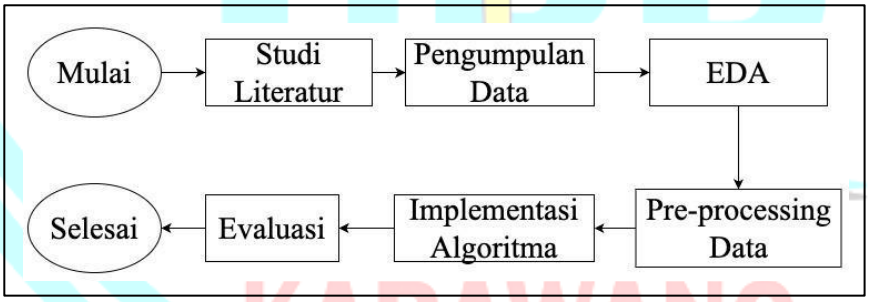
BAB III
METODE PENELITIAN

3.1 Objek Penelitian

Penelitian ini akan difokuskan pada dataset yang berjudul Cardiovascular_Disease_Dataset yang berasal dari link https://data.mendeley.com/datasets/dzz48mvjht/1 dari universitas Lincoln. Data ini berjumlah 1000 baris dan memiliki 14 kolom *variabel* yaitu *patientid, age, gender, chestpain, restingBP, serumcholestrol, fastingbloodsugar, restingrelectro, maxheartrate, exerciseangia, oldpeak, slope, noofmajorvessels* dan *target.*

3.2 Prosedur Penelitian

Penelitian ini diawali dengan tahap identifikasi masalah, studi literatur, pengumpulan data, *preprocessing* dan implementasi klasifikasi dengan Random Forest dan Gradient Boosting. Langkah selanjutnya adalah melakukan evaluasi terhadap algoritma yang digunakan, seperti yang terlihat pada Gambar 3.1.



Gambar 3. 1 Alur Penelitian

3.2.1 Studi Literatur

Pada tahap ini dilakukan pencarian sumber bahan Pustaka yang terkait dengan objek dan topik penelitian. Proses ini mencakup pencarian jurnal, artikel dan sumber informasi lainnya untuk mendalami topik penelitian. Setelah dilakukan studi literatur dapat diambil kesimpulan bahwa masih sedikitnya studi yang membahas penyakit jantung dengan *machine learning* dan belum ada studi yang membahas tentang kasus penyakit jantung dengan algoritma *Gradient Boosting* dan *Random Forest.*

3.2.2 Pengumpulan Data

Setelah mengidentifikasi permasalahan dan melakukan studi literatur, Langkah selanjutnya adalah mengumpulkan, pencarian dan mempersiapkan

dataset. Data yang akan digunakan adalah data dari website Mendeley dengan link https://data.mendeley.com/datasets/dzz48mvjht/1 data tersebut berisi 1000 baris dan 14 fitur kolom, data tersebut diambil pada tanggal 11 November 2024. Untuk rincain dataset bisa dilihat pada Tabel 3.3.

Tabel 1.3 Informasi dataset

Feature Name	*Description*
patientid	*Patient Identification Number*
age	*Age In Years*
gender	*gender (0= female, 1 = male)*
chestpain	*Chest pain type (0: typical angina, 1: atypical angina, 2: non-anginal pain, 3: asymptomatic)*
restingBP	*Resting blood pressure 94-200 (in mm HG)*
serumcholestrol	*Serum cholesterol 126-564 (in mg/dl)*
fastingbloodsugar	*Fasting blood sugar 0,1 > 120 mg/dl (0 = false , 1 = true)*
restingrelectro	*Resting electrocardiogram results 0,1,2 (0: normal, 1: Abnormal ST-T wave, 2 : Possible/definite LVH)*
maxheartrate	*Maximum heart rate achieved (71-202)*
exerciseangia	*Exercise induced angina (0 = no, 1 = yes)*
oldpeak	*Oldpeak = ST (0-6.2)*
slope	*Slope of the peak exercise ST segment (1-upsloping, 2-flat, 3-downsloping)*
noofmajorvessels	*Number of major vessels (0,1,2,3)*
target	*Classification Heart Disease (0= Absence , 1= Presence)*

3.2.3 . Exploratory Data Analysis (EDA)

Pada tahap ini, data dieksplorasi lebih dalam untuk mengungkap karakteristiknya dan memperoleh pemahaman yang lebih menyeluruh. Proses EDA dilakukan dengan memanfaatkan visualisasi data, seperti grafik, diagram, atau bahkan melalui penyajian teks sederhana yang mungkin belum tergambarkan secara jelas dalam dataset.

