Tugas 4: Tugas Mandiri 4 – Logistic Regression

**Amaya Eshia - 0110224102\***

1 Teknik Informatika, STT Terpadu Nurul Fikri, Depok

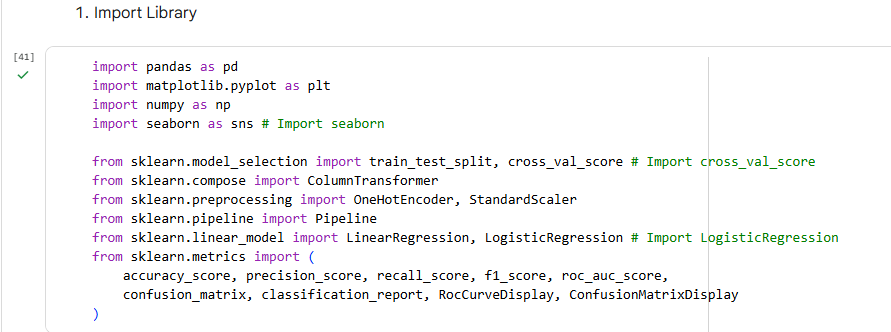
\*E-mail: [name@institution.edu](mailto:name@institution.edu) **– 0110224102@student.nurulfikri.ac.id**

**Abstract.** Penelitian ini bertujuan untuk membangun dan mengevaluasi model prediktif keputusan pembelian mobil oleh calon pelanggan menggunakan Regresi Logistik (Logistic Regression) berdasarkan data survei. Fitur yang digunakan dalam pemodelan meliputi Usia, Penghasilan, Status, Kelamin, dan Jumlah Kepemilikan Mobil.

**Kata Kunci:** Regresi Logistik, Prediksi Pembelian, Odds Ratio, Akurasi, Regresi, Logistik, DataSurvei, Model Prediktif

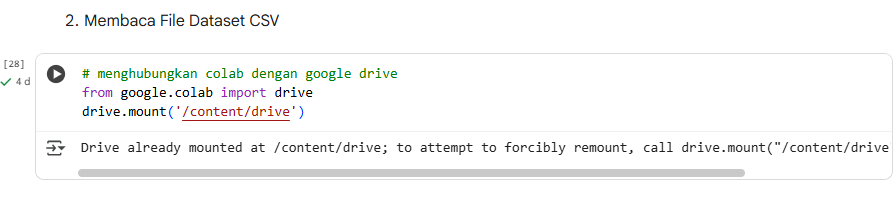
1. Penjelasan Koding Program

1. Import Library



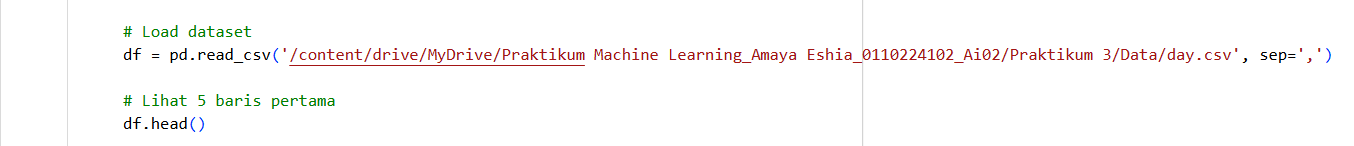
ini adalah melakukan import terhadap seluruh library yang dibutuhkan. Library berfungsi sebagai kumpulan fungsi dan modul pendukung agar proses pemodelan, analisis data, dan visualisasi dapat dilakukan dengan lebih mudah dan efisien.

2. Membaca File Dataset CSV



Selanjutnya, menggunakan library Pandas untuk membaca file data. Variabel path menyimpan lokasi folder di Google Drive / tempat file dataset berada. Fungsi pd.read\_csv() kemudian membaca file tersebut dan menyimpannya ke dalam sebuah DataFrame.

