

BAB III METODE PENELITIAN

3.1. Objek Penelitian

Pada Penelitian ini objek yang akan diambil adalah SMK PGRI 2 Karawang, dengan mengambil data nilai rapor siswa kelas XII jurusan Teknik Komputer dan Jaringan tahun pelajaran 2020-2021. Data yang diambil merupakan nilai dari semester 1 sampai dengan semester 6 meliputi semua mata pelajaran.

Penelitian ini dilaksanakan di Kampus utama SMK PGRI 2 Karawang di Jalan Pangkal Perjuangan/By Pass Paracis Kelurahan Tanjungpura Kecamatan Karawang Barat Kabupaten Karawang. Adapun waktu pelaksanaan penelitian ini dimulai dari bulan Oktober 2021. Perincian waktu penelitian terdapat pada Tabel 3.1.

Tabel 3.1 Perinci Kegiatan

No	Kegiatan	2021				2022				2022				2022							
		Okt				Feb				Mar				April				Mei			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Studi literatur																				
2	Pengumpulan data																				
3	Seleksi data																				
4	Perhitungan manual Perhitungan																				
5	Rapidminer dan program Python																				
6	Evaluasi																				
7	Kesimpulan																				

3.2. Peralatan Penelitian

Peralatan yang digunakan pada penelitian ini meliputi perangkat keras dan perangkat lunak dengan rincian sebagai berikut :

1. Perangkat keras
 - a. Laptop HP 250 G5 *Notebook* PC dengan spesifikasi sebagai berikut :

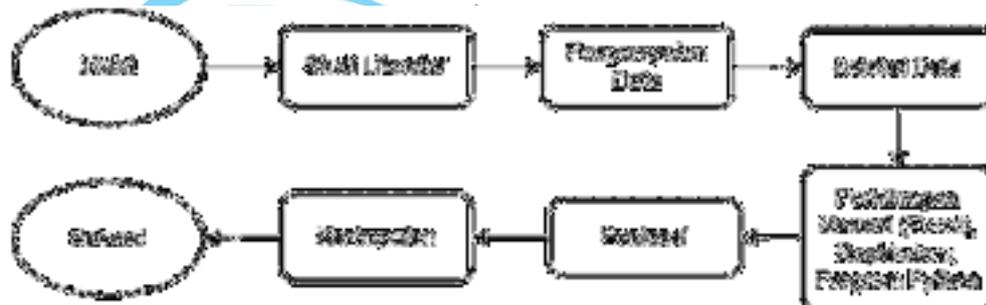
- Processor Intel(R) Core(TM) i3-6006U
- RAM 4 GB
- 64-bit *Operating System*, x64-based processor

2. Perangkat lunak

- a. *Microsoft Office Word 2019*
- b. *Microsoft Office Excel 2019*
- c. *RapidMiner Studio Version 9.10*
- d. *Python*
- e. *Google Colab*

3.3. Prosedur Penelitian

Tahapan prosedur penelitian dapat dilihat pada gambar 3.1.



Gambar 3.1 Prosedur Penelitian

1.3.1. Studi Literatur

Peneliti melakukan studi literatur dengan mempelajari buku dan jurnal karya ilmiah yang berkaitan dengan teknik data mining dengan konsep clustering menggunakan algoritma *K-Means* dan *Fuzzy C-Means* yang bersumber di internet dan perpustakaan.

1.3.2. Pengumpulan data

Dalam pengumpulan data, penulis mendatangi pihak sekolah dan data yang didapat merupakan data nilai rapor siswa kelas XII jurusan Teknik Komputer dan Jaringan SMK PGRI 2 Karawang tahun pelajaran 2020-2021. Data berupa *softfile* dengan jumlah 105 siswa.

1.3.3. Seleksi data

Seleksi data ialah proses penyeleksian data dari atribut yang tidak diperlukan. Setelah tahap ini selesai maka akan menghasilkan dataset yang akan digunakan pada perhitungan manual dan penerapan algoritma *K-Means Clustering* pada aplikasi *RapidMiner* dan *Python*.

