

**KOMPARASI ALGORITMA KNN DAN SVM UNTUK KLASIFIKASI  
KESEHATAN MENTAL PADA USIA REMAJA**

**SKRIPSI**

**DISUSUN OLEH**

**RIZKY HIDAYAT HASIBUAN**

**2109020130**



**UMSU**

Unggul | Cerdas | Terpercaya

**PROGRAM STUDI TEKNOLOGI INFORMASI  
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI  
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA**

**MEDAN**

**2026**

LEMBAR PENGESAHAN

Judul Skripsi : KOMPARASI ALGORITMA KNN DAN SVM UNTUK  
KLASIFIKASI KESEHATAN MENTAL PADA USIA  
REMAJA  
Nama Mahasiswa : RIZKY HIDAYAT HASIBUAN  
NPM : 2109020130  
Program Studi : TEKNOLOGI INFORMASI

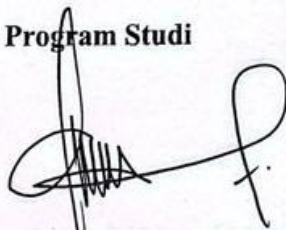
Menyetujui  
Komisi Pembimbing



(Mulkan Azhari, S.Kom., M.Kom.)

NIDN. 0108129402

Ketua Program Studi



(Fatma Sari Hutagalung, S.Kom., M.kom.)

NIDN. 0117019301

Dekan



(Dr. Al-Khowidzi, S.Kom., M.Kom.)

NIDN. 0127099201

## PERNYATAAN ORISINALITAS

### KOMPARASI ALGORITMA KNN DAN SVM UNTUK KLASIFIKASI KESEHATAN MENTAL PADA USIA REMAJA

#### SKRIPSI

Saya menyatakan bahwa karya tulis ini adalah hasil karya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya.

Medan, 2026 Mei 23

Yang membuat pernyataan



Kizky Muayad Hasibuan

NPM. 2109020130

**PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK  
KEPENTINGAN AKADEMIS**

Sebagai sivitas akademika Universitas Muhammadiyah Sumatera Utara, saya bertanda tangan di bawah ini:

Nama : Rizky Hidayat Hasibuan  
NPM : 2109020130  
Program Studi : Teknologi Informasi  
Karya Ilmiah : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Muhammadiyah Sumatera Utara Hak Bebas Royalti Non-Eksekutif (Non-Exclusive Royalty free Right) atas penelitian skripsi saya yang berjudul:

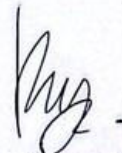
**KOMPARASI ALGORITMA KNN DAN SVM UNTUK KLASIFIKASI  
KESEHATAN MENTAL PADA USIA REMAJA**

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksekutif ini, Universitas Muhammadiyah Sumatera Utara berhak menyimpan, mengalih media, memformat, mengelola dalam bentuk database, merawat dan mempublikasikan Skripsi saya ini tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis dan sebagai pemegang dan atau sebagai pemilik hak cipta.

Demikian pernyataan ini dibuat dengan sebenarnya.

Medan, 2026

Yang membuat pernyataan



Rizky Hidayat Hasibuan

NPM. 2109020130

## RIWAYAT HIDUP

### DATA PRIBADI

Nama Lengkap : Rizky Hidayat Hasibuan  
Tempat dan Tanggal Lahir : Medan, 17 Juli 2003  
Alamat Rumah : Jln. Titipahlawan, Paya Pasir, Kec. Medan  
Marelan  
Telepon/Faks/HP : 0811614707  
E-mail : rizkyhidayathasibuan5@gmail.com

### DATA PENDIDIKAN

SD : SD Dr. Wahidin Sudirohusodo TAMAT: 2015  
SMP : SMP Negeri 5 Medan TAMAT: 2018  
SMA : SMA Alfityan School Medan TAMAT: 2021

## KATA PENGANTAR



### Pendahuluan

Penulis tentunya berterima kasih kepada berbagai pihak dalam dukungan serta doa dalam penyelesaian skripsi. Penulis juga mengucapkan terima kasih kepada:

1. Bapak Prof. Dr. Akrim. M.Pd. Rektor Universitas Muhammadiyah Sumatera Utara (UMSU)
2. Bapak Dr. Al-Khowarizmi, S.Kom., M.Kom. Dekan Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
3. Ibu Fatma Sari Hutagalung, S.Kom., M.Kom Ketua Program Studi Teknologi Informasi.
4. Bapak Okvi Nugroho, S.Kom., M.Kom Sekretaris Program Studi Teknologi Informasi.
5. Pembimbing saya, yaitu Bapak Mulkan Azhari S.Kom., M.Kom.
6. Kedua orang tua saya yang sudah mendukung saya dalam penyelesaian kuliah ini.

Semua pihak yang terlibat langsung ataupun tidak langsung yang tidak dapat penulis ucapkan satu-persatu yang telah membantu penyelesaian skripsi ini.

# KOMPARASI ALGORITMA KNN DAN SVM UNTUK KLASIFIKASI KESEHATAN MENTAL PADA USIA REMAJA

## ABSTRAK

Kesehatan mental pada usia remaja merupakan isu krusial yang terus mengalami peningkatan dan sering kali tidak terdeteksi secara dini akibat keterbatasan metode penilaian yang masih bersifat subjektif. Penelitian ini bertujuan untuk mengklasifikasikan tingkat kesehatan mental remaja berdasarkan tingkat stres menggunakan pendekatan machine learning serta membandingkan kinerja algoritma KNN dan SVM. Dataset yang digunakan terdiri dari 1.100 data remaja yang diperoleh dari GitHub dengan 11 atribut prediktor dan satu atribut target, yaitu *stress\_level* yang diklasifikasikan ke dalam tiga kelas: rendah, sedang, dan tinggi. Tahapan penelitian meliputi preprocessing data, EDA, feature selection, penanganan ketidakseimbangan data menggunakan SMOTE, pemodelan, dan evaluasi. Pengujian dilakukan menggunakan beberapa rasio pembagian data latih dan data uji. Evaluasi kinerja model menggunakan confusion matrix. Hasil penelitian menunjukkan bahwa algoritma SVM memberikan performa terbaik dengan akurasi sebesar 89,55% dan F1-Score sebesar 89,58% pada rasio data 80:20 sebelum penerapan SMOTE. Secara keseluruhan, SVM terbukti lebih stabil dan akurat dibandingkan KNN dalam mengklasifikasikan tingkat kesehatan mental remaja, sehingga berpotensi digunakan sebagai dasar sistem deteksi dini kesehatan mental berbasis data.

**Kata kunci:** Kesehatan Mental Remaja, Klasifikasi Tingkat Stres, K-Nearest Neighbor, *Support Vector Machine*

## COMPARISON OF KNN AND SVM ALGORITHMS FOR MENTAL HEALTH CLASSIFICATION IN ADOLESCENTS

### ABSTRACT

*Mental health among adolescents is a critical issue that continues to increase and is often not detected early due to the limitations of assessment methods that remain largely subjective. This study aims to classify adolescent mental health levels based on stress levels using a machine learning approach and to compare the performance of the KNN and SVM algorithms. The dataset used consists of 1,100 adolescent records obtained from GitHub, comprising 11 predictor attributes and one target attribute, namely stress\_level, which is classified into three categories: low, moderate, and high. The research stages include data preprocessing, EDA, feature selection, handling class imbalance using the SMOTE, modeling, and evaluation. Model testing was conducted using several training–testing split ratios. Model performance was evaluated using confusion matrix. The results indicate that the SVM algorithm achieved the best performance with an accuracy of 89.55% and an F1-Score of 89.58% using an 80:20 data split prior to the application of SMOTE. Overall, SVM demonstrated higher stability and accuracy compared to KNN in classifying adolescent mental health levels, indicating its strong potential as a data-driven early detection tool for adolescent mental health issues.*

**Keywords:** *Adolescent Mental Health, Stress Level Classification, K-Nearest Neighbor, Support Vector Machine.*

## DAFTAR ISI

LEMBAR PENGESAHAN.....	i
PERNYATAAN ORISINALITAS .....	ii
PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS .....	iii
RIWAYAT HIDUP .....	iv
KATA PENGANTAR .....	v
ABSTRAK.....	vi
ABSTRACT.....	vii
DAFTAR ISI .....	viii
DAFTAR TABEL .....	x
DAFTAR GAMBAR .....	xi
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang Masalah .....	1
1.2 Rumusan Masalah.....	4
1.3 Batasan Masalah .....	4
1.4 Tujuan Penelitian .....	5
1.5 Manfaat Penelitian .....	5
<b>BAB II LANDASAN TEORI.....</b>	<b>6</b>
2.1 Landasan Teori .....	6
2.1.1 Klasifikasi .....	6
2.1.2 Kesehatan Mental .....	7
2.1.3 K-Nearest Neighbor (KNN) .....	8
2.1.4 Support Vector Machine (SVM) .....	10
2.1.5 <i>Confusion Matrix</i> .....	13
2.1.6 <i>Tools</i> yang digunakan .....	14
2.2 Tinjauan Pustaka.....	15
<b>BAB III METODOLOGI PENELITIAN.....</b>	<b>18</b>
3.1 <i>Data Collection</i> .....	18
3.2 Preprocessing Data .....	20
3.2.1 <i>Cleaning Data</i> .....	20
3.2.2 Menghapus <i>Outlier</i> .....	21
3.3 <i>Exploratory Data Analysis</i> (EDA).....	21
3.4 <i>Splitting Data</i> .....	22
3.5 Feature Selection.....	22
3.6 <i>Oversampling</i> .....	23
3.7 <i>Modelling</i> .....	24
3.8 Evaluasi Model .....	24
3.9 Tempat dan Waktu Penelitian.....	27
<b>BAB IV HASIL DAN PEMBAHASAN.....</b>	<b>29</b>
4.1 <i>Modelling</i> .....	29
4.1.1 Implementasi K-Nearest Neighbor (KNN).....	29
4.1.2 Implementasi <i>Support Vector Machine</i> (SVM) .....	29
4.1.3 Hasil Implementasi pada Berbagai Rasio Split.....	30
4.2 <i>Evaluation</i> .....	33
4.4.1 Evaluasi Model Terbaik (SVM, Split 80:20, Sebelum SMOTE) ...	33
4.4.2 Analisis Hasil Evaluasi .....	34
<b>BAB V KESIMPULAN DAN SARAN .....</b>	<b>35</b>

