BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Sejumlah studi sebelumnya telah mengkaji penerapan berbagai algoritma *machine learning* dalam upaya prediksi penyakit jantung. Kajian tersebut menjadi dasar dalam memahami perkembangan penelitian-penelitian yang telah dilakukan sebelumnya. Berikut adalah ringkasan dari penelitian-penelitian yang relevan terkait pada Tabel 2.1 Penelitian Terdahulu.

Tabel 2. 1 Penelitian Terdahulu

No	Peneliti	Judul	Dataset	Metode	Hasil
1	Sreekumari [7]	*A Comparative Study of Heart Disease Prediction sing Machine Learning*	*Cleveland Heart Disease Dataset (UCI), Amerika Serikat*	*Decision Tree* *Random Forest* *Logistic Regression* *K-Nearest Neighbors* *Decision Tree + Grid Search*	*Decision Tree: 92%* *Random Forest: 91%* *Logistic Regression: 89%* *K-Nearest Neighbors: 72%* *Decision Tree & GridSearch: 88%*

No	Peneliti	Judul	Dataset	Metode	Hasil
2	Mohammed [13]	*The Prediction of Heart Disease Using Machine Learning Algorithms*	*Heart Disease Dataset (UCI), Amerika Serikat*	*Naïve Bayes* *Decision Tree* *Logistic Regression* *K-Nearest Neighbors* *Support Vector Machine* *Random Forest* *Multi Layer Perceptron*	*Naïve Bayes: 97%* *Decision Tree: 97%* *Logistic Regression: 97%* *K-Nearest Neighbors: 97%* *Support Vector Machine: 97%* *Random Forest: 97%* *Multi Layer Perceptron: 97%*

No	Peneliti	Judul	Dataset	Metode	Hasil
3	Madhani Aritonang [14]	IMPLEMENTASI ALGORITMA *LOGISTIC REGRESSION* DALAM MEMPREDIKSI PENYAKIT JANTUNG	Data pasien penyakit jantung (1025 data), Indonesia	*Logistic Regression*	Model *Logistic Regression* berhasil digunakan untuk prediksi penyakit jantung berbasis 1025 data pasien

4	Asyafiiyah [15]	Prediksi Pasien Terindikasi Penyakit Jantung Menggunakan Metode *Logistic Regression*	Data pasien penyakit jantung, Indonesia	*Logistic Regression*	*Logistic Regression* digunakan untuk klasifikasi *biner* risiko penyakit jantung berdasarkan data pasien
5	Ayankoya [10]	*Heart Disease Risk Prediction Using Machine Learning and Health Datasets*	*Health/Heart Disease Dataset, Internasional*	*Logistic Regression* *Random Forest* *XGBoost*	*Logistic Regression: 89%* *Random Forest: 92%* *XGBoost: 93%*
6	Paras Negi [16]	*Evaluation of Machine Learning Model's Accuracy for Heart Attack Prediction*	*Heart Disease Dataset, Amerika Serikat*	*Decision Tree* *Logistic Regression* *Support Vector Machine*	*Decision Tree: 80%* *Logistic Regression: 88%* *Support Vector Machine: 87%* *Random Forest: 83%*

				Random Forest	

No	Peneliti	Judul	Dataset	Metode	Hasil
7	Al-Karaki [17]	*Predicting Coronary Heart Disease Using a Suite of Machine Learning Models*	*Coronary Heart Disease Dataset, Internasional*	*Logistic Regression* *Linear Discriminant Analysis* *Support Vector Machine* *Random Forest* *Decision Tree* *Naïve Bayes* *K-Nearest Neighbors* *XGBoost*	*Logistic Regression: 67%* *Linear Discriminant Analysis: 68%* *Support Vector Machine: 66%* *Random Forest: 67%* *Decision Tree: 57%* *Naïve Bayes: 80%* *K-Nearest Neighbors: 64%* *XGBoost: 67%*

No	Peneliti	Judul	Dataset	Metode	Hasil
8	Ujjwal Daharwal [5]	*Comparison of Machine Learning Algorithms for Heart Disease Prediction*	*Heart Disease Dataset (UCI), Amerika Serikat*	*Naïve Bayes* *K-Nearest Neighbors* *Logistic Regression*	*Naïve Bayes: 86,6%* *K-Nearest Neighbors: 90,8%* *Logistic Regression: 91,4%*

