

BAB III

METODOLOGI PENELITIAN

3.1. Subjek dan Objek Penelitian

Subjek pada penelitian ini adalah kombinasi *K-Means* dengan *Particle Swarmi Optimization*. Objek penelitian ini adalah data penjualan pada Supermarket Hunter's e-grocery.

3.2. Alat dan Bahan Penelitian

3.2.1 Perangkat Keras (*hardware*)

- a. Laptop HP 14s-dq261TU
- b. *Processor* Intel Core i3 2,1 GHz
- c. RAM 4 Gb
- d. *Storage* 256 SSD

3.2.2 Perangkat Lunak (*software*)

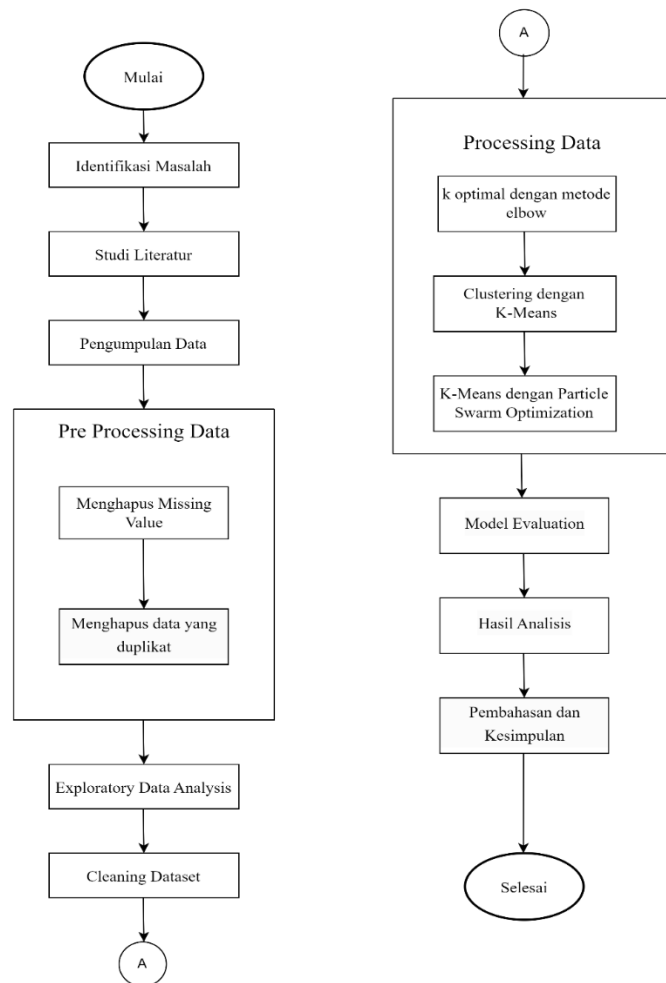
- a. Sistem operasi Windows 11
- b. Python

3.2.3 Bahan/Dataset

Dataset yang digunakan pada riset ini diperoleh dari Kaggle yang disediakan oleh Rupes Kumar "Supermarket dataset for predictive marketing 2023" yang memiliki 2019501 baris dan 12 kolom

3.3. Diagram Alur Penelitian

Diagram alir penelitian berfungsi untuk mengetahui tahapan apa saja yang akan peneliti lakukan dalam penelitian Kombinasi *K-Means* dengan *Particle Swarm Optimization* sehingga jalannya penelitian lebih terstruktur dan mudah untuk dilakukan. Gambar 3.1 menunjukkan mengenai alur penelitian sebagai berikut:



Gambar 3.1 Diagram Alur Penelitian

Berdasarkan Gambar 3.1 mengenai diagram alur penelitian, maka penjelasan untuk setiap tahapannya dapat diuraikan sebagai berikut:

3.3.1. Identifikasi Permasalahan

Tahap ini dilakukan untuk mengidentifikasi masalah dengan konteks urgensi topik penelitian. Masalah akan dilihat secara umum dari sudut pandang objek dan secara rinci sesuai dengan studi kasus subjek penelitian yang dilakukan, yaitu data penjualan di supermarket. Permasalahan yang diidentifikasi kemudian akan dirumuskan agar dapat dicari solusinya

3.3.2. Studi Literatur

Penelitian ini harus memiliki referensi sebagai dasar langkah penelitian serta referensi yang digunakan oleh penulis dari jurnal sebelumnya yang terkait dengan penelitian sebagai referensi. Adapun buku-buku terkait digunakan untuk menambah wawasan penulis dalam melakukan penelitian ini.