5.1 Kesimpulan .....	35
5.2 Saran .....	36
<b>DAFTAR PUSTAKA .....</b>	<b>37</b>

## DAFTAR TABEL

	Halaman
Tabel 2. 1 <i>Confusion Matrix</i> untuk dua kelas .....	13
Tabel 2. 2 <i>Literature review</i> .....	16
Tabel 3. 1 Dataset Kesehatan Mental .....	19
Tabel 3. 2 Atribut yang digunakan .....	23
Tabel 3. 3 Waktu Penelitian .....	28

## DAFTAR GAMBAR

	Halaman
Gambar 2. 1 <i>Hyperplane</i> Dua Dimensi .....	11
Gambar 3. 1 Metodologi Penelitian.....	18
Gambar 3. 2 Atribut <i>missing values</i> .....	20
Gambar 3. 3 Mengatasi <i>outlier</i> .....	21
Gambar 3. 4 Proses setelah menghapus <i>outlier</i> .....	22
Gambar 3. 5 <i>Oversampling</i> menggunakan SMOTE .....	24
Gambar 3. 6 Tahapan Penelitian dengan <i>Fishbone</i> .....	25
Gambar 4. 1 Perbandingan rata-rata skor model sebelum dan sesudah SMOTE .	32
Gambar 4. 2 Hasil Evaluasi Detail Model Terbaik .....	33

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang Masalah**

Kesehatan mental pada usia remaja menjadi salah satu isu penting yang semakin mendapat perhatian di berbagai negara, termasuk Indonesia. Masa remaja merupakan fase transisi dari anak-anak menuju dewasa, di mana individu mengalami perubahan signifikan secara fisik, emosional, dan sosial. Ketidakstabilan emosi, tekanan akademik, serta pengaruh lingkungan sosial seringkali menyebabkan munculnya gangguan kesehatan mental seperti stres, depresi, maupun kecemasan. Fenomena ini semakin mengkhawatirkan karena berdasarkan berbagai survei dan laporan kesehatan, angka gangguan kesehatan mental pada remaja terus meningkat dari tahun ke tahun.

Masalah yang muncul adalah banyaknya kasus kesehatan mental remaja yang tidak terdeteksi sejak dini. Hal ini disebabkan oleh beberapa faktor, antara lain kurangnya kesadaran individu terhadap kondisi psikologisnya, keterbatasan tenaga profesional di bidang kesehatan mental, serta stigma sosial yang masih kuat di masyarakat. Akibatnya, banyak remaja baru mendapatkan perhatian ketika kondisi mental mereka sudah tergolong berat. Metode penilaian yang digunakan selama ini juga masih dominan bersifat subjektif, seperti wawancara atau kuesioner manual, yang sangat bergantung pada interpretasi responden maupun penilai. Pendekatan ini seringkali menghasilkan hasil yang tidak konsisten dan sulit digunakan untuk pemantauan secara luas.

Proses klasifikasi kondisi kesehatan mental pada individu dewasa muda bersifat kompleks dan membutuhkan metode yang sistematis serta presisi tinggi. Pendekatan tradisional dalam proses pengkategorian sering kali mengandalkan penilaian subjektif dan keahlian klinis, yang rentan terhadap bias serta ketidakakuratan. Dalam beberapa tahun terakhir, kemajuan teknologi komputasi serta penerapan metode pembelajaran mesin semakin menarik perhatian dalam penelitian mengenai kesehatan mental. Klasifikasi kesehatan mental pada usia remaja dapat dilakukan dengan *machine learning* seperti menggunakan metode KNN dan SVM. *K-Nearest Neighbor* (KNN) adalah algoritma *machine learning* yang bekerja berdasarkan prinsip bahwa data yang memiliki kesamaan fitur akan memiliki klasifikasi yang mirip. Metode KNN yang sederhana dalam penerapannya sering digunakan karena kemampuannya untuk mengelola data dalam berbagai bentuk dan ukuran tanpa memerlukan asumsi khusus tentang distribusi data. Meskipun, KNN memiliki keterbatasan, terutama dalam hal kecepatan perhitungan pada dataset besar dan sensitivitasnya terhadap data yang bising. Beberapa penelitian terbaru telah menyelidiki cara untuk meningkatkan kinerja KNN, seperti melalui pemilihan fitur yang lebih optimal dan penggunaan teknik pra-pemrosesan yang canggih (Hadi et al., 2025).

*Support Vector Machine* (SVM) adalah algoritma yang lebih kompleks dan dirancang untuk menemukan *hyperplane* yang memisahkan dua kelas data dengan margin maksimum. SVM dikenal efektif dalam menangani kumpulan data dengan banyak fitur dan dapat memberikan kinerja yang baik meskipun pada data yang tidak dapat dipisahkan secara *linear*. SVM juga memiliki keunggulan dalam kemampuan penyebaran yang lebih baik dibandingkan KNN. Walau begitu, SVM

memerlukan penyetelan parameter yang lebih teliti, seperti pemilihan inti yang sesuai, untuk mencapai hasil yang maksimal (Hadi et al., 2025).

Penelitian sebelumnya oleh (Arrafi & Ramadhanu, 2025) yang berjudul “Implementasi Metode Algoritma *Principal Component Analysis* (PCA) dan Algoritma *K-Nearest Neighbor* (KNN) dalam Klasifikasi Buah Jambu Madu Jambu Merah dan Manggis” menunjukkan bahwa penggunaan *Principal Component Analysis* (PCA) dan *K-Nearest Neighbor* (KNN) efektif dalam melakukan klasifikasi terhadap buah jambu madu, buah jambu merah dan buah manggis dengan hasil akurasi mencapai 95%. Sementara itu, studi berikutnya oleh (Firdaus et al., 2025), dalam penelitian berjudul “Optimasi Kinerja Analisis Kernel *Support Vector Machine* (SVM) untuk Klasifikasi Operasi *Caesar* Persalinan” memperoleh tingkat akurasi pengujian sebesar 83.36%. Penelitian lain yang dilakukan oleh (Mulyani et al., 2025), dengan judul “Perbandingan Kinerja Algoritma KNN dan SVM Menggunakan SMOTE untuk Klasifikasi Penyakit Diabetes” Hasil dari penelitian dengan menggunakan algoritma SVM dengan kernel *Radial Basis Function* (RBF) mencapai akurasi sebesar 81,67%. Di sisi lain, algoritma KNN dengan nilai  $k=3$  yang ditemukan melalui *cross-validation* mencapai akurasi sebesar 83,33%. Studi lainnya yang dilakukan oleh (Hadi et al., 2025), berjudul “Analisis Perbandingan Algoritma *K-Nearest Neighbor* (KNN) dan *Support Vector Machine* (SVM) Dalam Klasifikasi Data Perbankan” menunjukkan bahwa SVM lebih baik daripada KNN dalam mengklasifikasikan risiko kredit pada data perbankan, terutama untuk pinjaman rumah. SVM dengan kernel dot dan ANOVA mencapai akurasi 80,56%, dibandingkan dengan 69,44% untuk KNN dengan  $k=50$ . Berdasarkan permasalahan tersebut, penelitian ini akan mencoba menerapkan

perbandingan algoritma *K-Nearest Neighbor* (KNN) dengan *Support Vector Machine* (SVM) dalam klasifikasi kesehatan mental pada kelompok usia remaja.

Berdasarkan pemaparan dari fenomena tersebut yang menunjukkan adanya kenaikan pada kesehatan mental remaja yang buruk dan dampak serius yang akan ditimbulkan kedepannya seperti perilaku menyimpang, kelainan psikologis atau adanya usaha untuk mengakhiri hidup, serta penelitian terdahulu, maka dari itu, Peneliti ingin membuat klasifikasi kesehatan mental usia remaja menggunakan algoritma *K-Nearest Neighbor* (KNN) dan *Support Vector Machine* (SVM) agar dapat digunakan sebagai bentuk pencegahan dan penanggulangan atau deteksi dini dari dampak fenomena yang sudah terjadi.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang penelitian yang telah dijelaskan, penelitian ini bertujuan untuk mengklasifikasikan kesehatan mental pada usia remaja dengan memanfaatkan algoritma *K-Nearest Neighbor* (KNN) dan *Support Vector Machine* (SVM), sekaligus menganalisis performa masing-masing algoritma dalam menangani klasifikasi tersebut.

## **1.3 Batasan Masalah**

Agar penelitian ini lebih terfokus dan tidak menyimpang dari tujuan utama, maka dibuat batasan masalah sebagai berikut:

1. Data yang digunakan dalam penelitian ini adalah data kesehatan mental dengan jumlah data sebanyak 1.100 baris bersumber dari *platform* Github.
2. Algoritma yang digunakan dalam penelitian ini membandingkan hasil akurasi algoritma *K-Nearest Neighbor* (KNN) dengan *Support Vector Machine* (SVM).

3. Bahasa pemrograman pada penelitian ini menggunakan bahasa pemrograman *Python*.
4. Metrik perhitungan performa model pada penelitian ini menggunakan *Confusion Matrix*.
5. *Script Python* yang digunakan dalam penelitian ini dijalankan dengan *platform Google Colab*.

#### **1.4 Tujuan Penelitian**

Mengetahui klasifikasi kesehatan mental usia remaja menggunakan algoritma *K-Nearest Neighbor* (KNN) dan *Support Vector Machine* (SVM) dan hasil performa dari masing-masing algoritma yang meliputi *Accuracy*, *Precision*, *Recall* dan *F1-Score*.

#### **1.5 Manfaat Penelitian**

Manfaat dari penelitian ini adalah:

1. Bagi Institusi Pendidikan, Penelitian ini dapat menjadi referensi bagi peneliti lain yang ingin melakukan penelitian khususnya pada kesehatan mental usia remaja.
2. Bagi Peneliti, Penelitian ini memungkinkan penulis untuk memperkaya pengetahuan dan wawasannya tentang berbagai metode dan teori yang relevan dengan bidang penelitian yang dipilih.
3. Bagi Masyarakat, Penelitian ini diharapkan dapat membantu dalam analisa dan deteksi dini pada tingkat kesehatan mental usia remaja.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Landasan Teori**

Landasan teori ini berisi tentang dasar-dasar teori dan konsep-konsep yang berkaitan dengan masalah dalam penelitian yang akan dijadikan sebagai referensi yang diambil dari buku, jurnal, artikel atau skripsi yang akan mendukung dalam penelitian ini.

##### **2.1.1 Klasifikasi**

Klasifikasi adalah penggalian data untuk menemukan pola dan hubungan dalam dataset, dengan tujuan untuk mengelompokan dan mengklasifikasi data ke dalam kategori yang telah ditentukan sebelumnya. Pada tahap awal, klasifikasi melibatkan pengumpulan data yang relevan untuk proses klasifikasi. Data ini dapat berasal dari berbagai sumber seperti basis data perusahaan, aplikasi perangkat lunak, atau sumber data eksternal lainnya. Penting untuk memastikan kualitas data yang baik dan relevan dalam proses ini untuk mendapatkan hasil yang akurat dan bermakna.

