Time Series: ARIMA and SETAR

Contents

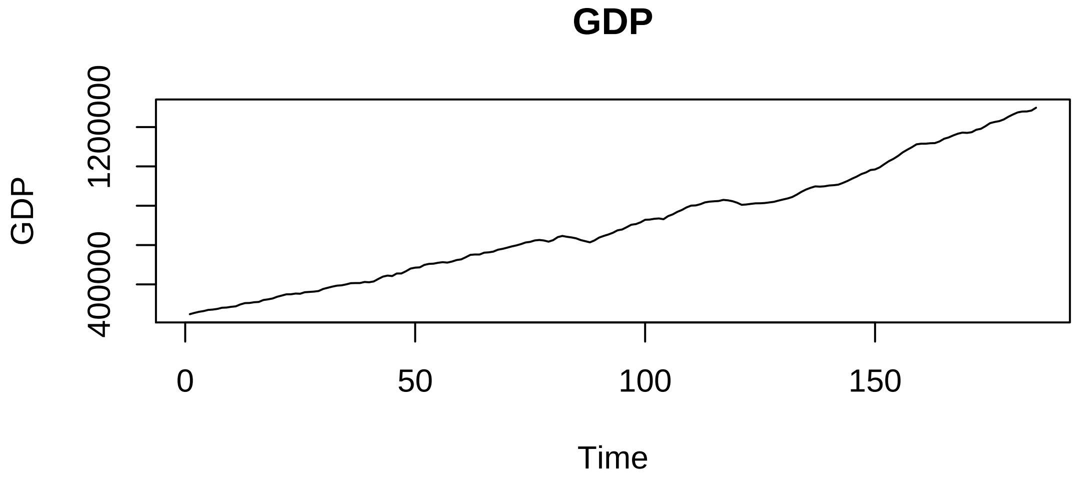
1 Linear forecast of the GDP Data: ARIMA model

1.1 Substract the Bussiness Cycle

1.1.1 Plot of the data

I this profject, we will use seasonal adjusted GDP data to do some forecasting. We will first plot the data to have a sense of them first:

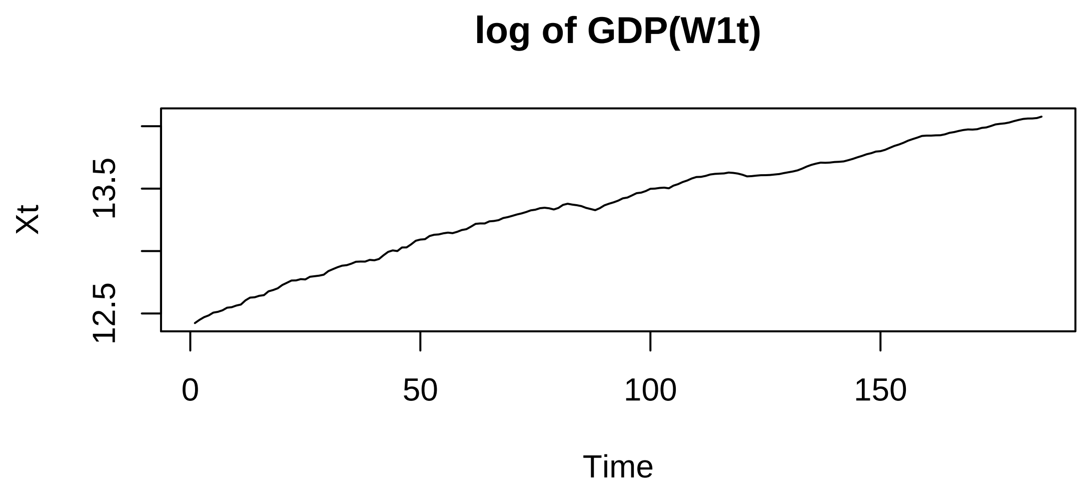
```r
options(warn = -1)

gdp_cons = read.csv('GDP_CONS_Canada.csv',header = T)
W1t = gdp_cons$GDP

plot(W1t, main = 'GDP', ylab = 'GDP', xlab ='Time', type = 'l')
```



Comment: They are the data from the first quarter of the year 1961 till the first quarte of 2007. And the two series are both increasing in a way we showed above.

In order to have a better analysis of the data, we first define $X_t = ln(W_{1t})$. We can see that the plot of Xt is:

```
Xt = log(W1t)
plot(Xt, main ='log of GDP(W1t)', xlab = 'Time',type = 'l')
```



Now we can see that it is still an increasing series. But the data now are much well behaved, and the difference between any two Xt represents the percentage change now.

1.1.2 Trend Stationary Approach and Difference Stationary Approach

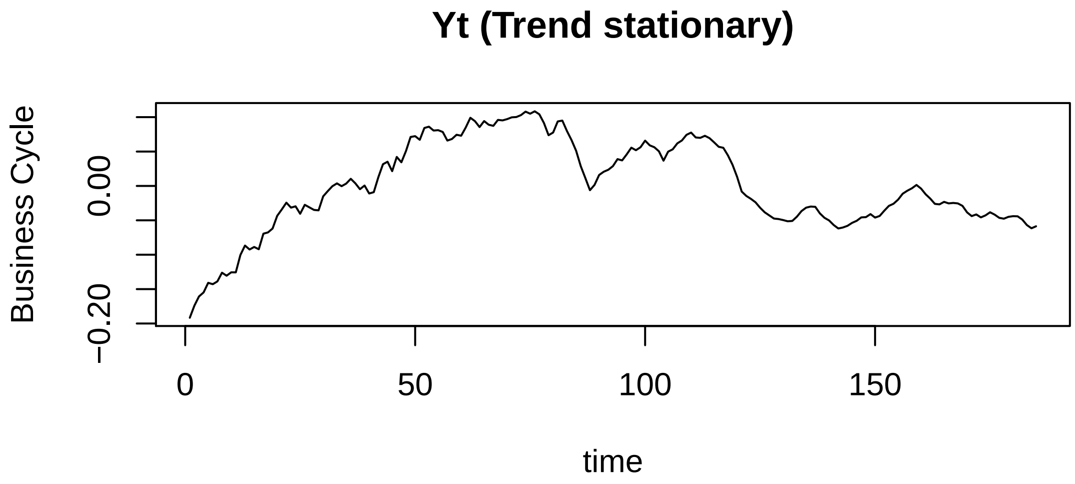
Then we will construct the measure of the business cycle Yt using trend stationary approach and difference stationary approach as follow:

Trend Stationary Approach: $X_t = \alpha + \mu t + Yt$

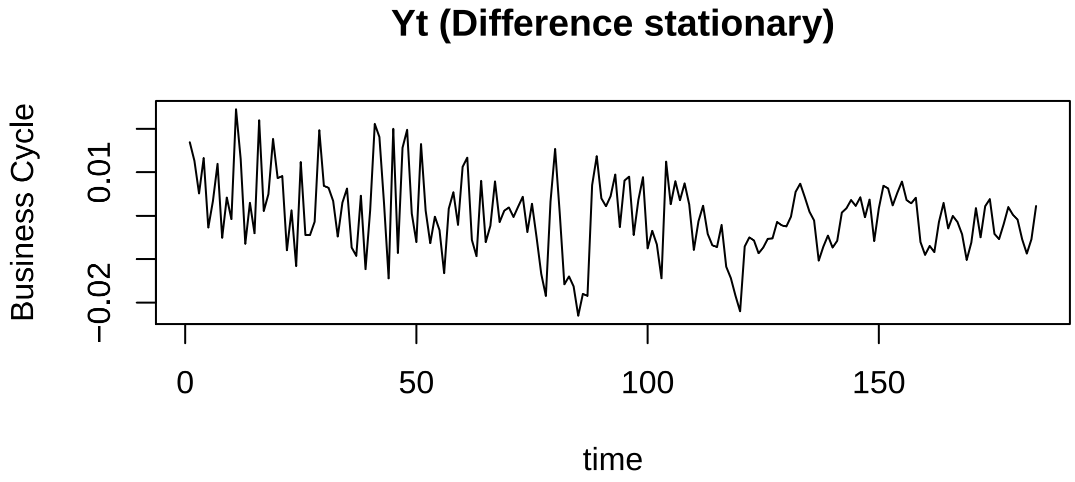
Difference Stationary Approach: $\delta X_t = \mu + Yt$

Then we get the two cycles showing as below:

```
tSeq <- 1:length(Xt)

model_ts <- lm(Xt~tSeq)
Yt_ts <- model_ts$residuals

dXt <- diff(Xt)
model_ds <- lm(dXt ~ 1)
Yt_ds <- model_ds$residuals

plot(Yt_ts,main = 'Yt (Trend stationary)', ylab = 'Business Cycle', type = 'l', xlab = '
```



```
plot(Yt_ds,main = 'Yt (Difference stationary)', ylab = 'Business Cycle', type = 'l', xla
```



1.1.3 Dickey-Fuller Test

Now question becomes: If we want to decide wether we should use TS or DS approach to substract bussiness cycle from raw time series, we will need to use Augmented Dickey-Fuller (ADF) test:

Let $\Delta X_t = \alpha + \mu t + \phi X_{t-1} + \theta_1 \Delta X_{t-1} + \theta_2 \Delta X_{t-2} + \theta_3 \Delta X_{t-3} + \theta_4 \Delta X_{t-4} + \theta_5 \Delta X_{t-5} + a_t$

H_0: $\phi = 0$ which is equivalent to using DS approach.

H_a: $\phi \neq 0$ which is equivalent to using TS approach.

Under the null hypothesis, we have:$t_{stat} = \frac{\hat{\phi}-0}{se(\hat{\phi})}$, where $t_{stat} \sim t_{n-8}$, with the $t_{crit}(\alpha = 0.05, 2sided, df = n-8)$

```
library(tseries)
adf.test(Xt, k =5)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  Xt
## Dickey-Fuller = -2.6929, Lag order = 5, p-value = 0.2865
## alternative hypothesis: stationary
```

We perform the ADF test, and we find that $t_{stat} = -2.6929$, making the associated p-value 0.2865, which is bigger than 0.05. So we do not reject the null hypothesis that we should use DS approach.

1.2 ARIMA model for modelling Bussiness Cycle

1.2.1 Box-Jenkins Identification

Now we will be using the bussiness cylcle subtracted using difference stationary approach and model it using ARIMA model. In order to identify the cycle, we will use Box-Jenkins indentification.

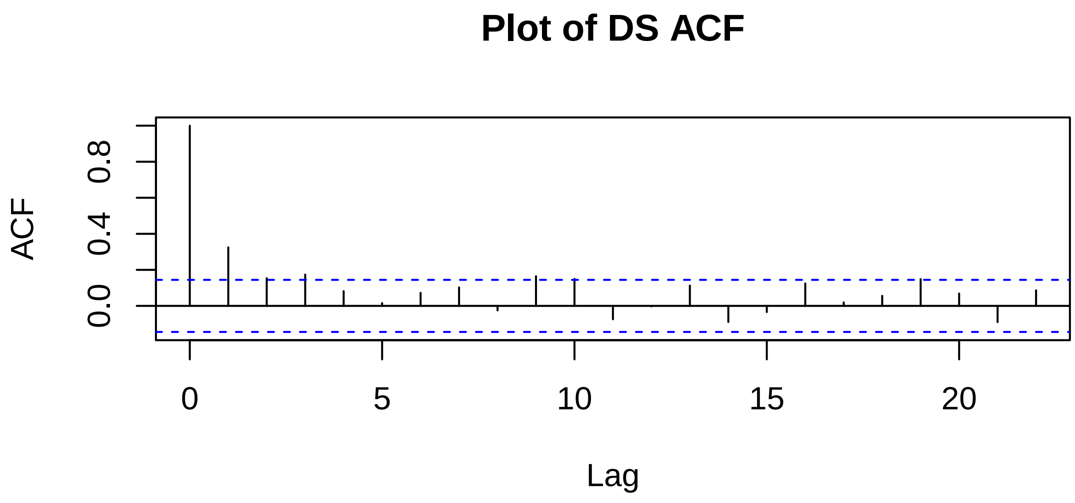
Box-Jenkins Identification saying that we can use the property of the acf and pacf to determine the AR or MA or ARMA process.We will show this relationship using the following table:

```
cutoff.property <- data.frame(
  model = c('AR(P)','MA(q)','ARMA(p,q)'),
  acf = c('damped exponential','cut-off at k=q','damped exponential'),
  pacf = c('cut-off at k=q','damped exponential','damped exponential')
)

library(kableExtra)
kable(cutoff.property,align='c')%>%
  column_spec(c(1,2,3),width='4cm')
```

