

## **STUDI KOMPARATIF *NAIVE BAYES* DAN *DECISION TREE* PADA *DATASET* BUNGA IRIS: EVALUASI AKURASI DAN EFISIENSI**

**Daffa Ariftha Eryana<sup>1)</sup>, Filius Deivivi<sup>2)</sup>, M. Ilham Baehaqi<sup>3)</sup>, Arya Wicaksono<sup>4)</sup>, Zalfa Alykha Puspita<sup>5)</sup>, Herwis Gultom<sup>6)</sup>**

Prodi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pamulang

Correspondence author: H. Gultom, dosen02535@unpam.ac.id, Tangerang Selatan, Indonesia

### **Abstract**

This study presents a comparative analysis between the *Naive bayes* algorithm and *Decision tree* algorithms in various domains, including iris species *Classification*, meat composition analysis, subject *Classification*, and slope analysis in tourism areas. This study aims to evaluate the performance, advantages, and limitations of both algorithms in various data contexts. The methodology used involves applying both algorithms to various *datasets* and comparing their *accuracy*, *precision*, *recall*, and *F1-score* metrics. The results show that both algorithms exhibit competitive performance, with varying strengths depending on the characteristics of the data. *Decision trees* demonstrate better interpretability and can handle non-linear relationships, while *Naive bayes* demonstrates strong performance with independent features and large *training datasets*. These findings contribute to understanding the appropriate application context for each algorithm in data mining tasks.

**Keywords:** *comparative analysis, naive bayes, decision tree, Classification*

### **Abstrak**

Penelitian ini menyajikan analisis komparatif antara algoritma *Naive bayes* dan *Decision tree* di berbagai domain, termasuk klasifikasi spesies iris, analisis komposisi daging, klasifikasi subjek, dan analisis kemiringan di area pariwisata. Penelitian ini bertujuan untuk mengevaluasi kinerja, keunggulan, dan keterbatasan kedua algoritma dalam berbagai konteks data. Metodologi yang digunakan melibatkan penerapan kedua algoritma pada berbagai *dataset* dan membandingkan akurasi, presisi, *recall*, serta metrik *F1-score*. Hasil menunjukkan bahwa kedua algoritma memiliki kinerja yang kompetitif dengan kekuatan yang bervariasi tergantung pada karakteristik data. *Decision tree* menunjukkan interpretabilitas yang lebih baik dan mampu menangani hubungan non-linear, sedangkan *Naive bayes* menunjukkan kinerja yang kuat dengan fitur-fitur independen dan *dataset* pelatihan yang besar. Temuan ini berkontribusi dalam memahami konteks aplikasi yang tepat untuk setiap algoritma dalam tugas data mining.

**Kata Kunci:** *perbandingan, naive bayes, decision tree, klasifikasi, data mining*

## A. PENDAHULUAN

Di era digital saat ini, volume data yang dihasilkan terus meningkat secara eksponensial. Hal ini mendorong perlunya teknik analisis data yang efektif untuk mengekstrak informasi berharga dari data tersebut. Penambangan data, sebagai salah satu cabang ilmu komputer yang berfokus pada penemuan pola dalam kumpulan data besar, menjadi sangat relevan dalam konteks ini (Sarno et al., 2022).

Klasifikasi merupakan salah satu tugas mendasar dalam penambangan data yang bertujuan untuk memprediksi kategori atau kelas dari data yang diberikan. Dua algoritma yang sering digunakan dalam tugas klasifikasi adalah *Naive bayes* dan *Decision tree*. Kedua algoritma tersebut memiliki karakteristik yang berbeda dalam hal pendekatan, kompleksitas, dan kinerja (Pradnyana et al., 2020).

*Naive bayes* merupakan algoritma klasifikasi probabilistik yang berbasis pada teorema Bayes dengan asumsi independensi yang kuat antar fitur. Algoritma ini dikenal karena kesederhanaannya, kecepatan pelatihan yang tinggi, dan kinerja yang baik pada berbagai jenis data (Iriadi et al., 2020). Di sisi lain, *Decision tree* merupakan algoritma berbasis aturan yang membangun model dalam bentuk struktur pohon untuk membuat keputusan klasifikasi (Primartha, 2018).

Pemilihan algoritma yang tepat sangat penting dalam mencapai kinerja klasifikasi yang optimal. Karakteristik data dan domain aplikasi berperan penting dalam menentukan algoritma yang cocok. Evaluasi kelengkapan masing-masing algoritma dalam berbagai konteks menjadi penting bagi praktisi dan peneliti di bidang penambangan data (Akbar & Yustanti, 2024).