No	Peneliti	Judul	Dataset	Metode	Hasil
9	Glen Fierre Sijabat [18]	Studi Komparatif *Model Machine Learning* untuk Klasifikasi Penyakit Jantung dengan *SMOTE* pada *Data Imbalanced*	*Heart Disease Dataset Imbalanced, Amerika Serikat*	*Logistic Regression, Random Forest, LightGBM, dan XGBoost*	*Logistic Regression: 84,78% (tanpa SMOTE)* *Random Forest + SMOTE: 86,96%*
10	Chandrasekhar [19]	*Enhancing Heart Disease Prediction Accuracy through Machine Learning Techniques and Optimization*	*Heart Disease Dataset, Amerika Serikat*	*Logistic Regression AdaBoost Ensemble*	*Logistic Regression: 90,16%* *AdaBoost: 90%* *Ensemble: 93,44% & 95%*

Berdasarkan Tabel 2.1 Penelitian Terdahulu, terlihat bahwa berbagai algoritma *machine learning* telah digunakan dalam prediksi penyakit jantung, seperti *Decision Tree, Random Forest, Logistic Regression, Support Vector Machine*, serta *Naïve Bayes* hingga metode ensemble seperti *AdaBoos*t dan *XGBoost*. Selain menggunakan metode yang beragam, penelitian-penelitian

tersebut juga memanfaatkan dataset yang berbeda-beda, mulai dari *Cleveland Heart Disease Dataset* dan *Heart Disease Dataset* dari *UCI Machine Learning Repository* hingga data pasien penyakit jantung yang dikumpulkan secara lokal. Perbedaan karakteristik dataset, teknik *preprocessing*, serta konfigurasi model menyebabkan variasi performa yang cukup signifikan pada setiap penelitian.

Penelitian oleh Sreekumari [7] menggunakan *Cleveland Heart Disease Dataset* dari *UCI Machine Learning Repository* dan menunjukkan bahwa *Decision Tree* memperoleh akurasi tertinggi sebesar 92%, diikuti *Random Forest* sebesar 91%, serta *Logistic Regression* sebesar 89%. Selanjutnya, Mohammed [13] menggunakan *Heart Disease Dataset* dari *UCI* dan melaporkan bahwa seluruh algoritma yang diuji mencapai akurasi sebesar 97%. Hasil tersebut menunjukkan bahwa karakteristik dataset dan proses pengolahan data memiliki pengaruh yang signifikan terhadap performa model yang dihasilkan.

Pada penelitian Paras Negi [16], yang menggunakan *Heart Disease Dataset, Logistic Regression* mencapai akurasi sebesar 88% dan menjadi algoritma dengan performa terbaik dibandingkan *Support Vector Machine* (87%), *Random Forest* (83%), dan *Decision Tree* (80%). Sementara itu, penelitian oleh Al-Karaki [17] menggunakan *Coronary Heart Disease Dataset* dan menunjukkan variasi performa yang cukup lebar, dengan *Naïve Bayes* memperoleh akurasi tertinggi sebesar 80%, sedangkan algoritma lainnya berada pada rentang 57%–68%. Temuan tersebut mengindikasikan bahwa performa suatu model sangat dipengaruhi oleh karakteristik dataset yang digunakan.

Penelitian oleh Daharwal [5] dan Ayankoya [10] menunjukkan bahwa *Logistic Regression* tetap menjadi salah satu *algoritma* dengan performa yang konsisten tinggi dengan akurasi di atas 89%. Selain itu, Glen Fierre Sijabat [18] menunjukkan bahwa penerapan *Synthetic Minority Oversampling Technique (SMOTE)* pada data yang tidak seimbang mampu meningkatkan performa model dibandingkan tanpa penanganan ketidakseimbangan kelas. Sementara itu, Chandrasekhar [19] menunjukkan bahwa metode *ensemble* mampu menghasilkan performa terbaik dengan akurasi mencapai 95%, lebih tinggi dibandingkan model tunggal seperti *Logistic Regression*.

Berdasarkan kajian terhadap penelitian-penelitian terdahulu, terlihat bahwa sebagian besar penelitian masih menggunakan dataset internasional, khususnya *Cleveland Heart Disease Dataset* dan berbagai *Heart Disease Dataset* lainnya yang berasal dari luar Indonesia. Meskipun menghasilkan tingkat akurasi yang tinggi, penggunaan dataset tersebut belum tentu merepresentasikan karakteristik populasi Indonesia yang memiliki perbedaan dalam pola hidup, kondisi kesehatan, serta faktor risiko penyakit jantung.