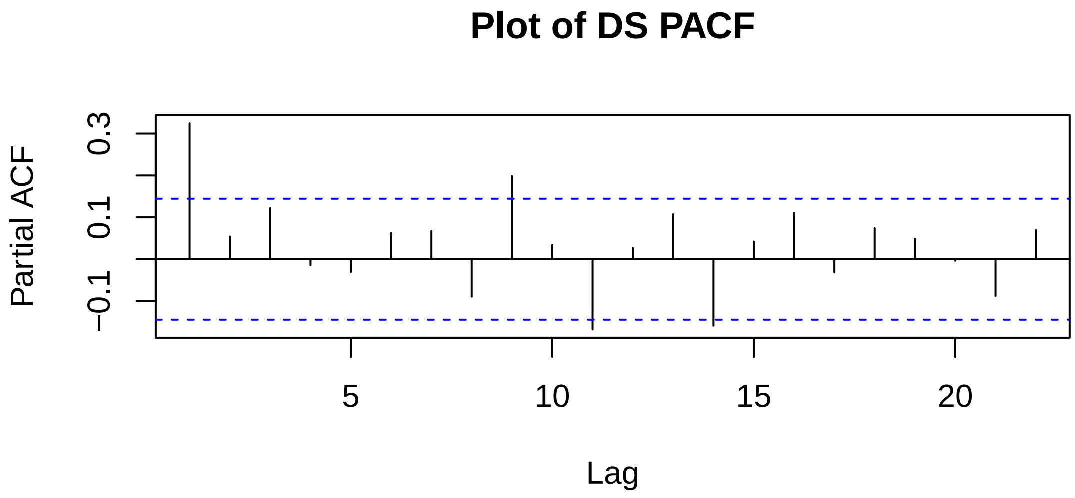
model	acf	pacf
AR(P)	damped exponential	cut-off at k=q
MA(q)	cut-off at k=q	damped exponential
ARMA(p,q)	damped exponential	damped exponential

Knowing the above property, we first plot the plots of acf and pacf of DS Yt:

```
acf(Yt_ds, main = 'Plot of DS ACF')
```



```
pacf(Yt_ds, main ='Plot of DS PACF')
```



The plots show that there are too many noise that we can not direct draw our conclusion base on that. So we will use hypothesis testing to test the cut-off.

Noticing that the distribution of acf(k) $\hat{\rho}(k) = N(0, 1/n)$, under the null hypothesis that $\rho(k) = 0$, we have the test statistics $\frac{|\hat{\rho}(k)-\rho(k)|}{se(\hat{r}ho(k))}$ should be expected to be smaller than 1.96 (or 2 roughly). So we just need to check if $|\hat{\rho}(k)| < 2se(\hat{\rho}(k)) = \frac{2}{\sqrt{n}} = 0.147442$ is true , then $\rho(k) = 0$. Same asymptotic distribution applies for partial acf, we just need to check if $|\phi_{kk}| < \frac{2}{\sqrt{n}} = 0.147442$.

We show the values of acf and pacf in the table:

```
lag <- c('1','2','3','4','5','6')
acf <- c(0.325, 0.154, 0.174, 0.082, 0.016, 0.072)
pacf <- c(0.325, 0.054, 0.122, -0.014, -0.031, 0.062)

kable(rbind(lag,acf,pacf),align = 'c')
```

lag	1	2	3	4	5	6
acf	0.325	0.154	0.174	0.082	0.016	0.072
pacf	0.325	0.054	0.122	-0.014	-0.031	0.062

Finding that there is a cut-off at 3 in acf and there is a cut-off at 1 in pacf. That means the process is either AR(1) or MA(3).

1.2.2 Bayesian Information Criteria

An alternative way to find the optimal lag for AR(p) or MA(q) is through using Bayesian Information Criteria.

As we know that, the formular of BIC is:

$$BIC(k) = ln(\hat{\sigma}_k^2) + \frac{ln(n) * k}{n}$$

where k = 1,. . . ,8

If the value of BIC reaches the minimum value at lag k, then we conclude that the optimal lag for the AR or MA process is k.

Let’s use the table to show the BIC values with different lags for the business cycle we get:

```
BIC <- function(res, k, N) {
  bic <- log(sum(res^2) / N)
  bic <- bic + log(N) * k / N
  bic
}



bic.ar.ds <- rep(NA,8)
bic.ma.ds <- rep(NA,8)

N_ds <- length(Yt_ds)

for(ii in 1:8) {
  model.ar.ds <- arima(Yt_ds, order = c(ii,0,0), include.mean = F)
  res.ar.ds <- model.ar.ds$residuals
  bic.ar.ds[ii] <- round(BIC(res.ar.ds, ii, N_ds),4)
  model.ma.ds <- arima(Yt_ds, order = c(0,0,ii), include.mean = F)
  res.ma.ds <- model.ma.ds$residuals

  bic.ma.ds[ii] <- round(BIC(res.ma.ds, ii, N_ds),4)
}


#put them to table
BIC_table <- data.frame(rbind(c('1','2','3','4','5','6','7','8'),bic.ar.ds, bic.ma.ds))
colnames(BIC_table) <- c()
rownames(BIC_table)  <- c('q','BIC.ar.ds','BIC.ma.ds')
BIC_table
```

```
##
## q                 1       2       3       4       5       6       7       8
## BIC.ar.ds -9.5547 -9.5298 -9.5172 -9.4889 -9.4617 -9.4377 -9.4148 -9.3957
## BIC.ma.ds -9.5406  -9.518 -9.5116 -9.4881 -9.4598 -9.4323 -9.4245 -9.3996
```

So from the table we know that the smallest BIC for difference stationary Yt happens in the lag of p = 1 for both AR and MA model. Then we choose that the optimal lag is p = 1. Then we can fit an AR(1) process to the Yt we got from the difference stationary approach.