3.3.3. Pengumpulan Data

Dataset yang digunakan bersumber dari Kaggle yang berisi data penjualan di Supermarket terdiri 2019501 baris dan 12 kolom.

order_id	user_id	order_nur	order_dov	order_hou	days_sinc	product_i	add_to_ca	reordered	departme	departme	product_name
2425083	49125	1	2	18	17	1	0	13	pantry	baking ingredients	
2425083	49125	1	2	18	91	2	0	16	dairy eggs	soy lactosefree	
2425083	49125	1	2	18	36	3	0	16	dairy eggs	butter	
2425083	49125	1	2	18	83	4	0	4	produce	fresh vegetables	
2425083	49125	1	2	18	83	5	0	4	produce	fresh vegetables	
2425083	49125	1	2	18	91	6	0	16	dairy eggs	soy lactosefree	
2425083	49125	1	2	18	120	7	0	16	dairy eggs	yogurt	
2425083	49125	1	2	18	59	8	0	15	canned go	canned meals beans	
2425083	49125	1	2	18	35	9	0	12	meat seaf	poultry counter	
1944304	162867	1	3	17	37	1	0	1	frozen	ice cream ice	
1944304	162867	1	3	17	24	2	0	4	produce	fresh fruits	
1944304	162867	1	3	17	83	3	0	4	produce	fresh vegetables	
1944304	162867	1	3	17	84	4	0	16	dairy eggs	milk	
1944304	162867	1	3	17	91	5	0	16	dairy eggs	soy lactosefree	
1944304	162867	1	3	17	24	6	0	4	produce	fresh fruits	
1944304	162867	1	3	17	24	7	0	4	produce	fresh fruits	

Gambar 3.2 Dataset

Tabel 3.1 Deskripsi Variabel

Variabel	Deskripsi
order_id	Identifikasi unik untuk setiap pesanan yang dibuat oleh pengguna. Setiap pesanan memiliki ID yang unik.
user_id	Identifikasi unik untuk setiap pengguna yang melakukan pembelian. Setiap pengguna memiliki ID yang unik.
order_number	Nomor urut dari pesanan yang dibuat oleh pengguna.
order_dow	Hari dalam seminggu saat pesanan dilakukan.
order_hour_of_day	Jam dalam hari saat pesanan dilakukan.
days_since_prior_order	Jumlah hari sejak pesanan terakhir oleh pengguna yang sama.
product_id	Identifikasi unik untuk setiap produk yang dipesan. Setiap produk dalam katalog memiliki ID yang unik.
add_to_cart_order	Urutan dimana produk ditambahkan ke keranjang belanja dalam satu pesanan.
reordered	Penanda apakah produk tersebut telah dipesan sebelumnya oleh pengguna. Nilai 1 menunjukkan produk telah dipesan ulang, sedangkan nilai 0 menunjukkan produk tersebut baru pertama kali dipesan.
department_id	Identifikasi unik untuk setiap departemen dalam supermarket.
department	Nama dari department yang terkait.
product_name	Nama produk yang dipesan.

3.3.4. Preprocessing Data

Preprocessing data adalah proses yang dilakukan untuk mengubah data mentah menjadi format yang mudah untuk diproses atau dipahami. Proses ini memiliki peran penting dalam awal pengolahan data, karena seringkali terdapat data dalam format yang tidak berstruktur. Pada *preprocessing* data ini dilakukan penghapusan *missing value*, menghapus kolom yang tidak representatif atau tidak dipakai, dan menghapus data yang duplikat.

