



Published in Data Folks Indonesia



Handhika Yanuar Pratama

Aug 14, 2021 · 7 min read · Listen

Simple Moving Average (SMA) Indicator Using Machine Learning



Photo by [Maxim Hopman](#) on [Unsplash](#)

Disclaimer: This is not an financial advice, this for educational purpose only, any loss or profit because of this is beyond my responsibility

As pandemic grows, people tend to change the way of his/her life. Working offline may not the best choices today, many of them more likely becoming an investor or trader. But, trust me, trading without knowledge will lead you into

Sign in to Medium with Google

Sahil Chaudhari
sahilchaudhari2709@gmail.com

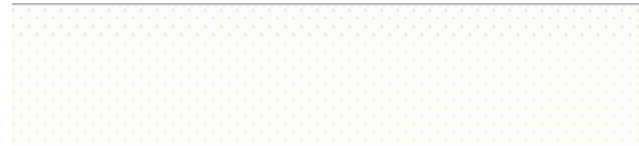
Sahil Chaudhari
sahilchaudhari0927@gmail.com

big loss. In this article, I want to tell you about using Simple Moving Average (SMA) indicator. Well, I understand that not everyone knows it. So, let's begin exploring.

Simple Moving Average

Actually, I'm not full time trader so for this section, I will bring Charles Schwab's explanation about SMA. SMA or Simple Moving Average is a straightforward calculation of a stock's average price (usually using close price) over a set number of days.

Okay, for example, you are setting 10 days moving average in close price of your favorite stock, so you will adding all the 10 days number and then divide it by 10, each day passed, the moving average will moving too, like this.




There is more advanced indicator called EMA (Exponential Moving Average). In this article, I will using SMA, because it's beginner friendly. Also Charles Schwab said that SMA is great for detecting daily pattern, the consequence is that SMA is not really good in detecting hourly pattern. In this article we will use 5 years data, so I will using SMA rather than EMA. Okay, let's go into the next steps.


Getting the Dataset

Yahoo! Finance is media property of Yahoo! network since 2017 that provides financial news, data, and everything about stock price including financial reports. This is legit source to collect stock price data. In this example I would like to predict the PGAS.JK stock market price.

