PERBANDINGAN ALGORITMA MACHINE LEARNING TRADISIONAL UNTUK PREDIKSI SERANGAN JANTUNG DI INDONESIA DENGAN ONE-HOT ENCODING DAN ORDINAL ENCODING

TUGAS AKHIR

Diajukan sebagai syarat menyelesaikan jenjang strata Satu (S-1) di Program Studi Teknik Informatika, Fakultas Teknologi Industri, Institut Teknologi Sumatera

Oleh:

Ahmad Zain Mahmud

121140232

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNOLOGI INDUSTRI

INSTITUT TEKNOLOGI SUMATERA

LAMPUNG SELATAN

2025

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

DAFTAR RUMUS

BAB I
PENDAHULUAN

1.1 Latar Belakang

Penyakit kardiovaskular, khususnya serangan jantung, merupakan penyebab utama kematian di dunia, termasuk di Indonesia. Berdasarkan data WHO (2021), lebih dari 17 juta orang meninggal setiap tahun akibat penyakit ini. Di Indonesia, prevalensi penyakit jantung semakin meningkat seiring dengan perubahan gaya hidup, pola makan, serta faktor risiko lain seperti hipertensi, diabetes, dan merokok. Deteksi dini terhadap risiko serangan jantung menjadi sangat penting untuk menurunkan angka kematian dan meningkatkan kualitas hidup masyarakat.

Perkembangan teknologi informasi, khususnya dalam bidang machine learning, membuka peluang besar untuk melakukan prediksi penyakit dengan akurasi tinggi. Berbagai algoritma seperti Random Forest, Support Vector Machine (SVM), Gradient Boosting, dan XGBoost telah banyak digunakan untuk prediksi dalam bidang kesehatan karena kemampuannya dalam menangani data yang kompleks dan nonlinear seperti pada tabel 2.1. Namun, setiap algoritma memiliki karakteristik dan performa yang berbeda tergantung pada jenis data dan teknik praproses yang digunakan.

Dalam proses preprocessing, teknik encoding seperti One-Hot Encoding dan Ordinal Encoding juga memegang peranan penting, terutama dalam mengubah data kategorikal menjadi numerik agar dapat diproses oleh algoritma. Penelitian oleh Fitria, L. dan Nugroho, T. [1] menunjukkan bahwa metode encoding yang tepat dapat meningkatkan akurasi model secara signifikan.

Dengan adanya dataset Heart Attack Prediction in Indonesia dari Kaggle, penelitian ini bertujuan untuk melakukan analisis komparatif terhadap empat algoritma klasifikasi dalam memprediksi risiko serangan jantung, sekaligus mengevaluasi metode encoding mana yang paling optimal serta mengidentifikasi fitur-fitur yang paling berpengaruh dalam proses prediksi.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, rumusan masalah dalam penelitian ini adalah:

1. Algoritma mana di antara Random Forest, SVM, Gradient Boosting, dan XGBoost yang memberikan performa terbaik dalam memprediksi serangan jantung di Indonesia?
2. Metode encoding mana yang memberikan hasil lebih optimal dalam meningkatkan performa model?
3. Apa saja variabel atau fitur yang paling berpengaruh dalam memprediksi risiko serangan jantung berdasarkan hasil model?

1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Menganalisis dan membandingkan performa empat algoritma klasifikasi (Random Forest, SVM, Gradient Boosting, dan XGBoost) dalam prediksi serangan jantung.
2. Membandingkan hasil prediksi antara penggunaan metode One-Hot Encoding dan Ordinal Encoding.
3. Mengidentifikasi variabel paling berpengaruh dalam prediksi risiko serangan jantung.