Penelitian ini memberikan kontribusi terhadap pemahaman kapan dan mengapa menggunakan setiap algoritma dengan

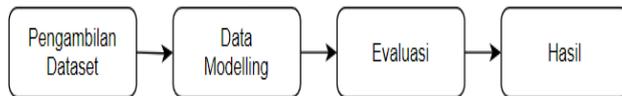
menyediakan analisis komparatif yang komprehensif di berbagai domain. Studi ini menggunakan lima *dataset* algoritma pada empat domain yang berbeda untuk memberikan wawasan tentang kinerja relatif mereka dalam berbagai kondisi data.

*Dataset* yang digunakan dalam penelitian ini meliputi: (1) *Dataset* Bunga Iris sebagai *dataset* utama yang akan dibandingkan dengan empat *dataset* lainnya; (2) *Dataset* Analisis Klasifikasi Komposisi Daging; (3) *Dataset* Analisis Mata Pelajaran Menggunakan *Machine learning*; (4) *Dataset* Analisis Kemiringan Lahan; dan (5) *Dataset* Kawasan Wisata di Desa Trunyan dan Toya Bungkah. Keberagaman domain pada *dataset* ini memungkinkan evaluasi yang komprehensif terhadap kemampuan adaptasi kedua algoritma pada berbagai karakteristik data.

Tujuan utama penelitian ini adalah untuk melakukan evaluasi komparatif yang sistematis antara algoritma *Naive bayes* dan *Decision tree* dalam hal akurasi dan efisiensi komputasi. Hasil penelitian ini diharapkan dapat memberikan panduan praktis bagi peneliti dan praktisi dalam memilih algoritma klasifikasi yang tepat sesuai dengan karakteristik data dan kebutuhan aplikasi mereka.

## B. METODE PENELITIAN

Dalam penelitian ini, model dibangun menggunakan bahasa pemrograman Python untuk menyusun model klasifikasi. Beberapa *library* untuk menjalankan pemodelan data dijelaskan dengan Pandas oleh bahasa pemrograman Python serta dijalankan menggunakan *Visual Studio Code* sebagai *Integrated Development Environment* (IDE). Secara garis besar, penelitian dilakukan dengan rangkaian kegiatan mulai dari data modeling, evaluasi model, dan menambah *Feature importance*.



Gambar 1. Tahapan penelitian

### Pengambilan Dataset

Dataset yang digunakan dalam penelitian ini diperoleh dari Kaggle dan tersedia pada situs website <https://www.kaggle.com/datasets/uciml/iris?resource=download>. Dataset tersebut digunakan dalam makalah klasik R.A. Fisher tahun 1936 yang berjudul "The Use of Multiple Measurements in Taxonomic Problems", dan juga dapat ditemukan di UCI Machine learning Repository.

Dataset ini mencakup tiga spesies bunga iris dengan masing-masing 50 sampel, serta beberapa properti (ciri-ciri) dari setiap bunga. Salah satu spesies bunga dapat dipisahkan secara linear dari dua spesies lainnya.

Tabel 1. Evaluasi Variabel Data

No	Variabel	Keterangan	Jenis Data
1	SepalLength	Panjang Sepal (Kelopak Bunga) dalam satuan sentimeter (cm).	Numerik (float).
2	PetalLength	Panjang Petal (Mahkota Bunga) dalam satuan sentimeter (cm).	Numerik (float).
3	SepalWidth	Lebar Sepal (Kelopak Bunga) dalam satuan sentimeter (cm).	Numerik (float).
4	PetalWidth	Lebar Petal (Mahkota Bunga) dalam satuan sentimeter (cm).	Numerik (float).
5	Species	Jenis atau Spesies bunga iris (Iris-setosa, Iris-versicolor, Iris-virginica).	Kategorikal (string/tabel).

### Data Modelling

Terdapat dua model prediksi yang dibangun dengan teknik Machine learning

dalam pembangunan model klasifikasi, yaitu *Decision tree* dan *Naive bayes*. Penerapan beberapa model *Machine learning* bertujuan untuk mendeteksi penyakit jantung dengan sempurna (Depari et al., 2022). Kedua model yang digunakan termasuk dalam kategori model *Machine learning* berbasis pohon keputusan. Penjelasan detail terdapat pada sebagai berikut.

