JSRCS Journal of Students' Research in Computer Science

e-ISSN: 2722 – 290X
Vol. 6 No. 2 (November 2025), Page: 135 – 148
https://doi.org/10.31599/tvt7s936
Available Online at http://ejurnal.ubharajaya.ac.id/index.php/JSRCS

Model Prediksi Kondisi Kesehatan dari Data *Medical Check-Up* Menggunakan *K-Nearest Neighbors* dan *Decision Tree*

Tata Arya Cahyaaty [1], Herlawati Herlawati [1,*], Andy Achmad Hendhar Setiawan [1]

* Korespondensi: e-mail: herlawati@ubharajaya.ac.id

[1] Informatika, Fakultas Ilmu Komputer; Universitas Bhayangkara Jakarta Raya, Jl. Perjuangan No. B1, Marga Mulya, Bekasi utara, Bekasi, Jawa Barat 17143, Telp/Fax: (021) 88955882; e-mail: cahyaatytataarya@gmail.com, herlawati@ubharajaya.ac.id, andy.achmad@dsn.ubharajaya.ac.id

Submitted : **2 September 2025**
Revised : **1 Oktober 2025**
Accepted : **4 November 2025**
Published : **30 November 2025**

Abstract

Medical Check-Up (MCU) is an essential procedure for the early detection of health disorders. However, manual analysis of MCU results requires time and may be subject to the interpretation of medical personnel. This study aims to develop an automatic classification system to predict health conditions based on MCU results using the K-Nearest Neighbors (KNN) and Decision Tree algorithms. The MCU data used includes blood pressure, body temperature, heart rate, as well as heart and blood pressure assessments. The models were trained and evaluated using the CRISP-DM methodology. The results show that the Decision Tree achieved an accuracy of 91.31%, while KNN achieved an accuracy of 89.75%. This system is implemented as a web-based application with a simple user interface to support the early diagnosis process at RS EMC Cibitung.

Keywords: *Decision Tree, K-Nearest Neighbors, Medical Check-Up, CRISP-DM*

Abstrak

Medical Check-Up (MCU), merupakan prosedur penting dalam deteksi dini gangguan kesehatan. Namun, analisis manual terhadap hasil MCU memerlukan waktu dan subjektivitas tenaga medis. Penelitian ini bertujuan membangun system klasifikasi otomatis untuk memprediksi kondisi Kesehatan berdasarkan hasil MCU menggunakan algoritma *k-Nearest Neighbors* (KNN) dan *Decesion Tree*. Data MCU yang digunakan meliputi tekanan darah, suhu tubuh, denyut nadi, serta Kesimpulan jantung dan tekanan darah. Model dilatih dan diuji menggunakan *CRISP-DM*. Hasil penelitian menunjukkan bahwa *Decision Tree* memiliki akurasi 91,31% dan KNN memiliki akurasi 89,75%. System ini diimplementasikan dalam bentuk aplikasi berbasis web dengan antarmuka sederhanan, untuk mendukung proses diagnosis awal di RS EMC Cibitung.

Kata kunci: *Decision Tree, K-Nearest Neighbors, Medical Check-Up, CRISP-DM*

1. Pendahuluan

Medical Check-Up (MCU) di RS EMC Cibitung menghasilkan data vital seperti tekanan darah, suhu, dan denyut jantung. Analisis manual atas data ini belum efisien dan beresiko subjektif. Penelitian ini bertujuan mengembangkan sistem prediksi berbasis *Machine Learning* Menggunakan algoritma KNN dan *Decision Tree* yang terbukti efektif dalam klasifikasi data

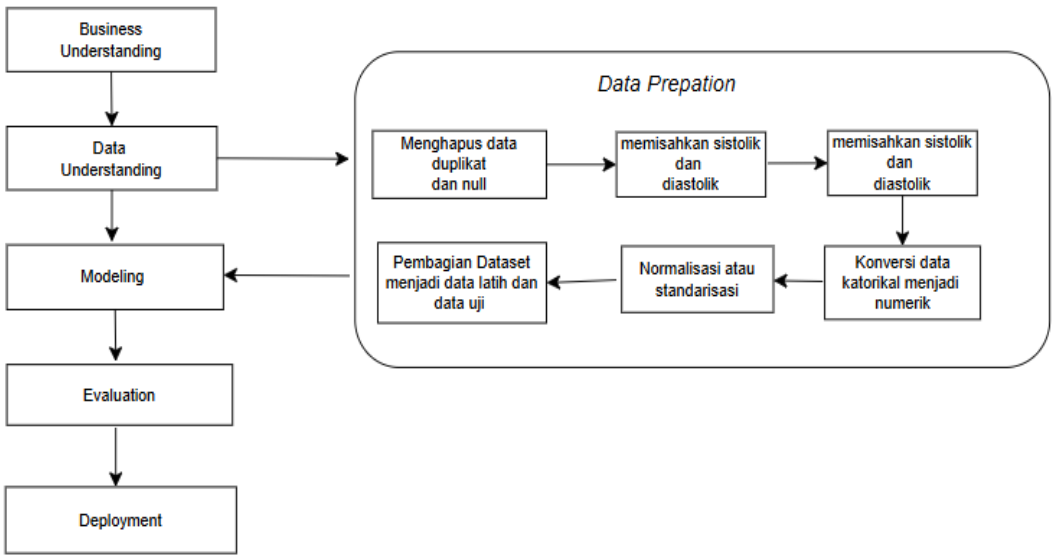
numerik dan katagorikal Dengan kemajuan teknologi informasi, metode telah banyak dimanfaatkan dalam bidang kesehatan, khususnya dalam klasifikasi dan prediksi data medis. Dua algoritma yang populer dan efektif dalam pengolahan data jenis ini adalah *K-Nearest Neighbors* (KNN) dan *Decision Tree*. Keduanya dikenal memiliki keunggulan dalam memproses data numerik maupun kategorikal secara cepat dan akurat, serta mudah diimplementasikan dan mampu menangani data yang mengandung nilai ekstrem (*outlier*) maupun distribusi yang tidak seragam.