1.4 Batasan Masalah

Penelitian ini dibatasi pada hal-hal berikut:

1. Dataset yang digunakan adalah Heart Attack Prediction in Indonesia dari Kaggle dengan 10.000 data setelah di-shuffle.
2. Algoritma yang dibandingkan adalah *Random Forest, SVM, Gradient Boosting, dan XGBoost*.
3. Teknik encoding yang digunakan hanya One-Hot Encoding dan Ordinal Encoding.
4. Penilaian performa model dilakukan berdasarkan metrik evaluasi seperti akurasi, precision, recall, dan F1-score.

5. Penelitian ini dilakukan menggunakan bahasa pemrograman Python dengan library scikit-learn dan XGBoost.

1.5 Manfaat Penelitian

Adapun manfaat yang diperoleh dari hasil penelitian ini adalah sebagai berikut:

1. Memberikan referensi bagi praktisi data dan peneliti dalam memilih algoritma terbaik untuk prediksi serangan jantung.
2. Memberikan wawasan mengenai pengaruh teknik encoding terhadap performa model klasifikasi.
3. Meningkatkan kesadaran akan pentingnya pemanfaatan teknologi informasi dalam bidang kesehatan, khususnya dalam upaya deteksi dini penyakit kardiovaskular.

1.6 Sistematika Penulisan

Sistematika penulisan skripsi ini dibagi menjadi beberapa bab sebagai berikut:

1.6.1 Bab I Pendahuluan

Bab ini berisi latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, dan sistematika penulisan.

1.6.2 Bab II Tinjauan Pustaka

Bab ini menjelaskan hasil kajian pustaka terkait topik yang diteliti, serta teori-teori pendukung seperti algoritma machine learning, teknik encoding, dan metrik evaluasi model.

1.6.3 Bab III Metodologi Penelitian

Bab ini membahas metode penelitian yang digunakan, termasuk desain penelitian, alur penelitian, langkah-langkah implementasi, teknik praproses data, serta metode pengujian performa model.

1.6.4 Bab IV Hasil dan Pembahasan

Bab ini menyajikan hasil pengolahan data, evaluasi performa masing-masing model, analisis terhadap hasil yang diperoleh, serta interpretasi terhadap pentingnya fitur dalam prediksi.

1.6.5 Bab V Kesimpulan dan Saran

Bab ini berisi kesimpulan dari hasil penelitian serta saran untuk penelitian selanjutnya yang berkaitan dengan pengembangan model prediksi serangan jantung atau penerapan di bidang lain.

BAB II
TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Penelitian mengenai penerapan algoritma machine learning untuk prediksi penyakit, khususnya serangan jantung, telah banyak dilakukan dalam beberapa tahun terakhir. Namun, masing-masing studi memiliki pendekatan yang berbeda, baik dari sisi algoritma yang digunakan, metode praproses data, hingga fokus analisis terhadap variabel yang berpengaruh.

Tabel 2.1 Literasi Penelitian Terdahulu

No	Penulis [Tahun] [Judul]	Permasalahan	Metode	Hasil
1.	Fitria & Nugroho [2021] [Pengaruh Metode Encoding pada Prediksi Diabetes]	Pengaruh metode encoding belum banyak diteliti	One-Hot, Ordinal Encoding, Logistic Regression	Encoding memengaruhi akurasi hingga 5%
2.	Rafiqi et al. [2023] [Prediksi Serangan Jantung dengan Random Forest, SVM, dan KNN]	Akurasi rendah dan ketidakseimbangan data	SMOTE, Random Forest, SVM, KNN	Random Forest paling akurat, fitur penting belum dianalisis
3.	Kusuma et al. [2020] [Prediksi Penyakit Jantung dengan SVM dan Grid Search]	Tidak membandingkan dengan algoritma lain	SVM dengan optimasi parameter (grid search)	Akurasi 88%, encoding terbatas

No	Penulis [Tahun] [Judul]	Permasalahan	Metode	Hasil
4.	Dewi & Wulandari [2021] [Deteksi Penyakit Jantung Menggunakan Gradient Boosting]	Dataset lokal kecil, fitur tidak dijelaskan dengan baik	Gradient Boosting	Akurasi 85%, namun validasi terbatas
5.	Putra & Syafitri [2022] [Komparasi XGBoost dan Naïve Bayes untuk Prediksi Jantung]	Kurangnya analisis terhadap metode encoding	XGBoost, Naïve Bayes	XGBoost unggul dalam akurasi dan interpretabilitas

