



FREELANCE BRAND DESIGNER

Machine Learning

Prediksi Harga Saham Tesla Menggunakan Support Vector Regression

STAND THE TEST OF TIME

About Group 2

Group Members:

- Dava Virgio Kertawijaya - ooooooo56848
- Nigel Andrian - ooooooo55946
- Michael Owen Kohar - ooooooo56755
- Samuel Andrew - ooooooo56975



Content

01 LATAR BELAKANG & RUMUSAN MASALAH

04 BATASAN MASALAH

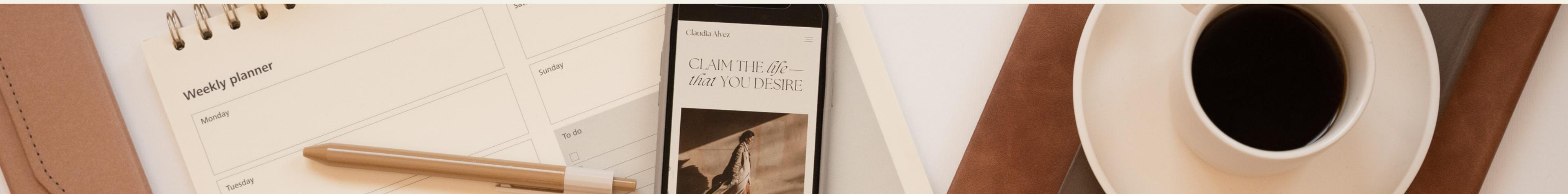
06 TUJUAN & MANFAAT

09 TELAAH LITELATAR

10 SUPPORT VECTOR REGRESSION (SVR)

16 FUNGSI KERNEL

18 DAFTAR REFERENSI



Latar Belakang & Rumusan Masalah



Latar Belakang

Korporasi yang menerbitkan saham menentukan besarnya pendapatan yang diterima dari pemegang saham (emiten). Jika emiten mampu menghasilkan keuntungan yang signifikan, pemegang saham juga akan mendapatkan keuntungan yang cukup besar. Risiko yang terkait dengan investasi akan meningkat berbanding lurus dengan keuntungan yang ditawarkan [3]. Akibatnya, penting untuk mengekstrapolasi harga saham saat ini dari harga saham kemarin.

Rumusan Masalah

Bagaimana cara menggunakan Support Vector Regression untuk memprediksi harga saham Tesla di masa depan?

Batasan Masalah



Batasan Masalah

1. Penelitian ini akan fokus pada prediksi harga saham Tesla menggunakan algoritma Support Vector Regression (SVR).
2. Penelitian ini hanya akan menggunakan data harga saham Tesla dari tahun sebelumnya untuk melakukan prediksi harga saham di masa depan.
3. Penelitian ini tidak akan mempertimbangkan faktor eksternal seperti berita atau peristiwa pasar yang mungkin mempengaruhi harga saham Tesla.
4. Penelitian ini hanya akan mempertimbangkan faktor internal seperti harga saham sebelumnya dan volume perdagangan sebagai variabel dalam algoritma SVR.
5. Penelitian ini akan menggunakan metric evaluasi accuracy untuk mengukur tingkat keakuratan prediksi model harga saham Tesla.

Tujuan & Manfaat



RIO STERLING

BRAND DESIGNER

Tujuan

Penelitian ini bertujuan untuk memprediksi tren harga saham Tesla menggunakan algoritma SVR.



Manfaat

Memberikan prediksi yang akurat terhadap harga saham Tesla di masa depan. Dengan memiliki prediksi yang akurat, investor dapat membuat keputusan investasi yang lebih baik dan mengoptimalkan keuntungan mereka. Hal ini juga dapat membantu meningkatkan kepercayaan investor dalam pasar saham dan memperkuat stabilitas pasar.

Telaah Literatur



Support Vector Regression (SVR)

SVR merupakan pengembangan SVM yang dapat mengatasi overfitting dalam estimasi fungsi, dengan cara meminimalkan batas atas dari generalization error [2]. SVR didasarkan pada konsep risk minimization dan dapat memberikan prediksi harga saham Tesla yang akurat. SVM sendiri menggunakan hyperplane tunggal pada ruang berdimensi banyak yang dapat diselesaikan secara non-linier, dengan metode quadratic programming [1].

Support Vector Regression (SVR)

Fungsi regresi dari metode SVR sebagai berikut:

$$f(x) = w^T \varphi(x) + b$$

Dengan keterangan:

- w merupakan vector pembobot,
- $\varphi(x)$ merupakan fungsi yang memetakan x dalam suatu dimensi,
- b merupakan bias

Support Vector Regression (SVR)

Untuk memaksimalkan hyperplane dengan meminimalkan nilai Loss function

$$R(f(x)) = \frac{1}{2} \|w\|^2 + \frac{C}{n} \sum_{i=1}^n L_\epsilon(y_i, f(x_i))$$

dimana

$$L_\epsilon(y_i, f(x_i)) = \begin{cases} 0 & ; |y_i - f(x_i)| \leq \epsilon \\ |y_i - f(x_i)| - \epsilon; & \text{lainnya} \end{cases}$$

Dengan keterangan:

L_ϵ merupakan ϵ -insensitive loss function,

ϵ dan C merupakan parameter.

