BAB III
METODE PENELITIAN

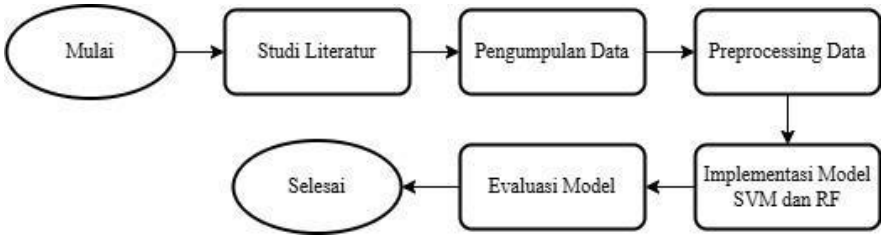
3.1 Objek Penelitian

Objek penelitian ini adalah data rekam medis pasien yang terdiagnosis hipertensi pada tahun 2023 dan 2024, serta pada tahun 2025 hanya mencakup data bulan Januari hingga Februari. Dataset tersebut memuat informasi seperti tanggal pemeriksaan, nik, nama pasien, tanggal lahir, usia, jenis kelamin, provinsi asal pasien, kota/kabupaten asal, alamat, riwayat penyakit pada keluarga/diri sendiri, kebiasaan merokok, kurang aktivitas fisik, konsumsi gula, garam, dan lemak berlebihan, kurang makan buah dan sayur, konsumsi alkohol, hasil pengukuran tekanan darah (sistolik dan diastolik), tinggi badan (cm), berat badan (kg), pemeriksaan gula darah, serta diagnosis medis.

Penelitian ini bertujuan untuk mengembangkan model klasifikasi yang mampu mengidentifikasi hipertensi berdasarkan data rekam medis pasien, serta membandingkan hasil akurasi algoritma *Support Vector Machine (SVM)* dan *Random Forest* untuk menentukan algoritma yang lebih unggul dalam klasifikasi. Proses klasifikasi dilakukan untuk mengelompokkan pasien ke dalam dua kategori, yaitu normal dan hipertensi berdasarkan hasil diagnosis medis yang telah tersedia.

3.2 Prosedur Penelitian

Penelitian ini disusun dengan pendekatan yang terstruktur dan sistematis untuk memastikan bahwa setiap tahapan dapat berjalan secara optimal. Langkah-langkah penelitian dirancang secara berurutan agar saling melengkapi dalam mencapai tujuan yang telah ditentukan. Untuk memberikan penjelasan yang lebih rinci dan jelas, prosedur penelitian ini disajikan dalam bentuk diagram alur *(flowchart)* yang menggambarkan keseluruhan proses secara menyeluruh seperti yang ditunjukkan pada Gambar 3.1.



Gambar 3. 1 *Flowchart* Prosedur Penelitian

3.3 Studi Literatur

Studi literatur ini bertujuan untuk mempelajari dan menganalisis penerapan algoritma *Support Vector Machine (SVM)* dan *Random Forest* dalam klasifikasi penyakit hipertensi, dengan meninjau berbagai penelitian yang telah dilakukan sebelumnya untuk menilai performa, akurasi, dan keunggulan masing-masing algoritma dalam mengidentifikasi pasien hipertensi berdasarkan data medis dan faktor risiko yang relevan.

3.4 Pengumpulan Data

Data dalam penelitian ini diperoleh dari rekam medis pasien di salah satu puskesmas yang menjadi lokasi studi. Pengumpulan data dilakukan pada tanggal 18 November 2024 dengan izin dan koordinasi bersama pihak puskesmas, khususnya bagian pelayanan medis dan pengelola data. Total data yang berhasil dikumpulkan berjumlah 3.230 entri pasien, yang mencakup periode tahun 2023, 2024, serta pada tahun 2025 hanya di bulan Januari hingga Februari. Dataset tersebut terdiri dari pasien dengan berbagai kategori diagnosis, yaitu hipertensi, diabetes, dan normal. Informasi yang terdapat dalam dataset meliputi tanggal pemeriksaan, nik, nama pasien, tanggal lahir, usia, jenis kelamin, provinsi dan kota/kabupaten asal pasien, alamat, riwayat penyakit pada keluarga/diri sendiri, kebiasaan merokok, kurang aktivitas fisik, konsumsi gula, garam, dan lemak berlebihan, kurang konsumsi buah dan sayur, konsumsi alkohol, hasil pengukuran tekanan darah (sistolik dan diastolik), tinggi badan (cm), berat badan (kg), hasil pemeriksaan gula darah, serta diagnosis medis. Seluruh data disusun dalam format *Excel* untuk memudahkan analisis dan pengolahan lebih lanjut.

