Homework 2

Vladislav Zakatov

23 October 2015

This document has been created as part of the second homework assignment in Machine Learning at CMF.

Initially, the working directory, system locale are set and the required packages are loaded.

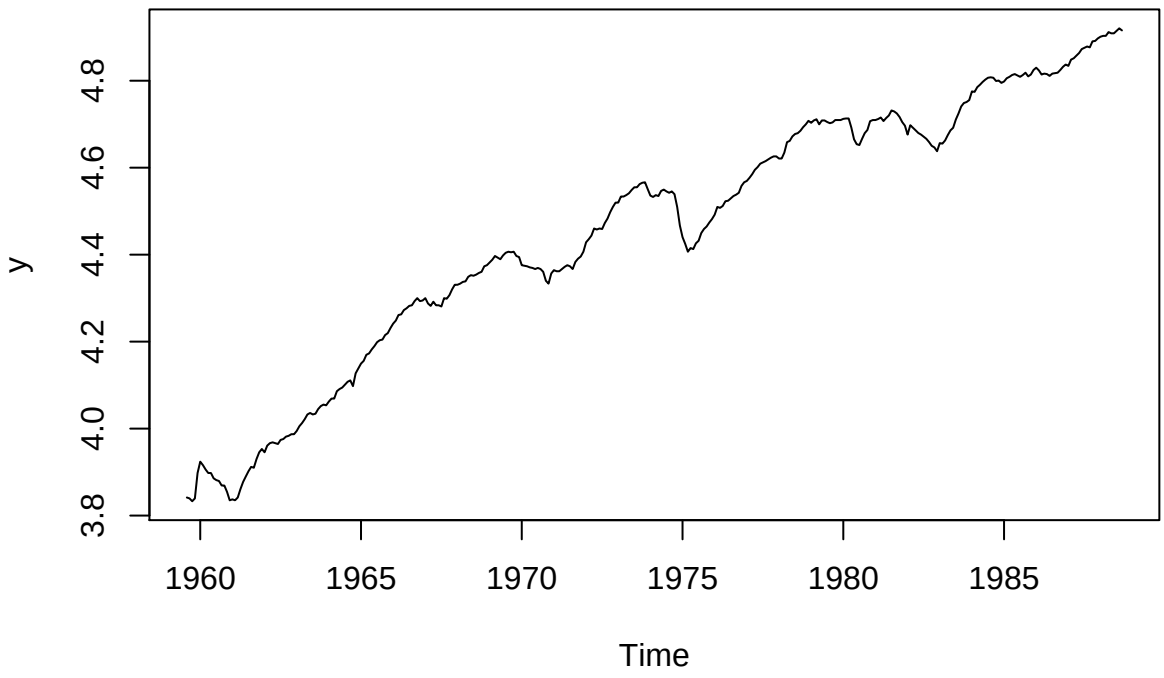
```
##### Initialisation #####
setwd("~/CMF/Courses/Machine Learning/2. Time series decomposition, seasonal, trend & cycle/Homework 2")
library(ggplot2)
library(stats)
library(mFilter)
library(forecast)
Sys.setlocale("LC_ALL","English")
```

The data from csv file is then loaded into R and the logarithm of the series is taken in order to work with an additive model.

```
tmp = read.csv("Data/prod_indx_train.csv", header = TRUE)[,1]
y <- ts(log(tmp), start = c(1959, 8), freq = 12)
```

