



Search

Write



Guide to Time Series Analysis with Python — 2: Moving Average Process



Buse Köseoğlu · Follow

5 min read · Jul 15, 2023



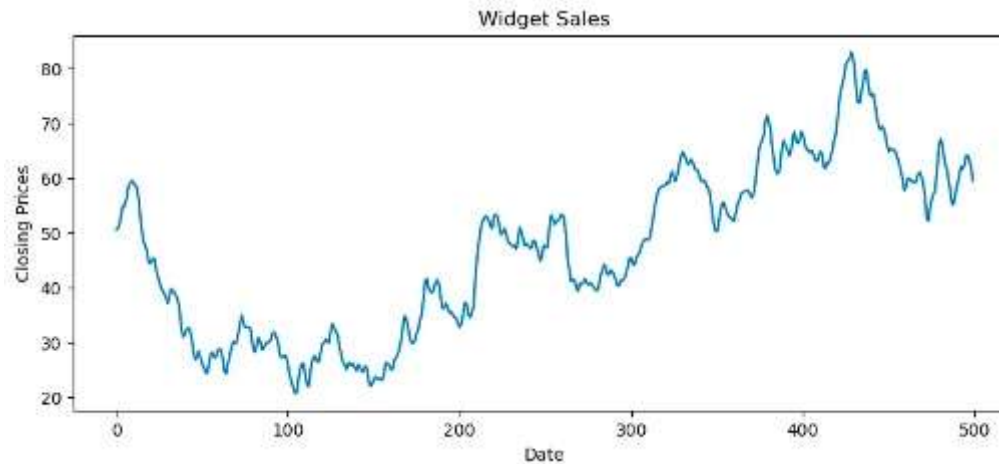
108



In my previous article, I talked about how time series analysis should be done, analysis methods and baseline model creation. If you do not know about these topics, you can first review this article.

1. [Guide to Time Series Analysis with Python — 1: Analysis Techniques and Baseline Model](#)

In this article, we will examine the Moving Average Process, a model used in time series together. We will use widget sales data during these reviews. You can find all analysis codes on [github repo](#).



Moving Average Process (MA)

Moving average process indicates that the current value depends on the current and past error rates. If we examine the formula:

$$y_t = \mu + \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \dots + \theta_q \epsilon_{t-q}$$

- μ : mean of the series
- ϵ_t : Current error value

- ϵ_{t-1} : Historical error value
- Theta-q : The effect of past mistakes on the present

The important parameter we need to determine for MA is q. This parameter determines the number of historical error terms that affect the current value.

MA is used to model univariate time series. So if we're going to use MA we only have the date and the data we want to predict, we can't use any extra arguments.

Specifying the q Parameter

To determine this parameter, we should use the ACF chart that we mentioned in the first article. In an ACF graph, we normally expect coefficients to be non-significant, but often this is not the case. So what should we do if there is a non-significant situation? We have a roadmap to determine this.

