

Nama : Dany Dzahab

Kelas : 3KA17

Npm : 11119578

1.

https://github.com/Danydzhb/UTS/blob/main/UTS_PSD_Dany_Dzahab_11119578_3KA17.ipynb

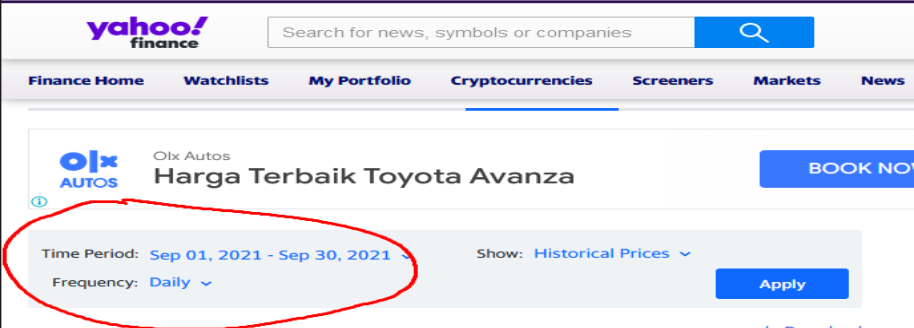
2.

Ubah dataset dengan rentang waktu 1 sept 2021 – 30 sept 2021

Ubah dataset dengan mengunjungi

<https://finance.yahoo.com/quote/GOOG/history?p=GOOG>

Ubah dibagian yg dilingkari merah sesuai rentang waktu yg diinginkan lalu download .



yahoo! finance

Search for news, symbols or companies

Finance Home Watchlists My Portfolio Cryptocurrencies Screeners Markets News

OlX AUTOS Olx Autos

Harga Terbaik Toyota Avanza

BOOK NOW

Time Period: Sep 01, 2021 - Sep 30, 2021

Frequency: Daily

Show: Historical Prices

Apply

Currency in USD

Download

Date	Open	High	Low	Close*	Adj Close**	Volume
Sep 29, 2021	2,742.19	2,747.97	2,685.00	2,690.42	2,690.42	1,316,900
Sep 28, 2021	2,781.77	2,792.13	2,714.00	2,723.68	2,723.68	2,109,500
Sep 27, 2021	2,831.71	2,850.00	2,810.00	2,830.02	2,830.02	942,200
Sep 24, 2021	2,818.92	2,858.07	2,817.01	2,852.66	2,852.66	747,500
Sep 23, 2021	2,832.19	2,845.05	2,821.93	2,836.53	2,836.53	863,600
Sep 22, 2021	2,801.01	2,831.67	2,789.44	2,818.77	2,818.77	1,103,400
Sep 21, 2021	2,802.34	2,816.23	2,778.11	2,792.93	2,792.93	906,500
Sep 20, 2021	2,780.00	2,787.25	2,741.06	2,780.34	2,780.34	1,745,900
Sep 17, 2021	2,875.97	2,884.99	2,821.23	2,829.27	2,829.27	3,002,000
Sep 16, 2021	2,902.42	2,904.00	2,868.33	2,887.47	2,887.47	1,014,600
Sep 15, 2021	2,875.18	2,911.63	2,845.12	2,904.12	2,904.12	1,032,400

3.

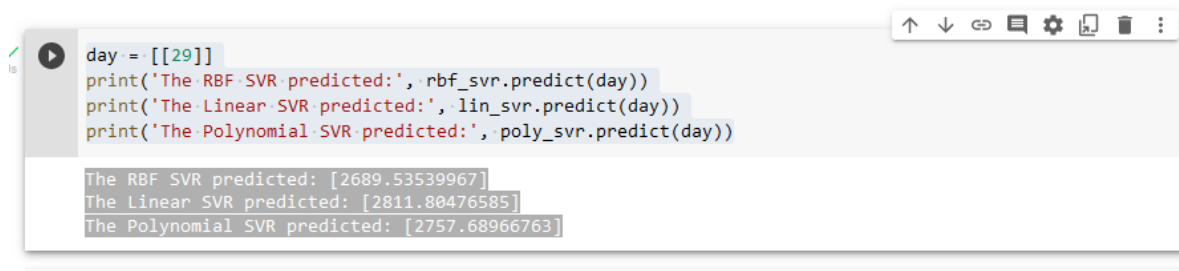
Tentukan hasil prediksi untuk tgl 29 september 2021

