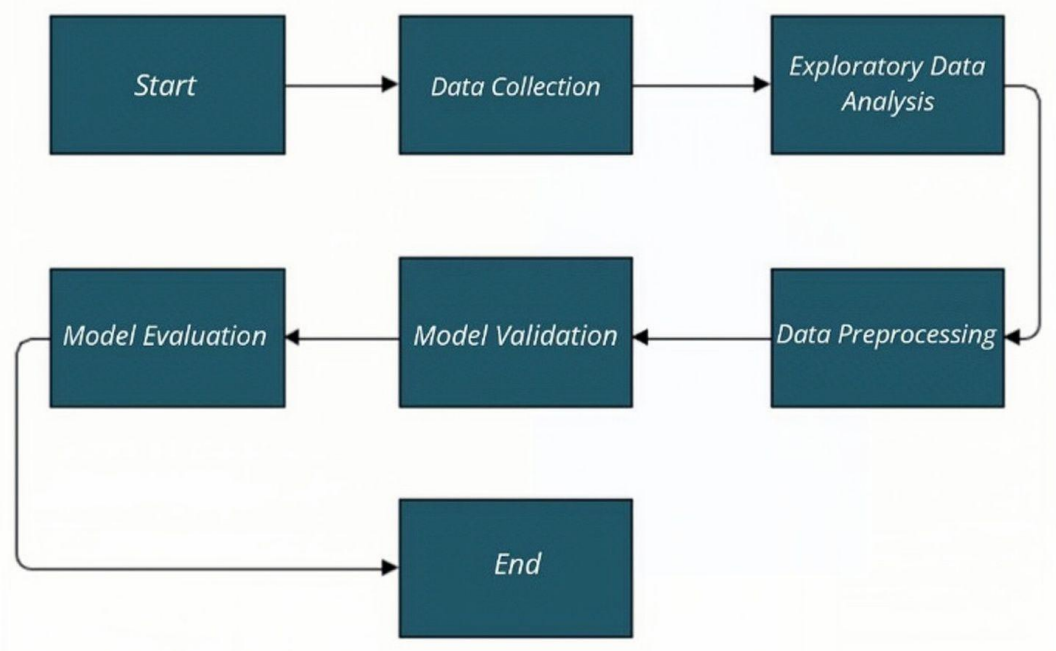
BAB III
METODELOGI PENELITIAN

3.1 Tahapan Penelitian

Dalam penelitian ini dilakukan menggunakan dua algoritma *yaitu Random Forest* dan *Decision Tree* dalam percobaan menggunakan dataset dari *Kaggle.com.* Metode penelitian yang digunakan pada penelitian ini menggunakan proses *data mining*.

Adapun tahapan penelitian yang dilakukan sebagai berikut.



Gambar 3.1 Tahapan Penelitian [7].

3.1.1 *Data Collection*

Penelitian ini menggunakan *dataset* riwayat penyakit resiko terkena penyakit jantung yang diperoleh dari repositori *Kaggle*. Dataset yang digunakan berjumlah 1888 data penderita resiko terkena penyakit jantung dengan variabel berisi *age, sex, cp, trestbps, chol, fbs, restecg, thalachh, exang, oldpeak, slope, ca, thal*, dan target. Data dalam bentuk *comma separated values (csv)*. Pada dataset tersebut terdiri dari 14 atribut.

	age	sex	cp	trestbps	chol	fbs	restecg	thalachh	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1883	60	1	0	140	207	0	0	138	1	1.9	2	1	3	0
1884	46	1	0	140	311	0	1	120	1	1.8	1	2	3	0
1885	59	1	3	134	204	0	1	162	0	0.8	2	2	2	0
1886	54	1	1	154	232	0	0	164	0	0.0	2	1	2	0
1887	53	1	0	110	335	0	1	143	1	3.0	1	1	3	0