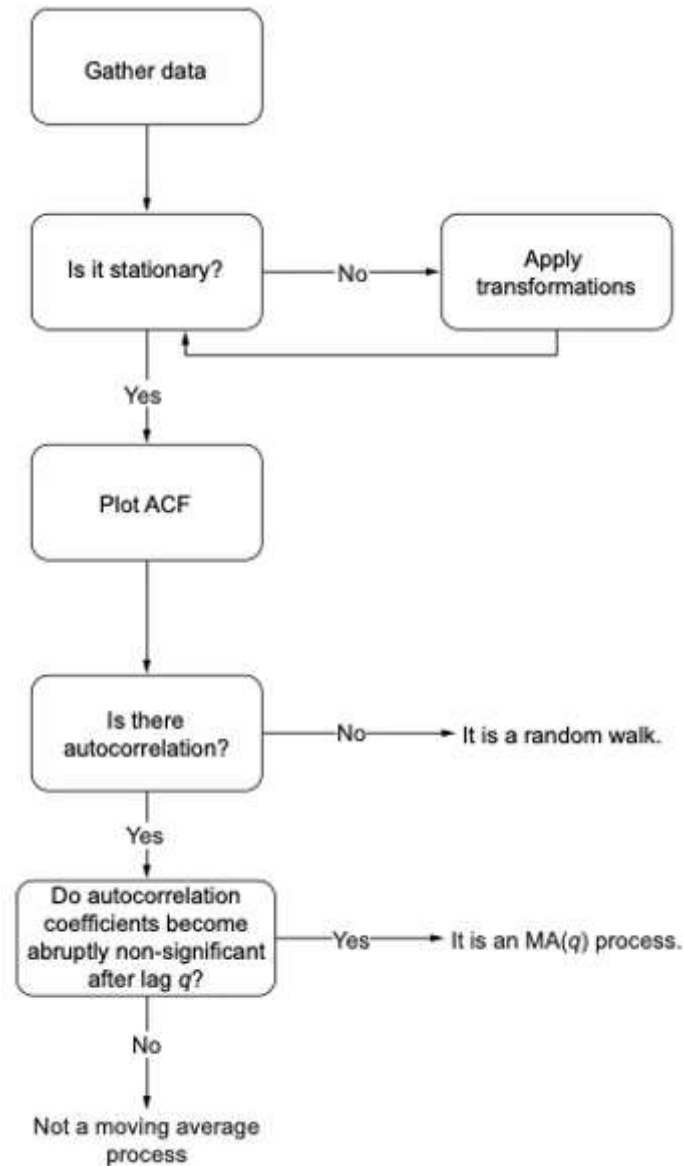


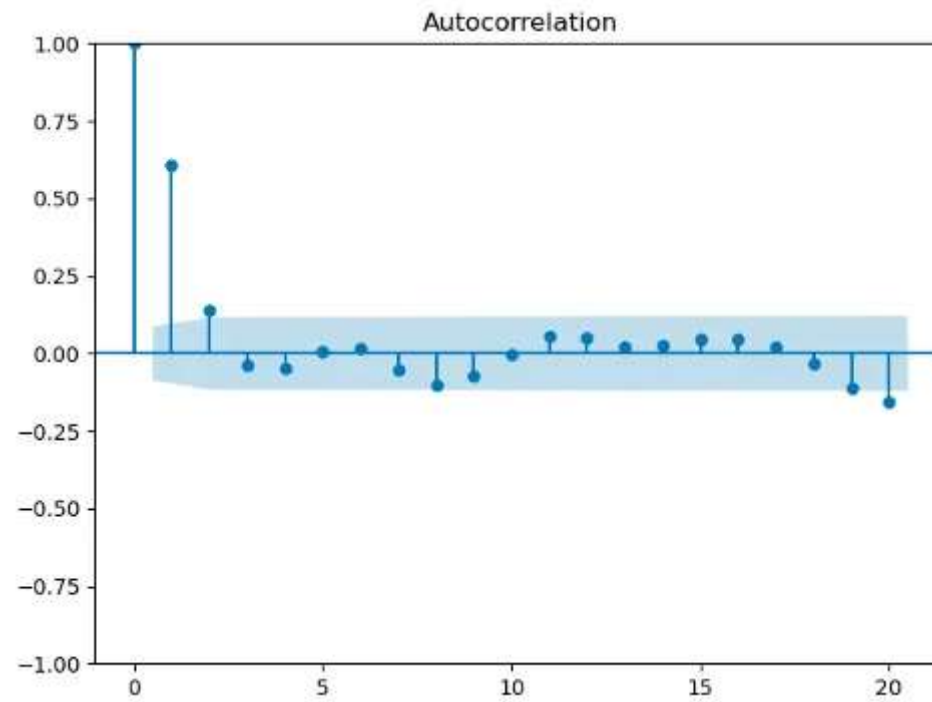
Figure from Time series forecasting in python — Marco Peixerio

The map above shows us a scenario that can develop after plotting the ACF graph. Accordingly, if there is autocorrelation, that is, if all coefficients are

not non-significant, we need to find out after which lag the coefficients are non-significant. Let's explain with an example.

```
# We made stationary series
widget_sales_diff = np.diff(df['widget_sales'], n=1)
plot_acf(widget_sales_diff, lags=20);

plt.tight_layout()
```



The above ACF graph was drawn after the data was stationary. We can determine the $MA(q)$ value according to this graph. In our previous review, all coefficients were non-significant (i.e. they were all in the blue confidence interval). But now, after the second lag, the coefficients have entered the confidence interval. So we can accept the parameter q as 2. There is only one significant coefficient seen in Lag 20. We can ignore this as it has no continuity and has never been seen before 20.

