**UNIVERSITAS ISLAM NEGERI SALATIGA**

# (UIN) SALATIGA

Jalan Lingkar Salatiga Km. 2, Pulutan, Sidorejo, Kota Salatiga, Jawa Tengah Telp. (0298) 323706 - Fax. (0298) 323433 e-mail: [administrasi@uinsalatiga.ac.id](mailto:administrasi@uinsalatiga.ac.id)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Mata Kuliah/SKS | : | Supervised Machine  Learning / 3 SKS | Nama  Anggota | : | 1. Bintang Gilangkasa Syailendra 2. Nabila Mutiara Sifia 3. Alya Salma Rosanty 4. Diah Tri Utami |
| Prodi/Jenjang | : | Sains Data /S1 | NIM  Anggota | : | 1. 23090230002 2. 23090230005 3. 23090230003 4. 23090230004 |
| Dosen Pengampu | : | Rahmatul Mahdalina, M.T.I | Kelompok | : | Grub 3 |

**LAPORAN PENGERJAAN UAS SUPERVISED MACHINE LEARNING**

## Deskripsi Praktek

* + 1. Nama Data : Data set harga saham Intel dari tahun 1980 hingga 2024.

Link Data : <https://www.kaggle.com/datasets/mhassansaboor/intel-stock-data-1980-2024>

Jumlah Data : 11.290 Data

Keterangan Data : Dataset ini dapat digunakan untuk analisis tren harga saham, peramalan harga, dan penelitian terkait pasar modal.

* + 1. Dalam tugas ini kami menggunakan dua model regresi yaitu: Linear Regression dan Decision Tree.
* Alasan Menggunakan Linear Regression

1. Kesederhanaan dan Interpretabilitas: Linear Regression adalah model yang sederhana dan mudah dipahami. Ini memungkinkan pengguna untuk dengan cepat menginterpretasikan hubungan antara variabel independen dan dependen, yang sangat berguna dalam konteks bisnis atau penelitian.
2. Asumsi Linearitas: Model ini bekerja dengan baik ketika hubungan antara variabel adalah linier. Jika data menunjukkan pola linier, Linear Regression dapat memberikan prediksi yang akurat.
3. Efisiensi Komputasi: Linear Regression relatif cepat dalam hal komputasi, membuatnya cocok untuk dataset besar.
4. Statistik yang Kuat: Model ini menyediakan berbagai statistik yang membantu dalam mengevaluasi kualitas model, seperti nilai R-squared, koefisien, dan p-value, yang memberikan wawasan tentang kekuatan dan signifikansi hubungan antar variabel.

* Alasan Menggunakan Decision Tree

1. Kemudahan Interpretasi: Decision Tree memiliki struktur visual yang jelas, mirip dengan pohon keputusan, membuatnya mudah dipahami bahkan oleh orang yang tidak memiliki latar belakang teknis. Ini sangat berharga dalam situasi di mana transparansi pengambilan keputusan diperlukan.
2. Penanganan Data Kategorikal dan Numerik: Decision Tree dapat menangani berbagai jenis data tanpa memerlukan transformasi khusus, sehingga cocok untuk dataset yang mengandung variabel kategorikal.
3. Kemampuan Menangani Non-linearitas: Model ini dapat menangkap hubungan non-linear antara fitur dan target variabel, menjadikannya fleksibel untuk berbagai jenis pola dalam dataset.
4. Minimal Preprocessing: Decision Tree tidak memerlukan normalisasi atau standarisasi data, serta dapat menangani nilai yang hilang secara efektif.
5. Pemilihan Fitur Otomatis: Dalam proses pembentukan pohon, Decision Tree secara otomatis melakukan seleksi fitur, membantu mengidentifikasi atribut-atribut paling informatif dalam dataset.
   * 1. Dalam tugas ini kami menggunakan Evaluasi Residual, alasan kami menggunakan evaluasi ini yaitu:
6. Memahami Kesalahan Prediksi

Residual adalah selisih antara nilai aktual dan nilai yang diprediksi oleh model. Dengan menganalisis residual, kita dapat memahami seberapa baik model dalam memprediksi data. Residual yang kecil menunjukkan bahwa model bekerja dengan baik, sementara residual yang besar menunjukkan adanya kesalahan yang perlu diperbaiki.

