Pemodelan Prediksi Harga Saham Tesla

line 1: Rafi Anas Naufal
line 2: *00000104240*

line 1: Yoel Beny Christian
line 2: *00000104698*

line 1: Michael Raynara Pradipta
line 2: *00000100867*

line 1: Nikolas Lyen Agung
line 2: *000000098199*

line 1: Enrico Felix Khosalim
line 2: *00000103230*

Dalam laporan ini, data saham Tesla dianalisis untuk menemukan pola, tren, dan elemen yang mempengaruhi pergerakan harga saham. Ini dilakukan dengan menggunakan data historis seperti harga pembukaan, penutupan, volume perdagangan, dan indikator teknis lainnya. Eksplorasi data, analisis statistik, dan visualisasi seperti grafik tren dan pola candlestick adalah semua teknik yang digunakan. Model prediksi berbasis regresi dan algoritma pembelajaran mesin juga digunakan untuk memproyeksikan pergerakan harga saham. Hasil analisis menunjukkan bahwa saham Tesla sangat berfluktuasi karena berita pasar, kemajuan teknologi, dan ekonomi global. Laporan ini membantu dalam membuat keputusan investasi yang lebih baik, terutama dalam memahami dinamika pasar saham Tesla.

Keywords (*Tesla, Regression, Random Forest, Crisp DM****)***

I. Introduction

Saham Tesla, Inc. (TSLA) telah menjadi salah satu saham paling populer di pasar modal dunia. Dengan inovasi di sektor kendaraan listrik dan energi terbarukan, perusahaan ini menarik perhatian investor global. Namun, volatilitas saham Tesla sering kali menjadi tantangan bagi investor dalam mengambil keputusan investasi yang tepat. Oleh karena itu, analisis data saham Tesla diperlukan untuk memahami pola pergerakan harga dan faktor-faktor yang mempengaruhinya.

Tujuan dari proyek ini adalah untuk menganalisis data saham Tesla guna mengidentifikasi pola historis, memahami tren harga, dan memberikan prediksi yang dapat membantu investor dalam pengambilan keputusan. Selain itu, proyek ini bertujuan untuk mengembangkan wawasan mendalam tentang dinamika pasar saham Tesla.

Dengan volatilitas pasar saham yang tinggi, terutama pada saham-saham teknologi seperti Tesla, kebutuhan untuk memiliki analisis yang akurat menjadi semakin penting. Investor membutuhkan alat dan wawasan untuk mengelola risiko dan memanfaatkan peluang. Tanpa pemahaman yang jelas, keputusan investasi dapat menjadi spekulatif dan berisiko tinggi.

Dalam proyek ini, kami akan fokus pada aspek berikut:

1.Identifikasi Tren: Memahami pergerakan harga saham Tesla dari waktu ke waktu.

2.Analisis Volatilitas: Mengukur tingkat fluktuasi harga saham untuk mengidentifikasi risiko.

3.Faktor Eksternal: Menganalisis dampak berita, laporan keuangan, dan kondisi ekonomi terhadap pergerakan harga saham.

4.Prediksi Harga: Membuat model prediksi berbasis data historis untuk memproyeksikan harga saham di masa depan.

II. Metodologi

A. Teori

Random Forest

Random Forest adalah algoritma ensemble learning untuk regresi dan klasifikasi yang membangun banyak pohon keputusan. Hasil dari pohon-pohon tersebut digabungkan (rata-rata untuk regresi, voting mayoritas untuk klasifikasi) untuk menghasilkan prediksi yang lebih akurat dan stabil dibandingkan pohon tunggal.

Regression

Metode statistik dan machine learning yang digunakan untuk memodelkan hubungan antara variabel independen (predictors) dan variabel dependen (target). Tujuannya adalah untuk memprediksi nilai target berdasarkan nilai prediktor. Contohnya adalah memprediksi harga rumah berdasarkan ukuran dan lokasi. Regresi linear adalah bentuk paling sederhana, tetapi ada juga variasi lain seperti regresi polinomial, regresi logistik, dan regresi ridge untuk menangani masalah yang lebih kompleks.

