PERBANDINGAN ALGORITMA *LOGISTIC REGRESSION* DAN *K-NEAREST NEIGHBOR* (KNN) DALAM DATASET PREDIKSI GAGAL JANTUNG

SKRIPSI

DISUSUN OLEH

JULIA NAMIRA NASUTION

2109020080

PROGRAM STUDI TEKNOLOGI INFORMASI

FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA

MEDAN

2025

PERBANDINGAN ALGORITMA *LOGISTIC REGRESSION* DAN *K-NEAREST NEIGHBOR* (KNN) DALAM DATASET PREDIKSI GAGAL JANTUNG

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer (S.Kom) dalam Program Studi Tekonologi Informasi pada Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara

JULIA NAMIRA NASUTION
NPM. 2109020080

PROGRAM STUDI TEKONOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
MEDAN
2025

LEMBAR PENGESAHAN

Judul Skripsi	:	PERBANDINGAN ALGORITMA *LOGISTIC REGRESSION* DAN *K-NEAREST NEIGHBOR* (KNN) DALAM DATASET PREDIKSI GAGAL JANTUNG
Nama Mahasiswa	:	JULIA NAMIRA NASUTION
NPM	:	2109020080
Program Studi	:	TEKNOLOGI INFORMASI

Menyetujui
Komisi Pembimbing

(Dr. Zainal Azis, M.Si.)
NIDN. 0113126301

Ketua Program Studi	**Dekan**
(Fatma Sari Hutagalung, S,Kom., M,kom.)	**(Dr. Al-Khowarizmi, S.Kom., M.Kom.)**
NIDN. 0117019301	**NIDN. 0127099201**

PERNYATAAN ORISINALITAS

PERBANDINGAN ALGORITMA *LOGISTIC REGRESSION* DAN *K-NEAREST NEIGHBOR* (KNN) DALAM DATASET PREDIKSI GAGAL JANTUNG

SKRIPSI

Saya menyatakan bahwa karya tulis ini adalah hasil karya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya.

Medan, 2025

Yang membuat pernyataan

Julia Namira Nasution

NPM. 2109020080

PERNYATAAN PERSETUJUAN PUBLIKASI

KARYA ILMIAH UNTUK KEPENTINGAN

AKADEMIS

Sebagai sivitas akademika Universitas Muhammadiyah Sumatera Utara, saya bertanda tangan dibawah ini:

Nama	: Julia Namira Nasution
NPM	: 2109020080
Program Studi	: Teknologi Informasi
Karya Ilmiah	: Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Muhammadiyah Sumatera Utara Hak Bedas Royalti Non-Eksekutif *(Non-Exclusive Royalty free Right)* atas penelitian skripsi saya yang berjudul:

PERBANDINGAN ALGORITMA *LOGISTIC REGRESSION* DAN *K-NEAREST NEIGHBOR* (KNN) DALAM DATASET PREDIKSI GAGAL JANTUNG

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksekutif ini, Universitas Muhammadiyah Sumatera Utara berhak menyimpan, mengalih media, memformat, mengelola dalam bentuk database, merawat dan mempublikasikan Skripsi saya ini tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis dan sebagai pemegang dan atau sebagai pemilik hak cipta.

Demikian pernyataan ini dibuat dengan sebenarnya.

Medan, 2025

Yang membuat pernyataan

Julia Namira Nasution

NPM. 2109020080

RIWAYAT HIDUP

DATA PRIBADI

Nama Lengkap	: Julia Namira Nasution
Tempat dan Tanggal Lahir	: Medan, 18 Juli 2003
Alamat Rumah	: Jl. Sederhana Gg. Raya 37 No. 2
Telepon/Faks/HP	: 087813098847
E-mail	: julianamira33@gmail.com
Instansi Tempat Kerja	: -
Alamat Kantor	: -

DATA PENDIDIKAN

SD	: SDN 060834 MEDAN	TAMAT: 2015
SMP	: SMPN 19 MEDAN	TAMAT: 2018
SMA	: SMAN 4 MEDAN	TAMAT: 2021

KATA PENGANTAR

Alhamdulillah, segala puji dan Syukur penulis panjatkan kehadiran Allah SWT yang telah melimpahkan Rahmat dan karunianya yang penuh dengan ilmu kepada penulis, sehingga penulis dapat menyelesaikan tugas akhir ini yang berjudul tentang "Perbandingan Algoritma *Logistic Regression* dan *K-Nearest Neighbor* (KNN) dalam Dataset Prediksi Gagal Jantung" untuk memenuhi persyaratan dalam jenjang strata satu dan mencapai gelar Sarjana Komputer dijurusan Teknologi Informasi, Fakultas Teknologi Informasi dan Ilmu Komputer, Universitas Muhammadiyah Sumatera Utara. Sholawat serta salam selalu tercurahkan kepada junjungan Nabi besar Muhammad SAW, keluarga dan sahabatnya yang syafaatnya kita nantikan diakhir zaman nanti. Dalam penyusunan Tugas Akhir ini, penulis telah mendapatkan banyak bantuan dan bimbingan dari berbagai pihak. Oleh karena itu pada kesempatan ini penulis temtunya berterimakasih kepada pihak dalam dukungan serta doa dalam penyelesaian skripsi. Penulis juga berterimakasih kepada :

1. Bapak Prof. Dr. Agussani, M.AP., selaku Rektor Universitas Muhammadiyah Sumatera Utara (UMSU).
2. Bapak Dr. Al-Khowarizmi, S.Kom., M.Kom., selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
3. Bapak Halim Maulana, S.T., M.Kom., selaku Wakil Dekan I Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.

4. Bapak Lutfi Basit, S.Sos., M.I.Kom., selaku Wakil Dekan II Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
5. Ibu Fatma Sari Hutagalung, S.Kom., M.Kom., selaku Ketua Program Studi Teknologi Informasi.
6. Bapak Mhd. Basri, S.Si., M.Kom., selaku Sekretaris Program Studi Teknologi Informasi.
7. Bapak Dr. Zainal Azis, M.Si., selaku dosen pembimbing skripsi yang telah meluangkan waktu, dan pikiran untuk memberikan arahan, bimbingan, dan saran yang berharga selama proses penyusunan skripsi ini.
8. Dosen-dosen Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara, atas ilmu yang diberikan selama masa perkuliahan.
9. Seluruh staf akademik Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara, atas fasilitas dan dukungan administratif yang diberikan selama masa perkuliahan hingga penyusunan skripsi.
10. Kedua orang tua tercinta Bapak Zulfikar Nasution dan Ibu Alfida yang selalu memberikan doa, dukungan, dan semangat dalam segala keadaan.
11. Kedua abang penulis dan juga kedua kakak ipar penulis yang telah memberikan dukungan dan doa kepada penulis.
12. Sahabat-sahabat digital forensik yaitu Dea Ikwa Cahya Syahfitri, Farah Zhafira Munthe, Aqilah Tahara, Suci Indah Ismana, Laila Salsabila, Umi

Khoirunnisa, yang selalu memberikan dukungan dan semangat selama masa perkuliahan.

13. Seluruh teman-teman KKN yang selalu memberikan motivasi, dukungan dan juga pengalaman yang tidak terlupakan.
14. Semua pihak yang terlibat, baik secara langsung maupun tidak langsung, dalam penyelesaian skripsi ini, yang mungkin namanya tidak disebutkan satu per satu.
15. Tidak lupa, penulis menyampaikan rasa terima kasih yang tulus kepada diri sendiri atas kesabaran, ketekunan, dan semangat yang terus dijaga selama proses penyusunan skripsi ini.

Penulis menyadari bahwa Tugas Akhir ini masih jauh dari sempurna dikarenakan keterbatasan pengetahuan serta kemampuan yang dimiliki penulis. Oleh karena itu, penulis mengharapkan kritik dan saran yang bersifat membangun untuk menyempurnakan penulisan skripsi ini. Semoga skripsi ini dapat bermanfaat bagi penulis khususnya dan bagi semua yang membutuhkan.

Wassalamu'alaikum warahmatullahi wabarakaatuh

Medan, 2025

Penulis

Julia Namira Nasution

PERBANDINGAN ALGORITMA *LOGISTIC REGRESSION* DAN *K-NEAREST NEIGHBOR* (KNN) DALAM DATASET PREDIKSI GAGAL JANTUNG

ABSTRAK

Penyakit gagal jantung merupakan salah satu penyebab utama kematian di seluruh dunia. Deteksi dini terhadap risiko gagal jantung menjadi krusial untuk meminimalisir dampak serius yang dapat ditimbulkan. Penelitian ini bertujuan untuk membandingkan performa dua algoritma machine learning, yaitu *Logistic Regression* dan *K-Nearest Neighbor* (KNN**)** dalam memprediksi gagal jantung menggunakan dataset dari platform *Kaggle*. Tahapan penelitian meliputi preprocessing data, normalisasi, pembagian data latih dan uji, serta implementasi model dan evaluasi menggunakan *confusion matrix*. Evaluasi dilakukan berdasarkan metrik akurasi, presisi, *recall*, dan f1-*score*. Hasil penelitian menunjukkan bahwa *Logistic Regression* memiliki akurasi sebesar 88,04% dan waktu eksekusi 0,022 detik, sedangkan KNN memperoleh akurasi sebesar 85,51% dan waktu eksekusi 0,158 detik. *Logistic Regression* unggul dalam *recall* dan f1-score, menjadikannya lebih efektif untuk deteksi dini gagal jantung. Dengan demikian, algoritma *Logistic Regression* dinilai lebih optimal dibandingkan KNN dalam konteks penelitian ini. Tetapi algoritma Logistic Regression tidak selalu lebih unggul dari K-Nearest Neighbor, karena hasil prediksi sangat bergantung pada karakteristik studi kasus.

Kata Kunci: Gagal Jantung, *Machine Learning, Logistic Regression, K-Nearest Neighbor*, Prediksi.

COMPARISON OF LOGISTIC REGRESSION AND K-NEAREST NEIGHBOR (KNN) ALGORITHMS IN HEART FAILURE PREDICTION DATASET

ABSTRACT

Heart failure is one of the leading causes of death worldwide. Early detection of heart failure risk is crucial to minimize its serious consequences. This study aims to compare the performance of two machine learning algorithms, namely Logistic Regression and K-Nearest Neighbor (KNN), in predicting heart failure using a dataset from the Kaggle platform. The research stages include data preprocessing, normalization, splitting into training and testing data, model implementation, and evaluation using a confusion matrix. Evaluation is based on accuracy, precision, recall, and F1-score metrics. The results show that Logistic Regression achieved an accuracy of 88.04% with an execution time of 0.022 seconds, while KNN achieved an accuracy of 85.51% with an execution time of 0.158 seconds. Logistic Regression outperformed in recall and F1-score, making it more effective for early detection of heart failure. Therefore, Logistic Regression is considered more optimal than KNN in the context of this study. However, Logistic Regression is not always superior to K-Nearest Neighbor, as prediction results highly depend on the characteristics of the specific case.

