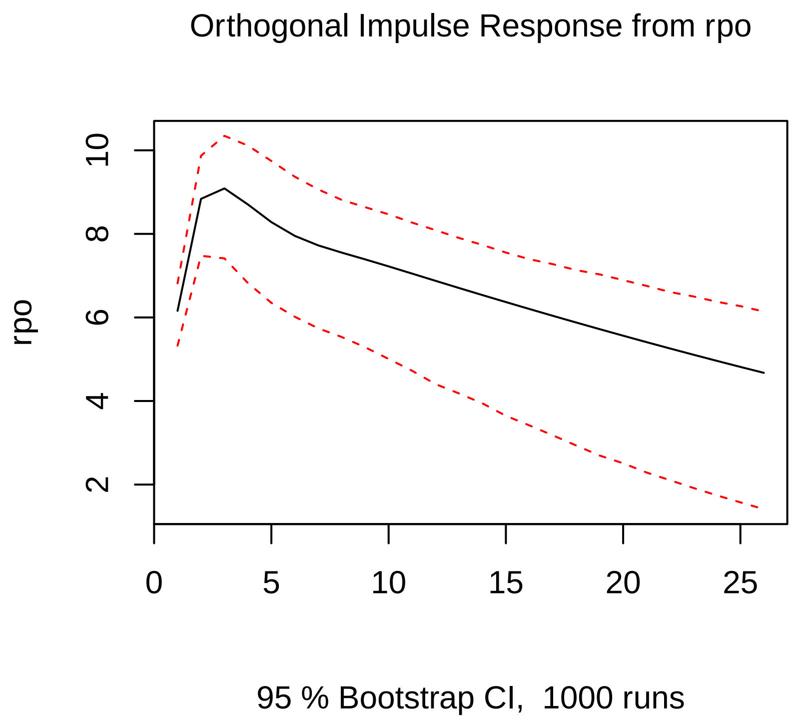


95 % Bootstrap CI, 1000 runs

Orthogonal Impulse Response from rea



95 % Bootstrap CI, 1000 runs


Orthogonal Impulse Response from rpo
rpo
10
8
6
4
2
0
5
10
15
20
25
95 % Bootstrap CI, 1000 runs