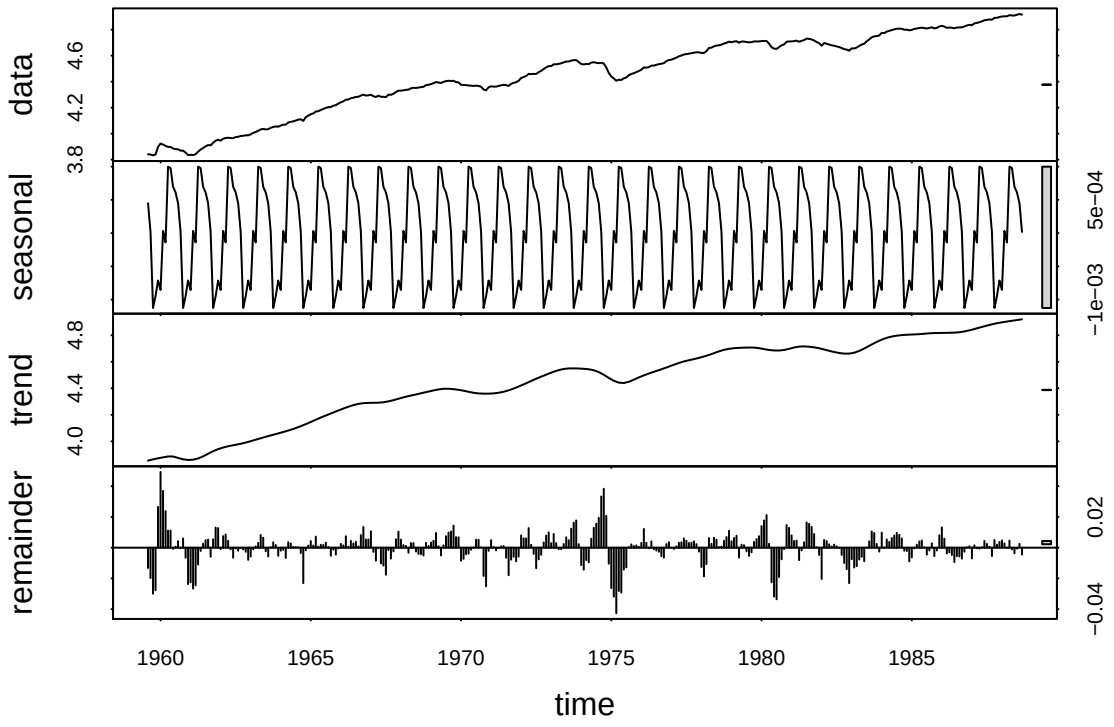
A graph of the log of the series is represented below.

```
##### Time series decompostion #####
plot(y)
```



Time series is then decomposed into trend, seasonal and remainder components using STL procedure.

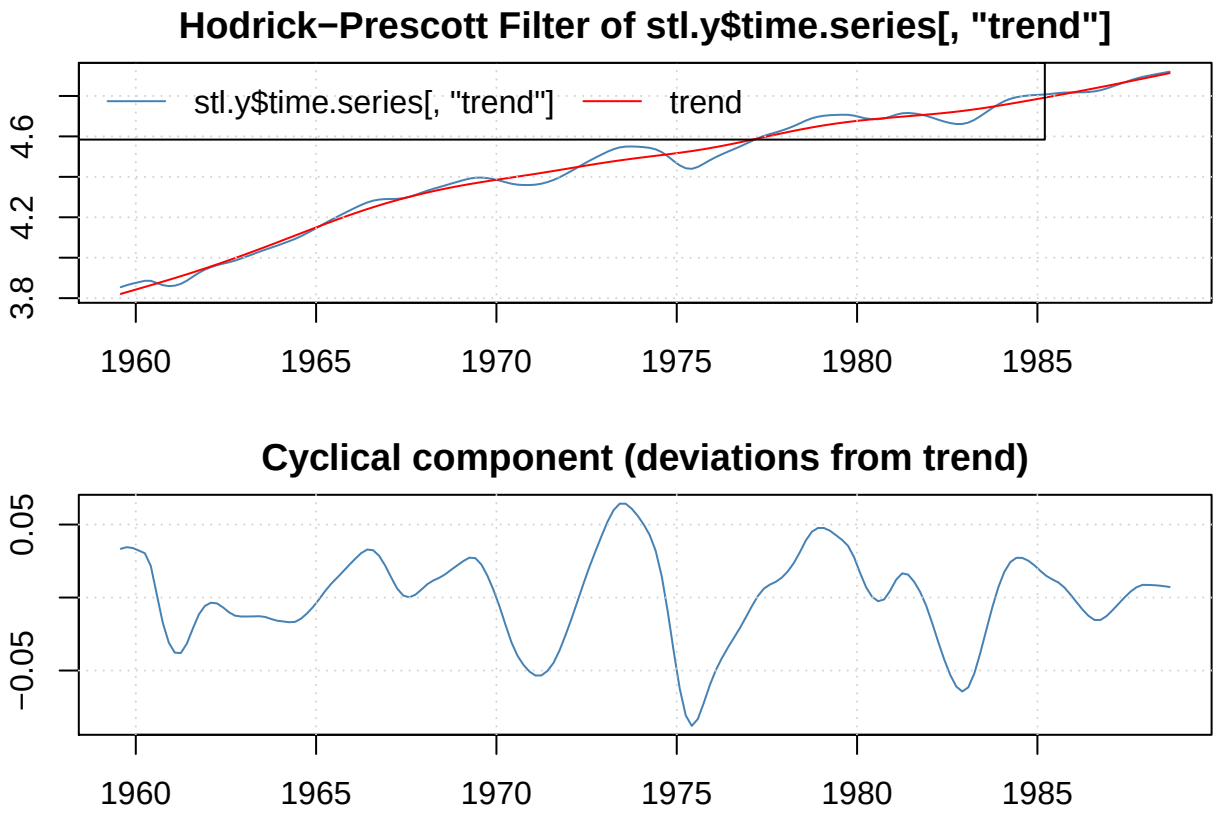
```
## decompose ts into trend, seasonal and remainder data
stl.y <- stl(y, s.window="periodic", robust=FALSE)
plot(stl.y)
```




```
ts = stl.y$time.series
```