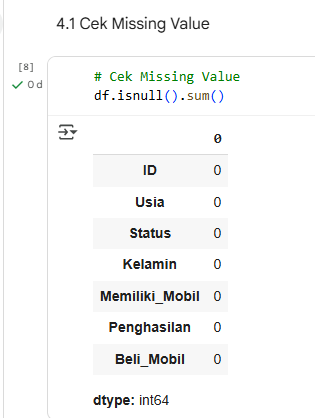
3. Melihat Informasi Umum Dataset



Langkah selanjutnya adalah membaca dataset calonpembelimobilt.csv menggunakan library pandas, kemudian menampilkan informasi struktur data menggunakan fungsi df.info(). Berdasarkan hasil dari df.info()

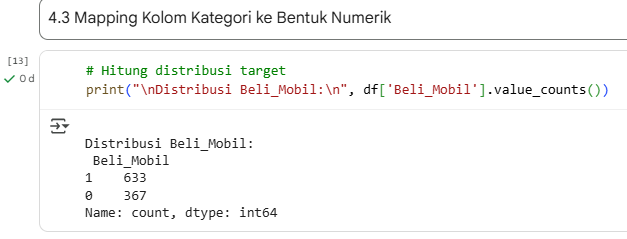
4. DataPre-processing

1.4 Cek Missing Value



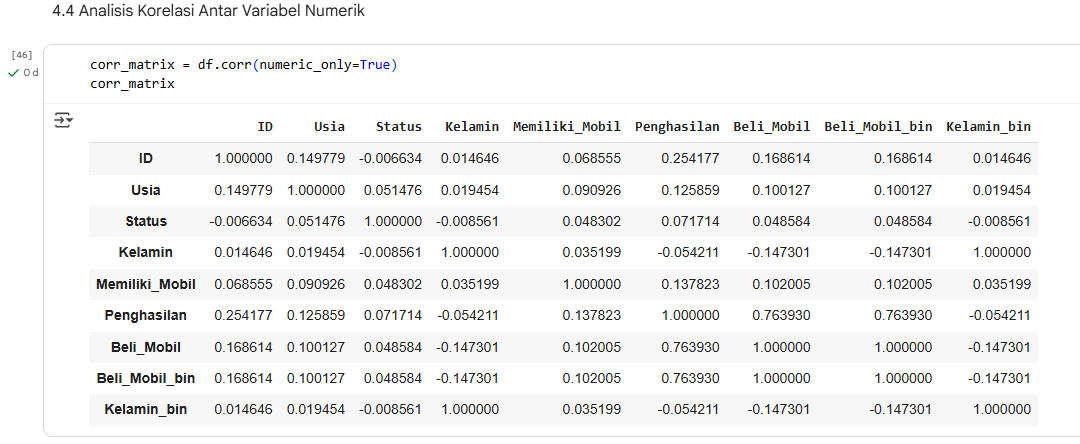
Perintah ini digunakan untuk memastikan apakah terdapat data yang hilang pada setiap kolom. Hasil menunjukkan seluruh kolom (ID, Usia, Status, Kelamin, Memiliki\_mobil, Penghasilan, Beli\_mobil) memiliki nilai 0 pada jumlah missing value. Artinya, tidak ada data kosong, sehingga dataset dapat langsung digunakan tanpa proses imputasi.

