

Penerapan Algoritma *K-Nearest Neighbor* pada Klasifikasi Penyakit Jantung

Rifda Naifa Saniy¹, Yessica Tarida Sheevana Sitorus², Naomi Angela Meyana³, Sandrina Najwa⁴,
Anindya Apriliyanti Pravitasari⁵, Fajar Indrayatna⁶.
Universitas Padjadjaran^{1,2,3,4,5,6}
rifda20002@mail.unpad.ac.id

Abstract

Abstrak. Jantung adalah salah satu organ vital manusia yang berfungsi untuk memompa darah ke seluruh tubuh. Peran jantung yang sangat penting membuat penyakit jantung menjadi faktor kematian yang paling tinggi di dunia per tahun 2021. Metode klasifikasi *K-Nearest Neighbor* (KNN) diterapkan untuk mempermudah ahli jantung dalam penyusunan dan penggolongan data. Pada penelitian ini penulis melakukan simulasi penerapan Metode KNN pada dataset penyakit jantung yang diperoleh dari *Kaggle* untuk mengetahui tingkat akurasi dari penggunaan metode KNN pada klasifikasi penyakit jantung. Dari penelitian ini didapatkan nilai K yang paling efektif pada K= 7 dengan tingkat akurasi 90.16%.

Kata kunci: penyakit jantung, klasifikasi, *K-Nearest Neighbor*.

I. PENDAHULUAN

Penyakit jantung adalah kondisi ketika jantung mengalami gangguan dan tidak bisa menjalankan fungsi utamanya memompa darah ke seluruh tubuh secara optimal. Jantung adalah salah satu organ vital yang memengaruhi kinerja seluruh organ tubuh manusia untuk menunjang kehidupan. Jika fungsi jantung terganggu, maka sirkulasi darah dari dan ke seluruh tubuh akan terganggu, gangguan ini lama-kelamaan akan berdampak ke organ tubuh yang lain juga, yang membuat penyakit jantung menjadi mematikan. Hasil Riset Kesehatan Dasar Tahun 2019 menunjukkan sebesar 1,5% atau 15 dari setiap 1.000 penduduk Indonesia menderita penyakit jantung. Dengan jumlah penduduk Indonesia yang mencapai 250 juta penduduk, maka setidaknya 250.000 penduduk memiliki penyakit jantung. Hal ini dibuktikan dengan data Kementerian Kesehatan pada tahun 2020 bahwa sebanyak 1,25 juta kematian di Indonesia diakibatkan oleh penyakit jantung. Tidak sebanding dengan banyaknya penderita penyakit jantung, jumlah dokter yang menangani penyakit jantung di Indonesia masih terbatas. Oleh karena itu, dengan kemajuan teknologi, diusahakan berbagai cara untuk membantu kinerja para dokter untuk menangani penderita penyakit jantung, salah satunya adalah bantuan teknologi *machine learning* dengan algoritma *K-Nearest Neighbor* (KNN).

K-Nearest Neighbor (KNN) adalah salah satu metode yang digunakan untuk klasifikasi objek baru berdasarkan sejumlah K tetangga terdekat. Algoritma KNN relatif sederhana dan mudah dipahami, sehingga cukup umum digunakan. Pada penerapan algoritma ini, pengklasifikasian terhadap sebuah gambar berdasarkan jarak terdekat dengan tetangganya. Nilai jarak ini akan digunakan sebagai nilai kemiripan antara data uji dan data latih.

Penelitian ini sudah pernah dilakukan oleh beberapa pihak, diantaranya oleh Hasran [1] pada tahun 2020 dan dipublikasikan dalam bentuk artikel yang berjudul “Klasifikasi Penyakit Jantung Menggunakan Metode *K-Nearest Neighbor*”. Pada artikel tersebut, Hasran membahas penelitiannya dalam menggunakan algoritma KNN terhadap *dataset* penyakit jantung yang didapat dari pusat *dataset UCI Machine Learning Repository* dan memperoleh hasil nilai K=6 dengan akurasi sebesar 85%.