Keywords: Heart Failure, Machine Learning, Logistic Regression, K-Nearest Neighbor, Prediction.

DAFTAR ISI

DAFTAR TABEL

HALAMAN

DAFTAR GAMBAR

HALAMAN

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Gagal jantung adalah kondisi medis yang serius di mana jantung tidak dapat memompa darah dengan cukup efisien untuk memenuhi kebutuhan tubuh. Kondisi ini sering berkembang secara bertahap dan bisa disebabkan oleh berbagai faktor seperti penyakit jantung koroner, tekanan darah tinggi, diabetes dan banyak penyebab lainnya (Febrian, 2024). Jantung adalah organ utama yang harus bekerja dengan benar karena berfungsi untuk memompa darah ke seluruh tubuh sehingga oksigen dan zat-zat gizi dapat tersalurkan. Jika jantung tidak bekerja dengan benar, akan sangat mengganggu fungsi organ tubuh lainnya bahkan beresiko menyebabkan gagal jantung. Dengan kata lain bahwa penyakit kardiovaskular khususnya penyakit jantung adalah salah satu penyakit paling mematikan baik di negara maju maupun berkembang. Perhatian terhadap penyakit tersebut sangatlah penting dan sangat diperlukan.

Dengan adanya perkembangan teknologi, banyak hal yang dapat dilakukan untuk memberikan kemudahan kepada manusia. Deteksi dini kegagalan jantung sangat penting untuk mencegah komplikasi lebih lanjut dan memperbaiki prognosis pasien serta tantangan dalam diagnosa . Namun, gejala kegagalan jantung sering kali tidak spesifik, yang membuat diagnosis klinis menjadi tantangan. Oleh karena itu, pendekatan berbasis pembelajaran mesin (*machine learning*) digunakan untuk membantu dalam

prediksi risiko kegagalan jantung dengan menganalisis data medis secara otomatis.

Model prediksi berbasis data dapat membantu para praktisi medis dalam mengidentifikasi pasien yang berisiko tinggi mengalami kegagalan jantung lebih dini, memungkinkan intervensi medis yang lebih tepat waktu (Nugraha et al., 2024). *Machine learning* merupakan proses di mana komputer dapat bekerja lebih akurat saat mengumpulkan dan belajar dari data yang diberikan.

Penelitian ini menggunakan dataset publik spesifik dari Kaggle, yaitu *heart failure prediction,* yang mungkin belum banyak dieksplorasi dalam penelitian sebelumnya untuk prediksi gagal jantung menggunakan metode komparatif. Penulis mencoba menggunakan dua algoritma yang berbeda dan membandingkannya. Penelitian ini berfokus pada identifikasi algoritma yang paling optimal dalam konteks prediksi gagal jantung dengan menggunakan model evaluasi untuk mengukur keefektifan masing masing algoritma. Algoritma *Logistic Regression* adalah sebuah metode pembelajaran terbimbing yang digunakan untuk masalah regresi dan klasifikasi. Metode ini menggunakan fungsi logistik untuk memodelkan hubungan antara atribut independen dengan probabilitas klasifikasi data kategorikal. Sedangkan algoritma *K-Nearest Neighbor* adalah algoritma generalisasi untuk aturan tetangga terdekat, *offset* induktifnya adalah label kelas k-sampel dengan label kelas yang akan diuji paling mirip dengan yang terdekat.

Dalam penelitian ini penulis akan mencoba melakukan prediksi dengan membandingkan akurasi yang didapat untuk memberikan pemahaman lebih mendalam tentang potensi penggunaan *machine learning* dalam mendukung deteksi dini gagal jantung. Penelitian dilakukan melalui tahapan pengumpulan data, *preprocessing* data, implementasi, dan evaluasi. Evaluasi model menggunakan *Confusion Matrix* dengan metrik *accuracy*, *precision, recall*, dan f1-*score* (Andi Irfan Daeng Mappa, 2025).

1.2. Rumusan Masalah

Mengenai latar belakang tersebut, maka rumusan masalah yang dapat dijadikan pertimbangan dalam penelitian ini adalah sebagai berikut:

1. Bagaimana performa algoritma *K-Nearest Neighbor* (K-NN) dalam memprediksi gagal jantung?
2. Bagaimana performa algoritma *Logistic Regression* dalam memprediksi gagal jantung?
3. Algoritma mana yang paling efektif dalam memprediksi gagal jantung?
4. Bagimana perbedaan waktu hitung antara algoritma *Logistic Regression* dan *K-Nearest Neighbor* (K-NN)?

1.3. Batasan Masalah

Adapun batasan masalah dalam penelitian ini adalah sebagai berikut :

1. Penelitian ini akan menggunakan dataset yang relevan untuk memprediksi penyakit gagal jantung, dengan total 918 data dan 12 atribut.
2. Data yang digunakan dalam penelitian ini akan mencakup rentang waktu tertentu, sesuai dengan ketersediaan data yang dapat diakses di kaggle

https://www.kaggle.com/code/tanmay111999/heart-failure-prediction-cv-score-90-5-models

3. Atribut data yang akan digunakan untuk prediksi penyakit gagal jantung antara lain : *age, gender, chest pain type, resting bps, cholesterol, fasting blood sugar, resting ecg, max heart rate, exercise angina, oldpeak, st slope, target*.
4. *Output* yang dihasilkan pada penelitian ini berupa hasil evaluasi performa pengujian algoritma.
5. Penelitian ini dilakukan dengan evaluasi model *Confusion Matrix* untuk mengukur nilai dari *accuracy*, *precision*, *recall*, dan F1-*score*.

1.4. Tujuan Penelitian

Adapun tujuan dari penelitian ini yaitu :

1. Mengetahui performa algoritma *K-Nearest Neighbor* (K-NN) dalam memprediksi gagal jantung
2. Mengetahui performa algoritma *Logistic Regression* dalam memprediksi gagal jantung
3. Mengetahui algoritma mana yang paling efektif dalam memprediksi gagal jantung
4. Mengetahui perbedaan waktu hitung antara algoritma *Logistic Regression* dan *K-Nearest Neighbor* (K-NN)

1.5. Manfaat Penelitian

Pada penelitian ini diperoleh manfaat yaitu antara lain :

1. Model prediksi yang dibuat dapat mendukung diagnosis awal penyakit gagal jantung, sehingga memungkinkan intervensi dan perawatan yang lebih cepat.
2. Penelitian ini dapat menjadi langkah awal untuk pengembangan teknologi kesehatan yang lebih maju dan inovatif dalam mendukung diagnosis serta pengelolaan penyakit kardiovaskular.
3. Dengan adanya model prediksi ini bisa menambah wawasan mengenai kinerja berbagai algoritma, tetapi juga memberi rekomendasi spesifik untuk aplikasi klinis.

BAB II

LANDASAN TEORI

2.1. Gagal Jantung

Gagal jantung dapat disebabkan oleh beberapa kondisi penyakit jantung. Menurut WHO, penyakit jantung atau yang disingkat sebagai *Cardiovascular Disease* (CVD) adalah kelompok gangguan pada jantung dan pembuluh darah yang termasuk diataranya : *coronary heart disease*, *cerebrovascular disease*, *rheumatic heart disease*, dan kondisi jantung lainnya. Lebih dari 4 dari 5 kematian atas CVD disebabkan oleh gagal jantung dan strokes, dan sepertiga dari angka kematian tersebut merupakan orang yang mati dibawah umur 70 (Tamba & -, 2022).

Gejala utama gagal jantung adalah sesak napas, mudah lelah, serta pembengkakan pada kaki dan pergelangan kaki. Gejala ini dapat berkembang secara bertahap atau muncul secara tiba-tiba.

Pencegahan utama gagal jantung adalah dengan menjalani gaya hidup sehat, yaitu dengan mengonsumsi makanan bergizi seimbang, membatasi asupan garam, gula, dan lemak jenuh, serta berolahraga secara rutin. Selain itu, pemeriksaan kesehatan secara rutin, terutama tekanan darah, gula darah, dan kolesterol, juga perlu dilakukan untuk mendeteksi gangguan kesehatan yang dapat menyebabkan gagal jantung (dr. Pittara, 2022).

2.1.1. Faktor-Faktor Penyebab Gagal Jantung

Berikut beberapa hal yang dapat meningkatkan faktor risiko terkena gejala gagal jantung :

1. **Mengalami Penyakit Jantung Koroner**

   Penyakit jantung koroner terjadi karena pembuluh arteri tidak cukup mengalirkan darah ke jantung. Kondisi dapat menimbulkan sejumlah gejala, mulai dari nyeri dada, kesulitan bernapas, dan serangan jantung.

2. **Tekanan Darah Tinggi**

   Tekanan darah tinggi atau hipertensi merupakan kondisi ketika tekanan darah berada di atas batas normal sekitar 130/80 mmHg atau lebih. Kondisi ini mampu mengakibatkan komplikasi penyakit serius, seperti penyakit jantung dan stroke.

3. **Diabetes**

   Penderita diabetes berisiko lebih tinggi mengalami penyakit gagal jantung. Kondisi ini terjadi akibat kadar gula darah tinggi dan tidak terkontrol dengan baik. Diabetes mampu menyebabkan kerusakan pada ginjal dan pembuluh darah jantung. Hal inilah yang mengakibatkan fungsi jantung menurun dan memicu timbulnya gagal jantung (Tim Konten Kesehatan, 2024).

2.2. Prediksi

Prediksi adalah proses untuk meramalkan suatu variabel di masa mendatang dengan berdasarkan pertimbangan data pada masa lampau. Data yang sering digunakan untuk melakukan prediksi adalah data yang berupa data kuantitatif. Prediksi tidak harus memberikan jawaban secara pasti

kejadian yang akan terjadi, melainkan berusaha untuk mencari jawaban sedekat mungkin yang akan terjadi (Neural et al., 2024).

Teori di balik prediksi dalam *machine learning* mencakup beberapa konsep dasar, termasuk pembelajaran dari data, model pembelajaran, dan evaluasi kinerja model (Febrian, 2024).

