

## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Objek dan Subjek Penelitian

Objek pada penelitian ini adalah penyakit ginjal kronis, dengan subjek penelitian algoritma *K-Nearest Neighbor* berbasis *AdaBoost* dan penelitian dilakukan secara langsung, mengambil data publik di *UCI Machine Learning Repository*.

#### 3.2 Alat dan Bahan Penelitian

Berdasarkan objek dan subjek, berikut adalah alat dan bahan yang digunakan pada penelitian.

##### 3.2.1 Perangkat Lunak

Penggunaan perangkat lunak pada penelitian ini, dapat dilihat pada Tabel 3.1.

Tabel 3. 1 Perangkat Lunak dalam Penelitian

Perangkat Lunak	Kegunaan
<i>Windows 11</i>	Sistem operasi.
<i>Rapidminer Studio, Version 9.10</i>	<i>Software</i> yang menjalankan operator dengan prinsip data mining.
<i>Microsoft Excel</i>	Program dari paket <i>Microsoft Office</i> , mengolah angka menggunakan <i>spreadsheet</i> yang terdiri dari baris dan kolom untuk mengeksekusi perintah.

##### 3.2.2 Perangkat Keras

Perangkat keras pada penelitian ini, dapat dilihat dalam bentuk Tabel 3.2.

Tabel 3. 2 Perangkat Keras dalam Penelitian

Perangkat Keras	Tipe/Ukuran
<i>Processor</i>	AMD <i>Ryzen 5 3500U, Radeon Vega Mobile Gfx, 2.10 GHz.</i>
<i>Display</i>	14 <i>inch</i>
<i>Memory</i>	8 GB RAM
<i>System type</i>	64-bit <i>operating system</i>

### 3.2.3 Bahan Penelitian

Bahan penelitian atau dataset yang digunakan dalam penelitian ini, terdiri dari 24 atribut dan satu *class*. Setiap atribut memiliki tipe atribut dan keterangan berdasarkan data yang diperoleh. Data awal dilampirkan pada lampiran 1. Berikut Tabel adalah atribut, keterangan dan tipe atribut.

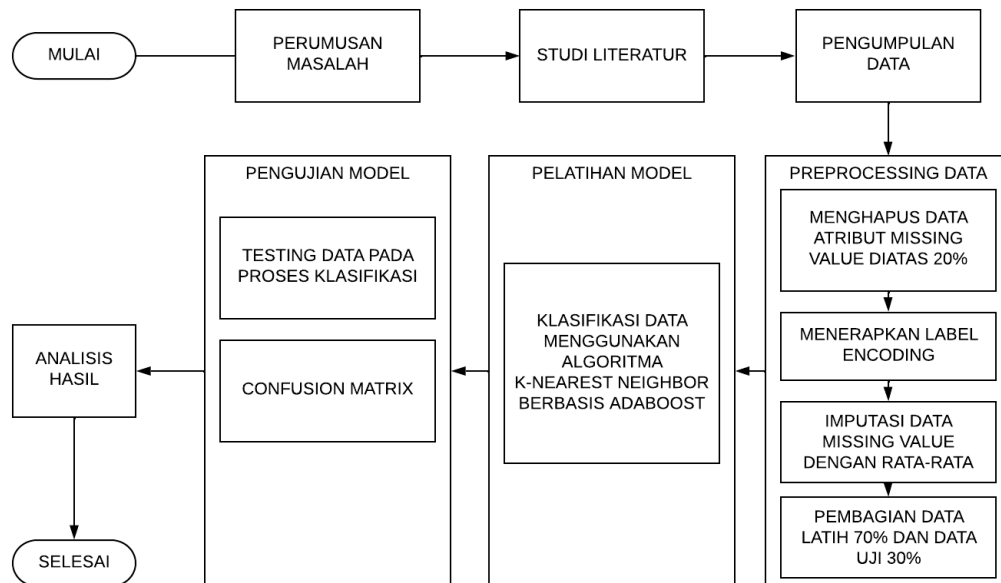
Tabel 3. 3 Dataset Berdasarkan Atribut, Keterangan dan Tipe Atribut

No.	Atribut	Deskripsi Atribut	Tipe Atribut
1.	<i>Age</i>	Usia pasien penderita penyakit ginjal kronis paling muda adalah usia 2 tahun dan paling tua adalah 82 tahun	Numerik
2.	<i>Blood Pressure (BP)</i>	<i>mm/Hg</i>	Numerik
3.	<i>Specific Gravity (SG)</i>	1.005, 1.010, 1.015, 1.020, 1.025	Nominal
4.	<i>Albumin (AL)</i>	0, 1, 2, 3, 4, 5	Nominal
5.	<i>Sugar (SU)</i>	0, 1, 2, 3, 4, 5	Nominal
6.	<i>Red Blood Cells (RBC)</i>	<i>Normal, abnormal</i>	Nominal
7.	<i>Pus Cell (PC)</i>	<i>Normal, abnormal</i>	Nominal
8.	<i>Puss Cell Clumps (PCC)</i>	<i>Present, notpresent</i>	Nominal
9.	<i>Bacteria (BA)</i>	<i>Present, notpresent</i>	Nominal