Terinspirasi dari penelitian yang dilakukan Hasran [1], pada penelitian ini akan dilakukan juga penerapan algoritma KNN dengan menggunakan *dataset* berbeda, *dataset* yang digunakan adalah data penyakit jantung diunduh dari *Kaggle*.

Berdasarkan latar belakang penelitian yang sudah dipaparkan, timbul rumusan masalah yang akan dibahas pada artikel ini, yaitu bagaimana hasil dari penerapan algoritma KNN dalam klasifikasi penyakit jantung.

Dari rumusan masalah tersebut, penelitian ini bertujuan untuk menerapkan algoritma KNN pada klasifikasi penyakit jantung dan memperoleh tingkat akurasi terbaik.

Manfaat yang didapat dari penelitian ini:

1. Bagi penulis, penelitian ini dapat memperkuat pemahaman terhadap metode klasifikasi *K-Nearest Neighbor* (KNN) dan mendapat gambaran tentang penerapannya dalam kehidupan nyata.
2. Bagi Pembaca, penelitian ini diharapkan dapat menambah wawasan mengenai metode klasifikasi, terutama metode *K-Nearest Neighbor* (KNN).

II. METODE PENELITIAN

Penelitian ini akan membahas penerapan algoritma *K-Nearest Neighbor* (KNN) pada klasifikasi penyakit jantung. Langkah yang diperlukan dalam melakukan penerapan adalah (1) menyiapkan data yang akan digunakan; (2) *data preprocessing* dengan memisahkan *dataset* menjadi data latih dan data uji dengan perbandingan yang ditentukan; (3) menerapkan algoritma KNN.

2.1 Data Preparation

Untuk mempermudah estimasi, penelitian ini menggunakan *dataset* yang sudah tersedia untuk umum dari situs *Kaggle*. *Dataset* yang digunakan memiliki judul *Heart Disease* dan memuat 302 data dengan 14 atribut, dimana 13 atribut berupa input dan 1 atribut berupa target. Target yang dihasilkan adalah '0' untuk data pasien yang tidak mengalami penyakit jantung, dan '1' untuk data pasien yang mengalami penyakit jantung. Untuk deskripsi masing-masing atribut dapat dilihat pada table dibawah ini,

Tabel 1. Deskripsi Masing-Masing Atribut

No	Atribut	Deskripsi
1	age = <i>age</i>	Usia
2	sex = <i>sex</i>	Jenis Kelamin
3	cp = <i>chest pain type</i>	Jenis nyeri dada. Terdapat 4 nilai yaitu: 0: <i>typical angina</i> 1: <i>atypical angina</i> 2: <i>non-anginal pain</i> 3: <i>asymptomatic</i>
4	trestbps = <i>resting blood pressure</i>	Tekanan darah saat istirahat (dalam mmHg)
5	chol = serum <i>cholestorol</i>	Kadar kolesterol (dalam mg/dl)
6	fbs = <i>fasting blood sugar</i>	Kadar gulah darah > 120 mg/ml. Terdapat 2 nilai yaitu: 1: <i>true</i> 0: <i>false</i>
7	restecg = <i>resting electrocardiographic result</i>	Kondisi ECG ketika dalam keadaan istirahat. Terdapat 3 nilai: 0: normal 1: <i>ST-T wave abnormality</i> yaitu keadaan dimana gelombang <i>inversions</i> T dan atau ST meningkat maupun menurun lebih dari 0,5 mV. 2: keadaan dimana <i>ventricular</i> kiri mengalami hipertropi
8	thalach = <i>maximum heart rate achieved</i>	Detak jantung maksimal
9	exang = <i>exercise induced angina</i>	Latihan yang diinduksi angina. Terdapat 2 nilai yaitu: 0: ya 1: tidak
10	oldpeak = <i>old peak</i>	Depresi ST yang diinduksi oleh olahraga

11	slope = <i>the slope of the peak exercise</i>	Kemiringan puncak ST setelah berolahraga. Terdapat 3 nilai yaitu: 0 : <i>downsloping</i> 1 : <i>flat</i> 2 : <i>upsloping</i>
12	ca = <i>number of major vessels</i>	Jumlah pembuluh darah yang terdeteksi melalui proses pewarnaan <i>flourosopy</i> .
13	thal = <i>thal</i>	Kelainan darah yang disebut 'Thalassemia'. Terdapat 3 nilai yaitu: 0 : <i>normal</i> 2 : <i>fixed detected</i> 3 : <i>reversable detected</i> .