Klasifikasi merupakan teknik data mining yang menggunakan data latih berlabel kelas sebagai dasar pembelajaran memprediksi kelas suatu objek yang tidak diketahui label kelasnya. Ada 2 proses dalam klasifikasi yaitu proses *training* dan *testing*. Untuk melakukan pembangunan model digunakan data *training* dan melakukan tes menggunakan data *testing* (Pebdika et al., 2023). Model klasifikasi dapat membantu mengidentifikasi pola perilaku dan preferensi peserta.

Algoritma klasifikasi berupaya membentuk serangkaian aturan, yang digunakan sebagai panduan untuk memprediksi kelas dari data prediksi.

### **2.1.2 Kesehatan Mental**

WHO mendefinisikan kesehatan mental sebagai keadaan sejahtera dimana seseorang mampu mewujudkan potensi dirinya, mengatasi tekanan hidup yang normal, bekerja secara produktif dan sukses, serta berkontribusi pada komunitasnya. Dalam buku kesehatan mental, kesehatan mental berkaitan dengan beberapa hal. Pertama adalah bagaimana seseorang berpikir, merasakan, dan menjalani hidupnya. Yang kedua adalah bagaimana seseorang memandang dirinya sendiri dan orang lain. dan ketiga, bagaimana seseorang mengevaluasi berbagai alternatif solusi dan mengambil keputusan terkait dengan situasi yang dihadapi Yusuf (Kumowal et al., 2022). Kesehatan mental adalah bagian mendasar dari definisi kesehatan. Kesehatan mental yang baik memungkinkan orang untuk mencapai potensi mereka, mengatasi tekanan kehidupan sehari-hari, bekerja secara produktif, dan berkontribusi pada komunitas mereka. Oleh karena itu, keberadaan gangguan jiwa tidak boleh dianggap remeh. Sebab, jumlah kasusnya masih sangat memprihatinkan. Sekitar 450 juta orang di seluruh dunia menderita gangguan mental dan perilaku. Diperkirakan satu dari empat orang akan menderita gangguan jiwa semasa hidupnya. Menurut WHO SEARO, India memiliki jumlah kasus gangguan depresi tertinggi (56.675.969 kasus atau 4,5% dari populasi) dan Maladewa memiliki jumlah kasus terendah (12.739 kasus atau 4,5% dari populasi). ). Sedangkan jumlah orang yang terinfeksi di Indonesia sebanyak 9.162.886 orang atau setara dengan 3,7% dari jumlah penduduk.

### 2.1.3 K-Nearest Neighbor (KNN)

*K-Nearest Neighbors* (KNN) adalah algoritma pembelajaran mesin yang digunakan untuk klasifikasi dan regresi. Algoritma ini bekerja dengan mencari sejumlah  $k$  tetangga terdekat dari data yang ingin diprediksi, berdasarkan metrik jarak tertentu seperti jarak *Euclidean*. Prediksi dilakukan dengan menganalisis label atau nilai dari tetangga-tetangga terdekat tersebut. *K-Nearest Neighbors* (KNN) dikenal karena kesederhanaannya dan efektivitasnya dalam berbagai aplikasi, seperti pengenalan pola dan pengambilan keputusan. *K-Nearest Neighbors* (KNN) merupakan salah satu algoritma pembelajaran mesin yang bersifat sederhana namun efektif dalam tugas klasifikasi dan regresi. Prinsip dasar dari algoritma ini adalah asumsi bahwa objek-objek yang berada dalam jarak yang dekat satu sama lain cenderung memiliki karakteristik yang serupa. Dengan kata lain, jika karakteristik suatu objek telah diketahui, maka status objek lain dapat diprediksi berdasarkan kemiripan dengan tetangga terdekatnya (Sakti & Daulay, 2024).

Sebagai pengembangan dari teknik *K-Nearest Neighbors*, algoritma KNN mengklasifikasikan data baru berdasarkan suara mayoritas dari  $K$  tetangga terdekat. Nilai  $K$  merupakan bilangan bulat positif yang menentukan jumlah tetangga yang akan dipertimbangkan dalam proses klasifikasi. Umumnya, nilai  $K$  dipilih dalam jumlah ganjil untuk menghindari kemungkinan hasil seri dalam proses pemungutan suara (*voting*) antar tetangga.

Sebagai metode *non-parametrik*, KNN tidak bergantung pada asumsi distribusi data tertentu. Hal ini menjadikannya fleksibel dalam menangani berbagai jenis dataset. Dalam tugas klasifikasi, KNN menentukan kategori

suatu objek berdasarkan mayoritas kelas dari K tetangga terdekat. Sementara itu, dalam regresi, nilai yang diprediksi dihitung sebagai rata-rata atau median dari nilai-nilai tetangga terdekat.

Dalam berbagai bidang, KNN telah banyak diterapkan untuk berbagai kebutuhan analisis data. Salah satu penerapannya dalam studi sosial ekonomi adalah dalam analisis kemiskinan, di mana algoritma ini dapat digunakan untuk mengelompokkan individu atau rumah tangga berdasarkan karakteristik ekonomi, pendidikan, dan sosial. Dengan membandingkan atribut individu dengan data yang telah diklasifikasikan sebelumnya, algoritma KNN dapat membantu dalam menentukan status kemiskinan suatu kelompok masyarakat.

Keunggulan utama dari KNN terletak pada kemudahannya dalam implementasi serta kemampuannya dalam menangani data yang tidak memiliki pola distribusi yang jelas. Namun, kelemahan utama algoritma ini adalah sensitivitasnya terhadap jumlah K yang dipilih serta kompleksitas komputasi yang meningkat seiring bertambahnya jumlah data. Oleh karena itu, pemilihan nilai K yang optimal dan penggunaan teknik optimasi, seperti *feature scaling* dan *dimensionality reduction*, menjadi faktor penting dalam meningkatkan performa

algoritma K-NN.  $d_{(a,b)} = \sqrt{\sum_{g=1}^P (x_{ag} - x_{bg})^2}$

Keterangan:

$d_{(a,b)}$  = jarak antara objek *a* dengan *b*

$x_{(a,g)}$  = nilai objek *training a* pada variabel ke-*g*

$x_{(b,g)}$  = nilai objek *training b* pada variabel ke-*g*

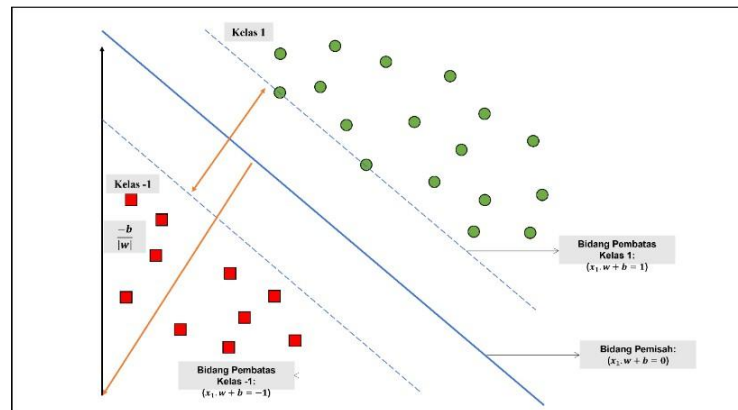
*P* = banyaknya variabel bebas

Algoritma pengerjaan metode *K-Nearest Neighbor* adalah sebagai berikut:

1. Tetapkan parameter  $K$  (jumlah tetangga terdekat).
2. Hitung jarak data baru/data *testing* menggunakan jarak *Euclidean* dengan semua data yang ada pada data *training* (persamaan 2)
3. Tentukan tetangga mana yang paling dekat berdasarkan jarak ke- $K$  yang paling minimal.
4. Menguraikan kategori dari tetangga terdekat.
5. Menggunakan kategori yang paling dapat diandalkan dari tetangga terdekat sebagai prediksi data baru.

#### **2.1.4 Support Vector Machine (SVM)**

*Support Vector Machine* (SVM) adalah metode pembelajaran *supervised* yang menganalisis data dan mengenali pola untuk klasifikasi dan regresi, SVM bekerja menggunakan cara mencari *hyperplane* terbaik dengan memaksimalkan jarak antar kelas, *hyperplane* merupakan sebuah fungsi yang digunakan untuk pemisah antar kelas. SVM merupakan sebuah pengklasifikasi maka diberi suatu himpunan pelatihan yang ditandai sebagai milik salah satu dari kelas kategori, algoritma SVM membangun sebuah model yang memprediksi apakah data yang baru diproses merupakan salah satu dalam kategori yang lain (Noviriandini et al., 2022). Berikut merupakan gambar *Hyperplane* dua dimensi yang dapat dilihat pada Gambar 2.1.



**Gambar 2. 1** *Hyperplane* Dua Dimensi

(Sumber : Shedriko, 2021)

Pada Gambar 2.1, setiap kelas terdefinisi oleh vektor  $w$  dan nilai  $b$ , yang bersama-sama mendefinisikan sebuah bidang *hyperplane* dalam ruang fitur (Shedriko, 2021). Perubahan pada nilai  $b$  akan mengakibatkan pergeseran pada posisi *hyperplane* tersebut. Dalam kasus dua kelompok objek yang berasal dari kelas yang berbeda, terdapat sebuah *hyperplane* optimal yang memisahkan keduanya. Untuk menemukan *hyperplane* optimal, tujuan utama adalah memaksimalkan margin, yaitu jarak terbesar antara *hyperplane* dengan titik data terdekat dari kedua kelas. Titik data yang berada pada kelas -1 memenuhi persamaan  $w \cdot x_i + b = -1$ , sedangkan titik data yang berada pada kelas +1 memenuhi persamaan  $w \cdot x_i + b = 1$ . Setelah *hyperplane* ditemukan, langkah selanjutnya adalah menerapkan algoritma *Support Vector Machine* (SVM) untuk melakukan klasifikasi. Langkah-langkah dalam algoritma SVM untuk menemukan *hyperplane* optimal dan mengklasifikasikan data yaitu sebagai berikut.

