

**DETEKSI SERANGAN JANTUNG BERDASARKAN DATA REKAM MEDIS
MENGUNAKAN *SUPPORT VECTOR MACHINE***

SKRIPSI

Oleh:
ALIF RADIFAN PIANDY
NIM. 200605110114



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2024**

**DETEKSI SERANGAN JANTUNG BERDASARKAN DATA REKAM
MEDIS MENGGUNAKAN *SUPPORT VECTOR MACHINE***

SKRIPSI

Diajukan kepada:

Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh:

ALIF RADIFAN PIANDY
NIM. 200605110114

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2024**

HALAMAN PERSETUJUAN

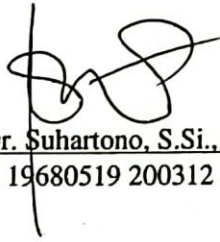
**DETEKSI SERANGAN JANTUNG BERDASARKAN DATA REKAM
MEDIS MENGGUNAKAN *SUPPORT VECTOR MACHINE***

SKRIPSI

Oleh:
ALIF RADIFAN PIANDY
NIM. 200605110114

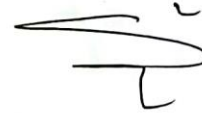
Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 7 Maret 2024

Pembimbing I,



Prof. Dr. Suhartono, S.Si., M.Kom
NIP. 19680519 200312 1 001

Pembimbing II,



Syahiduz Zaman, M.Kom
NIP. 19700502 200501 1 005

Mengetahui,

Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi

Universitas Islam Negeri Maulana Malik Ibrahim Malang




Dr. Fachri Kurniawan, M.MT, IPM
NIP. 19771020 200912 1 001

HALAMAN PENGESAHAN

DETEKSI SERANGAN JANTUNG BERDASARKAN DATA REKAM
MEDIS MENGGUNAKAN *SUPPORT VECTOR MACHINE*

SKRIPSI

Oleh:
ALIF RADIFAN PIANDY
NIM. 200605110114

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 21 Maret 2024

Susunan Dewan Penguji

Ketua Penguji : Fajar Rohman Hariri, M.Kom
NIP. 19890515 201801 1 001

Anggota Penguji I : Shoffin Nahwa Utama, M.T
NIP. 19860703 202012 1 003


Anggota Penguji II : Prof. Dr. Suhartono, S.Si., M.Kom
NIP. 19680519 200312 1001

Anggota Penguji III : Syahiduz Zaman, M.Kom
NIP. 19700502 200501 1 005

()
()
()
()

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang




Dr. Faed Pratiwi Kurniawan, M.MT, IPM
NIP. 19771020 200912 1 001

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Alif Radifan Piandy

NIM : 200605110114

Fakultas / Jurusan : Sains dan Teknologi / Teknik Informatika

Judul Skripsi : Deteksi Serangan Jantung Berdasarkan Data Rekam Medis
Menggunakan *Support Vector Machine*

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 1 April 2024

Yang membuat pernyataan,



Alif Radifan Piandy

NIM. 200605110114

MOTTO

“Whoever goes out to seek knowledge will be in the way of Allah until he returns”

“Hiduplah seolah engkau mati besok. Belajarlah seolah engkau hidup selamanya”

HALAMAN PERSEMBAHAN

Saya persembahkan karya ini kepada:

Ayah saya,

Ir. Sandy Adiwidjaja

Yang telah mendukung dan menyemangati saya hingga sampai titik ini

Bunda saya,

Poppy Yolanda, S.T

Yang telah mendukung dan menyemangati saya hingga sampai titik ini

Saudara kembar saya,

Alif Rafidan Piandy

Yang telah mendukung dan menyemangati saya hingga sampai titik ini

Sahabat-sahabat saya,

Achmad Fahreza Alif Pahlevi, Abdurrozzaq Ashshiddiqi Zuhri, Muhammad
Haris Ibrahim, Aqilla Darin Makkyah, Wikke Alvina Putri, Renisa Maulidifa, dan
Hikmatul Maulidia Putri

Yang telah memberikan kebahagiaan selama perkuliahan ini

Teman-teman seperjuangan,

Teknik Informatika Angkatan 2020

Semoga kita semua selalu diberi kemudahan oleh Allah SWT

KATA PENGANTAR

Assalamualaikum Warahmatullahi Wabarakatuh.

Segala puji hanya milik Allah Subhanahu Wa Ta'ala atas segala nikmat dan kasih sayang-Nya yang telah memudahkan penulis untuk menyelesaikan skripsi yang berjudul “Deteksi Serangan Jantung Berdasarkan Data Rekam Medis Menggunakan *Support Vector Machine*”. Semoga shalawat dan salam senantiasa terlimpah kepada Nabi Muhammad Sallallahu ‘Alaihi wa Sallam. Dan semoga kita semua mendapat syafaatnya di hari kiamat nanti, Aamiin.

Penulis mengucapkan rasa syukur dan terima kasih yang tak terhingga kepada semua pihak-pihak yang selalu memberikan bantuan dan motivasi kepada penulis untuk dapat menyelesaikan skripsi ini. Ucapan ini penulis sampaikan kepada:

1. Prof. Dr. H. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Prof. Dr. Hj. Sri Hariani, M.Si., selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Fachrul Kurniawan, M.MT., IPM, selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Prof. Dr. Suhartono, S.Si., M.Kom selaku dosen pembimbing I dan Syahiduz Zaman, M.Kom selaku dosen pembimbing II yang telah memberikan bantuan dan arahan kepada penulis, sehingga bisa menuntaskan skripsi ini.

5. Fajar Rohman Hariri, M.Kom selaku dosen penguji I dan Shoffin Nahwa Utama, M.T selaku dosen penguji II yang telah menguji serta memberikan masukan sehingga penulis dapat menuntaskan skripsi dengan baik.
6. Segenap Dosen, Admin, Laboran dan Mahasiswa Program Studi Teknik Informatika yang telah memberikan banyak dukungan dan bimbingan selama pengerjaan skripsi ini.
7. Bunda, Ayah, serta saudara kembar saya yang selalu memberikan dukungan dan motivasi untuk terus berusaha, dan doa yang tak putus-putusnya selalu disampaikan agar dapat menuntaskan skripsi ini dengan lancar dan baik.

Akhir kata, penulis mengakui bahwa penulisan pada skripsi ini masih banyak kekurangan. Saya berharap semoga skripsi ini diterima sebagai amal ibadah yang tulus dan bermanfaat di sisi Allah Subhanahu Wa Ta'ala. Semoga karya ini menjadi bagian dari kontribusi yang tak terputus dalam rangka memperkuat dan mengembangkan ilmu pengetahuan, serta melaksanakan tugas sebagai hamba Allah yang berkomitmen.

Wassalamualaikum Warahmatullahi Wabarakatuh.

Malang, 21 Maret 2024

Penulis

DAFTAR ISI

HALAMAN PENGAJUAN	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
PERNYATAAN KEASLIAN TULISAN	v
MOTTO	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiii
ABSTRAK	xiv
ABSTRACT	xv
مستخلص البحث	xvi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Pernyataan Masalah	5
1.3 Batasan Masalah	5
1.4 Tujuan Penelitian	5
1.5 Manfaat Penelitian	6
BAB II KAJIAN PUSTAKA	7
2.1 Penelitian Terkait	7
2.2 Serangan Jantung	13
2.3 Data Rekam Medis	14
2.4 <i>Support Vector Machine</i>	14
BAB III METODE PENELITIAN	16
3.1 Desain Sistem	16
3.2 Pengumpulan Data	17
3.3 <i>Preprocessing Data</i>	19
3.3.1 <i>Label Encoding</i>	21
3.3.2 <i>Feature Scaling</i>	22
3.4 <i>Split Data</i>	25
3.5 <i>Support Vector Machine</i>	27
3.6 Evaluasi Performa	30
3.7 Skenario Pengujian	34
BAB IV UJI COBA DAN PEMBAHASAN	39
4.1 Langkah-Langkah Uji Coba	39
4.2 Hasil Uji Coba	40
4.2.1 Hasil Uji Coba Skenario A	41
4.2.2 Hasil Uji Coba Skenario B	43

4.2.3 Hasil Uji Coba Skenario C	46
4.2.4 Hasil Uji Coba Skenario D	48
4.3 Pembahasan	50
BAB V KESIMPULAN DAN SARAN.....	62
5.1 Kesimpulan.....	62
5.2 Saran	63
DAFTAR PUSTAKA	

DAFTAR TABEL

Tabel 2.1 Penelitian Terkait	10
Tabel 3.1 Atribut <i>Dataset</i>	18
Tabel 3.2 Contoh <i>Dataset</i>	219
Tabel 3.3 Contoh Data Sebelum Menjalani Proses <i>Label Encoding</i>	21
Tabel 3.4 Contoh Data Setelah Menjalani Proses <i>Label Encoding</i>	22
Tabel 3.5 Contoh Data Sebelum Proses <i>Feature Scaling</i>	24
Tabel 3.6 Contoh Data Setelah Proses <i>Feature Scaling</i>	25
Tabel 3.7 Skenario Pembagian <i>Dataset</i>	34
Tabel 3.8 Sub-skenario Parameter Model SVM Kernel Linear	37
Tabel 3.9 Sub-skenario Parameter Model SVM Kernel RBF.....	37
Tabel 3.10 Sub-skenario Parameter Model SVM Kernel Polynomial	37
Tabel 4.1 Performa Sistem Skenario A.....	40
Tabel 4.2 Performa Sistem Skenario B	42
Tabel 4.3 Performa Sistem Skenario C	45
Tabel 4.4 Performa Sistem Skenario D.....	47

DAFTAR GAMBAR

Gambar 2.1 Optimal <i>Hyperplane</i> pada <i>Support Vector Machine</i>	15
Gambar 3.1 Desain Sistem	16
Gambar 4.1 Grafik Perbandingan Nilai Akurasi Setiap Skenario.....	53
Gambar 4.2 Grafik Perbandingan Nilai Presisi Setiap Skenario.....	54
Gambar 4.3 Grafik Perbandingan Nilai <i>Recall</i> Setiap Skenario.....	54
Gambar 4.4 Grafik Perbandingan Nilai <i>F1-score</i> Setiap Skenario.....	55

ABSTRAK

Piandy, Alif Radifan. 2024. **Deteksi Serangan Jantung Berdasarkan Data Rekam Medis Menggunakan Support Vector Machine**. Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Prof. Dr. Suhartono, S.Si., M.Kom. (II) Syahiduz Zaman, M.Kom.

Kata kunci: *Serangan Jantung, Support Vector Machine, Deteksi*

Serangan jantung merupakan kondisi serius yang dapat menyebabkan kematian jika tidak ditangani dengan cepat. Penyebab utama serangan jantung antara lain penyumbatan arteri koroner, tekanan darah tinggi, diabetes, merokok, dan obesitas. Banyak pasien yang mengalami serangan jantung terlambat mendapatkan penanganan medis karena tidak mengenali gejala awal, seperti nyeri dada, sesak napas, atau kelelahan berlebihan. Untuk mengatasi permasalahan tersebut, pada penelitian ini akan dirancang sistem deteksi serangan jantung berdasarkan data rekam medis pasien menggunakan metode *Support Vector Machine*, dengan tujuan untuk mengetahui performa metode *Support Vector Machine* dalam mendeteksi serangan jantung berdasarkan data rekam medis. Data yang digunakan diperoleh dari situs *Elsevier Mendeley Data Repository* yang berjudul *Heart Attack Dataset*, yang berisikan data rekam medis pasien sebanyak 1319 sampel dengan 9 atribut. Penelitian ini melibatkan beberapa langkah penting. Langkah awal adalah pra-pemrosesan data, yang mencakup *Label Encoding* menggunakan *LabelEncoder* dan *Feature Scaling* menggunakan *StandardScaler*. Kemudian, data dibagi menjadi empat bagian dengan rasio yang berbeda, yaitu 85% data latih : 15% data uji, 80% data latih : 20% data uji, 75% data latih : 25% data uji, dan 70% data latih : 30% data uji. Selanjutnya, penelitian ini membandingkan tiga kernel *Support Vector Machine* dan lima nilai *hyperparameter C* yang berbeda. Untuk mengevaluasi performa metode dalam melakukan deteksi, pengukuran dilakukan dengan memanfaatkan *confusion matrix* untuk mendapatkan nilai akurasi, presisi, *recall*, dan *f1-score*. Hasil analisis menunjukkan bahwa pada rasio 70:30 dengan kernel linear dan nilai *hyperparameter C*=100, diperoleh performa terbaik dengan nilai akurasi 93%, presisi 92%, *recall* 93%, dan *f1-score* 93%.

ABSTRACT

Piandy, Alif Radifan. 2024. **Heart Attack Detection Based on Medical Record Data Using Support Vector Machine.** Undergraduate Thesis. Informatics Engineering Study Program, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Advisors: (I) Prof. Dr. Suhartono, S.Si., M.Kom. (II) Syahiduz Zaman, M.Kom.

Keywords: *Heart Attack, Support Vector Machine, Detection*

Heart attack is a serious condition that can cause death if not treated quickly. The main causes of heart attacks include coronary artery blockage, high blood pressure, diabetes, smoking, and obesity. Many patients who experience a heart attack are late in getting medical treatment because they do not recognize the early symptoms, such as chest pain, shortness of breath, or excessive fatigue. To address this issue, this research will design a heart attack detection system based on patient medical record data using the Support Vector Machine method, with the aim of determining the performance of the Support Vector Machine method in detecting heart attacks based on medical record data. The data used was obtained from the Elsevier Mendeley Data Repository site titled Heart Attack Dataset, which contains medical record data of 1319 samples with 9 attributes. This research involves several important steps. The initial step is data preprocessing, which includes Label Encoding using LabelEncoder and Feature Scaling using StandardScaler. Then, the data is divided into four parts with different ratios, namely 85% training data: 15% test data, 80% training data: 20% test data, 75% training data: 25% test data, and 70% training data: 30% test data. Furthermore, this research compares three Support Vector Machine kernels and five different hyperparameter C values. To evaluate the performance of the method in performing detection, measurements are carried out by utilizing a confusion matrix to obtain accuracy, precision, recall, and f1-score values. The analysis results show that at a 70:30 ratio with a linear kernel and hyperparameter C=100, the best performance is obtained with an accuracy of 93%, precision of 92%, recall of 93%, and f1-score of 93%.

مستخلص البحث

فياندي، أليف راديفان. ٢٠٢٤. الكشف عن النوبة القلبية بناء على بيانات السجلات الطبية باستخدام آلة المتجه الداعم. البحث الجامعي. قسم الهندسة المعلوماتية، كلية العلوم والتكنولوجيا بجامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف الأول: أ. د. سوهارتونو، الماجستير. المشرف الثاني: شاهد الزمان، الماجستير. الكلمات الرئيسية. كشف، نوبة قلبية، آلة المتجه الداعم

النوبة القلبية هي حالة خطيرة يمكن أن تسبب الوفاة إذا لم يتم علاجها بسرعة. تشمل الأسباب الرئيسية للنوبات القلبية انسداد الشريان التاجي وارتفاع ضغط الدم والسكري والتدخين والسمنة. يحصل العديد من المرضى الذين يعانون من نوبة قلبية على رعاية طبية بعد فوات الأوان لأنهم لا يتعرفون على الأعراض المبكرة، مثل ألم الصدر أو ضيق التنفس أو التعب المفرط. للتغلب على هذه المشاكل، سيقوم هذا البحث بتصميم نظام للكشف عن النوبات القلبية بناء على بيانات السجلات الطبية للمريض باستخدام طريقة آلة المتجه الداعم للأمراض، بهدف معرفة أداء تلك الطريقة في الكشف عنها خلال البيانات الطبية. تم الحصول على البيانات المستخدمة من موقع مستودع بيانات إلزفير ومندلي بمجموعة بيانات عن النوبة القلبية، والتي تحتوي على بيانات السجلات الطبية للمرضى ما يصل إلى ١٣١٩ عينة مع ٩ سمات. تضمن هذا البحث عدة خطوات مهمة. الخطوة الأولى هي المعالجة المسبقة للبيانات، والتي شملت ترميز المصنقات باستخدام *LabelEncoder* وقياس الميزات باستخدام *StandardScaler*. بعد ذلك، تم تقسيم البيانات إلى أربعة أجزاء بنسب مختلفة، وهي ٨٥٪ بيانات التدريب: ١٥٪ بيانات الاختبار، ٨٠٪ بيانات التدريب: ٢٠٪ بيانات الاختبار، ٧٥٪ بيانات التدريب: ٢٥٪ بيانات الاختبار، و ٧٠٪ بيانات التدريب: ٣٠٪ بيانات الاختبار. بعد ذلك، قارن البحث ثلاث حبات لآلة المتجه الداعم وخمس قيم مختلفة للمعلمات الفائقة ج. لتقييم أداء الطريقة في إجراء الكشف، تم إجراء القياسات باستخدام مصفوفة الارتباك للحصول على والضبط والدقة والاستدعاء ودرجة ف١. أظهرت نتائج التحليل أنه بنسبة ٧٠:٣٠ مع نواة خطية وقيمة معلمة فائقة ج. = ١٠٠، تم الحصول على أفضل أداء بقيمة ضبط ٩٣٪، دقة ٩٢٪، استدعاء ٩٣٪، ودرجة ف١ ٩٣٪.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Penyakit jantung atau kardiovaskular menjadi persoalan kesehatan penting secara global, dan merupakan kondisi medis kronis yang tidak menular antar individu (Rahayu et al., 2021). Data *Global Disease Burden* (GDB) dan *Institute for Health Metrics and Evaluation* (IHME) tahun 2014 hingga 2019 menunjukkan bahwa penyakit kardiovaskular menjadi penyebab kematian terbanyak di Indonesia. Hasil survei Riset Kesehatan Dasar (Riskesdas) juga mengindikasikan peningkatan prevalensi penyakit jantung di Indonesia. Tahun 2013 menunjukkan prevalensi penyakit jantung tercatat 0,5%, yang kemudian meningkat menjadi 1,5% pada tahun 2018. Penyakit jantung juga telah memberikan tekanan ekonomi terbesar pada sistem layanan kesehatan di Indonesia, dengan pengeluaran anggaran kesehatan terbesar untuk penyakit jantung sebesar 7,7 triliun rupiah pada tahun 2021 (Rokom, 2022). Menurut data tahun 2021 dari *World Health Organization*, Setiap tahunnya, 17,9 juta orang kehilangan nyawa akibat penyakit jantung. Serangan jantung hanyalah satu jenis dari sekian banyak penyakit jantung yang ada. Serangan jantung adalah kondisi medis yang serius ketika terhambatnya suplai darah ke jantung (Gaye, 2021). Tanpa penanganan yang tepat, kondisi ini dapat menyebabkan kerusakan parah pada otot jantung dan berpotensi mengakibatkan kematian.