1. *Decision tree*: bisa disebut dengan algoritma pohon keputusan merupakan algoritma dalam *Machine learning* yang menggunakan struktur pohon, yaitu membagi *dataset* menjadi beberapa bagian atau percabangan kecil untuk menentukan keputusan (Kirono & Nataliani, 2024). Model ini direpresentasikan sebagai struktur diagram yang menyerupai pohon, setiap cabang merupakan hasil pengujian, dan titik akhir atau daun merupakan kelas yang menandakan kategori.
2. *Naive bayes*: adalah kombinasi dari beberapa pohon keputusan untuk memaksimalkan algoritma dan akurasi yang didapat, dimana setiap pohon bertanggung jawab atas setiap nilai dari *random vector* serta perseberangan yang diberikan kepada pohon keputusan lainnya (Sarno et al., 2022).

Membangun model klasifikasi menggunakan dua algoritma tersebut dilakukan dengan melatih model *Decision tree* menggunakan parameter tertentu, kali ini menggunakan "max\_depth" untuk mengontrol kedalaman pohon keputusan. Sedangkan untuk *Naive bayes*, melatih model Gaussian *Naive bayes* untuk melakukan pelatihan klasifikasi model pada data latih. Jumlah data latih yang ingin diuji bisa diatur sesuai kebutuhan penelitian (Pradnyana et al., 2020).

Selanjutnya, *Feature importance* akan ditambahkan pada tiap model karena ketiga model sama-sama memiliki fitur tersebut. Fitur ini digunakan untuk mengetahui

faktor risiko dari penyakit gagal jantung menurut variabel *training* pada *dataset*. Proses *Feature importance* adalah konsep yang mengevaluasi kontribusi setiap fitur individu terhadap prediksi pada model (Iriadi et al., 2020). Skor *Feature importance* yang diberikan dapat menentukan pentingnya variabel dalam diagnosis dan prediksi penyakit jantung. Tiap nilai kontribusi akan dihitung nilai rata-rata lalu diurutkan berdasarkan besar ke kecilnya nilai tersebut sehingga dapat diketahui diagnosis target.

### Evaluasi

Proses evaluasi adalah proses untuk mengukur tingkat kinerja dari model yang digunakan (Primartha, 2018). Evaluasi model pada penelitian mencakup *precision*, *recall*, *F1-score*, dan *accuracy*. *Precision* bertujuan mengukur kelas positif yang diprediksi benar positif. *Recall* bertujuan menentukan semua instance positif sebenarnya. *F1-score* adalah nilai rata-rata dari presisi dan *recall*. *Accuracy* digunakan untuk mengukur tingkat ketepatan model dalam memprediksi kelas dengan benar.

Menguji performa masing-masing model pada data uji dilakukan untuk memastikan kemampuan generalisasi algoritma pada data yang belum pernah dilihat sebelumnya. Menghitung matrik evaluasi meliputi akurasi, *Confusion Matrix*, dan *Classification Report* yang memberikan gambaran komprehensif tentang kinerja model. Membandingkan hasil evaluasi antara *Decision tree* dan *Naive bayes* dilakukan secara sistematis untuk mengidentifikasi kelebihan dan kekurangan masing-masing algoritma pada berbagai jenis *dataset*. Visualisasi hasil berupa *Decision tree Plot*, *Confusion Matrix*, dan Distribusi Fitur digunakan untuk memberikan interpretasi yang lebih mudah dipahami mengenai karakteristik dan performa kedua algoritma dalam menangani variasi data yang berbeda.

### C. HASIL DAN PEMBAHASAN

Sebelum menerapkan model *Machine learning*, terlebih dahulu dilakukan *split* data atau data pengujian untuk pelatihan dengan data test atau data pengujian. Setiap variabel yang diperoleh menjadi *data train* dan *data test* dengan rasio perbandingan untuk setiap model adalah 80% *data training* dan 20% *data testing*. Pembagian ini dilakukan menggunakan fungsi *train\_test\_split* guna menjamin data *training* dan *testing* untuk setiap algoritma baru. Untuk menjaga proporsi nilai pasien normal dan gagal jantung tetap seimbang pada data *training* dan *testing* maka digunakan parameter *stratify*. Ketiga model tersebut akan dievaluasi dengan data *training* serta diprediksi dengan data *testing* agar nilai evaluasi tiap model tetap konsisten dengan menerapkan parameter *random\_state*. Hasil evaluasi performa setiap model terdapat pada Tabel 2.