Selain itu, sebagian besar penelitian berfokus pada peningkatan nilai akurasi model tanpa membahas aspek interpretabilitas model dan kemampuan deteksi pasien berisiko tinggi secara mendalam. Pada konteks kesehatan, kemampuan model dalam meminimalkan kesalahan prediksi negatif palsu (*false negative*) menjadi sangat penting karena berkaitan langsung dengan risiko keterlambatan penanganan pasien.

Berdasarkan kondisi tersebut, penelitian ini mengadopsi penggunaan algoritma *Logistic Regression* yang telah menunjukkan performa kompetitif pada berbagai penelitian terdahulu. Selain itu, penelitian ini mengadopsi penggunaan teknik *Synthetic Minority Oversampling Technique (SMOTE), Grid Search Cross Validation*, penyesuaian *threshold* klasifikasi, serta analisis *Odds Ratio* untuk meningkatkan performa dan interpretabilitas model dalam memprediksi risiko penyakit jantung.

Penelitian ini menggunakan *Heart Attack Prediction Indonesia Dataset* yang terdiri dari 158.355 data dan dirancang berdasarkan karakteristik populasi Indonesia. Penggunaan dataset tersebut diharapkan dapat menghasilkan model yang lebih representatif terhadap kondisi kesehatan masyarakat Indonesia dibandingkan penelitian-penelitian terdahulu yang mayoritas menggunakan dataset internasional.

Evaluasi model dilakukan secara komprehensif menggunakan metrik *accuracy*, *precision*, *recall*, *F1-score*, dan *ROC-AUC*. Aspek interpretabilitas juga dianalisis menggunakan *Odds Ratio* sehingga model tidak hanya menghasilkan prediksi, tetapi juga mampu menjelaskan pengaruh masing-masing faktor risiko terhadap kemungkinan terjadinya penyakit jantung. Dengan demikian, penelitian

ini tidak hanya mereplikasi penelitian terdahulu, tetapi juga memberikan kontribusi melalui penggunaan data lokal, optimasi model, evaluasi yang lebih komprehensif, serta interpretasi hasil yang lebih relevan dalam konteks medis.

2.2 Teori yang Berkaitan

Sebagai dasar dalam pelaksanaan penelitian, diperlukan pemahaman terhadap teori-teori yang relevan. Berikut merupakan beberapa teori yang berkaitan dengan penelitian ini.

2.2.1 Penyakit Jantung

Penyakit jantung merujuk pada sekelompok gangguan yang berdampak pada struktur maupun fungsi jantung [20], terdiri atas penyakit arteri koroner, gagal jantung, gangguan irama jantung (aritmia), maupun kelainan pada katup jantung. Faktor risiko yang mendasari kondisi ini dapat dibedakan menjadi dua kategori: faktor yang tidak dapat diubah seperti usia, jenis kelamin, dan riwayat keluarga; serta faktor yang dapat dikendalikan meliputi hipertensi, diabetes, dislipidemia, obesitas, kebiasaan merokok, kurangnya aktivitas fisik, dan pola makan yang tidak sehat.

Mengingat potensi komplikasi yang dapat ditimbulkan seperti infark miokard dan kematian mendadak, upaya identifikasi secara dini memegang peranan penting dalam penanggulangan penyakit tersebut.

2.2.2 Machine Learning

Machine learning (*ML*) merupakan cabang kecerdasan buatan yang memungkinkan sistem komputer untuk belajar dari data dan menghasilkan prediksi secara mandiri, tanpa perlu *deprograme* secara eksplisit untuk setiap tugas tertentu [21]. Salah satu paradigma utama dalam *ML* adalah *supervised learning*, di mana model dibangun menggunakan data berlabel untuk mempelajari pola keterkaitan antara fitur *input* dan variabel *target* yang ingin diprediksi. Dalam konteks prediksi penyakit jantung, pendekatan ini diterapkan dengan melatih model menggunakan data rekam medis pasien yang telah dilabeli berdasarkan ada atau tidaknya kejadian serangan jantung.