B. Metode Pendekatan

Data yang digunakan dalam laporan ini berasal dari file WIKI-TSLA.csv, yang berisi informasi historis mengenai harga saham Tesla Inc. Data ini mencakup kolom-kolom seperti tanggal, harga pembukaan, harga tertinggi, harga terendah, harga penutupan, dan volume perdagangan. Analisis data saham sangat penting bagi investor dan analis pasar untuk memahami tren dan pola yang dapat mempengaruhi keputusan investasi.

Random Forest

Metode : CRISP DM

Kelebihan :Akurasi Tinggi, Mengurangi overfitting dengan ensemble.

Kekurangan :Komputasi Berat. Memerlukan waktu dan memori lebih.

Sulit Di Interpretasi, Kompleks untuk analisis mendalam.

Regression

Metode: Simple Linear Regression

Kelebihan: Sederhana dalam implementasi dan interpretasi. Dapat digunakan untuk berbagai jenis data, tidak terbatas pada deret waktu.

Kekurangan: Tidak selalu dapat menangkap pola musiman atau tren jangka panjang jika hanya menggunakan variabel independen tanpa mempertimbangkan waktu.

C. Perbandingan Antara Metodologi

Dalam dunia analisis data, regression dan Random Forest adalah dua metode populer yang sering digunakan untuk memprediksi nilai atau memahami hubungan antar variabel. Regression, terutama regresi linier, adalah metode analitik yang sederhana dan efektif dalam memodelkan hubungan antara variabel independen (predictor) dan variabel dependen (target) dalam bentuk fungsi matematis. Dengan pendekatan ini, regression memberikan keunggulan dalam hal interpretasi yang mudah dan transparansi model. Selain itu, metode ini sangat efisien dalam menangani data kecil hingga sedang yang memiliki hubungan linier, serta dapat memberikan wawasan mengenai kontribusi masing-masing variabel independen terhadap target.

Namun, regression memiliki beberapa keterbatasan yang signifikan. Metode ini kurang efektif dalam menangani data yang kompleks atau memiliki hubungan non-linear antara variabel. Selain itu, regression rentan terhadap outlier yang dapat mempengaruhi hasil prediksi secara drastis. Masalah lain seperti multikolinearitas—di mana variabel independen saling berkorelasi—juga dapat mengganggu akurasi model. Oleh karena itu, regression lebih cocok untuk situasi di mana data menunjukkan pola yang sederhana dan linier.

Sebaliknya, Random Forest adalah algoritma berbasis ensemble yang memanfaatkan banyak pohon keputusan (decision trees) untuk menghasilkan prediksi yang lebih akurat dan tangguh. Metode ini unggul dalam menangani data yang kompleks, non-linear, atau memiliki banyak fitur. Dengan menggabungkan prediksi dari beberapa pohon, Random Forest mampu mengurangi risiko overfitting yang sering menjadi masalah pada pohon keputusan tunggal. Selain itu, Random Forest juga tahan terhadap outlier dan dapat digunakan untuk menentukan variabel mana yang paling penting dalam model prediksi.

Namun, Random Forest memiliki kelemahan dalam hal interpretabilitas dan efisiensi. Karena terdiri dari banyak pohon, model ini lebih sulit dipahami dibandingkan regression. Selain itu, Random Forest memerlukan lebih banyak sumber daya komputasi, baik dalam hal waktu maupun memori, terutama ketika bekerja dengan dataset besar. Dengan demikian, pemilihan antara regression dan Random Forest sangat bergantung pada karakteristik data dan tujuan analisis. Regression ideal untuk model yang sederhana dan mudah dipahami, sementara Random Forest lebih cocok untuk data besar dan kompleksitas yang tinggi.

Metode pengembangan yang digunakan crisp dm. CRISP-DM (Cross-Industry Standard Process for Data Mining) adalah metodologi yang digunakan untuk pengembangan proyek data mining.

III. HASIL ANALISA

Data understanding

```
[15]: df.info()

<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 1949 entries, 2018-03-27 to 2010-06-29
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   Open         1949 non-null   float64
 1   High         1949 non-null   float64
 2   Low          1949 non-null   float64
 3   Close        1949 non-null   float64
 4   Volume       1949 non-null   float64
 5   Ex-Dividend  1949 non-null   float64
 6   Split Ratio  1949 non-null   float64
 7   Adj. Open    1949 non-null   float64
 8   Adj. High    1949 non-null   float64
 9   Adj. Low     1949 non-null   float64
 10  Adj. Close   1949 non-null   float64
 11  Adj. Volume  1949 non-null   float64
dtypes: float64(12)
memory usage: 197.9 KB
```

Open: Harga pembukaan saham pada hari tertentu.