**Tabel 2.** Evaluasi Setiap Model

Hasil	Model	
	<i>Decision tree</i>	Naive Bayes
<i>Accuracy</i>	0.366	0.3
<i>Precision</i>	0.34	0.39
<i>Recall</i>	0.36	0.33
<i>F1-score</i>	0.33	0.29

Berdasarkan Tabel 2 hasil performa model *Decision tree* untuk prediksi klasifikasi diperoleh nilai *accuracy* yang lebih tinggi dibandingkan *Naive bayes*, yaitu sebesar 0.366 atau 36.6%. Model *Decision tree* memiliki hasil *accuracy*, *recall*, dan *F1-score* yang lebih baik dibandingkan *Naive bayes*, namun untuk *Precision*, *Naive bayes* menunjukkan performa yang sedikit lebih unggul dengan nilai 0.39 atau 39%.

Berdasarkan hasil evaluasi setiap model pada Tabel 2, selanjutnya, dibahas penggunaan *Feature importance* untuk tiap

model. Model *Decision tree* menunjukkan kinerja yang lebih stabil secara keseluruhan dengan *accuracy* tertinggi, meskipun masih dalam kategori rendah. Hal ini menunjukkan bahwa kedua algoritma memiliki tantangan dalam mengklasifikasikan *dataset* yang digunakan, kemungkinan disebabkan oleh kompleksitas data atau keterbatasan fitur yang tersedia untuk membedakan kelas target secara efektif.

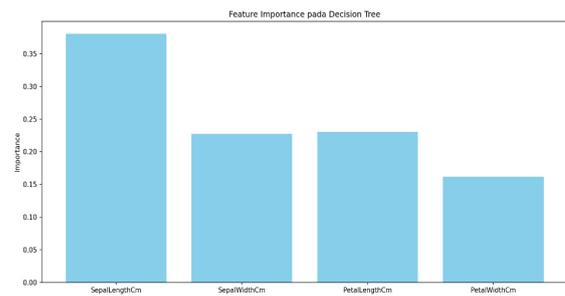
### Decision tree

Dengan menerapkan *Feature importance* pada model *Decision tree* didapatkan variabel *SI\_Slope* memiliki tingkat kepentingan tertinggi dengan nilai 39%, dilanjutkan dengan variabel *MaxHR* senilai 13%, dan *Cholesterol* senilai 12%. Variabel lain memiliki tingkat kepentingan di bawah 10%. Penerapan kode *Feature importance* untuk model *Decision tree* tercantum di Gambar 2. Hasil selengkapnya sebagai berikut pada Gambar 2 dan Tabel 3.

Berdasarkan Gambar 3, dapat dilihat bahwa fitur *SepalLengthCm* memiliki tingkat kepentingan tertinggi dalam model *Decision tree* dengan nilai *importance* sekitar 0.38, diikuti oleh *SepalWidthCm* dan *PetalLengthCm* dengan nilai yang hampir sama sekitar 0.23, sedangkan *PetalWidthCm* memiliki kontribusi terendah dengan nilai sekitar 0.16. Distribusi *Feature importance* ini menunjukkan bahwa karakteristik sepal, terutama panjang sepal, menjadi faktor paling determinan dalam proses klasifikasi menggunakan algoritma *Decision tree* pada *dataset* Iris menggunakan algoritma *Decision tree* pada *dataset* Iris.

```
# Print feature importance
feature_importance = pd.DataFrame({
    'Feature': X.columns,
    'Importance': dt_classifier.feature_importances_
})
print("\nFeature Importance:")
print(feature_importance.sort_values('Importance', ascending=False))
```

Gambar 2. Penerapan *feature importance* pada model *decision tree*



Gambar 3. *Feature importance* dengan *decision tree*

Tabel 3. Mean *Feature importance* Dengan *Decision tree*

No	Variabel	<i>Feature importance</i>
1	SepalLength	0.380587
2	PetalLength	0.230272
3	SepalWidth	0.227518
4	PetalWidth	0.161623

### Naive bayes

Dengan menerapkan analisis mean dan variansi fitur per kelas pada model *Naive bayes* didapatkan distribusi statistik yang menunjukkan karakteristik setiap kelas dalam *dataset* Iris. Berdasarkan Tabel 4, dapat dilihat bahwa setiap kelas memiliki pola distribusi yang berbeda untuk masing-masing fitur. Kelas Iris-versicolor memiliki nilai rata-rata *SepalLengthCm* tertinggi sebesar 5.864286, sedangkan Iris-virginica memiliki nilai *PetalLengthCm* tertinggi sebesar 3.570732. Penerapan kode analisis mean dan variansi untuk model *Naive bayes* tercantum di Gambar 4.

