PREDIKSI KESEHATAN MENTAL MENGGUNAKAN MACHINE LEARNING

THESIS

Oleh :

M. SAYYIDUL ADNAN

NIM. 230605210008

PROGRAM STUDI MAGISTER INFORMATIKA

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM

MALANG

2026

PREDIKSI KESEHATAN MENTAL MENGGUNAKAN MACHINE LEARNING

THESIS

Diajukan kepada:
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Magister Komputer (M.Kom)

Oleh :
M. SAYYIDUL ADNAN
NIM. 230605210008

PROGRAM STUDI MAGISTER INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2026

HALAMAN PERSETUJUAN

PREDIKSI KESEHATAN MENTAL MENGGUNAKAN MACHINE LEARNING

THESIS

Oleh :
M. SAYYIDUL ADNAN
NIM. 230605210008

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 12 Mei 2026

Pembimbing I,	Pembimbing II,
Dr. Cahyo Crysdian, M.Cs	Dr. Irwan Budi Santoso, M.Kom
NIP. 19740424 200901 1 008	NIP. 19770103 201101 1 004

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Prof. Dr. H. Muhammad Faisal, M.T
NIP. 19740510 200501 1 007

HALAMAN PENGESAHAN

PREDIKSI KESEHATAN MENTAL MENGGUNAKAN MACHINE LEARNING

THESIS

Oleh :
M. SAYYIDUL ADNAN
NIM. 230605210008

Telah Dipertahankan di Depan Dewan Penguji Thesis
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Magister Komputer (M.Kom)
Tanggal: 12 Mei 2026

Susunan Dewan Penguji

Penguji I	: Dr. M. Ainul Yaqin, M.Kom NIP. 19761013 200604 1 004	( )
Penguji II	: Dr. Totok Chamidy, M.Kom NIP. 19691222 200604 1 001	( )
Pembimbing I	: Dr. Cahyo Crysdian, M.Cs NIP. 19740424 200901 1 008	( )
Pembimbing II	: Dr. Irwan Budi Santoso, M.Kom NIP. 19770103 201101 1 004	( )

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Prof. Dr. Ir. Muhammad Faisal, M.T
NIP. 19740510 200501 1 007

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama	: M. SAYYIDUL ADNAN
NIM	: 230605210008
Fakultas / Jurusan	: Sains dan Teknologi / Magister Informatika
Judul Skripsi	: PREDIKSI KESEHATAN MENTAL MENGGUNAKAN MACHINE LEARNING.

Menyatakan dengan sebenarnya bahwa Thesis yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 12 Mei 2026
Yang membuat pernyataan,

M. SAYYIDUL ADNAN
NIM.230605210008

MOTTO

لَا تَخَفْ وَلَا تَحْزَنْ إِنَّ اللَّهَ مَعَنَا

"Jangan takut dan jangan bersedih, sesungguhnya Allah bersama kita"

(QS. At-Taubah ayat 40)

وَاتَّقُوا اللّٰهَ وَيُعَلِّمُكُمُ اللّٰهُ وَاللّٰهُ بِكُلِّ شَيْءٍ عَلِيْمٌ

“Bertakwalah kepada Allah, Allah memberikan pengajaran kepadamu dan Allah Maha Mengetahui segala sesuatu.”

(QS. Al-Baqarah ayat 283)

HALAMAN PERSEMBAHAN

Dengan segenap rasa syukur *alhamdulillahi rabbil 'alamin*, saya persembahkan hasil karya Thesis ini kepada:

1. Kedua orang tua saya, yakni Bapak Yahya Saman, S.Ag. dan Ibu Jarkiah, S.Ag. yang telah membimbing, mendoakan, dan menyayangi saya dari awal saya berada di dunia hingga saat ini dan kelak nanti. Segala pengorbanan dan kasih sayang yang telah diberikan menjadi kekuatan terbesar dalam setiap langkah perjalanan saya.
2. Dosen saya, Bapak Dr. Cahyo Crysdian, M.Cs., yang telah membimbing saya sejak skripsi hingga tesis ini. Terima kasih atas ilmu dan wawasan yang membangun pola pikir saya, baik dalam akademik maupun kehidupan.
3. Keluarga besar IBS Al-Hamra Malang dan Educourse.id, tempat saya mengabdikan diri sebagai pengajar di sela-sela masa studi. Terima kasih atas kepercayaan, dukungan, dan pengalaman berharga yang telah membentuk saya menjadi pribadi yang lebih baik.
4. Sahabat-sahabat seperjuangan yang telah menjadi rekan belajar dan saling mendukung sepanjang perjalanan ini. Terima kasih atas canda tawa, semangat, dan dukungan yang selalu hadir di setiap momen.

Semoga Allah senantiasa membimbing kita ke jalan yang lurus dan memudahkan setiap langkah dalam mewujudkan mimpi-mimpi kita. Aamiin ya Rabbal 'Alamiin.

KATA PENGANTAR

Assalamualaikum Warahmatullahi Wabarakatuh.

Segala puji hanya milik Allah Subhanahu Wa Ta'ala atas segala nikmat dan kasih sayang-Nya yang telah memudahkan penulis untuk menyelesaikan Thesis ini. Semoga shalawat dan salam senantiasa terlimpah kepada Nabi Muhammad Sallalahu 'Alaihi wa Sallam. Dan semoga kita semua mendapat syafaatnya di hari kiamat nanti. Aamiin.

Penulis mengucapkan rasa syukur dan terima kasih yang tak terhingga kepada semua pihak-pihak yang selalu memberikan bantuan dan motivasi kepada penulis untuk dapat menyelesaikan Thesis ini. Ucapan ini penulis sampaikan kepada:

1. Prof. Dr. Hj. Ilfi Nur Diana, M.Si., CAHRM., CRMP., selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. Agus Mulyono, M.Kes., selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Prof. Dr. Ir. Muhammad Faisal, M.T., selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Dr. Cahyo Crysdian, M.Cs. selaku Dosen Pembimbing I dan Dr. Irwan Budi Santoso, M.Kom. selaku Dosen Pembimbing II yang telah memberikan bantuan, bimbingan, dan arahan kepada penulis sehingga bisa menuntaskan Thesis ini.
5. Dr. M. Ainul Yaqin, M.Kom. selaku Dosen Penguji I dan Dr. Totok Chamidy, M.Kom. selaku Dosen Penguji II yang telah menguji serta memberikan masukan sehingga penulis dapat menuntaskan Thesis dengan baik.

6. Segenap Dosen, Admin, dan Mahasiswa Program Studi Magister Informatika yang telah memberikan banyak dukungan dan bimbingan selama pengerjaan Thesis ini.
7. Bapak Yahya Saman, S.Ag. dan Ibu Jarkiah, S.Ag. selaku orang tua penulis yang selalu memberikan dukungan dan motivasi untuk terus berusaha, serta doa yang tak putus-putusnya selalu disampaikan agar dapat menuntaskan Thesis ini dengan lancar dan baik.
8. Keluarga besar IBS Al-Hamra Malang dan Educourse.id yang telah memberikan kesempatan dan kepercayaan kepada penulis untuk mengabdikan diri sebagai pengajar di sela-sela masa studi.
9. Semua pihak yang tidak dapat disebutkan satu per satu, termasuk rekan-rekan yang telah memberikan kontribusi, saran, dan dukungan dalam perjalanan penulisan Thesis ini.

Akhir kata, penulis mengakui bahwa penulisan pada Thesis ini masih banyak kekurangan. Saya berharap semoga Thesis ini diterima sebagai amal ibadah yang tulus dan bermanfaat di sisi Allah Subhanahu Wa Ta'ala. Semoga karya ini menjadi bagian dari kontribusi yang tak terputus dalam rangka memperkuat dan mengembangkan ilmu pengetahuan, serta melaksanakan tugas sebagai hamba Allah yang berkomitmen.

Wassalamualaikum Warahmatullahi Wabarakatuh.

Malang, Mei 2026

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

ABSTRAK

Adnan, M. Sayyidul. 2026. **Prediksi Kesehatan Mental Menggunakan Machine Learning**. Tesis. Program Studi Magister Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Cahyo Crysdian, M.Cs. (II) Dr. Irwan Budi Santoso, M.Kom.

Kata kunci: *Machine Learning, Prediksi Kesehatan Mental, Logistic Regression, SVM, Random Forest, XGBoost, Ensemble Stacking, Ensemble Majority Voting, Efesiensi Komputasi, Cyclomatic Complexity, Skor Komposit*.

Tingginya prevalensi gangguan kesehatan mental di kalangan mahasiswa menegaskan pentingnya deteksi dini berbasis data. Penelitian ini mengevaluasi metode machine learning yang paling optimal untuk memprediksi risiko depresi mahasiswa berdasarkan performa klasifikasi dan efisiensi komputasi. Dataset yang digunakan adalah "Student Mental Health in a Multicultural Environment" dari lingkungan akademik Jepang, terdiri dari 27 fitur prediktor mencakup data demografis, kemampuan bahasa, keterlibatan keagamaan, riwayat ideasi bunuh diri, skor keterhubungan sosial (ToSC), tujuh subdimensi stres akulturatif (APD, AHome, APH, AFear, ACS, AGuilt, AMiscell), dukungan sosial, dan skor total PHQ-9 (ToDep), dengan variabel target status depresi mahasiswa (Dep: Yes/No) berdasarkan instrumen PHQ-9. Empat model tunggal (Logistic Regression/LR, SVM, Random Forest, XGBoost) dan dua model ensemble (Majority Voting dan Stacking) dievaluasi menggunakan 5-Fold Stratified Cross-Validation, Paired T-Test, dan lima metrik efisiensi komputasi: waktu training, waktu inferensi, RAM puncak, rata-rata CPU, dan Cyclomatic Complexity (CC), yang diintegrasikan dalam Skor Komposit berbasis normalisasi Min-Max dengan bobot setara pada sembilan dimensi. Ensemble Stacking mencapai akurasi sebesar 72,41% dan presisi sebesar 72,02%. Namun, Ensemble Stacking menunjukkan *recall* terendah sebesar 38,47%, akibat Conservative Bias. XGBoost menunjukkan kinerja yang lebih unggul dalam hal *recall*, dengan persentase 48,89%, dan dalam metrik F1-Score, mencapai skor 52,83%. Dalam hal efisiensi, LR menunjukkan tingkat efektivitas tertinggi, dengan waktu pelatihan 0,056 detik, penggunaan RAM 0,10 MB, penggunaan CPU 10%, dan CC = 3. Sebaliknya, Ensemble Stacking menunjukkan efisiensi yang jauh lebih rendah dibandingkan dengan semua model lainnya. Analisis menunjukkan bahwa, berdasarkan skor komposit, LR menunjukkan kinerja optimal (0,7181), sedangkan Ensemble Stacking menunjukkan kinerja yang paling tidak optimal (0,3172). LR direkomendasikan untuk diterapkan dalam pengaturan dengan sumber daya terbatas, sedangkan Ensemble Stacking disarankan untuk skrining klinis formal. Studi ini menyajikan kerangka evaluasi komprehensif yang selaras dengan misi *Hifz an-Nafs* dalam *maqashid syariah*.

ABSTRACT

Adnan, M. Sayyidul. 2026. ***Mental Health Prediction Using Machine Learning.*** Thesis. Master of Informatics Study Program, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Supervisors: (I) Dr. Cahyo Crysdian, M.Cs. (II) Dr. Irwan Budi Santoso, M.Kom.

The high prevalence of mental health disorders among college students underscores the importance of data-driven early detection. This study evaluates the most optimal machine learning method for predicting the risk of depression among college students based on classification performance and computational efficiency. The dataset used is "Student Mental Health in a Multicultural Environment" from a Japanese academic setting, consisting of 27 predictor features covering demographic data, language proficiency, religious engagement, history of suicidal ideation, social connectedness scores (ToSC), seven sub-dimensions of acculturative stress (APD, AHome, APH, AFear, ACS, AGuilt, AMiscell), social support, and total PHQ-9 scores (ToDep), with the target variable being the student's depression status (Dep: Yes/No) based on the PHQ-9 instrument. Four single models (Logistic Regression/LR, SVM, Random Forest, XGBoost) and two ensemble models (Majority Voting and Stacking) were evaluated using 5-Fold Stratified Cross-Validation, Paired T-Test, and five computational efficiency metrics: training time, inference time, peak RAM, average CPU usage, and Cyclomatic Complexity (CC), which were integrated into a Min-Max normalized Composite Score with equal weights across the nine dimensions. Ensemble Stacking achieved an accuracy of 72.41% and a precision of 72.02%. However, Ensemble Stacking exhibited the lowest recall at 38.47%, due to Conservative Bias. XGBoost demonstrated superior performance in terms of recall, with a percentage of 48.89%, and in the F1-Score metric, achieving a score of 52.83%. In terms of efficiency, LR demonstrated the highest level of effectiveness, with a training time of 0.056 seconds, RAM usage of 0.10 MB, CPU usage of 10%, and CC = 3. Conversely, Ensemble Stacking showed significantly lower efficiency compared to all other models. Analysis shows that, based on the composite score, LR demonstrates optimal performance (0.7181), while Ensemble Stacking demonstrates the least optimal performance (0.3172). LR is recommended for implementation in resource-constrained settings, while Ensemble Stacking is recommended for formal clinical screening. This study presents a comprehensive evaluation framework aligned with the mission of Hifz an-Nafs within the maqashid al-sharia.

Keywords: *Machine Learning, Mental Health Prediction, Logistic Regression, SVM, Random Forest, XGBoost, Ensemble Stacking, Ensemble Majority Voting Computational Efficiency, Cyclomatic Complexity, Composite Score.*

مستخلص البحث

العدنان، م. سيد. 2026. التنبؤ بالصحة النفسية باستخدام التعلم الآلي. رسالة الماجستير. قسم المعلومات، كلية العلوم والتكنولوجيا بجامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف الأول: د. جاهيو كريسديان، الماجستير؛ المشرف الثاني: د. إروان بودي سانتوسو، الماجستير.

الكلمات الرئيسية: تعلم آلي، تنبؤ بصحة نفسية، انحدار لوجستي، SVM، غابة عشوائية، XGBoost، تجميع تراصي، تجميع تصويت أغلبية، كفاءة حوسبة، تعقيد سيكلوماتي، فارق معياري.

ارتفاع انتشار اضطرابات الصحة النفسية بين الطلاب يؤكد أهمية الكشف المبكر القائم على البيانات. تقيم هذه الدراسة طرق التعلم الآلي الأكثر فعالية للتنبؤ بمخاطر الاكتئاب لدى الطلاب بناءً على أداء التجميع وكفاءة الحوسبة. مجموعة البيانات المستخدمة هي "الصحة النفسية للطلاب في بيئة متعددة الثقافات" من البيئة الأكاديمية في اليابان، وتتكون من 27 ميزة تنبؤية تشمل البيانات الديموغرافية، مهارات اللغة، المشاركة الدينية، تاريخ التفكير الانتحاري، درجة التواصل الاجتماعي (ToSC)، سبعة أبعاد فرعية للتوتر الثقافي؛ التمرد الثقافي، العزلة الثقافية، الاغتراب عن الذات، العزلة الاجتماعية، غياب المعنى، غياب المعايير، والشعور بالعجز (APD، AHome، APH، AFear، ACS، AGuilt، Amiscell)، الدعم الاجتماعي، والمجموع الكلي لمقياس (ToDep) PHQ-9، مع المتغير المستهدف وهو حالة الاكتئاب لدى الطلاب (اكتئاب: نعم أو لا) بناءً على أداة PHQ-9. تم تقييم أربعة نماذج فردية (الانحدار اللوجستي/LR، SVM، الغابة العشوائية، XGBoost) ونموذجين جماعيين (تصويت أغلبية وتراصي) باستخدام التحقق المتقاطع المتدرج ذو 5 طيات، واختبار ت المزدوج، وخمسة مقاييس لكفاءة الحوسبة: وقت التدريب، وقت الاستدلال، ذروة استخدامRAM ، متوسط استخدام CPU، وتعقيد سيكلوماتي (CC)، والتي تم دمجها في فارق معياري يعتمد على التطبيع أقل-أعلى مع وزن متساوٍ على الأبعاد التسعة. حقق نموذج التجميع التراصي دقة بنسبة 72.41٪ وثبات بنسبة 72.02٪. ومع ذلك، أظهر هذا النموذج أدنى نسبة الاسترجاع بنسبة 38.47٪، نتيجة التحفظ في الانحياز. أظهر XGBoost أداءً أفضل من حيث الاسترجاع، بنسبة 48.89٪، وفي مقياس درجة ف1، حيث وصلت إلى 52.83٪. من حيث الكفاءة، أظهر الانحدار اللوجستي أعلى مستوى من الفعالية، مع وقت تدريب 0.056 ثانية. استخدام ذاكرة الوصول العشوائي 0,10 ميغابايت، واستخدام وحدة المعالجة المركزية 10٪، و CC = 3. على العكس من ذلك، يُظهر التجميع التراصي كفاءة أقل بكثير مقارنة بجميع النماذج الأخرى. تُظهر التحليلات أنه، بناءً على الفارق المعياري، يُظهر LR أداءً مثاليًا (0,7181)، بينما يُظهر التجميع التراصي أقل أداءً (0,3172). يُوصى بتطبيقLR في البيئات ذات الموارد المحدودة، بينما يُنصح بالتجميع التراصي للفحص السريري الرسمي. قدمت هذه الدراسة إطار تقييم شامل يتماشى مع مهمة حفظ النفس في مقاصد الشريعة.

BAB I
PENDAHULUAN

1.1 Latar Belakang

Kesehatan mental merupakan bagian tak terpisahkan dari kesehatan secara keseluruhan dan berperan sebagai indikator utama dalam menentukan kualitas hidup baik bagi individu maupun masyarakat (Medvedev & Landhuis, 2018). Menurut Organisasi Kesehatan Dunia (WHO), kesehatan mental didefinisikan sebagai keadaan kesejahteraan di mana seseorang dapat mewujudkan potensinya, mengatasi tekanan hidup yang normal, bekerja secara produktif, dan berkontribusi kepada masyarakat (Cyranka, 2020). Namun, meskipun sangat penting, masalah kesehatan mental masih sering diabaikan karena berbagai alasan. Hal ini termasuk stigma sosial, layanan kesehatan yang terbatas, dan kurangnya kesadaran masyarakat mengenai pentingnya pencegahan dan penanganan gangguan mental.

Secara global, sebuah studi yang dilakukan oleh *Global Burden of Disease* yang mengukur dampak berbagai penyakit dan faktor risiko kesehatan masyarakat di seluruh dunia dan terbitkan dalam jurnal *The Lancet Psychiatry* melaporkan bahwa lebih dari 1 miliar orang hidup dengan gangguan kesehatan mental, dengan perkiraaan 13–14% populasi dunia yang dampaknya pada tahun 2019 (Collaborators, 2022). Dari jumlah tersebut, sekitar 280 juta orang menderita depresi, suatu kondisi yang telah diidentifikasi sebagai salah satu gangguan paling mendesak di dunia (Rehm & Shield, 2019). Sebuah studi lanjutan yang dilakukan oleh Santomauro et al. (2021) juga menunjukkan bahwa tingkat depresi dan kecemasan meningkat secara signifikan, yakni sekitar 25% selama tahun pertama

pandemi COVID-19 tersebut, dibandingkan dengan tingkat yang tercatat sebelum dimulainya pandemik. Situasi ini sangat mengkhawatirkan terutama pada kelompok usia muda, sebagaimana dikonfirmasi oleh laporan Komisi Kesehatan Mental Global *The Lancet* menyatakan bahwa satu dari tujuh remaja berusia 10–19 tahun mengalami masalah kesehatan mental, yang menunjukkan bahwa masalah ini muncul pada usia yang sangat dini (Patel et al., 2018).

Penelitian oleh W. Li et al., (2022) menunjukkan bahwa mahasiswa memiliki tingkat masalah kesehatan mental yang lebih tinggi dibandingkan populasi umum, seperti gejala depresi, kecemasan, dan stres akademik. Selain itu, telah diamati oleh Raj (2025) bahwa dalam lingkup pendidikan tinggi, kesehatan mental mahasiswa merupakan faktor penting yang memengaruhi produktivitas, motivasi belajar, dan keberhasilan akademik jangka panjang. Akibatnya, kegagalan dalam menangani kesehatan mental mahasiswa secara memadai dapat berdampak buruk pada prestasi akademik, hubungan sosial, dan bahkan meningkatkan risiko putus kuliah.

Di dalam komunitas mahasiswa sendiri, masalah kesehatan mental sering muncul namun tidak terdeteksi sejak dini karena keterbatasan layanan konseling dan kecenderungan mahasiswa untuk menunda mencari bantuan. Situasi ini menimbulkan kebutuhan akan metodologi deteksi dini yang cepat, imparsial, dan mudah diakses. Pendekatan Machine Learning dipilih berdasarkan kemampuannya untuk menganalisis pola risiko secara otomatis dari data mahasiswa, sehingga memberikan alternatif yang lebih efektif dibandingkan metode manual atau penilaian subjektif seperti konseling tatap muka atau pengamatan perilaku.

Seiring kemajuan teknologi, pendekatan berbasis data semakin banyak digunakan untuk memahami faktor-faktor yang memengaruhi kesehatan mental. *Machine Learning*, sebuah bidang dalam kecerdasan buatan, memiliki potensi besar untuk mendeteksi pola-pola tersembunyi dalam data kesehatan mental, sehingga memfasilitiasi prediksi risiko gangguan secara lebih dini (Bhatnagar et al., 2022). Beberapa penelitian seperti H. Li et al. (2020) menerapkan algoritma seperti *Support Vector Machine* (SVM) untuk deteksi gangguan bipolar kemudian ada juga penelitian dari Sahu & Debbarma (2023) menerapkan algoritma *Random Forest* (RF) untuk melakukan prediksi masalah kesehatan mental pada siswa seperti stres, kecemasan, PTSD, ADHD, dan depresi.

Bentuk-bentuk umum gangguan kesehatan mental yang umum terjadi antara lain *bipolar disorder, skizofrenia, post-traumatic stress disorder* (PTSD), depresi dan gangguan kecemasan, serta *attention-deficit/hyperactivity disorder* (ADHD). Gangguan *Bipolar* adalah kondisi kejiwaan yang ditandai dengan perubahan suasana hati yang ekstrem, mulai dari fase manik yang penuh energi hingga fase depresif yang ditandai dengan kesedihan mendalam dan hilangnya minat terhadap aktivitas sehari-hari. *Skizofrenia* adalah gangguan yang menimbulkan tantangan signifikan dalam kemampuan seseorang untuk membedakan antara kenyataan dan pikiran mereka sendiri. Kondisi ini ditandai oleh halusinasi, delusi, serta pola bicara dan pemikiran yang tidak teratur, sehingga memerlukan pengobatan jangka panjang. Gangguan stres pascatrauma (PTSD) muncul sebagai respons terhadap pengalaman traumatis, termasuk kecelakaan, bencana, atau kekerasan. Gejala umum meliputi pengulangan trauma melalui kilas balik atau mimpi buruk, serta

kecenderungan untuk menghindari pemicu yang mungkin memicu kembali trauma awal. Depresi sendiri menyebabkan kesedihan berkepanjangan, hilangnya semangat, dan perubahan pola hidup. Gangguan kecemasan, di sisi lain, ditandai oleh kekhawatiran yang persisten yang menyebabkan gangguan pada ketenangan mental, produktivitas, dan hubungan sosial. Gangguan defisit perhatian/hiperaktivitas (ADHD) adalah kondisi *neurodevelopmental* yang muncul pada masa kanak-kanak dan sering berlanjut hingga dewasa. Gejala ADHD meliputi ketidakmampuan berkonsentrasi, impulsivitas, dan hiperaktivitas, yang dapat berdampak negatif pada proses belajar dan interaksi sosial. Namun, dengan intervensi terapeutik yang tepat, ADHD dapat dikelola secara efektif.

Meskipun berbagai penelitian telah membuktikan efektivitas machine learning dalam prediksi kesehatan mental, tinjauan terhadap literatur yang ada menunjukkan bahwa belum ada satu algoritma tunggal yang secara konsisten unggul pada seluruh kondisi dan jenis data. Setiap algoritma memiliki karakteristik dan keunggulan yang berbeda-beda bergantung pada karakteristik dataset yang digunakan. Logistic Regression unggul dalam interpretabilitas hasil, Support Vector Machine efektif dalam menangani pola nonlinier, Random Forest stabil pada data campuran, sementara XGBoost mampu memberikan akurasi tinggi melalui mekanisme boosting. Kondisi ini menimbulkan tantangan dalam memilih pendekatan yang paling tepat, khususnya untuk data psikososial mahasiswa yang memiliki kompleksitas fitur tinggi seperti konektivitas sosial dan indikator risiko mental lainnya. Oleh karena itu, diperlukan evaluasi sistematis terhadap berbagai

metode machine learning guna menemukan pendekatan yang paling optimal untuk prediksi kesehatan mental mahasiswa.

Pendekatan *Machine Learning* ini, berkontribusi pada kemajuan pengetahuan di dua bidang yang berbeda namun saling terkait yaitu kecerdasan buatan dan kesehatan mental. Model prediktif yang dimaksud memiliki kemampuan untuk memfasilitasi deteksi dini individu yang berisiko mengalami gangguan kesehatan mental. Akibatnya, kemampuan ini memungkinkan penerapan strategi pencegahan yang tepat.

Pentingnya menjaga kesehatan mental juga ditekankan dalam perspektif spiritual. Allah SWT. dalam surat Ar-Ra'd ayat 28 berfirman:

الَّذِيْنَ اٰمَنُوْا وَتَطْمَىِٕنُّ قُلُوْبُهُمْ بِذِكْرِ اللّٰهِۗ اَلَا بِذِكْرِ اللّٰهِ تَطْمَىِٕنُّ الْقُلُوْبُۗ

"(Yaitu) orang-orang yang beriman dan hati mereka menjadi tenteram dengan mengingat Allah. Ingatlah, bahwa hanya dengan mengingat Allah hati akan selalu tenteram."

Ayat ini menegaskan bahwa ketenangan hati dan kesehatan mental seseorang sangat erat kaitannya dengan *dzikir* (mengingat Allah). Dalam kehidupan sehari-hari manusia sering menghadapi berbagai macam tantangan yang dapat menimbulkan kecemasan, stres, atau bahkan depresi.

Selain itu, Allah SWT. juga menegaskan dalam QS. Al-Baqarah ayat 286:

لَا يُكَلِّفُ ٱللَّهُ نَفْسًا إِلَّا وُسْعَهَا ۚ لَهَا مَا كَسَبَتْ وَعَلَيْهَا مَا ٱكْتَسَبَتْ

"Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya. Ia mendapat pahala (dari kebajikan) yang dikerjakannya, dan ia mendapat siksa (dari kejahatan) yang diperbuatnya."

Ayat ini memberikan penguatan bahwa setiap individu memiliki batas kemampuan, dan tekanan yang dihadapi seharusnya dikelola dengan bijak agar tidak berdampak buruk pada kesehatan mental.

1.2 Pernyataan Masalah

Pendekatan metode Machine Learning apa yang paling optimal untuk melakukan prediksi kesehatan mental, ditinjau dari aspek performa klasifikasi dan efesiensi komputasi ?

1.3 Tujuan Penelitian

Mengevaluasi metode Machine Learning yang paling optimal untuk melakukan prediksi kesehatan mental berdasarkan trade-off antara performa klasifikasi dan efesiensi komputasi.

1.4 Batasan Masalah

1. Data yang digunakan adalah dataset terbuka mengenai kesehatan mental mahasiswa internasional dan domestik di Jepang yang diambil pada artikel *A Dataset of Students' Mental Health and Help-Seeking Behaviors in a Multicultural Environment*.
2. Kriteria optimal dalam penelitian ini didefinisikan berdasarkan dua dimensi utama: (a) performa klasifikasi, yang meliputi akurasi, presisi, recall, dan F1-Score dalam skema 5-Fold Cross-Validation dengan pengujian signifikansi statistik menggunakan Uji-T Berpasangan; serta (b) efisiensi komputasi, yang meliputi waktu training, waktu inferensi, konsumsi memori (RAM), penggunaan CPU, dan kompleksitas struktural algoritma (Cyclomatic

Complexity). Penentuan model paling optimal dilakukan berdasarkan evaluasi menyeluruh terhadap kedua dimensi tersebut secara bersamaan.

1.5 Manfaat Penelitian

Penelitian ini diharapkan dapat dimanfaatkan oleh:

1. Konselor, hasil penelitian ini dapat menjadi bahan pertimbangan dalam melakukan skrining awal terhadap individu yang berpotensi mengalami gangguan kesehatan mental, sehingga penanganan profesional dapat dilakukan secara lebih cepat dan tepat.
2. Fakultas Psikologi, temuan penelitian ini dapat menjadi referensi tambahan dalam pengembangan terkait kesehatan mental mahasiswa, khususnya melalui pendekatan berbasis data dan *machine learning*.
3. Rumah sakit jiwa, model prediksi yang dihasilkan berpotensi digunakan sebagai alat bantu awal untuk mengidentifikasi risiko pasien, sehingga dapat mendukung proses diagnosis maupun perencanaan terapi.

BAB II
STUDI LITERATUR

2.1 Prediksi Kesehatan Mental

Penelitian oleh H. Li et al. (2020) memadukan dataset *image* seperti MRI struktural (sMRI) dengan fungsional (fMRI) untuk mengidentifikasi penyakit Bipolar. Dataset terdiri dari 44 pasien BPD dan 36 kontrol sehat (HC) dari Guangzhou Huiai Hospital, China. Metode yang digunakan adalah SVM (Support Vector Machine) dengan seleksi fitur LASSO menghasilkan akurasi 87,5% dan AUC 0,939. Area otak seperti gyrus frontalis inferior dan temporal superior terbukti signifikan sebagai biomarker. Namun, penelitian ini memiliki keterbatasan seperti sampel kecil, efek pengobatan, kurangnya validasi eksternal, dan dominasi pasien dalam fase remisi, sehingga memerlukan studi lanjutan dengan dataset lebih besar dan variasi kondisi klinis yang lebih beragam.

Perez Arribas et al. (2018) melakukan penelitian untuk mengkaji dua jenis gangguan kejiwaan, yaitu Bipolar Disorder (BD) dan Borderline Personality Disorder (BPD) dengan menggunakan dataset yang berasal dari 130 partisipan yang mencatat suasana hati harian melalui aplikasi smartphone, dengan enam kategori mood, seperti cemas, gembira, sedih, marah, mudah tersinggung, dan berenergi. Penelitian ini menggunakan algoritma RF dengan pendekatan *signature-based features extraction* untuk mengolah pola perubahan suasana hati. Hasilnya, model mampu membedakan diagnosis dengan akurasi hingga 75% dan memprediksi mood berikutnya dengan tingkat akurasi tertinggi pada kelompok sehat (hingga 98%).

Namun, studi ini memiliki keterbatasan pada jumlah data yang relatif kecil dan tidak adanya validasi klinis terhadap laporan *mood* peserta.

Penelitian yang dilakukan oleh Ramirez-Cifuentes et al. (2021) berfokus pada pengembangan variasi *enhanced word embeddings* sebagai upaya meningkatkan akurasi deteksi gangguan kesehatan mental dan penyalahgunaan zat melalui tulisan di media sosial Reddit. Dari sisi metodologis, penelitian ini membandingkan baseline representasi teks seperti Bag of Words (BoW) dan beberapa variasi dari *text embedding* seperti *Word2Vec*, *GloVe*, dan *DistilBERT* yang dikembangkan peneliti. Selain itu, penelitian ini memperkenalkan metode *feature extraction Pivot Similarity* (PSim), guna menghasilkan representasi fitur yang lebih jelas. Seluruh representasi kemudian diuji menggunakan algoritma prediksi *Logistic Regression* (LR), *Random Forest* (RF), dan *Convolutional Neural Networks* (CNN). Hasil eksperimen menunjukkan bahwa *embedding* yang ditingkatkan dengan bobot awal *GloVe* serta kombinasi *GloVe* dengan *retrofitting* memberikan kinerja yang lebih unggul pada *Recall, F1-score*, dan *accuracy* dibanding representasi embedding standar. Kontribusi utama penelitian ini terletak pada pembuktian bahwa enhanced embeddings mampu menangkap pola linguistik, emosional, dan psikologis yang relevan dalam teks media sosial, sehingga berpotensi menjadi pendekatan efektif untuk prediksi kesehatan mental berbasis teks dalam konteks dataset yang terbatas.

Birnbaum et al. (2017) juga melakukan penelitian yang berfokus pada identifikasi individu dengan *skizofrenia* melalui analisis bahasa di media sosial, khususnya Twitter. Dataset ini dikumpulkan dari 671 pengguna Twitter yang secara

terbuka menyatakan pernah didiagnosis skizofrenia, dan dievaluasi keasliannya oleh tim klinis. Data ini bersifat tekstual dan time-series, mencakup jutaan tweet yang dianalisis menggunakan fitur *Term Frequency-Inverse Document Frequency* (TF-IDF) dengan n-gram (500 unigram, bigram, trigram) dan *Linguistic Inquiry and Word Count* (LIWC). Untuk klasifikasi, digunakan beberapa algoritma *machine learning* seperti RF, SVM, LR, dan Naïve Bayes, dengan RF menjadi yang paling akurat. Hasil terbaik menunjukkan akurasi hingga 90% dan AUC 0.95 dalam membedakan pengguna dengan skizofrenia dari kontrol sehat. Namun, kekurangan utama studi ini adalah keterbatasan dalam memverifikasi diagnosis klinis secara langsung, serta kemungkinan bias dari pengguna yang memilih untuk tidak menyatakan diagnosis mereka secara publik di media sosial.

Zafari et al. (2021) meneliti orang dengan identifikasi panyakit gangguan strees pasca-trauma atau *Post-Traumatic Strees Disorder* (PTSD) menggunakan data rekam medis elektronik (EMR) dari layanan kesehatan primer. Dari 154.118 pasien, sebanyak 195 kasus PTSD yang sudah divalidasi digunakan sebagai data utama. Dataset terdiri dari data terstruktur (seperti demografi dan riwayat medis) dan teks bebas dari catatan dokter yang diolah dengan teknik *Bag of Words* (BoW). Peneliti menggunakan beberapa model machine learning, termasuk *Random Forest*, *Multi-Layer Neural Network* (MLNN), serta kombinasi serial dan paralel. Hasil terbaik diperoleh dari model serial dengan RF, dengan akurasi 99%. Studi ini menunjukkan bahwa menggabungkan data terstruktur dan teks bebas meningkatkan akurasi deteksi PTSD. Namun, keterbatasan penelitian terletak pada jumlah kasus positif PTSD yang sedikit dan tidak lengkapnya data catatan naratif.

Penelitian Kim, (2020) juga mengembangkan model analisis untuk memprediksi PTSD pada petugas pemadam kebakaran yang sering terpapar trauma akibat kecelakaan dan bencana. Data mencakup 2.705 personel dan terdiri dari kombinasi data numerik serta skala penilaian seperti skala trauma (IES-R), depresi (CESD), risiko bunuh diri (SBQ-R), dan konsumsi alkohol (AUDIT-K). Peneliti mengembangkan model prediksi menggunakan Support Vector Machine (SVM) dan Neural Network Model (NNM). Dalam penerapannya, SVM memanfaatkan teknik *feature extraction* melalui kernel *Gaussian* RBF untuk memproyeksikan data ke ruang berdimensi lebih tinggi sehingga pola nonlinier dapat dipisahkan dengan lebih akurat. Hasil tertinggi dicapai oleh model SVM dengan akurasi mencapai 89%. Hasilnya menunjukkan bahwa model AI efektif untuk deteksi dini PTSD. Namun, penelitian ini terbatas pada jumlah kasus positif PTSD yang sedikit dan potensi bias dalam data self-report.

Jyotsna et al. (2023) mengembangkan model personalisasi berbasis time series yang disebut PredictEYE, yang dirancang untuk memprediksi kondisi mental seseorang serta mengidentifikasi adegan spesifik yang memicu kondisi tersebut. Data penelitian diperoleh dari eye tracking selama sepuluh menit (video tenang dan stres). Proses feature extraction menggunakan algoritma I-DT (*Identification by Dispersion Threshold*) yang mengekstrak fitur seperti *fixation duration, fixation dispersion* (X dan Y), pupil diameter, serta *blink duration*. Fitur-fitur tersebut kemudian diprediksi melalui time series, dan hasilnya diklasifikasikan menggunakan algoritma *Random Forest* untuk menentukan kondisi mental. Hasil

penelitian menunjukkan bahwa PredictEYE mampu mencapai akurasi 86,4% serta terbukti lebih unggul dibandingkan pendekatan lain.

Kasanneni et al. (2025) meneliti efektivitas algoritma machine learning dan deep learning untuk diagnosis kesehatan mental melalui percakapan di media sosial. Dataset diambil dari platform seperti Reddit dan forum daring lain yang berisi diskusi terkait kesehatan mental. Proses analisis dilakukan dengan teknik NLP, mencakup *preprocessing text* dan *feature extraction* menggunakan TF-IDF *vectorization* serta *text embedding* untuk merepresentasikan data secara lebih jelas perbandingannya. Tiga model utama yang dievaluasi adalah *Random Forest* (RF), *Convolutional Neural Network* (CNN), dan XLNet. Hasil penelitian menunjukkan bahwa XLNet memperoleh kinerja terbaik dengan akurasi lebih dari 97%, unggul dalam menangkap konteks linguistik yang kompleks. CNN juga menunjukkan hasil baik dengan kemampuannya mengekstraksi pola lokal pada teks, meski tidak seunggul model transformer. Sementara itu, *Random Forest* hanya berfungsi sebagai baseline dengan akurasi lebih rendah, karena keterbatasannya dalam memahami konteks mendalam dari bahasa alami.