1.2.3 Model Estimation

Note that when we estimate the AR(1) model, we are using OLS method. And recall from the OLS estimation that the estimated coefficient vector of $Y = XB + a$ is $\hat{\beta}_{OLS} = (X'X)^{-1}X'Y$.

```
model.ar1 <- arima(Yt_ds, order = c(1,0,0), include.mean = F)
model.ar1
```

```
##
## Call:
## arima(x = Yt_ds, order = c(1, 0, 0), include.mean = F)
##
## Coefficients:
##          ar1
##       0.3295
## s.e.  0.0701
##
## sigma^2 estimated as 6.888e-05:  log likelihood = 620.5,  aic = -1237
```

By calling ARIMA, we can get the estimate in that way as:

$$Y_t = \underset{(0.0701)}{0.3295} Y_{t-1} + \epsilon_t$$

where the numbers in the brackets underneath are the corresponding standard error.

```
rhos = ARMAacf(coef(model.ar1),lag.max = 6)
#rhos

sigma2.hat = sum(model.ar1$residuals^2)/N_ds
gamma0.sqrt = sqrt(sigma2.hat/(1-coef(model.ar1)%*%rhos[2]))
#gamma0.sqrt
```

If we want to know something about the correlation of the lags, also we know that:$\gamma(0) = \frac{\sigma^2}{1-\phi_1\rho(1)}$. And also from the model estimation we know $\rho(k) = \phi_1\rho(k-1)$, with $\rho(0) = 1$ and $\rho(k) = \rho(-k)$. Then we get the following values:

$\rho_1 = 0.329536915$, $\rho_2 = 0.108594578$, $\rho_3 = 0.035785922$, $\rho_4 = 0.011792782$, $\rho_5 = 0.003886157$,$\rho_6 = 0.001280632$

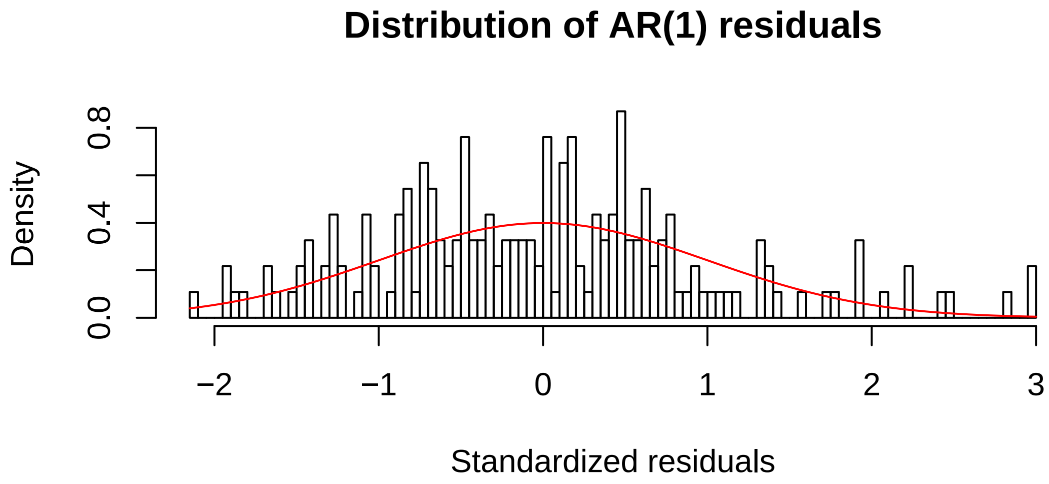
So finally we get the $\gamma(0)^{1/2} = 0.008790685$, which is quite small.

1.3 Diagnostic Test

We will be using AR(1) and run the diagnostic tests for it to see if the assumptions are satisfied for the model:

We first do that by checking if the residuals are normal:

```
ar1_ds <- arima(Yt_ds, order=c(1,0,0), include.mean = F)

st.ar1_res_ds <- ar1_ds$residuals/sqrt(ar1_ds$sigma2)



hist(st.ar1_res_ds ,main ='Distribution of AR(1) residuals',xlab='Standardized residuals
curve(dnorm(x),col = "red", add = T)
```



Looking from the histogram, we find that the standardized residuals are all appear to be roughly normally distributed. Also, we may want to perform some formal diagnostic test on that.

1.3.1 Box-Pierce Test

Sometimes we would like to have a test for residuals of their joint correlation. Here is where the Box-Pierce test comes in.

Under the H_0: $\rho_\epsilon(k) = 0$ for k = 1,. . . ,M. M is the total number of the residuals. We will have $\sqrt{n}\rho_\epsilon(k) \sim$ N(0,1) asymptotically for a particular k. And then we have:

$$Q_{stat} = \sum_k (\sqrt{n}\rho_\epsilon(k))^2 = n \sum_k \rho_\epsilon^2(k)$$

where $Q_{stat} \sim \chi^2(2)$ asymptotically.

```r
M <- ceiling(sqrt(length(Yt_ds)))
pval <- Box.test(x=ar1_ds$residuals,type='Box-Pierce',lag=M)
#pval > 1.96/sqrt(length(Yt))
pval
```

```
##
##  Box-Pierce test
##
## data:  ar1_ds$residuals
## X-squared = 31.255, df = 14, p-value = 0.005105
```

We run Box-Pierce test for both AR(1) and MA(3):

H_0 : there is no joint autocorrelation up to order $M \approx 14$

Test statistics is $Q_{stat} = n\sum_{k=1}^{14}\hat{\rho}_{\epsilon}^2(k) \sim \chi^2(14)$

For AR(1),$p - value = Pr(\chi^2(14) > Q_{stat}) = 0.005105 < 0.05$, we reject the H_0 at 95% confidence level.

