

Implementasi Grid Search CV KNN dengan Preprocessing Z-Score Outlier Removal untuk Sistem Prediksi Risiko Kehamilan

Ivan Maulana Anggita, Muhammad Naufal*, Farrikh Alzami

Fakultas Ilmu Komputer, Teknik Informatika, Universitas Dian Nuswantoro, Semarang, Indonesia

Email: ¹111202113298@mhs.dinus.ac.id, ^{2,*}m.naufal@dsn.dinus.ac.id, ³alzami@dsn.dinus.ac.id

Email Penulis Korespondensi: m.naufal@dsn.dinus.ac.id

Submitted: 11/08/2025; Accepted: 01/09/2025; Published: 04/09/2025

Abstrak—Penelitian ini bertujuan untuk mengoptimalkan algoritma K-Nearest Neighbors (KNN) dalam memprediksi tingkat risiko kehamilan menggunakan dataset “maternal health risk” dari UCI Machine Learning Repository. Metode yang digunakan mencakup preprocessing data melalui deteksi dan penghapusan outlier dengan Z-score, normalisasi menggunakan Standard Scaling, serta encoding kategorikal ke numerik pada label target. Selanjutnya dilakukan hyperparameter tuning menggunakan GridSearchCV untuk mencari kombinasi optimal parameter KNN (jumlah tetangga, bobot jarak, dan metrik jarak). Hasil pengujian menunjukkan bahwa model KNN tanpa optimasi hanya mencapai akurasi 69,46%, sedangkan setelah optimasi akurasi meningkat menjadi 82,00% dengan macro average precision 81,91%, recall 82,89%, dan F1-score 82,23%. Evaluasi menggunakan confusion matrix juga menunjukkan peningkatan akurasi signifikan pada kelas risiko tinggi. Model yang telah dioptimalkan kemudian di-deploy dalam bentuk aplikasi web menggunakan framework Flask dan Docker melalui platform Hugging Face Spaces, memungkinkan prediksi risiko kehamilan dilakukan secara online dengan lebih cepat dan efisien. Hasil ini menunjukkan bahwa kombinasi KNN dengan GridSearchCV dan normalisasi data dapat meningkatkan performa model prediksi risiko kehamilan secara signifikan dan dapat diterapkan dalam sistem pendukung keputusan bidang kesehatan.

Kata Kunci: K-Nearest Neighbors; GridSearchCV; Risiko Kehamilan; Machine Learning; Standard Scaling

Abstract—This study aims to optimize the K-Nearest Neighbors (KNN) algorithm in predicting pregnancy risk levels using the “maternal health risk” dataset from the UCI Machine Learning Repository. The methodology includes data preprocessing through outlier detection and removal using Z-score, normalization with Standard Scaling, and categorical encoding on the target labels. Hyperparameter tuning is performed using GridSearchCV to identify the optimal combination of KNN parameters (number of neighbors, distance weight, and distance metric). The results show that the unoptimized KNN model achieved an accuracy of only 69.46%, whereas the optimized model reached an accuracy of 82.00%, with macro average precision of 81.91%, recall of 82.89%, and F1-score of 82.23%. Evaluation using a confusion matrix also revealed significant performance improvement, especially in the high-risk category. The optimized model was deployed as a web application using the Flask framework and Docker via Hugging Face Spaces, enabling real-time and efficient online pregnancy prediction. These findings indicate that combining KNN with GridSearchCV and data normalization significantly enhances prediction performance and offers practical application in healthcare decision support systems.

Keywords: K-Nearest Neighbors; GridSearchCV; Pregnancy Risk; Machine Learning; Standard Scaling

1. PENDAHULUAN

Kehamilan merupakan suatu proses alami yang dialami oleh manusia, terutama wanita, untuk meneruskan keturunan, di mana janin berkembang dan tumbuh di dalam rahim. Selama periode ini, tubuh sang ibu mengalami berbagai perubahan fisik dan mental demi mendukung kesehatan ibu dan janin sampai waktu kelahiran [1]. Kehamilan dengan risiko tinggi merupakan kondisi yang dapat memengaruhi kesehatan ibu maupun janin apabila ditangani dengan prosedur yang sama seperti kehamilan normal. Risiko tinggi atau komplikasi kebidanan adalah penyimpangan dari kondisi normal yang secara langsung dapat meningkatkan kemungkinan terjadinya kesakitan maupun kematian pada ibu dan bayi [2].

Secara global, rasio kematian ibu menurun namun risiko kematian ibu jauh lebih tinggi di negara miskin (1 dari 45 wanita) dibanding negara maju (1 dari 5.400 wanita) [3]. Pernyataan ini didukung oleh *Journal of Global Health* didapatkan data tingkat kematian saat kehamilan seperti pendarahan obstetrik seperti pendarahan selama atau setelah persalinan (28,8% dari kematian), gangguan hipertensi dalam kehamilan (22,1% dari kematian), komplikasi non-obstetrik meliputi penyakit yang tidak terkait langsung dengan kehamilan seperti penyakit jantung sebesar (18,8% dari kematian), yang mana data ini didapat dari daerah sub-sahara Afrika [4]. Untuk membantu dalam menurunkan angka risiko global kematian dalam kehamilan terutama di wilayah negara miskin diperlukan adanya solusi untuk mendiagnosis keadaan ibu selama mengandung, maka dari itu pemanfaatan teknologi seperti pembelajaran mesin (*machine learning*) hendaknya bisa menjadi solusi yang inovatif dan relevan dalam permasalahan ini [5].

Teknik penambangan data (*data mining*) bersama dengan pembelajaran mesin (*machine learning*) telah banyak digunakan sebagai pendukung pengambilan keputusan dalam bidang kesehatan, khususnya dalam menganalisis data besar, memprediksi dan mengklasifikasi risiko penyakit, serta meningkatkan akurasi diagnosis dan perencanaan perawatan pasien [6]. Beberapa metode dalam penambangan data (*data mining*) seperti *preprocessing* data, digunakan untuk melakukan pembersihan data dari nilai yang tidak valid, duplikat, dan data yang tidak relevan, serta transformasi data agar memiliki format yang seragam, reduksi data untuk menghilangkan atribut atau data yang kurang penting, dan *splitting* data menjadi bagian pelatihan dan pengujian. Semua tahapan ini bertujuan untuk meningkatkan akurasi, performa, dan efisiensi model dalam proses analisis dan prediksi [7]. Begitu juga dalam pembelajaran mesin (*machine*

learning), terdapat banyak metode yang dapat diterapkan dalam bidang medis dan kesehatan untuk melakukan analisis data secara mendalam maupun prediksi hasil yang akurat [8]. Cara-cara ini dapat mengidentifikasi pola yang rumit dalam data kesehatan, seperti hasil uji laboratorium, gambar medis, dan rekam kesehatan elektronik, sehingga mendukung profesional kesehatan dalam menentukan diagnosis, terapi, serta membuat keputusan klinis dengan lebih akurat dan efektif [9], [10].

Pada pembelajaran mesin (*machine learning*) terdapat metode pembelajaran tidak diawasi (*unsupervised learning*) dan pembelajaran diawasi (*supervised learning*) [11]. *Supervised learning* seperti contoh *Logistic Regression*, *Decision Trees*, *Random Forest*, *Support Vector Machine (SVM)*, *Naive Bayes*, dan *K-Nearest Neighbors (KNN)*, sedangkan *unsupervised learning* *K-Means Clustering*, *DBSCAN*, dan *Principal Component Analysis (PCA)* [12]. Pada penelitian ini menerapkan *supervised learning* sebagai metode yang digunakan untuk pemodelan *machine learning*, dikarenakan data yang terpilih memiliki label sebagai referensi selama proses pelatihan, untuk model dapat belajar dari contoh-contoh dengan hasil yang sudah diketahui [13].

