



Klasifikasi Penyakit Ginjal Kronis Menggunakan K-Nearest Neighbors dengan Feature selection Pearson Correlation Coefficient

Moh Rizan Khadiki, Vivi Aida Fitria*

Fakultas Teknologi dan Desain, Teknik Informatika, Institut Teknologi dan Bisnis Asia, Malang
Jl. Soekarno Hatta-Rembuksari No.1 A, Mojolangu, Kec. Lowokwaru, Kota Malang, Jawa Timur, Indonesia

Email: ¹putrasembons@gmail.com, ^{2,*}viviaida@asia.ac.id

Email Penulis Korespondensi: viviaida@asia.ac.id

Submitted: 17/03/2025; Accepted: 30/04/2025; Published: 30/04/2025

Abstrak—Penyakit ginjal kronis (PGK) adalah masalah kesehatan global yang berdampak pada kualitas hidup dan angka kematian. PGK tidak memiliki gejala pada tahap awal sehingga sering disebut sebagai "penyakit diam", membuat deteksi dini menjadi sulit. Penelitian ini bertujuan untuk membangun model klasifikasi PGK menggunakan algoritma K-Nearest Neighbors (KNN) dengan penerapan metode seleksi fitur Pearson Correlation Coefficient untuk meningkatkan kinerja model. Feature selection digunakan untuk mengurangi dimensi data dan mencegah overfitting. Dataset Kaggle "Chronic Kidney Disease" digunakan dalam penelitian ini. Hasil evaluasi menunjukkan bahwa model dengan feature selection mencapai akurasi 93.37%, precision 91.9%, recall 93.37%, dan F1-score 91.48%, sedangkan model tanpa feature selection menunjukkan akurasi 91.27%, precision 87.24%, recall 91.27%, dan F1-score 88.99%. Kontribusi penelitian ini adalah meningkatkan performa klasifikasi penyakit ginjal kronis dengan metode seleksi fitur untuk mencapai keseimbangan antara precision dan recall serta mengurangi kesalahan klasifikasi.

Kata Kunci: Feature Selection; Ginjal Kronis; Klasifikasi; KNN; Pearson Correlation Coefficient; Overfitting

Abstract—Chronic kidney disease (CKD) is a global health issue that impacts quality of life and mortality rates. CKD often shows no symptoms in its early stages, earning it the nickname "silent disease," which complicates early detection efforts. This study aims to develop a classification model for CKD using the K-Nearest Neighbors (KNN) algorithm combined with the Pearson Correlation Coefficient feature selection method to enhance model performance. Feature selection is employed to reduce data dimensionality and prevent overfitting. The Kaggle "Chronic Kidney Disease" dataset is used in this study. Evaluation results show that the model with feature selection achieved an accuracy of 93.37%, precision of 91.9%, recall of 93.37%, and F1-score of 91.48%, while the model without feature selection achieved an accuracy of 91.27%, precision of 87.24%, recall of 91.27%, and F1-score of 88.99%. The contribution of this research is to improve the classification performance of chronic kidney disease by utilizing feature selection methods to achieve a better balance between precision and recall while reducing classification errors.

Keywords: Chronic Kidney; Classification; Feature Selection; KNN; Pearson Correlation Coefficient; Overfitting

1. PENDAHULUAN

Ginjal adalah organ penting yang melakukan banyak hal untuk menjaga komposisi darah tetap sehat, seperti menjaga keseimbangan cairan dalam tubuh dan mencegah penumpukan kotoran [1]. Selain itu, ginjal menghasilkan enzim dan hormon yang membantu mengendalikan tekanan darah dalam tubuh dan menjaga tulang sehat dan kuat [1]. Karena menjaga kesehatan ginjal manusia sangat penting untuk mencegah penyakit ginjal. Retention urea dan sampah nitrogen lainnya dalam darah menyebabkan gangguan ginjal karena tubuh tidak dapat mempertahankan keseimbangan cairan dan elektrolit [2]. Karena air yang digunakan untuk metabolisme tubuh manusia belum dibersihkan, tubuh akan mengalami kerusakan, yang dapat menyebabkan berbagai penyakit karena gangguan ginjal sulit dideteksi [3]. Gagal ginjal, bahkan jika tidak diobati, dapat menyebabkan kondisi kronis yang menghentikan kinerja ginjal selamanya, yang merupakan salah satu gangguan ginjal [3]. Penyakit Ginjal Kronis (PGK) menjadi permasalahan kesehatan global akibat peningkatan prevalensi dan tingginya angka gagal ginjal yang terus bertambah setiap tahun. Karena seringkali tidak menunjukkan gejala, penyakit ini sering disebut sebagai "penyakit diam". Namun, penyakit ini sulit disembuhkan. Diabetes dan tekanan darah tinggi adalah penyebab utamanya [4]. PGK adalah ketika ginjal tidak berfungsi secara normal untuk waktu yang lama [5]. Penyakit ini menyebabkan penurunan fungsi ginjal yang seringkali tidak terlihat, meskipun gejalanya muncul secara bertahap. Akibatnya, penderita gagal ginjal hanya perlu menunggu kematian ketika kondisinya menjadi parah [6]. Data Global Burden of Disease (GBD) 2019 dari Institute for Health Metrics and Evaluation (IHME) mengungkapkan bahwa penyakit ginjal kronis merupakan salah satu dari sepuluh penyebab kematian tertinggi di Indonesia [7]. Menurut Dr. Eva Susanti, Direktur Pencegahan dan Pengendalian Penyakit Tidak Menular Kementerian Kesehatan RI, jumlah kematian akibat penyakit ginjal kronis di Indonesia telah melebihi 42 ribu [7]. Dengan banyaknya jumlah manusia yang terkena PGK, maka diperlukan pemanfaatan teknologi untuk mengatasi permasalahan tersebut yaitu pengklasifikasian penyakit gagal ginjal kronis menggunakan algoritma machine learning [8]. Machine learning merupakan Ilmu komputer yang bekerja tanpa diprogram secara eksplisit. Machine learning adalah kecerdasan buatan yang belajar membuat data. Ini diperlukan untuk menggunakan metode yang cepat dan ampuh untuk menemukan masalah baru [9].