Individu yang gaya hidup serta pola makannya tidak sehat seringkali berisiko mengalami serangan jantung. Merokok merupakan salah satu faktor risiko utama serangan jantung (Alpert, 2023). Selain kebiasaan merokok, konsumsi makanan tinggi lemak jenuh dan karbohidrat olahan juga dapat memicu terjadinya serangan jantung (Abu Bakar et al., 2022). Beberapa makanan dengan kadar kolesterol dalam jumlah tinggi seperti gorengan, daging sapi olahan, dan kuning telur akan menyebabkan peningkatan kadar kolesterol dalam darah, sehingga berisiko menumpuk di dinding pembuluh darah dan menyumbat arteri (Muhammad Fauzan A et al., 2022). Arteri yang tersumbat tersebut dapat menyebabkan serangan jantung. Makanan memiliki peranan yang penting pada kesehatan jantung. Oleh sebab itu, manusia perlu untuk memperhatikan apa yang akan dimakan. Dalam Surat Yunus ayat 57, Allah berfirman:

يَا أَيُّهَا النَّاسُ قَدْ جَاءَتْكُمْ مَوْعِظَةٌ مِّن رَّبِّكُمْ وَشِفَاءٌ لِّمَا فِي الصُّدُورِ وَهُدًى وَرَحْمَةٌ لِّلْمُؤْمِنِينَ

“Hai manusia, sesungguhnya telah datang kepadamu pelajaran dari Tuhanmu dan penyembuh bagi penyakit-penyakit (yang berada) dalam dada dan petunjuk serta rahmat bagi orang-orang yang beriman.” (QS. Yunus: 57)

QS. Yunus: 57 menyatakan bahwa Allah SWT telah memberikan pelajaran, penyembuh, petunjuk, dan rahmat bagi manusia, khususnya orang-orang yang beriman. Pelajaran ini mencakup ilmu pengetahuan dan teknologi yang dapat digunakan untuk kemaslahatan umat. Penyembuh ini mencakup obat-obatan dan metode pengobatan yang dapat menyembuhkan penyakit fisik dan psikis. Petunjuk ini mencakup ajaran-ajaran agama dan moral yang dapat membimbing manusia ke jalan yang benar. Rahmat ini mencakup karunia-karunia Allah SWT yang tidak

terhingga dan tidak terbatas. Penelitian yang berjudul “Deteksi Serangan Jantung Berdasarkan Data Rekam Medis Menggunakan *Support Vector Machine*” adalah salah satu contoh penerapan pelajaran dan penyembuh yang telah diberikan Allah SWT kepada manusia.

Serangan jantung adalah kondisi medis yang fatal dan menyebabkan kematian (Aloyses et al., 2022). Banyak orang yang mengalami serangan jantung seringkali tidak menyadari gejala yang mereka alami atau bahkan tidak mengalami gejala sama sekali, yang mengakibatkan mereka tidak mendapatkan pertolongan medis yang diperlukan (Ahmed et al., 2020). Serangan jantung juga tidak dapat dideteksi secara langsung tanpa pemeriksaan jantung yang mendalam, seperti angiografi. Angiografi adalah salah satu teknik diagnostik andalan yang kerap diterapkan dokter untuk mendiagnosis serangan jantung (Alfaidi et al., 2022). Namun, proses diagnosis ini memerlukan analisis banyak faktor dan dianggap sebagai prosedur yang mahal. Hal ini menjadi persoalan di negara-negara berkembang yang mengalami keterbatasan sarana diagnostik yang diperlukan, jumlah dokter spesialis jantung, dan sumber daya pendukung lainnya.

Berdasarkan permasalahan yang disebutkan di atas, penting untuk diketahui bahwa mengabaikan faktor risiko serangan jantung dapat mengakibatkan peningkatan jumlah orang yang menderita serangan jantung. Maka, sangat penting menggunakan pendekatan yang dapat mengidentifikasi status pasien apakah mereka berisiko mengalami serangan jantung atau tidak. Salah satu metode atau pendekatan yang dapat diterapkan yaitu *Support Vector Machine*. SVM adalah metode klasifikasi yang memanfaatkan pemisahan optimal antara kategori atau

kelas data dalam sebuah *dataset*. SVM mencari *hyperplane* terbaik yang dapat memisahkan data ke dalam kelas-kelas yang berbeda. SVM juga merupakan metode yang efisien dan telah sukses diaplikasikan dalam berbagai bidang, terutama klasifikasi (Piccialli & Sciandrone, 2022).

Pada penelitian yang sebelumnya dilakukan oleh Musthofa Galih Pradana dkk pada tahun 2022, dilakukan perbandingan algoritma pembelajaran mesin dalam mengklasifikasi peluang penyakit serangan jantung. *UCI Machine Learning Repository* merupakan asal dari *dataset* yang digunakan. Dalam penelitian tersebut dihasilkan bahwa metode *Naïve Bayes* memiliki tingkat akurasi yang lebih rendah dibandingkan dengan metode *Support Vector Machine* dalam klasifikasi peluang penyakit serangan jantung. Metode SVM mampu untuk mencapai akurasi sebesar 87%, presisi sebesar 92% dan *recall* sebesar 87%. Hal ini menunjukkan bahwa SVM merupakan metode yang cocok dan lebih akurat dalam melakukan klasifikasi data (Pradana et al., 2022).

Melakukan uji akurasi pada hasil klasifikasi membantu untuk mengidentifikasi kesalahan dalam klasifikasi, sehingga dapat ditentukan persentase klasifikasi yang akurat (Bashit et al., 2019). Pengujian akurasi dilakukan untuk menilai sejauh mana hasil klasifikasi sesuai dengan data sebenarnya. Dalam dunia kesehatan, persentase akurasi sangat penting sebagai tolak ukur, karena dapat menentukan tingkat keandalan hasil klasifikasi yang harus sesuai dengan persyaratan dan standar klinis.

Dalam penelitian ini, metode SVM digunakan untuk mengklasifikasikan pasien penderita serangan jantung berdasarkan data rekam medis. Akurasi pada

hasil klasifikasi kemudian dievaluasi menggunakan tabel *confusion matrix* untuk menilai seberapa baik hasil klasifikasi selaras dengan diagnosis pasien yang sebenarnya. Diharapkan hasil yang diperoleh bermanfaat bagi dunia kesehatan dan mampu untuk mendiagnosis lebih awal pasien yang rentan terkena serangan jantung.

1.2 Pernyataan Masalah

Bagaimana performa metode *Support Vector Machine* dalam mendeteksi serangan jantung berdasarkan data rekam medis?

1.3 Batasan Masalah

Batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Data yang digunakan hanya berupa data rekam medis pasien, tidak melibatkan data diagnostik lainnya seperti hasil angiografi, *electrocardiogram*, foto *rontgen* dada, dan lainnya.
2. Tidak mempertimbangkan faktor-faktor di luar tubuh yang dapat memengaruhi serangan jantung seperti riwayat keluarga, pola makan, gaya hidup, dan lainnya.
3. Hanya fokus pada deteksi serangan jantung, tanpa mempertimbangkan penyakit jantung lainnya.

1.4 Tujuan Penelitian

Mengetahui performa metode *Support Vector Machine* dalam mendeteksi serangan jantung berdasarkan data rekam medis.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah sebagai berikut:

1. Membantu para peneliti atau akademisi lain yang ingin mengembangkan lebih lanjut penelitian ini.
2. Membantu tenaga medis dalam melakukan diagnosis dini serangan jantung pada pasien yang memiliki faktor risiko serangan jantung.

BAB II

KAJIAN PUSTAKA

2.1 Penelitian Terkait

Shanbhag et al., (2021) melakukan penelitian mengenai analisis probabilitas serangan jantung menggunakan pembelajaran mesin. Penelitian ini bertujuan untuk menganalisis probabilitas terjadinya serangan jantung dengan menggunakan algoritma pembelajaran mesin. Pada penelitian ini juga dijelaskan berbagai algoritma atau metode pembelajaran mesin yang digunakan untuk menganalisis probabilitas serangan jantung. Dalam penelitian ini, digunakan *dataset* yang berasal dari situs Kaggle yang berjudul *Heart Disease Dataset*. *Dataset* tersebut berisi 14 atribut dan 303 data pasien. Atribut-atribut yang tercantum berguna untuk menilai kemungkinan risiko terjadinya serangan jantung. Hasil penelitian menunjukkan bahwa dari beberapa algoritma pembelajaran mesin yang digunakan, *Support Vector Machine* menjadi yang paling efektif dalam menganalisis probabilitas serangan jantung dengan akurasi sebesar 85.7%.

Chowdhury et al., (2021) melakukan penelitian mengenai prognosis penyakit jantung menggunakan teknik klasifikasi pembelajaran mesin. Penelitian ini bertujuan untuk mengembangkan suatu sistem berbasis pembelajaran mesin dalam prognosis penyakit jantung dengan akurasi tinggi, serta dapat mengurangi keterlibatan tenaga medis dan tes medis tambahan. Penelitian ini juga bertujuan untuk meningkatkan akurasi prognosis penyakit jantung dengan mengidentifikasi fitur-fitur penting melalui penerapan berbagai algoritma pembelajaran mesin. Dalam penelitian ini digunakan *dataset* yang berasal dari berbagai rumah sakit dan

layanan kesehatan di wilayah Sylhet, Bangladesh dan terdiri dari 564 data pasien dan 18 atribut. Dalam penelitian ini, ada 5 algoritma pembelajaran mesin seperti *Decision Tree*, *Logistic Regression*, *K-Nearest Neighbors (K-NN)*, *Naïve Bayes*, dan *Support Vector Machine (SVM)*. Di antara algoritma pembelajaran mesin tersebut, *Support Vector Machine* bisa memberikan kinerja terbaik dengan tingkat akurasi sebesar 91%.

Garg et al., (2021) melakukan penelitian mengenai prediksi penyakit jantung menggunakan teknik pembelajaran mesin. Tujuan utama dari penelitian ini adalah mengeksplorasi beberapa teknik pembelajaran mesin dan membandingkan performanya. *Dataset* yang digunakan diperoleh dari Kaggle dengan judul *Heart Disease UCI* dan terdiri dari 303 data dengan 13 atribut yang berbeda. Setiap data mewakili seorang pasien dan atribut-atribut yang digunakan untuk mengidentifikasi pasien menderita penyakit jantung atau tidak. *Dataset* ini digunakan untuk melatih dan menguji berbagai teknik pembelajaran mesin yang digunakan untuk memprediksi penyakit jantung. Hasil penelitian menunjukkan bahwa algoritma K-NN memiliki performa terbaik dalam memprediksi penyakit jantung dengan akurasi sebesar 86.885%. Selain itu, peneliti juga menunjukkan bahwa teknik pembelajaran mesin dapat membantu dalam memprediksi penyakit jantung dan dapat digunakan sebagai alat untuk deteksi dini dan pencegahan penyakit jantung.

Penelitian yang dilakukan Furqan et al., (2020) bertujuan meningkatkan akurasi dan efisiensi prediksi penyakit jantung dengan menggunakan beberapa algoritma pembelajaran mesin seperti K-NN, *Logistic Regression*, dan *Random Forest*. *UCI dataset of heart disease* merupakan *dataset* publik yang digunakan.

Dataset ini berisi informasi klinis tentang pasien yang terkait dengan penyakit jantung. *Dataset* ini digunakan untuk menguji beberapa algoritma pembelajaran mesin seperti K-NN, *Logistic Regression*, dan *Random Forest* untuk memprediksi penyakit jantung pada pasien. Hasil penelitian menunjukkan bahwa algoritma *Logistic Regression* memiliki performa terbaik dalam memprediksi penyakit jantung dengan akurasi sebesar 85%. Selain itu, peneliti juga menunjukkan bahwa teknik pembelajaran mesin dapat membantu dalam memprediksi penyakit jantung dan dapat digunakan sebagai alat untuk deteksi dini dan pencegahan penyakit jantung.

Sajja et al., (2021) melakukan penelitian mengenai penerapan pembelajaran mesin dalam klasifikasi dan prediksi penyakit jantung. Mengidentifikasi faktor risiko yang berkontribusi terhadap penyakit jantung dan membangun model pembelajaran mesin yang dapat memprediksi kemungkinan seseorang menderita penyakit jantung merupakan tujuan penelitian ini. *Cleveland Heart Disease* yang diperoleh dari *UCI Machinery Heart Disease Data Collection* merupakan *dataset* yang digunakan. *Dataset* ini kemudian digunakan untuk melatih model pembelajaran mesin yang dapat memprediksi kemungkinan seseorang menderita penyakit jantung. Hasil dari penelitian tersebut menunjukkan bahwa teknik pembelajaran mesin, khususnya *Support Vector Machine*, dapat digunakan untuk memprediksi risiko penyakit jantung dengan akurasi lebih dari 90%. Dengan akurasi yang tinggi, para dokter dan tenaga medis dapat melakukan prediksi penyakit jantung dengan tepat.

Tabel 2.1 Penelitian Terkait

No.	Peneliti (tahun)	Metode	Hasil	Perbedaan
1	Shanbhag dkk. (2021)	K-NN, <i>Decision Tree</i> , <i>Random Forest</i> , <i>Logistic Regression</i> , <i>Support Vector Machine</i>	<p>SVM memperoleh akurasi sebesar 85.7%.</p> <p><i>Decision Tree</i> memperoleh akurasi sebesar 70.33%.</p> <p>K-NN memperoleh akurasi sebesar 84.2%.</p> <p><i>Random Forest</i> memperoleh akurasi sebesar 79%.</p> <p><i>Linear Regression</i> memperoleh akurasi sebesar 85%.</p>	<p>Studi kasus nya sama, namun data yang digunakan berbeda, yaitu data terkait serangan jantung yang berasal dari situs web Kaggle dengan judul <i>Heart Disease Dataset</i> (Shanbhag et al., 2021).</p> <p>Belum menerapkan <i>feature scaling</i> menggunakan <i>StandardScaler</i> untuk menormalisasi skala nilai dari setiap fitur dalam <i>dataset</i> pada tahap <i>preprocessing data</i>.</p> <p>Belum menerapkan <i>tuning hyperparameter C</i> untuk menentukan seberapa toleran model SVM terhadap kesalahan klasifikasi.</p>
2	Chowdhury dkk. (2021)	<i>Decision Tree</i> , <i>Logistic Regression</i> , K-NN, <i>Naive Bayes</i> , <i>Support Vector Machine</i>	<p><i>Decision Tree</i> memperoleh akurasi sebesar 87.546%.</p> <p><i>Naive Bayes</i> memperoleh akurasi sebesar 84%.</p> <p><i>Logistic Regression</i> memperoleh akurasi sebesar 86%.</p> <p>SVM memperoleh akurasi sebesar 91%.</p>	<p>Studi kasus nya sama, namun data yang digunakan berbeda, yaitu data terkait penyakit jantung yang berasal dari berbagai rumah sakit dan layanan kesehatan di wilayah Sylhet, Bangladesh (Chowdhury et al., 2021).</p> <p>Belum menerapkan <i>feature scaling</i> menggunakan <i>StandardScaler</i> untuk menormalisasi skala nilai dari setiap fitur dalam <i>dataset</i> pada</p>

No.	Peneliti (tahun)	Metode	Hasil	Perbedaan
2			K-NN memperoleh akurasi sebesar 87%.	tahap <i>preprocessing data</i> . Belum menerapkan <i>tuning hyperparameter C</i> untuk menentukan seberapa toleran model SVM terhadap kesalahan klasifikasi.
3	Garg dkk. (2021)	K-NN, <i>Random Forest</i>	K-NN memperoleh akurasi sebesar 86.885%. <i>Random Forest</i> memperoleh akurasi sebesar 81.967%.	Studi kasus nya sama, namun data yang digunakan berbeda, yaitu data terkait penyakit jantung yang berasal dari situs web Kaggle dengan judul <i>Heart Disease UCI</i> (Garg et al., 2021). Belum menerapkan <i>feature scaling</i> menggunakan <i>StandardScaler</i> untuk menormalisasi skala nilai dari setiap fitur dalam <i>dataset</i> pada tahap <i>preprocessing data</i> . Belum menerapkan <i>tuning hyperparameter C</i> untuk menentukan seberapa toleran model SVM terhadap kesalahan klasifikasi.
4	Furqan dkk. (2020)	<i>Logistic Regression</i> , K-NN, <i>Random Forest</i>	<i>Logistic Regression</i> memperoleh akurasi sebesar 87.88%. K-NN memperoleh akurasi sebesar 74.19%. <i>Random Forest</i> memperoleh	Studi kasus nya sama, namun data yang digunakan berbeda, yaitu data terkait penyakit jantung yang berasal dari situs web Kaggle dengan judul <i>UCI dataset of heart disease</i> (Furqan et al., 2020). Belum menerapkan <i>feature scaling</i>

No.	Peneliti (tahun)	Metode	Hasil	Perbedaan
4			akurasi sebesar 86.81%.	menggunakan StandardScaler untuk menormalisasi skala nilai dari setiap fitur dalam <i>dataset</i> pada tahap <i>preprocessing data</i> . Belum menerapkan <i>tuning hyperparameter C</i> untuk menentukan seberapa toleran model SVM terhadap kesalahan klasifikasi.
5	Sajja dkk. (2021)	C4.5, ID3, <i>Random Forest</i> , K-NN, SVM	C4.5 memperoleh akurasi sebesar 79%. ID3 memperoleh akurasi sebesar 71%. <i>Random Forest</i> memperoleh akurasi sebesar 81%. K-NN memperoleh akurasi sebesar 82%. SVM memperoleh akurasi sebesar 92%.	Studi kasus nya sama, namun data yang digunakan berbeda, yaitu data terkait penyakit jantung yang berasal dari situs web Kaggle dengan judul <i>Cleveland Heart Disease</i> (Sajja et al., 2021). Belum menerapkan <i>feature scaling</i> menggunakan StandardScaler untuk menormalisasi skala nilai dari setiap fitur dalam <i>dataset</i> pada tahap <i>preprocessing data</i> . Belum menerapkan <i>tuning hyperparameter C</i> untuk menentukan seberapa toleran model SVM terhadap kesalahan klasifikasi.

