LAPORAN UAS

KECERDASAN BUATAN

Disusun :

Nama : Olivia Ramadhani

NIM : 231011402280

Kelas : 05TPLE013

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS PAMULANG

2026

BAGIAN 1 — Pemahaman Konsep (Teori)

1.1 Konsep Dasar Decision Tree

- Definisi
    - Decision Tree adalah metode pembelajaran mesin yang membangun struktur pohon untuk memetakan fitur input menjadi output (prediksi).
    - Untuk regresi, outputnya nilai kontinu (contoh: Exam_Score).
- Cara kerja umum
    - Data dipecah berulang (split) berdasarkan fitur tertentu agar tiap cabang menjadi semakin "homogen" terhadap target.
    - Proses pemecahan berhenti saat mencapai kondisi tertentu (mis. kedalaman maksimal, jumlah data minimum, atau tidak ada peningkatan kualitas split).

1.2 Struktur Pohon (Komponen)

- Node
    - Titik pada pohon yang merepresentasikan keputusan/aturan (mis. Attendance <= 80).
- Root
    - Node paling atas (awal). Split pertama yang biasanya paling "informatif".
- Leaf
    - Node akhir tempat prediksi dibuat.
    - Pada regresi, leaf menyimpan nilai prediksi (biasanya rata-rata target pada node tersebut).
- Splitting
    - Proses memilih fitur dan ambang batas untuk memisahkan data.
    - Tujuan: membuat masing-masing subset data lebih "seragam" terhadap target.
- Pruning
    - Teknik untuk mengurangi kompleksitas pohon dengan memangkas cabang yang kurang penting.
    - Tujuan: mengurangi overfitting dan meningkatkan generalisasi.

1.3 Kriteria Split pada Regresi (Impurity)

- Tujuan split regresi
    - Meminimalkan error dalam node/leaf.

- Contoh kriteria yang umum
    - MSE / squared_error: memilih split yang menurunkan Mean Squared Error.
    - friedman_mse: variasi untuk membantu performa pada beberapa kondisi.

.

1.4 Perbedaan Decision Tree, Random Forest, dan Gradient Boosting

- Decision Tree
    - Satu pohon, mudah diinterpretasi, tetapi rentan overfitting bila terlalu dalam.
- Random Forest
    - Kumpulan banyak decision tree (ensemble) dengan teknik bagging dan random feature selection.
    - Lebih stabil dan biasanya lebih akurat daripada single tree.
- Gradient Boosting
    - Ensemble yang membangun model secara bertahap (stage-wise), tiap pohon memperbaiki error model sebelumnya.
    - Biasanya sangat kuat, tetapi lebih sensitif terhadap parameter dan berpotensi lebih mudah overfit jika tidak dituning.

1.5 Kelebihan dan Kekurangan Tree-Based Methods

- Kelebihan
    - Bisa menangani hubungan non-linear dan interaksi fitur.
    - Tidak membutuhkan scaling seperti model linear tertentu.
    - Bisa menangani data campuran (numerik + kategorikal) dengan encoding yang tepat.
    - Interpretabilitas relatif baik (khususnya single tree).
- Kekurangan
    - Single tree mudah overfitting.
    - Sensitif terhadap perubahan kecil pada data (tree bisa berubah struktur).
    - Untuk kategorikal dengan banyak kategori (one-hot), dimensi fitur bisa membesar.
    - Interpretasi jadi sulit jika pohon sangat dalam / jumlah leaf banyak.

BAGIAN 3 — Metodologi

3.1 Ringkasan Dataset dan Tujuan

- Dataset: Student Performance Factors
- Target: Exam_Score (regresi)
- Tujuan: memprediksi nilai ujian berdasarkan faktor akademik, kebiasaan belajar, dan faktor lingkungan.