Penelitian ini mengambil dataset dari *UCI Machine Learning Repository* yaitu “*maternal health risk*”, dataset ini terdiri dari 1013 data dan mencakup 6 fitur dan 1 target klasifikasi yaitu risiko kehamilan yang dibagi menjadi 3 jenis tingkatan kategorikal tertera yaitu *low risk*, *med risk*, *high risk* [14]. Beberapa penelitian terdahulu telah membahas dataset ini dengan membangun model klasifikasi untuk memprediksi risiko saat kehamilan seperti [15] [16], [17] diperoleh akurasi dari algoritma *K-Nearest Neighbors (KNN)* lebih rendah sebesar <70% jika dibandingkan dengan algoritma lain seperti *Light Gradient Boosting* dengan akurasi rata-rata >80%.

Beberapa metode yang bisa dilakukan dalam mengoptimalkan *machine learning* seperti melakukan *hyperparameter tuning* dimana proses eksperimen dilakukan untuk menemukan kombinasi optimal parameter eksternal dalam model *machine learning* yang tidak dipelajari selama pelatihan, tetapi diatur sebelumnya untuk mengontrol proses pembelajaran agar mendapatkan akurasi seakurat mungkin [18]. Selain itu, untuk meningkatkan performa model juga menerapkan *Z-Score* yang mampu menjaga distribusi data sekaligus lebih efektif dalam mengatasi nilai ekstrem (*outlier*). Begitu juga *Standard Scaler* yang berfungsi menyetarakan skala fitur sehingga data memiliki rata-rata mendekati nol dan standar deviasi mendekati satu [18]. *Z-score* juga dapat digunakan untuk menghilangkan pengaruh posisi dan skala pada titik data, sehingga memungkinkan berbagai himpunan data yang berbeda dapat dibandingkan secara langsung [19].

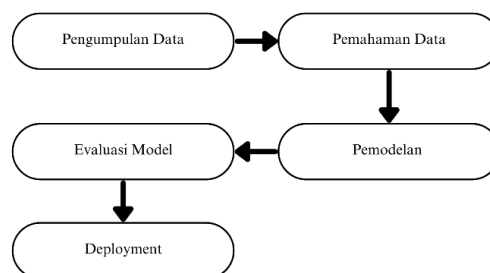
Beberapa penelitian terdahulu yang telah membahas tentang *hyperparameter tuning* menunjukkan bahwa algoritma *K-Nearest Neighbors (KNN)* berhasil mendapatkan akurasi dengan *gridsearch* meningkat 4% dari 91% menjadi 95%, sementara akurasi model *SVM* meningkat 11% dari 83% menjadi 94% [20]. *GridsearchCV* juga meningkatkan akurasi secara signifikan model *KNN* untuk memprediksi penyakit diabetes berhasil mencapai akurasi 88%, ini lebih tinggi jika dibandingkan tanpa menggunakan *gridsearch* sebesar 72% meningkat sebesar 16% [21]. Penelitian sebelumnya memperlihatkan bahwa metode ini memiliki performa yang lebih unggul dibandingkan teknik serupa.

Penelitian ini bertujuan mengoptimalkan algoritma *K-Nearest Neighbors (KNN)* dengan bantuan *GridSearchCV* serta *preprocessing* data menggunakan *Z-Score* untuk menangani *outlier* dan *Standard Scaler* untuk penyetaraan fitur. Pendekatan ini diharapkan menghasilkan prediksi risiko kehamilan yang lebih akurat dan efisien.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Penelitian ini melakukan klasifikasi kepada ibu hamil apakah memiliki indikasi pada kehamilannya. Pada hasil penelitian terdahulu dan tinjauan pustaka yang telah dilakukan algoritma *K-Nearest Neighbors (KNN)* yang digunakan dalam klasifikasi identifikasi, dengan memanfaatkan *GridSearchCV* untuk melakukan optimasi parameter agar mendapatkan tingkat akurasi yang terbaik. Penelitian ini melalui beberapa langkah seperti Gambar 1. untuk menyelesaikan masalah dan mencapai tujuan penelitian.



Gambar. 1 Alur Penelitian

2.2 Dataset

Pada penelitian ini dataset didapat dari *UCI Machine Learning Repository (Maternal Health Risk - UCI Machine Learning Repository)*. Data orisinal didapatkan melalui berbagai rumah sakit, klinik Kesehatan tentang kasus-kasus

komplikasi risiko kehamilan dari daerah pedesaan Bangladesh menggunakan perangkat berbasis *internet of things* (IoT) yang diinisiasi oleh universitas internasional Daffodil, Dhaka, Bangladesh [22]. Dataset ini memiliki 6 variabel yang digunakan sebagai atribut predictor (*Age, SystolicBP, DiastolicBP, BS, BodyTemp, HeartRate*) dan 1 variabel target (*RiskLevel*) [14]. Informasi dataset dapat dilihat pada Tabel 1, seperti berikut.

Tabel 1. Informasi Dataset

No	Atribut	Deskripsi
1	Age	Usia saat ibu mengandung
2	SystolicBP	Nilai tekanan darah atas diukur dalam satuan mmHg
3	DiastolicBP	Nilai tekanan darah atas diukur dalam satuan mmHg
4	BS	Kadar gula dalam darah dengan satuan molar
5	BodyTemp	Suhu badan dalam satuan fahrenheit
6	HeartRate	Jumlah detak jantung dalam satu menit
7	RiskLevel	Prediksi level resiko dibagi dalam tiga level (<i>low, mid, high</i>)

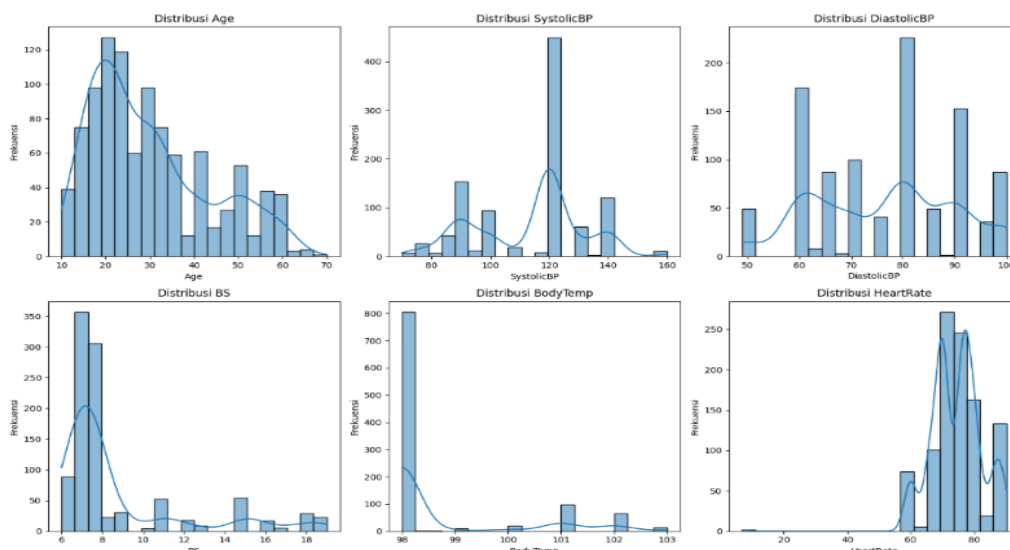
2.3 Exploratory Data Analysis (EDA)

Setelah data penelitian didapatkan dilanjutkan dengan *Exploratory Data Analysis* (EDA), merupakan proses penting untuk mendalami data dengan tujuan menemukan informasi seperti pola dalam data, *outlier*, serta meninjau asumsi pada hipotesis awal baik secara statistik maupun visual [23], metode ini dimanfaatkan guna menambah pemahaman yang lebih baik tentang wawasan dari data. Pada tabel 2 menunjukkan detail statistik data seperti *percentile, mean, std* dan lainnya dari dataset.

