

Perbandingan Efektivitas Random Forest dan K-Nearest Neighbors (KNN) dalam Memprediksi Risiko Penyakit Diabetes Melitus

Ryandra Surya Nandika^{1*}

¹ STMIK AMIKOM Surakarta

*Corresponding Email: ryandra.10344@mhs.amikomsolo.ac.id

Abstrak: Identifikasi penyakit diabetes melitus masih menjadi tantangan karena pola data yang sangat beragam, sehingga diperlukan sebuah metode klasifikasi yang lebih akurat. Penelitian ini dilakukan bertujuan untuk melakukan perbandingan performa algoritma Random Forest dan K-Nearest Neighbors (KNN) dalam mengklasifikasikan data penyakit diabetes menggunakan dataset dari Kaggle yang memuat 8 atribut. Proses analisis meliputi preprocessing, normalisasi, pembagian data latih dan data uji, dan optimasi parameter dengan GridSearchCV. Hasil menunjukkan bahwa algoritma Random Forest lebih baik dengan akurasi 74,7% dibandingkan KNN yang hanya 70,8% serta lebih efektif dalam mendeteksi kelas positif berdasarkan precision, recall, F1-score, dan confusion matrix. Berdasarkan hasil ini maka menegaskan bahwa algoritma Random Forest lebih baik dalam melakukan klasifikasi penyakit diabetes melitus dan memungkinkan untuk mendukung pengembangan sistem diagnosis penyakit diabetes berbasis machine learning dibidang kesehatan.

Kata kunci: Diabetes Melitus, Random Forest, K-Nearest Neighbors (KNN), Klasifikasi, Machine Learning.

***Abstract:** Identification of diabetes mellitus remains a challenge due to the highly diverse data patterns, necessitating a more accurate classification method. This study aimed to compare the performance of the Random Forest and K-Nearest Neighbors (KNN) algorithms in classifying diabetes data using a dataset from Kaggle containing 8 attributes. The analysis process included preprocessing, normalization, training data and test data sharing, and parameter optimization with GridSearchCV. The results showed that the Random Forest algorithm was better with an accuracy of 74.7% compared to KNN which was only 70.8% and was more effective in detecting positive classes based on precision, recall, F1-score, and confusion matrix. Based on these results, it confirms that the Random Forest algorithm is better in classifying diabetes mellitus and allows it to support the development of a machine learning-based diabetes diagnosis system in the health sector.*

***Keywords:** Diabetes Melitus, Random Forest, K-Nearest Neighbors (KNN), Classification, Machine Learning.*

PENDAHULUAN

Diabetes melitus saat ini menjadi salah satu penyakit dengan jumlah penderita yang terus bertambah di berbagai negara, termasuk di Indonesia. Penyakit ini terjadi ketika kadar glukosa dalam darah meningkat akibat gangguan produksi atau fungsi insulin. Jika tidak terdeteksi dan ditangani dengan tepat, diabetes dapat menyebabkan berbagai komplikasi serius seperti penyakit jantung, gagal ginjal, kebutaan, hingga amputasi (Yolanda, 2022). Tantangan terbesar dari kondisi ini adalah bagaimana mengenali dan menanganinya dengan cepat, terutama karena gejalanya tidak selalu jelas dan kondisi setiap penderita berbeda-beda. Sehingga dibutuhkan metode yang efektif dalam mendiagnosis penyakit diabetes secara cepat dan akurat (Diranisha, 2024).

Perkembangan teknologi, khususnya dalam bidang kecerdasan buatan dan machine learning, membuka peluang baru dalam membantu proses identifikasi penyakit. Pemanfaatan teknologi ini, memungkinkan tenaga kesehatan dapat memperoleh dukungan tambahan dalam mengambil sebuah keputusan yang lebih tepat. Meski begitu, kinerja machine learning sangat dipengaruhi oleh algoritma yang digunakan serta kemampuannya dalam memahami karakter data medis yang kompleks (Zuhri, 2025). Data diabetes sendiri memiliki karakter yang beragam, perbedaan skala nilai, dan hubungan antar fitur yang tidak selalu linear, sehingga pemilihan algoritma klasifikasi menjadi faktor penting untuk menghasilkan model yang tepat.