High: Harga tertinggi yang dicapai oleh saham pada hari tertentu.

Low: Harga terendah yang dicapai oleh saham pada hari tertentu.

Close: Harga penutupan saham pada hari tertentu.

Volume: Jumlah saham yang diperdagangkan pada hari tertentu.

Ex-Dividend: Tanggal dimana saham mulai diperdagangkan tanpa hak untuk mendapatkan dividen yang diumumkan.

Split Ratio: Rasio pemecahan saham, yang menunjukkan berapa banyak saham baru yang diterima pemegang saham untuk setiap saham yang mereka miliki.

Adj. Open: Harga pembukaan yang disesuaikan, biasanya untuk memperhitungkan dividen dan pemecahan saham.

Adj. High: Harga tertinggi yang disesuaikan.

Adj. Low: Harga terendah yang disesuaikan.

Adj. Close: Harga penutupan yang disesuaikan.

Adj. Volume: Volume perdagangan yang disesuaikan.

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 1949 entries, 2018-03-27 to 2010-06-29
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Open          1949 non-null   float64
 1   High          1949 non-null   float64
 2   Low           1949 non-null   float64
 3   Close         1949 non-null   float64
 4   Volume        1949 non-null   float64
 5   Ex-Dividend   1949 non-null   float64
 6   Split Ratio   1949 non-null   float64
 7   Adj. Open     1949 non-null   float64
 8   Adj. High     1949 non-null   float64
 9   Adj. Low      1949 non-null   float64
 10  Adj. Close    1949 non-null   float64
 11  Adj. Volume   1949 non-null   float64
dtypes: float64(12)
memory usage: 197.9 KB
```

```
df.head()
```

Date	Open	High	Low	Close	Volume	Ex-Dividend	Split Ratio	Adj. Open	Adj. High	Adj. Low	Adj. Close	Adj. Volu
2018-03-27	304.00	304.27	277.18	279.18	13696168.0	0.0	1.0	304.00	304.27	277.18	279.18	136961
2018-03-26	307.34	307.59	291.36	304.18	8324639.0	0.0	1.0	307.34	307.59	291.36	304.18	832463
2018-03-23	311.25	311.61	300.45	301.54	6600538.0	0.0	1.0	311.25	311.61	300.45	301.54	660053
2018-03-22	313.89	318.82	308.18	309.10	4914307.0	0.0	1.0	313.89	318.82	308.18	309.10	491430
2018-03-21	310.25	322.44	310.19	316.53	5927881.0	0.0	1.0	310.25	322.44	310.19	316.53	592788

```
df.tail()
```

Date	Open	High	Low	Close	Volume	Ex-Dividend	Split Ratio	Adj. Open	Adj. High	Adj. Low	Adj. Close	Adj. Volum
2010-07-06	20.00	20.0000	15.83	16.11	6866900.0	0.0	1.0	20.00	20.0000	15.83	16.11	6866900.
2010-07-02	23.00	23.1000	18.71	19.20	5139800.0	0.0	1.0	23.00	23.1000	18.71	19.20	5139800.
2010-07-01	25.00	25.9200	20.27	21.96	8218800.0	0.0	1.0	25.00	25.9200	20.27	21.96	8218800.
2010-06-30	25.79	30.4192	23.30	23.83	17187100.0	0.0	1.0	25.79	30.4192	23.30	23.83	17187100.
2010-06-29	19.00	25.0000	17.54	23.89	18766300.0	0.0	1.0	19.00	25.0000	17.54	23.89	18766300.

Data preparation

```
# Handling Missing Values
df = df.dropna()
```

Kode ini secara otomatis mencari dan menghapus semua baris yang memiliki setidaknya satu nilai kosong (NaN) di dalamnya. Ini berguna untuk membersihkan data sebelum analisis lebih lanjut, karena nilai kosong dapat menyebabkan kesalahan atau hasil yang tidak akurat.