1.5 Mapping Kolom Kategori ke Bentuk Numerik



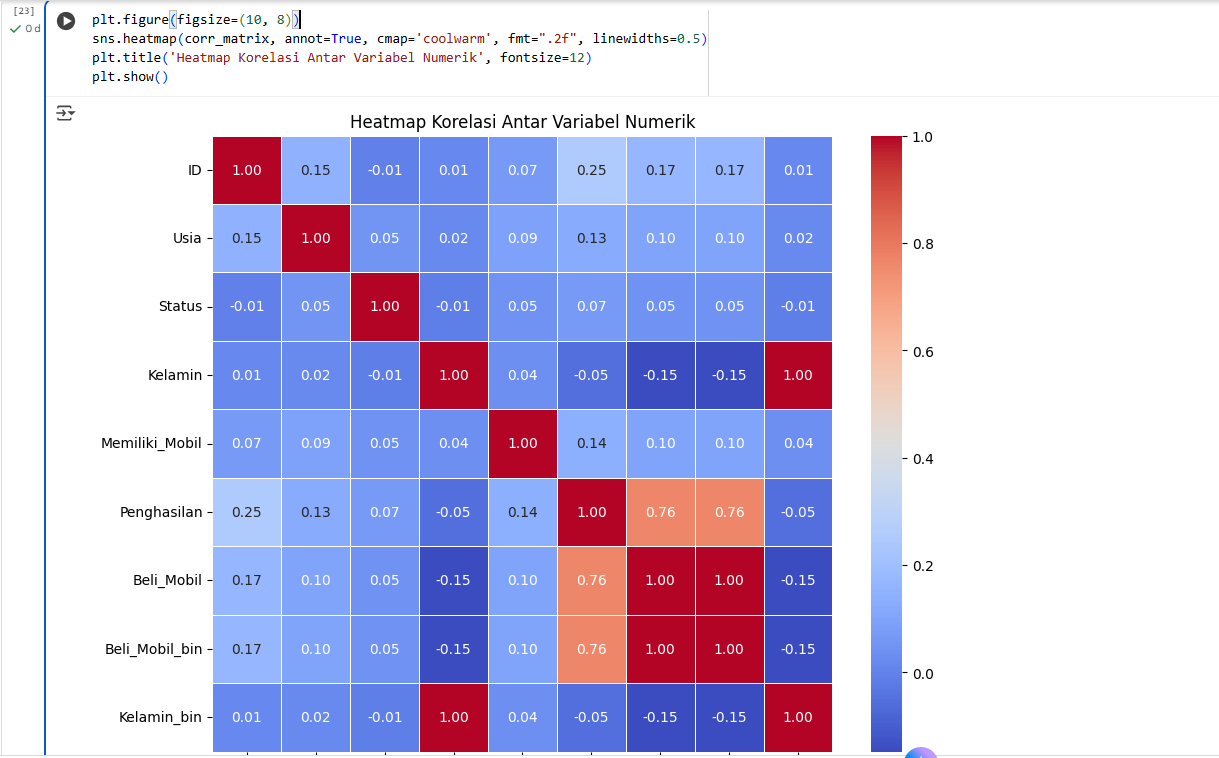
Kode di atas mengubah nilai teks menjadi bentuk numerik agar dapat diproses model

1.6 Analisis Korelasi Antar Variabel Numerik



Menghitung matriks korelasi variabel numerik menggunakan .corr(numeric\_only=True) dan menyimpannya dalam corr\_matrix.

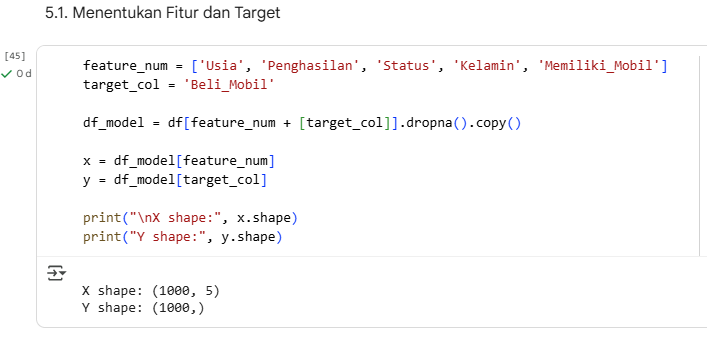
1.7 Visualisasi Heatmap Korelasi



Membuat heatmap korelasi menggunakan Seaborn dengan plt.figure(figsize=(10, 8)), sns.heatmap(corr\_matrix, annot=True, cmap='coolwarm', fmt=".2f", linewidths=0.5), dan plt.title('Heatmap Korelasi Antar Variabel Numerik') untuk visualisasi hubungan antar variabel.