Tabel 2. Detail Statistik

	Age	SystolicBP	DiastolicBP	BS	BodyTemp	HeartRate
Count	1014	1014	1014	1014	1014	1014
Mean	29.8923	113.0813	76.1989	8.7262	98.6137	74.3914
Std	13.4495	18.7521	14.1031	3.3215	1.2942	7.5427
Min	10	70	49	6	98	7
25%	19	100	65	6.9	98	70
50%	26	120	80	7.5	98	76
75%	39	120	90	8	98	80
Max	70	160	100	19	103	90

Gambar 2 menampilkan histogram dari setiap fitur pada dataset. Kolom Age menunjukkan sebaran usia ibu hamil yang didominasi pada usia 20 tahun. Fitur SystolicBP dan DiastolicBP merepresentasikan tekanan darah atas dan bawah pasien, sedangkan BS (Blood Sugar) menunjukkan kadar gula darah. Fitur BodyTemp menggambarkan sebaran suhu tubuh pasien dalam satuan Fahrenheit, dan HeartRate menunjukkan jumlah detak jantung per menit.

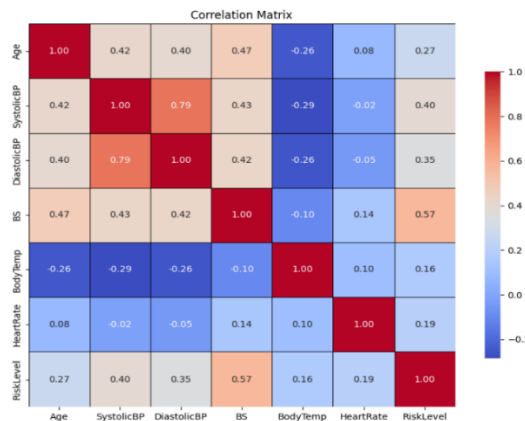


Gambar. 2 Visualisasi Tiap Fitur Dari Dataset Menggunakan Histogram

Dari Gambar 2 distribusi variabel dalam dataset menunjukkan bahwa fitur seperti usia distribusi data cenderung *skewer* ke kiri dimana didominasi usia dibawah 40 tahun, selanjutnya distribusi tekanan darah sistolik dan diastolik cenderung normal dengan data paling banyak berurutan pada 120 dan 80. Untuk *blood preasure* (BS) distribusi memperlihatkan mayoritas data berada pada nilai rendah hingga sedang antara 6 sampai 10, begitu pula distribusi pada

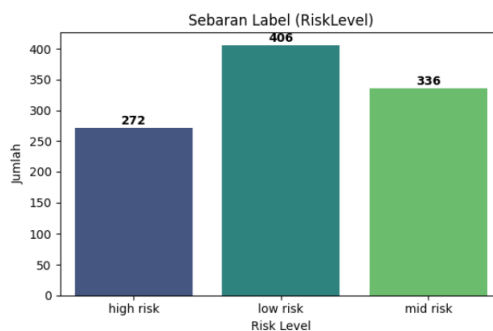
suhu tubuh (*BodyTemp*) mayoritas data berada pada suhu tubuh rendah pada 98 derajat fahrenheit. Distribusi detak jantung (*Heart Rate*) memiliki pola yang mengelompok pada rentang 60–80 dan terdapat outlier dibawah 20.

Selanjutnya dibuat correlation matrix untuk menggambarkan hubungan korelasi antar variable seperti pada gambar 3. Dari visualisasi diperoleh temuan seperti korelasi positif *SystolicBP* dan *DiastolicBP* yang menunjukkan bahwa kenaikan tekanan darah sistolik umumnya diikuti oleh kenaikan tekanan darah diastolik dengan nilai sebesar 0.79. Selain itu BS (*Blood Sugar*) memiliki korelasi paling tinggi terhadap *RiskLevel* sebesar 0.57 jika dibandingkan fitur lainnya, *SystolicBP* 0.40 dan *DiastolicBP* sebesar 0.35. Hal tersebut menunjukkan bahwa peningkatan kadar gula darah berkaitan erat dengan peningkatan tingkat risiko.



Gambar. 3 Heatmap Hubungan Korelasi Antar Variable

Kemudian proses EDA dilanjutkan untuk mengetahui sebaran data seperti pada Gambar 4, label tersebar merata antar kelas level resiko dan tidak terjadi ketidakseimbangan ekstrem dengan minimum 1:3, yaitu kelas high risk 272, low risk 406, dan mid risk 336, sehingga tidak dilakukan proses resampling seperti oversampling dan undersampling dengan alasan untuk menghindari overfitting terhadap pola buatan dan gagal menggeneralisasi data baru, serta menghindari distorsi distribusi asli yang membuat model yang dihasilkan tidak mencerminkan kondisi realitas lapangan.



Gambar. 4 Sebaran Label Level Risiko Kehamilan Dalam Dataset

2.4 Modeling

Pada proses *modelling* dilakukan beberapa tahapan untuk membangun model pada penelitian ini seperti *preprocessing*, dalam *machine learning preprocessing* dilakukan untuk mempersiapkan dan membersihkan dataset yang berupa data mentah agar menjadi lebih berkualitas dan sesuai untuk digunakan dalam proses pelatihan model [24]. Dalam penelitian ini dilakukan beberapa tahapan *preprocessing* berdasarkan hasil dari EDA, tahapan ini bertujuan agar data yang diperoleh memiliki format yang konsisten, dan dapat meningkatkan efektivitas serta efisiensi selama proses pemodelan *machine learning* berlangsung.

Tahapan awal dilakukan *data cleaning* untuk memastikan dataset yang digunakan terbebas dari ketidaksesuaian yang mana akan mempengaruhi hasil prediksi [25]. Pada penelitian ini, Tahap pembersihan data dimulai dengan pengecekan apakah terdapat nilai yang tidak lengkap atau kosong (*missing value*) dalam dataset.

Setelah dilakukan *data cleaning*, data dianalisis untuk mendeteksi *outlier*. Nilai yang terdeteksi sebagai *outlier* kemudian dihapus. Pada penelitian ini menggunakan *Z-Score* dengan batas yang digunakan sebesar $|Z| > 3.15$ guna mendeteksi outlier yang mungkin mengganggu akurasi model agar dapat meningkatkan ketepatan klasifikasi model [26]. Metode *Z-Score* dipilih karena mampu mendeteksi nilai ekstrem (*outlier*) secara efektif dengan mengukur jarak data terhadap rata-rata dalam satuan standar deviasi, dibandingkan metode Min-Max, yang membatasi nilai data pada rentang tertentu [27].

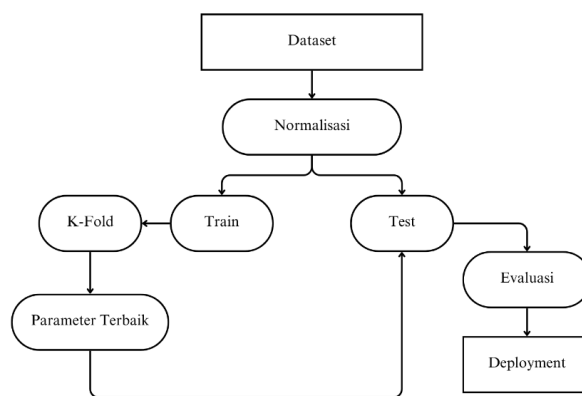
Tahapan selanjutnya, melibatkan proses *scaling* data menggunakan metode *Standard Scaling* yang secara matematis identik dengan *Z-Score* untuk menyesuaikan rentang nilai fitur numerik agar skala antar fitur menjadi seragam tanpa kehilangan informasi penting yang dapat mempengaruhi akurasi model [28]. Metode ini dipilih karena kemampuannya dalam mengubah skala data tanpa mengubah distribusi asli, sehingga setiap variabel memiliki bobot yang seimbang dalam proses pelatihan model [29]. Karena metode normalisasi berbasis *Z-Score*, yang secara matematis identik dengan proses *Standard Scaling*. Kedua metode ini menggunakan rumus yang sama seperti pada persamaan (1).

$$Z = \frac{x - \mu}{\sigma} \tag{1}$$

Dimana menghitung seberapa jauh suatu nilai data (x) menyimpang dari rata-rata (μ) dalam satuan standar deviasi (σ). Pendekatan ini tidak hanya menyetarakan skala antar fitur, tetapi juga efektif dalam mendeteksi nilai ekstrem (outlier).

