

# **LAPORAN TUGAS BESAR KECERDASAN BUATAN**

## **Prediksi Tingkat Stres Mahasiswa Selama Pembelajaran Daring Menggunakan Decision Tree**



Disusun oleh:

Rosdiana Putri Purwani – 2306003

Sintia Juniarti – 2306008

Dosen Pengampu Mata Kuliah:

Leni Fitriani, S.Kom, M.Kom

**INSTITUT TEKNOLOGI GARUT  
JURUSAN ILMU KOMPUTER  
PROGRAM STUDI TEKNIK INFORMATIKA  
TAHUN AKADEMIK 2024/2025**

## **PENDAHULUAN**

Perubahan sistem pendidikan dari tatap muka menjadi pembelajaran daring selama pandemi COVID-19 berdampak langsung pada psikologis mahasiswa. Mahasiswa mengalami tekanan akademik yang cukup berat akibat sistem baru ini, seperti peningkatan beban tugas, kurangnya pemahaman materi, dan minimnya interaksi sosial. Penelitian Lubis et al. (n.d.) menunjukkan bahwa sebesar 39,2% mahasiswa mengalami stres akademik kategori sedang, 27% kategori tinggi, dan hanya 5,4% sangat rendah, yang mengindikasikan bahwa kuliah daring memberikan tekanan nyata terhadap kondisi mental mahasiswa.

Senada dengan itu, studi literatur yang dilakukan oleh Fauziyyah, Awinda, dan Besral (2021) mengungkapkan bahwa rata-rata 55,1% mahasiswa di Indonesia mengalami stres dan 40% mengalami kecemasan selama perkuliahan jarak jauh. Mereka menemukan bahwa perubahan metode belajar, gangguan jaringan, tumpukan tugas, dan keterbatasan interaksi sosial menjadi faktor pemicu utama. Penelitian ini juga menegaskan bahwa pembelajaran jarak jauh yang dilakukan tanpa kesiapan menyeluruh dapat menjadi sumber stres yang signifikan bagi mahasiswa.

Harahap dkk.(2020)menyatakan bahwa 75% mahasiswa mengalami stres akademik sedang, dan 13% mengalami stres tinggi selama pembelajaran jarak jauh. Penelitian ini memperkuat temuan bahwa sistem pembelajaran daring belum optimal dalam mengurangi tekanan psikologis mahasiswa.

## 1. BUSINESS UNDERSTANDING

### 1.1 Permasalahan dunia nyata

Perubahan sistem pembelajaran dari tatap muka menjadi daring (online) telah membawa dampak besar dalam dunia pendidikan, terutama di kalangan mahasiswa. Meskipun pembelajaran daring menawarkan fleksibilitas dalam mengakses materi dan waktu belajar, namun di sisi lain sistem ini menyebabkan munculnya tekanan dan stres baru bagi mahasiswa. Beban akademik yang tinggi, keterbatasan interaksi sosial, masalah teknis seperti koneksi internet yang tidak stabil, serta lingkungan rumah yang kurang mendukung menjadi penyebab utama meningkatnya tingkat stres selama proses belajar daring. Stres yang berlebihan dapat menurunkan konsentrasi, motivasi, dan performa akademik mahasiswa, bahkan berisiko menyebabkan gangguan kesehatan mental seperti kecemasan dan depresi. Dalam konteks ini, deteksi dini terhadap tingkat stres mahasiswa menjadi sangat penting agar intervensi yang tepat dapat dilakukan, misalnya melalui konseling, pengurangan beban tugas, atau pendekatan akademik yang lebih humanis.

### 1.2 Tujuan Proyek

Tujuan utama dari proyek ini adalah membangun dan mengimplementasikan sebuah sistem klasifikasi berbasis kecerdasan buatan yang mampu memprediksi tingkat stres mahasiswa (kategori: rendah, sedang, tinggi) selama proses pembelajaran daring. Sistem ini dikembangkan menggunakan algoritma Decision Tree karena sifatnya yang mudah dipahami, efisien, dan mampu menangani kombinasi fitur numerik dan kategorikal dengan baik.

### 1.3 Siapa user/pengguna sistem

Sistem prediksi tingkat stres mahasiswa ini dirancang untuk memberikan manfaat kepada berbagai pihak yang terlibat dalam dunia pendidikan, terutama pada level perguruan tinggi. Pengguna sistem dapat berasal dari kalangan akademik maupun non-akademik yang berkepentingan terhadap kesejahteraan psikologis mahasiswa.

