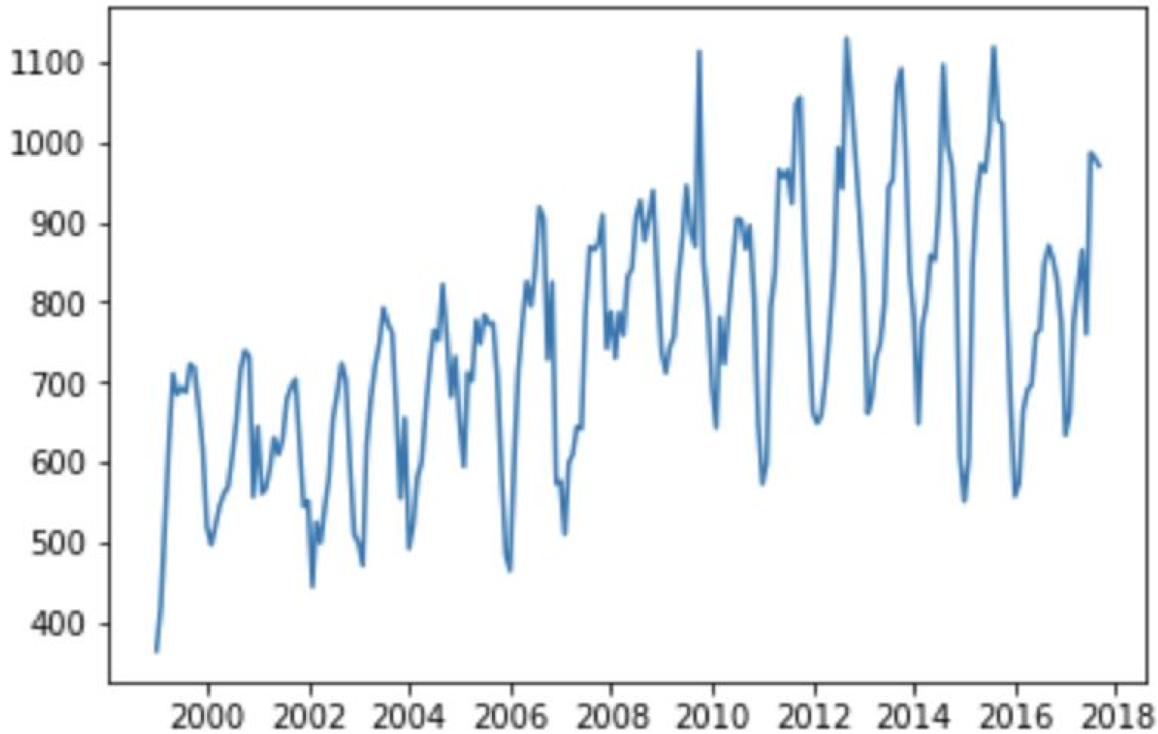
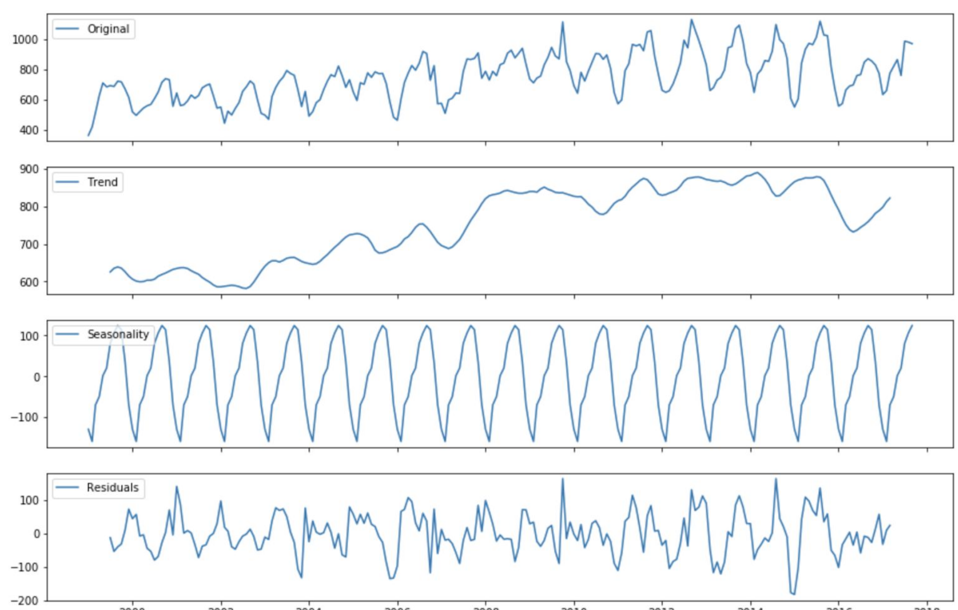
After reading and preparing the data we can plot the data:



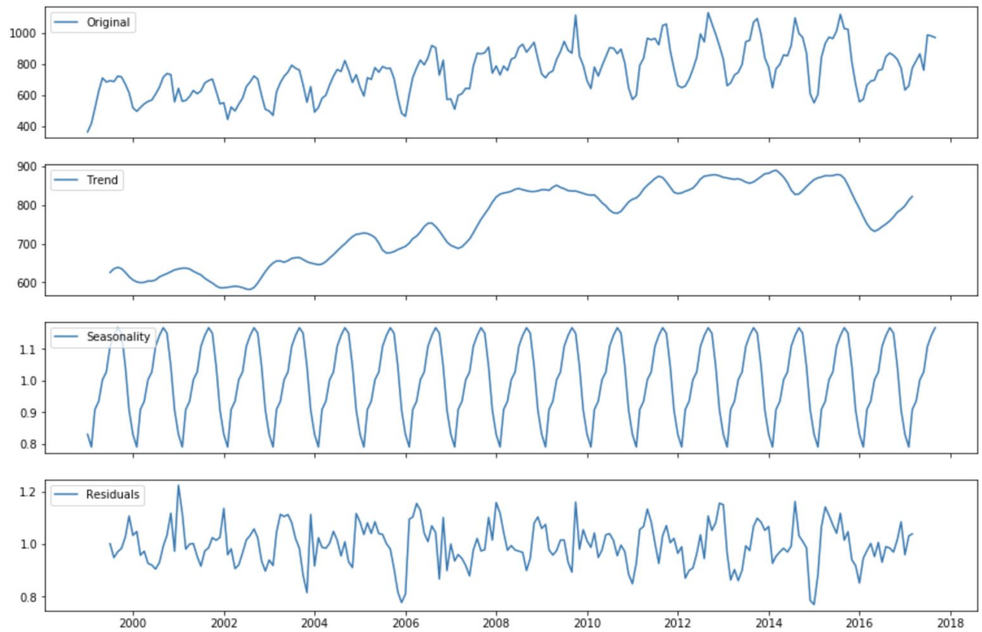
We can see that there is a little linear trend over years and our data is quite volatile, so we can expect seasonality.

1. **Additive VS Multiplicative**

Additive plots:



Multiplicative Plots:



Based on the plots we can assume that our data has an additive structure because we see not only trend but also additive seasonality, so let's check the summary statistics further to see if it's true.

I splitted the data into chunks and calculated mean and variance over those chunks.

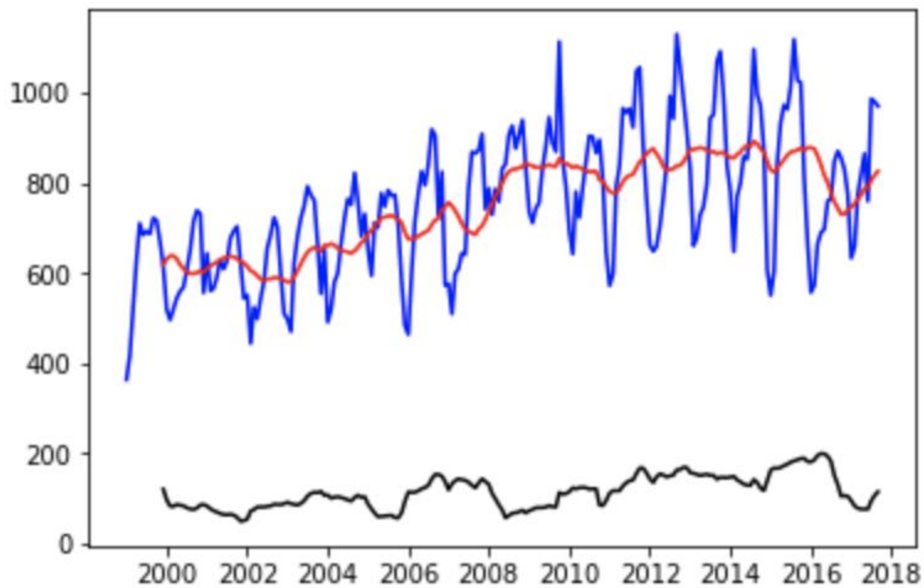
Chunk	Mean	Variance
0	613.435	8041.81
1	615.517	9166.44
2	721.078	11426.4
3	798.222	11663.8
4	814.645	18521.1
5	872.707	21944.4
6	822.625	21193.0

Actually, we see not so rapid increase in mean over chunks, which means that the production volume was increasing more or less gradually from year to year. Variance is constantly increasing in our data which can point to some heteroscedasticity which we would like to deal with.