1. Identifikasi Pola dan Anomali

Evaluasi residual dapat membantu dalam mengidentifikasi pola atau anomali dalam data. Misalnya, jika residual menunjukkan pola tertentu (seperti tren atau siklus), ini bisa menjadi indikasi bahwa model tidak sepenuhnya menangkap hubungan dalam data. Hal ini penting untuk kedua model, karena dapat membantu dalam mengoptimalkan dan memperbaiki model.

1. Normalitas dan Homoskedastisitas

Dalam regresi linear, asumsi dasar adalah bahwa residual harus terdistribusi normal dan memiliki varians yang konstan (homoskedastisitas). Dengan memeriksa distribusi residual, kita dapat memastikan bahwa asumsi ini terpenuhi. Jika residual tidak memenuhi asumsi ini, maka hasil dari analisis regresi mungkin tidak valid.

1. Penyesuaian Model

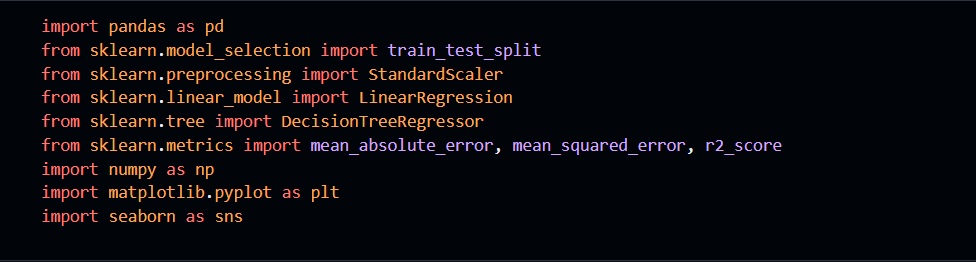
Untuk model decision tree, evaluasi residual juga digunakan untuk menyesuaikan model. Misalnya, jika residual menunjukkan kesalahan sistematik, kita dapat melakukan tuning pada hyperparameter atau memilih fitur yang lebih relevan untuk meningkatkan akurasi model.

1. Perbandingan Model

Evaluasi residual memungkinkan perbandingan antara berbagai model. Misalnya, ketika menggunakan decision tree dan regresi linear pada dataset yang sama, analisis residual dapat membantu menentukan model mana yang memberikan prediksi lebih baik berdasarkan ukuran kesalahan seperti MAE (Mean Absolute Error) atau RMSE (Root Mean Squared Error).

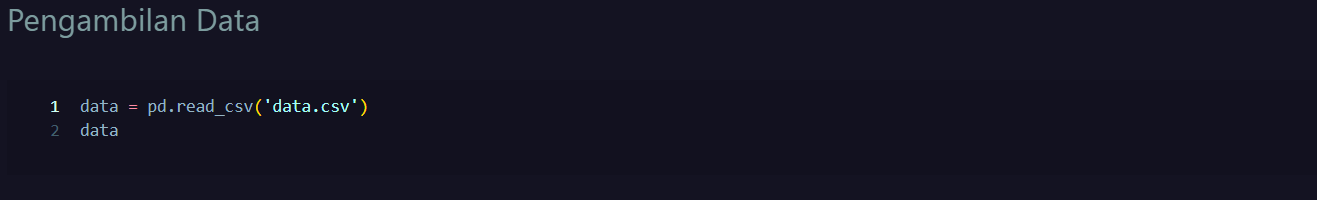
Dengan demikian, evaluasi residual adalah alat yang sangat berharga dalam pengembangan dan pemeliharaan model prediktif, baik itu decision tree maupun regresi linear. Analisis ini tidak hanya membantu dalam memahami kinerja model tetapi juga memberikan wawasan untuk perbaikan lebih lanjut.

* + **Praktek**



Kode berisi *import library* untuk beberapa pustaka python. Berikut adalah penjelasannya:

1. Import pandas: pandas adlah pustaka yang digunakan untuk manipulasi dan analisis data berbasis tabel
2. from sklearn.model\_selection import train\_test\_split: Fungsi train\_test\_split digunakan untuk membagi dataset menjadi dua bagian, yaitu data pelatihan (*training set*) dan data pengujian (*testing set*), dengan tujuan memvalidasi performa model
3. from sklearn.preprocessing import StandardScaler: StandardScaler digunakan untuk menstandarisasi fitur dengan menghilangkan rata-rata dan menskalakan ke varians satuan
4. from sklearn.linear\_model import LinearRegression: LinearRegression adalah model pembelajaran mesin untuk membuat prediksi berbasis regresi linear
5. from sklearn.tree import DecisionTreeRegressor: DecisionTreeRegressor adalah model regresi berbasis pohon keputusan
6. from sklearn.metrics import mean\_absolute\_error, mean\_squared\_error, r2\_score: mean\_absolute\_error (MAE) digunakan untuk mengukur rata-rata perbedaan absolut antara prediksi dan nilai aktual. mean\_squared\_error (MSE) digunakan untuk mengukur rata-rata kuadrat perbedaan antara prediksi dan nilai aktual. Lebih sensitif terhadap kesalahan besar dibanding MAE. r2\_score: Menilai seberapa baik model dapat menjelaskan variansi dalam data (*goodness of fit*).
7. import numpy as np: NumPy adalah pustaka untuk komputasi numerik, seperti operasi pada array multidimensi, fungsi matematika, aljabar linier, dan lain-lain.
8. import matplotlib.pyplot as plt: Matplotlib adalah pustaka visualisasi untuk membuat grafik atau plot
9. import seaborn as sns: eaborn adalah pustaka visualisasi data tingkat tinggi yang dibangun di atas Matplotlib
10. Pengambilan data



Baris pertama, pengambilan data CSV dilakukan dengan menggunakan fungsi pd.read\_csv(). Data CSV diberi nama ‘data.csv’. Data adalah variabel yang digunakan untuk menyimpan data yang telah dimuat dalam bentuk *DataFrame* Pandas. Baris kedua, variabel data berisi data file CSV yang dimuat.

1. Preprocessing



Pada fungsi data = data.ffill(), ffill() adalah metode dari Pandas yang digunakan untuk menerapkan *forward fill* pada data yang hilang (*missing values*). Tujuannya adalah untuk menangani *missing values* tanpa kehilangan data.

Pada fungsi X = data.drop(['Date', 'Close'], axis=1), drop() digunakan untuk menghapus kolom atau baris dalam *DataFrame*. ['Date', 'Close'] menunjukkan kolom yang akan dihapus. axis=1 menunjukkan bahwa penghapusan dilakukan pada kolom (bukan baris).

Fungsi train\_test\_split berguna membagi dataset menjadi data pengujian. X dan Y masing-masing adlah fitur dan target. test\_size=0.2 artinya 20% dari data akan digunakan untuk pengujian. random\_state=42 Digunakan untuk memastikan pembagian data tetap konsisten setiap kali kode dijalankan. X\_train & y\_train merupakan data pelatihan untuk melatih model. Sedangkan X\_test & y\_test merupakan Data pengujian untuk mengevaluasi model.

Fungsi StandardScaler berguna untuk menstandarisasi fitu (skala pada data) agar memiliki rata-rata 0 dan standar devisiasi 1. scaler.fit\_transform(X\_train) berfungsi untuk melatih skala berdasarkan data pelatihan (fit) dan menerapkan transformasi pada data pelatihan (transform). Selanjutnya, scaler.transform(X\_test) berfungsi menerapkan skala yang sama dari pelatihan data uji.

1. Training data



Alur pertama merupakan membuat dan melatih model. LinearRegression() berfungsi membuat model objek regresi linear yang berasal dari Scikit-Learn. Model ini akan bekerja menemukan hubungan linear antara fitu (x) dan targat (y). selanjutnya, fit(X\_train\_scaled, y\_train) digunakan untuk melatih model regresi linear menggunakan data pelatihan dengan X\_train\_scaled adalah data fitur yang telah diberi skala dan y\_train adlah target atau label yang terkait dengan data pelatihan.

Alur kedua merupakan langkah untuk membuat prediksi. predict(X\_test\_scaled) berfungsi menggunakan model yang telah dilatih untuk memprediksi target berdasarkan data pengujian, yaitu X\_test\_scaled dimana data fitur pengujian telah diskalakan. Variabel y\_pred\_lr merupakan prediksi target untuk data yang diuji.

Alur ketiga merupakan evaluasi model. mean\_absolute\_error(y\_test, y\_pred\_lr) adalah fungsi yang mengukur rata-rata kesalahan absolut antara nilai aktual (y\_test) dan prediksi (y\_pred\_lr). MAE adalah metrik sederhana untuk mengukur kesalahan prediksi. mean\_squared\_error(y\_test, y\_pred\_lr) adalah fungsi yang mengukur perbedaan rata-rata kuadrat antara nilai aktual dan prediksi. np.sqrt(mse\_lr) adalah fungsi yang digunakan untuk menghitung akar kuadrat dari MSE untuk mendapatkan RMSE (Root Mean Squared Error). Selanjutnya, r2\_score(y\_test, y\_pred\_lr) berfungsi sebagai penghitung *coefficient of determination* (R²), yang menunjukkan seberapa baik model dapat menjelaskan variansi dalam data.