Penelitian terkait yang menggunakan algoritma machine learning dilakukan oleh Setiawati and Wibawa (2020) [10] yang menemukan bahwa algoritma K-Nearest Neighbors (KNN) dapat digunakan untuk

mengklasifikasikan penyakit gagal jantung. Tingginya prevalensi gagal jantung perlu mendapat perhatian serius agar dapat ditangani sejak dini dan mengurangi risiko terjadinya komplikasi. Oleh karena itu, penelitian tersebut menerapkan algoritma KNN dan menghasilkan akurasi terbaik dengan $k = 7$, dimana setelah melalui evaluasi dengan confusion matrix, akurasi klasifikasi gagal jantung dengan KNN sebesar 91%. Penelitian serupa yang dilakukan oleh Indrianti, Dkk (2024) [11] menggunakan algoritma K-Nearest Neighbor (KNN) untuk klasifikasi penyakit ginjal kronis menunjukkan hasil yang cukup memuaskan. Dalam studi ini, 75% data digunakan sebagai data latih dan 25% sebagai data uji. Hasil klasifikasi penyakit ginjal kronis menunjukkan bahwa sistem mampu mengklasifikasikan data dengan akurasi sebesar 92,59%, nilai presisi 89,85%, recall sebesar 87,32%, dan f1-score mencapai 88,57%. Penelitian berikutnya dilakukan oleh Putri Aisyiyah Rakhma Devi (2024) [12] membahas penggunaan metode K-Nearest Neighbor (K-NN) dalam mengklasifikasikan stadium penyakit gagal ginjal kronis. Gagal ginjal kronis merupakan kondisi penurunan fungsi ginjal yang terjadi secara bertahap dan terbagi menjadi tiga stadium: ringan, sedang, dan berat. Dengan klasifikasi stadium, dokter dapat menentukan pengobatan yang tepat sesuai tingkat keparahan penyakit. Dalam penelitian ini, metode K-NN diuji menggunakan lima data uji dan menghasilkan akurasi sebesar 90% dengan tingkat kesalahan 10%, baik pada nilai K ganjil maupun genap. Hasil tersebut menunjukkan bahwa algoritma K-NN dapat membantu dalam mendiagnosis dan menentukan stadium gagal ginjal kronis secara lebih akurat. Penelitian selanjutnya dilakukan oleh Ahmadi and Abdi (2024) [13] klasifikasi menggunakan dua metode, yaitu K-Nearest Neighbor (KNN) dan Decision Tree ID3, dengan pengujian menggunakan variasi persentase data uji sebesar 10%, 20%, 30%, dan 40%. Hasil pengujian menunjukkan bahwa metode KNN memberikan akurasi tertinggi sebesar 99,16% pada saat menggunakan 30% data uji dengan nilai $k = 3$. Sementara itu, metode ID3 memperoleh akurasi tertinggi sebesar 98,33% pada persentase data uji yang sama, yaitu 30%.

Dalam keempat penelitian terdahulu, algoritma KNN telah terbukti efektif untuk mengklasifikasikan penyakit, termasuk gagal jantung dan gagal ginjal kronis, dengan tingkat akurasi yang tinggi (antara 90% hingga 99,16%), namun belum ada yang mengintegrasikan metode seleksi fitur berbasis Pearson Correlation Coefficient untuk mengoptimalkan kinerja KNN dalam mendeteksi penyakit ginjal kronis secara spesifik. Alasan penggunaan pilihan fitur adalah untuk meningkatkan kinerja model, menghindari overfitting, mengurangi dimensi data, dan menghemat waktu komputasi [14]. Metode feature selection yang digunakan yaitu pearson correlation coefficient dikarenakan pearson correlation dapat digunakan untuk variabel numerik [15]. Selain itu, pearson correlation secara efisien mampu mengatasi fitur bertipe campuran [16]. Penerapan Pearson Correlation Coefficient dalam feature selection bertujuan untuk menentukan fitur mana yang paling relevan dengan variabel target, khususnya dengan mengukur hubungan linear antara fitur. Fitur-fitur yang memiliki korelasi tinggi (positif atau negatif) dengan target dipertahankan, sementara fitur yang kurang signifikan diabaikan. Ini membantu mengurangi dimensionalitas data, meningkatkan performa model klasifikasi, serta mencegah overfitting [17]. Sedangkan, algoritma KNN digunakan karena mempunyai kelebihan yaitu teknik yang digunakan sederhana, efisien dan efektif dalam pengolahan objek data dan lain-lain [18]. Algoritma KNN adalah algoritma supervised learning yang dapat digunakan untuk classification dan regression. Label kelas atau nilai sampel diprediksi dengan mencari titik data K yang paling dekat dengan sampel. Algoritma KNN bekerja dengan prinsip yang sederhana namun efektif. Dalam proses klasifikasi, algoritma ini akan mencari K tetangga terdekat dari data uji berdasarkan metrik jarak, seperti Euclidean [19]. Untuk mengklasifikasikan objek baru, algoritma klasifikasi digunakan berdasarkan atribut dan data pelatihan yang digunakan, semakin kecil nilai K, semakin sensitif model terhadap data pelatihan [18]. Hasil penelitian yaitu performa akurasi model KNN dengan feature selection pearson correlation coefficient dalam mengklasifikasi penyakit gagal ginjal kronis.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Gambar 1 menunjukkan alur proses klasifikasi penyakit ginjal kronis menggunakan algoritma K-Nearest Neighbors (KNN) yang dikombinasikan dengan metode feature selection pearson correlation coefficient (PCC). Proses dimulai dari tahap input data dan preprocessing, dilanjutkan dengan seleksi fitur, pembagian data, klasifikasi, hingga evaluasi hasil prediksi.



Gambar 1. Tahapan Penelitian

Berikut ini merupakan uraian secara menyeluruh dari setiap tahapan yang ditampilkan pada Gambar 1. Tahapan tersebut disusun secara sistematis untuk menggambarkan langkah-langkah yang dilakukan selama proses penelitian, mulai dari input dataset, tahap pra-pemrosesan, pembagian data, pelatihan model, hingga evaluasi hasil klasifikasi.



2.1.1 Input Dataset

Pada tahap ini, dataset mengenai penyakit ginjal kronis diperoleh dari website Kaggle, dengan dataset yang digunakan berjudul “Chronic kidney Disease Dataset” yang terdiri dari 1659 data sampel. Sampel pasien masing-masing diidentifikasi dengan sebuah PatientID unik. Dataset ini melibatkan berbagai fitur yang berhubungan dengan kondisi kesehatan pasien, termasuk Age (usia), Gender (jenis kelamin), dan Ethnicity (etnis). Informasi terkait gaya hidup juga tercatat, faktor-faktor medis yang signifikan. Selain itu, berbagai parameter klinis lainnya juga dicatat. Dataset ini juga mengukur fungsi ginjal dengan mencantumkan GFR (Glomerular Filtration Rate), kadar Protein in Urine, dan ACR (Albumin-to-Creatinine Rasio), serta beragam parameter lainnya.

2.1.2 Preprocessing Z-Score

Pada tahap ini, sejumlah tindakan diambil untuk memastikan bahwa data yang digunakan dalam proses pemodelan berada dalam kondisi terbaik. Langkah pertama adalah pembersihan data untuk mengatasi nilai kosong (missing value) atau data yang tidak valid (outlier). Missing values dapat ditangani dengan cara penghapusan atau pengisian nilai berdasarkan rata-rata atau metode lain. Setelah itu, dilakukan normalisasi data untuk memastikan setiap fitur berada dalam rentang yang sama, sehingga fitur dengan skala yang lebih besar tidak mendominasi proses pembelajaran. Pada penelitian ini, normalisasi data menggunakan Z-score normalization, metode ini menggunakan nilai yang didasarkan pada rata-rata dan deviasi standar dari data [20]. Keunggulan dari menerapkan metode z-score normalisasi dalam prapemrosesan data yakni menyamakan skala data, menghindari bias model, dan meningkatkan konvergensi dalam pelatihan [21].