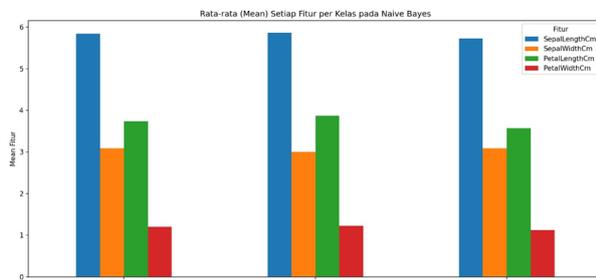
Berdasarkan Gambar 5, dapat dilihat bahwa distribusi rata-rata fitur per kelas menunjukkan pola yang konsisten di antara ketiga spesies Iris. *SepalLengthCm* memiliki nilai tertinggi pada semua kelas dengan rentang 5.7-5.9, diikuti oleh *PetalLengthCm* dengan nilai sekitar 3.6-3.9. *SepalWidthCm* menunjukkan nilai yang relatif stabil di sekitar 3.0-3.1 untuk semua kelas, sedangkan *PetalWidthCm* memiliki nilai terendah sekitar 1.1-1.2. Pola

distribusi ini memberikan insight tentang bagaimana algoritma *Naive bayes* menggunakan probabilitas kondisional dari setiap fitur untuk melakukan klasifikasi berdasarkan karakteristik statistik masing-masing kelas.

```
# Analisis mean dan variansi fitur per kelas
print("\nMean setiap fitur per kelas:")
print(pd.DataFrame(nb_classifier.theta_, columns=X.columns, index=nb_classifier.classes_))

print("\nVariansi setiap fitur per kelas:")
print(pd.DataFrame(nb_classifier.var_, columns=X.columns, index=nb_classifier.classes_))
```

**Gambar 4.** Penerapan perbandingan nilai rata-rata (mean) dan variansi (variance) setiap fitur pada masing-masing kelas



**Gambar 5.** Rata-rata (mean) Setiap Fitur per Kelas pada *Naive bayes*

**Tabel 4.** Mean Fitur per Kelas pada *Naive bayes*

Class	SepalLength Cm	SepalWidth Cm	PetalLengt hCm	PetalWidht hCm
Iris-setosa	5.837838	3.086486	3.740541	1.197297
Iris-versicolor	5.864286	3.002381	3.869048	1.22619
Iris-virginica	5.726829	3.087805	3.570732	1.12439

### Studi Lanjutan Pada *Dataset* Berbeda

Untuk menguji konsistensi kinerja algoritma *Naive bayes* dan *Decision tree* pada berbagai domain aplikasi, studi ini diperluas dengan menggunakan empat *dataset* tambahan yang mencerminkan variasi struktur dan karakteristik data.

Untuk mendukung generalisasi hasil penelitian kami, digunakan empat *dataset* tambahan dari berbagai domain. Masing-masing *dataset* memiliki karakteristik yang berbeda sehingga memungkinkan evaluasi algoritma *Naive bayes* dan *Decision tree* secara lebih luas. Seluruh *dataset* diproses dan dianalisis dengan metode yang sama seperti pada *dataset* Iris.

**Tabel 5.** Deskripsi *Dataset* Tambahan

No	Nama <i>Dataset</i>	Domain	Tujuan
1	Komposisi Daging	Pangan	Mendeteksi kualitas jenis daging
2	Mata Pelajaran	Pendidikan	Mengklasifikasi tingkat minat siswa
3	Kemiringan Lahan	Geospasial	Menentukan kelas risiko kemiringan
4	Rekomendasi Lensa	Kesehatan mata	Menentukan jenis lensa kontak

1. Komposisi Daging (Domain: Pangan): *Dataset* ini memuat fitur-fitur kandungan kimia pada berbagai jenis daging (seperti protein, lemak, air). Tujuannya adalah untuk mengelompokkan jenis atau kualitas daging dengan memanfaatkan pola distribusi nilai-nilai kimiawi tersebut. Sangat cocok untuk menguji algoritma pada data dengan fitur numerik dan korelasi antar atribut.
2. Mata Pelajaran (Domain: Pendidikan): *Dataset* ini menyajikan data tentang minat siswa terhadap berbagai mata pelajaran, dengan mempertimbangkan faktor seperti nilai, kehadiran, dan latar belakang. Tujuannya adalah untuk mengklasifikasikan siswa ke dalam kategori minat tinggi, sedang, atau rendah, dan dapat digunakan untuk mendukung personalisasi pembelajaran.
3. Kemiringan Lahan (Domain: Geospasial): *Dataset* ini berisi

persentase dan derajat kemiringan lahan yang digunakan untuk mengkategorikan area berdasarkan tingkat risiko atau kesesuaian penggunaan lahan (misalnya: aman, waspada, berisiko tinggi). *Dataset* ini sangat penting dalam studi lingkungan dan pengelolaan wilayah.