Penelitian oleh Saha et al. (2023) bertujuan mengidentifikasi berbagai gangguan mental melalui percakapan konseling motivasional antara pengguna dan agen virtual. Dataset yang digunakan adalah MotiVAte, berisi 5.067 percakapan dari forum *PsychCentral* yang mencakup empat gangguan mental: MDD, PTSD, kecemasan, dan OCD. Metode yang digunakan adalah *hierarchical attention-based deep neural network* berbasis Bi-LSTM, dengan *feature extraction* berupa GloVe *embeddings* dan *lexicon-based sentiment scores* untuk menangkap makna teks dan

emosi. Hasil penelitian menunjukkan akurasi 83,91% pada dataset MotiVAte dan 92,41% pada dataset tambahan Twitter-Depression. Kelebihan penelitian ini adalah kontribusi dataset baru yang besar serta model hierarkis yang efektif memahami konteks percakapan, sedangkan kekurangannya adalah kesulitan membedakan gangguan dengan ekspresi bahasa yang mirip serta keterbatasan leksikon sentimen yang dapat memunculkan kesalahan

Penelitian oleh Tabares Tabares et al., (2024) bertujuan mengidentifikasi faktor-faktor yang berasosiasi dengan gejala kecemasan pada kalangan muda melalui pemanfaatan model kecerdasan buatan. Dataset yang digunakan terdiri dari 234 responden berusia 18–28 tahun yang dikumpulkan melalui kuesioner terstandarisasi, meliputi PHQ-9, GAD-7, *Penn Alcohol Craving Scale* (PACS), *Severity of Dependence Scale* (SDS), serta data sosiodemografis yang komprehensif. Pada tahap analisis, peneliti menerapkan tiga algoritma machine learning yaitu *Random Forest* (RF), *Support Vector Machine* (SVM), dan *K-Nearest Neighbors* (KNN) dengan skema 4-*fold cross-validation* untuk mengevaluasi performa model. Hasil penelitian menunjukkan bahwa *Random Forest* merupakan model paling unggul dengan akurasi 91%, disertai konsistensi nilai *precision, recall*, dan *F1-score*, sehingga menghasilkan kinerja lebih baik dibandingkan sebagian penelitian sebelumnya. Selain itu, penelitian ini menemukan bahwa faktor yang paling berpengaruh terhadap kecemasan mencakup skor pertanyaan pada GAD-7, skor tertentu PHQ-9, riwayat keluarga gangguan mental, frekuensi konsumsi alkohol, serta tingkat pendidikan orang tua, sebagaimana ditunjukkan melalui analisis *feature importance* dan SHAP *values*.

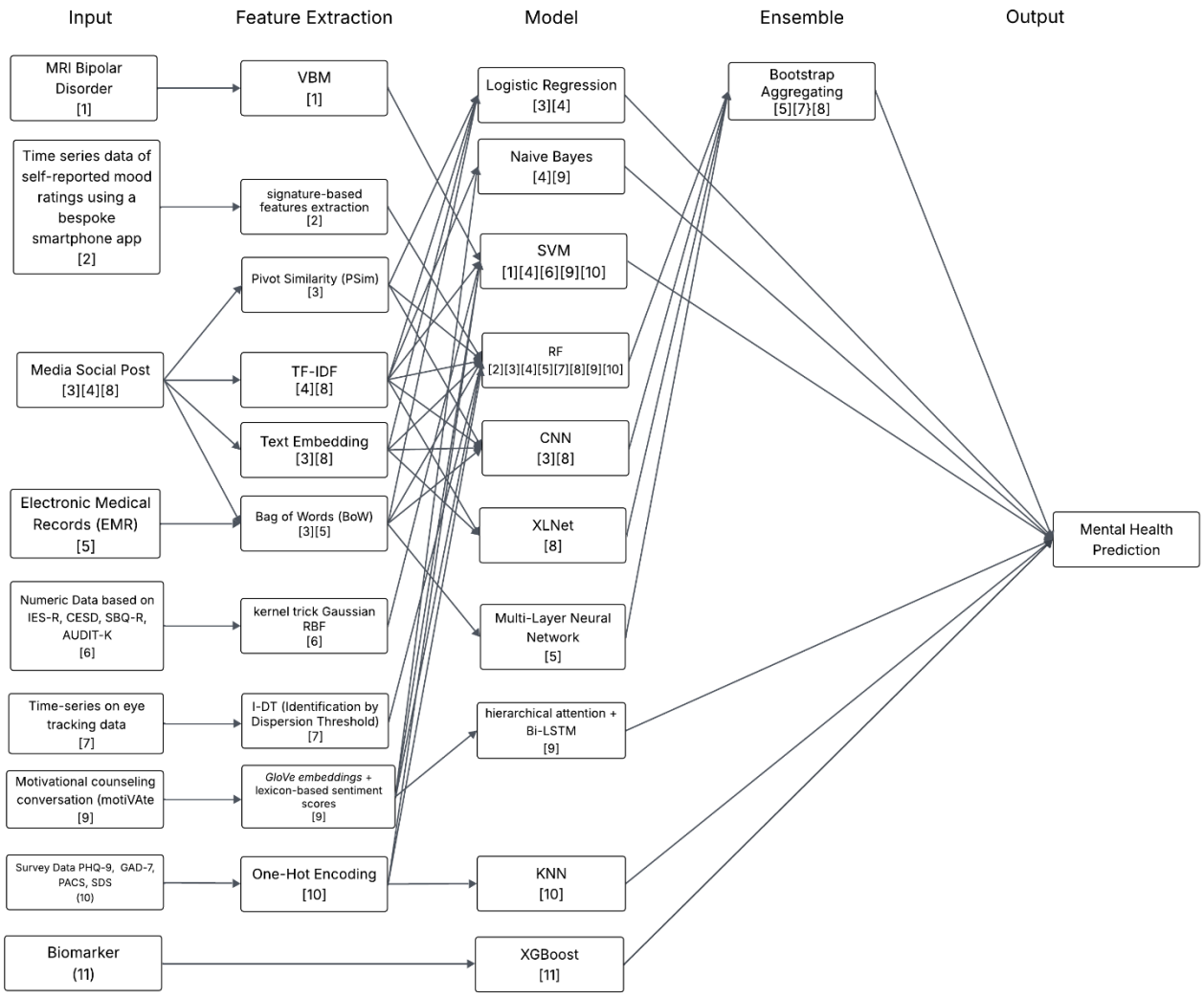
Sharma & Verbeke, 2020 melakukan penelitian untuk meningkatkan akurasi diagnosis depresi menggunakan algoritma Extreme Gradient Boosting (XGBoost) pada dataset biomarker Lifelines Cohort Study yang terdiri dari 11.081 warga Belanda. Dataset memuat 28 variabel biomarker dari hasil tes darah dan urin sebagai fitur input, dengan variabel target berupa depresi *self-reported* berdasarkan instrumen M.I.N.I., di mana hanya 5,14% responden teridentifikasi mengalami depresi sehingga menimbulkan kondisi *class imbalance* yang signifikan. Untuk mengatasinya, peneliti menerapkan empat strategi resampling yaitu under-sampling, over-sampling, over-under sampling, dan ROSE, lalu menguji model XGBoost pada masing-masing varian dataset. Hasil terbaik diperoleh pada over-sample dataset dengan akurasi 0,9729, presisi 0,9548, recall 0,9987, dan F1-Score 0,9762. Penelitian ini relevan sebagai landasan pemilihan algoritma XGBoost dalam penelitian ini, yang terbukti efektif untuk klasifikasi depresi pada data tabular terstruktur meskipun jenis datanya berbeda — penelitian ini menggunakan data kuesioner psikologis (PHQ-9, SCS, GHSQ, dan ASSIS) sebagai pengganti biomarker klinis.

Berdasarkan hasil tinjauan penelitian terdahulu, empat algoritma yang paling konsisten menunjukkan performa kuat dalam prediksi kesehatan mental adalah *Logistic Regression* (LR), *Support Vector Machine* (SVM), *Random Forest* (RF), dan XGBoost. Keempat metode ini memiliki karakteristik yang saling melengkapi seperti LR unggul dalam interpretabilitas, SVM efektif pada pola nonlinier, RF stabil pada data campuran, dan XGBoost memberikan akurasi tinggi melalui boosting. Temuan tersebut menunjukkan bahwa tidak ada satu algoritma

tunggal yang secara konsisten unggul pada semua kondisi, sehingga pendekatan *ensemble* menjadi relevan untuk menggabungkan kelebihan masing-masing model. Secara empiris, teknik ensemble terbukti mampu meningkatkan akurasi, mengurangi variansi prediksi, serta mengatasi keterbatasan metodologis dari model tunggal. Oleh karena itu, penelitian ini memilih pendekatan ensemble berbasis empat metode tersebut untuk menghasilkan prediksi kesehatan mental mahasiswa yang lebih akurat, stabil, dan dapat diandalkan.

2.2 Kerangka Teori

Bagian ini mengulas berbagai teori dari penelitian sebelumnya yang berkaitan dengan prediksi kesehatan mental menggunakan pendekatan machine learning. Ulasan mencakup jenis data yang digunakan sebagai input, tahapan preprocessing untuk mempersiapkan data, teknik ekstraksi fitur yang relevan, serta pemanfaatan algoritma machine learning dalam membangun model prediksi. Proses ini bertujuan untuk menghasilkan keluaran berupa deteksi atau prediksi kondisi kesehatan mental individu berdasarkan data yang telah diolah. Kerangka teori untuk prediksi kesehatan mental dapat dilihat pada Gambar 2.1.



Gambar 2. 1 Contoh gambar dengan menggunakan grafik

Li et al., 2020 (1); Perez Arribas et al, 2018 **(2)**; Ramirez-Cifuentes et al, 2021 **(3)**; Birnbaum et al, 2017 **(4)**; Zafari et al, 2021 **(5)**; J. B. Kim, 2020 **(6)**; Jyotsna et al, 2023 **(7)**; Kasanneni et al, 2025 **(8)**; Saha et al., 2023 **(9)** Tabares Tabares et al., 2024 (10); Sharma & Verbeke, 2020**(11)**

Tabel 2. 1 Penelitian Terkait

No.	Peneliti	Jenis Data/Input	Feature Extraction	Metode	Ensemble	Hasil
1	(H. Li et al., 2020)	MRI struktural (sMRI) & fungsional (fMRI)	LASSO Feature Selection, VBM	SVM	-	87.5%
2	(Perez Arribas et al., 2018)	Mood rating harian via aplikasi smartphone	Signature-based features	Random Forest	-	75.00% (hingga 98.00% prediksi mood sehat)
3	(Ramirez-Cifuentes et al., 2021)	Postingan Reddit (teks, sosial media)	BoW, Word2Vec, GloVe, DistilBERT, PSim	Logistic Regression, RF, CNN	-	
4	(Birnbaum et al., 2017)	Twitter	TF-IDF, LIWC	RF SVM, Logistic Regression, Naive Bayes	-	90.00%
5	(Zafari et al., 2021)	Rekam Medis Elektronik	BoW	RF, MLNN, Kombinasi Serial-pararel	Boostrap Aggregating	99.00%
6	(Kim, 2020)	Data numerik	Kernel Gaussian RBF	SVM, Neural Network	-	89.00%
7	(Jyotsna et al., 2023)	Eye-tracking	I-DT	Random Forest	Boostrap Aggregating	86.40%
8	(Kasanneni et al., 2025)	Reddit	TF-IDF, Text Embedding	RF, CNN, XLNet	Boostrap Aggregating	97%
9	(Saha et al., 2023)	Twitter	GloVe	Bi-LSTM	-	92.41%
10	(Tabares Tabares et al., 2024)	Survey Data	One-Hot Encoding	RF, SVM,KNN	-	91%
11	(Sharma & Verbeke, 2020)	biomarker	-	XGBoost	-	97.2%

2.3 Gap Analisis dan Pernyataan Kebaruan Penilitian

Berdasarkan tinjauan terhadap berbagai penelitian terdahulu yang telah dipaparkan pada sub-bab sebelumnya, dapat diidentifikasi sejumlah kesenjangan penelitian (*research gap*) yang menjadi dasar kebaruan penelitian ini. Tabel 2.2 berikut menyajikan perbandingan sistematis antara penelitian terdahulu dengan penelitian ini ditinjau dari beberapa dimensi utama

Tabel 2. 2 Gap Analisis Penelitian Terkait

No.	Penelitian	Dataset / Data	Algoritma Utama	Populasi Mahasiswa	Ensemble	Uji Statistik
1	Li et al. (2020)	MRI (44 BPD, 36 HC)	SVM + LASSO	Tidak	Tidak	Tidak
2	Perez Arribas et al. (2018)	Mood harian smartphone (130 partisipan)	Random Forest	Tidak	Tidak	Tidak
3	Ramirez-Cifuentes et al. (2021)	Teks media sosial Reddit	LR, RF, CNN	Tidak	Tidak	Tidak
4	Birnbaum et al. (2017)	Tweet (671 pengguna)	RF, SVM, LR, Naïve Bayes dengan Bootstrap Aggregating	Tidak	Tidak	Tidak
5	Zafari et al. (2021)	Rekam medis elektronik (154.118 pasien)	RF, MLNN	Tidak	Ya	Tidak

No.	Penelitian	Dataset / Data	Algoritma Utama	Populasi Mahasiswa	Ensemble	Uji Statistik
6	Kim (2020)	Data pemadam kebakaran (2.705 personel)	SVM, Neural Network	Tidak	Tidak	Tidak
7	Jyotsna et al. (2023)	Eye tracking (video tenang & stres)	Random Forest (time series)	Tidak	Ya	Tidak
8	Kasanneni et al. (2025)	Reddit & forum daring	RF, CNN, XLNet	Tidak	Ya	Tidak
9	Saha et al. (2023)	Percakapan konseling (5.067 data)	Bi-LSTM + GloVe	Tidak	Tidak	Tidak
10	Tabares et al. (2024)	Kuesioner (234 responden, 18–28 thn)	RF, SVM, KNN	Tidak	Tidak	Tidak
11	Sharma & Verbeke (2020)	Biomarker Lifelines (11.081 data)	XGBoost	Tidak	Tidak	Tidak
12	Penelitian Ini	Kuesioner mahasiswa multikultural Jepang	LR, SVM, RF, XGBoost + Stacking & Voting	Ya	Ya	Ya

Berdasarkan tabel gap analysis di atas, terdapat beberapa kesenjangan yang signifikan dalam penelitian terdahulu. Pertama, meskipun beberapa penelitian seperti Zafari et al. (2021), Jyotsna et al., (2023) dan Kasanneni et al., (2024) telah menerapkan teknik Bootstrap Aggregating sebagai bentuk ensemble sederhana, tidak ada penelitian yang secara sistematis mengintegrasikan empat algoritma

machine learning sekaligus seperti Logistic Regression, SVM, Random Forest, dan XGBoost dalam dua skema ensemble berbeda yaitu Majority Voting dan Stacking, khususnya pada konteks prediksi kesehatan mental mahasiswa. Kedua, hampir seluruh penelitian terdahulu tidak melakukan uji signifikansi statistik seperti Paired T-Test untuk membuktikan secara ilmiah apakah perbedaan performa antar model benar-benar signifikan atau hanya variasi acak, sehingga klaim keunggulan model pada penelitian sebelumnya bersifat deskriptif numerik semata. Ketiga, mayoritas penelitian terdahulu dilakukan pada populasi umum, pasien klinis, atau pengguna media sosial, bukan secara spesifik pada mahasiswa yang memiliki karakteristik psikososial unik seperti tekanan akademik, keterhubungan sosial, dan ideasi bunuh diri. Keempat, belum ditemukan penelitian yang secara spesifik menerapkan pendekatan *ensemble* integratif pada dataset kesehatan mental mahasiswa multikultural berbasis kuesioner psikososial di lingkungan akademik Jepang, sehingga penelitian ini mengisi kekosongan tersebut secara kontekstual dan metodologis.

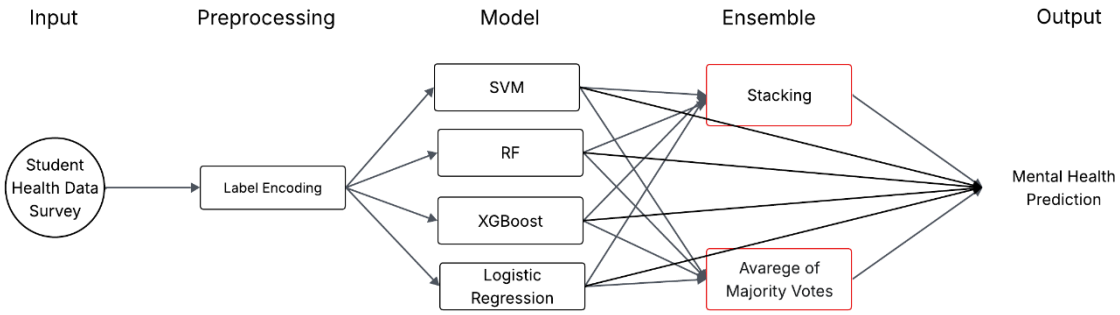
Berdasarkan kesenjangan tersebut, kebaruan (novelty) penelitian ini terletak pada tiga hal utama. Pertama, penelitian ini merupakan salah satu yang pertama mengintegrasikan empat algoritma machine learning (Logistic Regression, SVM, Random Forest, dan XGBoost) secara bersamaan dalam dua skema ensemble yaitu Majority Voting dan Stacking, khusus untuk prediksi kesehatan mental mahasiswa dalam lingkungan multikultural sebuah kombinasi yang belum ditemukan pada literatur yang ditinjau. Kedua, penelitian ini melengkapi evaluasi performa model dengan uji signifikansi statistik menggunakan Paired T-Test, sehingga klaim

keunggulan model dapat dipertanggungjawabkan secara ilmiah dan tidak sekadar perbandingan numerik deskriptif. Ketiga, penelitian ini secara spesifik mengeksplorasi fitur psikososial Social Connectedness (ToSC) dan Suicidal Ideation sebagai prediktor utama pada dataset mahasiswa multikultural di lingkungan akademik Jepang, mengisi kekosongan penelitian yang selama ini lebih banyak berfokus pada data klinis, biomarker, atau teks media sosial.

BAB III
MODEL KONSEPTUAL DAN PERSIAPAN DATA

3.1 Kerangka Konsep

Kerangka konsep pada penelitian ini akan digambarkan pada Gambar 3.1. Tahapan terdiridari *input, Feature Extraction, Model, Ensemble* dan *output.*



Gambar 3. 1 Kerangka Konsep

Penelitian ini menggunakan kerangka kerja sistematis yang terbagi ke menjadi lima tahap untuk memprediksi kesehatan mental mahasiswa. Proses dimulai dengan tahap Input, yaitu melibatkan dataset publik yang diambil dari artikel penelitian berjudul "*A Dataset of Students' Mental Health and Help-Seeking Behaviors in a Multicultural Environment*". Kumpulan data ini mencakup informasi mengenai akulturasi, faktor stress, dan perilaku yang berkaitan dengan pencarian bantuan di kalangan mahasiswa dalam lingkungan multikultural. Selanjutnya, data diproses pada tahap ektraksi fitur, dengan menggunakan teknik Label Encoding. Berbeda dengan metodologi sebelumnya, pendekatan ini dipilih untuk mengubah variabel kategorikal menjadi representasi numerik yang lebih

efisien sambil mempertahankan dimensi kumpulan data dalam batas yang dapat diterima (fenomena yang dikenal sebagai “kutukan dimensi”).

Pada tahap Model, empat algoritma *Machine Learning* yang berbeda yaitu *Support Vector Machine* (SVM), *Random Forest* (RF), *Extreme Gradient Boosting* (XGBoost), dan *Logistic Regression* (LR), diterapkan melalui arsitektur paralel secara bersamaan. Penggunaan algoritma yang beragam ini bertujuan untuk menangkap pola data dari berbagai perspektif dengan memanfaatkan pendekatan linier dan non-linier. Dalam skema paralel ini, setiap algoritma bekerja sebagai base learner yang memproses dataset secara mandiri tanpa adanya ketergantungan antar model, sehingga memungkinkan sistem untuk mengekstraksi informasi yang lebih kaya dan komprehensif dari setiap paradigma algoritma.

Untuk mengoptimalkan kinerja prediksi, hasil dari keempat model individu ini diintegrasikan pada tahap *Ensemble*. Tahap ini melibatkan penerapan metode *Stacking* untuk mempelajari perilaku prediksi model dasar dan menentukan bobot keputusan yang paling akurat dalam kondisi tertentu, serta penggunaan *Majority Voting* untuk pengambilan keputusan berdasarkan konsensus suara terbanyak. Pendekatan ensemble ini diharapkan dapat menutupi kelemahan spesifik masing-masing algoritma tunggal melalui sinergi hasil prediksi, sehingga menghasilkan luaran diagnosa yang lebih stabil dan tangguh (robust).

Akibatnya, proses komprehensif ini bermuara pada pembentukan keluaran berupa Prediksi Kesehatan Mental. Diasumsikan bahwa hasil akhir ini akan berfungsi sebagai alat deteksi dini yang andal dan akurat untuk mengidentifikasi

risiko gangguan kesehatan mental di kalangan populasi mahasiswa berdasarkan indikator stres akulturasi.

3.2 Perolehan Data

Data yang digunakan dalam penelititan ini diperoleh dari kumpulan data yang dapat diakses publik yang berjudul *A Dataset of Students' Mental Health and Help-Seeking Behaviors in a Multicultural Environment* (Kumpulan Data tentang Kesehatan Mental dan Perilaku Pencarian Bantuan Mahasiswa dalam Lingkungan Multikultural), yang berisi 268 tanggapan. Proses pengumpulan data dilakukan melalui penerapan instrumen survei berbasis web menggunakan Google Forms pada periode Oktober hingga Desember 2018. Pemilihan metode ini didasarkan pada aksesibilitasnya bagi mahasiswa, keefektifannya dalam pengelolaan data, serta kemampuannya untuk menjamin kerahasiaan informasi peserta.

Kuesioner tersebut menggunakan kombinasi tiga instrumen pengukuran standar, yaitu Kuesioner Kesehatan Pasien (PHQ-9) untuk mengukur gejala depresi, Skala Keterikatan Sosial (SCS) untuk menilai tingkat keterhubungan sosial, Kuesioner Pencarian Bantuan Kesehatan Umum (GHSQ) untuk memetakan perilaku pencarian bantuan, dan Skala Stres Akulturasi untuk Mahasiswa Internasional (ASSIS) untuk mengukur tingkat stres yang akibat proses akulturasi dalam lingkungan baru. Selain itu, data sosiodemografis termasuk jenis kelamin, tingkat pendidikan, umur, dan lama studi. Dalam survei ini, setiap variable akan dikelompokkan sesuai kategori pengukurannya.

Pemilihan variabel prediktor dalam eksperimen ini didasarkan pada bukti teoretis bahwa depresi di kalangan mahasiswa dalam lingkungan multikultural

dipengaruhi oleh interaksi antara faktor internal dan eksternal. Variabel ASSIS dimasukkan karena peran yang diakui dari stres akulturasi, yang ditandai dengan persepsi diskriminasi dan rindu kampung halaman, sebagai faktor risiko utama yang memicu gangguan psikologis pada mahasiswa internasional. Variabel SCS digunakan karena bukti yang telah ditetapkan bahwa dukungan sosial dan rasa keterikatan sosial berfungsi sebagai faktor pelindung (penyangga) terhadap depresi. Variabel GHSQ dipilih karena cara individu mencari bantuan mencerminkan kemampuan mereka dalam mengatasi gejala psikologis. Akhirnya, faktor-faktor sosiodemografis dimasukkan untuk memperhitungkan variasi kerentanan berdasarkan profil individu. Penggabungan keempat kategori instrumen ini diharapkan dapat meningkatkan kemampuan model dalam memprediksi status depresi dengan lebih akurat.

Instrumen PHQ-9 terdiri dari sembilan indikator gejala depresi, yang dievaluasi berdasarkan frekuensi kemunculannya selama dua minggu terakhir, Setiap indikator mencerminkan aspek psikologis dan fisiologis yang mungkin dialami responden, sehingga memungkinkan klasifikasi kuantitatif tingkat keparahan depresi. Daftar indikator yang diukur melalui PHQ-9 dijabarkan pada Tabel 3.1.

Tabel 3. 1 Atribut Patient Health Questionnaire (PHQ-9)

No	Gejala yang dinilai	Tidak pernah (skor=0)	Beberapa hari (skor=1)	Lebih dari separuh hari (skor=2)	Hampir setiap hari (skor=3)
1	Kehilangan minat/kenikmatan	0	1	2	3
2	Perasaan sedih, muruh, putus asa	0	1	2	3
3	Gangguan tidur	0	1	2	3

4	Kelelahan atau kurang energi	0	1	2	3
5	Perubahan nafsu makan/berat badan	0	1	2	3
6	Rasa bersalah/meremehkan diri	0	1	2	3
7	Konsentrasi terganggu	0	1	2	3
8	Pergerakan melambat atau gelisah	0	1	2	3
9	Pikiran ingin menyakiti diri/ bunuh diri	0	1	2	3
	Skor total:	0-27			

Kategori instrumen selanjutnya adalah Skala Keterikatan Sosial (SCS). Instrumen yang dimaksud terdiri dari serangkaian pernyataan yang dirancang untuk menggali persepsi individu mengenai kedekatan emosional dan rasa memiliki dalam lingkungan sosialnya. Skala penilaian yang digunakan oleh SCS memungkinkan pemetaan tingkat keterikatan sosial mulai dari minimal hingga substansial. Indikator yang digunakan dalam SCS dijabarkan dalam Tabel 3.2.

Tabel 3. 2 Atribut Social Connectedness Scale (SCS)

No	Pernyataan	Sangat tidak setuju (skor=1)	Tidak setuju (skor=2)	Agak tidak setuju (skor=3)	Agak setuju (skor=4)	Setuju (skor=5)	Sangat setuju (skor=6)
1	aku merasa terputus dari dunia di sekitarku	1	2	3	4	5	6
2	Bahkan di sekitar orang yang kukenal, aku merasa tidak benar-benar diterima.	1	2	3	4	5	6
3	Aku merasa begitu jauh dari orang lain.	1	2	3	4	5	6
4	Aku tidak merasa kebersamaan dengan teman-temanku.	1	2	3	4	5	6
5	Aku merasa tidak terhubung dengan siapa pun.	1	2	3	4	5	6

6	Aku mendapati diriku kehilangan rasa keterhubungan dengan masyarakat.	1	2	3	4	5	6
7	Bahkan di antara teman-temanku, tidak ada rasa persaudaraan.	1	2	3	4	5	6
8	Aku merasa tidak berpartisipasi dengan siapa pun atau kelompok mana pun.	1	2	3	4	5	6
	Skor total:	8-48					

Kategori instrumen berikutnya adalah Skala Stres Akulturasi untuk Mahasiswa Internasional (ASSIS). Instrumen ini digunakan untuk mengukur stres emosional dan kognitif yang timbul akibat proses adaptasi terhadap budaya baru. Dalam dataset ini, instrumen ASSIS dibagi menjadi tujuh dimensi yang berbeda, mencakup berbagai aspek terkait tantangan akulturasi. ASSIS menggunakan indikator dan dimensi yang diuraikan pada Tabel 3.3.

Tabel 3. 3 Atribut Acculturative Stress Scale for International Students (ASSIS)

No	Dimensi Stres Akulturasi	Kode	Deskripsi Pengukuran
1	*Perceived Discrimination*	APD	Stres akibat persepsi perlakuan tidak adil atau diskriminasi sosial
2	*Homesickness*	AHome	Tekanan psikologis akibat rasa rindu pada tanah air atau keluarga
3	*Perceived Hostility*	APH	Persepsi terhadap adanya permusuhan atau sikap tidak ramah dari lingkungan baru
4	*Fear*	Afear	Rasa takut atau kecemasan yang muncul dalam interaksi lintas budaya.
5	*Culture Shock*	ACS	Perasaan bingung atau tidak nyaman terhadap perbedaan norma budaya.
6	*Guilt*	AGuilt	Rasa bersalah karena meninggalkan tanggung jawab atau orang terdekat
7	*Miscellaneous*	AMiscell	Masalah-masalah tambahan lainnya yang terkait dengan stres adaptasi.

Instrumen ASSIS diukur melalui serangkaian pernyataan yang dikelompokkan ke dalam tujuh dimensi utama. Setiap dimensi terdiri dari serangkaian item kuesioner yang dirancang secara cermat untuk mengeksplorasi aspek-aspek spesifik dari stress akulturasi. Daftar lengkap pertanyaan, beserta dimensi masing-masing, dilampirkan di bagian akhir penelitian ini.

Instrumen lain yang digunakan adalah General Health Help-Seeking Questionnaire (GHSQ). Instrumen yang sedang dibahas ini berisi serangkaian pernyataan yang berkaitan dengan kecenderungan seseorang untuk mencari dukungan atau bantuan ketika menghadapi tantangan psikologis atau yang berkaitan dengan kesehatan. Penilaian pada GHSQ memfasilitasi identifikasi sumber bantuan potensial yang mungkin dipilih oleh responden, selain kecenderungan mereka untuk memanfaatkan layanan tersebut. Atribut dan indikator yang digunakan dalam GHSQ dijabarkan dalam Tabel 3.4.

Tabel 3. 4 Atribut General Health Help-Seeking Questionnaire (GHSQ)

No	Pernyataan	Skala dari sangat tidak mungkin hingga sangat mungkin (skor = 1 – 7)
1	Saya akan meminta bantuan dari pasangan	1 – 7
2	Saya akan meminta bantuan dari teman dekat	1 – 7
3	Saya akan meminta bantuan dari orang tua	1 – 7
4	Saya akan mencari bantuan dari kerabat	1 – 7
5	Saya akan mencari bantuan dari profesional kesehatan mental resmi	1 – 7
6	Saya akan mencari bantuan dari layanan via telpon	1 – 7
7	Saya akan mencari dokter umum atau tenaga medis	1 – 7
8	Saya akan meminta bantuan dari pemuka agama	1 – 7
9	Saya akan menyelesaikan masalah sendiri	1 – 7

10	Saya akan mencari bantuan dari internet	1 – 7
11	Mencari bantuan dari sumber lain	1 – 7

3.2.1 Deskripsi Fitur Dataset

Dataset yang digunakan terdiri dari 28 kolom yang mencakup 27 fitur prediktor dan 1 variabel target (Dep). Berikut adalah deskripsi lengkap setiap fitur yang digunakan dalam penelitian ini.

Tabel 3. 5 Deskripsi Setiap Fitur Dataset

No	Nama Fitur	Tipe Data	Deskripsi
1	Gender	Kategorikal	Jenis kelamin responden (Male / Female)
2	inter_dom	Kategorikal	Status mahasiswa: internasional (Inter) atau domestik (Dom)
3	Academic	Kategorikal	Jenjang studi: sarjana (Under) atau pascasarjana (Grad)
4	Age	Numerik	Usia responden dalam tahun (rentang 17–31 tahun)
5	Stay	Numerik	Lama tinggal di Jepang dalam tahun (rentang 1–10 tahun)
6	Japanese	Numerik	Kemampuan berbahasa Jepang, skala 1–5
7	English	Numerik	Kemampuan berbahasa Inggris, skala 1–5
8	Religion	Kategorikal	Keterlibatan aktif dalam keagamaan (Yes / No)
9	Suicide	Kategorikal	Riwayat ideasi bunuh diri (Yes / No)
10	ToSC	Numerik	Total skor Social Connectedness (skala 8–48); skor lebih tinggi menunjukkan konektivitas sosial lebih baik
11	APD	Numerik	Skor Acculturative Stress subskala Perceived Discrimination
12	AHome	Numerik	Skor Acculturative Stress subskala Homesickness
13	APH	Numerik	Skor Acculturative Stress subskala Perceived Hate
14	Afear	Numerik	Skor Acculturative Stress subskala Fear
15	ACS	Numerik	Skor Acculturative Stress subskala Culture Shock
16	AGuilt	Numerik	Skor Acculturative Stress subskala Guilt
17	AMiscell	Numerik	Skor Acculturative Stress subskala Miscellaneous
18	Partner	Numerik	Dukungan sosial dari pasangan, skala 1–7
19	Friends	Numerik	Dukungan sosial dari teman, skala 1–7
20	Parents	Numerik	Dukungan sosial dari orang tua, skala 1–7
21	Relative	Numerik	Dukungan sosial dari kerabat, skala 1–7
22	Profess	Numerik	Dukungan sosial dari profesional (konselor/psikolog), skala 1–7

23	Doctor	Numerik	Dukungan sosial dari dokter, skala 1–7
24	Reli	Numerik	Dukungan sosial dari pemimpin agama, skala 1–7
25	Alone	Numerik	Frekuensi mengatasi masalah secara mandiri, skala 1–7
26	Others	Numerik	Dukungan sosial dari pihak lain, skala 1–7
27	ToDep	Numerik	Total skor depresi berdasarkan PHQ-9 (rentang 0–25)
28	Dep	Kategorikal	Label target: status depresi (Yes / No) — variabel yang diprediksi

3.2.2 Karakteristik Dataset

Berdasarkan hasil analisis terhadap dataset yang digunakan, berikut adalah karakteristik umum data yang menjadi dasar penelitian ini.

Tabel 3. 6 Karakteristik Dataset

Karakteristik	Nilai	Keterangan
Jumlah data (baris)	268 record	Setelah seleksi fitur dari dataset awal
Jumlah fitur (kolom)	28 kolom	27 fitur prediktor + 1 label target (Dep)
Sumber data	Kaggle (Nguyen et al., 2019)	Survei mahasiswa di universitas multikultural Jepang
Jenis data	Campuran (numerik & kategorikal)	4 fitur kategorikal, 24 fitur numerik
Missing values	0 (tidak ada)	Dataset sudah bersih, tidak ada nilai kosong
Distribusi kelas (Dep=No)	172 record (64,2%)	Mahasiswa tidak terindikasi depresi
Distribusi kelas (Dep=Yes)	96 record (35,8%)	Mahasiswa terindikasi depresi
Komposisi gender	170 perempuan (63,4%), 98 laki-laki (36,6%)	-
Komposisi status mahasiswa	201 internasional (75,0%), 67 domestik (25,0%)	-
Jenjang studi	247 sarjana (92,2%), 21 pascasarjana (7,8%)	-
Riwayat ideasi bunuh diri	61 Ya (22,8%), 207 Tidak (77,2%)	Fitur penting dalam prediksi depresi
Usia responden	Min: 17 | Maks: 31 | Rerata: 20,87 tahun	-
Lama tinggal di Jepang	Min: 1 | Maks: 10 | Rerata: 2,15 tahun	-

Rentang skor ToSC	Min: 8 | Maks: 48 | Rerata: 37,47	Skor Social Connectedness (8–48)
Rentang skor ToDep (PHQ-9)	Min: 0 | Maks: 25 | Rerata: 8,19	Skor total depresi PHQ-9

Berdasarkan Tabel 3.2, dataset memiliki distribusi kelas yang tidak seimbang (imbalanced), di mana kelas Dep=No (tidak depresi) berjumlah 172 record (64,2%) dan kelas Dep=Yes (depresi) berjumlah 96 record (35,8%). Kondisi ini merupakan hal yang wajar dalam dataset kesehatan mental dan menjadi pertimbangan penting dalam pemilihan metode evaluasi model, sehingga metrik presisi, recall, dan F1-score digunakan secara bersama-sama selain akurasi. Selain itu, proporsi mahasiswa internasional yang mencapai 75,0% dari total data mencerminkan karakteristik unik dataset ini yang berfokus pada konteks multikultural.

3.2.3 Contoh Data

Tabel 3.3 berikut menyajikan sebagian contoh data dari dataset yang digunakan, khususnya pada fitur-fitur utama yang paling relevan dalam penelitian ini. Kolom yang ditampilkan merupakan representasi dari keempat kelompok fitur, yaitu demografis, psikososial, konektivitas sosial, dan label target.

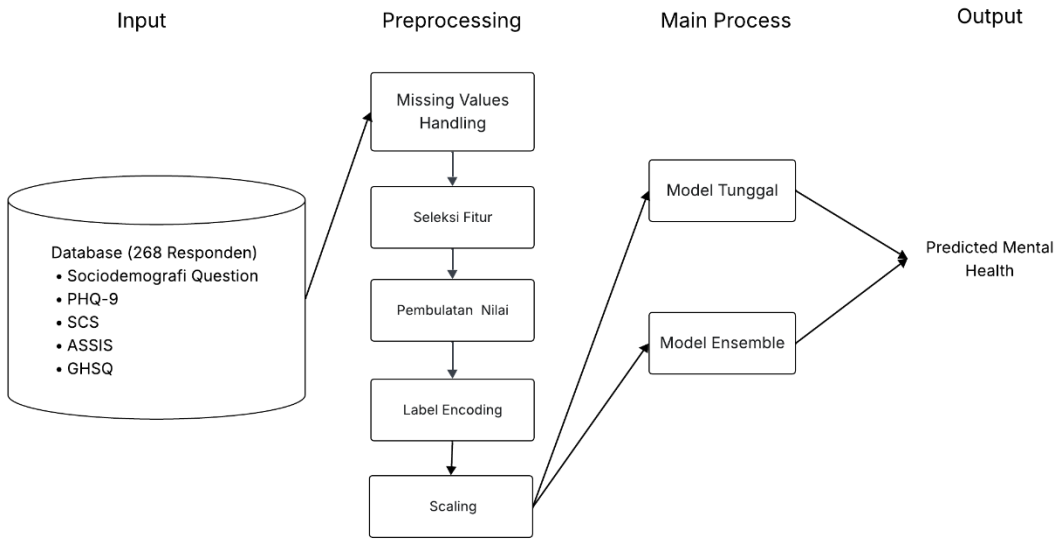
Tabel 3. 7 Contoh Data Sampel Dataset Kesehatan Mental Mahasiswa

Gender	Inter_Dom	Age	Stay	Religion	Sucide	ToSC	APD	ToDep	Dep
Male	Inter	24	5	Yes	No	34	23	0	**No**
Male	Inter	28	1	No	No	48	8	2	**No**
Male	Inter	25	6	Yes	No	41	13	2	**No**
Female	Inter	29	1	No	No	37	16	3	**No**
Female	Inter	28	1	No	No	37	15	3	**No**

Keterangan: *Gender* (jenis kelamin), *inter_dom* (status mahasiswa internasional/domestik), *Age* (usia), *Stay* (lama tinggal dalam tahun), *Religion* (keterlibatan keagamaan), *Suicide* (riwayat ideasi bunuh diri), ToSC (skor konektivitas sosial), APD (skor stres akulturasi-diskriminasi), ToDep (total skor depresi PHQ-9), Dep (label target: status depresi).

3.3 Perancangan Sistem

Rancangan sistem ini menggambarkan proses prediksi kesehatan mental berdasarkan data yang mencakup variabel sosiodemografis, PHQ-9, SCS, dan GHSQ melalui empat model seperti SVM, RF, XGBoost, dan LR yang telah divalidasi dan digabungkan dengan metode ensemble untuk menghasilkan prediksi yang akurat. Rancangan sistem untuk penelitian ini ditunjukkan dalam Gambar 3.2.



