

**PERBANDINGAN EFEKTIVITAS METODE *K-NEAREST NEIGHBOR*  
DAN *NAÏVE BAYES* DALAM DATA PENGAMATAN KESEHATAN  
TANAMAN**

**SKRIPSI**



Oleh :

**Nur Aizah**  
**2021503126**

**PROGRAM STUDI TEKNOLOGI INFORMASI  
FAKULTAS SAINS DAN TEKNOLOGI UNIVERSITAS IBRAHIMY  
SITUBONDO**

**2025**

**PERBANDINGAN EFEKTIVITAS METODE *K-NEAREST NEIGHBOR*  
DAN *NAÏVE BAYES* DALAM DATA PENGAMATAN KESEHATAN  
TANAMAN**

**SKRIPSI**



Oleh :

**Nur Aizah**

2021503126

**PROGRAM STUDI TEKNOLOGI INFORMASI  
FAKULTAS SAINS DAN TEKNOLOGI UNIVERSITAS IBRAHIMY  
SITUBONDO  
2025**

**HALAMAN JUDUL****PERBANDINGAN EFEKTIVITAS METODE *K-NEAREST NEIGHBOR*  
DAN *NAÏVE BAYES* DALAM DATA PENGAMATAN KESEHATAN  
TANAMAN****SKRIPSI**

Diajukan untuk Memenuhi Salah Satu Persyaratan dalam Menyelesaikan Program Sarjana (S-1) pada Program Studi Teknologi Informasi Fakultas Sains dan Teknologi Universitas Ibrahimy.

Oleh :

**NUR AIZAH**

2021503126

**PROGRAM STUDI TEKNOLOGI INFORMASI  
FAKULTAS SAINS DAN TEKNOLOGI UNIVERSITAS IBRAHIMY  
SITUBONDO**

**2025**

i

**PERNYATAAN KEASLIAN TULISAN**

Saya yang bertanda tangan dibawah ini:

Nama : Nur Aizah  
NPM/NIRM : 2021503126  
Program Studi : S-1 Teknologi Informasi  
Fakultas : Fakultas Sains dan Teknologi

Menyatakan dengan sebenarnya, bahwa laporan Hasil Penelitian (SKRIPSI) dengan judul “**Perbandingan Eektivitas Metode *K-Nearest Neighbor* dan *Naïve Bayes* dalam Data Pengamatan Kesehatan Tanaman**” secara keseluruhan adalah hasil penelitian atau karya saya sendiri, kecuali pada bagian-bagian yang dirujuk sebagai sebagai sumber referensi dan disebutkan dalam daftar Pustaka. Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa laporan Hasil Penelitian (SKRIPSI) ini hasil plagiasi, maka saya siap menerima sanksi atas perbuatan tersebut.

Situbondo, 20 Agustus 2025

Saya yang menyatakan,



**Nur Aizah**

**LEMBAR PERNYATAAN  
KESEDIAAN PUBLIKASI KARYA ILMIAH**

Saya yang bertanda tangan di bawah ini :

Nama : Nur Aizah  
NPM : 2021503126  
Program Studi : S-1 Teknologi Informasi  
Fakultas : Fakultas Sains dan Teknologi  
Jenis Karya Ilmiah : Hasil Penelitian

Demikian pengembangan ilmu pengetahuan, menyetujui untuk memberikan Hak Bebas Royalti Non-eksklusif (*Non-exclusive Royalty-Free Right*) kepada Perpustakaan Universitas Ibrahimy atas karya ilmiah saya berupa hasil penelitian yang berjudul :

**PERBANDINGAN EFEKTIVITAS METODE *K-NEAREST NEIGHBOR*  
DAN *NAÏVE BAYES* DALAM DATA PENGAMATAN KESEHATAN  
TANAMAN**

Dengan Hak Bebas Royalti Non-eksklusif ini Pusat Perpustakaan Universitas Ibrahimy berhak menyimpan, alih media/format, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat untuk dapat dipergunakan sebagaimana mestinya.

Situbondo, 20 Agustus 2025

Yang Menyatakan



**Nur Aizah**

**PERSETUJUAN PEMBIMBING**

Nama : **Nur Aizah**

NPM/NIRM : **2021503126**

Judul : **PERBANDINGAN EFEKTIVITAS METODE *K-NEAREST NEIGHBOR* DAN *NAÏVE BAYES* DALAM DATA PENGAMATAN KESEHATAN TANAMAN**

Telah ditelaah dan disetujui oleh pembimbing untuk diujikan pada sidang skripsi/munaqosah.



Telah disetujui oleh :

**Pembimbing I**

**Ahmad Homaidi, M. Kom.**

NIDN. 0705078901

**Pembimbing II**

**Lukman Fakhri Lidimilah, M. Kom.**

NIDN. 0715099001

**HALAMAN PENGESAHAN**

**SKRIPSI**

**PERBANDINGAN EFEKTIVITAS METODE *K-NEAREST NEIGHBOR*  
DAN *NAÏVE BAYES* DALAM DATA PENGAMATAN KESEHATAN  
TANAMAN**

**NUR AIZAH**

2021503126

Telah dipertahankan didepan Dewan Penguji Sidang/munaqasah skripsi pada hari  
Senin, 18 Agustus 2025 sebagai salah satu syarat memperoleh gelar sarjana  
(S.Kom) pada Fakultas Sains dan Teknologi Universitas Ibrahimiy.

Tim Penguji,

Ketua Sidang,



**Ahmad Homaidi, M.Kom.**  
NIDN. 0705078901

Sekretaris Sidang,



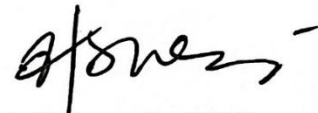
**Ahmad Ilzam, S.Kom.**  
NIDN.-

Penguji I,



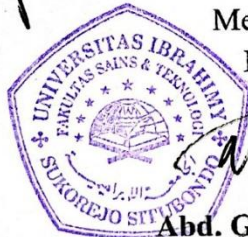
**Firman Santoso, M.Kom.**  
NIDN. 0722129201

Penguji II,



**Adi Susanto, M.Kom.**  
NIDN. 0708079104

Mengetahui  
Dekan,



**Abd. Ghofur, M. Kom.**  
NIDN. 0711088803

## MOTTO

رَبَّنَا إِنَّكَ جَامِعُ النَّاسِ لِيَوْمٍ لَا رَيْبَ فِيهِ ۗ إِنَّ اللَّهَ لَا يُخْلِفُ الْمِيعَادَ

"Ya Tuhan kami, sesungguhnya Engkau akan mengumpulkan manusia pada hari yang tak ada keraguan padanya.

Sungguh, Allah tidak menyalahi janji."

(QS. Ali Imran 3:9)

***"Fortes Fortuna Adiuvat"***

"Keberuntungan membantu orang-orang yang berani."



## PERSEMBAHAN

*Bismillahirrahmanirrahim...*

Segala puji dan syukur penulis panjatkan ke hadirat Allah SWT. atas segala limpahan rahmat, nikmat, dan karunia-Nya. Karya sederhana ini penulis persembahkan sebagai bentuk ungkapan rasa syukur, cinta, dan terima kasih yang mendalam kepada mereka yang telah menjadi bagian penting dalam perjalanan panjang, penuh makna, serta penuh ujian dalam proses penyusunan skripsi ini.

1. Untuk Ibu dan Bapak tercinta, sang pemilik doa, mahabbah, dukungan mental, finansial, moral, dan spiritual tak terhingga. Yang menjadi pondasi terkuat penulis untuk mampu berdiri sampai di titik ini. Semoga Allah membalas segala kebaikan keduanya dengan pahala yang tiada putus.
2. Para Masyayikh Ma'had Salafiyah Syafi'iyah Sukorejo, dosen pembimbing dan seluruh civitas akademik Fakultas Sains dan Teknologi Universitas Ibrahimi, yang telah membimbing dan memberikan ilmu yang luar biasa serta menjadi bagian penting dalam perjalanan penulis, baik secara akademis dan spiritual.
3. Teman-teman prodi TI angkatan 2021, keluarga besar BEM SAINTEK, dan PMII UNIB Situbondo yang telah menjadi rekan seperjuangan dalam menghadapi segala dinamika perkuliahan serta telah memberi ruang untuk bertumbuh, belajar, dan mengenal makna khidmat.
4. Seluruh sahabat dan teman hidup yang namanya tidak mampu saya sebutkan satu persatu, yang telah memberi dukungan, semangat, dan kontribusi berharga dalam berbagai bentuk selama proses penyusunan skripsi ini.
5. Untuk diri sendiri, yang tidak pernah lelah untuk bekerja keras, sabar, dan konsisten dalam menyelesaikan setiap langkah menempuh sarjana. Terima kasih diri, atas usaha yang tidak pernah surut meskipun banyak tantangan yang harus dihadapi. Semoga proses ini menjadi pengalaman berharga yang dapat memberikan pembelajaran untuk terus berkembang dan mencapai tujuan-tujuan berikutnya.

## KATA PENGANTAR

Segala puji syukur peneliti sampaikan kepada Allah SWT, karena atas rahmat dan hidayah-Nya, perencanaan, pelaksanaan dan penyelesaian tugas akhir/skripsi dengan judul **“Perbandingan Eektivitas Metode *K-Nearest Neighbor* dan *Naïve Bayes* dalam Data Pengamatan Kesehatan Tanaman”** sebagai salah satu syarat penyelesaian program diplomat/sarjana dapat terselesaikan dengan baik dan lancar.

Kesuksesan ini dapat peneliti peroleh karena dukungan beberapa pihak. Peneliti menyampaikan terima kasih kepada :

1. **KHR. Ach. Azaim Ibrahimi, S. Sy, M. HI** Selaku Pengasuh Pondok Pesantren Salafiyah Syafi'iyah Sukorejo Situbondo.
2. **KH. Ach. Fadlail, SH, M.H** Selaku Rektor Universitas Ibrahimi.
3. Bapak **Abd. Ghofur, M. Kom** Selaku Dekan Fakultas Sains dan Teknologi Universitas Ibrahimi.
4. Bapak **Ahmad Homaidi, M. Kom** selaku pembimbing I
5. Bapak **Lukman Fakhid Lidimilah, M. Kom** selaku pembimbing II
6. Bapak **Firman Santoso, M.Kom** selaku Ketua Program Studi Teknologi Informasi
7. Teruntuk Ibu dan Aba dan seluruh keluarga besar yang selalu mencurahkan kasih sayang yang tiada batas, yang tanpa lelah mendo'akan dan mendukung peneliti untuk tetap kuat sehingga dapat menyelesaikan tugas akhir ini.

*Jazaakumullahu Khoiron.*

Situbondo, 18 Agustus 2025

Peneliti,



Nur Aizah

**DAFTAR ISI**

<b>HALAMAN JUDUL .....</b>	<b>i</b>
<b>PERNYATAAN KEASLIAN TULISAN .....</b>	<b>ii</b>
<b>LEMBAR PERNYATAAN .....</b>	<b>ii</b>
<b>KESEDIAAN PUBLIKASI KARYA ILMIAH .....</b>	<b>iii</b>
<b>PERSETUJUAN PEMBIMBING .....</b>	<b>iv</b>
<b>HALAMAN PENGESAHAN .....</b>	<b>v</b>
<b>MOTTO.....</b>	<b>vi</b>
<b>PERSEMBAHAN .....</b>	<b>vii</b>
<b>KATA PENGANTAR.....</b>	<b>viii</b>
<b>DAFTAR ISI .....</b>	<b>ix</b>
<b>DAFTAR TABEL.....</b>	<b>xii</b>
<b>DAFTAR GAMBAR.....</b>	<b>xiii</b>
<b>DAFTAR RUMUS .....</b>	<b>xiv</b>
<b>ABSTRAK.....</b>	<b>xv</b>
<b>ABSTRACT.....</b>	<b>xvi</b>
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang.....	1
1.2 Identifikasi Masalah.....	5
1.3 Rumusan Masalah.....	6
1.4 Batasan Masalah .....	6
1.5 Tujuan Penelitian .....	7
1.6 Manfaat Penelitian .....	7
1.7 Metode Penelitian .....	8
1.7.1 Jenis penelitian .....	8
1.7.2 Tahapan pengumpulan data.....	8
1.8 Sistematika Pembahasan .....	9
<b>BAB II KAJIAN TEORI .....</b>	<b>11</b>
2.1 Penelitian Terdahulu .....	11

2.1.1	Perbandingan Metode <i>K-Nearest Neighbor</i> dan <i>Naive Bayes</i> Dalam Identifikasi Penyakit Daun Tanaman Obat[17] .....	11
2.1.2	<i>Identification of Corn Plant Diseases and Pests Based on Digital Images using Multinomial Naïve Bayes and K-Nearest Neighbor</i> [18].....	12
2.1.3	Klasifikasi Tanaman Herbal Menggunakan Metode <i>Naïve Bayes Clasifier</i> Dan <i>K-Nearest Neighbor</i> [19].....	13
2.2	Landasan Teori .....	14
2.2.1	Kesehatan Tanaman .....	14
2.2.2	Data Mining.....	15
2.2.3	<i>K-Nearest Neighbor</i> .....	16
2.2.4	<i>Naïve Bayes</i> .....	18
2.2.5	Dataset.....	18
2.2.6	<i>Python</i> .....	19
2.2.7	<i>Visual Studio Code</i> .....	19
<b>BAB III METODE PENELITIAN .....</b>		<b>21</b>
3.1	Metode Penelitian .....	21
3.1.1	Jenis dan Pendekatan Penelitian.....	21
3.1.2	Lokasi dan Waktu Penelitian.....	21
3.1.3	Populasi dan Sampel.....	21
3.1.4	Variabel Penelitian.....	22
3.1.5	Metode Pengumpulan Data.....	22
3.1.6	Instrumen Penelitian .....	23
3.1.7	Metode Analisis Data.....	23
3.2	Metode Perancangan Sistem .....	24
3.2.1	Pemahaman Bisnis ( <i>Business Understanding</i> ).....	25
3.2.2	Pemahaman Data ( <i>Data Understanding</i> ).....	26
<b>BAB IV HASIL DAN PEMBAHASAN.....</b>		<b>31</b>
4.1	Persiapan Data ( <i>Data Preparation</i> ) .....	31
4.2	Permodelan (Modeling) .....	36
a.	Algoritma <i>K-Nearest Neighbor</i> .....	37
b.	Algoritma <i>Naïve Bayes</i> .....	41

4.3	Evaluasi (Evaluation).....	44
4.4	Penyebaran (Deployment).....	51
<b>BAB V PENUTUP.....</b>		<b>53</b>
5.1	Kesimpulan.....	53
5.2	Saran .....	56
<b>DAFTAR PUSTAKA .....</b>		<b>57</b>
<b>LAMPIRAN .....</b>		<b>61</b>

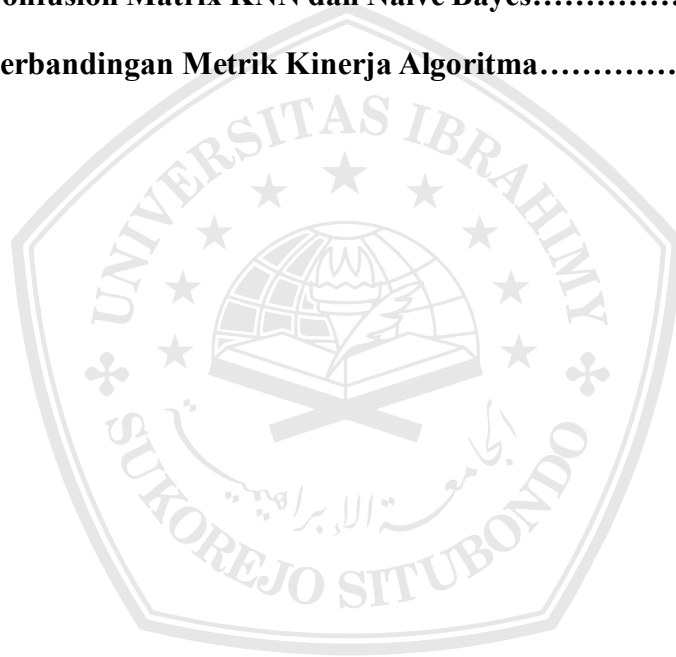


## DAFTAR TABEL

Tabel 1.1	Algoritma Klasifikasi Yang Umum Digunakan.....	3
Tabel 3.1	Data Kesehatan Tanaman.....	30
Tabel 4.1	Tabel Data Latih dan Data Uji.....	38
Tabel 4.2	Hasil Perhitungan ED.....	39
Tabel 4.3	Hasil Sortir Perhitungan ED dengan $K=5$ .....	39
Tabel 4.4	Hasil Perhitungan $P(X C)$ / <i>Likelihood</i> .....	42
Tabel 4.5	Tabel Hasil Pengujian Data .....	46
Tabel 4.6	Tabel Perbandingan Kinerja Algoritma.....	48
Tabel 5.1	Tabel Perbandingan Kinerja KNN vs <i>Naïve Bayes</i> .....	52

**DAFTAR GAMBAR**

<b>Gambar 3.1 Alur Metode CRISP-DM.....</b>	<b>24</b>
<b>Gambar 3.2 Distribusi Status Kesehatan Tanaman.....</b>	<b>26</b>
<b>Gambar 3.3 Distribusi Fitur Numerik.....</b>	<b>27</b>
<b>Gambar 3.4 Heatmap Korelasi antar Fitur Numerik.....</b>	<b>28</b>
<b>Gambar 4.1 Distribusi Kelas pada Training dan Validation Set.....</b>	<b>36</b>
<b>Gambar 4.2 Confusion Matrix KNN dan Naïve Bayes.....</b>	<b>45</b>
<b>Gambar 4.3 Perbandingan Metrik Kinerja Algoritma.....</b>	<b>49</b>



## DAFTAR RUMUS

(1) Rumus <i>Manhattan Distance</i> .....	17
(2) Rumus <i>Euclidean Distance</i> .....	17
(3) Rumus <i>Minkowski Distance</i> .....	17
(4) Rumus Teorema <i>Bayes</i> .....	18
(5) Tahap Perhitungan <i>Euclidean Distance</i> .....	36
(6) Tahap Perhitungan Probabilitas Prior.....	41
(7) Tahap Perhitungan <i>Likelihood</i> .....	42
(8) Tahap Perhitungan Probabilitas Posterior.....	43
(9) Rumus Perhitungan Akurasi.....	45
(10) Tahap Perhitungan <i>Precision</i> .....	47
(11) Tahap Perhitungan <i>Recall/Sensitivitas</i> .....	47
(12) Tahap Perhitungan <i>F1-Score</i> .....	48

## ABSTRAK

Nur Aizah, 2025. **Perbandingan Efektivitas Metode K-Nearest Neighbor dan Naïve Bayes dalam Data Pengamatan Kesehatan Tanaman.** Skripsi, Program Studi Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Ibrahimy. Pembimbing: (1) Ahmad Homaidi, M.Kom., (2) Lukman Fakhri Lidimilah, M.Kom.