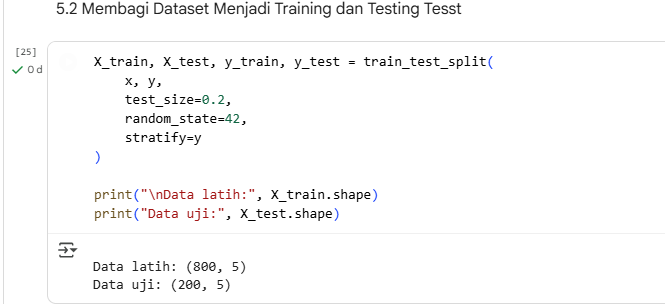
5. Pembagian Dataset (Training dan Testing)

5.1 Menentukan Fitur dan Target



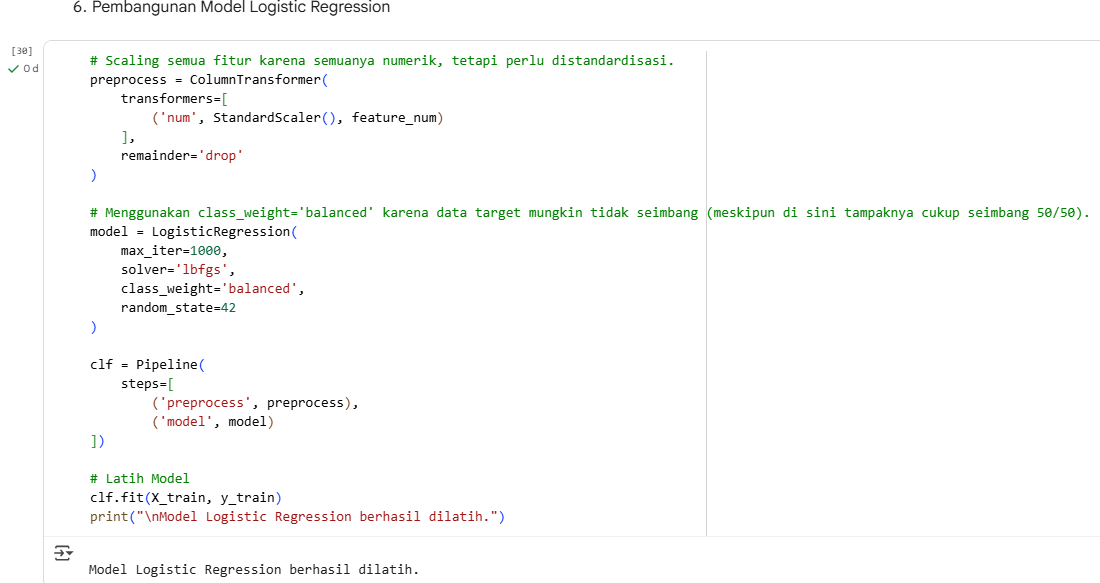
Memilih fitur numerik seperti ['Usia', 'Penghasilan', 'Status', 'Kelamin', 'Memiliki\_Mobil'] sebagai X dan 'Beli\_Mobil' sebagai y. Menghapus baris dengan missing values menggunakan .dropna() dan menampilkan shape X serta y.