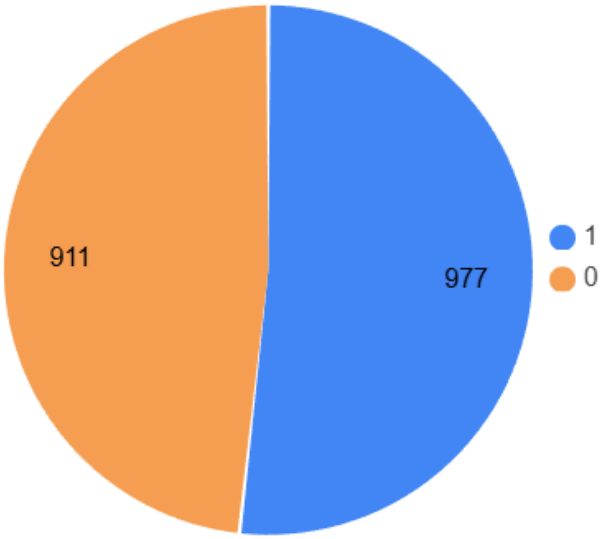
Gambar 3.2 Sampel Data

Tabel 3.1 Metadata Penyakit Jantung

Atribute	*Description*	*Value*	*Type Data*
Age	*Umur pasien dalam tahun*	*0-100*	*Numerical*
Sex	*Jenis Kelamin (1: Laki-laki, 0: Perempuan)*	*0 atau 1*	*Categorical*
Cp	*Nyeri dada dengan nilai 0(tidak ada), 1(sangat ringan), 2(ringan), 3(signifikan), 4(heabt/keras)*	*0;1;2;3;4*	*Categorical*
Trestbps	*Tekanan Darah*	*100-200*	*Numerical*
Chol	*Konsentrasi kolestrol dalam mg/dl*	*200-240*	*Numerical*
Fbs	*Konsentrasi gula setelah puasa dalam 8 jam dengan nilai 1(>*	*0 atau 1*	*Categorical*

	120 mg/dl) dan 0(<120 md/dl)		
Restecg	*Hasil elektrokardiografi istirahat (0= tidak ada kelainan, 1= memiliki kelainan gelombang ST-T, 2 = hipertrofi ventrikel kiri).*	*0;1;2*	*Categorical*
Thalachh	*Detak jantung maksimum yang dicapai.*	*0-200*	*Numerical*
Exang	*Angina yang diinduksi oleh olahrga (1= ya, 0= tidak).*	*0 atau 1*	*Categorical*
Oldpeak	*Depresi ST yang induksi oleh olahraga relatif terhadap istirahat.*		*Numerical*
Slope	*Kemiringan segmen ST pada punccak Latihan (0= upsloping, 1= flat, 2= downsloping).*	*0;1;2*	*Categorical*
Ca	*Jumlah pembuluh darah utama (0-3) yang diwarnai oleh fluoroskopi.*	*0-3*	*Numerical*
Thal	*Jenis-jenis thallasemia (1= normal, 2= cacat tetap, 3= cacat reversibel*	*0;1;2;3*	*Categorical*

Target	*Terkena prediksi penyakit jantung (1= penyakit jantung, 0= tidak memiliki penyakit jantung)*	*0 atau 1*	*Categorical*



Gambar 3.3 Jumlah Penderita dan Bukan Penderita Penyakit Jantung

Berdasarkan visualisasi gambar 3.3 seseorang yang tidak terkena penyakit jantung ditampilkan dengan label 0 berjumlah 911 data dan penderita yang terkena penyakit jantung ditampilkan dengan label 1 berjumlah 977 data.

3.1.2 *Exploratory Data Analysis (EDA)*

Dalam penelitian ini terdiri dalam beberapa langkah pengerjaan yaitu tahapan *analyzing* dataset kemudian dilanjutkan dengan deskripsi dataset menggunakan *EDA (Exploratory Data Analysis)* dengan melakukan EDA, akan sangat berguna dalam mendeteksi kesalahan dari awal, dapat mengidentifikasi outlier, mengetahui hubungan antar data serta dapat menggali faktor-faktor penting dari data. Data yang telah terkumpul akan dianalisis terlebih dahulu untuk mengetahui karakteristik data

yang memiliki nilai yang null, jumlah data dan tipe data yang tidak konsisten dan juga fungsi EDA adalah mengenali kesalahan dataset yang menguasai pola suatu data dan menemukan hubungan antar variabel.