Pada penelitian ini, akan dilakukan klasifikasi pasien penderita serangan jantung berdasarkan data rekam medis. *Support Vector Machine* merupakan metode klasifikasi yang akan digunakan, dan hasil dari klasifikasi ini akan dievaluasi

menggunakan tabel *confusion matrix* untuk mengukur sejauh mana performa sistem klasifikasi dapat diandalkan.

Terdapat tiga aspek baru dalam penelitian ini yang belum pernah dilakukan sebelumnya. Pertama, penggunaan *dataset* berisi rekam medis pasien yang menderita serangan jantung dari *Elsevier Mendeley Data Repository: Heart Attack Dataset*. Kedua, *tuning hyperparameter C* untuk mengoptimalkan SVM guna menghindari kesalahan klasifikasi pada setiap sampel dalam *dataset* pelatihan. Terakhir, pengaplikasian *StandardScaler* untuk melakukan normalisasi data atau *feature scaling*, sehingga semua data numerik dalam *dataset* memiliki rentang nilai yang serupa. Hal ini membuat proses pengolahan *dataset* dengan metode SVM menjadi lebih efisien.

2.2 Serangan Jantung

Penyumbatan arteri koroner terjadi saat suplai darah ke otot jantung terhambat, yang menyebabkan serangan jantung. Nama ilmiah dari serangan jantung adalah *myocardial infarction*. Penyebab utamanya biasanya aterosklerosis atau penumpukan plak di dalam arteri (Gaye, 2021). Rasa nyeri pada dada merupakan salah satu tanda seseorang mungkin terkena serangan jantung, meskipun tidak semua orang mengalami gejala tersebut (Breuckmann & Rassaf, 2020). Faktor risiko serangan jantung antara lain riwayat penyakit jantung dalam keluarga, kebiasaan merokok, tekanan darah tinggi, diabetes, kadar kolesterol tinggi, obesitas, kurang aktivitas fisik, dan pola makan yang tidak sehat (Tamlander et al., 2022). Deteksi dan pengobatan dini pada serangan jantung sangatlah penting, seperti angioplasti untuk membuka kembali arteri yang tersumbat (Rohayati &

Widani, 2022). Upaya pencegahan serangan jantung, seperti perubahan gaya hidup yang sehat dan menjalani pengobatan, sangatlah penting untuk mengendalikan faktor risiko dan mencegah serangan jantung di masa depan.

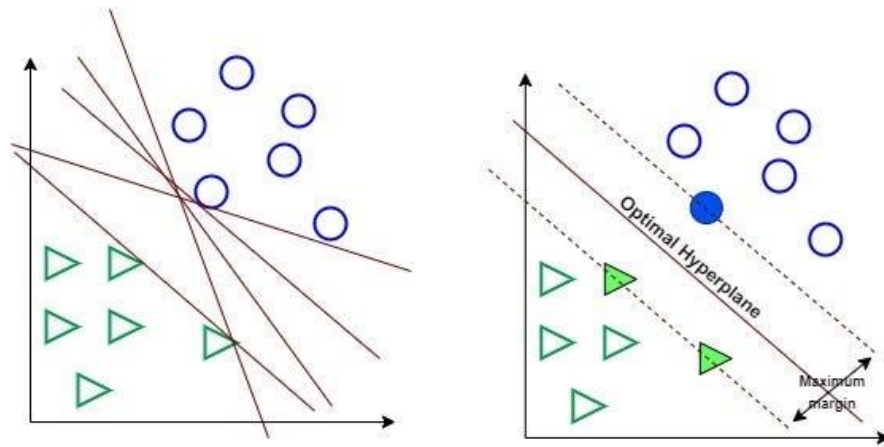
2.3 Data Rekam Medis

Data rekam medis merupakan catatan medis pasien yang berisi informasi mengenai kondisi kesehatan, riwayat penyakit, pemeriksaan, diagnosis, perawatan, tindakan medis, dan hasil laboratorium seseorang (Ordila et al., 2020). Data rekam medis dibuat dan disimpan oleh tenaga kesehatan. Informasi yang tercantum dalam data rekam medis antara lain riwayat penyakit pasien, diagnosa dokter, hasil pemeriksaan laboratorium dan radiologi, tindakan medis yang dilakukan, obat-obatan yang diberikan, tindak lanjut pengobatan, dan catatan perkembangan kondisi kesehatan pasien (Helling et al., 2019). Data rekam medis sangatlah penting karena berisi informasi medis lengkap mengenai pasien yang dibutuhkan tenaga kesehatan dalam memberikan layanan dan perawatan kesehatan secara tepat dan terbaik untuk pasien.

2.4 Support Vector Machine

Algoritma *supervised machine learning* yang bisa diterapkan dalam klasifikasi adalah *Support Vector Machine* (Handayani, 2021). Beberapa algoritma *machine learning* lainnya ketika dibandingkan dengan *Support Vector Machine* cenderung memberikan hasil klasifikasi dengan nilai akurasi lebih baik menjadi salah satu alasan *Support Vector Machine* diterapkan dalam klasifikasi. (Shanbhag et al., 2021). Dalam masalah klasifikasi menggunakan SVM, tujuannya adalah

untuk menentukan batas keputusan optimal (*hyperplane*) dengan memaksimalkan jarak (*margin*) yang mengisolasi setiap kategori data.



Gambar 2.1 Optimal *Hyperplane* pada *Support Vector Machine*

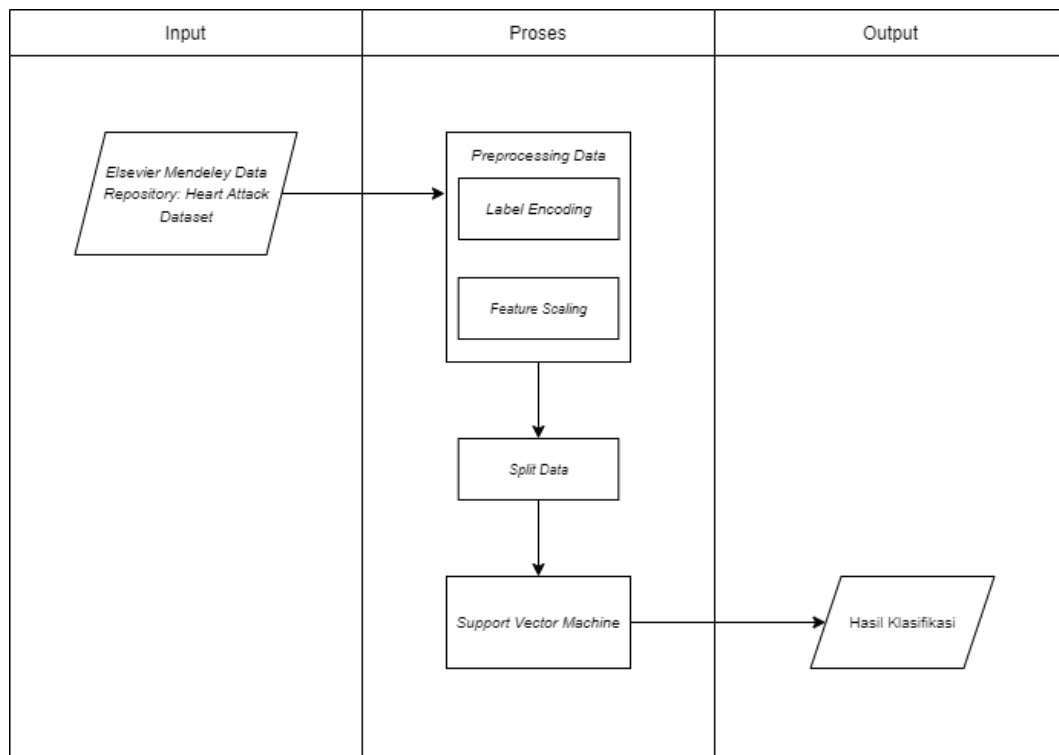
Pada Gambar 2.1, gambar di sebelah kiri menampilkan beberapa garis pemisah potensial (*hyperplane*) yang dapat memisahkan titik data lingkaran dan titik data segitiga. Algoritma SVM kemudian berupaya mengidentifikasi *hyperplane* optimal yang dapat membedakan dua kelas titik data secara efektif. Bisa dilihat pada gambar sebelah kanan, *hyperplane* yang optimal berhasil ditemukan dan mampu membedakan kedua kelas tersebut dengan cara memberikan margin atau jarak yang maksimal di antara keduanya.

BAB III

METODE PENELITIAN

3.1 Desain Sistem

Gambar 3.1 menggambarkan desain sistem yang digunakan dalam penelitian ini. Sistem ini terdiri dari beberapa langkah yang saling terkait dalam rangkaian proses.



Gambar 3.1 Desain Sistem

Penelitian ini bertujuan untuk mengembangkan sistem deteksi serangan jantung dengan menggunakan bahasa pemrograman python. Sistem ini menggunakan data input yang berasal dari *dataset* publik yang tersedia di *Elsevier Mendeley Data Repository*. *Dataset* ini memiliki judul *Heart Attack Dataset* dan

berisi data tentang faktor-faktor risiko, gejala, dan hasil diagnosis serangan jantung pada 1319 pasien. Data tersebut kemudian melalui tahap *preprocessing data* yang meliputi *Label Encoding*, dan *Feature Scaling*. Tahap selanjutnya adalah klasifikasi data dengan menggunakan metode *Support Vector Machine* (SVM). Metode SVM adalah salah satu metode pembelajaran mesin yang dapat digunakan untuk mengklasifikasikan data menjadi dua atau lebih kelas dengan menggunakan fungsi kernel yang sesuai. Dalam penelitian ini, fungsi kernel yang digunakan adalah fungsi kernel linear, RBF, dan polynomial. Setelah hasil klasifikasi diperoleh, dilakukan evaluasi performa untuk mengukur keberhasilan metode SVM dalam mengklasifikasikan data. Evaluasi performa dilakukan dengan menggunakan tabel *confusion matrix* yang menunjukkan jumlah data yang diklasifikasikan dengan benar dan salah untuk setiap kelas. Tabel *confusion matrix* juga dapat digunakan untuk menghitung metrik evaluasi lainnya, seperti akurasi, presisi, *recall*, dan *F1-score*.

3.2 Pengumpulan Data

Penelitian ini menggunakan data sekunder yang berasal dari *Elsevier Mendeley Data Repository*, yaitu sebuah *platform online* yang menyimpan dan membagikan data penelitian dari berbagai bidang ilmu. Data sekunder adalah data yang tidak dikumpulkan langsung oleh peneliti, melainkan diperoleh dari sumber lain yang telah melakukan pengumpulan data sebelumnya. Data sekunder ini kemudian diolah dan dianalisis oleh peneliti untuk tujuan penelitian, yaitu menguji metode klasifikasi dan mendapatkan hasil prediksi yang akurat. Penelitian ini

menggunakan *dataset* yang sama dari *Elsevier Mendeley Data Repository* sebagai data latih dan data uji.

Karena data yang digunakan adalah data sekunder yang dikumpulkan oleh pihak lain, maka peneliti perlu melakukan pemeriksaan awal terhadap kualitas data. Salah satu hal yang perlu diperiksa adalah adanya nilai yang hilang atau *missing value* dalam data. Nilai yang hilang adalah nilai yang seharusnya ada pada variabel tertentu, tetapi tidak tercatat atau tidak diketahui. Nilai yang hilang dapat menyebabkan bias atau kesalahan dalam analisis data jika tidak ditangani dengan tepat. Untuk melakukan pemeriksaan nilai yang hilang, peneliti menggunakan *package* *panda*, yaitu sebuah *library* Python yang menyediakan struktur data dan fungsi analisis data yang efisien dan mudah digunakan.

Dataset yang digunakan dalam penelitian ini adalah *dataset* publik yang tersedia di *Elsevier Mendeley Data Repository* dengan judul “Heart Attack Dataset”. *Dataset* ini berisi data rekam medis pasien mengenai serangan jantung. *Dataset* ini memiliki 1.319 baris data dan 9 kolom atribut. Terdapat 509 pasien yang didiagnosis positif menderita serangan jantung dan 810 pasien yang didiagnosis negatif menderita serangan jantung. Penjelasan lebih lanjut mengenai *dataset* ini dapat dilihat pada Tabel 3.1.

Tabel 3.1 Atribut *Dataset*

No.	Nama Atribut	Deskripsi Atribut
1	<i>Age</i>	Umur
2	<i>Gender</i>	Jenis kelamin (Pria “1” dan Wanita “0”)
3	<i>Heart rate</i>	Detak jantung
4	<i>Systolic BP</i>	Tekanan darah pada saat jantung berkontraksi dan memompa darah ke seluruh tubuh
5	<i>Diastolic BP</i>	Tekanan darah pada saat jantung sedang beristirahat (relaksasi) atau sebelum kembali memompa darah ke seluruh tubuh
6	<i>Blood sugar</i>	Kadar glukosa atau gula yang terdapat dalam darah
7	CK-MB	CK-MB atau <i>creatine kinase</i> -MB adalah enzim yang diproduksi oleh

No.	Nama Atribut	Deskripsi Atribut
7		otot jantung. Ketika jaringan otot jantung mengalami kerusakan, CK-MB akan dilepaskan ke dalam darah sehingga menyebabkan kadar CK-MB meningkat di dalam darah. Pengukuran kadar CK-MB dalam darah dapat membantu dokter mendiagnosis serangan jantung
8	Troponin	Troponin adalah protein yang terdapat pada otot jantung dan berperan dalam kontraksi otot jantung. Ketika terjadi kerusakan pada otot jantung, troponin akan dilepaskan ke dalam darah sehingga kadarnya dalam darah akan meningkat. Hal ini penting karena pengukuran kadar troponin dalam darah dapat membantu dokter mendiagnosis serangan jantung
9	Result	Klasifikasi ini bergantung pada 8 atribut yang telah disebutkan sebelumnya. Terdapat dua nilai yang berbeda untuk menunjukkan hasil klasifikasi, yaitu: Positif (ada risiko serangan jantung) dan Negatif (tidak ada risiko serangan jantung)

Untuk contoh *dataset* yang digunakan bisa dilihat pada Tabel 3.2.

Tabel 3.2 Contoh *Dataset*

Age	Gender	Heart rate	Systolic blood pressure	Diastolic blood pressure	Blood sugar	CK-MB	Troponin	Result
55	1	64	112	65	300	1.08	0.003	negative
58	0	61	112	58	87	1.83	0.004	negative
32	0	40	179	68	102	0.71	0.003	negative
63	1	60	214	82	87	3.00	2.370	positive
44	0	60	154	81	135	2.35	0.004	negative
67	1	61	160	95	100	2.84	0.011	negative
44	0	60	166	90	102	2.39	0.006	negative
63	0	60	150	83	198	2.39	0.013	negative
64	1	60	199	99	92	3.43	5.370	positive
54	0	94	122	67	97	1.42	0.012	negative

3.3 Preprocessing Data

Sebelum data dapat digunakan untuk penelitian, data perlu melalui tahap pra-pemrosesan data. Tahap ini bertujuan untuk membersihkan, mengubah, dan menstandarkan data agar sesuai dengan kebutuhan penelitian. Ada dua langkah utama dalam pra-pemrosesan data, yaitu *Label Encoding* dan *Feature Scaling*.

Label Encoding adalah proses mengubah data kategorikal menjadi data numerik. Data kategorikal adalah data yang memiliki nilai berupa kategori atau label, seperti jenis kelamin, warna, atau hasil. Data numerik adalah data yang

memiliki nilai berupa angka, seperti usia, berat, atau tinggi. Kebanyakan model pembelajaran mesin hanya dapat bekerja dengan data numerik, sehingga data kategorikal perlu diubah menjadi data numerik. Dalam penelitian ini, peneliti menggunakan *LabelEncoder*, yaitu sebuah fungsi dari *library* sklearn, untuk mengubah data kategorikal pada atribut “Result” menjadi data numerik. Atribut “Result” menunjukkan hasil klasifikasi pasien, yaitu positif atau negatif. Dengan menggunakan *LabelEncoder*, data kategorikal ini diubah menjadi data numerik, yaitu 1 untuk positif dan 0 untuk negatif.

