Project 3 - Forecasting U.S. House Prices

Aliaksandr Panko

January 30, 2018

Contents

1 Project Objectives

Forecast Case-Shiller U.S. National Home Price Index using an ARMA model based on the historical data. Test the model and suggest exogenous variables that can be introduced which can improve the forecasts.

2 Case-Shiller U.S. National Home Price Index

The S&P/Chase-Shiller U.S. National Home Price Index is an index that measures the change in value of the U.S. residential housing market. It tracks the growth in value of real estate by following the purchase price and resale value of homes that have undergone a minimum of two arm's-length transactions.

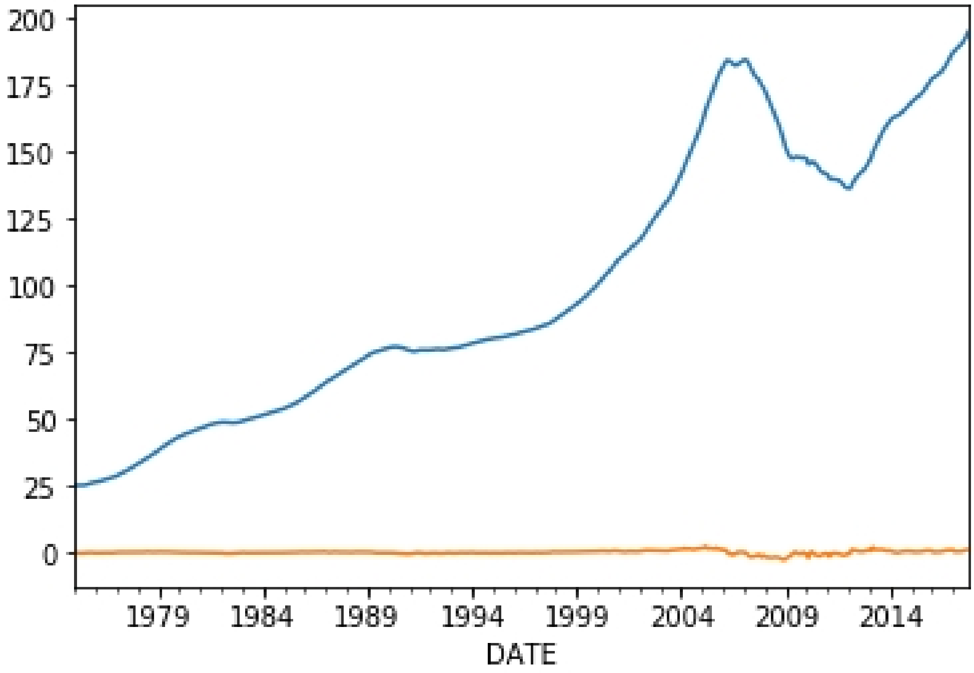
3 Implementation Details

3.1 Data Download

The Case-Shiller U.S. National Home Price Index historical monthly data (seasonally adjusted series) from January 1975 – current date are downloaded from The FRED Database. For this purposes *Pandas DataReader* object was used:

```
from pandas_datareader import data as pdr
pdr.DataReader()
```

To have a brief look I plotted the data (it is in blue):



3.2 Augmented Dickey-Fuller Test

Frankly speaking, it is obvious that the data is not stationary but to prove it ADF test was implemented.

```
from statsmodels.tsa.stattools import adfuller
adfuller()
```

Test results as expected proves the claim:

- ADF Statistic: -0.258178
- p-value: 0.931222

3.3 Reduce non-stationarity

In this project Differencing method is used. In this technique, the difference of the observation at a particular instant with that at the previous instant is taken.

```
DataFrame.shift()
```