2.1.3 Feature Selection: PCS

Seleksi fitur dilakukan untuk meningkatkan efisiensi dan akurasi model. Metode feature selection yang digunakan yaitu pearson correlation coefficient dikarenakan pearson correlation dapat digunakan untuk variabel numerik [15]. Selain itu, pearson correlation secara efisien mampu mengatasi fitur bertipe campuran [16]. Dengan demikian, proses ini membantu mengurangi dimensi data, yang tidak hanya membuat model lebih efisien, tetapi juga mengurangi risiko overfitting. Pearson Correlation merupakan nilai korelasi yang memiliki rentang -1 hingga 1 yang dimana jika mendekati 1 (baik minus atau positif) yang memiliki arti indikasi kuat terhadap parameter lain [22]. Formula dari pearson adalah sebagai berikut yakni pada Persamaan (1) [23].

$$r_{xy} = \frac{n \sum xy - (\sum x)(\sum y)}{\sqrt{[n \sum x^2 - (\sum x)^2][n \sum y^2 - (\sum y)^2]}} \quad (1)$$

Dalam persamaan tersebut, x merepresentasikan variabel pertama yang meliputi atribut seperti usia (Age), etnis (Ethnicity), status sosial ekonomi (Socioeconomic Status), tingkat pendidikan (Education Level), hingga literasi kesehatan (Health Literacy). Sementara itu, y menggambarkan variabel kedua, yaitu hasil diagnosis, dan n menyatakan jumlah keseluruhan data pengamatan yang digunakan dalam analisis.

2.1.4 Split Data

Split data dilakukan untuk membagi dataset menjadi dua kelompok, yaitu data pelatihan (training) dan data pengujian (testing). Pada penelitian ini, digunakan pembagian dengan rasio 80:20, di mana 80% data digunakan untuk melatih model dan 20% sisanya digunakan untuk menguji kinerja model tersebut. Pendekatan ini merupakan praktik umum dalam machine learning untuk memastikan bahwa model tidak hanya dapat mempelajari pola dari data pelatihan, tetapi juga mampu melakukan prediksi yang baik terhadap data baru yang belum dikenalnya.

2.1.5 Klasifikasi Menggunakan KNN

Algoritma KNN merupakan algoritma supervised learning yang dapat diterapkan pada tugas classification maupun regression [19]. Metode KNN menempatkan objek dalam kategori berdasarkan seberapa dekat mereka dengan data pembelajaran [24]. Ini dilakukan dengan menemukan K objek dalam data latih yang paling dekat dengannya pada data baru atau data uji [25]. Tahapan-tahapan ini mencakup:

a. Menentukan parameter K

KNN berfungsi dengan mencari K objek yang paling dekat atau paling mirip dari data pelatihan dengan objek pada data baru atau data uji [26].

b. Hitung jarak data baru terhadap data lama

Dalam metode KNN, pemilihan data dilakukan berdasarkan jarak atau kedekatan dengan tetangga terdekat [27]. Rumus Euclidean distance digunakan untuk menguji kedekatan data. Penerapan rumus ini pada tetangga terdekat adalah seperti yang ditunjukkan dalam Persamaan (2) [28].

$$D(a, b) = \sqrt{\sum (a_k - b_k)^2} \quad (2)$$

Pada rumus di atas, a merepresentasikan data uji (testing data), sedangkan b adalah data sampel (sample data) yang dibandingkan. Nilai D menunjukkan jarak antara kedua data tersebut. Simbol k merujuk pada jumlah dimensi data atau banyaknya atribut yang dihitung, dan setiap a_k serta b_k adalah nilai dari masing-masing variabel ke- k dalam data uji dan data sampel.

c. Distribusi kelas

Persamaan (3) adalah formula untuk menghitung distribusi kelas terhadap hasil jarak [29].

$$E(a, b) = \begin{cases} 1 & \text{if } a = b \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

Rumus pada Persamaan (3) digunakan untuk menentukan distribusi kelas berdasarkan hasil perhitungan jarak antar data. Dalam formula tersebut, nilai $E(a, b)$ akan bernilai 1 apabila nilai a sama dengan b . Di sini, a merujuk pada hasil klasifikasi dari data latih ke- i menggunakan metode $K=3$, sedangkan b adalah kelas yang menjadi acuan.

d. Probabilitas kelas

Formula untuk menghitung probabilitas pada setiap kelas adalah seperti yang ada pada Persamaan (4) [29]

$$p(c_j)_{(x_{tl})} = \frac{E(y_i, c_j)}{k} \quad (4)$$

Menghitung probabilitas kemunculan kelas ke- j dalam lingkungan data X_{tl} dengan $E(y_i, c_j)$ membagi yaitu banyaknya tetangga dari keseluruhan k tetangga terdekat yang termasuk dalam kelas c_j dengan k itu sendiri, yaitu jumlah tetangga terdekat yang diperhitungkan.

e. Output dan hasil diagnosis

Untuk perhitungan Output Probabilitas kelas terbesar menggunakan KNN adalah menggunakan Persamaan (5) [29]

$$y_1 = \arg \max \{p(c_1)_{(x_{tk})}, p(c_2)_{(x_{tk})}, \dots, p(m)_{(x_{tk})}\} \quad (5)$$

2.1.6 Evaluasi

Metode untuk mengevaluasi kinerja masalah klasifikasi dalam situasi di mana output dapat berupa 2 (dua) atau lebih kategori kelas adalah confusion matrix [30]. True Positive (TP) dan True Negative (TN) menunjukkan prediksi yang berhasil, sedangkan False Positive (FP) dan False Negative (FN) menunjukkan prediksi yang salah. Precision digunakan untuk menilai seberapa relevan prediksi terhadap kelas positif, sedangkan recall mengukur sejauh mana model mampu mengenali seluruh instance dari kelas positif. F1-score berperan dalam menyeimbangkan precision dan recall, yang menjadi sangat krusial ketika menghadapi dataset dengan distribusi kelas yang tidak seimbang [31]. Evaluasi model menggunakan confusion matrix terdapat 4 (empat) performa yang dapat dihitung yakni sebagai berikut [32] [33]:

- a. Akurasi mengukur seberapa akurat model yang dihasilkan dalam memprediksi kelas data [34]. Persamaan akurasi dapat dilihat pada Persamaan (6).

$$Akurasi = \frac{TP+TN}{TP+FN+FP+TN} \quad (6)$$

Nilai akurasi diperoleh dengan menghitung jumlah prediksi benar, yaitu True Positive (TP) dan True Negative (TN), dibandingkan dengan total seluruh prediksi.

- b. Precision adalah rasio antara jumlah dokumen teks yang relevan dengan keseluruhan dokumen teks yang telah dipilih. Persamaan precision dapat dilihat pada Persamaan (7).

$$Precision = \frac{TP}{TP+FP} \quad (7)$$

Di mana precision dihitung dengan membagi jumlah true positive (TP) dengan jumlah true positive (TP) ditambah false positive (FP).

- c. Recall (sensitivity), mengukur seberapa banyak dokumen teks relevan yang berhasil diambil dibandingkan dengan total keseluruhan dokumen teks relevan yang ada. Persamaan untuk recall dapat ditemukan pada Persamaan (8).