Feature Scaling adalah proses menstandarkan skala nilai dari setiap atribut dalam *dataset*. Setiap atribut dalam *dataset* dapat memiliki rentang nilai yang berbeda-beda, misalnya usia berkisar antara 29 hingga 77, sedangkan tekanan darah berkisar antara 94 hingga 200. Perbedaan skala ini dapat menyebabkan atribut yang memiliki rentang nilai yang lebih besar memiliki pengaruh yang lebih besar terhadap model pembelajaran mesin, sehingga mengurangi akurasi prediksi. Untuk mengatasi hal ini, perlu dilakukan *Feature Scaling*, yaitu proses menyesuaikan skala nilai dari setiap atribut agar memiliki rata-rata 0 dan variansi 1. Dengan demikian, setiap atribut memiliki kontribusi yang sama terhadap model pembelajaran mesin. Dalam penelitian ini, peneliti menggunakan *StandardScaler*, yaitu sebuah fungsi dari *library* sklearn, untuk melakukan *Feature Scaling*. *StandardScaler* menghitung rata-rata dan standar deviasi dari setiap atribut, kemudian mengurangi setiap nilai dengan rata-rata dan membaginya dengan standar deviasi. Hasilnya adalah data yang memiliki skala yang seragam.

3.3.1 Label Encoding

Salah satu langkah dalam pra-pemrosesan data adalah *Label Encoding*, yaitu proses mengubah data kategorik menjadi data numerik. Data kategorik adalah data yang memiliki nilai berupa kategori atau label, seperti jenis kelamin, warna, atau hasil. Data numerik adalah data yang memiliki nilai berupa angka, seperti usia, berat, atau tinggi. Kebanyakan model pembelajaran mesin hanya dapat bekerja dengan data numerik, sehingga data kategorik perlu diubah menjadi data numerik. Dalam penelitian ini, peneliti menggunakan *LabelEncoder*, yaitu sebuah fungsi dari *library* sklearn, untuk mengubah data kategorik pada atribut “Result” menjadi data numerik. Atribut “Result” menunjukkan hasil klasifikasi pasien, yaitu positif atau negatif. Dengan menggunakan *LabelEncoder*, data kategorik ini diubah menjadi data numerik, yaitu 1 untuk positif dan 0 untuk negatif.

Untuk memperlihatkan perbedaan antara data sebelum dan sesudah *Label Encoding*, peneliti akan menampilkan 5 sampel data dari *dataset* yang digunakan dalam penelitian ini. Data ini terdiri dari 6 atribut, yaitu *Systolic BP*, *Diastolic BP*, *Blood Sugar*, CK-MB, dan Troponin. Atribut terakhir adalah atribut “Result”, yang merupakan variabel target atau kelas yang ingin diprediksi oleh model pembelajaran mesin. Tabel 3.3 menunjukkan data sebelum *Label Encoding*, sedangkan Tabel 3.4 menunjukkan data sesudah *Label Encoding*.

Tabel 3.3 Contoh Data Sebelum Menjalani Proses *Label Encoding*

Systolic BP	Diastolic BP	Blood sugar	CK-MB	Troponin	Result
160	83	160.0	1.80	0.012	negative
98	46	296.0	6.75	1.060	positive
160	77	270.0	1.99	0.003	negative
120	55	270.0	13.87	0.122	positive
112	65	300.0	1.08	0.003	negative
160	83	160.0	1.80	0.012	negative

Setelah melakukan *Label Encoding* pada atribut “Result”, peneliti akan menampilkan data yang telah diubah dalam bentuk tabel. Data ini berisi 6 sampel data dari *dataset* yang digunakan dalam penelitian ini. Tabel 3.4 menunjukkan data yang telah diubah dengan menggunakan *LabelEncoder*.

Tabel 3.4 Contoh Data Setelah Menjalani Proses *Label Encoding*

Systolic BP	Diastolic BP	Blood sugar	CK-MB	Troponin	Result
160	83	160.0	1.80	0.012	0
98	46	296.0	6.75	1.060	1
160	77	270.0	1.99	0.003	0
120	55	270.0	13.87	0.122	1
112	65	300.0	1.08	0.003	0
160	83	160.0	1.80	0.012	0

3.3.2 Feature Scaling

Feature scaling adalah proses yang digunakan untuk menstandarkan nilai-nilai dalam atribut atau fitur yang ada dalam *dataset*. Proses ini bertujuan untuk membuat semua fitur memiliki rentang nilai yang serupa, sehingga tidak ada fitur yang mendominasi atau diabaikan oleh algoritma *machine learning*. Hal ini penting karena algoritma *machine learning* biasanya menggunakan jarak atau perbedaan antara nilai fitur untuk menghitung kemiripan atau keterkaitan antara data. Jika ada fitur yang memiliki rentang nilai yang jauh berbeda dengan fitur lain, maka fitur tersebut akan memiliki pengaruh yang lebih besar atau lebih kecil pada hasil algoritma *machine learning*. Misalnya, jika kita memiliki *dataset* yang berisi informasi tentang pelanggan, seperti usia, pendapatan, dan tingkat kepuasan. Fitur usia mungkin berkisar antara 0 sampai 100 tahun, fitur pendapatan mungkin berkisar antara 0 sampai 10 juta rupiah, dan fitur tingkat kepuasan mungkin berkisar antara 1 sampai 5. Jika kita menggunakan algoritma *machine learning* untuk mengelompokkan pelanggan berdasarkan kemiripan mereka, maka fitur

pendapatan akan memiliki bobot yang lebih besar daripada fitur usia dan tingkat kepuasan, karena perbedaan nilai pendapatan antara pelanggan lebih besar daripada perbedaan nilai fitur lain. Hal ini dapat menyebabkan hasil yang tidak akurat, karena kita mungkin mengabaikan fitur lain yang juga penting untuk menentukan kemiripan antara pelanggan.

Untuk mengatasi masalah ini, kita dapat menggunakan berbagai teknik *feature scaling* untuk menyesuaikan rentang nilai fitur agar lebih seimbang. Salah satu teknik yang umum digunakan adalah StandardScaler. Teknik ini mengubah nilai fitur menjadi nilai *z-score*, yaitu nilai yang menunjukkan berapa banyak standar deviasi yang terdapat antara nilai fitur tersebut dengan rata-rata fitur tersebut. Dengan demikian, nilai fitur yang lebih tinggi atau lebih rendah dari rata-rata akan memiliki nilai *z-score* yang lebih besar atau lebih kecil dari nol, sedangkan nilai fitur yang sama dengan rata-rata akan memiliki nilai *z-score* yang sama dengan nol. Nilai *z-score* ini memiliki rentang yang serupa untuk semua fitur, tergantung pada distribusi data. Dengan menggunakan nilai *z-score* ini, kita dapat membuat algoritma *machine learning* lebih adil dalam memperlakukan semua fitur. Rumus matematika untuk menghitung nilai *z-score* menggunakan teknik StandardScaler adalah sebagai berikut:

$$z = \frac{x - \mu}{\sigma} \quad (3.1)$$

Keterangan:

z : nilai *z-score*

x : nilai fitur asli

μ : rata-rata fitur

σ : deviasi standar fitur

Dalam Tabel 3.5, peneliti akan menunjukkan 5 fitur atau atribut yang belum menjalani proses *feature scaling* menggunakan StandardScaler. Terdapat 5 data pada setiap fitur.

Tabel 3.5 Contoh Data Sebelum Proses *Feature Scaling*

Systolic BP	Diastolic BP	Blood sugar	CK-MB	Troponin
160	83	160.0	1.80	0.012
98	46	296.0	6.75	1.060
160	77	270.0	1.99	0.003
120	55	270.0	13.87	0.122
112	65	300.0	1.08	0.003

Untuk mengimplementasikan persamaan (3.1) pada Tabel 3.5, langkah pertama adalah menghitung rerata dan deviasi standar dari tiap atribut. Rerata bisa dihitung menggunakan persamaan (3.2).

$$\mu = \frac{\text{Jumlah seluruh nilai}}{\text{Jumlah sampel data}} \quad (3.2)$$

Cara untuk mengubah nilai pada atribut *Systolic BP* adalah dengan menghitung rata-rata dari atribut tersebut. Dari data yang ada, nilai rata-rata *Systolic BP* adalah 167.738 dibagi 1319, yaitu sekitar 127.171. Standar deviasi bisa dihitung menggunakan persamaan (3.3).

$$\sigma = \sqrt{\frac{\sum(x-\mu)^2}{N}} \quad (3.3)$$

Keterangan:

x : nilai fitur asli

μ : nilai rata-rata dari semua data

N : jumlah data

σ : nilai standar deviasi dari data

σ yang didapatkan yaitu $= \sqrt{(160-127.171)^2 + (98-127.171)^2 + (160-127.171)^2 + (120-127.171)^2 + (112-127.171)^2 + \dots / (1319)} = 26.1227$.

Setelah mendapatkan rerata dan deviasi standar dari tiap atribut, maka substitusikan dengan persamaan (3.1) dan didapatkan $\sigma = (160-127.171)/26.1227 = 1.25672307993$ atau 1.26. Dalam Tabel 3.6, peneliti akan menunjukkan 5 fitur atau atribut yang telah menjalani proses *feature scaling* menggunakan StandardScaler.

Tabel 3.6 Contoh Data Setelah Proses *Feature Scaling*

Systolic BP	Diastolic BP	Blood sugar	CK-MB	Troponin
1.26	0.76	0.18	-0.29	-0.30
-1.12	-1.87	1.99	-0.18	0.61
1.26	0.34	1.65	-0.29	-0.31
-0.27	-1.23	1.65	-0.03	-0.21
-0.58	-0.52	2.05	-0.31	-0.31

Nilai *z-score* bisa positif atau negatif, tergantung pada apakah nilai fitur di atas atau di bawah rata-rata populasi. Nilai *z-score* yang negatif menunjukkan bahwa nilai fitur lebih rendah dari rata-rata populasi, sedangkan nilai *z-score* yang positif menunjukkan bahwa nilai fitur lebih tinggi dari rata-rata populasi. Nilai negatif bukan berarti bahwa fitur itu salah atau tidak berguna, tetapi hanya menunjukkan posisi relatif fitur itu dalam distribusi populasi. Nilai *z-score* juga bisa digunakan untuk mengukur seberapa jauh nilai fitur dari rata-rata populasi dalam satuan standar deviasi. Semakin besar nilai *z-score*, semakin jauh nilai fitur dari rata-rata populasi.

3.4 Split Data

Split data adalah salah satu langkah penting dalam proses pengembangan model *machine learning*. Langkah ini melibatkan pemisahan suatu *dataset* menjadi dua bagian, yaitu data latih dan data uji. Data latih adalah data yang digunakan

untuk melatih model *machine learning*, yaitu proses yang mengajarkan model untuk mengenali pola atau hubungan antara fitur dan target dalam data. Data uji adalah data yang digunakan untuk menguji kinerja model *machine learning*, yaitu proses yang mengukur seberapa baik model dapat melakukan prediksi atau klasifikasi pada data baru yang belum pernah dilihat sebelumnya.

Split data perlu dilakukan karena model *machine learning* harus dapat beradaptasi dengan data yang bervariasi dan tidak terbatas. Jika model hanya dilatih dan diuji dengan data yang sama, maka model tersebut akan mengalami *overfitting*, yaitu kondisi di mana model terlalu spesifik pada data latih dan tidak dapat menggeneralisasi pada data baru. *Overfitting* dapat menyebabkan model memiliki kinerja yang buruk pada data uji, karena model tidak dapat menangkap variasi atau ketidakpastian yang ada pada data uji. Oleh karena itu, *split data* dapat membantu peneliti untuk mencegah *overfitting* dan meningkatkan kinerja model.

Salah satu metode yang umum digunakan untuk melakukan *split data* adalah dengan menggunakan fungsi *train_test_split* dari pustaka *scikit-learn*. Pustaka *scikit-learn* adalah pustaka python yang menyediakan berbagai alat dan algoritma untuk *machine learning* dan *data science*. Fungsi *train_test_split()* dapat memisahkan *dataset* menjadi data latih dan data uji dengan cara yang acak dan proporsional. Peneliti dapat menentukan ukuran atau persentase data latih dan data uji yang diinginkan. Fungsi ini akan mengembalikan empat objek, yaitu *x_train*, *x_test*, *y_train*, dan *y_test*, yang merupakan fitur dan target dari data latih dan data uji.

3.5 Support Vector Machine

Metode *Support Vector Machine* (SVM) diterapkan dalam penelitian ini dengan tiga jenis kernel, yaitu kernel linear, RBF, dan polynomial. SVM (*Support Vector Machine*) adalah sebuah metode pembelajaran mesin yang digunakan untuk melakukan klasifikasi biner, yaitu memisahkan dua kelas dengan sebuah *hyperplane* yang optimal.

Hyperplane adalah sebuah bidang yang membagi ruang menjadi dua bagian. Dalam kasus dua dimensi, *hyperplane* adalah sebuah garis lurus. Dalam kasus tiga dimensi, *hyperplane* adalah sebuah bidang datar. Dalam kasus yang lebih tinggi, *hyperplane* adalah sebuah subruang yang berdimensi satu kurang dari ruang asal.

Hyperplane yang optimal adalah *hyperplane* yang memiliki margin yang maksimal. Margin adalah jarak terkecil antara *hyperplane* dan titik-titik data dari kedua kelas. Dengan memiliki margin yang maksimal, kita dapat meningkatkan generalisasi dari model, yaitu kemampuan untuk memprediksi kelas dari data baru yang belum pernah dilihat sebelumnya.

Untuk menemukan *hyperplane* yang optimal, kita perlu menentukan parameter dari *hyperplane*, yaitu bobot dan bias. Bobot adalah sebuah vektor yang menentukan arah dari *hyperplane*. Bias adalah sebuah skalar yang menentukan jarak dari *hyperplane* ke titik asal.

Untuk menentukan bobot dan bias yang optimal, kita perlu meminimalkan sebuah fungsi biaya yang mengukur seberapa baik *hyperplane* memisahkan data. Fungsi biaya yang digunakan oleh SVM adalah sebagai berikut:

$$J(w, b) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^m \max(0, 1 - y_i(w^T x_i - b)) \quad (3.4)$$

Keterangan:

w adalah bobot

b adalah bias

C adalah sebuah konstanta yang mengontrol seberapa besar penalti untuk data yang salah diklasifikasikan atau berada di dalam margin. Nilai C yang lebih besar berarti penalti yang lebih besar, dan sebaliknya.

m adalah jumlah data

x_i adalah data ke- i

y_i adalah label kelas dari data ke- i , yang bernilai $+1$ atau -1

$\|w\|^2$ adalah norma kuadrat dari bobot, yang mengukur panjang dari bobot

$\max(0, 1 - y_i(w^T x_i - b))$ adalah sebuah fungsi yang disebut *hinge loss*, yang mengukur seberapa jauh data ke- i dari margin. Jika data ke- i berada di luar margin dan diklasifikasikan dengan benar, maka *hinge loss* bernilai 0. Jika data ke- i berada di dalam margin atau diklasifikasikan dengan salah, maka *hinge loss* bernilai lebih besar dari 0.

Fungsi biaya ini memiliki dua komponen: komponen pertama ($\frac{1}{2} \|w\|^2$) adalah untuk memaksimalkan margin, dan komponen kedua ($C \sum_{i=1}^m \max(0, 1 - y_i(w^T x_i - b))$) adalah untuk meminimalkan *hinge loss*. Dengan meminimalkan fungsi biaya ini, kita dapat menemukan bobot dan bias yang optimal.

Untuk meminimalkan fungsi biaya ini, kita dapat menggunakan sebuah metode yang disebut *gradient descent* (penurunan gradien). *Gradient descent* adalah sebuah algoritma yang secara iteratif meng-*update* bobot dan bias dengan mengurangi nilai mereka sebesar turunan parsial dari fungsi biaya terhadap bobot dan bias, dikalikan dengan laju pembelajaran. Laju pembelajaran adalah sebuah konstanta yang mengontrol seberapa besar langkah yang diambil oleh algoritma untuk mencapai nilai minimum. Nilai laju pembelajaran yang terlalu besar dapat

menyebabkan algoritma melompati nilai minimum, dan nilai laju pembelajaran yang terlalu kecil dapat menyebabkan algoritma berjalan terlalu lambat. Turunan parsial dari fungsi biaya terhadap bobot dan bias dapat ditulis sebagai berikut:

$$\frac{\partial J}{\partial w} = w + C \sum_{i=1}^m -y_i x_i I(y_i(w^T x_i - b) < 1) \quad (3.5)$$

$$\frac{\partial J}{\partial b} = C \sum_{i=1}^m -y_i I(y_i(w^T x_i - b) < 1) \quad (3.6)$$

Keterangan:

I adalah sebuah fungsi yang disebut indikator, yang bernilai 1 jika kondisi di dalam kurung benar, dan bernilai 0 jika kondisi di dalam kurung salah.

Dengan menggunakan turunan parsial ini, kita dapat meng-*update* bobot dan bias dengan rumus berikut:

$$w := w - \alpha \frac{\partial J}{\partial w} \quad (3.7)$$

$$b := b - \alpha \frac{\partial J}{\partial b} \quad (3.8)$$

Keterangan:

α adalah laju pembelajaran

$:=$ adalah sebuah simbol yang berarti ditugaskan menjadi

SVM mengimplementasikan algoritma *gradient descent* ini dengan menggunakan sebuah *loop* untuk meng-*update* bobot dan bias secara berulang sebanyak jumlah iterasi yang ditentukan. *Loop* ini juga menggunakan sebuah kondisi untuk memeriksa apakah data ke- i memenuhi syarat margin atau tidak.