2.2 Data Preprocessing

a. Pemisahan data

Pada pemisahan data dilakukan pemisahan dataset menjadi dua bagian yaitu data latih dan data uji. Pembagian tersebut akan dipilih secara acak. Perbandingan jumlah data latih dan data uji yang digunakan pada penelitian ini adalah 80:20. Karena *dataset* yang digunakan dalam penelitian ini memiliki 302 data pasien jantung, maka jumlah data latih yang digunakan adalah 242 dengan 61 data sisanya digunakan sebagai data uji. Data latih digunakan untuk melatih *classifier* dalam mengenali karakteristik pasien yang terkena penyakit jantung maupun yang tidak terkena penyakit jantung. Data uji digunakan dalam uji coba terhadap model klasifikasi yang dihasilkan, serta menentukan performa dari model klasifikasi dengan cara membandingkan hasil klasifikasi model terhadap tiap data dalam data testing dengan label sebenarnya [8].

b. Normalisasi data

Metode normalisasi data yang digunakan pada *dataset* penelitian ini adalah Normalisasi min-maks. Metode normalisasi min-maks merupakan metode normalisasi yang melakukan transformasi linear terhadap data asli. Rumus penghitungan normalisasi min-max dapat dilihat pada rumus berikut:

$$X_n = \frac{X_0 - X_{min}}{X_{0=maks} - X_{min}} \quad (1)$$

Keterangan:

X_n \equiv Nilai baru untuk variabel x

X_0 \equiv Nilai lama untuk variabel x

X_{min} \equiv Nilai minimum untuk variabel x

X_{maks} \equiv Nilai maksimum untuk variabel x

Keuntungan dari metode ini adalah keseimbangan nilai perbandingan antardata saat sebelum dan sesudah proses normalisasi, serta tidak ada bias yang dihasilkan oleh metode ini. Kekurangan dari metode ini adalah ketika ada data baru, metode ini memungkinkan terjebak "out of the bound" error [6].

2.3 K-Nearest Neighbor (KKN)

Algoritma KNN adalahh algoritma yang digunakan dalam klasifikasi terhadap objek baru berdasarkan sejumlah K tetangga terdekatnya. Algoritma KNN dapat digolongkan sebagai *supervised learning*, *lazy learning algorithm*, dan *instance-based learning*, atau *memory-based learning* [7].

Pada algoritma KNN, dilakukan klasifikasi terhadap sekumpulan data berdasarkan pembelajaran data yang sudah tersusun sebelumnya, termasuk dalam *supervised learning* dimana hasil *query instance* yang baru disusun berdasarkan mayoritas kedekatan jarak dari kategori yang ada dalam KNN. Algoritma KNN bekerja berdasarkan jarak terbendek dari sampel uji ke sampel latih untuk menentukan hasil klasifikasinya. Setelah didapat hasil klasifikasi, diambil mayoritas

data dari hasil klasifikasi untuk dijadikan prediksi dari sampel uji. Dekat atau jauhnya tetangga bisa dihitung menggunakan rumus *euclidian*:

$$Euclidian = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (2)$$

Keterangan:

$p_i \equiv$ data latih atau *data training*

$q_i \equiv$ data uji atau *data testing*

Cara kerja algoritma KKN:

1. Menentukan nilai K, dengan $K \equiv$ jumlah tetangga terdekat.
2. Menghitung jarak antara data baru dengan semua data latih.
3. Mengurutkan data tersebut dan menetapkan tetangga terdekat berdasarkan jarak minimum ke K.
4. Memeriksa kelas dari jarak terdekat.
5. Menggunakan mayoritas sederhana dari kelas tetangga terdekat sebagai nilai prediksi data baru.