Pemrosesan dilanjutkan dengan *encoding*, metode yang diterapkan adalah *label encoding* pada target untuk mengonversi data kategori berupa teks menjadi format numerik, sehingga dapat diproses oleh algoritma *machine learning* yang hanya menerima data dalam bentuk angka. *Label encoding* dipilih karena caranya yang sederhana, yaitu menggantikan setiap label kategori dengan angka [30].

Data yang telah melalui proses normalisasi selanjutnya dilakukan pembagian dataset menjadi data pengujian dan data pelatihan. Teknik split data merupakan metode untuk membagi dataset menjadi beberapa bagian dan termasuk salah satu faktor penting yang memengaruhi performa model klasifikasi dalam algoritma pembelajaran mesin [31]. Pada tahap ini, data dipisahkan dengan perbandingan 80:20, dimana 80% dialokasikan untuk melatih model, sedangkan 20% digunakan untuk menguji performa model. Berikut pada Gambar 5. diilustrasikan bagaimana proses *modelling* dilaksanakan.



Gambar. 5 Alur Penelitian

2.5 K-Nearest Neighbors (KNN)

Pada penelitian ini algoritma K-Nearest Neighbors (KNN) diterapkan dalam proses pembangunan model prediksi *maternal health risk*. Algoritma ini merupakan metode klasifikasi yang memanfaatkan jarak antar titik data untuk memperkirakan label pada data yang belum diketahui [32]. Dalam KNN, label untuk data baru ditentukan berdasarkan mayoritas label dari “k” tetangga terdekatnya [32]. Untuk menghitung jarak antar data baru KNN menggunakan metrik penghitung jarak tertentu seperti *manhattan*, *cosine*, atau *euclidean* setelahnya algoritma menentukan “k” tetangga terdekat berdasar jarak yang dihitung, kemudian data baru diklasifikasikan dengan menentukan label yang paling banyak muncul di antara tetangga terdekatnya [33]. Rumus jarak metode KNN yang diujikan dapat dilihat dari persamaan (3), (4), (5) secara berurut cosine, manhattan, euclidean, juga menggunakan pembobotan jarak *Distance* sebagaimana dirumuskan pada persamaan (6) untuk meningkatkan akurasi.

$$\text{Cosine Distance}(A, B) = 1 - \cos(\theta) = \frac{A \cdot B}{\|A\| \cdot \|B\|} \tag{3}$$

Cosine digunakan untuk mengukur tingkat kesamaan arah antara dua vektor data A dan B dalam ruang berdimensi n . Nilai kesamaan dihitung menggunakan rasio antara *dot product* ($A \cdot B$) dengan hasil kali panjang masing-masing vektor ($\|A\| \cdot \|B\|$), sehingga diperoleh nilai $\cos(\theta)$ yang berada pada rentang -1 hingga 1. Semakin mendekati 1, semakin mirip arah kedua vektor tersebut.

$$\text{Manhattan Distance}(A, B) = \sum_{i=1}^n |A_i - B_i| \tag{4}$$

Sementara itu, Manhattan Distance atau *taxicab distance* mengukur jarak absolut antar titik dengan menjumlahkan perbedaan nilai pada setiap dimensi, yaitu $\sum |A_i - B_i|$. Metode ini menekankan total perbedaan linier antar dimensi dan sering digunakan ketika distribusi data bersifat grid atau berbasis nilai absolut.

$$\text{Euclidean Distance}(A, B) = \sqrt{\sum_{i=1}^n (A_i - B_i)^2} \quad (5)$$

Euclidean Distance mengukur jarak lurus (*straight-line distance*) antar titik dalam ruang multidimensi. Nilai dihitung dengan akar kuadrat dari jumlah kuadrat selisih tiap dimensi, yaitu $\sqrt{\sum (A_i - B_i)^2}$. Ukuran ini umum digunakan sebagai representasi jarak geometris yang paling intuitif karena menggambarkan kedekatan spasial antar data.

$$w_i = \frac{1}{d_i + \epsilon} \quad (6)$$

Pembobotan jarak pada algoritma KNN memberikan pengaruh lebih besar pada tetangga terdekat dengan menentukan bobot (w_i) sebagai kebalikan dari jarak (d_i), ditambah konstanta kecil (ϵ) untuk menghindari pembagian nol, sehingga hasil klasifikasi menjadi lebih akurat.

2.6 GridSearchCV

Untuk mengimprovisasi algoritma *K-Nearest Neighbors* (KNN) dilakukan *Hyperparameter Tuning* yang menerapkan teknik *gridsearchCV*, dengan maksud untuk mengidentifikasi parameter-parameter yang digunakan dalam model sehingga menghasilkan performa yang paling optimal [34]. Proses ini diawali dengan menentukan kumpulan nilai untuk setiap *hyperparameter*, seperti jumlah tetangga (k) {3, 5, 7, 9, 11, ..., 21}, bobot jarak (*weights*) {*distance*}, dan metode jarak (*metric*) {euclidean, manhattan, cosine}. *GridSearchCV* melakukan pengujian terhadap seluruh kombinasi *hyperparameter* yang tersedia dengan menggunakan metode validasi silang untuk menilai performa model. *Hyperparameter* yang memberikan hasil terbaik inilah kemudian dipilih guna meningkatkan akurasi model secara keseluruhan.

Proses GridSearchCV kemudian mengikuti beberapa langkah utama. Pertama, seluruh kombinasi nilai dari parameter-parameter tersebut disusun dalam bentuk grid. Selanjutnya, untuk setiap kombinasi, dilakukan proses *k-fold cross-validation* guna menilai stabilitas dan performa model secara menyeluruh. Proses ini melibatkan pembagian data latih menjadi beberapa lipatan, di mana model dilatih pada sebagian data dan diuji pada bagian sisanya secara bergiliran. Evaluasi dilakukan menggunakan metrik kinerja tertentu, seperti akurasi atau *F1-score*, dan hasil rata-ratanya digunakan sebagai dasar pemilihan.

2.7 Evaluasi

Metode evaluasi yang digunakan pada penelitian ini adalah *confusion matrix*, alat evaluasi ini memperlihatkan banyaknya prediksi yang tepat dan keliru untuk tiap kelas, sehingga memberikan gambaran yang lebih menyeluruh tentang kinerja model klasifikasi [35]. Parameter evaluasi yang digunakan meliputi seperti *accuracy*, *precision*, *recall*, dan *F1-score* seperti yang dirumuskan berikut.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (7)$$

$$\text{Precision}(i) = \frac{TP_i}{TP_i + FP_i} \quad (8)$$

$$\text{Recall}(i) = \frac{TP_i}{TP_i + FN_i} \quad (9)$$

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision}_i \cdot \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i} \quad (10)$$

2.7 Deployment

Pada tahapan *deployment* yang mana merupakan proses akhir setelah evaluasi model *machine learning*, dilakukan bertujuan agar model ini dapat diaplikasikan kedalam bentuk website dan dapat diakses siapapun. Alasan menggunakan website sebagai basis *deployment* seperti kemudahan akses darimana saja hanya dengan browser dan akses internet selain itu juga kemudahan dalam update dan pemeliharaan sistem jika sewaktu waktu dibutuhkan. Dari berapa pertimbangan ini aplikasi berbasis website cocok untuk dipilih dalam penelitian ini.