Setelah bobot dan bias yang optimal ditemukan, kita dapat menggunakan *hyperplane* yang optimal untuk memprediksi kelas dari data baru. Untuk melakukan prediksi, kita dapat menggunakan rumus berikut:

$$\hat{y} = \text{sign}(w^T x - b) \quad (3.9)$$

Keterangan:

\hat{y} adalah prediksi kelas, yang bernilai +1 atau -1
 sign adalah sebuah fungsi yang mengembalikan tanda dari argumennya, yaitu +1 jika argumennya positif, -1 jika argumennya negatif, dan 0 jika argumennya nol.

3.6 Evaluasi Performa

Evaluasi performa adalah salah satu tahapan terakhir dalam penelitian yang menggunakan metode pembelajaran mesin. Tahapan ini bertujuan untuk mengetahui seberapa baik model atau algoritma yang telah dibuat dapat memprediksi atau mengklasifikasikan data dengan benar. Evaluasi performa ini penting dilakukan untuk mengukur kualitas dan keandalan dari model atau algoritma, serta untuk mengidentifikasi kelebihan dan kekurangan dari model atau algoritma tersebut.

Salah satu cara untuk melakukan evaluasi performa adalah dengan menggunakan tabel *confusion matrix*. Tabel *confusion matrix* adalah sebuah tabel yang menampilkan perbandingan antara kelas prediksi yang dihasilkan oleh model atau algoritma dengan kelas data yang sebenarnya. Tabel *confusion matrix* dapat digunakan untuk menghitung berbagai metrik evaluasi, seperti akurasi, presisi, *recall*, dan F1-score. Tabel *confusion matrix* memiliki empat komponen utama, yaitu:

1. *True Positive* (TP): TP mewakili jumlah kasus positif yang berhasil diidentifikasi dengan benar oleh model.
2. *True Negative* (TN): TN mewakili jumlah kasus negatif yang berhasil diidentifikasi dengan benar oleh model.
3. *False Positive* (FP): FP mewakili jumlah kasus negatif yang keliru diidentifikasi sebagai positif oleh model.
4. *False Negative* (FN): FN mewakili jumlah kasus positif yang keliru diidentifikasi sebagai negatif oleh model.

Akurasi adalah salah satu metrik evaluasi yang sering digunakan untuk mengukur kinerja dari model pembelajaran mesin, khususnya dalam tugas klasifikasi. Tugas klasifikasi adalah tugas yang mengelompokkan data menjadi beberapa kelas yang telah ditentukan sebelumnya, seperti kelas positif dan negatif. Model pembelajaran mesin adalah model yang dapat belajar dari data dan membuat prediksi atau keputusan berdasarkan data tersebut.

Akurasi mengukur seberapa sering model dapat membuat prediksi yang benar dari seluruh data yang ada. Prediksi yang benar adalah prediksi yang sesuai dengan kelas data yang sebenarnya. Misalnya, jika model memprediksi bahwa sebuah email adalah spam, dan email tersebut memang spam, maka prediksi tersebut adalah benar. Sebaliknya, jika model memprediksi bahwa sebuah email adalah bukan spam, tetapi email tersebut sebenarnya spam, maka prediksi tersebut adalah salah. Nilai akurasi dapat dihitung menggunakan rumus (3.10).

$$Akurasi = \frac{TP+TN}{(TP+TN+FP+FN)} \times 100\% \quad (3.10)$$

Presisi adalah salah satu metrik evaluasi yang sering digunakan untuk mengukur kinerja dari model pembelajaran mesin, khususnya dalam tugas klasifikasi. Tugas klasifikasi adalah tugas yang mengelompokkan data menjadi beberapa kelas yang telah ditentukan sebelumnya, seperti kelas positif dan negatif. Model pembelajaran mesin adalah model yang dapat belajar dari data dan membuat prediksi atau keputusan berdasarkan data tersebut.

Presisi mengukur seberapa sering model dapat membuat prediksi yang benar untuk kelas positif dari seluruh prediksi positif yang dibuat. Prediksi positif adalah prediksi yang mengatakan bahwa data termasuk dalam kelas positif. Prediksi yang benar adalah prediksi yang sesuai dengan kelas data yang sebenarnya. Misalnya, jika model memprediksi bahwa sebuah email adalah spam, dan email tersebut memang spam, maka prediksi tersebut adalah benar dan positif. Sebaliknya, jika model memprediksi bahwa sebuah email adalah spam, tetapi email tersebut sebenarnya bukan spam, maka prediksi tersebut adalah salah dan positif. Nilai presisi dapat dihitung dengan rumus (3.11).

$$Presisi = \frac{TP}{(TP+FP)} \times 100\% \quad (3.11)$$

Recall adalah salah satu metrik evaluasi yang sering digunakan untuk mengukur kinerja dari model pembelajaran mesin, khususnya dalam tugas klasifikasi. Tugas klasifikasi adalah tugas yang mengelompokkan data menjadi beberapa kelas yang telah ditentukan sebelumnya, seperti kelas positif dan negatif.

Model pembelajaran mesin adalah model yang dapat belajar dari data dan membuat prediksi atau keputusan berdasarkan data tersebut.

Recall mengukur seberapa sering model dapat menemukan data yang benar-benar positif dari seluruh data yang positif. Data positif adalah data yang termasuk dalam kelas positif. Data yang benar-benar positif adalah data yang diprediksi sebagai kelas positif dan benar-benar merupakan kelas positif. Misalnya, jika model memprediksi bahwa sebuah email adalah spam, dan email tersebut memang spam, maka email tersebut adalah data yang benar-benar positif. Sebaliknya, jika model memprediksi bahwa sebuah email adalah bukan spam, tetapi email tersebut sebenarnya spam, maka email tersebut adalah data yang benar-benar positif tetapi tidak ditemukan oleh model. Nilai *recall* dapat dihitung dengan rumus (3.12).

$$Recall = \frac{TP}{(TP+FN)} \times 100\% \quad (3.12)$$

F1-score adalah metrik evaluasi yang mencerminkan keseimbangan antara presisi dan *recall*. Presisi adalah proporsi data yang benar-benar positif dari data yang diprediksi positif. *Recall* adalah proporsi data yang diprediksi positif dari data yang benar-benar positif. Kedua metrik ini memiliki *trade-off*, yaitu jika kita meningkatkan presisi, maka *recall* akan menurun, dan sebaliknya. Oleh karena itu, kita perlu mencari nilai optimal dari kedua metrik ini, yang dapat diwakili oleh F1-score.

F1-score adalah rata-rata harmonik dari presisi dan *recall*. Rata-rata harmonik adalah rata-rata yang memberikan bobot lebih besar pada nilai yang lebih rendah. Dengan menggunakan rata-rata harmonik, F1-score akan memiliki nilai

yang tinggi jika presisi dan *recall* memiliki nilai yang tinggi, dan akan memiliki nilai yang rendah jika presisi atau *recall* memiliki nilai yang rendah. Nilai *F1-score* dapat dihitung dengan rumus (3.13).

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (3.13)$$

3.7 Skenario Pengujian

Skenario pengujian adalah sebuah metode yang digunakan untuk menguji kinerja dari sistem yang telah dibuat dengan menggunakan data yang berbeda. Dalam penelitian ini, sistem yang dibuat adalah sebuah model pembelajaran mesin yang dapat melakukan klasifikasi biner pada *dataset* yang telah dijelaskan pada sub-bab 3.2. *Dataset* ini memiliki dua kelas, yaitu kelas positif dan kelas negatif.

Untuk menguji kinerja dari model pembelajaran mesin, *dataset* ini dibagi menjadi dua bagian, yaitu data pelatihan dan data pengujian. Data pelatihan adalah data yang digunakan untuk melatih model pembelajaran mesin, yaitu proses yang mengajarkan model untuk mengenali pola atau hubungan antara fitur dan target dalam data. Data pengujian adalah data yang digunakan untuk menguji kinerja model pembelajaran mesin, yaitu proses yang mengukur seberapa baik model dapat melakukan prediksi atau klasifikasi pada data baru yang belum pernah dilihat sebelumnya.

Dalam penelitian ini, *dataset* dibagi menjadi empat skenario pengujian yang berbeda, dengan distribusi persentase data pelatihan dan data pengujian yang berbeda pula. Tujuan dari pembagian ini adalah untuk menemukan skenario pengujian yang dapat memberikan akurasi terbaik untuk model pembelajaran

mesin. Akurasi adalah proporsi data yang diprediksi dengan benar dari seluruh data. Skenario pengujian yang berbeda ini dapat mempengaruhi kinerja model pembelajaran mesin, karena jumlah dan variasi data pelatihan dan data pengujian dapat mempengaruhi kemampuan model untuk belajar dan menggeneralisasi.

Distribusi persentase data pelatihan dan data pengujian dalam setiap skenario pengujian dapat dilihat pada Tabel 3.7 berikut.

Tabel 3.7 Skenario Pembagian *Dataset*

Skenario	Persentase Data Latih	Persentase Data Uji
A	85%	15%
B	80%	20%
C	75%	25%
D	70%	30%

Skenario A menggunakan total 1319 data, yang terdiri dari 1121 data latih dan 198 data uji. Dari 1121 data latih, 692 pasien diklasifikasikan sebagai positif serangan jantung, sedangkan 429 pasien diklasifikasikan sebagai negatif serangan jantung. Pada 198 data uji, terdapat 118 pasien positif serangan jantung dan 80 pasien negatif serangan jantung.

Skenario B menggunakan total 1319 data, yang terbagi menjadi 1055 data latih dan 264 data uji. Dari 1055 data latih, 647 pasien termasuk dalam kategori positif serangan jantung, sementara 408 pasien termasuk dalam kategori negatif serangan jantung. Pada 264 data uji, terdapat 163 pasien positif serangan jantung dan 101 pasien negatif serangan jantung.

Skenario C juga menggunakan total 1319 data, dengan pembagian 989 data latih dan 330 data uji. Dari 989 data latih, 608 pasien diklasifikasikan sebagai positif serangan jantung, sedangkan 381 pasien diklasifikasikan sebagai negatif serangan

jantung. Pada 330 data uji, terdapat 202 pasien positif serangan jantung dan 128 pasien negatif serangan jantung.

Skenario D, seperti skenario sebelumnya, menggunakan total 1319 data. Namun, pembagian data latih dan data uji sedikit berbeda, yaitu 923 data latih dan 396 data uji. Dari 923 data latih, 569 pasien termasuk dalam kategori positif serangan jantung, sementara 354 pasien termasuk dalam kategori negatif serangan jantung. Pada 396 data uji, terdapat 241 pasien positif serangan jantung dan 155 pasien negatif serangan jantung.

Sub-skenario parameter uji coba adalah sebuah metode yang digunakan untuk menguji kinerja dari model pembelajaran mesin dengan menggunakan parameter yang berbeda. Parameter adalah nilai yang dapat diubah untuk mengatur perilaku dari model pembelajaran mesin. Dalam penelitian ini, model pembelajaran mesin yang digunakan adalah *Support Vector Machine* (SVM), yang merupakan sebuah metode klasifikasi biner yang dapat memisahkan dua kelas dengan sebuah *hyperplane* yang optimal.

SVM memiliki beberapa parameter yang dapat diubah, seperti *hyperparameter C* dan fungsi kernel. *Hyperparameter C* adalah sebuah nilai yang menentukan seberapa besar toleransi terhadap kesalahan klasifikasi. Nilai C yang lebih besar menunjukkan bahwa model lebih sensitif terhadap kesalahan klasifikasi, sehingga model akan mencoba memisahkan data dengan margin yang sempit. Nilai C yang lebih kecil menunjukkan bahwa model lebih toleran terhadap kesalahan klasifikasi, sehingga model akan mencoba memisahkan data dengan margin yang lebar.

Fungsi kernel adalah sebuah fungsi yang dapat mengubah data dari ruang asal ke ruang yang lebih tinggi, di mana data dapat dipisahkan dengan *hyperplane* yang linear. Ada berbagai jenis fungsi kernel yang dapat digunakan, seperti kernel linear, RBF, dan polynomial. Kernel linear adalah kernel yang paling sederhana, yang tidak mengubah data dari ruang asal. Kernel RBF adalah kernel yang menggunakan fungsi radial basis, yang dapat membuat *hyperplane* yang melengkung. Kernel polynomial adalah kernel yang menggunakan fungsi polinomial, yang dapat membuat *hyperplane* yang berbentuk polinomial.

Dalam penelitian ini, terdapat lima nilai *hyperparameter* C dan tiga fungsi kernel SVM yang akan diuji coba pada setiap skenario pembagian *dataset* (Afandi et al., 2023). Skenario pembagian *dataset* adalah sebuah metode yang digunakan untuk membagi data menjadi data pelatihan dan data pengujian dengan proporsi yang berbeda. Data pelatihan adalah data yang digunakan untuk melatih model, sedangkan data pengujian adalah data yang digunakan untuk menguji kinerja model. Dalam penelitian ini, terdapat empat skenario pembagian *dataset*, yaitu 85% data pelatihan dan 15% data pengujian, 80% data pelatihan dan 20% data pengujian, 75% data pelatihan dan 25% data pengujian, dan 70% data pelatihan dan 30% data pengujian.

Tujuan dari sub-skenario parameter uji coba adalah untuk menemukan kombinasi terbaik dari *hyperparameter* C dan fungsi kernel SVM yang dapat memberikan akurasi terbaik untuk model pada setiap skenario pembagian *dataset*. Akurasi adalah proporsi data yang diprediksi dengan benar dari seluruh data. Dengan menggunakan sub-skenario parameter uji coba, peneliti dapat mengetahui

pengaruh dari parameter terhadap kinerja model, serta dapat memilih parameter yang optimal untuk model.

Tabel 3.8 Sub-skenario Parameter Model SVM Kernel Linear

Parameter	
Tipe SVM	Classification
Kernel SVM	Linear
Nilai Hyperparameter C	0.01, 0.1, 1, 10, 100

Tabel 3.9 Sub-skenario Parameter Model SVM Kernel RBF

Parameter	
Tipe SVM	<i>Classification</i>
Kernel SVM	RBF
Nilai <i>Hyperparameter C</i>	0.01, 0.1, 1, 10, 100

Tabel 3.10 Sub-skenario Parameter Model SVM Kernel Polynomial

Parameter	
Tipe SVM	Classification
Kernel SVM	Polynomial
Nilai Hyperparameter C	0.01, 0.1, 1, 10, 100
Degree	3, 5, 7, 9, 11

BAB IV

UJI COBA DAN PEMBAHASAN

4.1 Langkah-Langkah Uji Coba

Untuk mengevaluasi kinerja sistem yang telah dikembangkan, serangkaian tes dilakukan untuk mengukur tingkat *error* pada model. Ada beberapa tahap yang perlu dilalui sebelum mendapatkan nilai *error* dari model yang diuji.

Dalam penelitian ini, tes dilakukan untuk mendeteksi serangan jantung berdasarkan data medis menggunakan algoritma pembelajaran mesin *Support Vector Machine*. Langkah-langkah yang dilakukan terdiri dari:

1. *Dataset* yang digunakan berasal dari *dataset* publik yang diambil dari *Elsevier Mendeley Data Repository*. Terdapat dua kelas data, yaitu kelas Serangan Jantung dan kelas Non Serangan Jantung. *Dataset* ini terdiri dari 1319 sampel, dengan 9 fitur pada setiap sampel.
2. *Dataset* kemudian dipisahkan menjadi dua bagian, yaitu data latih dan data uji, sesuai dengan skenario pembagian *dataset* yang dijelaskan pada subbab 3.7. Pembagian *dataset* dilakukan menggunakan metode `train_test_split()`.
3. Melakukan percobaan dengan berbagai kombinasi parameter SVM, seperti fungsi kernel, nilai *hyperparameter C*, dan *degree* (Kernel Polynomial) untuk menemukan kombinasi parameter terbaik.
4. Melatih model SVM menggunakan data latih. Model SVM akan memetakan data ke dalam ruang berdimensi tinggi, tergantung pada fungsi kernel yang dipilih, seperti Kernel RBF dan Polynomial.

5. Menguji model SVM yang telah dilatih dengan data uji untuk mengevaluasi model dalam mengklasifikasikan serangan jantung. Evaluasi model mencakup perhitungan nilai akurasi, presisi, *recall*, dan *f1-score* berdasarkan *confusion matrix* hasil klasifikasi data uji.
6. Menganalisis dan membandingkan performa model SVM dengan berbagai skenario pembagian data, parameter kernel, dan nilai *hyperparameter C* untuk mendapatkan kesimpulan model SVM terbaik untuk deteksi serangan jantung.

4.2 Hasil Uji Coba

Pada subbab ini, penulis akan membahas hasil uji coba sistem deteksi serangan jantung yang menggunakan metode *Support Vector Machine (SVM)*. Metode SVM merupakan salah satu metode pembelajaran mesin yang dapat digunakan untuk mengklasifikasikan data berdasarkan fitur-fitur yang dipilih. Tingkat keberhasilan metode SVM dalam sistem ini ditentukan berdasarkan hasil perbandingan antara nilai y_{test} dan y_{pred} . Nilai y_{test} adalah kelas sebenarnya dari data uji yang diberikan, sedangkan nilai y_{pred} adalah kelas yang diprediksi oleh model SVM yang telah dilatih dengan data latih. Setelah mendapatkan nilai y_{test} dan y_{pred} , penulis akan menghitung akurasi, presisi, *recall*, dan *F1-score* dari sistem deteksi.