2. **Descriptive part (Moving Average, Exponential Smoothing, Differencing, Stationarity)**

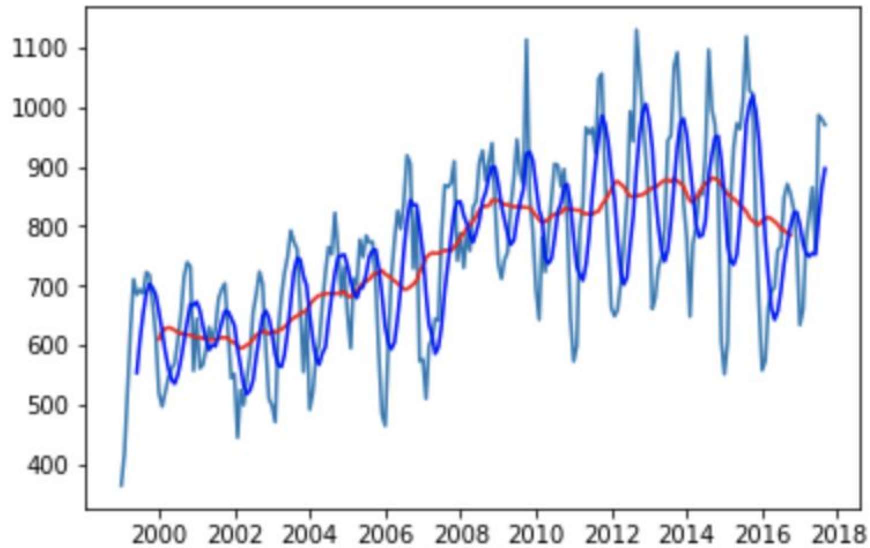
Let`s check if our raw data is stationary. In addition to checking summary statistics over chunks where we already found increasing variance, we can use Augmented Dickey-Fuller test which shows us the following results:

```
Test Statistic                  -1.530376
p-value                          0.518435
#Lags Used                      15.000000
Number of Observations Used    209.000000
Critical Value (1%)             -3.462032
Critical Value (5%)             -2.875471
Critical Value (10%)            -2.574195
dtype: float64
```



We immediately see a very big p-value which indicates that data is non-stationary because we can't reject the null hypothesis.

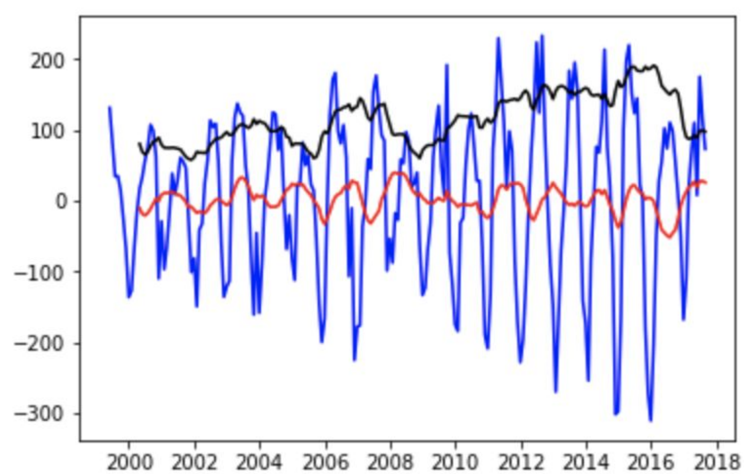
Moving Average:



We see that moving average with the window of length 24 is smoothing the data quite strong which is obviously making it more plain than smoothing with a window of 6. But again we observe some changes further in the data due to not constant variance.

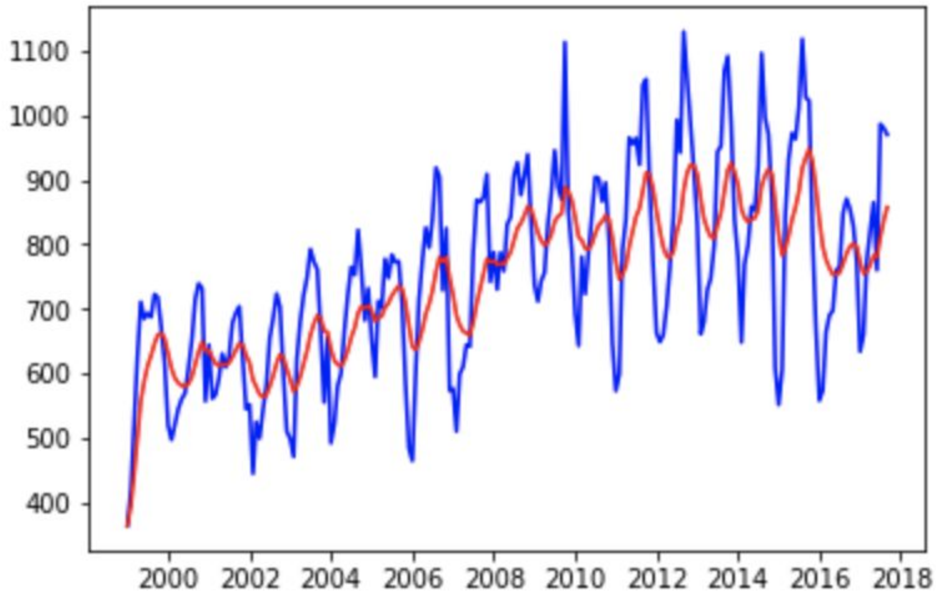
I calculated shocks and run ADF test to see if MA works:

```
Test Statistic                -6.374162e+00
p-value                        2.305736e-08
#Lags Used                     1.300000e+01
Number of Observations Used    2.060000e+02
Critical Value (1%)           -3.462499e+00
Critical Value (5%)           -2.875675e+00
Critical Value (10%)          -2.574304e+00
dtype: float64
```