Berdasarkan tinjauan terhadap penelitian terdahulu, dapat disimpulkan bahwa berbagai studi telah dilakukan untuk meningkatkan akurasi prediksi penyakit jantung menggunakan beragam algoritma seperti Random Forest, Support Vector Machine (SVM), Gradient Boosting, XGBoost, hingga Naïve Bayes. Beberapa penelitian seperti yang dilakukan oleh Rafiqi et al. [2] dan Kusuma et al. [3] menyoroti pentingnya penanganan data tidak seimbang dan optimasi parameter, sementara Dewi dan Wulandari [4] serta Putra dan Syafitri [5] menekankan pada penggunaan algoritma ensemble dan analisis fitur penting. Di sisi lain, Fitria dan Nugroho [1] menunjukkan bahwa pemilihan metode encoding data dapat berdampak signifikan terhadap performa model prediksi penyakit kronis. Namun demikian, belum ditemukan studi yang secara menyeluruh membandingkan pengaruh metode encoding (seperti One-Hot dan Ordinal Encoding) terhadap performa masing-masing algoritma prediktif dalam konteks penyakit jantung, terutama dengan data lokal di Indonesia. Selain itu, kombinasi evaluasi performa model dan identifikasi fitur paling berpengaruh juga masih jarang dijadikan fokus utama secara bersamaan.

Oleh karena itu, penelitian ini akan mengisi celah tersebut dengan melakukan analisis komparatif beberapa algoritma klasifikasi (Random Forest, SVM, Gradient Boost, dan XGBoost) menggunakan dua pendekatan encoding data, serta

mengevaluasi variabel yang paling memengaruhi hasil prediksi serangan jantung. Penelitian ini diharapkan dapat memberikan kontribusi nyata dalam pengembangan sistem pendukung keputusan di bidang kesehatan berbasis data Indonesia.

2.2 Dasar Teori

Pada bagian ini akan dijelaskan teori-teori dan konsep yang digunakan dalam penelitian ini, yaitu mengenai algoritma klasifikasi, teknik encoding, serta metrik evaluasi model.

2.2.1 Machine Learning

Machine Learning adalah cabang dari kecerdasan buatan (AI) yang memungkinkan sistem belajar dari data dan membuat prediksi atau keputusan tanpa diprogram secara eksplisit. Dalam konteks klasifikasi medis, algoritma supervised learning banyak digunakan karena menyediakan label yang jelas (positif/negatif penyakit).

2.2.1.1 Supervised Learning

Supervised learning merupakan salah satu jenis machine learning di mana algoritma dilatih menggunakan dataset yang sudah diberi label. Tujuan utamanya adalah mempelajari hubungan antara fitur (input) dan label (output) sehingga model dapat memprediksi output untuk data baru. Contoh kasus supervised learning adalah klasifikasi, seperti memprediksi apakah seseorang berisiko terkena serangan jantung atau tidak.

2.2.2 Algoritma Klasifikasi

Berikut adalah empat algoritma utama yang digunakan dalam penelitian ini:

2.2.2.1 Random Forest

Random Forest adalah algoritma ensemble learning yang menggabungkan banyak pohon keputusan (decision tree) untuk menghasilkan prediksi yang lebih akurat dan stabil. Setiap pohon dilatih menggunakan subset acak dari data (bagging),

dan hasil prediksi ditentukan berdasarkan voting mayoritas. Kelebihan Random Forest adalah mampu menangani data yang besar dan kompleks serta mengurangi risiko overfitting.