Dalam klasifikasi data medis, termasuk kasus diabetes melitus, Random Forest dan K-Nearest Neighbor (KNN) merupakan dua algoritma yang cukup sering digunakan karena memiliki karakteristik yang berbeda dalam menangani data. Random Forest dikenal mampu menangani data dengan kompleksitas tinggi dan mengurangi risiko overfitting melalui mekanisme ensemble pada pohon keputusan (Inzaghi, 2025). Sementara itu, KNN

mengandalkan kedekatan antar data sehingga lebih sensitif terhadap pola lokal dan variasi distribusi data, yang umum ditemukan pada data pasien diabetes (Salmon, 2024). Walaupun keduanya sering digunakan, belum ada kesimpulan yang benar-benar jelas mengenai algoritma mana yang paling sesuai untuk menangani karakteristik data diabetes yang beragam.

Performa masing-masing algoritma sangat dipengaruhi oleh kualitas dataset, tahapan proses data, serta pengaturan parameter model yang digunakan (Tangelobo, 2023). Oleh karena itu, penelitian ini dilakukan untuk menganalisis dan membandingkan performa algoritma Random Forest dan K-Nearest Neighbor (KNN) dalam mengklasifikasikan data penyakit diabetes. Hasil dari perbandingan ini diharapkan dapat memberikan gambaran yang lebih jelas mengenai algoritma yang lebih optimal untuk mendeteksi dini diabetes melitus, sehingga keputusan medis dapat diambil dengan tingkat akurasi yang lebih baik (Hadi dan Sirodj, 2023).

Penelitian ini perlu dilakukan karena meskipun berbagai algoritma machine learning telah diterapkan dalam klasifikasi penyakit diabetes melitus, belum menghasilkan kesimpulan yang konsisten terkait algoritma yang paling sesuai untuk karakteristik data diabetes yang kompleks dan heterogen. Perbedaan pendekatan antara Random Forest dan K-Nearest Neighbor (KNN) menjadikan keduanya relevan untuk dianalisis secara lebih kontekstual, khususnya dengan menggunakan metrik evaluasi yang tidak hanya berfokus pada akurasi. Dengan demikian, penelitian ini bertujuan untuk mengisi kekosongan tersebut melalui analisis kinerja algoritma dalam mendukung diagnosis dini diabetes melitus.

Berdasarkan kebutuhan akan diagnosis yang lebih cepat dan tepat, penelitian ini berfokus pada proses analisis dan perbandingan kinerja Random Forest dan KNN dalam mengklasifikasikan data diabetes. Evaluasi dilakukan menggunakan metrik akurasi, seperti akurasi, precision, recall, dan F1-score agar hasil kinerja model lebih menyeluruh (Fadillah, 2024). Melalui penelitian ini, diharapkan dapat diperoleh pemahaman yang lebih baik mengenai pemilihan algoritma yang sesuai untuk sistem pendukung diagnosis diabetes berbasis machine learning yang tidak hanya akurat, tetapi juga efisien dan mudah diterapkan dalam praktik medis.

LANDASAN TEORI

Sejumlah penelitian sebelumnya telah mencoba membandingkan kinerja algoritma Random Forest dan K-Nearest Neighbors (KNN) dalam mengklasifikasikan penyakit diabetes. Salah satu studi yang dilakukan oleh (Salmon, 2024) menunjukkan bahwa Random Forest mampu mencapai tingkat akurasi yang sangat tinggi, hingga 99,75%, yang melampaui performa KNN pada dataset yang digunakan. Meski hasil tersebut terlihat menjanjikan, penerapan tahapan seperti normalisasi data, seleksi fitur, atau penanganan data tidak seimbang, sehingga berpotensi memberikan pengaruh yang cukup signifikan terhadap kinerja algoritma khususnya KNN yang sangat sensitif terhadap skala dan distribusi data. Temuan ini sejalan dengan penelitian (Tangelobo, 2023), yang menyebutkan bahwa Random Forest cenderung menunjukkan performa yang lebih stabil pada berbagai metrik seperti akurasi, presisi, recall, dan F1-score jika disandingkan dengan KNN maupun Naive Bayes.