#### 1. Dosen dan Tenaga Pengajar

Dosen dapat menggunakan sistem ini untuk:

- Menyesuaikan jumlah tugas atau metode pembelajaran jika mayoritas mahasiswa berada dalam kondisi stres tinggi.

- Mengidentifikasi mahasiswa yang mungkin mengalami tekanan mental dan membutuhkan pendekatan pembelajaran yang lebih personal.
- Melakukan evaluasi terhadap dampak kebijakan akademik terhadap kesehatan mental mahasiswa.

## 2. Layanan Konseling dan Psikolog Kampus

Layanan konseling merupakan pihak yang paling diuntungkan dari sistem ini karena mereka bisa:

- Menentukan prioritas penanganan berdasarkan tingkat stres yang diprediksi oleh sistem.
- Menjalankan program intervensi atau terapi yang lebih terarah dan berbasis data.
- Menggunakan hasil prediksi sebagai bahan laporan atau evaluasi program kesejahteraan mahasiswa.

## 3. Pihak Manajemen atau Administrasi Akademik

Pihak manajemen seperti dekanat, program studi, atau biro kemahasiswaan dapat memanfaatkan sistem untuk:

- Menyusun kebijakan akademik yang lebih berorientasi pada kesehatan mental mahasiswa.
- Menyediakan fasilitas pendukung (seperti klinik psikologi, waktu rehat akademik, dan pelatihan manajemen stres).
- Menyusun laporan evaluatif tentang kesejahteraan mental mahasiswa selama semester berlangsung.

## 4. Mahasiswa Itu Sendiri

Mahasiswa dapat menggunakan sistem ini secara mandiri untuk:

- Melihat dan memahami tingkat stres mereka berdasarkan aktivitas harian dan kondisi pribadi.
- Menjadi lebih sadar akan pentingnya menjaga keseimbangan hidup selama pembelajaran daring.
- Mengambil langkah preventif lebih awal seperti mengatur waktu belajar, mengelola tugas, atau mencari bantuan profesional.

## 5. Peneliti dan Akademisi

Sistem ini juga berguna bagi peneliti yang sedang meneliti topik terkait psikologi pendidikan, pembelajaran daring, atau penerapan kecerdasan buatan dalam bidang sosial. Mereka dapat:

- Menggunakan sistem sebagai alat bantu eksperimen.
- Menyumbangkan hasil penelitian sebagai masukan kebijakan institusi.

#### 1.4 Manfaat implementasi AI

Implementasi teknologi kecerdasan buatan dalam proyek ini, khususnya melalui penggunaan algoritma Decision Tree, memberikan banyak keuntungan yang signifikan dalam bidang pendidikan tinggi, terutama dalam menghadapi masalah stres pada mahasiswa selama proses belajar secara daring.

Peran IT juga kerap dimanfaatkan dalam mendukung proses belajar, baik di sekolah maupun untuk pembelajaran secara mandiri. Di masa depan aktivitas belajar akan lebih banyak mengimplementasikan kecerdasan buatan. AI dapat dimanfaatkan untuk menyampaikan materi pembelajaran, melakukan penilaian, memberikan umpan balik proses belajar. Kecerdasan Buatan sebagai elemen dari Teknologi Informasi hadir sebagai jawaban dalam pendidikan selama masa Pandemi Covid-19. Synthetic Kecerdasan adalah sebuah penggunaan dan petunjuk yang berkaitan dengan pengkodean komputer untuk melaksanakan sesuatu yang dalam perspektif manusia memiliki kecerdasan atau dapat dimengerti sebagai suatu penelitian mengenai cara merakit komputer dapat melakukan tindakan-tindakan yang saat ini bisa dilakukan dengan lebih baik daripada manusia. Lingkup dan penerapan Kecerdasan Buatan sangatlah luas termasuk pula dalamnya sektor Pendidikan.<sup>1062 (n.d.)</sup>