2.2.2.2 Support Vector Machine (SVM)

Algoritma ini mencari hyperplane terbaik yang memisahkan kelas-kelas dalam data. SVM efektif pada data berdimensi tinggi dan dapat digunakan dengan kernel untuk menangani data non-linear. Namun, SVM bisa sensitif terhadap skala data dan parameter tuning.

2.2.2.3 Gradient Boosting

Gradient Boosting adalah metode boosting yang membangun model secara bertahap, di mana setiap model baru memperbaiki kesalahan dari model sebelumnya. Algoritma ini fokus pada data yang salah diklasifikasikan dan memperkuat pembelajaran terhadap data tersebut. Gradient Boosting dikenal memiliki performa tinggi namun membutuhkan tuning parameter yang tepat agar tidak overfitting.

2.2.2.4 XGBoost

Extreme Gradient Boosting (XGBoost) adalah pengembangan dari Gradient Boosting yang lebih cepat dan efisien. XGBoost menggunakan regularisasi L1 dan L2 untuk menghindari overfitting serta mendukung paralelisasi saat pelatihan. Karena efisiensi dan akurasinya, XGBoost sering digunakan dalam kompetisi data science.

2.2.3 Preprocessing Data

Preprocessing adalah tahap awal yang bertujuan untuk menyiapkan data mentah menjadi format yang bisa digunakan oleh algoritma. Berikut merupakan teknik preprocessing yang diterapkan pada dataset:

2.2.3.1 Standard Scaler

Standard Scaler digunakan untuk menstandarkan fitur numerik agar memiliki rata-rata 0 dan standar deviasi 1. Ini penting agar semua fitur memiliki kontribusi yang seimbang dalam proses pembelajaran model machine learning, terutama pada algoritma yang sensitif terhadap skala seperti SVM dan KNN. Berikut merupakan rumus dari standard scaler:

$$z = \frac{x - \mu}{\sigma} \quad \text{(Rumus 2.1)}$$

Dengan z merupakan merupakan nilai fitur setelah dinormalisasi, x yaitu nilai asli fitur, yaitu rata-rata (mean) dari fitur, dan merupakan standar deviasi dari fitur.

2.2.3.2 One-Hot Encoding

One-Hot Encoding mengubah setiap kategori menjadi vektor biner. Contohnya, atribut ”Jenis Kelamin” dengan dua kategori (Pria dan Wanita) akan diubah menjadi dua kolom: Pria dan Wanita, dengan nilai 1 atau 0. Teknik ini mencegah asumsi ordinalitas antar kategori, tetapi dapat menyebabkan dimensi data meningkat drastis jika jumlah kategori besar.

2.2.3.3 Ordinal Encoding

Ordinal Encoding mengubah setiap kategori menjadi angka urut. Misalnya, kategori ”Rendah”, ”Sedang”, dan ”Tinggi” dapat diubah menjadi 1, 2, dan 3. Metode ini lebih hemat memori, namun berisiko memberikan makna urutan pada kategori yang sebenarnya tidak memiliki hubungan ordinal.

2.2.4 Metrik Evaluasi Model

2.2.4.1 Accuracy

Accuracy (Akurasi) mengukur proporsi prediksi yang benar terhadap total data. Meskipun populer, akurasi bisa menyesatkan jika data tidak seimbang (imbalance class). Rumus perhitungan dari accuracy dapat dilihat pada rumus 2.2:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad \text{(Rumus 2.2)}$$

Dengan TP (True Positive) merupakan atau jumlah data positif yang diprediksi

benar sebagai positif oleh model, TN (True Negative) yaitu jumlah data negatif yang diprediksi benar sebagai negatif oleh model, FP (False Positive) yaitu jumlah data negatif yang diprediksi salah sebagai positif oleh model, dan FN (False Negative) yaitu jumlah data positif yang diprediksi salah sebagai negatif oleh model.