Kesehatan tanaman merupakan aspek penting dalam menjaga produktivitas pertanian dan ketahanan pangan. Namun, metode identifikasi tradisional yang bersifat visual dan subjektif sering kali kurang akurat dan rentan terhadap kesalahan. Penelitian ini bertujuan untuk membandingkan efektivitas dua algoritma klasifikasi, yaitu *K-Nearest Neighbor* (K-NN) dan *Naïve Bayes* (NB), dalam mengklasifikasikan status kesehatan tanaman berdasarkan data pengamatan sensor lingkungan. Data yang digunakan terdiri dari 1.200 entri yang diperoleh dari *Kaggle*, mencakup 11 fitur seperti kelembaban tanah, suhu, pH, kandungan hara, dan sinyal elektrokimia, dengan label kesehatan tanaman (*Healthy*, *Moderate Stress*, *High Stress*). Penelitian menggunakan pendekatan CRISP-DM dengan tahapan *business understanding*, *data preparation*, *modeling*, *evaluation*, dan *deployment*. Evaluasi model menunjukkan bahwa *Naïve Bayes* memiliki performa lebih tinggi dengan akurasi 76,25% dan *F1-score* rata-rata 76,07%, dibandingkan K-NN yang hanya mencapai akurasi 56,67% dan *F1-score* 59,53%. Uji-T berpasangan membuktikan perbedaan performa kedua algoritma signifikan secara statistik. Hasil ini menunjukkan bahwa *Naïve Bayes* lebih efektif dan efisien untuk klasifikasi kesehatan tanaman berbasis data sensor numerik.

**Kata Kunci:** *Kesehatan Tanaman, K-Nearest Neighbor, Naïve Bayes, Klasifikasi, CRISP-DM.*

## **ABSTRACT**

Nur Aizah, 2025. *Comparison of the Effectiveness of the K-Nearest Neighbor and Naïve Bayes Methods in Plant Health Observation Data*. Undergraduate Thesis, Information Technology Study Program, Faculty of Science and Technology, Ibrahimy University. Supervisors: (1) Ahmad Homaidi, M.Kom., (2) Lukman Fakhid Lidimilah, M.Kom

*Plant health plays a crucial role in maintaining agricultural productivity and food security. However, traditional identification methods that rely on subjective visual observation are often inaccurate and prone to error. This study aims to compare the effectiveness of two classification algorithms, namely K-Nearest Neighbor (K-NN) and Naïve Bayes (NB), in predicting plant health status based on environmental sensor data. The dataset consists of 1,200 records obtained from Kaggle, containing 11 features such as soil moisture, temperature, pH, nutrient levels, and electrochemical signals, with plant health labels (Healthy, Moderate Stress, High Stress). The research follows the CRISP-DM approach, including business understanding, data preparation, modeling, evaluation, and deployment stages. Model evaluation shows that Naïve Bayes outperforms K-NN with an accuracy of 76.25% and an average F1-score of 76.07%, compared to K-NN with 56.67% accuracy and 59.53% F1-score. A paired T-test confirms that the performance difference between the two algorithms is statistically significant. These findings indicate that Naïve Bayes is more effective and efficient for classifying plant health using numerical sensor data.*

**Keywords:** *Plant Health, K-Nearest Neighbor, Naïve Bayes, Classification, CRISP-DM.*

## BAB I

### PENDAHULUAN

#### 1.1 Latar Belakang

Kesehatan tanaman memainkan peran yang sangat penting dan berpengaruh terhadap ketahanan pangan dan stabilitas ekonomi global. Menurut *Food and Agriculture Organization* (FAO), sekitar 40% produksi tanaman pangan global hilang setiap tahun akibat hama dan penyakit, yang menyebabkan kerugian ekonomi miliaran dolar serta mengancam sumber pangan bagi jutaan orang. Penurunan produktivitas tanaman tidak hanya mengurangi hasil panen, tetapi juga meningkatkan biaya produksi karena kebutuhan pestisida tambahan, perawatan intensif, dan pemulihan lahan[1].

Perubahan iklim dapat memperburuk risiko ini, membuat tanaman lebih rentan terhadap stres lingkungan seperti kekeringan atau serangan patogen baru. Tanaman dapat mengalami gangguan kesehatan yang disebabkan oleh tidak terpenuhinya kandungan unsur hara pada lingkungan sekitar sehingga menyebabkan penurunan produktivitas bahkan kematian yang berdampak langsung pada sektor pertanian dan ekonomi masyarakat[2]. Adapun unsur hara yang di serap oleh tanaman berasal dari udara dan tanah dengan kelembaban dan temperatur tertentu. Zat yang dibutuhkan tanaman seperti karbon, hidrogen, dan oksigen yang berasal dari udara serta fosfor, kalium, dan air yang berasal dari tanah. Berbagai variabel seperti kelembapan tanah, suhu tanah dan lingkungan, intensitas cahaya, kandungan klorofil, dan tingkat unsur hara (nitrogen, fosfor, kalium) dapat

digunakan untuk mengevaluasi tingkat kesehatan tanaman[3]. Data yang tersebut dapat memberikan informasi yang berharga mengenai kondisi tanaman yang dapat digunakan untuk mendeteksi stres atau gejala awal penyakit pada tanaman.

Setiap tanaman membutuhkan penanganan dan penanggulangan yang berbeda berdasarkan tingkat stress yang dialami. Diperlukan pengetahuan yang memadai untuk mengidentifikasi ciri-ciri tanaman yang mengalami stress sehingga penanganan dapat sesuai kebutuhan dan tidak berlebihan. Pada umumnya, identifikasi tingkat stress tanaman sering dilakukan dengan mengamati pola kerusakan berdasarkan kandungan unsur hara yang ada disekitarnya. Saat ini, perbedaan hasil pengamatan antar individu masih menjadi permasalahan yang cukup umum, terutama karena penilaian yang bersifat subjektif. Hal ini umumnya dipengaruhi oleh faktor-faktor seperti kelelahan dan kurangnya konsentrasi, yang menyebabkan manusia rentan melakukan kesalahan dalam identifikasi.

Ilmu pengetahuan tentang teknologi berkembang dengan pesat, begitu pula dengan teknologi klasifikasi. Klasifikasi adalah salah satu cabang dalam ilmu data yang berfokus pada pengelompokan atau pengkategorian data ke dalam kelas-kelas tertentu berdasarkan karakteristik atau atribut yang dimiliki oleh sebuah data. Klasifikasi bertujuan untuk membuat keputusan berdasarkan pola yang ditemukan dalam data historis. Secara teknis, klasifikasi merupakan bagian dari pembelajaran mesin (*machine learning*) yang termasuk dalam kategori pembelajaran terawasi (*supervised learning*)[4]. Dalam proses ini, algoritma dilatih menggunakan dataset yang telah diberi label atau kategori tertentu sebelumnya. Data pelatihan ini mengandung pasangan *input* (atribut) dan *output* (label) yang memungkinkan

algoritma belajar mengenali hubungan atau pola antara keduanya. Setelah pelatihan selesai, model klasifikasi diharapkan mampu mengklasifikasikan kelas dari data baru yang belum pernah dilihat sebelumnya [5]. Berikut beberapa algoritma klasifikasi yang umum dijumpai sebagai *baseline* dan karakteristiknya.

**Tabel 1.1 Algoritma Klasifikasi Yang Umum Digunakan**

Algoritma	Penggunaan	Alasan
<i>Naïve Bayes</i> (NB)	Sangat umum	Cepat, berbasis probabilistik, sederhana, cocok untuk data besar & teks[6]
<i>K-Nearest Neighbor</i> (KNN)	Sangat umum	Intuitif, non-parametrik, efektif untuk data numerik, tidak memerlukan pelatihan awal[7]
<i>C4.5 (Decision Tree)</i>	Kadang	Mudah diinterpretasikan, tapi memerlukan proses <i>pruning</i> dan pemisahan atribut kontinu [8]
<i>Support Vector Machine</i>	Tidak umum	Kompleks, memerlukan <i>tuning</i> parameter seperti kernel dan C, lebih cocok untuk evaluasi lanjutan[9]
<i>Random Forest</i> (RF)	Tidak umum	Akurat dan kuat terhadap <i>overfitting</i> , tetapi merupakan model <i>ensemble</i> yang lebih kompleks[10]
<i>Neural Network</i> (NN)	Tidak umum	Butuh data besar, tuning banyak parameter, dan proses training yang berat[11]
<i>Logistic Regression</i> (LR)	Kadang	Sederhana dan interpretable, tetapi hanya efektif untuk masalah klasifikasi linier sederhana[12]

Dalam penelitian ini, akan digunakan dua metode pembandingan, yaitu *K-Nearest Neighbor* (K-NN) dan *Naïve Bayes*, untuk mengatasi permasalahan klasifikasi yang bersifat subjektif dan rentan terhadap kesalahan manusia. Pemilihan kedua metode ini didasarkan pada keunggulan masing-masing dalam melakukan klasifikasi data. Kedua metode tersebut mewakili pendekatan yang

sederhana namun kuat dalam *machine learning*, sangat cocok untuk perbandingan awal, dan sering digunakan dalam penelitian karena mudah dipahami serta cepat diimplementasikan. Metode *K-Nearest Neighbor* sering dibahas dalam penelitian sebelumnya karena berhasil mengklasifikasikan faktor-faktor pada lingkungan yang mempengaruhi kesuburan tanaman, yakni kelembapan dan tingkat keasaman tanah. Metode *K-Nearest Neighbor* (K-NN) memiliki keunggulan dalam kemudahan implementasi serta kemampuannya menangani data yang besar dan kompleks[13]. Metode ini juga efektif untuk pelatihan data dalam jumlah besar dan tetap tangguh meskipun terdapat *noise*[14]. Klasifikasi dilakukan secara sederhana dengan menghitung jarak *Euclidean* untuk menentukan kedekatan antara data latih dan data uji, sehingga dapat menghasilkan akurasi yang baik.

Sebaliknya, dalam konteks kandungan unsur lingkungan yang mempengaruhi kesehatan tanaman, *Naïve Bayes* berhasil menghitung kemungkinan setiap status kesehatan berdasarkan fitur-fitur yang ada, seperti suhu tanah, kelembapan udara, dan kandungan unsur hara[15]. Metode ini efektif dalam menangani karakteristik data karena mampu melakukan klasifikasi meskipun jumlah data latih terbatas, serta menghasilkan error yang lebih kecil jika data latihnya banyak. Dengan penerapan Teorema *Bayes* dan asumsi independensi antar fitur, metode ini juga memungkinkan klasifikasi sederhana dan tetap menunjukkan akurasi tinggi pada dataset yang besar[16].

Berdasarkan beberapa studi literatur, penelitian mengenai identifikasi kesehatan tanaman menggunakan data numerik sensor tanah dan udara masih tergolong jarang. Sebagian besar penelitian yang ada saat ini hanya fokus pada

identifikasi penyakit tanaman berbasis citra dan berdasarkan faktor-faktor tanah seperti pH juga kelembapan, sehingga ketersediaan literatur yang relevan masih terbatas. Oleh karena itu, penting untuk melanjutkan penelitian dengan mempertimbangkan fitur lain seperti suhu, intensitas cahaya, klorofil dan lainnya.

Berdasarkan uraian tersebut, kedua metode ini memiliki kemampuan yang baik dalam menangani masalah klasifikasi terkait karakteristik data dan dalam mengidentifikasi kandungan unsur hara di lingkungan dengan hasil yang optimal. Oleh karena itu, untuk menentukan metode yang paling sesuai dalam mengidentifikasi tingkat stres pada tanaman, penting untuk melakukan perbandingan performa antara metode K-NN dan *Naïve Bayes*. Dengan membandingkan performa kedua metode ini, diharapkan penelitian dapat memberikan solusi yang lebih objektif dan akurat dalam proses klasifikasi, serta mengurangi pengaruh subjektivitas dalam pengambilan keputusan.

## 1.2 Identifikasi Masalah

Berdasarkan latar belakang diatas, maka didapatkan suatu identifikasi masalah yaitu:

### 1. Stress pada Tanaman

Tanaman rentan mengalami gangguan kesehatan akibat ketidakseimbangan unsur hara, stres lingkungan, dan ketidakmampuan sistem pemantauan tradisional untuk mendeteksi gejala awal penyakit secara akurat. Kondisi ini mengakibatkan kerugian ekonomi yang besar dan mengancam ketersediaan pangan, khususnya di negara-negara yang bergantung pada sektor pertanian.

## 2. Perbedaan Metode Klasifikasi

Metode *K-Nearest Neighbor* (K-NN) dan *Naïve Bayes* memiliki pendekatan berbeda dalam klasifikasi data. K-NN mengandalkan kedekatan data untuk menentukan kelas, sedangkan *Naïve Bayes* menggunakan pendekatan probabilistik. Perbedaan ini menimbulkan tantangan dalam memilih metode yang lebih efektif, karena K-NN unggul dalam akurasi, sementara *Naïve Bayes* lebih cepat namun kurang akurat pada data dengan fitur saling berkorelasi.

### 1.3 Rumusan Masalah

Rumusan masalah yang dapat diambil dari latar belakang diatas yaitu "Bagaimana cara efektif memantau kesehatan tanaman dan menentukan metode klasifikasi yang lebih akurat dan efisien antara *K-Nearest Neighbor* (K-NN) dan *Naïve Bayes*".

### 1.4 Batasan Masalah

Untuk memudahkan penulisan dalam penyusunan penelitian ini, dan agar pembahasan penelitian ini tidak menyimpang dari apa yang telah dirumuskan, maka penulis membatasi permasalahan sebagai berikut:

- a. Data yang akan dianalisis adalah data sekunder terkait kesehatan tanaman, dengan tujuan untuk mengukur dan mengklasifikasikan tingkat stres tanaman
- b. Penelitian ini dibatasi pada analisis kesehatan tanaman berdasarkan data lingkungan seperti suhu, kelembapan, intensitas cahaya, dan kandungan unsur hara.

- c. Fokus penelitian ini adalah pada perbandingan dua metode klasifikasi, *K-Nearest Neighbor* (KNN) dan *Naïve Bayes* menilai kinerja setiap metode.

### 1.5 Tujuan Penelitian

Tujuan pengamatan ini adalah untuk menentukan algoritma terbaik antara *K-Nearest Neighbor* (K-NN) dan *Naïve Bayes* dalam mengklasifikasikan kesehatan tanaman berdasarkan data pengamatan seperti suhu, kelembapan, intensitas cahaya, dan kandungan unsur hara. Evaluasi membandingkan atribut kinerja meliputi akurasi, presisi, *recall*, dan *f1-score* guna menentukan metode yang lebih optimal untuk pemantauan kesehatan tanaman secara efektif dan efisien.

### 1.6 Manfaat Penelitian

Penulis berharap analisis dan penelitian ini dapat memberikan kontribusi yang dapat diketahui oleh berbagai pihak khususnya penulis, dan pada umumnya oleh semua pihak yang terlibat dalam penyusunan proposal ini, antara lain:

- a. Memberikan wawasan dalam penerapan algoritma *K-Nearest Neighbor* (KNN) dan *Naïve Bayes* dalam analisis data pertanian serta memperkaya literatur ilmiah di bidang *Data Mining* dalam sektor pertanian.
- b. Menjadi acuan praktisi pertanian dalam memilih metode klasifikasi yang paling efektif untuk pemantauan kesehatan tanaman secara akurat dan efisien sehingga mampu meningkatkan produktivitas dan kualitas hasil pertanian.

## 1.7 Metode Penelitian

### 1.7.1 Jenis penelitian

#### a. *Library Research*

Data yang berkaitan dengan tingkat stres pada kesehatan tanaman, serta penggunaan algoritma *K-Nearest Neighbor* (KNN) dan *Naïve Bayes*, dihimpun dari berbagai referensi ilmiah, meliputi buku, jurnal, dan artikel yang diperoleh melalui perpustakaan digital.

#### b. Deskriptif Kuantitatif

Metode ini berfokus pada penyajian data dalam bentuk tabel, grafik, atau statistik deskriptif seperti rata-rata, median, modus, dan persentase untuk memberikan informasi tentang pola, hubungan, atau kecenderungan tertentu dalam data kesehatan tanaman.

### 1.7.2 Tahapan pengumpulan data

Penelitian ini menggunakan teknik studi pustaka yaitu mengumpulkan data dari berbagai sumber seperti buku, jurnal, artikel dan situs-situs yang dapat dipercaya sesuai dengan topik yang digunakan. Teknik pengumpulan data yang digunakan pada penelitian ini sebagai berikut:

#### a. Teknik Pengumpulan Data Sekunder

Data yang digunakan oleh penulis merupakan data sekunder yang disediakan oleh penyedia dataset online melalui website <https://www.kaggle.com/>

## b. Studi Pustaka

Teknik pengumpulan data yang kedua adalah studi pustaka. Studi pustaka bertujuan mencari informasi yang mendukung penelitian ini, baik melalui media seperti buku, internet, maupun media informasi lainnya.

## 1.8 Sistematika Pembahasan

### **BAB I : PENDAHULUAN**

Pada bab ini, latar belakang permasalahan yang menjadi dasar penelitian akan disajikan, diikuti dengan identifikasi masalah, rumusan masalah, serta tujuan dan manfaat penelitian. Bagian ini akan menguraikan batasan masalah yang ditetapkan dalam penelitian dan berfokus pada penyusunan landasan awal dan penjelasan mengenai pentingnya penelitian ini, serta arah penelitian yang akan diambil.