Sementara itu, berikut ini merupakan penjelasan untuk setiap feature yang ada pada Tabel 3.3 (Doppala et al., 2022).

1. ***patientid***: ID unik setiap pasien.
2. ***age****:* Usia pasien (semakin tua, risikonya lebih tinggi).
3. ***gender***: Jenis kelamin (1 = pria, 0 = wanita).

4. ***chestpain***: Jenis nyeri dada (normal atau tanda gangguan jantung).
5. ***restingBP***: Tekanan darah saat istirahat (tinggi = risiko jantung).
6. ***serumcholestrol***: Kadar kolesterol dalam darah (tinggi = bisa sumbat pembuluh darah).
7. ***fastingbloodsugar***: Gula darah setelah puasa (>120 = berisiko).
8. ***restingrelectro****:* Hasil EKG saat istirahat (lihat aktivitas jantung).
9. ***maxheartrate****:* Detak jantung maksimal saat olahraga.
10. ***exerciseangia***: Nyeri dada saat olahraga (1 = ya, 0 = tidak).
11. ***oldpeak****:* Perbedaan EKG saat olahraga dan istirahat (tinggi = potensi masalah).
12. ***slope***: Bentuk grafik EKG saat olahraga (naik/datar/turun).
13. ***noofmajorvessels****:* Jumlah pembuluh darah besar yang terlihat (lebih banyak = lebih parah).
14. ***target****:* Hasil diagnosis (1 = ada penyakit jantung, 0 = tidak ada).

3.2.4 Preprocessing Data

Pada tahap ini dilakukan beberapa tahapan seperti pemeriksaan *missing value*, pemeriksaan data duplikat, *feature selection*, normalisasi dan *split* data.

1. Pemeriksaan *Missing Value*

Pada tahap ini dataset diperiksa untuk memastikan apakah ada nilai yang hilang (*missing value*) untuk kemudian dilakukan penanganan *missing value*. Dalam dataset kali ini tidak ditemukan *missing value* atau nilai yang hilang.

2. Pemeriksaan Data Duplikat

Dataset diperiksa untuk memastikan tidak ada data yang duplikat sebelum masuk ke tahap selanjutnya. Jika ditemukan data duplikat biasanya akan dilakukan penghapusan dan disisakan satu data saja. Dalam kasus kali ini tidak ditemukan data duplikat dalam dataset.

3. Feature Selection

Langkah berikutnya adalah *feature selection* dengan cara memisahkan antara *variabel y* yaitu kolom target yang akan dilakukan klasifikasi dan *variabel x* yang terdiri dari beberapa kolom selain kolom target. Pemilihan kolom yang akan

digunakan untuk *variabel x* diambil dari 10 kolom yang memiliki relevansi paling tinggi dengan kolom target.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 14 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   patientid           1000 non-null   int64
 1   age                 1000 non-null   int64
 2   gender              1000 non-null   int64
 3   chestpain           1000 non-null   int64
 4   restingBP           1000 non-null   int64
 5   serumcholestrol     1000 non-null   int64
 6   fastingbloodsugar   1000 non-null   int64
 7   restingrelectro     1000 non-null   int64
 8   maxheartrate        1000 non-null   int64
 9   exerciseangia       1000 non-null   int64
 10  oldpeak             1000 non-null   float64
 11  slope               1000 non-null   int64
 12  noofmajorvessels    1000 non-null   int64
 13  target              1000 non-null   int64
dtypes: float64(1), int64(13)
memory usage: 109.5 KB
```

Gambar 3.2 *before feature selection*

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 10 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   gender              1000 non-null   int64
 1   chestpain           1000 non-null   int64
 2   restingBP           1000 non-null   int64
 3   serumcholestrol     1000 non-null   int64
 4   fastingbloodsugar   1000 non-null   int64
 5   restingrelectro     1000 non-null   int64
 6   maxheartrate        1000 non-null   int64
 7   oldpeak             1000 non-null   float64
 8   slope               1000 non-null   int64
 9   noofmajorvessels    1000 non-null   int64
dtypes: float64(1), int64(9)
memory usage: 78.2 KB
```