2.2.4.2 Precision

Precision (Presisi) adalah rasio antara jumlah prediksi positif yang benar (True Positives) dengan seluruh prediksi positif. Tinggi rendahnya presisi menunjukkan seberapa andal model dalam memprediksi kelas positif. Rumus perhitungan dari accuracy dapat dilihat pada rumus 2.3:

$$Recall = \frac{TP}{TP + FP} \qquad \text{(Rumus 2.3)}$$

Dengan TP (True Positive) merupakan atau jumlah data positif yang diprediksi benar sebagai positif oleh model, dan FP (False Positive) yaitu jumlah data negatif yang diprediksi salah sebagai positif oleh model.

2.2.4.3 Recall

Recall mengukur seberapa banyak kasus positif yang berhasil ditemukan oleh model dari seluruh kasus positif yang sebenarnya. Metrik ini penting dalam kasus deteksi penyakit karena kesalahan melewatkan kasus positif bisa berdampak fatal. Rumus perhitungan dari recall dapat dilihat pada rumus 2.4:

$$Recall = \frac{TP}{TP + FN} \qquad \text{(Rumus 2.4)}$$

Dengan TP (True Positive) merupakan atau jumlah data positif yang diprediksi benar sebagai positif oleh model, dan FN (False Negative) yaitu jumlah data positif yang diprediksi salah sebagai negatif oleh model.

2.2.4.4 F1-Score

F1-Score adalah rata-rata harmonis dari presisi dan recall. Metrik ini digunakan saat kita menginginkan keseimbangan antara presisi dan recall, terutama pada dataset yang tidak seimbang. Rumus perhitungan dari recall dapat dilihat pada rumus 2.5:

$$F1\text{-}Score = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \qquad \text{(Rumus 2.5)}$$

2.2.5 K-Fold Cross Validation

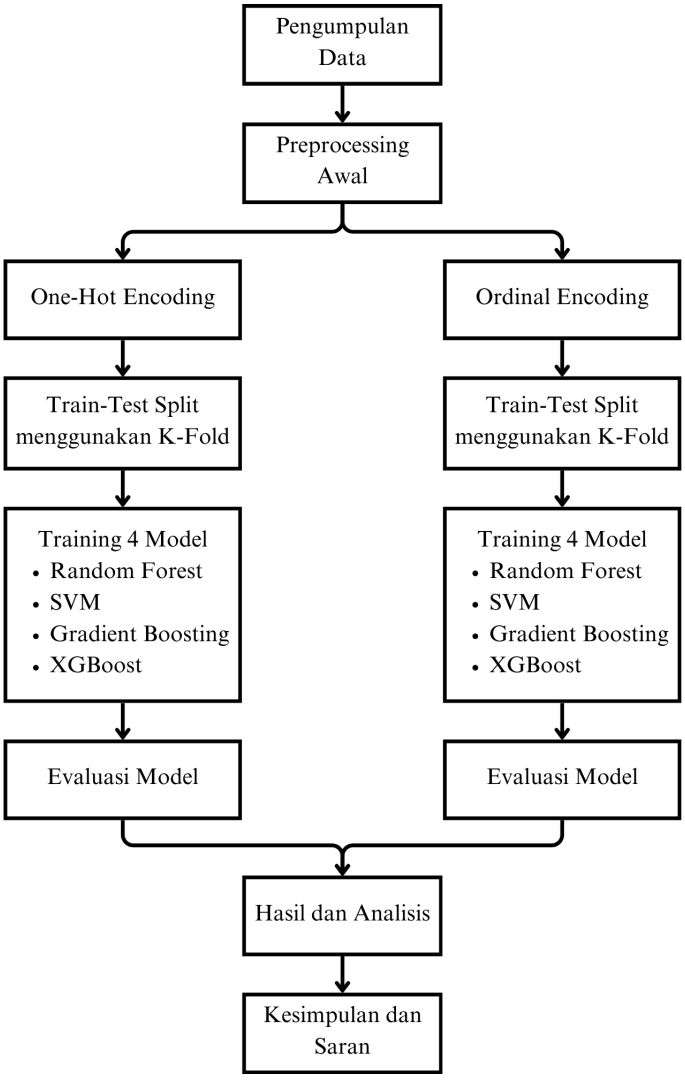
K-Fold Cross Validation adalah teknik validasi silang yang membagi dataset menjadi k bagian yang kurang lebih sama besar. Model akan dilatih dan divalidasi sebanyak k kali, di mana setiap fold secara bergiliran digunakan sebagai data validasi dan sisanya sebagai data pelatihan. Teknik ini membantu mengurangi variansi hasil evaluasi yang mungkin terjadi karena pembagian data yang tidak representatif.