No.	Atribut	Deskripsi Atribut	Tipe Atribut
10.	<i>Blood Glucose Random (BGR)</i>	<i>mg/dl</i>	Numerik
11.	<i>Blood Urea (BU)</i>	<i>mg/dl</i>	Numerik
12.	<i>Serum Creatinine (SC)</i>	<i>mg/dl</i>	Numerik
13.	<i>Sodium (SOD)</i>	<i>mmol/L</i>	Numerik
14.	<i>Potassium (POT)</i>	<i>mEq/L</i>	Numerik
15.	<i>Hemoglobin (HEMO)</i>	<i>g/dl</i>	Numerik
16.	<i>Packed Cell Volume (PCV)</i>	Angka terendah adalah 14 dan angka tertinggi adalah 54	Numerik
17.	<i>White Blood Cell Count (WC)</i>	<i>cells/cumm</i>	Numerik
18.	<i>Red Blood Cell Count (RC)</i>	<i>millions/cmm</i>	Numerik
19.	<i>Hypertension (HTN)</i>	<i>Yes, no</i>	Nominal
20.	<i>Diabetes Mellitus (DM)</i>	<i>Yes, no</i>	Nominal
21.	<i>Coronary Artery Disease (CAD)</i>	<i>Yes, no</i>	Nominal
22.	<i>Appetite (APPET)</i>	<i>Good, poor</i>	Nominal
23.	<i>Pedal Edema (PE)</i>	<i>Good, poor</i>	Nominal
24.	<i>Anemia (ANE)</i>	<i>Yes, no</i>	Nominal
25.	<i>Class</i>	<i>Ckd, notckd</i>	Nominal

### 3.3 Diagram Alir Penelitian

Penerapan proses penelitian dimulai dari perumusan masalah, studi literatur, pengumpulan data, *preprocessing* data, pelatihan model, pengujian model hingga analisis hasil. Gambar 3.1 berikut adalah proses penelitian.



Gambar 3. 1 Proses Penelitian

#### 3.3.1 Perumusan masalah

Pada fase ini, peneliti mengetahui penggunaan metode melalui permasalahan yang diturunkan dari latar belakang. Langkah yang tepat adalah mengadopsi algoritma *K-Nearest Neighbor* berbasis *AdaBoost* untuk klasifikasi penyakit ginjal kronis. Mengetahui hasil *precision*, *recall* dan *accuracy* penerapan algoritma *K-Nearest Neighbor* berbasis *AdaBoost*.

#### 3.3.2 Studi literatur

Fase dalam mencari dan mengidentifikasi jurnal yang sesuai dengan penelitian, merupakan fase studi literatur. Memanfaatkan referensi dari jurnal, mengambil *record* dari *UCI Repository of Machine Learning Databases* dalam kebutuhan penelitian ini. Jurnal penelitian dan materi penerapan metode diambil berdasarkan penelitian sebelumnya terkait implementasi algoritma *K-Nearest-*

*Nearest* berbasis *AdaBoost*. Sebelum fase *preprocessing* data dan penerapan metode.

### 3.3.3 Pengumpulan data

Pengambilan data dilakukan secara langsung, melalui dataset *public* dari *UCI Repository of Machine Learning Databases*. Memanfaatkan rekam data *chronic kidney disease (ckd)* Juli 2015. Mengambil 24 atribut (*age, blood pressure, specific gravity, albumin, sugar, red blood cells, pus cell, pus cell clumps, bacteria, blood glucose random, blood urea, serum creatinine, sodium, potassium, hemoglobin, packed cell volume, white blood cell count, red blood cell count, hypertension, diabetes mellitus, coronary artery disease, appetite, pedal edema, anemia* dan memiliki kelas (*ckd* dan *notckd*). Pada masing-masing atribut dan kelas memiliki 400 data. Atribut dan kelas adalah rekomendasi dalam kumpulan dataset. Data tidak berupa *spreadsheet* atau file *Excel*, tetapi data disimpan dalam file *notepad* dan tanda koma untuk mengatur urutan atribut data, dalam hal ini data diolah berbentuk file *Excel* untuk diproses menggunakan alat *Rapidminer*.