5.2 Membagi Dataset Menjadi Training dan Testing



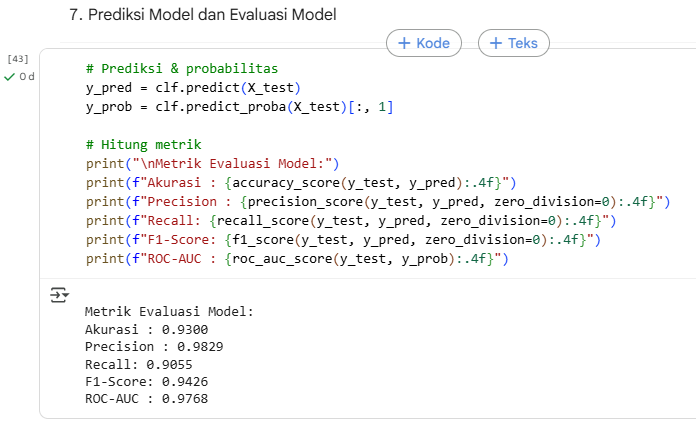
Membagi data menjadi training (80%) dan testing (20%) menggunakan train\_test\_split dengan test\_size=0.2, random\_state=42, dan stratify=y untuk menjaga keseimbangan kelas. Menampilkan ukuran data training dan testing untuk verifikasi.

6. Pembangunan Model Logistic Regression



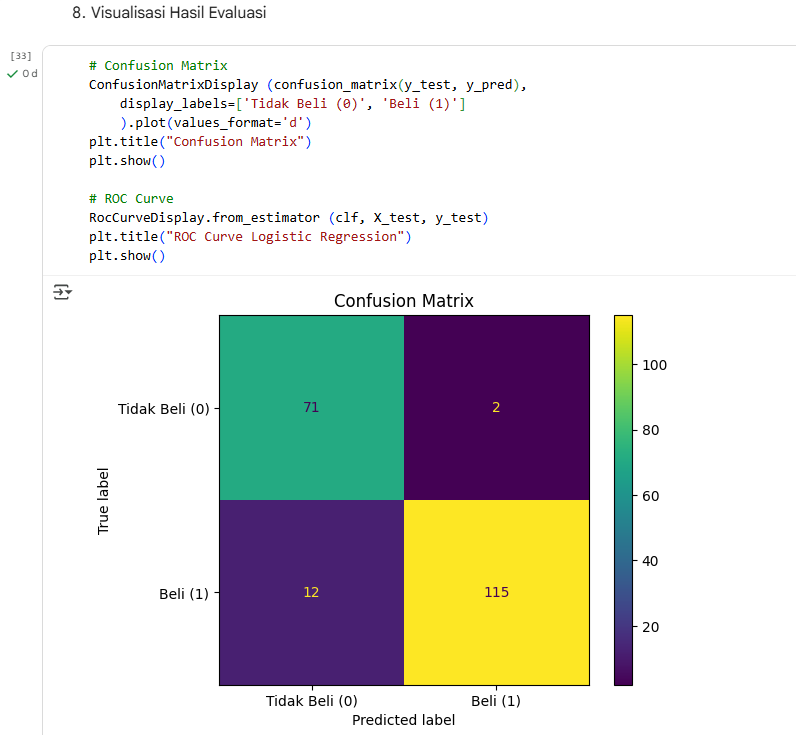
Membuat preprocessor dengan ColumnTransformer yang menerapkan StandardScaler pada fitur numerik. Membuat model LogisticRegression dengan max\_iter=1000, solver='lbfgs', class\_weight='balanced', dan random\_state=42. Menggabungkan preprocessor dan model dalam Pipeline bernama clf. Melatih model menggunakan clf.fit(X\_train, y\_train) dan menampilkan pesan sukses.