Dengan adanya sistem prediktif ini, lembaga pendidikan bisa lebih awal mengenali mahasiswa yang berisiko mengalami stres tinggi, sehingga memungkinkan untuk melakukan intervensi dengan lebih cepat dan tepat. Selain itu, penerapan AI juga membuat proses evaluasi kondisi mahasiswa dapat dilakukan secara otomatis dan efisien, tanpa perlu penilaian manual yang menyita waktu. Keputusan yang dihasilkan pun menjadi lebih berbasis data, objektif, dan konsisten, karena didasarkan pada pola yang diperoleh dari data yang sebenarnya. Model Decision Tree yang diterapkan juga mudah untuk dipahami dan divisualisasikan, sehingga hasilnya dapat

diinterpretasikan dengan baik oleh orang-orang non-teknis seperti dosen. Secara keseluruhan, penerapan kecerdasan buatan dalam proyek ini tidak hanya memaksimalkan efektifitas dalam mengidentifikasi stress pada mahasiswa, tetapi juga mendorong pengguna teknologi cerdas untuk menciptakan lingkungan pendidikan yang lebih memperhatikan kesehatan mental.

## 2. **DATA UNDERSTANDING**

### 2.1 Sumber Data

Data set yang kami ambil itu dari Kaggle yang berjudul **“Student Mental Health Analysis During Online Learning”** dan berisi 1.000 entri data yang merepresentasikan informasi mengenai kondisi psikologis dan kebiasaan mahasiswa selama pembelajaran daring. Dataset ini mencakup data kategorikal dan numerik, serta target klasifikasi berupa tingkat stres.

### 2.2 Deskripsi setiap fitur

Berikut adalah penjelasan dari masing-masing atribut dalam dataset:

- Name: Nama responden (string) – tidak digunakan dalam pemodelan karena bersifat identitas.
- Gender: Jenis kelamin (Male/Female) – data kategorikal.
- Age: Usia responden (integer).
- Education Level: Jenjang pendidikan (misal: Class 8, BTech, MSc) – kategorikal.
- Screen Time (hrs/day): Rata-rata durasi waktu di depan layar per hari dalam jam – numerik.
- Sleep Duration (hrs): Rata-rata waktu tidur per hari dalam jam – numerik.
- Physical Activity (hrs/week): Rata-rata waktu olahraga per minggu dalam jam – numerik.
- Stress Level: Target klasifikasi (Low, Medium, High) – kategorikal.
- Anxious Before Exams: Apakah mahasiswa merasa cemas sebelum ujian (Yes/No) – kategorikal.
- Academic Performance Change: Perubahan performa akademik selama daring (Improved, Declined, Same) – kategorikal.

### 2.3 Ukuran dan Format Data

- Jumlah entri: 1.000

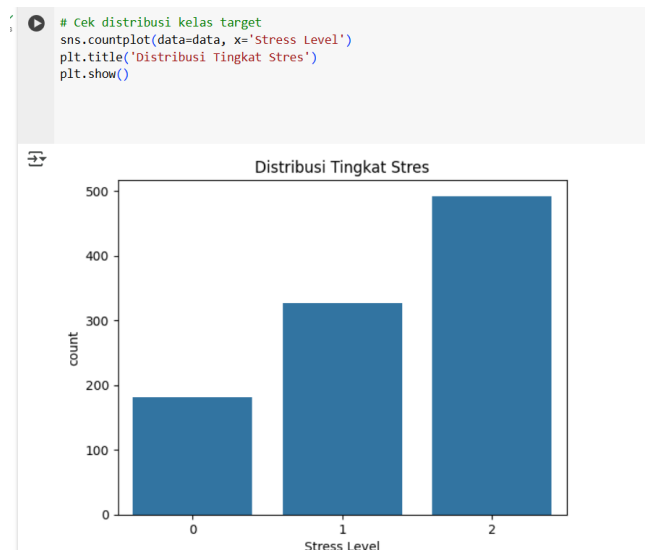
- Jumlah fitur: 10
- Format: .CSV
- Ukuran file: ±78 KB