3.2 Ringkasan Tahapan Eksperimen

- EDA
    - Jumlah data: 6.607 baris
    - Tipe data: kombinasi numerik dan kategorikal
    - Missing value: kecil (<1%), diisi dengan median/modus
    - Korelasi tertinggi terhadap Exam_Score:
        - Attendance (0.58)
        - Hours Studied (0.45)
        - Parental Involvement (+1.7 poin efek rata-rata)
- Preprocessing
    - Imputasi median untuk data numerik.
    - Imputasi modus untuk data kategorikal.
    - One-Hot Encoding untuk kolom kategorikal.
- Split data
    - Train-test split 80:20 (random_state=42)
- Modeling
    - Baseline: DecisionTreeRegressor default
    - Tuning: GridSearchCV (cv=5) untuk parameter:
        - max_depth
        - min_samples_split
        - min_samples_leaf

- criterion

3.3 Hasil Evaluasi Model

- Baseline Decision Tree (Default)
  - MAE : 1.7421
  - MSE : 12.5061
  - RMSE : 3.5364
  - R^2 : 0.1152 (opsional)
- **Best Model (Setelah Tuning)**
  - Best Params: {'model__criterion': 'squared_error', 'model__max_depth': None, 'model__min_samples_leaf': 10, 'model__min_samples_split': 2}
  - MAE: 1.4851
  - MSE: 6.2545
  - RMSE: 2.5009
  - R^2: 0.5575 (opsional)

3.4 Model Terbaik Berdasarkan Hasil Eksperimen

- Model terbaik dipilih berdasarkan metrik utama:
  - Prioritas: MAE terendah (karena scoring tuning juga MAE)
  - Didukung oleh MSE/RMSE yang lebih kecil
- Alasan pemilihan:
  - Baseline (default) menggunakan parameter bawaan Decision Tree, sehingga struktur pohon bisa terlalu kompleks atau kurang sesuai dengan karakter data. Akibatnya performa di data uji bisa kurang optimal (lebih berisiko overfitting).
  - Tuned (setelah GridSearchCV) memilih kombinasi parameter terbaik (misalnya max_depth, min_samples_leaf, min_samples_split, dan criterion) sehingga kompleksitas pohon lebih terkontrol. Dampaknya, model biasanya lebih mampu generalisasi ke data uji dan menghasilkan error yang lebih kecil.
  - Secara metrik, jika hasil tuning lebih baik maka terlihat dari MAE/MSE/RMSE yang menurun (dan opsional R^2 meningkat) dibanding baseline

3.5 Faktor yang Mempengaruhi Performa Model

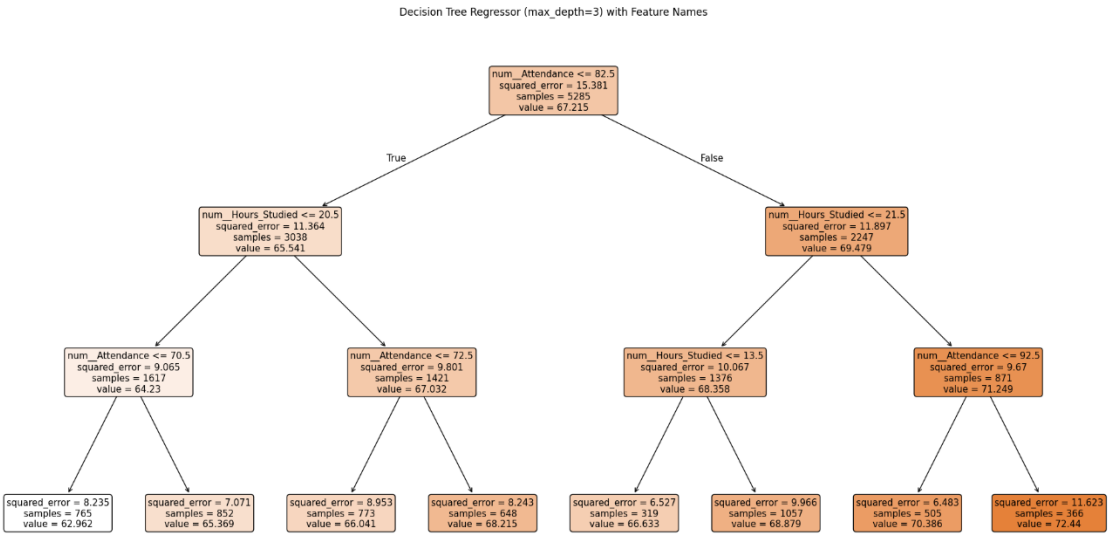
- Kompleksitas model (overfitting vs underfitting)