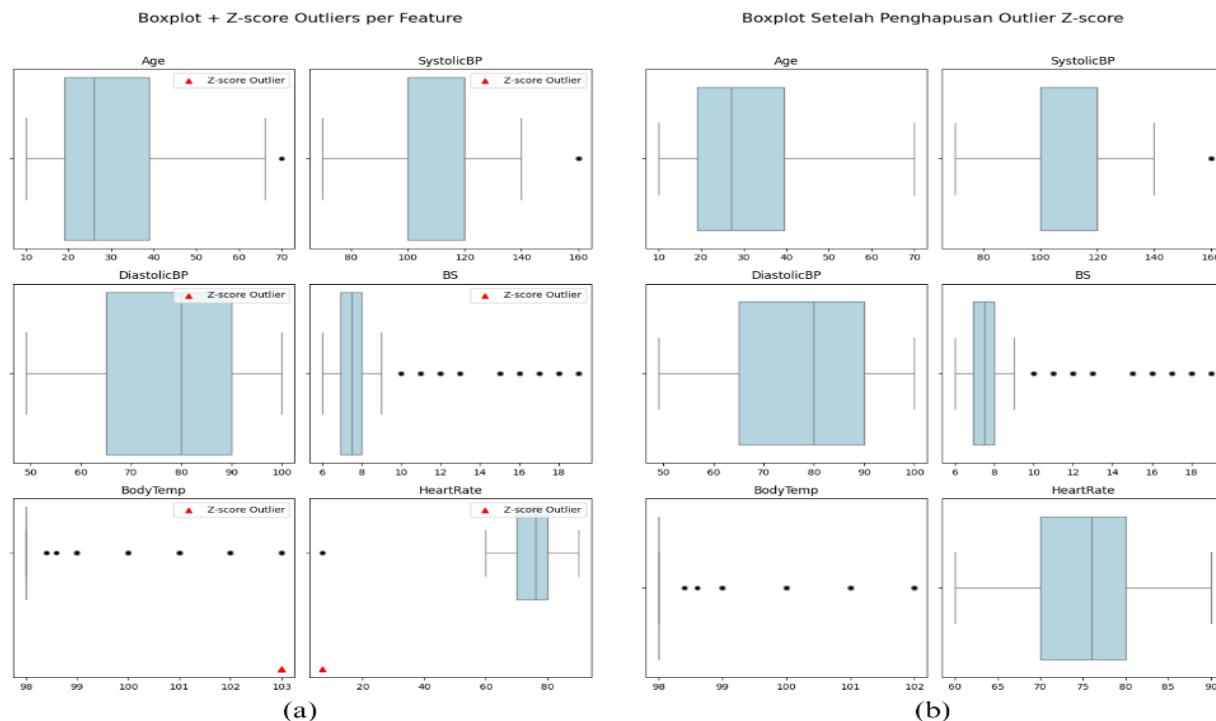
3. HASIL DAN PEMBAHASAN

3.1 Normalisasi

Langkah pertama yang dilakukan dalam proses normalisasi adalah penanganan *outlier* untuk itu dilakukan pengecekan nilai outlier. Selanjutnya penanganan *outlier* yang digunakan dalam penelitian ini adalah dengan cara menghapus data yang terdeteksi sebagai nilai yang *outlier*, cara ini dipilih untuk memudahkan dalam penanganan nilai ini. Penggunaan Z-Score untuk deteksi *outlier* didasari karena jika menggunakan deteksi boxplot IQR jumlah outlier yang didapatkan terlalu banyak dan jika dilanjutkan pemrosesan dengan menghapus outlier akan mengurangi jumlah data yang bisa digunakan selanjutnya. Dengan menggunakan metode IQR, total terdapat 394 baris data yang mengandung setidaknya satu nilai outlier, yang mencakup sekitar 38.86% dari keseluruhan data. Outlier terbanyak ditemukan pada fitur Blood Sugar (BS) dan Body Temperature, masing-masing sebanyak 210 kasus. Jika seluruh baris ini dihapus, proporsi data yang hilang akan cukup besar dan berpotensi mengganggu proses pelatihan model. Sebaliknya, deteksi menggunakan

metode Z-Score menunjukkan bahwa hanya terdapat 15 baris data yang mengandung outlier, atau sekitar 1.48% dari total data. Outlier paling banyak ditemukan pada fitur Body Temperature sebanyak 13 kasus, serta Heart Rate sebanyak 2 kasus. Fitur lainnya tidak mengandung outlier signifikan berdasarkan kriteria Z-Score.

Tahapan selanjutnya untuk memastikan bahwa proses penghapusan outlier berhasil, dilakukan visualisasi menggunakan boxplot. Data yang diklasifikasikan sebagai outlier oleh metode Z-Score ditandai menggunakan ikon segitiga berwarna merah, sebagai pembeda terhadap outlier yang terdeteksi oleh metode IQR. Visualisasi ini ditampilkan pada Gambar 6, yang memberikan gambaran mengenai distribusi data dan perbedaan hasil deteksi outlier dari kedua metode.



Gambar. 6 Boxplot Sebelum (a) dan Sesudah (b) Penghapusan Outlier

Proses *scaling* menggunakan *Standard Scaler* dilakukan hanya pada data latih tanpa melibatkan label prediksi. Tabel 3 menunjukkan bahwa setelah *scaling*, seluruh fitur memiliki nilai rata-rata yang mendekati nol dan standar deviasi yang mendekati satu, menandakan bahwa skala data telah seragam dan siap digunakan pada tahap pengujian, karena terbatas maka hanya empat digit dibelakang koma yang ditulis.

Tabel 3. Detail Data Setelah Dilakukan Scaling

	Age	SystolicBP	DiastolicBP	BS	BodyTemp	HeartRate
Count	799	799	799	799	799	799
Mean	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Std	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Min	-1.4799	-2.2988	-1.9297	-8.2130	-4.7452	-1.9092
25%	-8.1037	-9.6483	-7.9457	-5.5018	-4.7452	-5.8261
50%	-2.8958	3.6918	2.6968	-3.6942	-4.7452	2.1335
75%	6.0319	3.6918	9.7918	-2.1879	-4.7452	7.4399
Max	2.9839	2.5036	1.6886	3.0950	2.6180	2.0706

3.2 Modeling

Selanjutnya penelitian dimulai dengan melakukan proses pembagian dataset menjadi dua subset yakni *train* dan *test* dengan perbandingan yang sudah ditentukan sebesar 80:20, tindakan ini dengan tujuan setelah dilakukan pembelajaran mesin pada data *train* dilakukan pengujian menggunakan data *test* untuk mengevaluasi performa model memprediksi terhadap data baru.

Proses pelatihan model dilanjutkan dengan mencari parameter terbaik dari data *train* dengan menjalankan prosedur *hyperparameter tuning* menggunakan *gridsearch* dengan tujuan mengetahui parameter optimal dari pada algoritma KNN. Tahapan berikutnya Cross-validation digunakan untuk memastikan bahwa hyperparameter pada model *machine learning* yang dihasilkan benar-benar optimal dalam menangani data baru berdasarkan pada parameter terbaik yang telah dipilih dengan *gridsearch*. Parameter yang akan dicari pada KNN dengan *gridsearch* yaitu dapat dilihat sebagai berikut.

Tabel 4. Pengaturan Hyperparameter

Parameter	Setting
K	3-21
Distance	Cosine, Manhattan, Euclidean
K-CV	10

Dimana K adalah jumlah tetangga terdekat yang dipertimbangkan saat mengklasifikasikan data baru dengan settingan 3-21 dengan interval ganjil, distance adalah metrik yang digunakan untuk menghitung jarak antar titik data dengan pilihan cosine, manhattan, dan euclidean. Dan weight bobot distance tetangga yang lebih dekat diberi bobot lebih besar. Serta nilai K-Fold pada *GridSearchCV* sebesar 10 dimana data dibagi menjadi 10 bagian untuk dilakukan validasi silang 10 kali, selain itu *stratify* digunakan untuk memastikan proporsi kelas target di setiap *fold* konsisten dengan proporsi kelas dalam data asli, sehingga evaluasi model menjadi lebih representatif dan seimbang.