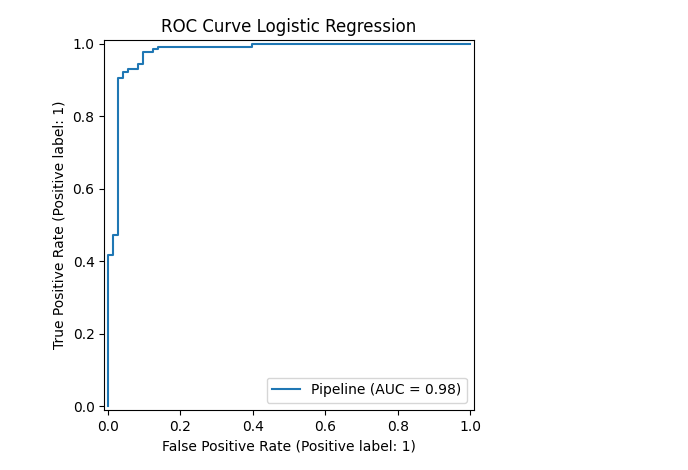
7. Prediksi Model dan Evaluasi Model



Melakukan prediksi pada data testing menggunakan clf.predict(X\_test) dan probabilitas menggunakan clf.predict\_proba(X\_test)[:, 1]. Menghitung metrik evaluasi seperti accuracy\_score, precision\_score, recall\_score, f1\_score, dan roc\_auc\_score, kemudian menampilkannya dengan format 4 desimal.

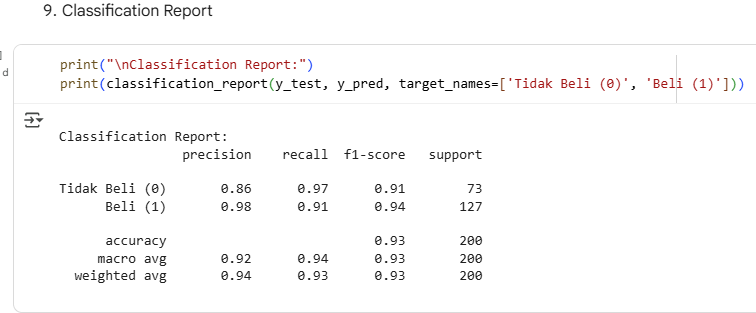
8. Visualisasi Hasil Evaluasi





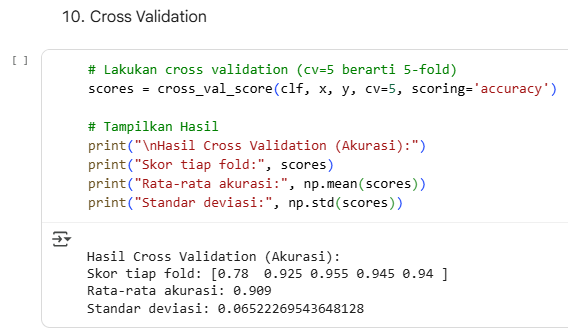
· Membuat Confusion Matrix Display menggunakan ConfusionMatrixDisplay dengan display\_labels=['Tidak Beli (0)', 'Beli (1)'] dan plt.title("Confusion Matrix"). · Membuat ROC Curve menggunakan RocCurveDisplay.from\_estimator dengan plt.title("ROC Curve Logistic Regression").

9. Classification Report



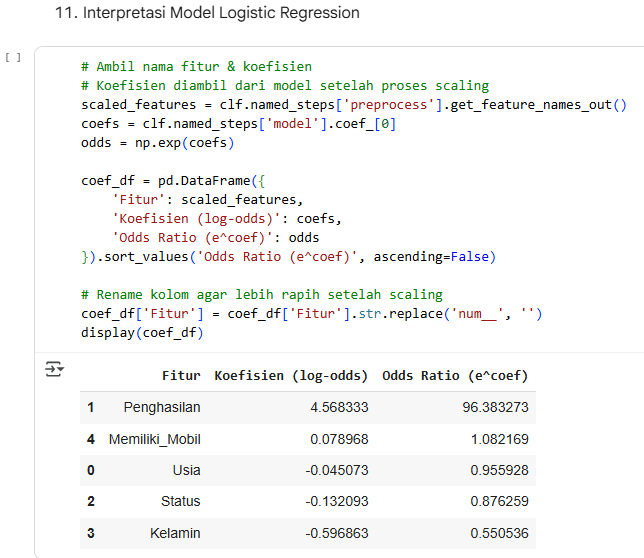
Menampilkan classification\_report dengan target\_names=['Tidak Beli (0)', 'Beli (1)'] untuk detail precision, recall, dan F1-score per kelas.