## 2.4 Tipe data dan target klasifikasi

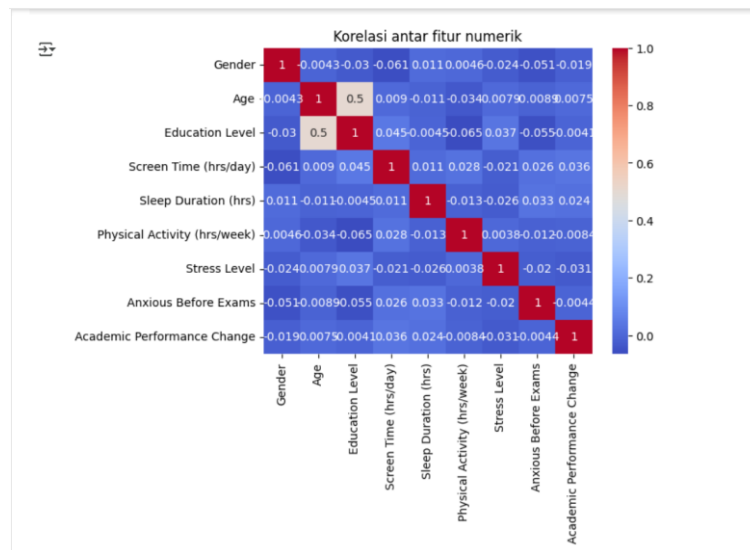
- Numerik: Age, Screen Time (hrs/day), Sleep Duration (hrs), Physical Activity (hrs/week)
- Kategorikal: Gender, Education Level, Anxious Before Exams, Academic Performance Change, Stress Level
- Target (Label): Stress Level

## 3. *EXPLORATORY DATA ANALYSIS (EDA)*

### 3.1 Visualisasi distribusi data.



### 3.2 Analisis Kolerasi antar fitur



## 4. DATA PREPARATION

### a) Pembersihan Data (null Value, duplikasi)

```
data.isna().sum()

Name      0
Gender     0
Age        0
Education Level  0
Screen Time (hrs/day)  0
Sleep Duration (hrs)  0
Physical Activity (hrs/week)  0
Stress Level      0
Anxious Before Exams  0
Academic Performance Change  0

dtype: int64

[9] # Hapus kolom tidak relevan
data = data.drop('Name', axis=1)
```

Pada fase pemeriksaan data, perintah `data.isna().sum()` dipakai untuk menghitung jumlah nilai yang tidak ada di setiap kolom dalam dataset. Output yang diperoleh menunjukkan bahwa semua kolom, yaitu Name, Gender, Age, Education Level, Screen Time (hrs/day), Sleep Duration (hrs), Physical Activity (hrs/week), Stress Level, Anxious Before Exams, dan Academic Performance Change, tidak memiliki nilai yang hilang, yang terlihat dari angka nol pada tabel hasil pemeriksaan. Situasi ini mengindikasikan bahwa dataset telah bersih dan tidak memerlukan langkah lanjutan untuk mengatasi data yang kosong. Setelah memastikan tidak adanya nilai yang hilang, langkah selanjutnya adalah menghapus kolom yang dianggap tidak penting untuk analisis. Dalam hal ini, kolom Name dihapus dengan menggunakan perintah `data`.



drop('Name', axis=1). Kolom Name dihilangkan karena hanya berisi identitas atau nama responden yang unik dan tidak berperan dalam memprediksi tingkat stres. Dengan menghapus kolom ini, dataset menjadi lebih sederhana dan hanya memuat variabel yang langsung terkait dengan proses pemodelan.

b) Proses transformasi: normalisasi/standardisasi data numerik.

```
# Import LabelEncoder dan StandardScaler
from sklearn.preprocessing import LabelEncoder, StandardScaler

# Encode fitur kategorikal
label_enc = LabelEncoder()
data['Gender'] = label_enc.fit_transform(data['Gender'])
data['Education Level'] = label_enc.fit_transform(data['Education Level'])
data['Anxious Before Exams'] = label_enc.fit_transform(data['Anxious Before Exams'])
data['Academic Performance Change'] = label_enc.fit_transform(data['Academic Performance Change'])
data['Stress Level'] = label_enc.fit_transform(data['Stress Level']) # Target

# Normalisasi fitur numerik
scaler = StandardScaler()
num_features = ['Screen Time (hrs/day)', 'Sleep Duration (hrs)', 'Physical Activity (hrs/week)']
data[num_features] = scaler.fit_transform(data[num_features])
print("Data setelah preprocessing:")
display(data.head())
```

Pada langkah pertama, fitur-fitur kategorikal dikehendaki untuk dilakukan pengkodean menggunakan LabelEncoder yang disediakan oleh scikit-learn. Proses ini berfungsi untuk mengonversi data kategorikal yang berbentuk teks menjadi nilai numerik yang mampu diproses oleh algoritma pembelajaran mesin. Fitur yang mengalami pengkodean mencakup 'Gender', 'Education Level', 'Anxious Before Exams', dan 'Academic Performance Change', sementara 'Stress Level' ditetapkan sebagai variabel target yang juga mesti di-encode.