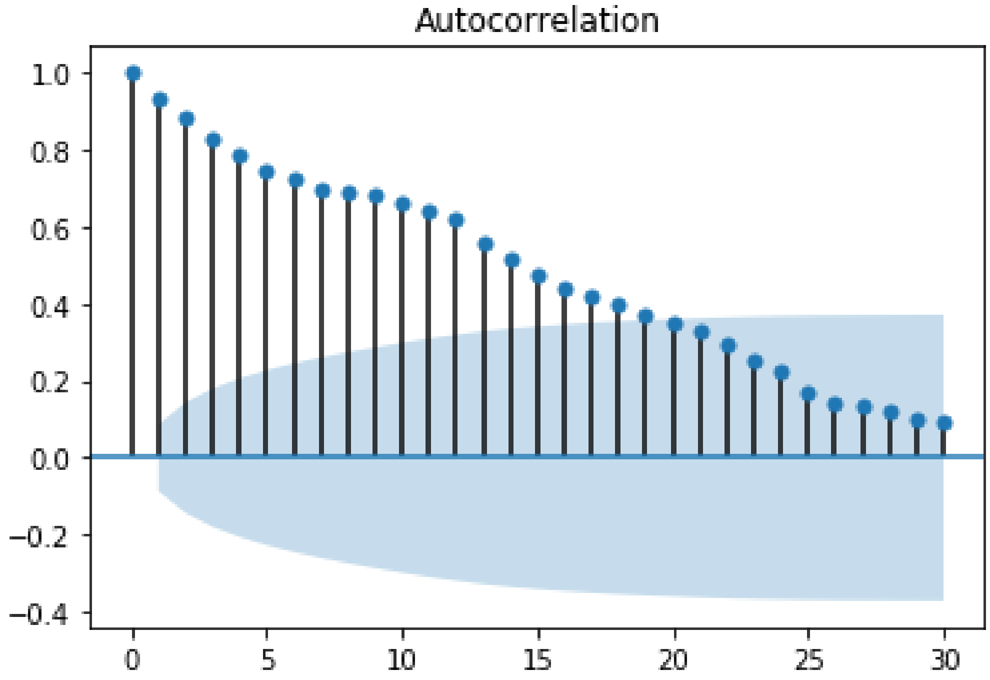
After that Augmented Dickey-Fuller Test passed:

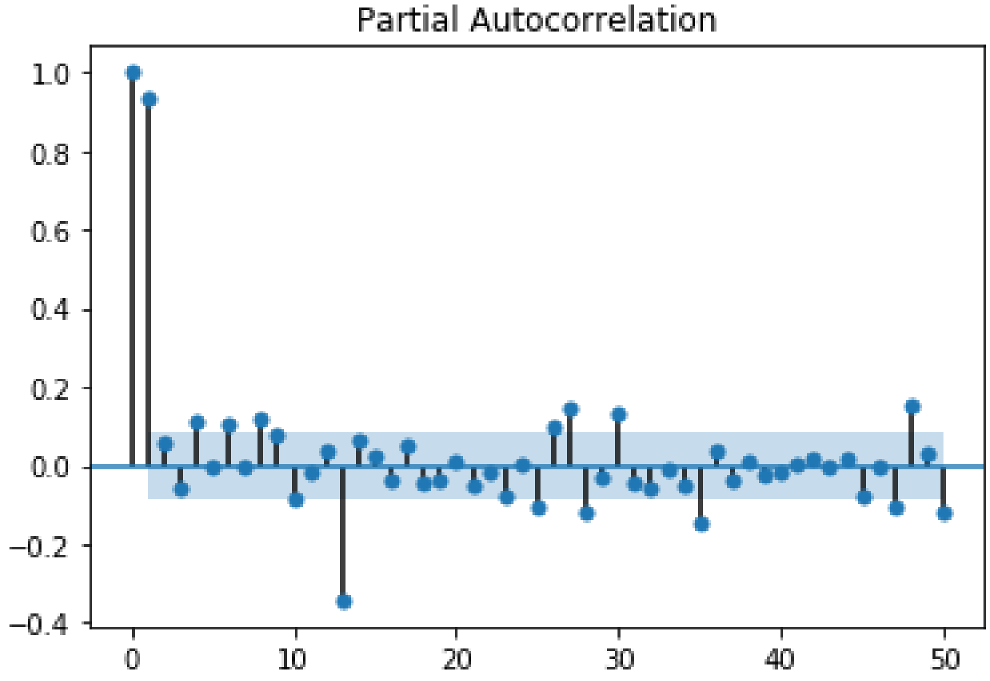
- ADF Statistic: -3.202595
- p-value: 0.019835

Brown line on the previous plot shows the modified data.