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

Dengan membagi jumlah true positive (TP) dengan jumlah true positive (TP) ditambah false negative (FN). recall berfokus pada kemampuan model dalam menemukan semua contoh positif yang sebenarnya.

- d. F-1 score menunjukkan perbandingan precision dan recall yang dibobotkan rata-rata. Persamaan (9) menunjukkan skor persamaan f1-score.

$$F1 - Score = 2 \times \frac{Recall \times Precision}{Recall + Precision} \quad (9)$$

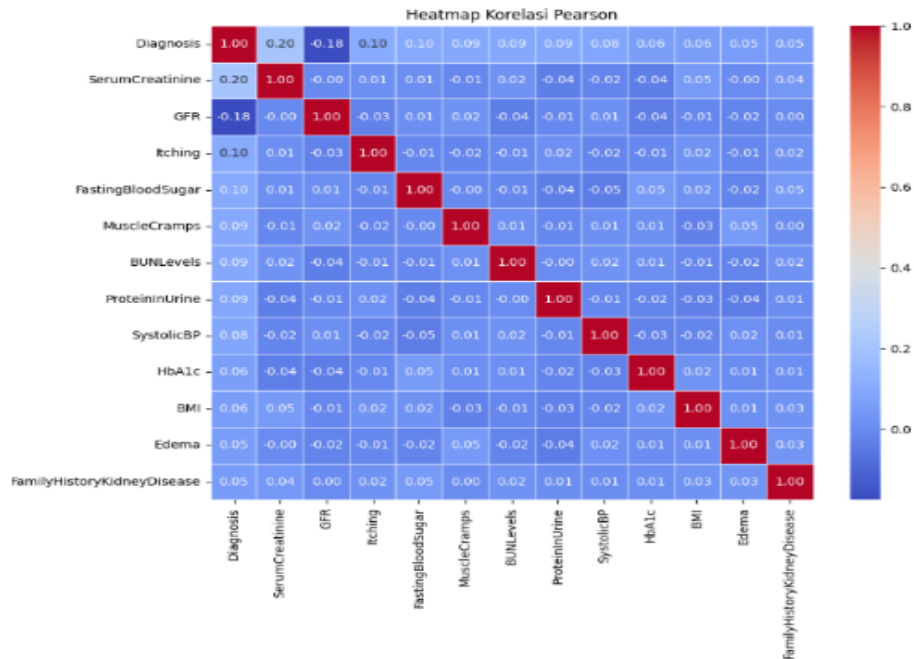
F1-score memberikan satu nilai tunggal untuk menilai keseimbangan antara precision dan recall. Matrix ini sangat berguna terutama ketika terdapat ketidakseimbangan kelas dalam dataset. Semakin mendekati angka 1, semakin baik keseimbangan antara akurasi prediksi positif dan kemampuan model dalam menangkap data positif yang sebenarnya, yang mencerminkan kinerja model yang stabil dan andal dalam berbagai kondisi klasifikasi.

3. HASIL DAN PEMBAHASAN

Penelitian ini bertujuan untuk mengklasifikasikan penyakit ginjal dengan menggunakan algoritma K-Nearest Neighbors (KNN) melalui dua pendekatan berbeda, yaitu dengan menerapkan feature selection dan tanpa menggunakan feature selection. Pendekatan ini dilakukan untuk mengevaluasi pengaruh seleksi fitur terhadap performa klasifikasi yang dihasilkan oleh algoritma KNN. Dalam implementasinya, digunakan beberapa library populer dalam ekosistem Python, seperti pandas untuk manipulasi dan analisis data dalam format tabular, matplotlib dan seaborn untuk visualisasi data dan hasil analisis, serta modul-modul dari scikit-learn yang digunakan untuk membangun dan mengevaluasi model pembelajaran mesin. Proses prapemrosesan data dilakukan dengan menggunakan MinMaxScaler dan StandardScaler dari sklearn.preprocessing untuk memastikan bahwa seluruh fitur memiliki skala yang seragam. Pembagian data menjadi data latih dan data uji dilakukan dengan fungsi train_test_split dari sklearn.model_selection. Untuk proses klasifikasi, digunakan KNeighborsClassifier dari sklearn.neighbors, yang bekerja dengan prinsip menghitung kedekatan antar data untuk menentukan kelas dari data yang tidak dikenal. Evaluasi terhadap kinerja model dilakukan dengan menggunakan metrik evaluasi seperti confusion matrix dan classification report yang disediakan oleh sklearn.metrics. Dataset yang digunakan terdiri dari 1659 entri dan 54 atribut, mencakup informasi demografis seperti PatientID, Age, Gender, dan Ethnicity, serta berbagai faktor yang berkaitan dengan gaya hidup, kondisi kesehatan, dan informasi medis lainnya, termasuk hasil diagnosis dan dokter yang menangani pasien. Hasil dari kedua pendekatan dibandingkan untuk menilai efektivitas penggunaan feature selection dalam meningkatkan akurasi dan efisiensi model klasifikasi berbasis KNN.

3.1 Klasifikasi Menggunakan KNN dengan Feature Selection

Klasifikasi dengan feature selection dilakukan setelah tahap normalisasi data menggunakan z-score normalization. Proses seleksi fitur diawali dengan menghitung korelasi pearson antara setiap fitur dan variabel target (Diagnosis), kemudian diurutkan berdasarkan nilai absolut tertinggi untuk mengabaikan korelasi lemah. Fitur dengan nilai korelasi absolut di atas 0,05 disaring sebagai kandidat utama. Selanjutnya, subset fitur yang relevan dipilih dari correlation matrix untuk membentuk correlation_subset, yang berisi fitur-fitur dengan korelasi signifikan terhadap target. Visualisasi dilakukan menggunakan seaborn.heatmap untuk menampilkan hubungan antar fitur secara visual, dengan parameter tambahan seperti annot=True, cmap='coolwarm', dan fmt=".2f" untuk meningkatkan keterbacaan. Hasil visualisasi yang ditampilkan pada Gambar 2 untuk mempermudah pemahaman terhadap keterkaitan antar fitur yang telah diseleksi.