Beberapa penelitian sebelumnya menunjukkan efektivitas metode klasifikasi dalam bidang medis. Andriani et al. (2020) menggunakan algoritma KNN untuk mengklasifikasikan hasil pemeriksaan laboratorium darah dan memperoleh akurasi sebesar 87%. Sementara itu, Wibowo dan Sari (2021) menerapkan Decision Tree dalam pendeteksian potensi hipertensi berdasarkan tekanan darah dan riwayat keluarga, dengan akurasi mencapai 89%. Temuan-temuan tersebut menunjukkan bahwa metode klasifikasi mampu memberikan hasil prediktif yang akurat dalam konteks kesehatan (Adiputra et al., 2021)

Tinjauan pustaka ini menjadi dasar teoritis dalam penerapan algoritma KNN dan Decision Tree untuk mengklasifikasikan hasil pemeriksaan *Medical Check-Up*, dengan tujuan mendeteksi potensi kondisi kesehatan pasien di RS EMC Cibitung secara lebih dini dan efisien.

2. Metode Penelitian

Penelitian ini menggunakan pendekatan *CRISP-DM* (*Cross Industry Standard Process for Data Mining*) sebagai kerangka kerja dalam pembangunan sistem klasifikasi kondisi kesehatan berdasarkan data *Medical Check-Up* (MCU). Metodologi ini dipilih karena terstruktur dan cocok untuk proyek berbasis data mining. *CRISP-DM* terdiri dari enam tahapan utama.



Sumber: Hasil Penelitian (2025)

Gambar 1. Kerangka Penelitian

2.1. *Machine learning*

Machine Learning merupakan cabang dari kecerdasan buatan yang berfokus pada pengembangan algoritma dan model yang memungkinkan komputer untuk belajar dan mebuat prediksi atau Keputusan tanpa perlu deprogram secara eksplisit. Dengan kata lain mengikuti instruksi yang diberikan, system *machine learning* berjalan dari data yang diberikan agar bisa jalan sesuai perintah yang diinginkan (Dietterich, 2005).

2.2. *Data Mining*

Data Mining merupakan proses penggalian informasi atau pola tersembunyi dari Kumpulan data dalam jumlah besar. Dalam bidang Kesehatan, *data mining* digunakan untuk menganalisis hasil pemeriksaan medis guna mengidentifikasi pola tertentu yang dapat menunjukkan indikasi penyakit. Beberapa Teknik umum dalam *Data Mining* meliputi klasifikasi, clustering, asosiasi, dan deteksi anomali. Dengan teknik ini, data hasil MCU dapat diolah untuk mengungkap kemungkinan kondisi kesehatan pasien (Pei & Tong, 2015).

2.3. Metode Klasifikasi

Metode klasifikasi dalam *data mining* bertujuan untuk memetakan data ke dalam kelas-kelas tertentu. Dua metode yang digunakan dalam penelitian ini adalah *K-Nearest Neighbors* (KNN) dan *Decision Tree.*

2.3.1. *K-Nearest Neighbors* (*KNN*)

KNN merupakan algoritma klasifikasi yang bekerja dengan cara mecari sejumlah data tetangga terdekat (berdasarkan jarak, biasanya *Euclidean*) dari data yang akan diklasifikasi. Nilai K menentukan jumlah tetangga yang dihiting. Algoritma ini cocok untuk dataset berukuran kecil hingga menengah. Namun, kelemahan algoritma ini terletak pada sensitivitas terhadap data yang tidak ternormalisasi serta performa yang menurun pada dataset besar.(Hasran, 2020)

$$d\ (x,y) = \sqrt{\sum_{i=1}^{n}(x_{i-}\ y_i)^2} \qquad (1)$$

Keterangan:

d (x, y) = Jarak *Euclidean* antara dua titik data x dan y

xi, yi = nilai fitur ke-I pada data x dan y

n = jumlah total fitur

2.3.2. *Decision Tree*

Decision Tree merupakan algoritma klasifikasi yang menggunakan struktur pohon untuk mengambil keputusan. Setiap node pada pohon mewakili atribut yang diuji, cabang merepresentasikan hasil dari pengujian, dan daun menunjukkan kelas akhir (Purnomo et al., 2023)

a. *Entropy* (Mengukur Ketidakteraturan Data)

$$E(S) = -\sum_{i=1}^{c} p_i\ log_2\ p_i \qquad (2)$$

Keterangan:

S = himpunan data pada node

C = jumlah kelas

Pi = proporsi data pada kelas ke-i

b. Information Gain (Menentukan Atribut Terbaik untuk Pemisahan Data)

$$IG(A) = E(S) - \sum_{v \in Values(A)} | \frac{s_v}{s} | E(S_v) \quad (3)$$

Keterangan:

Gain (S,A) = informasi yang diperoleh jika atrimut A digunakan untuk membagi dataset S

Values (A) = himpunan nilai berbeda pada atribut A

Sv = subset dari S di mana atribut A bernilai v

|Sv|/|S| = proporsi subset dibandingkan keseluruhan dataset

2.4. *CRISP-DM* (*Cross Industry Standard Process for Data Mining*)

CRISP-DM adalah standar proses yang umum digunakan dalam proyek Data Mining. Model ini terdiri dari enam fase utama: a) Business Understanding: Memahami tujuan proyek dan kebutuhan bisnis; b) Data Understanding: Mengumpulkan, mengeksplorasi, dan memahami data yang tersedia; c) Data Preparation: Membersihkan, mengolah, dan menyiapkan data agar dapat digunakan untuk modeling; d) Modeling: Membangun model prediktif menggunakan algoritma seperti KNN dan Decision Tree; e) Evaluation: Mengevaluasi model berdasarkan metrik performa untuk menilai keakuratannya; f) Deployment: Mengimplementasikan model dalam lingkungan nyata agar dapat digunakan secara langsung. *CRISP-DM* memberikan kerangka kerja sistematis yang mendukung proses analisis data dari awal hingga akhir (Rianti et al., 2023).

3. Hasil dan Pembahasan

3.1. *Business Understanding*

Penelitian ini menggunakan pendekatan *CRISP-DM* (Cross Industry Standard Process for Data Mining) sebagai kerangka kerja dalam pembangunan sistem klasifikasi kondisi kesehatan berdasarkan data *Medical Check-Up* (MCU). Metodologi ini dipilih karena terstruktur` dan cocok untuk proyek berbasis data mining,*CRISP-DM* terdiri dari enam tahap utama.

3.2. *Data Understanding*

Data yang digunakan merupakan data hasil pemeriksaan MCU pasien di RS EMC Cibitung pada tahun 2024. Tahapan *Data Understanding* bertujuan untuk memahami struktur, tipe, dan pola data yang akan digunakan dalam proses klasifikasi. Pada gambar tersebut, ditampilkan data yang sudah diproses dari hasil *Medical Check-Up* pasien, dengan beberapa atribut penting. Data berasal dari hasil MCU tahun 2024 di RS EMC Cibitung atribut yang dikumpulkan.

Data berasal dari hasil MCU 2024 di RS EMC Cibitung untuk melakukan MCU pada pasien yaitu pemprosesan untuk menyamakan skala antar fitur agar tidak ada yang

mendominasi dalam proses klasifikasi. Total jumlah fitur yang digunakan sebanyak 12 (termasuk label) sebelum modeling, data telah melalui proses *encoding* untuk data kategorikal seperti jenis kelamin dan kondisi jantung dan normalisasi menggunakan Min-Mix atau standard scaler (Mart et al., 2019). Label target kondisi asli digunakan untuk membandingkan performa prediksi model data digunakan dalam pembagian training dan testing untuk validasi model.

	JENIS KELAMIN	BODY MASS INDEX	DENYUT NADI	SUHU	PEMERIKSAAN VISUS	DAYA LIHAT WARNA	JANTUNG	UMUR	TEKANAN SISTOLIK	TEKANAN DIASTOLIK	VISUS KIRI DENOMINATOR	VISUS KANAN DENOMINATOR
0	0.0	1.095713	-0.812170	-0.019754	0.0	0.0	1.0	1.683079	-0.395070	-1.841339	0.0	7.00
1	1.0	0.454777	-1.312224	-1.706993	0.0	0.0	1.0	1.205882	-0.632899	-0.455281	0.0	0.00
2	0.0	2.065777	-0.097807	1.245675	0.0	0.0	1.0	1.086582	0.140044	0.604645	1.0	1.00
3	1.0	0.870519	0.116502	-0.019754	1.0	0.0	1.0	1.205882	1.329186	1.012310	2.0	0.25
4	1.0	0.731939	1.259484	0.402055	1.0	0.0	1.0	1.205882	0.912986	1.338441	0.0	0.00
...	...	...	...	...	...	...	...	...	...	...	...	...
358	0.0	1.216971	-1.312224	1.667485	0.0	0.0	0.0	1.683079	1.567014	1.746105	10.0	5.00
359	0.0	-0.688513	-1.240788	0.402055	0.0	0.0	0.0	-0.941506	-0.692356	-0.618347	0.0	0.25
360	1.0	-1.381416	-0.383552	0.823865	0.0	0.0	0.0	-1.538003	-1.108555	-1.678273	0.0	0.00
361	0.0	-0.861739	-1.455097	0.402055	0.0	0.0	0.0	-1.060806	-0.989641	-1.189076	3.0	0.00
362	0.0	1.216971	-1.312224	1.667485	0.0	0.0	0.0	1.683079	1.567014	1.746105	10.0	5.00

363 rows × 12 columns

Sumber: Hasil Penelitian (2025)