1.3.4. Perhitungan

1.3.4.1. Manual (Excel)

Pada proses perhitungan manual penulis menggunakan aplikasi *Microsoft Excel* 2019 dan mengimplementasikan algoritma *K-Means* dan *Fuzzy C-Means* dalam melakukan *clustering*. Data akan dikelompokkan menjadi 3 yaitu berprestasi, prestasi sedang dan tidak berprestasi.

a. Implementasi algoritma *K-Means*

Langkah-langkah dalam algoritma *K-Means Clustering* (AE Pramitasari & Yessica Nataliani, 2021) sebagai berikut :

1. Memasukan data
2. Tentukan jumlah *cluster*
3. Ambil sembarang data sebanyak *cluster* secara acak sebagai pusat *cluster* (*centroid*)
4. Hitung jarak antara data dengan *centroid*, dengan menggunakan persamaan *Euclidean distance* :

$$D_{ij} = \sqrt{(x_{1j} - v_{1j})^2 + \dots + (x_{ij} - v_{kj})^2}$$

Dimana:

D_{ij} = jarak data i ke pusat cluster ke k

x_{ij} = data ke i pada atribut ke j

v_{kj} = titik pusat ke k pada atribut ke j

5. Hitung kembali pusat *cluster* (*centroid*) dengan keanggotaan *cluster* yang baru, berikut rumus untuk menghitung *centroid* :

$$C_1 = \frac{1}{M} \sum_{j=1}^M x_j$$

Jika pusat *cluster* tidak berubah maka proses *cluster* telah selesai, jika belum maka ulangi langkah ke 4 sampai pusat *cluster* tidak berubah lagi.

b. Implementasi algoritma *Fuzzy C-Means*

Langkah-langkah dalam algoritma FCM (Agustina & Prihandoko, 2018) sebagai berikut :

1. Memasukan data yang akan dikelompokan X, berupa matriks berukuran $n \times m$
2. Tentukan :
 - a. Jumlah *cluster* = c
 - b. Pangkat = w
 - c. Maksimum iterasi = Max Iter
 - d. *Error* terkecil yang diharapkan = ϵ
 - e. Fungsi objektif awal: $P_0 = 0$ 
 - f. Iterasi awal: $t = 1$;
3. Bangkitkan bilangan acak (μ_{ik}), dengan $i = 1, 2, \dots, n$; $k = 1, 2, \dots, n$; dan c sebagai elemen-elemen matriks partisi awal U.

Hitung jumlah setiap kolom :

$$Q_i = \sum_k^c \mu_{ik} \text{ dengan } j = 1, 2, \dots, n$$

Hitung :

$$\mu_{ik} = \frac{\mu_{ik}}{Q_i}$$

4. Hitung pusat *cluster* ke-k : V_{kj}

$$V_{kj} = \frac{\sum_i^n \mu_{ik}^w x_{ij}}{\sum_i^n \mu_{ik}^w}$$

5. Hitung fungsi objektif pada iterasi ke-t

$$P_t = \sum_{t=1}^n \sum_{k=1}^c ([\sum_{j=1}^m (X_{ij} - V_{kj})^2] (\mu_{ik})^w)$$

6. Hitung perubahan matriks partisi

$$\mu_{ik} = \frac{[\sum_{j=1}^m (X_{ij} - V_{kj})^2]^{\frac{-1}{w-1}}}{\sum_{k=1}^c \sum_{j=1}^m [(X_{ij} - V_{kj})^2]^{\frac{-1}{w-1}}}$$

7. Cek kondisi berhenti :
 - a. Jika $(|P_t - P_{t-1}| \leq \text{MaxIter})$ maka berhenti.
 - b. Jika tidak $t = t + 1$, ulangi langkah ke 4. Output yang dihasilkan dari FCM merupakan deretan pusat *cluster* dan beberapa derajat keanggotaan untuk tiap-tiap titik data.

1.3.4.2. Rapidminer

Setelah melakukan perhitungan manual peneliti akan membandingkan hasilnya dengan perhitungan menggunakan *RapidMiner*. Peneliti menggunakan *Rapidminer Studio* versi 9.10. Berikut merupakan langkah-langkahnya (Ulya & Sarah, 2020) :

1. Siapkan dataset dalam format excel
2. Buka aplikasi *Rapidminer* lalu dengan import data yang akan diolah yaitu dengan klik tombol “Add Data” yang terdapat dalam panel *repository*
3. Pilih *My Computer*
4. Cari file data dan klik *button* “Next”
5. Pilih *cells* yang akan diimportkan, lalu klik *button* “Next”
6. Pada tahap “*format your columns*” pilih tipe data integer. Kemudian, klik “Next”
7. Klik “*finish*”
8. Selanjutnya klik *button* “Design”
9. Klik data yang telah diimportkan tersebut kemudian tarik ke panel *process*
10. Selanjutnya, cari algoritma k-means pada panel *Operator*, dengan mengetikkan “*k-means*” lalu klik algoritma tersebut dan tarik ke panel *process*
11. Hubungkan data ke *clustering* dan hubungkan *clustering* ke sumbu yang ada di pojok kanan atas panel *process*
12. Klik *clustering* dan masukan jumlah cluster yang diinginkan lalu klik *run*
13. Data telah selesai dan berhasil diproses