Dengan menggunakan rata-rata dari hasil evaluasi di setiap fold, diperoleh estimasi kinerja model yang lebih stabil dan dapat digeneralisasi.

BAB III
METODE PENELITIAN

3.1 Alur Penelitian

Alur penelitian ini mencerminkan langkah-langkah utama yang dilakukan pada penelitian ini, dapat dilihat pada Gambar 3.1.



Gambar 3.1 Alur Penelitian

3.2 Penjabaran Langkah Penelitian

3.2.1 Pengumpulan Data

Data diambil dari Kaggle dengan judul dataset ”Heart Attack Prediction in Indonesia”. Dataset diunduh dalam bentuk csv.

3.2.2 Preprocessing Awal

Dataset asli dikurangi jumlahnya sebanyak 10000 baris sekaligus menyemimbangkan data target (heart_attack = 1 sebanyak 4994 dan heart_attack = 0 sebanyak 5006). Kemudian mengecek tipe data, data kosong, dan mengaplikasikan StandardScaler (penting untuk SVM).

3.2.3 Encoding

Pada bagian Encoding tebagi menjadi 2 cabang, One-Hot Encoding dan Ordinal Encoding. Encoding digunakan untuk variabel kategorik.

3.2.4 Splitting Data

Data kemudian dibagi menjadi 5 bagian (fold) dimana setiap fold secara bergantian dipakai sebagai data testing, dan 4 fold sisanya dipakai sebagai training. Splitting dilakukan secara terpisah untuk 2 metode encoding yang berbeda.

3.2.5 Training

Data kemudian dilatih untuk 4 model yang berbeda dengan metode encoding yang berbeda juga. model yang dipakai yaitu:

1. Random Forest Classifier
2. Support Vector Classifier
3. Gradient Boosting Classifier
4. XGBoost Classifier

3.2.6 Evaluasi

Model-model yang sudah dilatih kemudian diuji pada data test dan dievaluasi menggunakan 4 metrik:

1. Accuracy: Ketepatan model
2. Precision: Kemampuan mendeteksi positif secara benar
3. Recall: Kemampuan menangkap semua kasus positif
4. F1 Score: Harmoni antara precision dan recall

Nilai-nilai ini kemudian dibandingkan antar model dan antar metode encoding.

Selain itu, dilihat variabel apa saja yang paling mempengaruhi antar model dan antar metode encoding.

3.2.7 Hasil dan Analisis

Dari seluruh percobaan dilakukan perbandingan hasil performa antar model dan metode encoding. Dari hasil tersebut ditentukan manakah model dan metode encoding yang terbaik. Lalu identifikasi fitur yang paling berpengaruh pada prediksi serangan jantung.

3.2.8 Kesimpulan dan Saran

Pada langkah akhir ini disimpulkan mana model terbaik berdasarkan performanya. Kemudian memberikan saran mengenai model dan metode mana yang cocok untuk kasus prediksi serangan jantung, tindakan preventif dari serangan jantung berdasarkan urutan fitur yang berpengaruh dalam model, dan pengembangan yang bisa dilakukan dari penelitian ini.

3.3 Alat dan Bahan Tugas Akhir

Dalam menjalani penelitian, beberapa alat dan bahan digunakan untuk memastikan penelitian berjalan dengan baik.