- max_depth terlalu besar → overfitting
- max_depth terlalu kecil → underfitting

- Regularisasi via parameter minimum sampel
  - min_samples_leaf lebih besar → model lebih stabil, biasanya generalisasi lebih baik
  - min_samples_split mengontrol kapan node boleh dipecah
- Kualitas preprocessing
  - Penanganan missing value (imputasi) mempengaruhi stabilitas split
  - One-hot encoding meningkatkan jumlah fitur → bisa mempengaruhi struktur tree
- Pembagian data
  - train-test split dan variasi data mempengaruhi hasil evaluasi

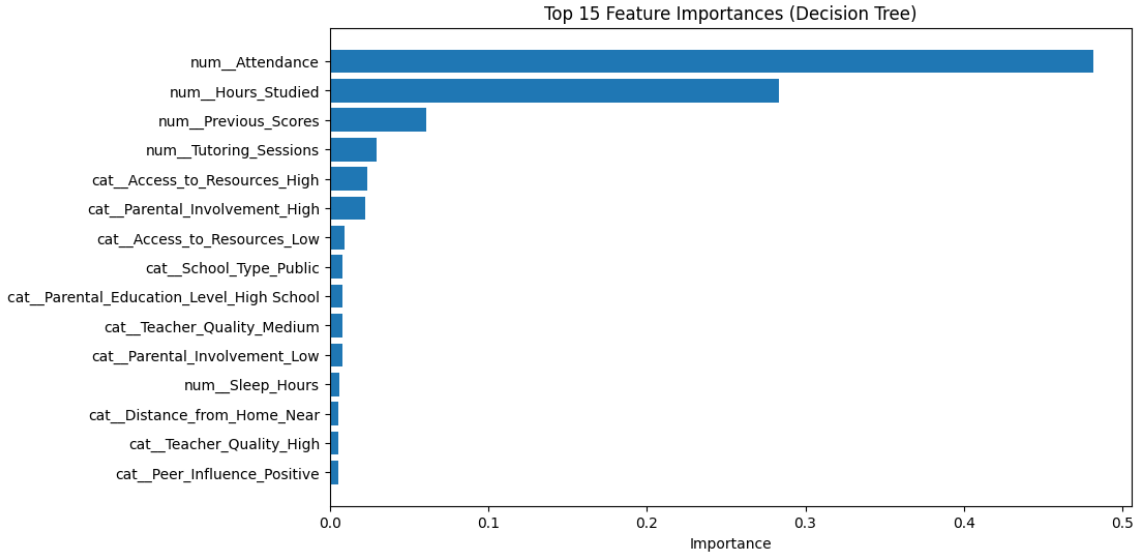
3.6 Visualisasi Pohon dan Interpretasi

- Visualisasi pohon dengan max_depth=3 digunakan agar aturan keputusan mudah dibaca.



- Interpretasi umum:
  - Split awal menunjukkan fitur yang paling berpengaruh dalam mengurangi error prediksi.
- (Opsional) Feature importance:

  Top fitur berdasarkan feature_importances



- Kaitkan fitur penting dengan konteks pendidikan (misalnya attendance, jam belajar, akses sumber belajar).

3.7 Kelebihan Tree-Based Methods pada Studi Kasus Ini

- Mampu menangkap pola non-linear antara faktor siswa dan nilai ujian.
- Mudah menjelaskan aturan keputusan (terutama pada pohon dangkal).
- Cocok untuk dataset dengan gabungan fitur numerik dan kategorikal (dengan encoding).

3.8 Kesimpulan Akhir

- Model Decision Tree dapat memprediksi Exam_Score dengan performa [sebutkan singkat: baik/cukup] berdasarkan MAE/MSE/RMSE.
- Tuning parameter membantu mengontrol overfitting dan meningkatkan generalisasi.
- Faktor paling berpengaruh terhadap prediksi (berdasarkan importance/split awal) adalah [isi fitur-fitur utama].