Meski begitu, KNN tetap menunjukkan potensi yang cukup kompetitif pada kondisi tertentu. Hadi dan Sirodj, (2023) menyatakan bahwa KNN bekerja cukup efektif ketika digunakan pada dataset dengan jumlah sampel yang tidak terlalu besar. Fadillah, (2024) juga mengungkapkan bahwa Random Forest menunjukkan performa yang lebih stabil untuk data yang memiliki banyak fitur, sementara KNN justru bisa lebih unggul pada dataset dengan dimensi kecil dan struktur data yang sederhana. Namun, kedua penelitian tersebut belum menguraikan proses penyesuaian parameter secara sistematis, sehingga hasil yang diperoleh masih belum mencerminkan kemampuan dari masing-masing algoritma.

Penelitian lain juga menunjukkan bahwa kinerja algoritma Random Forest dan K-Nearest Neighbor (KNN) dalam klasifikasi penyakit diabetes melitus sangat dipengaruhi oleh karakteristik data dan konfigurasi model yang digunakan. Yolanda, (2023) dalam penelitiannya menggunakan dataset yang konsisten menunjukkan bahwa akurasi Random Forest dalam mengklasifikasikan penyakit diabetes mencapai 80,91%, sedangkan KNN hanya sebesar 74,06%. Hal ini menegaskan bahwa performa algoritma sangat dipengaruhi oleh karakteristik data yang digunakan. Faktor lain yang juga berperan penting adalah parameter dalam pemodelan. Diranisha, (2024) menekankan bahwa tuning parameter pada Random Forest dapat meningkatkan performa model secara signifikan. Sementara itu, Zuhri dan Ramadhan, (2025) membuktikan bahwa konfigurasi parameter model yang seragam sangat mempengaruhi performa yang diperoleh. Hal ini memudahkan interpretasi hasil apakah perbedaan performa berasal dari karakteristik algoritma atau dari konfigurasi parameter yang digunakan.

Berdasarkan hasil tinjauan terhadap penelitian-penelitian terdahulu, dapat disimpulkan bahwa Random Forest dan K-Nearest Neighbor (KNN) telah banyak diterapkan dalam klasifikasi penyakit diabetes melitus, namun hasil yang diperoleh masih menunjukkan variasi kinerja yang dipengaruhi oleh perbedaan dataset, teknik praproses, konfigurasi parameter, serta metode evaluasi yang digunakan masing-masing penelitian. Kondisi ini menyebabkan belum adanya gambaran yang konsisten mengenai kesesuaian kedua algoritma tersebut pada karakteristik data tertentu. Oleh karena itu, penelitian ini dilakukan sebagai pelengkap terhadap penelitian-penelitian sebelumnya dengan menerapkan Random Forest dan KNN pada dataset yang sama. Dengan pendekatan tersebut, penelitian ini diharapkan dapat memberikan perbandingan kinerja yang lebih objektif sekaligus melengkapi pemahaman mengenai penerapan algoritma machine learning dalam mendukung klasifikasi dan diagnosis dini penyakit diabetes melitus.

METODE



Gambar 1. Flowchart Metode Penelitian

Berdasarkan gambar 1, alur penelitian dimulai dari pengumpulan data, kemudian data tersebut diproses agar siap digunakan. Pada tahap preprocessing, dilakukan pengecekan nilai yang hilang, pemisahan antara fitur dan target, pembagian data menjadi data latih dan data uji, dan normalisasi agar skala datanya seragam. Setelah itu, barulah metode machine learning diterapkan. Proses ini mencakup pelatihan model, penyesuaian parameter, hingga memilih model yang memberikan hasil terbaik. Model kemudian digunakan untuk melakukan prediksi, dan pada tahap akhir dilakukan evaluasi untuk melihat performa model.