4. Rekomendasi Lensa (Domain: Kesehatan Mata): *Dataset* ini berisi informasi tentang umur, status

penglihatan, astigmatisme, dan tingkat produksi air mata, yang digunakan untuk merekomendasikan jenis lensa kontak yang sesuai (misalnya: hard, soft, atau tidak disarankan memakai lensa). *Dataset* ini cocok untuk pengujian algoritma klasifikasi berbasis aturan (rule-based).

### Hasil Evaluasi

**Tabel 6.** Evaluasi dari keseluruhan *Dataset*

<i>Dataset</i>	<i>Algoritma</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
Iris	<i>Decision tree</i>	37.78	38.19	37.96	37.6
Iris	<i>Naive bayes</i>	28.89	34.52	30.74	28.67
Kemiringan	<i>Decision tree</i>	100	100	100	100
Kemiringan	<i>Naive bayes</i>	100	100	100	100
Komposisi Daging	<i>Decision tree</i>	78	76	79	77
Komposisi Daging	<i>Naive bayes</i>	72	70	71	70
Lensa	<i>Decision tree</i>	83.33	83.33	88.89	86.05
Lensa	<i>Naive bayes</i>	33.33	33.33	22.22	26.67
Mata Pelajaran	<i>Decision tree</i>	65	63	64	63
Mata Pelajaran	<i>Naive bayes</i>	72	71	70	70

- a. *Accuracy*: Persentase prediksi yang benar dari seluruh data.
- b. *Precision*: Seberapa banyak prediksi positif yang benar.
- c. *Recall*: Seberapa banyak data positif yang berhasil terdeteksi.
- d. *F1-score*: Rata-rata harmonis antara *precision* dan *recall*.

1. Iris (Domain: Botanik). *Dataset* klasik yang berisi pengukuran morfologi bunga iris seperti panjang dan lebar sepal serta petal untuk mengklasifikasikan spesies iris. *Decision tree* mencapai *Accuracy* 37.78% dan *F1-score* 37.6% yang menunjukkan performa cukup rendah, sementara *Naive bayes* lebih rendah lagi

dengan *F1-score* hanya 28.67% yang kurang akurat dalam mengklasifikasikan varietas bunga berdasarkan fitur morfologi yang kompleks.

2. Kemiringan Lahan (Domain: Geospasial). *Dataset* ini berisi persentase dan derajat kemiringan lahan yang digunakan untuk

mengklasifikasikan area berdasarkan tingkat risiko dan kesesuaian lahan misalnya aman, waspada, berisiko tinggi. *Dataset* ini sangat penting dalam studi lingkungan dan perencanaan lahan. Kedua algoritma *Decision tree* dan *Naive bayes* mencapai performa sempurna dengan 100% untuk semua metrik, yang artinya *dataset* ini sangat mudah diklasifikasikan atau datanya sangat bersih dan terpisah jelas sehingga cocok untuk pengujian algoritma klasifikasi berbasis aturan rule-based.

3. Komposisi Daging (Domain: Pangan). *Dataset* ini memuat fitur-fitur kandungan kimia pada berbagai jenis daging seperti protein, lemak, dan air. Tujuannya adalah untuk mengelompokkan jenis atau kualitas daging dengan memanfaatkan pola distribusi nilai-nilai kimiawi tersebut. Nilai *Accuracy* 78% dan *F1-score* 77% menunjukkan bahwa *Decision tree* cukup andal dalam mengklasifikasikan data komposisi daging, sedangkan *Naive bayes* juga cukup baik dengan *F1-score* 70% tetapi tidak seakurat *Decision tree* dalam menguji algoritma pada data dengan fitur numerik dan korelasi antar atribut.
4. Rekomendasi Lensa (Domain: Kesehatan Mata). *Dataset* ini berisi informasi tentang umur, status penglihatan, astigmatisme, dan tingkat produksi air mata, yang digunakan untuk merekomendasikan jenis lensa kontak yang sesuai misalnya hard, soft, atau tidak disarankan memiliki lensa. *Dataset* ini cocok untuk pengujian algoritma klasifikasi berbasis aturan rule-based. *Decision tree* menunjukkan performa yang sangat bagus dengan *Accuracy* 83.33% dan *F1-score* 86.05%, sedangkan *Naive bayes* jauh lebih rendah dengan *F1-score* 26.67%

yang artinya *Decision tree* lebih cocok untuk *dataset* ini.