Modelling

There is an important point that we should not forget when we are going to model for $MA(q)$. The MA model does not allow us to predict 50 days ahead. The $MA(q)$ model can only predict the future q steps. Therefore, if we want to predict the far future, we need to add the predicted q value to the data and then re-estimate it.

An important point here is that we should do the modeling on the data that we have made stationary. In other words, we will use the `widet_sales_diff` data above while modeling.

```
df_diff = pd.DataFrame({'widet_sales_diff': widet_sales_diff})
```

```
train = df_diff[:449]
test = df_diff[449:]
```

We create a dataframe with the data we have made stationary and divide it into train and test.

With the following function, we will create both the MA model and our base models for comparison. This function will be used for both moving average and autoregressive processes.

```
def rolling_forecast(df: pd.DataFrame, train_len: int, horizon: int, window: int):
    total_len = train_len + horizon

    if method == 'mean':
        pred_mean = []

        for i in range(train_len, total_len, window):
            mean = np.mean(df[:i].values)
            pred_mean.extend(mean for _ in range(window))

        return pred_mean

    elif method == 'last':
        pred_last_value = []

        for i in range(train_len, total_len, window):
            last_value = df[:i].iloc[-1].values[0]
```

```
pred_last_value.extend(last_value for _ in range(window))

return pred_last_value

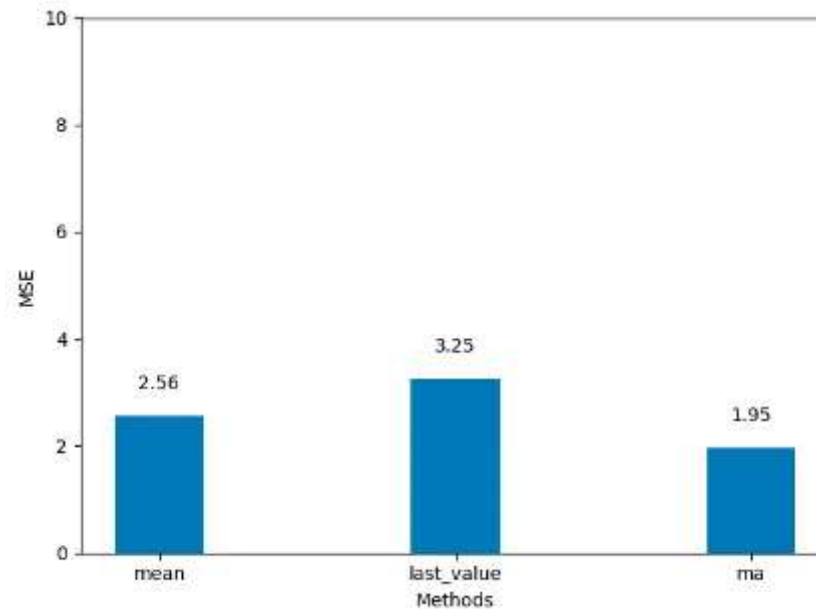
elif method == 'MA':
    pred_MA = []

    for i in range(train_len, total_len, window):
        model = SARIMAX(df[:i], order=(0,0,2))
        res = model.fit(dispatch=False)
        predictions = res.get_prediction(0, i + window - 1)
        oos_pred = predictions.predicted_mean.iloc[-window:]
        pred_MA.extend(oos_pred)

    return pred_MA
```

“**model = SARIMAX(df[:i], order=(0,0,2))**” is the part where we create the model and specify the q parameter we found. After doing all this, only the training process remains.

Above, we created both our base models and our MA(2) model. We can now compare these three models.