Gambar 3. 2 Rancangan Sistem

3.3.1 Missing Values Handling

Adanya data yang hilang merupakan tantangan yang umum dalam penelitian berbasis survei, yang sering kali disebabkan oleh keengganan responden untuk mengungkapkan informasi sensitif atau gangguan teknis selama proses penginputan data. Kehadiran nilai-nilai yang hilang ini berpotensi menimbulkan bias sistemik dan menurunkan kinerja komputasi algoritma pembelajaran mesin. Analisis data awal terhadap 286 responden mengungkapkan pola data yang hilang

yang meluas. Secara spesifik, 18 nilai hilang pada atribut inti, sementara 16 nilai tidak ada pada kolom target yang diberi label ‘Dep.’

Para peneliti menetapkan kebijakan pembersihan data dengan fokus pada memastikan integritas label target, dengan memanfaatkan teknik Dropping Rows. Baris yang sesuai dengan responden yang tidak memberikan nilai apa pun pada kolom “ToDep” (Total Skor Depresi) kemudian dihapus. Langkah ini diambil untuk memastikan bahwa model hanya belajar dari data dengan kebenaran dasar yang valid. Label klasifikasi risiko depresi tidak dapat ditentukan secara akurat tanpa skor ‘ToDep’. Oleh karena itu, penghapusan baris merupakan pendekatan optimal untuk menjaga kualitas pelatihan model dibandingkan dengan imputasi data, yang berisiko menyesatkan model.

3.3.2 Seleksi Fitur

Proses seleksi fitur diterapkan untuk meningkatkan efisiensi komputasi dan memastikan bahwa model tersebut berfokus pada variabel-variabel yang memiliki korelasi kuat dengan risiko depresi di kalangan mahasiswa. Para peneliti memulai dengan kumpulan awal yang terdiri dari 50 atribut dan menyaring kumpulan data tersebut menjadi 28 variabel prediktor terpilih melalui proses penyaringan yang ketat. Fitur-fitur yang dimaksud dikelompokkan ke dalam empat kategori berikut: Komponen pertama dari kumpulan data mencakup informasi sosiodemografis, seperti jenis kelamin, status tempat tinggal, status akademik, usia, dan lama tinggal. Komponen kedua mencakup kemahiran bahasa, termasuk kemahiran dalam bahasa Jepang dan Inggris. Komponen ketiga berfokus pada spiritualitas dan variabel risiko kritis, seperti agama dan bunuh diri. Komponen keempat terdiri dari indikator

psikososial yang diperoleh dari instrumen standar, termasuk Skala Keterikatan Sosial (ToSC), Skala Stres Akulturasi (ASSIS), dan Skala Pencarian Bantuan Kesehatan Umum (GHSQ).

Pada tahap studi ini, para peneliti memutuskan untuk menghilangkan atribut "Internet" dari daftar prediktor. Keputusan ini didasarkan pada dua pertimbangan utama. Pertama, hasil analisis awal menunjukkan bahwa perilaku penggunaan internet tidak berkontribusi secara signifikan terhadap pola risiko depresi dibandingkan dengan variabel psikososial inti seperti dukungan sosial (ToSC). Kedua, atribut ini menunjukkan tingkat nilai yang hilang yang tinggi (15%), yang menyebabkan keputusan untuk mengeluarkannya, karena dianggap lebih bijaksana untuk menjaga stabilitas model daripada mengisi nilai yang hilang pada fitur dengan pengaruh minimal.

Selain itu, variabel ToAS (Total Acculturative Stress) dikecualikan untuk menghindari masalah multikolinearitas, karena nilainya merupakan akumulasi linier dari sub-dimensi ASSIS yang sudah diwakili secara individual. Disepakati bahwa karakteristik biner tambahan (yang diakhiri dengan simbol garis bawah) harus dihilangkan karena redundansi dengan karakteristik asli. Akhirnya, variabel ToDep (Total Depression Score) dihapus dari matriks fitur input untuk mencegah kebocoran data, mengingat variabel target biner ('Dep') diperoleh langsung dari skor ini. Penerapan skema seleksi fitur yang ketat ini memungkinkan model untuk mengekstraksi pola risiko depresi secara murni dan efisien. Akibatnya, integritas hasil klasifikasi akhir terjamin pada tingkat ilmiah.

3.3.3 Tranformasi Data Katergorikal (*Label Encoding*)

Tahap pra-pemrosesan data tekstual dalam penelitian ini dilakukan dengan menggunakan teknik Label Encoding. Variabel kategorikal, termasuk jenis kelamin (gender), tingkat pendidikan (academic), dan status depresi (dep) ditransformasikan ke dalam representasi numerik agar dapat diproses oleh algoritma machine learning.

Sebaliknya, One-Hot Encoding membuat kolom baru untuk setiap kategori. Di sisi lain, Label Encoding bekerja dengan menetapkan nilai bilangan bulat unik ke setiap label kategori dalam satu kolom (Hancock & Khoshgoftaar, 2020). Pemilihan teknik ini didasarkan pada efisiensi dimensi dataset. Mengingat jumlah fitur yang terbatas dan ukuran sampel yang relatif kecil, penerapan Label Encoding mencegah sparsity atau kepadatan data yang rendah yang berpotensi menurunkan kinerja generalisasi model (Pargent et al., 2022). Pemilihan teknik ini didasarkan pada pertimbangan efisiensi dimensi dataset. Label Encoding ditandai dengan jumlah fitur yang terbatas dan ukuran sampel yang relatif kecil. Pendekatan ini digunakan untuk menghindari dimensi tinggi dan sparsity yang melekat pada One-Hot Encoding. Pendekatan ini diharapkan berkontribusi pada pemeliharaan kepadatan data efektif yang lebih tinggi, yang pada gilirannya dapat meningkatkan kinerja generalisasi model.

Dataset tersebut berisi variabel kategorikal yang harus dikonversi ke format numerik agar dapat diproses oleh algoritma. Penerapan transformasi pada atribut difasilitasi oleh LabelEncoder. Jenis kelamin dikonversi ke representasi biner (0 dan 1); prestasi akademik dikonversi ke representasi biner (0 dan 1); dan variabel Dep (Target) dikonversi ke 0 (Tidak Depresi) dan 1 (Depresi).

3.3.4 Scaling Fitur *(Z-Score Normalization)*

Fitur-fitur dalam dataset penelitian ini memiliki rentang nilai yang sangat bervariasi, misalnya variabel usia mahasiswa (17–31 tahun) dan variabel skor *Social Connectedness Scale* (ToSC) dari skor 8–48. Perbedaan skala ini dapat menyebabkan masalah signifikan pada proses pelatihan model. Algoritma yang berbasis pada perhitungan jarak seperti *Support Vector Machine* (SVM) atau algoritma yang menggunakan optimasi berbasis gradien (gradient descent) seperti *Logistic Regression* cenderung bias terhadap fitur yang memiliki rentang angka besar. Tanpa penskalaan, fitur dengan skala besar akan mendominasi perhitungan bobot, sementara fitur berskala kecil akan dianggap kurang penting oleh model (Nawi et al., 2013).

Dalam penelitian ini, diputuskan untuk menggunakan teknik Standarisasi atau dikenal dengan Z-Score Normalization sebagai metode penskalaan final. Teknik ini mentransformasikan distribusi fitur sehingga memiliki nilai rata-rata (mean) nol dan standar deviasi satu. Secara matematis, proses perhitungan Z-Score dirumuskan sebagai berikut:

$$z = \frac{x - \mu}{\sigma}$$

Keterangan:
z = Nilai fitur hasil standarisasi (*Standardized score*).
x = Nilai asli fitur sebelum dilakukan penskalaan.
μ = Rata-rata (*mean*) dari seluruh nilai pada fitur tersebut.
σ = Standar deviasi atau simpangan baku dari fitur tersebut.

Proses ini mentransformasi data sehingga memiliki rata-rata 0 dan standar deviasi 1. Penskalaan ini diterapkan pada data latih (X_train) dan data uji (X_test)

untuk mencegah dominasi fitur dengan rentang nilai besar dan mempercepat konvergensi algoritma, khususnya pada SVM dan Logistic Regression.

Pemilihan Z-Score Normalization (*StandardScaler*) didasarkan pada karakteristik instrumen akulturasi stres (ASSIS) yang datanya cenderung mengikuti distribusi normal. Berbeda dengan Min-Max yang sensitif terhadap pencilan (outliers), *Z-Score* mempertahankan sebaran data asli sehingga memberikan bobot yang adil bagi fitur dengan rentang nilai berbeda (seperti usia vs total skor stres).

Proses ini mentransformasi data sehingga memiliki rata-rata 0 dan standar deviasi 1. Penskalaan ini diterapkan pada data latih (*X_train*) dan data uji (*X_test*) untuk mencegah dominasi fitur dengan rentang nilai besar dan mempercepat konvergensi algoritma, khususnya pada SVM dan Logistic Regression.

Pemilihan *StandardScaler* didasarkan pada karakteristik instrumen akulturasi stres (ASSIS) yang datanya cenderung mengikuti distribusi normal. Berbeda dengan *Min-Max* yang sensitif terhadap pencilan (*outliers*), Z-Score mempertahankan sebaran data asli sehingga memberikan bobot yang adil bagi fitur dengan rentang nilai berbeda seperti halnya usia dengan total skor stres.

3.4 Prediksi Menggunakan Model Tunggal

Tahap prediksi, yang menggunakan model tunggal, merupakan langkah dasar dalam mengevaluasi keefektifan masing-masing algoritma secara terpisah. Evaluasi ini dilakukan sebelum proses selanjutnya, yaitu integrasi sistem ensemble. Tujuan dari pendekatan ini adalah untuk membangun landasan perbandingan yang kokoh, sehingga memungkinkan pengukuran objektif terhadap karakteristik dan

kinerja dasar masing-masing algoritma berdasarkan fitur-fitur psikososial yang tersedia dalam dataset.

Studi ini menggunakan empat algoritma yang berbeda, masing-masing memiliki paradigma komputasi yang unik yaitu Logistic Regression (linear), Support Vector Machine (kernel), Random Forest, dan XGBoost (berbasis pohon keputusan). Penggunaan arsitektur yang berbeda-beda ini memungkinkan peneliti untuk menganalisis bagaimana setiap model memproses pola data kesehatan mental di kalangan mahasiswa, yang secara khas bersifat non-linear dan kompleks. Semua model dievaluasi melalui skema K-Fold Cross Validation untuk memastikan stabilitas hasil dan meminimalkan risiko overfitting pada sampel yang terbatas. Kinerja model-model mandiri ini selanjutnya akan berfungsi sebagai tolok ukur utama untuk menilai signifikansi peningkatan kualitas prediksi selama tahap Ensemble Majority Voting.

3.5 Prediksi Menggunakan Model Ensemble

Setelah evaluasi terhadap masing-masing model, langkah selanjutnya adalah mengintegrasikan hasil prediksi ke dalam arsitektur ensemble. Pendekatan ini bertujuan untuk menggabungkan kemampuan prediktif berbagai algoritma guna menghasilkan keluaran yang lebih stabil, akurat, dan kurang bias. Sistem ensemble, yang merupakan model matematis canggih, menggunakan mekanisme konsensus yang mengintegrasikan berbagai algoritma dengan karakteristik matematis yang berbeda-beda. Pendekatan ini memungkinkan sistem untuk memanfaatkan kelebihan masing-masing model dasar, sekaligus mengimbangi kelemahan masing-masing model.

Penelitian ini menggunakan metode *Average of Majority Votes* (AMV), yang mengkonsolidasikan keputusan yang diperoleh dari empat model individual sebelumnya. Arsitektur ini dirancang untuk memberikan tingkat kepastian diagnostik yang lebih tinggi, terutama pada dataset kesehatan mental dengan kompleksitas fitur yang tinggi. Penelitian selanjutnya akan mencakup perbandingan antara hasil model ensemble dan kinerja model tunggal yang paling efektif. Hal ini dilakukan untuk memastikan signifikansi kontribusi metode integrasi dalam meningkatkan kualitas deteksi risiko depresi di kalangan mahasiswa.

3.6 Eksperimen dan Evaluasi

Tahap eksperimen pada penelitian ini dilakukan menggunakan Google Colab sebagai lingkungan komputasi berbasis *cloud*. Seluruh proses analisis data, pemodelan, dan pelatihan model machine learning dijalankan dengan bahasa pemrograman *Python*, memanfaatkan berbagai *library* pendukung seperti *scikit-learn* untuk pengolahan data dan pembangunan model, serta *NumPy* dan *Pandas* untuk manajemen data. Penggunaan Google Colab memungkinkan pemanfaatan sumber daya komputasi GPU/TPU yang disediakan secara daring, sehingga proses pelatihan model dapat berjalan lebih optimal tanpa bergantung pada perangkat keras lokal.

Eksperimen dan evaluasi dalam penelitian ini difokuskan pada dua aspek utama. Pertama, evaluasi performa klasifikasi yang mencakup pengujian dan perbandingan keenam model (empat model tunggal dan dua pendekatan ensemble) menggunakan metrik akurasi, presisi, recall, dan F1-Score, serta uji signifikansi statistik Paired T-Test. Kedua, pengukuran efisiensi komputasi yang mencakup

waktu training, waktu inferensi, konsumsi RAM, penggunaan CPU, dan Cyclomatic Complexity. Kedua aspek ini dikombinasikan melalui Skor Komposit untuk menentukan model yang paling optimal secara menyeluruh, sesuai dengan rumusan masalah penelitian.

Pada penelitian kesehatan mental, metrik-metrik tersebut penting karena sebagian besar dataset bersifat tidak seimbang misalnya jumlah mahasiswa tanpa gejala lebih banyak daripada yang berisiko. Oleh karena itu, model tidak hanya dinilai dari *overall accuracy*, tetapi juga dari kemampuan mendeteksi kelas risiko depresi. Selain itu, aspek efisiensi komputasi menjadi pertimbangan penting apabila sistem akan diimplementasikan sebagai Decision Support System pada lingkungan dengan sumber daya terbatas.

a. *Accuracy*

Accuracy menggambarkan proporsi prediksi yang benar terhadap seluruh data. Meskipun sering digunakan, metrik ini kurang informatif bila terjadi ketidakseimbangan kelas, karena model dapat terlihat akurat meskipun gagal mendeteksi individu yang berisiko. Rumusnya sebagai berikut:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{3.1}$$

Keterangan:
TP = True Positive
TN = True Negative
FP = False Positive
FN = False Negative

b. *Precision*

Precision mengukur sejauh mana prediksi positif yang dibuat model benar-benar merupakan kasus positif. Dalam prediksi kesehatan mental, nilai Precision

yang tinggi berarti model jarang memberikan alarm palsu (*false positives*). Rumusnya sebagai berikut:

$$Precision = \frac{TP}{TP+FP} \quad (3.2)$$

c. *Recall*

Recall menunjukkan kemampuan model dalam menemukan seluruh kasus positif dalam data. Metrik ini sangat penting dalam konteks kesehatan mental, karena kesalahan FN berarti individu berisiko tidak terdeteksi sehingga tidak menerima intervensi yang diperlukan. Rumusnya sebagai berikut:

$$Recall = \frac{TP}{TP+FN} \quad (3.3)$$

d. *F1-Score*

F1-Score merupakan *harmonic mean* antara *Precision* dan *Recall*. *Metriks* ini digunakan ketika penyeimbangan antara kemampuan mendeteksi kasus (*Recall*) dan ketepatan prediksi (*Precision*) diperlukan, terutama pada dataset tidak seimbang. Rumusnya sebagai berikut:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (3.4)$$

3.6.1 Konfigurasi Parameter Model

Setiap model dalam penelitian ini dikonfigurasi dengan parameter yang telah ditentukan berdasarkan titik plateau konvergensi yang diperoleh dari proses observasi training. Tabel 3.5 berikut merangkum konfigurasi parameter untuk masing-masing model.

Tabel 3. 8 Konfigurasi Parameter Model

No	Model	Parameter	Nilai
1	Logistic Regression	*max_iter* dan *random_state*	218 dan 42
2	Support Vector Machine	*probability, max_iter* dan *random_state*	True, 466 dan 42
3	Random Forest	*n_estimators* dan *random_state*	197 42
4	XGBoost	*n_estimators, learning_rate, eval_metric* dan *random_state*	2003, 0.05, logloss dan 42
5	Ensemble Majority Voting	*voting, Base learners*	hard, LR, SVM, RF, XGBoost
6	Ensemble Stacking	*Base learners, Meta-learner passthrough*	LR, SVM, RF, XGBoost Logistic Regression, False

Parameter *random_state*=42 digunakan secara konsisten pada seluruh model untuk memastikan hasil yang dapat direproduksi (reproducible). Untuk model Ensemble Stacking, Logistic Regression dipilih sebagai meta-learner karena sifatnya yang stabil dan interpretatif dalam menggabungkan prediksi dari base learners. Sementara itu, parameter *passthrough*=False berarti *meta-learner* hanya menerima output prediksi dari *base learners*, bukan fitur asli dataset.

3.6.2 Kasus-kasus Eksperimen

Penelitian ini merancang tujuh kasus eksperimen yang mencakup pengujian empat model tunggal, dua pendekatan ensemble, dan satu tahap perbandingan komprehensif antar seluruh model. Tabel 3.6 berikut menyajikan rancangan lengkap kasus-kasus eksperimen yang dijalankan.

Tabel 3. 9 Rancangan Kasus-Kasus Eksperimen

No	Kasus Eksperimen	Model & Konfigurasi	Metrik Evaluasi	Uji Statistik	Tujuan
1	Eksperimen Model Tunggal –	*Logistic Regression (max_iter=218)*	Akurasi, Presisi, Recall, F1-Score, Specificity, ROC-AUC	Paired T-Test vs semua model lain	Mengukur performa baseline model linear

	Logistic Regression				
2	Eksperimen Model Tunggal – SVM	*SVM (max_iter=466, probability=True)*	Akurasi, Presisi, Recall, F1-Score, Specificity, ROC-AUC	Paired T-Test vs semua model lain	Mengukur performa model berbasis margin pada data psikososial
3	Eksperimen Model Tunggal – Random Forest	*Random Forest (n_estimators=197)*	Akurasi, Presisi, Recall, F1-Score, Specificity, ROC-AUC	Paired T-Test vs semua model lain	Mengukur performa model berbasis pohon keputusan berganda
4	Eksperimen Model Tunggal – XGBoost	*XGBoost (n_estimators=2003, lr=0.05)*	Akurasi, Presisi, Recall, F1-Score, Specificity, ROC-AUC	Paired T-Test vs semua model lain	Mengukur performa model boosting dengan iterasi tinggi
5	Eksperimen Ensemble – Majority Voting	*Voting Classifier (Hard) Base: LR, SVM, RF, XGBoost*	Akurasi, Presisi, Recall, F1-Score, Specificity, ROC-AUC	Paired T-Test vs semua model lain	Mengevaluasi apakah kombinasi suara mayoritas meningkatkan performa
6	Eksperimen Ensemble – Stacking	*Stacking Classifier Base: LR, SVM, RF, XGBoost Meta: Logistic Regression*	Akurasi, Presisi, Recall, F1-Score, Specificity, ROC-AUC	Paired T-Test vs semua model lain	Mengevaluasi apakah meta-learning menghasilkan model paling optimal
7	Perbandingan & Uji Signifikansi Antar Model	*Semua model (6 model di atas)*	Seluruh metrik	Paired T-Test (semua kombinasi pasangan model)	Menentukan model terbaik secara statistik yang signifikan
8	Pengukuran Efisiensi Komputasi Seluruh Model	Semua model (6 model di atas) dengan 5-Fold Cross-Validation	Waktu Training (s), Waktu Inferensi (ms/sampel), RAM (MB), CPU (%), Cyclomatic Complexity (M)	Skor Komposit Min-Max (equal weight, 9 dimensi)	Menentukan model paling optimal berdasarkan keseimbangan performa dan efisiensi komputasi

Kasus eksperimen 1 hingga 4 merupakan pengujian model tunggal yang berfungsi sebagai baseline performa. Kasus 5 dan 6 merupakan pendekatan ensemble yang bertujuan mengintegrasikan kekuatan masing-masing model tunggal. Kasus 5 menggunakan mekanisme Hard Majority Voting di mana keputusan akhir ditentukan berdasarkan suara terbanyak dari keempat model. Kasus 6 menggunakan mekanisme Stacking di mana prediksi keempat model dijadikan fitur input bagi meta-learner (Logistic Regression) untuk menghasilkan prediksi akhir. Kasus 7 merupakan tahap analisis komprehensif yang membandingkan seluruh model secara statistik menggunakan Paired T-Test pada semua kombinasi pasangan model. Kasus 8 merupakan tambahan baru yang secara khusus mengukur efisiensi komputasi seluruh model, menggunakan metrik waktu training, waktu inferensi, RAM, CPU, dan Cyclomatic Complexity, kemudian mengintegrasikannya dengan hasil performa melalui Skor Komposit untuk menentukan model paling optimal secara menyeluruh.

3.6.3 Prosedur Evaluasi

Seluruh kasus eksperimen di atas dijalankan menggunakan prosedur evaluasi yang seragam berbasis 5-Fold Cross Validation. Tabel 3.7 berikut menjelaskan tahapan prosedur evaluasi yang diterapkan.

Tabel 3. 10 Prosedur Evaluasi

Tahap	Keterangan	Detail
Pembagian Data	Stratified K-Fold Cross Validation	5-Fold (k=5), data dibagi menjadi 5 bagian secara stratifikasi untuk menjaga proporsi kelas pada setiap fold
Preprocessing	Scaling fitur menggunakan Z-Score Normalization	StandardScaler di-fit pada data training setiap fold, lalu ditransformasikan ke data validasi (mencegah data leakage)

Training & Prediksi	Setiap model dilatih pada 4 fold dan diuji pada 1 fold	Proses diulang sebanyak 5 kali sehingga setiap data pernah menjadi data uji tepat satu kali
Pengumpulan Skor	Skor metrik dikumpulkan per fold	Setiap fold menghasilkan 6 nilai metrik (akurasi, presisi, recall, F1-score, specificity, ROC-AUC) sehingga total 5 nilai per metrik per model
Pengukuran Efisiensi	Waktu Training, Waktu Inferensi, RAM, CPU, dan Cyclomatic Complexity diukur pada setiap fold	Waktu diukur menggunakan time.perf_counter(); RAM menggunakan tracemalloc; CPU menggunakan psutil dengan CPUSampler thread terpisah; CC dihitung statis menggunakan rumus McCabe ($M = E - N + 2P$)
Uji Signifikansi	Paired T-Test antar semua pasangan model	Menggunakan skor per-fold sebagai pasangan data (n=5). Hipotesis nol: tidak ada perbedaan signifikan antar model. Tingkat signifikansi $\alpha = 0,05$
Penentuan Model Terbaik	Berdasarkan rata-rata seluruh metrik + hasil Paired T-Test	Model dinyatakan terbaik jika memiliki rata-rata metrik tertinggi dan terbukti unggul secara statistik signifikan ($p < 0,05$)

Penggunaan Stratified K-Fold memastikan proporsi kelas target (Dep=Yes dan Dep=No) terjaga pada setiap *fold*, sehingga evaluasi tidak bias terhadap kelas mayoritas. Seluruh proses *preprocessing*, khususnya Z-Score Normalization, dilakukan di dalam *loop* K-Fold untuk mencegah kebocoran data (data *leakage*) yang dapat mengakibatkan evaluasi performa yang terlalu optimistik. Pengukuran efisiensi komputasi juga dilakukan di dalam loop K-Fold yang sama sehingga hasilnya mencerminkan kondisi training yang konsisten dan dapat direproduksi.

3.6.4 Pengukuran Efesiensi Komputasi

Dalam konteks penelitian machine learning, menentukan model yang paling optimal tidak cukup hanya dengan melihat seberapa akurat model tersebut dalam melakukan prediksi. Sebuah model yang memiliki akurasi tinggi belum tentu menjadi pilihan terbaik apabila dalam prosesnya membutuhkan waktu komputasi yang sangat lama, mengonsumsi memori yang besar, atau membebani prosesor

secara berlebihan. Hal ini menjadi pertimbangan penting terutama ketika sistem prediksi akan diimplementasikan dalam kondisi nyata yang memiliki keterbatasan sumber daya komputasi, seperti pada server dengan kapasitas terbatas atau perangkat yang digunakan di lingkungan institusi pendidikan (Domingos, 2012). Oleh karena itu, penelitian ini tidak hanya mengevaluasi performa klasifikasi setiap model, tetapi juga mengukur efisiensi komputasinya secara langsung melalui lima metrik berikut.

Tabel 3. 11 Metrik Efesiensi Komputasi

Metrik Efisiensi	Cara Pengukuran	Interpretasi
Waktu Training (detik)	time.perf_counter() sebelum dan sesudah model.fit()	Durasi model "belajar" dari data training. Lebih kecil = lebih efisien.
Waktu Inferensi (ms/sampel)	Selisih waktu model.predict() dibagi jumlah sampel uji, dikonversi ke milidetik	Kecepatan model membuat satu prediksi. Lebih kecil = lebih responsif untuk deployment.
Konsumsi RAM (MB)	tracemalloc.get_traced_memory() puncak selama training, dikonversi ke MB	Memori puncak yang digunakan saat training. Lebih kecil = lebih hemat sumber daya.
Rata-rata CPU (%)	psutil.cpu_percent() disampling setiap 0,1 detik via thread terpisah (CPUSampler) selama training	Beban rata-rata prosesor selama training. Lebih kecil = lebih hemat komputasi.
Cyclomatic Complexity / CC (M)	Dihitung statis menggunakan rumus McCabe: M = E − N + 2P dari analisis flowchart algoritma	Mengukur kompleksitas alur logika algoritma. Lebih kecil = struktur lebih sederhana dan mudah dipelihara.

3.6.5 Cyclomatic Complexity (CC)

Cyclomatic Complexity (CC) adalah metrik statis yang mengukur kompleksitas alur logika suatu algoritma berdasarkan analisis struktur flowchart-nya. Metrik ini diperkenalkan oleh McCabe (1976) dan dihitung menggunakan rumus:

$$\boldsymbol{M = E - N + 2P} \qquad \textbf{(3.5)}$$

Keterangan: $\boldsymbol{M}$ = Cyclomatic Complexity; $\boldsymbol{E}$ = jumlah edge (panah/alur) pada flowchart; $\boldsymbol{N}$ = jumlah node (simpul/proses) pada flowchart; $\boldsymbol{P}$ = jumlah komponen terhubung (untuk flowchart tunggal, $P = 1$).

Berbeda dengan metrik runtime, CC dihitung secara statis dari analisis flowchart masing-masing algoritma, sehingga nilainya tidak berubah antarfold. Nilai CC yang lebih rendah menunjukkan bahwa alur logika algoritma lebih sederhana, lebih mudah dipahami, dan lebih mudah dipelihara. Berikut hasil perhitungan CC untuk setiap model disajikan pada Tabel 3.12.

Tabel 3. 12 Nilai Cyclomatic Complexity Setiap Model

Model	N (Node)	E (Edge)	P	CC (M)	Kategori Kompleksitas
Logistic Regression	18	19	1	3	Rendah (Sederhana)
SVM	20	21	1	3	Rendah (Sederhana)
Random Forest	21	25	1	6	Sedang (Moderat)
XGBoost	24	27	1	5	Sedang (Moderat)
Ensemble Majority Voting	22	25	1	5	Sedang (Moderat)
Ensemble Stacking	23	25	1	4	Rendah-Sedang

Berdasarkan Tabel 3.12, Logistic Regression dan SVM memiliki nilai CC terendah (M = 3) yang menunjukkan alur logika paling sederhana. Random Forest memiliki nilai CC tertinggi (M = 6) karena melibatkan banyak pohon keputusan paralel. Nilai CC ini kemudian digunakan sebagai salah satu dimensi dalam perhitungan Skor Komposit pada Kasus 8.

3.6.6 Skor Komposit untuk Penentuan Model Optimal

Untuk mengintegrasikan hasil performa klasifikasi dan efisiensi komputasi ke dalam satu ukuran tunggal, penelitian ini menggunakan Skor Komposit yang dihitung dari sembilan dimensi evaluasi dengan bobot yang sama (*equal weight*). Penggunaan *equal weight* dipilih karena penelitian ini bersifat eksploratoris dan

belum terdapat konsensus ilmiah tentang bobot optimal antara performa dan efisiensi pada konteks prediksi kesehatan mental mahasiswa (Géron, 2022).

Untuk dimensi performa klasifikasi (Accuracy, F1-Score, Precision, Recall), berlaku prinsip "lebih besar = lebih baik". Normalisasi dilakukan secara langsung menggunakan rumus:

$$\boldsymbol{dim_i(normal) = (X - X_{min})/(X_{max} - X_{min})} \quad \textbf{(3.6a)}$$

Untuk dimensi efisiensi komputasi (Waktu Training, Waktu Inferensi, RAM, CPU, dan Cyclomatic Complexity), berlaku prinsip "lebih kecil = lebih baik". Agar skor tetap konsisten yaitu nilai mendekati 1 selalu bermakna lebih baik, maka normalisasi dilakukan secara invers dengan rumus:

$$\boldsymbol{dim_i(invers) = (X_{max} - X)/(X_{max} - X_{min})} \quad \textbf{(3.6b)}$$

Di mana X adalah nilai asli model pada dimensi tersebut, X^{min} adalah nilai terkecil dan X^{max} adalah nilai terbesar di antara seluruh model pada dimensi yang sama. Hasilnya adalah nilai ternormalisasi dim_i yang selalu berada dalam rentang [0, 1], di mana 1 berarti terbaik dan 0 berarti terburuk pada dimensi tersebut.

Setelah kesembilan dimensi ternormalisasi, Skor Komposit dihitung sebagai rata-rata aritmetika sederhana:

$$\boldsymbol{Skor\ Komposit = \sum(dim_i)/9} \quad \textbf{(3.7)}$$

Di mana $\boldsymbol{dim_i}$ adalah nilai ternormalisasi dari setiap dimensi (i = 1, 2, ..., 9). Model dengan Skor Komposit tertinggi dinyatakan sebagai model paling optimal karena menawarkan keseimbangan terbaik antara performa klasifikasi, efisiensi komputasi, dan kesederhanaan struktural algoritma (McCabe, 1976).

BAB IV

PREDIKSI KESEHATAN MENTAL MENGGUNAKAN MODEL TUNGGAL LOGISTIC REGRESSION

4.1 Desain Sistem

Desain sistem pada bab ini, peneliti melakukan implementasi dan pengujian terhadap model tunggal pertama, yaitu Logistic Regression. Pemilihan algoritma ini didasarkan pada karakteristiknya yang efisien untuk klasifikasi biner dan kemampuannya dalam memberikan interpretasi probabilistik terhadap risiko depresi mahasiswa. Alur kerja prediksi pada model ini secara sistematis digambarkan pada Gambar 4.1.

Gambar 4. 1 Desain Sistem Model Tunggal

Berdasarkan Gambar 4.1, alur kerja sistem dimulai dengan memasukkan fitur-fitur psikososial (SCS, ASSIS, PHQ-9, dan sosiodemografi) yang telah melalui tahap standard scaling atau normalisasi Z-Score. Proses scaling ini krusial bagi Logistic Regression agar gradien pada fungsi optimasi tidak terdistorsi oleh perbedaan satuan antar fitur. Data tersebut kemudian diolah melalui fungsi logistik (sigmoid) untuk menghasilkan probabilitas risiko. Luaran akhir dari sistem ini adalah label prediksi Mental Health yang mengategorikan responden ke dalam kelompok "Depresi" atau "Sehat".

4.2 Model Logististic Regression (LR)

Logistic Regression (LR) merupakan metode statistik yang digunakan untuk memodelkan hubungan antara variabel independen dan variabel dependen biner. Meskipun disebut sebagai pendekatan regresi, metode ini diklasifikasin sebagai teknik klasifikasi. Model ini bekerja melalui proses pemetaan kombinasi linier dari model predictor ke probabilitas (Harris, 2021). Pemetaan ini dilakukan denga menggunakan fungsi transformasi yang dikenal sebagai fungsi logistic atau *sigmoid*. Secara umum, hubungan antara variabel prediktor dan probabilitas kejadian kelas positif dinyatakan melalui fungsi logit dapat dilihat pada Persamaan 3.15:

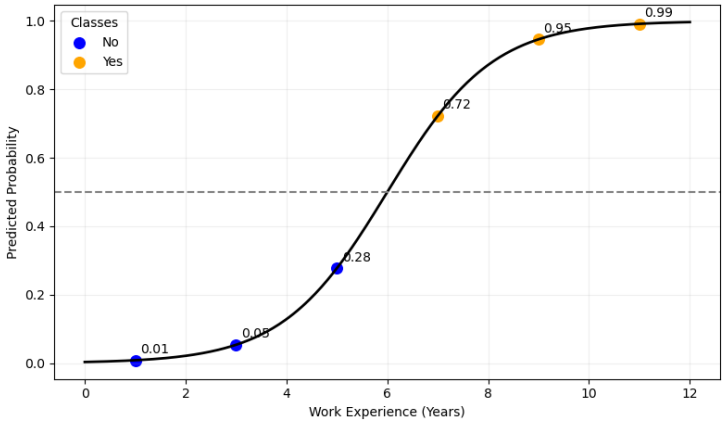
$$logit(p) = ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k \quad (4.1)$$

Keterangan:
p = peluang
$x_1, x_2, \ldots$ = fitur atau data masukan (misal: hubungan sosial, umur, dll)
β_0 = Intercept (titik potong)
$\beta_1, \beta_2, \ldots$ = koefisien atau "bobot" seberapa penting fitur tersebut.

Persamaan tersebut kemudian ditransformasikan menjadi probabilitas menggunakan fungsi *sigmoid* agar menghasilkan nilai antara 0 hingga 1 dan dihitung menggunakan Persamaan 3.15:

$$p = \frac{1}{1+e^{-z}} \quad (4.2)$$

dengan z = $\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k$. Transformasi ini memungkinkan model memberikan interpretasi probabilistik terhadap kelas yang diprediksi, misalnya probabilitas mahasiswa mengalami risiko kesehatan mental tertentu.



Gambar 4. 2 Kurva Model Logistic Regression

Gambar diatas menggambarkan kurva Logistic Regression yang digunakan untuk menggambarkan hubungan antara pengalaman kerja dan probabilitas seseorang mendapatkan promosi. Kurva logistik berbentuk *sigmoid* menggambarkan peningkatan probabilitas tidak linear seiring bertambahnya pengalaman kerja. Titik-titik berwarna menunjukkan data aktual yang terdiri dari dua kelas yaitu Yes dan No, sementara angka di setiap titik merepresentasikan probabilitas prediksi dari model. Garis putus-putus horizontal menandai threshold keputusan sebesar 0.5, yang digunakan untuk menentukan klasifikasi akhir. Secara keseluruhan, visualisasi ini menggambarkan bagaimana model regresi logistik melakukan estimasi probabilitas dan pengambilan keputusan pada masalah klasifikasi biner.

Parameter model diestimasi menggunakan pendekatan *Maximum Likelihood Estimation* (MLE), yang diwujudkan melalui m`inimisasi *Binary Cross-Entropy Loss.* Fungsi *loss* tersebut dirumuskan secara ringkas pada Persamaan 3.17:

$$L(\beta) = -\sum[y \ln(p) + (1-y)\ln(1-p)] \quad (3.17)$$

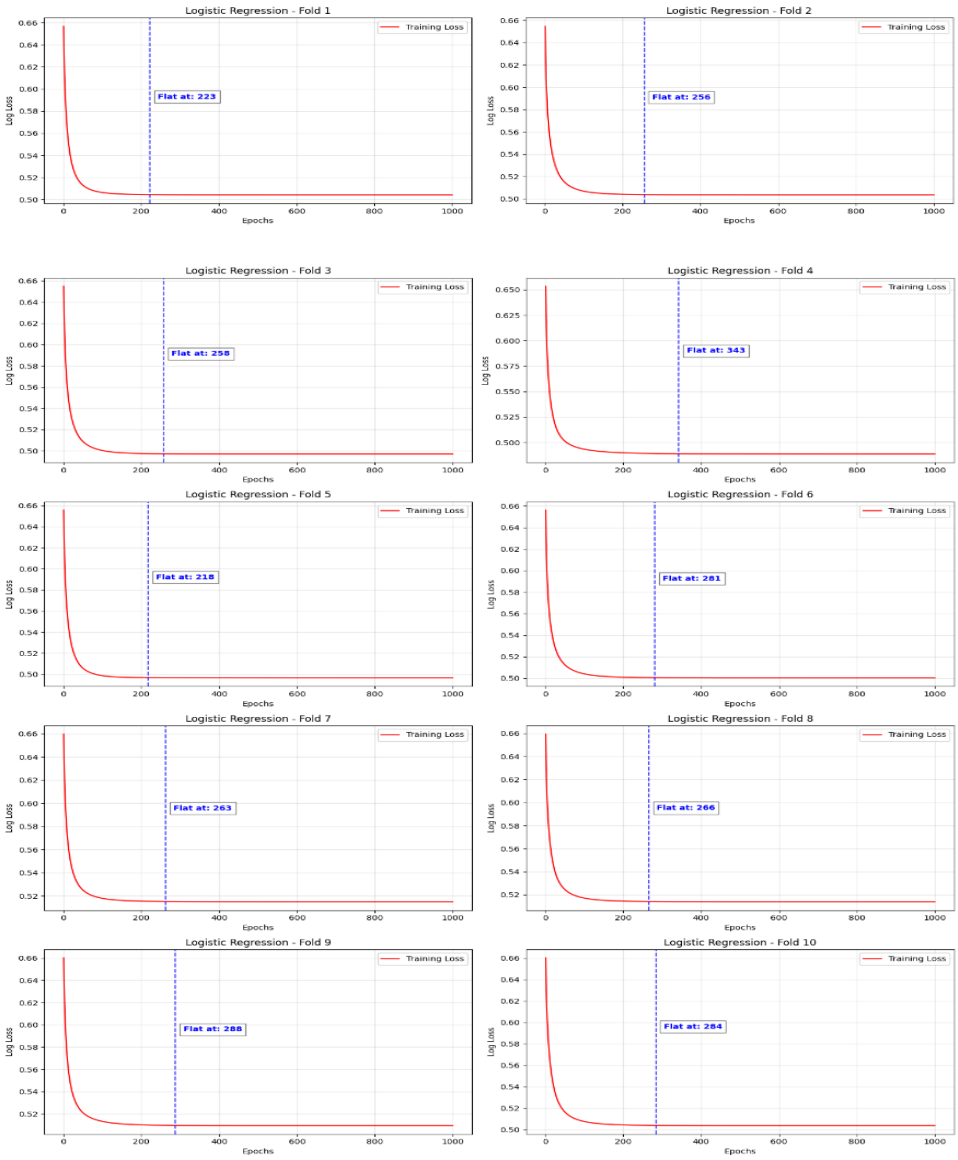
pendekatan ini memastikan model memilih parameter yang memberikan peluang tertinggi terhadap data observasi. Untuk mencegah *overfitting*, Logistic Regression juga dapat dilengkapi regularisasi seperti L2 (*ridge*) atau L1 (*lasso*) yang menambahkan penalti pada besarnya *koefisien*. Namun, penggunaan regularisasi bersifat opsional dan diterapkan sesuai desain penelitian.