Sumber data

Sumber data yang digunakan dalam penelitian ini adalah dataset klasifikasi diabetes yang berasal dari Kaggle. Dataset ini terdiri dari sejumlah variabel medis yang dapat digunakan untuk

memprediksi kemungkinan seseorang menderita penyakit diabetes. Data ini diperoleh dalam format CSV agar lebih mudah untuk dilakukan pengolahan pada Google Collab (Shiva, 2023).

Preprocessing Data

Sebelum dilakukan analisis dan implementasi model machine learning, dataset ini melalui beberapa tahap preprocessing agar dataset siap digunakan untuk proses analisis lebih lanjut sebagai berikut.

Pemeriksaan Nilai Hilang

Teknik ini memastikan bahwa tidak ada data kosong yang dapat mempengaruhi performa model, sehingga tidak diperlukan penghapusan data.

Pemisahan Fitur dan Target

Dataset terdiri dari 8 fitur medis dan 1 label target (Outcome), yang mengindikasikan apakah seseorang menderita diabetes (1) atau tidak (0). Fitur-fitur medis yang digunakan mencerminkan kondisi pasien dan digunakan indikator dalam diagnosis diabetes. Fitur Pregnancies menunjukkan jumlah kehamilan yang pernah dialami pasien, dan Glucose menunjukkan kadar glukosa dalam darah. BloodPressure menunjukkan tekanan darah, sedangkan SkinThickness menunjukkan ketebalan lipatan kulit yang berkaitan dengan lemak tubuh. Selanjutnya fitur Insulin menunjukkan kadar insulin dalam darah, sementara BMI (Body Mass Index) digunakan untuk mengukur indeks massa tubuh pasien. Faktor keturunan diabetes ditunjukkan oleh fitur DiabetesPedigreeFunction, serta fitur Age menunjukkan usia pasien.

Pemisahan Data Latih dan Data Uji

Dataset dibagi menjadi 80% untuk pelatihan dan 20% untuk pengujian menggunakan metode train-test split. Pemisahan ini dilakukan untuk memastikan model dapat diuji pada data yang tidak terlihat selama proses pelatihan. Dari hasil pemisahan data latih dan data uji didapatkan data latih sebanyak 615 data dan data uji sebanyak 155 data. Data yang digunakan untuk penelitian ini adalah data uji dengan total 155 data.

Normalisasi Data

Data numerik dinormalisasi menggunakan StandardScaler untuk memastikan setiap fitur memiliki skala yang seragam, sehingga dapat meningkatkan kinerja model machine learning.

Perbandingan Skenario Preprocessing Data

Untuk menganalisis pengaruh preprocessing terhadap kinerja algoritma klasifikasi, diterapkan dua skenario pengolahan data, yaitu penggunaan data tanpa normalisasi dan penggunaan data yang sudah dinormalisasi dengan StandardScaler. Perbandingan kedua skenario ini dilakukan untuk memahami peran preprocessing dalam meningkatkan performa model, terutama pada algoritma K-Nearest Neighbors (KNN).

Pada skenario pertama, Random Forest dan KNN dilatih serta diuji menggunakan data asli tanpa dilakukan normalisasi. Pendekatan ini digunakan untuk memperoleh gambaran kinerja awal masing-masing algoritma sebagai acuan sebelum dilakukan pengolahan data.