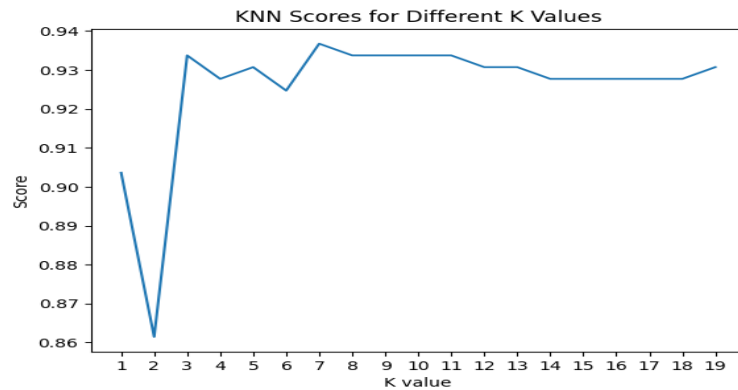
Gambar 2. Output Heatmap Korelasi Pearson

Output heatmap korelasi Pearson pada Gambar 2 menampilkan hubungan antara 12 fitur dalam dataset, seperti Diagnosis, SerumCreatinine, GFR, FastingBloodSugar, dan lainnya. Nilai korelasi pada setiap sel menunjukkan sejauh mana dua fitur saling berhubungan. Korelasi positif mendekati 1 menunjukkan hubungan yang kuat dan bergerak ke arah yang sama, sedangkan korelasi negatif mendekati -1 menunjukkan hubungan yang kuat tetapi bergerak ke arah berlawanan. Dalam heatmap ini, sebagian besar hubungan antar fitur memiliki korelasi lemah atau mendekati nol, seperti antara Diagnosis dan GFR (-0,18) atau antara ProteinInUrine dan Edema (0,02), yang menunjukkan bahwa fitur-fitur tersebut tidak saling bergantung secara signifikan. Hubungan moderat terlihat antara Diagnosis dan SerumCreatinine (0,20), yang mengindikasikan potensi relevansi fitur tersebut dalam analisis

atau pemodelan. Warna dalam heatmap membantu mengidentifikasi pola hubungan, di mana warna merah tua menunjukkan korelasi kuat (biasanya pada diagonal karena fitur selalu berkorelasi sempurna dengan dirinya sendiri), sedangkan warna biru pucat menunjukkan korelasi lemah atau tidak signifikan. Secara keseluruhan, heatmap ini memberikan wawasan awal untuk memahami hubungan antar fitur dan membantu memilih fitur-fitur yang relevan untuk pemodelan lebih lanjut.

Setelah proses normalisasi selesai dilakukan, tahap selanjutnya adalah melakukan pemilihan fitur dari data yang telah dinormalisasi tersebut. Proses ini menghasilkan sebuah Dataframe yang hanya berisi fitur-fitur yang dianggap relevan untuk proses klasifikasi, seperti Diagnosis, SerumCreatinine, GFR, dan fitur-fitur penting lainnya. Data yang telah tersaring ini kemudian digunakan dalam tahap pembagian data (split data) menjadi dua bagian, yaitu data pelatihan (training data) dan data pengujian (testing data). Pada proses ini, digunakan rasio pembagian sebesar 80:20, artinya 80% dari keseluruhan data digunakan untuk melatih model, sedangkan 20% sisanya digunakan untuk menguji performa model. Seperti yang dapat dilihat pada Gambar 3, setelah data selesai dipersiapkan melalui tahap normalisasi dan seleksi fitur, data tersebut kemudian dimasukkan ke dalam model klasifikasi K-Nearest Neighbor (KNN). Nilai K yang digunakan dalam model ini berperan penting dalam menentukan akurasi awal dari proses klasifikasi yang dilakukan.

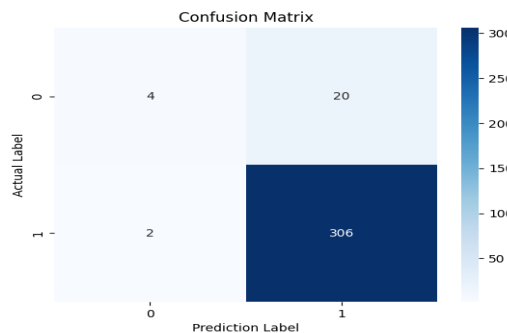
Berikut merupakan Gambar 3 yang memperlihatkan grafik hubungan antara variasi nilai K dengan skor akurasi yang didapatkan. Terobservasi bahwa skor akurasi mengalami perubahan pada setiap nilai K, dengan puncak akurasi tercapai pada nilai K tertentu.



Gambar 3. Score Akurasi KNN Dengan Feature selection Berdasarkan Nilai K

Berdasarkan Gambar 3, dapat terlihat bahwa nilai K (1 hingga 19) memiliki nilai akurasi yang berbeda, tetapi nilai K=3 adalah yang paling akurat, dengan akurasi 93.67%. Ini menunjukkan bahwa model KNN mencapai tingkat akurasi tertinggi sebesar 93.67% pada data pengujian, yang diperoleh dari nilai K tertentu. Sehingga dilakukan inisialisasi model KNN dengan menggunakan nilai K=3. Dalam prosesnya `_pred = knn.predict(X_test)` digunakan untuk melakukan prediksi terhadap data pengujian `X_test` menggunakan model KNN yang telah dilatih sebelumnya. Di sini, `X_test` adalah matriks yang berisi fitur-fitur dari data pengujian yang tidak diketahui label targetnya. Model menggunakan data pelatihan yang telah disimpan untuk menghitung jarak antara sampel dalam `X_test` dengan sampel dalam `X_train` dan menentukan prediksi berdasarkan tetangga terdekat.

Untuk menilai performa model KNN setelah melalui proses feature selection, dilakukan evaluasi menggunakan confusion matrix. Visualisasi dari confusion matrix tersebut ditampilkan pada Gambar 4, yang menggambarkan distribusi hasil prediksi terhadap data aktual dalam empat kategori utama (TP, FP, TN, FN).



Gambar 4. Confusion Matrix KNN Dengan Feature selection

Berdasarkan Gambar 4 confusion matrix, diperoleh informasi bahwa model memiliki akurasi yang sangat tinggi dalam mendeteksi data kelas 1, di mana sebanyak 306 data berhasil diprediksi dengan benar sebagai kelas 1. Sementara itu, hanya terdapat 4 data dari class 0 yang berhasil diklasifikasikan dengan benar. Terdapat kesalahan dalam mendeteksi kelas 0, di mana 20 data dari class 0 justru salah diprediksi sebagai class 1. Selain itu, ditemukan

pula 2 data dari kelas 1 yang keliru diprediksi sebagai class 0. Hasil ini menunjukkan performa model yang cukup baik namun masih terdapat sedikit kesalahan dalam klasifikasi antar kelas.

Secara keseluruhan, model menunjukkan kinerja yang sangat baik untuk mendeteksi class 1 (mengandung sangat sedikit FN), namun performanya kurang baik dalam mendeteksi kelas 0 (dengan jumlah FP yang cukup tinggi). Hal ini dapat mengindikasikan ketidakseimbangan data atau bias model terhadap kelas mayoritas. Selanjutnya adalah melakukan evaluasi menggunakan empat matrix evaluasi confusion matrix didapatkan accuracy 93.37%, precision 91.9%, recall 93.37%, dan F1-score 91.48%. Persentase prediksi model yang benar ditunjukkan oleh nilai akurasi dan nilai presisi. Laporan klasifikasi model dengan pemilihan fitur ditunjukkan pada Gambar 5.