Support Vector Regression (SVR)

Konsep dari kuadratik loss function adalah meminimumkan nilai sebagai berikut::

$$R(w, \xi, \xi^*) = \frac{1}{2} \|w\|^2 + C \left(\sum_{i=1}^n (\xi_i + \xi_i^*) \right) \text{ dengan batasan}$$

$$w\varphi(x_i) + b - y_i \leq \varepsilon + \xi_i^*; \quad y_i - w\varphi(x_i) - b \leq \varepsilon + \xi_i \quad \text{dan} \quad \xi_i, \xi_i^* \geq 0$$

Support Vector Regression (SVR)

Dengan menggunakan penyelesaian lagrange dalam bentuk:

$$L(w, b, \xi, \xi^*, \alpha_i, \alpha_i^*, \beta_i, \beta_i^*) = \frac{1}{2} \|w\|^2 + C \left(\sum_{i=1}^n (\xi_i + \xi_i^*) \right) - \sum_{i=1}^n \alpha_i [w \varphi(x_i) + b - y_i + \varepsilon + \xi_i] - \sum_{i=1}^n \alpha_i^* [y_i - w \varphi(x_i) - b_i + \varepsilon + \xi_i^*] - \sum_{i=1}^n (\beta_i \xi_i + \beta_i^* \xi_i^*)$$

Dengan menggunakan pendekatan Karush-Kuhn-Tuck didapatkan sebagai berikut:

$$Q(\alpha, \alpha^*) = -\frac{1}{2} \sum_{i,j=1}^l (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) K(x_i, x_j) - \varepsilon \sum_{i=1}^l (\alpha_i + \alpha_i^*) + \sum_{i=1}^l y_i (\alpha_i - \alpha_i^*) \quad \text{dengan batasan}$$

$$\sum_{i=1}^l (\alpha_i - \alpha_i^*) = 0 ; 0 \leq \alpha_i \leq C ; 0 \leq \alpha_i^* \leq C \quad \text{dimana } K(x_i, x_j) \text{ merupakan fungsi kernel.}$$

Fungsi Kernel

Metode analisis data mining umumnya menggunakan fungsi linier, namun SVM dengan pendekatan kernel dapat memisahkan data nonlinier secara linier pada feature space yang baru. Dengan kemampuan ini, SVM dapat memberikan hasil yang akurat dalam analisis data mining untuk mengidentifikasi informasi penting dari data nonlinier.

Fungsi Kernel

Fungsi kernel yang digunakan pada metode SVR adalah sebagai berikut:

1. Linear: $\mathbf{x}^T \mathbf{x}$

2. Polinomial: $(\mathbf{x}^T \mathbf{x} + 1)^n$

3. Radial Basis Function (RBF): $\exp\left(-\frac{1}{2\sigma^2} \|\mathbf{x} - \mathbf{x}_i\|^2\right)$

Metodologi Penelitian

Jenis Penelitian

Dalam penelitian ini, digunakan pendekatan penelitian kuantitatif dengan tujuan untuk melakukan prediksi trend harga saham Tesla.

Sumber Data

Data yang digunakan dalam penelitian ini merupakan data sekunder yang terdiri dari data harian harga saham Tesla dari 29 Juni 2010 hingga 17 Maret 2017. Data ini terdiri dari 1692 data harga saham harian. Data akan dibagi menjadi dua bagian, yaitu data training dan data testing, dengan perbandingan 8 banding 2. Dalam penelitian ini, diasumsikan bahwa harga saham hari ini hanya akan terpengaruh oleh harga saham hari-hari sebelumnya.

Metric Evaluasi

Metric Evaluasi yang digunakan dalam peneltian ini adalah Akurasi menggunakan R-square, Dikarenakan harga saham yang tidak bisa diprediksi secara 100% akurat sehingga kami memutuskan memakai akurasi untuk melihat seberapa besar akurasi dari prediksi terhadap harga saham tesla yang telah kami gunakan

Read CSV

Read CSV

Read Dataset

```
In [3]: ➜ tesla = pd.read_csv('Tesla.csv')
tesla.head()
```

Out[3]:

	Date	Open	High	Low	Close	Volume	Adj Close
0	6/29/2010	19.000000	25.00	17.540001	23.889999	18766300	23.889999
1	6/30/2010	25.790001	30.42	23.299999	23.830000	17187100	23.830000
2	7/1/2010	25.000000	25.92	20.270000	21.959999	8218800	21.959999
3	7/2/2010	23.000000	23.10	18.709999	19.200001	5139800	19.200001
4	7/6/2010	20.000000	20.00	15.830000	16.110001	6866900	16.110001

```
In [4]: ➜ tesla.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1692 entries, 0 to 1691
Data columns (total 7 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   Date        1692 non-null    object 
 1   Open         1692 non-null    float64
 2   High         1692 non-null    float64
 3   Low          1692 non-null    float64
 4   Close        1692 non-null    float64
 5   Volume       1692 non-null    int64  
 6   Adj Close    1692 non-null    float64
dtypes: float64(5), int64(1), object(1)
memory usage: 92.7+ KB
```

Pre Processing

```
In [10]: ▶ ^ = tesla[['Open', 'High', 'Low']].values  
y = tesla['Close'].values
```

```
In [11]: ▶ from sklearn.model_selection import train_test_split  
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
In [12]: ▶ import numpy as np  
  
correlation_matrix = np.corrcoef(X.T, y)  
correlation_with_target = correlation_matrix[-1, :-1]  
  
print(correlation_with_target)  
  
[0.99923326 0.99969092 0.9996561 ]
```

```
In [13]: ▶ scaler = MinMaxScaler()  
X_scaled = scaler.fit_transform(X)
```

```
In [14]: ▶ print(np.unique(y))  
  
[ 15.8 16.110001 17.049999 ... 282.26001 284.119995 286.040009]
```

Mencari Model Optimal

```
In [15]: svr = SVR(kernel='rbf', C=0.1, gamma=0.1, epsilon=0.1)
```

```
In [16]: from sklearn.model_selection import cross_val_score
scores = cross_val_score(svr, X, y, cv=5, scoring='r2')
```

```
In [17]: # Menampilkan hasil validasi silang
print('R-squared scores:', scores)
print('R-squared mean:', scores.mean())
print('R-squared std:', scores.std())
```

```
R-squared scores: [-2019.38960196 -2684.90951015 -3.30158869 -62.45166163
-49.20031584]
R-squared mean: -963.8505356542116
R-squared std: 1153.079871338242
```

```
In [18]: svr.fit(X_train, y_train)
```

```
Out[18]: SVR(C=0.1, gamma=0.1)
```

```
In [19]: y_pred = svr.predict(X_test)
```

```
In [20]: mse = mean_squared_error(y_test, y_pred)
rmse = np.sqrt(mse)
mae = mean_absolute_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print("Mean Squared Error:", mse)
print("Root Mean Squared Error:", rmse)
print("Mean Absolute Error:", mae)
print("R-squared:", r2)
```

```
Mean Squared Error: 9778.692437946775
Root Mean Squared Error: 98.88727136465428
Mean Absolute Error: 89.75919412717663
R-squared: -0.04855369558381595
```

Hasil dan Diskusi

Hasil Prediksi Model

C	Gamma	Epsilon	MSE	RMSE	MAE	R-Square
0,1	0,1	0,1	9778.692437946775	98.88727136465428	89.75919412717663	-0.04855369558381595
0,1	0,01	0,1	7576.57624150215	87.0435307274593	79.33478140357875	0.18757573486300294
0,1	0,001	0,1	4835.13255162105	69.5351173984847	62.25300227266322	0.48153639785562374
0,1	0,0001	0,1	2548.3977677130915	50.4816577353903	44.61614665642437	0.7267393453562616
0,1	0,1	0,01	9785.311386347861	98.92073284376669	89.76901713602618	-0.04926343493300611
0,1	0,01	0,01	7578.462023220843	87.05436245944739	79.33743977652306	0.18737352547736708
0,1	0,001	0,01	4831.369105627526	69.50805065334177	62.24669590664343	0.48193994620625247
0,1	0,0001	0,01	2549.420564015889	50.49178709469381	44.61553151439951	0.7266296725293543
0,1	0,1	0,001	9785.974172187969	98.92408287261485	89.76999943691114	-0.049334504408443225
0,1	0,01	0,001	7579.00104709407	87.05745830825794	79.3382666407961	0.1873157268015352
0,1	0,001	0,001	4831.369105627526	69.50805065334177	62.24669590664343	0.48193994620625247
0,1	0,0001	0,001	2549.4205640158875	50.4917870946938	44.6155315143995	0.7266296725293544
0,1	0,1	0,0001	9786.040459681979	98.92441791429444	89.77009766699963	-0.04934161231139211
0,1	0,01	0,0001	7579.054950984987	87.0577678957196	79.33834894168106	0.18730994677272395
0,1	0,001	0,0001	4831.369105627526	69.50805065334177	62.24669590664343	0.48193994620625247
0,1	0,0001	0,0001	2549.420564015889	50.49178709469381	44.61553151439951	0.7266296725293542
1	0,1	0,1	3115.928738412167	55.82050464132483	46.65183009000264	0.6658838986325751
1	0,01	0,1	741.1941038872787	27.224880236417548	13.982137785079981	0.920522930677312
1	0,001	0,1	182.76205399167696	13.518951660231535	5.723994619769737	0.9804027145406672
1	0,0001	0,1	3115.866782113317	55.81994967852727	46.64615551051413	0.6658905421083299
1	0,1	0,01	38.83615023859141	6.231865710891997	3.1107459272957545	0.9958356611465864
1	0,01	0,01	741.4947150211656	27.230400566667498	13.985534221284254	0.9204906966217498