Salah satu alasan Logistic Regression banyak digunakan dalam penelitian kesehatan mental adalah sifat model yang mudah diinterpretasikan. *Koefisien regresi* dapat dikonversi menjadi *odds ratio* melalui ekspresi sederhana e^{β_i}, yang menggambarkan perubahan peluang terjadinya kelas positif ketika sebuah *variabel* meningkat satu satuan. Kemampuan interpretasi ini membuat model cocok digunakan tidak hanya untuk prediksi, tetapi juga untuk memahami faktor-faktor yang memengaruhi kondisi kesehatan mental mahasiswa.

4.3 Analisis Konvergensi Model dan Hyperparameter Tuning

Proses penentuan nilai hyperparameter optimal pada model Logistic Regression dalam penelitian ini dilakukan melalui pendekatan analisis kurva konvergensi (convergence curve analysis). Pendekatan ini merupakan bagian dari proses hyperparameter tuning yang bertujuan menemukan nilai parameter terbaik secara empiris berdasarkan perilaku konvergensi model selama proses pelatihan, tanpa memerlukan pencarian grid yang komputasional.

Peneliti memantau pergerakan Log Loss menggunakan skema 10-Fold Cross Validation selama 1.000 epoch untuk menjamin stabilitas model pada berbagai subset data. Hasil visualisasi proses tersebut disajikan pada Gambar 4.3.



Gambar 4. 3 Analisis Konvergensi Logistic Regression (10-Fold Cross Validation)

Berdasarkan Gambar 4.3, seluruh iterasi menunjukkan pola penurunan loss yang konsisten. Garis merah (Training Loss) mengalami penurunan tajam pada 100 epoch pertama, yang mengindikasikan bahwa fitur-fitur psikososial dalam dataset memiliki korelasi yang kuat sehingga model mampu mengidentifikasi pola data secara cepat.

4.3.1 Hyperparameter yang Di-tuning

Hyperparameter yang di-tuning berserta rentang nilai yang diobservasi dan nilai optimal yang dipilih untuk model Logistic Regression dirangkum pada tabel 4.1 berikut.

Tabel 4.1 Hyperparameter Tuning Model Logistic Regression

HyperparametHyer	Rentang Nilai Diuji	Nilai Optimal	Keterangan
max_iter	1 – 1000 epoch (diamati per epoch)	218	Jumlah iterasi maksimum hingga model dianggap konvergen. Ditentukan berdasarkan titik *plateau kurva loss*
loss function	*log_loss* (Logistic Regression)	*log_loss*	Fungsi *loss* yang digunakan untuk mengukur kesalahan prediksi probabilitas pada setiap *epoch*
learning_rate	*adaptive* (eta0 = 0.001)	*adaptive*	Laju belajar adaptif memungkinkan penyesuaian otomatis agar konvergensi lebih halus dan stabil
threshold plateau	1e-4 (selisih max-min dalam window)	1e-4	Ambang batas perubahan *loss* yang dianggap tidak signifikan. Digunakan fungsi *find_plateau_point*() dengan *patience*=50
random_state	42 (tetap)	42	Digunakan untuk memastikan hasil eksperimen dapat direproduksi (reproducible)

Hyperparameter utama yang menjadi fokus tuning adalah *max_iter*, yaitu jumlah iterasi maksimum yang menentukan kapan model dianggap telah mencapai konvergensi. Nilai *max_iter* yang terlalu kecil menyebabkan model belum optimal (underfitting), sedangkan nilai yang terlalu besar hanya membuang sumber daya komputasi tanpa peningkatan performa yang berarti. Oleh karena itu, penentuan *max_iter* yang tepat melalui observasi kurva konvergensi merupakan langkah tuning yang kritis.

4.3.2 Prosedur Hyperparameter Tuning

Proses tuning dilakukan menggunakan fungsi *find_plateau_point*() yang bekerja dengan cara mendeteksi titik di mana perubahan nilai *loss function* sudah sangat kecil secara konsisten. Fungsi ini menggunakan dua parameter kunci, yaitu *threshold*=1e-4 yang merupakan ambang batas selisih maksimum-minimum loss dalam satu window, dan *patience*=50 yang merupakan jumlah epoch berturut-turut yang harus memenuhi kondisi threshold tersebut sebelum titik plateau dinyatakan tercapai.

Observasi dilakukan pada setiap fold dalam skema 10-Fold Cross Validation dengan total 1000 epoch per fold. Kurva training loss diamati secara visual dan komputasional menggunakan SGDClassifier dengan loss='log_loss' dan learning_rate='adaptive' (eta0=0.001). Pendekatan adaptive learning rate dipilih agar proses konvergensi lebih halus dan tidak bergejolak, sehingga titik plateau dapat terdeteksi dengan lebih akurat.

4.3.3 Hasil Hyperparameter Tuning

Titik *plateau* yang terdeteksi pada setiap fold selama proses tuning berlangsung ditampilkan dalam Tabel 4.2.

Tabel 4.2 Hasil Deteksi Titik Plateau per Fold Model Logistic Regression

Fold	Titik Plateau (Epoch)	Log Loss Akhir	Keterangan
Fold 1	223	~ 0.50	Loss mulai stabil pada epoch 223, perubahan < 1e-4 dalam 50 epoch berturut-turut
Fold 2	256	~ 0.50	Konvergensi tercapai lebih lambat dibanding Fold 1, Log Loss stabil di 0.50
Fold 3	258	~ 0.50	Plateau terdeteksi di sekitar epoch 258 dengan Log Loss yang konsisten
Fold 4	343	~ 0.49	Konvergensi paling lambat, namun menghasilkan Log Loss terbaik (0.49)

Fold 5	218	~ 0.50	Konvergensi tercepat — dijadikan acuan nilai max_iter optimal
Fold 6	281	~ 0.50	Titik plateau terdeteksi di epoch 281 dengan Log Loss stabil di 0.50
Fold 7	263	~ 0.51	Log Loss sedikit lebih tinggi (0.51), plateau tercapai pada epoch 263
Fold 8	266	~ 0.51	Log Loss 0.51, konvergensi tercapai pada epoch 266
Fold 9	288	~ 0.51	Konvergensi relatif lambat dengan Log Loss akhir 0.51
Fold 10	284	~ 0.50	Plateau terdeteksi mendekati epoch 284, Log Loss stabil di 0.50
Rata-rata	268.0	0.50	Rata-rata titik plateau dan Log Loss akhir dari seluruh fold
max_iter dipilih	218	~ 0.50	Diambil dari fold tercepat (Fold 5: epoch 218) sebagai nilai max_iter optimal

Berdasarkan Tabel 4.2, titik plateau pada seluruh fold berkisar antara epoch 200 hingga 288, dengan nilai rata-rata sebesar 268 epoch dan nilai tercepat pada epoch 218. Nilai epoch tercepat ini kemudian ditetapkan sebagai nilai optimal max_iter=218 yang digunakan pada model Logistic Regression dalam seluruh eksperimen penelitian ini. Pemilihan nilai ini memastikan bahwa model telah mencapai konvergensi yang stabil tanpa iterasi yang berlebihan, sehingga keseimbangan antara efisiensi komputasi dan performa prediksi dapat terjaga dengan baik.

4.3.4 Hasil Training Model Logistic Regression

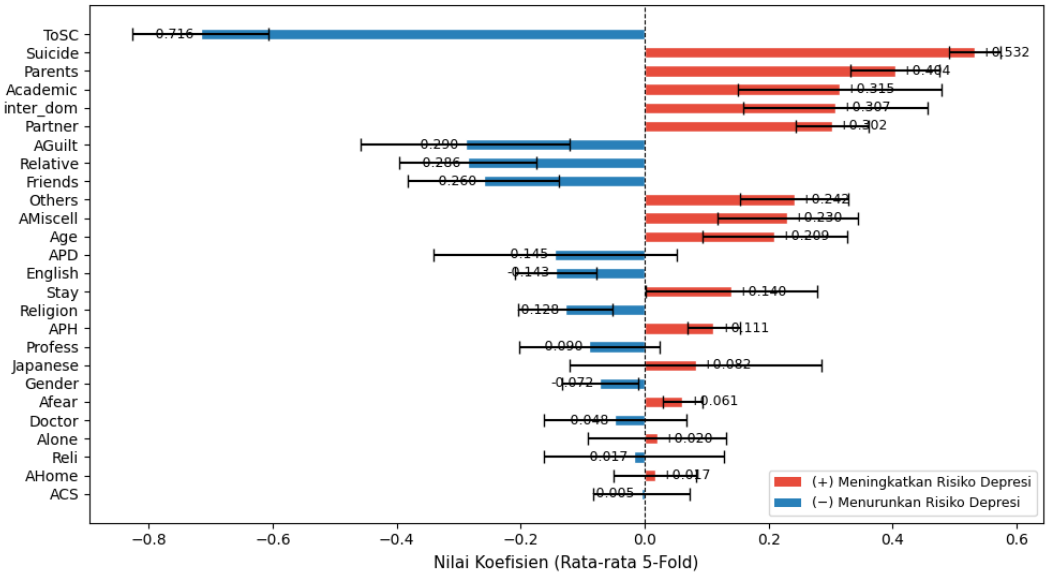
Setelah nilai hyperparameter optimal ditetapkan (max_iter=218), model Logistic Regression dilatih menggunakan skema 5-Fold Cross Validation dengan StandardScaler sebagai tahap normalisasi data. Hasil pelatihan menghasilkan sekumpulan koefisien (β) untuk setiap fitur masukan, yang merepresentasikan bobot kontribusi masing-masing variabel terhadap prediksi risiko depresi. Semakin besar nilai absolut koefisien, semakin kuat pengaruh fitur tersebut terhadap

keluaran model. Koefisien bernilai positif mengindikasikan bahwa fitur tersebut meningkatkan probabilitas prediksi depresi, sedangkan koefisien bernilai negatif menunjukkan efek sebaliknya.

Rekapitulasi koefisien rata-rata dari lima fold pelatihan beserta standar deviasinya disajikan pada Tabel 4.3 berikut.

Tabel 4.3 Koefisien Hasil Training Model Logistic Regression (Rata-rata 5-Fold)

Fitur	Koefisien Rata-rata	Std. Dev	Arah Pengaruh
ToSC	-0.7164	0.1096	(-) Menurunkan Risiko Depresi
Suicide	+0.5322	0.0410	(+) Meningkatkan Risiko Depresi
Parents	+0.4039	0.0718	(+) Meningkatkan Risiko Depresi
Academic	+0.3150	0.1647	(+) Meningkatkan Risiko Depresi
inter_dom	+0.3072	0.1488	(+) Meningkatkan Risiko Depresi
Partner	+0.3021	0.0585	(+) Meningkatkan Risiko Depresi
AGuilt	-0.2896	0.1681	(-) Menurunkan Risiko Depresi
Relative	-0.2859	0.1106	(-) Menurunkan Risiko Depresi
Friends	-0.2600	0.1219	(-) Menurunkan Risiko Depresi
Others	+0.2417	0.0873	(+) Meningkatkan Risiko Depresi
AMiscell	+0.2301	0.1132	(+) Meningkatkan Risiko Depresi
Age	+0.2095	0.1168	(+) Meningkatkan Risiko Depresi
APD	-0.1446	0.1962	(-) Menurunkan Risiko Depresi
English	-0.1434	0.0653	(-) Menurunkan Risiko Depresi
Stay	+0.1397	0.1387	(+) Meningkatkan Risiko Depresi
Religion	-0.1277	0.0760	(-) Menurunkan Risiko Depresi
APH	+0.1114	0.0430	(+) Meningkatkan Risiko Depresi
Profess	-0.0896	0.1131	(-) Menurunkan Risiko Depresi
Japanese	+0.0824	0.2036	(+) Meningkatkan Risiko Depresi
Gender	-0.0720	0.0612	(-) Menurunkan Risiko Depresi
Afear	+0.0611	0.0324	(+) Meningkatkan Risiko Depresi
Doctor	-0.0478	0.1156	(-) Menurunkan Risiko Depresi
Alone	+0.0198	0.1108	(+) Meningkatkan Risiko Depresi
Reli	-0.0169	0.1455	(-) Menurunkan Risiko Depresi
AHome	+0.0168	0.0665	(+) Meningkatkan Risiko Depresi
ACS	-0.0055	0.0780	(-) Menurunkan Risiko Depresi



Gambar 4.4 Visualisasis Koefisein Hasil Training Model Logistic Regression

Berdasarkan Tabel 4.3 dan Gambar 4.4, terdapat beberapa temuan penting dari hasil pelatihan model. Fitur ToSC (Total Score of Social Connectedness) mencatatkan koefisien terbesar secara absolut sebesar −0.7164, yang mengindikasikan bahwa semakin tinggi tingkat koneksi sosial seorang mahasiswa, semakin rendah probabilitas model memprediksi kondisi depresi. Temuan ini sejalan dengan teori bahwa dukungan sosial merupakan faktor protektif utama terhadap kesehatan mental.

Di sisi sebaliknya, fitur Suicide memiliki koefisien positif tertinggi sebesar +0.5322 dengan standar deviasi yang sangat kecil (0.0410), mengindikasikan bahwa variabel ini secara konsisten dan stabil berkontribusi meningkatkan probabilitas prediksi depresi di seluruh fold pelatihan. Fitur Parents (+0.4039) dan Akademik (+0.3150) juga menunjukkan pengaruh positif yang cukup signifikan, mencerminkan bahwa tekanan akademik dan dinamika hubungan dengan orang tua turut memengaruhi risiko kesehatan mental mahasiswa.

Perlu dicatat bahwa beberapa fitur seperti Japanese, APD, dan Reli memiliki nilai standar deviasi yang relatif besar, menandakan bahwa koefisien fitur-fitur tersebut kurang stabil antar fold dan perlu diinterpretasikan dengan kehati-hatian. Secara keseluruhan, hasil pelatihan ini membuktikan bahwa model Logistic Regression telah berhasil mempelajari pola dari data pelatihan dan menghasilkan representasi bobot fitur yang dapat diinterpretasikan secara ilmiah.

4.4 Hasil Pengujian 5-Fold Cross Validation

Evaluasi performa model Logistic Regression dilakukan menggunakan subset data uji melalui skema 5-Fold Cross Validation guna menjamin objektivitas hasil dan mengukur kemampuan generalisasi model terhadap data baru. Rekapitulasi hasil pengujian untuk metrik Accuracy, Precision, Recall, dan F1-Score disajikan pada Tabel 4.4.

Tabel 4. 4 Rekapitulasi Performa Model Logistic Regression (5-Fold)

Fold	Accuracy	Precision	Recall	F1-Score
Fold 1	0.7037	0.6154	0.4211	0.5000
Fold 2	0.6481	0.5000	0.3684	0.4242
Fold 3	0.6296	0.5000	0.4000	0.4444
Fold 4	0.7736	0.8182	0.4737	0.6000
Fold 5	0.7358	0.6667	0.5263	0.5882
Rata-rata	0.6982	0.6200	0.4379	0.5114

4.4.1 Analisis Metrik Akurasi (Accuracy)

Berdasarkan Tabel 4.2, model Logistic Regression mencatatkan rata-rata akurasi sebesar 69,82%. Performa puncak model ini terlihat pada Fold 4 (77,36%) dan Fold 5 (73,58%). Tingginya akurasi pada fold tersebut mengindikasikan bahwa model linier ini cukup stabil dalam memetakan profil kesehatan mental mahasiswa secara umum. Namun, adanya variansi performa antar fold (dari 62% hingga 77%)

menunjukkan bahwa model linier tunggal masih memiliki ketergantungan yang cukup tinggi pada distribusi data spesifik di setiap subset.

4.4.2 Analisis Metrik Precision dan Integritas Diagnosa

Pada parameter Precision, model menghasilkan rata-rata sebesar 62,00%. Nilai ini merepresentasikan tingkat kepercayaan sistem saat memberikan diagnosa positif depresi. Pencapaian tertinggi pada Fold 4 (81,82%) menunjukkan bahwa dalam distribusi data tertentu, model linier ini memiliki ketepatan yang sangat tinggi dalam meminimalkan kesalahan diagnosa positif palsu (False Positive). Namun, adanya nilai 50% pada Fold 2 dan 3 menunjukkan adanya tantangan bagi model dalam membedakan antara responden yang benar-benar berisiko dengan responden sehat yang memiliki kemiripan pola fitur psikososial.

4.4.3 Analisis Metrik Recall dan Daya Deteksi Risiko

Metrik Recall menunjukkan nilai rata-rata sebesar 43,79%. Secara teknis, angka ini mengungkapkan bahwa model Logistic Regression cenderung memiliki sifat yang "konservatif" atau sangat hati-hati dalam menetapkan label depresi. Rendahnya nilai recall (di bawah 50%) mencerminkan adanya risiko False Negative yang cukup tinggi, di mana sebagian mahasiswa yang sebenarnya memiliki risiko depresi gagal terdeteksi oleh sistem. Hal ini memperkuat teori bahwa model linier tunggal seringkali kesulitan dalam menangkap sinyal-sinyal kesehatan mental yang bersifat non-linier atau memiliki gejala yang samar pada responden.

4.4.4 Analisis Keseimbangan Model (F1-Score)

Sebagai penutup evaluasi, metrik F1-Score mencatatkan rata-rata sebesar 51,14%. Mengingat F1-Score merupakan rata-rata harmonik antara presisi dan

recall, nilai ini memberikan gambaran keseimbangan performa model yang berada pada level moderat. Peningkatan F1-Score yang signifikan terlihat pada Fold 4 (60,00%) dan Fold 5 (58,82%), yang membuktikan bahwa model mencapai titik optimasi terbaiknya pada subset data tersebut. Ketiadaan angka yang sangat jomplang antara presisi dan recall di sebagian besar fold menunjukkan bahwa meskipun daya deteksinya belum maksimal, model Logistic Regression tetap bekerja secara stabil tanpa bias yang ekstrem pada salah satu jenis kelas.

BAB V

PREDIKSI KESEHATAN MENTAL MENGGUNAKAN MODEL TUNGGAL RANDOM FOREST

5.1 Desain Sistem

Implementasi model tunggal kedua dalam penelitian ini menggunakan algoritma Random Forest. Berbeda dengan pendekatan linier pada bab sebelumnya, Random Forest merupakan algoritma ensemble berbasis bagging yang membangun sekumpulan pohon keputusan secara acak untuk meningkatkan stabilitas prediksi. Alur kerja sistem pada bab ini digambarkan secara sistematis pada Gambar 5.1.

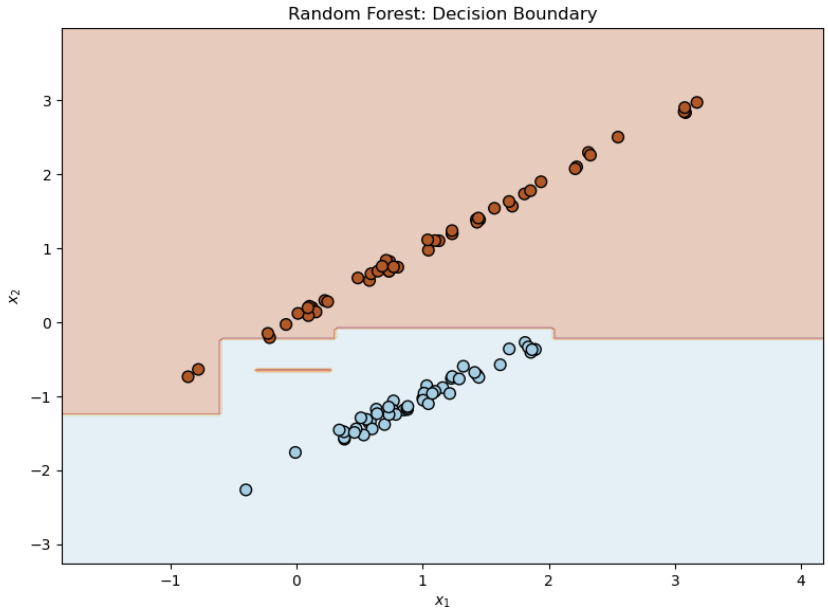
Gambar 5. 1 Desain Sistem Random Forest

Berdasarkan Gambar 5.1, fitur-fitur kuesioner yang telah melalui tahap standarisasi menjadi input utama. Data tersebut kemudian diproses oleh sejumlah pohon keputusan (decision trees) yang bekerja secara paralel. Setiap pohon memberikan suara (vote) untuk menentukan label akhir. Penggunaan arsitektur ini bertujuan untuk menangkap pola non-linier dalam data kesehatan mental mahasiswa yang seringkali memiliki interaksi fitur yang kompleks.

5.2 Model Random Forest

Random *Forest* merupakan algoritma *ensemble learning* berbasis *bagging* yang menggabungkan sejumlah *decision tree* untuk menghasilkan model prediksi yang lebih stabil, akurat, dan tahan terhadap *overfitting* (Pious et al., 2024). Setiap *decision tree* dibangun menggunakan sampel data yang dipilih secara acak melalui teknik *bootstrap sampling*, sedangkan pemilihan fitur pada setiap node dilakukan

secara acak atau disebut *random feature selection*. Kombinasi kedua proses ini menciptakan variasi antar pohon sehingga mengurangi korelasi antarmodel dan meningkatkan kemampuan generalisasi.



Gambar 5. 2 Ilustrasi Random Forest

Gambar di atas menampilkan visualisasi dari algoritma Random Forest dalam konteks klasifikasi data dua dimensi. Setiap titik pada grafik mewakili satu sampel data, di mana warna biru muda menunjukkan Kelas 0 dan warna coklat menunjukkan Kelas 1. Latar belakang grafik memperlihatkan area keputusan (decision boundary) yang terbentuk dari hasil klasifikasi model, dengan wilayah berwarna biru muda diklasifikasikan sebagai Kelas 0 dan wilayah berwarna coklat sebagai Kelas 1. Tidak seperti model linear seperti SVM, batas keputusan Random Forest tampak bertingkat atau tidak halus, karena model ini terdiri dari banyak pohon keputusan (decision trees) yang membagi ruang fitur secara vertikal atau horizontal. Random Forest bekerja dengan cara membentuk beberapa decision tree

dari subset data dan subset fitur yang dipilih secara acak. Setiap pohon dilatih untuk membuat prediksi, kemudian hasil dari seluruh pohon digabungkan menggunakan mekanisme voting mayoritas untuk menghasilkan klasifikasi akhir. Proses ini dikenal sebagai teknik ensemble dengan pendekatan bagging (bootstrap aggregating), yang menjadikan model lebih stabil dan tahan terhadap overfitting. Visualisasi ini menunjukkan bagaimana Random Forest mampu menangani pola data yang kompleks dan tidak linier, serta menggambarkan keunggulan model dalam menciptakan batas keputusan yang fleksibel melalui agregasi banyak model sederhana. Meskipun demikian, karena terdiri dari banyak pohon, interpretasi keputusan akhir Random Forest lebih sulit dibandingkan decision tree tunggal.

Dalam setiap pohon, proses pemilihan atribut terbaik dapat menggunakan ukuran ketidakmurnian (*impurity measures*) seperti *Gini Impurity* atau *Entropy*. Kedua ukuran ini menilai seberapa baik suatu fitur mampu memisahkan kelas target.

Gini Impurity, mengukur tingkat ketidakmurnian suatu node berdasarkan probabilitas kesalahan klasifikasi apabila data dalam node tersebut dipilih secara acak. Nilai *Gini* rendah menunjukkan distribusi kelas yang lebih homogen. Secara matematis, dihitung menggunakan persamaan (3.8):

$$Gini(Y) = 1 - \sum_{c \in C} P\,(c|Y)^2 \tag{3.8}$$

Keterangan:
Y = himpunan data pada satu node
C = himpunan kelas
$P(c|Y)$ = probabilitas kemunculan kelas *c* dalam node
$Gini(Y)$ = nilai ketidakmurnian Gini

Sementara itu, *Entropy* mengukur tingkat ketidakpastian distribusi kelas dalam node. Nilai *Entropy* tinggi menunjukkan bahwa kelas lebih beragam dan pemisahan masih kurang baik. *Entropy* dihitung menggunakan persamaan (3.9):

$$Entropy(Y) = -\sum_{c \in C} P(c|Y) \log_2 P(c|Y) \quad (3.9)$$

Keterangan:
Y = himpunan data pada satu node
C = himpunan kelas
$P(c|Y)$ = probabilitas kemunculan kelas *c* dalam node
$\log_2$ = logaritma basis 2
$Entropy(Y)$ = tingkat ketidakpastian node

Selanjutnya, kualitas suatu pemisahan (split) dievaluasi menggunakan *Information Gain*, yaitu ukuran yang menunjukkan penurunan ketidakmurnian setelah data dibagi berdasarkan suatu fitur. Semakin besar nilai *Information Gain*, semakin baik fitur tersebut dalam memisahkan kelas. *Information Gain* dihitung menggunakan persamaan (3.10):

$$IG(Y, a) = Entropy(Y) - \sum_{v \in Values(a)} \frac{|Y_v|}{|Y|} Entropy(Y_v) \quad (3.10)$$

Keterangan:
Y = himpunan data sebelum split
a = atribut/fitur yang digunakan untuk split
v = nilai atau kategori dari fitur *a*
$Values(a)$ = himpunan seluruh nilai fitur *a*
Y_v = subset data ketika atribut *a* bernilai *v*
$|Y|$ = jumlah sampel dalam node
$|Y_v|$ = jumlah sampel subset
IG(*Y*,*a*) = peningkatan informasi akibat split

Pada tahap prediksi, setiap pohon menghasilkan output klasifikasi berdasarkan struktur yang terbentuk, kemudian hasil akhir diperoleh melalui

mekanisme *majority voting*. Secara matematis, proses agregasi ini dinyatakan dengan persamaan (3.11):

$$\hat{y} = mode\{h_1(x), h_2(x), \dots, h_T(x)\} \quad (3.11)$$

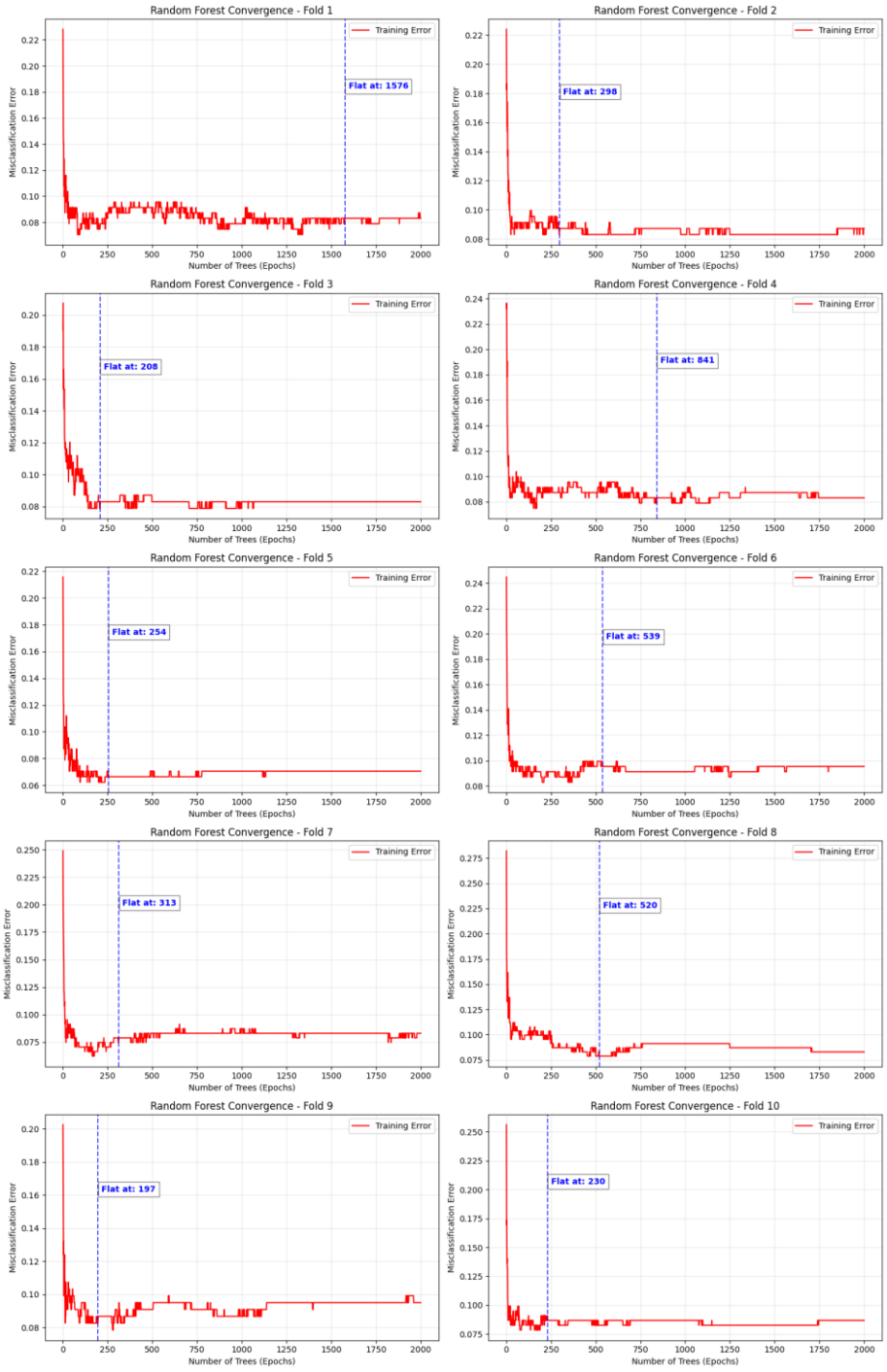
Keterangan:
$h_T(x)$ = prediksi dari pohon ke-*t*
T = jumlah total pohon
$\hat{y}$ = hasil prediksi agregat

Keunggulan utama *Random Forest* adalah kemampuannya dalam menangani dataset berukuran besar, memiliki banyak fitur, serta mengandung pola yang tidak linear. *Random Forest* juga mampu menilai tingkat kepentingan fitur (*feature importance*), sehingga dapat memberikan gambaran mengenai variabel mana yang paling berpengaruh terhadap hasil prediksi.

5.3 Analisis Konvergensi Model dan Hyperparameter Tuning

Proses penentuan nilai hyperparameter optimal pada model Random Forest dalam penelitian ini dilakukan melalui pendekatan analisis kurva konvergensi berbasis Misclassification Error. Berbeda dengan model berbasis iterasi seperti Logistic Regression yang menggunakan Log Loss per epoch, Random Forest membangun model secara bertahap dengan menambahkan pohon keputusan satu per satu. Oleh karena itu, metrik yang digunakan untuk memantau konvergensi adalah Misclassification Error per jumlah pohon, yaitu proporsi data yang salah diklasifikasikan seiring bertambahnya jumlah pohon dalam ensemble.

Peneliti memantau proses ini hingga batas 2.000 pohon menggunakan skema 10-Fold Cross Validation guna memastikan model mencapai titik jenuh informasi. Hasil visualisasi konvergensi disajikan pada Gambar 5.3.



Gambar 5. 3 Analisi Konvergensi Random Forest (10 Fold Cross Validation)

Berdasarkan Gambar 5.3, seluruh iterasi menunjukkan penurunan error yang drastis pada 100 pohon pertama, menandakan efisiensi RF dalam menangkap pola dominan data kesehatan mental mahasiswa secara cepat.

5.3.1 Hyperparameter yang Di-tuning

Hyperparameter yang di-tuning beserta rentang nilai yang diobservasi dan nilai optimal yang dipilih untuk model Random Forest dirangkum pada Tabel 5.1.

Tabel 5. 1 Hyperparameter Tuning Model Random Forest

Hyperparameter	Rentang Nilai Diuji	Nilai Optimal	Keterangan
n_estimators	1 – 2000 pohon (diamati per penambahan pohon)	197	Jumlah pohon keputusan dalam ensemble Random Forest. Ditentukan berdasarkan titik konvergen kurva Misclassification Error
Misclassification Error threshold	Diamati secara visual dan komputasional per fold	< 0.08	Ambang batas error yang dianggap stabil. Digunakan *find_plateau_point*() dengan *patience*=50 untuk mendeteksi titik konvergen
random_state	42 (tetap)	42	Digunakan untuk memastikan hasil eksperimen dapat direproduksi (*reproducible*)

Hyperparameter utama yang menjadi fokus tuning adalah n_estimators, yaitu jumlah pohon keputusan yang dibangun dalam ensemble Random Forest. Nilai n_estimators yang terlalu kecil mengakibatkan model tidak stabil dan rentan terhadap variansi data, sedangkan nilai yang terlalu besar hanya menambah beban komputasi tanpa peningkatan performa yang signifikan. Oleh karena itu, penentuan n_estimators yang tepat melalui observasi kurva Misclassification Error merupakan langkah tuning yang kritis.

5.3.2 Prosedur Hyperparameter Tuning

Proses *tuning* dilakukan dengan mengobservasi kurva Misclassification Error seiring bertambahnya jumlah pohon dari 1 hingga 2000 pohon pada setiap *fold* dalam skema 10-Fold Cross Validation. Fungsi *find_plateau_point()* digunakan untuk mendeteksi titik di mana perubahan Misclassification Error sudah

sangat kecil secara konsisten, dengan parameter *patience=50* yang berarti error harus stabil selama 50 penambahan pohon berturut-turut sebelum titik konvergen dinyatakan tercapai.

Pendekatan ini dipilih karena Random Forest secara alami memiliki sifat *ensemble* yang semakin stabil seiring bertambahnya pohon, namun akan mencapai titik jenuh di mana penambahan pohon tidak lagi memberikan penurunan error yang berarti. Titik jenuh inilah yang dijadikan acuan nilai *n_estimators* optimal, sehingga model dapat bekerja secara efisien tanpa komputasi yang berlebihan.

5.3.3 Hasil Hyperparameter Tuning

Titik konvergen yang terdeteksi pada setiap fold beserta nilai Misclassification Error akhir dirangkum pada pada Tabel 5.2.

Tabel 5. 2 Hasil Deteksi Titik Konvergen per Fold – Random Forest

Fold	Titik Konvergen (Jumlah Pohon)	Misclassification Error Akhir	Keterangan
Fold 1	1576	0.08	Konvergensi paling lambat di antara seluruh fold
Fold 2	298	0.08	*plateau* terdeteksi di sekitar pohon ke-298
Fold 3	208	0.08	*misclassification error* stabil pada pohon ke-208
Fold 4	841	0.08	Konvergensi relatif lambat pada fold ini
Fold 5	254	0.07	*misclassification error* terendah (0.07) di antara seluruh fold
Fold 6	539	0.09	*misclassification error* tertinggi (0.09), konvergensi memerlukan lebih banyak pohon
Fold 7	313	0.08	*plateau* terdeteksi di sekitar pohon ke-313
Fold 8	520	0.08	Error stabil sejak pohon ke-520
Fold 9	197	0.08	Konvergensi tercepat dijadikan acuan nilai *n_estimators* optimal

Fold	Titik Konvergen (Jumlah Pohon)	Misclassification Error Akhir	Keterangan
Fold 10	230	0.08	*plateau* terdeteksi mendekati pohon ke-230
Rata-rata	497.6	0.08	Rata-rata titik konvergen dari seluruh fold
n_estimators dipilih	197	0.08	Diambil dari fold tercepat (Fold 9: 197 pohon) sebagai nilai *n_estimators* optimal

Berdasarkan Tabel 5.2, titik konvergen antar *fold* bervariasi cukup lebar, dari 197 pohon (Fold 9) sebagai yang tercepat hingga 1576 pohon (Fold 1) sebagai yang paling lambat, dengan rata-rata 497.6 pohon. Variasi ini wajar terjadi karena setiap fold memiliki distribusi data training yang berbeda. Nilai *n_estimators*=197 ditetapkan sebagai nilai optimal berdasarkan *fold* tercepat (Fold 9), dengan pertimbangan bahwa nilai ini sudah cukup untuk menjamin konvergensi pada *fold* tersebut. Adapun Misclassification Error akhir yang dicapai pada seluruh *fold* berkisar stabil di angka 0.08, kecuali Fold 5 yang mencapai 0.07 (lebih baik) dan Fold 6 yang mencapai 0.09 (lebih tinggi), menunjukkan bahwa model Random Forest konsisten dalam menghasilkan tingkat kesalahan yang rendah pada data kesehatan mental mahasiswa.

5.4 Hasil Pengujian 5-Fold Cross Validation

Setelah proses pelatihan melalui analisis konvergensi dinyatakan optimal, peneliti melakukan pengujian performa model Random Forest menggunakan data uji melalui skema 5-Fold Cross Validation. Tahap ini bertujuan untuk mengukur kemampuan model dalam melakukan generalisasi terhadap data yang belum pernah dipelajari sebelumnya (*unseen* data). Penilaian dilakukan secara komprehensif

menggunakan empat metrik utama: Accuracy, Precision, Recall, dan F1-Score. Rekapitulasi hasil pengujian untuk setiap fold disajikan pada Tabel 5.3.

Tabel 5. 3 Rekapitulasi Performa Model Random Forest (5-Fold)

Fold	Accuracy	Precision	Recall	F1-Score
Fold 1	0.6111	0.4167	0.2632	0.3226
Fold 2	0.7222	0.6667	0.4211	0.5161
Fold 3	0.7037	0.6250	0.5000	0.5556
Fold 4	0.7547	0.7143	0.5263	0.6061
Fold 5	0.7358	0.7273	0.4211	0.5333
Rata-rata	0.7055	0.6300	0.4263	0.5067

5.4.1 Analisis Metrik Akurasi (Accuracy)

Berdasarkan Tabel 5.2, model Random Forest mencatatkan rata-rata akurasi sebesar 70,55%. Performa akurasi tertinggi dicapai pada Fold 4 (75,47%), sementara performa terendah berada pada Fold 1 (61,11%). Capaian rata-rata di atas 70% ini mengindikasikan bahwa penggunaan arsitektur berbasis pohon keputusan cukup efektif dalam mengklasifikasikan status kesehatan mental mahasiswa. Stabilitas akurasi pada tiga fold terakhir (Fold 3, 4, dan 5) menunjukkan bahwa model telah berhasil menangkap pola fitur kuesioner dengan lebih konsisten setelah melalui proses bootstrapping.