2.4 Evaluasi Model

Setelah melakukan proses model *training*, dilakukan evaluasi model. Pengujian ini akan menghasilkan nilai akurasi dari sebuah model yang sudah melalui proses training. Namun, tidak hanya nilai akurasi yang diperhatikan dalam melakukan evaluasi, tetapi diperlukan juga metrik evaluasi lainnya yang lebih komprehensif seperti *Precision*, *Recall*, dan *F1-score*. Untuk menghitung keempat metrik, diperlukan penyusunan *Confusion Matrix* terlebih dahulu.

Confusion Matrix

Confusion matrix adalah salah satu metode yang dapat digunakan untuk mengukur kinerja suatu metode klasifikasi. Metode ini akan menampilkan dan membandingkan nilai aktual atau nilai sebenarnya dengan nilai hasil prediksi model yang dapat digunakan untuk mendapatkan metrik evaluasi seperti *Accuracy*, *Precision*, *Recall* dan *F1-Score*. Pada pengukuran kinerja menggunakan *confusion matrix*, terdapat 4 istilah sebagai representasi hasil proses klasifikasi. Keempat istilah tersebut adalah *True Positive (TP)*, *True Negative (TN)*, *False Positive (FP)* dan *False Negative (FN)* [3].

<i>Actual Values</i>	0	TP	FP
	1	FN	TN
		0	1
		<i>Predicted Values</i>	

Gambar 1. *Confusion Matrix* dengan Algoritma KNN

Keterangan:

True Positive (TP): Jumlah data yang bernilai Positif dan diprediksi benar sebagai Positif

False Positive (FP): Jumlah data yang bernilai Negatif tetapi diprediksi sebagai Positif

True Negative (TN): Jumlah data yang bernilai Negatif dan diprediksi benar sebagai Negatif

False Negative (FN): Jumlah data yang bernilai Positif tetapi diprediksi sebagai Negatif

Accuracy

Accuracy/Akurasi didefinisikan sebagai tingkat kedekatan antara nilai prediksi dengan nilai sebenarnya. Akurasi dihitung dengan rumus sebagai berikut:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

Precision

Precision/Presisi didefinisikan sebagai rasio dari item relevan yang dipilih terhadap jumlah item yang terpilih. *Precision* dapat diartikan sebagai kecocokan antara permintaan informasi dengan jawaban terhadap permintaan tersebut. *Precision* dihitung dengan rumus sebagai berikut:

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

Recall

Recall didefinisikan sebagai rasio dari item relevan yang dipilih terhadap total jumlah item relevann yang tersedia. *Recall* dihitung dengan rumus sebagai berikut:

$$Recall = \frac{TP}{TP+FN} \quad (5)$$

F-1 Score

F-Measure adalah *harmonic mean* antara nilai *Precision* dan *Recall*, *F-Measure* juga kadang disebut dengan *F-1 Score*. *F-1 Score* dihitung dengan rumus sebagai berikut

$$F1_Score = \frac{2(Precision.Recall)}{Precision+Recall} \quad (6)$$

III.HASIL DAN PEMBAHASAN

Seperti yang sudah dipaparkan sebelumnya, langkah-langkah dalam mengestimasi tingkat akurasi penerapan *K-Nearest Neighbor* pada klasifikasi penyakit jantung diawali dengan menyiapkan data. Dataset penyakit jantung yang kami gunakan terdiri dari 302 data dengan 14 atribut. Dataset dapat dilihat pada tabel 2.