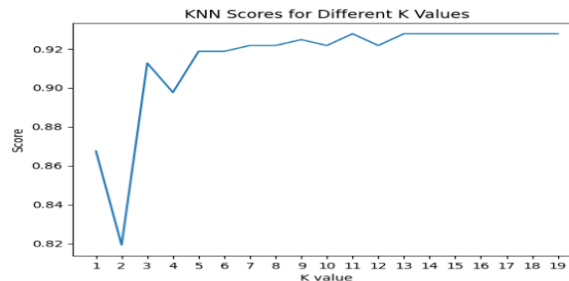
	precision	recall	f1-score	support
0	0.6667	0.1667	0.2667	24
1	0.9387	0.9935	0.9653	308
accuracy			0.9337	332
macro avg	0.8027	0.5801	0.6160	332
weighted avg	0.9190	0.9337	0.9148	332

Gambar 5. Laporan Klasifikasi Model Dengan Feature selection

Gambar 5 menampilkan laporan klasifikasi yang memuat nilai precision, recall, dan f1-score untuk masing-masing kelas (0 dan 1). Terlihat bahwa model memiliki kinerja lebih baik pada kelas 1 dibandingkan kelas 0. Selain itu, ditampilkan pula nilai rata-rata macro dan weighted, dengan akurasi keseluruhan mencapai 93,37%. Nilai macro average dihitung tanpa mempertimbangkan distribusi jumlah sampel antar kelas, sedangkan weighted average memperhitungkannya. Hasil ini mengonfirmasi adanya ketidakseimbangan data antar kelas.

3.2 Klasifikasi Menggunakan KNN Tanpa Feature Selection

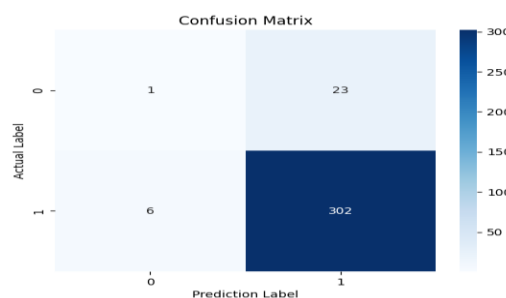
Klasifikasi menggunakan KNN tanpa feature selection diterapkan dengan terlebih dahulu melakukan tahap data preparation, di mana proses normalisasi data dilakukan menggunakan metode z-score normalization untuk memastikan setiap fitur berada pada skala yang sama. Setelah normalisasi, dataset dibagi menjadi data latih dan data uji dengan rasio 80:20 untuk memastikan evaluasi model yang adil dan representatif. Data yang telah diproses kemudian dimasukkan ke dalam model KNN, dan proses klasifikasi dilakukan dengan mengevaluasi akurasi model berdasarkan variasi nilai K untuk menentukan nilai K terbaik yang memberikan performa optimal..



Gambar 6. Score Akurasi KNN Tanpa Feature selection Berdasarkan Nilai K

Berikut merupakan Gambar 6 yang memperlihatkan grafik hubungan antara variasi nilai K dengan skor akurasi yang didapatkan tanpa menggunakan feature selection. Terobservasi bahwa skor akurasi mengalami perubahan pada setiap nilai K, dengan puncak akurasi tercapai pada nilai K tertentu.

Gambar 6 menunjukkan bahwa nilai K (1 hingga 19) memiliki nilai akurasi yang berbeda, tetapi nilai K=3 memiliki nilai akurasi tertinggi sebesar 92.77%, menunjukkan bahwa model KNN mencapai tingkat akurasi tertinggi sebesar 92.77% pada data pengujian. Karena nilai K=3 menunjukkan nilai akurasi terbaik, model KNN dimulai dengan nilai K=3. Model yang dibangun tersebut dilakukan evaluasi menggunakan confusion matrix yang hasilnya terlihat pada Gambar 7.



Gambar 7. Confusion Matrix KNN Tanpa Feature selection

Berdasarkan Gambar 7 confusion matrix, performa model ditunjukkan melalui empat elemen utama. True Positive (TP) mencatat sebanyak 302 data dari kelas 1 yang berhasil diprediksi dengan benar sebagai kelas 1, menandakan tingkat akurasi yang tinggi dalam mendeteksi kelas tersebut. True Negative (TN) hanya berjumlah 1 data dari kelas 0 yang tepat diklasifikasikan. Sementara itu, terdapat 23 data dari kelas 0 yang salah diprediksi sebagai kelas 1 (False Positive/FP), mengindikasikan adanya kesalahan dalam prediksi kelas 0. Selain itu, sebanyak 6 data dari kelas 1 salah diprediksi sebagai kelas 0 (False Negative/FN).

Berdasarkan hasil evaluasi yang ditunjukkan oleh Gambar 7 secara keseluruhan, model menunjukkan performa yang sangat baik untuk mendeteksi kelas 1 (dengan sangat sedikit FN), namun performanya kurang baik dalam mendeteksi kelas 0 (dengan jumlah FP yang cukup tinggi). Hal ini dapat mengindikasikan ketidakseimbangan data atau bias model terhadap kelas mayoritas. Evaluasi model dilanjutkan dengan melakukan evaluasi menggunakan empat metrix dengan hasil yang didapatkan accuracy 91.27%, precision 87.24%, recall 91.27%, dan F1-score 88.99%. Persentase prediksi model yang benar ditunjukkan oleh nilai akurasi dan nilai presisi. Evaluasi kinerja model tanpa feature selection dilakukan menggunakan laporan klasifikasi, yang mencakup matrix precision, recall, f1-score, serta jumlah sampel (support) untuk tiap kelas. Hasil evaluasi ini dapat dilihat pada Gambar 8.

	precision	recall	f1-score	support
0	0.1429	0.0417	0.0645	24
1	0.9292	0.9805	0.9542	308
accuracy			0.9127	332
macro avg	0.5360	0.5111	0.5094	332
weighted avg	0.8724	0.9127	0.8899	332

Gambar 8. Laporan Klasifikasi Model Dengan Feature selection

Output pada Gambar 8 merupakan laporan klasifikasi yang lebih terperinci untuk masing-masing class 0 serta 1. Kelas 1 menunjukkan kinerja yang lebih baik daripada kelas 0. Selain itu, laporan ini mencakup rata-rata macro dan weighted untuk seluruh dataset, serta accuracy yang mencapai 91.27%. Rata-rata macro menghitung nilai rata-rata tanpa memperhatikan ketidakseimbangan kelas, sedangkan weighted avg menghitung rata-rata dengan mempertimbangkan proporsi sampel di setiap kelas. Berikut pada Tabel 1 merupakan hasil akhir evaluasi model yang diterapkan.

Tabel 1. Hasil Akhir Evaluasi

Model	Akurasi	Precision (Weighted Average)	Recall (Weighted Average)	F1-Score (Weighted Average)
KNN Dengan Feature selection	93.37%	91.9%	93.37%	91.48%
KNN Tanpa Feature selection	91.27%	87.24%	91.27%	88.99%

Berdasarkan Tabel 1, evaluasi menunjukkan bahwa model dengan feature selection memiliki performa lebih baik dibandingkan tanpa feature selection. Model ini mencapai accuracy 93.37% dengan precision 91.9%, recall 93.37%, dan F1-score 91.48%, Hasil ini terlihat lebih seimbang antara precision dan recall, yang mengindikasikan bahwa model dapat mengidentifikasi kedua kelas (Positif dan Negatif) dengan baik. F1-score yang tinggi (91.48%) juga menunjukkan keseimbangan yang baik antara presisi dan juga recall. Sebaliknya, model tanpa feature selection memiliki accuracy 91.27%, namun precision untuk kelas negatif sangat rendah (14.29%). Hal ini menunjukkan bahwa model kesulitan dalam memprediksi kelas Negatif secara akurat. Recall untuk kelas 1 (Positif) cukup tinggi (98.05%), namun precision secara keseluruhan untuk model ini lebih rendah dibandingkan dengan model yang menggunakan feature selection. F1-score yang lebih rendah (88.99%) juga mencerminkan ketidakseimbangan prediksi antara kedua kelas. Secara keseluruhan, feature selection terbukti meningkatkan akurasi dan stabilitas model secara signifikan.