3.3.1 Alat

Dalam membuat pengukuran frekuensi denyut nadi non-kontak dalam penelitian, berikut adalah alat-alat yang digunakan:

1. Laptop dengan spesifikasi sistem operasi Windows 11, processor Intel core i5-8250U CPU @ 8 core/1,80 GHz, RAM 8GB, grafis NVIDIA GeForce MX150, SSD 512 GB

2. Miniconda versi 25.3.1
3. Jupyter Notebook versi 7.3.2
4. Scikit-Learn versi 1.2.2
5. Python versi 3.10.16
6. NumPy versi 1.26.4
7. Pandas versi 2.2.3
8. Matplotlib versi 3.10.0
9. Overleaf untuk membuat dokumen Latex

3.3.2 Bahan

1. Dataset dari kaggle dengan judul ”Heart Attack Prediction in Indonesia” dalam format csv,
2. Jurnal ilmiah dan literatur sebelumnya sebagai dasar teori dan pembanding

3.4 Ilustrasi Preprocessing Data

Berikut dijelaskan ilustrasi perhitungan preprocessing data yang digunakan sebelum dilakukan pelatihan model machine learning. Teknik preprocessing yang digunakan meliputi *One-Hot Encoding, Ordinal Encoding*, dan *Standard Scaler*. Ilustrasi menggunakan sampel data dari beberapa baris dan kolom dalam dataset untuk menggambarkan bagaimana transformasi dilakukan.

3.4.1 One-Hot Encoding

One-Hot Encoding digunakan untuk mengubah fitur kategorikal menjadi representasi numerik biner. Fitur yang diencoding secara one-hot antara lain: gender, region, dan smoking_status. Misalnya, untuk kolom gender yang memiliki dua kategori yaitu ”Male” dan ”Female”, akan diubah menjadi dua kolom gender_Male dan gender_Female.

Tabel 3.1 Contoh Sebelum One-Hot Encoding

gender	region	smoking_status
Male	Urban	Never

Tabel 3.1 Contoh Sebelum One-Hot Encoding

gender	region	smoking_status
Male	Urban	Never
Male	Rural	Past

Tabel 3.1 menunjukkan 3 baris pertama dari 3 contoh kolom katoegrikal pada dataset yang akan di-*encoding* agar bisa diolah oleh model.

Tabel 3.2 Contoh Setelah One-Hot Encoding

gender_Male	gender_Female	region_Urban	region_Rural	smoking_Never	smoking_Past
1	0	1	0	1	0
1	0	1	0	1	0
1	0	0	1	0	1

Tabel 3.2 menunjukkan 3 baris pertama dari 3 contoh kolom katoegrikal pada dataset yang sudah di-*encoding* menggunakan *One-Hot Encoding*.

3.4.2 Ordinal Encoding

Ordinal Encoding digunakan pada fitur kategorikal yang memiliki urutan atau skala. Sebagai contoh, fitur income_level dengan urutan Low < Middle < High akan di-encode menjadi angka sesuai urutannya.

Tabel 3.3 Contoh Sebelum dan Sesudah Ordinal Encoding

income_level(asli)	income_level(ordinal)
Low	0
Middle	1
High	2

Pada tabel 3.3, Mapping yang digunakan adalah: Low = 0, Middle = 1, High = 2

3.4.3 Standard Scaler

Standard Scaler digunakan untuk melakukan normalisasi pada fitur numerik agar memiliki rata-rata 0 dan standar deviasi 1. Misalnya, pada kolom age, cholesterol_level, dan waist_circumference.

Tabel 3.4 Contoh Sebelum Standard Scaler

age	cholesterol_level	waist_circumference
61	190	106
53	215	98
45	242	80

Tabel 3.5 Contoh Setelah Standard Scaler

age_scaled	cholesterol_scaled	waist_scaled
1,22	-1,00	1,30
0,00	0,00	0,43
-1,22	1,00	-1,73

Catatan: Nilai di atas hanya ilustrasi, dihitung dari 3 baris pertama menggunakan rumus 2.1

3.5 Rancangan Pengujian

Rancangan pengujian dilakukan untuk mengevaluasi performa model machine learning dalam memprediksi risiko serangan jantung berdasarkan data yang telah dipra-proses. Pengujian dilakukan melalui beberapa skenario untuk mengetahui pengaruh metode encoding dan algoritma terhadap hasil prediksi.