### **BAB II : KAJIAN TEORI**

Pada bab ini akan diulas penelitian terdahulu yang memiliki topik serupa agar memberikan pemahaman lebih dalam mengenai perkembangan penelitian di bidang tersebut. Serta tinjauan pustaka yang relevan dengan mencakup konsep-konsep dasar yang digunakan. Hal ini bertujuan untuk memberikan landasan teori yang kokoh sebagai dasar dalam pelaksanaan penelitian.

### **BAB III : METODE PENELITIAN**

Pada bab ini, metode penelitian yang digunakan akan dijelaskan secara rinci, mencakup jenis penelitian, perancangan sistem, serta tahapan yang digunakan dalam pemahaman dan analisis data. Bab ini bertujuan untuk memberikan

gambaran menyeluruh mengenai proses penelitian, serta prosedur yang diterapkan untuk memperoleh data yang valid dan dapat dipercaya.

#### **BAB IV : HASIL DAN PEMBAHASAN**

Bab ini menyajikan hasil-hasil yang diperoleh dari penelitian yang telah dilakukan. Hasil-hasil tersebut akan dianalisis dan dibahas secara mendalam, dengan penekanan pada interpretasi data yang diperoleh. Pembahasan ini bertujuan untuk mengevaluasi temuan-temuan penelitian dan memberikan pemahaman yang lebih baik mengenai implikasi dari hasil yang diperoleh.

#### **BAB IV : HASIL DAN PEMBAHASAN**

Bab ini menyajikan hasil-hasil yang diperoleh dari penelitian yang telah dilakukan. Hasil-hasil tersebut akan dianalisis dan dibahas secara mendalam, dengan penekanan pada interpretasi data yang diperoleh. Pembahasan ini bertujuan untuk mengevaluasi temuan-temuan penelitian dan memberikan pemahaman yang lebih baik mengenai implikasi dari hasil yang diperoleh.

#### **BAB V : PENUTUP**

Bab ini menyajikan kesimpulan yang dapat diambil berdasarkan hasil penelitian yang telah dilakukan. Selain itu, saran-saran untuk penelitian lebih lanjut atau penerapan praktis dari hasil penelitian akan diberikan. Bab ini bertujuan untuk merangkum seluruh hasil dan pembahasan yang telah dilakukan, serta memberikan pandangan akhir mengenai kontribusi penelitian.

## BAB II

### KAJIAN TEORI

#### 2.1 Penelitian Terdahulu

##### 2.1.1 Perbandingan Metode *K-Nearest Neighbor* dan *Naive Bayes* Dalam Identifikasi Penyakit Daun Tanaman Obat[17]

Penelitian ini mengacu pada tesis yang dilakukan oleh Putu Prianka Vedanty, Program Studi Ilmu Komputer Program Pascasarjana Universitas Pendidikan Ganesha Singaraja yang dipublikasikan pada 16 Februari 2024.

Penelitian ini bertujuan untuk mengidentifikasi penyakit daun pada tanaman obat dengan membandingkan metode *K-Nearest Neighbor* dan *Naive Bayes* guna meningkatkan akurasi deteksi penyakit daun tanaman obat. Identifikasi dini sangat penting karena petani di Indonesia masih mengandalkan pengamatan visual secara manual, yang seringkali tidak akurat dan berisiko menyebabkan salah diagnosis. Kesalahan ini dapat berdampak pada pemberian penanganan yang tidak tepat, seperti penggunaan pestisida berlebihan, yang akhirnya mengurangi hasil panen, meningkatkan biaya produksi, dan menghambat pertumbuhan tanaman.

Untuk mengatasi masalah tersebut, penelitian ini menggunakan pendekatan otomatis dengan membandingkan dua metode klasifikasi, yaitu *K-Nearest Neighbor* (K-NN) dan *Naive Bayes*, untuk mengidentifikasi penyakit daun tanaman obat berdasarkan ciri warna dan tekstur. Penelitian ini melakukan 12 skenario pengujian dengan menggunakan dataset citra tanaman obat yang sudah melalui proses augmentasi. Metode K-NN menunjukkan hasil yang lebih unggul, dengan akurasi tertinggi sebesar 89% pada skenario pengujian kelima menggunakan nilai

$K=2$  dan dataset berukuran 700x700 piksel. Sebaliknya, metode *Naïve Bayes* hanya mencapai akurasi 57,67% pada skenario pengujian keempat dengan dataset berukuran 500x500 piksel. Hasil ini menunjukkan bahwa K-NN lebih efektif dalam mengidentifikasi penyakit daun pada tanaman obat dibandingkan *Naïve Bayes*, meskipun *Naïve Bayes* masih dapat digunakan untuk analisis data yang lebih sederhana.

### **2.1.2 Identification of Corn Plant Diseases and Pests Based on Digital Images using Multinomial Naïve Bayes and K-Nearest Neighbor [18]**

Penelitian ini dilakukan oleh Yulia Resti, Chandra Irsan, Mega Tiara Putri, Irsyadi Yani, Ansyori Ansyori, dan Bambang Suprihatin. Diterbitkan oleh jurnal *Science & Technology Indonesia* Vol. 7 No. 1 halaman 29-35, pada Januari 2022.

Penelitian ini mengkaji penerapan *machine learning* statistik dalam identifikasi penyakit dan hama pada tanaman jagung (*Zea mays L*) menggunakan metode *Multinomial Naïve Bayes* (MNB) dan *K-Nearest Neighbor* (K-NN). Jagung merupakan bahan pangan yang penting sebagai sumber karbohidrat selain gandum dan beras, dan sangat rentan terhadap serangan hama dan penyakit yang dapat menurunkan kuantitas dan kualitas produksi. Oleh karena itu, mengidentifikasi penyakit dan hama pada tanaman jagung sangat penting untuk menjaga kualitas hasil pertanian. Dalam penelitian ini, dilakukan analisis terhadap 761 gambar digital yang menggambarkan enam kelas penyakit dan hama tanaman jagung.

Hasil penelitian menunjukkan bahwa metode *K-Nearest Neighbor* (K-NN) memberikan kinerja prediktif yang lebih baik dibandingkan dengan *Multinomial Naïve Bayes* (MNB). Pada metode MNB, dengan dua kategori, akurasi mencapai

92,72%, dengan nilai presisi 79,88%, recall 79,24%, F1-score 78,17%, kappa 72,44%, dan AUC 71,91%. Sedangkan pada metode K-NN dengan nilai  $k=3$ , akurasi mencapai 99,54%, dengan presisi 88,57%, recall 94,38%, F1-score 93,59%, kappa 94,30%, dan AUC 95,45%. Ini menegaskan bahwa K-NN lebih efektif dalam mendeteksi penyakit dan hama pada tanaman jagung, memberikan hasil yang lebih akurat dan konsisten dibandingkan dengan MNB.

Penelitian ini menunjukkan potensi penggunaan metode klasifikasi berbasis gambar digital sebagai solusi dalam deteksi dini penyakit dan hama tanaman, yang dapat membantu petani dalam mengambil langkah pencegahan yang lebih cepat dan tepat guna mengurangi kerugian produksi.

### **2.1.3 Klasifikasi Tanaman Herbal Menggunakan Metode *Naïve Bayes Clasifier* Dan *K-Nearest Neighbor*[19]**

Penelitian ini dilakukan oleh Fingki Junaidi Manurung, Sutejo, Lisawita, dan Fajrizal pada Prosiding Seminar Nasional Teknologi Informasi & Ilmu Komputer (SEMASTER) Vol. 3 No. 1 halaman 411-422, diterbitkan pada tanggal 12 November 2024.

Penelitian ini bertujuan untuk menganalisis dan membandingkan kinerja metode *Naïve Bayes Classifier* dan *K-Nearest Neighbor* (KNN) dalam proses klasifikasi jenis-jenis tanaman herbal. Tanaman herbal memiliki peran penting dalam kehidupan manusia, seperti sebagai sumber oksigen, bahan makanan, obat-obatan, dan bahan kosmetika. Untuk mengidentifikasi berbagai jenis tanaman herbal secara akurat, diperlukan metode klasifikasi yang efektif. Dalam penelitian

ini, digunakan fitur VGG16 untuk ekstraksi ciri-ciri daun herbal, dengan total 6 fitur yang dianalisis.

Metode *Naïve Bayes Classifier* dipilih karena kesederhanaan dan kecepatannya dalam melakukan klasifikasi berbasis probabilitas, sedangkan metode *K-Nearest Neighbor* (KNN) digunakan karena kemampuannya mengelompokkan data berdasarkan jarak terdekat antar objek. Berdasarkan hasil uji coba, metode *Naïve Bayes Classifier* menunjukkan performa yang lebih baik dengan akurasi 92,86%, sementara metode KNN hanya mencapai akurasi 82,14%. Hasil ini menunjukkan bahwa *Naïve Bayes* lebih unggul dalam mengklasifikasikan data tanaman herbal, baik dari segi kecepatan pemrosesan maupun ketepatan klasifikasi, dibandingkan KNN yang meskipun sederhana, kurang efektif untuk data dengan pola distribusi kompleks.

## 2.2 Landasan Teori

Landasan teori dalam penelitian ini mencakup berbagai referensi dan definisi dari sejumlah kata kunci yang relevan dengan topik yang akan dibahas. Berikut ini disajikan beberapa definisi tersebut sebagai acuan utama dalam penelitian.

### 2.2.1 Kesehatan Tanaman

Kesehatan tanaman adalah kondisi fisik dan fisiologis tanaman yang mencerminkan kemampuannya untuk tumbuh dan berkembang dengan baik, bebas dari gangguan hama, penyakit, kekurangan nutrisi, atau stres lingkungan[20]. Tanaman yang sehat umumnya memiliki ciri-ciri seperti daun yang hijau segar, batang yang kuat, akar yang aktif dalam menyerap nutrisi, serta kemampuan

reproduksi yang optimal. Dalam ilmu agronomi, kesehatan tanaman juga diukur oleh keseimbangan berbagai faktor yang memengaruhi pertumbuhannya, seperti unsur hara yang cukup, kelembaban tanah yang tepat, intensitas cahaya yang sesuai, serta suhu dan pH tanah yang mendukung.

Dari sisi biologi, kesehatan tanaman juga dipengaruhi oleh interaksi genetik tanaman dengan mikroorganisme yang ada di rizosfer (zona perakaran). Faktor-faktor lingkungan seperti kelembaban dan suhu tanah, berperan penting dalam mendukung proses fotosintesis, metabolisme, dan pertumbuhan akar[21]. Kesehatan tanaman juga sangat bergantung pada ketersediaan unsur hara esensial seperti nitrogen, fosfor, dan kalium yang mendukung pembentukan daun, akar, dan buah. Selain itu, kandungan klorofil pada tanaman menjadi indikator utama dalam menunjukkan seberapa efektif tanaman melakukan fotosintesis untuk menghasilkan energi[22]. Secara keseluruhan, kesehatan tanaman adalah hasil interaksi kompleks antara faktor genetik, lingkungan, dan mikroorganisme yang mendukung pertumbuhannya secara optimal.

### 2.2.2 Data Mining

*Data Mining*, atau penggalian data, merupakan proses eksplorasi informasi dari kumpulan data untuk mengidentifikasi pola, tren, dan hubungan yang dapat mendukung pengambilan keputusan strategis. Dalam praktiknya, penggalian data digunakan untuk mengolah data dalam jumlah besar menjadi informasi spesifik yang relevan sehingga dapat memberikan solusi atas berbagai permasalahan. Penggalian data menjadi salah satu elemen penting dalam analisis data dan merupakan bagian integral dari disiplin ilmu data. Pada tingkat yang lebih

kompleks, proses ini melibatkan pendekatan sistematis yang dikenal sebagai *Knowledge Discovery in Database* (KDD)[23], yaitu metodologi yang dirancang untuk mengumpulkan, memproses, dan menganalisis data guna menghasilkan pengetahuan baru yang berguna.

Salah satu metode dalam *Data Mining* adalah klasifikasi, yang digunakan untuk mengelompokkan data ke dalam kategori tertentu berdasarkan pola-pola yang ditemukan dalam dataset, sehingga memudahkan dalam pengambilan keputusan dan klasifikasi[24]. *Data Mining* sendiri merupakan gabungan dari berbagai disiplin ilmu, yang mencakup teknik-teknik seperti pembelajaran mesin, pengenalan pola, statistik, pengelolaan basis data, dan visualisasi. Semua teknik tersebut bekerja bersama-sama untuk memecahkan masalah dalam pengambilan informasi dari basis data besar secara efisien dan efektif[25].

### 2.2.3 *K-Nearest Neighbor*

*K-Nearest Neighbor* (KNN) merupakan algoritma berbasis *instance* yang mengklasifikasikan data berdasarkan kedekatannya dengan data yang sudah diketahui labelnya. Disebut juga sebagai pembelajaran malas (*Lazy Learning*) karena tidak menggunakan titik data pelatihan untuk membangun model. Dengan kata lain algoritma KNN mempunyai fase pelatihan yang sangat minim. Tujuannya adalah mengklasifikasikan objek baru berdasarkan kemiripan dengan sampel dalam data pelatihan. KNN melakukan klasifikasi dengan mengidentifikasi nilai K data terdekat dari suatu titik berdasarkan perhitungan jarak[26]. Pendekatan ini memungkinkan KNN untuk menyesuaikan diri terhadap berbagai pola dalam data dan menghasilkan klasifikasi berdasarkan informasi lokal.

a. *Euclidean Distance*

Metode ini merupakan metode pengukuran jarak paling umum dalam algoritma *K-Nearest Neighbors* (KNN). Jarak ini menggambarkan jarak lurus (seperti garis diagonal) antara dua titik dalam ruang berdimensi banyak [27]. *Euclidean Distance* cocok digunakan pada data berskala kontinu dan tidak terlalu terpengaruh oleh *outlier*.

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (1)$$

b. *Manhattan Distance*

*Manhattan distance* atau dikenal juga sebagai *L1-norm* atau *Taxicab Distance*. Jarak ini menghitung selisih absolut antar titik secara horizontal dan vertikal, menyerupai cara taksi berpindah di jalan berbentuk *grid* [28]. Metode ini lebih tahan terhadap *outlier* dibandingkan *Euclidean*.

$$d(x, y) = \sum_{i=1}^n |x_i - y_i| \quad (2)$$

c. *Minkowski Distance*

*Minkowski Distance* adalah bentuk generalisasi dari kedua metode sebelumnya. Parameter  $p$  dapat disesuaikan untuk mengubah karakteristik jaraknya. Jika  $p=1$ , maka sama dengan *Manhattan Distance*, dan jika  $p=2$ , maka akan menjadi *Euclidean Distance* [29].

$$d(x, y) = \left( \sum_{i=1}^n |x_i - y_i|^p \right)^{\frac{1}{p}} \quad (3)$$

Pemilihan nilai  $K$  (jumlah tetangga terdekat) merupakan faktor yang sangat penting dalam algoritma *K-Nearest Neighbors* (KNN). Nilai  $K$  sebaiknya

disesuaikan dengan karakteristik data yang digunakan. Jika data mengandung banyak *outlier* atau *noise*, maka pemilihan nilai  $K$  yang lebih besar dapat memberikan hasil yang lebih stabil. Selain itu, penggunaan nilai  $K$  ganjil lebih disarankan agar dapat menghindari hasil klasifikasi yang imbang atau seri dalam proses voting klasifikasi.

#### 2.2.4 Naïve Bayes

*Naïve Bayes* adalah salah satu algoritma pembelajaran mesin berbasis probabilistik dalam klasifikasi data serta memiliki kecepatan dan efisiensi yang tinggi. Sehingga menjadi pilihan populer dalam aplikasi yang memerlukan analisis cepat pada jumlah data yang besar. Algoritma ini berlandaskan pada Teorema Bayes, yang pertama kali diperkenalkan oleh seorang matematikawan Inggris, Thomas Bayes, pada abad ke-18 [30].

$$P(C|X) = \frac{P(C) * P(X|C)}{P(X)} \quad (4)$$

Keterangan :

$P(C|X)$  = Probabilitas posterior, yaitu probabilitas kelas  $C$  berdasarkan data fitur  $X$

$P(C)$  = Probabilitas prior dari kelas  $C$

$P(X|C)$  = *Likelihood*, yaitu probabilitas data fitur  $X$  muncul dalam kelas  $C$

$P(X)$  = *Evidence*, yaitu probabilitas keseluruhan data fitur  $X$  muncul

#### 2.2.5 Dataset

Dataset merupakan kumpulan data yang terstruktur dan tersusun secara sistematis. Setiap barisnya merepresentasikan satu entitas dan setiap kolom menunjukkan atribut atau variabel yang menjelaskan karakteristik entitas

tersebut[31]. Dalam bidang ilmu data dan pembelajaran mesin, dataset merupakan dasar utama untuk proses pelatihan, pengujian, serta evaluasi model dalam mengenali pola dan hubungan antar data. Dataset umumnya dibagi menjadi *training set* untuk melatih model dan *validation set* untuk mengukur kinerjanya. Kualitas dataset sangat memengaruhi hasil akhir model, sehingga perlu dilakukan tahapan seperti normalisasi, pengisian nilai yang hilang, dan pembersihan data[32]. Dataset dapat berupa data numerik, kategorikal, teks, atau kombinasi dari berbagai jenis data, tergantung pada kebutuhan dan tujuan analisis.

#### 2.2.6 Python

*Python* adalah bahasa pemrograman tingkat tinggi ilmu data, pemrograman pembelajaran mesin (*machine learning*), analisis data, dan automasi[33]. Beberapa pustaka penting untuk pengolahan dan analisis data yang digunakan adalah *NumPy* dan *Pandas*, yang memungkinkan manipulasi dan pembersihan data secara efisien. *Scikit-learn* adalah pustaka utama dalam klasifikasi serta menyediakan berbagai fungsi untuk evaluasi model menggunakan metrik seperti akurasi, presisi, *recall*, dan *F1-score*. Untuk visualisasi data dan hasil analisis, *Python* dilengkapi dengan *Matplotlib* dan *Seaborn*, yang memungkinkan pengguna untuk membuat grafik dan plot yang menggambarkan distribusi data, hubungan antar variabel, serta kinerja model[34].