Hasil Prediksi Model

1	0,001	0,01	182.82516642013775	13.521285679259119	5.727556473525405	0.9803959470949662
1	0,0001	0,01	38.82746626320429	6.231168932327569	3.1085548511122325	0.9958365923155073
1	0,1	0,001	3115.8998650388207	55.82024601377909	46.647433054246534	0.6658869946786486
1	0,01	0,001	741.4545832722796	27.229663664325336	13.985415832557267	0.9204949998855932
1	0,001	0,001	182.79947123645465	13.520335470558955	5.727464023723256	0.9803987023487981
1	0,0001	0,001	38.90382732650471	6.237293269239848	3.109878984468498	0.9958284042396852
1	0,1	0,0001	3115.903160822177	55.82027553516891	46.647562404384026	0.6658866412770215
1	0,01	0,0001	741.4407753078965	27.229410116781754	13.985395039852294	0.920496480491735
1	0,001	0,0001	182.79846095733873	13.52029810904104	5.727515103465411	0.9803988106794269
1	0,0001	0,0001	38.90759830455955	6.237595554743795	3.1100533373413217	0.9958279998836835
10	0,1	0,1	593.5526798803174	24.36293660214871	10.931244296627888	0.9363542866327368
10	0,01	0,1	38.68663553997865	6.219858160760473	2.600888830648178	0.9958516933708096
10	0,001	0,1	3.44360923652481	1.8556964289788376	1.1049455163286541	0.9996307472380364
10	0,0001	0,1	38.68663553997865	6.219858160760473	2.600888830648178	0.9958516933708096
10	0,1	0,01	593.5711205734611	24.36331505713993	10.933293100053778	0.9363523092664306
10	0,01	0,01	38.56972078868568	6.210452542986355	2.5978530401191806	0.9958642299543369
10	0,001	0,01	3.431406507378138	1.8524056001259923	1.1030264339302216	0.9996320557173473
10	0,0001	0,01	4.514465332460753	2.124727119528706	1.3212154822135076	0.9995159210356624
10	0,1	0,001	593.5817068018748	24.363532313724026	10.933022304760422	0.9363511741219319
10	0,01	0,001	38.58944369927392	6.212040220352241	2.598992798053232	0.9958621150978859
10	0,001	0,001	3.4288990348301756	1.8517286612325725	1.103194643733742	0.9996323245896555
10	0,0001	0,001	4.543810374860177	2.1316215364975504	1.324803154624826	0.9995127744132638

Hasil Prediksi Model Dengan Error Terendah

PARAMATER :

C : 10

Gamma : 0,001

Epsilon : 0,001

Hasil :