Langkah kedua melibatkan normalisasi data numerik dengan menggunakan StandardScaler. Proses ini diterapkan pada tiga fitur numerik, yaitu 'Screen Time (hrs/day)', 'Sleep Duration (hrs)', dan 'Physical Activity (hrs/week)'. StandardScaler berfungsi untuk menyesuaikan distribusi data agar memiliki rata-rata 0 dan deviasi standar 1, yang membantu algoritma pembelajaran mesin untuk beroperasi lebih efektif, terutama pada algoritma yang sensitif terhadap skala data seperti jaringan saraf atau SVM. Setelah menyelesaikan kedua proses preprocessing, kode akan menunjukkan pesan konfirmasi serta pratinjau dari data yang telah diolah. Gabungan antara pengkodean dan normalisasi ini merupakan praktik umum dalam tahap preprocessing data untuk keperluan pembelajaran mesin. Pengkodean menjamin bahwa semua data kategorikal dapat dipahami oleh model, sedangkan normalisasi memastikan tidak ada fitur numerik yang menguasai proses pelatihan karena perbedaan skala. Penggunaan fit\_transform() pada kedua teknik preprocessing ini menunjukkan bahwa model akan

mempelajari pola transformasi dari data pelatihan dan menerapkannya dengan konsisten, yang krusial untuk menjaga keseragaman antara data pelatihan dan pengujian.

c) Split data :

Hasil pembagian data menunjukkan dataset total 1000 baris dibagi menjadi 800 baris untuk training dan 200 baris untuk testing (rasio 80:20). Distribusi kelas Stress Level tetap proporsional pada kedua set: Data Training: Level 2 (394), Level 1 (261), Level 0 (145). Data Testing: Level 2 (98), Level 1 (66), Level 0 (36) Stratifikasi berhasil mempertahankan proporsi yang sama antara training dan testing set, dengan Level 2 (stress tinggi) sebagai kelas mayoritas, Level 1 (stress sedang) di tengah, dan Level 0 (stress rendah) sebagai kelas minoritas. Claude can make mistakes. Please double-check responses.

## 5. **MODELING**

- a) Algoritma yang di pilih adalah decision tree . Decision tree adalah sebuah struktur data yang terdiri dari simpul (node) dan rusuk (edge). Simpul pada sebuah pohon dibedakan menjadi tiga, yaitu simpul akar (root/node), simpul percabangan/internal (branch/internal node) dan simpul daun (leaf node) . Pohon keputusan merupakan representasi sederhana dari teknik klasifikasi untuk sejumlah kelas berhinga, dimana simpul internal maupun simpul akar ditandai dengan nama atribut, rusuk-rusuknya diberi label nilai atribut yang mungkin dan simpul daun ditandai dengan kelas-kelas yang berbeda. Solehuddin et al. (2022).
- b) Alasan memilih Decision Tree karena mudah dipahami dan Mudah Dipahami dan Diinterpretasikan. Decision Tree menyajikan hasil dalam bentuk pohon keputusan yang intuitif, sehingga mudah dipahami oleh non-teknikal seperti dosen, pihak kampus, atau layanan konseling. Menangani Data Kategorik dan Numerik Sekaligus ,cocok dengan dataset ini karena memiliki kombinasi fitur kategorikal (seperti Gender, Education Level) dan numerik (seperti Screen Time, Sleep Duration). Tidak Membutuhkan Normalisasi Fitur, Meskipun kita sudah menormalisasi fitur, Decision Tree sebenarnya tidak sensitif terhadap skala fitur, sehingga tidak perlu preprocessing kompleks. Menangani Masalah Non-Linear dengan Baik , hubungan antara stres dan variabel seperti screen time atau aktivitas fisik kemungkinan tidak linear, dan Decision Tree bisa mengatasinya tanpa perlu transformasi khusus. Cocok untuk Dataset Kecil