The trend component from the STL procedure includes the trend itself and the cycle which we want now to separate using the Hodrick-Prescott filter. We set lambda parameter to 129600 as our data is monthly.

```
## separate trend from cycles
hp.trend <- hpfilter(stl.y$time.series[,"trend"], type = "lambda", freq = 129600, drift = FALSE)
plot(hp.trend)
```



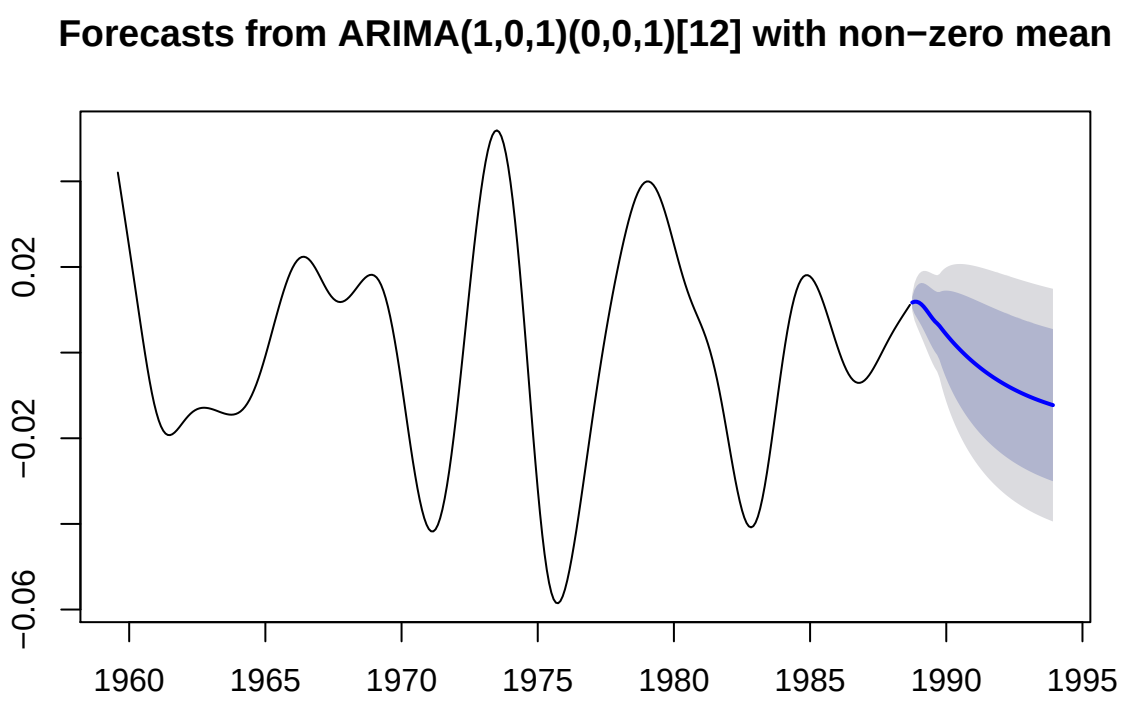
We apply the Hodrick-Prescott filter again to the cycle component with a lower lambda to smoothen it more.

```
## repeat H-P for smoother cycle
cycle <- hpfilter(hp.trend$cycle,type="lambda",freq=1600)$trend
plot(cycle)
```

We now want to predcit new 63 values for our decomposed series.

We start with applying the ARIMA model to the cycle component.