3.5 *Preprocessing* Data

Tahapan *preprocessing* data pada penelitian ini bertujuan untuk memastikan bahwa dataset yang digunakan dalam analisis klasifikasi hipertensi berada dalam kondisi optimal. Proses ini mencakup beberapa langkah, mulai mengatasi nilai yang kosong *(missing values)*, menghapus data duplikat, *transformasi* data, normalisasi data, pembagian data, dan pemilihan fitur yang relevan agar model dapat bekerja secara efisien dan akurat. Berikut ini adalah beberapa langkah dalam *preprocessing* data:

1. Mengatasi *Missing Values*

Mengatasi *missing values* dilakukan untuk mengidentifikasi kolom yang memiliki nilai kosong *(missing values)*. Jika kolom penting, isi nilai kosong dengan metode imputasi, contohnya mean/median untuk kolom numerik. Modus (nilai yang paling sering muncul) untuk kolom kategorikal. Kemudian jika kolom tidak relevan, maka dapat dihapus. Contoh cara mengatasi *missing values* ditunjukkan pada Tabel 3.1 sebelum mengatasi *missing values* dan Tabel 3.2 setelah mengatasi *missing values*.

Tabel 3. 1 Data Awal Sebelum Mengatasi Missing Values

Data Awal	
Nama Pasien	**Riwayat Penyakit Pada Keluarga/Diri Sendiri**
Kaspi	Penyakit Diabetes
Amah Maryamah	Penyakit Diabetes
Dadang	NaN
Emis	Penyakit Hipertensi
Yati	NaN
Carti	Penyakit Hipertensi

Tabel 3. 2 Data Setelah Mengatasi *Missing Values*

Data Setelah Mengatasi *Missing Values*	
Nama Pasien	**Riwayat Penyakit Pada Keluarga/Diri Sendiri**

Kaspi	Penyakit Diabetes
Amah Maryamah	Penyakit Diabetes
Dadang	Tidak Ada
Emis	Penyakit Hipertensi
Yati	Tidak Ada
Carti	Penyakit Hipertensi

2. Menghapus Data Duplikat

Menghapus data duplikat dilakukan untuk meningkatkan akurasi, dan kualitas data dengan memastikan bahwa setiap entri yang disimpan bersifat unik dan tidak berulang. Data duplikat ini dilakukan dengan mencari data pasien yang tercatat lebih dari satu kali, kemudian hapus data duplikat atau gabungkan data jika informasi tambahan tersedia. Contoh cara hapus data duplikat ditunjukkan pada Tabel 3.3 sebelum hapus duplikat dan Tabel 3.4 setelah hapus duplikat.

Tabel 3. 3 Data Awal Sebelum Hapus Duplikat

Data Awal	
Nama Pasien	**Jenis Kelamin**
Kaspi	Laki-Laki
Kaspi	Laki-Laki
Amah Maryamah	Perempuan
Dadang	Laki-Laki
Emis	Perempuan
Yati	Perempuan

Tabel 3. 4 Data Setelah Hapus Duplikat

Data Setelah di Hapus	
Nama Pasien	**Jenis Kelamin**
Kaspi	Laki-Laki
Amah Maryamah	Perempuan
Dadang	Laki-Laki
Emis	Perempuan

Yati	Perempuan

3. *Transformasi* Data

Transformasi data dilakukan untuk mengubah format, skala, atau distribusi data agar lebih sesuai untuk analisis atau pemodelan. Dalam penelitian ini, *transformasi* data dilakukan dengan menerapkan *label encoding* untuk mengonversi variabel kategorikal menjadi format numerik. Contoh *transformasi* data menggunakan *label encoding* ditunjukkan pada Tabel 3.5.