### 3.3.4 *Preprocessing data*

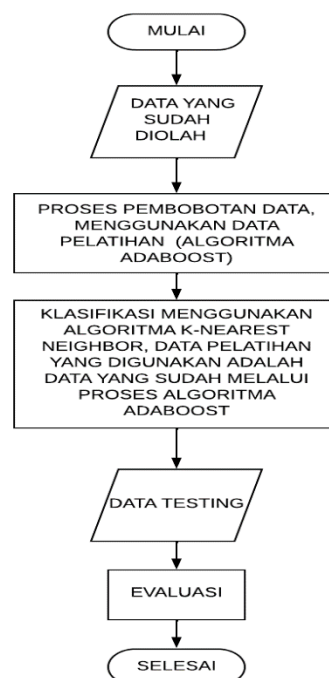
Fase pengolahan data memiliki empat tahap pra-pemrosesan, sebelum melakukan tahap pelatihan model. Pertama, menghitung persentase data *missing value* pada atribut, untuk menindaklanjuti penggunaan atribut yang digunakan. Kedua, menerapkan teknik *label encoding*, tipe data nominal diubah menjadi tipe data numerik yaitu 0 dan 1. Ketiga, tahap imputasi data dengan menetapkan nilai rata – rata pada atribut, dengan menggunakan operator yang ada pada *Rapidminer*. Keempat, membagi data dengan rasio 70% data latih dan 30% pada data uji.

Setelah melakukan perhitungan persentase, atribut yang digunakan pada penelitian ini adalah *age, blood pressure, specific gravity, albumin, sugar, pus cell, pus cell clumps, bacteria, blood glucose random, blood urea, serum creatinine, hemoglobin, packed cell volume, hypertension, diabetes mellitus, coronary artery disease, appetite, pedal edema, anemia* dan memiliki *class*. Pada perubahan tipe nominal data atribut, yang menjadi tipe numerik adalah *Specific gravity, Albumin,*

*Sugar, Pus cell, Pus cell clumps, Bacteria, Hypertension, Diabetes mellitus, Coronary artery disease, Appetite, Pedal edema, dan Anemia.*

### 3.3.5 Pelatihan model

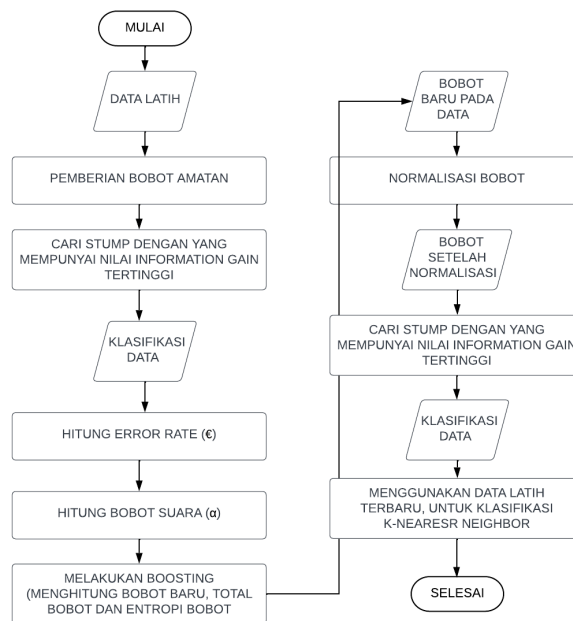
Dalam fase pelatihan model, peneliti mengenakan dua metode klasifikasi: algoritma *K-Nearest-Neighbor* dan algoritma *K-Nearest-Neighbor* berbasis *AdaBoost*. Parameter *k* yang digunakan adalah 3, 5, 7 Algoritma menggunakan *split data* untuk pemisahan antara pelatihan data dan pengujian data. Operator yang dijalankan oleh perangkat *Rapidminer* bekerja secara otomatis untuk mencari nilai perhitungan jarak *euclidean* pada proses perhitungan algoritma *K-Nearest Neighbor* berbasis *AdaBoost*. Sama dengan perbaikan bobot, operator pada perangkat *Rapidminer* bekerja secara otomatis. Pada proses perhitungan manual pembobotan data atau proses perhitungan *AdaBoost*, adanya perhitungan nilai *entropy* dan *information gain* dalam proses perhitungan algoritma *AdaBoost*. Pemodelan operator dengan alat *Rapidminer* ditunjukkan dengan *flowchart* pada proses klasifikasi *K-Nearest Neighbor* berbasis *AdaBoost* sesuai Gambar 3.2.