- a. Tentukan titik data  $\{x_1, x_2, \dots, x_n\}$  yang merupakan atribut pada data.
- b. Menentukan kelas :  $y \in \{-1, +1\}$ , -1 sebagai Malignant dan +1 sebagai *Benign*.

$$(x + a)^n = \sum_{k=0}^n \binom{n}{k} x^k a^{n-k}$$

- c. Menentukan data kelas berdasarkan rumus  $\{(x_i, y_i)\}_{i=1}^n$  dimana  $x_i$  merupakan vektor baris fitur ke- $i$ ,  $y_i$  merupakan label kelas dari  $x_i$ , dan  $N$  merupakan banyaknya data.
- d. Memaksimalkan fungsi berdasarkan persamaan (2.1).

$$Ld = \sum_{i=1}^N d_i - \sum_{i=1}^N \sum_{j=1}^N k_i k_j y_i y_j K(x_i, x_j), 0 \leq k_i \leq C \text{ dan } \sum_{i=1}^N k_i y_i = 0 \quad (2.1)$$

Dengan  $Ld$  yaitu dualitas *Langrange Multiplier*,  $K$  yaitu nilai bobot setiap titik data,  $C$  yaitu Nilai Konstanta,  $N$  yaitu banyaknya data, dan  $d$  yaitu jarak antar setiap data ke *hyperplane*.

- e. Menghitung nilai  $w$  dan  $b$  berdasarkan persamaan (2.2) dan (2.3).

$$w = \sum_{i=1}^N a_i y_i x_i \quad (2.2)$$

$$b = \frac{-1}{2} (w \cdot x^i + w \cdot x) \quad (2.3)$$

dimana  $w$  adalah nilai yang berkaitan dengan margin,  $N$  adalah banyaknya data,  $a$  merupakan nilai angka terendah,  $x$  merepresentasikan vektor input,  $y$  merupakan  $\{-1, +1\}$ , dan  $b$  merupakan bias.

- f. Menghitung fungsi keputusan klasifikasi  $sign(f(x))$  menggunakan persamaan (2.4).

$$f(x) = w \cdot x + b \quad (2.4)$$

atau dengan persamaan (2.5)

$$f(x) = \sum_{i=1}^m a_i y_i K(x, x_j) + b \quad (2.5)$$

dengan  $m$  adalah jumlah titik data yang dimiliki  $a_i > 0$ , dan  $K(x, x_i)$  merupakan fungsi kernel. Pada penelitian ini kernel yang digunakan adalah kernel *linier*. Adapun pendefinisian pada persamaan (2.6).

$$K(x, y) = x \cdot y \quad (2.6)$$

### 2.1.5 Confusion Matrix

*Confusion Matrix* atau *error matrix* adalah ringkasan hasil prediksi pada permasalahan klasifikasi. Jumlah klasifikasi yang benar dan yang salah dikumpulkan dengan nilai hitung kemudian dipecah oleh setiap kelas. Sehingga *Confusion Matrix* tidak hanya memberikan informasi kesalahan yang dibuat *Classifier* tetapi juga jenis kesalahannya (Umar et al., 2020). Pada Tabel 2.1 merupakan tabel *Confusion Matrix*.

**Tabel 2. 1** *Confusion Matrix* untuk dua kelas

		Aktual	
		Positif	Negatif
Prediksi	Positif	TP	FP
	Negatif	FN	TN

*Confusion Matrix* dapat dilihat dengan beberapa ketentuan sebagai berikut :

- a. *Positive* (P): aktual bernilai positif
- b. *Negative* (N): aktual bernilai negatif
- c. *True Positive* (TP): aktual bernilai positif, dan diprediksi positif
- d. *True Negative* (TN): aktual bernilai negatif, dan diprediksi negatif
- e. *False Positive* (FP): aktual bernilai negatif, tetapi diprediksi positif
- f. *False Negative* (FN): aktual bernilai positif, tetapi diprediksi negatif juga

Untuk dapat menghitung akurasi pada tabel *Confusion Matrix* dapat menggunakan rumus Persamaan 2.6 sebagai berikut.

$$Accuracy = \frac{(TP + TN)}{(TP + FN + FP + TN)} \quad (2.6)$$

Rasio dari semua hal relevan yang dipilih untuk semua item yang dipilih dikenal sebagai presisi. Oleh karena itu, presisi dapat dipahami sebagai korespondensi antara permintaan informasi dan tanggapan (Wulan Dari & Triloka, 2022). Rumus Persamaan 2.7 di bawah ini dapat digunakan untuk menghitung presisi sebagai berikut.

$$Precision = \frac{TP}{(TP + FP)} \quad (2.7)$$

*Recall* adalah proporsi dari hal-hal relevan yang dipilih untuk semua item relevan yang dapat diakses (Wulan Dari & Triloka, 2022). Rumus Persamaan 2.8 berikut ini dapat digunakan untuk menentukan *Recall* sebagai berikut.

$$Recall = \frac{TP}{(TP + TN)} \quad (2.8)$$

Angka antara 0 dan 1 atau perhitungan persentase (1–100%) dapat digunakan untuk mencetak *Precision* dan *Recall*. Rata-rata *Recall* dan *Precision* dibandingkan kemudian menghasilkan *F1 Score* (Powers, 2020). Rumus Persamaan 2.9 di bawah ini dapat digunakan untuk menghitung *F1 Score* sebagai berikut.

$$F_1Score = 2 \times \frac{precision \times recall}{precision + recall} \quad (2.9)$$

### 2.1.6 Tools yang digunakan

Adapun *tools* yang digunakan dalam penelitian ini adalah sebagai berikut.

### **A. *Python***

*Python* adalah bahasa scripting tingkat tinggi yang kuat, interaktif, dan berorientasi objek. *Python* merupakan bahasa pemrograman yang *freeware*, tidak ada batasan dalam penyalinannya atau mendistribusikannya lengkap dengan *source code* nya, *debugger* dan *profiler*, fungsi sistem, GUI (antar muka pengguna grafis) dan basis datanya (Saabith et al., 2019).

### **B. *Google Colaboratory***

*Google Colaboratory* atau *Colab* adalah proyek yang memiliki tujuan untuk menyebarkan pendidikan dan penelitian pembelajaran mesin. *Colaboratory* menyediakan *runtime Python* 2 dan 3 yang telah dikonfigurasi sebelumnya dengan pembelajaran mesin yang penting dan perpustakaan kecerdasan buatan, seperti *Tensorflow*, *Matplotlib*, dan *Keras*. Layanan ini menyediakan *runtime* yang dipercepat GPU, juga sepenuhnya dikonfigurasi dengan perangkat lunak yang sebelumnya diuraikan. Infrastruktur Kolaborasi *Google* di-host di *platform Google Cloud* (Carneiro et al., 2018).

## **2.2 Tinjauan Pustaka**

Dalam penyusunan penelitian ini menggunakan beberapa penelitian sebelumnya yang sudah ada dalam bentuk jurnal dengan penggunaan metode penelitian yang sama dan berkaitan untuk digunakan sebagai perbandingan terhadap penelitian yang sedang dilakukan. Penelitian terkait yang sudah dilakukan oleh peneliti sebelumnya disajikan pada Tabel 2.2 berikut.

Tabel 2. 2 Literature review

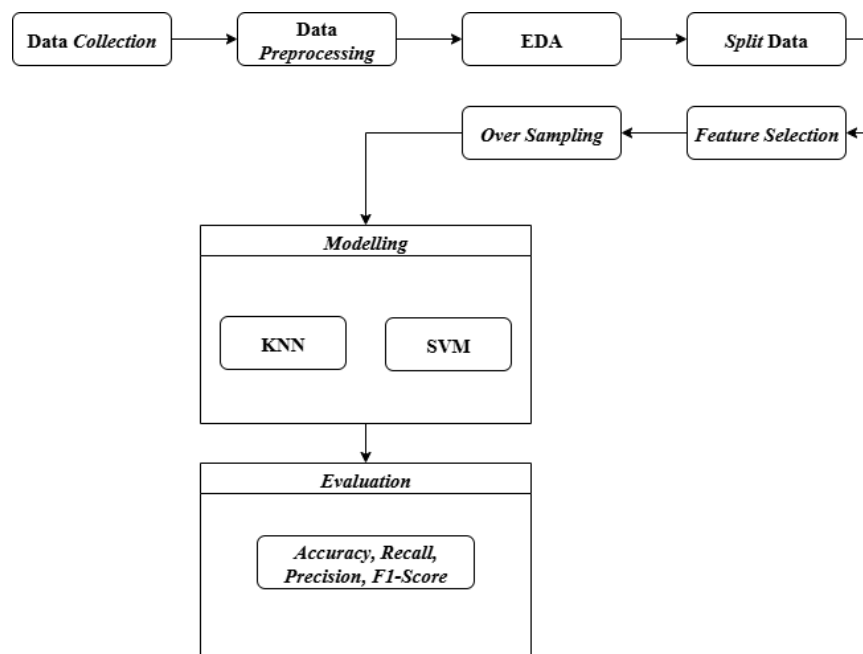
No.	Peneliti	Judul	Sumber	Metode	Hasil
1.	Alfarezy, R., Ermatita, E., & Wadu, R. M. B (2022)	Implementasi Algoritma <i>Naïve</i> <i>Bayes</i> untuk Analisis Klasifikasi Survei Kesehatan Mental (Studi Kasus: <i>Open</i> <i>Sourcing</i> Mental Illness)	Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)	<i>Naïve Bayes</i>	Hasil dari penelitian ini menunjukkan bahwa hasil akurasi yang didapatkan untuk analisis klasifikasi yang dilakukan dengan algoritma <i>Naïve Bayes</i> adalah 72% pada ukuran data uji 30% dan data latih 70%, yang merupakan hasil akurasi tertinggi yang didapatkan setelah dilakukan percobaan dengan ukuran pembagian data lain pada saat percobaan.
2.	Putra, H. D., Khairani, L., & Hastari, D. (2023)	Comparison of <i>Naïve Bayes</i> <i>Classifier</i> and <i>Support Vector</i> <i>Machine</i> <i>Algorithms for</i> <i>Classifying Student</i> <i>Mental Health</i> Data	SENTIMAS: Seminar Nasional Penelitian dan Pengabdian Masyarakat	<i>Naïve Bayes</i> <i>Classifier</i> dan <i>Support</i> <i>Vector</i> <i>Machine</i>	Penelitian ini menggunakan algoritma klasifikasi <i>Naïve Bayes</i> dan <i>Support Vector Machine</i> . Hasil akurasi untuk algoritma SVM sebesar 94,37% dan <i>Naïve Bayes</i> sebesar 86,87%.
3.	Mulyani, A., Khoerunisa, S., & Kurniadi, D. (2024)	Perbandingan Kinerja Algoritma KNN dan SVM Menggunakan SMOTE untuk Klasifikasi Penyakit Diabetes	Jurnal Nasional Teknik Elektro dan Teknologi Informasi	SVM dan KNN	Hasil penelitian menunjukkan bahwa kedua algoritma SVM dengan kernel <i>Radial Basis Function</i> (RBF) mencapai akurasi sebesar 81,67%. Di sisi lain, algoritma KNN dengan nilai $k = 3$ yang ditemukan melalui <i>cross-validation</i> mencapai akurasi sebesar 83,33%. Berdasarkan evaluasi <i>Confusion Matrix</i> , KNN