Tabel 3. 5 *Label Encoding*

Kategori	**Nilai Asli**	***Label Encoding***
Jenis Kelamin	Laki-laki	0
	Perempuan	1
Penyakit Pada Keluarga/Diri Sendiri	Penyakit Asma	0
	Penyakit Diabetes	1
	Penyakit Hipertensi	2
	Penyakit Jantung	3
	Penyakit Kolestrol	4
	Penyakit Stroke	5
	Tidak Ada	6
Merokok	Tidak	0
	Ya	1
Kurang Aktivitas Fisik	Tidak	0
	Ya	1
Gula Berlebihan	Tidak	0
	Ya	1
Garam Berlebihan	Tidak	0
	Ya	1
Lemak Berlebihan	Tidak	0
	Ya	1
Kurang Makan Buah dan Sayur	Tidak	0
	Ya	1
Konsumsi Alkohol	Tidak	0
	Ya	1
Diagnosis	Normal	0
	Hipertensi	1

4. Normalisasi Data

Normalisasi data merupakan salah satu tahap penting dalam *preprocessing* yang bertujuan untuk menyamakan skala antar fitur numerik antara 0 sampai 1, agar seluruh variabel memiliki kontribusi yang seimbang dalam proses pelatihan model.

Dalam penelitian ini, normalisasi dilakukan menggunakan metode *Min-Max*, yang mengubah nilai setiap fitur ke dalam skala yang seragam. Proses ini diterapkan pada fitur usia, tekanan darah sistolik (sistol), tekanan darah diastolik (diastol), tinggi badan (cm), dan berat badan (kg). Karena setiap fitur memiliki rentang nilai yang berbeda, normalisasi diperlukan agar tidak ada fitur yang mendominasi model hanya karena memiliki skala angka yang lebih besar.

5. Pembagian Data

Setelah data melalui tahapan *preprocessing*, data dibagi menjadi dua bagian: 70% data digunakan untuk pelatihan *(training)* dan 30% digunakan untuk pengujian *(testing)*. Sebagian data digunakan untuk melatih model, sedangkan data yang tersisa digunakan untuk menilai kinerja model yang dilatih

Langkah-langkah Pembagian Data:

1. Mulai
2. Persiapkan Dataset yang Akan Dibagi:
   Fitur (X) untuk data input
   Label (Y) untuk data output
3. Tentukan Proporsi Pembagian Data:
   Contoh proporsi: 70% untuk data pelatihan *(training)* dan 30% untuk data pengujian *(testing)*.
4. Hitung Jumlah Data untuk Setiap Bagian:
   Jumlah total data = total baris dalam dataset.
   Data Pelatihan *(training)* = 70% dari total data.
   Data Pengujian (*testing*) = 30% dari total data.
5. Acak Urutan Baris Data:
   Lakukan pengacakan pada baris dataset agar distribusi data lebih merata dan tidak mengikuti pola tertentu.
6. Bagi Dataset:
   Data Pelatihan *(training)*: Ambil 70% dari dataset yang telah diacak.
   Data Pengujian (*testing*): Sisakan 30% dari dataset.
7. Pisahkan Fitur dan Label:
   Data Pelatihan: Fitur *(X_train, Label (y_train)*
   Data Pengujian: Fitur *(X_test), Label (y_test)*

8. Output Hasil Pembagian:
   Data Pelatihan: *(X_train, y_train)*
   Data Pengujian: *(X_test, y_test)*
9. Tampilkan Data Pembagian:
   Data *training* (70%)
   Data *testing* (30%)
10. Selesai

6. Pemilihan Fitur

Pemilihan fitur dilakukan untuk memilih atribut yang paling relevan terhadap target klasifikasi. Dalam penelitian ini, digunakan visualisasi *heatmap* korelasi untuk mengidentifikasi hubungan antar fitur numerik dan target. *Heatmap* memudahkan dalam melihat fitur mana yang memiliki korelasi tinggi terhadap status hipertensi, seperti tekanan darah sistolik, diastolik, dan usia. Fitur dengan korelasi rendah atau redundan dapat dihilangkan agar model lebih optimal. Pemilihan fitur berbasis korelasi ini bertujuan untuk meningkatkan kinerja dan akurasi algoritma klasifikasi seperti SVM dan *Random Forest*.