Skenario kedua data dilakukan proses normalisasi menggunakan StandardScaler, setiap fitur disesuaikan sehingga memiliki nilai rata-rata nol dan standar deviasi satu. Ini dilakukan bertujuan agar skala antar fitur tidak ada variabel yang mendominasi. Selanjutnya, hasil kedua skenario dibandingkan menggunakan metrik evaluasi yang sama, meliputi akurasi, precision, recall, F1-Score, serta confusion matrix. Dengan cara ini penelitian juga dapat memberikan gambaran yang lebih menyeluruh mengenai dampak preprocessing terhadap kinerja model klasifikasi.

Tuning (Mencari Nilai Terbaik)

Mencari nilai K optimal dalam rentang 1-20 menggunakan GridSearchCV dengan 5-fold cross-validation untuk memilih parameter terbaik untuk meningkatkan performa model KNN.

Prediksi dengan Model Terbaik

Menggunakan model KNN dengan K terbaik yang telah ditemukan untuk melakukan prediksi pada data uji untuk menguji performa model pada data yang belum pernah dilihat sebelumnya.

Evaluasi Model

Evaluasi Model dilakukan dengan menghitung confusion matrix, classification report, dan akurasi untuk mengukur performa klasifikasi serta memahami tingkat kesalahan dan efektivitas model.

Accuracy

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

Gambar 2. Rumus Accuracy

Gambar 2 menunjukkan Rumus Accuracy. Accuracy dilakukan untuk mengukur tingkat ketepatan model dalam mengklasifikasikan seluruh data secara benar, baik kelas positif maupun negatif.

Precision

$$\text{Precision} = \frac{TP}{TP + FP}$$

Gambar 3. Rumus Precision

Gambar 3 menunjukkan Rumus Precision. Precision dilakukan untuk mengukur tingkat ketepatan prediksi positif yang dihasilkan oleh model, yaitu seberapa besar proporsi data yang diprediksi positif benar-benar merupakan data positif.

Recall

$$\text{Recall} = \frac{TP}{TP + FN}$$

Gambar 4. Rumus Recall

Gambar 4 menunjukkan Rumus Recall. Recall dilakukan untuk mengukur kemampuan model dalam mendeteksi seluruh data yang benar-benar positif.

F1-Score

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Gambar 5. Rumus F1-Score

Gambar 5 menunjukkan Rumus F1-Score. F1-Score merupakan rata-rata dari precision dan recall yang digunakan untuk memberikan keseimbangan antara kedua metrik.

Metode Machine Learning

Penelitian ini menggunakan dua algoritma, yaitu Random Forest dan KNN, karena keduanya mampu mengolah data yang beragam dan menghasilkan prediksi yang cukup akurat. Random Forest dipilih karena kemampuannya dalam mengolah data dengan jumlah fitur yang banyak, lebih tahan terhadap masalah overfitting dan relatif robust ketika menghadapi data yang mengandung missing value maupun outlier. Sementara itu, KNN digunakan karena pendekatannya yang sederhana dan mudah dipahami, tidak memerlukan asumsi yang kompleks

terhadap data, serta menentukan klasifikasi berdasarkan tingkat kedekatan antar pola data. Pendekatan ini membuat KNN cukup efektif untuk menangani data dengan hubungan nonlinier.

HASIL DAN PEMBAHASAN

Hasil

Pada penelitian ini setelah dilakukan klasifikasi menggunakan algoritma Random Forest dan KNN, hasil klasifikasi akan dilakukan perbandingan untuk mengetahui model terbaik.

Perbandingan Akurasi Model Berdasarkan Skenario Preprocessing Data

Tabel 1. Perbandingan Akurasi Berdasarkan Skenario Preprocessing

Model	Tanpa Normalisasi	Normalisasi (StandardScaler)
Random Forest	74,7%	74,7%
KNN	67,5%	70,8%

Berdasarkan Tabel 1, dapat dilihat bahwa normalisasi data memberikan dampak yang berbeda pada masing-masing algoritma. Random Forest menunjukkan nilai akurasi yang sama 74,7% pada kedua skenario preprocessing, sedangkan KNN mengalami peningkatan akurasi dari 67,5% menjadi 70,8% setelah dilakukan normalisasi data.