2.2.3 Logistic Regression

Logistic Regression adalah salah satu *algoritma* dalam *supervised learning* yang dirancang untuk menangani permasalahan klasifikasi *biner* [12]. Berbeda dengan *linear regression* yang menghasilkan keluaran berupa nilai kontinu, *Logistic Regression* memanfaatkan fungsi *sigmoid* untuk mentransformasikan kombinasi linear dari fitur-fitur input ke dalam rentang probabilitas [0,1]. Fungsi *sigmoid* didefinisikan sebagai berikut pada Rumus 2.1 *Linear Combination*.

$$z = \beta 0 + \beta 1x1 + \beta 2x2 + \cdots + \beta nxn$$

Rumus 2. 1 Linear Combination

Dengan z merupakan kombinasi *linear* dari seluruh fitur input yang digunakan sebagai prediktor.

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Rumus 2. 2 Sigmoid Function

$$P(Y = 1 \mid X) = \sigma(z)$$

Rumus 2.3 Probability Function

Persamaan tersebut menghasilkan probabilitas terjadinya kelas positif berdasarkan nilai kombinasi linear fitur-fitur *input* yang dimiliki suatu sampel.

$$L = -1/n \, \Sigma \, [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)]$$

Rumus 2.4 Binary Cross-Entropy Loss Function

Di mana y_i adalah nilai *aktual*, $\hat{y}_i$ adalah nilai probabilitas yang diprediksi model, dan n adalah jumlah *sampel*. Minimisasi fungsi *log-loss* dilakukan menggunakan metode optimasi numerik, seperti *Gradient Descent*, *Newton Method*, maupun *solver Liblinear* yang tersedia pada implementasi *Logistic Regression*. Prediksi akhir ditentukan berdasarkan ambang batas probabilitas (*threshold*) tertentu. Secara umum nilai threshold yang digunakan adalah 0,5, namun nilai tersebut dapat disesuaikan untuk meningkatkan performa model sesuai kebutuhan kasus yang dihadapi.

Selain menggunakan nilai *threshold* standar sebesar 0,5, beberapa penelitian menerapkan penyesuaian *threshold* klasifikasi untuk memperoleh

keseimbangan yang lebih baik antara *precision* dan *recall*. Dalam kasus prediksi penyakit, peningkatan nilai *recall* menjadi penting untuk meminimalkan jumlah pasien berisiko yang tidak terdeteksi oleh model. Oleh karena itu, penyesuaian *threshold* dapat digunakan sebagai salah satu strategi untuk meningkatkan sensitivitas model tanpa mengubah algoritma dasar yang digunakan.

2.2.4 Data Preprocessing

Data preprocessing merupakan serangkaian proses persiapan yang dilakukan terhadap data mentah sebelum digunakan dalam pelatihan model *machine learning*. Proses ini meliputi penanganan *missing values*

melalui teknik imputasi maupun penghapusan data, konversi variabel kategorikal ke dalam format numerik menggunakan *label encoding* atau *one-hot encoding*, serta normalisasi atau standardisasi fitur numerik guna menyeragamkan skala di antara fitur-fitur yang digunakan.

Pada dataset yang memiliki distribusi kelas tidak seimbang *(imbalanced data)*, teknik *Synthetic Minority Oversampling Technique (SMOTE)* dapat digunakan untuk menghasilkan sampel sintetis pada kelas minoritas sehingga distribusi data menjadi lebih seimbang. Penerapan *SMOTE* bertujuan meningkatkan kemampuan model dalam mengenali pola pada kelas minoritas dan mengurangi kecenderungan model untuk bias terhadap kelas mayoritas.

2.2.5 SMOTE

Synthetic Minority Oversampling Technique (SMOTE) merupakan metode *oversampling* yang digunakan untuk mengatasi permasalahan ketidakseimbangan kelas (*class imbalance*) pada *dataset*. Berbeda dengan metode *random oversampling* yang hanya menggandakan data pada kelas minoritas, *SMOTE* menghasilkan data sintetis baru berdasarkan karakteristik data yang sudah ada pada kelas minoritas. Pendekatan ini bertujuan untuk meningkatkan representasi kelas minoritas sehingga model dapat mempelajari pola data secara lebih seimbang.

Proses pembentukan data sintetis pada *SMOTE* dilakukan dengan memilih salah satu sampel dari kelas minoritas, kemudian mencari beberapa

tetangga terdekat (*K-Nearest Neighbors*). Data baru dibentuk pada garis yang menghubungkan sampel asli dengan salah satu tetangga terdekatnya. Secara matematis, proses pembentukan sampel sintetis dapat dinyatakan sebagai berikut pada Rumus 2.5 *SMOTE Synythetic Sample Generation*.

$$x_{new} = x_i + \lambda(x_{nn} - x_i)$$

Rumus 2.5 SMOTE Synthetic Sample Generation

dengan:

- x_{new}= data sintetis yang dihasilkan,
- x_i= sampel dari kelas minoritas,
- x_{nn}= salah satu tetangga terdekat dari x_i,
- λ= bilangan acak pada rentang 0 sampai 1.