2.3. Kaggle

Kaggle adalah sebuah komunitas online yang dibentuk oleh Anthony Goldbloom sebagai CEO dan Ben Hamner sebagai CTO di tahun 2010 (Nadiyah Ramalia, 2024). Komunitas online ini menampung para pegiat data *science* yang ingin belajar lebih dalam tentang *machine learning* dan ilmu-ilmu terkait lainnya.

Di dalamnya, ada berbagai kegiatan yang dilakukan, salah satunya yang paling terkenal adalah kompetisi *machine learning*. Kaggle sendiri menyatakan bahwa selain berkompetisi, anggota komunitasnya bisa bersama-sama menulis dan membagikan kode serta mempelajari berbagai hal.

Gambar 2.1 Kaggle
Sumber : Google

Selain ikut kompetisi, pengguna juga bisa mengakses ribuan dataset untuk dijadikan bahan latihan dalam mengasah *skill Data Science*. Karena

banyaknya dataset yang ada, pengguna hampir bisa menemukan dataset tentang topik apapun disini (Fasya Al Rahmah, 2021).

2.3.1. Cara Mengunduh Dataset dari Kaggle

Berikut merupakan tahap-tahap cara mengunduh dataset dari kaggle :

1. **Buat Akun Kaggle**

   Jika belum memiliki akun Kaggle, langkah pertama adalah membuatnya. Kunjungi situs web Kaggle dan daftar menggunakan alamat email. Setelah masuk, maka akan memiliki akses ke berbagai kumpulan data.

2. **Jelajahi Kumpulan Data Kaggle**

   Kaggle menawarkan koleksi kumpulan data yang luas tentang berbagai topik, mulai dari keuangan dan perawatan kesehatan hingga pemrosesan bahasa alami dan visi komputer. Gunakan bilah pencarian dan filter untuk menemukan kumpulan data yang sesuai dengan minat penelitian (Egor Zyryanov, 2024). Manfaatkan tag dan urutkan berdasarkan popularitas atau kebaruan untuk menemukan kumpulan data terkini.

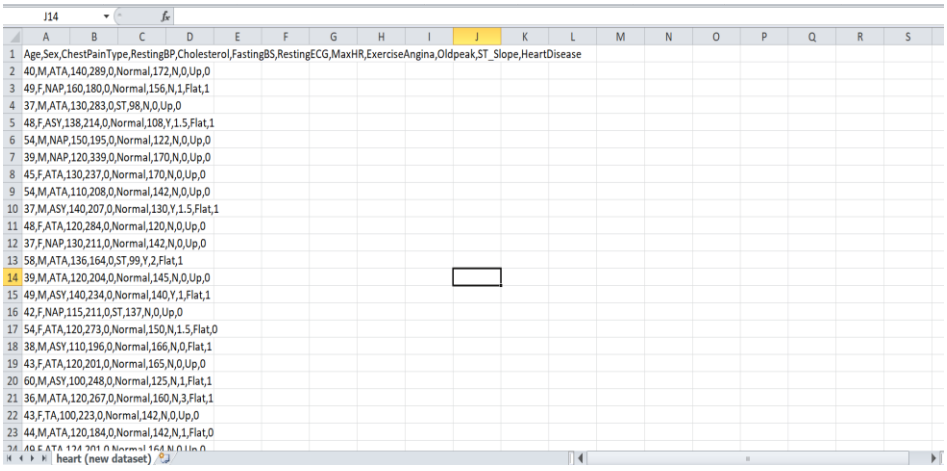
3. **Unduh Dataset**

   Setelah menemukan kumpulan data yang sesuai dengan kebutuhan penelitian, maka dapat mengunduhnya langsung dari Kaggle. Sebagian besar kumpulan data tersedia dalam format

umum seperti CSV. Klik tombol "Unduh" untuk menyimpan kumpulan data ke computer.

2.4. Dataset

Dataset adalah kumpulan data yang berisi beberapa catatan atau *record*, yang sudah diatur dalam format yang terstruktur serta diperoleh dari informasi-informasi tertentu. Dataset biasanya disajikan dalam bentuk tabel, yang setiap kolom menjelaskan variable tertentu. Jadi, dataset terdiri dari beberapa data. Dataset yang diambil dari *kaggle* tersedia dalam berbagai format seperti CSV, JSON, SQLite, dan BigQuery (Team, 2024). Pada penelitian ini dataset yang penulis gunakan yaitu menggunakan format CSV. Penelitian ini menggunakan dataset *Heart Failure Prediction* yang diperoleh dari *Kaggle*.



	A
1	Age,Sex,ChestPainType,RestingBP,Cholesterol,FastingBS,RestingECG,MaxHR,ExerciseAngina,Oldpeak,ST_Slope,HeartDisease
2	40,M,ATA,140,289,0,Normal,172,N,0,Up,0
3	49,F,NAP,160,180,0,Normal,156,N,1,Flat,1
4	37,M,ATA,130,283,0,ST,98,N,0,Up,0
5	48,F,ASY,138,214,0,Normal,108,Y,1.5,Flat,1
6	54,M,NAP,150,195,0,Normal,122,N,0,Up,0
7	39,M,NAP,120,339,0,Normal,170,N,0,Up,0
8	45,F,ATA,130,237,0,Normal,170,N,0,Up,0
9	54,M,ATA,110,208,0,Normal,142,N,0,Up,0
10	37,M,ASY,140,207,0,Normal,130,Y,1.5,Flat,1
11	48,F,ATA,120,284,0,Normal,120,N,0,Up,0
12	37,F,NAP,130,211,0,Normal,142,N,0,Up,0
13	58,M,ATA,136,164,0,ST,99,Y,2,Flat,1
14	39,M,ATA,120,204,0,Normal,145,N,0,Up,0
15	49,M,ASY,140,234,0,Normal,140,Y,1,Flat,1
16	42,F,NAP,115,211,0,ST,137,N,0,Up,0
17	54,F,ATA,120,273,0,Normal,150,N,1.5,Flat,0
18	38,M,ASY,110,196,0,Normal,166,N,0,Flat,1
19	43,F,ATA,120,201,0,Normal,165,N,0,Up,0
20	60,M,ASY,100,248,0,Normal,125,N,1,Flat,1
21	36,M,ATA,120,267,0,Normal,160,N,3,Flat,1
22	43,F,TA,100,223,0,Normal,142,N,0,Up,0
23	44,M,ATA,120,184,0,Normal,142,N,1,Flat,0

Gambar 2.2 Dataset Format CSV

2.5. Pemrograman Python

Python merupakan bahasa pemrograman komputer yang biasa dipakai untuk membangun situs, *software*/aplikasi, mengotomatiskan tugas dan melakukan analisis data. Bahasa pemrograman ini termasuk bahasa tujuan

umum. Artinya, *python* bisa digunakan untuk membuat berbagai program berbeda, bukan khusus untuk masalah tertentu saja.

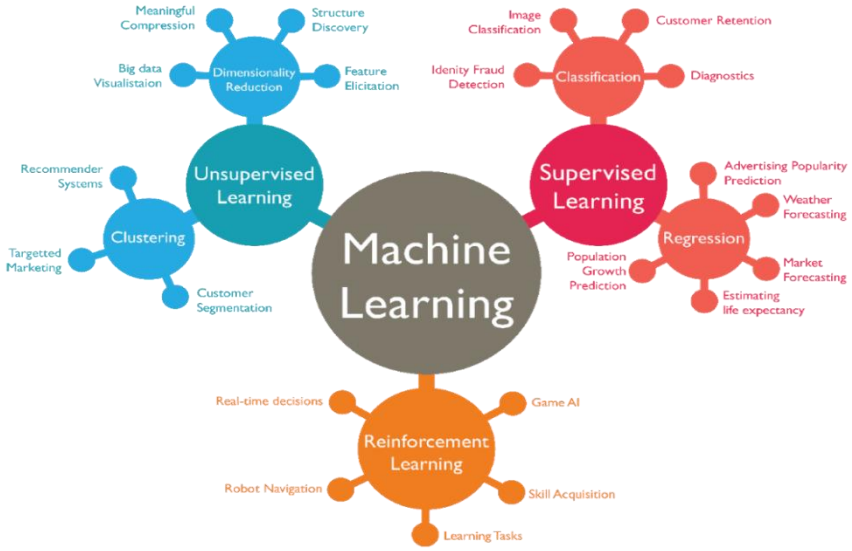
Gambar 2.3 *Python*
Sumber : *Google*

Python telah menjadi andalan dalam ilmu data. Bahasa pemrograman ini memungkinkan analisis data untuk melakukan perhitungan statistik yang rumit, membuat visualisasi data serta algoritma *machine learning*. *Python* juga bisa digunakan untuk memanipulasi, menganalisis data, dan menyelesaikan berbagai tugas lain terkait data. Selain itu, *python* bisa membantu membangun berbagai visualisasi data yang berbeda. Misalnya, grafik garis, batang, diagram lingkaran, histogram, dan lain sebagainya (Dicoding Intern, 2023).

2.6. Machine Learning

Machine learning adalah pembelajaran mesin yang ditujukan untuk memahami dan membangun suatu metode yang memanfaatkan data untuk meningkatkan kinerja dan dikembangkan untuk bisa mendapatkan berbagai informasi secara mandiri. Jadi pengembang hanya perlu memprogramnya sekali dengan algoritma tertentu (Meilinaeka, 2022).

Dengan *machine learning*, komputer dapat mempelajari pola dari data yang ada, membuat prediksi, dan mengambil keputusan tanpa memerlukan program yang dirancang khusus untuk setiap tugas. *Machine learning* tidak hanya memungkinkan sistem untuk membuat keputusan yang lebih baik, tetapi juga memungkinkan prediksi yang lebih akurat.



Gambar 2.4 Model-Model *Machine Learning*

Sumber : *Google*

2.6.1. Model – Model *Machine Learning*

Model-model *machine learning* terbagi menjadi tiga kategori: *supervised learning*, *unsupervised learning*, dan *reinforcement learning.*

1. Supervised Learning

Supervised learning adalah model *machine learning* yang paling mudah dipahami. Proses *learning* dalam model *supervised* meliputi pembuatan fungsi yang dapat dilatih menggunakan data set latihan, kemudian diaplikasikan ke data

yang baru untuk menciptakan prediksi terhadap data tersebut. Tujuannya adalah untuk membangun fungsi yang mampu mengeneralisir data yang belum dilihat sebelumnya.

Ada sejumlah algoritma yang ada pada *supervised learning*, yaitu *Logistic Regression, K-nearest neighbor* (KNN), *Support vector machine* (SVM), *Random forest*, dan beberapa algoritma lainnya.