#### 2.2.7 Visual Studio Code

*Visual Studio Code (VS Code)* adalah *source-code* editor lintas *platform* yang dikembangkan oleh Microsoft dan dirilis pertama kali pada tahun 2015 [35]. *VS Code* bersifat *open source* dan gratis, mendukung berbagai bahasa pemrograman

melalui *extension* atau *plugin* yang dapat diunduh dari *Visual Studio Code Marketplace*. Editor ini dilengkapi dengan fitur-fitur seperti *syntax highlighting*, *IntelliSense* (penyelesaian kode cerdas), *debugger* terintegrasi, *version control* berbasis Git, dan terminal bawaan [36], sehingga memudahkan pengembang dalam menulis, menguji, dan mengelola kode. Dukungan terhadap berbagai sistem operasi seperti *Windows*, *macOS*, dan *Linux* menjadikan *VS Code* fleksibel digunakan oleh banyak kalangan, mulai dari pelajar hingga profesional di bidang pengembangan perangkat lunak.



## BAB III

### METODE PENELITIAN

#### 3.1 Metode Penelitian

##### 3.1.1 Jenis dan Pendekatan Penelitian

Jenis penelitian yang digunakan adalah kuantitatif deskriptif, karena penelitian ini menekankan pada pengumpulan dan analisis data numerik untuk mengidentifikasi hubungan antar variabel dan mengukur tingkat akurasi klasifikasi[37].

Pendekatan yang digunakan adalah eksperimen dengan studi komparatif, yaitu membandingkan dua algoritma klasifikasi (*K-Nearest Neighbor* dan *Naïve Bayes*) dalam mendeteksi tingkat kesehatan tanaman berdasarkan data sensor.

##### 3.1.2 Lokasi dan Waktu Penelitian

- Lokasi Penelitian

Penelitian ini dilakukan secara *in-silico* (berbasis komputer), menggunakan data sekunder dari *platform* Kaggle, sehingga tidak dilakukan di lokasi fisik tertentu[38].

- Waktu Pelaksanaan

Penelitian dilakukan pada periode April 2025 hingga Juli 2025.

##### 3.1.3 Populasi dan Sampel

Populasi dalam penelitian ini adalah seluruh data observasi yang berkaitan dengan kondisi fisiologis tanaman yang tersedia dalam dataset *Plant Health Data*. Sampel yang digunakan berjumlah 1.200 baris data, yang masing-masing

mencakup parameter hasil pembacaan sensor seperti suhu, kelembapan, kandungan unsur hara (nitrogen, fosfor, kalium), intensitas cahaya, serta sinyal elektrokimia. Teknik pengambilan sampel yang digunakan adalah *total sampling*, yaitu seluruh data yang tersedia dalam dataset dimanfaatkan sepenuhnya sebagai sampel penelitian[39].

### 3.1.4 Variabel Penelitian

a. Variabel Independen (X):

1. *Soil Moisture*
2. *Ambient Temperature*
3. *Soil Temperature*
4. *Humidity*
5. *Light Intensity*
6. *Soil pH*
7. *Nitrogen Level*
8. *Phosphorus Level*
9. *Potassium Level*
10. *Chlorophyll Content*
11. *Electrochemical Signal*

b. Variabel Dependen (Y):

1. Status Kesehatan Tanaman (*Healthy, Moderate Stress, High Stress*)

### 3.1.5 Metode Pengumpulan Data

Teknik yang digunakan untuk pengumpulan data dalam penelitian ini adalah melalui studi dokumentasi dan pemanfaatan sensor atau alat ukur. Studi

dokumentasi dilakukan dengan menggunakan dataset sekunder yang diperoleh dari *platform* Kaggle, yaitu kumpulan data yang telah tersedia dan terdokumentasi dengan baik[40]. Sementara itu, data awal dalam dataset tersebut berasal dari pengukuran sensor lingkungan dan fisiologis tanaman, seperti kelembapan tanah, suhu, pH, kadar unsur hara, cahaya, dan sinyal elektrokimia, yang menjadi dasar dalam proses klasifikasi kesehatan tanaman.

### 3.1.6 Instrumen Penelitian

#### a. Instrumen

- Dataset digital *Plant Health Data*
- Bahasa pemrograman *Python* dengan pustaka pendukung seperti *Scikit-Learn*, *Pandas*, *Matplotlib*, dan *Seaborn*
- Perangkat lunak pendukung analisis: *Visual Studio Code*

#### b. Validitas dan Reliabilitas

Karena data bersifat sekunder dan bersumber dari *platform* terpercaya (Kaggle), maka validitas awal diasumsikan memadai. Reliabilitas diuji melalui replikasi eksperimen dan uji performa model.

### 3.1.7 Metode Analisis Data

Analisis data dilakukan menggunakan metode:

- Statistik Deskriptif: Distribusi kelas, korelasi fitur, visualisasi *heatmap*.
- Algoritma Klasifikasi
  1. *K-Nearest Neighbor* (K-NN): dengan parameter  $k = 5$  dan menggunakan jarak *Euclidean*.
  2. *Naïve Bayes* (GNB): menggunakan distribusi *Gaussian*.

- Evaluasi Model

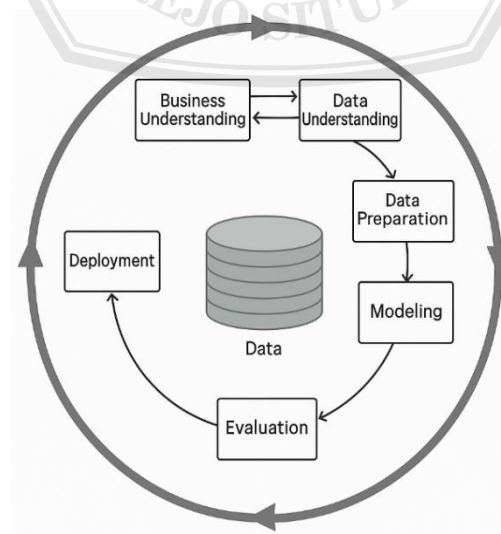
Menggunakan metrik *Accuracy*, *Precision*, *Recall*, dan *F1-Score*.

- Uji Statistik

Menggunakan Uji-T Berpasangan (*Paired T-Test*) untuk mengetahui perbedaan signifikan antara performa kedua model.

### 3.2 Metode Perancangan Sistem

Perancangan sistem pada penelitian ini mengacu pada pendekatan *Cross Industry Standard Process for Data Mining* (CRISP-DM), yang merupakan suatu metodologi standar dalam pengembangan sistem berbasis data. Pendekatan ini dipilih karena mampu menyediakan kerangka kerja yang terstruktur namun tetap fleksibel, sehingga memudahkan proses pengolahan data hingga tahap implementasi[41]. CRISP-DM terdiri atas enam tahapan utama, yaitu pemahaman terhadap konteks bisnis, pemahaman data, persiapan data, pembuatan model (*modeling*), evaluasi model (*evaluation*), serta penyebaran hasil (*deployment*).



**Gambar 3.1** Alur Metode CRISP-DM

### 3.2.1 Pemahaman Bisnis (*Business Understanding*)

*Business understanding* adalah tahap awal dalam proses pengembangan sistem berbasis data, di mana fokus utamanya adalah memahami tujuan utama bisnis, kebutuhan pengguna atau organisasi, serta permasalahan yang ingin diselesaikan.

#### a. Tujuan Bisnis

Tujuan utama dari proyek ini adalah membandingkan algoritma *machine learning* *K-Nearest Neighbor* (KNN) dan *Naïve Bayes* dalam hal akurasi dan efektivitas untuk klasifikasi kesehatan tanaman berdasarkan data sensor sebanyak 1.200 baris. Algoritma terbaik akan dijadikan dasar dalam membangun model klasifikasi yang dapat digunakan oleh sistem pertanian cerdas dalam pengambilan keputusan secara otomatis.

#### b. Kebutuhan Bisnis

- Mengembangkan sistem klasifikasi berbasis *machine learning* yang akurat, efisien, dan siap digunakan.
- Menyediakan bukti kuantitatif (akurasi, *precision*, *recall*, dan *F1-score*) dari kedua algoritma untuk mendukung keputusan pemilihan model.
- Menghasilkan model yang mampu mengklasifikasikan data tanaman baru dengan tingkat kepercayaan tinggi, untuk membantu intervensi lebih cepat dalam perawatan tanaman.

#### c. Permasalahan

- Pengamatan visual bersifat subyektif, hasilnya tidak seragam, memakan waktu, dan rawan error.

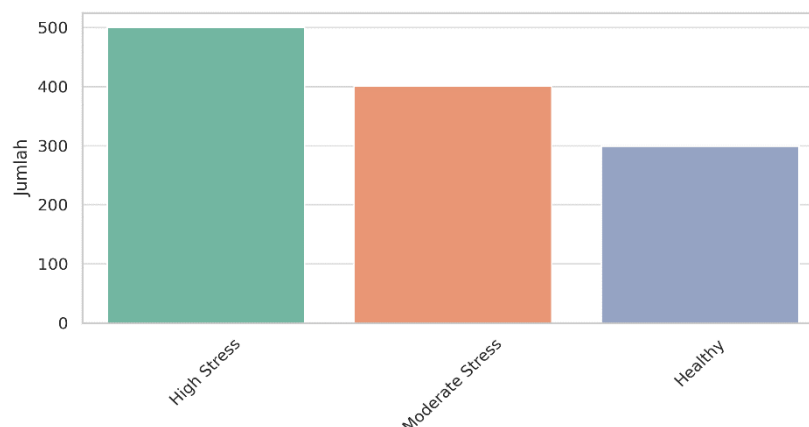
- Dibutuhkan bukti kuantitatif dalam pengambilan keputusan berupa metrik terukur sebelum mengadopsi model AI.

### 3.2.2 Pemahaman Data (*Data Understanding*)

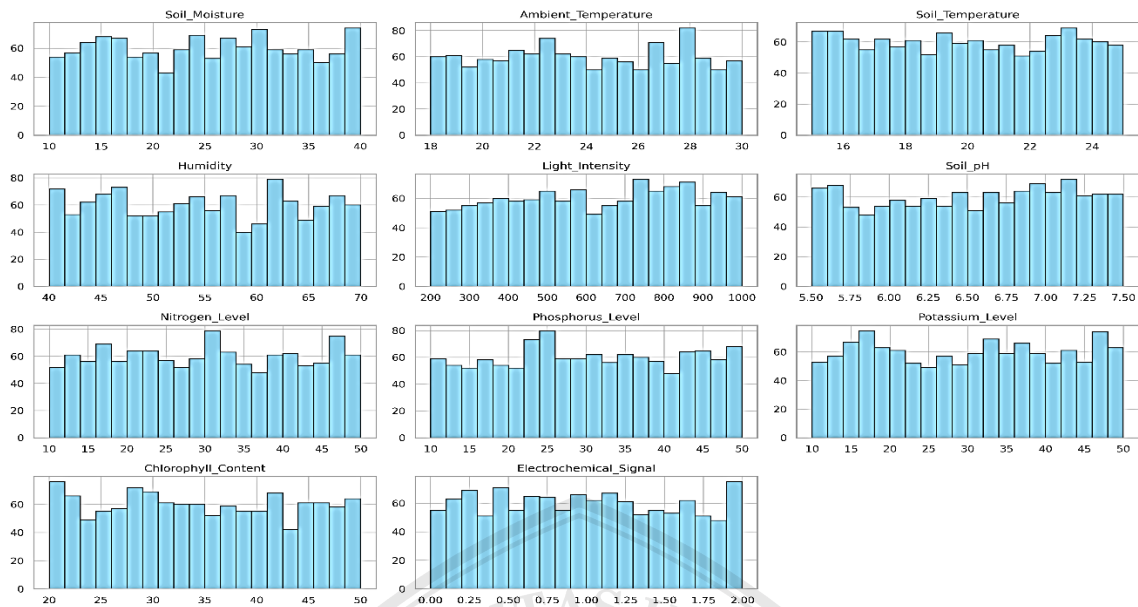
Proses ini bertujuan untuk mengevaluasi apakah data yang dimiliki cukup, bersih, dan layak untuk diproses sebagai pondasi dari seluruh proses data *mining*.

Penelitian menggunakan data sekunder yang diperoleh dari repositori *online* Kaggle, yaitu *Plant Health Data* dengan tautan akses <https://www.kaggle.com/datasets/ziya07/plant-health-data>. Data sekunder adalah data yang diperoleh dari sumber yang sudah ada dan tidak dikumpulkan secara langsung oleh peneliti[42]. Dataset ini berisi 1.200 entri dengan 14 kolom, mencakup berbagai parameter lingkungan dan nutrisi tanaman serta status kesehatan tanaman.

Grafik berikut menunjukkan distribusi status kesehatan tanaman dalam dataset dan memberikan gambaran awal tentang keseimbangan kelas yang penting untuk analisis lanjutan seperti klasifikasi. Terdapat 500 data kesehatan tanaman dengan label *High Stress*, 401 data *Moderate Stress*, dan 299 data *Healthy*.



**Gambar 3.2 Distribusi Status Kesehatan Tanaman**

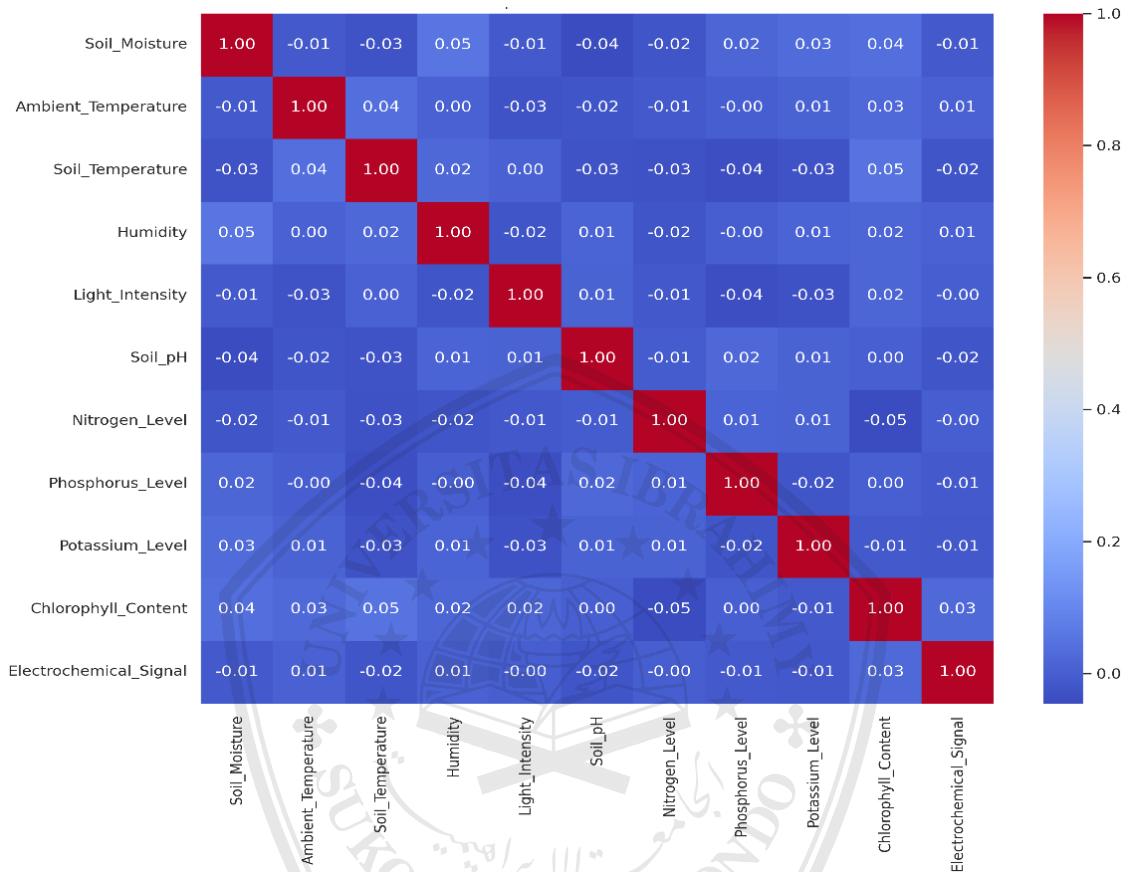


**Gambar 3.3 Distribusi Fitur Numerik**

Berdasarkan gambar distribusi fitur numerik, terlihat bahwa beberapa fitur memiliki distribusi yang hampir merata (*uniform*). Hal ini menunjukkan tidak adanya dominasi nilai tertentu. Selain itu, tidak tampak adanya *outlier* signifikan yang menandakan bahwa data bersih dan stabil untuk proses pelatihan model.

*Heatmap* adalah grafik visual yang digunakan untuk menunjukkan hubungan atau korelasi antar fitur numerik dalam suatu dataset. Grafik ini memakai gradasi warna untuk mempermudah pembacaan kekuatan hubungan antar fitur. Dalam penelitian ini, heatmap menggambarkan korelasi *Pearson*, yaitu ukuran statistik yang menunjukkan sejauh mana dua variabel numerik saling berhubungan secara linear. Nilai korelasi berkisar dari +1 hingga -1, di mana +1 menunjukkan korelasi positif sempurna (kedua fitur meningkat bersama), 0 berarti tidak ada hubungan linear, dan -1 menunjukkan korelasi negatif sempurna (satu fitur meningkat, lainnya menurun). Jika dua fitur memiliki korelasi sangat tinggi, salah satunya

dapat dihilangkan untuk menyederhanakan model tanpa kehilangan informasi penting.



**Gambar 3.4 Heatmap Korelasi antar Fitur Numerik**

Berdasarkan hasil analisis *heatmap* korelasi antar fitur numerik, diperoleh informasi bahwa sebagian besar pasangan fitur dalam dataset memiliki nilai korelasi yang sangat rendah, yaitu mendekati nol. Nilai korelasi yang rendah ini menandakan bahwa tidak terdapat hubungan linear yang kuat di antara fitur-fitur tersebut. Dengan kata lain, perubahan nilai pada suatu fitur tidak secara signifikan diikuti oleh perubahan nilai pada fitur lainnya.

Kondisi ini menunjukkan bahwa setiap fitur bersifat relatif independen, sehingga tidak saling memengaruhi secara langsung. Hal ini penting karena ketiadaan hubungan linear yang kuat mengurangi risiko terjadinya

multikolinearitas, yaitu keadaan di mana beberapa fitur memiliki hubungan yang sangat erat sehingga dapat mengganggu proses pembelajaran model. Multikolinearitas sering kali menjadi masalah pada model yang sensitif terhadap korelasi antar fitur, namun pada dataset ini, masalah tersebut tidak ditemukan.