1.3.2 Overfitting

The method we use to test the over fitting is to use the likelihood ratio test. We derive the log likelihood function of Yt as $l(\theta, \sigma^2) = -\frac{n}{2}ln(\sigma^2) - \frac{n}{2\sigma^2}\sum\hat{\epsilon}_t^2$. Then we maximize the log likelihood function and get the estimated value for θ and σ^2. Use that we construct the the test statistics:

$$LRT = -2ln(\frac{l_{reduced}}{l_{full}}) = nln(\frac{\sigma_{red}^2}{\sigma_{full}^2})$$

```r
ar5_ds <- arima(Yt_ds, order = c(5,0,0), include.mean = FALSE)

LR_ds <- N_ds*log(ar1_ds$sigma2/ar5_ds$sigma2)
1- pchisq(LR_ds,4)
```

```
## [1] 0.4423065
```

We run a LRT:

For AR(1) model:

H_0 : AR(1) model is more adequate than AR(5)

Test statistics is $LRT = n(\frac{\sigma_r^2}{\sigma_f^2}) = 3.740197 \sim \chi^2(4)$

$p - value = Pr(\chi^2(4) > 3.740197) = 0.4423065 > 0.05$, we do not reject the H_0 at 95% confidence level.

1.4 K-step ahead forecast of Growth Rate of GDP using Yt

Noticing that

$$\Delta X_t = \Delta ln(W_t) = ln(W_2) - ln(W_1) = ln(\frac{W_2}{W_1}) = ln(1 + \frac{\Delta W}{W_1}) \approx \frac{\Delta W}{W_1}$$

The growth rate we are forecasting is actually the ΔX_t.

We know that using the Wold Representation we can construct the Expectation and Variance of Yt by using:

$$E_tY_{t+k} = \sum_{j}^{\inf}\psi_{k+j}a_{t-j}$$

$$Var_tY_{t+k} = \sigma^2(1 + \psi_1^2 + ... + \psi_k^2)$$

Then using this, we can find the growth rate of the raw series W_t by using:$\Delta X_t = ln(W_t) - ln(W_{t-1})$ with 95% confidence interval:

$$E_t[\Delta X_{t+k}] = \pm 2\sqrt{Var_t[\Delta X_{t+k}]}$$

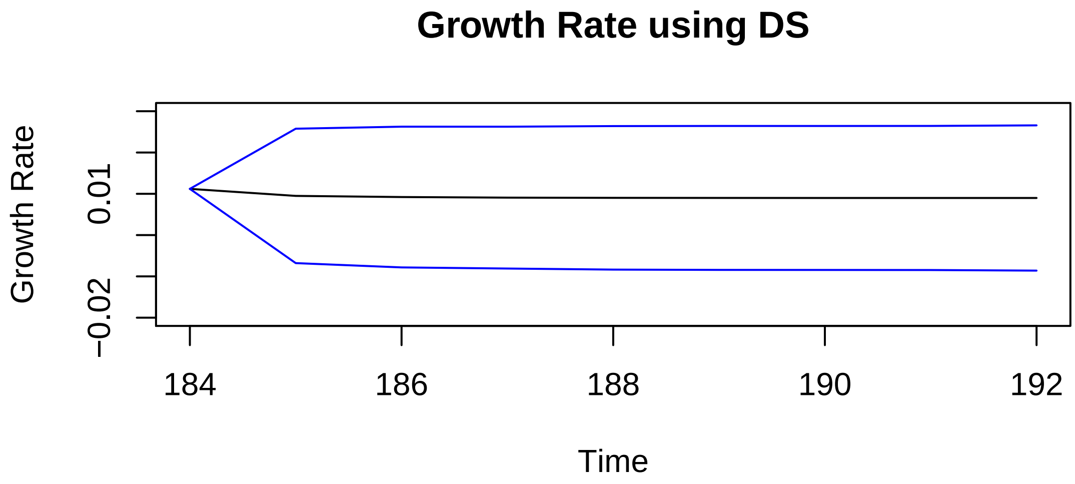
For the Difference Stationary Forecasting:

Since $\Delta X_t = \mu + Y_t$, we have $E_t[\Delta X_{t+k}] = \mu + E_t[Y_{t+k}]$.

Also, $Var_t[\Delta X_{t+k}] = Var_t[\mu + Y_{t+k}] = Var_t[Y_{t+k}] = \sigma^2 \sum_{j=0}^{k-1} \psi_j^2$

Then the result we can got are shown in the graphs below:

```
phi_ds <- model.ar.ds$coef



#DS
#E of Xt
E_ds <- rep(NA,9)
E_ds[1] <- Yt_ds[N_ds]
E_ds[2] <- phi_ds[1]*Yt_ds[N_ds] + phi_ds[2]*Yt_ds[N_ds - 1]
E_ds[3] <- phi_ds[1]*E_ds[2] + phi_ds[2]*Yt_ds[N_ds]

for(i in 4:9){
  E_ds[i] <- phi_ds[1]*E_ds[i-1] + phi_ds[2]*E_ds[i-2]
}

E_ds <- E_ds+ model_ds$coefficients

#var of Xt = var of Yt
V_ds <- rep(NA,9)
psis_ds <- c(1,ARMAtoMA(ar = coef(model.ar.ds),lag.max = 7))
V_ds[1] <- 0
for(i in 2:9){
  V_ds[i] <- model.ar.ds$sigma2*(sum(psis_ds[1:(i-1)]^2))
}
#V_ds

#confidence interval
E_ds.low <- rep(NA,9)
E_ds.high <- rep(NA,9)
for(j in 1:9){
  E_ds.low[j] <- E_ds[j] - 2*sqrt(V_ds[j])
  E_ds.high[j] <- E_ds[j] + 2*sqrt(V_ds[j])
}

plot(N_ds:(N_ds+8), E_ds, xlab='Time', ylab = 'Growth Rate', type = 'l',ylim=c(-0.02,0.0
points(N_ds:(N_ds+8), E_ds.low,type='l',col = 'blue')
points(N_ds:(N_ds+8), E_ds.high,type='l', col = 'blue')
```



We see that for the seasonal adjusted data, we have an estimated grwoth rate of GDP of 1%, quaterly.