3.6 Implementasi Model SVM dan RF

Implementasi model *Support Vector Machine (SVM)* dan *Random Forest* adalah langkah penting dalam klasifikasi penyakit hipertensi. Kedua algoritma ini memiliki pendekatan yang berbeda dalam memproses data.

3.6.1 *Support Vector Machine (SVM)*

Pada Penelitian ini menggunakan algoritma SVM dengan kernel *Radial Basis Function (RBF)* dengan nilai *C=1* dan *gamma='scale'*. Model dibangun dengan menggunakan data pelatihan yang telah melalui tahap *preprocessing*, kemudian kinerjanya dievaluasi menggunakan data pengujian. Adapun implementasi model klasifikasi SVM ditunjukkan pada Tabel 3.6.

Tabel 3. 6 Membuat Model Klasifikasi dengan SVM

Klasifikasi dengan Algoritma SVM
Input : Dataset *Training*
Output : Hasil prediksi model SVM pada data pengujian

1. Baca Dataset
2. Menyiapkan dataset pelatihan dan pengujian
3. Tentukan model yang akan digunakan dengan kernel (RBF)
4. Tentukan nilai *hyperparameter* (C = 1 dan gamma = ‘scale’)
5. Melatih model SVM menggunakan data pelatihan untuk mempelajari pola dan membuat *hyperplane* pemisah antar kelas
6. Prediksi label (Normal atau Hipertensi) pada data pengujian menggunakan model yang telah dilatih

3.6.2 *Random Forest*

Pada penelitian ini digunakan algoritma Random Forest dengan jumlah pohon keputusan *(n_estimators)* sebanyak 100 dan pengaturan *random_state = 42* untuk memastikan hasil yang konsisten. Model ini juga dibangun menggunakan data pelatihan yang telah melalui tahap *preprocessing*, kemudian kinerjanya dievaluasi menggunakan data pengujian. Adapun implementasi model klasifikasi *Random Forest* ditunjukkan pada Tabel 3.7.

Tabel 3. 7 Membuat Model Klasifikasi dengan *Random Forest*

Klasifikasi dengan Algoritma *Random Forest*
Input : Dataset *Training* Output : Hasil prediksi model *Random Forest* pada data pengujian 1. Baca dataset 2. Menyiapkan dataset pelatihan dan pengujian 3. Membuat objek menggunakan *classifier* dengan *hyperparameter* (*n_estimators = 100* dan *random_state = 42*) 4. Melatih model menggunakan data pelatihan untuk membentuk sejumlah pohon keputusan 5. Prediksi label (Normal atau Hipertensi) pada data pengujian menggunakan model yang telah dilatih

3.7 Evaluasi Model

Evaluasi model dalam penelitian ini dilakukan menggunakan *confusion matrix* untuk mengukur kemampuan model dalam memprediksi klasifikasi hipertensi. *Confusion matrix* memberikan penjelasan yang lebih rinci mengenai cara model mengelompokkan data ke dalam masing-masing kategori, yaitu normal dan hipertensi. *Confusion matrix* menunjukkan jumlah prediksi yang benar dan salah, yang dibagi menjadi empat bagian: *True Positive*, *True Negative*, *False Positive*, dan *False Negative*. Dari matriks ini dihitung metrik evaluasi seperti akurasi, presisi, *recall*, dan *f1-score*. Evaluasi dilakukan pada dua model, yaitu *Support Vector Machine (SVM)* dan *Random Forest*, dengan menggunakan data uji yang telah dipreproses. Hasil evaluasi ini digunakan untuk mengetahui model mana yang lebih baik dalam mengklasifikasikan pasien yang terdiagnosis hipertensi dan normal.

UBP KARAWANG