Hasil Klasifikasi Menggunakan Random Forest

Penelitian ini, algoritma Random Forest digunakan untuk melakukan klasifikasi dengan optimasi hyperparameter menggunakan GridSearchCV. Setelah proses pelatihan dan evaluasi, model memperoleh akurasi sebesar 74,7%.

Evaluasi Model Random Forest

Tabel 2. Evaluasi Model Random Forest

Kelas	Precision	Recall	F1-Score	Support
0 (Negatif)	81%	80%	80%	99
1 (Positif)	64%	65%	65%	55

Berdasarkan Tabel 2 diatas, dapat dilihat bahwa model lebih baik dalam mendeteksi kelas negatif, dengan precision untuk kelas negatif adalah **81%**, yang berarti dari seluruh prediksi negatif yang dibuat oleh model, **81% benar**. Sementara itu, precision untuk kelas positif hanya **64%**, yang menunjukkan bahwa masih terdapat kesalahan dalam mengklasifikasikan sampel positif.

Confusion Matrix Random Forest

Tabel 3. Confusion Matrix Random Forest

	Positif (1)	Negatif (0)
Positif (1)	36	20
Negatif (0)	19	79

Berdasarkan Tabel 3, dapat dilihat bahwa model mampu mengenali 36 sampel positif dengan tepat, yang menunjukkan bahwa sebagian data positif berhasil diklasifikasikan dengan benar. Meskipun begitu, masih ada 20 sampel negatif yang justru terbaca sebagai positif, menandakan adanya kesalahan dalam proses prediksi. Selain itu, terdapat 19 sampel positif yang tidak terdeteksi dan diklasifikasikan sebagai negatif, sehingga menunjukkan bahwa model

masih melewatkan beberapa kasus penting. Di sisi lain, model cukup baik dalam mengidentifikasi sampel negatif, dengan 79 data berhasil diklasifikasikan secara akurat.

Hasil Klasifikasi Menggunakan K-Nearest Neighbors (KNN)

Pada penelitian ini, algoritma K-Nearest Neighbors (KNN) digunakan untuk melakukan klasifikasi dengan optimasi jumlah tetangga terdekat yang menghasilkan nilai optimal 11. Setelah proses pelatihan dan evaluasi, model memperoleh akurasi sebesar 70,8%.

Evaluasi Model KNN

Tabel 4. Evaluasi Model

Kelas	Precision	Recall	F1-Score	Support
0 (Negatif)	76%	79%	78%	99
1 (Positif)	60%	56%	58%	55

Berdasarkan Tabel 4, dapat dilihat bahwa model lebih baik dalam mendeteksi kelas negatif, dengan Precision untuk kelas negatif adalah **76%**, yang berarti dari seluruh prediksi negatif yang dibuat oleh model, **76% benar**. Sementara itu, precision untuk kelas positif hanya **60%**, yang menunjukkan bahwa masih terdapat kesalahan dalam mengklasifikasikan sampel positif.

Confusion Matrix KNN

Tabel 5. Confusion Matrix

	Positif (1)	Negatif (0)
Positif (1)	31	21
Negatif (0)	24	78

Berdasarkan Tabel 5, dapat dilihat bahwa model KNN mampu mengenali 31 sampel positif dengan benar sebagai True Positive. Namun, model ini juga masih menghasilkan kesalahan, yaitu ketika 21 sampel negatif justru diklasifikasikan sebagai positif, yang disebut sebagai False Positive. Selain itu, terdapat 24 sampel positif yang tidak teridentifikasi dengan tepat dan malah dianggap sebagai negatif, sehingga masuk dalam kategori False Negative. Di sisi lain, model berhasil mengklasifikasikan 78 sampel negatif dengan benar sebagai True Negative.