```
# Menggunakan IQR untuk mendeteksi outliers
Q1 = df['Close'].quantile(0.25)
Q3 = df['Close'].quantile(0.75)
IQR = Q3 - Q1
```

Fungsi dari metode IQR (Interquartile Range) dalam mendeteksi outliers di Python adalah untuk mengidentifikasi nilai-nilai yang berada jauh di luar rentang normal dari data.

```
# Menghapus outliers
df = df[~((df['Close'] < (Q1 - 1.5 * IQR)) | (df['Close'] > (Q3 + 1.5 * IQR)))]
```

Kode diatas membersihkan data dari outliers yang diidentifikasi menggunakan rentang antar kuartil (IQR). Baris dengan nilai di luar batas wajar (di bawah Q1 atau di atas Q3) dihapus untuk meningkatkan kualitas data dan analisis.

```
# Normalisasi menggunakan StandardScaler
scaler = StandardScaler()
df[['Open', 'High', 'Low', 'Close', 'Volume']] = scaler.fit_transform(df[['Open', 'High', 'Low', 'Close', 'Volume']])
```

digunakan untuk standarisasi data. Standarisasi adalah proses mengubah skala fitur-fitur data sehingga mereka memiliki distribusi dengan: Mean (Rata-rata): 0, Standard Deviation (Standar Deviasi): 1

```
# Membuat bin untuk harga penutupan
bins = [0, 0.5, 1, 1.5, 2, np.inf]
labels = ['Very Low', 'Low', 'Medium', 'High', 'Very High']
df['Close_Binned'] = pd.cut(df['Close'], bins=bins, labels=labels)
```

Binning adalah teknik dalam data preprocessing yang digunakan untuk mengelompokkan data numerik ke dalam beberapa rentang atau kategori (disebut bins).

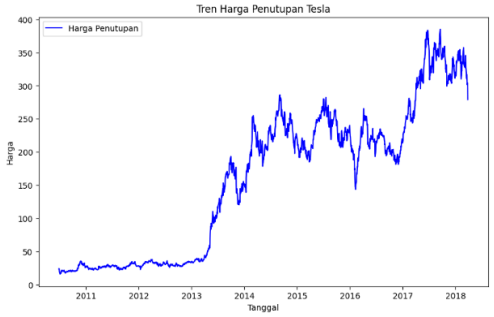
```
# Encoding untuk kolom kategorikal
df = pd.get_dummies(df, columns=['Close_Binned'], drop_first=True)
```

digunakan untuk mengonversi data kategori (categorical data) menjadi representasi numerik, yang sering disebut sebagai one-hot encoding. Hal ini berguna dalam machine learning karena sebagian besar algoritma hanya dapat bekerja dengan data numerik.

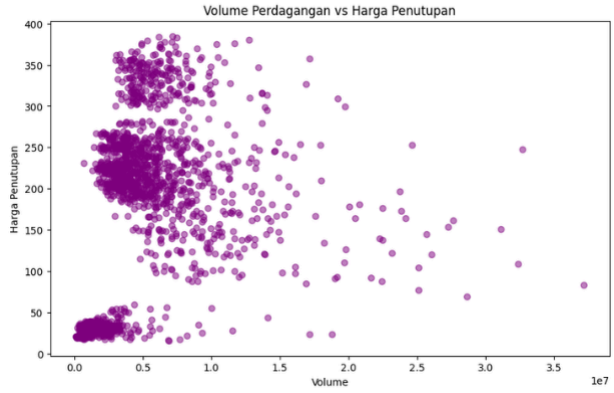
```
# Mengelompokkan data berdasarkan tahun
df['Year'] = df.index.year
grouped_df = df.groupby('Year').mean()
```

digunakan untuk mengelompokkan data berdasarkan satu atau lebih kolom dan kemudian melakukan operasi tertentu (seperti agregasi, transformasi, atau analisis)

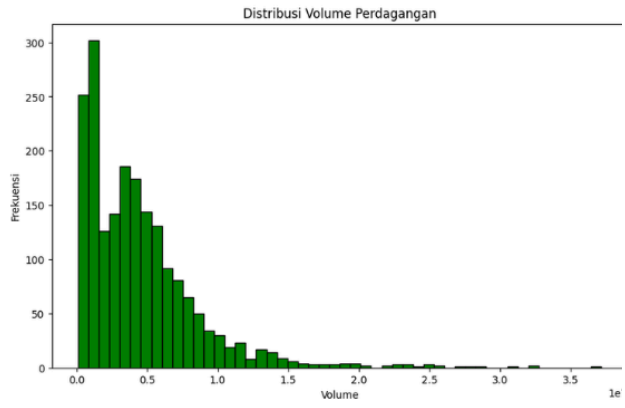
```
[12]: # Line plot untuk tren harga penutupan
plt.figure(figsize=(10, 6))
plt.plot(data['Date'], data['Close'], label='Harga Penutupan', color='blue')
plt.title('Tren Harga Penutupan Tesla')
plt.xlabel('Tanggal')
plt.ylabel('Harga')
plt.legend()
plt.show()
```



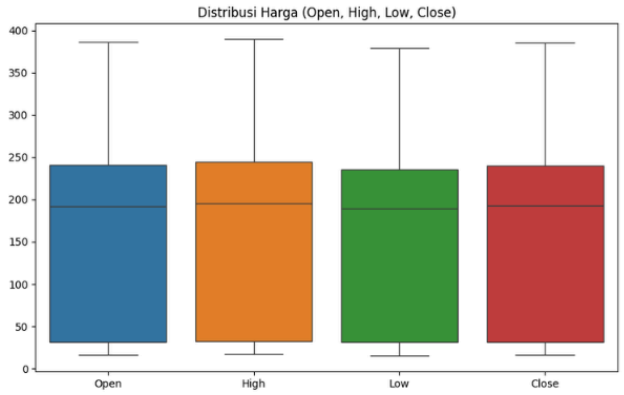