Code: `day = [[29]]`

```
print('The RBF SVR predicted:', rbf_svr.predict(
day))
print('The Linear SVR predicted:', lin_svr.predi
ct(day))
print('The Polynomial SVR predicted:', poly_svr.
predict(day))
```

Output : The RBF SVR predicted: [2689.53539967]
The Linear SVR predicted: [2811.80476585]
The Polynomial SVR predicted: [2757.68966763]

SS:



```
day = [[29]]
print('The RBF SVR predicted:', rbf_svr.predict(day))
print('The Linear SVR predicted:', lin_svr.predict(day))
print('The Polynomial SVR predicted:', poly_svr.predict(day))
```

The RBF SVR predicted: [2689.53539967]
The Linear SVR predicted: [2811.80476585]
The Polynomial SVR predicted: [2757.68966763]

4. Model terbaik dari grafik di atas ini tampaknya adalah RBF yang merupakan model Regresi Vektor Pendukung yang menggunakan kernel yang disebut fungsi basis radial.

5.

Hal pertama yang ingin saya lakukan sebelum menulis satu baris kode adalah memberikan deskripsi di komentar tentang apa yang dilakukan kode tersebut. Dengan cara ini saya dapat melihat kembali kode saya dan tahu persis apa fungsinya.

```
# Description: This program predicts the price of GOOG stock for a
#             specific day
#             using the Machine Learning algorithm called Support
#             Vector Regression (SVR)
#
```

Sekarang impor paket /libraries untuk memudahkan penulisan program.

```
✓ [2] #Import the libraries
    from sklearn.svm import SVR
    import numpy as np
    import pandas as pd
    import matplotlib.pyplot as plt
    plt.style.use('seaborn-darkgrid')
```

Selanjutnya saya akan memuat data saham Google (GOOG) yang saya dapatkan dari finance.yahoo.com ke dalam sebuah variabel bernama 'df' kependekan dari data frame.

Note : data berasal dari rentang waktu waktu 1 sept 2021 – 30 sept 2021

```
[3] #Load the data
    #from google.colab import files # Use to load data on Google Colab
    #uploaded = files.upload() # Use to load data on Google Colab
    df = pd.read_csv('GOOG.csv')
    df
```

Load data

```
[4] actual_price = df.tail(1)
    actual_price
```

Tampilkann data row terakhir

```
[5] df = df.head(len(df)-1)
    df
```

Tampilkan semua baris kecuali baris terakhir

```
✓ [6] #Create the lists / X and y data set
0s days = list()
adj_close_prices = list()
```

```
✓ [7] df_days = df.loc[:, 'Date']
0s df_adj_close = df.loc[:, 'Adj Close']
```

```
✓ [8] #Create the independent data set
0s for day in df_days:
    days.append( [int(day.split('-')[2])] )
#Create the dependent data set
for adj_close_price in df_adj_close:
    adj_close_prices.append( float(adj_close_price) )
```

Membuat variabel independent dan dependent data set

```
✓ [10] #Create and train an SVR model using a linear kernel
1s lin_svr = SVR(kernel='linear', C=1000.0)
lin_svr.fit(days,adj_close_prices)#Create and train an SVR model using a polynomial kernel
poly_svr = SVR(kernel='poly', C=1000.0, degree=2)
poly_svr.fit(days, adj_close_prices)#Create and train an SVR model using a RBF kernel
rbf_svr = SVR(kernel='rbf', C=2500.0, gamma=0.20)
rbf_svr.fit(days, adj_close_prices)

SVR(C=2500.0, gamma=0.2)
```

Membuat dan melatih model SVR

```
✓ [11] #Plot the models on a graph to see which has the best fit
0s plt.figure(figsize=(16,8))
plt.scatter(days, adj_close_prices, color = 'black', label='Original Data')
plt.plot(days, rbf_svr.predict(days), color = 'green', label='RBF Model')
plt.plot(days, poly_svr.predict(days), color = 'orange', label='Polynomial Model')
plt.plot(days, lin_svr.predict(days), color = 'purple', label='Linear Model')
plt.xlabel('Days')
plt.ylabel('Adj Close Price')
plt.title('Support Vector Regression')
plt.legend()
plt.show()
```

Membuat grafik model

```
✓ [12] day = [[29]]
0s print('The RBF SVR predicted:', rbf_svr.predict(day))
print('The Linear SVR predicted:', lin_svr.predict(day))
print('The Polynomial SVR predicted:', poly_svr.predict(day))

The RBF SVR predicted: [2689.53539967]
The Linear SVR predicted: [2811.80476585]
The Polynomial SVR predicted: [2757.68966763]
```

Membuat prediksi