**LAPORAN SINGKAT**

**1. Deskripsi Dataset**

* Nama file: income.csv (Adult Income Dataset).
* Ukuran awal: 48.842 baris × 15 kolom (ditampilkan saat pembacaan).
* Kolom utama (sample): age, workclass, fnlwgt, education, educational-num, marital-status, occupation, relationship, race, gender, capital-gain, capital-loss, hours-per-week, native-country, income.
* Missing values (terdeteksi sebelum preprocessing):
* workclass: 2.799
* occupation: 2.809
* native-country: 857
* (kolom lain: 0 missing)
* Preprocessing yang diterapkan dalam skrip:
* Trim whitespace pada cell string.
* Ganti ? → NaN.
* Menghapus (drop) semua baris yang mengandung missing (cara sederhana; catatan: ini mengurangi jumlah baris sebelum training).
* Pemisahan fitur numerik dan kategorikal; numerik distandar (StandardScaler), kategorikal di-OneHotEncode.
* Catatan: histogram umur (report/hist\_age.png) dibuat sebagai bagian dari EDA.

**2. Model yang digunakan**

Dua algoritma klasifikasi yang dipakai (pipeline = preprocessing + classifier):

1. Logistic Regression

* Solver: liblinear (atau default di skrip).
* Digunakan sebagai baseline linear, probabilistik (menghasilkan predict\_proba).

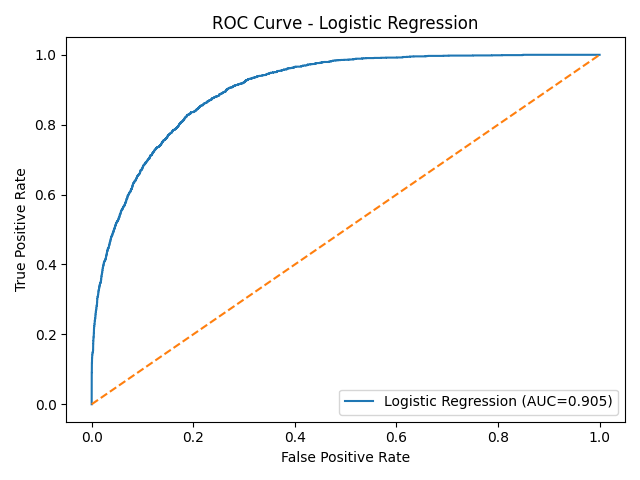
1. Decision Tree Classifier

* Random state diset (untuk reproduksibilitas).
* Memberi model non-linear / tree-based dengan interpretasi aturan.

**3. Hasil Evaluasi dan Pembahasan**

3.1 Confusion Matrix & metrik (di test set)

**Logistic Regression**

****

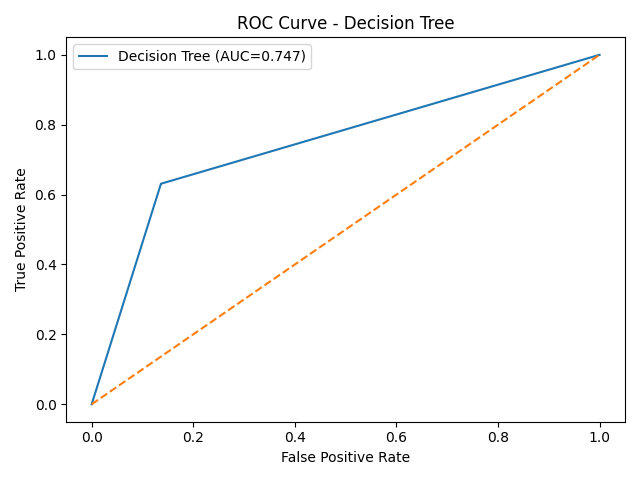
* Confusion Matrix:

[[6313 490]

[ 899 1343]]

* True Negative (TN) = 6313
* False Positive (FP) = 490
* False Negative (FN) = 899
* True Positive (TP) = 1343
* Metrik:
* Accuracy : 0.8464
* Precision: 0.7327
* Recall : 0.5990
* F1-score : 0.6591
* ROC-AUC : 0.9047

**Decision Tree**

****

* Confusion Matrix:

[[5875 928]

[ 827 1415]]

* TN = 5875
* FP = 928
* FN = 827
* TP = 1415
* Metrik:
* Accuracy : 0.8060
* Precision: 0.6039
* Recall : 0.6311
* F1-score : 0.6172
* ROC-AUC : 0.7473

3.2 ROC Curve (lokasi file)

* report/roc\_Logistic\_Regression.png → AUC = 0.905 (sekitar 0.9047) — kurva ROC mendekati sudut kiri-atas (performansi sangat baik).
* report/roc\_Decision\_Tree.png → AUC = 0.747 (sekitar 0.7473) — performansi moderat.

3.3 Pembahasan singkat

* Logistic Regression tampil lebih baik secara keseluruhan:
* AUC jauh lebih tinggi (0.905 vs 0.747) — artinya model logistic regression lebih baik membedakan kelas >50K vs <=50K di berbagai threshold.
* Precision dan F1-score Logistic lebih tinggi → trade-off antara false positives dan false negatives lebih baik terjaga.
* Decision Tree menunjukkan recall sedikit lebih tinggi (0.6311 vs 0.5990):
* Artinya Decision Tree menemukan lebih banyak instance positif (lebih sedikit FN), tetapi membayar dengan banyak FP (precision turun).
* Jika tujuanmu adalah menangkap sebanyak mungkin individu berpenghasilan >50K (toleran terhadap FP), Decision Tree atau threshold yang dioptimalkan bisa dipertimbangkan.
* Accuracy: Logistic > Decision Tree (0.8464 vs 0.8060). Namun accuracy sendiri tidak selalu cukup andal bila kelas imbalanced — AUC dan F1 memberi gambaran lebih lengkap.
* Catatan tentang data & preprocessing:
* Skrip menghapus baris yang mengandung missing (dropna). Ini sederhana tetapi dapat mengurangi representasi kelompok tertentu (mis. kategori workclass atau native-country). Imputasi (mode/most frequent atau model-based) bisa meningkatkan data utilitas.
* Banyak fitur kategorikal (OneHot) → model linear (Logistic) bekerja baik setelah encoding dan scaling.
* Decision Tree cenderung overfit bila tidak disetel hyperparameter (max\_depth, min\_samples\_leaf). Hasil AUC lebih rendah menunjukkan perlu tuning atau ensemble.

1. **Kesimpulan**

Berdasarkan eksperimen saat ini, Logistic Regression memberikan performa terbaik secara keseluruhan untuk tugas prediksi pendapatan (Accuracy 0.846, F1 0.659, ROC-AUC ≈ 0.905), sedangkan Decision Tree memiliki recall sedikit lebih tinggi namun precision dan AUC lebih rendah (Accuracy 0.806, F1 0.617, ROC-AUC ≈ 0.747). Oleh karena itu, untuk kebutuhan umum (menyeimbangkan kesalahan tipe I & II) Logistic Regression direkomendasikan sebagai model baseline yang solid; langkah selanjutnya adalah melakukan hyperparameter tuning, perbaikan penanganan missing data, dan mencoba ensemble untuk potensi peningkatan performa lebih lanjut.