2. Unsupervised Learning

Model *machine learning* yang satu ini juga tergolong sederhana. Namun, seperti namanya, metode *learning* ini tidak memiliki *feedback* sehingga tidak ada ukuran untuk performanya. Tujuan metode ini adalah untuk membangun fungsi *mapping* yang mengategorikan data menjadi beberapa kelas berdasarkan ciri yang tersembunyi dalam data yang diproses. Pengimplementasian *unsupervised learning* biasanya menggunakan beragam algoritma, yaitu *K-Means Clustering, Principal Component Analysis, Deep Belief Network*, dan beberapa algoritma lainnya.

3. Reinforcement Learning

Reinforcement learning memiliki kemampuan tidak hanya mempelajari *input* dan *output* suatu data, tetapi juga memetakan serangkaian *input* dan output dengan dependensi. Selama proses *learning*, algoritma secara acak mengeksplorasi pasangan keadaan-tindakan dalam beberapa lingkungan (untuk

membangun tabel pasangan keadaan-tindakan). Salah satu contoh algoritma yang menggunakan *reinforcement learning* adalah *Q-learning* (algoritma, 2022).

2.7. Algoritma *Logistic Regression*

Logistic Regression adalah sebuah metode yang digunakan untuk masalah regresi dan klasifikasi. Metode ini menggunakan fungsi logistik untuk memodelkan hubungan antara atribut independen dengan probabilitas klasifikasi data kategorikal (Azzahra et al., 2024). *Logistic Regression* menunjukkan keterkaitan antara output dalam bentuk pengklasifikasian biner terhadap variabel-variabel independen berdasarkan probabilitas dengan memprediksi nilai variabel dependen (Mohammad Fahry Sholahuddin et al., 2023).

Logistic Regression merupakan algoritma klasifikasi yang digunakan untuk memprediksi probabilitas variabel dependen kategori. Variabel terikat dalam *Logistic Regression* adalah variabel biner yang memiliki nilai 1 (ya) atau 0 (tidak). Untuk klasifikasi biner, *Logistic Regression* adalah algoritma statistik yang tujuan utamanya adalah memprediksi kemungkinan bahwa suatu contoh akan masuk ke salah satu dari dua kelas. Dalam konteks prediksi gagal jantung, *Logistic Regression* mencoba memprediksi apakah seorang pasien memiliki resiko gagal jantung (positif) atau tidak (negatif). Fungsi logistik sigmoid atau logistik yang membatasi *output* menjadi nilai antara 0 dan 1. Fungsi probabilitas sigmoid yang mengubah input linear menjadi probabilitas dan mengestimasi parameter. Untuk mengevaluasi ikatan antara sejumlah

variabel dan variabel biner atau acak, *Logistic Regression* biner adalah teknik analisis data yang umum. Variabel respon biner (y) dan variabel prediktor (x) terdiri dari dua ketegori sukses dan gagal, yang diwakili oleh nilai y=1 (sukses) dan nilai y=(gagal) (Setyawan & Wakhidah, 2025).

2.7.1. Kelebihan *Logistic Regression*

1. Cepat dan mudah digunakan.
2. Hasilnya mudah dijelaskan dan bisa tahu seberapa besar pengaruh tiap faktor (fitur) terhadap hasil prediksi.
3. Memiliki performa yang baik pada dataset yang *linearly separable* (dapat dipisahkan secara linear).

2.7.2. Kekurangan *Logistic Regression*

1. Tidak bekerja dengan baik jika hubungan antara variabel tidak linear (kecuali ditransformasikan).
2. Jika fitur saling berkorelasi, hasil model bisa menjadi tidak stabil.
3. Asumsi dasar dari model dapat membatasi fleksibilitasnya pada data nyata yang kompleks.

2.8. Algoritma *K- Nearest Neighbor* (KNN)

K-Nearest Neighbors (KNN) adalah metode pembelajaran *instance* dalam *supervised learning* yang termasuk dalam teknik *lazy learning* (Mohammad Fahry Sholahuddin et al., 2023). Metode KNN adalah sebuah metode pembelajaran yang didasarkan oleh perwujudan, yang mana fungsi-fungsinya merupakan nilai pendekatan secara lokal dan segala perhitungan ditahan sampai proses klasifikasi. Data yang dilatih

diproyeksikan ke dalam ruang yang memiliki banyak dimensi, di mana masing masing dimensi mengekstrak fitur dari data. Dekat atau jauhnya tetangga biasanya dihitung berdasarkan jarak *Euclidean* (Simanjuntak et al., 2022).

Berikut penjelasan setiap tahapan dari algoritma KNN :

1. Menentukan jumlah tetangga terdekat yang disimbolkan dengan nilai parameter K. Nilai pada parameter K yang akurat untuk algoritma ini tergantung pada data training yang digunakan.
2. Menghitung dan menentukan nilai kuadrat jarak data testing objek terhadap data training yang diberikan dengan menggunakan persamaan 1.
3. Melakukan pengurutan dari hasil perhitungan no 2 secara ascending (urut dari nilai rendah ke nilai tinggi).
4. Mengumpulkandata pada kategori Y (Klasifikasi tetangga terdekat berdasarkan nilai K).
5. Kategori Y yang paling banyak muncul menjadi hasil akhir (Cholil et al., 2021).

2.8.1. Kelebihan *K-Nearest Neighbor*

1. Fleksibel untuk data linear maupun non-linear karena tidak mengasumsikan bentuk hubungan antar fitur.
2. Bisa digunakan untuk data yang bentuknya tidak beraturan.

2.8.2. Kekurangan *K-Nearest Neighbor*

1. Proses prediksi lambat karena harus menghitung jarak ke semua data latih.

2. Titik data yang menyimpang dapat mempengaruhi hasil klasifikasi.
3. Harus distandarisasi terlebih dahulu karena menggunakan perhitungan jarak.

2.9. Confusion Matrix

Confusion matrix berfungsi untuk mengevaluasi keampuhan metode klasifikasi dalam memprediksi kelas data dimana teknik ini memberikan perbandingan nilai kelas aslinya dengan nilai prediksi (Habibi et al., 2023). Contoh tabel *Confusion Matrix* dapat dilihat di bawah ini.

Tabel 2.1 Confusion Matrix

	ACTUAL	
PREDICTED	*FALSE*	*TRUE*
FALSE	*TN (True Negative)*	*FP (False Positive)*
TRUE	*FN (False Negative)*	*TP (True Positive)*

Pada tabel 2.1 *confusion matrix* terdapat empat nilai yang dikeluarkan antara lain :

1. *True Positive* (TP), ialah nilai terkini yang bernilai positif dan terprediksi benar.
2. *True Negative* (TN), ialah nilai terkini yang bernilai negatif dan terprediksi benar.
3. *False Positive* (FP), ialah nilai terkini yang bernilai negatif dan terprediksi positif.
4. *False Negative* (FN), ialah nilai terkini yang bernilai positif dan terprediksi negatif.

Melalui 4 data tersebut, dapat diperoleh data data lain yang sangat berguna untuk mengukur perfoma sebuah model, diantaranya:

1. *Accuracy* merupakan presentase jumlah data yang dilakukan pada klasifikasi atau prediksi secara benar oleh algoritma. Berikut merupakan persamaan dari *accuracy*.

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

2. *Precision* yaitu nilai dari ketepatan dari metode yang digunakan dalam klasifikasi. Nilai tersebut menunjukkan banyaknya data yang dapat terklasifikasi di kelas yang benar dalam beberapa pengujian. Berikut merupakan persamaan dari *precision*.

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

3. *Recall* yaitu nilai yang dapat mengukur hasil berapa presentase data yang terklasifikasikan dengan benar. Berikut merupakan persamaan dari *Recall*.

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

4. F1 *score* digunakan untuk mengukur nilai rata-rata harmonik dari nilai *precision* dan *recall*. Berikut merupakan persamaan dari F1 *score*.

$$\text{F1 } Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (6)$$

2.10. Penelitian Terdahulu

Tabel 2.2 Peneltian Terdahulu

No.	Nama Peneliti	Judul	Metode	Kesimpulan
1.	Nur Devita, Ambarwati, Anita Desiani, Sri Indra, Indri Ramayanti (2024)	Perbandingan Algoritma *K-Nearest Neighbor* Dan *Logistic Regression* dalam Klasifikasi Penyakit Kanker Serviks	*K-Nearest Neighbor* dan *Logistic Regression*	Pada perbandingan tersebut KNN menghasilkan nilai akurasi sebesar 91% dengan teknik *percentage split* dan 93% dengan *K-fold cross validation. Logistic Regression* memperoleh nilai akurasi 94% dengan *percentage split* dan pada *K-fold cross validation* akurasi yang diperoleh sebesar 96%.
2.	Kadek Adi, Awaldi Rizki, Dody Kristianto, Herman RuswanSuwarman (2024)	Prediksi Gagal Jantung Berbasis *Machine learning* Menggunakan *Support Vector*	*Support Vector Machine* dan Regresi Logistik	Pada Penelitian ini Regresi Logistik menghasilkan akurasi 82% dengan *precision* 0.80, *recall* 0.85, dan F1-*Score* 0.82.