Sign in to Medium with Google

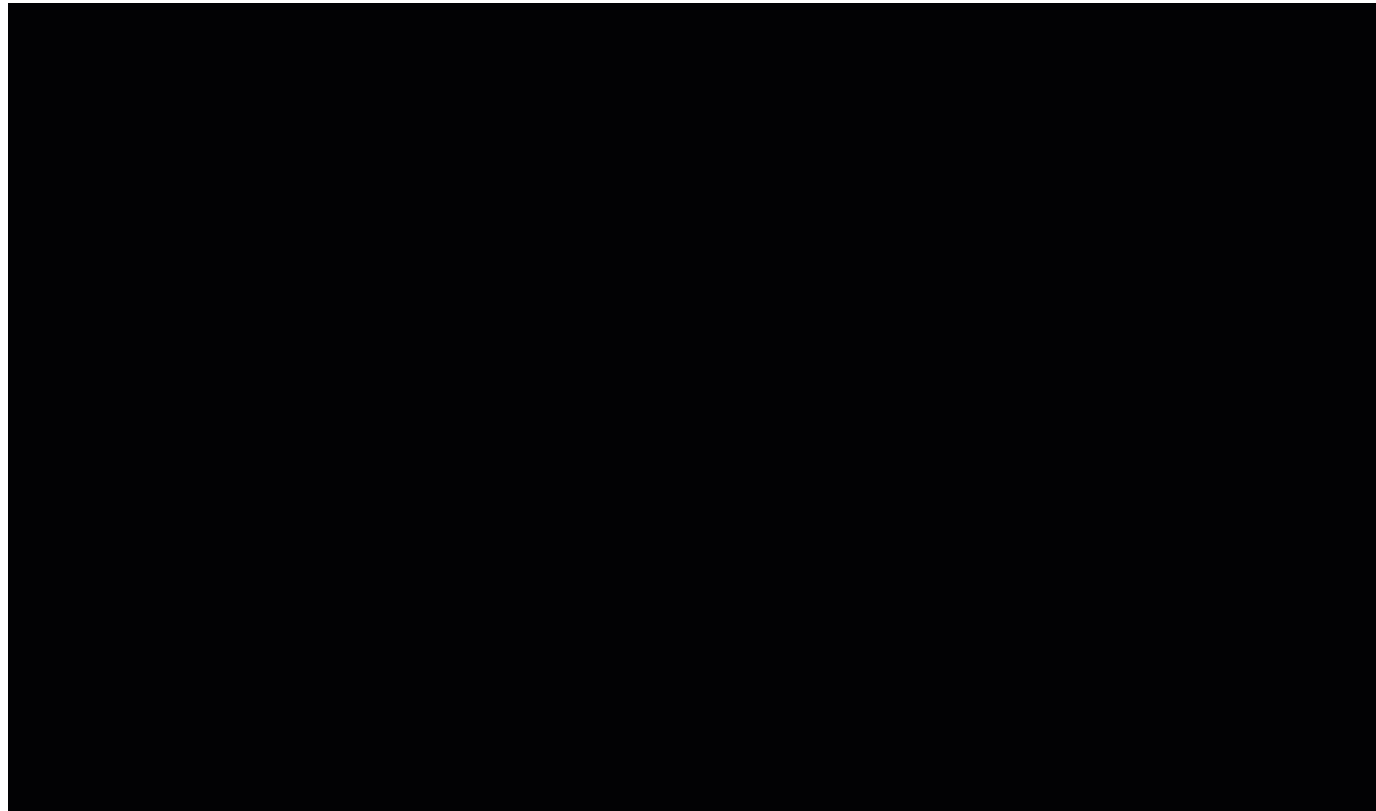


Sahil Chaudhari
sahilchaudhari2709@gmail.com



Sahil Chaudhari
sahilchaudhari0927@gmail.com

PGAS.JK owned by **PT Perusahaan Gas Negara Tbk (PGAS.JK)**, this stock belongs to IDX (Indonesian Stock Exchange). Why I use this, because, I ever getting profit from this stock and it would be nice if I used it as example. Recently, this stock price is back just same as June-July 2020, why not test it right? Who knows if this stock will be great again in December 2021. To collect PGAS.JK data in the last 5 years you can see this GIF below




Okay, we already understand about what is SMA and had the data, let's go into the coding section.


Coding Section

Data

Before do the code in real data, I want to explore some features from the dataset. Okay, if you open the dataset that you download before, maybe the data will look like broken, but this is CSV file. You can open it using Pandas or you can add

Sign in to Medium with Google

 **Sahil Chaudhari**
sahilchaudhari2709@gmail.com

 **Sahil Chaudhari**
sahilchaudhari0927@gmail.com

configuration for displaying csv file into your Excel app. This picture below is displayed using Pandas.

	Open	High	Low	Close	Adj Close	Volume
2016-08-04	3300.0	3330.0	3240.0	3300.0	2778.076416	38220800.0
2016-08-05	3300.0	3370.0	3280.0	3350.0	2820.168457	35901300.0
2016-08-08	3360.0	3370.0	3250.0	3270.0	2752.821289	44279000.0
2016-08-09	3280.0	3310.0	3180.0	3230.0	2719.147705	43741400.0
2016-08-10	3240.0	3260.0	3180.0	3180.0	2677.055420	23591800.0
...
2021-07-28	1030.0	1035.0	1000.0	1005.0	1005.000000	97526100.0
2021-07-29	1015.0	1015.0	990.0	1005.0	1005.000000	108477500.0
2021-07-30	1010.0	1010.0	970.0	975.0	975.000000	240537200.0
2021-08-02	980.0	990.0	960.0	990.0	990.000000	86357700.0
2021-08-03	995.0	1015.0	985.0	1010.0	1010.000000	93092400.0


1261 rows x 6 columns

Okay, the picture above actually is a data frame, but to make it beginner friendly let us say that the picture is a table. From the table, we had 6 columns with one index. The details for them are as follows:

- 1. Open — It’s the market opening price in the current day
- 2. High — It’s the highest price of the stock in the current day
- 3. Low — It’s the lowest price of the stock in the current day
- 4. Close — It’s the market close price in the current day
- 5. Adj Close — It’s the closing price that had been amends with closing price that day and the corporation values.
- 6. Volume — It’s the market volume of stock price that traded in the current day


Investopedia said that the Adj Close commonly being used to check the historical returns or analyze the past performance. So, because this is recommended, let use this as the parameter.

Sign in to Medium with Google



Sahil Chaudhari

sahilchaudhari2709@gmail.com



Sahil Chaudhari

sahilchaudhari0927@gmail.com

Code

This project is actually one piece of my 180 Days Of Data Science. All the code and the dataset can be downloaded in my GitHub [here](#).