Mean Squared Error: 3.4288990348301756

Root Mean Squared Error: 1.8517286612325725

Mean Absolute Error: 1.103194643733742

R-squared: 0.9996323245896555

Validasi Model SVR

```
# Melakukan prediksi pada data latih dan data uji
y_train_pred = svr.predict(X_train)
y_test_pred = svr.predict(X_test)

# Menghitung koefisien determinasi (R-squared) untuk data latih dan data uji
r_squared_train = r2_score(y_train, y_train_pred)
r_squared_test = r2_score(y_test, y_test_pred)

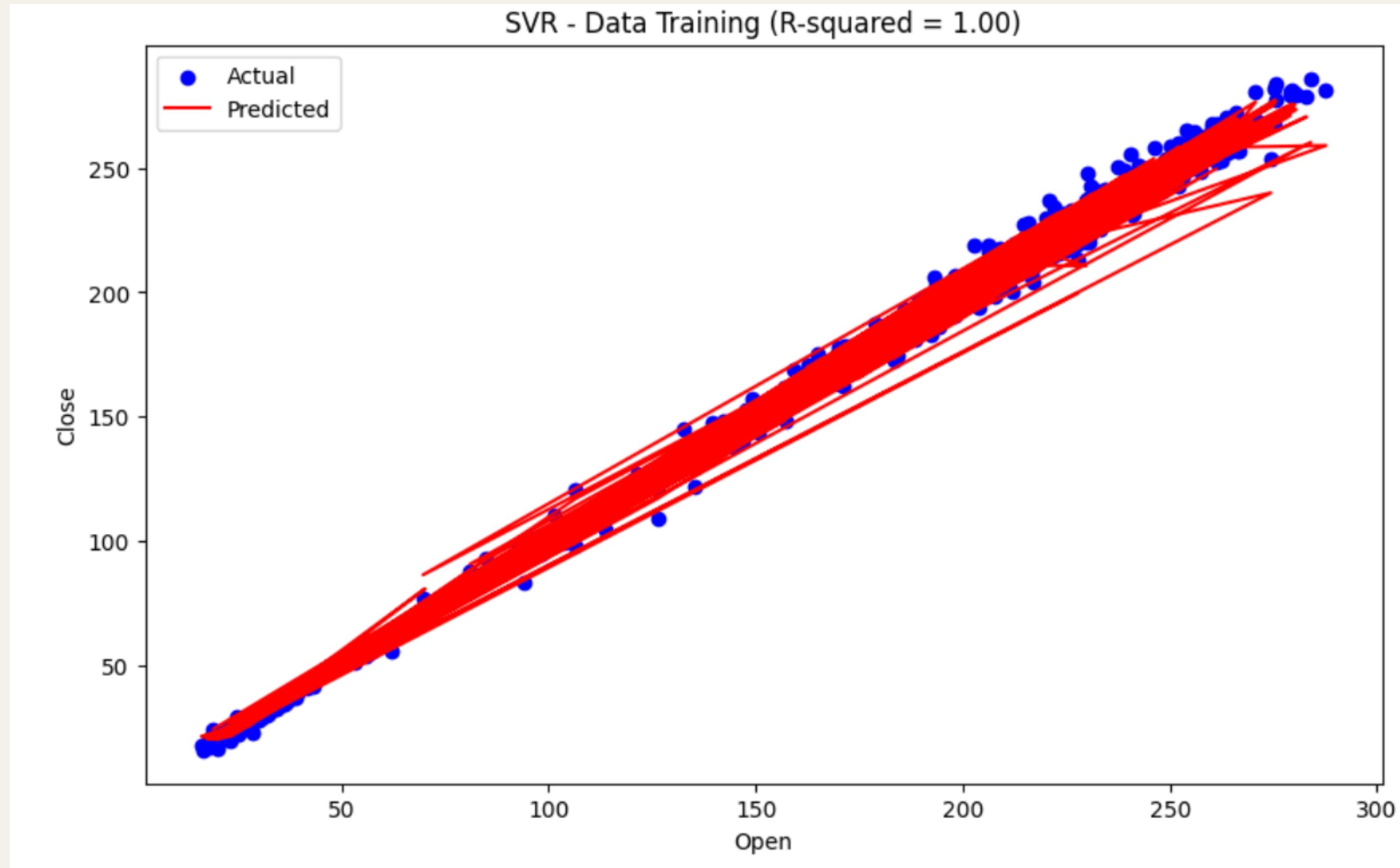
print("Koefisien Determinasi (R-squared) untuk Data Latih:", r_squared_train)
print("Koefisien Determinasi (R-squared) untuk Data Uji:", r_squared_test)
```

Hasil :

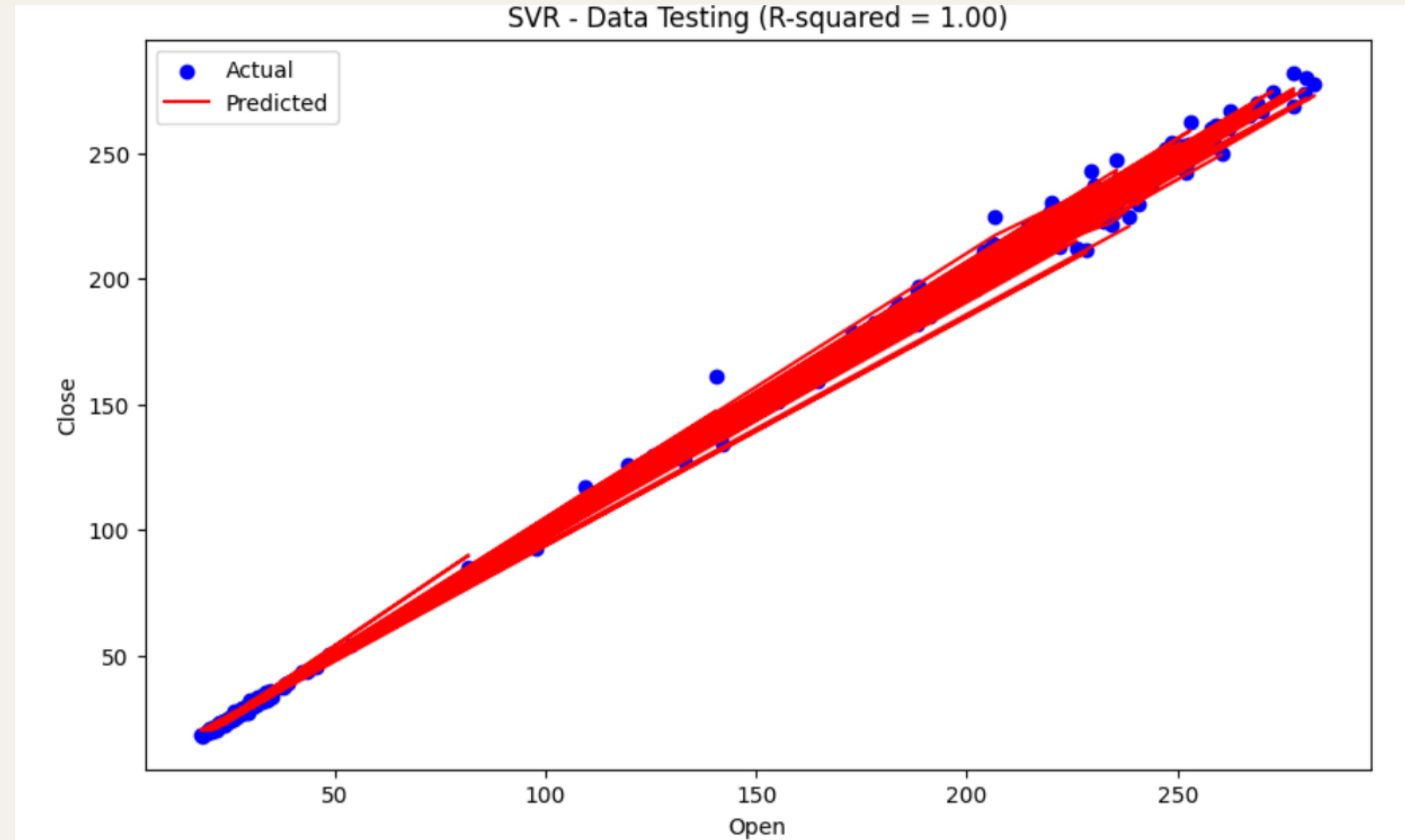
Koefisien Determinasi (R-squared) untuk Data Latih: 0.999436928173908