		Machine dan Regresi Logistik		Sebaliknya, SVM menunjukkan performa lebih baik dengan akurasi 85%, *precision* 0.82, *recall* 0.88, dan F1-*Score* 0.85, serta rata-rata probabilitas prediksi 0.87.
3.	Windari Oktapia, Arif Bijaksana, Rina Septriana (2023)	Perbandingan Algoritma *Logistic Regression* dan *Random Foret* (Studi Kasus : Klasifikasi Emosi Tweet)	*Logistic Regression* dan *Random Forest*	Hasil penelitian ini menunjukkan bahwa model klasifikasi *Logistic Regresion* memiliki nilai *accuracy* tertinggi sebesar 78.22%. Sedangkan model klasifikasi *Random Forest* memiliki nilai *accuracy* tertinggi sebesar 72.41%.
4.	Mohammad Fahry, Abdul Holik, Chelvin Suprapto, Iqbal Izha Mahendra, Sadewa	Perbandingan Model *Logistic Regression* dan *K-Nearest Neighbors* Dalam	*Logistic Regression* dan *K-Nearest Neighbors*	Pada penelitian ini *k-nearest neighbors* memiliki nilai akurasi, presisi, dan *recall* yang lebih tinggi

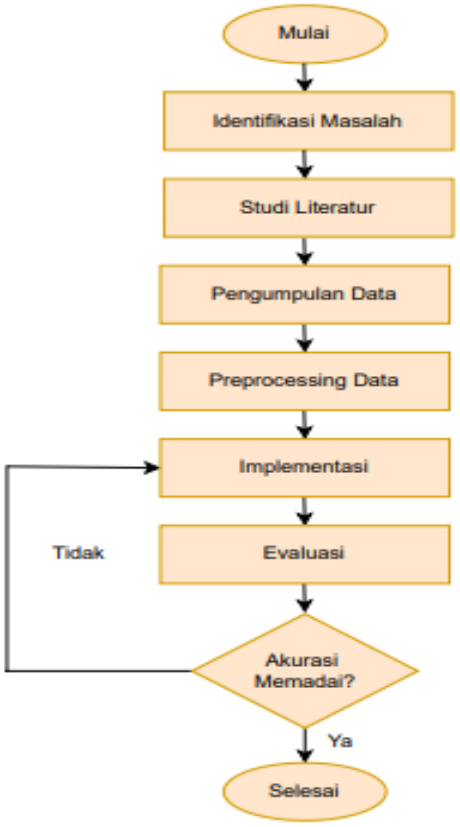
	Wibawanto, Muchamad Kurniawan (2023)	Prediksi Pembatalan Hotel		dibandingkan dengan Regresi Logistik, dengan nilai akurasi sebesar 0.81, presisi 0.81, dan *recall* 0.80.

BAB III

METODOLOGI PENELITIAN

3.1. Desain Penelitian

Pada penelitian ini menggunakan pendekatan kuantitatif. Pendekatan kuantitatif digunakan untuk menganalisis data orang yang terkena penyakit jantung dengan menerapkan metode *Logistic Regression* dan *K-Nearest Neighbor* (KNN). Kemudian kedua metode tersebut akan dibandingkan untuk mengetahui metode mana yang lebih akurat untuk memprediksi berdasarkan dari nilai akurasi kedua metode tersebut.



Gambar 3.1 Alur Penelitian

3.1.1. Identifikasi Masalah

Permasalahan pada penelitian ini adalah tingginya angka kematian yang disebabkan gagal jantung, tantangan utama dalam penanganan gagal jantung adalah diagnosis yang terlambat, gagal jantung dapat disebabkan oleh berbagai faktor dan diperlukannya deteksi dini untuk meningkatkan peluang bertahan hidup.

3.1.2. Studi Literatur

Melakukan kajian pustaka yang menggunakan berbagai algoritma *machine learning* untuk prediksi sesuatu, dan memahami kelebihan serta kelemahan dari masing-masing algoritma. Penelitian ini menggunakan algoritma *Logistic Regression dan K-Nearest Neighbor (KNN)* dikarenakan kedua algoritma tersebut sering digunakan untuk memprediksi sesuatu dalam bidang kesehatan.

3.1.3. Pengumpulan Data

Penelitian ini menggunakan dataset *Heart Failure Prediction* yang diperoleh dari Kaggle. Dataset ini dibuat dengan menggabungkan berbagai dataset yang sebelumnya tersedia secara terpisah, namun belum pernah digabungkan sebelumnya. Dalam dataset ini, lima dataset tentang penyakit jantung digabungkan berdasarkan 11 fitur umum. Lima dataset yang digunakan untuk penyusunannya adalah sebagai berikut: Cleveland dengan 303 observasi, Hungarian dengan 294

observasi, Switzerland dengan 123 observasi, dan Long Beach VA dengan 198 observasi.

Tabel 3.1 Variabel Dataset

No.	Variabel	Keterangan	Contoh Data
1.	Age	Umur	40
2.	Sex	Jenis kelamin	**M**= Laki-laki **F**= Perempuan
3.	ChestPainType	Jenis nyeri dada	**TA**= *Typical Angina* **ATA**=*Atypical Angina* **NAP**= *Non-Anginal Pain* **ASY**= *Asymptomatic*
4.	RestingBP	Tekanan darah saat istirahat	140 [mm Hg]
5.	Cholestrol	Kadar kolesterol pasien	195 [mm/dl]
6.	FastingBS	Kadar gula darah saat puasa	**1**= *FastingBS* >120 mm/dl **0**= *FastingBS*<120 mm/dl
7.	RestingECG	Hasil ECG saat pasien istirahat(berbaring)	**Normal**= Normal **ST**= Gelombang ST-T yang *abnormal* dan elevasi atau depresi> 0.05 mV **LVH**= menunjukan *hypertrophy ventricular* kiri yang mungkin atau pasti berdasarkan kriteria Estes
8.	MaxHR	Detak jantung	130

		maksimal	
9.	ExerciseAngina	Nyeri dada yang diakibatkan oleh olahraga	**Y** = Yes **N** = No
10.	Oldpeak	Hasil depresi ST	1,2 mm
11.	ST_Slope	Kemiringan dari puncak segmen ST	***Up*** *= upsloping* ***Flat*** *= flat* ***Down*** *=downsloping*
12.	Target	Resiko gagal jantung	1= resiko 0= tidak resiko

3.1.4. Preprocessing Data

Tahap ini sangat penting untuk memastikan bahwa data yang digunakan bersih, relevan dan siap untuk diolah, meliput:

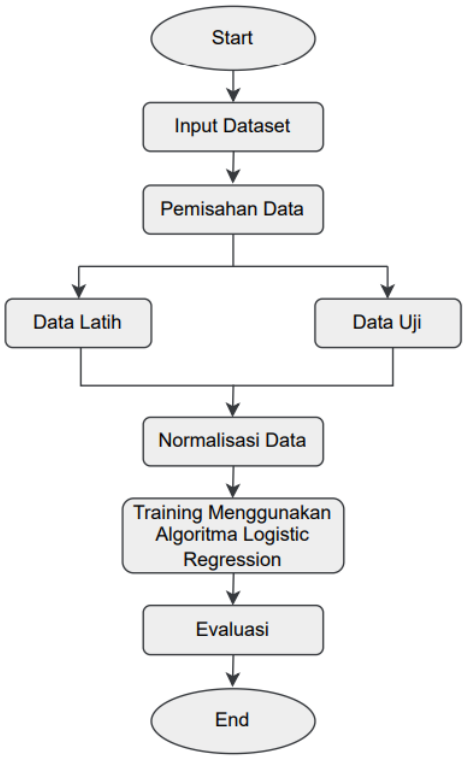
1. Membagi data menjadi 2 yaitu data *training* (data latih) dan data *testing* (data uji).
2. Normalisasi data untuk menghilangkan redundansi data (pengulangan) dan menstandarisasi informasi untuk alur kerja data yang lebih baik.

3.1.5. Implementasi

Pada tahap ini, dilakukan penerapan algoritma *Logistic Regression* dan *K-Nearest Neighbor* untuk mendapatkan nilai akurasi dari kedua algoritma tersebut. Implementasi ini mencakup beberapa langkah utama, yaitu pemodelan, pelatihan, serta

mengetahui hasil prediksi berdasarkan data yang telah diproses sebelumnya.

1. Algoritma Logistic Regression

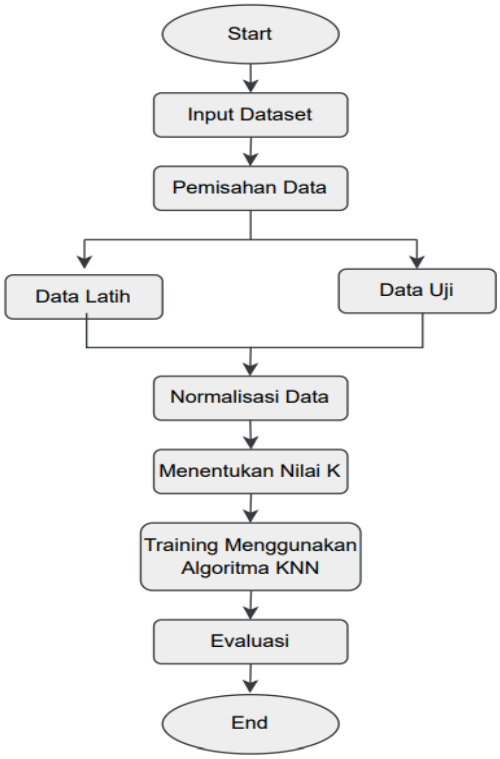


Gambar 3.2 Flowchart Algoritma Logistic Regression

Berdasarkan *flowchart* diatas proses dimulai dari input dataset yang akan digunakan. Selanjutnya yaitu tahap pemisahan data training dan data testing. Setelah itu, lanjut ke tahap normalisasi data. Normalisasi data adalah menghilangkan redundansi data (pengulangan) dan menstandarisasi informasi untuk alur kerja data yang lebih baik. Kemudian dataset yang sudah dibagi menjadi data pelatihan (*training*) dan data percobaan (*testing*) diterapkan ke model. Tahap selanjutnya

adalah evaluasi model untuk menghitung akurasi algoritma. Dalam evaluasi, data *testing* yang akan dihitung adalah *accuracy*, *precision*, *recall*, dan F-1 *score* nya.

2. Algoritma *K-Nearest Neighbor*



Gambar 3.3 Flowchart K-Nearest Neighbor

Berdasarkan *flowchart* diatas proses dimulai dari input dataset yang akan digunakan. Selanjutnya yaitu tahap pemisahan data training dan data testing. Setelah itu, lanjut ke tahap normalisasi data. Normalisasi data adalah menghilangkan redundansi data (pengulangan) dan menstandarisasi informasi untuk alur kerja data yang lebih baik. Setelah melakukan normalisasi data, masukkan nilai K yang telah ditentukan yaitu jumlah tetangga terdekat yang akan digunakan dalam algoritma

K-NN untuk membuat prediksi. Berdasarkan jumlah tetangga terdekat ini, dilakukanlah prediksi. Tahap selanjutnya adalah evaluasi model untuk menghitung akurasi algoritma. Dalam evaluasi, data *testing* yang akan dihitung adalah *accuracy, precision, recall,* dan F-1 *score* nya.

3.1.6. Evaluasi

Tahap evaluasi bertujuan untuk mengukur seberapa baik model diuji pada data uji untuk menilai kemampuannya dalam memprediksi resiko gagal jantung dengan menggunakan metrik evaluasi seperti *accuracy, precision, recall,* dan F-1 *score*.