					menunjukkan kinerja yang lebih unggul dibandingkan SVM dalam hal akurasi.
4.	Muhammad Ikhsan Al-Arrafi & Agung Ramadhanu (2025)	Implementasi Metode Algoritma <i>Principal Component Analysis</i> (PCA) dan Algoritma <i>K-Nearest Neighbor</i> (KNN) dalam Klasifikasi Buah Jambu Madu Jambu Merah dan Manggis	<i>Journal of Science and Social Research</i>	KNN	Hasil dari penelitian ini menunjukkan bahwa penggunaan <i>Principal Component Analysis</i> (PCA) dan <i>K-Nearest Neighbor</i> (KNN) efektif dalam melakukan klasifikasi terhadap buah jambu madu, buah jambu merah dan buah manggis dengan hasil akurasi mencapai 95%.
5.	Abdul, A. F., Egi Pratama, Nurul, N. S., Rahayu, L. K., Wiyanto (2025)	Optimasi Kinerja Analisis Kernel <i>Support Vector Machine</i> (SVM) untuk Klasifikasi Operasi <i>Caesar</i> Persalinan	JATI (Jurnal Mahasiswa Teknik Informatika)	SVM	Hasil dari penelitian ini menghasilkan tingkat akurasi pada pengujian 83.36% menggunakan bantuan kernel RBF sebagai akurasi terbaik, dibandingkan dengan kernel <i>linear</i> sebesar 78.92%, kernel <i>sigmoid</i> sebesar 65.05%, kernel <i>polynomial</i> sebesar 82.39%.
6.	Rosalia, H., Ni Luh Gede Pivin Suwirmayanti, I Gusti Ngurah, A. K., I Gusti, A. D. S., Putu Devi Novayanti (2025)	Analisis Perbandingan Algoritma <i>K-Nearest Neighbor</i> (KNN) dan <i>Support Vector Machine</i> (SVM) Dalam Klasifikasi Data Perbankan	Jurnal informasi dan Komputer	KNN dan SVM	Hasil dari penelitian ini menunjukkan bahwa SVM lebih baik daripada KNN dalam mengklasifikasikan risiko kredit pada data perbankan, terutama untuk pinjaman rumah. SVM dengan kernel <i>dot</i> dan ANOVA mencapai akurasi 80,56%, dibandingkan dengan 69,44% untuk KNN dengan k=50.

## BAB III

### METODOLOGI PENELITIAN

Metodologi penelitian ini dirancang untuk mengkaji klasifikasi kesehatan mental pada usia remaja dengan menggunakan algoritma KNN dan SVM. Bab ini akan membahas langkah-langkah yang dilakukan mulai dari pengumpulan data, *Preprocessing data*, implementasi model, hingga evaluasi kinerja algoritma seperti yang disajikan pada Gambar 3.1.



**Gambar 3. 1** Metodologi Penelitian

#### **3.1 Data Collection**

Tahap awal dalam penelitian ini dimulai dengan proses data *Collection* atau pengumpulan data, di mana langkah ini dilakukan dengan mengumpulkan informasi yang berkaitan dengan kesehatan mental. Dalam penelitian ini, dataset yang digunakan berasal dari *platform* GitHub dengan total 1.100 baris data dan 12 kolom.

**Tabel 3. 1** Dataset Kesehatan Mental

anx iet y_l eve l	ment al_he alth_ histor y	de pr es sion	he ad ac he	sle ep_ qua lity	breat hing_ pro blem	livin g_c ondi tion s	acade mic_p erfor manc e	stu dy_ l oad	future _care er_co ncern s	extrac urricul ar_acti vities	str ess_ le vel
14	0	11	2	2	4	3	3	2	3	3	1
15	1	15	5	1	4	1	1	4	5	5	2
12	1	14	2	2	2	2	2	3	2	2	1
16	1	15	4	4	3	2	2	4	4	4	2
16	0	7	2	5	1	2	4	3	2	0	1
20	1	21	3	1	4	2	2	5	5	4	2
4	0	6	1	4	1	4	5	1	1	2	0
17	1	22	4	1	5	1	1	3	4	4	2
13	1	12	3	2	4	3	3	3	3	2	1
6	0	27	4	6	2	5	2	2	5	3	1
17	1	25	4	5	3	2	1	3	4	4	2
17	1	22	3	6	5	2	1	3	4	5	2
5	0	8	1	4	2	3	5	2	1	1	0
9	1	24	4	8	0	2	1	2	3	0	2
2	0	3	1	4	2	3	4	2	1	2	0
11	0	14	3	5	4	2	3	3	3	2	1
6	0	1	1	4	2	4	5	1	1	2	0
7	0	3	1	4	2	4	4	2	1	1	0
11	0	12	3	2	2	2	2	3	2	2	1
21	1	25	4	1	4	1	1	5	5	4	2
3	0	0	1	4	1	3	5	2	1	2	0
18	1	21	4	1	3	1	2	5	4	4	2
7	0	5	1	4	1	3	4	2	1	2	0
20	1	26	3	5	4	2	1	3	4	4	2
13	1	14	3	2	2	2	3	2	2	3	1

Setelah *dataset* didapatkan, kolom target yang digunakan pada dataset tersebut yaitu kolom '*stress\_level*' yang berisi angka 0 untuk tingkat stres Rendah, 1 untuk tingkat stres Sedang, dan 2 untuk tingkat stres Tinggi yang selanjutnya akan dilakukan proses pembersihan data atau *Preprocessing data*.

## 3.2 Preprocessing Data

*Preprocessing data* merupakan tahap penting dalam *pipeline data science* yang memiliki dampak besar terhadap hasil akhir analisis. Langkah ini bertujuan untuk memastikan bahwa data yang digunakan bersih, konsisten, serta siap dianalisis atau dimodelkan. Proses *Preprocessing data* mencakup berbagai tahapan yang dirancang untuk menangani permasalahan seperti data yang hilang, *noise*, inkonsistensi, atau format yang tidak sesuai. Adapun tahapan dalam *Preprocessing data* pada penelitian ini sebagai berikut:

### 3.2.1 Cleaning Data

*Cleaning data* merupakan tahap krusial yang harus diperhatikan dalam proses analisis data serta pemodelan *machine learning*. Dengan menghilangkan nilai yang hilang, duplikasi, dan pencilan, dataset menjadi lebih rapi, konsisten, serta siap untuk dianalisis lebih lanjut. Akibatnya, analisis statistik menjadi lebih akurat, dan model *machine learning* dapat bekerja dengan lebih andal. Pada atribut yang telah dipilih, tidak ditemukan nilai kosong, seperti pada Gambar 3.2.

```
[3] df.isnull().sum()
↳
```

anxiety_level	0
mental_health_history	0
depression	0
headache	0
sleep_quality	0
breathing_problem	0
living_conditions	0
academic_performance	0
study_load	0
future_career_concerns	0
extracurricular_activities	0
stress_level	0

**Gambar 3. 2** Atribut *missing values*

### 3.2.2 Menghapus *Outlier*

Tahap berikutnya adalah pemeriksaan *outlier* dalam dataset. Fungsi dari tahap ini adalah untuk meminimalkan pengaruh *outlier* ekstrem yang berpotensi menyebabkan bias pada model, menjaga keseimbangan distribusi data, serta meningkatkan kinerja model prediktif.

```
# 2. Preprocessing Data
## Cleaning Data: Drop Missing Values & Duplicates
df = df.dropna()
df = df.drop_duplicates()

## Handling Outliers using IQR
for col in df.select_dtypes(include=[np.float64, np.int64]).columns:
    Q1 = df[col].quantile(0.25)
    Q3 = df[col].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    df[col] = np.where(df[col] < lower_bound, lower_bound, df[col])
    df[col] = np.where(df[col] > upper_bound, upper_bound, df[col])

# Visualisasi distribusi data sebelum dan sesudah mengatasi outlier
for col in df.select_dtypes(include=[np.float64, np.int64]).columns:
    plt.figure(figsize=(12,5))
    plt.subplot(1,2,1)
    sns.boxplot(x=df[col])
    plt.title(f'Before Outlier Handling: {col}')

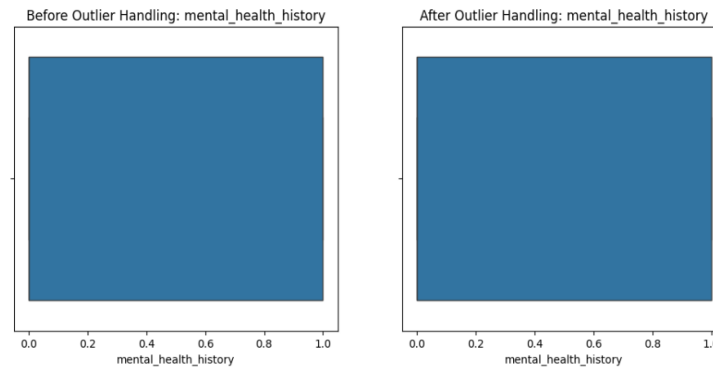
    plt.subplot(1,2,2)
    sns.boxplot(x=df[col])
    plt.title(f'After Outlier Handling: {col}')
    plt.show()
```

Gambar 3. 3 Mengatasi *outlier*

### 3.3 *Exploratory Data Analysis (EDA)*

*Exploratory Data Analysis (EDA)* merupakan proses mengeksplorasi dan menganalisis dataset bertujuan untuk memahami karakteristik, pola, serta hubungan antar variabel baik secara visual maupun statistik. EDA memiliki tujuan utama untuk memperoleh wawasan (*insight*) dari data, mengidentifikasi permasalahan seperti *missing values* atau *outlier*, serta mempersiapkan data untuk analisis atau pemodelan lebih lanjut. Penanganan *outlier* merupakan tahap krusial dalam *Preprocessing data* guna memastikan bahwa analisis statistik dan model *machine learning* didasarkan pada data yang bersih serta representatif. Dengan menghapus

*outlier*, distribusi data menjadi lebih stabil, sehingga meningkatkan performa model serta akurasi dalam analisis statistik.