Penerapan *SMOTE* dalam penelitian ini bertujuan untuk menyeimbangkan distribusi kelas pada dataset penyakit jantung sehingga model *Logistic Regression* dapat mengenali pola pada kelas pasien berisiko secara lebih baik. Penggunaan *SMOTE* juga diharapkan dapat meningkatkan nilai *recall* dan mengurangi kemungkinan terjadinya kesalahan prediksi negatif palsu (*false negative*) yang sangat penting dalam konteks prediksi penyakit jantung.

2.2.6 Evaluasi Model

Evaluasi model dalam penelitian ini dilakukan menggunakan sejumlah metrik yang saling melengkapi. Akurasi menunjukkan rasio prediksi yang benar dibandingkan jumlah seluruh data. Presisi menggambarkan ketepatan prediksi positif yang dihasilkan model, dihitung sebagai berikut pada Rumus 2.6 *Precision*.

$$Precision = \frac{TP}{TP+FP}.$$

Rumus 2. 6 Precision

Recall atau sensitivitas menggambarkan seberapa baik model dapat mengidentifikasi semua kasus positif yang benar-benar ada, dengan formula sebagai berikut pada Rumus 2.7 *Recall* dan Rumus 2.8 *F1-Score*.

$$Recall = \frac{TP}{TP+FN}.$$

Rumus 2. 7 Recall

$$F_1 = 2 \times \backslash frac\{Precision \times Recall\}\{Precision + Recall\}$$

Rumus 2. 8 F1-Score

F1-score merepresentasikan keseimbangan antara *precision* dan *recall* melalui perhitungan rata-rata harmonisnya. *ROC-AUC* menggambarkan kapasitas diskriminasi model dalam membedakan kelas positif dan negatif pada berbagai nilai *threshold*. Sementara itu, *confusion matrix* menyajikan ringkasan distribusi hasil prediksi benar dan salah secara terstruktur. Dalam aplikasi medis, *recall* menjadi metrik yang paling kritis, mengingat kegagalan dalam mendeteksi pasien berisiko tinggi yang dikenal sebagai *false negative* dapat menimbulkan konsekuensi yang fatal.

2.2.7 Cross Validation

Cross Validation adalah suatu metode untuk mengevaluasi performa model yang bekerja dengan membagi dataset ke dalam *k fold* secara berulang, dengan tujuan menghasilkan estimasi performa yang lebih stabil dan bebas dari *bias*. Dalam skema *K-Fold Cross Validation*, dataset dipartisi menjadi *k subset* yang sama besar, pada setiap iterasi, satu *fold* berperan sebagai data pengujian sementara *k-1 fold* yang tersisa digunakan sebagai data pelatihan. Proses ini dijalankan sebanyak k kali secara bergantian, dan nilai akhir yang dilaporkan merupakan rata-rata dari metrik evaluasi yang diperoleh di seluruh iterasi. Dengan pendekatan ini, risiko *overfitting* dapat ditekan dan gambaran mengenai kemampuan generalisasi model menjadi lebih representatif.

2.2.8 GridSearchCV

GridSearchCV merupakan metode optimasi *hyperparameter* yang mengintegrasikan pendekatan *grid search* dengan *cross validation* secara sistematis. *Hyperparameter* sendiri didefinisikan sebagai *parameter* yang tidak diperoleh melalui proses pelatihan model, melainkan ditetapkan terlebih dahulu sebelum pelatihan dimulai. Salah satu contohnya adalah *parameter C* pada *Logistic Regression* yang berfungsi mengatur kekuatan regularisasi. *Grid search* bekerja dengan mendefinisikan sekumpulan nilai kandidat untuk setiap *hyperparameter*, kemudian menelusuri seluruh kombinasi yang mungkin secara *exhaustive*. Setiap kombinasi dievaluasi menggunakan *cross validation* guna memperoleh estimasi performa yang stabil dan tidak bias. Kombinasi yang menghasilkan nilai metrik terbaik selanjutnya ditetapkan sebagai konfigurasi optimal model. Dibandingkan dengan pencarian manual, penggunaan *GridSearchCV* memungkinkan proses tuning *hyperparameter* dilakukan secara lebih objektif, terstruktur, dan menghasilkan model yang lebih optimal.