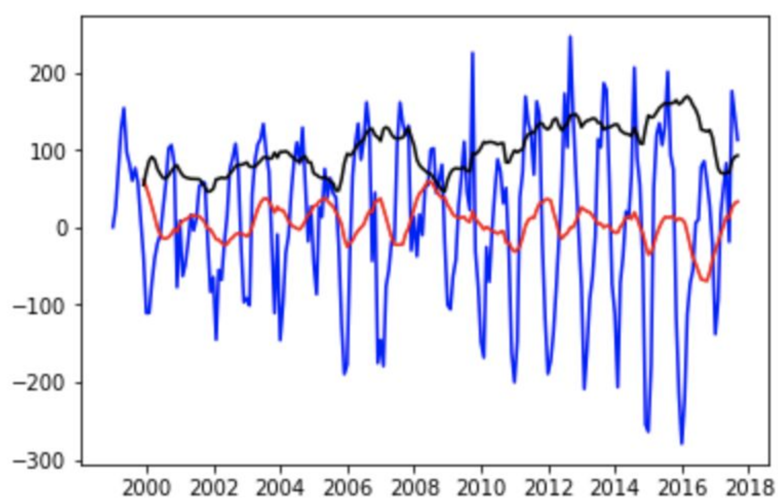


Moving Average Smoothing helps us reach stationarity in data.Based on p-value we can be sure that this approach makes our data more or less stationary.

Exponential Smoothing:




```
Test Statistic                   -5.640578
p-value                           0.000001
#Lags Used                       13.000000
Number of Observations Used     211.000000
Critical Value (1%)              -3.461727
Critical Value (5%)              -2.875337
Critical Value (10%)             -2.574124
dtype: float64
```



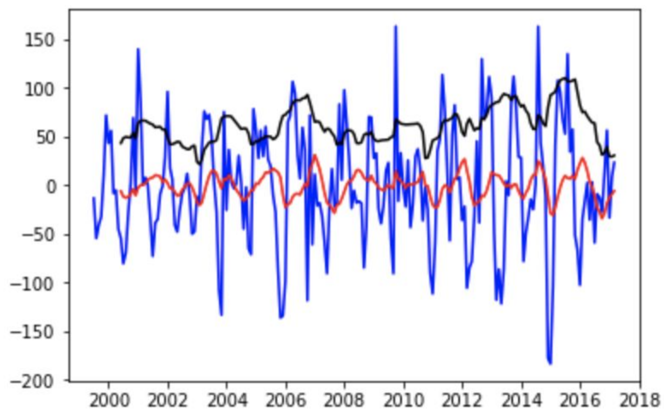
ADF test shows us much better results after the data is smoothed. P-value which is less than 0.05 so theoretically we can reject the null hypothesis and say that our data is stanionary but still p-value is not that small and doesn't imply that we can be actually confident. Also MA shows us better result based on p-value than EWMA.

I think this is because Exponentially weighted MA puts more weight on last results which in our data are more volatile and has increased variance.

3. Decomposition VS Differencing

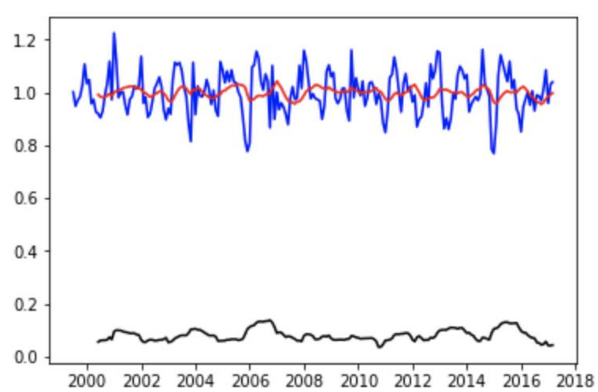
Stationarity Test on additive decomposed data results:

```
Test Statistic                -7.611192e+00
p-value                        2.252834e-11
#Lags Used                     1.400000e+01
Number of Observations Used    1.980000e+02
Critical Value (1%)           -3.463815e+00
Critical Value (5%)           -2.876251e+00
Critical Value (10%)          -2.574611e+00
dtype: float64
```



Stationarity Test on multiplicative decomposition:

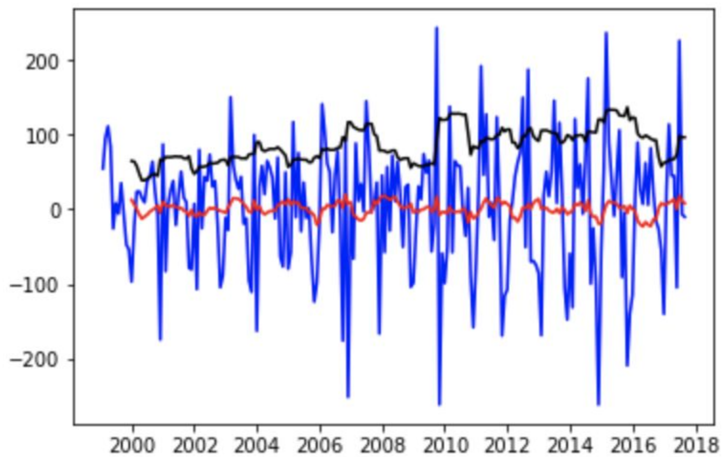
```
Test Statistic                -7.465135e+00
p-value                        5.228626e-11
#Lags Used                     1.200000e+01
Number of Observations Used    2.000000e+02
Critical Value (1%)           -3.463476e+00
Critical Value (5%)           -2.876102e+00
Critical Value (10%)          -2.574532e+00
dtype: float64
```



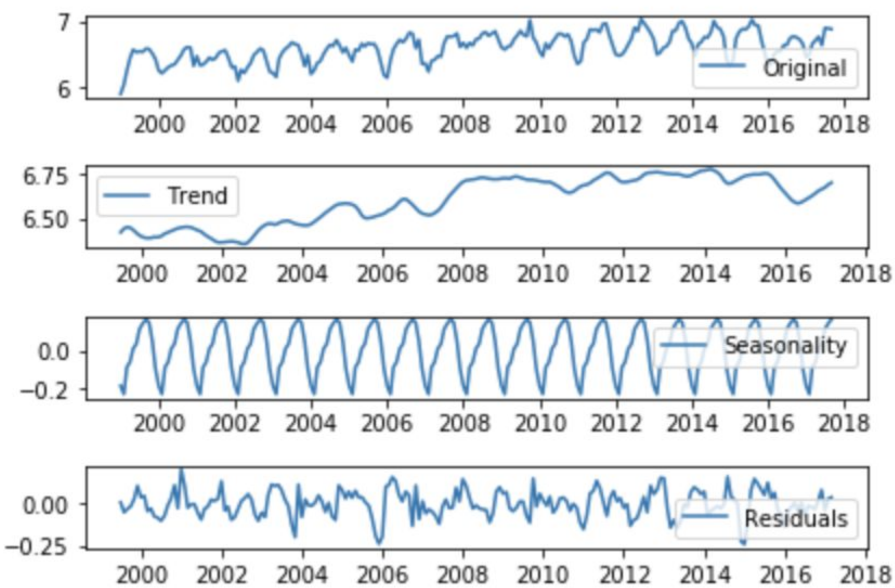
Both decompositions make our data more stationary, but dditive decompositin works better based on summary statistics of ADF test. Multiplicative decomposition makes standard deviation really small, so our series doesn't look so volatile.

Differencing

```
Test Statistic                  -6.079618e+00
p-value                          1.100212e-07
#Lags Used                       1.400000e+01
Number of Observations Used      2.090000e+02
Critical Value (1%)             -3.462032e+00
Critical Value (5%)             -2.875471e+00
Critical Value (10%)            -2.574195e+00
dtype: float64
```

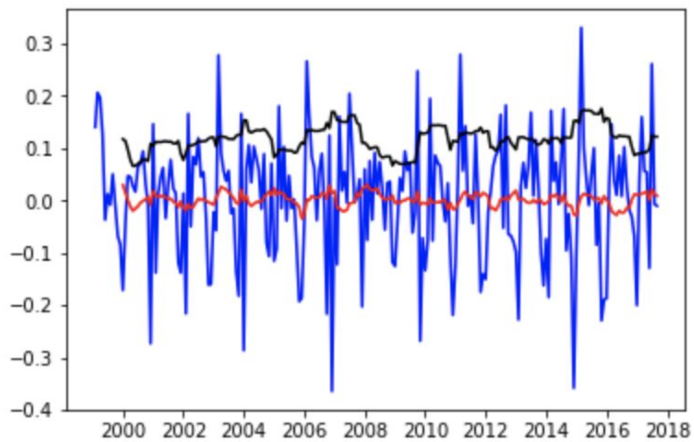


Logarithmic data with additive decomposition:



Logarithmic data with differencing:

```
Test Statistic                -6.122430e+00
p-value                        8.789786e-08
#Lags Used                     1.400000e+01
Number of Observations Used    2.090000e+02
Critical Value (1%)           -3.462032e+00
Critical Value (5%)           -2.875471e+00
Critical Value (10%)          -2.574195e+00
dtype: float64
```