Gambar 2. Hasil atribut yang digunakan

3.3. *Data Preparation*

Tahap ini mencakup proses pembersihan dan transformasi data agar siap digunakan oleh algoritma *Machine Learning.* Langkah-langkahnya meliputi:

a. Pemilihan Atribut (*Feature Selection)*

Atribut-atribut yang digunakan dalam penelitian ini dipilih berdasarkan relevansinya terhadap kondisi kesehatan pasien. Atribut yang digunakan adalah: (1) Tekanan Darah Sistoling (mmHg); (2) Tekanan Darah Diastolik (mmHg); (3) Denyut Nadi (bpm); (4) Suhu Tubuh (°C); (5) Kesimpulan Tekanan Darah (Normal/Pre Hipertensi/Hipertensi); (6) Kesimpulan Jantung (Normal/Tidak Normal). Tarket klasifikasi adalah kondisi Kesehatan pasien dengan 0 (sehat) dan 1(tidak sehat).

b. Pembersihan data (*Data Cleaning)*

Pada tahap ini dilakukan pemeriksaan tahapan data, seperti: (1) Pemeriksaan nilai kosong (missing values): Data diperiksa untuk memastikan tidak ada nilai yang hilang pada setiap kolom. Jika ditemukan nilai kosong, data akan diisi menggunakan Teknik imputasi atau dihapus sesuai dengan kebutuhan; (2) Identifika outlier: Data yang berada jauh dari distribusi normal, seperti nilai nilai tekanan darah atau denyut nadi yang tidak wajar, pemeriksaan dan ditangani; (3) Pengapusan data duplikat: Jika terdapat data yang sama persis, data tersebut dihapus untuk mencegah bias pada model. Pemeriksaan inkonsistensi data: Contoh, memastika bahwa tekanan sistolik selalu lebih besar diastolik.

c. Implementasi Form Input Pada Aplikasi

Sebagai hasil dari tahap persiapan data, form input pada aplikasi prediksi kesehatan telah dirancang sedemikian rupa agar sesuai dengan format data yang telag diproses. From ini

berfungsi untuk mengumpulkan data pasien yang akan diprediksi kondisi kesehatan berdasarkan fitur-fitur hasil *Medical Check-Up* (MCU).

Sumber: Hasil Penelitian (2025)

Gambar 3. Prediksi Kondisi Kesehatan Normal

Pada Gambar 3 Implementasi Form Input pada Aplikasi bisa kita lihat bahwa data yang dimasukkan pada form aplikasi: (1) Tekanan Sistolik: 120 mmHg, (2) Tekanan Diastolik: 80 mmHg, (3) Denyut Nadi: 70 bpm. (4) Suhu Tubuh: 35.0 °C, (5) Kesimpulan Tensi: Normal, (6) Kesimpulan Jantung: Normal

Sumber: Hasil Penelitian (2025)

Gambar 4. Prediksi Kondisi Kesehatan Tidak Normal

Pada Gambar 4 hasil prediksi menunjukan “Tidak Normal”, maka model mendeteksi bahwa data vital pasien mengindikasikan adanya potensi masalah Kesehatan. Prediksi “Tidak Normal” muncul berdasarkan kombinasi nilai-nilai input berikut: (1)Tekanan Sistolik dan Diastolik yang lebih tinggi dari rentang normal; (2) Denyut Nadi yang terlalu tinggi atau rendah; (3) Suhu Tubuh di luar rentang normal (hipotermia atau demam); (4) Kesimpulan

Tensi yang menunjukan Pre Hipertensi atau Hipertensi; (5) Kesimpula Jantung yang menunjukan adanya gangguan irama atau kondisi abnormal.

Hasil “Tidak Normal” menandakan adanya kemungkinan hipertensi dan gangguan pada sistem kondiovaskular pasien, sebagai mana pada inpu berikut: (1) Tekanan Sistolik: 150.0 mmHg; (2) Tekanan Diastolik: 98 mmHg; (3) Denyut Nadi: 120.0 bpm; (4) Suhu Tubuh: 38.0 °C; (5) Kesimpulan Tensi: Tinggi; (6) Kesimpulan Jantung: Tidak Normal.

Dari sini hipertensi adanya gangguan pada pasien, sehingga diperlukan perhatian medis lebih lanjut agar pasien dapet penanganan yang lebih baik seperti konsultasi ke dokter spesialis jantung, melakukan EKG atau laboratorium, dan melakukan perubahan gaya hidup yang lebih sehat.

3.4. *Modeling*

Di tahap modeling dalam proses ini adalah tahapan di mana model *machine learning* dibangun menggunakan data yang telah melalui tahap persiapan (*data preparation)*. Pada penelitian ini, dua algoritma digunakan untuk membangun model Klasifikasi kondisi Kesehatan pasien berdasarkan data *Medical Check-Up* (MCU), yaitu *K-Nearest Neighboars* (KNN) dan *Decision Tree*. Kedua algoritma dipilih karena memiliki karakteristik yang sesuai untuk data numerik dan katagorikal, sertas sering digunakan dalam kasus klasifikasi data kesehatan.

a. Pemilihan Fitur (*Feature Selection*)

   Fitur yang digunakan dalam penelitian ini dipilih berdasarkan hasil *Medical Check-Up* dan relevansi terhadap kondisi kesehatan, yaitu: (1) Tekanan darah: Sistolik dan Diastolik. (2) Denyut Nadi; (3) Suhu Tubuh; (4) Kesimpulan Tekanan Darah: Normal/Tinggi/Rendah; (5) Kesimpulan Kondisi Jantung: Normal/Tidak Normal. Target atau label klasifikasi adalah kondisi kesehatan yang dibagi menjadi dua kelas 0 (sehat) dan 1 (tidak sehat).

b. Pembagian Data

   Data dibagi menjadi dua bagian: (1) 80% untuk data latih (training set) ; (2) 20% untuk data uji (testing set). Pembagian dilakukan secara acak dengan menggunakan fungsi train_test_split dari pustaka scikit-learn dengan parameter random_state=42 untuk menjaga reprodusibilitas.

c. Pembagian Model

   (1) *K-Nearest Neighbors* (KNN), Model KNN dibangun dengan beberapa nilai k = 3, 5, dan 10. Model terbaik diperoleh pada nilai k = 3, dengan hasil akurasi tertinggi.KNN bekerja dengan mengukur jarak terdekat antara data uji dan data latih menggunakan metode *Eulicdean distance*.