3.4 ARIMA model parameters estimation

The parameters of the model are estimated using Box-Jenkins methodology. On this step autocorrelation function graph and partial autocorrelation function graph are plotted:



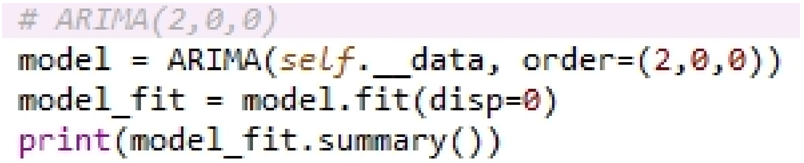


It's clear that ACF decays exponentially and PACF has 2 significant spikes. Since differencing is not required, the parameters of the model are ARIMA(2,0,0).

4 Forcasting

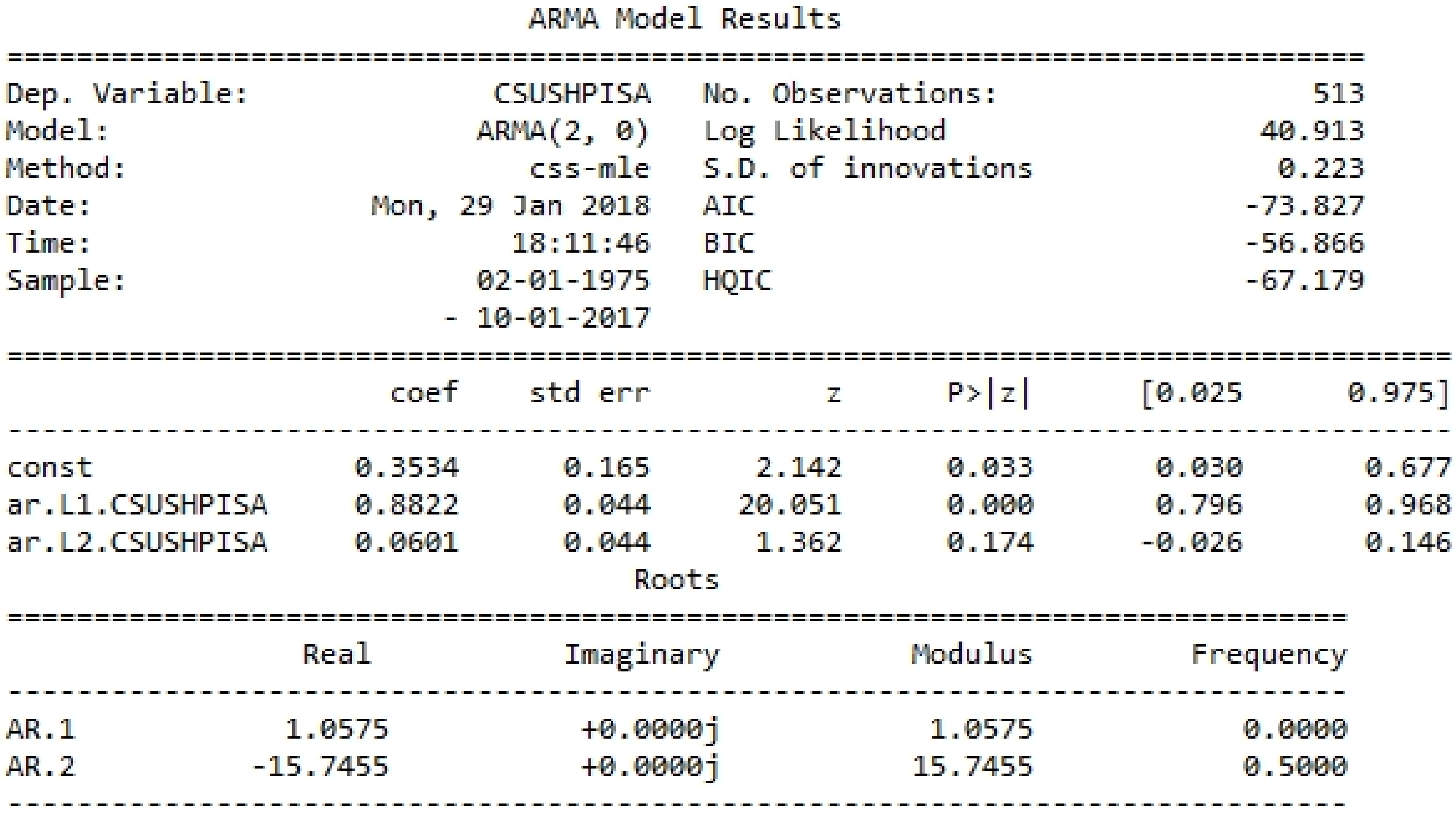
Now ARIMA model with estimated parameters are used to forecast future values.

```
from statsmodels.tsa.arima_model import ARIMA
pdr.DataReader()
```