Alur keempat yaitu mencetak hasil evaluasi. Kode print(f'Linear Regression MAE: {mae\_lr}, MSE: {mse\_lr}, RMSE: {rmse\_lr}, R-Squared: {r2\_lr}') akan mencetak hasil evaluai model MAE, MSE, RMSE, dan R².



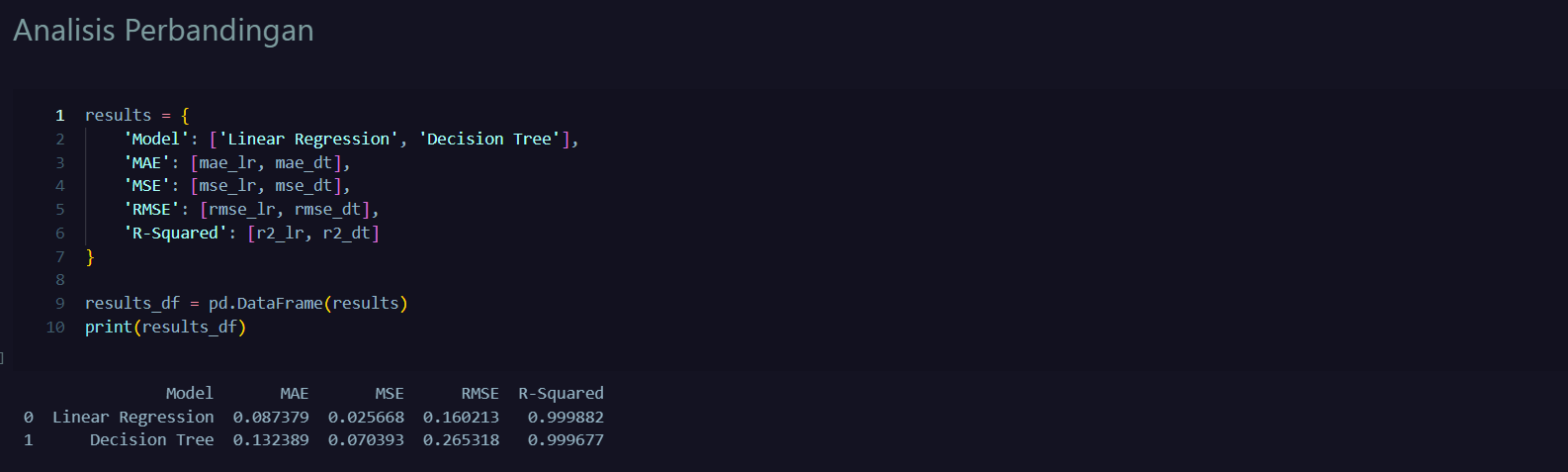
Langkah pertama yaitu membuat dan melatih model. Kode DecisionTreeRegressor() akan mengeksekusi dengan membuat objek regresi dengan decision tree. Selanjutnya, fit(X\_train\_scaled, y\_train) adalah kode yang melatih model dengan menggunakan data pelatihan dengan X\_train\_scaled adalah fitur data pelatihan yang telah diskalakan dan y\_train adalah target atau nilai yang ingin diprediksi.

Langkah kedua yaitu membuat prediksi. Dengan memanfaatkan fungsi predict(X\_test\_scaled) yang berfungsi memprediksi nilai target menggunaka model yang telah diuji berdasarkan data pengujian dengan X\_test\_scaled. y\_pred\_dt berfungsi sebagai prediksi target untuk data pengujian.

Langkah ketiga yaitu evaluasi model. Eksekusi diawali dengan fungsi mean\_absolute\_error(y\_test, y\_pred\_dt) yang akan mengukur rata-rata perbedaan absolut antara nilai aktual (y\_test) dan prediksi (y\_pred\_dt). Eksekusi selanjutnya dari fungsi mean\_squared\_error(y\_test, y\_pred\_dt) yang akan mengukur rata-rata kuadrat perbedaan antara nilai aktual dan prediksi. Setelah itu eksekusi kode berganti menjasi perhitungan akar kuadrat dari MSE untuk mendapatkan RMSE dengan menggunakan np.sqrt(mse\_dt). Selanjutnya eksekusi berpindah ke fungsi r2\_score(y\_test, y\_pred\_dt) dimana kode akan mengukur goodness of fit model terhadap data.