3.2. Jadwal Penelitian

Tabel 3.2 Jadwal Penelitian

No.	Kegiatan Penelitian	Tahun 2025					
		Februari	Maret	April	Mei	Juni	Juli
1.	Penyusunan Proposal						
2.	Seminar Proposal						
3.	Preprocessing Data						
4.	Evaluasi						
5.	Sidang						

BAB IV

HASIL DAN PEMBAHASAN

4.1. Deskripsi Data

Dataset yang digunakan dalam penelitian ini berjudul *Heart Failure Prediction Dataset* yang terdiri dari 918 data pasien. Dataset ini mencakup 12 atribut, yaitu 11 atribut fitur dan 1 atribut target.

Tabel 4.1 *Heart Failure Prediction Dataset*

No	Age	Sex	Chest Pain Type	Resting BP	……	ST Slope	Heart Disease
1	40	M	ATA	140	……	Up	0
2	49	F	NAP	160	……	Flat	1
3	37	M	ATA	130	……	Up	0
4	48	F	ASY	138	……	Flat	1
5	54	M	NAP	150	……	Up	0
6	39	M	NAP	120	……	Up	0
7	45	F	ATA	130	……	Up	0
8	54	M	ATA	110	……	Up	0
9	37	M	ASY	140	……	Flat	1
10	48	F	ATA	120	……	Up	0
11	37	F	NAP	130	……	Up	0
12	58	M	ATA	136	……	Flat	1
13	39	M	ATA	120	……	Up	0
14	49	M	ASY	140	……	Flat	1
15	42	F	NAP	115	……	Up	0
16	54	F	ATA	120	……	Flat	0
17	38	M	ASY	110	……	Flat	1
18	43	F	ATA	120	……	Up	0
19	60	M	ASY	100	……	Flat	1
20	36	M	ATA	120	……	Flat	1
21	43	F	TA	100	……	Up	0
22	44	M	ATA	120	……	Flat	0
23	49	F	ATA	124	……	Up	0
24	44	M	ATA	150	……	Flat	1
25	40	M	NAP	130	……	Up	0
26	36	M	NAP	130	……	Up	0
27	53	M	ASY	124	……	Flat	0

28	52	M	ATA	120	……	Up	0
29	53	F	ATA	113	……	Up	0
30	51	M	ATA	125	……	Up	0
31	53	M	NAP	145	……	Flat	1
32	56	M	NAP	130	……	Up	0
33	54	M	ASY	125	……	Flat	1
34	41	M	ASY	130	……	Flat	1
35	43	F	ATA	150	……	Up	0
36	32	M	ATA	125	……	Up	0
37	65	M	ASY	140	……	Flat	1
38	41	F	ATA	110	……	Up	0
39	48	F	ATA	120	……	Up	0
40	48	F	ASY	150	……	Flat	0
41	54	F	ATA	150	……	Up	0
42	54	F	NAP	130	……	Flat	1
43	35	M	ATA	150	……	Up	0
44	52	M	NAP	140	……	Up	0
45	43	M	ASY	120	……	Flat	1
46	59	M	NAP	130	……	Flat	0
47	37	M	ASY	120	……	Up	0
48	50	M	ATA	140	……	Up	0
49	36	M	NAP	112	……	Flat	0
50	41	M	ASY	110	……	Flat	1
51	50	M	ASY	130	……	Flat	1
52	47	F	ASY	120	……	Flat	1
53	45	M	ATA	140	……	Up	0
54	41	F	ATA	130	……	Up	0
55	52	F	ASY	130	……	Flat	0
56	51	F	ATA	160	……	Up	0
57	31	M	ASY	120	……	Flat	1
58	58	M	NAP	130	……	Flat	1
59	54	M	ASY	150	……	Up	0
60	52	M	ASY	112	……	Flat	1
61	49	M	ATA	100	……	Up	0
62	43	F	NAP	150	……	Up	0
63	45	M	ASY	140	……	Up	0
64	46	M	ASY	120	……	Flat	1
65	50	F	ATA	110	……	Up	0
66	37	F	ATA	120	……	Up	0
67	45	F	ASY	132	……	Up	0
68	32	M	ATA	110	……	Up	0

69	52	M	ASY	160	……	Flat	1
70	44	M	ASY	150	……	Up	0
71	57	M	ATA	140	……	Flat	1
72	44	M	ATA	130	……	Up	0
73	52	M	ASY	120	……	Flat	1
74	44	F	ASY	120	……	Up	0
75	55	M	ASY	140	……	Flat	1
76	46	M	NAP	150	……	Up	0
77	32	M	ASY	118	……	Flat	1
78	35	F	ASY	140	……	Up	0
79	52	M	ATA	140	……	Up	0
80	49	M	ASY	130	……	Flat	1
81	55	M	NAP	110	……	Up	0
82	54	M	ATA	120	……	Up	0
83	63	M	ASY	150	……	Flat	1
84	52	M	ATA	160	……	Up	0
85	56	M	ASY	150	……	Flat	1
86	66	M	ASY	140	……	Flat	1
87	65	M	ASY	170	……	Flat	1
88	53	F	ATA	140	……	Flat	0
89	43	M	TA	120	……	Flat	1
90	55	M	ASY	140	……	Flat	0
91	49	F	ATA	110	……	Up	0
92	39	M	ASY	130	……	Up	0
93	52	F	ATA	120	……	Up	0
94	48	M	ASY	160	……	Flat	1
95	39	F	NAP	110	……	Up	0
96	58	M	ASY	130	……	Flat	1
97	43	M	ATA	142	……	Up	0
98	39	M	NAP	160	……	Up	0
99	56	M	ASY	120	……	Up	0
100	41	M	ATA	125	……	Up	0
….	….	….	….	….	……	….	….
901	58	M	ASY	114	……	Down	1
902	58	F	ASY	170	……	Flat	1
903	58	M	ATA	125	……	Flat	0
904	56	M	ATA	130	……	Up	0
905	56	M	ATA	120	……	Down	0
906	67	M	NAP	152	……	Flat	1
907	55	F	ATA	132	……	Up	0
908	44	M	ASY	120	……	Down	1

909	63	M	ASY	140	……	Up	1
910	63	F	ASY	124	……	Flat	1
911	41	M	ATA	120	……	Up	0
912	59	M	ASY	164	……	Flat	1
913	57	F	ASY	140	……	Flat	1
914	45	M	TA	110	……	Flat	1
915	68	M	ASY	144	……	Flat	1
916	57	M	ASY	130	……	Flat	1
917	57	F	ATA	130	……	Flat	1
918	38	M	NAP	138	……	Up	0

4.2. Import Library

Untuk melakukan prediksi dan perbandingan performa model, dilakukan implementasi dua algoritma yaitu *Logistic Regression* dan *K-Nearest Neighbor (KNN)* menggunakan bahasa pemrograman *Python* dengan *library* pendukung dari *Scikit-learn*, *Pandas*, dan *Matplotlib.*

```
1  import numpy as np
2  import pandas as pd
3  import matplotlib.pyplot as plt
4  import seaborn as sns
5  import time
6  from matplotlib import pyplot as plt
7  from sklearn.model_selection import train_test_split
8  from sklearn.preprocessing import MinMaxScaler
9  from sklearn.linear_model import LogisticRegression
10 from sklearn.neighbors import KNeighborsClassifier
11 from sklearn.metrics import classification_report, confusion_matrix, accuracy_score
```

Gambar 4.1 *Library*

4.3. Membaca Data

Setelah library di *import*, langkah selanjutnya adalah membaca dataset ke dalam *Python* menggunakan *library Pandas*. Dataset disimpan dalam format csv dan dibaca menggunakan fungsi *read_csv*().

	Age	Sex	ChestPainType	RestingBP	Cholesterol	...	MaxHR	ExerciseAngina	Oldpeak	ST_Slope	HeartDisease
0	40	M	ATA	140	289	...	172	N	0.0	Up	0
1	49	F	NAP	160	180	...	156	N	1.0	Flat	1
2	37	M	ATA	130	283	...	98	N	0.0	Up	0
3	48	F	ASY	138	214	...	108	Y	1.5	Flat	1
4	54	M	NAP	150	195	...	122	N	0.0	Up	0

Gambar 4.2 Lima Data Teratas

4.4. Pra-pemrosesan Data

Sebelum dilakukan pelatihan model, dataset mengalami beberapa tahapan:

1. *One-Hot Encoding*

Fitur kategorikal seperti *Sex*, *ChestPainType*, *RestingECG*, *ExerciseAngina*, dan *ST_Slope* diubah menjadi representasi numerik menggunakan metode *One-Hot Encoding* agar dapat diproses oleh algoritma *machine learning*.

```
[5 rows x 12 columns]
Age             0
Sex             0
ChestPainType   0
RestingBP       0
Cholesterol     0
FastingBS       0
RestingECG      0
MaxHR           0
ExerciseAngina  0
Oldpeak         0
ST_Slope        0
HeartDisease    0
dtype: int64
```

Gambar 4.3 Jumlah Data Kosong

2. Pembagian Data

Data dibagi menjadi data latih dan uji dengan rasio 70:30 menggunakan *train_test_split*, menghasilkan data latih 70% dari total data dan data uji 30% dari total data.

3. Normalisasi Data

Fitur numerik dinormalisasi menggunakan *Min-Max Scaler* untuk membuat skala nilai berada antara 0 dan 1, agar performa algoritma seperti KNN tidak bias terhadap fitur berskala besar.

```
Data Training Setelah Normalisasi:
 [[0.47916667 0.8        0.58687943 ... 0.         1.         0.        ]
 [0.33333333 0.55       0.46808511 ... 0.         1.         0.        ]
 [0.1875     0.55       0.51241135 ... 1.         0.         0.        ]
 ...
 [0.6875     0.64       0.36879433 ... 0.         0.         1.        ]
 [0.125      0.6        0.54609929 ... 0.         0.         1.        ]
 [0.3125     0.56       0.5141844  ... 0.         0.         1.        ]]


Data Testing Seetelah Normalisasi:
 [[0.25       0.6        0.59574468 ... 0.         1.         0.        ]
 [0.39583333 0.5        0.28191489 ... 0.         0.         1.        ]
 [0.64583333 0.65       0.         ... 1.         0.         0.        ]
 ...
 [0.41666667 0.65       0.60460993 ... 0.         1.         0.        ]
 [0.79166667 0.7        0.38829787 ... 0.         1.         0.        ]
 [0.75       0.75       0.39893617 ... 0.         1.         0.        ]]
```