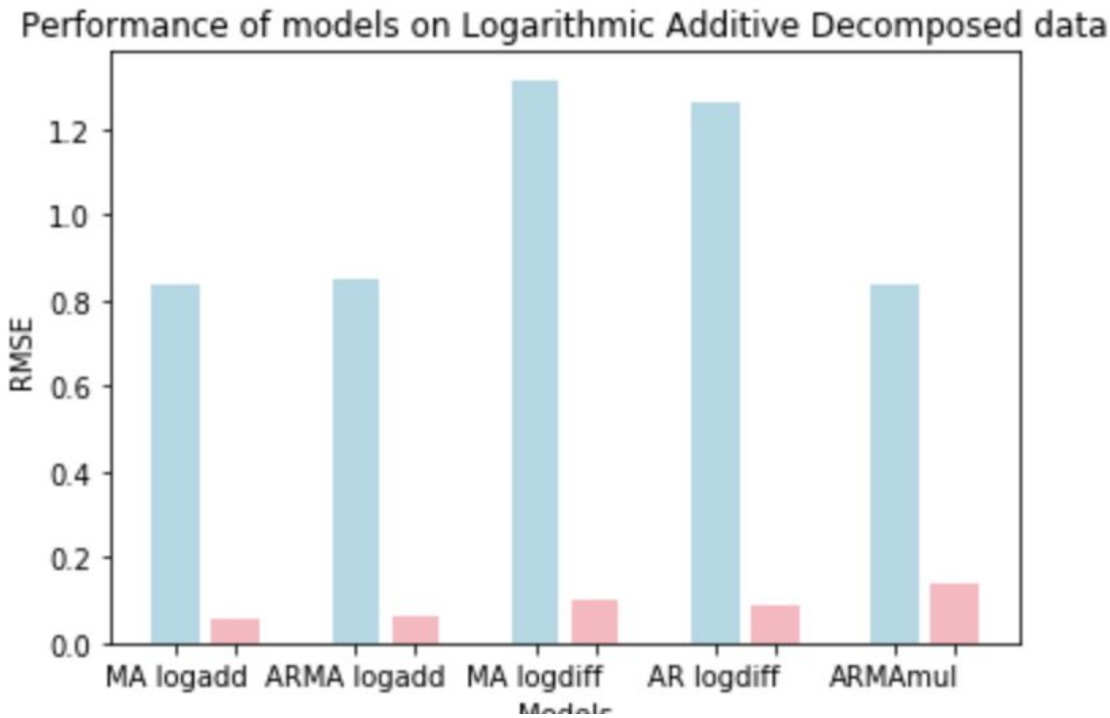
After this I have divided my data into 4 hypotheses on which we will train the models: Additive Decomposed data, Differenced data, Logarithmic data with additive decomposition and logarithmic data with differencing and split these data into train, validation and test sets.

MODELS:

Moving Average and Exponential Smoothing models did not perform very well, so I moved on.
I considered AR, MA, ARMA and ARIMA models for all 4 hypotheses.
Firstly I fitted models based on ACF and PACF, but to get even better results I optimized it using AIC. Optimizing process is commented in the Jupyter Notebook because it takes time, so if You want to check it - just remove comments and run.

Here I will include only final results, models I chose for testing:

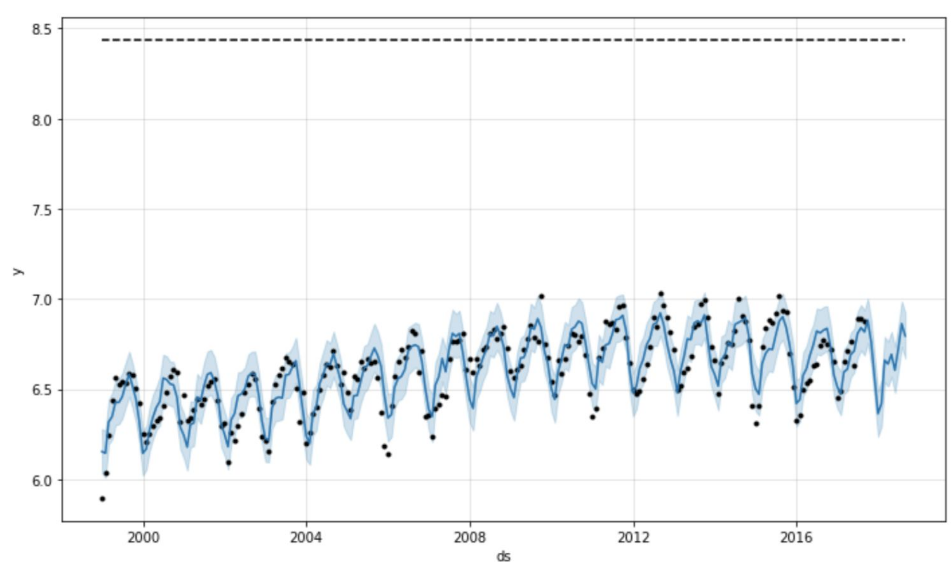
1. MA model on logarithmic additive data
2. ARMA model on logarithmic additive data
3. MA model on logarithmic differenced data
4. AR model on logarithmic differenced data
5. ARMA model on multiplicative