I started by importing the modules and setting the style of the matplotlib into seaborn-whitegrid (You can use your preferences)

```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np

# For chart style
plt.style.use('seaborn-whitegrid')
```

Next, I import the dataset and set the index into 'date' column. After set the index into 'date', I turn the format into `DatetimeIndex`. Because, this is a time-series data. If you don't want to turn it into the `DatetimeIndex`, the data can still be processed, but it takes a lot more time.


```
# Store the data
df = pd.read_csv("PGAS.JK (1).csv")

# Set the date columns as the index
df = df.set_index('Date')


# Set the index into datetime for efficiency
df = df.set_index(pd.DatetimeIndex(df.index.values))
```

If you want to display the table, you can write `df` in your notebook, and this table will be displayed

Sign in to Medium with Google



Sahil Chaudhari
sahilchaudhari2709@gmail.com



Sahil Chaudhari
sahilchaudhari0927@gmail.com

	Open	High	Low	Close	Adj Close	Volume
2016-08-04	3300.0	3330.0	3240.0	3300.0	2778.076416	38220800.0
2016-08-05	3300.0	3370.0	3280.0	3350.0	2820.168457	35901300.0
2016-08-08	3360.0	3370.0	3250.0	3270.0	2752.821289	44279000.0
2016-08-09	3280.0	3310.0	3180.0	3230.0	2719.147705	43741400.0
2016-08-10	3240.0	3260.0	3180.0	3180.0	2677.055420	23591800.0
...
2021-07-28	1030.0	1035.0	1000.0	1005.0	1005.000000	97526100.0
2021-07-29	1015.0	1015.0	990.0	1005.0	1005.000000	108477500.0
2021-07-30	1010.0	1010.0	970.0	975.0	975.000000	240537200.0
2021-08-02	980.0	990.0	960.0	990.0	990.000000	86357700.0
2021-08-03	995.0	1015.0	985.0	1010.0	1010.000000	93092400.0

1261 rows × 6 columns

You can check the basic information from the dataset by typing `df.info()` and you will get information, that all columns type are float64 and the index is `DatetimeIndex` which start from 2016-08-04 until 2021-08-03. The data length is 1261 and there is no null values.

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 1261 entries, 2016-08-04 to 2021-08-03
Data columns (total 6 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   Open        1260 non-null   float64
1   High        1260 non-null   float64
2   Low         1260 non-null   float64
3   Close       1260 non-null   float64
4   Adj Close   1260 non-null   float64
5   Volume      1260 non-null   float64
dtypes: float64(6)
memory usage: 69.0 KB
```

Data is collected, we also know the data type. The next step is visualizing the column that we used. In this case I use the 'Adj Close', so the syntax will be like this.

```
# Visualize the adj close price
plt.figure(figsize=(16,8))
```

 Sign in to Medium with Google



Sahil Chaudhari
sahilchaudhari2709@gmail.com



Sahil Chaudhari
sahilchaudhari0927@gmail.com

```
plt.plot(df['Adj Close'], label='PGAS')

# Adding text into the visualization
plt.title('Adj Close Price History', fontsize=18)
plt.ylabel('Adj Close Price', fontsize=18)
plt.xlabel('Date', fontsize=18)

# Give legend
plt.legend()

# Show the graph
plt.show()
```


The output will be like this




Beautiful right? But it doesn't done yet. After visualizing the data, we will make simple function for SMA

```
# Create function for calculating Simple Moving Average (SMA)
def SMA(data, period=30, column='Adj Close'):
    return data[column].rolling(window=period).mean()
```

Sign in to Medium with Google

 **Sahil Chaudhari**
sahilchaudhari2709@gmail.com

 **Sahil Chaudhari**
sahilchaudhari0927@gmail.com

The period=30 is just the default value for rolling function, so whenever you forget to put any value, the function will take first 30 data to build a SMA30 instead of bring error into your function.

If you don't get what I mean. Let's say I build SMA10 and SMA50, so I will use this syntax for saving the value.

```
# Create two new columns to store the 10 day and 50 day SMA
df['SMA10'] = SMA(df, 10)
df['SMA50'] = SMA(df, 50)
```


To understand the function, you can check the values of df['SMA10'] or df['SMA50']. In the df['SMA10'] the output will be like this.

df.SMA10[:15]	
2016-08-04	NaN
2016-08-05	NaN
2016-08-08	NaN
2016-08-09	NaN
2016-08-10	NaN
2016-08-11	NaN
2016-08-12	NaN
2016-08-15	NaN
2016-08-16	NaN
2016-08-18	2666.953394
2016-08-19	2650.116577
2016-08-22	2634.963428
2016-08-23	2625.703174
2016-08-24	2623.177612
2016-08-25	2635.805225
Name: SMA10, dtype: float64	
SMA10 is starting at the 10 index	


Did you get it? The first 10 of the data will be taken to build the SMA10, just like the illustrator that I used before.

Next, we build some signal, so whenever SMA10 is greater than SMA50 it will noted into the signal value as 1 and if not the signal value will be 0. After that,

Sign in to Medium with Google



Sahil Chaudhari
sahilchaudhari2709@gmail.com



Sahil Chaudhari
sahilchaudhari0927@gmail.com

Get started

Sign In

Search



Handhika Yanuar Pratama

132 Followers

A Stoic Developer || Junior Data Engineer || Dreamer

Follow

Related

Stock Price Prediction using ARIMA Model
Any kind of prediction is...

Stock Prediction using Prophet (Python)
Stock Price Prediction u...

Game Stop Stock Market price Prediction using RNN (LSTM)

Predicting the Stock Market
One of my biggest inspirations for w...

we build new column named ‘Position’, so whenever the pattern change the direction, it will store the data into the columns.