```
[13]: # Scatter plot untuk Volume vs Harga Penutupan
plt.figure(figsize=(10, 6))
plt.scatter(data['Volume'], data['Close'], alpha=0.5, color='purple')
plt.title('Volume Perdagangan vs Harga Penutupan')
plt.xlabel('Volume')
plt.ylabel('Harga Penutupan')
plt.show()
```



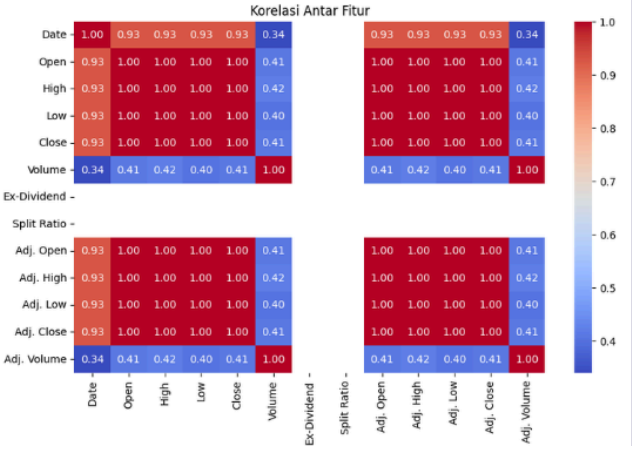

```
[14]: # Histogram distribusi volume perdagangan
plt.figure(figsize=(10, 6))
plt.hist(data['Volume'], bins=50, color='green', edgecolor='black')
plt.title('Distribusi Volume Perdagangan')
plt.xlabel('Volume')
plt.ylabel('Frekuensi')
plt.show()
```



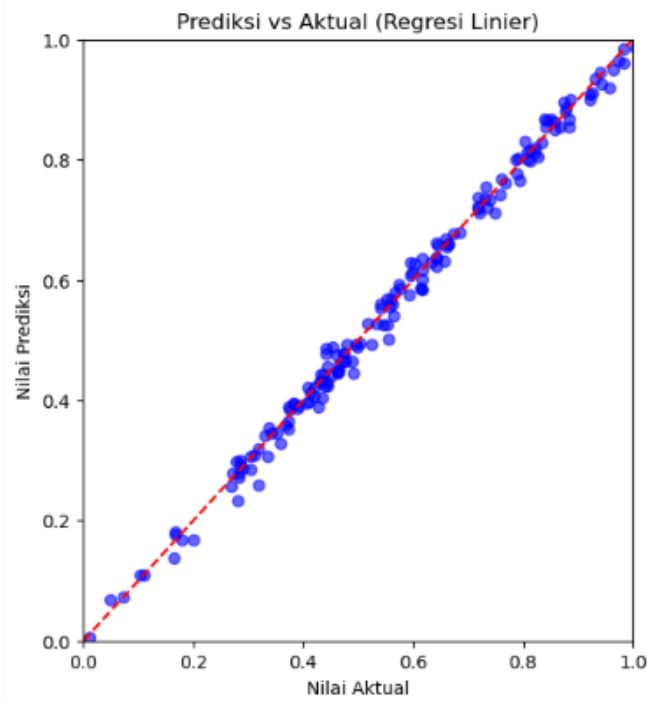