Koefisien Determinasi (R-squared) untuk Data Uji: 0.999632324589655

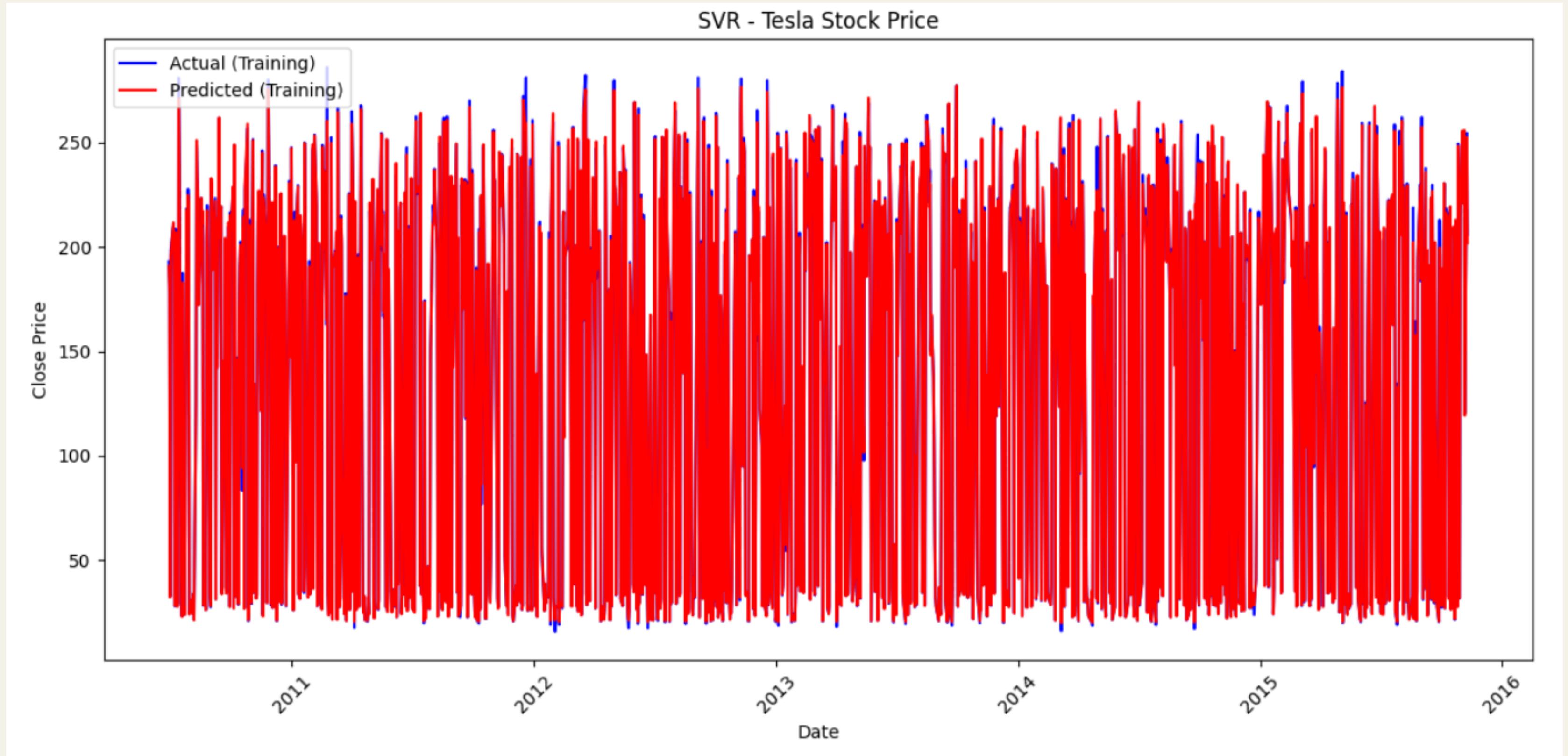
Visualisasi Scatter Plot Hasil Data Training