4. KESIMPULAN

Berdasarkan temuan penelitian yang telah dilakukan, hasil classification penyakit ginjal kronis dengan algoritma KNN dapat disimpulkan, menunjukkan bahwa model ini dapat melakukan klasifikasi dengan sangat akurat. Model KNN dapat membantu membedakan kelas penyakit ginjal kronis berdasarkan atribut yang tersedia dalam dataset, dan dapat memberikan hasil yang cukup untuk digunakan dalam klasifikasi medis. Dalam melakukan implementasi tersebut dilakukan dengan bantuan software Google collaboratory menggunakan bahasa pemrograman Python. Dataset yang digunakan dalam implementasi yaitu chronic kidney disease yang diambil dari platform Kaggle, selanjutnya dilakukan preprocessing data dengan melakukan pembersihan data untuk mengatasi missing value, outlier, dan melakukan normalisasi data. Selanjutnya dilakukan penerapan feature selection



menggunakan pearson correlation coefficient. Selanjutnya, data dibagi menjadi 80% data pelatihan dan 20% data pengujian. Kemudian, algoritma KNN diimplementasikan, dan terakhir, evaluasi model dilakukan menggunakan confusion matrix yang terdiri dari nilai akurasi, precision, recall, dan skor f1-score. Penggunaan feature selection terbukti dapat meningkatkan akurasi dan kinerja model klasifikasi. Model yang menggunakan feature selection mencapai akurasi sebesar 93,37%, dengan precision 91,9%, recall 93,37%, dan F1-score 91,48%. Sedangkan model yang tidak menggunakan feature selection mencapai akurasi 91,27%, dengan precision 87,24%, recall 91,27%, dan F1-score 88,99%. Pemilihan fitur yang relevan menjadikan akurasi pada model menjadi lebih baik dalam memprediksi kelas penyakit ginjal kronis.

REFERENCES

- [1] Q. A'yunyah et al., "Implementasi Algoritma Naïve Bayes Classifier (NBC) untuk Klasifikasi Penyakit Ginjal Kronik," *J. Sist. Komput. dan Inform.*, vol. 4, no. 1, hal. 72, 2022, doi: 10.30865/json.v4i1.4781.
- [2] C. J. G. Paath, G. Masi, and F. Onibala, "Study Cross Sectional : Dukungan Keluarga Dengan Kepatuhan Hemodialisa Pada Pasien Gagal Ginjal Kronis," *J. Keperawatan*, vol. 8, no. 1, hal. 106, 2020, doi: 10.35790/jkp.v8i1.28418.
- [3] A. T. Akbar, N. Yudistira, and A. Ridok, "Identifikasi Gagal Ginjal Kronis dengan Mengimplementasikan Metode Support Vector Machine beserta K-Nearest Neighbour (SVM-KNN)," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 10, no. 2, hal. 301–308, 2023, doi: 10.25126/jtiik.20231026059.
- [4] M. Rizal, M. Z. Syahaf, S. R. Priyambodo, and Y. Rhamdani, "Optimasi Algoritma Naïve Bayes Menggunakan Forward Selection Untuk Klasifikasi Penyakit Ginjal Kronis," *Naratif J. Nas. Riset, Apl. dan Tek. Inform.*, vol. 5, no. 1, hal. 71–80, 2023, doi: 10.53580/naratif.v5i1.200.
- [5] V. Wulandari, W. J. Sari, Z. Alfian, L. Legito, and T. Arifianto, "Implementasi Algoritma Naïve Bayes Classifier dan K-Nearest Neighbor untuk Klasifikasi Penyakit Ginjal Kronik," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 4, no. 2, hal. 710–718, 2024, doi: 10.57152/malcom.v4i2.1229.
- [6] N. P. Nugraha, R. Azim, S. Z. Daffa, and P. S. Ningayu, "Perbandingan Akurasi Metode Naïve Bayes dan Metode KNN untuk Memprediksi Gagal Ginjal Kronis," *J. Rekayasa Elektro Sriwij.*, vol. 5, no. 1, hal. 1–10, 2023, doi: 10.36706/jres.v5i1.63.
- [7] R. Salsabilla, "Data Kemenkes: Lebih dari 700 Ribu Orang RI Menderita Ginjal Kronis," *cnbcindonesia*, 2024. <https://www.cnbcindonesia.com/lifestyle/20240116111340-33-506206/data-kemenkes-lebih-dari-700-ribu-orang-ri-menderita-ginjal-kronis> [Accessed: 14-Feb-2025].
- [8] S. N. Chotimah and A. R. Rozzaqi, "Klasifikasi Diagnosis Penyakit Ginjal Kronis Dengan Menerapkan Konsep Algoritma Naïve Bayes," *JIPETIKJurnal Ilm. Penelit. Teknol. Inf. Komput.*, vol. 4, no. 1, hal. 8–15, 2023, doi: 10.26877/jipetik.v4i1.16174.
- [9] E. V. Tjahjadi and B. Santoso, "Klasifikasi Malware Menggunakan Teknik Machine Learning," *Jurnal Inovasi Ilmu Komputer (JIJK)*, vol. 2, no. 1, hal. 60–70, Desember 2023.
- [10] M. F. Akbarollah, W. Wiyanto, D. Ardiatma, and A. T. Zy, "Penerapan Algoritma K-Nearest Neighbor Dalam Klasifikasi Penyakit Jantung," *J. Comput. Syst. Informatics*, vol. 4, no. 4, hal. 850–860, 2023, doi: 10.47065/josyc.v4i4.4071.
- [11] N. Fatimah Indrianti, A. Kania Ningsih, and R. Ilyas, "Implementasi Data Mining Untuk Klasifikasi Penyakit Gagal Ginjal Kronis Menggunakan Metode K-Nearest Neighbor," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 8, no. 2, hal. 2255–2260, 2024, doi: 10.36040/jati.v8i2.9464.
- [12] M. Knn, S. Kasus, R. S. Di, K. A. B. Gresik, P. Aisyiyah, and R. Devi, "Klasifikasi penyakit gagal ginjal kronis dengan metode knn (studi kasus rs di kab gresik)," vol. 9, no. 3, hal. 1739–1748, 2024.
- [13] M. F. Ahmadi and N. F. Abdi, "Klasifikasi Penyakit Ginjal Kronis (CKD) dengan Algoritma KNN dan Decision Tree ID3," *Journal of Informatics and Advanced Computing (JIAC)*, vol. 5, no. 2, hal. 52–57, November 2024, doi: 10.35814/jiac.v5i2.7189.
- [14] D. Leni, A. Dwiherzandis, R. Sumiati, H. Haris, and S. Afriyani, "Seleksi Fitur Berdasarkan Korelasi Pearson dalam Pemodelan Efisiensi Energi Bangunan," *Tek. Sains J. Ilmu Tek.*, vol. 8, no. 2, hal. 103–115, 2023, doi: 10.24967/teksis.v8i2.2525.
- [15] J. Brownlee, "How to Choose a Feature Selection Method For Machine Learning," *Machine Learning Mastery*, 2020. [Online]. Available: <https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/>. [Accessed: 15-Feb-2025].
- [16] M. B. Hanif and G. G. Setiaji, "Meningkatkan Kinerja Decision Tree C4.5 dengan Seleksi Fitur Korelasi Pearson pada Deteksi Penyakit Diabetes," *Indones. J. Comput. Sci.*, vol. 11, no. 2, hal. 685–695, 2022, doi: 10.33022/ijcs.v11i2.3087.
- [17] S. Ratnasingam and J. Muñoz-Lopez, "Distance Correlation-Based Feature Selection in Random Forest," *Entropy*, vol. 25, no. 9, 2023, doi: 10.3390/e25091250.
- [18] C. Yanasari and T. Arifin, "Implementasi Algoritma K-Nearest Neighbor Untuk Klasifikasi Penerimaan Beasiswa Program Indonesia Pintar," *J. Sist. Inf. dan Ilmu Komput.*, vol. 1, no. 4, hal. 178–194, 2023.
- [19] S. D. Prasetyo, S. S. Hilabi, and F. Nurapriani, "Analisis Sentimen Relokasi Ibukota Nusantara Menggunakan Algoritma Naïve Bayes dan KNN," *J. KomtekInfo*, vol. 10, hal. 1–7, 2023, doi: 10.35134/komtekinfo.v10i1.330.
- [20] G. Gerry, B. S. Santoso, and J. P. Tanjung, "Classification of Wheat Seeds Using Neural Network Backpropagation Algorithm," *J. Informatics Telecommun. Eng.*, vol. 4, no. 2, hal. 335–342, 2021, doi: 10.31289/jite.v4i2.4449.
- [21] M. Sciences and K. Saikumar, "Optimized Building of Machine Learning Models for The Diagnosis of Thyroid Problems in Human Beings," *Int. J. Pharm. Res.*, vol. 12, no. Special Issue 3, 2020, doi: 10.31838/ijpr/2020.sp3.058.
- [22] A. Yilmaz, "Prediction of Type 2 Diabetes Mellitus Using Feature Selection-Based Machine Learning Algorithms," *Heal. Probl. Civiliz.*, vol. 16, no. 2, pp. 128–139, 2022, doi: 10.5114/hpc.2022.114541.
- [23] M. Miftahuddin, A. Pratama, and I. Setiawan, "Hubungan Antara Kelembaban Relatif Dengan Beberapa Variabel Iklim Dengan Pendekatan Korelasi Pearson di Samudera Hindia," *J. Siger Mat.*, vol. 2, no. 1, hal. 25–33, 2021, doi: 10.23960/jsm.v2i1.2753.