Tabel 2. *Dataset* Penyakit Jantung

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
...
298	57	0	0	140	241	0	1	123	1	0.2	1	0	3	0
299	45	1	3	110	264	0	1	132	0	1.2	1	0	3	0
300	68	1	0	144	193	1	1	141	0	3.4	1	2	3	0
301	57	1	0	130	131	0	1	115	1	1.2	1	1	3	0
302	57	0	1	130	236	0	0	174	0	0.0	1	1	2	0

303 rows x 14 columns

3.1 Data preprocessing

Setelah menyiapkan data, langkah selanjutnya merupakan data preprocessing, dimana pada tahap ini terdiri dari tahap split data dan tahap normalization data. Pada split data, dataset dibagi 2 secara acak menjadi 80% data latih dan 20% data uji. Selanjutnya, proses normalisasi data dilakukan dengan metode Min-Max Normalization pada persamaan (1). Normalisasi berperan agar nilai setiap atribut berada dalam interval yang sama, pada setiap atribut akan di ubah pada rentang 0-1. Hasil dataset setelah dilakukan normalisasi dapat dilihat pada tabel 3.

Tabel 3. *Dataset* Penyakit Jantung Setelah di Normalisasi

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal
0	0.708333	1.0	1.000000	0.481132	0.244292	1.0	0.0	0.603053	0.0	0.370968	0.0	0.00	0.333333
1	0.166667	1.0	0.666667	0.339623	0.283105	0.0	0.5	0.885496	0.0	0.564516	0.0	0.00	0.666667
2	0.250000	0.0	0.333333	0.339623	0.178082	0.0	0.0	0.770992	0.0	0.225806	1.0	0.00	0.666667
3	0.562500	1.0	0.333333	0.245283	0.251142	0.0	0.5	0.816794	0.0	0.129032	1.0	0.00	0.666667
4	0.583333	0.0	0.000000	0.245283	0.520548	0.0	0.5	0.702290	1.0	0.096774	1.0	0.00	0.666667
...
298	0.583333	0.0	0.000000	0.433962	0.262557	0.0	0.5	0.396947	1.0	0.032258	0.5	0.00	1.000000
299	0.333333	1.0	1.000000	0.150943	0.315068	0.0	0.5	0.465649	0.0	0.193548	0.5	0.00	1.000000
300	0.812500	1.0	0.000000	0.471698	0.152968	1.0	0.5	0.534351	0.0	0.548387	0.5	0.50	1.000000
301	0.583333	1.0	0.000000	0.339623	0.011416	0.0	0.5	0.335878	1.0	0.193548	0.5	0.25	1.000000
302	0.583333	0.0	0.333333	0.339623	0.251142	0.0	0.0	0.786260	0.0	0.000000	0.5	0.25	0.666667

303 rows x 13 columns

3.2 Implementasi dan Evaluasi Model KNN

Tahap selanjutnya adalah implementasi metode KNN pada data uji dan data latih yang sudah melalui tahap *preprocessing* sebelumnya. Untuk pengukuran jarak, akan digunakan persamaan (2) dan untuk nilai K akan menggunakan nilai K = 3. Untuk melihat tingkat akurasi dan nilai performa yang terbaik, nantinya akan dicoba dengan menggunakan nilai K = 1 hingga K = 10. Setelah itu dilanjutkan dengan evaluasi performa KNN, pada tahap ini akan diperoleh *output* berupa *confusion matrix* dan performa algoritma KNN pada klasifikasi penyakit jantung berupa *Accuracy*, *Precision*, *Recall* dan *F1-Score*.

Dengan menggunakan bahasa pemrograman *Python* dengan *tools Jupyter Notebook* dalam aplikasi *Anaconda Navigator*, diperoleh hasil dari evaluasi model dengan algoritma KNN pada klasifikasi penyakit jantung ketika nilai K = 3 sebagai berikut:

Actual Values	0	TP=23	FP=4
	1	FN=4	TN=30
		0	1
		Predicted Values	

Gambar 2. *Confusion Matrix* dengan Algoritma KNN untuk K = 3

Dari hasil penerapan algoritma KNN pada klasifikasi penyakit jantung untuk K = 3, didapat hasil:

Accuracy : 0.868852459

Precision : 0.8518518519

Recall : 0.8518518519

F1-Score : 0.8518518519

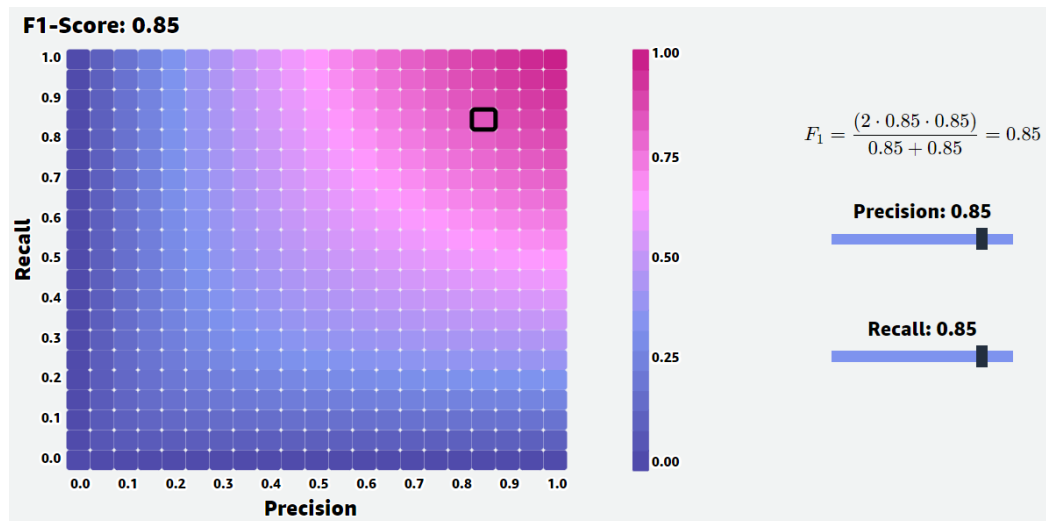
Untuk simulasi perhitungan performa algoritma KNN ketika K = 3 dapat dihitung dengan menggunakan rumus pada persamaan (3), (4), (5) dan (6) seperti berikut:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} = \frac{23 + 30}{23 + 4 + 4 + 30} = 0.868852459$$

$$Precision = \frac{TP}{TP + FP} = \frac{23}{23 + 4} = 0.8518518519$$

$$Recall = \frac{TP}{TP + FN} = \frac{23}{23 + 4} = 0.8518518519$$

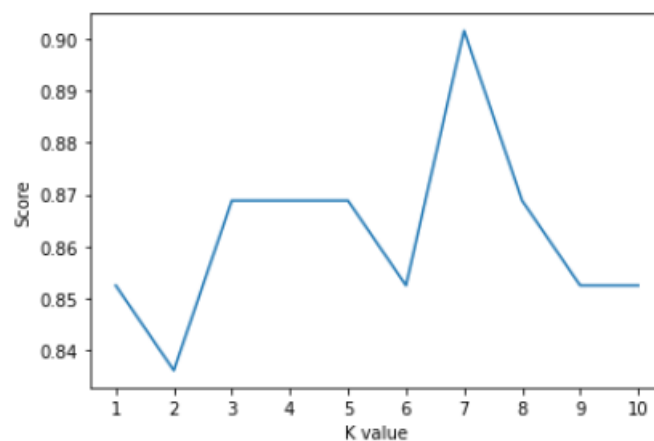
$$F1_Score = \frac{2(Precision \cdot Recall)}{Precision + Recall} = \frac{2(0.8518518519 \times 0.8518518519)}{0.8518518519 + 0.8518518519} = 0.8518518519$$



Seperti yang sudah dijelaskan sebelumnya, untuk memperoleh tingkat akurasi dan nilai performa terbaik akan dicoba dengan menggunakan nilai $K = 1$ hingga $K = 10$. Hasil perhitungan dapat dilihat pada tabel 4 dan visualisasi dari tingkat akurasi dengan menggunakan nilai $K = 1$ hingga $K = 10$ algoritma KNN dapat dilihat pada gambar 3.