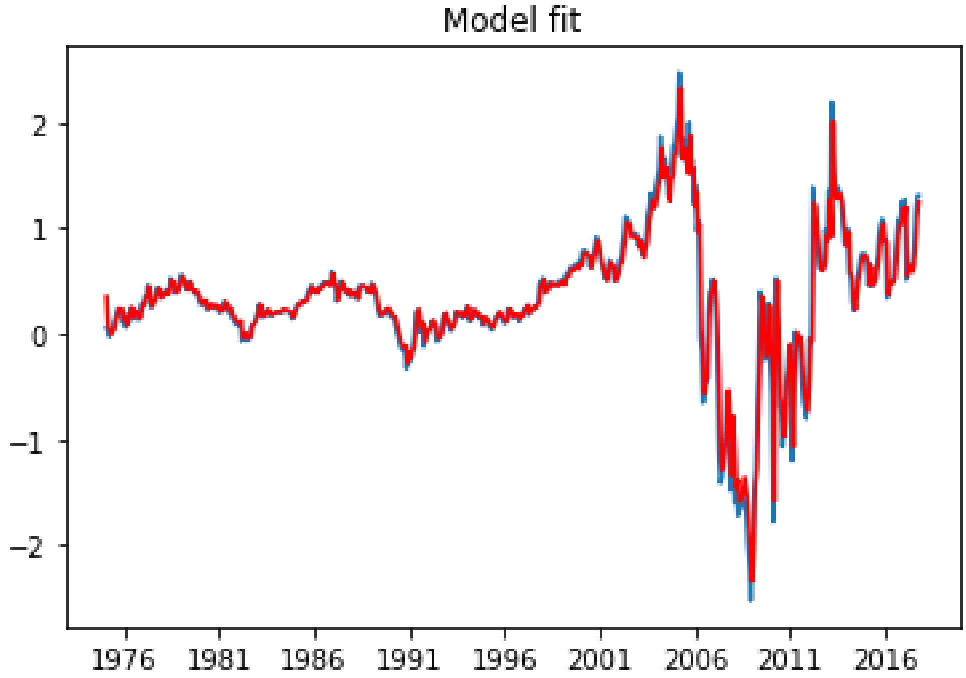
```
# ARIMA(2,0,0)
model = ARIMA(self.__data, order=(2,0,0))
model_fit = model.fit(disp=0)
print(model_fit.summary())
```

The model gives next results:

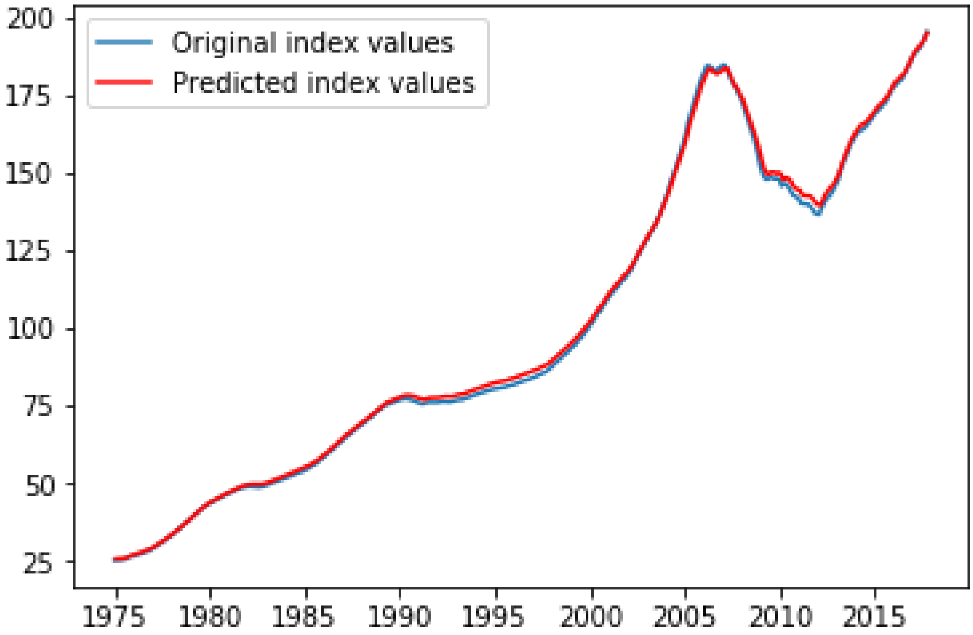


```
                              ARMA Model Results
==============================================================================
Dep. Variable:              CSUSHPISA   No. Observations:                  513
Model:                     ARMA(2, 0)   Log Likelihood                  40.913
Method:                       css-mle   S.D. of innovations              0.223
Date:                Mon, 29 Jan 2018   AIC                            -73.827
Time:                        18:11:46   BIC                            -56.866
Sample:                    02-01-1975   HQIC                           -67.179
                         - 10-01-2017
=================================================================================
                     coef    std err          z      P>|z|      [0.025      0.975]
---------------------------------------------------------------------------------
const              0.3534      0.165      2.142      0.033       0.030       0.677
ar.L1.CSUSHPISA    0.8822      0.044     20.051      0.000       0.796       0.968
ar.L2.CSUSHPISA    0.0601      0.044      1.362      0.174      -0.026       0.146
                                    Roots
=============================================================================
                 Real           Imaginary           Modulus         Frequency
-----------------------------------------------------------------------------
AR.1           1.0575           +0.0000j            1.0575            0.0000
AR.2         -15.7455           +0.0000j           15.7455            0.5000
-----------------------------------------------------------------------------
```

Firstly, using **model_fit.fittedvalues** I get prediction for stationary data:



Then for original data:



The plot shows that historical data are predicted almost perfectly!
Regarding expected values for 1,2 and 3rd month:

1. 196.42472046
2. 197.63067588
3. 198.790293

model_fit.forecast() was used

5 Conclusion

Overall, implemented model describes original data very well. Regarding exogenous variables that can be introduced which can improve the forecasts, in my opinion, they are:

- inflation rate
- salary growth rate
- population growth rate
- interest rate
- unemployment rate
- GDP

6 Additional Information

6.1 ARMA model

- **Autoregressive Moving Average(ARMA)** - mathematical model that is used for stationary time series.
- **time series** is set of pairs: variable value and the time that it is generated.
- **Stationary** time series is a time series that describe a stationary stochastic process.
- **stochastic process** is a collection of random variables that describes a random process. Basically it is a set of pairs: (random variable, time)
- **stationary stochastic process** is a stochastic process that does not change its characteristics over time(random variable mean, variance and so on)

- **Autoregressive process** is a stochastic process used in statistical calculations in which future values are estimated based on a weighted sum of past values. Autoregressive processes are used by investors in technical analysis. Trends, moving averages and regressions take into account past prices in an effort to create forecasts of future price movement.

- **Moving-average (MA) Process** is a process that is used in time series analysis for modeling univariate time series. A MA process shows that the current values of the time series depends on the previous (unobserved) random shocks.

- **univariate time series** is a time series for only 1 variable.

6.2 Augmented Dickey-Fuller Test

The Augmented Dickey-Fuller test can be used to test for a unit root in a univariate process in the presence of serial correlation. The null hypothesis of the Augmented Dickey-Fuller is that there is a unit root, with the alternative that there is no unit root. If the p-value is above a critical size, then we cannot reject that there is a unit root. Unit root existence means non-stationarity.

6.3 Reduce non-stationarity

Though stationarity assumption is taken in many TS models, almost none of practical time series are stationary. Its almost impossible to make a series perfectly stationary, but it possible to make it close to stationary one.

There are 2 major reasons behind non-stationaruty of a TS:

1. **Trend** – varying mean over time. For eg, in this case we saw that on average, the number of passengers was growing over time.

2. **Seasonality** – variations at specific time-frames. eg people might have a tendency to buy cars in a particular month because of pay increment or festivals.