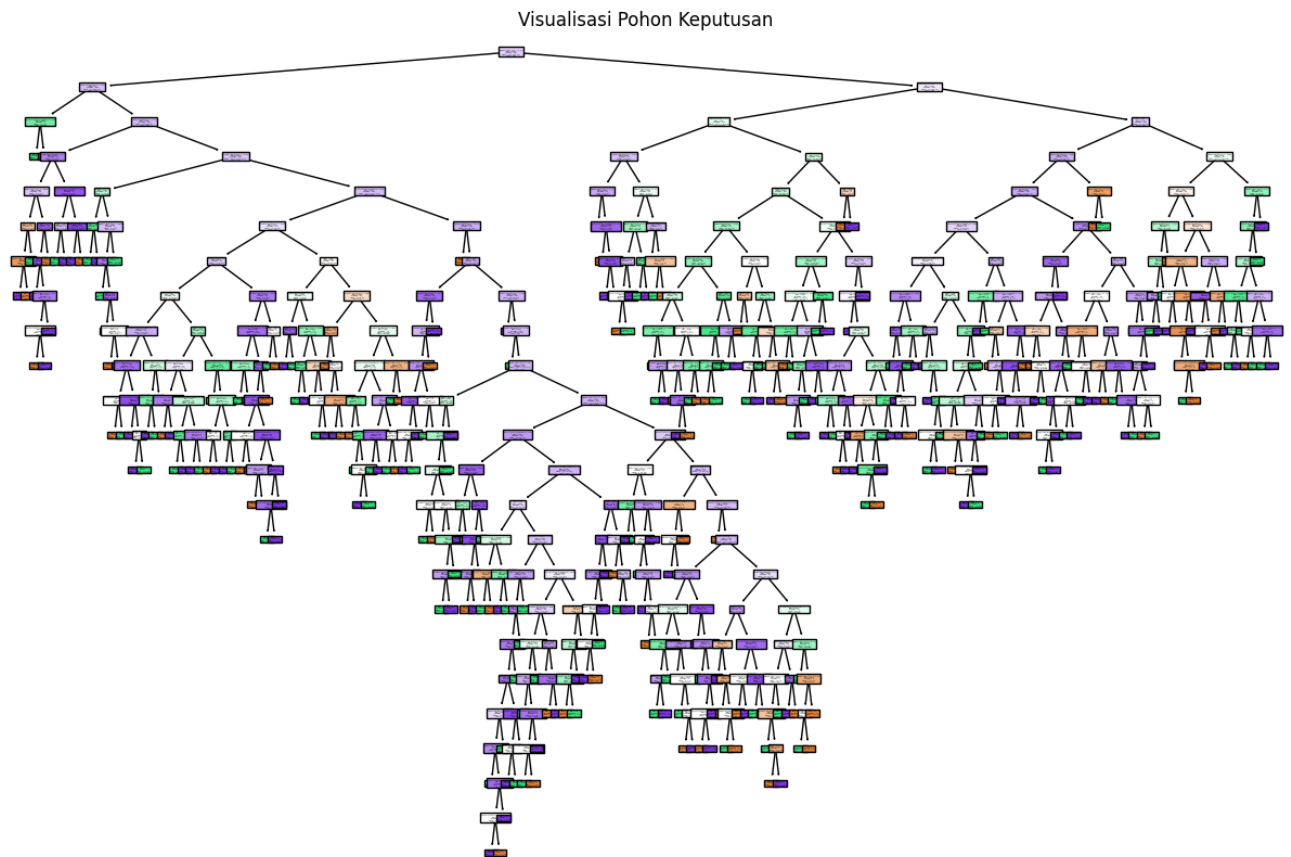
hingga Menengah, dataset ini tidak besar, dan Decision Tree dapat berkinerja baik tanpa memerlukan jumlah data yang sangat banyak seperti beberapa algoritma lain (misalnya Neural Network).