```
[15]: # Box plot untuk distribusi harga
plt.figure(figsize=(10, 6))
sns.boxplot(data=data[['Open', 'High', 'Low', 'Close']])
plt.title('Distribusi Harga (Open, High, Low, Close)')
plt.show()
```



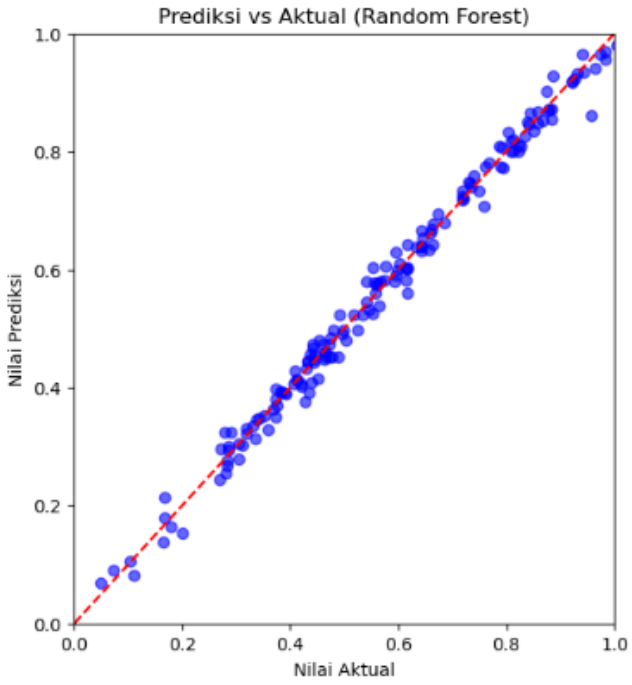