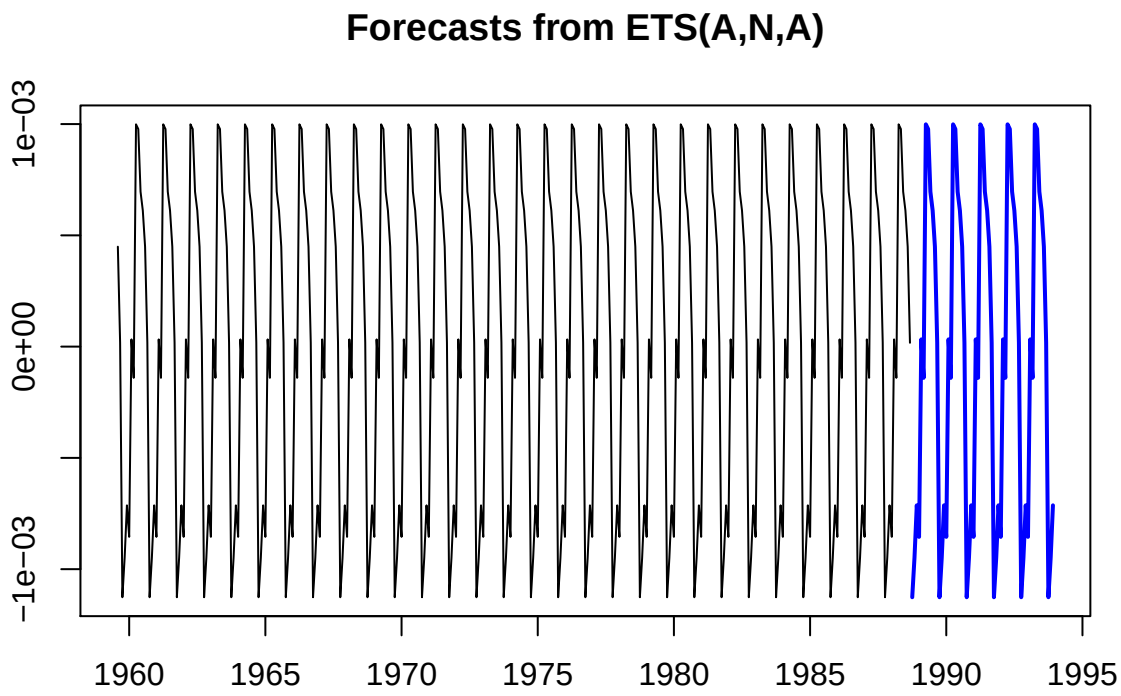
```
## use ARIMA to predict cycle
fit = auto.arima(cycle)
pred.cycle = forecast(fit,63)
plot(pred.cycle)
```




```
res.cycle = c(cycle,pred.cycle$mean)
res.cycle <- ts(res.cycle, start = c(1959, 8), freq = 12)
```

We now predict the next values for the seasonal component.

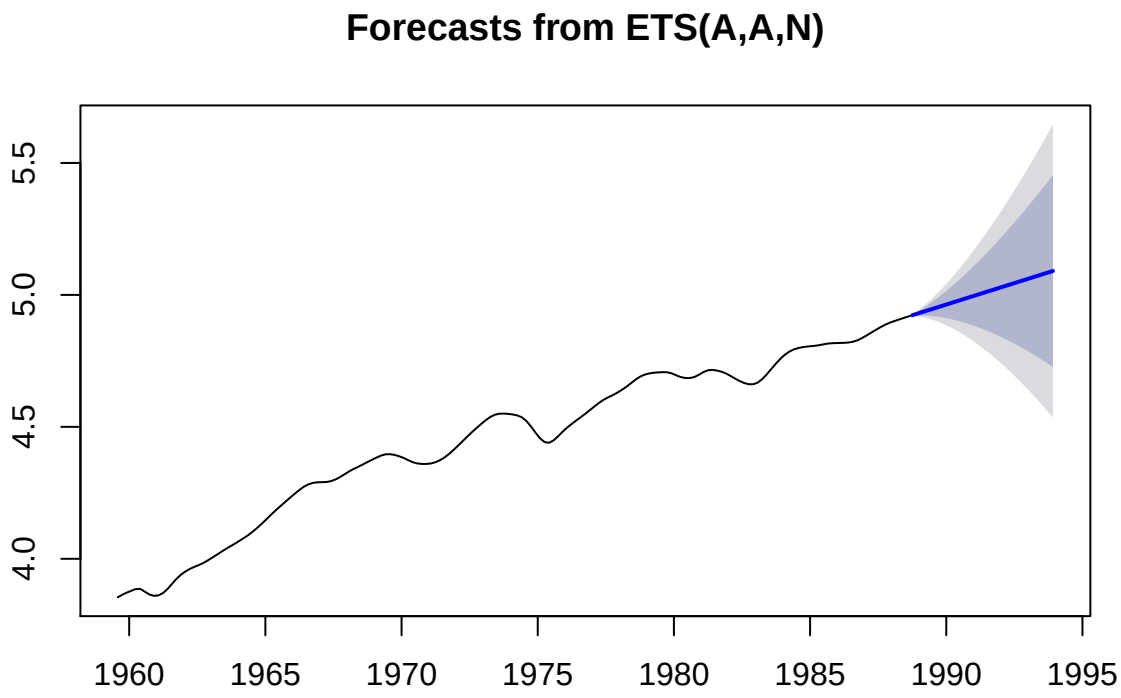
```
## predict seasonal component
pred.season = forecast(ts[,1],63)
plot(pred.season)
```




```
res.season = c(ts[,1],pred.season$mean)
res.season <- ts(res.season, start = c(1959, 8), freq = 12)
```