   (2) *Decision Tree* Model *Decision Tree* dibangun menggunakan parameter default terlebih dahulu, kemudian dilakukan eksperimen pada nilai max_depth untuk menghindari *overfitting*. *Decision Tree* memiliki kelebihan dalam kemudahan interpretasi dan visualisasi aturan klasifikasi.

Model KNN bekerja dengan cara membandingkan jarak data baru dengan jumlah tetangga terdekat di data latih. Dalam penelitian ini digunakan nilah k = 3, k = 5, dan k = 10 untuk mengevaluasi performa model. Jarak antar data hitung menggunakan *Euclidean Destance*.

d. Evaluasi Model

Model yang telah dilatih menggunakan data uji. Evaluasi dilkukan dengan metrik Confusion Matrix, Accuracy, Precision, Recall, dan F1-Score. Metrik ini memberikan gambaran performa model dalam memprediksi dua kelas, yaitu: (1) Kelas 0 = Kondisi Sehat; (2) Kelas 1 = Kondisi Tidak Sehat

```
Confusion Matrix:
[[45  5]
 [ 6 13]]

Classification Report:
              precision    recall  f1-score   support

         0.0       0.88      0.90      0.89        50
         1.0       0.72      0.68      0.70        19

    accuracy                           0.84        69
   macro avg       0.80      0.79      0.80        69
weighted avg       0.84      0.84      0.84        69
```

Sumber: Hasil Penelitian (2025)

Gambar 5 *Confusion Matrix* Prediksi Model

Confusion matrix tersebut menunjukkan bahwa model berhasil mengklasifikasikan 45 data positif secara benar dan melakukan 5 kesalahan, sementara seluruh data pada kelas kedua (10 data) berhasil diprediksi dengan benar. Laporan klasifikasi memperlihatkan bahwa kelas pertama memiliki precision 0,90, recall 0,90, dan f1-score 0,90, sedangkan kelas kedua menunjukkan precision 0,72, recall 0,88, dan f1-score 0,78. Secara keseluruhan, model mencapai akurasi 0,84 dengan nilai *macro average* dan *weighted average* yang konsisten, sehingga menandakan performa yang cukup baik dalam memprediksi kedua kelas meskipun masih terdapat ketidakseimbangan kinerja antar kelas.

3.5. *Evaluation*

Dari Gambar 5 dapat dijelaskan a) Akurasi: 84% menunjukkan bahwa model mampu memprediksi kondisi kesehatan secara keseluruhan dengan cukup baik; b) Precision kelas 0: 0.88, lebih tinggi dibanding kelas 1 (0.72), yang berarti model lebih akurat dalam mengenali pasien sehat dibanding pasien tidak sehat; c) *Recall* kelas 1: 0.68, yang mengindikasikan bahwa sekitar 32% pasien tidak sehat tidak terdeteksi oleh model (*false negative*); d) F1-Score kelas 1: 0.70, menunjukkan bahwa model masih memiliki ruang untuk ditingkatkan dalam mengklasifikasikan pasien tidak sehat secara konsisten.

Dalam proses evaluasi model, data yang digunakan adalah data uji (testing set) yang diperoleh dari proses pembagian dataset awal menjadi dua bagian, yaitu data latih (training set) dan data uji (testing set). Pembagian ini bertujuan untuk menguji kemampuan geralisasi model terhadap data yang belum pernah dilihat sebelumnya, sehingga performa model dapat dinilai secara objektif (Ramadhan, 2019).

Pada penelitian ini, pembagian data dilakukan dengan proporsi 80% untuk data latih dan 20% untuk data uji. Artinya, sebagaian besar data digunakan untuk melatih model agar dapat mengenali pola dari data, sementara sisanya digunakan untuk menguji seberapa baik model dapat melakukan prediksi terhadap data baru (Gunawan et al., 2020).

a. Hasil Evaluasi per model

Evaluasi dilakukan untuk membandingkan performa dua algoritma klasifikasi yang digunakan dalam penelitian, yaitu *K-Nearest Neighbors* (KNN) dan *Decision Tree.* Evaluasi dilakukan terhadap data uji dengan menggunakan metrik Accuracy, Precision, Recall, dan F1-Score. Hasil evaluasi disajikan dalam tabel berikut:

Tabel 1. Hasil Evaluasi KNN dan *Decision Tree*

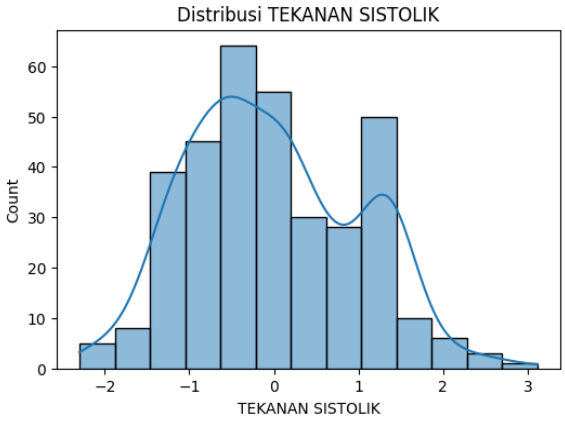
Metrik Evaluasi	*KNN (K-3)*	*Decision Tree*
Accuracy	89.85%	91.30%
Precision	90.65%	91.00%
Recall	89.85%	91.00%
F1-Score	89.89%	91.00%

Sumber: Hasil Penelitian (2025)

Decision Tree menunjukkan hasil evaluasi yang lebih unggul secara keseluruhan dibandingkan *KNN*, terutama dalam hal akurasi serta konsistensi antara precision dan recall. Model *KNN* tetap menunjukkan performa yang kompetitif dengan selisih akurasi yang tidak terlalu jauh dibanding *Decision Tree*, serta *Decision Tree* juga lebih unggul dalam aspek interpretabilitas karena menghasilkan struktur pohon keputusan yang mudah dipahami oleh pengguna non-teknis seperti dokter atau tenaga medis.

b. Visualisasi

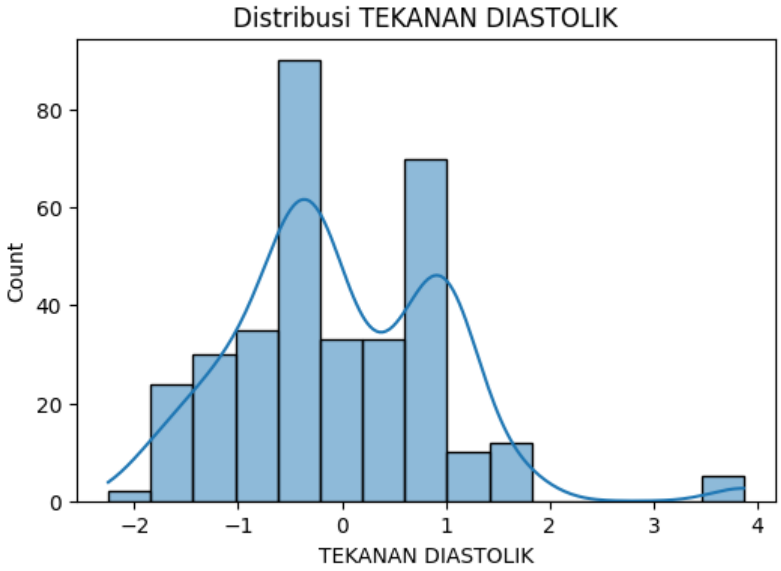
Gambar 6, menunjukan distribusi nilai tekanan sistolik berdasarkan data *Medical Check-Up* (MCU) yang telah melalui proses normalisasi atau standardisasi histogram ini menggambarkan jumlah sampel (count) pada berbagai rentang nilai tekanan sistolik, dilengkapi dengan kurva KDE (*Kernel Density Estimation*) untuk menunjukkan pola distribusi secara lebih halus (Anggrawan & Mayadi, 2023).



Sumber: Hasil Penelitian (2025)

Gambar 6. Distribusi Tekanan Darah

Interpretasi Grafik: Sebagian besar nilai tekanan sistolik berada di sekitar nilai rata-rata (titik nol pada sumbu x), menunjukkan distribusi data yang cenderung normal atau simetris. Adanya dua puncak pada kurva distribusi mengindikasikan kemungkinan terdapat dua kelompok utama — pasien dengan tekanan darah normal dan pasien dengan tekanan darah tinggi atau rendah. Bagian paling kiri dan kanan kurva menunjukkan adanya outlier, yaitu data dengan tekanan sistolik sangat rendah atau sangat tinggi, namun jumlahnya relatif sedikit.



Sumber: Hasil Penelitian (2025)

Gambar 7. Distribusi Tekanan Diastolik

Gambar 7, menunjukan histogram distribusi nilai tekanan diastolic setelah dilakukan proses normalisasi atau standarisasi (ditunjukan oleh rentang sumbu X yang menggunakan nilai Z-score, seperti -2, -1, 0, 1).

Ciri-ciri distribusi:

1) Bentuk Distribusi campuran (Multimodal)

   Terlihat ada dua puncak (bimodal)**,** yaitu sekitar nilai z-score -0.5 dan +0.8**,** yang menunjukkan bahwa terdapat dua kelompok besar dalam data tekanan diastolic. Ini bisa mengindikasikan dua populasi berbeda, misalnya kelompok dengan tekanan darah normal dan kelompok dengan tekanan darah lebih tinggi.