Dalam melakukan analisis data yang terkait dengan resiko terkena penyakit jantung dengan menerapkan *Exploratory Data Analysis* (EDA) seperti teknik visualisasi berikut :

a) Tipe Data

   Proses EDA dimulai dengan memahami struktur dan tipe data yang dimiliki. Yaitu mencakup pemeriksaan bentuk dataset, jenis variabel, (numerik atau kategori) serta melihat ringkasan statistic dari dataset.

b) Missing Values

   Memeriksa apakah terdapat nilai yang hilang atau null, untuk itu pada tahapan ini diperlukan identifikasi *missing values* untuk dapat meningkatkan akurasi kinerja model yang dapat guna meningkatkan keakuratan data.

c) Outlier

   Pada penelitian ini pemeriksaan outlier menggunakan visualisasi boxplot dan *IQR (Inter Quartile Range)* digunakan untuk mendeteksi *outlier* dalam data dengan cara menentukan batas bawah *(lower)* dan batas *(upper)* diluar mana data yang dianggap sebagai *outlier*.

   Mengidentifikasi nilai pada datast apakah logic dan rasional sesuai dengan fakta. Data yang tidak sesuai dengan faktanya biasanya disebut sebagai data outlier atau data yang sangat menyimpang pada sebaran didalam dataset. Data outlier dapat mempengaruhi performa model karena dapat mengakibatkan *overfitting* dan *underfitting*, maka perlu dilakukan pemeriksaan outlier. Data dikatakan sebagai outlier karena data tersebut melebihi nilai maksimum atau jauh dari nilai rata-rata.

   Penanganan outlier dilakukan menggunakan pendekatan IQR dengan metode *winsorizing*, yaitu menyesuaikan nilai-nilai ekstrem agar berada dalam batas normal (tidak menghapus data). Proses ini diimplementasikan

melalui fungsi *Python handle_outliers_iqr*, di mana nilai-nilai diatas batas atas diganti dengan nilai batas atas. Visualisasi boxplot digunakan untuk memverifikasi keberhasilan proses ini secara grafis, memperlihatkan bahwa data berada dalam distribusi yang lebih terkendali setelah penyesuaian. Berikut penjelasan mengenai rumus IQR (*Inter Quartile Range*) yang di terapkan untuk menangani data outlier :

- Q1 dan Q3 adalah kuartil 1 dan kuartil 3 dari kolom yang ditentukan.
- IQR adalah selisih antara Q3 dan Q1, menggambarkan rentang tengah 50% data.
- *Lower_bound* dan *upper_bound* adalah batas bawah dan atas untuk deteksi outlier, berdasarkan rumus :

  Lower Bound= Q1-1.5 x IQR

  Upper Bound= Q3+1.5 x IQR
- *np.clip*() digunakan untuk mengganti nilai outlier yang lebih rendah dari *lower_bound* dengan nilai *lower_bound*, dan nilai yang lebih tinggi dari *upper_bound* menjadi *upper_bound*.

d) *Univariate Analysis*

Pada tahap ini analisis univariate, adalah tahapan untuk langkah memahami distribusi dan karakteristik dari masing-masing variabel dalam dataset.

e) *Multivariate Analysis*

Tahapan menganalisis hubungan antar variabel numerik dalam dataset.

3.1.3 Data Preprocessing

Pada tahapan ini akan dilakukan tahapan persiapan data, yaitu:

a. Standarisasi Data

Standarisasi data adalah bagian dari tahapan transformasi/pra-pemrosesan data yang harus dilakukan sebelum tahap modelling.