2 Non-linear Forecast of Bussiness Cylcle: SETAR(m) Model

2.1 Model setup

Now we will use Self-excited autoregressive model (SETAR) model to model the business cycle Yt from difference stationary approach:

A 2 regime SETAR model in general has a form like this:

$$Y_t = \begin{cases} \beta_0^1 + \sum_{i=1}^{p_i} \beta_i^1 Y_{t-i} + \epsilon_t^1, Y_{t-d} < \gamma \\ \beta_0^2 + \sum_{i=1}^{p_i} \beta_i^2 Y_{t-i} + \epsilon_t^2, Y_{t-d} \geq \gamma \end{cases}$$

where γ is the threshold value, Y_{t-d} is threshold variable, d is the time delay. Note that if we set d = 0, threshold variable we will use then will be Y_t. So we will use the default value d = 0. And for now we assume that the model has two regime and we will test it later when we come to the diagnose part.

2.2 Lag and Threshold value Specification

In this model we have to figure out two things:/enskip 1. What is the lag up tp? 2. What is the threshold value that we should seperate the data.

Here we will the grid search on Akaike information criterion (AIC) to determine these two:

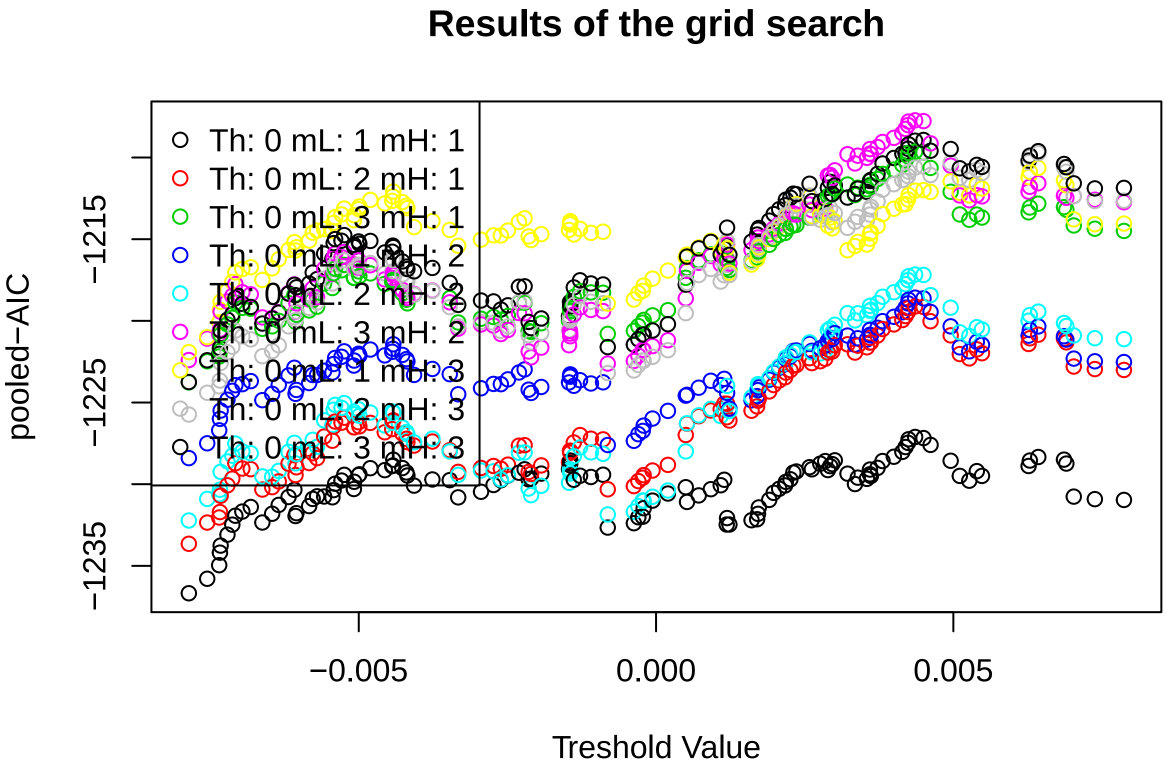
$$AIC(\hat{p}_i, \hat{\gamma}, \hat{d}) = min[AIC(k_i)], i = 1, 2$$

Grid-search is a way to select the best of a family of models, parametrized by a grid of parameters. By minimizing the AIC using grid search, we can get the optimal lags for the two regime and threshold value and threshold dalay which we use to calculate the estimated time delay d as well.

From the seaching algorithm we get a series of AIC values, then we show the result in table:

```
library(tsDyn)

selectSETAR(Yt_ds, m=3)
```

```
## Using maximum autoregressive order for low regime: mL = 3
## Using maximum autoregressive order for high regime: mH = 3
## Searching on 125 possible threshold values within regimes with sufficient ( 15% ) num
## Searching on  1125  combinations of thresholds ( 125 ), thDelay ( 1 ), mL ( 3 ) and M
```



```
## Results of the grid search for 1 threshold
##    thDelay mL mH            th pooled-AIC
## 1        0  1  1 -0.0078577950  -1236.677
## 2        0  1  1 -0.0075487764  -1235.791
## 3        0  1  1 -0.0073471581  -1234.958
## 4        0  1  1 -0.0073345826  -1234.193
## 5        0  1  1 -0.0073231466  -1233.759
## 6        0  2  1 -0.0078577950  -1233.648
## 7        0  1  1 -0.0072081482  -1233.091
## 8        0  1  1 -0.0008122907  -1232.656
## 9        0  1  1 -0.0071265322  -1232.479
## 10       0  1  1  0.0012335958  -1232.470
```

From the smallest AIC value, we notice that time delay d = 0, lag of the two regime are 1 and 1, and also the threshold value γ should be -0.0078577950, under the assumption of two regime. Then we can start to estimate the parameters base on those values.

2.3 Parameters Estimation

For estimation of SETAR(2) model, we will be using OLS estimation.

In the general case, if we let the y_t to be the univariate time series and $Y_{t-1} = (1, y_{t-1}, y_{t-2}, ..., y_{t-p})$ which is a $k \times 1$ vector with $k = p + 1$. A SETAR(m) model with m regimes takes the form:

$$y_t = \alpha'_1 Y_{t-1} I_{1t}(\gamma, d) + ... + \alpha'_m Y_{t-m} I_{mt}(\gamma, d) + e_t$$

where t = 1,..,T. T is the total number of the data. $\alpha_i, i = 1, ...m$ are $k \times 1$ coefficient vector for the corresponding Y_{t-i} in the ith regime. And $\gamma = (\gamma_1, ..., \gamma_{m-1})$ with $\gamma_1 < ... < \gamma_{m-1}$ is a vector of threshold parameter. $I(\gamma, d)$ is an indicator fuction that controls which piece of the function to use given the data.