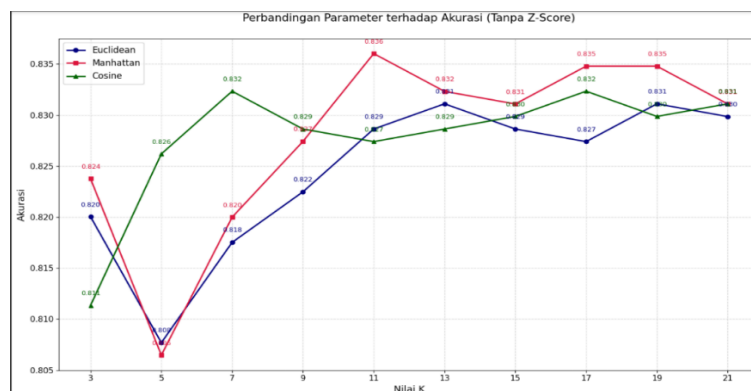
3.3 Hasil Pelatihan

Bagian ini membahas hasil pelatihan model berdasarkan setiap pengujian parameter yang dilakukan. Pembahasan difokuskan pada perbandingan hasil *GridSearch* yang diterapkan pada dataset yang telah dilakukan penghapusan *outlier* menggunakan Z-Score dengan dataset yang tidak menghapus *outlier*. Untuk tabel 5 didapatkan rerata akurasi dari hasil *GridSearch* pada dataset tanpa Z-Score didapatkan akurasi terendah dan tertinggi pada K = 5 dan K = 19 dengan akurasi berurutan 81.34% dan 83.19%.

Tabel 5. Akurasi Rata-Rata Berdasarkan Nilai K

K Neighbors	Average Accuracy
3	81,83%
5	81,34%
7	82,32%
9	82,61%
11	83,06%
13	83,00%
15	82,98%
17	83,15%
19	83,19%
21	83,06%

Sementara itu, Gambar 7 menyajikan komparasi performa dari masing-masing parameter jarak yakni *cosine*, *euclidean*, dan *manhattan* yang digunakan pada setiap nilai K ganjil mulai dari 3 hingga 21. Berdasarkan hasil tersebut, terlihat bahwa metrik *manhattan* menunjukkan performa yang lebih unggul mulai dari nilai K = 11 sampai 21 secara keseluruhan dibandingkan dua metrik lainnya.



Gambar. 7 Grafik Komparasi Parameter Tanpa Z-Score

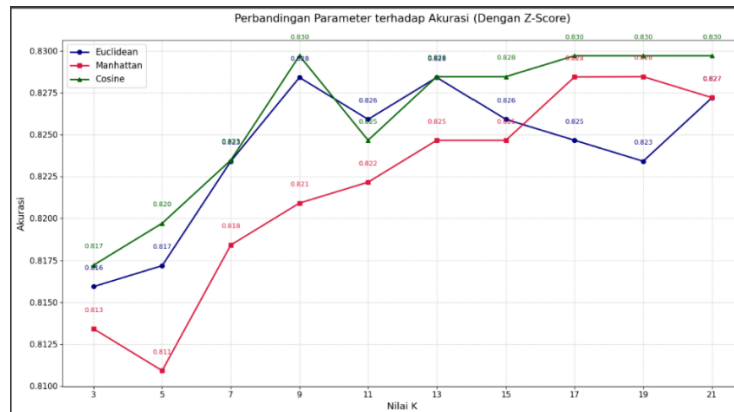
Selanjutnya, pada Tabel 6 ditampilkan hasil evaluasi pada dataset yang menggunakan Z-Score. Melalui proses *GridSearch*, diperoleh akurasi terendah sebesar 81,55% pada K = 3, dan akurasi tertinggi sebesar 82,80% pada K = 21. Hasil ini menunjukkan adanya peningkatan akurasi seiring dengan bertambahnya nilai K dalam rentang yang diuji.

Tabel 6. Akurasi Rata-Rata Berdasarkan Nilai K dengan Z-Score

K Neighbors	Average Accuracy
3	81,55%
5	81,59%
7	82,18%

K Neighbors	Average Accuracy
9	82,63%
11	82,43%
13	82,72%
15	82,64%
17	82,76%
19	82,72%
21	82,80%

Sementara itu, Gambar 8 menyajikan komparasi performa dari masing-masing parameter jarak *cosine*, *euclidean*, dan *manhattan* yang digunakan pada setiap nilai K ganjil dari 3 hingga 21. Secara keseluruhan, metrik *cosine* menunjukkan performa yang paling unggul, khususnya pada nilai K = 9 serta K = 15 hingga K = 21.

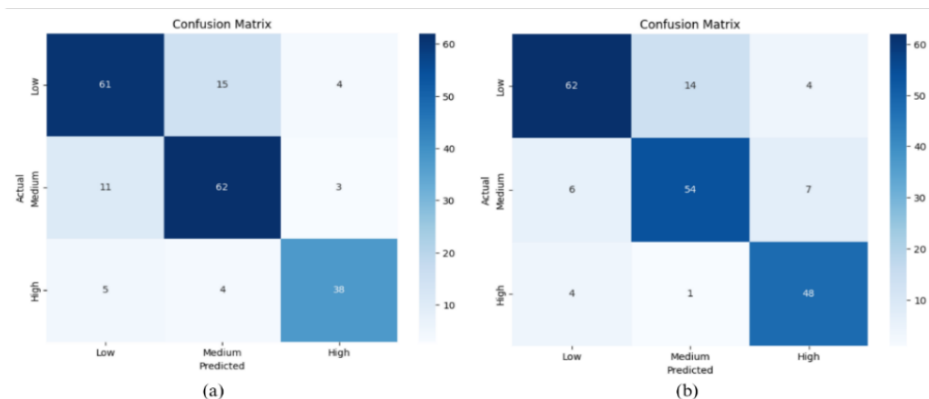


Gambar. 8 Grafik Komparasi Parameter dengan Z-Score

Setelah dilakukan proses GridSearch cross-validation, diperoleh parameter terbaik untuk masing-masing dataset. Pada dataset tanpa Z-Score, parameter optimal adalah ($k=11$, $metric=manhattan$), sedangkan pada dataset dengan Z-Score, parameter terbaik adalah ($k=9$, $metric=cosine$). Hasil ini menunjukkan bahwa Manhattan lebih sesuai digunakan pada data mentah karena mampu mempertahankan makna skala asli tiap fitur, sementara Cosine lebih optimal pada data yang telah dinormalisasi dengan Z-Score karena dapat merepresentasikan pola relatif antar fitur secara lebih seimbang.

3.4 Evaluasi

Evaluasi model dimulai setelah proses *train* selesai dilanjutkan dengan menguji model menggunakan data baru yang berasal dari dataset *test*, dari sinilah dapat diamati bagaimana reaksi model terhadap data baru yang belum pernah dilihat sebelumnya. Gambar 9 (a) menunjukkan *confusion matrix* hasil prediksi model KNN pada dataset tanpa Z-Score dengan parameter terbaik ($K=11$, $metric=manhattan$). Model mampu mengklasifikasikan kelas *Low* dan *Medium* dengan cukup baik, namun akurasi pada kelas *High* masih relatif rendah, ditandai dengan jumlah kesalahan klasifikasi yang lebih tinggi ke kelas *Low* maupun *Medium*.



Gambar. 9 Confusion Matrix dari (a) Tanpa Z-Score dan (b) dengan Z-Score

Sementara itu, Gambar 9 (b) menunjukkan confusion matrix model KNN pada dataset dengan Z-Score ($K=9$, $metric=cosine$). Terlihat peningkatan akurasi signifikan pada kelas *High* dan penurunan kesalahan klasifikasi, meskipun akurasi kelas *Medium* sedikit menurun. Secara keseluruhan, Z-Score menghasilkan distribusi prediksi yang lebih seimbang antar kelas.