2) Penyebaran Data

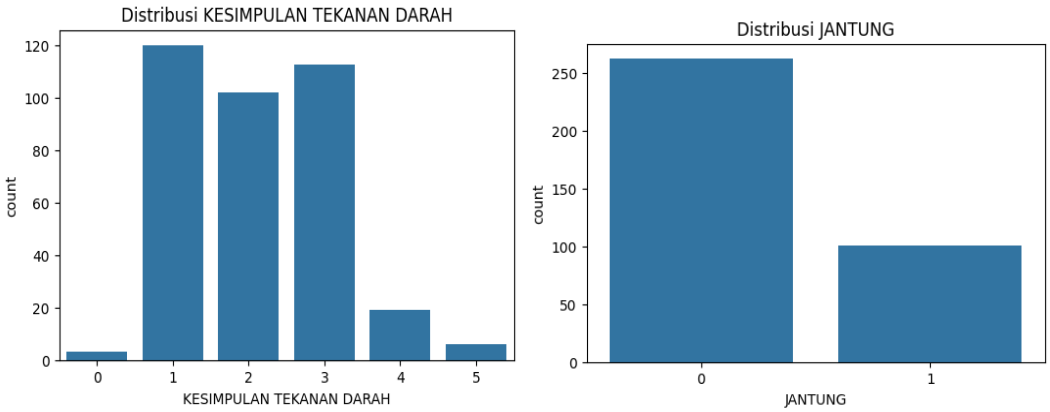
   Sebagian besar data berada antra z-score -2 hingga +2, yang berarti kebanyakan nilai tekanan diastolic masih berada dalam jangkauan yang wajar setelah standarisasi. Hanya sedikit data yang ekstrem (outlier), misalnya di z-score > 3.

3) *Skewness* (Kemencengan)

   Distribusi tidak simetris sempurna – ada kecendrungan miring ke kanan sedikit, artinya ada beberapa data dengan nilai tekanan diastolik tinggi.

4) Jumlah Data

Tinggi tertinggi di histogram menunjukan jumlah pasien terbanyak memiliki nilai tekanan diastolic yang dinormalilasi di sekitas 0 (bearti dekat dengan rata-rata populasi)



Sumber: Hasil Penelitian (2025)

Gambar 8. Distribusi Kesimpulan Tekanan Darah dan Distribusi Jantung

Pada Gambar 8, Distribusi Jantung menunjukan jumlah pasien berdasarkan kondisi Kesehatan jantung yang dikodekan dalam bentuk biner: (1) 0: Tidak ada indikasi gangguan jantung; (2) 1: Terindikasi memiliki masalah jantung.

Dari hasil distribusi jantung ini mayoritas label 0 (tidak bermasalah dengan jantung), yaitu sekitar 260 pasien. Sementara pasien yang memiliki indikasi masalah jantung 1, berjumlah sekitar 100 pasien. Distribusi ini menunjukan bahwa Sebagian besar pasien dalam dataset berada dalam kondisi jantung sehat, sementara 28–30% dari total populasi menunjukkan adanya indikasi gangguan jantung. Perbandingan ini penting dalam konteks:(Kurniawan, 2019).

Model prediktif: karena terdapat ketidak seimbangan data (*class imbalance*) model klasifikasi perlu diimbangi agar tidak bias terhadap kelas mayoritas (0). Analisis risiko: sekitar 1 dari 3 pasien dalam dataset ini memerlukan perhatian atau tindak lanjut terkait Kesehatan jantung .

Data yang didistribusikan menunjukkan bahwa sebagian besar pasien tidak mengalami masalah jantung. Namun, sekitar 28 persen pasien menunjukkan indikasi bahwa mereka memiliki gangguan jantung. Dalam proses pelatihan model klasifikasi, ketimpangan distribusi ini penting untuk diperhatikan karena dapat menyebabkan bias terhadap kelas mayoritas. Oleh karena itu, diterapkan metode penyeimbangan kelas seperti oversampling atau penggunaan metrik evaluasi yang sensitif terhadap data tidak seimbang.

3.6. *Deployment*

Tahap deployment merupakan bagian akhir dari proses data mining berdasarkan framework *CRISP-DM*, yang bertujuan untuk mengimplementasikan model klasifikasi ke dalam

system nyata yang dapat digunakan oleh pengguna akhir. Dalam penelitian ini, system klasifikasi hasil *Medical Check-Up* (MCU) dikembangkan dalam bentuk aplikasi web berbasis Streamlit (Ardianto & Rushendra, 2025).

Langkah-Langkah Deployment yang Dilakukan sebagai berikut: a) Menyimpan Model Terlatih: Setelah proses pelatihan model klasifikasi dengan algoritma K-Nearest Neighbors (KNN) dan Decision Tree selesai, model dengan performa terbaik disimpan dalam format.pkl menggunakan library pickle. File model ini disiapkan agar dapat digunakan kembali saat aplikasi menerima input untuk dilakukan proses prediksi; b) Pembuatan Antarmuka Aplikasi: Antarmuka pengguna dikembangkan menggunakan Streamlit, sebuah framework berbasis Python yang sederhana dan cocok untuk aplikasi data science. Aplikasi ini menerima masukan berupa data vital pasien, seperti tekanan darah (sistolik dan diastolik), suhu tubuh, denyut nadi, serta hasil kesimpulan tekanan darah dan jantung; c) Proses Prediksi: Setelah pengguna mengisi formulir input, data tersebut akan diproses dan dikonversi terlebih dahulu —terutama untuk variabel kategorikal seperti kesimpulan tekanan darah dan kondisi jantung. Selanjutnya, data yang telah diproses dikirim ke model klasifikasi untuk menghasilkan prediksi kondisi pasien (sehat atau tidak sehat). Hasil ini ditampilkan langsung pada tampilan web beserta rangkuman input yang telah dimasukkan; d) Deployment ke Platform Hosting: Aplikasi ini diunggah ke Streamlit Community Cloud secara gratis, yang terhubung dengan GitHub sebagai sumber file aplikasi (misalnya app.py, model.pkl, dan file lain yang diperlukan). Setelah integrasi dengan repositori GitHub berhasil dilakukan, aplikasi dapat diakses publik melalui URL khusus dan digunakan oleh tenaga medis untuk membantu analisis hasil MCU.