And finally we get new predicted values for the trend.

```
## predict cycle
pred.trend = forecast(ts[,2],63)
plot(pred.trend)
```

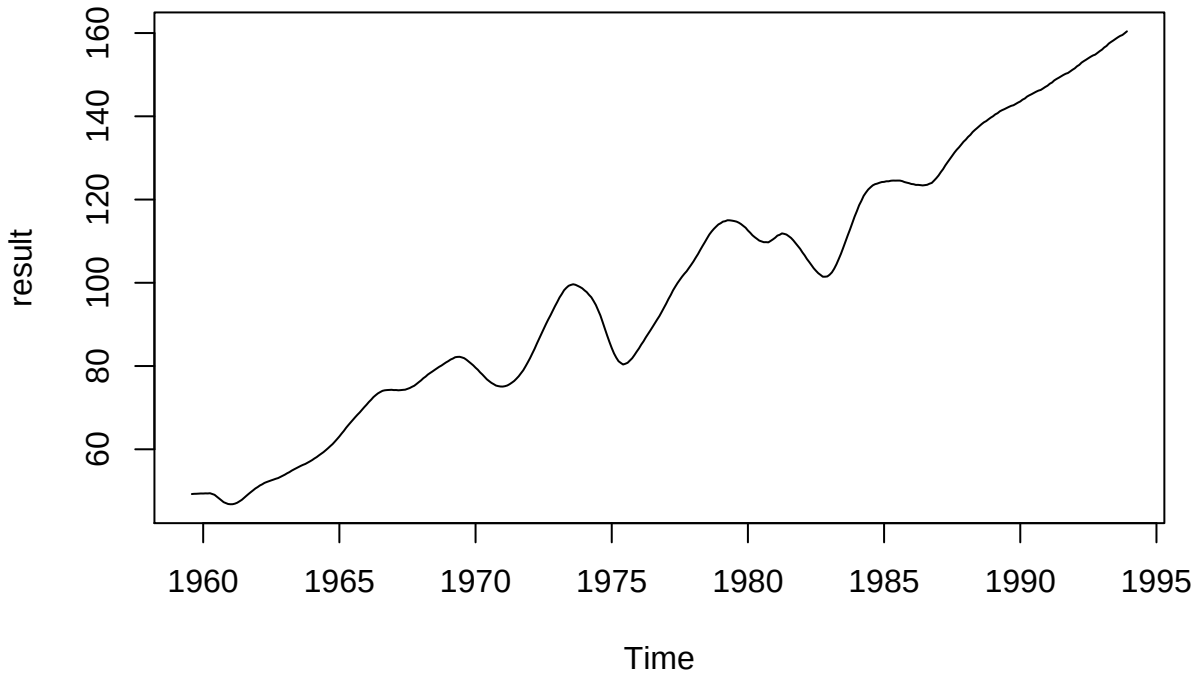



```
res.trend = c(ts[,2],pred.trend$mean)
res.trend <- ts(res.trend, start = c(1959, 8), freq = 12)
```

Now we combine all the components and take the exponent of the series in order to get back to the original scale.

```
##### Combine decompositions #####
result = res.trend + res.season + res.cycle
result = exp(result)

plot(result)
```


result
60
80
100
120
140
160
1960
1965
1970
1975
1980
1985
1990
1995
Time