Tabel 4. Hasil Uji performa KNN

K	Accuracy	Precision	Recall	$F1_Score$
1	0.852459	0.88888890	0.8	0.842105
2	0.836066	0.9259259	0.757576	0.833334
3	0.8688525	0.8518519	0.8518519	0.85185185
4	0.8688525	0.8888889	0.827586	0.85714
5	0.8688525	0.8148148	0.88	0.8461539
6	0.852459	0.8148148	0.88	0.842105
7	0.90164	0.85185185	0.92	0.8846
8	0.8688525	0.8888889	0.827586	0.85714
9	0.852459	0.777778	0.875	0.82353
10	0.852459	0.777778	0.875	0.82353



Maximum KNN Score is 90.16%

Gambar 3. Hasil Akurasi KNN

Akurasi terbaik dari klasifikasi penyakit jantung menggunakan metode KNN adalah 90.16% dengan $K=7$, dengan *precision* bernilai 0.92, *recall* bernilai 85.19%, dan *F1-score* bernilai 88.46%. *Precision* bernilai 0.92 berarti kecocokan antara permintaan informasi dengan jawaban terhadap permintaan tersebut bernilai 0.92. Lalu, rasio dari item relevan yang dipilih terhadap total jumlah item relevan yang tersedia atau *recall* bernilai 85.19%. *Harmonic mean* antara nilai *precision* dan *recall* atau *F1-score* adalah 88.46%.

IV. KESIMPULAN

Berdasarkan penelitian ini, dapat ditarik kesimpulan yaitu dengan simulasi nilai K dari rentang 1-10, diperoleh nilai K dengan akurasi, *precision* dan *F1-Score* terbaik berada pada saat nilai $K = 7$, sedangkan untuk nilai *recall* terbaik berada pada saat nilai $K = 2$. Nilai akurasi, *recall* dan *F1-Score* dari metode KNN dalam klasifikasi penyakit jantung saat nilai $K = 7$ berturut-turut adalah 90.16%, 92% dan 89%. Sedangkan nilai *precision* dari metode KNN dalam klasifikasi penyakit jantung saat nilai $K = 2$ adalah 93%.

DAFTAR PUSTAKA

- [1] Hasran. (2020). Klasifikasi Penyakit Jantung Menggunakan Metode K-Nearest Neighbour. *Indonesian Journal of Data Science* (pp. 6-10). Makassar: Universitas Muslim Indonesia.
- [2] Kaggle. (2019). "Heart Disease – Classification (Machine Learning)". Diakses pada 12 Juni 2022 dari <https://www.kaggle.com/code/cdabakoglu/heart-disease-classifications-machine-learning>
- [3] Karsito, & Susanti, S. (2019). Wibisono, A. B., & Fahrurrozi, A. (2019). Klasifikasi Kelayakan Peserta Pengajuan Kredit Rumah dengan Algoritma Naive Bayes di Perumahan Azzura Residencia. *Jurnal Teknologi Pelita Bangsa Vol.9, No.3*.
- [4] Lestari, M.E.I., 2015. Penerapan algoritma klasifikasi Nearest Neighbor (K-NN) untuk mendeteksi penyakit jantung. *Faktor Exacta*, 7(4), pp.366-371.
- [5] MUJAHIDIN, S. 2015. Klasifikasi Warna Kulit berdasarkan Ruang Warna RGB. In Seminar Nasional Aplikasi Teknologi Informasi (SNATI) (pp. 17–19). Yogyakarta: Universitas Islam Indonesia.
- [6] Mustaffa, Z., & Yusof, Y. (2011). A Comparison of Normalization Techniques in Predicting Dengue Outbreak. *International Conference on Business and Economics Research*, 1, 345–349.
- [7] Primartha, R. (2018). *Belajar Machine Learning*. Bandung: Informatika.
- [8] Wibisono, A. B., & Fahrurrozi, A. (2019). Wibisono, A. B., & Fahrurrozi, A. (2019). PERBANDINGAN ALGORITMA KLASIFIKASI DALAM PENGKLASIFIKASIAN DATA PENYAKIT JANTUNG KORONER. *Jurnal Ilmiah Teknologi dan Rekayasa Vol.24, No.3*.