### c) Implementasi Model

```
# Buat dan latih model Decision Tree
model_dt = DecisionTreeClassifier(random_state=42)
model_dt.fit(X_train, y_train)
```

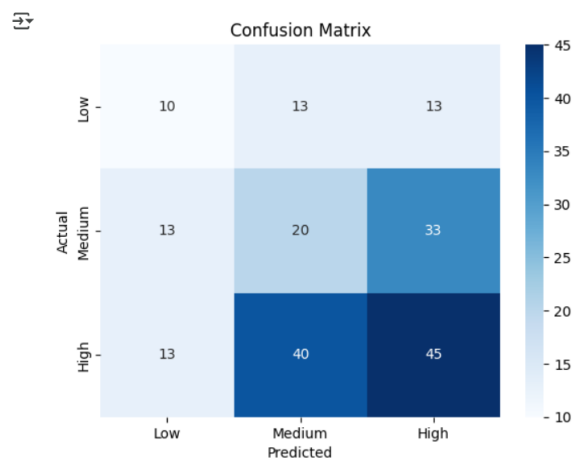
Kode ini digunakan untuk membuat dan melatih model klasifikasi Pohon Keputusan dengan memanfaatkan pustaka scikit-learn di Python. Pertama, sebuah model Decision Tree dihasilkan dengan memanggil kelas `DecisionTreeClassifier` dan menyimpannya dalam variabel `model\_dt`. Parameter `random\_state=42` ditambahkan agar hasil tetap konsisten setiap kali kode dijalankan, karena parameter ini mengatur seed acak yang dipakai dalam pembuatan pohon. Selanjutnya, model dilatih dengan menjalankan metode `fit()` pada objek `model\_dt` menggunakan data pelatihan `x\_train` (fitur) dan `y\_train` (label/target). Proses pelatihan ini akan membentuk struktur pohon keputusan dengan memilih fitur optimal untuk membedakan data secara berulang hingga memenuhi kriteria tertentu, seperti kedalaman maksimum atau jumlah sampel minimum di leaf node. Pohon Keputusan adalah model yang sederhana dan mudah dipahami, sesuai untuk berbagai tugas pengelompokan. Setelah proses pelatihan selesai, model dapat digunakan untuk menghasilkan prediksi pada data yang baru.

#### d) Visualisasi Model



## 6. EVALUATION

### 6.1 Confusion matrix



Dari matriks ini, terlihat model ini memiliki kinerja yang cukup baik dengan prediksi yang benar pada diagonal utama. Model ini mampu memprediksi 20 sampel dengan benar dari kategori "sedang" dan 10 sampel dari kategori "rendah", dengan 45

sampel dari kategori "tinggi". Warna biru gelap diagonal menunjukkan nilai -nilai tinggi ini, menunjukkan bahwa setiap kelas memiliki akurasi yang baik.

Namun, model ini juga menampilkan beberapa kesalahan prediksi dari angka off-diagonal utama. Sebagai contoh, 13 sampel yang diklasifikasikan sebagai "rendah" diprediksi sebagai "sedang", dan 13 diprediksi sebagai "tinggi". Ada juga 13 sampel dalam kategori "sedang", yang diharapkan menjadi "rendah" dan 33 sunpets sebagai "tinggi." Untuk kategori "tinggi", model membuat kesalahan dengan memprediksi 13 sampel sebagai "rendah" dan 40 sampel sebagai "medium."

a) Matrik Evaluasi : Accuary,Precision,Recall, f1-score

Accuracy: 0.375					
Classification Report:					
	precision	recall	f1-score	support	
Low	0.28	0.28	0.28	36	
Medium	0.27	0.30	0.29	66	
High	0.49	0.46	0.48	98	
accuracy			0.38	200	
macro avg	0.35	0.35	0.35	200	
weighted avg	0.38	0.38	0.38	200	

b) Penjelasan kinerja model berdasarkan metrik tersebut.

Model Decision Tree yang diproduksi menghasilkan nilai akurasi 0,375. Ini hanya menunjukkan sekitar 37,5% dari prediksi model sesuai dengan karakter aktual dalam data uji. Nilai akurasi ini relatif rendah, menunjukkan bahwa model kesulitan mengenali pola data untuk membedakan tingkat stres bagi siswa dalam kategori rendah, menengah dan tinggi. Jika model dalam laporan klasifikasi ditampilkan secara lebih rinci, keakuratan model akan bervariasi di seluruh kelas. Akurasi tertinggi dapat ditemukan di kelas high-end 0,49. Ini berarti bahwa semua prediksi yang sesuai dengan beban tinggi sekitar 49% benar. Kelas menengah memiliki ketepatan hanya 0,27 dan rendah 0,28. Model ini sering menunjukkan bahwa kelas data ini memprediksi, meskipun termasuk dalam kategori lain.