Keadaan tersebut juga memberikan keuntungan pada penerapan model pembelajaran mesin seperti *K-Nearest Neighbors* (KNN) dan *Naive Bayes*. KNN bekerja dengan mengukur jarak antar titik data pada ruang fitur tanpa memerlukan asumsi korelasi antar fitur, sedangkan *Naive Bayes* bahkan mengasumsikan bahwa setiap fitur saling bebas secara kondisional. Oleh karena itu, dataset dengan tingkat korelasi rendah antar fitur seperti ini dapat mendukung kinerja kedua algoritma tersebut secara optimal.

Berdasarkan hasil visualisasi *heatmap*, pasangan fitur dengan nilai korelasi tertinggi adalah *Soil\_Temperature* dan *Chlorophyll\_Content*, yaitu sebesar 0,05. Meskipun nilai ini masih tergolong sangat lemah, korelasi tersebut merupakan yang paling tinggi di antara semua pasangan fitur yang ada. Pada *heatmap*, perbedaan ini ditunjukkan oleh warna biru yang sedikit lebih terang pada sel perpotongan keduanya, dan angka 0,05 tercantum secara eksplisit di dalam sel tersebut. Hal ini menguatkan kesimpulan bahwa secara umum, hubungan linear antar fitur dalam dataset ini sangat lemah, dengan tidak ada pasangan fitur yang menunjukkan korelasi yang signifikan.

**Tabel 3.1. Data Kesehatan Tanaman**

No	Timestamp	Plant_ID	Soil_Moist	Ambient_Temp	Soil_Temp	Humidity	Light_Intensity	Soil_pH	Nitrogen	Phosphorus_Level	Potassium_Level	Chlorophyll_Content	Electrochemical	Plant_Health_Status
1	2024-10-03 10:54:53.407995	1	27,521	22,240	21,900	55,292	556,173	5,582	10,004	45,807	39,076	35,703	0,941	High Stress
2	2024-10-03 10:54:53.426551	2	34,995	20,225	19,522	66,291	846,819	7,032	12,474	14,963	44,974	49,788	0,963	High Stress
3	2024-10-03 10:54:53.439542	3	31,901	18,408	15,375	45,282	472,839	6,752	37,519	20,889	33,436	22,676	1,578	Healthy
4	2024-10-03 10:54:53.449536	4	32,516	26,853	23,113	47,474	839,613	5,683	43,378	40,899	33,880	49,083	1,943	Healthy
5	2024-10-03 10:54:53.458530	5	13,653	22,662	20,980	54,418	800,285	7,233	15,283	18,058	31,671	41,851	0,673	High Stress
6	2024-10-03 10:54:53.467524	6	18,746	27,642	22,679	43,860	210,667	6,388	23,414	29,392	39,931	30,715	0,825	High Stress
7	2024-10-03 10:54:53.473520	7	21,518	22,361	19,834	54,203	416,583	7,200	45,801	22,255	27,825	24,326	0,115	Moderate Stress
8	2024-10-03 10:54:53.480516	8	16,781	26,204	24,584	54,868	576,543	6,156	36,519	41,221	39,429	27,704	0,189	High Stress
9	2024-10-03 10:54:53.486512	9	23,220	28,751	19,808	69,087	906,363	6,338	15,063	40,966	37,691	40,157	1,032	Moderate Stress
10	2024-10-03 10:54:53.493508	10	25,123	22,063	24,395	58,781	652,939	5,692	23,716	19,506	27,263	29,777	0,052	Moderate Stress
11	2024-10-03 16:54:53.407995	1	14,836	21,707	18,681	63,949	596,137	7,136	30,713	25,394	17,945	27,993	0,165	High Stress
12	2024-10-03 16:54:53.426551	2	16,578	28,678	24,781	51,413	980,282	7,151	47,657	19,913	33,855	31,772	1,969	High Stress
...	.....	...	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....
1200	2024-11-02 04:54:53.493508	10	24,406	25,725	23,359	47,505	304,202	6,122	29,659	34,373	10,486	41,977	0,048	Moderate Stress

## BAB IV

### HASIL DAN PEMBAHASAN

#### 4.1 Persiapan Data (*Data Preparation*)

- a. Menghapus kolom yang tidak relevan

Kolom identitas yang tidak memiliki kontribusi prediktif terhadap status kesehatan tanaman dihilangkan untuk meningkatkan efisiensi pemodelan.

##### Segmen Program 4.1 Menghapus kolom *plant\_id* dan *timestamp*

```
#hilangkan plant_id dan timestamp
1 df = df.drop(columns = ['Timestamp', 'Plant_ID'] )
2 df.head(10)
```

- b. Definisi fitur (X) dan target (Y)

Fitur (X) terdiri atas kolom parameter fisiologis tanaman seperti kelembaban tanah, suhu lingkungan dan tanah, pH, kadar nitrogen, fosfor, kalium, kandungan klorofil, serta sinyal elektrokimia. Sementara itu, target (Y) yang akan diklasifikasi adalah *Plant\_Health\_Status*.

##### Segmen Program 4.2 Definisi fitur dan target

```
# Definisi Fitur (X) dan Target (y)
1 fitur_kolom = [
    'Soil_Moisture', 'Ambient_Temperature',
    'Soil_Temperature', 'Humidity',
    'Light_Intensity', 'Soil_pH', 'Nitrogen_Level',
    'Phosphorus_Level', 'Potassium_Level',
    'Chlorophyll_Content', 'Electrochemical_Signal']
2 target_kolom = 'Plant_Health_Status'
3 X = df[fitur_kolom]
4 y = df[target_kolom]
```

c. Transformasi variabel target

Transformasi variabel target dari bentuk kategorikal ke numerik untuk menyesuaikan format data dengan kebutuhan algoritma pembelajaran mesin yang secara umum hanya dapat mengolah input numerik. Dengan mengonversi label seperti “*Healthy*”, “*Moderate Stress*”, dan “*High Stress*” agar proses pelatihan model menjadi lebih efisien dan terstruktur.

**Segmen Program 4.3 Transformasi variabel target**

```

# Encoding Target
1  le = LabelEncoder()
2  y_encoded = le.fit_transform(y)

# Untuk melihat mapping: print(le.classes_)
3  print(f"Kelas asli: {le.classes_}")
4  print(f"Kelas terencode: {np.unique(y_encoded)}")

```

Berikut hasil yang ditampilkan oleh program :

```
Kelas asli: ['Healthy' 'High Stress' 'Moderate Stress']
```

```
Kelas terencode: [0 1 2]
```

d. Pembagian data latih dan data uji

Pembagian dataset menggunakan fungsi *train\_test\_split* dari pustaka *scikit-learn* dengan parameter *test\_size=0.2* menunjukkan bahwa 20% data digunakan untuk validasi dan sisanya 80% untuk pelatihan. Parameter *stratify=y\_encoded* memastikan bahwa proporsi kelas dalam data target tetap seimbang pada kedua subset. Setelah pembagian, akan ditampilkan distribusi kelas (jumlah sampel per kategori status kesehatan tanaman) dalam masing-masing set. Hal ini bertujuan untuk memverifikasi bahwa pembagian data dilakukan secara adil dan representatif.

### Segmen Program 4.4 Pembagian data latih dan data uji

```

# Pembagian Data Training dan Validation (Stratified Split)
# test_size=0.2 berarti 20% untuk validasi, 80% untuk
training
1 X_train, X_val, y_train, y_val = train_test_split(X,
y_encoded, test_size=0.2, random_state=42,
stratify=y_encoded)

2 print(f"\nUkuran Training Set (X_train): {X_train.shape[0]}
sampel")
3 print(f"Ukuran Validation Set (X_val): {X_val.shape[0]}
sampel")

4 print("\nDistribusi kelas di Training set (setelah
encoding):")
unique_train, counts_train = np.unique(y_train,
5 return_counts=True)
print(pd.Series(counts_train,
index=le.inverse_transform(unique_train)).sort_index())

6 print("\nDistribusi kelas di Validation set (setelah
encoding):")
7 unique_val, counts_val = np.unique(y_val,
8 return_counts=True)
print(pd.Series(counts_val,
index=le.inverse_transform(unique_val)).sort_index())

```

Berikut hasil yang ditampilkan oleh program :

```

Ukuran Training Set (X_train): 960 sampel
Ukuran Validation Set (X_val): 240 sampel

```

```

Distribusi kelas di Training set (setelah encoding):
Healthy          239
High Stress      400
Moderate Stress  321
dtype: int64

```

```

Distribusi kelas di Validation set (setelah encoding):
Healthy          60
High Stress      100
Moderate Stress  80
dtype: int64

```

### Segmen Program 4.5 Membuat Visualisasi Distribusi Kelas

```

# Konversi array numpy hasil unique ke Pandas Series untuk
memudahkan plotting
1 train_counts = pd.Series(np.unique(y_train,
return_counts=True)[1],
2 index=le.inverse_transform(np.unique(y_train)))
val_counts = pd.Series(np.unique(y_val,
return_counts=True)[1],
index=le.inverse_transform(np.unique(y_val)))

# Gabungkan kedua Series ke dalam satu DataFrame untuk
plotting yang lebih mudah
3 dist_df = pd.DataFrame({
'Training Set': train_counts,
'Validation Set': val_counts
}).fillna(0)

# Pastikan urutan kelas konsisten untuk plotting
4 dist_df = dist_df.loc[le.classes_]

# Menyiapkan data untuk seaborn (format long-form)
5 plot_data = dist_df.stack().reset_index()
6 plot_data.columns = ['Health Status', 'Dataset', 'Count']

# Membuat dan menampilkan plot
7 plt.figure(figsize=(10, 6)) #Ukuran figure yang sedikit
lebih kecil jika ingin lebih ringkas
8 sns.barplot(x='Health Status', y='Count', hue='Dataset',
data=plot_data, palette='viridis')

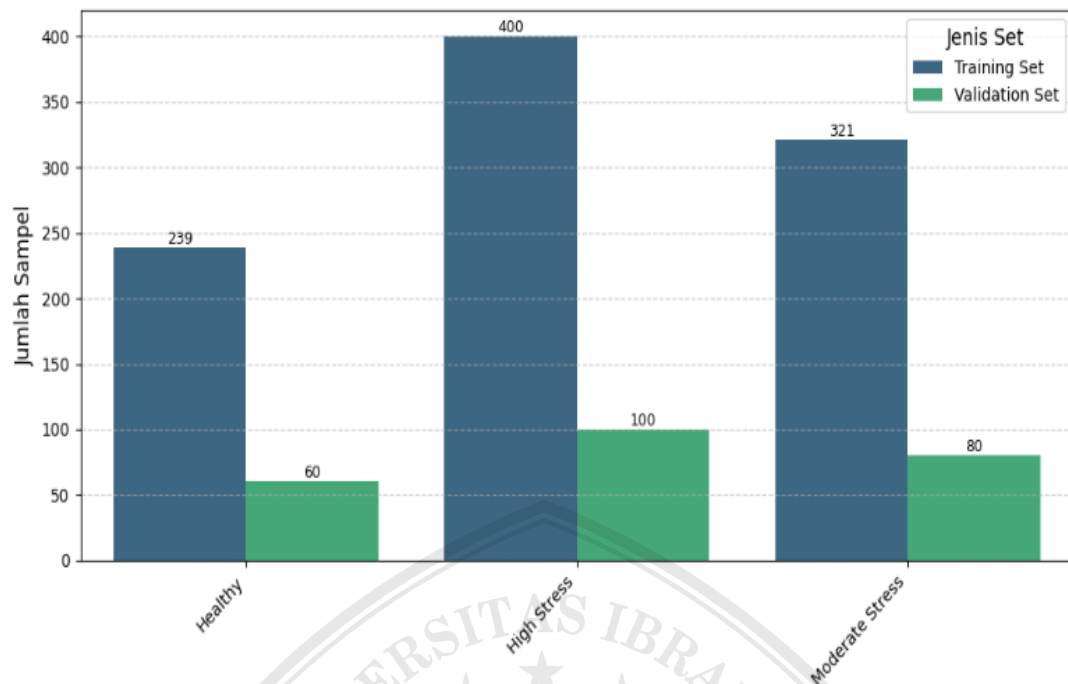
# Menambahkan label nilai di atas setiap bar (opsional, tapi
sangat direkomendasikan)
9 for container in plt.gca().containers:
10 plt.gca().bar_label(container, fmt='%d',
label_type='edge', fontsize=9)

11 plt.title('Distribusi Kelas pada Training dan Validation
Set', fontsize=14, fontweight='bold')
12 plt.xlabel('Status Kesehatan Tanaman', fontsize=12)
13 plt.ylabel('Jumlah Sampel', fontsize=12)
14 plt.xticks(rotation=45, ha='right', fontsize=10)
15 plt.yticks(fontsize=10)
16 plt.grid(axis='y', linestyle='--', alpha=0.7)
17 plt.legend(title='Jenis Set', fontsize=10,
18 title_fontsize=12, loc='upper right')
plt.tight_layout()
19 plt.show()

```

Kode tersebut digunakan untuk membuat visualisasi distribusi kelas pada data *training* dan *validation* dalam bentuk grafik batang (*bar plot*) menggunakan

*Pandas*, *Seaborn*, dan *Matplotlib*. Proses dimulai dengan menghitung jumlah data di setiap kelas menggunakan `np.unique()` pada variabel `y_train` dan `y_val`. Nilai label numerik kemudian dikonversi kembali ke bentuk teks aslinya menggunakan `le.inverse_transform()`, dan hasilnya disimpan dalam format *Pandas Series* agar mudah diolah. Selanjutnya, kedua *Series* tersebut digabungkan menjadi sebuah *DataFrame* dengan kolom “*Training Set*” dan “*Validation Set*”, di mana nilai kosong diisi dengan nol menggunakan `.fillna(0)`. *DataFrame* kemudian diurutkan kembali sesuai urutan kelas asli menggunakan `dist_df.loc[le.classes_]` untuk menjaga konsistensi tampilan. Agar dapat digunakan oleh fungsi `sns.barplot()`, *DataFrame* diubah ke format *long-form* menggunakan `.stack().reset_index()` sehingga menghasilkan tiga kolom utama, yaitu “*Health Status*” untuk label kelas, “*Dataset*” untuk jenis *set* (*Training/Validation*), dan “*Count*” untuk jumlah data. Visualisasi dibuat dengan `sns.barplot()` menggunakan parameter `hue` untuk membedakan warna antara *Training Set* dan *Validation Set*, serta palet warna “*viridis*” agar tampil lebih menarik. Label jumlah data juga ditambahkan di atas setiap batang menggunakan `bar_label()` untuk mempermudah pembacaan. Terakhir, pengaturan judul, label sumbu, rotasi teks, garis *grid*, legenda, dan tata letak dilakukan agar grafik lebih rapi dan mudah dipahami, sebelum akhirnya ditampilkan dengan `plt.show()`.



**Gambar 4.1** Distribusi Kelas pada *Training* dan *Validation Set*

Gambar tersebut menunjukkan distribusi jumlah sampel pada data *training* dan *validation* untuk masing-masing kategori status kesehatan tanaman. Terlihat bahwa kelas *High Stress* memiliki jumlah terbanyak, diikuti *Moderate Stress* dan *Healthy*, dengan proporsi yang konsisten antara kedua subset. Hal ini menunjukkan bahwa pembagian data menggunakan metode *stratified split* berhasil menjaga keseimbangan kelas, sehingga model dapat dilatih secara adil dan representatif tanpa bias terhadap kelas tertentu.

#### 4.2 Permodelan (Modeling)

Pemodelan dilakukan menggunakan algoritma *K-Nearest Neighbor* (K-NN) dan *Naïve Bayes* dari pustaka *Scikit-Learn*. Model dilatih dengan data latih menggunakan fungsi `.fit()` dan diuji pada data uji dengan `.predict()`. Evaluasi dilakukan melalui `accuracy_score` dan `classification_report`. Implementasi

permodelan menggunakan pustaka dinilai efisien dan akurat karena memanfaatkan fungsi siap pakai tanpa perlu perhitungan manual, sehingga cocok untuk sistem klasifikasi kesehatan tanaman berbasis data.

a. Algoritma *K-Nearest Neighbor*

1) Pemilihan Nilai K dan Perhitungan Jarak *Euclidean*

Pilih nilai K, misalnya K=5. Hitung jarak antara data yang akan diklasifikasi dengan data latih menggunakan rumus jarak *Euclidean* untuk menghitung jarak geometris antar titik berdasarkan fitur-fitur dalam dataset.

$$(X, Y) = \sqrt{\sum_{i=1}^n (X_i - Y_i)^2} \quad (5)$$

$X$  adalah data latih dan  $Y$  adalah data uji. Data uji akan ditentukan kedekatannya dengan menggunakan rumus *Euclidean Distance* tersebut, misalkan kelembapan tanah 29,896, suhu udara 25,749, suhu tanah 21,357, kelembapan udara 46,014, intensitas cahaya 795,427, pH tanah 7,081, kadar nitrogen 27,858, fosfor 11,412, dan kalium 45,996, kandungan klorofil 39,454, serta sinyal elektrokimia 1,170. Pada perhitungan jarak seperti *Manhattan Distance*, fungsi *absolute* berperan untuk memastikan selisih antara dua nilai fitur menjadi positif sebelum dikuadratkan, sehingga jarak yang dihitung tidak dipengaruhi oleh tanda (positif atau negatif) dari selisih tersebut. Sedangkan pada *Euclidean Distance*, fungsi *absolute* tidak lagi digunakan karena proses kuadrat otomatis membuat semua nilai menjadi positif.