Untuk memvisualisasikan hasil deteksi, penulis akan menggunakan *confusion matrix*. *Confusion matrix* adalah tabel yang menunjukkan distribusi data uji berdasarkan kelas sebenarnya dan kelas prediksi. *Confusion matrix* dapat

digunakan untuk menghitung akurasi, presisi, *recall*, dan *F1-score* dengan rumus-rumus tertentu. Dengan menggunakan *confusion matrix*, penulis dapat melihat seberapa baik sistem klasifikasi dalam mendeteksi serangan jantung.

4.2.1 Hasil Uji Coba Skenario A

Sebanyak 1319 data pasien telah disiapkan dan dibagi menjadi dua kategori, yaitu data pelatihan dan data pengujian, dengan rasio 85:15. Dengan demikian, terdapat 1121 data pelatihan dan 198 data pengujian.

Tabel 4.1 Performa Sistem Skenario A

Nilai Hyperparameter C	Kernel	Akurasi	Presisi	Recall	F1-Score
0.01	Linear	68%	71%	62%	61%
	RBF	60%	30%	50%	37%
	Polynomial Degree 3	60%	30%	50%	37%
	Polynomial Degree 5	60%	30%	50%	37%
	Polynomial Degree 7	61%	68%	51%	41%
	Polynomial Degree 9	60%	55%	51%	41%
	Polynomial Degree 11	60%	55%	51%	42%
0.1	Linear	74%	74%	72%	72%
	RBF	66%	68%	60%	58%
	Polynomial Degree 3	60%	55%	51%	41%
	Polynomial Degree 5	60%	55%	51%	41%
	Polynomial Degree 7	60%	55%	51%	42%
	Polynomial Degree 9	60%	58%	51%	43%
	Polynomial Degree 11	60%	55%	51%	42%
1	Linear	77%	76%	76%	76%
	RBF	71%	70%	67%	68%
	Polynomial Degree 3	65%	67%	58%	56%
	Polynomial Degree 5	63%	64%	55%	50%
	Polynomial Degree 7	61%	60%	52%	44%

Nilai Hyperparameter C	Kernel	Akurasi	Presisi	Recall	F1-Score
	Polynomial Degree 9	61%	62%	52%	43%
	Polynomial Degree 11	61%	64%	52%	44%
10	Linear	85%	84%	85%	84%
	RBF	76%	76%	73%	74%
	Polynomial Degree 3	70%	70%	67%	67%
	Polynomial Degree 5	68%	69%	62%	61%
	Polynomial Degree 7	63%	64%	56%	52%
	Polynomial Degree 9	60%	55%	52%	45%
	Polynomial Degree 11	60%	57%	52%	45%
100	Linear	92%	91%	92%	92%
	RBF	81%	81%	79%	80%
	Polynomial Degree 3	77%	77%	75%	75%
	Polynomial Degree 5	70%	71%	66%	66%
	Polynomial Degree 7	67%	67%	61%	60%
	Polynomial Degree 9	64%	63%	57%	55%
	Polynomial Degree 11	63%	63%	56%	52%

Tabel 4.1 menunjukkan hasil performa sistem (Skenario A) yang mencakup akurasi, presisi, *recall*, dan *f1-score*. Dengan pengaturan *hyperparameter C* sebesar 0.01, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 68%, presisi tertinggi sebesar 71%, *recall* tertinggi sebesar 62%, dan *f1-score* tertinggi sebesar 61%. Dengan pengaturan *hyperparameter C* sebesar 0.1, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 74%, presisi tertinggi sebesar 74%, *recall* tertinggi sebesar 72%, dan *f1-score* tertinggi sebesar 72%. Dengan pengaturan *hyperparameter C* sebesar 1, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 77%, presisi tertinggi sebesar 76%, *recall* tertinggi sebesar 76%, dan *f1-*

score tertinggi sebesar 76%. Dengan pengaturan *hyperparameter* C sebesar 10, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 85%, presisi tertinggi sebesar 84%, *recall* tertinggi sebesar 85%, dan *f1-score* tertinggi sebesar 84%. Dengan pengaturan *hyperparameter* C sebesar 100, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 92%, presisi tertinggi sebesar 91%, *recall* tertinggi sebesar 92%, dan *f1-score* tertinggi sebesar 92%.

Berdasarkan tabel 4.1, terlihat bahwa ketika C=100, SVM dengan kernel Linear mencapai performa puncak dengan akurasi 92%, presisi 91%, *recall* 92%, dan *f1-score* 92%. Dengan demikian, berdasarkan hasil uji coba skenario A, dapat disimpulkan bahwa konfigurasi *hyperparameter* C=100 dan kernel Linear memberikan hasil terbaik untuk model SVM.

4.2.2 Hasil Uji Coba Skenario B

Sebanyak 1319 data pasien telah disiapkan dan dibagi menjadi dua kategori, yaitu data pelatihan dan data pengujian, dengan rasio 80:20. Dengan demikian, terdapat 1055 data pelatihan dan 264 data pengujian.

Tabel 4.2 Performa Sistem Skenario B

Nilai Hyperparameter C	Kernel	Akurasi	Presisi	Recall	F1-Score
0.01	Linear	69%	70%	62%	61%
	RBF	62%	31%	50%	38%
	Polynomial Degree 3	62%	31%	50%	38%
	Polynomial Degree 5	62%	31%	50%	38%
	Polynomial Degree 7	62%	69%	51%	41%
	Polynomial Degree 9	62%	56%	51%	41%
	Polynomial Degree 11	62%	56%	51%	42%

Nilai Hyperparameter C	Kernel	Akurasi	Presisi	Recall	F1-Score
0.1	Linear	77%	76%	74%	75%
	RBF	69%	72%	62%	60%
	Polynomial Degree 3	62%	56%	51%	42%
	Polynomial Degree 5	62%	69%	51%	41%
	Polynomial Degree 7	62%	59%	51%	43%
	Polynomial Degree 9	63%	66%	53%	45%
	Polynomial Degree 11	63%	65%	52%	44%
1	Linear	80%	78%	79%	78%
	RBF	73%	72%	70%	70%
	Polynomial Degree 3	67%	68%	60%	58%
	Polynomial Degree 5	65%	68%	56%	51%
	Polynomial Degree 7	64%	68%	54%	47%
	Polynomial Degree 9	64%	68%	53%	46%
	Polynomial Degree 11	64%	68%	53%	46%
10	Linear	86%	85%	87%	86%
	RBF	80%	79%	77%	78%
	Polynomial Degree 3	73%	72%	69%	69%
	Polynomial Degree 5	69%	69%	62%	61%
	Polynomial Degree 7	65%	64%	57%	54%
	Polynomial Degree 9	62%	58%	53%	49%
	Polynomial Degree 11	62%	56%	52%	46%
100	Linear	92%	91%	92%	92%
	RBF	83%	83%	81%	82%
	Polynomial Degree 3	78%	77%	75%	76%
	Polynomial Degree 5	70%	68%	65%	66%
	Polynomial Degree 7	66%	64%	60%	59%
	Polynomial Degree 9	66%	65%	59%	58%
	Polynomial Degree 11	66%	65%	58%	56%

Tabel 4.2 menunjukkan hasil performa sistem (Skenario B) yang mencakup akurasi, presisi, *recall*, dan *f1-score*. Dengan pengaturan *hyperparameter* C sebesar 0.01, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 69%, presisi tertinggi sebesar 70%, *recall* tertinggi sebesar 62%, dan *f1-score* tertinggi sebesar 61%. Dengan pengaturan *hyperparameter* C sebesar 0.1, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 77%, presisi tertinggi sebesar 76%, *recall* tertinggi sebesar 74%, dan *f1-score* tertinggi sebesar 75%. Dengan pengaturan *hyperparameter* C sebesar 1, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 77%, presisi tertinggi sebesar 80%, *recall* tertinggi sebesar 79%, dan *f1-score* tertinggi sebesar 78%. Dengan pengaturan *hyperparameter* C sebesar 10, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 86%, presisi tertinggi sebesar 85%, *recall* tertinggi sebesar 87%, dan *f1-score* tertinggi sebesar 86%. Dengan pengaturan *hyperparameter* C sebesar 100, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 92%, presisi tertinggi sebesar 91%, *recall* tertinggi sebesar 92%, dan *f1-score* tertinggi sebesar 92%.

Berdasarkan tabel 4.2, terlihat bahwa ketika $C=100$, SVM dengan kernel Linear mencapai performa puncak dengan akurasi 92%, presisi 91%, *recall* 92%, dan *f1-score* 92%. Dengan demikian, berdasarkan hasil uji coba skenario B, dapat disimpulkan bahwa konfigurasi *hyperparameter* $C=100$ dan kernel Linear memberikan hasil terbaik untuk model SVM.

4.2.3 Hasil Uji Coba Skenario C

Sebanyak 1319 data pasien telah disiapkan dan dibagi menjadi dua kategori, yaitu data pelatihan dan data pengujian, dengan rasio 75:25. Dengan demikian, terdapat 989 data pelatihan dan 330 data pengujian.

Tabel 4.3 Performa Sistem Skenario C

Nilai Hyperparameter C	Kernel	Akurasi	Presisi	Recall	F1-Score
0.01	Linear	69%	69%	62%	62%
	RBF	61%	31%	50%	38%
	Polynomial Degree 3	61%	31%	50%	38%
	Polynomial Degree 5	61%	31%	50%	38%
	Polynomial Degree 7	62%	81%	51%	40%
	Polynomial Degree 9	62%	59%	51%	41%
	Polynomial Degree 11	62%	58%	51%	42%
0.1	Linear	75%	73%	72%	72%
	RBF	67%	67%	60%	59%
	Polynomial Degree 3	62%	59%	51%	41%
	Polynomial Degree 5	62%	71%	51%	41%
	Polynomial Degree 7	61%	56%	51%	41%
	Polynomial Degree 9	62%	64%	52%	44%
	Polynomial Degree 11	62%	63%	52%	43%
1	Linear	77%	76%	76%	76%
	RBF	71%	69%	68%	69%
	Polynomial Degree 3	67%	68%	59%	57%
	Polynomial Degree 5	65%	67%	55%	51%
	Polynomial Degree 7	63%	66%	53%	46%
	Polynomial Degree 9	62%	64%	52%	44%
	Polynomial Degree 11	63%	68%	52%	44%
10	Linear	86%	85%	87%	86%
	RBF	77%	76%	75%	76%
	Polynomial Degree 3	71%	70%	67%	68%
	Polynomial	66%	65%	60%	59%

Nilai Hyperparameter C	Kernel	Akurasi	Presisi	Recall	F1-Score
10	Degree 5				
	Polynomial Degree 7	63%	61%	55%	53%
	Polynomial Degree 9	63%	61%	54%	51%
	Polynomial Degree 11	62%	59%	53%	48%
100	Linear	92%	92%	92%	92%
	RBF	81%	80%	79%	79%
	Polynomial Degree 3	77%	76%	75%	76%
	Polynomial Degree 5	69%	67%	64%	65%
	Polynomial Degree 7	64%	61%	58%	57%
	Polynomial Degree 9	63%	60%	56%	54%
	Polynomial Degree 11	64%	62%	56%	53%

Tabel 4.3 menunjukkan hasil performa sistem (Skenario C) yang mencakup akurasi, presisi, *recall*, dan *f1-score*. Dengan pengaturan *hyperparameter C* sebesar 0.01, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 69%, presisi tertinggi sebesar 69%, *recall* tertinggi sebesar 62%, dan *f1-score* tertinggi sebesar 62%. Dengan pengaturan *hyperparameter C* sebesar 0.1, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 75%, presisi tertinggi sebesar 73%, *recall* tertinggi sebesar 72%, dan *f1-score* tertinggi sebesar 72%. Dengan pengaturan *hyperparameter C* sebesar 1, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 77%, presisi tertinggi sebesar 76%, *recall* tertinggi sebesar 76%, dan *f1-score* tertinggi sebesar 76%. Dengan pengaturan *hyperparameter C* sebesar 10, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 86%, presisi tertinggi sebesar 85%, *recall* tertinggi sebesar 87%, dan *f1-score* tertinggi sebesar 86%. Dengan pengaturan *hyperparameter C* sebesar 100, SVM dengan kernel

Linear mencapai akurasi tertinggi sebesar 92%, presisi tertinggi sebesar 92%, *recall* tertinggi sebesar 92%, dan *f1-score* tertinggi sebesar 92%.

Berdasarkan tabel 4.3, terlihat bahwa ketika $C=100$, SVM dengan kernel Linear mencapai performa puncak dengan akurasi 92%, presisi 92%, *recall* 92%, dan *f1-score* 92%. Dengan demikian, berdasarkan hasil uji coba skenario B, dapat disimpulkan bahwa konfigurasi *hyperparameter* $C=100$ dan kernel Linear memberikan hasil terbaik untuk model SVM.

4.2.4 Hasil Uji Coba Skenario D

Sebanyak 1319 data pasien telah disiapkan dan dibagi menjadi dua kategori, yaitu data pelatihan dan data pengujian, dengan rasio 70:30. Dengan demikian, terdapat 923 data pelatihan dan 396 data pengujian.

Tabel 4.4 Performa Sistem Skenario D

Nilai Hyperparameter C	Kernel	Akurasi	Presisi	Recall	F1-Score
0.01	Linear	69%	70%	63%	63%
	RBF	61%	30%	50%	38%
	Polynomial Degree 3	61%	30%	50%	38%
	Polynomial Degree 5	61%	30%	50%	38%
	Polynomial Degree 7	61%	81%	51%	39%
	Polynomial Degree 9	62%	71%	51%	40%
	Polynomial Degree 11	62%	63%	51%	42%
0.1	Linear	74%	73%	71%	72%
	RBF	68%	69%	61%	59%
	Polynomial Degree 3	61%	61%	51%	40%
	Polynomial Degree 5	61%	61%	51%	40%
	Polynomial Degree 7	61%	60%	51%	42%
	Polynomial Degree 9	62%	68%	52%	43%
	Polynomial Degree 11	62%	68%	52%	43%

Nilai Hyperparameter C	Kernel	Akurasi	Presisi	Recall	F1-Score
0.1	Polynomial Degree 11	62%	64%	52%	43%
1	Linear	78%	76%	77%	77%
	RBF	71%	69%	68%	69%
	Polynomial Degree 3	67%	68%	59%	57%
	Polynomial Degree 5	64%	68%	54%	48%
	Polynomial Degree 7	62%	68%	52%	43%
	Polynomial Degree 9	62%	68%	52%	43%
	Polynomial Degree 11	63%	70%	53%	45%
10	Linear	87%	87%	88%	87%
	RBF	76%	75%	74%	74%
	Polynomial Degree 3	69%	68%	66%	66%
	Polynomial Degree 5	66%	65%	60%	59%
	Polynomial Degree 7	64%	62%	56%	53%
	Polynomial Degree 9	62%	59%	53%	48%
	Polynomial Degree 11	62%	61%	54%	49%
100	Linear	93%	92%	93%	93%
	RBF	79%	78%	77%	78%
	Polynomial Degree 3	75%	74%	72%	73%
	Polynomial Degree 5	67%	65%	62%	62%
	Polynomial Degree 7	64%	62%	59%	58%
	Polynomial Degree 9	63%	60%	56%	55%
	Polynomial Degree 11	63%	60%	55%	52%

Tabel 4.4 menunjukkan nilai performa sistem yang mencakup akurasi, presisi, *recall*, dan *f1-score*. Dengan pengaturan *hyperparameter C* sebesar 0.01, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 69%, presisi tertinggi sebesar 70%, *recall* tertinggi sebesar 63%, dan *f1-score* tertinggi sebesar 63%. Dengan pengaturan *hyperparameter C* sebesar 0.1, SVM dengan kernel

Linear mencapai akurasi tertinggi sebesar 74%, presisi tertinggi sebesar 73%, *recall* tertinggi sebesar 71%, dan *f1-score* tertinggi sebesar 72%. Dengan pengaturan *hyperparameter* C sebesar 1, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 78%, presisi tertinggi sebesar 76%, *recall* tertinggi sebesar 77%, dan *f1-score* tertinggi sebesar 77%. Dengan pengaturan *hyperparameter* C sebesar 10, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 87%, presisi tertinggi sebesar 87%, *recall* tertinggi sebesar 88%, dan *f1-score* tertinggi sebesar 87%. Dengan pengaturan *hyperparameter* C sebesar 100, SVM dengan kernel Linear mencapai akurasi tertinggi sebesar 93%, presisi tertinggi sebesar 92%, *recall* tertinggi sebesar 93%, dan *f1-score* tertinggi sebesar 93%.

Berdasarkan tabel 4.4, terlihat bahwa ketika $C=100$, SVM dengan kernel Linear mencapai performa puncak dengan akurasi 93%, presisi 92%, *recall* 93%, dan *f1-score* 93%. Dengan demikian, berdasarkan hasil uji coba skenario B, dapat disimpulkan bahwa konfigurasi *hyperparameter* $C=100$ dan kernel Linear memberikan hasil terbaik untuk model SVM.