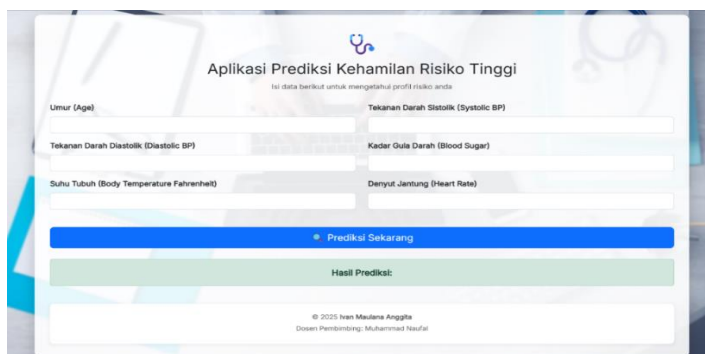
Selanjutnya pada Tabel 7 menunjukkan perbandingan hasil evaluasi model *K-Nearest Neighbors* (KNN) sebelum dan sesudah dilakukan proses optimasi. Model KNN dasar sebagaimana dikutip dari [17] menghasilkan akurasi sebesar 66,50%, dengan nilai presisi 67,09% dan recall 66,50%. Setelah dilakukan optimasi *hyperparameter* menggunakan *GridSearchCV*, terjadi peningkatan performa yang signifikan, ditunjukkan oleh akurasi yang meningkat menjadi 79.31%, presisi 80.07%, dan recall 79.56%. Lebih lanjut, dengan melalui proses penghapusan outlier menggunakan *Z-Score* pada data, performa model meningkat lebih lanjut dengan akurasi mencapai 82,00%, presisi 81,91%, dan recall 82,23%. Hasil ini menunjukkan bahwa kombinasi *hyperparameter* tuning dan *Z-Score* berkontribusi besar dalam meningkatkan kinerja model KNN secara keseluruhan.

Tabel 7. Tabel Evaluasi KNN Optimisasi

Algoritma	Akurasi	Presisi	Recall
KNN [17]	66.50	67.09	66.50
KNN + Hyperparameter	79.31	80.07	79.56
KNN + Hyperparameter dan Z-Score	82.00	81.91	82.23

3.5 Deployment

Tahap deployment menghasilkan prototype aplikasi yang dapat diakses melalui website. Diharapkan dapat memberikan manfaat berupa kemudahan, efisiensi, dan kecepatan dalam melakukan prediksi tingkat risiko kehamilan, sehingga dapat mendukung tenaga kesehatan dalam proses pengambilan keputusan berbasis data. Dengan pendekatan ini, aplikasi prediksi dapat berjalan secara online dan diakses publik melalui URL https://risetkami.my.id/mahasiswa/mhr_kehamilan.html [36], seperti ditunjukkan pada Gambar 9.



Gambar. 9 Tampilan Utama Halaman

4. KESIMPULAN

Penelitian ini membuktikan bahwa algoritma *K-Nearest Neighbors* (KNN) yang dipadukan dengan normalisasi data serta *hyperparameter* tuning menggunakan *GridSearchCV* mampu meningkatkan akurasi prediksi pada dataset *Maternal Health Risk* dalam mengklasifikasikan tingkat risiko kehamilan. Tahapan penelitian meliputi *Exploratory Data Analysis* (EDA), normalisasi (*Z-Score* dan *Standard Scaling*), serta label encoding yang mempersiapkan dataset secara optimal sebelum pelatihan model. Hasil evaluasi menunjukkan bahwa KNN tanpa optimasi hanya mencapai akurasi 69,46%, sedangkan dengan *hyperparameter* akurasi meningkat menjadi 79.31% dan bisa ditingkatkan lebih lanjut dengan *Z-Score* hingga mendapatkan akurasi 82.00% dengan distribusi prediksi antar kelas menjadi lebih seimbang, sebagaimana ditunjukkan pada confusion matrix. Implementasi prototype model ke dalam website dapat memberikan manfaat berupa kemudahan akses, efisiensi, dan kecepatan dalam melakukan prediksi tingkat risiko kehamilan, sehingga dapat mendukung tenaga kesehatan dalam pengambilan keputusan yang lebih tepat berbasis data. Meskipun demikian, penelitian ini memiliki beberapa keterbatasan, di antaranya ukuran dataset yang relatif kecil serta fokus yang hanya pada algoritma KNN tanpa adanya perbandingan dengan algoritma lain. Oleh karena itu, penelitian selanjutnya disarankan untuk menggunakan dataset yang lebih besar dan beragam, melakukan perbandingan dengan algoritma lain seperti *Random Forest* atau *XGBoost*.

REFERENCES

- [1] A. Raza, H. U. R. Siddiqui, K. Munir, M. Almutairi, F. Rustam, and I. Ashraf, "Ensemble learning-based feature engineering to analyze maternal health during pregnancy and health risk prediction," *journals.plos.org*, vol. 17, no. 11 November, Nov. 2022, doi: 10.1371/JOURNAL.PONE.0276525.
- [2] A. Bayuana *et al.*, "Komplikasi Pada Kehamilan, Persalinan, Nifas dan Bayi Baru Lahir: Literature Review," *Jurnal Wacana Kesehatan*, vol. 8, no. 1, p. 26, Jul. 2023, doi: 10.52822/jwk.v8i1.517.
- [3] A. Kurjak, M. Stanojević, and J. Dudenhausen, "Why maternal mortality in the world remains tragedy in low-income countries and shame for high-income ones: Will sustainable development goals (SDG) help?," *J Perinat Med*, vol. 51, no. 2, pp. 170–181, Feb. 2023, doi: 10.1515/jpm-2022-0061.