```
df['Position'] = df['Signal'].diff()

df['Buy'] = np.where(df['Position'] == 1, df['Adj Close'], np.NAN)
df['Sell'] = np.where(df['Position'] == -1, df['Adj Close'], np.NAN)
```

Everything had been setup, let’s display it into chart.

```
# Visualize the close price with SMA also Buy and Sell Signals
plt.figure(figsize=(16,8))
plt.plot(df['Adj Close'], alpha=0.5, label='Actual Close Price')
plt.plot(df['SMA10'], alpha=0.5, label='SMA10')
plt.plot(df['SMA50'], alpha=0.5, label='SMA50')

# Make buy or sell signal
plt.scatter(df.index, df['Buy'], alpha=1, label='Buy Signal',
            marker='^', color='green')
plt.scatter(df.index, df['Sell'], alpha=1, label='Sell Signal',
            marker='v', color='red')


# Adding text into the visualization
plt.title('Adj Close Price w/ Buy and Sell Signals', fontsize=18)
plt.ylabel('Adj Close Price', fontsize=18)
plt.xlabel('Date', fontsize=18)

# Give legend
plt.legend()


# Show the graph
plt.show()
```

The output will be like this.

Sign in to Medium with Google



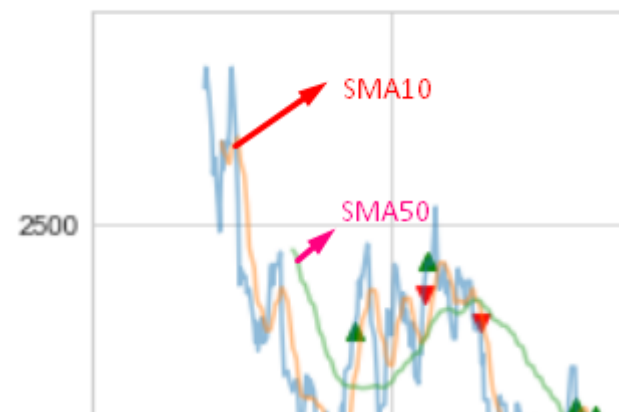
Sahil Chaudhari
sahilchaudhari2709@gmail.com



Sahil Chaudhari
sahilchaudhari0927@gmail.com



Remember before I built new column named 'Position?', the column will be displayed as mark, to display whenever SMA10 is bigger than SMA50 and otherwise. This sometimes used as indicator, we should buy stock or not (Green and Red Triangle). If you watch more detail, you will get that SMA10 will be start first.



Well that's all about how to build SMA indicator in stock price, you can use it into your favorite stock, also you can improve it into better machine.

Conclusion



Sign in to Medium with Google


Sahil Chaudhari
sahilchaudhari2709@gmail.com

Sahil Chaudhari
sahilchaudhari0927@gmail.com

In this article, I demonstrated how to build SMA indicator in the stock price using Machine Learning. I went through three major step. First, I share about SMA. Next, we gather the data from legit source. Finally, we process the data to build SMA Indicator.


References

 Sign in to Medium with Google 



Sahil Chaudhari

sahilchaudhari2709@gmail.com



Sahil Chaudhari

sahilchaudhari0927@gmail.com

Source Code

180-days-of-data-science/day67 at main · theDreamer911/180-days-of-data-science

Just like the name, let's start and see you in New Year - 180-days-of-data-science/day67 at main · ...

github.com

mer911/180-days-science

ne, let's start and see you in New

0

Issues

0

Stars

0

Forks

Have A Nice Code 🙌

Sign up for Data Folks Indonesia

By Data Folks Indonesia


A Newsletter to keep you up to date with the data world [Take a look.](#)


Your email


✉️

Get this newsletter

By signing up, you will create a Medium account if you don't already have one. Review our [Privacy Policy](#) for more information about our privacy practices.

 Sign in to Medium with Google

 Sahil Chaudhari
sahilchaudhari2709@gmail.com

 Sahil Chaudhari
sahilchaudhari0927@gmail.com

https://medium.com/data-folks-indonesia/simple-moving-average-sma-indicator-using-machine-learning-e8951f61dd9b

12/12