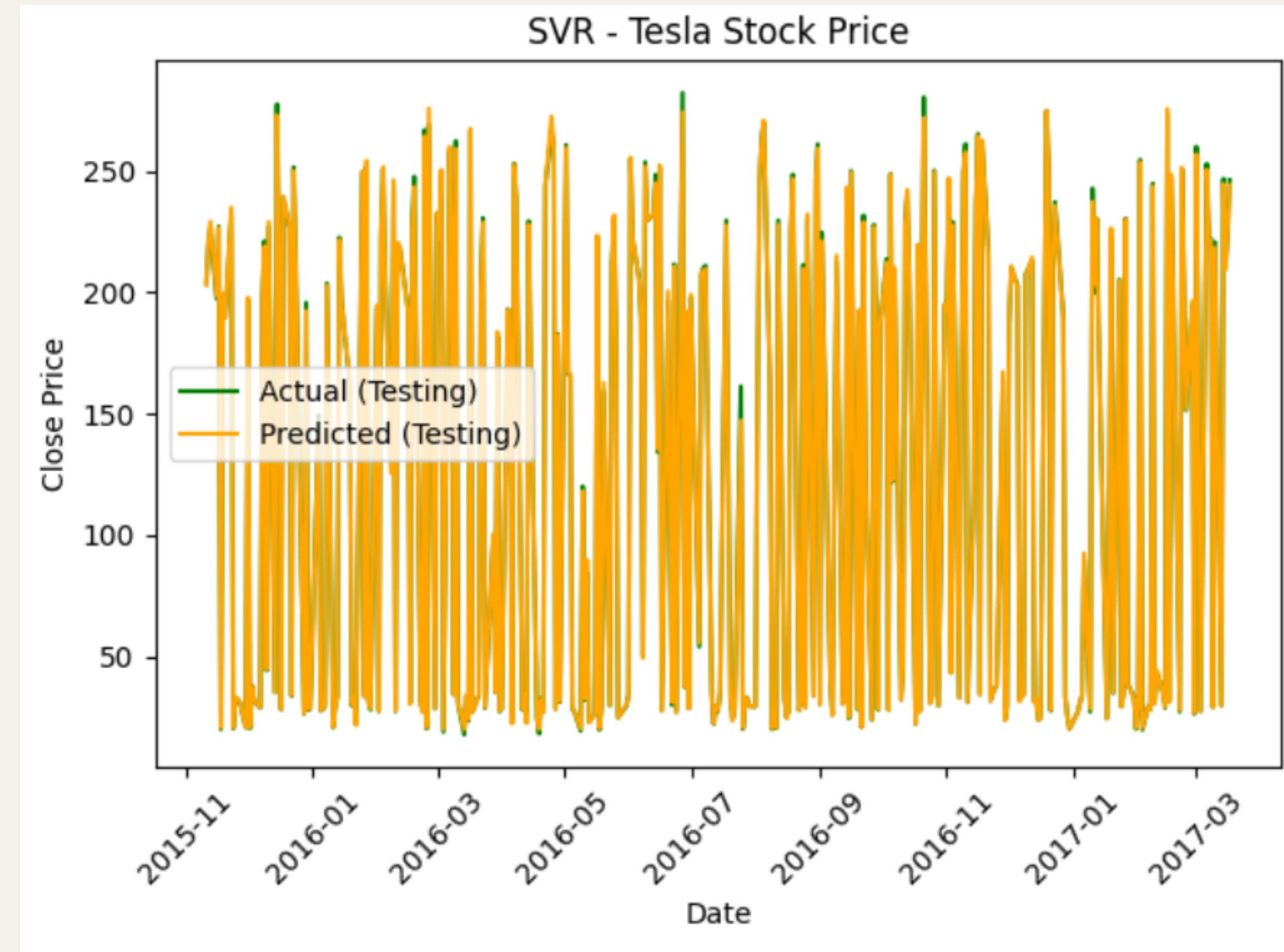
Visualisasi Scatter Plot Hasil Data Testing