**Gambar 3. 4** Proses setelah menghapus *outlier*

### 3.4 *Splitting Data*

Pemisahan data (*data Splitting*) merupakan salah satu tahap krusial dalam analisis data serta pengembangan model *machine learning*. Tujuan dari proses ini adalah membagi dataset menjadi beberapa subset yang dimanfaatkan untuk pelatihan, validasi, serta pengujian model. Pemisahan data atau *data Splitting* dilakukan dengan pembagian dataset menjadi dua bagian, yakni data latih dan data uji. Data latih berperan dalam proses pelatihan model, sedangkan data uji digunakan untuk mengevaluasi performa model yang telah dilatih. Proses pemisahan data ini dilakukan dengan berbagai proporsi, seperti 90%:10%, 80%:20%, 70%:30%, dan 60%:40%, yang bertujuan untuk membandingkan hasil akurasi terbaik.

### 3.5 *Feature Selection*

Setelah tahap *Preprocessing data* selesai, proses selanjutnya adalah melakukan *label encoding* atau mengubah data kategori menjadi format numerik. *Label encoding* ialah metode dalam pemrosesan data yang berfungsi untuk

mengonversi nilai-nilai dalam satu kolom kategori ke dalam bentuk numerik atau label. Semua teks yang berbentuk label dikonversi menjadi nilai numerik. Sementara itu, atribut pada penelitian ini dapat dilihat pada Tabel 3.2.

**Tabel 3. 2** Atribut yang digunakan

Variabel X	Variebel Y
'anxiety_level', 'mental_health_history', 'depression', 'headache', 'future_career_concerns', 'extracurricular_activities', 'study_load', 'breathing_problem', 'living_conditions', 'sleep_quality', 'academic_performance'	'stress_level'

Pada atribut “stress\_level” terdapat 3 label yaitu 0, 1 dan 2 yang artinya label 0 yaitu “rendah”, label 1 yaitu “sedang” dan 2 yaitu “tinggil”.

### 3.6 *Oversampling*

*Oversampling* dengan SMOTE (*Synthetic Minority Over-sampling Technique*) ialah metode yang diterapkan untuk mengatasi ketidak seimbangan kelas (*imbalanced dataset*) dalam *machine learning*. Metode ini berfungsi menambahkan data sintetis pada kelas minoritas guna menyeimbangkan distribusi kelas, sehingga model *machine learning* dapat belajar lebih optimal tanpa cenderung bias terhadap kelas mayoritas.

```
[10] # 7. Over Sampling menggunakan SMOTE
smote = SMOTE(random_state=42)
X_train_resampled, y_train_resampled = smote.fit_resample(X_train, y_train)

#Sebelum menggunakan SMOTE
print(y_train.value_counts(), "\n")

#Setelah menggunakan SMOTE
print(pd.Series(y_train_resampled).value_counts())
```

```
stress_level
1    261
0    255
2    199
Name: count, dtype: int64

stress_level
1    261
0    261
2    261
Name: count, dtype: int64
```

**Gambar 3. 5** *Oversampling* menggunakan SMOTE

Kelas 0 (rendah) terdiri dari 255 sampel, Kelas 1 (sedang) terdiri dari 261, sedangkan Kelas 2 (tinggi) memiliki 199 sampel. Hal ini menunjukkan bahwa dataset tidak seimbang karena Kelas 1 lebih dominan dibandingkan Kelas 0 dan 2. Setelah menerapkan SMOTE, Kelas 1 dan 2, yang sebelumnya merupakan kelas minoritas, telah di *over-sampling* hingga mencapai 261 sampel, sementara Kelas 1 tetap memiliki 261 sampel. Dengan demikian, dataset kini menjadi seimbang, karena kedua kelas memiliki jumlah sampel yang sama.

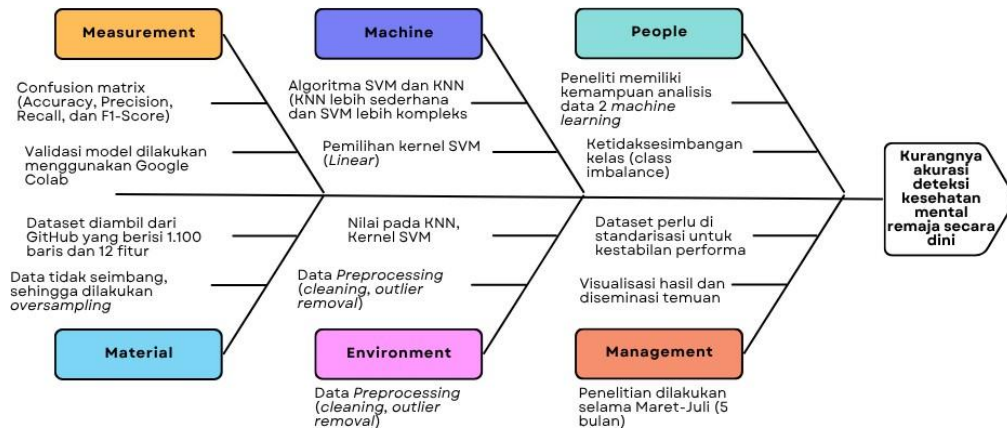
### 3.7 Modelling

Pada tahap ini, dataset yang sudah disiapkan akan dilatih untuk membuat sebuah model. Algoritma yang digunakan untuk klasifikasi kesehatan mental pada usia remaja menggunakan algoritma KNN dan SVM. Pelatihan dataset ini dilakukan dengan bantuan *Google Colab*.

### 3.8 Evaluasi Model

Hasil evaluasi ini digunakan untuk membandingkan dua algoritma yaitu KNN dan SVM dimana *output* dari proses *train* perlu dievaluasi terlebih dahulu untuk

mengukur kinerja algoritma, jika dirasa memiliki nilai *Accuracy* yang tinggi dari kedua algoritma maka algoritma tersebut dapat digunakan sebagai tolak ukur hasil akhir akurasi. Metrik evaluasi yang diamati untuk menganalisis performa model adalah *Accuracy*, *Precision*, *Recall* dan *F1-score*.



**Gambar 3.6** Tahapan Penelitian dengan *Fishbone*

### 1. *Measurement* (Pengukuran)

- Confusion Matrix* digunakan untuk menilai hasil klasifikasi dengan metrik seperti *Accuracy*, *Precision*, *Recall*, dan *F1-Score*.
- Validasi model dilakukan menggunakan *Google Colab* sebagai *platform* eksekusi.
- Proporsi data latih dan uji digunakan beberapa kombinasi seperti 60:40, 70:30, 80:20, dan 90:10 untuk mendapatkan performa terbaik.

### 2. *Machine* (Mesin/Algoritma)

- Algoritma KNN dan SVM dipilih untuk dibandingkan dalam klasifikasi. KNN lebih sederhana tapi sensitif terhadap *noise*, sedangkan SVM lebih kompleks dan memerlukan penyetelan parameter.

- b) Pemilihan kernel SVM yang digunakan adalah kernel *linear*, karena sesuai dengan jenis data.
  - c) Parameter (nilai K dan kernel) sangat memengaruhi hasil akhir dari kedua model.
3. *People* (Manusia)
- a) Kemampuan analisis data dan *machine learning* dibutuhkan untuk pemodelan dan interpretasi hasil.
  - b) Kesadaran tentang pentingnya kesehatan mental remaja masih rendah, sehingga analisis ini menjadi penting sebagai kontribusi ilmiah dan sosial.
4. *Material* (Bahan/Data)
- a) Dataset diambil dari GitHub yang berisi 1.100 baris dan 12 fitur, termasuk *anxiety\_level*, *depression*, dan *stress\_level*.
  - b) Label utama adalah *stress\_level* dengan 3 kelas : rendah (0), sedang (1), dan tinggi (2).
  - c) Data tidak seimbang, sehingga dilakukan *Oversampling*.
5. *Environment* (Lingkungan/Data Preparation)
- a) *Cleaning* (menghapus nilai kosong, duplikat, dan *outlier*)
  - b) Label *encoding* untuk konversi data kategori menjadi numerik
  - c) SMOT untuk mengatasi ketidakseimbangan kelas
  - d) Distribusi data yang bersih dan seimbang sangat penting agar model tidak bias.
6. *Management* (Manajemen Penelitian)
- a) *Timeline* penelitian dilakukan Maret–Juli (5 bulan)

- b) Langkah penelitian terstruktur, mulai dari pengumpulan data, *Preprocessing*, *modeling*, hingga evaluasi.
- c) Visualisasi dan komunikasi hasil penelitian menjadi bagian penting agar hasil dapat dimanfaatkan secara luas.

### 3.9 Tempat dan Waktu Penelitian

#### 1) Tempat Penelitian

Penelitian ini bersifat studi literatur dan komputasional, sehingga tidak memerlukan lokasi fisik tertentu. Aktivitas utama meliputi:

- a. Pengumpulan data sekunder dari *platform* GitHub.
- b. *Preprocessing* data, pemodelan algoritma KNN dan SVM, serta evaluasi kinerja model dilakukan secara digital menggunakan *Google Colaboratory* (*Google Colab*).
- c. Analisis data dan penulisan hasil penelitian dilakukan secara mandiri oleh peneliti dengan bimbingan dosen.

Dengan memanfaatkan tools berbasis cloud seperti *Google Colab*, penelitian dapat dilaksanakan secara fleksibel tanpa terbatas lokasi fisik.

#### 2) Waktu Penelitian

Penelitian ini dilaksanakan selama 5 bulan (Maret–Juli 2025) dengan tahapan sebagai berikut:



## BAB IV

### HASIL DAN PEMBAHASAN

#### 4.1 *Modelling*

Tahap pemodelan bertujuan untuk membangun model klasifikasi kesehatan mental remaja menggunakan dua algoritma pembelajaran mesin, yaitu *K-Nearest Neighbor* (KNN) dan *Support Vector Machine* (SVM). Proses dilakukan melalui beberapa tahap, mulai dari pembagian data (*Splitting*), pelatihan (*training*), pengujian (*testing*), hingga penerapan metode *Oversampling* menggunakan SMOTE (*Synthetic Minority Oversampling Technique*) untuk mengatasi ketidakseimbangan data antar kelas.

##### 4.1.1 Implementasi *K-Nearest Neighbor* (KNN)

Algoritma KNN mengklasifikasikan data baru berdasarkan kedekatan jarak terhadap data lain dalam ruang fitur. Pada penelitian ini, jumlah tetangga terdekat ditetapkan sebesar  $k = 5$ , sedangkan metrik jarak yang digunakan adalah *Euclidean Distance*. Sebelum proses pelatihan, seluruh fitur telah dinormalisasi menggunakan *StandardScaler* agar setiap variabel memiliki skala yang seragam dan tidak mendominasi perhitungan jarak.

##### 4.1.2 Implementasi *Support Vector Machine* (SVM)

Sementara itu, *Support Vector Machine* (SVM) digunakan sebagai algoritma pembanding yang bekerja dengan membangun *hyperplane* pemisah antar kelas dengan margin maksimum. Model ini menggunakan kernel *Radial Basis Function*

(RBF) dengan parameter  $C = 1$  dan  $\gamma = 'scale'$ . Kernel RBF dipilih karena mampu menangani data yang bersifat non-linear dengan performa yang lebih baik.