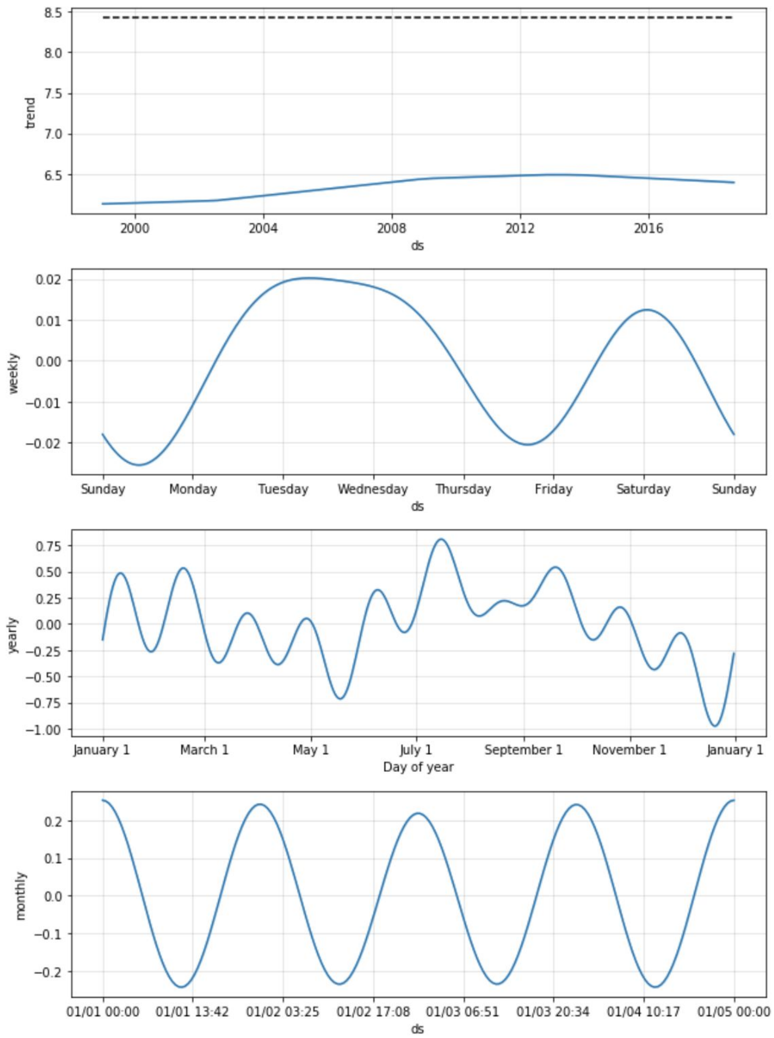


This is a barchart which describes the performance of models based on their RMSEs. In most cased ARMA does better than MA and AR and there is no wonder because it takes into account both factors of AR and MA models.

Facebook Prophet

I trained FP model on original data without decomposition or logarithms because it gave much better results. So, after setting a required dataset and training the model I obtained the following forecast:

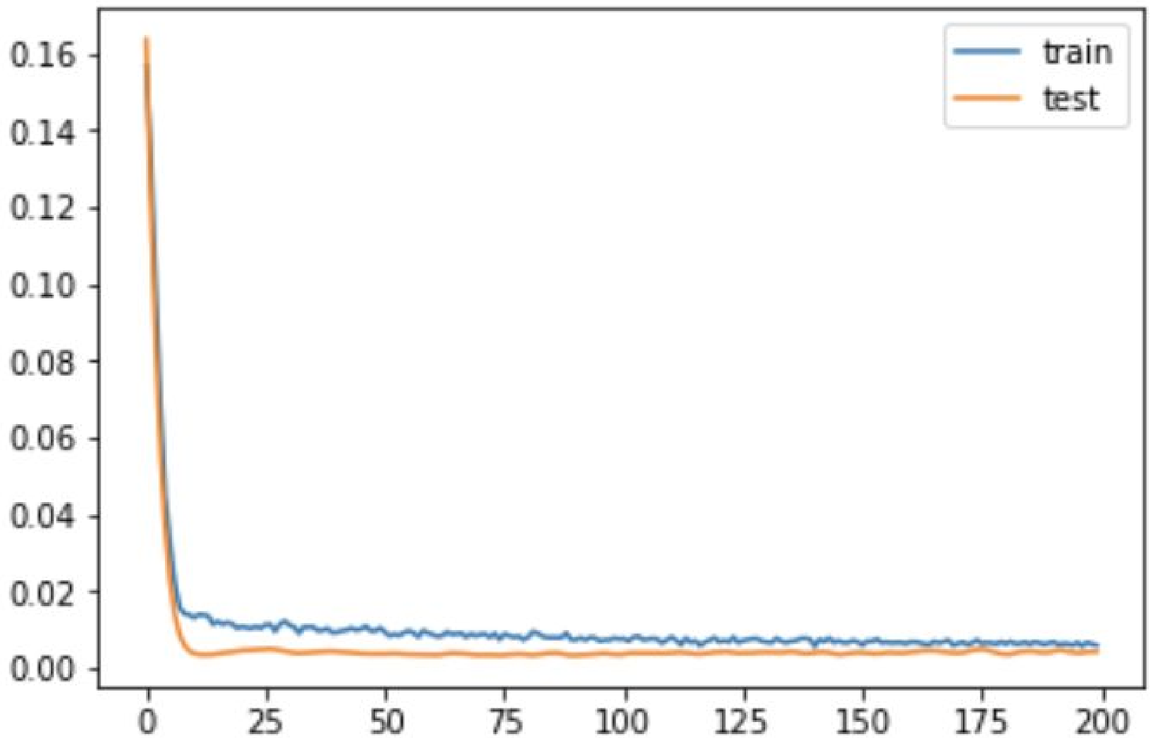




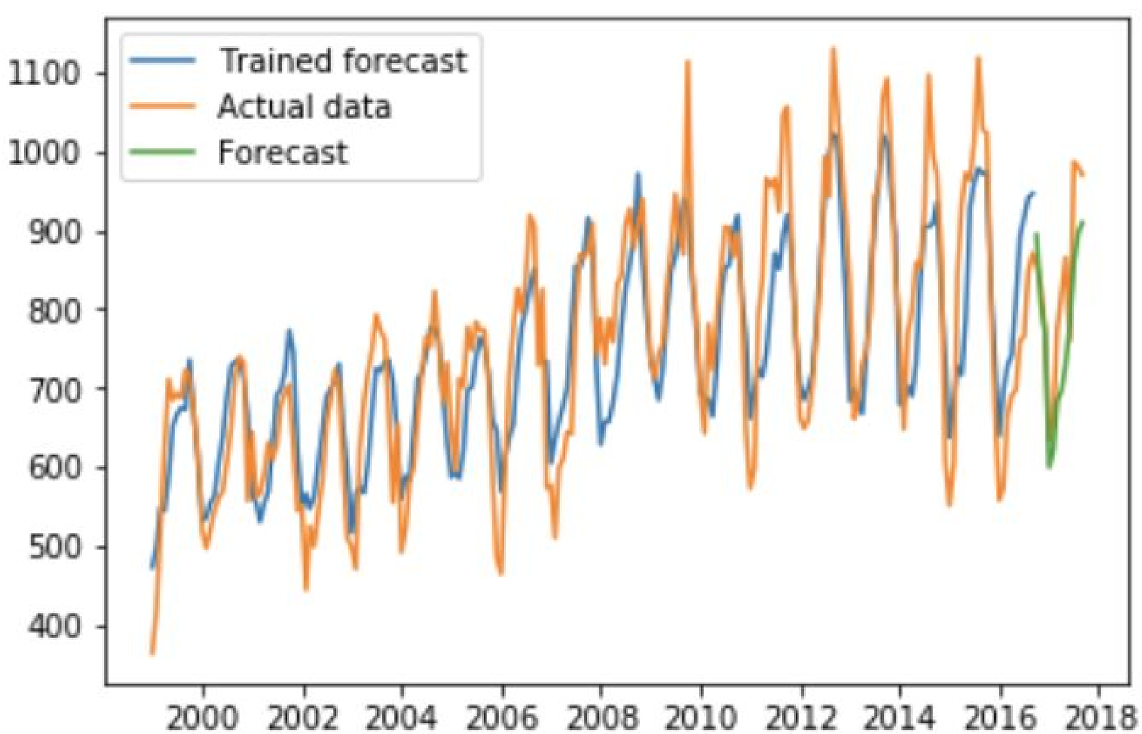
FP model follows the data accurately and make accurate predictions, if we analyze the components we observe that in this region production was quite stable over the years but there exist a yearly seasonality which shows us that during cold months starting from October will the end of January production is lower that during spring and summer.

Deep Learning methods

For deep learning models I tried multiple variants with different layers and loss functions and the best performing was mean squared logarithmic error. T



The model converges really good and has training RMSE: 0.005 and testing RMSE: 0.004



Models which I included in the report are best from all the rest which I tried on different datasets, but to conclude the most accurate ones if choosing from AR, MA, ARMA and ARIMA is ARMA which is the best with all kinds of data and decomposition, however they are poor for predictions, can only be user with short term predictions. For future forecast better choice is FP or Deep Learning but they require mostly raw data without logarithms.