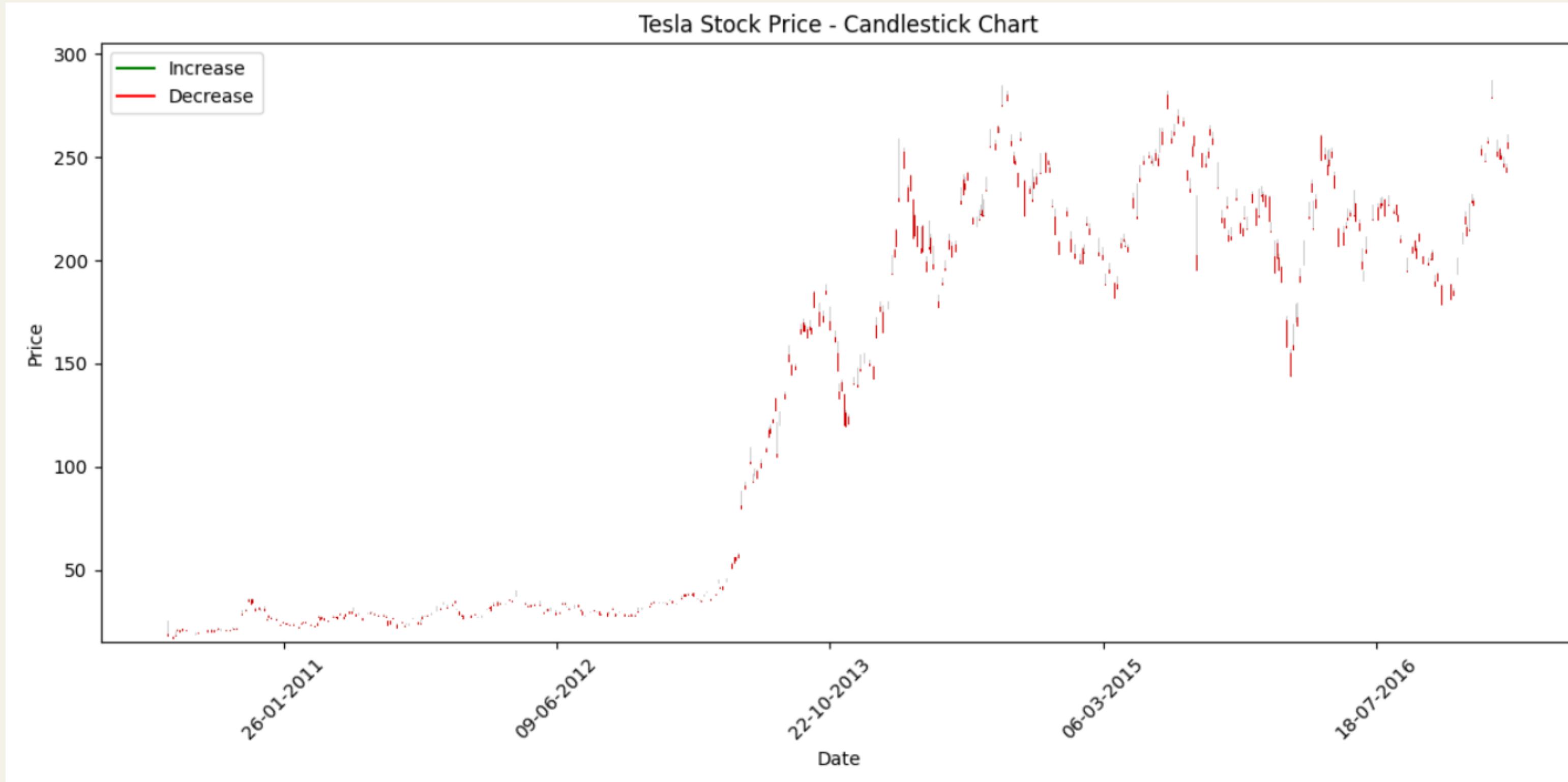
Grafik Hasil Aktual dan Prediksi dari Data Training



Grafik Hasil Aktual dan Prediksi dari Data Testing



Visualisasi Candlestick Saham Tesla



Simpulan

Setelah melakukan analisis, dapat disimpulkan bahwa model SVR terbaik menggunakan fungsi kernel RBF dengan parameter $C = 10$, $\gamma = 0,001$, dan $\epsilon_{\text{silon}} = 0,001$. Model ini memiliki tingkat akurasi yang cukup tinggi, yaitu 99,94% untuk data training dan 99,96% untuk data testing, sehingga dapat diandalkan dalam memprediksi harga saham Tesla.

Daftar Referensi

- [1] Bertsimas, D. dan Shioda, R., Classification and Regression Via Integer Optimization, *Operation Research*, 2007, Vol. 55, No. 2: 252-271.
- [2] Gunn, S., *Support Vector Machines for Classification*, University of Southampton, 1998.
- [3] Pai, P.F and Lin, C.S., A Hybrid ARIMA and Support Vector Machine Model in Stock Proce Forecasting, *Omega*, 2005: 497-505.

**SEKIAN DAN
TERIMA KASIH :)**

Sehat selalu ya orang baik