2.2.9 Odds Ratio

Odds Ratio merupakan ukuran statistik yang digunakan untuk menginterpretasikan pengaruh suatu variabel *independen* terhadap peluang terjadinya suatu kejadian pada *model Logistic Regression*. Nilai *Odds Ratio* diperoleh dengan melakukan transformasi eksponensial terhadap koefisien *Logistic Regression*, yang dapat dinyatakan sebagai pada Rumus 2.9 *Odds Ratio*.

$$OR = e^{\beta}$$

Rumus 2. 9 Odds Ratio

dengan β merupakan koefisien dari variabel yang bersangkutan. Nilai *Odds Ratio* lebih besar dari 1 menunjukkan bahwa peningkatan variabel tersebut akan meningkatkan peluang terjadinya kejadian yang diprediksi. Sebaliknya, nilai *Odds Ratio* kurang dari 1 menunjukkan adanya pengaruh yang menurunkan peluang kejadian.

Dalam konteks prediksi penyakit jantung, *Odds Ratio* digunakan untuk mengidentifikasi faktor-faktor risiko yang memiliki pengaruh paling besar terhadap kemungkinan terjadinya penyakit jantung. Pendekatan ini memberikan interpretasi yang lebih mudah dipahami dibandingkan hanya melihat nilai koefisien model secara langsung sehingga lebih sesuai digunakan dalam analisis medis.

2.3 *CRISP DM* (*Framework* Penelitian)

Framework yang digunakan dalam penelitian ini adalah *CRISP-DM (Cross Industry Standard Process for Data Mining),* yakni metodologi yang telah banyak diadopsi dalam proyek *data mining* dan *machine learning* berkat alur kerjanya yang sistematis, terstruktur, dan bersifat iteratif. *CRISP-DM* memfasilitasi peneliti untuk memahami permasalahan secara mendalam, mengolah data secara efektif, serta membangun model dengan performa dan interpretabilitas yang baik. Secara keseluruhan, *framework* proses ini meliputi enam tahap inti, yakni *business understanding, data understanding, data preparation, modeling, evaluation,* dan *deployment.*

2.3.1 Business Understanding

Tahap *business understanding* merupakan fase pembuka dalam *CRISP-DM* yang menitikberatkan pada pemahaman terhadap permasalahan penelitian dan tujuan yang hendak dicapai. Pada tahap ini dilakukan identifikasi terhadap latar belakang masalah, kebutuhan penelitian, serta ruang lingkup kajian yang akan dibahas. Dalam konteks penelitian ini, permasalahan utama yang diangkat adalah tingginya risiko penyakit jantung di Indonesia yang memerlukan pendekatan deteksi dini berbasis data.

Sasaran utama dari penelitian ini ialah merancang model machine learning yang dapat memperkirakan risiko penyakit jantung menggunakan data pasien, dengan penekanan khusus pada representasi populasi Indonesia. Di samping itu, penelitian ini juga diarahkan untuk mengevaluasi performa model secara menyeluruh serta mengungkap faktor-faktor yang memberikan kontribusi terbesar terhadap hasil prediksi. Langkah ini menjadi dasar dalam menetapkan strategi analisis, pemilihan metode, serta tolok ukur keberhasilan

penelitian secara menyeluruh.

2.3.2 Data Understanding

Tahap *data understanding* bertujuan untuk memahami karakteristik dataset yang digunakan dalam penelitian. Dataset yang digunakan adalah *Heart Attack Prediction Indonesia Dataset* yang berisi berbagai atribut terkait kondisi kesehatan individu. Pada tahap ini dilakukan eksplorasi data atau *Exploratory Data Analysis* untuk memahami distribusi data, mengidentifikasi tipe data baik numerik maupun kategorikal, serta menganalisis hubungan antar variabel. Selain itu, dilakukan pula deteksi *missing values*, serta analisis distribusi kelas target untuk mengidentifikasi potensi ketidakseimbangan kelas *(class imbalance)* pada dataset.

2.3.3 Data Preparation

Tahap *data preparation* dilakukan untuk mempersiapkan data agar siap digunakan pada proses pemodelan. Proses tersebut mencakup penanganan *missing values*, konversi variabel kategorikal ke representasi numerik menggunakan *label encoding*, standardisasi fitur numerik menggunakan *StandardScaler*, serta penerapan *Synthetic Minority Oversampling Technique (SMOTE)* untuk menangani ketidakseimbangan kelas.