- [4] R. Musarandega *et al.*, “Causes of maternal mortality in Sub Saharan Africa: A systematic review of studies published from 2015 to 2020,” *J Glob Health*, Oct, 2021, doi: 10.7189/jogh.11.04048.
- [5] R. G. Wardhana, G. Wang, and F. Sibuea, “Penerapan Machine Learning Dalam Prediksi Tingkat Kasus Penyakit Di Indonesia,” *Journal of Information System Management (JOISM)*, vol. 5, no. 1, pp. 40–45, Jul. 2023, doi: 10.24076/JOISM.2023V5I1.1136.
- [6] S. M. D. A. C. Jayatilake and G. U. Ganegoda, “Involvement of Machine Learning Tools in Healthcare Decision Making,” *J Healthc Eng*, vol. 2021, no. 1, p. 6679512, Jan. 2021, doi: 10.1155/2021/6679512.
- [7] M. Rijal *et al.*, “Prediksi Depresi: Inovasi Terkini Dalam Kesehatan Mental Melalui Metode Machine Learning,” *Journal Pharmacy and Application of Computer Sciences*, vol. 2, no. 1, pp. 9–14, Feb. 2024, doi: 10.59823/JOPACS.V2I1.47.
- [8] M. R. S. Rao, D. Yadav, and V. Anbarasu, “An Improvised Machine Learning Model KNN for Malware Detection and Classification,” *2023 International Conference on Computer Communication and Informatics, ICCCI 2023*, 2023, doi: 10.1109/ICCCI56745.2023.10128189.
- [9] I. Mayla Faiza, Gunawan, and W. Andriani, “Tinjauan Pustaka Sistematis: Penerapan Metode Machine Learning untuk Deteksi Bencana Banjir,” *Jurnal Minfo Polgan*, vol. 11, no. 2, pp. 59–63, Aug. 2022, doi: 10.33395/JMP.V11I2.11657.
- [10] L. Rubinger, A. Gazendam, S. Ekhtiari, and M. Bhandari, “Machine learning and artificial intelligence in research and healthcare,” *Injury*, vol. 54, pp. S69–S73, May 2023, doi: 10.1016/J.INJURY.2022.01.046.
- [11] Y. Chen, M. Mancini, X. Zhu, and Z. Akata, “Semi-Supervised and Unsupervised Deep Visual Learning: A Survey,” *IEEE Trans Pattern Anal Mach Intell*, vol. 46, no. 3, pp. 1327–1347, Mar. 2024, doi: 10.1109/TPAMI.2022.3201576.
- [12] R. S. Nurhalizah, R. Ardianto, and P. Purwono, “Analisis Supervised dan Unsupervised Learning pada Machine Learning: Systematic Literature Review,” *Jurnal Ilmu Komputer dan Informatika*, vol. 4, no. 1, pp. 61–72, Aug. 2024, doi: 10.54082/JIKI.168.
- [13] N. L. P. C. Savitri, R. A. Rahman, R. Venyutzky, and N. A. Rakhmawati, “Analisis Klasifikasi Sentimen Terhadap Sekolah Daring pada Twitter Menggunakan Supervised Machine Learning,” *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 7, no. 1, pp. 2443–2229, Apr. 2021, doi: 10.28932/JUTISI.V7I1.3216.
- [14] “Maternal Health Risk - UCI Machine Learning Repository.” Accessed: Jun. 12, 2025. [Online]. Available: <https://archive.ics.uci.edu/dataset/863/maternal+health+risk>
- [15] H. B. Mutlu, F. Durmaz, N. Yücel, E. Cengil, and M. Yıldırım, “Prediction of Maternal Health Risk with Traditional Machine Learning Methods,” *NATURENGS*, vol. 4, no. 1, pp. 16–23, Jun. 2023, doi: 10.46572/NATURENGS.1293185.
- [16] M. N. Raihen and S. Akter, “Comparative Assessment of Several Effective Machine Learning Classification Methods for Maternal Health Risk,” *Computational Journal of Mathematical and Statistical Sciences*, vol. 3, no. 1, pp. 161–176, Apr. 2024, doi: 10.21608/cjmss.2024.259490.1036.
- [17] T. R. Noviany, S. I. Nainggolan, R. Raihan, I. Firmansyah, and R. Idroes, “Maternal Health Risk Detection Using Light Gradient Boosting Machine Approach,” *Infolitika Journal of Data Science*, vol. 1, no. 2, pp. 48–55, Dec. 2023, doi: 10.60084/ijds.v1i2.123.
- [18] K. Cabello-Solorzano, L. Ortigosa de Araujo, M. Peña, L. Correia, and A. J. Tallón-Ballesteros, “The Impact of Data Normalization on the Accuracy of Machine Learning Algorithms: A Comparative Analysis,” *Lecture Notes in Networks and Systems*, vol. 750 LNNS, pp. 344–353, 2023, doi: 10.1007/978-3-031-42536-3_33.
- [19] P. V. Anusha, C. Anuradha, P. S. R. Chandra Murty, and C. S. Kiran, “Detecting outliers in high dimensional data sets using Z-score methodology,” *International Journal of Innovative Technology and Exploring Engineering*, vol. 9, no. 1, pp. 48–53, Nov. 2019, doi: 10.35940/IJITEE.A3910.119119.
- [20] W. Aprilliandhika and F. F. Abdulloh, “Comparison Of K-Nearest Neighbor And Support Vector Machine Algorithm Optimization With Grid Search Cv On Stroke Prediction,” *Jurnal Teknik Informatika (Jutif)*, vol. 5, no. 4, pp. 991–1000, Jul. 2024, doi: 10.52436/1.JUTIF.2024.5.4.1951.
- [21] A. Yaqin, D. Kurniawan, and J. Zeniarja, “Optimasi Algoritma K-Nearest Neighbors Menggunakan GridSearchCV untuk Klasifikasi Penyakit Diabetes,” *Infotekmesin*, vol. 16, no. 1, pp. 75–84, Jan. 2025, doi: 10.35970/INFOTEKMESIN.V16I1.2557.
- [22] M. Ahmed, M. A. Kashem, M. Rahman, and S. Khatun, “Review and Analysis of Risk Factor of Maternal Health in Remote Area Using the Internet of Things (IoT),” *Lecture Notes in Electrical Engineering*, vol. 632, pp. 357–365, 2020, doi: 10.1007/978-981-15-2317-5_30.
- [23] V. Da Poian *et al.*, “Exploratory data analysis (EDA) machine learning approaches for ocean world analog mass spectrometry,” *Frontiers in Astronomy and Space Sciences*, vol. 10, p. 1134141, May 2023, doi: 10.3389/FSPAS.2023.1134141/BIBTEX.
- [24] V. Werner de Vargas, J. A. Schneider Aranda, R. dos Santos Costa, P. R. da Silva Pereira, and J. L. Victória Barbosa, “Imbalanced data preprocessing techniques for machine learning: a systematic mapping study,” *Springer*, vol. 65, no. 1, pp. 31–57, Jan. 2023, doi: 10.1007/S10115-022-01772-8.
- [25] J. Elektronika and D. Komputer, “Mengoptimalkan Proses Pembersihan Data dalam Analisis Big Data Menggunakan Pipeline Berbasis AI,” *Elkom: Jurnal Elektronika dan Komputer*, vol. 17, no. 2, pp. 657–666, Dec. 2024, doi: 10.51903/ELKOM.V17I2.2311.
- [26] I. M. K. Karo and H. Hendriyana, “Klasifikasi Penderita Diabetes menggunakan Algoritma Machine Learning dan Z-Score,” *Jurnal Teknologi Terpadu*, vol. 8, no. 2, pp. 94–99, Dec. 2022, doi: 10.54914/JTT.V8I2.564.
- [27] P. P. Allorerung, A. Erna, M. Bagussahrir, and S. Alam, “Analisis Performa Normalisasi Data untuk Klasifikasi K-Nearest Neighbor pada Dataset Penyakit,” *JISKA (Jurnal Informatika Sunan Kalijaga)*, vol. 9, no. 3, pp. 178–191, Sep. 2024, doi: 10.14421/jiska.2024.9.3.178-191.
- [28] V. R. Prasetyo, M. Mercifia, A. Averina, L. Sunyoto, and B. Budiarjo, “Prediksi Rating Film Pada Website Imdb Menggunakan Metode Neural Network,” *Network Engineering Research Operation*, vol. 7, no. 1, p. 1, Apr. 2022, doi: 10.21107/NERO.V7I1.268.
- [29] V. R. Prasetyo, M. F. Naufal, and Budiarjo, “Implementation of K-Means and K-Nearest Neighbor Methods for Laptop Recommendation Websites,” *Proceedings of the 4th International Conference on Informatics, Technology and Engineering 2023 (InCITE 2023)*, pp. 457–469, Nov. 2023, doi: 10.2991/978-94-6463-288-0_38.



- [30] X. Yang, L. Hou, Y. Zhou, W. Wang, and J. Yan, “Dense label encoding for boundary discontinuity free rotation detection,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 15814–15824, 2021, doi: 10.1109/CVPR46437.2021.01556.
- [31] R. Oktafiani, A. Hermawan, and D. Avianto, “Pengaruh Komposisi Split data Terhadap Performa Klasifikasi Penyakit Kanker Payudara Menggunakan Algoritma Machine Learning,” *Jurnal Sains dan Informatika*, vol. 9, no. 1, pp. 19–28, Jun. 2023, doi: 10.34128/JSI.V9I1.622.
- [32] S. Zhang, “Challenges in KNN Classification,” *IEEE Trans Knowl Data Eng*, vol. 34, no. 10, pp. 4663–4675, Oct. 2022, doi: 10.1109/TKDE.2021.3049250.
- [33] H. Al Azies and M. Naufal, “A Stacking Approach to Enhance K-Nearest Neighbors Performance for Autism Screening,” *Jurnal Teknologi Informasi dan Terapan (J-TIT)*, vol. 11, no. 2, Dec. 2024, doi: 10.25047/JTIT.V11I2.5517.
- [34] Z. Maisat, E. Darmawan, and A. Fauzan Dianta, “Implementasi Optimasi Hyperparameter GridSearchCV Pada Sistem Prediksi Serangan Jantung Menggunakan SVM,” *Teknologi: Jurnal Ilmiah Sistem Informasi*, vol. 13, no. 1, pp. 8–15, Jan. 2023, doi: 10.26594/TEKNOLOGI.V13I1.3098.
- [35] Muljono, S. A. Wulandari, H. Al Azies, M. Naufal, W. A. Prasetyanto, and F. A. Zahra, “Breaking Boundaries in Diagnosis: Non-Invasive Anemia Detection Empowered by AI,” *IEEE Access*, vol. 12, pp. 9292–9307, 2024, doi: 10.1109/ACCESS.2024.3353788.
- [36] “Aplikasi Prediksi Resiko Kehamilan.” Accessed: Aug. 09, 2025. [Online]. Available: https://risetkami.my.id/mahasiswa/mhr_kehamilan.html