Tabel 4.1 Tabel Data Latih dan Data Uji

Parameter	ID_1	ID_2	ID_n	Data Baru
<i>Soil_Moisture</i>	27,521	34,995	.....	29,896
<i>Ambient_Temperature</i>	22,240	20,224	.....	25,749
<i>Soil_Temperature</i>	21,9	19,522	.....	21,357
<i>Humidity</i>	55,292	66,29	.....	46,014
<i>Light_Intensity</i>	556,173	846,818	.....	795,427
<i>Soil_PH</i>	5,582	7,031	.....	7,081
<i>Nitrogen_level</i>	10,003	12,473	.....	27,858
<i>Phosporus_Level</i>	45,807	14,926	.....	11,412
<i>Potassium_Level</i>	39,076	44,973	.....	45,996
<i>Chlorophyll_Content</i>	35,703	49,788	.....	39,454
<i>Electrochemical_Signal</i>	0,941	0,963	.....	1,170
<i>Plant_Health_Status</i>	<i>High Stress</i>	<i>High Stress</i>	.....	--

$$\begin{aligned}
 id1: & \sqrt{(27,521 - 29,896)^2 + (22,240 - 25,749)^2 + (21,9 - 21,357)^2} \\
 & + (55,292 - 46,014)^2 + (556,173 - 795,427)^2 + (5,582 - 7,081)^2 \\
 & + (10,003 - 27,858)^2 + (45,807 - 11,412)^2 + (39,076 - 13,375)^2 \\
 & + (35,703 - 21,067)^2 + (0,941 - 1,170)^2 \\
 = & \sqrt{5,640 + 12,315 + 0,295 + 86,072 + 57242,806 +} \\
 & 2,248 + 1,984 + 1182,986 + 47,886 + 14,073 + 0,052} = 242,72
 \end{aligned}$$

$$\begin{aligned}
 id2: & \sqrt{(34,995 - 29,896)^2 + (20,224 - 25,749)^2 + (19,522 - 21,357)^2} \\
 & + (66,290 - 46,014)^2 + (846,818 - 795,427)^2 + (7,031 - 7,081)^2 \\
 & + (12,473 - 27,858)^2 + (14,926 - 11,412)^2 + (44,973 - 13,375)^2 \\
 & + (49,788 - 21,067)^2 + (0,963 - 1,170)^2 \\
 = & \sqrt{26 + 30,525 + 3,367 + 411,116 + 2641,035} \\
 & + 0,002 + 236,698 + 12,607 + 1,046 +} \\
 & 106,793 + 0,043} = 58,9
 \end{aligned}$$

Tabel 4.2 Hasil Perhitungan ED

<i>Timestamp</i>	<i>ID</i>	<i>Soil_Moist</i>	<i>Ambient_Temp</i>	<i>Soil_Temp</i>	<i>Humidity</i>	...	<i>Plant_Health_Status</i>	<i>ED</i>
2024-10-03	1	27,521	22,240	21,900	55,292	...	High Stress	242,72
2024-10-03	2	34,995	20,225	19,522	66,291	...	High Stress	58,90
2024-10-03	3	31,901	18,408	15,375	45,282	...	Healthy	323,70
2024-10-03	4	32,516	26,853	23,113	47,474	...	Healthy	57,60
2024-10-03	5	13,653	22,662	20,980	54,418	...	High Stress	27,95
2024-10-03	6	18,746	27,642	22,679	43,860	...	High Stress	585,27
2024-10-03	7	21,518	22,361	19,834	54,203	...	Moderate Stress	380,36
2024-10-03	8	16,781	26,204	24,584	54,868	...	High Stress	222,08
2024-10-03	9	23,220	28,751	19,808	69,087	...	Moderate Stress	118,33
2024-10-03	10	25,123	22,063	24,395	58,781	...	Moderate Stress	145,06

2) Penentuan *K-Neighbors* dan Klasifikasi Kelas

Setelah seluruh jarak tiap data dihitung, akan diurutkan nilai jarak dari yang terkecil hingga terbesar untuk menentukan *K-Neighbors* (tetangga terdekat) dan ditentukan label berdasarkan mayoritas kelas dari  $K=5$  tetangga terdekat.

Tabel 4.3 Hasil Sortir Perhitungan ED dengan  $K=5$ 

<i>Timestamp</i>	<i>ID</i>	<i>Soil_Moist</i>	<i>Ambient_Temp</i>	<i>Soil_Temp</i>	<i>Humidity</i>	...	<i>Plant_Health_Status</i>	<i>ED</i>
2024-10-13	9	29,896	25,749	21,357	46,014	...	<i>Moderate Stress</i>	0,00
2024-10-03	4	25,161	21,053	23,899	43,214	...	<i>Moderate Stress</i>	18,61
2024-10-23	4	27,57	25,273	15,848	45,641	...	<i>High Stress</i>	25,33
2024-10-06	7	37,6	18,632	23,252	44,875	...	<i>Healthy</i>	25,61
2024-10-20	6	21,532	22,385	24,026	42,018	...	<i>Moderate Stress</i>	25,82

### 3) Implementasi KNN dengan pustaka *Scikit-Learn*

#### Segmen Program 4.6 Implementasi Model KNN

```

1  # Inisialisasi model KNN (contoh dengan k=5)
   knn_model = KNeighborsClassifier(n_neighbors=5)
2  # Latih model Menggunakan X_train dan X_val secara langsung
   karena scaling dilewati
   knn_model.fit(X_train, y_train)
3  # Klasifikasi pada data uji
   y_pred_knn = knn_model.predict(X_val)
4  # Evaluasi kinerja KNN and classification report
   print("Accuracy KNN: ", accuracy_score(y_val,
   y_pred_knn))
5  print("Classification Report KNN:")
6  print(classification_report(y_val, y_pred_knn,
   target_names=le.classes_))
7  cm_knn = confusion_matrix(y_val, y_pred_knn)

   # Menampilkan Laporan Klasifikasi
8  plt.figure(figsize=(7, 6))
9  sns.heatmap(cm_knn, annot=True, fmt='d', cmap='Blues',
   xticklabels=le.classes_, yticklabels=le.classes_)
10 plt.title('Confusion Matrix KNN')
11 plt.xlabel('Klasifikasi')
12 plt.ylabel('Aktual')
13 plt.show()

```

Pada tahap ini, algoritma *K-Nearest Neighbors* (KNN) diimplementasikan menggunakan pustaka *Scikit-Learn* tanpa perhitungan manual. Model diinisialisasi dengan `n_neighbors=5`, yang berarti klasifikasi kelas ditentukan berdasarkan lima tetangga terdekat. Nilai `k` ini dipilih sebagai titik awal sebelum dilakukan pengujian nilai lain pada tahap *tuning*. Proses pelatihan dilakukan dengan `.fit()` menggunakan `X_train` sebagai masukan fitur dan `y_train` sebagai label target. *Feature scaling* dilewati karena dataset memiliki skala fitur yang sebanding. Setelah pelatihan, klasifikasi terhadap data validasi (`X_val`) dilakukan menggunakan `.predict()`, menghasilkan label klasifikasi (`y_pred_knn`). Evaluasi kinerja model dilakukan dengan menghitung akurasi menggunakan `accuracy_score` dan menampilkan

*classification report* yang memuat *precision*, *recall*, dan *F1-score* untuk setiap kelas. Selanjutnya, dibuat *confusion matrix* dengan *confusion\_matrix* dan divisualisasikan sebagai heatmap menggunakan *Seaborn*. Visualisasi ini memudahkan analisis kesalahan klasifikasi dengan melihat distribusi klasifikasi benar dan salah pada tiap kelas.

Berikut hasil yang ditampilkan oleh program :

```
Accuracy KNN: 0.5666666666666667
Classification Report KNN:
              precision    recall  f1-score   support

   Healthy          0.67      0.57      0.61         60
  High Stress          0.66      0.67      0.67        100
 Moderate Stress      0.40      0.44      0.42         80

 accuracy                   0.57         240
 macro avg          0.58      0.56      0.57         240
 weighted avg       0.58      0.57      0.57         240
```

#### b. Algoritma *Naïve Bayes*

##### 1) Perhitungan Probabilitas Prior

Hitung probabilitas prior untuk setiap kelas berdasarkan distribusi data latih.

Probabilitas prior dihitung sebagaimana contoh berikut.

$$P(C"health") = \frac{\text{Jumlah data kelas health (299)}}{\text{Total data (1200 data)}} \quad (6)$$

$P(C)$  merepresentasikan peluang awal data pada seluruh kelas *health*, *moderate*, dan *high*. Sehingga didapatkan hasil

- $P(C) \text{ Health} = \frac{299}{1200} = 0,25$
- $P(C) \text{ Moderate Stress} = \frac{402}{1200} = 0,33$

- $P(C) \text{ High Stress} = \frac{500}{1200} = 0,42$

## 2) Perhitungan *Likelihood* untuk Setiap Fitur

Untuk setiap fitur dalam dataset, hitung probabilitas *likelihood* berdasarkan distribusi fitur tersebut dalam setiap kelas. Misalnya, untuk data kontinu, digunakan distribusi *Gaussian* dengan rumus:

$$P(X|C) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right) \quad (7)$$

$$P(\text{soil moist}|\text{health}) = \frac{1}{\sqrt{2 * 3.14 * 9.2221}} * \exp\left(-\frac{(29,896 - 25.0093)^2}{2 * 9.2221}\right)$$

$$= 0,131369862 * 0,273978868 = 0,0360$$

Berdasarkan perhitungan tersebut terhadap seluruh fitur dan kelas data yang ada, maka diperoleh hasil perhitungan sebagaimana berikut.

**Tabel 4.4 Hasil Perhitungan P(X|C) / *Likelihood***

FITUR	Y	HEALTH	MODERATE	HIGH
<i>Soil_Moisture</i>	29,90	0,0360	0,0495	0,0447
<i>Ambient_Temperature</i>	25,75	0,0997	0,1037	0,1015
<i>Soil_Temperature</i>	21,36	0,1189	0,1187	0,1253
<i>Humidity</i>	46,01	0,0274	0,0274	0,0273
<i>Light_Intensity</i>	795,43	0,0013	0,0013	0,0013
<i>Soil_pH</i>	7,08	0,4316	0,4345	0,4345
<i>Nitrogen_Level</i>	27,86	0,0436	0,0364	0,0314
<i>Phosphorus_Level</i>	11,41	0,0088	0,0092	0,0089
<i>Potassium_Level</i>	46,00	0,0136	0,0133	0,0137
<i>Chlorophyll_Content</i>	39,45	0,0394	0,0392	0,0395
<i>Electrochemical_Signal</i>	1,17	0,6708	0,6686	0,6456

### 3) Perhitungan Probabilitas Posterior

Seluruh *likelihood* dan probabilitas prior dikalikan menggunakan teorema

*Bayes* :

$$P(C|X) = \frac{p(X|C) * P(C)}{P(X)} \quad (8)$$

Sehingga diperoleh nilai probabilitas dari masing-masing perkalian sebagaimana berikut:

- *Health* = 2,21053E-16
- *Moderate* = 0,00000E+00
- *High* = 2,50498E-14

Karena  $P(X)$  adalah konstan untuk semua kelas, maka fokus perhitungan berada pada pembilang  $P(C)*P(X|C)$ . Data yang dihitung mendekati label *High Stress* dengan skor probabilitas tertinggi.

### 4) Implementasi dengan pustaka *Scikit-Learn*

#### Segmen Program 4.7 Implementasi Model *Naïve Bayes*

```

1 #implemantasi model naive bayes
2 gnb_model = GaussianNB()
3 gnb_model.fit(X_train, y_train)
4 y_pred_gnb = gnb_model.predict(X_val)
5 print("Accuracy Naive Bayes: ",
6 accuracy_score(y_val, y_pred_gnb))
7 print("Classification Report Naive Bayes:")
8 print(classification_report(y_val, y_pred_gnb,
9 target_names=le.classes_))

```

Berikut hasil yang ditampilkan oleh program :

```
Accuracy Naive Bayes: 0.7625
Classification Report Naive Bayes:
              precision    recall  f1-score   support

   Healthy           0.77       0.78       0.78         60
  High Stress        0.83       0.79       0.81        100
 Moderate Stress     0.68       0.71       0.70         80

 accuracy                   0.76         240
 macro avg           0.76       0.76       0.76         240
 weighted avg        0.77       0.76       0.76         240
```

#### Segmen Program 4.8 Menampilkan *Confusion Matrix Naïve Bayes*

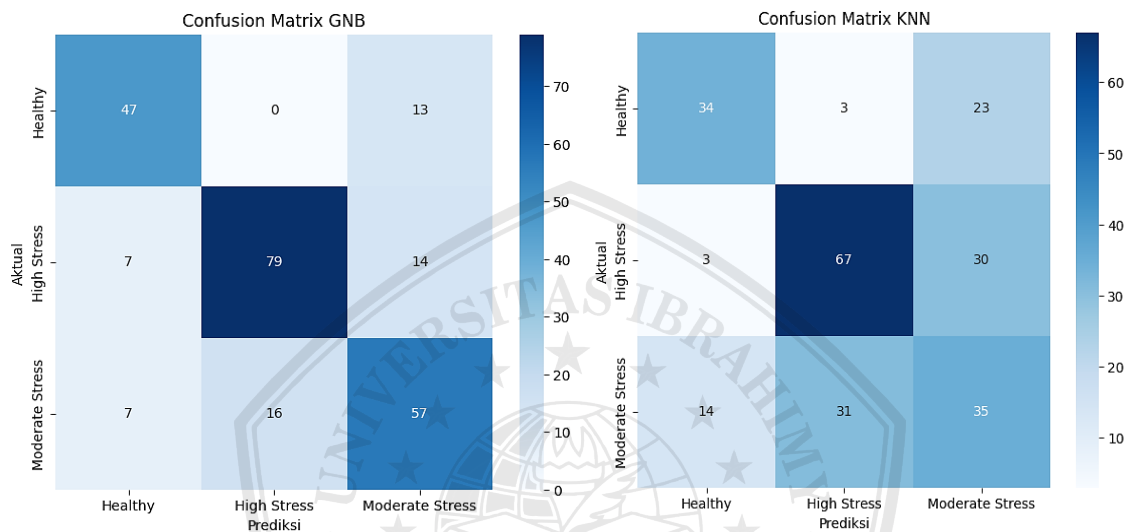
```
1 cm_naive = confusion_matrix(y_val, y_pred_gnb)
2 plt.figure(figsize=(7, 6))
3 sns.heatmap(cm_naive, annot=True, fmt='d',
4             cmap='Blues',
5             xticklabels=le.classes_,
6             yticklabels=le.classes_)
7 plt.title('Confusion Matrix GNB')
8 plt.xlabel('Klasifikasi')
9 plt.ylabel('Aktual')
10 plt.show()
```

### 4.3 Evaluasi (Evaluation)

Tahap *Evaluation* (Evaluasi) bertujuan untuk menilai dan membandingkan kinerja model *K-Nearest Neighbors* (KNN) dan *Naïve Bayes* secara objektif. Pada tahap ini, digunakan metode uji statistik *T-Test* untuk mengukur apakah terdapat perbedaan yang signifikan secara statistik antara performa kedua model berdasarkan hasil klasifikasi yang telah dihasilkan. Proses evaluasi ini memastikan bahwa perbandingan tidak hanya didasarkan pada nilai akurasi semata, tetapi juga mempertimbangkan signifikansi perbedaan performa, sehingga pemilihan model terbaik dapat dilakukan secara lebih tepat dan terukur. Berikut metrik evaluasi yang akan dibandingkan dalam penelitian ini.

a. *Confusion matrix*

*Confusion matrix* menyajikan perbandingan antara label sebenarnya dan label klasifikasi dalam bentuk tabel. Metrik ini memberikan detail mengenai jenis kesalahan (*false positive* dan *false negative*) yang dilakukan model.



**Gambar 4.2 Confusion Matrix KNN dan *Naïve Bayes***

b. Akurasi (*accuracy*)

Akurasi merupakan metrik evaluasi yang mengukur proporsi jumlah klasifikasi yang benar terhadap keseluruhan data uji yang digunakan. Nilai akurasi diperoleh dengan membagi jumlah klasifikasi benar (baik positif maupun negatif) dengan total seluruh klasifikasi yang dilakukan. Metrik ini sesuai digunakan ketika distribusi kelas dalam dataset seimbang atau tidak terdapat perbedaan yang signifikan antara jumlah data pada masing-masing kelas, sehingga hasil pengukuran tidak bias terhadap kelas mayoritas.

Rumus:

$$\text{Akurasi} = \frac{TP_{total}}{TP + TN + FP + FN} \quad (9)$$

Keterangan:

- TP (*True Positive*): Klasifikasi positif yang benar
- TN (*True Negative*): Klasifikasi negatif yang benar
- FP (*False Positive*): Klasifikasi positif yang salah
- FN (*False Negative*): Klasifikasi negatif yang salah

**Tabel 4.5 Tabel Hasil Pengujian Data**

Kelas	TP	FP	FN
<i>Naïve Bayes</i>			
<i>Healthy</i>	47	14	13
<i>High Stress</i>	79	16	21
<i>Moderate Stress</i>	57	27	23
<i>K-Nearest Neighbor</i>			
<i>Healthy</i>	34	17	26
<i>High Stress</i>	67	34	33
<i>Moderate Stress</i>	35	53	45

**Akurasi *Naïve Bayes*:**

$$\text{Total} = 47 + 0 + 13 + 7 + 79 + 14 + 7 + 16 + 57 = 240$$

$$\text{Akurasi} = \frac{47 + 79 + 57}{240} = \frac{183}{240} = \mathbf{0.7625}$$

**Akurasi *K-Nearest Neighbor*:**

$$\text{Total} = 34 + 3 + 23 + 3 + 67 + 30 + 14 + 31 + 35 = 240$$

$$\text{Akurasi} = \frac{34 + 67 + 35}{240} = \frac{136}{240} = \mathbf{0.5667}$$

c. Presisi (*precision*)

*Precision* mengukur seberapa banyak klasifikasi positif yang benar dari semua yang diklasifikasi positif. Misalkan jika model mengklasifikasikan 10 tanaman

sebagai sakit, dan hanya 7 yang benar-benar sakit, maka  $precision = 7/10 = 0.7$ .

Artinya, model cukup tepat dalam menyatakan tanaman sakit.