```
[16]: # Heatmap untuk korelasi fitur
plt.figure(figsize=(10, 6))
sns.heatmap(data.corr(), annot=True, cmap='coolwarm', fmt='.2f')
plt.title('Korelasi Antar Fitur')
plt.show()
```



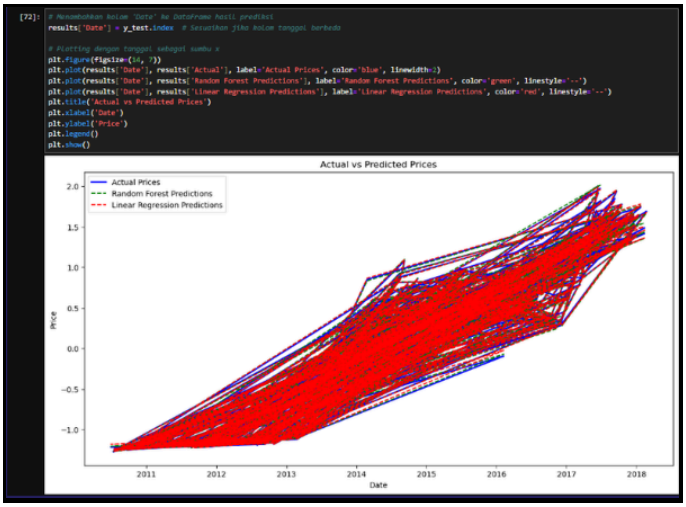
Regression



random forest



- **Sumbu X (Nilai Aktual):** Menunjukkan nilai sebenarnya dari data.
- **Sumbu Y (Nilai Prediksi):** Menunjukkan nilai yang diprediksi oleh model regresi linier.
- **Titik Biru:** Mewakili setiap pasangan nilai aktual dan nilai prediksi.
- **Garis Merah Putus-putus:** Mewakili garis referensi di mana nilai prediksi sama dengan nilai aktual (ideal).
- Grafik Regression ini menunjukkan bahwa model regresi linier cukup efektif dalam memprediksi nilai, dengan banyak titik yang dekat dengan garis referensi.
- Grafik Random Forest ini menunjukkan bahwa model Random Forest mampu memprediksi nilai dengan baik, dengan banyak titik yang dekat dengan garis referensi.



Visualisasi ini menunjukkan perbandingan antara harga aktual dan harga prediksi menggunakan dua model machine learning yang berbeda dalam rentang waktu 2011-2018.

Komponen utama dalam grafik:

- **Garis biru solid:** menunjukkan harga aktual
- **Garis putus-putus hijau:** prediksi menggunakan model Random Forest
- **Garis putus-putus merah:** prediksi menggunakan model Linear Regression

Dari visualisasi ini, dapat disimpulkan bahwa kedua model machine learning cukup baik dalam memprediksi harga, meskipun ada beberapa titik di mana terdapat perbedaan dengan harga aktual, terutama pada periode dengan volatilitas tinggi.

Evaluation

	Model	MSE	RMSE	MAE
0	Random Forest	0.000385	0.019622	0.012474
1	Linear Regression	0.000255	0.015969	0.010549

Mean Squared Error (MSE):
Random Forest: 0.000385
Regresi Linier: 0.000255

MSE untuk model Regresi Linier lebih rendah dibandingkan dengan Random Forest. Ini menunjukkan bahwa model Regresi Linier memiliki kesalahan kuadrat rata-rata yang lebih kecil, yang berarti prediksi model ini lebih dekat dengan nilai aktual.

Root Mean Squared Error (RMSE):

- Random Forest: 0.019622
- Regresi Linier: 0.015969

RMSE juga menunjukkan bahwa model Regresi Linier lebih baik, dengan nilai yang lebih rendah. RMSE memberikan ukuran kesalahan dalam satuan yang sama dengan variabel target, sehingga lebih mudah untuk diinterpretasikan.

Mean Absolute Error (MAE):

- Random Forest: 0.012474
- Regresi Linier: 0.010549

MAE untuk model Regresi Linier juga lebih rendah, menunjukkan bahwa rata-rata kesalahan absolut dari prediksi model ini lebih kecil dibandingkan dengan Random Forest. Ini menunjukkan bahwa model Regresi Linier lebih akurat dalam memprediksi nilai aktual

IV. KESIMPULAN

Mengapa Regresi Linier Lebih Akurat

Simplicity and Interpretability: Regresi Linier adalah model yang lebih sederhana dan lebih mudah diinterpretasikan. Dalam banyak kasus, model yang lebih sederhana dapat memberikan hasil yang lebih baik, terutama jika hubungan antara variabel independen dan dependen adalah linear atau mendekati linear.

Overfitting: Random Forest, meskipun merupakan model yang kuat, dapat cenderung overfit pada data pelatihan, terutama jika tidak diatur dengan baik (misalnya, jumlah pohon yang terlalu banyak atau kedalaman pohon yang terlalu dalam). Overfitting dapat menyebabkan model berkinerja buruk pada data yang tidak terlihat (data uji).

[1] [1] M. Fahrul Rizki Aditya, Nuril Lutvi Azizah. Prediksi Penyakit Hipertensi Menggunakan metode Decision Tree dan Random Forest. Jurnal Ilmiah KOMPUTASI, Volume 23 No:1, Maret 2024. https://ejournal.jak-stik.ac.id/files/journals/1/articles/Vol23No1Mar2024/3503/3503.pdf

[2] [2] Alvian, Edi Purnomo Putra. (2022). Perbedaan Classification, Regression, Dan Time Series Association. https://sis.binus.ac.id/2022/01/24/topik-8-perbedaan-classification-regression-dan-time-series-association/

[3] [3] L. N. Rupaidah, M. H. Syarif, and U. Enri, "Perbandingan Algoritma SARIMA dan Linear Regression dalam Memprediksi Indeks Harga Saham Gabungan," Krea-TIF: Jurnal Teknik Informatika, vol. 9, no. 2, pp. 42-49, Nov. 2021. doi: 10.3

[4] Dataset: https://www.kaggle.com/datasets/oddyvirgantara/harga-saham-tesla 2832/kreatif.v9i2.6291. Available: http://ejournal.uika-