2.3.4 Modeling

Pada tahap pemodelan, algoritma *Logistic Regression* digunakan sebagai model utama. Untuk memperoleh konfigurasi model yang optimal, dilakukan optimasi *hyperparameter* menggunakan *GridSearchCV*. Selain itu, dilakukan penyesuaian *threshold* klasifikasi untuk meningkatkan kemampuan model dalam mendeteksi pasien berisiko penyakit jantung.

2.3.5 Evaluation

Tahap evaluasi dilaksanakan untuk mengukur dan menganalisis sejauh mana performa model yang telah dibangun dapat diandalkan. Penilaian tidak bertumpu pada satu metrik tunggal, melainkan menggunakan kombinasi beberapa metrik untuk memperoleh gambaran yang lebih menyeluruh. Metrik-metrik yang digunakan dalam penelitian ini mencakup akurasi sebagai ukuran proporsi prediksi yang tepat, presisi untuk menilai ketepatan dalam

mengidentifikasi kelas positif, *recall* untuk mengukur sensitivitas model terhadap kasus positif, *F1-score* sebagai nilai harmonis antara presisi dan *recall*, *ROC-AUC* untuk mengevaluasi kemampuan diskriminasi model secara keseluruhan, serta *Confusion Matrix* untuk memetakan distribusi hasil prediksi yang benar maupun yang keliru. Penerapan metode *Cross Validation* pada tahap ini dimaksudkan untuk memverifikasi kemampuan generalisasi model, sekaligus meminimalkan risiko *overfitting* dan menghasilkan penilaian yang lebih objektif.

Selain evaluasi performa klasifikasi, dilakukan pula analisis *Odds Ratio* berdasarkan koefisien *Logistic Regression* untuk mengidentifikasi faktor-faktor yang paling berpengaruh terhadap risiko penyakit jantung.

2.3.6 Deployment

Tahap *deployment* merupakan fase penutup dalam kerangka *CRISP-DM* yang berorientasi pada pemanfaatan hasil pemodelan. Dalam konteks penelitian ini, implementasi tidak diwujudkan dalam bentuk pengembangan sistem operasional secara langsung, melainkan berupa penyajian hasil analisis beserta model yang dapat dijadikan acuan atau referensi ilmiah.

Temuan dari penelitian ini diharapkan dapat memberikan sumbangsih pada sektor kesehatan, terutama dalam mendukung upaya deteksi dini penyakit jantung. Selain itu, model yang dihasilkan juga dapat menjadi dasar untuk pengembangan sistem pendukung keputusan *(decision support system)* di masa mendatang.

Secara keseluruhan, alur penelitian ini mengikuti tahapan yang sistematis, dimulai dari identifikasi dan pemahaman permasalahan (*business understanding*), eksplorasi serta pemahaman karakteristik data (*data understanding*), pengolahan dan persiapan data (*data preparation*), konstruksi model prediktif (*modeling*), evaluasi performa menggunakan *5-Fold Cross Validation* dan berbagai metrik klasifikasi (*evaluation*), hingga interpretasi hasil sebagai bentuk *deployment* dalam lingkup penelitian. Pendekatan terstruktur ini diharapkan dapat menghasilkan model yang tidak hanya unggul secara performa, tetapi juga mudah diinterpretasikan dan relevan untuk

diterapkan dalam kondisi nyata.

2.4 *Tools/Software* yang Digunakan

Penelitian ini memanfaatkan berbagai *tools* dan *software* untuk mendukung seluruh tahapan dalam proses *data mining*, mulai dari pengolahan data, pembangunan model, hingga evaluasi dan visualisasi hasil. Pemilihan *tools* didasarkan pada kemudahan penggunaan, kelengkapan fitur, serta dukungan komunitas yang luas. Adapun *tools* dan *software* yang digunakan pelaksanaan penelitian ini adalah sebagai berikut.