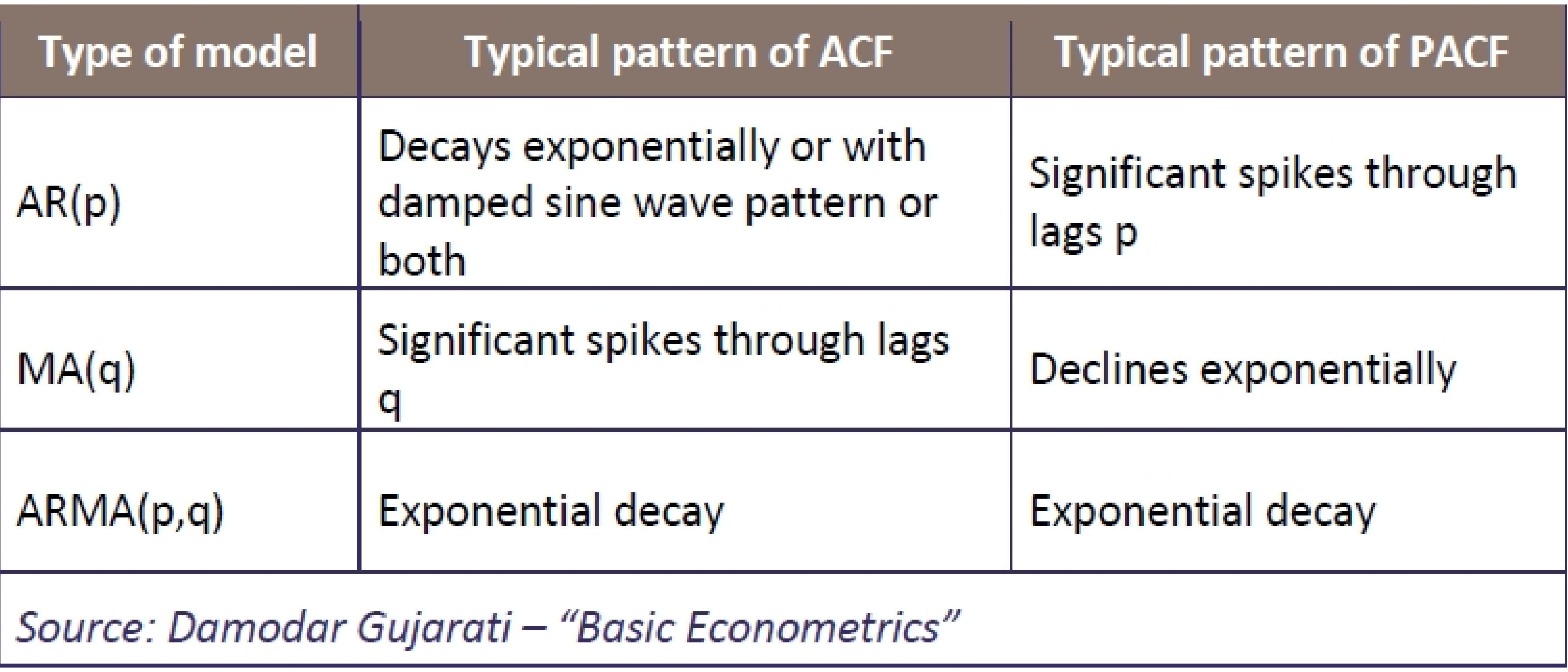
The underlying principle is to model or estimate the trend and seasonality in the series and remove those from the series to get a stationary series. Then statistical forecasting techniques can be implemented on this series. The final step would be to convert the forecasted values into the original scale by applying trend and seasonality constraints back.

There are several possible tricks to use to eliminating trend and seasonality:

- Transformation
- Subtract moving average
- Differencing
- Decomposition

6.4 Box-Jenkins methodology

Box-Jenkins method is applied at ARMA/ARIMA models for finding the best fit. In this project, based on provided patterns and table below, parameters of the model are determined.



Type of model	Typical pattern of ACF	Typical pattern of PACF
AR(p)	Decays exponentially or with damped sine wave pattern or both	Significant spikes through lags p
MA(q)	Significant spikes through lags q	Declines exponentially
ARMA(p,q)	Exponential decay	Exponential decay

Source: Damodar Gujarati – "Basic Econometrics"

ARIMA model has 3 parameters (p,d,q):

1. p - parameter of AR model
2. d - shows the number of differences
3. q - parameter of MA model