Also, we need the assumption of

$$E_{t-1}(e_t) = 0, Var(e_t) = \sigma^2 < \infty$$

If we let $\theta = (\alpha_1, .., \alpha_m, \gamma, d)$ denote the collection of parameters in the SETAR(m) model, then under the assumption of $E_{t-1}(e_t) = 0, Var(e_t) = \sigma^2 < \infty$ using the OLS estimation, we will obtain $\hat{\theta}$ by:

$$\hat{\theta} = \underset{\theta}{argmin} \sum_{t=1}^{T} (y_t - \alpha'_1 Y_{t-1} I_{1t}(\gamma, d)) - ... - \alpha'_m Y_{t-m} I_{mt}(\gamma, d))^2$$

The OLS problem for SETAR(2) can be reduced to:

$$S = \underset{d,\gamma,\alpha}{min} (y - X(\gamma, d)\alpha)'(y - X(\gamma, d)\alpha)$$

where $X(\gamma, d)$ is the $T \times 2k$ matrix whose ith row is $X_{t-1}(\gamma, d)'$, and $X_{t-1}(\gamma, d) = (X_{t-1}(\gamma, d)I_{1t}(\gamma, d), X_{t-1}I_{2t}(\gamma, d))'$, and X is the $T \times k$ matrix whose ith row is Y'_{t-1}

To solve this equation, we sovle it by two steps:

First, for given (d, γ), minimize over α. And the solution can be written as:

$$\hat{\alpha}(\gamma, d) = [X(\gamma, d)'X(\gamma, d)]'[X(\gamma, d)]$$

Then, find:

$$(\hat{\gamma}, \hat{d}) = \underset{\gamma, d}{argmin} S(\gamma, d|\hat{\alpha}(\gamma, d))$$

where $\alpha = (\alpha_1', \alpha_2')'$.

And since we are estimating the coefficients, then the parameter we are interested in is $\alpha = (\alpha_1', \alpha_2')'$.

2.4 Result report

```r
setar.model<-setar(Yt_ds,mL=1,mH=1,th = -0.0078577950,thDelay = 0)
```

```
##
##  1 T: Trim not respected:  0.147541 0.852459 from th: -0.007857795
```

```r
summary(setar.model)
```

```
##
## Non linear autoregressive model
##
## SETAR model ( 2 regimes)
## Coefficients:
## Low regime:
##     const.L      phiL.1
## 0.008643731 0.972864401
##
## High regime:
##       const.H        phiH.1
## -2.288056e-05  2.900574e-01
##
## Threshold:
## -Variable: Z(t) = + (1) X(t)
## -Value: -0.007858 (fixed)
## Proportion of points in low regime: 14.75%    High regime: 85.25%
##
## Residuals:
```

```
##         Min          1Q      Median          3Q         Max
## -0.01594482 -0.00565162 -0.00028742  0.00449685  0.02540662
##
## Fit:
## residuals variance = 6.633e-05,  AIC = -1762, MAPE = 128.4%
##
## Coefficient(s):
##
##            Estimate  Std. Error  t value Pr(>|t|)
## const.L  8.6437e-03  5.2402e-03   1.6495 0.100789
## phiL.1   9.7286e-01  3.7421e-01   2.5998 0.010103 *
## const.H -2.2881e-05  6.9202e-04  -0.0331 0.973660
## phiH.1   2.9006e-01  9.1604e-02   3.1664 0.001813 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Threshold
## Variable: Z(t) = + (1) X(t)
##
## Value: -0.007858 (fixed)
```

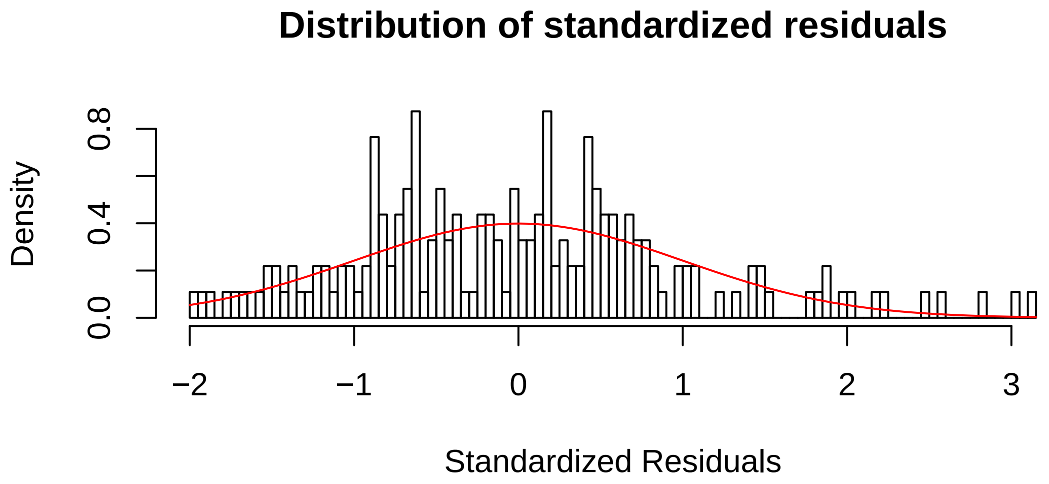
From the OLS, we get the following estimate:

$$\hat{y}_t = \begin{cases} \underset{(5.2402e-03)}{8.6437e-03} + \underset{(3.7421e-01)}{9.7286e-01} Y_{t-1} + \epsilon_t^1 & , Y_t < -0.007858 \\ \underset{(6.9202e-04)}{-2.2881e-05} + \underset{(9.1604e-02)}{2.9006e-01} Y_{t-1} + \epsilon_t^2 & , Y_t \geq -0.007858 \end{cases}$$