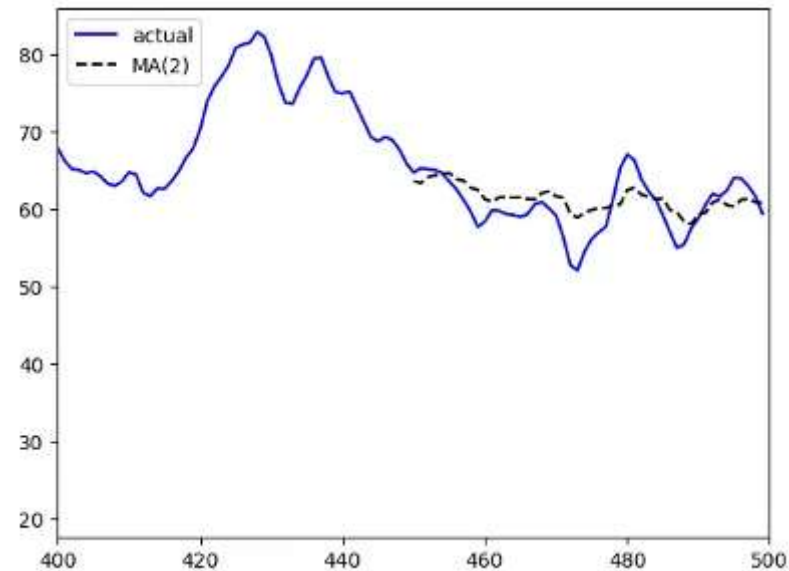


When we examine the model comparisons, we see that the best model is the MA(2) model, as we expected. But to get the real metrics, we have to reverse the difference we applied.

```
df['pred_widget_sales'] = pd.Series()  
df['pred_widget_sales'][450:] = df['widget_sales'].iloc[450] + pred_df['pred_MA']
```

When we reverse this process, we can now obtain the actual sales values predicted by the model. You can see the actual values and the estimation of

the model in the chart below.



Thank you for reading. In the next article, we will make predictions using Autoregressive Process. Stay tuned!

References

- Time series forecasting in python — Marco Peixerio
- Tan, Y.-F.; Ong, L.-Y.; Leow, M.-C.; Goh, Y.-X. Exploring Time-Series Forecasting Models for Dynamic Pricing in Digital Signage Advertising. Future Internet 2021, 13, 241. <https://doi.org/10.3390/fi13100241>

- Machine learning with time series — Markus Löning
- <https://www.influxdata.com/time-series-forecasting-methods/>
- <https://builtin.com/data-science/time-series-forecasting-python>
- <https://www.machinelearningplus.com/time-series/time-series-analysis-python/>
- <https://www.projectpro.io/article/moving-average-time-series-model/716>

Time Series Analysis

Moving Average

Simple Moving Average

Time Series Forecasting

Artificial Intelligence



Written by Buse Köseoğlu

Follow

448 Followers

BAU Software Engineering | Data Science | <https://www.linkedin.com/in/busekoseoglu/>

More from Buse Köseoğlu

- The formula for R^2 is

$$R^2 = 1 - \frac{SS_{Residual}}{SS_{Total}}$$

while the formula for adjusted R^2 is

$$R^2_{adj} = 1 - \frac{MS_{Residual}}{MS_{Total}}$$

 Buse Köseoğlu in BAU Yazılım ve Bilişim Kulübü

Model Performansını Değerlendirmek: Regresyon

Regresyonda bağımlı X değişkenlerini alarak bağımsız Y değişkenini tahmin etmeye...

3 min read · Feb 11, 2021




136



2



 Buse Köseoğlu

One Layer Neural Network From Scratch—Classification

Classification is a supervised learning method. In supervised learning, we have...