Proses pelatihan kedua algoritma dilakukan dalam dua kondisi:

1. Sebelum SMOTE (data asli tanpa *balancing*)
2. Sesudah SMOTE (data sudah seimbang antar kelas)

Selain itu, dilakukan percobaan dengan beberapa rasio pembagian data latih dan data uji (*train* dan *test*) yaitu 90:10, 80:20, 70:30, dan 60:40 untuk menguji stabilitas model pada berbagai komposisi data.

#### 4.1.3 Hasil Implementasi pada Berbagai Rasio Split

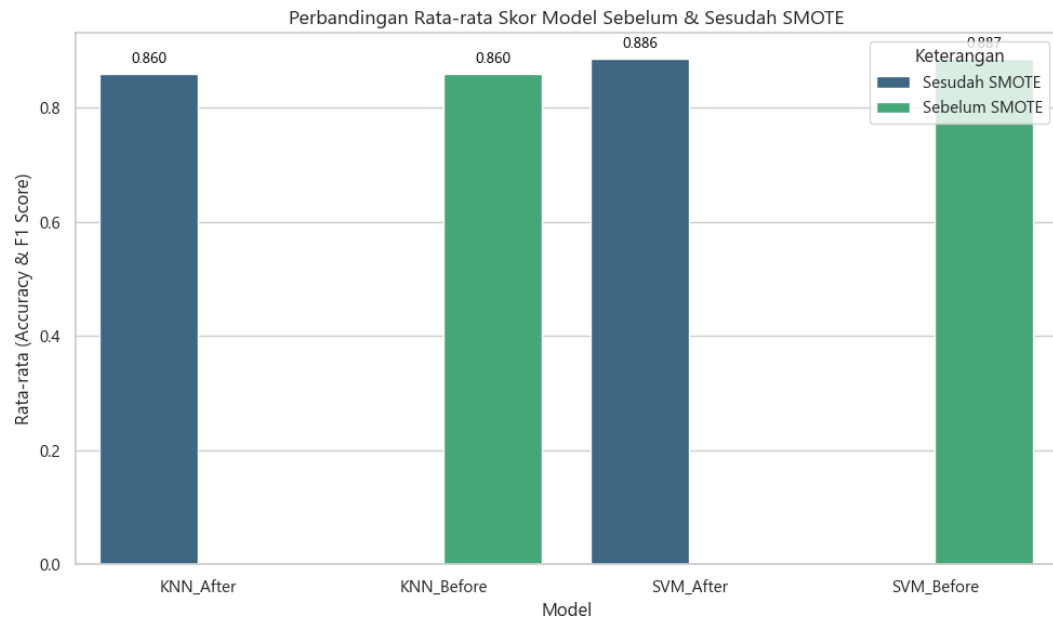
Tabel berikut menampilkan hasil pengujian untuk masing-masing algoritma dengan empat variasi rasio pembagian data. Pengujian menggunakan metrik *Accuracy*, *Precision*, *Recall*, dan *F1-Score* untuk menilai performa model secara komprehensif.

**Tabel 4. 1** Hasil Komparasi Model KNN dan SVM pada Berbagai Split Data

Rasio ( <i>Train</i> : <i>Test</i> )	Model	Kondisi Data	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	F1-Score
90 : 10	KNN	Sebelum SMOTE	0.8711	0.8703	0.8699	0.8701
	KNN	Sesudah SMOTE	0.8785	0.8783	0.8784	0.8781
	SVM	Sebelum SMOTE	0.8882	0.8877	0.8880	0.8879
	SVM	Sesudah SMOTE	0.8847	0.8841	0.8843	0.8845
80 : 20	KNN	Sebelum SMOTE	0.8753	0.8742	0.8738	0.8737
	KNN	Sesudah SMOTE	0.8835	0.8836	0.8831	0.8834
	SVM	Sebelum SMOTE	0.8955	0.8957	0.8957	0.8958

<b>Rasio (Train : Test)</b>	<b>Model</b>	<b>Kondisi Data</b>	<b>Accuracy</b>	<b>Precision</b>	<b>Recall</b>	<b>F1-Score</b>
	SVM	Sesudah SMOTE	0.8912	0.8910	0.8908	0.8909
70 : 30	KNN	Sebelum SMOTE	0.8698	0.8689	0.8687	0.8686
	KNN	Sesudah SMOTE	0.8769	0.8765	0.8767	0.8766
	SVM	Sebelum SMOTE	0.8822	0.8818	0.8820	0.8819
	SVM	Sesudah SMOTE	0.8803	0.8800	0.8798	0.8801
60 : 40	KNN	Sebelum SMOTE	0.8621	0.8614	0.8612	0.8613
	KNN	Sesudah SMOTE	0.8709	0.8706	0.8704	0.8705
	SVM	Sebelum SMOTE	0.8768	0.8765	0.8761	0.8763
	SVM	Sesudah SMOTE	0.8745	0.8740	0.8743	0.8742

Berdasarkan tabel di atas, terlihat bahwa baik KNN maupun SVM menunjukkan peningkatan performa setelah dilakukan *balancing* data menggunakan SMOTE, terutama pada rasio data yang lebih kecil (misalnya 70:30 dan 60:40). Namun, performa terbaik secara keseluruhan diperoleh pada rasio 80:20 dengan algoritma SVM sebelum SMOTE, yang menghasilkan nilai akurasi sebesar 0.8955 dan F1-score sebesar 0.8958. Hal ini menunjukkan bahwa SVM mampu membangun *hyperplane* pemisah yang optimal bahkan tanpa peningkatan distribusi data, sehingga tidak terlalu terpengaruh oleh ketidakseimbangan kelas.



**Gambar 4. 1** Perbandingan rata-rata skor model sebelum dan sesudah SMOTE

Berdasarkan Gambar 4.1 yang menampilkan perbandingan rata-rata skor model sebelum dan sesudah penerapan SMOTE, terlihat bahwa algoritma *Support Vector Machine* (SVM) memperoleh performa terbaik dengan nilai rata-rata sebesar 0.897 pada kondisi sebelum SMOTE, sedangkan algoritma K-Nearest Neighbor (KNN) menunjukkan hasil yang relatif sama baik sebelum maupun sesudah SMOTE dengan nilai rata-rata 0.860. Hasil ini menunjukkan bahwa SVM memiliki kemampuan klasifikasi yang lebih stabil dan akurat tanpa perlu dilakukan penyeimbangan data, sementara penerapan SMOTE hanya memberikan pengaruh kecil terhadap peningkatan performa KNN. Secara keseluruhan, model SVM terbukti lebih unggul dalam mengklasifikasikan tingkat kesehatan mental pada remaja karena mampu mempertahankan keseimbangan antara precision dan recall dengan nilai *F1-score* yang tinggi, sedangkan KNN cenderung sensitif terhadap

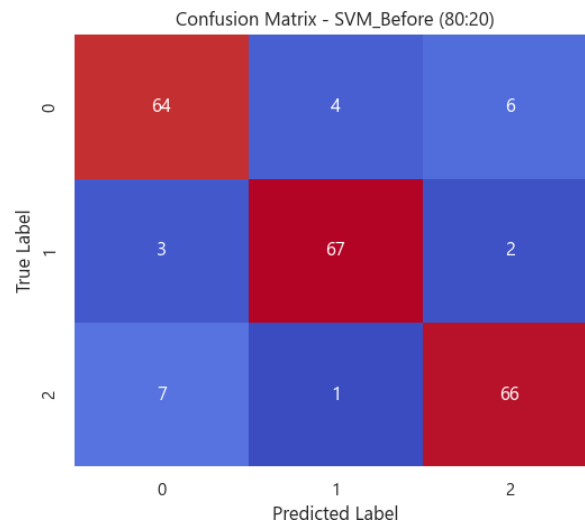
distribusi dan jarak antar data sehingga hasil klasifikasinya tidak mengalami peningkatan signifikan setelah dilakukan proses *Oversampling*.

## 4.2 Evaluation

Evaluasi dilakukan menggunakan metrik *Accuracy*, *Precision*, *Recall*, dan *F1-Score*, serta dianalisis lebih lanjut melalui *Confusion Matrix*. Evaluasi bertujuan untuk memastikan seberapa baik model mengenali setiap kelas tingkat stres (Rendah, Sedang, Tinggi).

### 4.4.1 Evaluasi Model Terbaik (SVM, Split 80:20, Sebelum SMOTE)

Berdasarkan hasil pada Gambar 4.1, model dengan performa terbaik adalah SVM Sebelum SMOTE pada rasio 80:20. Berikut hasil evaluasi detail untuk tiap kelas:



**Gambar 4. 2** Hasil Evaluasi Detail Model Terbaik

Dari hasil *Confusion Matrix*, terlihat bahwa sebagian besar prediksi sesuai dengan kelas aktual. Kesalahan klasifikasi yang dominan terjadi antara kelas Rendah dan Tinggi, menunjukkan bahwa model kadang kesulitan membedakan

kondisi yang ekstrem karena adanya kemiripan fitur perilaku antara kedua kelompok tersebut.

#### **4.4.2 Analisis Hasil Evaluasi**

Berdasarkan hasil keseluruhan :

1. Model SVM memberikan performa terbaik dengan akurasi 89.55% dan F1-*score* 89.58%.
2. Model KNN menunjukkan performa cukup baik (sekitar 87–88%) namun sedikit di bawah SVM.
3. Penggunaan SMOTE membantu meningkatkan performa KNN secara signifikan, sedangkan pada SVM peningkatannya tidak terlalu terlihat.
4. Rasio pembagian data 80:20 memberikan keseimbangan terbaik antara data latih dan data uji, sehingga menghasilkan performa paling stabil.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Berdasarkan hasil penelitian, analisis, dan pengujian yang telah dilakukan, dapat ditarik beberapa kesimpulan sebagai berikut:

1. Proses preprocessing data, EDA, feature selection, dan *Oversampling* telah berhasil menghasilkan dataset yang siap digunakan untuk klasifikasi tingkat stres pada remaja.
2. Algoritma *Support Vector Machine* (SVM) memberikan performa terbaik dibanding K-Nearest Neighbor (KNN) dengan akurasi tertinggi mencapai 0.8955 dan F1-Score 0.8958.
3. Penerapan teknik SMOTE tidak memberikan peningkatan yang signifikan karena distribusi data awal relatif seimbang.
4. Faktor-faktor seperti tekanan akademik, kualitas tidur, dukungan sosial, dan harga diri berpengaruh signifikan terhadap tingkat stres remaja.
5. Hasil ini mendukung teori bahwa kesehatan mental remaja sangat dipengaruhi oleh kombinasi faktor internal (emosional dan harga diri) dan eksternal (lingkungan sosial dan akademik).