Gambar 3.3 *after feature selection*

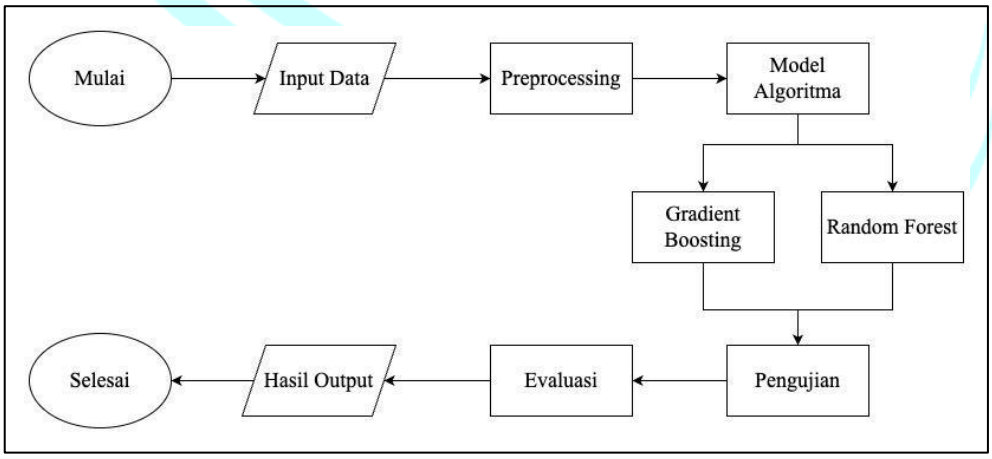
4. Normalisasi Data

Kemudian, dilakukan normalisasi data, di mana data dalam bentuk *numerik* diubah menjadi bentuk *numerik* yang setara agar lebih mudah diolah. Dalam penelitian kali ini yang digunakan adalah StandardScaler, ini teknik yang digunakan untuk mengubah skala data sehingga memiliki distribusi dengan rata-rata 0 dan standar deviasi 1.

5. Split Data

Sebelum masuk ke tahap implementasi algoritma, dataset di bagi menjadi 2 yakni 80% untuk data training dan 20% untuk data testing. Untuk data training berjumlah 800 baris dan data testing berjumlah 200 baris.

3.2.5 Implementasi Algoritma



Gambar 3.4 Flowchart implementasi model

Setelah tahap seleksi data selesai, langkah selanjutnya adalah menerapkan algoritma *Random Forest* dan *Gradient Boosting* untuk melakukan perhitungan, yang dilakukan menggunakan bahasa pemrograman Python. *Random Forest* dan *Gradient Boosting* dipilih karena mendapatkan hasil yang terbaik dibandingkan dengan beberapa algoritma lain dari hasil penelitian sebelumnya. Penjelasan lebih detail mengenai tahapan dan penerapan algoritma *Random Forest* dan *Gradient Boosting* disajikan pada Gambar 3.4.

Setelah tahap seleksi data selesai, langkah selanjutnya adalah menerapkan algoritma *Random Forest* dan *Gradient Boosting* untuk melakukan perhitungan, yang dilakukan menggunakan bahasa pemrograman Python. *Random Forest* dan *Gradient Boosting* dipilih karena mendapatkan hasil yang terbaik dibandingkan dengan beberapa algoritma lain dari hasil penelitian sebelumnya. Penjelasan lebih detail mengenai tahapan dan penerapan algoritma *Random Forest* dan *Gradient Boosting* disajikan pada Gambar 3.4.

3.2.6 Evaluasi

Setelah algoritma *Random Forest* dan *Gradient Boosting* diterapkan, langkah berikutnya adalah mengevaluasi kinerja model yang dihasilkan. Evaluasi dilakukan dengan menggunakan metrik seperti *accuracy*, *precision*, *recall*, dan *F1-score* untuk mengukur seberapa baik model dalam melakukan prediksi berdasarkan data yang telah diproses. Selain itu, evaluasi juga mencakup analisis terhadap *confusion matrix* untuk memahami distribusi kesalahan prediksi pada masing-masing kelas.