1.3.4.3. Program Python

Perhitungan dengan program python dimaksudkan untuk memperkuat perhitungan sebelumnya yakni perhitungan manual dan menggunakan *RapidMiner*.

Dalam membuat program python, peneliti menggunakan *library Google Colab* dengan langkah-langkah sebagai berikut :

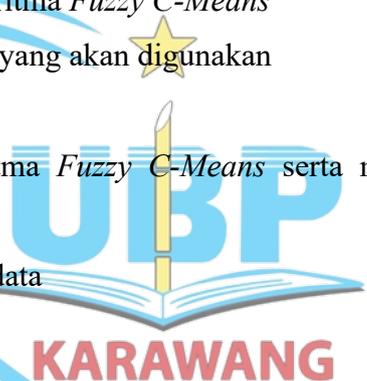
a. Algoritma *K-Means*

1. Mengimport *library* yang akan digunakan
2. Mengimport dataset
3. Menjalankan algoritma *K-Means* serta menampilkan data hasil *cluster*
4. Memvisualisasikan data

b. Algoritma *Fuzzy C-Means*

Langkah-langkah algoritma *Fuzzy C-Means* dalam Python tidak jauh berbeda dengan langkah-langkah dari algoritma *K-Means* berikut adalah langkah-langkahnya :

1. *Install Library Algoritma Fuzzy C-Means*
2. Mengimport *library* yang akan digunakan
3. Mengimport dataset
4. Menjalankan algoritma *Fuzzy C-Means* serta menampilkan data hasil *cluster*
5. Memvisualisasikan data



1.3.5. Evaluasi

Evaluasi pada penelitian ini adalah dengan menggunakan metode *Davies Bouldin Index (DBI)*. Metode ini merupakan cara untuk menyempurnakan kinerja sebuah metode atau algoritma optimasi demi mendapatkan jumlah *cluster* yang terbaik (Wen Gie & Deny Jollyta, 2020). Pada tahap inilah akan diukur algoritma manakah yang terbaik dalam mengelompokkan siswa berprestasi berdasarkan nilai.

Sum of Square Within Cluster (SSW) merupakan persamaan yang digunakan untuk mengetahui matrik kohesi dalam sebuah cluster ke-*i* yang dirumuskan sebagai berikut (Zulfa Nabila, et al., 2021) :

$$SSW_i = \frac{1}{m_i} \sum_{j=1}^{m_i} d(x_j, c_i)$$

Keterangan :

m = jumlah data dalam *cluster* ke-*i*

c = *centroid cluster* ke-*i*

d = jarak

Untuk menghitung separasi antar *cluster* dapat menggunakan *Sum of Square Between Cluster* (SSB) dengan penjabaran sebagai berikut (Zulfa Nabila, *et al.*, 2021) :

$$SSB_{ij} = d(c_i, c_j)$$

Keterangan :

d = jarak

c = centroid cluster ke- i

Setelah memperoleh nilai kohesi dan separasi langkah selanjutnya melakukan pengukuran rasio (R_{ij}) untuk mengetahui nilai perbandingan antar *cluster* ke- i dan ke- j . *Cluster* yang memiliki nilai kohesi terkecil dan separasi yang terbesar adalah *cluster* yang baik. Berikut persamaan dalam menghitung nilai rasio (Zulfa Nabila, *et al.*, 2021) :

$$DBI = \frac{1}{k} \sum_{i=1}^k \max_{i \neq j} (R_{ij})$$

k merupakan jumlah *cluster* yang digunakan. Semakin kecil nilai DBI yang diperoleh (non-negatif ≥ 0), maka semakin baik *cluster* yang diperoleh.

1.3.6. Kesimpulan

Setelah semua tahapan selesai, tahap selanjutnya ialah menarik kesimpulan dari penelitian ini dan pada tahap inilah peneliti akan menyebutkan hasil dari penelitian ini termasuk menarik kesimpulan algoritma mana yang lebih baik