Perbandingan Model

Berdasarkan hasil klasifikasi menggunakan Random Forest dan K-Nearest Neighbors (KNN), berikut adalah analisis perbandingan performa kedua model:

Akurasi Model

Random Forest memiliki akurasi 74,7%, lebih tinggi dibandingkan KNN yang memiliki akurasi 70,8%. Hal ini menunjukkan bahwa Random Forest lebih baik dalam mengklasifikasikan data dengan benar dibandingkan KNN.

Kesalahan Klasifikasi

Tabel 6. Kesalahan Klasifikasi

Martrix	Random Forest	KNN
False Positive (FP)	20	21
False Negative (FN)	19	24
True Positive (TP)	36	31
True Negative (TN)	79	78

Berdasarkan tabel 6, dapat dilihat bahwa Random Forest menunjukkan kemampuan yang lebih baik dalam mengenali sampel positif secara tepat, yang ditunjukkan oleh jumlah True Positive yang lebih tinggi. Sebaliknya, KNN terlihat lebih sering melakukan kesalahan pada data negatif karena jumlah False Positive yang dihasilkan sedikit lebih banyak. Selain itu, KNN juga lebih sering gagal mendeteksi sampel positif dibandingkan Random Forest, sehingga jumlah False Negative pada model ini menjadi lebih besar. Untuk kelas negatif, kedua model sama-sama mampu mengidentifikasi sebagian besar sampel dengan benar, yang tercermin dari jumlah True Negative sebanyak 78 sampel yang terklasifikasi dengan tepat.

Pembahasan

Performa pada kelas Positif dan Negatif

Kelas Negatif (0):

Pada kelas negatif (0), kinerja kedua model terlihat seimbang dalam mengenali data yang benar-benar negatif. Random Forest mampu mengidentifikasi 79 sampel negatif dengan tepat, sedangkan KNN berhasil mengenali 78 sampel. Selisihnya tidak besar, tetapi KNN sedikit lebih sering melakukan kesalahan dengan mengklasifikasikan data negatif sebagai positif, yaitu sebanyak 21 kasus, sementara Random Forest hanya menghasilkan 20 kesalahan yang sama.

Kelas Positif (1):

Pada kelas positif (1), perbedaan kinerja kedua model terlihat lebih jelas, dengan Random Forest menunjukkan hasil yang lebih baik dibandingkan KNN. Hal ini dapat dilihat dari nilai recall Random Forest yang mencapai 65%, lebih tinggi dibandingkan KNN yang hanya 56%. Artinya, Random Forest mampu mengenali lebih banyak sampel positif secara tepat. Kondisi tersebut juga terlihat dari jumlah false negative yang lebih rendah, yakni 19 pada Random Forest, sedangkan KNN menghasilkan 24 kesalahan yang sama. Sebaliknya, KNN lebih sering gagal mendeteksi data positif dan justru mengklasifikasikannya sebagai negatif, ini dapat meningkatkan risiko kesalahan klasifikasi terutama pada kondisi identifikasi kelas positif menjadi aspek krusial.

Kesimpulan Pembahasan

Berdasarkan hasil evaluasi melalui confusion matrix, perbandingan model, akurasi model, dan kesalahan klasifikasi, Random Forest terbukti menunjukkan performa yang lebih baik dibandingkan KNN. Random Forest tidak hanya mampu menghasilkan akurasi yang tinggi tetapi juga lebih baik dalam mengenali kelas positif serta menjaga keseimbangan prediksi antar kelas. Sedangkan KNN masih mengalami kendala dalam mendeteksi data positif, yang dapat dilihat dari jumlah False Negative yang relatif lebih besar. Meski begitu, KNN tetap lebih sederhana dan mudah diimplementasikan. Namun, kinerja algoritma KNN sangat dipengaruhi tahapan preprocessing serta pemilihan parameter yang digunakan. Walaupun Random Forest membutuhkan proses yang lebih besar dan model yang lebih rumit, kelebihanannya terlihat dari hasil klasifikasi yang lebih baik. Prediksi yang dihasilkan juga lebih stabil, oleh karena itu Random Forest menjadi model yang lebih tepat digunakan.