5. Mata Pelajaran (Domain: Pendidikan). *Dataset* ini menyajikan data tentang minat siswa terhadap berbagai mata pelajaran, dengan mempertimbangkan faktor seperti nilai, kelakuan, dan latar belakang. Tujuannya adalah untuk mengklasifikasikan siswa ke dalam kategori minat tinggi, sedang, atau rendah dan dapat digunakan untuk mendukung personalisasi pembelajaran. *Accuracy* 72% dan *F1-score* 70% menunjukkan bahwa *Naive bayes* lebih tepat dalam menangkap pola pada data mata pelajaran, sementara *Decision tree* hanya mencatat *F1-score* 63% yang artinya model cenderung kurang stabil atau kompleksitas datanya tidak sesuai untuk struktur pohon.

#### Analisis Hasil Tiap *Dataset*

Berdasarkan hasil evaluasi model pada lima *dataset* yang berbeda, diperoleh beberapa temuan penting:

1. *Decision tree* menunjukkan kinerja sangat baik pada *dataset* Kemiringan dengan akurasi, presisi, *recall*, dan *F1-score* mencapai 100%, mengindikasikan bahwa algoritma ini sangat cocok untuk data dengan struktur fitur yang kompleks dan relasi antar atribut.
2. Komposisi Daging (Pangan): *Decision tree* (*F1-score* 77%) unggul atas *Naive bayes* (70%), cocok untuk fitur numerik dengan korelasi antar atribut.
3. Mata Pelajaran (Pendidikan): *Naive bayes* (*F1-score* 70%) lebih unggul dibanding *Decision tree* (63%), menunjukkan stabilitas lebih baik pada data yang sederhana.
4. Untuk *dataset* Lensa, *Decision tree* unggul signifikan dengan akurasi 83.33%, sedangkan *Naive bayes* jauh lebih rendah (33.33%), menandakan bahwa fitur-fitur pada *dataset* ini memiliki keterkaitan yang lebih cocok

dimodelkan oleh struktur pohon keputusan.

5. Iris sebagai *dataset* benchmark menunjukkan hasil sedang untuk *Decision tree* (37.78%) dan lebih rendah untuk *Naive bayes* (28.89%), namun pola ini tetap mencerminkan kekuatan *Decision tree* dalam menangkap hubungan antar fitur.

### Kesimpulan Lokal dari Studi Tambahan

Berdasarkan hasil evaluasi pada lima *dataset*, terlihat bahwa kedua algoritma memiliki keunggulan masing-masing yang bergantung pada struktur dan kompleksitas data:

1. *Decision tree* unggul dalam *dataset* dengan hubungan fitur yang kompleks (Kemiringan, Komposisi Daging, dan Lensa) dengan rata-rata akurasi tinggi dan *F1-score* mencapai 86%.
2. *Naive bayes* cenderung memberikan hasil lebih stabil pada *dataset* dengan fitur independen seperti Mata Pelajaran, meskipun tetap kalah dari *Decision tree* dalam konteks interpretasi dan fleksibilitas.

Secara lokal, temuan ini mendukung kesimpulan umum bahwa:

1. *Decision tree* cocok digunakan untuk *dataset* eksploratif dan analisis mendalam,
2. *Naive bayes* lebih cocok sebagai baseline cepat dan untuk data berskala besar dengan fitur yang saling bebas

### D. PENUTUP

Berdasarkan analisis komprehensif yang telah dilakukan dalam penelitian ini, dapat disimpulkan bahwa algoritma *Naive bayes* dan *Decision tree* memiliki karakteristik kinerja yang berbeda tergantung pada jenis dan struktur *dataset* yang digunakan. Evaluasi pada lima *dataset* dari berbagai domain menunjukkan bahwa

tidak ada algoritma yang secara konsisten unggul di semua kondisi, melainkan keunggulan masing-masing algoritma sangat bergantung pada karakteristik spesifik data yang dianalisis.

*Decision tree* menunjukkan performa superior pada *dataset* dengan struktur fitur yang kompleks dan memiliki hubungan antar atribut yang kuat. Hal ini terlihat jelas pada *dataset* Kemiringan Lahan yang mencapai akurasi sempurna 100%, *dataset* Komposisi Daging dengan *F1-score* 77%, dan *dataset* Rekomendasi Lensa dengan akurasi 83.33%. Keunggulan *Decision tree* terletak pada kemampuannya untuk menangkap hubungan non-linear dan memberikan interpretabilitas yang tinggi melalui struktur pohon keputusan yang mudah dipahami, sehingga sangat cocok untuk analisis eksploratif dan aplikasi yang membutuhkan transparansi dalam proses pengambilan keputusan.