4.3 Pembahasan

Dalam penelitian ini, penulis menggunakan 1.319 data rekam medis pasien yang berasal dari *Elsevier Mendeley Data Repository: Heart Attack Dataset* sebagai *dataset*. Data tersebut perlu melalui tahap *preprocessing* sebelum dapat digunakan oleh sistem. Tahap *preprocessing* meliputi *Label Encoding* dan *Feature Scaling* yang bertujuan untuk mengubah data kategorikal menjadi numerik dan menormalisasi rentang nilai data. Tahap ini sangat penting untuk meningkatkan performa sistem dalam mengolah data. Setelah data siap, penulis membagi data

menjadi data latih dan data uji dengan proporsi yang berbeda sesuai dengan skenario yang ditetapkan. Penulis kemudian melatih model *Support Vector Machine* (SVM) dengan data latih untuk mengklasifikasikan pasien yang berisiko mengalami serangan jantung. Penulis juga melakukan uji coba dengan mengubah nilai *hyperparameter C* dan jenis kernel SVM untuk mencari parameter terbaik yang memberikan akurasi prediksi tertinggi pada data uji.

Pada skenario A, *dataset* dibagi menjadi data latih (85%) dan data uji (15%). Model SVM dengan kernel Linear dan nilai *hyperparameter C*=100 memiliki performa terbaik dibandingkan model SVM lainnya dengan kernel dan nilai C yang berbeda. Alasan mengapa kernel Linear cocok untuk kasus ini adalah karena data kelas serangan jantung dan non-serangan jantung dapat dipisahkan secara linear. Oleh karena itu, *hyperplane* linear dapat membagi kedua kelas dengan baik. Selain itu, nilai C yang besar membuat model SVM lebih peka terhadap kesalahan klasifikasi, sehingga batas keputusan yang dihasilkan lebih tepat. Model SVM dengan kernel Linear dan nilai C=100 dapat mendeteksi serangan jantung pada data uji skenario A dengan akurasi 92%, presisi 91%, *recall* 92%, dan *f1-score* 92%.

Pada skenario B, *dataset* dibagi menjadi dua bagian: data latih (80%) dan data uji (20%). Skenario ini bertujuan untuk mengetahui pengaruh ukuran data uji terhadap kemampuan model SVM dalam mendeteksi serangan jantung. Hasilnya, model SVM yang menggunakan kernel Linear dan nilai C=100 adalah model terbaik di antara model SVM lainnya dengan kernel dan nilai C yang berbeda. Model terbaik ini dapat mendeteksi pasien serangan jantung dengan akurasi 92%, presisi 91%, *recall* 92%, dan *f1-score* 92%. Ini berarti model SVM dengan

parameter tersebut sangat akurat dan tepat dalam mengenali pasien serangan jantung dan non serangan jantung. Hasil ini juga menunjukkan bahwa model SVM dengan parameter terbaik tetap stabil meskipun ukuran data uji ditingkatkan menjadi 20%.

Pada skenario C, penulis menggunakan 75% data untuk melatih model SVM dan 25% data untuk menguji performanya. Skenario ini bertujuan untuk melihat apakah proporsi data uji yang lebih besar akan mempengaruhi hasil model SVM. Hasilnya menunjukkan bahwa model SVM yang paling baik adalah yang menggunakan kernel Linear dan nilai $C=100$. Model ini dapat mengklasifikasikan data uji dengan akurasi 92%, presisi 92%, *recall* 92%, dan *f1-score* 92%. Nilai-nilai ini menunjukkan bahwa model SVM sangat akurat dan tepat dalam membedakan pasien serangan jantung dan non serangan jantung. Hal ini menunjukkan bahwa model SVM telah belajar dengan baik dari data latih dan dapat diterapkan pada data uji. Selain itu, hasil model SVM pada skenario C tidak berbeda jauh dengan hasil model SVM pada skenario A dan B, yang menggunakan proporsi data uji yang lebih kecil. Hal ini menunjukkan bahwa model SVM stabil dan tidak sensitif terhadap perubahan proporsi data uji.

Pada skenario D, penulis menggunakan 70% data untuk melatih model SVM dan 30% data untuk menguji performanya. Model SVM yang paling baik adalah yang menggunakan kernel Linear dan nilai $C=100$. Model ini dapat mengklasifikasikan data uji dengan akurasi 93%, presisi 92%, *recall* 93%, dan *f1-score* 93%. Nilai-nilai ini menunjukkan bahwa model SVM sangat akurat dan tepat dalam membedakan pasien serangan jantung dan non-serangan jantung. Hal ini

menunjukkan bahwa model SVM telah belajar dengan baik dari data latih dan tidak terpengaruh oleh ukuran data uji yang lebih besar.

Dari empat skenario yang telah diuji, penulis dapat menyimpulkan bahwa model SVM dengan kernel Linear dan nilai $C=100$ adalah model terbaik untuk mendeteksi serangan jantung pada *dataset* ini. Model ini selalu memiliki performa prediksi yang tinggi, dengan akurasi, presisi, *recall*, dan *f1-score* lebih dari 90% pada setiap proporsi data uji yang digunakan.

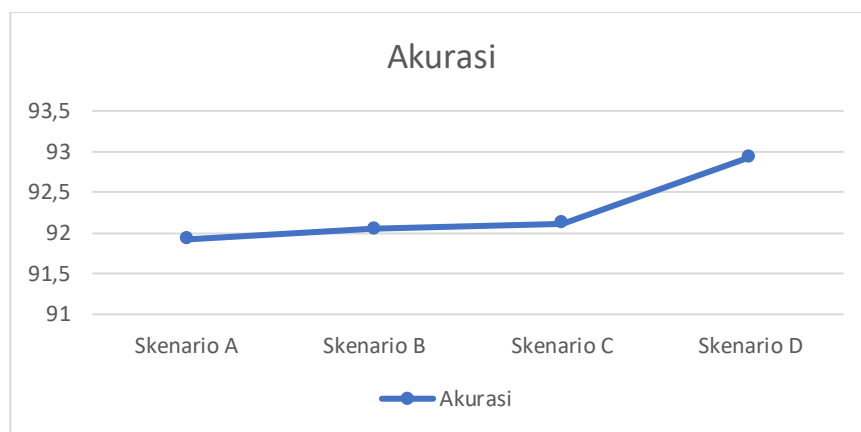
Nilai C adalah parameter regularisasi pada model SVM yang menentukan lebar margin *hyperplane* dan batas kesalahan yang diperbolehkan. Nilai C yang besar menghasilkan margin yang sempit, sehingga model lebih sesuai dengan data latih dan memiliki performa yang tinggi, namun rentan terhadap *overfitting*. Nilai C yang kecil menghasilkan margin yang luas, sehingga model lebih generalisasi dan toleran terhadap kesalahan, namun kurang sesuai dengan data latih. Pada kasus ini, nilai $C=100$ adalah nilai optimal yang membuat model dapat meminimalkan kesalahan pada data latih dan uji dengan baik.

Jenis kernel adalah fungsi yang mengubah data ke ruang fitur dengan dimensi yang lebih tinggi. Pada penelitian ini, kernel Linear adalah pilihan yang tepat dan lebih sesuai untuk *dataset* yang digunakan daripada jenis kernel lainnya, karena *dataset* dapat dipisahkan secara linear (Mohd Hatta et al., 2020).

Rasio data latih-uji berguna untuk menguji konsistensi performa model pada beragam proporsi data latih dan uji. Performa model yang konsisten meski proporsi data uji ditingkatkan mengindikasikan model terlatih dengan baik sehingga mampu menggeneralisasi dengan akurat pada data baru. Pada kasus ini, performa prediksi

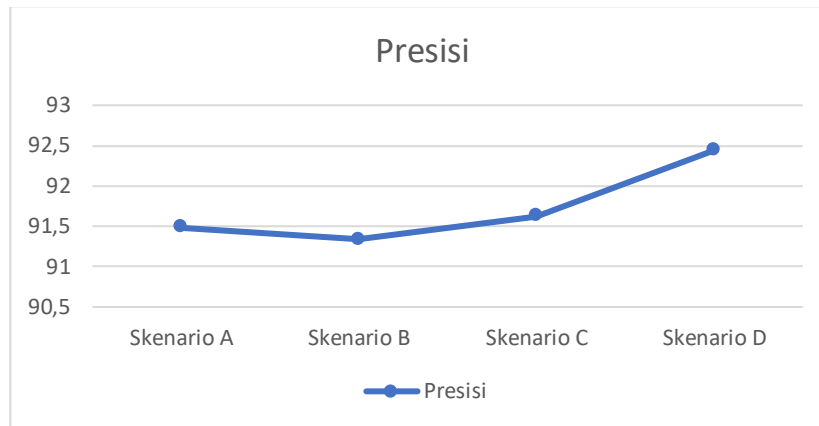
model SVM dengan parameter terbaik cukup stabil pada rentang proporsi data uji 30%, yang menunjukkan model cukup *robust* dan tidak *overfitting*.

Jadi, kombinasi $C=100$, kernel Linear, dan rasio data latih-uji 70:30 adalah konfigurasi parameter terbaik untuk SVM pada kasus deteksi serangan jantung ini, karena menghasilkan performa prediksi yang sangat akurat dan konsisten pada data uji.



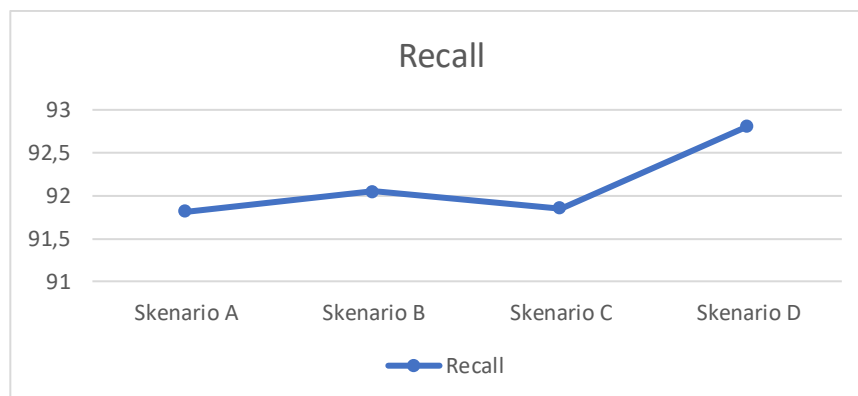
Gambar 4.5 Grafik Perbandingan Nilai Akurasi Setiap Skenario

Berdasarkan hasil uji coba, skenario A memiliki nilai akurasi sebesar 91.92%, skenario B memiliki nilai akurasi sebesar 92.05%, skenario C memiliki nilai akurasi sebesar 92.12%, dan skenario D memiliki nilai akurasi sebesar 92.93%. Akurasi diukur sebagai persentase prediksi yang tepat dari total data pasien, memberikan gambaran yang dapat diandalkan terkait dengan kemungkinan menderita atau tidak menderita serangan jantung.



Gambar 4.6 Grafik Perbandingan Nilai Presisi Setiap Skenario

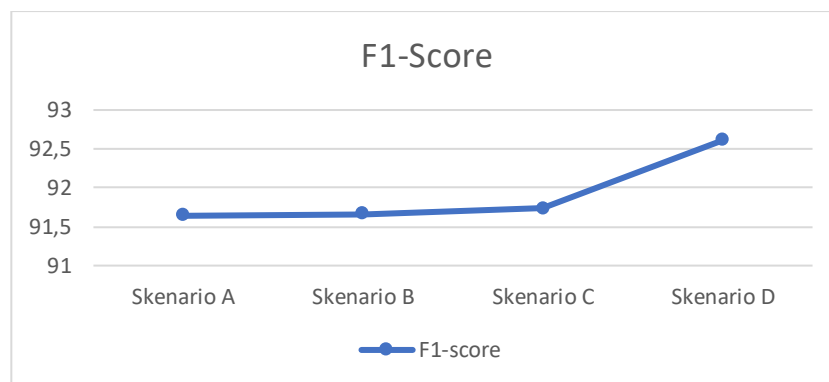
Hasil pengujian pada skenario A mengungkapkan bahwa presisi mencapai 91,49%. Skenario B menunjukkan hasil presisi sebesar 91,34%, sementara skenario C mencapai presisi 91,62%. Pada skenario D, hasil pengujian menunjukkan presisi sebesar 92,44%. Presisi dalam konteks ini menggambarkan presentase pasien yang secara akurat dideteksi menderita serangan jantung dari total pasien yang diprediksi menderita kondisi tersebut.



Gambar 4.7 Grafik Perbandingan Nilai Recall Setiap Skenario

Hasil pengujian pada skenario A menunjukkan tingkat *recall* sebesar 91,81%. Pada skenario B, nilai *recall* yang dicapai adalah 92,05%, sementara skenario C

mencapai 91,85%. Sementara itu, skenario D menunjukkan kinerja tertinggi dengan nilai *recall* sebesar 92,81%. *Recall* diukur sebagai persentase pasien yang secara tepat diprediksi mengalami serangan jantung dibandingkan dengan total pasien yang sebenarnya menderita kondisi tersebut.



Gambar 4.8 Grafik Perbandingan Nilai F1-score Setiap Skenario

Analisis hasil uji coba pada empat skenario menunjukkan performa model yang beragam. *f1-score* untuk skenario A mencapai 91.64%, skenario B mencapai 91.66%, skenario C mencapai 91.73%, dan skenario D mencapai 92.61%. Diperoleh informasi bahwa *f1-score* mengindikasikan keseimbangan yang baik antara kemampuan model dalam mengidentifikasi kasus positif (pasien yang menderita serangan jantung) dan menghindari kesalahan dalam memprediksi kasus positif (memprediksi serangan jantung padahal tidak terjadi).

Tingkat akurasi yang tinggi menunjukkan kemampuan metode SVM dalam mendeteksi serangan jantung secara handal berdasarkan pola dan kecenderungan pada data rekam medis. Model SVM terlatih dengan baik sehingga dapat memisahkan kedua kelas data uji (serangan jantung dan non-serangan jantung) tanpa kesalahan klasifikasi yang signifikan. Hasil pengujian menunjukkan potensi

dan manfaat metode SVM dalam mendukung diagnosis dini serangan jantung melalui analisis otomatis data rekam medis pasien dengan menggunakan teknik pembelajaran mesin. Dengan tingkat akurasi yang tinggi, model SVM yang dikembangkan dapat menjadi alat yang sangat berguna bagi dokter dan tenaga medis dalam skrining awal pasien yang berisiko mengalami serangan jantung di masa depan.

Sistem deteksi serangan jantung ini merupakan hasil dari penelitian yang bertujuan mengidentifikasi dan mendeteksi serangan jantung pada pasien yang berisiko tinggi. Sistem ini dapat memberitahu pasien tentang potensi risiko serangan jantung sehingga mereka dapat segera mendapatkan penanganan yang tepat. Sistem ini memiliki nilai penting dalam konteks medis karena dapat mengurangi risiko kesalahan diagnosis dini secara manual yang mungkin dilakukan oleh tenaga medis terhadap pasien yang rentan terhadap serangan jantung sebelum melakukan tes laboratorium lebih lanjut. Meskipun sistem ini memerlukan peningkatan dan perbaikan lebih lanjut untuk meningkatkan kinerjanya, diharapkan dapat memberikan manfaat yang lebih besar bagi kualitas hidup manusia dengan memberikan penanganan serangan jantung yang lebih cepat.

Serangan jantung terjadi ketika arteri koroner yang mengalirkan darah ke jantung tersumbat oleh gumpalan darah atau plak, menyebabkan kurangnya oksigen dan nutrisi yang dapat merusak atau bahkan membunuh jaringan jantung. Kesehatan merupakan amanah yang harus dijaga oleh setiap individu dan masyarakat. Oleh karena itu, penting bagi manusia untuk mencegah penyakit

dengan mendeteksi gejala-gejala awal seperti serangan jantung guna memelihara kesehatan.

Dalam Islam, mencari pengobatan ketika sakit adalah hal yang penting. Hal ini juga berlaku untuk serangan jantung, yang membutuhkan perawatan medis dan usaha penyembuhan yang sesuai dengan Islam. Selain itu, Islam juga mengajarkan pentingnya memberikan dukungan sosial dan emosional kepada orang-orang yang sakit. Nilai-nilai Islam mencakup menunjukkan rasa simpati, memberikan dorongan moral, dan membantu secara praktis orang-orang yang menderita serangan jantung.

Kondisi kesehatan tubuh yang optimal tidak selalu dirasakan oleh setiap orang, ada kalanya seseorang mengalami berbagai macam penyakit, yang bisa menyerang bagian luar ataupun dalam tubuh. Berkaitan dengan hal ini, Allah SWT tidak menghendaki hamba-Nya untuk menyerah ketika sakit, tetapi Allah SWT mengharuskan dan memerintahkan kita untuk berikhtiar sebaik mungkin agar bisa sembuh dari penyakit yang kita derita. Seperti yang telah dijelaskan oleh Nabi Muhammad SAW kepada para sahabat dan umatnya, dianjurkan untuk mencari obat ketika sakit, karena tiap penyakit pasti ada obatnya, sebagaimana dalam hadist Rasulullah SAW:

حَدَّثَنَا هَارُونُ بْنُ مَعْرُوفٍ وَأَبُو الطَّاهِرِ وَأَحْمَدُ بْنُ عَيْسَى قَالُوا حَدَّثَنَا ابْنُ وَهْبٍ أَخْبَرَنِي عَمْرُو وَهُوَ ابْنُ الْحَارِثِ عَنْ عَبْدِ رَبِّهِ بْنِ سَعِيدٍ عَنْ أَبِي الزُّبَيْرِ عَنْ جَابِرٍ عَنْ رَسُولِ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ أَنَّهُ قَالَ لِكُلِّ دَاءٍ دَوَاءٌ فَإِذَا أُصِيبَ دَوَاءُ الدَّاءِ بَرَأَ بِإِذْنِ اللَّهِ عَزَّ وَجَلَّ

“Telah menceritakan kepada kami Harun bin Ma’ruf dan Abu Ath Thahir serta Ahmad bin ‘Isa mereka berkata; Telah menceritakan kepada kami Ibnu Wahb; Telah mengabarkan kepadaku ‘Amru, yaitu Ibnu al-Harits dari ‘Abdu Rabbih bin Sa’id dari Abu Az Zubair dari Jabir dari Rasulullah shallallahu ‘alaihi wasallam,

beliau bersabda: “Setiap penyakit ada obatnya. Apabila ditemukan obat yang tepat untuk suatu penyakit, akan sembuhlah penyakit itu dengan izin Allah ‘azza wajalla.” (HR. Muslim No. 4084)

Dalam salah satu kitabnya, Ibnu Qayyim menjelaskan bahwa kalimat “Setiap penyakit ada obatnya” bersifat umum. Allah SWT menciptakan obat-obatan untuk mengobati berbagai penyakit. Hadits tersebut menunjukkan bahwa Islam menganjurkan kita untuk berobat. Ibnu Qayyim juga mengatakan bahwa berobat tidak bertentangan dengan tawakkal, tetapi justru merupakan bagian dari tawakkal.