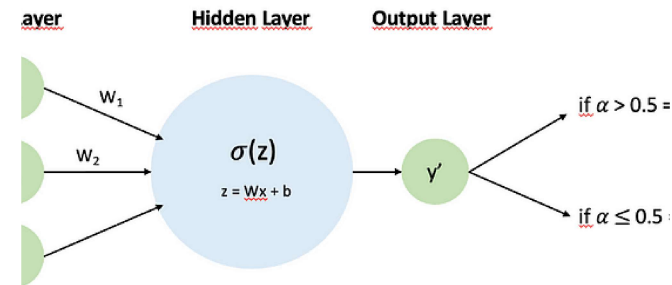
7 min read · Jul 28, 2023



222



2






 Buse Köseoğlu

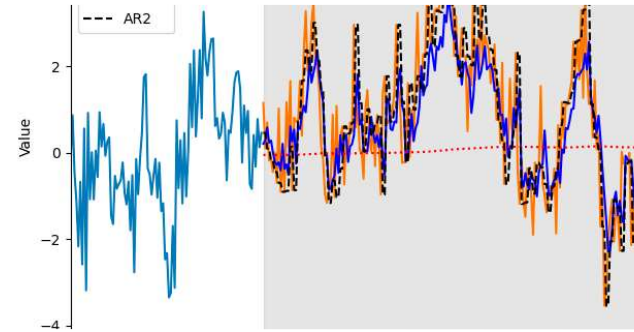
TEMEL SWITCH KONFIGÜRASYONU

Bu yazımda iki switch ve iki bilgisayar ile kurulmuş bir topolojide çalışacağız....

6 min read · Aug 28, 2020

 60 

 ...




 Buse Köseoğlu

Guide to Time Series Analysis with Python—3: Autoregressive...

In the previous article, we examined the moving average process. This time we will...

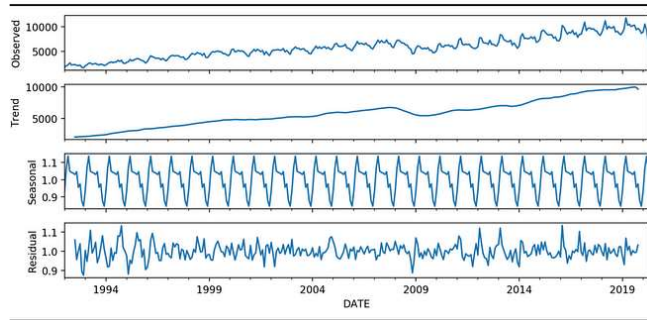
4 min read · Aug 8, 2023


 117 

 ...

See all from Buse Köseoğlu

Recommended from Medium



 Roshmita Dey

Time Series Decomposition

Time series decomposition is a vital technique that helps in understanding the...

5 min read · Jan 13, 2024



118



1



 Dr. Vipul Joshi in Time Series Using Python

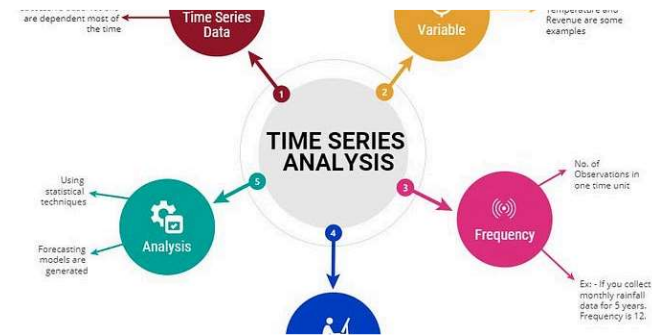
Introduction To Time Series Analysis -Part 1

What is Time Series and why is it special?

9 min read · Apr 17, 2024



56



Lists



AI Regulation

6 stories · 477 saves



Generative AI Recommended Reading

52 stories · 1121 saves



ChatGPT

21 stories · 673 saves



ChatGPT prompts

48 stories · 1654 saves



 Sarah Lea in Python in Plain English

Beginner's guide to predicting Time Series data with Python

This article delves into various modeling techniques such as linear regression,...

★ • 10 min read • Feb 15, 2024



 Max Zhang

A guide to time series forecasting with arima in python 3



 Zaki Nurkholis

A Practical Guide on Scikit-learn for Time Series Forecasting

While most machine learning algorithms available in scikit-learn (and various other...

14 min read • Dec 12, 2023



 Michael Whittle in Coinmonks

Artificial Intelligence (AI) models for Trading

Introduction

🌟 · 14 min read · Jan 21, 2024



135



Exploring Random Forests from Machine Learning

11 min read · Jun 4, 2024



311

6



See more recommendations

[Help](#) [Status](#) [About](#) [Careers](#) [Press](#) [Blog](#) [Privacy](#) [Terms](#) [Text to speech](#) [Teams](#)