Di sisi lain, *Naive bayes* menunjukkan stabilitas yang lebih baik pada *dataset* dengan fitur-fitur yang relatif independen, seperti yang terlihat pada *dataset* Mata Pelajaran dengan *F1-score* 70% yang unggul dibandingkan *Decision tree* (63%). Meskipun secara umum *Naive bayes* menunjukkan performa yang lebih rendah dalam penelitian ini, algoritma ini tetap memiliki nilai praktis sebagai baseline yang cepat dan efisien, terutama untuk *dataset* berskala besar dengan asumsi independensi fitur yang terpenuhi.

Temuan penelitian ini memberikan panduan praktis bagi peneliti dan praktisi dalam memilih algoritma klasifikasi yang tepat. *Decision tree* lebih direkomendasikan untuk *dataset* dengan struktur kompleks, hubungan fitur yang kuat, dan ketika interpretabilitas model menjadi prioritas utama. Sementara itu, *Naive bayes* lebih cocok digunakan sebagai pendekatan awal yang cepat, untuk *dataset* dengan fitur independen, atau ketika efisiensi komputasi menjadi pertimbangan penting. Penelitian

ini juga menggarisbawahi pentingnya melakukan evaluasi empiris pada data spesifik sebelum menentukan pilihan algoritma, karena karakteristik data memiliki pengaruh yang signifikan terhadap kinerja model klasifikasi.

Secara keseluruhan, penelitian ini berkontribusi dalam memberikan pemahaman yang lebih mendalam tentang konteks aplikasi yang tepat untuk setiap algoritma dalam tugas data mining. Hasil evaluasi menunjukkan bahwa pemilihan algoritma yang optimal tidak dapat ditentukan secara universal, melainkan harus disesuaikan dengan karakteristik data dan kebutuhan spesifik dari setiap aplikasi. Penelitian lanjutan dapat diarahkan untuk mengeksplorasi kombinasi algoritma atau teknik ensemble yang dapat memanfaatkan kelebihan masing-masing algoritma untuk meningkatkan performa klasifikasi secara keseluruhan.

## E. DAFTAR PUSTAKA

- Akbar, M., & Yustanti, W. (2024). Pemilihan Algoritma Klasifikasi Terbaik Untuk Prediksi Jenis Keluhan MI User Interface (MIUI) 14. *JINACS : Journal of Informatics and Computer Science*, 6(2), 445–452. <https://doi.org/10.26740/jinacs.v6n02.p445-452>
- Chotib, A. S. (2020). Sistem Informasi Penilaian Angka Kredit Dosen Berbasis Web (Studi Kasus PAK Online Kopertais Wilayah I DKI Jakarta). *Applied Information System and Management (AISM)*, 3(1), 21–30. <https://doi.org/10.15408/aism.v3i1.16189>
- Depari, D. H., Widiastiwi, Y., & Santoni, M. M. (2022). Perbandingan Model *Decision tree*, *Naive bayes* dan *Random Forest* untuk Prediksi Klasifikasi Penyakit Jantung. *Informatik : Jurnal Ilmu Komputer*, 18(3), 239–248. <https://doi.org/10.52958/iftk.v18i3.4694>
- Iriadi, N., Priatno, & Ishaq, A. (2020). *Penerapan Data Mining dengan Rapid Miner : Konsep Data Maining, Data Warehouse, Metode, Model, Teknik*. Yogyakarta : Graha Ilmu.
- Kirono, A. S., & Nataliani, Y. (2024). Perbandingan Algoritma *Machine learning* dalam Analisis Penyebab Penyakit Gagal Jantung. *JEPIN : Jurnal Edukasi & Penelitian Informatika*, 10(2), 296–301. <https://doi.org/10.26418/jp.v10i2.78369>
- Nistrina, K., & Lestari, T. A. (2024). Desain Inovatif Sistem Informasi Profil Hotel Damanaka Pangalengan Berbasis Website Menggunakan UML dan Figma. *JurnalSistemInformasi, J-SIKA*, 6, 8–17.
- Pradnyana, G. A., Darmawiguna, I. G. M., & Wijaya, I. N. S. W. (2020). *Data Mining: Menemukan Pengetahuan dalam Data*. Depok : PT Raja Grafindo Persada.
- Primartha, R. (2018). *Belajar Machine learning : Teori Dan Praktik*. Bandung : Informatika.
- Sarno, R., Sabilla, S. I., Malikhah, Purbawa, D. P., & Ardani, M. S. H. (2022). *Machine learning dan Deep Learning-Konsep dan Pemrograman Python*. Yogyakarta : Andi.