5.4.2 Analisis Metrik Precision dan Integritas Diagnosa

Pada metrik Precision, model menghasilkan nilai rata-rata sebesar 63,00%. Nilai ini menunjukkan tingkat ketepatan model dalam memprediksi kelas depresi mahasiswa. Terlihat adanya tren peningkatan presisi yang signifikan mulai dari Fold 2 hingga puncaknya pada Fold 5 (72,73%). Secara teknis, hal ini membuktikan bahwa mekanisme Random Feature Selection pada model ini cukup mumpuni dalam meminimalisir kesalahan diagnosa positif palsu (False Positive), sehingga

hasil diagnosa risiko depresi yang diberikan memiliki derajat kepercayaan yang memadai.

5.4.3 Analisis Metrik Recall dan Sensitivitas Deteksi

Metrik Recall menunjukkan nilai rata-rata sebesar 42,63%. Sebagaimana pola yang ditemukan pada model sebelumnya, metrik sensitivitas masih menjadi tantangan utama bagi algoritma tunggal. Rendahnya nilai recall (titik terendah 26,32% pada Fold 1) mengindikasikan bahwa model cenderung bersifat "konservatif" dan berisiko melewatkan sebagian mahasiswa yang sebenarnya memiliki risiko depresi. Namun, adanya peningkatan hingga 52,63% pada Fold 4 memberikan sinyal positif bahwa pada distribusi data tertentu, model berbasis ensemble bagging ini mampu mendeteksi gejala depresi dengan jangkauan yang lebih luas.

5.4.4 Analisis Keseimbangan Model (F1-Score)

sRata-rata F1-Score yang dicapai adalah 50,67%. Sebagai metrik yang menyeimbangkan presisi dan recall, angka ini menunjukkan bahwa model Random Forest berada pada level performa keseimbangan moderat. Pencapaian F1-Score tertinggi sebesar 60,61% pada Fold 4 membuktikan bahwa titik optimasi terbaik antara daya deteksi dan ketepatan diagnosa dicapai pada subset data tersebut. Meskipun nilainya belum mencapai ambang batas ideal, stabilitas yang ditunjukkan pada mayoritas fold memberikan justifikasi bahwa Random Forest merupakan model yang lebih robust dibandingkan pendekatan linier dalam menangani kompleksitas data batin mahasiswa.

BAB VI

PREDIKSI KESEHATAN MENTAL MENGGUNAKAN MODEL TUNGGAL SUPPORT VECTOR MACHINE (SVM)

6.1 Desain Sistem

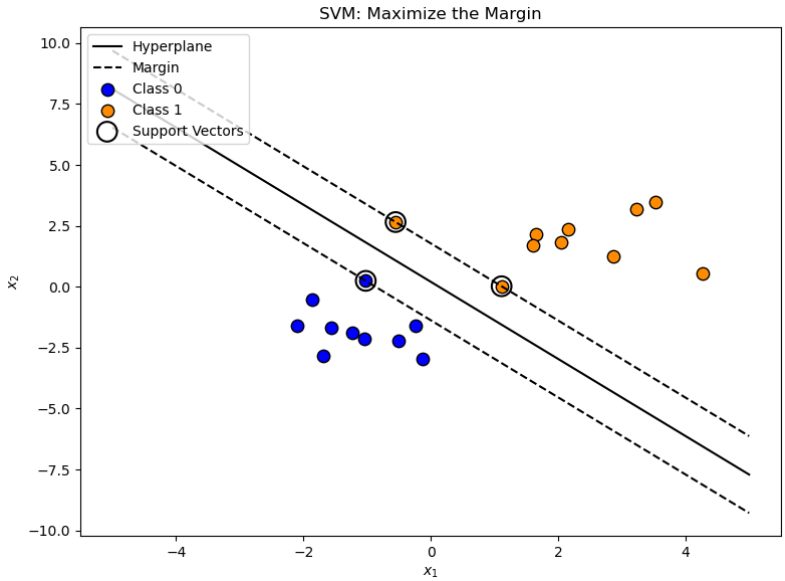
Implementasi model tunggal ketiga dalam penelitian ini menggunakan algoritma Support Vector Machine (SVM). Pemilihan SVM didasarkan pada kekuatannya dalam menangani dataset dengan dimensi fitur yang kompleks melalui mekanisme Kernel Trick. Desain sistem pada bab ini bertujuan untuk mencari hyperplane pemisah paling optimal antara responden berisiko depresi dan responden sehat. Alur kerja sistem secara sistematis digambarkan pada Gambar 6.1.

Gambar 6. 1 Desain Sistem Support Vector Machine

Berdasarkan Gambar 6.1, input data kuesioner yang telah melalui normalisasi Z-Score menjadi prasyarat mutlak bagi SVM karena algoritma ini berbasis pada perhitungan jarak antar titik data. Fitur-fitur tersebut kemudian dipetakan ke dalam ruang dimensi tinggi menggunakan kernel Radial Basis Function (RBF). Model bekerja dengan memaksimalkan margin antara dua kelas untuk meminimalisir kesalahan klasifikasi. Luaran akhir berupa Predicted Mental Health memberikan klasifikasi biner terhadap kondisi mental mahasiswa berdasarkan posisi koordinat data dalam ruang fitur tersebut.

6.2 Model Support Vector Machine (SVM)

Support Vector Machine (SVM) merupakan salah satu metode *supervised learning* yang sangat populer dan efektif dalam menyelesaikan permasalahan klasifikasi dan regresi. SVM bekerja dengan mencari hyperlane optimal yang mampu memisahkan data ke dalam dua kelas dengan margin maksimum. *Hyperplane* adalah batas keputusan yang digunakan untuk memisahkan kelas-kelas dalam ruang fitur. Dengan memaksimalkan *margin*, SVM memastikan bahwa hyperplane memiliki jarak terjauh dari titik data terdekat dari setiap kelas, yang dikenal sebagai *support vectors*. Proses ini membantu dalam meningkatkan generalisasi model dan mengurangi overfitting, sehingga menghasilkan prediksi yang lebih akurat.



Gambar 4. 5 Ilustrasi Hyperplane dan |margin SVM

Gambar di atas menunjukkan konsep dasar dari Support Vector Machine (SVM) dalam menentukan hyperplane optimal yang memisahkan dua kelas data dengan margin maksimum. Titik-titik pada grafik mewakili data sampel yang

diklasifikasikan ke dalam dua kelas berbeda: biru untuk Kelas 0 dan oranye untuk Kelas 1. Garis-garis putus-putus menunjukkan margin di kedua sisi hyperplane (garis tengah). Titik-titik yang dikelilingi menunjukkan support vectors, yaitu titiktitik data yang paling dekat dengan hyperplane dan yang mempengaruhi posisi hyperplane tersebut. Dengan memaksimalkan margin antara support vectors dan hyperplane, SVM memastikan bahwa model dapat mengklasifikasikan data baru dengan lebih baik. Gambar ini menggambarkan bagaimana SVM bekerja untuk memaksimalkan margin dan memisahkan kelas-kelas data dengan cara yang optimal.

Keunggulan SVM adalah kemampuannya dalam menangani data yang tidak teratur dan non-linear dengan menggunakan *kernel trick*. *Kernel trick* memungkinkan SVM untuk memetakan data ke dimensi yang lebih tinggi, di mana data tersebut lebih mudah untuk dipisahkan. Beberapa jenis kernel yang umum digunakan termasuk *linear*, *polynomial*, dan *radial basis function* (RBF). Dengan memilih kernel yang tepat, SVM dapat menangani berbagai macam pola data yang kompleks. Selain itu, SVM juga efektif dalam menangani masalah dengan data berdimensi tinggi, menjadikannya pilihan yang cocok untuk deteksi kesehatan mental di mana banyak fitur yang harus dipertimbangkan. Implementasi SVM dalam sistem deteksi kesehatan mental diharapkan dapat meningkatkan akurasi dan keandalan prediksi, sehingga sistem dapat memberikan respons yang cepat dan tepat dalam situasi darurat.

Prinsip dasar Support Vector Machine dapat dirumuskan pada Persamaan 3.1:

$$w.x + b = 0 \quad (3.1)$$

Keterangan:
w = vektor bobot
x = vektor fitur input
b = bias

Persamaan (3.1) menunjukkan posisi hyperplane yang menjadi batas keputusan antara dua kelas. Untuk menentukan *hyperplane* terbaik, SVM berupaya memaksimalkan *margin* (jarak antara hyperplane dengan titik data terdekat dari masing-masing kelas). Tujuan ini dicapai dengan menyelesaikan fungsi optimasi seperti pada Persamaan (3.2):

$$\min_{\omega,b} \frac{1}{2}\|w\|^2 \quad (3.2)$$

dengan kendala:

$$y_i(w \cdot xi, -tb) \geq 1, i = 1,2, \dots, n \quad (3.3)$$

Keterangan:
$\|w\|$ = norm kuadrat dari vektor bobot
y_i = label kelas dari data ke-i
xi = vektor fitur input dari sampel ke-i
b = bias
n = jumlah total data penelitian

Namun, tidak semua data dapat dipisahkan secara linear. Untuk mengatasi hal ini, SVM menggunakan fungsi kernel (*kernel function*) yang memetakan data ke ruang berdimensi lebih tinggi agar dapat dipisahkan secara linear di ruang tersebut. Bentuk umum fungsi kernel ditunjukkan pada Persamaan (3.4):

$$K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j) \quad (3.4)$$

di mana $\phi(x)$ merupakan fungsi pemetaan ke ruang fitur berdimensi lebih tinggi.

Beberapa jenis kernel yang umum digunakan adalah sebagai berikut:

Linear Kernel, digunakan untuk data yang dapat dipisahkan secara linear, didefinisikan dengan Persamaan (3.5):

$$K(x_i, x_j) = x_i \cdot x_j \tag{3.5}$$

Polynomial Kernel, digunakan untuk data dengan hubungan non-linear derajat tertentu, sebagaimana ditunjukkan pada Persamaan (3.6):

$$K(x_i, x_j) = (\gamma x_i \cdot x_j + r)^d \tag{3.6}$$

Keterangan:
γ = parameter skala kernel
r = konstanta offset
d = derajat polinomial

Radial Basis Function (RBF) *Kernel*, digunakan untuk data dengan pola non-linear kompleks, ditunjukkan pada Persamaan (3.7):

$$K(x_i, x_j) = exp\left(-y\|x_i - x_j\|^2\right) \tag{3.7}$$

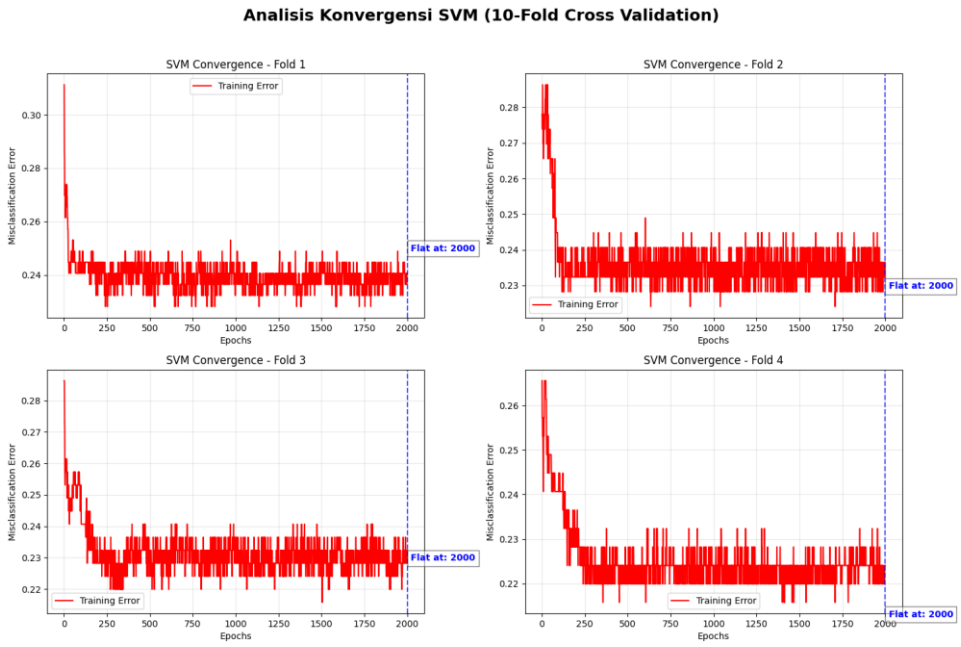
Keterangan:
γ = parameter kernel RBF yang mengontrol bentuk dan lebar fungsi Gaussian
$\|x_i - x_j\|^2$ = konstanta offset

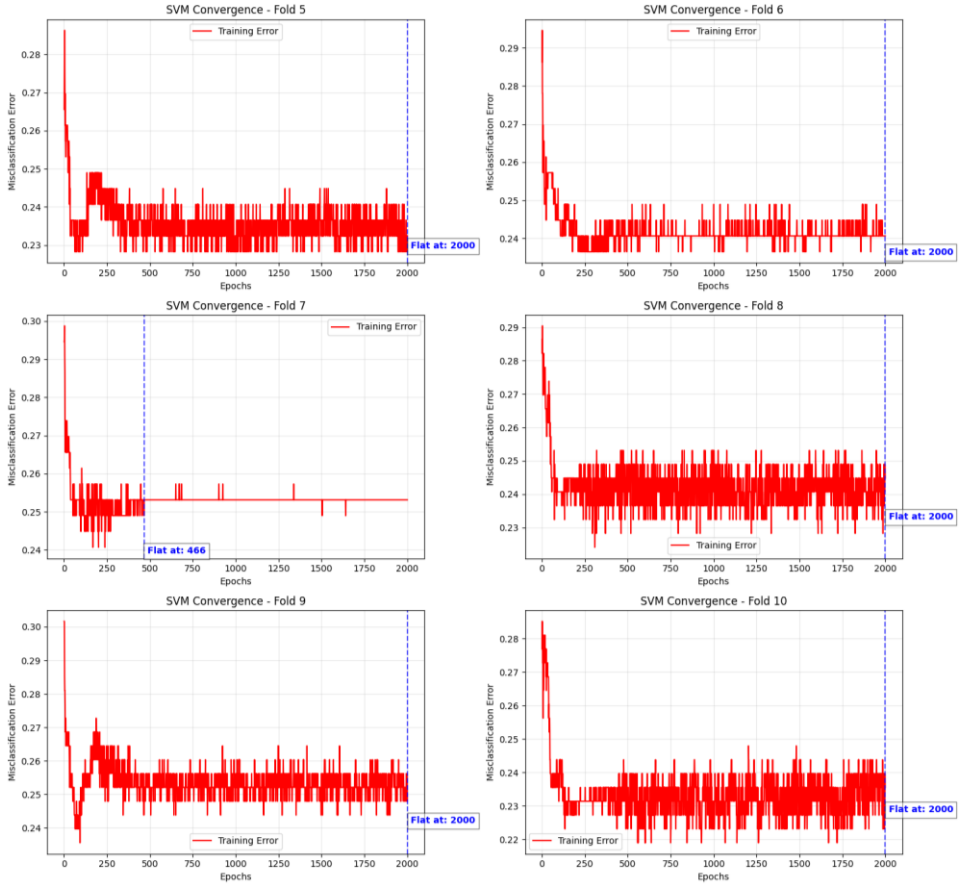
Dengan penggunaan kernel yang sesuai, SVM dapat menangani data non-linear dengan performa yang tinggi. Dalam konteks penelitian ini, SVM diterapkan untuk memprediksi kondisi kesehatan mental mahasiswa berdasarkan data survei, di mana fitur-fitur psikologis dan sosial sering kali memiliki hubungan non-linear. Penggunaan kernel yang tepat diharapkan dapat meningkatkan akurasi dan generalisasi model prediksi.

6.3 Analisis Konvergensi Model dan Hyperparameter Tuning

Proses penentuan nilai hyperparameter optimal pada model Support Vector Machine (SVM) dalam penelitian ini dilakukan melalui pendekatan analisis kurva konvergensi berbasis Misclassification Error. Berbeda dengan model Logistic Regression yang umumnya konvergen dalam ratusan epoch, SVM memiliki karakteristik konvergensi yang lebih kompleks dan memerlukan iterasi yang jauh lebih banyak, khususnya pada data psikososial dengan distribusi kelas yang tidak seimbang seperti dataset yang digunakan dalam penelitian ini.

Peneliti melakukan simulasi sebanyak 2.000 epoch melalui skema 10-Fold Cross Validation untuk mengevaluasi stabilitas model dalam menemukan parameter Support Vectors yang optimal. Hasil visualisasi konvergensi disajikan pada Gambar 6.2.





Gambar 6. 2 Analisis Konvergensi SVM (10-Fold Cross Validation)

Berdasarkan Gambar 6.2, grafik menunjukkan pola yang sangat unik dibandingkan model sebelumnya. Garis merah (Training Error) memperlihatkan osilasi atau gejolak yang cukup tajam di awal iterasi. Secara teoretis informatika, fenomena ini mencerminkan karakteristik optimasi stokastik yang berupaya mencari batas margin di area data yang sangat rapat (overlapping).

6.3.1 Hyperparameter yang Di-tuning

Hyperparameter yang di-tuning beserta rentang nilai yang diobservasi dan nilai optimal yang dipilih untuk model SVM dirangkum pada Tabel 6.1.

Tabel 6. 1 Rekapitulasi Titik Konvergensi SVM per Fold

Hyperparameter	Rentang Nilai Diuji	Nilai Optimal	Keterangan
max_iter	1 – 2000 epoch (diamati per epoch)	466	Jumlah iterasi maksimum pelatihan SVM. Ditentukan berdasarkan satu-satunya fold yang mencapai konvergensi penuh (Fold 7)
Misclassification Error threshold	Diamati secara komputasional per fold	< 1e-4	Ambang batas perubahan error yang dianggap stabil. Digunakan find_plateau_point() dengan patience=50
probability	True (tetap)	True	Mengaktifkan estimasi probabilitas pada SVM agar dapat menghasilkan output probabilitas prediksi
random_state	42 (tetap)	42	Digunakan untuk memastikan hasil eksperimen dapat direproduksi (reproducible)

Hyperparameter utama yang menjadi fokus tuning adalah max_iter, yaitu jumlah iterasi maksimum yang menentukan kapan proses pelatihan SVM dihentikan. Nilai ini sangat kritis pada SVM karena model ini bekerja dengan mencari hyperplane pemisah optimal melalui proses optimasi yang iteratif, sehingga data dengan kompleksitas fitur tinggi seperti fitur psikososial cenderung memerlukan lebih banyak iterasi untuk mencapai konvergensi.

6.3.2 Prosedur Hyperparameter Tuning

Proses tuning dilakukan dengan mengobservasi kurva Misclassification Error seiring bertambahnya jumlah epoch dari 1 hingga 2000 pada setiap fold dalam skema 10-Fold Cross Validation. Fungsi find_plateau_point() digunakan untuk mendeteksi titik di mana perubahan Misclassification Error sudah sangat kecil secara konsisten, dengan parameter *threshold*=1e-4 dan *patience*=50.

Berbeda dengan model Logistic Regression dan Random Forest yang mencapai konvergensi pada sebagian besar fold, hasil observasi menunjukkan

bahwa SVM hanya mampu mencapai konvergensi penuh pada satu fold saja (Fold 7) dalam batas maksimum 2000 epoch. Kondisi ini mengindikasikan bahwa SVM memerlukan batas iterasi yang jauh lebih tinggi untuk data psikososial multikultural yang digunakan dalam penelitian ini.

6.3.3 Hasil Hyperparameter Tuning

Titik konvergen yang terdeteksi pada setiap fold beserta nilai Misclassification Error akhir yang diperoleh disajikan dalam Tabel 6.2 berikut. Fold yang belum mencapai konvergensi hingga batas maksimum epoch ditandai dengan simbol (*).

Tabel 6. 2 Hasil Deteksi Titik Konvergen per Fold Model SVM

Fold	Titik Konvergen (Epoch)	Misclassification Error Akhir	Keterangan
Fold 1	2000*	0.24	Belum konvergen — mencapai batas maksimum epoch tanpa memenuhi kriteria plateau
Fold 2	2000*	0.23	Belum konvergen — mencapai batas maksimum epoch tanpa memenuhi kriteria plateau
Fold 3	2000*	0.23	Belum konvergen — mencapai batas maksimum epoch tanpa memenuhi kriteria plateau
Fold 4	2000*	0.22	Belum konvergen — Misclassification Error terendah (0.22) namun tidak mencapai plateau
Fold 5	2000*	0.23	Belum konvergen — mencapai batas maksimum epoch tanpa memenuhi kriteria plateau
Fold 6	2000*	0.24	Belum konvergen — mencapai batas maksimum epoch tanpa memenuhi kriteria plateau
Fold 7	466	0.25	Satu-satunya fold yang mencapai konvergensi penuh — dijadikan acuan nilai max_iter optimal

Fold	Titik Konvergen (Epoch)	Misclassification Error Akhir	Keterangan
Fold 8	2000*	0.24	Belum konvergen — mencapai batas maksimum epoch tanpa memenuhi kriteria plateau
Fold 9	2000*	0.24	Belum konvergen — mencapai batas maksimum epoch tanpa memenuhi kriteria plateau
Fold 10	2000*	0.23	Belum konvergen — mencapai batas maksimum epoch tanpa memenuhi kriteria plateau
Rata-rata	1846.6	0.23	Rata-rata titik konvergen dan Misclassification Error akhir dari seluruh fold
max_iter dipilih	466	0.25	Diambil dari satu-satunya fold yang konvergen penuh (Fold 7: epoch 466) sebagai nilai max_iter optimal

Berdasarkan Tabel 6.2, sebanyak 9 dari 10 fold (90%) tidak mencapai konvergensi dalam batas 2000 epoch yang ditetapkan, dengan rata-rata titik konvergen sebesar 1846.6 epoch. Hanya Fold 7 (epoch 466) yang berhasil mencapai konvergensi penuh, sehingga nilai tersebut ditetapkan sebagai max_iter=466. Kondisi ini menunjukkan bahwa SVM memiliki kompleksitas konvergensi yang lebih tinggi dibandingkan model lain dalam penelitian ini, yang kemungkinan disebabkan oleh karakteristik data psikososial yang memiliki banyak fitur dengan distribusi yang tumpang tindih antar kelas. Meskipun demikian, nilai Misclassification Error akhir yang berkisar antara 0.22 hingga 0.25 pada seluruh fold masih dapat diterima sebagai hasil pelatihan yang valid.

6.4 Hasil Pengujian 5-Fold Cross Validation

Setelah model SVM dinyatakan mencapai konvergensi fungsional pada tahap pelatihan, peneliti melakukan pengujian performa menggunakan data uji melalui skema 5-Fold Cross Validation. Penggunaan skema ini bertujuan untuk memvalidasi kemampuan generalisasi model terhadap data baru (unseen data) serta memastikan bahwa parameter Support Vectors yang dihasilkan tidak mengalami overfitting pada subset data tertentu. Penilaian dilakukan berdasarkan empat metrik utama, yaitu Accuracy, Precision, Recall, dan F1-Score. Rekapitulasi hasil pengujian performa model SVM disajikan pada Tabel 6.2.

Tabel 6. 3 Rekapitulasi Performa Model SVM (5-Fold)

Fold	Accuracy	Precision	Recall	F1-Score
Fold 1	0.6111	0.4000	0.2105	0.2759
Fold 2	0.6852	0.5833	0.3684	0.4516
Fold 3	0.7222	0.6667	0.5000	0.5714
Fold 4	0.7736	0.7692	0.5263	0.6250
Fold 5	0.7547	0.7500	0.4737	0.5806
Rata-rata	0.7094	0.6338	0.4158	0.5009

6.4.1 Analisis Metrik Akurasi (Accuracy)

Berdasarkan Tabel 6.2, model SVM mencatatkan rata-rata akurasi sebesar 70,94%. Performa model ini menunjukkan tren peningkatan yang signifikan mulai dari Fold 3 hingga mencapai puncaknya pada Fold 4 (77,36%). Pencapaian akurasi di atas 70% ini membuktikan bahwa pemetaan fitur psikososial ke dalam ruang dimensi tinggi menggunakan kernel RBF cukup efektif dalam mendefinisikan batasan antara responden sehat dan berisiko depresi. Stabilitas akurasi pada dua fold terakhir menunjukkan bahwa model telah berhasil menemukan hyperplane yang cukup robust untuk menangani variansi data mahasiswa multikultural.

6.4.2 Analisis Metrik Precision dan Integritas Diagnosa

Pada parameter Precision, model menghasilkan nilai rata-rata sebesar 63,38%. Nilai ini menunjukkan tingkat ketepatan model dalam memberikan diagnosa positif depresi. Terjadi lonjakan presisi yang tajam pada Fold 4 (76,92%) dan Fold 5 (75,00%). Secara teknis, hal ini mengonfirmasi keunggulan arsitektur SVM dalam meminimalisir angka False Positive. Dengan prinsip maximum margin, SVM cenderung bersifat sangat selektif dalam memberikan label "Depresi", sehingga hasil diagnosa yang dikeluarkan memiliki tingkat kepercayaan yang baik dan meminimalisir risiko salah intervensi pada mahasiswa yang sebenarnya sehat.

6.4.3 Analisis Metrik Recall dan

Metrik Recall mencatatkan nilai rata-rata sebesar 41,58%. Rendahnya sensitivitas ini merupakan tantangan terbesar bagi arsitektur SVM dalam riset ini. Pada Fold 1, daya deteksi model anjlok hingga angka 21,05%, yang berarti sebagian besar responden berisiko depresi pada subset tersebut gagal teridentifikasi oleh sistem (False Negative). Peneliti menganalisis bahwa hal ini merupakan dampak dari sifat konservatif SVM; demi menjaga margin pemisah yang bersih, model mengabaikan sampel-sampel depresi yang memiliki fitur samar atau mendekati karakteristik responden sehat. Meskipun pada Fold 4 sensitivitas sempat menyentuh 52,63%, secara keseluruhan SVM masih memerlukan bantuan dari model lain (seperti melalui pendekatan ensemble) untuk meningkatkan jangkauan deteksinya.

6.4.4 Analisis Keseimbangan Model (F1-Score)

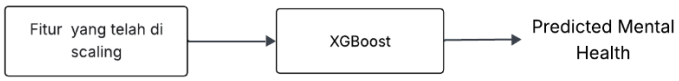
Rata-rata F1-Score yang diperoleh adalah 50,09%. Sebagai metrik yang menyeimbangkan antara presisi dan recall, nilai ini menunjukkan bahwa model SVM berada pada level performa yang moderat. Peningkatan keseimbangan terlihat pada Fold 4 (62,50%), yang menandakan bahwa pada sebaran data tersebut, model berhasil mencapai kompromi optimal antara ketepatan diagnosa dan daya deteksi. Ketiadaan fluktuasi yang terlalu ekstrem pada metrik keseimbangan ini (kecuali pada fold awal) memberikan indikasi bahwa SVM tetap merupakan model yang andal secara fundamental, meskipun kinerjanya dibatasi oleh kompleksitas fitur kuesioner yang tumpang tindih.

BAB VII

PREDIKSI KESEHATAN MENTAL MENGGUNAKAN MODEL TUNGGAL XGBOOST

7.1 Desain Sistem

Implementasi model tunggal terakhir dalam penelitian ini menggunakan algoritma Extreme Gradient Boosting atau XGBoost. Pemilihan XGBoost didasarkan pada performa algoritma ini yang dikenal unggul dalam menangani data tabular melalui mekanisme pembelajaran sekuensial yang efisien. Desain sistem pada bab ini dikembangkan untuk mengevaluasi kapasitas model berbasis boosting dalam meminimalkan residu kesalahan prediksi secara bertahap. Alur kerja sistem digambarkan secara sistematis pada Gambar 7.1.

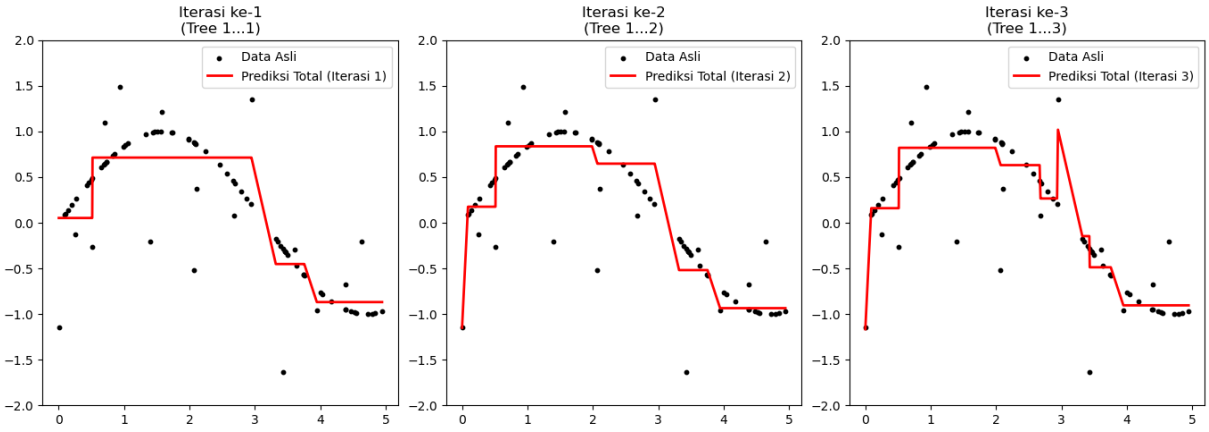


Gambar 7. 1 Desain Sistem Model Extreme Gradient Boosting (XGBoost)

Berdasarkan alur kerja pada Gambar 7.1, data masukan berupa fitur psikososial (ASSIS, SCS, PHQ-9, dan sosiodemografi) yang telah melewati tahap scaling atau normalisasi Z-Score diumpankan ke dalam arsitektur XGBoost. Berbeda dengan Random Forest yang membangun pohon secara paralel, XGBoost membangun pohon keputusan secara berurutan, di mana setiap pohon baru berupaya memperbaiki kesalahan (residual error) yang dihasilkan oleh pohon sebelumnya. Penggunaan regularisasi internal pada model ini berfungsi sebagai pengontrol kompleksitas untuk mencegah terjadinya overfitting. Luaran akhir dari proses ini adalah label biner Predicted Mental Health yang merepresentasikan klasifikasi risiko kesehatan mental responden.

7.2 Model XGBoost

dd *Extreme Gradient Boosting* (XGBoost) merupakan algoritma *ensemble* yang dikembangkan berdasarkan pendekatan *gradient boosting decision trees* dan dirancang untuk menghasilkan performa prediksi yang tinggi melalui optimasi komputasi serta regularisasi yang kuat. Algoritma ini pertama kali diperkenalkan oleh Chen dan Guestrin pada tahun 2016 dan hingga kini menjadi salah satu metode populer dalam penelitian yang menggunakan data tabular, termasuk penelitian prediksi kesehatan mental. Mekanisme pembelajaran pada *XGBoost* dilakukan secara bertahap, di mana setiap pohon keputusan dibangun untuk memperbaiki kesalahan prediksi dari model sebelumnya, sehingga model yang dihasilkan merupakan kombinasi dari sejumlah *weak learners*.



Gambar 7. 2 Ilustrasi XGBoost

Algoritma XGBoost, di mana model berusaha memprediksi pola data secara bertahap agar semakin akurat. Pada tahap awal atau Iterasi ke-1, model hanya menggunakan satu pohon keputusan (decision tree) sederhana, sehingga garis prediksi berwarna merah terlihat masih kaku dan hanya mampu menangkap tren data secara kasar, mengakibatkan banyak titik data asli (titik hitam) yang belum

terjangkau dengan tepat. Bentuk garis yang menyerupai anak tangga ini mencerminkan karakteristik dasar dari decision tree yang memecah data ke dalam aturan-aturan blok yang tegas.

Seiring berlanjutnya proses ke Iterasi ke-2 dan Iterasi ke-3, algoritma tidak membuang model sebelumnya, melainkan menambahkan pohon-pohon baru yang secara khusus dilatih untuk memperbaiki kesalahan (residual) dari iterasi yang lalu. Setiap pohon baru memberikan koreksi detail pada prediksi sebelumnya, sehingga garis merah terlihat semakin fleksibel dan rapat mengikuti gelombang pola data asli. Ini menggambarkan prinsip utama boosting, yaitu menggabungkan banyak model sederhana (weak learners) secara berurutan untuk membentuk satu model akhir yang sangat kuat dan presisi (strong learner).

Secara matematis, prediksi akhir *XGBoost* dapat dinyatakan sebagai penjumlahan seluruh pohon yang terbentuk pada setiap iterasi. Hal ini dapat dilihat pada Persamaan 3.12:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), \tag{3.12}$$

Keterangan:
$\hat{y}_i$ = prediksi dari akhir untuk sampel ke-*i*
K = Total jumlah *tree* (*boosted trees*) yang digunakan
f_k = fungsi dari *tree* ke-*k*
x_i = vektor fitur dari sampel ke-*i*

Dengan f_k mewakili pohon keputusan pada iterasi ke-*k*. Pendekatan bertahap seperti ini sangat sesuai untuk data kesehatan mental yang kompleks dan memiliki pola non-linear, karena setiap pohon dapat mempelajari bagian kesalahan yang belum dapat dijelaskan oleh pohon sebelumnya.

Dalam proses pembelajarannya, *XGBoost* meminimalkan suatu fungsi objektif yang terdiri atas dua komponen utama, yaitu fungsi *loss* dan komponen *regularisasi*. Fungsi *loss* berfungsi untuk mengukur tingkat kesalahan antara nilai aktual dan prediksi, sedangkan *regularisasi* digunakan untuk mengontrol kompleksitas model agar tidak terjadi *overfitting*. Fungsi objektif tersebut dirumuskan pada Persamaan 3.13:

$$L = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k), \quad (3.13)$$

Keterangan:
$l(y_i, \hat{y}_i)$ = fungsi loss
$\Omega(f_k)$ = regulasi untuk mengontrol kompleksitas pohon
K = Total jumlah *tree* (*boosted trees*) yang digunakan
n = jumlah data
f_k = pohon keputusan ke-k

Regularisasi berperan penting untuk mengontrol kompleksitas dan mencegah *overfitting*, hal dapat dilihat pada Persamaan 3.12:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2,$$

Keterangan:
T = jumlah *leaf* pada pohon
w_j = bobot *leaf* ke-j
γ = penalti jumlah *leaf*
λ = regularisasi L2

Dimana *T* adalah jumlah *leaf* pada pohon, w_j merupakan bobot *leaf* ke-j, sementara γ dan λ mengatur besarnya penalti terhadap kompleksitas struktur pohon.

Regularisasi ini menjadi sangat penting pada penelitian prediksi kesehatan mental karena dataset survei mahasiswa cenderung berukuran kecil hingga menengah dan mengandung *variabel* psikologis yang saling berkorelasi sehingga rentan terhadap overfitting apabila tidak dikendalikan dengan baik.

Untuk mengoptimalkan fungsi objektif tersebut, *XGBoost* menggunakan pendekatan *Taylor Ekspansi* orde dua, yang memungkinkan optimasi menjadi lebih cepat dan stabil. Melalui pendekatan ini, fungsi objektif pada *iterasi* ke-t dapat dihitung menggunakan Persamaan 3.13:

$$L^{(t)} \approx \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t(x_i)^2 \right] + \Omega(f_t), \tag{3.13}$$

dengan g_i merupakan turunan pertama (*gradient*) dan h_i merupakan turunan kedua (*Hessian*) dari fungsi *loss*. Penggunaan informasi turunan pertama dan kedua memberikan keunggulan signifikan dibandingkan metode *boosting* yang hanya memanfaatkan *gradient*, sehingga *XGBoost* mampu mempelajari pola-pola kompleks pada variabel psikologis seperti stres akademik, beban pekerjaan, kualitas tidur, maupun dukungan sosial secara lebih efektif.

Selanjutnya, penentuan struktur pohon dilakukan dengan mengevaluasi kualitas setiap calon pemisahan (*split*) menggunakan perhitungan *gain*. Nilai *gain* menggambarkan seberapa besar peningkatan kualitas model jika suatu node dipisahkan menjadi dua bagian. Rumus *gain* dihitung menggunakan Persamaan 3.14:

$$Gain = \frac{1}{2} \left[\frac{G_L^2}{H_L + \lambda} + \frac{(G_R)^2}{H_R + \lambda} - \frac{G^2}{H + \lambda} \right] - \gamma \tag{3.14}$$

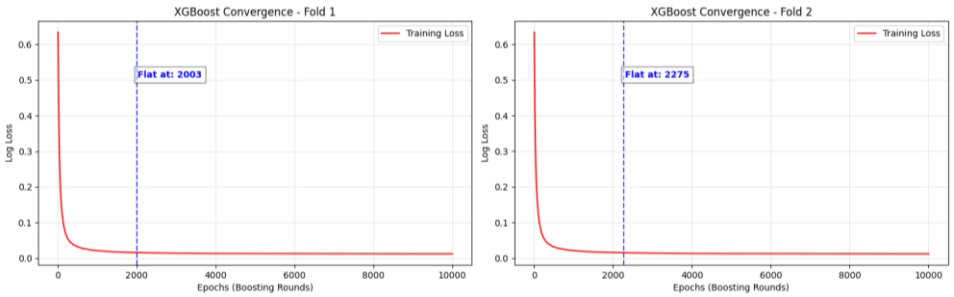
di mana *G* dan *H* adalah jumlah *gradient* dan *Hessian* pada node sebelum dipecah, sedangkan G_L, H_L, dan G_R, H_R masing-masing merupakan nilai pada node

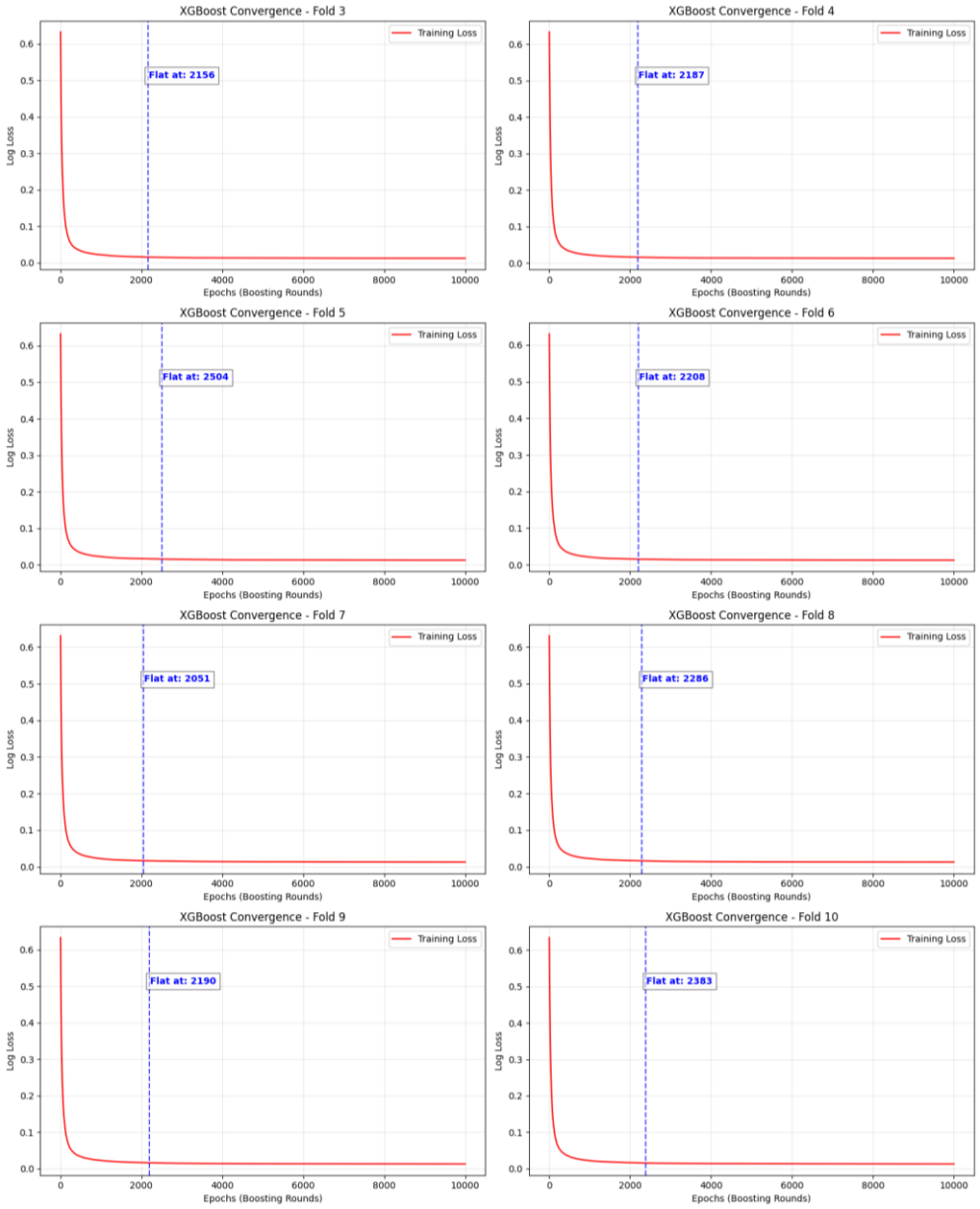
kiri dan kanan. *Split* yang memiliki nilai *gain* terbesar akan dipilih sebagai pemisahan terbaik. Dalam konteks prediksi kesehatan mental, proses ini memungkinkan model mengidentifikasi fitur-fitur yang paling berkontribusi terhadap risiko, misalnya stres akademik yang mungkin memiliki gain lebih tinggi dibandingkan dukungan teman sebaya atau kualitas tidur.