Adapun skenario pengujian yang dilakukan adalah sebagai berikut:

1. Pengujian terhadap dua metode encoding berbeda:
    - One-Hot Encoding
    - Ordinal Encoding
2. Pengujian terhadap empat algoritma klasifikasi berbeda:
    - Random Forest

- Support Vector Machine (SVM)
- Gradient Boosting
- XGBoost

3. Skema Pembagian Data

   Data dibagi menjadi 80% untuk pelatihan dan 20% untuk pengujian. Selain itu, untuk menjaga generalisasi model, digunakan teknik validasi silang (k-fold cross-validation) dengan nilai k = 5.

4. Metrik Evaluasi

   Kinerja model dievaluasi berdasarkan beberapa metrik, yaitu:

   - Accuracy
   - Precision
   - Recall
   - F1-Score

Nantinya, hasil akan terlihat seperti contoh tabel 3.6. Sebagai catatan, nilai pada tabel 3.6 hanya ilustrasi.

Tabel 3.6 Contoh Hasil Evaluasi Model dengan 5-Fold Cross Validation

Encoding	**Algoritma**	**Accuracy**	**Precision**	**Recall**	**F1-Score**
One-Hot	Random Forest	0.87	0.88	0.86	0.87
One-Hot	SVM	0.84	0.85	0.83	0.84
One-Hot	Gradient Boosting	0.86	0.87	0.85	0.86
One-Hot	XGBoost	0.88	0.89	0.87	0.88
Ordinal	Random Forest	0.85	0.86	0.84	0.85
Ordinal	SVM	0.82	0.83	0.81	0.82
Ordinal	Gradient Boosting	0.84	0.85	0.83	0.84
Ordinal	XGBoost	0.86	0.87	0.85	0.86

Selain evaluasi performa model, dilakukan juga analisis terhadap fitur-fitur paling berpengaruh menggunakan metode feature importance pada algoritma Gradient Boosting dan XGBoost.

DAFTAR PUSTAKA

[1] L. Fitria and T. Nugroho. “Pengaruh Metode Encoding terhadap Kinerja Model Prediksi Penyakit Diabetes”. *Jurnal Ilmiah Komputer dan Informatika* (2021). Analisis One-Hot vs Ordinal Encoding dalam prediksi penyakit kronis.

[2] A. Rafiqi, B. Susanto, and R. Lestari. “Prediksi Serangan Jantung Menggunakan Random Forest, SVM, dan KNN dengan Penanganan Data Imbalance”. *Jurnal Teknologi Informasi dan Ilmu Komputer* (2023). Menggunakan teknik SMOTE dan membandingkan tiga algoritma klasifikasi.

[3] H. Kusuma, Y. Santoso, and N. Rahayu. “Prediksi Penyakit Jantung Menggunakan Support Vector Machine dan Optimasi Grid Search”. *Seminar Nasional Teknologi Informasi* (2020). Akurasi tinggi namun tidak membandingkan algoritma lain.

[4] A. Dewi and S. Wulandari. “Efektivitas Algoritma Gradient Boosting dalam Deteksi Penyakit Jantung pada Dataset Lokal Indonesia”. *Jurnal Sistem Informasi dan Komputer Terapan Indonesia* (2021). Studi pada data lokal dengan preprocessing sederhana.

[5] R. Putra and D. Syafitri. “Perbandingan XGBoost dan Naïve Bayes dalam Prediksi Penyakit Jantung menggunakan Data Cleveland”. *Jurnal Informatika dan Komputasi* (2022). Analisis feature importance pada XGBoost.