Langkah keempat yaitu mencetak hasil evaluasi. Fungsi print(f'Decision Tree MAE: {mae\_dt}, MSE: {mse\_dt}, RMSE: {rmse\_dt}, R-Squared: {r2\_dt}') akan mencetak hasil evaluasi model Decision Tree Regressor yang terdiri dari MAE, MSE, RMSE, dan R².

1. Pemodelan data



Eksekusi pertama dimulai dengan membuat kamu atau dictionary yang terdiri dari Model, MAE, MSE, RMSE, dan R-Squard. Eksekusi kedua yaitu membuat dataFrame dengan fungsi pd.DataFrame(results) yang mengubah kamus results menjadi objek pandas dataframe/ eksekusi selanjutnya adalah menampilkan DataFrame dengan print(results\_df).

1. Visualisasi data

A screenshot of a computer

Description automatically generated

Langkah pertama yaitu membuat kanvas untuk plot. Diawali dengan fungsi plt.figure(figsize=(20, 8)) dimana plt.figure() sebagai pembuat kanvas utama untuk menggambar plot dan figsize=(20, 8) sebagai penentu ukuran kanvas dalam satuan inci.

Langkah kedua adalah membuat subplot dan visualisasi. Subplot 1, MAE, berisi beberapa fungsi, yaitu plt.subplot(2, 3, 1) membagi kanvas menjadi grid 2 baris × 3 kolom, dan memilih posisi pertama untuk plot dan sns.barplot() membuat diagram batang menggunaka data dari result\_df. Subplo 2, MSE, mirip dengan subplot pertama, tetapi visualisasi menggunakan metrik MSE (Mean Squared Error). Subplot 3, RMSE, sama juga dengan subplot sebelumnya, tetapi visualisasi menggunakan metrik RMSE (Root Mean Squared Error). Subplot 4, R-Squared, visualisasi untuk R-Squared menunjukkan *goodness of fit* dari model yang ditampilkan.

Langkah ketiga adalah menyesuaikan tata letak dan menampilkan plot. plt.tight\_layout() berfungsi menyesuaikan tata letak subplot agar tidak saling tumpang tindih.

A graph of a bar graph

Description automatically generated with medium confidence

1. Perhitungan akurasi, presisi, dan recall



Pertama, menghitung residuals pada baris pertama dan kedua. Residual adalah selisih antara nilai aktual (y\_test) dan nilai prediksi (y\_pred\_lr) untuk Linear Regression, y\_pred\_dt untuk Decision Tree).

Kedua, pada baris keempat adalah pembuatan kanvas plot dengan ukuran 12 inci x 6 inci.

Ketiga, residuals linear regression yang terdapat pada baris keenam hingga kesembilan. Di dalam kode ini ada beberapa eksekusi, yaitu membagi kanvas menjadi 1 baris x 2 kolom, membuat histogram distribusi residuals dari metode linear regression, menambahkan kurva, dan membagi residuals menjadi 30 bin.

Keempat, residuals decision tree, diman eksekusi kode ini sama dengan Residuals Linear Regression, tetapi untuk residuals model Decision Tree. Kelima, menyesuaikan tata letak dan menampilkan plot yang terdapat pada kode baris ke 16 dan ke 17.

A graph of a line graph

Description automatically generated with medium confidence

1. Evaluasi data

## Keterangan Tanggung Jawab Anggota

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama** | **NIM** | **Tanggungjawab dalam Kelompok** |
| 1. | Bintang Gilangkasa Syailendra | 23090230002 | Menginput data dan membuat kode python |
| 2. | Nabila Mutiara Sifia | 23090230005 | Mencari data dan membuat praktek |
| 3. | Alya Salma Rosanty | 23090230003 | Membuat soal |
| 4. | Diah Tri Utami | 23090230004 | Membuat deskripsi praktek dan laporan |

## Soal

* + 1. Tuliskan perbandingan akurasi masing-masing model !
    2. Model mana yang memiliki performa lebih baik untuk prediksi ?
    3. Tuliskan kesimpulan berkaitan tentang model yang telak dilakukan pada praktek !
    4. Tuliskan kesimpulan berkaitan dengan data ! (contoh : jenis kelamin yang rawan diabetes adalah laki-laki)