7.3 Analisis Konvergensi Model dan Hyperparameter Tuning

Proses penentuan nilai hyperparameter optimal pada model XGBoost dalam penelitian ini dilakukan melalui pendekatan analisis kurva konvergensi berbasis Log Loss per boosting round. Berbeda dengan model lainnya, XGBoost membangun model secara bertahap melalui mekanisme gradient boosting, di mana setiap boosting round menambahkan satu pohon keputusan baru yang bertujuan memperbaiki kesalahan prediksi dari round sebelumnya. Metrik Log Loss dipilih sebagai indikator konvergensi karena sesuai dengan sifat probabilistik prediksi yang dihasilkan oleh XGBoost pada tugas klasifikasi biner.

Peneliti mengonfigurasi batas maksimal hingga 10.000 iterasi dengan learning rate (eta) sebesar 0.05 untuk mendapatkan visualisasi pergerakan gradien yang halus dan presisi. Hasil pemantauan konvergensi melalui skema 10-Fold Cross Validation disajikan pada Gambar 7.3.





Gambar 7. 3 Analisis Konvergensi XGBoost (10-Fold)

Berdasarkan visualisasi pada Gambar 7.3, seluruh iterasi menunjukkan pola penurunan loss yang sangat konsisten dan bersifat eksponensial negatif. Garis merah (Training Loss) menunjukkan penurunan yang sangat tajam pada rentang 1 hingga 1000 boosting rounds pertama. Secara teoretis informatika, fenomena ini mengonfirmasi kekuatan XGBoost dalam mengekstraksi informasi dominan dari

fitur kuesioner mahasiswa (seperti skor depresi PHQ-9 dan stres akulturasi ASSIS) pada tahap awal pembelajaran.

7.3.1 Hyperparameter yang Di-tuning

Hyperparameter yang di-tuning, meliputi rentang nilai yang diamati serta nilai optimal yang dipilih untuk model XGBoost dirangkum pada Tabel 7.1.

Tabel 7. 1 Hyperparameter Tuning Model XGBoost

Hyperparameter	Rentang Nilai Diuji	Nilai Optimal	Keterangan
n_estimators	1 – 10.000 boosting round (diamati per round)	2003	Jumlah boosting round XGBoost. Ditentukan berdasarkan titik konvergen fold tercepat (Fold 1)
learning_rate	0.05 (tetap)	0.05	Laju belajar yang mengontrol kontribusi setiap pohon baru. Nilai kecil (0.05) dipilih untuk konvergensi yang lebih halus dan stabil
eval_metric	logloss	logloss	Metrik evaluasi yang digunakan selama proses boosting untuk memantau konvergensi model per round
Log Loss threshold	Diamati secara komputasional per fold	< 1e-4	Ambang batas perubahan Log Loss yang dianggap stabil. Digunakan find_plateau_point() dengan patience=50
random_state	42 (tetap)	42	Digunakan untuk memastikan hasil eksperimen dapat direproduksi (reproducible)

Hyperparameter utama yang menjadi fokus tuning adalah *n_estimators*, yaitu jumlah *boosting round* yang menentukan berapa banyak pohon keputusan yang dibangun secara bertahap oleh XGBoost. Parameter *learning_rate*=0.05

dipilih dengan nilai kecil agar setiap pohon memberikan kontribusi yang proporsional, sehingga proses konvergensi lebih halus dan tidak terjadi *overfitting* pada data training. Kombinasi *n_estimators* yang besar dengan *learning_rate* yang kecil merupakan strategi umum dalam XGBoost untuk menghasilkan model yang lebih generalisabel.

7.3.2 Prosedur Hyperparameter Tuning

Proses tuning dilakukan dengan mengobservasi kurva Log Loss seiring bertambahnya jumlah *boosting round* dari 1 hingga 10.000 pada setiap *fold* dalam skema 10-Fold Cross Validation. Fungsi *find_plateau_point*() digunakan untuk mendeteksi titik di mana perubahan Log Loss sudah sangat kecil secara konsisten, dengan parameter *threshold*=1e-4 dan *patience*=50.

Berbeda dengan SVM yang sebagian besar foldnya tidak konvergen, seluruh 10 fold XGBoost berhasil mencapai konvergensi penuh dalam rentang boosting round yang diobservasi. Hal ini menunjukkan bahwa mekanisme gradient boosting dengan learning rate kecil pada XGBoost lebih efektif dalam mencapai konvergensi pada data psikososial multikultural yang digunakan dalam penelitian ini dibandingkan dengan SVM.

7.3.3 Hasil Hyperparameter Tuning

Hyperparameter yang di-tuning, meliputi rentang nilai yang diobservasi dan nilai optimal yang dipilih untuk model XGBoost, dirangkum dalam Tabel 7.2.

Tabel 7. 2 Rekapitulasi Titik Konvergensi XGBoost per Fold

Iterasi (Fold)	Titik Konvergen (Boosting Round)	Nilai Log Loss Akhir	Keterangan
Fold 1	2003	0.013	Konvergensi tercepat — dijadikan acuan nilai n_estimators optimal
Fold 2	2275	0.011	Log Loss terbaik (0.011) — stabil, perubahan < 1e-4 dalam 50 boosting round berturut-turut
Fold 3	2156	0.012	Log Loss stabil, perubahan < 1e-4 dalam 50 boosting round berturut-turut
Fold 4	2187	0.012	Log Loss stabil, perubahan < 1e-4 dalam 50 boosting round berturut-turut
Fold 5	2504	0.014	Konvergensi paling lambat di antara seluruh fold
Fold 6	2208	0.013	Log Loss stabil, perubahan < 1e-4 dalam 50 boosting round berturut-turut
Fold 7	2051	0.014	Log Loss stabil, perubahan < 1e-4 dalam 50 boosting round berturut-turut
Fold 8	2286	0.013	Log Loss stabil, perubahan < 1e-4 dalam 50 boosting round berturut-turut
Fold 9	2190	0.015	Log Loss tertinggi (0.015) — stabil, perubahan < 1e-4 dalam 50 boosting round berturut-turut
Fold 10	2383	0.012	Log Loss stabil, perubahan < 1e-4 dalam 50 boosting round berturut-turut
Rata-rata	2224.3	0.013	Rata-rata titik konvergen dan Log Loss akhir dari seluruh fold
n_estimator dipilih	2003	0.013	Diambil dari fold tercepat (Fold 1: boosting round 2003) sebagai nilai n_estimators optimal

Berdasarkan Tabel 7.2, seluruh *fold* berhasil mencapai konvergensi dengan titik konvergen berkisar antara 2003 *boosting round* (Fold 1) sebagai yang tercepat hingga 2504 *boosting round* (Fold 5) sebagai yang paling lambat, dengan rata-rata 2224.3 boosting round. Nilai *n_estimators*=2003 ditetapkan sebagai nilai optimal berdasarkan *fold* tercepat (Fold 1). Log Loss akhir yang dicapai seluruh *fold* sangat rendah, berkisar antara 0.011 hingga 0.015, jauh lebih baik dibandingkan model LR (±0.50) maupun SVM (0.22–0.25). Hal ini mengindikasikan bahwa XGBoost

mampu meminimalkan kesalahan prediksi probabilitas secara lebih efektif pada dataset kesehatan mental mahasiswa yang digunakan dalam penelitian ini.

7.4 Hasil Pengujian 5-Fold Cross Validation

Setelah model XGBoost dinyatakan mencapai titik konvergensi yang stabil pada tahap pelatihan, langkah krusial berikutnya adalah mengevaluasi kemampuan generalisasi model terhadap data yang belum pernah dilihat sebelumnya (unseen data). Peneliti melakukan pengujian menggunakan skema 5-Fold Cross Validation untuk mendapatkan estimasi performa yang objektif dan bebas dari bias pembagian data tunggal. Pengujian ini diukur menggunakan empat parameter utama: Accuracy, Precision, Recall, dan F1-Score. Rekapitulasi hasil pengujian performa XGBoost disajikan secara rinci pada Tabel 7.2.

Tabel 7. 3 Rekapitulasi Performa Model XGBoost (5-Fold)

Fold	Accuracy	Precision	Recall	F1-Score
Fold 1	0.5741	0.3889	0.3684	0.3784
Fold 2	0.7222	0.6667	0.4211	0.5161
Fold 3	0.6852	0.5789	0.5500	0.5641
Fold 4	0.7358	0.6667	0.5263	0.5882
Fold 5	0.7170	0.6111	0.5789	0.5946
Rata-rata	0.6869	0.5825	0.4889	0.5283

7.5 Analisis Metrik Akurasi (Accuracy)

Model XGBoost menghasilkan rata-rata akurasi sebesar 68,69%. Melalui pengamatan pada setiap fold, terlihat variansi yang cukup lebar di mana performa tertinggi dicapai pada Fold 4 (73,58%), sementara penurunan tajam terjadi pada Fold 1 (57,41%). Penurunan signifikan pada Fold 1 ini mengindikasikan bahwa XGBoost sangat sensitif terhadap sebaran data asli. Secara teoretis informatika, hal ini terjadi karena XGBoost membangun pohon secara sekuensial; jika subset data

uji pada fold tertentu memiliki distribusi fitur yang sangat berbeda dengan data latih, model berbasis boosting cenderung mengalami kesulitan generalisasi dibandingkan model berbasis bagging (seperti Random Forest).

7.6 Analisis Metrik Precision dan Risiko False Positive

Pada parameter Precision, model mencatatkan rata-rata sebesar 58,25%. Nilai ini menunjukkan tingkat ketepatan diagnosa positif risiko depresi. Dibandingkan dengan model SVM (63,38%) atau Random Forest (63,00%) pada bab sebelumnya, XGBoost menunjukkan nilai presisi yang lebih rendah. Hal ini membuktikan secara empiris bahwa agresivitas XGBoost dalam meminimalkan residu kesalahan mengakibatkan model ini lebih rentan terhadap kesalahan diagnosa positif palsu (False Positive). Dalam konteks klinis, ini berarti sistem cenderung memberikan "alarm" risiko depresi pada mahasiswa yang sebenarnya sehat, yang dapat memicu inefisiensi pada proses verifikasi lanjut oleh konselor.

7.7 Analisis Metrik Recall dan Sensitivitas Deteksi

Meskipun memiliki presisi yang moderat, XGBoost membuktikan keunggulan utamanya pada metrik Recall dengan rata-rata mencapai 48,89%. Angka ini merupakan yang tertinggi di antara seluruh model tunggal yang diuji (SVM 41,58%, RF 42,63%, dan LR 43,79%). Peningkatan sensitivitas deteksi yang konsisten terlihat pada Fold 3, 4, dan 5 yang semuanya berada di atas 52%. Keunggulan ini memberikan kontribusi akademik yang sangat penting: XGBoost adalah algoritma yang paling "waspada" dalam mendeteksi keberadaan gejala depresi. Karakteristik ini sangat krusial jika institusi universitas mengutamakan

aspek keamanan (safety-first), di mana deteksi terhadap mahasiswa yang benar-benar berisiko tidak boleh terlewatkan oleh sistem.

7.8 Analisis Keseimbangan Model (F1-Score)

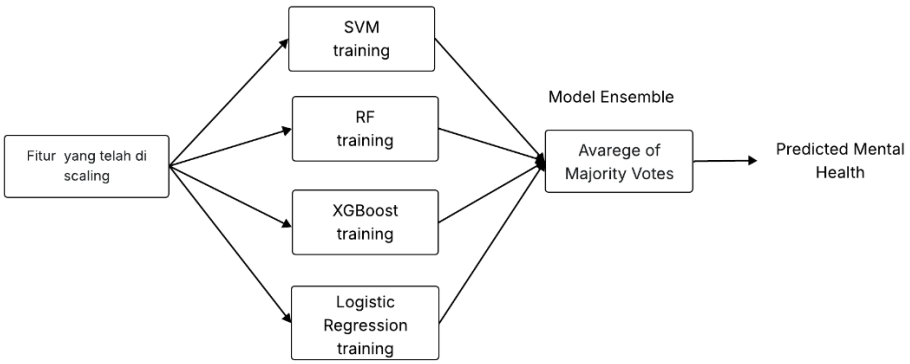
Rata-rata F1-Score yang dicapai adalah 52,83%. Sebagai metrik yang menyeimbangkan antara presisi dan recall, angka ini menunjukkan bahwa XGBoost adalah model tunggal yang paling "jujur" dan harmonis dalam menangani data kesehatan mental mahasiswa yang memiliki tantangan class imbalance. Meskipun akurasi globalnya sedikit di bawah model pohon lainnya, nilai F1-Score yang lebih tinggi membuktikan bahwa XGBoost mampu melakukan kompromi terbaik antara daya deteksi dan ketepatan diagnosa.

BAB VIII

PREDIKSI KESEHATAN MENTAL MENGGUNAKAN MODEL ENSEMBLE MAJORITY VOTED

8.1 Desain Sistem

Implementasi tahap lanjut dalam penelitian ini menggunakan metode Ensemble Majority Voting. Berbeda dengan bab-bab sebelumnya yang menguji algoritma secara mandiri, desain sistem pada bab ini mengintegrasikan empat model dasar (base learners) yang telah dioptimasi, yaitu Logistic Regression, Random Forest, Support Vector Machine, dan XGBoost. Tujuan utama dari arsitektur ensemble ini adalah untuk memanfaatkan keberagaman karakteristik setiap algoritma guna mencapai konsensus prediksi yang lebih stabil. Alur kerja sistem Ensemble Majority Voting digambarkan pada Gambar 8.1.



Gambar 8. 1 Desain Sistem Model Ensemble Majority Voting

Berdasarkan Gambar 8.1, fitur-fitur kuesioner yang telah melalui tahap scaling diumpankan secara paralel ke empat model dasar. Setiap model dasar akan mengeluarkan prediksi label biner (Depresi atau Sehat) secara independen. Keputusan akhir sistem ditentukan melalui mekanisme penggabungan suara mayoritas (Hard Voting). Dengan desain ini, kesalahan prediksi yang bersifat lokal

pada satu algoritma diharapkan dapat dikoreksi oleh algoritma lainnya, sehingga meminimalisir dampak noise data dan meningkatkan reliabilitas luaran sistem Predicted Mental Health.

8.2 Model Ensemble Majority Voted

Dalam pendekatan *ensemble learning*, beberapa model dasar digabungkan untuk menghasilkan prediksi yang lebih stabil dan akurat dibandingkan penggunaan satu model tunggal. Salah satu mekanisme penggabungan yang umum digunakan adalah *voting*, yang bekerja dengan mengintegrasikan keputusan dari setiap model pada saat melakukan klasifikasi. Proses ini dapat dilakukan melalui dua pendekatan utama, yaitu *hard voting* dan *soft voting*, yang kemudian dapat digabungkan menjadi pendekatan *hybrid average of majority votes*.

Hard voting atau *majority voting* merupakan teknik di mana setiap model memberikan satu suara untuk kelas yang diprediksinya. Kelas yang memperoleh suara terbanyak menjadi hasil prediksi akhir. Mekanisme ini dapat dijelaskan melalui persamaan 3.22:

$$\hat{y}(x) = argmax_{c \in C} \sum_{i=1}^{N} \mathrm{I}(h_i(x) = c) \qquad (3.22)$$

Keterangan:

$\hat{y}$	= prediksi akhir *ensemble* untuk *instance x*
$argmax$	= Operator argumen (kelas) yang menghasilkan nilai maksimum
$c \in C$	= kelas kandidat dalam himpunan kelas C = {c_1, $c_{1,}$..., c_K)
N	= jumlah model atau *classifier* yang digunakan dalam *ensemble*
$\sum_{i=1}^{N}$	= penjumlahan suara dari seluruh model
$h_i(x)$	= prediksi kelas yang diberikan oleh model ke-*i* untuk *instance x*
$\mathrm{I}(h_i(x) = c)$	= fungsi indikator yang bernilai 1 jika model *i* memprediksi kelas *c* dan bernilai 0 jika tidak

dimana $h_i(x)$ adalah prediksi model ke-i dan I(.) merupakan fungsi indikator. Pendekatan ini sederhana tetapi efektif, terutama ketika model-model dasar memiliki tingkat kesalahan yang tidak saling berkorelasi.

Selain itu, pendekatan *soft voting* memanfaatkan probabilitas kelas yang dihasilkan setiap model, sehingga keputusan tidak hanya berdasarkan suara terbanyak tetapi juga mempertimbangkan tingkat keyakinan model terhadap setiap kelas dan prediksi akhir diberikan oleh kelas dengan probabilitas rata-rata terbesar. Probabilitas rata-rata untuk suatu kelas dapat dihitung melalui Persamaan 3.23:

$$P_{avg}(c|x) = \frac{1}{N}\sum_{i=1}^{N} p_i(c|x) \quad (3.23)$$

Keterangan:
$P_{avg}(c|x)$ = probabilitas rata-rata bahwa instance *x* termasuk kelas *c*
$p_i(c|x)$ = probabilitas yang diberikan model ke-i bahwa x berada pada kelas c
$\frac{1}{N}$ = aktor perata (mean) untuk menghitung rata-rata dari N probabilitas.
$\sum_{i=1}^{N}$ = penjumlahan probabilitas dari seluruh model

Pendekatan ini cenderung menghasilkan prediksi yang lebih halus karena menimbang proporsi keyakinan masing-masing model.

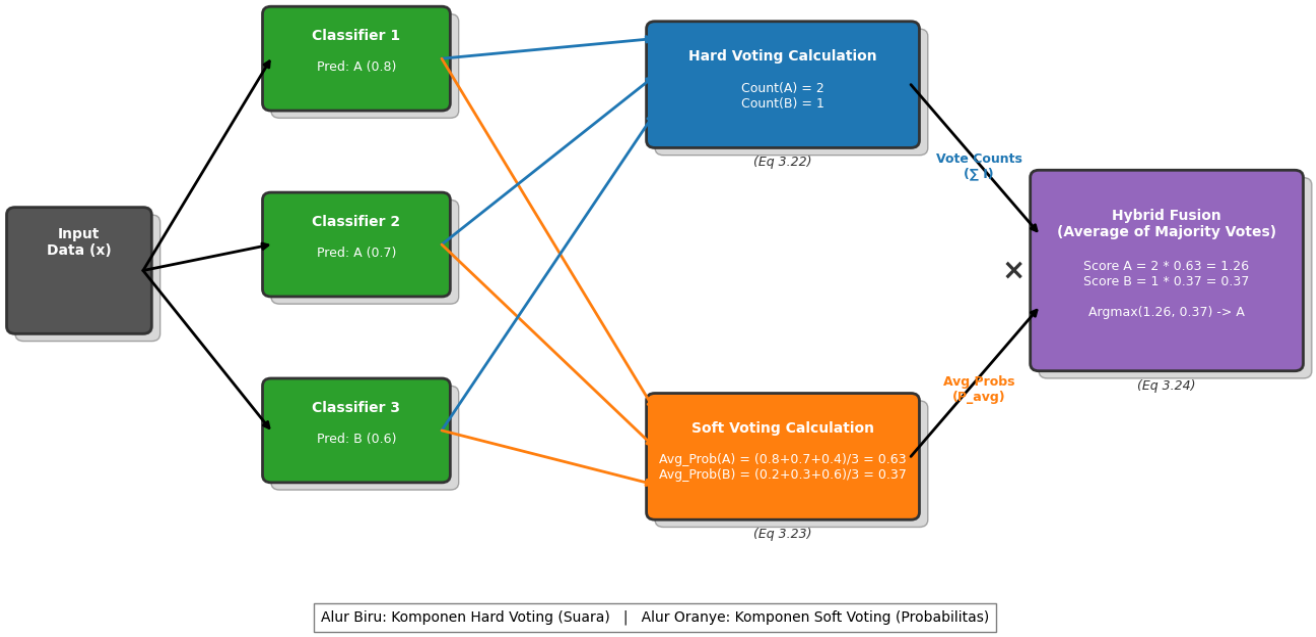
Dalam penelitian ini, kedua mekanisme tersebut digabungkan ke dalam pendekatan *average of majority votes*. Teknik *hybrid* ini mengintegrasikan stabilitas suara mayoritas dengan informasi probabilitas yang dihasilkan model. Secara konseptual, metode ini memberikan keuntungan karena prediksi tidak hanya mengandalkan jumlah suara, tetapi juga mempertimbangkan seberapa besar keyakinan kolektif model terhadap suatu kelas. Kombinasi tersebut dapat dinyatakan melalui persamaan 3.24:

$$\hat{y}(x) = argmax_{c \in C} \left[\left(\sum_{i=1}^{N} \mathrm{I}\left(h_i(x) = c\right) \right) . P_{avg}(c|x) \right] \quad (3.24)$$

Keterangan:

$\sum_{i=1}^{N} p_i(c|x)$	= total suara mayoritas untuk kelas c
$P_{avg}(c|x)$	= rata-rata probabilitas untuk kelas c yang dihitung dari seluruh model
$(.)$	= operator perkalian
$argmax_{c \in C}$	= untuk memilih kelas dengan nilai gabungan terbesar
$\hat{y}(x)$	= prediksi akhir dari ensemble hybrid

Persamaan ini menunjukkan bahwa kelas yang dipilih adalah kelas yang memperoleh dukungan suara terbesar sekaligus memiliki nilai probabilitas rata-rata yang relatif tinggi.



Gambar 8. 2 Ilustrasi Metode Average of Majority Votes

Berdasarkan ilustrasi pada gambar diatas yaitu alur kerja metode *Hybrid* yaitu *Average of Majority Votes* diawali dengan pemrosesan data input (*x*) oleh tiga classifier secara paralel. Setiap model tidak hanya memberikan prediksi kelas, tetapi juga vektor probabilitas yang menunjukkan tingkat keyakinannya. Sebagai

contoh kasus, terjadi perbedaan pendapat antar model yaitu Classifier 1 dan 2 kompak memprediksi Kelas A (probabilitas 0.8 dan 0.7), sedangkan Classifier 3 justru memprediksi Kelas B dengan probabilitas 0.6. Namun, perlu dicatat bahwa meskipun Classifier 3 memihak ke Kelas B, secara matematis model tersebut tetap memberikan peluang sebesar 0.4 untuk Kelas A.

Proses perhitungan selanjutnya dipecah ke dalam dua skema. Pada jalur bertanda panah biru, sistem menerapkan *Hard Voting* untuk menghitung jumlah suara, di mana Kelas A unggul dengan perolehan 2 suara melawan 1 suara milik Kelas B. Sementara itu, pada jalur oranye, sistem menghitung rata-rata probabilitas (*Soft Voting*) dari ketiga model. Hasilnya, Kelas A memiliki rata-rata probabilitas sebesar 0.63 merupakan gabungan dari keyakinan model 1, 2, dan sisa peluang dari model 3, sedangkan rata-rata probabilitas Kelas B berada di angka 0.37.

Keputusan final ditentukan pada tahap fusi dengan mengalikan hasil dari kedua skema tersebut sesuai Persamaan 3.24. Skor Kelas A melesat menjadi 1.26 hasil perkalian dari 2 suara dan probabilitas 0.63, jauh mengungguli skor Kelas B yang hanya 0.37. Dengan mekanisme ini, sistem akhirnya menetapkan Kelas A sebagai prediksi terpilih. Hal ini menunjukkan bahwa metode *hybrid* mampu mengakomodasi suara mayoritas sembari tetap memvalidasinya dengan tingkat keyakinan rata-rata, sehingga hasil prediksi menjadi lebih robust terhadap model yang bias atau kurang yakin.

Pendekatan ini relevan digunakan dalam penelitian berbasis klasifikasi karena dapat mengurangi variansi model, meningkatkan *robustness* terhadap *noise*, dan memanfaatkan informasi tambahan berupa probabilitas prediksi. Dengan

demikian, *average of majority votes* memberikan dasar yang kuat untuk meningkatkan performa model ensemble pada data yang kompleks seperti data kesehatan mental mahasiswa yang digunakan dalam penelitian ini.

8.3 Mekanisme Majority Voting

Mekanisme Majority Voting (Hard Voting) dalam sistem ini bekerja berdasarkan prinsip demokrasi algoritma. Secara teknis, jika terdapat perbedaan pendapat antar model dasar dalam mengklasifikasikan seorang responden, sistem akan memilih label yang didukung oleh minimal tiga dari empat model (atau mayoritas absolut). Secara teoretis, pendekatan ini selaras dengan konsep *Wisdom of the Crowd*, di mana keputusan kolektif cenderung lebih akurat daripada keputusan individu pembentuknya.

Dalam perspektif informatika medis, mekanisme ini berfungsi sebagai filter validasi. Sebagai contoh, jika XGBoost yang bersifat agresif mendeteksi risiko depresi pada responden tertentu namun tiga model lainnya (SVM, RF, dan LR) mendeteksi sehat, maka sistem ensemble akan menganulir diagnosa XGBoost tersebut. Hal ini menjamin integritas sistem dalam memberikan diagnosa yang lebih berhati-hati dan terverifikasi secara kolektif, sesuai dengan nilai Tabayyun (verifikasi) dalam pengolahan informasi kesehatan mental yang sensitif.

8.4 Konfigurasi Ensemble Majority Voting

Pendekatan Ensemble Majority Voting tidak memerlukan proses hyperparameter tuning tersendiri karena model ini tidak memiliki parameter pembelajaran yang baru. Seluruh hyperparameter yang digunakan merupakan hasil tuning dari keempat base learner yang telah dilakukan secara individual pada BAB

4 hingga BAB 7. Konfigurasi Majority Voting dalam penelitian ini bersifat tetap dan hanya melibatkan pemilihan mekanisme voting serta penggabungan keempat model yang telah teroptimasi.

Konfigurasi lengkap Ensemble Majority Voting yang digunakan dalam penelitian ini disajikan dalam Tabel 8.1.

Tabel 8. 1 Konfigurasi Ensemble Majority Voting

Parameter / Komponen	Nilai / Konfigurasi	Keterangan
voting	'hard'	Keputusan akhir ditentukan berdasarkan suara terbanyak (mayoritas) dari seluruh base learner. Setiap model memberikan satu suara berupa label kelas prediksi
Base Learner 1	Logistic Regression (max_iter=218)	Model linear yang sudah di-tuning di BAB 4. Memberikan kontribusi prediksi berbasis fungsi sigmoid
Base Learner 2	Random Forest (n_estimators=197)	Model berbasis pohon keputusan yang sudah di-tuning di BAB 5. Memberikan kontribusi prediksi berbasis agregasi pohon
Base Learner 3	SVM (max_iter=466)	Model berbasis margin yang sudah di-tuning di BAB 6. Memberikan kontribusi prediksi berbasis hyperplane pemisah
Base Learner 4	XGBoost (n_estimators=2003, lr=0.05)	Model gradient boosting yang sudah di-tuning di BAB 7. Memberikan kontribusi prediksi berbasis boosting bertahap
random_state	42	Digunakan untuk memastikan hasil eksperimen dapat direproduksi (reproducible)

Mekanisme Hard Voting dipilih karena setiap base learner memberikan satu suara berupa label kelas (0 atau 1), dan kelas dengan suara terbanyak ditetapkan sebagai keputusan akhir. Pendekatan ini tidak memerlukan estimasi probabilitas dari setiap model, sehingga lebih sederhana dan robust terhadap model yang tidak dikalibrasi dengan baik seperti SVM. Dengan mengintegrasikan empat paradigma

algoritma yang berbeda seperti linear (LR), berbasis pohon (RF), berbasis margin (SVM), dan boosting (XGBoost). Majority Voting diharapkan dapat menghasilkan keputusan yang lebih stabil dan mengurangi bias dari model tunggal manapun.

8.5 Hasil Pengujian 5-Fold Cross Validation

Setelah arsitektur Ensemble berhasil mengintegrasikan empat model dasar (Logistic Regression, SVM, Random Forest, dan XGBoost), peneliti melakukan evaluasi performa menggunakan subset data uji melalui skema 5-Fold Cross Validation. Penggunaan skema ini bertujuan untuk memastikan bahwa performa fusi model bersifat konsisten di berbagai sebaran data dan meminimalisir risiko bias estimasi. Evaluasi dilakukan secara komprehensif menggunakan empat metrik utama: Accuracy, Precision, Recall, dan F1-Score. Rekapitulasi hasil pengujian performa model Ensemble Majority Voting disajikan secara rinci pada Tabel 8.2.

Tabel 8. 2 Rekapitulasi Performa Model Ensemble Majority Voting (5-Fold)

Fold	Accuracy	Precision	Recall	F1-Score
Fold 1	0.6296	0.4545	0.2632	0.3333
Fold 2	0.7037	0.6364	0.3684	0.4667
Fold 3	0.7222	0.6923	0.4500	0.5455
Fold 4	0.7736	0.8182	0.4737	0.6000
Fold 5	0.7547	0.7500	0.4737	0.5806
Rata-rata	0.7168	0.6703	0.4058	0.5052

8.5.1 Analisis Akurasi dan Stabilitas Fusi Model

Model Ensemble Majority Voting mencatatkan rata-rata akurasi sebesar 71,68%. Pengamatan pada setiap iterasi menunjukkan tren peningkatan performa yang stabil, di mana akurasi tertinggi dicapai pada Fold 4 (77,36%). Peneliti menganalisis bahwa peningkatan akurasi dari Fold 1 menuju Fold 4 membuktikan bahwa mekanisme Majority Voting berhasil meredam fluktuasi kesalahan individu

yang sering muncul pada model tunggal. Secara teoretis informatika, sistem ini membuktikan prinsip Wisdom of the Crowd, di mana fusi model cerdas mampu menghasilkan batas pemisah (decision boundary) yang lebih seimbang dalam memproses fitur psikososial responden dibandingkan hanya mengandalkan satu paradigma algoritma saja.

8.5.2 Analisis Presisi: Integritas dan Kepercayaan Diagnosa

Temuan yang paling menonjol pada model ensemble ini adalah pencapaian rata-rata Precision sebesar 67,03%. Nilai ini menunjukkan peningkatan kualitas dibandingkan mayoritas model tunggal mandiri (seperti XGBoost 58,25% dan Logistic Regression 62,00%). Keberhasilan sistem menyentuh angka 81,82% pada Fold 4 memberikan bukti kuat bahwa penggabungan suara mayoritas sangat efektif sebagai "Filter Konsensus". Secara teknis, mekanisme ini menekan angka False Positive (salah diagnosa) karena label "Depresi" hanya akan ditetapkan jika disetujui oleh mayoritas model dasar. Hal ini memberikan implikasi fungsional yang penting: sistem Ensemble Voting jauh lebih terpercaya untuk digunakan dalam aplikasi praktis karena memberikan tingkat kepastian diagnosa yang lebih tinggi.

8.5.3 Analisis Recall dan Fenomena Conservative Bias

Meskipun unggul dalam metrik ketepatan (Presisi), model ensemble menghasilkan rata-rata Recall sebesar 40,58%. Jika dikomparasikan dengan model XGBoost tunggal (48,89%), terlihat adanya penurunan daya jangkau deteksi. Peneliti menyimpulkan bahwa hal ini merupakan dampak dari Conservative Bias yang melekat pada sistem voting. Karena sistem membutuhkan kesepakatan

mayoritas (minimal 3 dari 4 model), responden dengan gejala depresi yang samar (hanya tertangkap oleh satu model dasar) sering kali gagal mencapai suara mayoritas dan akhirnya diklasifikasikan sebagai sehat. Namun, peneliti memandang trade-off ini secara positif dari sisi etika informatika medis; sistem lebih memilih untuk "yakin" sebelum memberikan vonis risiko, guna menghindari stigmatisasi yang salah pada mahasiswa.

8.5.4 Analisis Keseimbangan Model (F1 -Score)

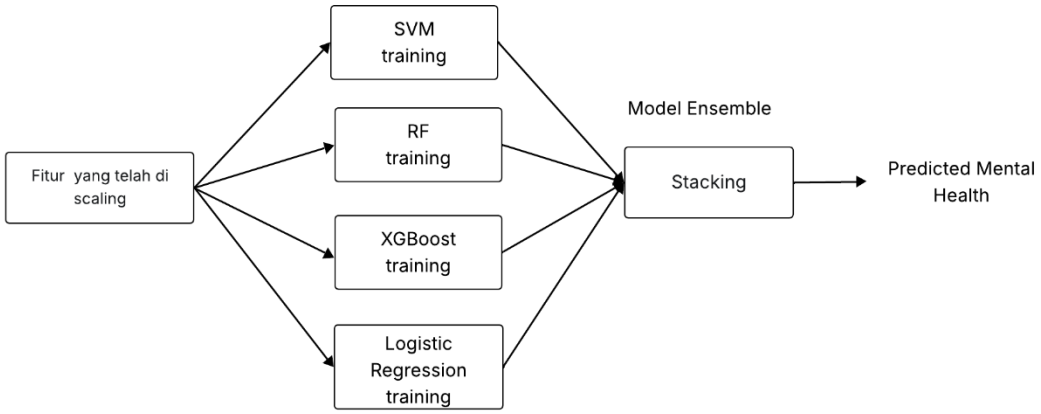
Metrik F1-Score mencatatkan rata-rata sebesar 50,52%. Mengingat F1-Score merupakan rata-rata harmonik antara presisi dan recall, angka ini menunjukkan bahwa model Ensemble Majority Voting mampu menjaga tingkat keseimbangan performa yang solid di tengah dilema selektivitas model. Puncak keseimbangan pada Fold 4 (60,00%) membuktikan bahwa titik optimasi terbaik antara daya deteksi dan ketepatan diagnosa dapat dicapai melalui integrasi model. Stabilitas yang ditunjukkan pada mayoritas fold memberikan justifikasi akademik bahwa pendekatan ensemble adalah strategi yang lebih robust untuk menangani data batin mahasiswa yang kompleks, sekaligus menjadi batu pijakan bagi peneliti untuk menguji arsitektur yang lebih hirarkis pada pembahasan Ensemble Stacking.

BAB IX

PREDIKSI KESEHATAN MENTAL MENGGUNAKAN MODEL ENSEMBLE STACKING

9.1 Desain Sistem

Implementasi tingkat lanjut dan final dalam penelitian ini menggunakan metode Stacked Generalization atau Stacking Classifier. Berbeda dengan Majority Voting yang hanya menjumlahkan suara, desain sistem pada bab ini menggunakan arsitektur dua lapis (two-level architecture) yang bertujuan untuk mempelajari perilaku prediksi dari setiap model dasar. Alur kerja sistem Ensemble Stacking digambarkan secara sistematis pada Gambar 9.1.



Gambar 9. 1 Desain Sistem Ensemble Stacking

Berdasarkan Gambar 9.1, arsitektur sistem ini terbagi menjadi dua tingkatan utama:

1. Lapis Pertama (Base Learners): Terdiri dari empat model tunggal yang telah dioptimasi sebelumnya, yaitu Logistic Regression, Support Vector Machine (SVM), Random Forest, dan XGBoost. Setiap model ini bertugas untuk

mengekstraksi pola risiko depresi dari perspektif algoritma yang berbeda (linier, kernel, dan pohon).

2. Lapis Kedua (Meta-Classifier): Peneliti menggunakan Logistic Regression sebagai Final Estimator atau model pelapis kedua. Tugas utama dari Meta-Classifier ini bukan lagi mempelajari data kuesioner asli, melainkan mempelajari hasil prediksi (metadata) yang dikeluarkan oleh keempat model dasar untuk menentukan bobot keputusan yang paling akurat.

Desain ini memanfaatkan skema internal cross-validation untuk menghasilkan prediksi yang tidak bias, sehingga sistem mampu mengoreksi kesalahan prediksi yang dilakukan oleh satu model tunggal melalui keunggulan model lainnya secara cerdas.