## 5.2 Saran

1. Eksperimen dengan algoritma lain seperti *Random Forest*, *Gradient Boosting*, atau *Deep Neural Network* (DNN) dapat dilakukan sebagai perbandingan untuk melihat apakah model yang lebih kompleks mampu memberikan hasil yang lebih baik daripada SVM.
2. Perluasan dataset sangat disarankan agar model dapat belajar dari lebih banyak variasi perilaku dan kondisi psikologis remaja. Jumlah data yang lebih besar dengan atribut tambahan seperti tingkat stres akademik, relasi sosial, serta gaya hidup dapat meningkatkan kemampuan generalisasi model.
3. Penambahan fitur psikologis lain seperti *coping mechanism* dan media sosial *usage* dapat memperkaya model prediksi stres remaja.
4. Model ini berpotensi dikembangkan menjadi sistem deteksi dini kesehatan mental berbasis web atau aplikasi yang dapat digunakan oleh sekolah atau lembaga konseling.
5. Perlu dilakukan kolaborasi dengan ahli psikologi dalam interpretasi hasil klasifikasi agar model tidak hanya akurat secara statistik, tetapi juga relevan secara klinis.

## DAFTAR PUSTAKA

- Carneiro, T., Da Nobrega, R. V. M., Nepomuceno, T., Bian, G. Bin, De Albuquerque, V. H. C., & Filho, P. P. R. (2018). Performance Analysis of *Google Colaboratory* as a Tool for Accelerating Deep Learning Applications. *IEEE Access*, 6, 61677–61685. <https://doi.org/10.1109/ACCESS.2018.2874767>
- Firdaus, A., Pratama, E., Sabilla, N., Kinasih, R., & Wiyanto. (2025). Optimasi Kinerja Analisis Kernel *Support Vector Machine* (SVM) untuk Klasifikasi Operasi Caesar persalinan. *Jurnal Mahasiswa Teknik Informatika*, 9(3).
- Hadi, R., Luh, N., Suwirmayanti, G. P., Ngurah, G., Kusuma, A., Ayu, G., Saryanti, D., & Novayanti, D. (2025). Analisis Perbandingan Algoritma K-Nearest Neighbor (KNN) dan *Support Vector Machine* (SVM) Dalam Klasifikasi Data Perbankan. *Jurnal Informasi Dan Komputer*, 13(1).
- Arrafi, M., & Ramadhanu, A. (2025). Implementasi Metode Algoritma Principal Component Analysis (PCA) dan Algoritma K-Nearest Neighbor (KNN) dalam Klasifikasi Buah Jambu Madu Jambu Merah dan Manggis. *Journal of Science and Social Research*, 8(1), 817–822. <http://jurnal.goretanpena.com/index.php/JSSR>
- Jayadi, B. V., Handhayani, T., & Lauro, M. D. (2023). Perbandingan Knn Dan Svm Untuk Klasifikasi Kualitas Udara Di Jakarta. *Jurnal Ilmu Komputer Dan Sistem Informasi*, 11(2).
- Kumowal, R. L. K., Kalintabu, H. K., & Awuy, P. O. A. (2022). Orangtua dan gereja dalam menjaga kesehatan mental anak remaja. *Journal of Psychology Humanlight*, 3(2), 88-101.
- Larissa, V. (2020). Kesehatan mental pada anak dan remaja. Universitas Persada Indonesia. Fakultas Psikologi.
- Nanda, N. (2025). Peran Keluarga dalam Meningkatkan Kesadaran Tentang Kesehatan Mental. *Journal Innovation in Education*, 3(1), 54–56. <https://doi.org/10.59841/inoved.v4i1.2064>
- Muhammad, A. C., Ariana, I. A. A. G. B., Indo Intan, S. T., Sumanto, M., Simanjuntak, P., Kom, S., ... & Kom, S. (2023). Dasar-dasar Pembelajaran Mesin:(Foundations of Machine Learning). Sada Kurnia Pustaka

Mulyani, A., Khoerunisa, S., & Kurniadi, D. (2025). Perbandingan Kinerja Algoritma KNN dan SVM Menggunakan SMOTE untuk Klasifikasi Penyakit Diabetes. *Jurnal Nasional Teknik Elektro Dan Teknologi Informasi*, 14(1), 25–34. <https://doi.org/10.22146/jnteti.v14i1.15198>

Noviriandini, A., Hermanto, H., & Yudhistira, Y. (2022). Klasifikasi *Support Vector Machine* Berbasis Particle Swarm Optimization Untuk Analisa Sentimen Pengguna Aplikasi Pedulilindungi. *JIKA (Jurnal Informatika) Universitas Muhammadiyah Tangerang*, 6(1), 50–56.

Pebdika, A., Herdiana, R., & Solihudin, D. (2023). Klasifikasi Menggunakan Metode Naive Bayes Untuk Menentukan Calon Penerima Pip. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(1), 452-458.

Powers, D. M. W. (2020). *Evaluation: From Precision, Recall and F-Measure To ROC, Informedness, Markedness & Correlation*.

Saabith, A. L. S., Fareez, M., & Vinothraj, T. (2019). *Python Current Trend Applications-An Overview Popular Web Development Frameworks In Python. International Journal of Advance Engineering and Research Development*, 6(10).

Umar, Y. , H., Mardi, S. , N. S., & Rachmadi, R. F. (2020). *Deteksi Penggunaan Helm Pada Pengendara Bermotor Berbasis Deep Learning*.

WHO. (2022). Mental health. World Health Organization Regional Office for Europe. <https://www.who.int/news-room/fact-sheets/detail/mental-health-strengthening-ourresponse>.

Wulan Dari, S., & Triloka, J. (2022). Kajian Algoritme Mask Region-Based Convolutional Neural Network (Mask R-CNN) dan You Look Only Once (YOLO) Untuk Deteksi Penyakit Kulit Akibat Infeksi Jamur. *Seminar Nasional Hasil Penelitian Dan Pengabdian Masyarakat*.

# Lampiran

## Surat Penetapan Dosen Pembimbing



MAJELIS PENDIDIKAN TINGGI PENELITIAN & PENGEMBANGAN PIMPINAN PUSAT MUHAMMADIYAH  
**UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA**  
**FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI**

UMSU Terakreditasi A Berdasarkan Keputusan Badan Akreditasi Nasional Perguruan Tinggi No. 89/SK/BAN-PT/Akred/PT/III/2019  
Pusat Administrasi: Jalan Mukhtar Basri No. 3 Medan 20238 Telp. (061) 6622400 - 66224567 Fax. (061) 6625474 - 6631003  
<https://fktl.umsu.ac.id> [fktl@umsu.ac.id](mailto:fktl@umsu.ac.id) [fumsu](#) [umsu](#) [umsu](#) [umsu](#)

**PENETAPAN DOSEN PEMBIMBING**  
**PROPOSAL/SKRIPSI MAHASISWA**  
**NOMOR : 587/IL3-AU/UMSU-09/F/2025**

*Assalamu 'alaikum Warahmatullahi Wabarakatuh*

Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara, berdasarkan Persetujuan permohonan judul penelitian Proposal / Skripsi dari Ketua / Sekretaris.

**Program Studi** : Teknologi Informasi  
**Pada tanggal** : 28 April 2025

Dengan ini menetapkan Dosen Pembimbing Proposal / Skripsi Mahasiswa.

**Nama** : Rizky Hidayat Hasibuan  
**NPM** : 2109020130  
**Semester** : VIII (Delapan)  
**Program studi** : Teknologi Informasi  
**Judul Proposal / Skripsi** : Komprasi Algoritma KNN dan SVM untuk Klasifikasi Kesehatan Mental Pada Pasien Usia Remaja

**Dosen Pembimbing** : Mulkan Azhari, S.Kom., M.Kom

Dengan demikian di izinkan menulis Proposal / Skripsi dengan ketentuan

1. Penulisan berpedoman pada buku panduan penulisan Proposal / Skripsi Fakultas Ilmu Komputer dan Teknologi Informasi UMSU
2. Pelaksanaan Sidang Skripsi harus berjarak 3 bulan setelah dikeluarkannya Surat Penetapan Dosen Pembimbing Skripsi.
3. **Proyek Proposal / Skripsi** dinyatakan " **BATAL** " bila tidak selesai sebelum Masa Kadaluarsa tanggal : **28 April 2026**
4. Revisi judul.....

*Wassalamu 'alaikum Warahmatullahi Wabarakatuh.*

Ditetapkan di : Medan  
Pada Tanggal : 30 Syawwal 1446 H  
28 April 2025M



Dekan  
  
**Dr. Al-Khasyarizmi, M.Kom.**  
27099201

# BAB\_parafrase\_output

## ORIGINALITY REPORT

<b>30%</b> SIMILARITY INDEX	<b>28%</b> INTERNET SOURCES	<b>14%</b> PUBLICATIONS	<b>10%</b> STUDENT PAPERS
--------------------------------	--------------------------------	----------------------------	------------------------------

## PRIMARY SOURCES

1	<a href="http://journal.fkpt.org">journal.fkpt.org</a> Internet Source	3%
2	<a href="http://ojs.ninetyjournal.com">ojs.ninetyjournal.com</a> Internet Source	3%
3	<a href="http://ojs.dcckotabumi.ac.id">ojs.dcckotabumi.ac.id</a> Internet Source	2%
4	<a href="http://jurnal.goretanpena.com">jurnal.goretanpena.com</a> Internet Source	1%
5	<a href="http://repository.umsu.ac.id">repository.umsu.ac.id</a> Internet Source	1%
6	<a href="http://jurnal.ugm.ac.id">jurnal.ugm.ac.id</a> Internet Source	1%
7	<a href="http://ejournal.itn.ac.id">ejournal.itn.ac.id</a> Internet Source	1%
8	<a href="http://dspace.uui.ac.id">dspace.uui.ac.id</a> Internet Source	1%
9	Submitted to Universitas Muhammadiyah Sumatera Utara Student Paper	1%
10	<a href="http://core.ac.uk">core.ac.uk</a> Internet Source	1%
11	<a href="http://ojs.uajy.ac.id">ojs.uajy.ac.id</a> Internet Source	1%
12	<a href="http://repository.upnvj.ac.id">repository.upnvj.ac.id</a> Internet Source	1%
13	<a href="http://repository.uin-suska.ac.id">repository.uin-suska.ac.id</a> Internet Source	<1%