Gambar 3. 2 Proses Klasifikasi Menggunakan Algoritma K-NN berbasis *AdaBoost*

Berikut langkah-langkah proses klasifikasi berdasarkan pada Gambar 3.2:

1. Menggunakan data yang sudah diolah pada tahap *preprocessing*.
2. Tahap selanjutnya melakukan proses perbaikan bobot pada data pelatihan yang akan digunakan pada klasifikasi *K-Nearest Neighbor*.
3. Melakukan klasifikasi *K-Nearest Neighbor* dengan data latih yang sudah melalui proses metode *AdaBoost*.
4. Data testing yang digunakan adalah data yang sudah diolah pada tahap *preprocessing*.
5. Dalam tahap evaluasi, menguji proses klasifikasi dengan menggunakan *confusion matrix*, untuk mendapatkan nilai *accuracy*, *recall* dan *precision*.



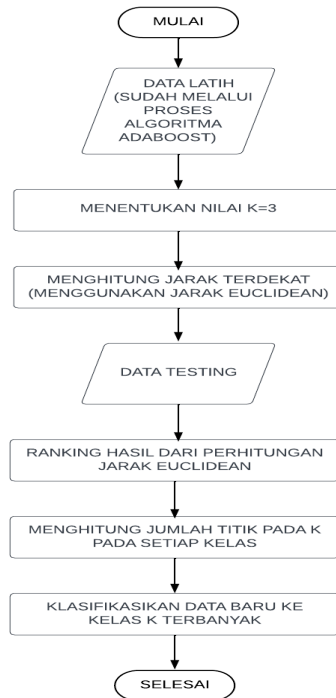
Gambar 3. 3 Proses Algoritma *AdaBoost*

Berikut langkah-langkah alur proses algoritma *AdaBoost* berdasarkan pada Gambar 3.3:

1. Menyiapkan data latih yang akan diolah.
2. Memberikan bobot amatan awal.

3. Mencari stump dengan mencari nilai *information gain* tertinggi, untuk mendapatkan nilai *information gain*, terlebih dahulu menghitung nilai entropi.
4. Nilai tertinggi *information gain* dari atribut, akan dijadikan acuan untuk mengklasifikasikan data.
5. Adanya *error* pada data klasifikasi, akan dilakakun perhitungan *error rate*.
6. Dari hasil *error* dapat dihitung bobot suara ( $\alpha$ ).
7. Melakukan *boosting*, menghitung bobot baru, total bobot dan entropi bobot.
8. Mendapatkan bobot baru pada data.
9. Menghitung keseluruhan bobot yang baru. Nilai bobot harus sama dengan 1, nilai bobot tidak sama dengan 1 akan dinormalisasi, untuk mendapatkan total bobot sama dengan 1.
10. Mendapatkan bobot data setelah dinormalisasi.
11. Mencari nilai stump kembali.
12. Mendapatkan data klasifikasi dari nilai tertinggi *information gain*.
13. Data dengan bobot sama dengan satu atau tidak adanya *error*, selanjutnya data latih tersebut digunakan pada klasifikasi *K-Nearest Neighbor*.





Gambar 3. 4 Proses Algoritma *K-Nearest Neighbor*

Berikut langkah-langkah alur proses algoritma *K-Nearest Neighbor* berdasarkan pada Gambar 3.4:

1. Menyiapkan data latih.
2. Tahap menentukan nilai k.
3. Menghitung jarak terdekat dapat menggunakan jarak *Euclidean*.
4. Meranking hasil perhitungan *Euclidean*.
5. Tahap selanjutnya menghitung jumlah titik pada k untuk setiap kelas.
6. Terakhir mengklasifikasikan data baru ke kelas k terbanyak.

### 3.3.6 Pengujian model

Pada termin ini adalah termin akhir untuk menerima *output accuracy*, *recall* dan *precision* berdasarkan klasifikasi *K-Nearest Neighbor* berbasis *AdaBoost*. Kemudian pada tahap analisis *output*, nilai klasifikasi yang dihasilkan memiliki

hasil yang akurat atau tidak. Representasi nilai *accuracy*, *recall* dan *precision* dengan metode *confusion matrix*.

### 3.3.7 Analisis hasil

Pada langkah ini, klasifikasi dianalisis. Proses klasifikasi *K-Nearest Neighbor* berbasis *AdaBoost* mengumpulkan skor *accuracy*, *recall* dan *precision*. Berdasarkan nilai parameter *k* dalam bentuk tabel dan mengilustrasikannya dengan visualisasi *bar chart*. Kemudian dapat dianalisis apakah hasil dari metode klasifikasi menggunakan algoritma *K-Nearest Neighbor* berbasis *AdaBoost* memiliki nilai akurasi yang akurat. Performa yang tinggi menunjukkan fungsi yang akurat.