Standarisasi data, juga dikenal sebagai normalisasi, Adalah proses mengubah nilai-nilai fitur numerik dalam dataset agar memiliki skala yang sama. Tujuannya a dalah untuk memastikan bahwa setiap fitur berkontribusi secara merata pada model machine learning dan tidak ada fitur yang

mendominasi hanya karena skalanya yang besar, oleh karena itu tahapan standarisasi data ini sangat penting dilakukan dalam tahapan sebelum memasuki tahapan pemodelan.

b. *Spliting* data

Dalam analisis data, pembagian data menjadi data training dan data testing sangat penting karena bertujuan untuk mengevaluasi kinerja model secara objektif dan menghindari overfitting.

Data training digunakan untuk melatih model agar mengenali pola antar fitur dan target, sedangkan Data testing digunakan untuk mengevaluasi kinerja model pada data yang belum pernah dilihat sebelumnya, yaitu menggunakan *confusion matrix.* Proses memisahkan data training dan data testing, yaitu dataset menjadi dua bagian yaitu data latih (training) dan data uji (testing). Data latih digunakan untuk melatih model agar mengenali pola dalam hubungan data. Dalam penelitian ini, proses split ditentukan data *training* 80% dan data *testing* 20%. Langkah ini penting agar model dapat menggeneralisasi dengan baik dan tidak hanya berfokus pada data latih, sehingga menghasilkan prediksi yang akurat saat digunakan. Rasio 80:20 dianggap sebagai titik keseimbangan yang baik. 80% data untuk training adalah jumlah cukup besar bagi algoritma untuk mempelajari pola-pola yang relavan dan kompleks dari data. Semakin banyak data yang digunakan untuk melatih model, semakin baik model tersebut dalam menangkap hubungan antar variabel. Sedangkan, 20% data untuk testing sudah cukup untuk mengukur kinerja model secara objektif tanpa mengorbankan terlalu banyak data yang seharusnya digunakan untuk melatih.

Secara umum, rasio 80:20 adalah standar yang teruji dan efektif untuk dataset berukuran sedang, karena mampu memberikan jumlah data yang cukup untuk proses pembelajaran (training) dan pengujian (testing) secara proporsional.

3.1.4 Model Validation

Validasi model dilakukan untuk mengevaluasi performa algoritma yang digunakan dalam penelitian ini, yaitu *Random Forest* dan *Decision Tree*. Langkah-langkah validasi yang dilakukan adalah sebagai berikut:

a) Training

Proses pelatihan model melibatkan penyesuaian parameter untuk meminimalkan eror dengan menggunakan data pelatihan untuk dapat menemukan pola atau hubungan antara fitur dan jumlah data. Setelah model dilatih, metrik evaluasi digunakan untuk menilai performa model pada pelatihan dan data pengujian.

b) Cross Validation

Tahap ini dilakukan evaluasi terhadap kinerja model menggunakan Teknik 5-fold cross-validation dengan proporsi pembagian data 80-20. Pertama, dataset dibagi menjadilima bagian yang hamper sama besar. Sebanyak 80% data digunakan untuk pelatihan model, sedangkan 20% sisanya digunakan untuk pengujian pada setiap fold. Proses ini dapat diulang sebanyak 5 kali, dengan setiap bagian data menjadi data uji sekali, sementara empat fold lainya digunakan sebagai data pelatihan. Dengan cara ini, model diuji secara menyeluruh pada berbagai bagian dataset, sehingga memberikan Gambaran yang lebih stabil dan mengurangi bias yang mungkin timbul akibat pembagian data.

c) Metrik Evaluasi

- Akurasi: Mengukur persentasi prediksi yang benar terhadap total data testing.
- *Precision*: Tingkat keakuratan model dalam memprediksi kelas positif (prediksi resiko terkena penyakit jantung).
- *Recall*: Kemampuan model dalam mendeteksi seluruh kasus positif.
- *F1-Score*: Kombinasi dari *precision* dan *recall* untuk menangani ketidakseimbangan data.

3.1.5 Model Evaluation

Pada tahapan ini mengevaluasi kinerja model menggunakan metrik yang relevan seperti *accuracy, precision, recall,* dan *F1-score.* Selanjutnya melakukan penilaian model menggunakan *confusion matrix* untuk memastikan performa model dalam memprediksi kelas tertentu.