1. *Python*

   Bahasa pemrograman yang digunakan sebagai tulang punggung dalam penelitian ini adalah *Python*. Pemilihannya didasarkan pada kesederhanaan sintaks yang dimiliki, sehingga proses pengembangan model *machine learning* dapat dilakukan secara lebih efisien dan mudah dipahami. Selain itu, Python memiliki ekosistem *library* yang sangat lengkap, khususnya dalam bidang *data science* dan *machine learning*, sehingga mampu mendukung seluruh kebutuhan penelitian, mulai dari *preprocessing data* hingga evaluasi model. Dukungan komunitas yang besar juga menjadi keunggulan *Python*, karena memudahkan peneliti dalam mencari referensi, dokumentasi, maupun solusi terhadap permasalahan yang dihadapi selama proses penelitian.

2. *Scikit-learn*

   Scikit-learn merupakan *library* utama yang dimanfaatkan dalam penelitian ini untuk mengimplementasikan algoritma *Logistic Regression*. Di luar fungsi pemodelan, *library* ini juga menyediakan berbagai utilitas untuk keperluan praproses data, di antaranya *StandardScaler* untuk normalisasi fitur numerik dan *LabelEncoder* untuk mengkonversi variabel kategorikal ke representasi numerik. Pada tahap evaluasi, *Scikit-learn* digunakan untuk menghitung sejumlah metrik performa, mencakup akurasi, presisi, *recall*, *F1-score*, dan *ROC-AUC*. Selain itu, *library* ini mendukung penerapan metode *Cross Validation*, termasuk *5-Fold Cross Validation*, yang

diterapkan untuk memverifikasi kemampuan generalisasi model sekaligus meminimalkan risiko *overfitting*. *Library* ini juga digunakan untuk implementasi *GridSearchCV* dalam proses optimasi *hyperparameter model*.

3. *Pandas* dan *NumPy*

   Dua *library* yang menjadi tulang punggung proses pengolahan data dalam penelitian ini adalah *Pandas* dan *NumPy*. *Pandas* berperan dalam pembacaan dataset, pelaksanaan eksplorasi data (*Exploratory Data Analysis*/*EDA*), serta berbagai operasi manipulasi data mencakup *filtering*, *grouping*, dan transformasi data. *Library* ini juga menyederhanakan penanganan *missing values* dan pengelolaan data berbentuk tabel melalui struktur *DataFrame*. Di sisi lain, *NumPy* dimanfaatkan untuk mendukung komputasi numerik yang efisien, khususnya dalam operasi *array* dan kalkulasi matematis. Kombinasi keduanya memungkinkan proses pengolahan data berlangsung secara lebih cepat, terstruktur, dan efisien.

4. *Matplotlib* dan *Seaborn*

   Matplotlib dan *Seaborn* digunakan sebagai *tools* untuk visualisasi data dalam penelitian ini. *Matplotlib* merupakan *library* dasar untuk membuat berbagai jenis grafik, sedangkan *Seaborn* digunakan untuk menghasilkan visualisasi yang lebih informatif dan estetis. Kedua *library* ini digunakan dalam tahap eksplorasi data untuk memahami distribusi data, hubungan antar variabel, serta mendeteksi pola tertentu. Selain itu, visualisasi juga digunakan dalam tahap evaluasi model, seperti pembuatan *confusion matrix, ROC curve*, serta visualisasi *Odds Ratio*. Visualisasi ini membantu dalam memahami performa model secara lebih intuitif dan mendukung interpretasi hasil penelitian.

5. *Jupyter Notebook*

   Dalam penelitian ini, *Jupyter Notebook* digunakan sebagai lingkungan pengembangan interaktif berbasis *web*. *Platform* ini memungkinkan penulisan dan eksekusi kode secara bertahap, sekaligus mengintegrasikan

kode, teks penjelasan, dan visualisasi dalam satu dokumen yang terpadu. Kemampuan tersebut sangat mendukung kelancaran proses eksplorasi data, pengembangan model, dan dokumentasi penelitian. Di samping itu, *Jupyter Notebook* juga mempermudah proses *debugging* dan eksperimentasi terhadap berbagai konfigurasi model yang diuji.

6. *Anaconda Navigator*

   Anaconda Navigator digunakan sebagai *platform* untuk mengelola *environment* dan dependensi dalam penelitian ini. Dengan menggunakan *Anaconda*, peneliti dapat mengatur berbagai *library* yang dibutuhkan tanpa mengalami konflik versi. *Anaconda* juga menyediakan kemudahan dalam instalasi *package* serta integrasi dengan *tools* seperti *Jupyter Notebook*. Penggunaan *Anaconda* membantu memastikan bahwa lingkungan pengembangan tetap stabil dan dapat direproduksi, sehingga mendukung validitas dan konsistensi hasil penelitian.