Sementara itu, penarikan di semua kelas juga relatif rendah, dengan nilai tertinggi di kelas tinggi 0,46. Ini berarti bahwa model hanya menemukan sekitar 46% dari semua data. Ini sebenarnya cukup menegangkan. Penarikan kembali callback

adalah kelas rendah dan rendah, hanya 0,30 dan 0,28, menunjukkan bahwa model sering tidak mendeteksi sampel kelas ini (false negative). Memang benar untuk mengatakan bahwa itu adalah akurasi yang harmonis rata-rata dan nilai rata-rata penarikan F1 juga rendah. Skor F1 kelas tinggi adalah 0,48, dengan kelas menengah rendah dan menengah 0,28 dan 0,29. Ini menunjukkan bahwa model diklasifikasikan sebagai kinerja yang tidak seimbang dan tidak mencukupi ketika mengenali tiga kategori tegangan. Secara keseluruhan, penilaian ini menunjukkan bahwa model pohon yang menentukan keputusan untuk percobaan ini gagal memberikan prediksi yang akurat dan andal. Nilai metrik rendah antar kelas menunjukkan perlunya perbaikan, seperti menyesuaikan model parameter, atau menambahkan tes kompleks lain dari data pelatihan dan algoritma. Selain itu, penilaian komprehensif oleh kelas mengkonfirmasi bahwa model agak "baik" untuk mengenali stres yang lebih tinggi daripada kategori lain, tetapi kinerjanya umumnya tidak ideal.

## KESIMPULAN

Berdasarkan hasil tugas besar ini, dapat disimpulkan bahwa penerapan algoritma Decision Tree dalam membangun sistem prediksi tingkat stres mahasiswa selama pembelajaran daring memberikan gambaran awal yang cukup informatif meskipun dengan tingkat akurasi yang masih rendah, yakni sebesar 37,5%. Model menunjukkan performa terbaik dalam mengidentifikasi kategori stres tinggi, namun masih kesulitan membedakan antara stres rendah dan sedang. Hal ini menunjukkan bahwa meskipun pendekatan ini potensial, diperlukan peningkatan lebih lanjut agar sistem dapat digunakan secara lebih andal dalam lingkungan nyata.

Untuk pengembangan ke depan, disarankan agar digunakan algoritma yang lebih kompleks seperti Random Forest atau Gradient Boosting yang mampu menangani data dengan lebih baik. Selain itu, perlu dilakukan penanganan ketidakseimbangan data antar kelas serta optimasi parameter model melalui teknik hyperparameter tuning. Penambahan fitur-fitur yang lebih relevan dan kontekstual, seperti faktor lingkungan dan ekonomi, juga dapat meningkatkan akurasi prediksi. Validasi model secara menyeluruh menggunakan metode *cross-validation* serta pengembangan sistem dalam bentuk aplikasi atau platform digital akan sangat membantu mahasiswa maupun institusi dalam mendeteksi dan menangani stres secara lebih dini dan tepat sasaran.

## DAFTAR PUSTAKA

1062. (n.d.).

Dampak Pembelajaran Jarak Jauh terhadap Tingkat Stres dan Kecemasan Mahasiswa selama Pandemi COVID-19. (2021). *Jurnal Biostatistik, Kependudukan, Dan Informatika Kesehatan*, 1(2). <https://doi.org/10.7454/bikfokes.v1i2.1011>

Harahap, A. C. P., Harahap, D. P., & Harahap, S. R. (2020). Analisis Tingkat Stres Akademik Pada Mahasiswa Selama Pembelajaran Jarak Jauh Dimasa Covid-19. *Biblio Couns : Jurnal Kajian Konseling Dan Pendidikan*, 3(1), 10–14. <https://doi.org/10.30596/bibliocouns.v3i1.4804>

Lubis, H., Ramadhani, A., & Rasyid, M. (n.d.). Stres Akademik Mahasiswa dalam Melaksanakan Kuliah Daring Selama Masa Pandemi Covid 19. *Maret*, 10(1), 31–39. <https://doi.org/10.30872/psikostudia>

Solehuddin, M., Syafei, W. A., & Gernowo, R. (2022). Metode Decision Tree untuk Meningkatkan Kualitas Rencana Pelaksanaan Pembelajaran dengan Algoritma C4.5. *Jurnal Penelitian Dan Pengembangan Pendidikan*, 6(3), 510–519. <https://doi.org/10.23887/jpppp.v6i3.52840>