SIMPULAN

Berdasarkan hasil penelitian ini, dapat disimpulkan bahwa Random Forest memiliki keunggulan dibandingkan KNN dalam berbagai aspek utama, termasuk akurasi, keseimbangan klasifikasi, serta kemampuan mendeteksi kelas positif. KNN cenderung kurang akurat dalam mengidentifikasi kelas positif, sebagaimana terlihat dari tingginya jumlah False Negative (FN) dibandingkan Random Forest. Hal ini menunjukkan bahwa model KNN lebih sering salah mengklasifikasikan data positif sebagai negatif, yang dapat berdampak pada analisis yang

membutuhkan deteksi akurat terhadap kelas positif. Selain itu, hasil pengujian menunjukkan bahwa skenario preprocessing data sebelum dilakukan normalisasi berpengaruh pada kinerja KNN, dimana setelah dilakukan normalisasi data mampu meningkatkan hasil klasifikasi. Sedangkan Random Forest menunjukkan hasil yang stabil pada berbagai kondisi pengolahan data. Secara keseluruhan, Random Forest menjadi model yang lebih optimal dalam penelitian ini, karena memberikan hasil yang lebih akurat dalam mengklasifikasikan kedua kelas serta memiliki performa yang lebih baik.

DAFTAR PUSTAKA

- Diranisha, N., Prasetyo, M., & Hidayat, A. (2024). Penerapan random forest dalam klasifikasi data medis untuk diagnosis diabetes. *Jurnal Informatika Medis*, 15(1), 78–89.
- Diranisha, N., Santoso, P., & Kurniawan, W. (2024). Optimalisasi parameter random forest untuk prediksi penyakit diabetes. *Jurnal Sistem Cerdas*, 20(1), 98–110.
- Fadillah, A., Setiawan, R., & Handayani, Y. (2024). Studi perbandingan random forest dan KNN dalam klasifikasi data medis. *Jurnal Data Science Kesehatan*, 16(3), 147–159.
- Hadi, H., & Sirodj, M. (2023). Evaluasi kinerja algoritma K-nearest neighbor dalam prediksi diabetes. *Jurnal Teknologi Informasi dan Kesehatan*, 11(1), 89–102.
- Inzaghi, M. R., & Susanto, E. R. (2025). Evaluasi kinerja model random forest dalam memprediksi diabetes berdasarkan dataset kesehatan di Indonesia [Evaluation of the performance of the random forest model in predicting diabetes based on health datasets in Indonesia]. *[Nama jurnal belum dicantumkan]*, 5(7), 1857–1866.
- Salmon, R., Nugraha, A., & Wahyudi, T. (2024). Perbandingan kinerja algoritma random forest dan K-nearest neighbor untuk prediksi diabetes. *Journal of Medical Data Science*, 18(4), 56–67.
- Shiva. (2023). *Diabetes classification* [Data set]. Kaggle. <https://www.kaggle.com/datasets/shiva8227/diabetes-classification>
- Tangelobo, B., Suryadi, F., & Hartanto, K. (2023). Analisis perbandingan algoritma machine learning untuk diagnosis penyakit diabetes. *Jurnal Ilmu Komputer dan Kesehatan*, 17(2), 134–145.
- Yolanda, R. (2022a). Analisis algoritma machine learning dalam mendiagnosis penyakit diabetes. *Jurnal Kesehatan Masyarakat*, 12(3), 45–56.
- Yolanda, R. (2022b). Penerapan machine learning dalam deteksi dini diabetes: Studi kasus random forest dan KNN. *Jurnal Kesehatan Digital*, 13(2), 67–78.
- Zuhri, M., Rahmad, S., & Purnomo, L. (2025). Implementasi algoritma K-nearest neighbor dalam prediksi diabetes. *Jurnal Teknik Informatika dan Kesehatan*, 10(2), 112–124.