Gambar 4.4 Data *Training* dan *Testing* Setelah Normalisasi

4.5. Implementasi Algoritma

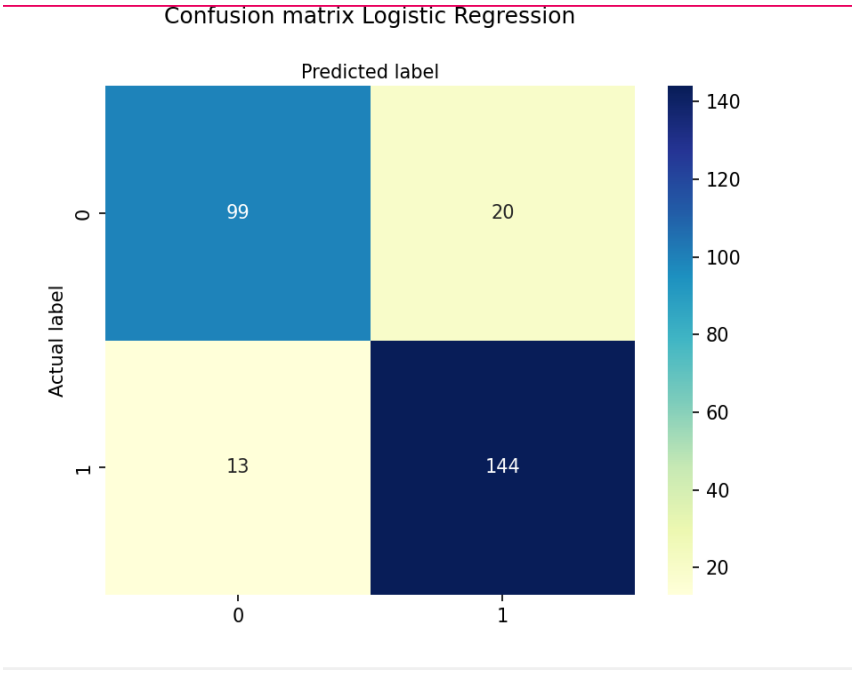
Pengimplementasian algoritma *Logistic Regression* pada prediksi gagal jantung dilakukan menggunakan modul *Scikit-learn* yang di dalamnya terdapat modul *Logistic Regression*. Mengimport modul *Logistic Regression* kemudian membuat objek *classifier* menggunakan fungsi *Logistic Regression*(). Setelah itu memasukkan data training (data latih) ke dalam fungsi *Logistic Regression* menggunakan fungsi fit() dan

dilakukan prediksi pada data testing (data uji) menggunakan fungsi *predict*().

4.6. Evaluasi Perbandingan

Untuk mengevaluasi algoritma *Logistic Regression* dan *K-Nearest Neighbor* penulis menggunakan metode *Confusion Matrix*. *Confusion Matrix* adalah matriks atau tabel yang berfungsi untuk mengevaluasi kinerja model klasifikasi. Untuk melihat bentuk *Confusion Matrix* dari data menggunakan fungsi *confusion_matrix*. Kemudian memvisualisasikannya menggunakan *library seaborn* dan *matplotlib*. Untuk memvisualisasikannya penulis menggunakan fungsi *heatmap*().

Berikut adalah *Confusion Matrix* dari masing-masing algoritma :



Gambar 4.5 Visualisasi *Confusion Matrix Logistic Regression*

1. $Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{144+99}{144+99+20+13} = \frac{243}{276} = 0{,}8804 = 88{,}04\% = 88\%$

Artinya: 88% dari seluruh prediksi model adalah benar (baik yang gagal jantung maupun tidak gagal jantung).

2. $Precision = \frac{TP}{TP+FP} = \frac{144}{144+20} = \frac{144}{164} = 0{,}8780 = 87{,}80\% = 88\%$

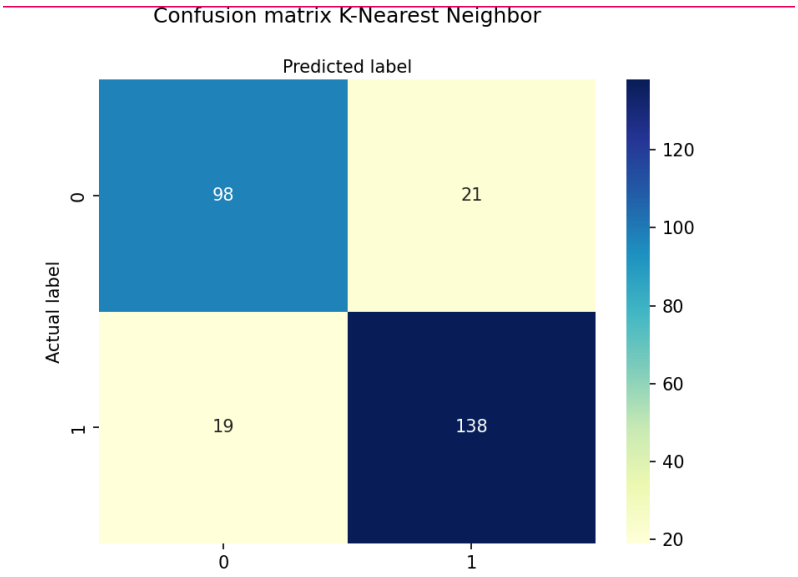
   Artinya: 88% prediksi "gagal jantung" memang benar-benar gagal jantung

3. $Recall = \frac{TP}{TP+FN} = \frac{144}{144+13} = \frac{144}{157} = 0{,}9171 = 91{,}71\% = 92\%$

   Artinya: dari semua yang seharusnya gagal jantung, hanya 92% berhasil diprediksi oleh model.

4. F1–Score $= 2 \text{ x } \frac{Precision \text{ x } Recall}{Precision + Recall} = 2 \text{ x } \frac{0{,}8780 \text{ x } 0{,}9171}{0{,}8780 \text{ x } 0{,}9171} = 2 \text{ x } \frac{0{,}8052}{1{,}7951} =$ $2 \text{ x } 0{,}4485 = \quad 0{,}8970\% = 89{,}70\% = 90\%$

   F1-Score adalah rata-rata harmonis antara *Precision* dan *Recall*, seimbang antara kualitas prediksi dan kelengkapannya.



Gambar 4.5 Visualisasi *Confusion Matrix K-Nearest Neighbor*

1. *Accuracy* $= \frac{TP+TN}{TP+TN+FP+FN} = \frac{138+98}{138+98+21+19} = \frac{236}{276} = 0{,}8550 = 85{,}50\% =$ 86%

Artinya: 86% dari seluruh prediksi model adalah benar (baik yang gagal jantung maupun tidak gagal jantung).

2. $Precision = \frac{TP}{TP+FP} = \frac{138}{138+21} = \frac{138}{159} = 0{,}8679 = 86{,}79\% = 87\%$

   Artinya: 87% prediksi "gagal jantung" memang benar-benar gagal jantung.

3. $Recall = \frac{TP}{TP+FN} = \frac{138}{138+19} = \frac{138}{157} = 0{,}8789 = 87{,}89\% = 88\%$

   Artinya: dari semua yang seharusnya gagal jantung, hanya 88% berhasil diprediksi oleh model.

4. $F1\text{–}Score = 2 \text{ x } \frac{Precision \text{ x } Recall}{Precision + Recall} = 2 \text{ x } \frac{0{,}8679 \text{ x } 0{,}8789}{0{,}8679 \text{ x } 0{,}8789} = 2 \text{ x } \frac{0{,}7627}{1{,}7468} = 2 \text{ x } 0{,}4366 = 0{,}8732\% = 87{,}32\% = 87\%$

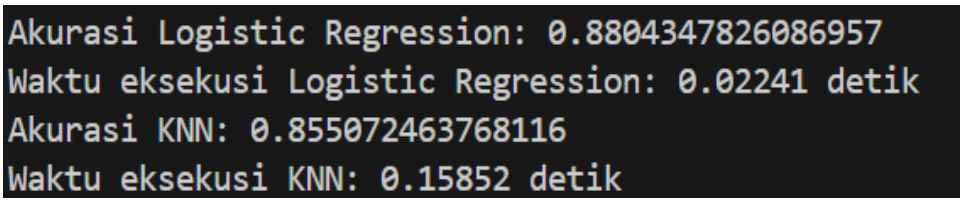
   F1-Score adalah rata-rata harmonis antara *Precision* dan *Recall*, seimbang antara kualitas prediksi dan kelengkapannya.

```
Nilai Presisi, Recall, dan F1 Score Model Logistic Regression:
              precision    recall  f1-score   support

           0       0.88      0.83      0.86       119
           1       0.88      0.92      0.90       157

    accuracy                           0.88       276
   macro avg       0.88      0.87      0.88       276
weighted avg       0.88      0.88      0.88       276

Nilai Presisi, Recall, dan F1 Score Model K-Nearest Neighbor:
              precision    recall  f1-score   support

           0       0.84      0.82      0.83       119
           1       0.87      0.88      0.87       157

    accuracy                           0.86       276
   macro avg       0.85      0.85      0.85       276
weighted avg       0.85      0.86      0.85       276
```

Gambar 4.6 *Output Classification Report*

Dari *classification report* di atas nilai akurasi *Logistic Regression* lebih tinggi ketimbang *K-Nearest Neighbor* (KNN), artinya algoritma

Logistic Regression lebih baik dari pada algoritma *K-Nearest Neighbor* (KNN). Namun dugaan ini masih belum dapat dipastikan karena masih ada langkah selanjutnya.



Gambar 4.7 Akurasi dan Waktu Eksekusi

Pada waktu eksekusi dari masing-masing algoritma, terlihat juga bahwa waktu eksekusi algoritma *Logistic Regression* lebih efisien daripada *algoritma K-Nearest Neighbor*. Selisih waktu antara kedua algoritma tersebut yaitu 0.13611 detik.