Rumus:

$$Precision = \frac{TP}{TP + FP} \quad (10)$$

#### **Precision Naïve Bayes**

$$Healthy = 47 / (47 + 14) = 0.770$$

$$High Stress = 79 / (79 + 16) = 0.832$$

$$Moderate = 57 / (57 + 27) = 0.679$$

#### **Precision KNN**

$$Healthy = 34 / (34 + 17) = 0.667$$

$$High Stress = 67 / (67 + 34) = 0.663$$

$$Moderate = 35 / (35 + 53) = 0.398$$

#### d. *Recall (sensitivity)*

*Recall* mengukur seberapa banyak kasus positif yang berhasil ditemukan oleh model dari seluruh kasus positif yang ada. Misalkan jika ada 10 tanaman yang benar-benar sakit, dan model hanya berhasil menemukan 7, maka  $recall = 7/10 = 0.7$ . Artinya, model berhasil menemukan 70% dari total tanaman yang sakit.

$$Recall = \frac{TP}{TP + FN} \quad (11)$$

#### **Recall Naïve Bayes**

$$Healthy = 47 / (47 + 13) = 0.783$$

$$High Stress = 79 / (79 + 21) = 0.790$$

$$Moderate = 57 / (57 + 23) = 0.712$$

#### **Recall KNN**

$$Healthy = 34 / (34 + 26) = 0.567$$

$$High Stress = 67 / (67 + 33) = 0.670$$

$$Moderate = 35 / (35 + 45) = 0.438$$

e. *F1-score*

*F1-score* adalah rata-rata harmonis antara *precision* dan *recall*. Berguna ketika terdapat ketidakseimbangan kelas dan kita ingin mempertimbangkan *trade-off* antara presisi dan *recall*.

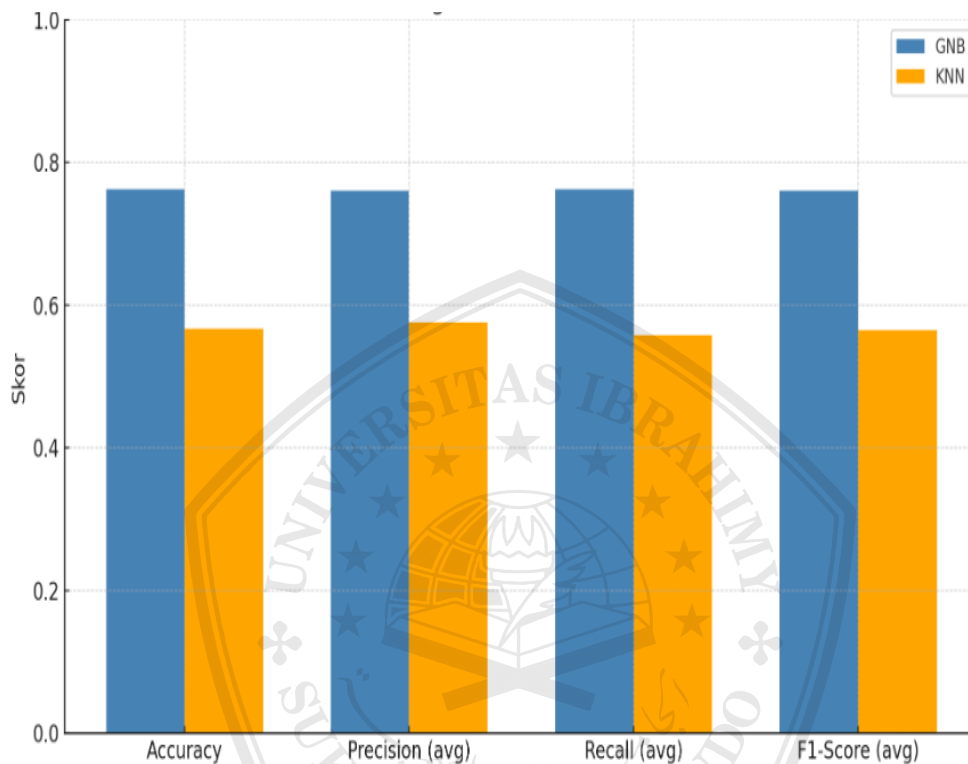
$$F1 - Score = 2 \times \frac{Precision * Recall}{Precision + Recall} \quad (12)$$

***F1-score Naïve Bayes****Healthy* = 0.777*High Stress* = 0.810*Moderate* = 0.695***F1-Score KNN****Healthy* = 0.613*High Stress* = 0.667*Moderate* = 0.417**Tabel 4.6 Tabel Perbandingan Kinerja Algoritma**

Metrik	GNB	KNN
<i>Accuracy</i>	0.7625	0.5292
<i>Precision (avg)</i>	0.7602	0.5759
<i>Recall (avg)</i>	0.7619	0.5581
<i>F1-Score (avg)</i>	0.7607	0.5953

Berdasarkan hasil evaluasi, algoritma *Gaussian Naive Bayes* (GNB) terbukti lebih unggul dibandingkan *K-Nearest Neighbors* (KNN) pada seluruh metrik yang diuji. GNB memperoleh akurasi sebesar 0,7625 (76,25%), lebih tinggi dibanding KNN yang hanya mencapai 0,5292 (52,92%). Pada *precision* rata-rata, GNB mencatat 0,7602 (76,02%), sedangkan KNN hanya 0,5759 (57,59%). Dari segi *recall*, GNB meraih 0,7619 (76,19%) dan KNN 0,5581 (55,81%). Sementara itu, *F1-Score* rata-rata GNB sebesar 0,7607 (76,07%) juga lebih baik dibanding KNN

yang hanya 0,5953 (59,53%). Secara keseluruhan, GNB lebih konsisten dan akurat dalam menghasilkan klasifikasi dibandingkan KNN pada dataset ini.



**Gambar 4.3 Perbandingan Metrik Kinerja Algoritma**

Selanjutnya akan dilakukan pengujian *t-test* berpasangan dengan tujuan untuk mengetahui apakah terdapat perbedaan yang signifikan secara statistik antara kinerja model KNN dan *Naïve Bayes*. Pengujian ini penting untuk memastikan bahwa perbedaan hasil klasifikasi antara kedua model bukan terjadi secara kebetulan, melainkan benar-benar mencerminkan perbedaan performa yang nyata. Dengan membandingkan rata-rata klasifikasi benar dari kedua model pada data yang sama, *t-test* dapat memberikan dasar statistik yang kuat dalam menarik kesimpulan mengenai keunggulan salah satu model secara objektif.

### Segmen Program 4.9 Uji-T Berpasangan (*Paired T-Test*)

```

1 print("\n--- 📄 **Hasil Uji-T Berpasangan** ---")
  alpha = 0.05

2 t_statistic,      p_value      =      stats.ttest_rel(correct_knn,
  correct_gnb)

  # Buat DataFrame untuk hasil T-test
3 ttest_results = pd.DataFrame({
  'Metrik Statistik': ['Statistik T', 'Nilai P',
  'Tingkat Signifikansi (Alpha)'],
  'Nilai': [f"{t_statistic:.4f}", f"{p_value:.7f}",
  f"{alpha:.2f}"]
  })

4 print(ttest_results.to_markdown(index=False))

5 alpha = 0.05

6 print("\n--- **Kesimpulan Uji-T Berpasangan** ---")
7 if p_value < alpha:
8 print("✅ **Tolak Hipotesis Nol (H0).**")
9 print("Terdapat **perbedaan signifikan secara statistik**
  antara kinerja KNN dan Naive Bayes.")

10 else:
11 print("❌ **Gagal menolak Hipotesis Nol (H0).**")
12 print("Tidak ada perbedaan signifikan secara statistik antara
  kinerja KNN dan Naive Bayes.")

```

Uji-T berpasangan dilakukan untuk membandingkan performa dua model klasifikasi, yaitu KNN dan *Naive Bayes* dalam hal jumlah klasifikasi yang benar terhadap label yang sama. Fungsi `ttest_rel()` dari pustaka `scipy.stats` digunakan untuk menghitung nilai statistik T dan nilai P (*p-value*).

Berikut hasil yang ditampilkan oleh program :

```

--- 📄 **Hasil Uji-T Berpasangan** ---
| Metrik Statistik          | Nilai |
|:-----:|:-----:|
| Statistik T              | -5.0636 |
| Nilai P                  | 8e-07 |
| Tingkat Signifikansi (Alpha) | 0.05 |

--- **Kesimpulan Uji-T Berpasangan** ---
✅ **Tolak Hipotesis Nol (H0).**

Terdapat **perbedaan signifikan secara statistik** antara
kinerja KNN dan Naive Bayes.

```

*P-Value* adalah nilai yang menunjukkan probabilitas untuk memperoleh perbedaan hasil yang sama atau lebih ekstrem dari yang diamati apabila pada kenyataannya tidak terdapat perbedaan di populasi. Dalam uji-t, pengujian diawali dengan Hipotesis Nol ( $H_0$ ) yang menyatakan bahwa tidak ada perbedaan kinerja antara KNN dan *Naive Bayes*. Nilai *P* digunakan untuk menentukan apakah data yang diperoleh cukup kuat untuk menolak  $H_0$ . Nilai *P* yang kecil menunjukkan bahwa kemungkinan perbedaan tersebut terjadi karena faktor acak sangat rendah, sedangkan nilai yang besar mengindikasikan sebaliknya. Pada hasil pengujian ini, nilai *P* sebesar 0.0000008 ( $8e-07$ ) jauh di bawah tingkat signifikansi (*alpha*) 0.05, sehingga  $H_0$  ditolak dan disimpulkan bahwa perbedaan kinerja KNN dan *Naive Bayes* signifikan secara statistik.

#### 4.4 Penyebaran (Deployment)

Deployment merupakan tahap akhir dari proses CRISP-DM yang berfokus pada implementasi model ke dalam sistem nyata, agar hasil klasifikasi dapat dimanfaatkan oleh pengguna akhir.

##### Segmen Program 4.10 Menyimpan Model dalam Bentuk *Pickle*

```
1 #saya ingin menyimpan model knn dan naive bayes
2 import pickle
3 pickle.dump(knn_model, open("knn_model.pkl", "wb"))
4 pickle.dump(gnb_model, open("gnb_model.pkl", "wb"))
5
6 #implementasi model
7 knn_model = pickle.load(open("knn_model.pkl", "rb"))
8 gnb_model = pickle.load(open("gnb_model.pkl", "rb"))
```

Dalam penelitian ini, deployment dilakukan dengan menyimpan model yang telah dilatih menggunakan algoritma *K-Nearest Neighbor* dan *Naive Bayes* dalam bentuk file *.pkl* (*pickle file*) menggunakan pustaka *Python pickle*. File model ini

kemudian dapat digunakan kembali untuk mengklasifikasikan data baru tanpa perlu melatih ulang model dari awal.

#### Segmen Program 4.11 Menyimpan Model dalam Bentuk *Pickle*

```

1  #load file testing
   df_test = pd.read_csv(mypath +
2  '/Dataset/test_plant_data.csv')
   df_test.head(10)

   #implementasi model
3  y_pred_knn = knn_model.predict(X_test)
4  y_pred_gnb = gnb_model.predict(X_test)

5  kelas_asli = ['Healthy' 'High Stress' 'Moderate Stress']
   # rubah klasifikasi menjadi kelas asli
6  y_pred_knn = le.inverse_transform(y_pred_knn)
   y_pred_gnb = le.inverse_transform(y_pred_gnb)

7  df_test["Predicted_Status_KNN"] = y_pred_knn
8  df_test["Predicted_Status_GNB"] = y_pred_gnb

9  output_excel_path = mypath + '/result_test_plant_data.xlsx'
10 df_test.to_excel(output_excel_path, index=False)

```

Setelah proses pelatihan dan evaluasi selesai, dilakukan pengujian model terhadap data uji (*test\_plant\_data.csv*). Model yang telah disimpan dipanggil kembali dan digunakan untuk mengklasifikasikan status kesehatan tanaman berdasarkan fitur-fitur sensor seperti kelembaban tanah, suhu, dan kadar unsur hara. Hasil klasifikasi dari kedua algoritma ditambahkan ke dalam dataset dan diekspor ke file *.xlsx* baru sebagai laporan akhir yang bisa dianalisis lebih lanjut atau digunakan dalam sistem pemantauan pertanian berbasis data.

## BAB V

### PENUTUP

#### 5.1 Kesimpulan

Pada bagian latar belakang telah dijelaskan bahwa penelitian ini menggunakan dua algoritma pembandingan, yaitu *K-Nearest Neighbor* (KNN) dan *Naïve Bayes* (NB). Keduanya dipilih karena sama-sama sederhana, kuat, dan sering digunakan sebagai *baseline* dalam penelitian klasifikasi. KNN dikenal efektif untuk data numerik, mudah diimplementasikan, serta tangguh terhadap data besar dan kompleks dengan mekanisme perhitungan jarak *Euclidean*. Sementara itu, *Naïve Bayes* unggul dalam kecepatan, efisiensi pada data berukuran besar, serta tetap mampu memberikan hasil baik meski data latih terbatas karena berbasis probabilistik dan asumsi independensi antar fitur.

Berdasarkan literatur terdahulu yang dikaji dalam penelitian ini, mayoritas penelitian menunjukkan bahwa KNN lebih unggul dibandingkan *Naïve Bayes*. Misalnya, penelitian oleh Putu Prianka Vedanty (2024) mengenai identifikasi penyakit daun tanaman obat menunjukkan bahwa KNN mencapai akurasi 89%, sedangkan *Naïve Bayes* hanya 57,67%. Demikian juga penelitian oleh Yulia Resti dkk. (2022) pada deteksi penyakit dan hama tanaman jagung, di mana KNN memperoleh akurasi sangat tinggi, yaitu 99,54%, lebih unggul dibandingkan *Multinomial Naïve Bayes* yang hanya 92,72%. Namun, ada juga penelitian yang hasilnya berbeda, seperti penelitian Fingki Junaidi Manurung dkk. (2024) pada klasifikasi tanaman herbal, yang menunjukkan bahwa *Naïve Bayes* lebih unggul dengan akurasi 92,86% dibandingkan KNN yang hanya 82,14%.

Menariknya, hasil penelitian ini justru berbeda dengan mayoritas penelitian sebelumnya. *Naïve Bayes* terbukti lebih unggul dengan akurasi 76,25%, sementara KNN hanya 56,67%. NB juga menunjukkan konsistensi performa pada semua kelas kesehatan tanaman, khususnya kelas *High Stress* dengan F1-score tertinggi (0,81). Sebaliknya, KNN gagal membedakan kelas “*Moderate Stress*” dengan baik (F1-score hanya 0,42), yang kemungkinan besar disebabkan oleh *overlap* antar fitur dalam dataset. Hal ini memperlihatkan bahwa keunggulan algoritma dapat sangat bergantung pada jenis data yang digunakan.

Dengan demikian, penelitian ini memberikan kontribusi baru dengan menunjukkan bahwa dalam konteks data sensor numerik kesehatan tanaman, *Naïve Bayes* lebih efektif dibandingkan KNN. Hal ini memperkaya literatur yang sebelumnya didominasi oleh penelitian berbasis citra dan memperkuat argumen bahwa pemilihan algoritma *machine learning* harus disesuaikan dengan karakteristik dataset, bukan hanya berdasarkan tren umum di penelitian terdahulu.

**Tabel 5.1** Tabel Perbandingan Kinerja KNN vs *Naïve Bayes*

Metrik Evaluasi	<i>K-Nearest Neighbor</i> (KNN)	<i>Naïve Bayes</i> (GNB)
Akurasi	56,67%	76,25%
<i>Precision</i> (avg)	57,59%	76,02%
<i>Recall</i> (avg)	55,81%	76,19%
F1-Score (avg)	59,53%	76,07%
Klasifikasi Tepat	136 dari 240 data	183 dari 240 data
Kelas Terbaik	High Stress (F1: 0.67)	High Stress (F1: 0.81)
Kelas Terlemah	Moderate Stress (F1: 0.42)	Moderate Stress (F1: 0.70)

Berdasarkan hasil evaluasi model, algoritma *Naïve Bayes* (GNB) menunjukkan performa yang lebih unggul dibandingkan *K-Nearest Neighbor* (KNN) dalam mengklasifikasikan status kesehatan tanaman. *Naïve Bayes* berhasil mengklasifikasikan 183 sampel dengan benar, dengan akurasi yang tinggi pada setiap kelas: 47 untuk "*Healthy*", 79 untuk "*High Stress*", dan 57 untuk "*Moderate Stress*". *Naïve Bayes* mencapai akurasi sebesar 76,25%, dengan nilai *f1-score* tertinggi pada kelas *High Stress* (0.81), dan performa yang baik pada kelas *Healthy* (*f1-score* 0.78) serta *Moderate Stress* (*f1-score* 0.70). KNN hanya mengklasifikasikan 136 sampel dengan benar (34 "*Healthy*", 67 "*High Stress*", dan 35 "*Moderate Stress*"), dengan nilai akurasi keseluruhan sebesar 56,67%. Berdasarkan *classification report* KNN, kelas "*High Stress*" memiliki performa terbaik dengan *precision* 0.66, *recall* 0.67, dan *f1-score* 0.67, sedangkan kelas "*Moderate Stress*" memiliki performa terendah. Hal ini menunjukkan bahwa KNN kesulitan saat menghadapi *overlap* antar kelas dan cenderung kurang efektif dalam membedakan kelas yang memiliki distribusi fitur yang tumpang tindih. Sedangkan NB lebih mampu menangani data numerik dengan distribusi yang bervariasi dan fitur-fitur yang relatif independen.

Dengan demikian, penelitian ini berhasil menunjukkan bahwa algoritma *Naïve Bayes* lebih cocok digunakan untuk klasifikasi kesehatan tanaman berbasis data sensor dalam skenario ini, dan dapat dijadikan dasar pengembangan sistem klasifikasi pertanian cerdas yang efisien, praktis, dan berbasis data.

## 5.2 Saran

Program perbandingan algoritma KNN dan *Naïve Bayes* dalam pengamatan kesehatan tanaman ini masih memiliki banyak keterbatasan, diantaranya:

- 1) Belum diimplementasikan dalam sistem monitoring pertanian berbasis aplikasi atau IoT
- 2) Perlu dikembangkan dengan menambahkan parameter lain seperti curah hujan atau citra tanaman
- 3) Disarankan membandingkan algoritma lanjutan seperti *Random Forest* atau SVM untuk hasil yang lebih optimal.

Oleh karena itu, program ini diharapkan dapat terus dikembangkan secara lebih luas dan menyeluruh, baik dari segi perancangan antarmuka, fungsionalitas sistem, maupun akurasi perhitungan agar mampu memberikan hasil yang lebih optimal dan bermanfaat dalam penerapannya.