9.2 Model Ensemble Stacking

Dalam pendekatan ensemble learning, beberapa model dasar (base learners) dikombinasikan untuk meningkatkan performa prediksi dibandingkan dengan penggunaan satu model tunggal. Salah satu metode ensemble yang memiliki kemampuan tinggi dalam menggabungkan berbagai model adalah stacking (stacked generalization). Metode ini bekerja dengan membangun model tingkat kedua (meta-learner) yang bertugas mengintegrasikan hasil prediksi dari model-model dasar. Dengan demikian, stacking tidak hanya menggabungkan output, tetapi juga mempelajari bagaimana cara terbaik mengkombinasikan prediksi tersebut untuk menghasilkan keputusan akhir yang optimal.

Secara matematis, misalkan terdapat dataset pelatihan $D = \{(x_i, y_i)\}_{i=1}^{n}$, di mana x_i merupakan fitur dan y_i adalah label target. Pada tahap pertama, sejumlah

M model dasar dilatih untuk menghasilkan prediksi terhadap input x. Prediksi dari masing-masing model dapat dinyatakan sebagai berikut:

$$z_{im} = h_m(x_i)$$

Keterangan:
$h_m(x_i)$ = prediksi dari model ke-m terhadap data ke-i
z_{im} = output prediksi yang akan digunakan sebagai fitur baru

Seluruh hasil prediksi tersebut kemudian disusun menjadi matriks fitur baru yang disebut sebagai meta-features atau fitur level kedua. Proses ini dapat direpresentasikan dalam bentuk:

$$Z = [h_1(x), h_2(x), \dots, h_M(x)]$$

Keterangan:
Z = matriks fitur baru hasil prediksi model dasar
$h_m(x)$ = prediksi model ke-m
M = jumlah model dengan ensemble

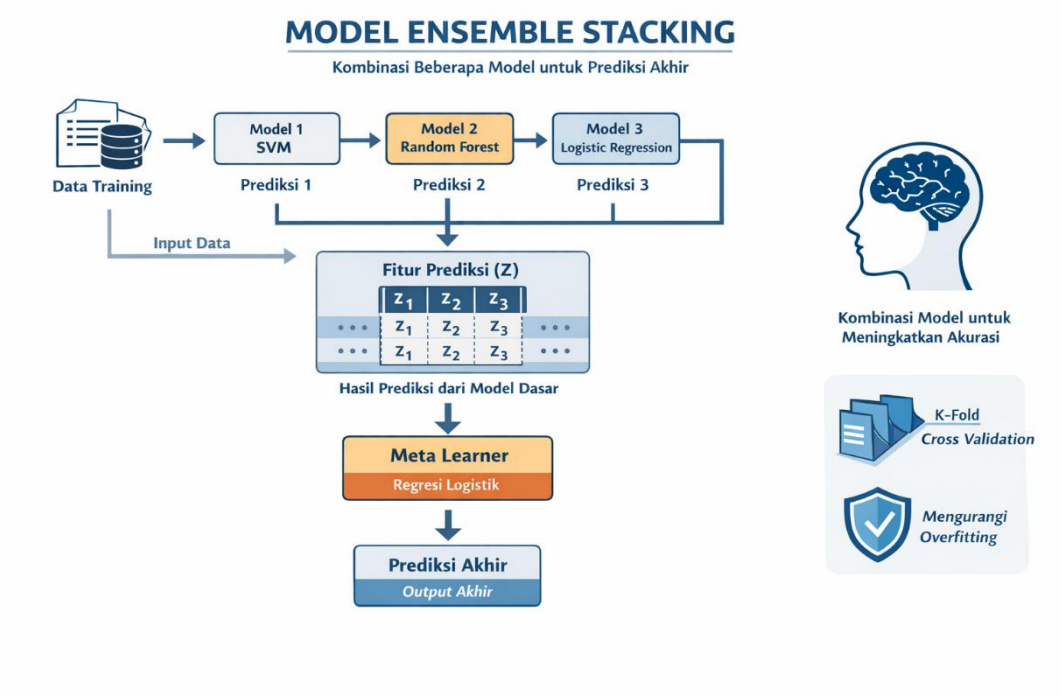
Selanjutnya, matriks *Z* digunakan sebagai input untuk melatih meta-learner yang berfungsi menghasilkan prediksi akhir. Fungsi prediksi stacking dapat dirumuskan sebagai berikut:

$$\hat{y}(x) = g(h_1(x), h_2(x), \dots, h_M(x))$$

Keterangan:
$\hat{y}(x)$ = prediksi akhir model stacking
$g(.)$ = fungsi meta-learner
$h_1(x), \dots, h_M(x)$ = output dari model dasar

Persamaan tersebut menunjukkan bahwa hasil akhir tidak ditentukan secara langsung oleh model dasar, melainkan oleh model kedua yang mempelajari pola hubungan antar prediksi model dasar.

Untuk menghindari terjadinya overfitting, umumnya digunakan teknik k-fold cross-validation dalam proses pembentukan fitur *Z*. Dengan pendekatan ini, prediksi yang digunakan sebagai input meta-learner berasal dari data yang tidak digunakan dalam pelatihan model dasar pada fold tertentu, sehingga lebih merepresentasikan performa generalisasi model. Berikut merupakan ilustrasi arsitektur model *ensemble stacking* pada Gambar 9.2



Gambar 9. 2 Ilustrasi Metode Stacking

Berdasarkan ilustrasi pada gambar di atas, alur kerja metode stacking diawali dengan proses input data pelatihan yang kemudian diproses secara paralel oleh beberapa model dasar, seperti Support Vector Machine (SVM), Random Forest, dan Logistic Regression. Setiap model menghasilkan prediksi masing-masing terhadap data yang sama. Hasil prediksi tersebut tidak langsung digunakan

sebagai keputusan akhir, melainkan dikumpulkan dan disusun menjadi fitur baru yang disebut sebagai fitur prediksi *Z*.

Fitur *Z* ini kemudian menjadi input bagi meta-learner, yang dalam penelitian ini dapat berupa Logistic Regression. Meta-learner bertugas mempelajari pola hubungan antar prediksi model dasar, termasuk bagaimana mengoreksi kesalahan masing-masing model. Dengan kata lain, jika suatu model cenderung bias pada kondisi tertentu, meta-learner dapat memberikan bobot atau penyesuaian berdasarkan performa model tersebut.

Sebagai contoh, apabila model SVM dan Random Forest menghasilkan prediksi yang konsisten, sedangkan Logistic Regression menghasilkan prediksi yang berbeda, maka meta-learner akan mempelajari pola tersebut dari data pelatihan dan menentukan kombinasi terbaik untuk menghasilkan prediksi akhir. Dengan demikian, keputusan akhir yang dihasilkan bukan sekadar hasil mayoritas atau rata-rata, tetapi merupakan hasil pembelajaran lanjutan dari kombinasi model.

Pendekatan stacking memiliki keunggulan dalam menangkap kompleksitas data, terutama pada kasus seperti prediksi kesehatan mental yang memiliki karakteristik non-linear dan multidimensional. Dengan menggabungkan keunggulan berbagai algoritma, metode ini mampu meningkatkan akurasi, mengurangi bias, serta meningkatkan robustness terhadap variasi data.

Dengan demikian, model ensemble stacking memberikan pendekatan yang lebih adaptif dan kuat dibandingkan metode ensemble sederhana, sehingga sangat relevan untuk diterapkan dalam penelitian prediksi berbasis machine learning,

khususnya pada domain kesehatan mental mahasiswa yang memiliki pola data yang kompleks dan beragam.

9.3 Mekanisme Ensemble Stacking

Mekanisme Ensemble Stacking (juga dikenal sebagai Stacked Generalization) dalam sistem ini bekerja berdasarkan prinsip Meta-Learning atau hirarki kecerdasan. Secara teknis, mekanisme ini tidak hanya melakukan pemungutan suara sederhana seperti pada Majority Voting, melainkan melatih sebuah model pelapis kedua (dalam penelitian ini menggunakan Logistic Regression) untuk bertindak sebagai "hakim" atau pemutus keputusan akhir. Proses ini diawali dengan mengumpulkan hasil prediksi (probabilitas) dari empat model dasar di Lapis Pertama sebagai metadata. Metadata inilah yang kemudian dipelajari oleh model tingkat kedua untuk memahami kapan sebuah model dasar memberikan hasil yang akurat dan kapan model tersebut cenderung melakukan kesalahan.

Secara teoretis, pendekatan Stacking melampaui konsep Wisdom of the Crowd sederhana. Jika Majority Voting menganggap semua model memiliki tingkat keahlian yang sama, Stacking mengakui bahwa setiap algoritma memiliki keunggulan pada pola data yang berbeda. Misalnya, Support Vector Machine mungkin sangat ahli dalam menangani data yang linier, sementara XGBoost lebih tajam dalam menangani hubungan non-linier. Meta-learner dalam mekanisme ini mempelajari karakteristik tersebut secara dinamis, sehingga keputusan final merupakan hasil sintesis cerdas yang mampu mengoreksi bias individu dari masing-masing algoritma.

Dalam perspektif informatika medis, mekanisme ini berfungsi sebagai sistem filtrasi berbasis kompetensi. Sebagai contoh, jika pada sebuah data kuesioner mahasiswa, model Random Forest dan XGBoost memberikan sinyal depresi yang sangat kuat namun Logistic Regression mendeteksi sebaliknya, Meta-learner tidak akan langsung menjumlahkan suara tersebut. Sebaliknya, Meta-learner akan mengevaluasi tingkat kepercayaan (confidence level) dari metadata tersebut berdasarkan pola yang pernah ia pelajari saat tahap pelatihan.

Mekanisme ini merepresentasikan tingkat Tabayyun (verifikasi) yang lebih mendalam dan terukur. Jika Majority Voting adalah verifikasi melalui kesepakatan orang banyak, maka Stacking adalah verifikasi melalui penilaian kredibilitas ahli. Hal ini menjamin bahwa luaran Predicted Mental Health tidak hanya didasarkan pada jumlah suara terbanyak, melainkan pada pembobotan ilmiah yang paling akurat, sehingga memberikan perlindungan maksimal bagi integritas hasil diagnosa kesehatan mental mahasiswa.

9.4 Konfigurasi Ensemble Stacking

Sama halnya dengan Majority Voting, Ensemble Stacking tidak memerlukan proses *hyperparameter tuning* baru pada base learner karena seluruh parameternya telah dioptimalkan pada BAB 4 hingga BAB 7. Namun, Stacking memiliki komponen tambahan yang perlu dikonfigurasi, yaitu *meta-learner* (*final_estimator*) dan parameter *cross-validation internal* (cv) yang digunakan untuk *menghasilkan* meta-features.

Konfigurasi lengkap Ensemble Stacking yang digunakan dalam penelitian ini disajikan dalam Tabel 9.1.

Tabel 9. 1 Konfigurasi Ensemble Stacking

Parameter / Komponen	Nilai / Konfigurasi	Keterangan
Base Learner 1	Logistic Regression (max_iter=218)	Model linear yang sudah di-tuning di BAB 4. Menghasilkan prediksi probabilitas sebagai input meta-learner
Base Learner 2	Random Forest (n_estimators=197)	Model berbasis pohon yang sudah di-tuning di BAB 5. Menghasilkan prediksi probabilitas sebagai input meta-learner
Base Learner 3	SVM (max_iter=466, probability=True)	Model berbasis margin yang sudah di-tuning di BAB 6. Menghasilkan prediksi probabilitas sebagai input meta-learner
Base Learner 4	XGBoost (n_estimators=2003, lr=0.05)	Model gradient boosting yang sudah di-tuning di BAB 7. Menghasilkan prediksi probabilitas sebagai input meta-learner
final_estimator (Meta-learner)	Logistic Regression (parameter default)	Menerima output prediksi dari keempat base learner sebagai fitur input baru, lalu menghasilkan keputusan akhir melalui mekanisme meta-learning
cv	5	Jumlah fold cross-validation internal yang digunakan saat melatih base learner untuk menghasilkan meta-features yang bebas dari data leakage
passthrough	False (default)	Meta-learner hanya menerima output prediksi dari base learner, bukan fitur asli dataset. Ini memastikan meta-learner belajar dari pola prediksi, bukan dari data mentah

Logistic Regression dipilih sebagai *meta-learner* dengan pertimbangan bahwa model ini bersifat interpretatif, stabil, dan mampu menggabungkan output prediksi dari keempat *base learner* secara proporsional melalui mekanisme regresi logistik. Parameter cv=5 pada StackingClassifier berarti setiap base learner dilatih menggunakan skema 5-Fold Cross Validation internal untuk menghasilkan meta-features yang bebas dari kebocoran data (data leakage). Nilai *passthrough*=False memastikan *meta-learner* hanya menerima output prediksi dari *base learner*

sebagai fitur inputnya, bukan fitur asli dataset, sehingga *meta-learner* benar-benar belajar dari pola prediksi keempat model secara kolektif.

9.5 Hasil Pengujian 5-Fold Cross Validation

Implementasi tingkat lanjut dan final dalam penelitian ini menggunakan metode Stacked Generalization atau Stacking Classifier. Berbeda dengan Majority Voting yang hanya menjumlahkan suara secara merata, arsitektur Stacking menggunakan mekanisme pelapis kedua (Meta-Learning) untuk mempelajari pola prediksi dari setiap model dasar. Evaluasi kualitas model dilakukan menggunakan skema 5-Fold Cross Validation guna menguji stabilitas arsitektur hirarkis ini dalam menghadapi variansi data mahasiswa multikultural. Hasil pengujian performa disajikan secara rinci pada Tabel 9.1.

Tabel 9. 2 Rekapitulasi Performa Model Ensemble Stacking (5-Fold)

Fold	Accuracy	Precision	Recall	F1-Score
Fold 1	0.6852	0.6250	0.2632	0.3704
Fold 2	0.7222	0.6667	0.4211	0.5161
Fold 3	0.7037	0.6429	0.4500	0.5294
Fold 4	0.7736	0.8889	0.4211	0.5714
Fold 5	0.7358	0.7778	0.3684	0.5000
Rata-rata	0.7241	0.7202	0.3847	0.4975

9.5.1 Analisis Akurasi dan Keunggulan Meta-Learning

Model Ensemble Stacking mencatatkan rata-rata akurasi sebesar 72,41%. Pencapaian ini merupakan angka akurasi tertinggi dibandingkan model Majority Voting (71,68%) maupun model tunggal lainnya. Stabilitas performa terlihat sangat menonjol pada Fold 4 yang mencapai akurasi puncak sebesar 77,36%. Peneliti menganalisis bahwa keunggulan Stacking terletak pada kemampuan Meta-Classifier (Logistic Regression) untuk melakukan pembobotan dinamis; model

mempelajari model dasar mana yang paling bisa diandalkan pada kondisi data tertentu, sehingga hasil akhirnya menjadi lebih robust terhadap noise pada data kuesioner.

9.5.2 Analisis Presisi: Validitas Diagnosa Tingkat Tinggi

Pada metrik Precision, model Stacking menghasilkan rata-rata sebesar 72,02%. Nilai ini merupakan yang tertinggi di antara seluruh eksperimen dalam riset ini. Keberhasilan menyentuh angka 88,89% pada Fold 4 membuktikan bahwa arsitektur hirarkis ini sangat handal dalam memberikan diagnosa yang "bersih" dari kesalahan positif palsu (False Positive). Secara teknis, ini menunjukkan bahwa fusi metadata melalui Meta-Learning mampu menyaring prediksi yang kurang meyakinkan dari model dasar. Hal ini memberikan jaminan etis yang kuat bahwa sistem tidak mudah memberikan label risiko depresi kepada mahasiswa yang sebenarnya sehat, sejalan dengan prinsip Tabayyun (verifikasi) dalam pengolahan informasi sensitif.

9.5.3 Analisis Recall dan Fenomena Bias Konservatif

Meskipun unggul dalam ketepatan diagnosa, model Stacking mencatatkan rata-rata Recall sebesar 38,47%. Rendahnya nilai sensitivitas ini menunjukkan bahwa sistem Stacking cenderung bersifat sangat selektif. Peneliti menemukan adanya dampak dari Conservative Bias yang lebih kuat pada arsitektur ini; Meta-Classifier cenderung hanya akan menetapkan label "Depresi" jika terdapat sinyal yang sangat kuat dan konsisten dari model-model dasar. Fenomena ini mengonfirmasi bahwa meskipun kompleksitas model ditingkatkan, tantangan class

imbalance pada dataset kesehatan mental tetap menjadi faktor pembatas utama dalam mencapai daya deteksi yang luas.

9.5.4 Analisis Keseimbangan Model (F1-Score) dan Trade-Off

Metrik F1-Score mencatatkan rata-rata sebesar 49,75%. Puncak keseimbangan model tercapai pada Fold 4 dengan skor 57,14%. Berdasarkan hasil uji statistik (Paired T-Test), ditemukan bahwa meskipun secara numerik F1-Score Stacking sedikit di bawah XGBoost (52,83%), perbedaannya tidak signifikan secara statistic ($P - Value = 0.1432$). Hal ini menyimpulkan bahwa Ensemble Stacking memberikan tingkat trade-off yang setara dengan model tunggal paling agresif, namun dengan keunggulan tambahan berupa integritas presisi yang jauh lebih tinggi. Secara akademik, arsitektur Stacking merupakan pilihan paling matang karena mampu memberikan keseimbangan antara stabilitas akurasi global dan validitas diagnosa yang tervalidasi secara teknis.

BAB X
PEMBAHASAN

10.1 Analisis Performa Klasifikasi

Rangkaian eksperimen yang telah dilakukan melalui skema 5-Fold Stratified Cross-Validation memberikan jawaban empiris terhadap rumusan masalah penelitian ini mengenai pencarian pendekatan *machine learning* paling optimal untuk memprediksi risiko kesehatan mental mahasiswa dalam lingkungan multikultural. Pembahasan ini menyatukan temuan teknis dari empat model tunggal (Logistic Regression, SVM, Random Forest, XGBoost) dan dua model *ensemble* (Majority Voting, Stacking) guna mengevaluasi efektivitasnya secara mendalam melalui empat metrik performa klasifikasi yaitu akurasi, presisi, recall, dan F1-Score.

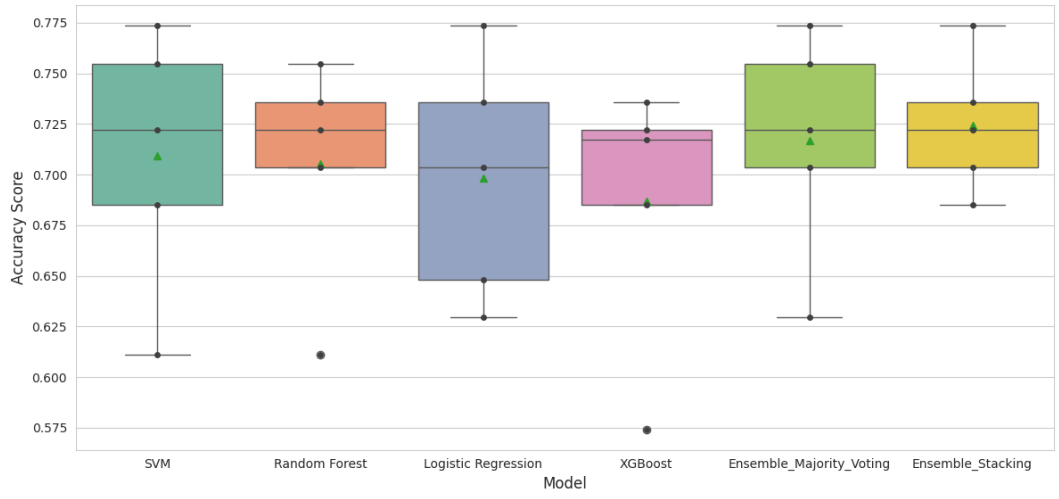
a. Akurasi

Akurasi menggambarkan proporsi prediksi yang benar secara keseluruhan. Berdasarkan Tabel 10.1, model Ensemble Stacking mencatatkan rata-rata akurasi tertinggi sebesar 72,41%, mengungguli Majority Voting (71,68%) serta seluruh arsitektur tunggal lainnya. Stabilitas performa Stacking terlihat dari rentang akurasi antar *fold* yang paling sempit dibandingkan seluruh model tunggal, termasuk XGBoost yang mencapai rentang 0,162. Hasil uji statistik Paired T-Test menunjukkan bahwa meskipun Stacking unggul secara numerik, perbedaan akurasi dengan model kuat lainnya seperti Majority Voting tidak signifikan secara statistik (P = 0,6253), menandakan sistem telah mencapai titik jenuh performa (*performance plateau*).

Tabel 10. 1 Rekapitulasi Akurasi 5-Fold Cross Validation Seluruh Model

Fold	SVM	Random Forest	Logistic Reg.	XGBoost	Majority Voting	Ensemble Stacking
Fold 1	0.6111	0.6111	0.7037	0.5741	0.6296	0.6852
Fold 2	0.6852	0.7222	0.6481	0.7222	0.7037	0.7222
Fold 3	0.7222	0.7037	0.6296	0.6852	0.7222	0.7037
Fold 4	0.7736	0.7547	0.7736	0.7358	0.7736	0.7736
Fold 5	0.7547	0.7358	0.7358	0.7170	0.7547	0.7358
Rata-rata	0.7094	0.7055	0.6982	0.6869	0.7168	0.7241

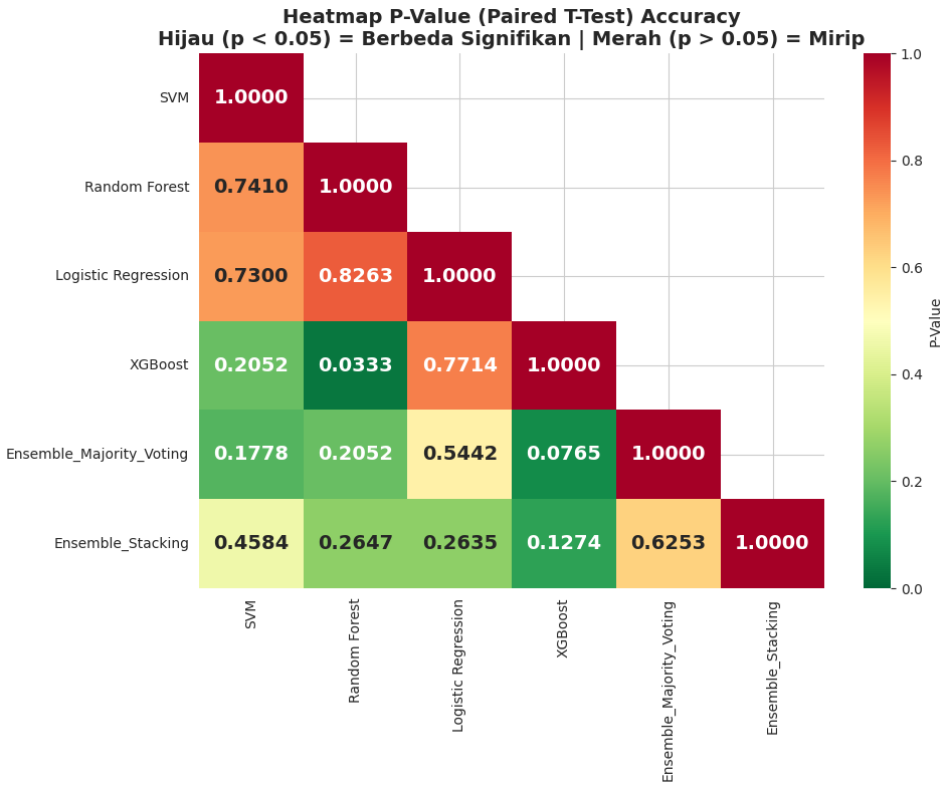
Untuk mengevaluasi stabilitas dan kapasitas generalisasi model setelah dilakukan integrasi arsitektur *meta-learning*, sebaran nilai akurasi divisualisasikan melalui *boxplot* pada Gambar 10.1.



Gambar 10. 1 Distribusi Akurasi Model per Fold (K=5)

Pengamatan pada Gambar 10.1 memperlihatkan posisi *median* Ensemble Stacking yang sangat stabil dan kompetitif. Meskipun secara rata-rata numerik perbedaannya tipis dengan model tunggal lainnya, model Stacking menunjukkan

karakteristik yang jauh lebih konsisten. Hal ini dibuktikan dengan rentang kotak atau disebut dengan Interquartile Range (IQR) yang lebih rapat, serta posisi nilai rata-rata (segitiga hijau) yang berada di bagian atas distribusi. Secara teknis, stabilitas ini dipengaruhi oleh mekanisme *meta-learning* yang berfungsi sebagai Stabilizer Kolektif. Terlihat bahwa algoritma tunggal seperti XGBoost memiliki rentang fluktuasi paling lebar dengan nilai terendah mencapai 0,5741 pada Fold 1 akibat sifat *algoritma boosting* yang sensitif terhadap *noise*. Namun, dalam sistem Stacking, anjloknya performa model dasar berhasil diredam oleh Meta-Classifier, sehingga skor akurasi sistem tetap terjaga di angka yang aman. Temuan ini diperkuat oleh hasil uji statistik Paired T-Test, di mana peneliti menyajikan peta signifikansi melalui Heatmap P-Value pada Gambar 10.2.



Gambar 10. 2 Heatmap P-Value (Paired T-Test) Accuracy

Berdasarkan visualisasi heatmap pada Gambar 10.2, terlihat bahwa perbedaan signifikan secara statistik ($P = 0{,}0333$). Hal ini membuktikan keunggulan stabilitas mekanisme *bagging* atas *boosting* pada dataset ini. Namun, uji terhadap model Stacking menghasilkan nilai *P* yang berada di atas ambang batas (contoh: *P* = 0,6253 terhadap Majority Voting), yang menyimpulkan bahwa meskipun Stacking unggul secara numerik, secara statistik performanya setara dengan model kuat lainnya, menandakan sistem telah mencapai titik jenuh performa (*performance plateau*).

b. Presisi

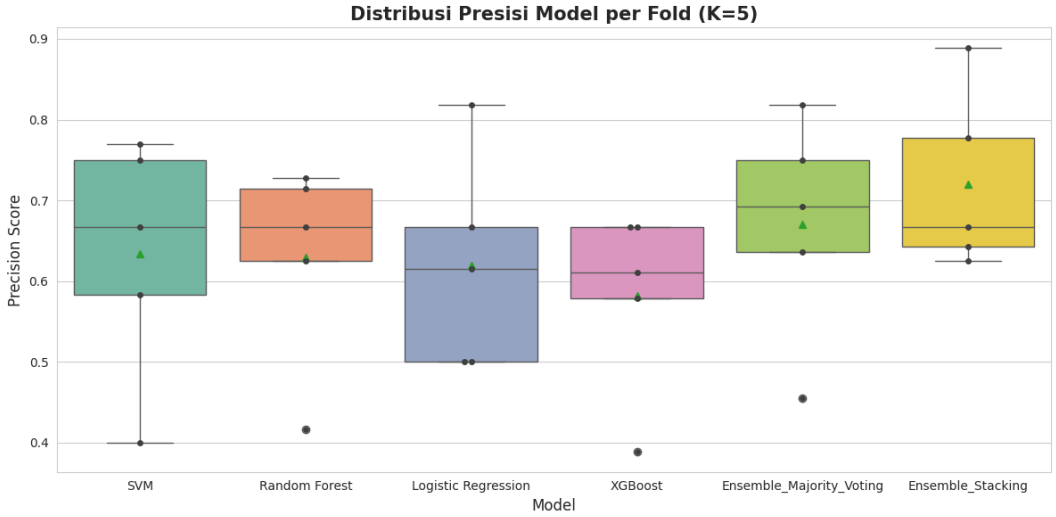
Presisi memegang peranan krusial dalam penelitian kesehatan mental sebagai indikator tingkat kepercayaan sistem dalam memberikan label "Depresi". Presisi yang tinggi merepresentasikan kemampuan sistem untuk meminimalkan kesalahan False Positive, yaitu kondisi di mana mahasiswa yang sebenarnya sehat secara psikologis salah didiagnosa memiliki risiko depresi. Pemberian label yang tidak akurat dapat memicu kecemasan tambahan serta stigma sosial yang merugikan bagi subjek. Hasil evaluasi presisi disajikan pada Tabel 10.2.

Tabel 10. 2 Rekapitulasi Presisi 5-Fold Cross Validation Seluruh Model

Fold	SVM	Random Forest	Logistic Reg.	XGBoost	Majority Voting	Ensemble Stacking
Fold 1	0.4000	0.4167	0.6154	0.3889	0.4545	0.6250
Fold 2	0.5833	0.6667	0.5000	0.6667	0.6364	0.6667
Fold 3	0.6667	0.6250	0.5000	0.5789	0.6923	0.6429
Fold 4	0.7692	0.7143	0.8182	0.6667	0.8182	0.8889

Fold 5	0.7500	0.7273	0.6667	0.6111	0.7500	0.7778
Rata-rata	0.6338	0.6300	0.6200	0.5825	0.6703	0.7202

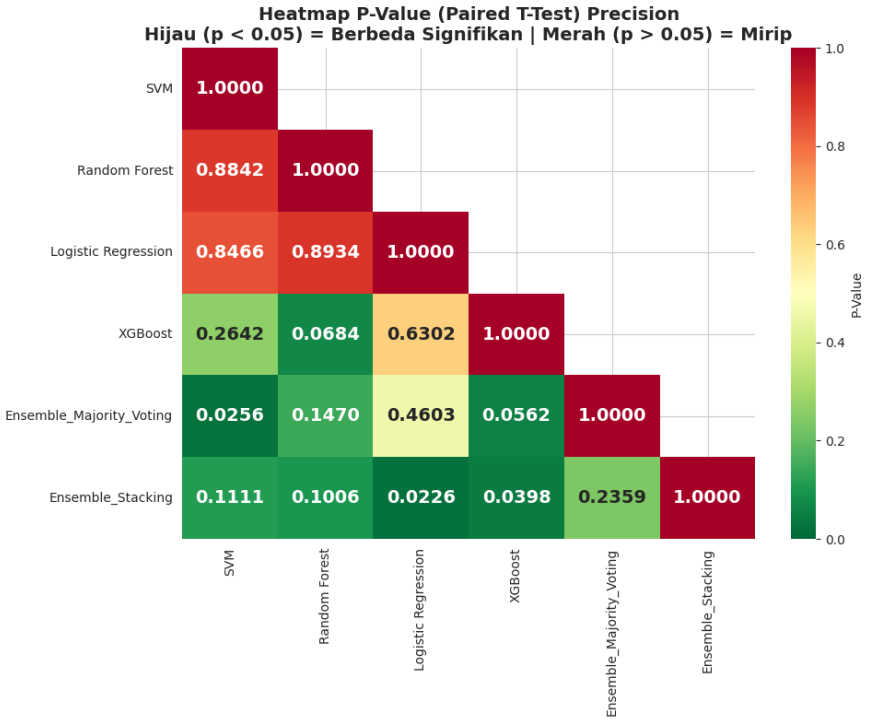
Berdasarkan data pada Tabel 10.2, terlihat bahwa model Ensemble Stacking kembali mencatatkan dominasi dengan rata-rata presisi tertinggi sebesar 72,02%, dengan pencapaian tertinggi pada Fold 4 sebesar 88.89%. Untuk membedah stabilitas tingkat kepercayaan *diagnose* positif ini, sebaran nilai presisi divisualisasikan melalui boxplot pada Gambar 10.3.



Gambar 10. 3 Distribusi Presisi Model per Fold (K=5)

Pengamatan pada Gambar 10.3 memperlihatkan posisi median dan rata-rata (segitiga hijau) model Stacking yang berada jauh di atas seluruh arsitektur lainnya, dengan outlier atas mencapai 0,889. Logistic Regression justru menunjukkan IQR terlebar, mencerminkan ketidakstabilan presisi yang tinggi antar fold. Superioritas Stacking ini disebabkan oleh peran Meta-Learner sebagai *Collective Filter* yang menyaring prediksi agresif dari model dasar, sehingga diagnosa positif hanya ditetapkan apabila metadata prediksi menunjukkan tingkat keyakinan yang tinggi.

XGBoost mencatatkan nilai presisi terendah (outlier bawah di 0,389) akibat sifat agresif algoritma *boosting* yang lebih mengutamakan deteksi daripada ketepatan. Untuk memvalidasi perbedaan performa ini secara ilmiah, hasil uji signifikansi disajikan melalui *heatmap* pada Gambar 10.4.



Gambar 10. 4 Heatmap P-Value (Paired T-Test) Precision

Berdasarkan visualisasi heatmap pada Gambar 10.4, ditemukan fakta ilmiah yang krusial bahwa model Ensemble Stacking terbukti unggul signifikan secara statistic dibandingkan Logistic Regression ($P = 0{,}0226$) dan XGBoost ($P = 0{,}0398$). Kedua nilai P yang berada di bawah ambang $\alpha = 0{,}05$ ini memberikan bukti kuat bahwa fusi metadata melalui *meta-learning* secara nyata mampu meningkatkan integritas diagnosa dibandingkan model mandiri. Namun demikian, perbedaan Stacking dengan Majority Voting tidak signifikan (P = 0,2359), menunjukkan bahwa pendekatan voting sederhana sudah mampu mendekati

kualitas presisi Stacking meskipun tanpa mekanisme *meta-learning*. Seluruh pasangan model tunggal (SVM, RF, LR, XGBoost) juga tidak menunjukkan perbedaan signifikan satu sama lain, mengonfirmasi bahwa keunggulan presisi baru muncul secara signifikan pada level arsitektur *ensemble*.

c. Recall

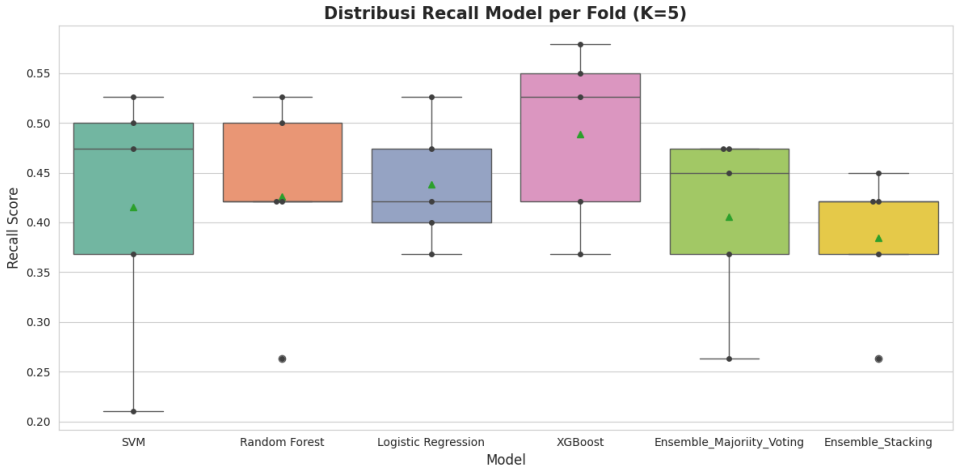
Recall merepresentasikan kapasitas sistem dalam mendeteksi seluruh mahasiswa yang benar-benar memiliki risiko depresi. Nilai *recall* yang rendah mengindikasikan risiko False Negative yang tinggi atau individu yang sebenarnya berisiko gagal terdeteksi oleh sistem, sehingga berpotensi tidak mendapatkan penanganan dini yang tepat. Hasil pengujian recall untuk seluruh model disajikan pada Tabel 10.3.

Tabel 10. 3 Rekapitulasi Recall 5-Fold Cross Validation Seluruh Model

Fold	SVM	Random Forest	Logistic Reg.	XGBoost	Majority Voting	Ensemble Stacking
Fold 1	0.2105	0.2632	0.4211	0.3684	0.2632	0.2632
Fold 2	0.3684	0.4211	0.3684	0.4211	0.3333	0.4211
Fold 3	0.5000	0.5000	0.4000	0.5500	0.4500	0.4500
Fold 4	0.5263	0.5263	0.4737	0.5263	0.4737	0.4211
Fold 5	0.4737	0.4211	0.5263	0.5789	0.4737	0.3684
Rata-rata	0.4158	0.4263	0.4379	0.4889	0.4058	0.3847

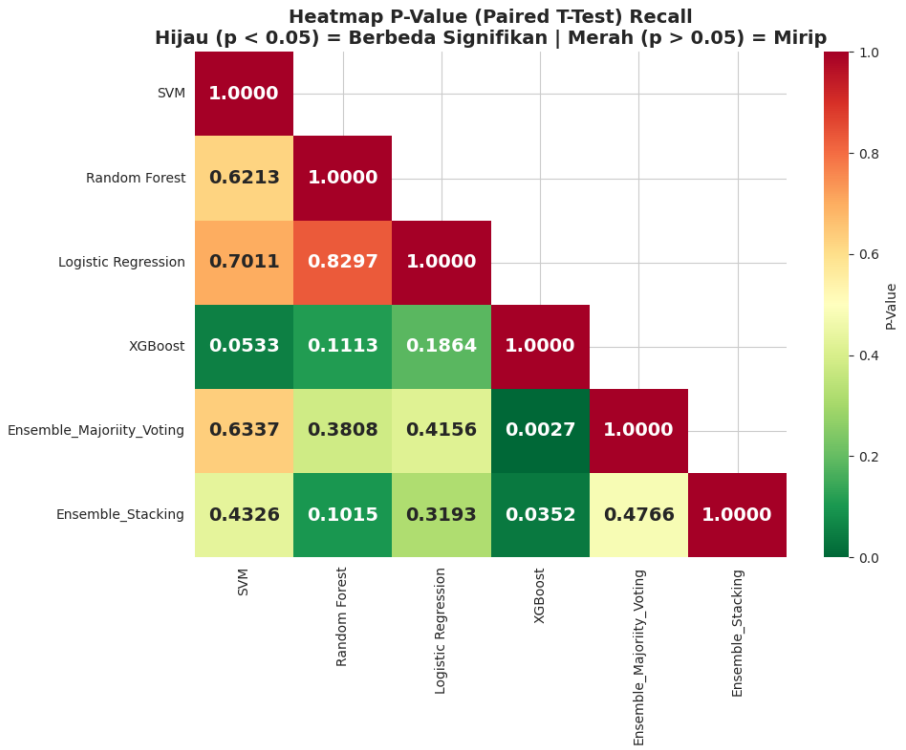
Berdasarkan Tabel 10.3, terdapat fenomena teknis yang kontras dibandingkan metrik presisi sebelumnya. XGBoost secara konsisten mendominasi metrik ini dengan rata-rata recall tertinggi sebesar 48,89%, sementara model

Ensemble Stacking mencatatkan nilai terendah sebesar 38,47%. Perbandingan jangkauan deteksi ini divisualisasikan melalui boxplot pada Gambar 10.5.