Tabel 4.2 Perbandingan Evaluasi Algoritma

Kriteria Evaluasi	Logistic Regression	K-Nearest Neighbor (KNN)
Akurasi	88,04%	85,51%
Precision	88%	87%
Recall	92%	88%
F1-Score	90% Tinggi (lebih baik)	87% Cukup baik
Waktu Eksekusi	0,022 detik	0,158 detik

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil penelitian berjudul *"Perbandingan Algoritma Logistic Regression dan K-Nearest Neighbor (KNN) dalam Dataset Prediksi Gagal Jantung"*, dapat disimpulkan beberapa hal sebagai berikut:

1. Algoritma *Logistic Regression* menunjukkan performa prediksi yang sangat baik dengan tingkat akurasi sebesar 88.04%. *Logistic Regression* juga memiliki keunggulan pada nilai *recall* dan *f1-score*, yang berarti lebih sensitif dalam mendeteksi pasien dengan risiko gagal jantung.
2. Algoritma *K-Nearest Neighbor* (KNN) juga menunjukkan performa tinggi dengan akurasi 85.51%, dan memiliki nilai *precision* yang sedikit lebih tinggi dibandingkan *Logistic Regression*, menunjukkan kekuatan dalam meminimalisir *false positive*.
3. Dari sisi efisiensi waktu, *Logistic Regression* terbukti lebih cepat dengan waktu eksekusi sekitar 0.022 detik, dibandingkan KNN yang memerlukan waktu 0.158 detik. Selisih waktu sebesar 0.136 detik menunjukkan bahwa *Logistic Regression* lebih efisien untuk digunakan pada skala data besar.
4. Secara keseluruhan, berdasarkan hasil evaluasi menggunakan *confusion matrix, accuracy, precision, recall, dan f1-score,* algoritma *Logistic Regression* dinilai lebih unggul dibandingkan KNN dalam konteks prediksi penyakit gagal jantung pada dataset ini. Tetapi algoritma Logistic Regression tidak selalu lebih unggul dari K-Nearest Neighbor, karena hasil prediksi sangat bergantung pada jenis dan pola data. Oleh

karena itu, pemilihan algoritma harus disesuaikan dengan karakteristik studi kasus.

5.2 Saran

Adapun saran yang dapat diberikan untuk pengembangan penelitian di masa mendatang adalah:

1. Pengembangan selanjutnya dapat memanfaatkan data *real-time* atau data dari rumah sakit lokal untuk menguji validitas model dalam kondisi nyata serta meningkatkan generalisasi model ke populasi yang lebih luas.
2. Penelitian selanjutnya dapat membandingkan algoritma yang lebih kompleks untuk melihat apakah terdapat peningkatan signifikan dalam akurasi dan generalisasi model.

DAFTAR PUSTAKA

algoritma. (2022). *Model Machine Learning*. Algoritma. https://algorit.ma/blog/model-machine-learning-2022/

Andi Irfan Daeng Mappa. (2025). *SKRIPSI PERBANDINGAN ALGORITMA MACHINE LEARNING UNTUK MEMPREDIKSI TINGKAT PROMOSI KARYAWAN PADA PT PELABUHAN TANJUNG PRIOK PROGRAM STUDI SISTEM DAN TEKNOLOGI INFORMASI*.

Azzahra, N. D., Ambarwati, A., Desiani, A., Maiyanti, S. I., & Ramayanti, I. (2024). Perbandingan Algoritma K-Nearest Neighbor Dan Logistic Regression Dalam Klasifikasi Penyakit Kanker Serviks. *Energy : Jurnal Ilmiah Ilmu-Ilmu Teknik*, *14*(1), 1–8. https://doi.org/10.51747/energy.v14i1.1843

Cholil, S. R., Handayani, T., Prathivi, R., & Ardianita, T. (2021). Implementasi Algoritma Klasifikasi K-Nearest Neighbor (KNN) Untuk Klasifikasi Seleksi Penerima Beasiswa. *IJCIT (Indonesian Journal on Computer and Information Technology)*, *6*(2), 118–127. https://doi.org/10.31294/ijcit.v6i2.10438

Dicoding Intern. (2023). *Python: Pengertian, Contoh Penggunaan, dan Manfaat Mempelajarinya*. Blog. https://www.dicoding.com/blog/python-pengertian-contoh-penggunaan-dan-manfaat-mempelajarinya/

dr. Pittara. (2022). *Gagal Jantung*. ALODOKTER. https://www.alodokter.com/gagal-jantung

Egor Zyryanov. (2024). *Cara Menggunakan Dataset Kaggle untuk Penelitian: 10 Langkah Penting*. Setronica. https://setronica.com/how-to-use-kaggle-datasets-for-research-a-step-by-step-guide/

Fasya Al Rahmah. (2021). *Asah Kemampuan Lewat Kompetisi Kaggle*. Algoritma. https://algorit.ma/blog/kaggle-project-kompetisi/

Febrian, M. R. (2024). *PREDIKSI PENYAKIT GAGAL JANTUNG MENGGUNAKAN ALGORITMA NAIVE BAYES PREDIKSI PENYAKIT GAGAL JANTUNG MENGGUNAKAN*.

Habibi, H. A. N. S., Nugroho, A., & Firliana, R. (2023). Perbandingan

Algoritma Naïve Bayes Classifier Dan K-Nearest Neighbors Untuk Analisis Sentimen Covid-19 Di Twitter. *Jurnal Ilmiah Informatika*, *11*(01), 54–62. https://doi.org/10.33884/jif.v11i01.7069

Meilinaeka. (2022). *Apa itu Machine Learning? Ketahui Penjelasannya Berikut Ini*. Direktorat Pusat Teknologi Informasi. https://it.telkomuniversity.ac.id/apa-itu-machine-learning/

Mohammad Fahry Sholahuddin, Abdul Holik , Chelvin Suprapto, Iqbal Izha Mahendra, Sadewa Sadewa Wibawanto, M. K. (2023). *Perbandingan Model Logistic Regression dan K-Nearest Neighbors Dalam Prediksi Pembatalan Hotel*. 137–143.

Nadiyah Ramalia. (2024). *Kaggle, Komunitas Belajar Data Science yang Bisa Jadi Sumber Uang*. Glints TapLoker. https://glints.com/id/lowongan/kaggle-adalah/

Neural, R., Rnn, N., & Long, D. A. N. (2024). *PREDIKSI CURAH HUJAN MENGGUNAKAN RECURRENT NEURAL NETWORK (RNN) DAN LONG SHORT-TERM MEMORY (LSTM)*.

Nugraha, W., Syarif, M., Bina, U., Informatika, S., Tree, D., & Forest, R. (2024). *Evaluasi Performa Algoritma Klasifikasi dalam Prediksi Gagal Jantung : Studi Kasus Dataset Heart Failure Prediction*. *23*(4), 897–908.

Setyawan, N. H., & Wakhidah, N. (2025). *Analisis perbandingan metode logistic regression, random forest, gradient boosting untuk prediksi diabetes*. *10*(1), 150–162.

Simanjuntak, T. I., Tanjung, J. P., Tanjung, M. A. P., Utari, C. T., & Muhathir. (2022). JITE (Journal of Informatics and Telecommunication Engineering) Numerical Analysis of Variations Distance Formulas on K Nearest. *JITE (Journal of Informatics and Telecommunication Engineering)*, *6*(July 2021), 325–335.

Tamba, S. P., & -, E. (2022). Prediksi Penyakit Gagal Jantung Dengan Menggunakan Random Forest. *Jurnal Sistem Informasi Dan Ilmu Komputer Prima(JUSIKOM PRIMA)*, *5*(2), 176–181. https://doi.org/10.34012/jurnalsisteminformasidanilmukomputer.v5i2.2445

Team. (2024). *Data Set Adalah: Pengertian, Jenis dan Contohnya*.

Codingstudio.Id. https://codingstudio.id/blog/data-set-adalah/

Tim Konten Kesehatan. (2024). *Apa Penyebab Gagal Jantung? Waspadai 14 Faktornya Ini*. Ciputra Hospital. https://ciputrahospital.com/penyebab-gagal-jantung/

LAMPIRAN CODING

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import time
from matplotlib import pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import MinMaxScaler
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score

# -------- MEMBACA DATA -------- #
df = pd.read_csv('heart_(new_dataset).csv')
print(df.head())

# -------- MENGECEK KEKOSONGAN DATA -------- #
print(df.isna().sum())

# -------- ENCODING FITUR KATEGORI DAN MEMBAGI DATA -------- #
df_encoded = pd.get_dummies(df, columns=['Sex', 'ChestPainType', 'RestingECG', 'ExerciseAngina', 'ST_Slope'])

X = df_encoded.drop('HeartDisease', axis=1)
Y = df_encoded['HeartDisease']
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.3, random_state=5)
```

```python
# -------- NORMALISASI DATA -------- #
scaler = MinMaxScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

print("Data Training Setelah Normalisasi: \n", X_train)
print('\n')
print("Data Testing Seetelah Normalisasi: \n", X_test)

# -------- PEMODELAN LOGISTIC REGRESSION -------- #
start_logreg = time.time()
logreg = LogisticRegression()
logreg.fit(X_train, Y_train)
predictlogreg = logreg.predict(X_test)
end_logreg = time.time()

# -------- PEMODELAN K-NEAREST NEIGHBORS -------- #
start_knn = time.time()
knn_model = KNeighborsClassifier(n_neighbors=5)
knn_model.fit(X_train, Y_train)
knn_pred = knn_model.predict(X_test)
end_knn = time.time()
```

```
# --------VISUALISASI CONF MATRIX LOGISTIC REGRESSION -------- #
class_names=[0,1]
fig, ax = plt.subplots()
tick_marks = np.arange(len(class_names))
plt.xticks(tick_marks, class_names)
plt.yticks(tick_marks, class_names)
sns.heatmap(pd.DataFrame(confusion_matrix(Y_test, predictlogreg)), annot=True, cmap="YlGnBu", fmt='g')
ax.xaxis.set_label_position("top")

plt.title('Confusion matrix Logistic Regression', y=1.1)
plt.ylabel('Actual label')
plt.xlabel('Predicted label')
plt.show()

# --------VISUALISASI CONF MATRIX KNN -------- #
class_names=[0,1]
fig, ax = plt.subplots()
tick_marks = np.arange(len(class_names))
plt.xticks(tick_marks, class_names)
plt.yticks(tick_marks, class_names)
sns.heatmap(pd.DataFrame(confusion_matrix(Y_test, knn_pred)), annot=True, cmap="YlGnBu", fmt='g')
ax.xaxis.set_label_position("top")

plt.title('Confusion matrix K-Nearest Neighbor', y=1.1)
plt.ylabel('Actual label')
plt.xlabel('Predicted label')
plt.show()
```

```
# -------- CLASSIFICATION REPORT -------- #
print("Nilai Presisi, Recall, dan F1 Score Model Logistic Regression: \n", classification_report(Y_test, predictlogr
print("Nilai Presisi, Recall, dan F1 Score Model K-Nearest Neighbor: \n", classification_report(Y_test, knn_pred))

# -------- AKURASI & WAKTU -------- #
print("Akurasi Logistic Regression:", accuracy_score(Y_test, predictlogreg))
print("Waktu eksekusi Logistic Regression: {:.5f} detik".format(end_logreg - start_logreg))

print("Akurasi KNN:", accuracy_score(Y_test, knn_pred))
print("Waktu eksekusi KNN: {:.5f} detik".format(end_knn - start_knn))
```