- [24] A. N. Dzulhijjah, S. Anraeni, and S. Sugiarti, “Klasifikasi Kematangan Citra Labu Siam Menggunakan Metode KNN (K-Nearest Neighbor) Dengan Ekstraksi Fitur HSV (Hue, Saturation, Value),” *Bul. Sist. Inf. dan Teknol. Islam*, vol. 2, no. 2, hal. 103–110, 2021, doi: 10.33096/busiti.v2i2.808.
- [25] P. D. Rinanda, B. Delvika, S. Nurhidayarnis, N. Abror, and A. Hidayat, “Perbandingan Klasifikasi Antara Naive Bayes dan K-Nearest Neighbor Terhadap Resiko Diabetes pada Ibu Hamil,” *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 2, no. 2, hal. 68–75, 2022, doi: 10.57152/malcom.v2i2.432.
- [26] A. Maslan, K. M. Bin Mohamad, and F. B. Mohd Foozy, “Feature selection for DDoS detection using classification machine learning techniques,” *IAES Int. J. Artif. Intell.*, vol. 9, no. 1, pp. 137–145, 2020, doi: 10.11591/ijai.v9.i1.pp137-145.
- [27] M. A. Bahrudin, R. K. Niswatin, and L. S. Wahyuniar, “Sistem Pendukung Keputusan Penentuan Ekstrakurikuler Siswa SMK Al-Asy’ariyah Prambon Dengan Metode K-Nearest Neighbor (KNN),” dalam *Prosiding Seminar Nasional Inovasi Teknologi Terapan (SNITT)*, vol. 1, hal. 185–192, Agustus 2021.
- [28] J. Nasir, R. Saputra, G. Efendi, A. Zahmi, and Y. L. Setiawan, “K-Nearest Neighbor untuk Frasa Guna Mendukung Keputusan dalam Mencari Guru Terbaik,” *J. Ilmu Komput. dan Agri-Informatika*, vol. 9, no. 1, hal. 13–22, 2022, doi: 10.29244/jika.9.1.13-22.
- [29] S. R. Gongada, M. Chakravarthy, and B. Mangu, “Power system contingency classification using machine learning technique,” *Bull. Electr. Eng. Informatics*, vol. 11, no. 6, pp. 3091-3098, 2022, doi: 10.11591/eei.v11i6.4031.
- [30] I. G. I. Sudipa et al., *METODE PENELITIAN BIDANG ILMU INFORMATIKA (Teori & Referensi Berbasis Studi Kasus)*. PT. Sonpedia Publishing Indonesia, 2023.
- [31] V. No, K. A. Wulandari, A. Nugraha, A. Luthfiarta, and L. R. Nisa, “Edumatic: Jurnal Pendidikan Informatika Peningkatan Akurasi Deteksi Dini Penyakit Parkinson melalui Pendekatan Ensemble Learning dan Seleksi Fitur Optimal,” vol. 8, no. 2, hal. 575–584, 2024, doi: 10.29408/edumatic.v8i2.27788.
- [32] K. T. Pamungkas, L. Aridinanti, and W. Wibowo, “Analisis Sentimen Pelaporan Masyarakat di Situs Media Centre Surabaya dengan Naive Bayes Classifier,” *J. Sains dan Seni ITS*, vol. 11, no. 2, 2022, doi: 10.12962/j23373520.v11i2.72566.
- [33] K. Wabang, O. D. Nurhayati, and Farikhin, “Application of The Naive Bayes Classifier Algorithm to Classify Community Complaints,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 5, pp. 872–876, 2022, doi: 10.29207/resti.v6i5.4498.
- [34] H. Suparwito, R. Gunawan, I. Binanto, R. A. Kumalasanti, and W. Widyastuti, *Pengantar Pembelajaran Mesin Menggunakan Bahasa Pemrograman Python*. Sanata Dharma University Press, 2023.