Manfaat *Deployment* sebagai berikut: Meningkatkan efisiensi klasifikasi hasil MCU dengan mengurangi kebutuhan analisis manual oleh tenaga medis. Menyediakan hasil prediksi secara real-time dengan tingkat objektivitas yang lebih tinggi. Mendukung aksesibilitas luas, karena sistem dapat dijalankan dari perangkat apa pun yang terhubung ke internet. Gambar 9 menunjukkan tampilan aplikasi prediksi kondisi Kesehatan.

Sumber: Hasil Penelitian (2025)

Gambar 9. Tampilan Aplikasi

4. Kesimpulan

Penelitian ini berhasil mengembangkan sistem klasifikasi untuk menilai kondisi kesehatan berdasarkan data *Medical Check-Up* (MCU) dari RS EMC Cibitung, dengan memanfaatkan algoritma *K-Nearest Neighbors* (KNN) dan *Decision Tree*. Sistem ini dirancang untuk mengolah data vital pasien seperti tekanan darah, suhu tubuh, dan denyut nadi, serta memberikan hasil prediksi secara otomatis melalui antarmuka web berbasis Streamlit. Berdasarkan hasil evaluasi, kedua algoritma menunjukkan kinerja yang baik, namun KNN tampil lebih stabil pada data uji terutama dalam aspek akurasi dan *F1-score*, sehingga dianggap lebih efektif dalam klasifikasi data kesehatan. Aplikasi ini telah berhasil diimplementasikan secara online melalui platform Streamlit Cloud, sehingga dapat diakses kapan saja dan di mana saja untuk mempercepat proses analisis hasil MCU secara objektif.

Daftar Pustaka

Adiputra, I. M. S., Trisnadewi, N. W., Oktaviani, N. P. W., Munthe, S. A., Hulu, V. T., Budiastutik, I., Faridi, A., Radeny Ramdany, Rosmauli Jerimia Fitriani, P. O. A. T., Rahmiati, B. F., Lusiana, S. A., Susilawaty, A., Sianturi, E., & Suryana. (2021). *Metodologi Penelitian Kesehatan*. Yayasan Kita Menulis. https://repositori.uin-alauddin.ac.id/19810/1/2021_Book Chapter_Metodologi Penelitian Kesehatan.pdf

Anggrawan, A., & Mayadi, M. (2023). Application of KNN Machine Learning and Fuzzy C-Means to Diagnose Diabetes. *MATRIK : Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer*, *22*(2), 405–418. https://doi.org/10.30812/matrik.v22i2.2777

Ardianto, M. R., & Rushendra, R. (2025). Prediksi Penyakit Diabetes Berdasarkan Perbandingan Klasifikasi Metode K-Nearest Neighbor, Naïve Bayes, Dan Decision Tree Menggunakan Rapid Miner. *Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)*, *10*(2), 973–985.

Dietterich, T. G. (2005). Machine Learning #1 - Overview. *To Appear in Annual Review of Computer Science*, *4*(1990), 46.

Gunawan, I., Agushybana, F., & Kartasurya, M. I. (2020). Perancangan Sistem Informasi Medical Check Up Guna Mempercepat Pelayanan MCU di RSUD Brebes. *Jurkes.Polije.Ac.Id*, *8*(1), 39–54. https://doi.org/10.25047/j-kes.v8i1.140

Hasran. (2020). Klasifikasi Penyakit Jantung Menggunakan Metode K-Nearest Neighbor. *Indonesian Journal of Data and Science*, *1*(1), 6–10. https://doi.org/https://doi.org/10.33096/ijodas.v1i1.3

Kurniawan, B. (2019). Analisis Pemanfaatan Layanan Medical Check-Up Di Rumah Sakit Tk.II Moh. Ridwan Meuraksa Jakarta Timur Tahun 2019. *Jurnal Medika Hutama*, *01*(01), 29–36. https://jurnalmedikahutama.com/index.php/JMH/article/view/18

Mart, F., Contreras-ochando, L., & Lachiche, N. (2019). *CRISP-DM Twenty Years Later : From Data Mining Processes to Data Science Trajectories*.

Pei, J. H. J., & Tong, H. (2015). *Data Mining Concepts And Techniques*.

Purnomo, H., Pambudi, R. E., & Irawan, R. (2023). Penerapan Decision Tree untuk Klasifikasi Penyakit Berdasarkan Data Rekam Medis. *Aisyah Journal of Informatics and Electrical Engineering*, *7*(1), 1–8. https://doi.org/https://doi.org/10.30604/jti.v7i1.655

Ramadhan, P. S. (2019). Penerapan K-Nearest Neighbor dalam Pendeteksian Abcessus. *InfoTekJar (Jurnal Nasional Informatika Dan Teknologi Jaringan)*, *3*(2), 61–70. https://doi.org/10.30743/infotekjar.v3i2.1003

Rianti, A., Majid, N. W. A., & Fauzi, A. (2023). CRISP-DM: Metodologi Proyek Data Science. *Prosiding Seminar Nasional Teknologi Informasi Dan Bisnis (SENATIB) 2023*, 107–114. https://ojs.udb.ac.id/index.php/Senatib/article/view/3015