10. Cross Validation



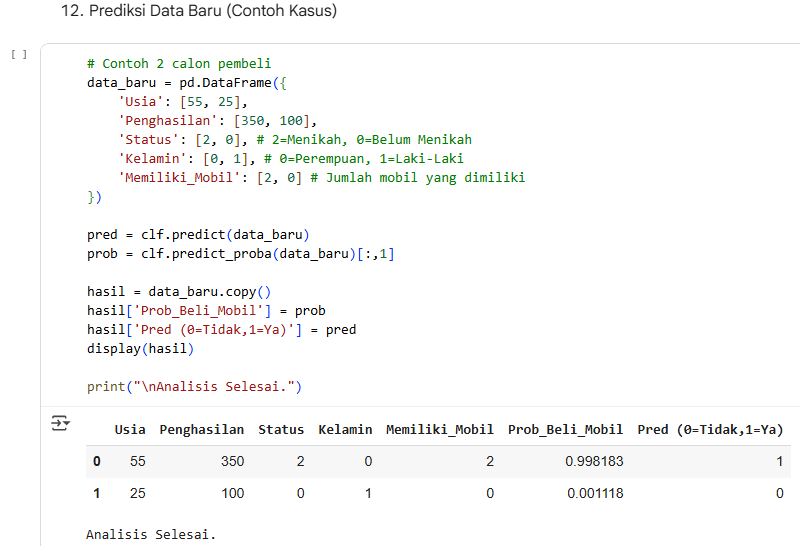
Melakukan 5-fold cross-validation menggunakan cross\_val\_score dengan scoring='accuracy'. Menampilkan skor tiap fold, rata-rata akurasi, dan standar deviasi.

11. Interpretasi Model Logistic Regression



Mengambil nama fitur setelah scaling dari clf.named\_steps['preprocess'].get\_feature\_names\_out(), koefisien dari clf.named\_steps['model'].coef\_[0], dan odds ratio menggunakan np.exp(coefs). Membuat DataFrame coef\_df dengan kolom 'Fitur', 'Koefisien (log-odds)', dan 'Odds Ratio (e^coef)', diurutkan descending berdasarkan odds ratio. Membersihkan nama fitur dengan .str.replace('num\_\_', '') dan menampilkan tabel.

12. Prediksi Data Baru (Contoh Kasus)



Membuat DataFrame data\_baru dengan 2 contoh calon pembeli (usia 55 dan 25). Melakukan prediksi menggunakan clf.predict dan probabilitas menggunakan clf.predict\_proba[:,1]. Menambahkan kolom 'Prob\_Beli\_Mobil' dan 'Pred (0=Tidak,1=Ya)' ke DataFrame hasil, kemudian menampilkannya. Menampilkan pesan "Analisis Selesai." dan kesimpulan probabilitas pembelian.

Referensi:

Munir, S., Seminar, K. B., Sudradjat, Sukoco, H., & Buono, A. (2022). The Use of Random Forest Regression for Estimating Leaf Nitrogen Content of Oil Palm Based on Sentinel 1-A Imagery. *Information*, *14*(1), 10. https://doi.org/10.3390/info14010010

Seminar, K. B., Imantho, H., Sudradjat, Yahya, S., Munir, S., Kaliana, I., Mei Haryadi, F., Noor Baroroh, A., Supriyanto, Handoyo, G. C., Kurnia Wijayanto, A., Ijang Wahyudin, C., Liyantono, Budiman, R., Bakir Pasaman, A., Rusiawan, D., & Sulastri. (2024). PreciPalm: An Intelligent System for Calculating Macronutrient Status and Fertilizer Recommendations for Oil Palm on Mineral Soils Based on a Precision Agriculture Approach. *Scientific World Journal*, *2024*(1). https://doi.org/10.1155/2024/1788726