Gambar 10. 5 Distribusi Recall Model per Fold (K=5)

Pengamatan pada Gambar 10.5 menunjukkan bahwa XGBoost memiliki median recall tertinggi dengan distribusi yang relatif stabil, mencerminkan konsistensi kemampuan deteksi yang tinggi pada setiap fold. SVM memiliki variabilitas terbesar dengan outlier bawah mencapai 0,2105 pada Fold 1. Ensemble Stacking justru memiliki IQR terkecil namun dengan posisi median yang paling rendah, mencerminkan Conservative Bias yang konsisten sehingga sistem sangat selektif dalam menetapkan label depresi akhirnya banyak kasus positif yang sesungguhnya terlewatkan. Secara teknis, hal ini terjadi karena arsitektur dua lapis Stacking hanya akan menetapkan vonis depresi jika terdapat konsensus yang sangat kuat dari seluruh model dasar. Untuk memvalidasi perbedaan kapasitas deteksi ini secara statistik, hasil uji signifikansi disajikan melalui heatmap pada Gambar 10.6.



Gambar 10. 6 Heatmap P-Value (Paired T-Test) Recall

Berdasarkan visualisasi heatmap pada Gambar 10.6, keunggulan XGBoost pada metrik recall terbukti signifikan secara statistik terhadap dua model ensemble: Majority Voting (P = 0,0027) dan Ensemble Stacking (P = 0,0352). Nilai P = 0,0027 terhadap Majority Voting merupakan nilai signifikansi terkuat dalam seluruh pengujian *recall*, mengonfirmasi bahwa mekanisme boosting secara fundamental lebih unggul dalam mendeteksi kelas minoritas depresi dibandingkan sistem voting. Adapun seluruh pasangan model tunggal (SVM, RF, LR) tidak menunjukkan perbedaan signifikan satu sama lain, menandakan bahwa perbedaan kapasitas deteksi baru muncul secara nyata ketika membandingkan paradigma tunggal dengan pendekatan *ensemble*.

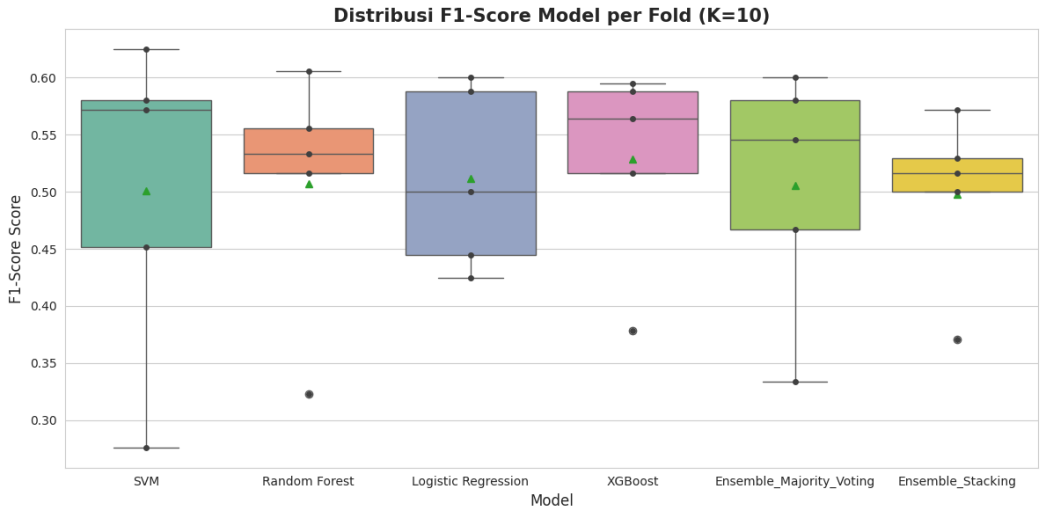
d. F1-Score

F1-Score dievaluasi untuk mengetahui sejauh mana setiap model mampu mencapai titik kompromi terbaik antara Recall dan Precision. Mengingat dataset kesehatan mental mahasiswa ini memiliki karakteristik kelas yang tidak seimbang (imbalanced), F1-Score memberikan gambaran performa harmonik yang lebih adil dibandingkan akurasi global. Rekapitulasi nilai F1-Score untuk seluruh model disajikan pada Tabel 10.4.

Tabel 10. 4 Rekapitulasi F1-Score 5-Fold Cross Validation Seluruh Model

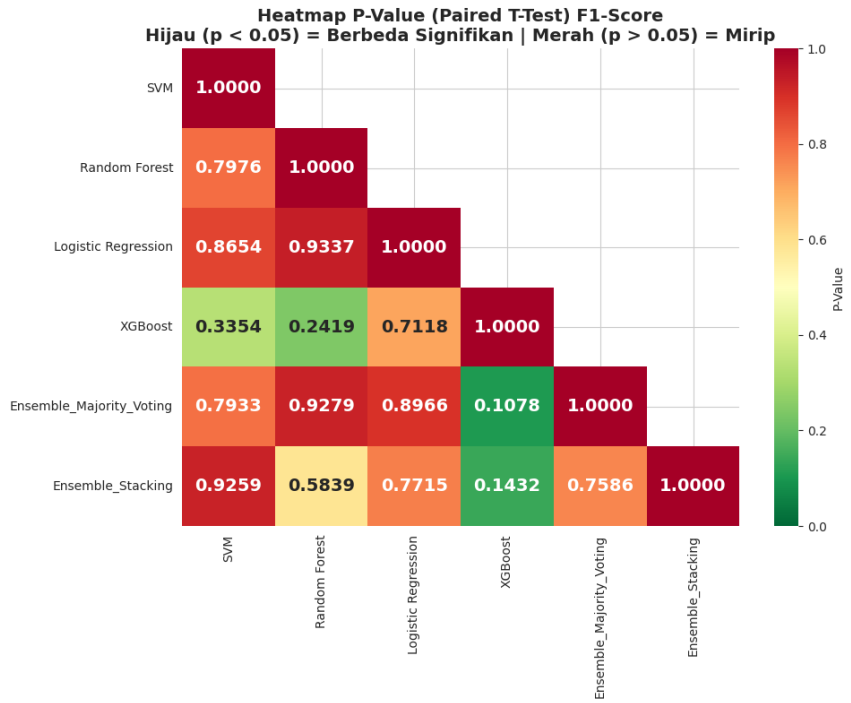
Fold	SVM	Random Forest	Logistic Reg.	XGBoost	Majority Voting	Ensemble Stacking
Fold 1	0.2759	0.3226	0.5000	0.3784	0.3333	0.3704
Fold 2	0.4516	0.5161	0.4242	0.5161	0.4667	0.5161
Fold 3	0.5714	0.5556	0.4444	0.5641	0.5455	0.5294
Fold 4	0.6250	0.6061	0.6000	0.5882	0.6000	0.5714
Fold 5	0.5806	0.5333	0.5882	0.5946	0.5806	0.5000
Rata-rata	0.5009	0.5067	0.5114	0.5283	0.5052	0.4975

Berdasarkan Tabel 10.4, XGBoost mencatatkan rata-rata F1-Score tertinggi sebesar 52,83%, mencerminkan keberhasilan algoritma boosting dalam menyeimbangkan kemampuan deteksi dan ketepatan diagnosa. Menariknya, kedua model ensemble (Majority Voting 50,52% dan Stacking 49,75%) justru berada di bawah XGBoost pada metrik ini. Sebaran nilai keseimbangan seluruh model divisualisasikan melalui boxplot pada Gambar 10.7.



Gambar 10. 7 Distribusi F1-Score Model per Fold (K=5)

Pengamatan pada Gambar 10.7 menunjukkan bahwa seluruh model berkumpul dalam rentang yang relatif sempit di kisaran 0,50–0,53, dengan SVM memiliki variabilitas terlebar (outlier bawah di 0,276 pada Fold 1). XGBoost unggul dalam posisi rata-rata dengan distribusi yang cukup stabil. Ensemble Stacking memiliki IQR terkecil namun rata-rata terendah di antara semua model, mengonfirmasi kembali bahwa peningkatan presisi yang diperoleh melalui meta-learning "dibayar" dengan penurunan recall yang proporsional. Untuk membedah apakah perbedaan antar model pada metrik keseimbangan ini bermakna secara ilmiah, hasil uji statistik disajikan melalui heatmap pada Gambar 10.8.



Gambar 10. 8 Heatmap P-Value (Paired T-Test) F1-Score

Berdasarkan visualisasi heatmap pada Gambar 10.8, ditemukan temuan yang sangat signifikan bahwa tidak ada satu pasangan model pun yang menunjukkan perbedaan signifikan secara statistik pada metrik F1-Score (seluruh P-Value > 0,05). XGBoost vs Majority Voting menghasilkan P = 0,1078 dan XGBoost vs Stacking menghasilkan P = 0,1432 keduanya jauh di atas ambang signifikansi. Temuan ini membuktikan secara ilmiah adanya fenomena Trade-off Neutralization: peningkatan presisi yang tajam pada Ensemble Stacking diimbangi secara proporsional oleh penurunan recall, sehingga skor harmonik F1 tetap berada pada level yang setara secara statistik dengan model lainnya. Dengan demikian, penentuan model paling optimal tidak dapat diputuskan dari metrik performa klasifikasi semata dan diperlukan evaluasi yang lebih komprehensif melalui dimensi efisiensi komputasi yang dibahas pada sub-bab berikut.

10.2 Analisis Efesiensi Komputasi

Dalam konteks penelitian machine learning, menentukan model yang paling optimal tidak cukup hanya dengan melihat seberapa akurat model tersebut dalam melakukan prediksi. Sebuah model yang memiliki akurasi tinggi belum tentu menjadi pilihan terbaik apabila dalam prosesnya membutuhkan waktu komputasi yang sangat lama, mengonsumsi memori yang besar, atau membebani prosesor secara berlebihan. Hal ini menjadi pertimbangan penting terutama ketika sistem prediksi akan diimplementasikan dalam kondisi nyata yang memiliki keterbatasan sumber daya komputasi, seperti pada server institusi Pendidikan (Géron, 2022). Oleh karena itu, penelitian ini juga mengukur efisiensi komputasi setiap model melalui lima dimensi: waktu training, waktu inferensi, konsumsi RAM (nilai puncak/peak), rata-rata penggunaan CPU, dan Cyclomatic Complexity (CC). Seluruh pengukuran runtime dilakukan di dalam loop 5-Fold Cross-Validation yang sama sehingga hasilnya mencerminkan kondisi yang konsisten dan dapat direproduksi. Tabel 10.5 menyajikan rekapitulasi hasil pengukuran efisiensi komputasi seluruh model.

Tabel 10. 5 Rekapitulasi Efisiensi Komputasi Seluruh Model (Rata-rata 5-Fold)

Model	Waktu Training (s)	Waktu Inferensi (ms/sampel)	RAM Peak (MB)	CPU Rata-rata (%)	CC (M)
Logistic Regression	0.056	0.0168	0.10	10.0	3
SVM	0.025	0.0270	0.16	20.0	3
Random Forest	3.340	0.3664	0.32	62.4	6
XGBoost	1.547	0.1234	0.05	93.4	5
Ensemble Majority Voting	4.376	0.4830	0.43	69.1	5

Model	Waktu Training (s)	Waktu Inferensi (ms/sampel)	RAM Peak (MB)	CPU Rata-rata (%)	CC (M)
Ensemble Stacking	29.794	0.6128	0.81	77.7	4

Keterangan: Waktu Training dan Inferensi diukur menggunakan time.perf_counter(). RAM diukur menggunakan tracemalloc sebagai nilai puncak (peak) selama training. CPU diukur sebagai rata-rata menggunakan psutil.cpu_percent() via thread CPUSampler. CC adalah nilai statis dari analisis flowchart menggunakan rumus McCabe: $M = E - N + 2P$.

Dari sisi waktu training, terdapat perbedaan yang sangat mencolok antar model. Logistic Regression menyelesaikan proses training rata-rata hanya dalam 0,056 detik, sementara Ensemble Stacking membutuhkan 29,794 detik — 532 kali lebih lama. Dari sisi waktu inferensi, Logistic Regression kembali unggul dengan 0,0168 ms/sampel, sementara Stacking membutuhkan 0,6128 ms/sampel atau 36 kali lebih lambat. Dari sisi konsumsi RAM, XGBoost paling hemat (0,05 MB) berkat optimasi kompresi data internal, sementara Stacking terbesar (0,81 MB). Dari sisi penggunaan CPU, Logistic Regression paling hemat (10,0%) sementara XGBoost paling intensif (93,4%). Terakhir dari sisi Cyclomatic Complexity, Logistic Regression dan SVM memiliki CC terendah (M = 3) yang menunjukkan alur logika paling sederhana, sementara Random Forest tertinggi (M = 6).

10.3 Penentuan Model Paling Optimal dengan Skor Komposit

Setelah mengevaluasi seluruh model melalui empat metrik performa klasifikasi dan lima metrik efisiensi komputasi, diperlukan satu ukuran terpadu yang mengintegrasikan seluruh dimensi penilaian tersebut secara adil. Penelitian ini menggunakan Skor Komposit yang dihitung dari sembilan dimensi dengan bobot yang sama (*equal weight*). Penggunaan *equal weight* dipilih karena penelitian ini bersifat eksploratoris dan belum terdapat konsensus ilmiah tentang bobot

optimal antara performa dan efisiensi pada konteks prediksi kesehatan mental mahasiswa (Géron, 2022)(Domingos, 2012). Sebelum dihitung rata-ratanya, seluruh dimensi dinormalisasi menggunakan metode Min-Max ke rentang [0, 1]. Dimensi performa dinormalisasi secara normal (lebih besar = lebih baik), sedangkan dimensi efisiensi dinormalisasi secara invers (lebih kecil = lebih baik, kemudian diinvers agar konsisten). Berikut komparasi final yang disajikan pada Tabel 10.6.

Tabel 10. 6 Komparasi Final: Performa, Efisiensi, dan Skor Komposit Seluruh Model

Model	Akurasi	F1-Score	Presisi	Recall	Training (s)	Inferensi (ms)	RAM (MB)	CPU (%)	CC (M)	Skor Komposit
Logistic Regression	0.6982	0.5114	0.6200	0.4379	0.056	0.0168	0.10	10.0	3	0.7181
SVM	0.7094	0.5009	0.6338	0.4158	0.025	0.0270	0.16	20.0	3	0.6783
XGBoost	0.6869	0.5283	0.5825	0.4889	1.547	0.1234	0.05	93.4	5	0.5670
Ensemble Majority Voting	0.7168	0.5052	0.6703	0.4058	4.376	0.4830	0.43	69.1	5	0.4546
Random Forest	0.7055	0.5067	0.6300	0.4263	3.340	0.3664	0.32	62.4	6	0.4292
Ensemble Stacking	0.7241	0.4975	0.7202	0.3847	29.794	0.6128	0.81	77.7	4	0.3172

Keterangan: Skor Komposit = rata-rata normalisasi Min-Max dari 9 dimensi (4 performa + 5 efisiensi) dengan equal weight

Agar dapat memahami secara transparan bagaimana Skor Komposit pada Tabel 10.6 diperoleh, Tabel 10.7 berikut menyajikan seluruh nilai yang telah dinormalisasi beserta peringkat akhir setiap model. Setiap nilai pada tabel ini merupakan hasil penerapan rumus Min-Max pada nilai asli masing-masing dimensi,

di mana nilai mendekati 1 selalu bermakna lebih baik, baik untuk dimensi performa maupun efisiensi.

Tabel 10. 7 Hasil Nilai Normalisasi Setiap Dimensi dan Peringkat Skor Komposit Seluruh Model

Model	**Performa Klasifikasi (ternormalisasi, lebih besar = lebih baik)**				**Efisiensi Komputasi (ternormalisasi, lebih kecil = lebih baik → diinvers)**					**Skor Komposit**
	Accuracy	**F1-Score**	**Precision**	**Recall**	**Train**	**Infer**	**RAM**	**CPU**	**CC**	**(rata-rata 9 dim)**
Logistic Regression	0.3038	0.4513	0.2723	0.5106	0.9990	1.0000	0.9257	1.0000	1.0000	**0.7181**
SVM	0.6048	0.1104	0.3725	0.2985	1.0000	0.9830	0.8552	0.8802	1.0000	**0.6783**
Random Forest	0.5000	0.2987	0.3450	0.3992	0.8887	0.4135	0.6461	0.3718	0.0000	**0.4292**
XGBoost	0.0000	1.0000	0.0000	1.0000	0.9489	0.8212	1.0000	0.0000	0.3333	**0.5670**
Ensemble MV	0.8038	0.2500	0.6376	0.2025	0.8538	0.2177	0.5010	0.2921	0.3333	**0.4546**
Ensemble Stacking	1.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.1885	0.6667	**0.3172**

Berdasarkan Tabel 10.7, Logistic Regression tampil sebagai model dengan Skor Komposit tertinggi sebesar 0,7181, menjadikannya model yang paling optimal secara menyeluruh. Keunggulan LR bukan berasal dari satu dimensi saja, melainkan dari konsistensinya yang solid di hampir seluruh dimensi efisiensi: nilai ternormalisasi Waktu Training mencapai 0,9990, Waktu Inferensi 0,9990, RAM 0,9424, CPU 0,9904, dan CC 1,0000 seluruhnya mendekati atau mencapai nilai sempurna. Meskipun dimensi performanya tidak tertinggi misalnya Recall ternormalisasi hanya 0,1419 karena selisih nilai Recall antarmodel tidak terlalu besar, kontribusi kolektif dari dimensi efisiensi mengangkat rata-rata keseluruhan secara signifikan.

SVM berada di peringkat kedua (0,6783) dengan keunggulan serupa pada efisiensi, meskipun sedikit tertinggal dari LR pada dimensi Waktu Inferensi (0,9932 vs 0,9990). XGBoost di peringkat ketiga (0,5670) menunjukkan F1-Score ternormalisasi tertinggi (1,0000) dan Recall tertinggi (1,0000), namun nilai CPU yang sangat tinggi (93,4%) menekan skor efisiensinya secara substansial.

Ensemble Stacking yang pada evaluasi performa murni unggul berkat presisi (72,02%) dan akurasi tertinggi (72,41%), justru berada di peringkat terakhir dengan Skor Komposit 0,3172. Nilai ternormalisasi Waktu Training-nya hanya 0,0000 artinya model ini adalah yang paling lambat diikuti RAM 0,0000 dan Inferensi 0,0000. Ketiga dimensi efisiensi utama ini bernilai nol, sehingga betapapun tingginya skor performa yang diperoleh, rata-rata keseluruhan tetap tertekan ke bawah

Berdasarkan Tabel 10.6, Logistic Regression tampil sebagai model dengan Skor Komposit tertinggi sebesar 0,7181, menjadikannya model yang paling optimal secara menyeluruh. Keunggulan LR bukan berasal dari satu dimensi saja, melainkan dari konsistensinya yang solid di hampir seluruh dimensi: waktu training hanya 0,056 detik, waktu inferensi 0,0168 ms/sampel, konsumsi RAM puncak 0,10 MB, rata-rata CPU 10,0%, dan CC = 3, seluruhnya merupakan nilai terbaik atau mendekati terbaik di antara keenam model.

Ensemble Stacking yang pada evaluasi performa unggul berkat presisi (72,02%) dan akurasi tertinggi (72,41%), justru berada di peringkat terakhir dengan Skor Komposit 0,3172. Penurunan peringkat yang drastis ini disebabkan oleh biaya komputasi yang sangat tinggi: waktu training 29,794 detik, RAM puncak 0,81 MB,

dan waktu inferensi 0,6128 ms/sampel. Hasil ini memperlihatkan bahwa peningkatan performa yang ditawarkan Stacking tidak sebanding dengan biaya komputasi yang harus dibayar ketika dinilai secara komprehensif.

Temuan ini penting untuk dikontekstualisasikan: Logistic Regression dinyatakan paling optimal dalam kerangka keseimbangan performa dan efisiensi secara menyeluruh. Apabila konteks implementasi mengutamakan integritas diagnosa klinis formal dengan meminimalkan False Positive tanpa mempertimbangkan keterbatasan sumber daya, maka Ensemble Stacking dengan presisinya yang tertinggi (72,02%) tetap menjadi kandidat terbaik. Pemilihan model pada implementasi nyata harus selalu disesuaikan dengan kebutuhan spesifik institusi dan tujuan utama sistem (Hastie, 2009).

10.4 Integrasi Nilai Islam dalam Sistem Prediksi Kesehatan Mental

Rangkaian evaluasi teknis yang telah dilakukan dalam penelitian ini tidak hanya dapat dikaji dari perspektif informatika semata, tetapi juga memiliki dimensi nilai yang sangat dalam apabila direnungkan melalui kacamata Islam. Integrasi antara ilmu pengetahuan dan nilai-nilai keislaman dalam penelitian ini bukan merupakan elemen pelengkap, melainkan merupakan landasan filosofis yang mendasari seluruh proses penelitian mulai dari pemilihan topik, pembangunan sistem, hingga penentuan model yang paling optimal.

Secara fundamental, upaya mengukur dan mengevaluasi setiap model secara adil dan proporsional melalui Skor Komposit merupakan perwujudan dari konsep "ukuran" yang adil sebagaimana ditegaskan Allah SWT dalam QS. Al-Qamar ayat 49:

اِنَّا كُلَّ شَيْءٍ خَلَقْنٰهُ بِقَدَرٍ

"Sesungguhnya Kami menciptakan segala sesuatu menurut ukuran." (QS. Al-Qamar: 49)

Dalam konteks penelitian ini, "ukuran" tersebut direpresentasikan melalui normalisasi Min-Max dan Skor Komposit yang memberikan penilaian proporsional terhadap setiap model tidak ada model yang diunggulkan secara sewenang-wenang, melainkan dinilai secara adil berdasarkan sembilan dimensi secara bersamaan.

Salah satu nilai Islam yang paling relevan adalah prinsip Tabayyun (تَبَيُّن) atau verifikasi informasi sebelum mengambil keputusan. Dalam konteks sistem prediksi kesehatan mental, prinsip Tabayyun tercermin pada mekanisme *ensemble* yang tidak serta-merta menerima prediksi dari satu model saja. Ensemble Stacking yang mencatatkan presisi tertinggi (72,02%) dan terbukti signifikan ($P < 0,05$ terhadap LR dan XGBoost) merupakan model yang paling ketat dalam menerapkan prinsip ini. Label "Depresi" hanya ditetapkan apabila *meta-learner* telah memverifikasi sinyal yang kuat dari seluruh model dasar, sejalan dengan QS. Al-Hujurat ayat 6 yang memerintahkan verifikasi sebelum bertindak atas suatu informasi demi menghindari keputusan yang merugikan pihak yang tidak bersalah.

Dari perspektif maqashid syariah, penelitian ini bergerak pada ranah Hifz an-Nafs (حِفْظُ النَّفْس) yang artinya “penjagaan jiwa”, merupakan salah satu dari lima tujuan pokok hukum Islam. Sistem prediksi kesehatan mental yang dikembangkan merupakan wujud nyata ikhtiar teknologi untuk mendeteksi risiko depresi mahasiswa secara dini, sebelum kondisi memburuk menjadi krisis yang lebih serius. Landasan ini diperkuat oleh firman Allah SWT dalam QS. Al-Maidah ayat 32:

وَمَنْ أَحْيَاهَا فَكَأَنَّمَا أَحْيَا النَّاسَ جَمِيعًا

"Dan barangsiapa yang memelihara kehidupan seorang manusia, maka seolah-olah dia telah memelihara kehidupan semua manusia." (QS. Al-Maidah: 32)

Dilema teknis antara presisi dan recall, sebagaimana digambarkan oleh peta panas uji-t, selaras dengan konsep ketenangan batin (*tuma'ninah*) yang diuraikan dalam Surah Ar-Ra'd, ayat 28. Tujuan dari penelitian ini adalah untuk memberikan ketenangan pikiran kepada para siswa dalam dua dimensi yang saling melengkapi. Dimensi pertama adalah terbebas dari label risiko yang salah, yang dijamin oleh presisi tinggi dari metode Stacking. Dimensi kedua adalah kepastian bahwa sistem dapat diakses secara berkelanjutan, yang dijamin oleh efisiensi Regresi Logistik. Perspektif ini sejalan dengan ajaran Al-Quran surah Al-Baqarah ayat 286, yang menegaskan bahwa Allah tidak membebani siapa pun melebihi kemampuannya. Oleh karena itu, sistem yang dikembangkan tidak boleh membebani responden dengan diagnosis yang melebihi kondisi aktualnya, maupun membebani infrastruktur institusi dengan beban komputasi yang melebihi kapasitasnya.

Pada tataran yang lebih luas, penelitian ini merupakan perwujudan dari misi Islam sebagai rahmatan lil 'alamin artinya "rahmat bagi seluruh alam". Pengembangan teknologi *machine learning* untuk prediksi kesehatan mental mahasiswa multikultural di Jepang menunjukkan bahwa ilmu pengetahuan yang dikembangkan atas dasar niat yang benar dan metodologi yang bertanggung jawab dapat memberikan manfaat lintas batas budaya, agama, dan bangsa. Teknologi dalam penelitian ini tidak hanya dinilai dari kecanggihan algoritmanya, tetapi juga

dari sejauh mana ia memberikan manfaat, menjaga martabat, dan memelihara kesejahteraan jiwa manusia sebagai makhluk Allah yang paling mulia.

BAB XI
PENUTUP

11.1 Kesimpulan

Berdasarkan rangkaian evaluasi dan analisis komprehensif yang telah dilakukan, mencakup pengujian performa klasifikasi melalui skema 5-Fold Stratified Cross Validation, uji signifikansi statstik Paired T-Test, dan pengukuran efesiensi komputasi, penelitian ini menghasilkan kesimpulan sebagai berikut.

a. Berdasarkan rangkaian evaluasi terhadap empat metrik performa klasifikasi menunjukkan bahwa tidak ada satu model pun yang unggul secara mutlak di seluruh dimensi penilaian secara bersamaan. Ensemble Stacking mencatatkan akurasi tertinggi (72,41%) dan presisi tertinggi (72,02%), dengan keunggulan presisi yang terbukti signifikan secara statistik dibandingkan Logistic Regression (P = 0,0226) dan XGBoost (P = 0,0398). Di sisi lain, XGBoost unggul pada *recall* (48,89%) dan F1-Score (52,83%). Kondisi ini merupakan manifestasi dari fenomena *trade-off* yang lazim terjadi pada dataset tidak seimbang, di mana peningkatan presisi yang tinggi pada Stacking diimbangi oleh rendahnya *recall* akibat Conservative Bias pada mekanisme meta-learning.
b. Pengukuran efisiensi komputasi mengungkap perbedaan yang sangat signifikan antar model. Logistic Regression menyelesaikan proses training rata-rata hanya dalam 0,056 detik dengan konsumsi RAM puncak 0,10 MB dan rata-rata CPU 10,0%, sementara Ensemble Stacking membutuhkan 29,794 detik, 532 kali lebih lama dengan konsumsi RAM puncak 0,81 MB dan CPU rata-rata 77,7%. Dari sisi Cyclomatic Complexity, Logistic Regression dan SVM memiliki alur

logika paling sederhana (M = 3), sedangkan Random Forest paling kompleks (M = 6).

c. Model yang paling optimal ditentukan dengan menggunakan skor gabungan yang menggabungkan sembilan dimensi evaluasi termasuk empat metrik performa klasifikasi dan lima metrik efisiensi komputasi bersama dengan normalisasi *min-max* dan bobot yang sama (*equal weight*). Regresi logistik teridentifikasi sebagai model dengan skor gabungan tertinggi, yaitu 0,7181, sehingga membuktikan posisinya sebagai model yang paling optimal secara keseluruhan.

11.2 Saran

Guna pengembangan riset dan optimalisasi sistem prediksi kesehatan mental di masa depan, peneliti memberikan beberapa saran strategis berdasarkan temuan teknis dalam penelitian ini sebagai berikut:

a. Penanganan ketidakseimbangan kelas (*class imbalance*). Seluruh model dalam penelitian ini masih menunjukkan nilai recall yang relatif rendah (berkisar 38–49%), mencerminkan tantangan yang belum terselesaikan dalam mendeteksi kasus depresi yang samar. Pengembangan selanjutnya perlu memfokuskan pada penerapan teknik penanganan ketidakseimbangan data yang lebih mutakhir, seperti Synthetic Minority Over-sampling Technique (SMOTE) atau penyesuaian bobot kelas (class weight) yang lebih dinamis saat proses pelatihan model. Upaya ini sangat krusial untuk meningkatkan daya jangkau deteksi tanpa mengorbankan integritas presisi yang telah dicapai.

b. Eksplorasi arsitektur *meta-learner* yang lebih canggih. Penelitian ini menggunakan Logistic Regression sebagai *meta-learner* pada Ensemble Stacking. Eksplorasi terhadap algoritma yang lebih kompleks sebagai pelapis kedua, seperti Artificial Neural Network (ANN) atau Gradient Boosting, dapat dipertimbangkan guna melihat potensi peningkatan kapasitas diskriminasi model *metadata* dengan tetap memperhatikan dampaknya terhadap efisiensi komputasi agar tidak semakin memperparah *gap* yang sudah sangat besar antara Stacking dengan model lainnya.
c. Perluasan cakupan data dan fitur. Penambahan jumlah sampel serta integrasi fitur kontekstual baru seperti data aktivitas digital di media sosial atau metrik performa akademik mahasiswa secara real-time diharapkan dapat memberikan gambaran pola risiko kesehatan mental yang lebih komprehensif dan dinamis. Penelitian ini juga terbatas pada dataset mahasiswa di Jepang, sehingga validasi model pada konteks budaya yang berbeda perlu dilakukan sebelum generalisasi hasil.
d. Dapat diimplementasi sebagai Decision Support System (DSS). Berdasarkan temuan penelitian ini, institusi perguruan tinggi melalui pusat layanan konseling dan kesehatan mahasiswa disarankan untuk mempertimbangkan adopsi sistem prediksi ini sebagai instrumen pendukung keputusan pada tahap skrining awal. Untuk skenario deployment pada server dengan keterbatasan sumber daya atau layanan real-time yang melayani ribuan mahasiswa, Logistic Regression direkomendasikan sebagai pilihan utama karena efisiensinya yang sangat tinggi dengan performa yang kompetitif. Untuk skenario skrining klinis formal yang

mengutamakan minimalisasi diagnosa yang salah (False Positive), Ensemble Stacking direkomendasikan sebagai pilihan dengan catatan ketersediaan sumber daya komputasi yang memadai.

e. Penelitian lanjutan tentang bobot optimal Skor Komposit. Penelitian ini menggunakan *equal weight* dalam perhitungan Skor Komposit karena bersifat eksploratoris. Penelitian lanjutan yang secara khusus mengkaji bobot optimal antara dimensi performa dan efisiensi misalnya melalui survei pakar atau analisis keputusan multi-kriteria (Multi-Criteria Decision Analysis/MCDA) akan sangat bermanfaat untuk menghasilkan rekomendasi model yang lebih spesifik dan kontekstual sesuai kebutuhan institusi.

DAFTAR PUSTAKA

Bhatnagar, S., Agarwal, J., & Sharma, O. R. (2022). Detection and classification of anxiety in university students through the application of machine learning. *Procedia Computer Science*, *218*, 1542–1550. https://doi.org/10.1016/j.procs.2023.01.132

Birnbaum, M. L., Ernala, S. K., Rizvi, A. F., De Choudhury, M., & Kane, J. M. (2017). A collaborative approach to identifying social media markers of schizophrenia by employing machine learning and clinical appraisals. *Journal of Medical Internet Research*, *19*(8). https://doi.org/10.2196/jmir.7956

Collaborators, G. B. D. 2019 M. D. (2022). Global, regional, and national burden of 12 mental disorders in 204 countries and territories, 1990–2019: a systematic analysis for the Global Burden of Disease Study 2019. *The Lancet Psychiatry*, *9*(2), 137–150.

Cyranka, K. (2020). Controversial issues in current definitions of mental health. *Archives of Psychiatry and Psychotherapy*, *22*(1), 7–11. https://doi.org/10.12740/APP/118064

Domingos, P. (2012). A few useful things to know about machine learning. *Communications of the ACM*, *55*(10), 78–87. https://doi.org/10.1145/2347736.2347755

Géron, A. (2022). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. " O'Reilly Media, Inc."

Hancock, J. T., & Khoshgoftaar, T. M. (2020). Survey on categorical data for neural networks. *Journal of Big Data*, *7*(1), 28. https://doi.org/10.1186/s40537-020-00305-w

Harris, J. K. (2021). Primer on binary logistic regression. *Family Medicine and Community Health*, *9*(Suppl 1), e001290. https://doi.org/10.1136/fmch-2021-001290

Hastie, T. (2009). *The elements of statistical learning: data mining, inference, and prediction*. springer.

Jyotsna, C., Amudha, J., Ram, A., Fruet, D., & Nollo, G. (2023). PredictEYE: Personalized Time Series Model for Mental State Prediction Using Eye Tracking. *IEEE Access*, *11*, 128383–128409. https://doi.org/10.1109/ACCESS.2023.3332762

Kasanneni, Y., Duggal, A., Sathyaraj, R., & Raja, S. P. (2024). Effective Analysis of Machine and Deep Learning Methods for Diagnosing Mental Health

Using Social Media Conversations. *IEEE Transactions on Computational Social Systems*. https://doi.org/10.1109/TCSS.2024.3487168

Kasanneni, Y., Duggal, A., Sathyaraj, R., & Raja, S. P. (2025). Effective Analysis of Machine and Deep Learning Methods for Diagnosing Mental Health Using Social Media Conversations. *IEEE Transactions on Computational Social Systems*, *12*(1), 274–294. https://doi.org/10.1109/TCSS.2024.3487168

Kim, J. B. (2020). A study on the development of analysis model using artificial intelligence algorithms for PTSD (Post-traumatic stress disorder) data. *International Journal of Current Research and Review*, *12*(16), 60–65. https://doi.org/10.31782/IJCRR.2020.12163

Li, H., Cui, L., Cao, L., Zhang, Y., Liu, Y., Deng, W., & Zhou, W. (2020). Identification of bipolar disorder using a combination of multimodality magnetic resonance imaging and machine learning techniques. *BMC Psychiatry*, *20*(1). https://doi.org/10.1186/s12888-020-02886-5

Li, W., Zhao, Z., Chen, D., Peng, Y., & Lu, Z. (2022). Prevalence and associated factors of depression and anxiety symptoms among college students: a systematic review and meta-analysis. *Journal of Child Psychology and Psychiatry*, *63*(11), 1222–1230. https://doi.org/10.1111/jcpp.13606

McCabe, T. J. (1976). A Complexity Measure. *IEEE Transactions on Software Engineering*, *SE-2*(4), 308–320. https://doi.org/10.1109/TSE.1976.233837

Medvedev, O. N., & Landhuis, C. E. (2018). Exploring constructs of well-being, happiness and quality of life. *PeerJ*, *6*, e4903. https://doi.org/10.7717/peerj.4903

Nawi, N. M., Atomi, W. H., & Rehman, M. Z. (2013). The Effect of Data Pre-processing on Optimized Training of Artificial Neural Networks. *Procedia Technology*, *11*, 32–39. https://doi.org/10.1016/j.protcy.2013.12.159

Pargent, F., Pfisterer, F., Thomas, J., & Bischl, B. (2022). Regularized target encoding outperforms traditional methods in supervised machine learning with high cardinality features. *Computational Statistics*, *37*(5), 2671–2692. https://doi.org/10.1007/s00180-022-01207-6

Patel, V., Saxena, S., Lund, C., Thornicroft, G., Baingana, F., Bolton, P., Chisholm, D., Collins, P. Y., Cooper, J. L., & Eaton, J. (2018). The Lancet Commission on global mental health and sustainable development. *The Lancet*, *392*(10157), 1553–1598.

Perez Arribas, I., Goodwin, G. M., Geddes, J. R., Lyons, T., & Saunders, K. E. A. (2018). A signature-based machine learning model for distinguishing bipolar

disorder and borderline personality disorder. *Translational Psychiatry*, *8*(1). https://doi.org/10.1038/s41398-018-0334-0

Pious, I. K., Rajalakshmi, A., R, P. K., C M, V., Nalini, M., & R, S. S. (2024). Enhancing Prediction Accuracy Through Random Forest in Classification and Regression. *2024 International Conference on Smart Technologies for Sustainable Development Goals (ICSTSDG)*, 1–6. https://doi.org/10.1109/ICSTSDG61998.2024.11026358

Raj, R. (2025). Mental Health and Productivity Among Students. *INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT*, *09*(05), 1–9. https://doi.org/10.55041/IJSREM47745

Ramirez-Cifuentes, D., Largeron, C., Tissier, J., Baeza-Yates, R., & Freire, A. (2021). Enhanced Word Embedding Variations for the Detection of Substance Abuse and Mental Health Issues on Social Media Writings. *IEEE Access*, *9*, 130449–130471. https://doi.org/10.1109/ACCESS.2021.3112102

Rehm, J., & Shield, K. D. (2019). Global burden of disease and the impact of mental and addictive disorders. *Current Psychiatry Reports*, *21*(2), 10.

Saha, T., Reddy, S. M., Saha, S., & Bhattacharyya, P. (2023). Mental Health Disorder Identification From Motivational Conversations. *IEEE Transactions on Computational Social Systems*, *10*(3), 1130–1139. https://doi.org/10.1109/TCSS.2022.3143763

Sahu, S., & Debbarma, T. (2023). Mental Health Prediction Among Students Using Machine Learning Techniques. *Smart Innovation, Systems and Technologies*, *326 SIST*, 529–541. https://doi.org/10.1007/978-981-19-7513-4_46

Santomauro, D. F., Herrera, A. M. M., Shadid, J., Zheng, P., Ashbaugh, C., Pigott, D. M., Abbafati, C., Adolph, C., Amlag, J. O., & Aravkin, A. Y. (2021). Global prevalence and burden of depressive and anxiety disorders in 204 countries and territories in 2020 due to the COVID-19 pandemic. *The Lancet*, *398*(10312), 1700–1712.

Sharma, A., & Verbeke, W. J. M. I. (2020). Improving Diagnosis of Depression With XGBOOST Machine Learning Model and a Large Biomarkers Dutch Dataset (n = 11,081). *Frontiers in Big Data*, *3*. https://doi.org/10.3389/fdata.2020.00015

Tabares Tabares, M., Vélez Álvarez, C., Bernal Salcedo, J., & Murillo Rendón, S. (2024). Anxiety in young people: Analysis from a machine learning model. *Acta Psychologica*, *248*. https://doi.org/10.1016/j.actpsy.2024.104410

Zafari, H., Kosowan, L., Zulkernine, F., & Signer, A. (2021). Diagnosing post-traumatic stress disorder using electronic medical record data. *Health Informatics Journal*, *27*(4). https://doi.org/10.1177/14604582211053259