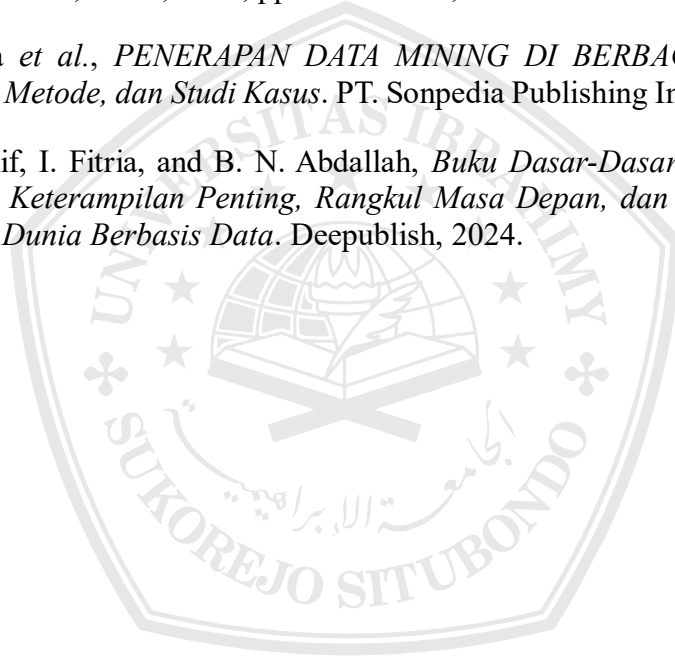
## DAFTAR PUSTAKA

- [1] F. and A. Organization, "Climate change fans spread of pests and threatens plants and crops, new FAO study Pests destroy up to 40 percent of global crops and cost \$220 billion of losses." Accessed: Feb. 02, 2025. [Online]. Available: <https://www.fao.org/newsroom/detail/Climate-change-fans-spread-of-pests-and-threatens-plants-and-crops-new-FAO-study>
- [2] S. T. Nasrul Zaman, *PUBLIC HEALTH FOR SUSTAINABLE DEVELOPMENT: Keadilan, Tantangan dan Kolaborasi Masyarakat Global*. Feniks Muda Sejahtera, 2025.
- [3] A. L. Abadi, M. A. Syib'li, L. Q. Aini, A. W. Sektiono, F. A. Choliq, and I. Trianti, *Pengelolaan Penyakit Tumbuhan Terpadu (Integrated Plant Disease Management)*. Universitas Brawijaya Press, 2023.
- [4] S. Junaidi *et al.*, *Buku Ajar Machine Learning*. PT. Sonpedia Publishing Indonesia, 2024.
- [5] P. W. Rahayu *et al.*, *Buku Ajar Data Mining*. PT. Sonpedia Publishing Indonesia, 2024.
- [6] R. M. Sari, V. Tasril, S. Wahyuni, and S. E. Putri, *Klasifikasi Forecasting Menggunakan Algoritma Naive Bayes*. Serasi Media Teknologi, 2024.
- [7] M. R. Haditama, "Analisis dan pembuatan dashboard prediksi kelulusan mahasiswa menggunakan metode random forest, naïve bayes dan support vector machine," 2023, *Fakultas Sains dan Teknologi UIN Syarif Hidayatullah Jakarta*.
- [8] M. Huda and M. Kom, *Algoritma Data Mining: Analisis Data Dengan Komputer*. bisakimia, 2019.
- [9] G. A. P. Febriyanti and A. Baita, "Comparison of Support Vector Machine and Decision Tree Algorithm Performance with Undersampling Approach in Predicting Heart Disease Based on Lifestyle," *J. Appl. Informatics Comput.*, vol. 9, no. 2, pp. 318–327, 2025.
- [10] I. Maulita and A. Wahid, "Prediksi Magnitudo Gempa Menggunakan Random Forest, Support Vector Regression, XGBoost, LightGBM, dan Multi-Layer Perceptron Berdasarkan Data Kedalaman dan Geolokasi (Predicting Earthquake Magnitude Using Random Forest, Support Vector Regression, XGBoost)," *J. Pendidik. Dan Teknol. Indones.*, vol. 4, pp. 221–232, 2024.

- [11] N. Rahayu, “Aggarwal, CC (2018). Artificial Neural Network and Deep Learning. Springer. Anderson, JA (1995). An Introduction to Neural Networks. The MIT Press. Bahdanau, D., Cho, K., & Bengio, Y.(2015). Neural Machine Translation,” *Deep Learn. Teor. Algoritm. dan Apl.*, vol. 9, p. 56, 2025.
- [12] N. A. Permana and H. Bunyamin, “Perbandingan Logistic Regression dengan Random Forest dalam Memprediksi Sentimen Pada IMDb Movie Review,” *J. Strateg. Maranatha*, vol. 6, no. 2, pp. 391–399, 2024.
- [13] N. A. Hanif, M. H. H. Ichsan, and A. S. Budi, “Rancangan Sistem Klasifikasi Kesuburan Tanah pada Tanaman Pangan berdasarkan PH dan Kelembapan berbasis Arduino Nano menggunakan Metode K-NN dan Aplikasi Android,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 6, no. 8, pp. 3629–3636, 2022.
- [14] E. V. Rahmadani, S. F. Pane, and N. H. Harani, *Algoritma C4. 5 dan K-Nearest Neighbors (KNN) untuk Memetakan Matakuliah dan Keterlambatan Kelulusan Mahasiswa*. Kreatif, 2020.
- [15] J. R. Sitinjak, M. H. H. Ichsan, and E. Setiawan, “Penerapan Metode Naive Bayes dalam Sistem Pendeteksi Kualitas Tanah pada Tanaman Kedelai,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 6, pp. 2617–2622, 2023.
- [16] U. Sa’adah, M. Y. Rochayani, D. W. Lestari, and D. A. Lusia, *Kupas Tuntas Algoritma Data Mining dan Implementasinya Menggunakan R*. Universitas Brawijaya Press, 2021.
- [17] P. P. Vedanty, “Perbandingan Metode K-Nearest Neighbor dan Naive Bayes Dalam Identifikasi Penyakit Daun Tanaman Obat,” 2024, *Universitas Pendidikan Ganesha*.
- [18] Y. Resti, C. Irsan, M. T. Putri, I. Yani, A. Ansyori, and B. Suprihatin, “Identification of corn plant diseases and pests based on dig[1] Y. Resti, C. Irsan, M. T. Putri, I. Yani, A. Ansyori, and B. Suprihatin, “Identification of corn plant diseases and pests based on digital images using multinomial naïve bayes and k-nearest n,” *Sci. Technol. Indones.*, vol. 7, no. 1, pp. 29–35, 2022.
- [19] F. J. Manurung, S. Sutejo, L. Lisnawita, and F. Fajrizal, “Klasifikasi Tanaman Herbal Menggunakan Metode Naïve Bayes Classifier Dan K-Nearest Neighbor,” in *SEMASTER: Seminar Nasional Teknologi Informasi & Ilmu Komputer*, 2024, pp. 411–422.
- [20] E. L. Arumingtyas, R. Mastuti, and L. Hakim, *Biologi tanaman hortikultura*. Universitas Brawijaya Press, 2021.

- [21] A. L. Adiredjo and L. Soetopo, *Pemuliaan Ketahanan Genetik Tanaman*. Universitas Brawijaya Press, 2021.
- [22] M. Malado *et al.*, *Pengendalian Hama dan Penyakit Tanaman Pertanian*. CV. Gita Lentera, 2024.
- [23] F. A. Sianturi, P. M. Hasugian, A. Simangunsong, and B. Nadeak, *DATA MINING: Teori dan Aplikasi Weka*, vol. 1. IOCS Publisher, 2019.
- [24] R. M. Sari, *Klasifikasi Data Mining*. Serasi Media Teknologi, 2024.
- [25] R. satria Wahono, "Data Mining," *Data Mining*.
- [26] G. S. Mahendra *et al.*, *Tren Teknologi AI: Pengantar, Teori, dan Contoh Penerapan Artificial Intelligence di Berbagai Bidang*. PT. Sonpedia Publishing Indonesia, 2024.
- [27] M. A. Naufal and B. Setiadi, "Perhitungan Jarak Dalam Sistem Deteksi Social Distancing Dengan Menggunakan Metode Euclidean Distance," in *Prosiding Industrial Research Workshop and National Seminar*, 2022, pp. 1431–1435.
- [28] R. Nooraeni and G. Nurfalah, "Kajian Penerapan Jarak Euclidean, Manhattan, Minkowski, dan Chebyshev pada Algoritma Clustering K-Prototype," *Sains, Apl. Komputasi dan Teknol. Inf.*, vol. 4, no. 2, pp. 72–82, 2022.
- [29] A. P. Agustin, A. C. Fauzan, and H. Harliana, "Implementasi K-Nearest Neighbor Dengan Jarak Minkowski Untuk Deteksi Dini Covid-19 Pada Citra CT-Scan Paru-Paru," *J. Ilm. Intech Inf. Technol. J. UMUS*, vol. 4, no. 01, pp. 23–30, 2022.
- [30] A. S. Huda, R. M. Awangga, and R. N. S. Fathonah, *Prediksi Penerimaan Pegawai Baru Dengan Metode Naive Bayes*, vol. 1. Kreatif, 2020.
- [31] F. Ramdani and I. Q. Utami, *Pengantar Data Science*. Bumi Aksara, 2024.
- [32] V. B. A. Pardosi *et al.*, *KECERDASAN KOMPUTASIONAL*. CV Rey Media Grafika, 2024.
- [33] B. Aribowo and S. Fairuz, *Panduan Praktis Machine Learning Klasifikasi Menggunakan Python: Diandra Kreatif*. Diandra Kreatif, 2024.
- [34] F. Elfaladonna, I. G. T. Isa, D. Sartika, and A. M. Putra, *Buku Ajar Dasar Exploratory Data Analysis (EDA)*. Penerbit NEM, 2024.
- [35] A. S. Rohman, *Master Coding HTML, CSS & Javascript: Menggunakan Visual Studio Code*. Anak Hebat Indonesia, 2024.

- [36] A. Gunawan, "Mobile Programming Menggunakan Flutter dan Visual Studio Code Untuk Pemula," 2024, *PT. Literasi Nusantara Abadi Grup*.
- [37] R. A. Siroj, W. Afgani, F. Fatimah, D. Septaria, and G. Z. Salsabila, "Metode Penelitian Kuantitatif Pendekatan Ilmiah untuk Analisis Data," *J. Rev. Pendidik. dan Pengajaran*, vol. 7, no. 3, pp. 11279–11289, 2024.
- [38] S. Khaerunnisa and R. Awaluddin, *Penelitian in silico untuk pemula*. Airlangga University Press, 2020.
- [39] B. Sumargo, *Teknik sampling*. Unj press, 2020.
- [40] G. Daruhadi and P. Sopiati, "Pengumpulan data penelitian," *J-CEKI J. Cendekia Ilm.*, vol. 3, no. 5, pp. 5423–5443, 2024.
- [41] G. Urva *et al.*, *PENERAPAN DATA MINING DI BERBAGAI BIDANG: Konsep, Metode, dan Studi Kasus*. PT. Sonpedia Publishing Indonesia, 2023.
- [42] M. I. Alif, I. Fitria, and B. N. Abdallah, *Buku Dasar-Dasar Analisis Data Pelajari Keterampilan Penting, Rangkul Masa Depan, dan Lejitkan Karir Anda di Dunia Berbasis Data*. Deepublish, 2024.



## LAMPIRAN

**PONDOK PESANTREN SALAFIYAH SYAFI'YAH SUKOREJO  
UNIVERSITAS IBRAHIMI  
PERPUSTAKAAN IBRAHIMI**

N P P . 3 5 1 2 1 4 2 F 2 0 0 6 5 6 7

Jl. KHR. Syamsul Arifin No. 1-2 PO. Box. 2 Kode Pos. 68374 Phone (0338) 452666 Fax. (0338) 453068  
SUMBEREJO BANYUPUTIH SITUBONDO JAWA TIMUR**SURAT KETERANGAN  
HASIL PEMERIKSAAN PLAGIASI**

Yang bertanda tangan di bawah ini

Nama : Muhammad Ali Ridla, M.Kom.  
Jabatan : Kepala Perpustakaan

Menyatakan dengan sebenarnya bahwa:

NIM : 2021503126  
Nama : NUR AIZAH  
Fakultas : Sains dan Teknologi  
Prodi : Teknologi Informasi  
Kecamatan : PANJI  
Kabupaten : SITUBONDO  
Provinsi : Jawa Timur  
Judul Skripsi : PERBANDINGAN EFEKTIVITAS METODE K-  
NEAREST NEIGHBOR DAN NAÏVE BAYES  
DALAM DATA PENGAMATAN KESEHATAN  
TANAMAN

Dengan dosen Pembimbing :

1. Ahmad Homaidi, M.Kom.
2. Lukman Fakhid Lidimilah, M.Kom

Telah dilakukan cek plagiasi di Perpustakaan Universitas Ibrahimi dengan persentase plagiasi terakhir sebesar 20% .

Demikian Surat Keterangan ini dibuat untuk dipergunakan sebagaimana mestinya.

Sukorejo, 16 Agustus 2025

Kepala Perpustakaan,



Muhammad Ali Ridla, M.Kom.

UU ITE No.11 Tahun 2008 Pasal 5 Ayat 1  
"Informasi Elektronik dan/atau Dokumen Elektronik  
dan/atau hasil cetaknya merupakan alat bukti yang sah."



**Prosiding Seminar Nasional  
Karsa Nusantara**

Kolaborasi Rekayasa dan Sains Nasional  
Untuk Teknologi, Riset, dan Kecerdasan Buatan  
ISSN 3090-1154 (Online)

Nomor : 001/LoA/KarsaNusantara/VII/2025  
Perihal : Letter of Acceptance (LoA)

**Kepada  
Yth.**

**Pemakalah Prosiding Seminar Nasional Karsa Nusantara 2025  
Di Tempat**

Pemimpin Redaksi Prosiding Seminar Nasional Karsa Nusantara telah melakukan evaluasi terhadap Naskah:

Judul : Perbandingan Efektivitas Metode K-Nearest Neighbor dan Naive Bayes dalam Data Pengamatan Kesehatan Tanaman  
Penulis Ke-1 : Nur Aizah\*1, Ahmad Homaidi2, Lukman Fakhri Lidimilah3  
Afiliasi : Universitas Ibrahimy Situbondo  
Korespondensi: Nur Aizah

Berdasarkan penilaian dari Reviewer terhadap Naskah yang telah disubmit, maka diputuskan bahwa Naskah tersebut DITERIMA, dan dapat dimuat pada terbitan prosiding Seminar Nasional Karsa Nusantara pada Volume 2 Tahun 2025.

Terimakasih untuk partisipasinya dalam mempercayakan naskahnya kepada prosiding kami.

Surabaya, 23 Juni 2025  
Pemimpin Redaksi



Dr. Ir. Anang Kukuh Adisusilo, ST., MT

**Pembimbing I:** Ahmad Homaidi, M.Kom

NO	TANGGAL	CATATAN	PARAF
1	08 Desember 2024	Pengajuan Judul & Dataset	[Signature]
2	14 Januari 2025	Bimbingan Bab I & III	[Signature]
3	02 Februari 2025	Bimbingan Bab III	[Signature]
4	13 April 2025	Bimbingan Revisi	[Signature]
5	11 Mei 2025	Revisi	[Signature]
6	18 Mei 2025	Pembahasan proposal	[Signature]
7	12 Juni 2025	Bimbingan Artikel	[Signature]
8	06 Juli 2025	Bimbingan Bab IV - V	[Signature]
9	06 Agustus 2025	ACC	[Signature]
10			
11			
12			

**Pembimbing II:** Lukman Fauzi L, M.Kom

NO	TANGGAL	CATATAN	PARAF
1	08 Desember 2024	Pengajuan judul	[Signature]
2	14 Januari 2025	Bimbingan Bab I - III	[Signature]
3	13 April 2025	Bimbingan Implementasi Sistem	[Signature]
4	11 Mei 2025	Bimbingan Sistem	[Signature]
5	18 Mei 2025	Pembahasan proposal	[Signature]
6	12 Juni 2025	Bimbingan Artikel	[Signature]
7	26 Juli 2025	Bimbingan Bab IV & V	[Signature]
8	09 Agustus 2025	ACC	[Signature]
9			
10			
11			
12			

Nomor : 092/03/KarsaNusantara/FT/VI/2025

# SERTIFIKAT

DIBERIKAN KEPADA :

## *Nur Aizah*

sebagai

### PRESENTER

pada Seminar Nasional Karsa Nusantara 2025 dengan tema  
"Menghubungkan Inovasi, Mewujudkan Masa Depan"  
di Universitas Wijaya Kusuma Surabaya tanggal 18 Juni 2025

Dekan Fakultas Teknik

Johan Paing Heru Waskito, ST, MT

ITS Institut Teknologi Sepuluh Nopember

Departemen Teknik Elektro Otomasi

IPB University

Sekolah Vokasi

DTPB - FTP UGM

## CURRICULUM VITAE

### Identitas Diri

Nama Lengkap : Nur Aizah  
 NPM : 2021503126  
 Tempat, Tanggal Lahir : Situbondo, 03 September 2002  
 Program Studi : Teknologi Informasi

### Nama Orang Tua

Ayah : Moh. Sujono  
 Ibu : Siti Hatija

### Latar Belakang Pendidikan

SD/MI : SDN 1 SELETRENG  
 SMP/MTs : SMP IBRAHIMY 3 SUKOREJO  
 SMA/SMK/MA : SMA IBRAHIMY SUKOREJO  
 Latar Organisasi : OSIM MISSPi, OSIS SMP IBRAHIMY 3, OSIM MDI, IKSASS SUB RAYON PANJI, BEM SAINTEK, DPM SAINTEK, PC IPPNU SITUBONDO, PK PMII UNIB, PC PMII SITUBONDO

### Alamat Rumah

Jalan : Jl. Gn. Arjuno RT.004/RW.013  
 Dusun : -  
 Desa/Kelurahan : Mimbaan  
 Kabupaten/Kota : Situbondo  
 Provinsi : Jawa Timur  
 No. Telepon : 085138029620  
 E-mail : [wittychaa@gmail.com](mailto:wittychaa@gmail.com)