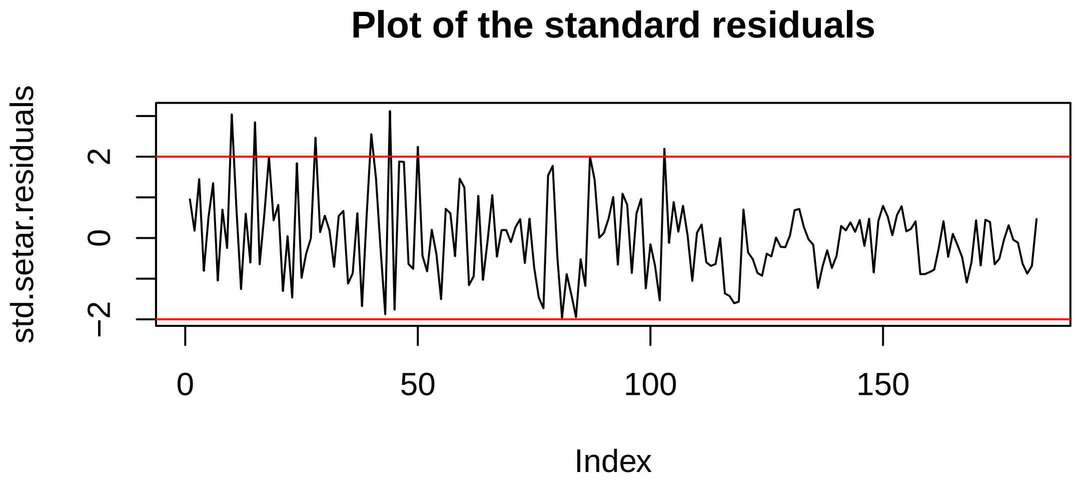
$$\sigma^2 = 6.633 \times 10^{-5}, AIC = -1762, MAPE = 128.4$$

2.5 Diagostic Test

```
setar.residuals <- setar.model$residuals
setar.residuals.sigma.sq <- sum(setar.residuals^2)/length(Yt_ds)

std.setar.residuals <- (setar.residuals - mean(setar.residuals))/sqrt(setar.residuals.si


hist(std.setar.residuals, breaks = 100, freq = F, main = "Distribution of standardized r
curve(dnorm(x),col = "red", add = T)
```



```r
plot(std.setar.residuals, type = 'l', main = 'Plot of the standard residuals')
abline(h=-2, col = 'red')
abline(h=2, col = 'red')
```



We plot the distribution of the standardized residuals, finding that it roughly fits a normal distribution. But there are some outliers. So, we can still do a formal test for the assumption of normality — Jaque-Berra Test:

2.5.1 Jaque-Berra Test

```r
K_3 <- sum(std.setar.residuals^3)/length(Yt_ds)
K_4 <- sum(std.setar.residuals^4)/length(Yt_ds)
```

```
#K_3
#K_4
J_stat <- length(Yt_ds)*((K_3)^2/6+(K_4-3)^2/24)
#J_stat
chisq.crit <- qchisq(0.95,2)
#J_stat < chisq.crit
```

We run a JB test:

\$H_0: \$ residuals are normal.

First, we get the $\hat{K}_3$ is 0.6041875 and $\hat{K}_4$ is 3.4696796.

Also, $JB_{stat} = n(\frac{\hat{K}_3^2}{6} + \frac{(\hat{K}_4-3)^2}{24}) = 12.8858945 \sim \chi^2(2)$

Since $JB_{stat} > JB_{crit}(\alpha = 0.05, df = 6) \approx 6$, we reject the H_0 that residuals are normal at 95% confidence level.

2.5.2 Box-Pierce Test

```
M <- ceiling(sqrt(length(Yt_ds)))
#M
pval <- Box.test(x=setar.residuals, type='Box-Pierce', lag=M)

pval
```

```
##
##  Box-Pierce test
##
## data:  setar.residuals
## X-squared = 27.546, df = 14, p-value = 0.01634
```

We run a Box-Pierce test:

H_0 : there is no joint autocorrelation up to order $M \approx 14$

Test statistics is $Q_{stat} = n\sum_{k=1}^{14} \hat{\rho}_{\epsilon}^2(k) \sim \chi^2(14)$

$p - value = Pr(\chi^2(26) > Q_{stat}) = 0.01634 < 0.05$, we reject the H_0 at 95% confidence level.

WOW, it failed all the diagnostic tests!

2.5.3 Test for over fitting

Let's go back to our original question that whether a SETAR(2) model is adequate compared to SETAR(3) or SETAR(1), which means whether we need to reconsider the number of threshold value we want to add into the model:

Let's denote the $S_m = \hat{e}'_m \hat{e}_m$ as the sum of squared residuals where $\hat{e}_m$ is the least square residuals from model SETAR(m).

The test statistics we are using to reject the null hypothese of SETAR(j) against SETAR(k) is:

$$F_{jk} = \underset{\gamma,d}{max} F_{jk}(\gamma, d) = \underset{\gamma,d}{max} T \frac{S_j - S_k(\gamma, d)}{S_k(\gamma, d)}$$

where F_{jk} asymptoticly follows F_{jk} when γ is identified.

Let's test the model adequacy of SETAR(2) vs SETAR(3):

H_0: SETAR(2) is adequate to SETAR(3)

```
setarTest(Yt_ds,m=1, thDelay = 0, test="2vs3")
```

```
## Test of setar(2) against  setar(3)
##
##         Test Pval
## 2vs3 2.66412    1
```

p-value > 0.05.

So we do not reject the null hypothesis that SETAR(2) is adequate to SETAR(3) at a 95% confidence level

But wait a minute, is that SETAR(2) better than the SETAR(1) which is linear AR in this case?

Now, let's test for the over fitting of SETAR(1) vs SETAR(2):

H_0: SETAR(1) is adequate to SETAR(2)

```
setarTest(Yt_ds, m =1, thDelay = 0, test="1vs")
```

```
## Test of linearity against setar(2) and setar(3)
##
##          Test Pval
## 1vs2 7.004008  0.5
## 1vs3 9.770092  0.7
```

p-value > 0.05.

So we do not reject the null hypothesis that SETAR(1) is more adequate to SETAR(2), which means that just an AR(p) model is better than SETAR model in this case.

This means that: We can just use the first section of the project to forecast, which is using ARIMA model to forecast the GDP growth rate. LOL. Thanks for your reading.