Salah satu ajaran yang terkandung dalam Al-Qur’an adalah mengenai kewajiban untuk saling tolong-menolong dan melakukan kebaikan demi kemaslahatan umat, sebagaimana yang ditegaskan dalam QS. Al-Ma’idah ayat 2. Ayat ini juga menjelaskan bahwa semua tindakan dan pemberian yang bermanfaat bagi orang lain harus dilandasi oleh niat yang baik dan rasa takut kepada Allah.

وَتَعَاوَنُوا عَلَى الْبِرِّ وَالتَّقْوَىٰ ۖ وَلَا تَعَاوَنُوا عَلَى الْإِثْمِ وَالْعُدْوَانِ ۗ وَاتَّقُوا اللَّهَ ۖ إِنَّ اللَّهَ شَدِيدُ الْعِقَابِ

“Dan tolong-menolonglah kamu dalam (mengerjakan) kebajikan dan takwa, dan jangan tolong-menolong dalam berbuat dosa dan pelanggaran. Dan bertakwalah kamu kepada Allah, sesungguhnya Allah amat berat siksa-Nya.” (QS. Al-Ma’idah: 2)

Penelitian ini dilakukan untuk menerapkan ayat yang telah disebutkan di atas, dengan cara memberikan kontribusi tentang bagaimana mendeteksi serangan jantung menggunakan metode *Support Vector Machine*, yang penulis harapkan dapat membantu menilai seseorang berpotensi mengalami serangan jantung atau tidak, juga mengevaluasi kinerja dari metode tersebut dan penelitian yang telah dilakukan dapat menyajikan informasi terkait hal-hal yang dapat memicu seseorang

menderita serangan jantung. Sistem yang telah dikembangkan dapat membantu mencegah serangan jantung lebih awal dan dapat dilakukan tindakan yang tepat. Terdapat hadits yang relevan dengan penelitian ini yaitu hadits riwayat Muslim, Dari Abu Hurairah radhiyallahu ‘anhu, Rasulullah shallallahu ‘alaihi wa sallam bersabda:

مَنْ نَفَسَ عَنْ مُؤْمِنٍ كُرْبَةً مِنْ كُرْبِ الدُّنْيَا نَفَسَ اللَّهُ عَنْهُ كُرْبَةً مِنْ كُرْبِ يَوْمِ الْقِيَامَةِ وَمَنْ يَسَّرَ عَلَى مُعْسِرٍ يَسَّرَ اللَّهُ عَلَيْهِ فِي الدُّنْيَا وَالْآخِرَةِ وَمَنْ سَتَرَ مُسْلِمًا سَتَرَهُ اللَّهُ فِي الدُّنْيَا وَالْآخِرَةِ وَاللَّهُ فِي عَوْنِ الْعَبْدِ مَا كَانَ الْعَبْدُ فِي عَوْنِ أَخِيهِ وَمَنْ سَلَكَ طَرِيقًا يَلْتَمِسُ فِيهِ عِلْمًا سَهَّلَ اللَّهُ لَهُ بِهِ طَرِيقًا إِلَى الْجَنَّةِ وَمَا اجْتَمَعَ قَوْمٌ فِي بَيْتٍ مِنْ بُيُوتِ اللَّهِ يَتْلُونَ كِتَابَ اللَّهِ وَيَتَدَارَسُونَهُ بَيْنَهُمْ إِلَّا نَزَلَتْ عَلَيْهِمُ السَّكِينَةُ وَغَشِيَتْهُمُ الرَّحْمَةُ وَحَفَّتْهُمُ الْمَلَائِكَةُ وَذَكَرَهُمُ اللَّهُ فِيمَنْ عِنْدَهُ وَمَنْ بَطَأَ بِهِ عَمَلُهُ لَمْ يُسْرَعِ بِهِ نَسَبُهُ

“Barangsiapa membebaskan seorang mukmin dari suatu kesulitan dunia, maka Allah akan membebaskannya dari suatu kesulitan pada hari kiamat. Barangsiapa memberi kemudahan kepada orang yang berada dalam kesulitan, maka Allah akan memberikan kemudahan di dunia dan akhirat. Barangsiapa menutupi aib seorang muslim, maka Allah akan menutup aibnya di dunia dan akhirat. Allah akan selalu menolong hamba-Nya selama hamba tersebut menolong saudaranya sesama muslim. Barangsiapa menempuh jalan untuk mencari ilmu, maka Allah akan memudahkan jalan ke surga baginya. Tidaklah sekelompok orang berkumpul di suatu masjid (rumah Allah) untuk membaca al-Qur'an, melainkan mereka akan diliputi ketenangan, rahmat, dan dikelilingi para malaikat, serta Allah akan menyebut-nyebut mereka pada malaikat-malaikat yang berada di sisi-Nya. Barangsiapa yang ketinggalan amalnya, maka nasabnya tidak juga meninggikannya.” (Hadits Riwayat Muslim, Shahîh Muslim, juz VIII, hal. 71, hadits no. 7028, dari Abu Hurairah r.a.)

Memberikan manfaat dan meringankan beban sesama adalah hal yang sangat dianjurkan dalam hadits ini. Umat Islam dituntut untuk senantiasa melakukan kebaikan dan bermanfaat bagi orang lain, baik dalam urusan kesehatan maupun lainnya. Salah satu contohnya adalah dengan mengedukasi masyarakat tentang pencegahan dan penanganan serangan jantung. Hal ini dapat dilakukan melalui

penerapan sistem deteksi serangan jantung menggunakan metode *Support Vector Machine*. Salah satu nilai yang diajarkan oleh Al-Qur'an adalah kewajiban seorang muslim untuk menginfakkan sebagian dari harta yang dimilikinya di jalan Allah SWT. Hal ini ditegaskan dalam surah Al-Baqarah ayat 195 yang berbunyi:

وَأَنْفِقُوا فِي سَبِيلِ اللَّهِ وَلَا تُلْقُوا بِأَيْدِيكُمْ إِلَى التَّهْلُكَةِ . وَأَحْسِنُوا . إِنَّ اللَّهَ يُحِبُّ الْمُحْسِنِينَ

“Dan infakkanlah (hartamu) di jalan Allah, dan janganlah kamu jatuhkan (diri sendiri) ke dalam kebinasaan dengan tangan sendiri, dan berbuatbaiklah. Sungguh, Allah menyukai orang-orang yang berbuat baik.” (QS. Al-Baqarah: 195)

Pengembangan sistem deteksi serangan jantung menggunakan metode *Support Vector Machine* dalam penelitian ini merupakan salah satu bentuk kontribusi dalam bidang ilmu yang bermanfaat bagi kesehatan dan keselamatan manusia.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Bab sebelumnya telah menjelaskan tentang pengujian yang dilakukan untuk mengetahui performa metode *Support Vector Machine* dalam mendeteksi serangan jantung. Proses pengujian ini melibatkan beberapa langkah penting. Langkah pertama adalah *preprocessing data*, yang mencakup *Label Encoding* menggunakan *LabelEncoder* dan *Feature Scaling* menggunakan *StandardScaler*. Setelah itu, data dibagi menjadi empat bagian dengan rasio yang berbeda, yaitu 85% data latih : 15% data uji, 80% data latih : 20% data uji, 75% data latih : 25% data uji, dan 70% data latih : 30% data uji. Langkah berikutnya adalah membandingkan tiga kernel *Support Vector Machine* dan lima nilai *hyperparameter C* yang berbeda. Hasil pengujian menunjukkan bahwa Model D, yang menggunakan kernel Linear dengan nilai *hyperparameter C*=100 dan rasio data latih 70% dan data uji 30%, mencapai akurasi tertinggi sebesar 93%. Model ini berhasil mendeteksi 368 dari 396 data uji dengan benar. Namun, Model D juga mengalami beberapa kesalahan deteksi. Terdapat 16 data pasien yang sebenarnya mengalami serangan jantung tetapi salah diklasifikasikan sebagai tidak mengalami serangan jantung, dan 12 data pasien yang sebenarnya tidak mengalami serangan jantung tetapi salah diklasifikasikan sebagai mengalami serangan jantung. Berdasarkan hasil akurasi yang diperoleh, dapat disimpulkan bahwa Model D memiliki kemampuan yang sangat baik dalam mendeteksi serangan jantung berdasarkan data target dan data prediksi.

5.2 Saran

Setelah melakukan analisis terhadap hasil pengujian, penulis menyadari bahwa penelitian ini masih memiliki kekurangan dan memerlukan penyempurnaan lebih lanjut. Sehubungan dengan hal tersebut, penulis mengajukan beberapa rekomendasi yang dapat dipertimbangkan untuk penelitian di masa mendatang:

1. Menerapkan dan menguji metode SVM terhadap *dataset* jenis penyakit jantung lainnya, seperti gagal jantung, penyakit jantung koroner, dan aritmia.
2. Menerapkan teknik *ensemble* atau kombinasi beberapa metode *machine learning* untuk meningkatkan performa sistem.
3. Menerapkan teknik seleksi fitur pada *dataset* yang bertujuan untuk memilih fitur-fitur yang paling berpengaruh dan relevan terhadap variabel target atau output yang ingin diprediksi.

DAFTAR PUSTAKA

- Abu Bakar, N. A. F., Ahmad, A., Wan Musa, W. Z., Shahril, M. R., Wan-Arfah, N., Abdul Majid, H., Piernas, C., Ramli, A. W., & Naing, N. N. (2022). Association Between a Dietary Pattern High in Saturated Fatty Acids, Dietary Energy Density, and Sodium With Coronary Heart Disease. *Scientific Reports*, *12*(1), 1–11. <https://doi.org/10.1038/s41598-022-17388-5>
- Afandi, W., Laksana, T. G., & Tanjung, N. A. F. (2023). Penerapan Metode Support Vector Machine Analisis Sentimen Tweet Pergantian Logo Halal di Indonesia. *Jurnal Elektronika Dan Komputer*, *16*(1), 45–52. <https://journal.stekom.ac.id/index.php/elkom/article/view/964%0A>
<https://doi.org/10.51903/elkom.v16i1.964>
- Ahmed, A. A. A., Al-Shami, A. M., Jamshed, S., Nahas, A. R. F., & Ibrahim, M. I. M. (2020). Public Awareness of and Action Towards Heart Attack Symptoms: an Exploratory Study. *International Journal of Environmental Research and Public Health*, *17*(23), 1–16. <https://doi.org/10.3390/ijerph17238982>
- Alfaidi, A., Aljuhani, R., Alshehri, B., Alwadei, H., & Sabbeh, S. (2022). Machine Learning: Assisted Cardiovascular Diseases Diagnosis. *International Journal of Advanced Computer Science and Applications*, *13*(2), 135–141. <https://doi.org/10.14569/IJACSA.2022.0130216>
- Aloyses, E., Bhaskar, C. V., Priyatharsini, G. S., & V, R. B. (2022). Heart Attack Prediction Using SMLT. *International Journal of Pharmaceutical Sciences Review and Research*, *73*(35), 197–202. <https://doi.org/10.47583/ijpsr.2022.v73i02.035>
- Alpert, J. S. (2023). New Coronary Heart Disease Risk Factors. *American Journal of Medicine*, *136*(4), 331–332. <https://doi.org/10.1016/j.amjmed.2022.08.002>
- Bashit, N., Prasetyo, Y., & Suprayogi, A. (2019). Klasifikasi Berbasis Objek untuk Pemetaan Penggunaan Lahan menggunakan Citra SPOT 5 di Kecamatan Ngaglik. *Teknik*, *40*(2), 122. <https://doi.org/10.14710/teknik.v39i3.23050>
- Breuckmann, F., & Rassaf, T. (2020). Early Heart Attack Care as a Prehospital Programme Designed to Supplement the Established Chest Pain Unit Concept in Germany. *European Heart Journal: Acute Cardiovascular Care*, *9*(1), 93–94. <https://doi.org/10.1177/2048872618759316>
- Chowdhury, M. N. R., Ahmed, E., Siddik, M. A. D., & Zaman, A. U. (2021). Heart Disease Prognosis Using Machine Learning Classification Techniques. *2021 6th International Conference for Convergence in Technology, I2CT 2021*, 1–6. <https://doi.org/10.1109/I2CT51068.2021.9418181>
- Furqan, M., Rajput, H., Narejo, S., Ashraf, A., & Awan, K. (2020). Heart Disease

Prediction using Machine Learning Algorithms. *International Conference on Computational Sciences and Technologies*, 65–70.
<https://doi.org/10.1109/SMARTGENCON56628.2022.10083907>

- Garg, A., Sharma, B., & Khan, R. (2021). Heart Disease Prediction Using Machine Learning Techniques. *IOP Conference Series: Materials Science and Engineering*, 1–9. <https://doi.org/10.1088/1757-899X/1022/1/012046>
- Helling, L. S., Wahyudi, E., & Hasanudin. (2019). Siremis: Sistem Informasi Rekam Medis Puskesmas Kecamatan Matraman Jakarta. *INTENSIF: Jurnal Ilmiah Penelitian Dan Penerapan Teknologi Sistem Informasi*, 3(2), 116–129. <https://doi.org/10.29407/intensif.v3i2.12597>
- Mohd Hatta, N., Ali Shah, Z., & Kasim, S. (2020). Evaluate the Performance of SVM Kernel Functions for Multiclass Cancer Classification. *International Journal of Data Science*, 1(1), 37–41. <https://doi.org/10.18517/ijods.1.1.37-41.2020>
- Muhammad Fauzan A, Hedian Nura, M., Mayonva, H., Yursal, M. D., Asnur, L., & Khairul Kamal, F. (2022). (Boring Sen) Cherry Leaf Dried Sponge Is Beneficial in Maintaining Cholesterol Levels and High Blood Pressure. *Spectrum*, 1(01), 36–47. <https://doi.org/10.54482/spectrum.v1i01.99>
- Ordila, R., Wahyuni, R., Irawan, Y., & Yulia Sari, M. (2020). Penerapan Data Mining Untuk Pengelompokan Data Rekam Medis Pasien Berdasarkan Jenis Penyakit Dengan Algoritma Clustering (Studi Kasus : Poli Klinik PT.Inecda). *Jurnal Ilmu Komputer*, 9(2), 148–153. <https://doi.org/10.33060/jik/2020/vol9.iss2.181>
- Piccialli, V., & Sciandrone, M. (2022). Nonlinear Optimization and Support Vector Machines. *Annals of Operations Research*, 15–47. <https://doi.org/10.1007/s10479-022-04655-x>
- Pradana, M. G., Wijaya, D. P., & Saputro, P. H. (2022). Komparasi Metode Support Vector Machine Dan Naïve Bayes Dalam Klasifikasi Peluang Penyakit Serangan Jantung. *Indonesian Journal of Business Intelligence*, 5(2), 87–91. <https://doi.org/10.21927/ijubi.v5i2.2659>
- Rahayu, D., Irawan, H., Santoso, P., Susilowati, E., Atmojo, D. S., & Kristanto, H. (2021). Deteksi Dini Penyakit Tidak Menular Pada Lansia. *Jurnal Peduli Masyarakat*, 3(1), 91–96. <https://doi.org/10.37287/jpm.v3i1.449>
- Rohayati, & Widani, N. L. (2022). Analisis Faktor yang Berhubungan dengan Serangan Jantung Berulang pada Pasien Post Percutaneous Transluminal Coronary Angioplasty. *Jurnal Kesehatan STIKES Bethesda Yakkum Yogyakarta*, 10(1), 25–39. <https://doi.org/10.35913/jk.v10i1.247>
- Sajja, G. S., Mustafa, M., Phasinam, K., Kaliyaperumal, K., Ventayen, R. J. M., & Kassaruk, T. (2021). Towards Application of Machine Learning in Classification and Prediction of Heart Disease. *Proceedings of the 2nd*

International Conference on Electronics and Sustainable Communication Systems, ICESC 2021, 1664–1669.
<https://doi.org/10.1109/ICESC51422.2021.9532940>

Shanbhag, A. A., Shetty, C., Ananth, A., Shetty, A. S., Kavanashree Nayak, K., & Rakshitha, B. R. (2021). Heart Attack Probability Analysis Using Machine Learning. *2021 IEEE International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics, DISCOVER 2021 - Proceedings*, 301–306. <https://doi.org/10.1109/DISCOVER52564.2021.9663631>

Tamlander, M., Mars, N., Pirinen, M., Widén, E., & Ripatti, S. (2022). Integration of Questionnaire-based Risk Factors Improves Polygenic Risk Scores for Human Coronary Heart Disease and Type 2 Diabetes. *Communications Biology*, 1-13. <https://doi.org/10.1038/s42003-021-02996-0>