1. Penghapusan *Missing Values*

Proses pembersihan data melibatkan pengecekan terhadap nilai kosong dalam lembar data. Pengecekan data kosong dilakukan dengan menggunakan perintah `df.isnull().sum()`, yang menghitung jumlah data yang kosong. Setelah pengecekan, tidak ada lagi data yang kosong dalam lembar data.

```
data.isnull().sum()
order_id          0
user_id           0
order_number      0
order_dow         0
order_hour_of_day 0
days_since_prior_order 0
product_id        0
add_to_cart_order 0
reordered         0
department_id     0
department        0
product_name      0
dtype: int64
```

Gambar 3.3. Proses Pengecekan Data Kosong

2. Mengecek Data Yang Duplikat

Proses pengecekan data duplikat dilakukan dengan menggunakan perintah `df.duplicated().sum()`.

```
#mengecek data duplikat
data.duplicated().sum()
0
```

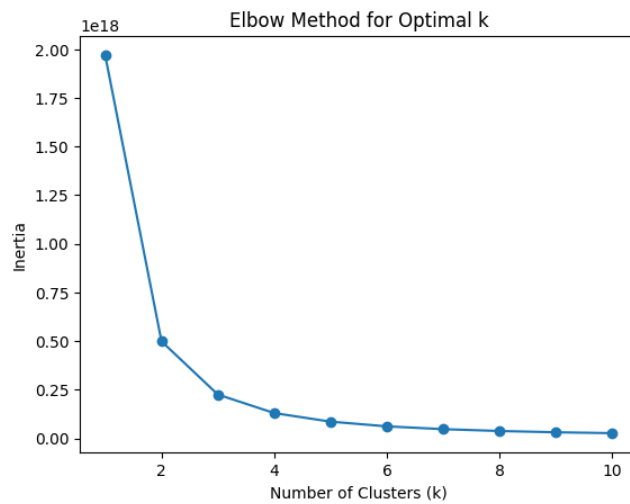
Gambar 3.4. Proses Pengecekan Data Yang Duplikat

3.3.5. *Exploratory Data Analysis (EDA)*

Exploratory Data Analysis (EDA) adalah tahap analisis data yang dilakukan sebelum menerapkan model statistik bertujuan untuk memahami data secara visual dan deskriptif. *Exploratory Data Analysis (EDA)* memiliki peran penting dalam mengidentifikasi tren, pola, dan wawasan yang dapat menjadi dasar untuk pengambilan keputusan yang tepat.

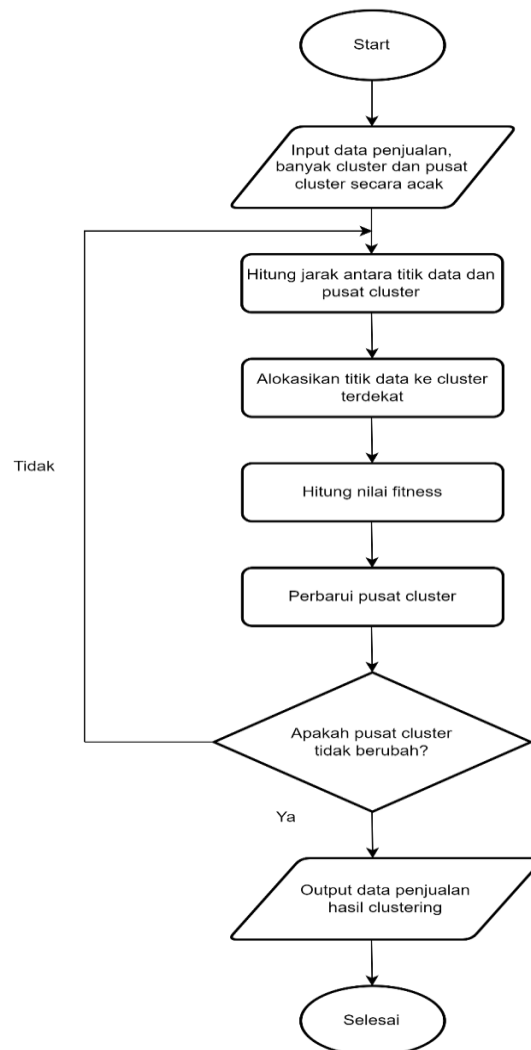
3.3.6 Processing Data

Proses ini diawali dengan mencari k optimal dengan menggunakan metode *elbow*. Metode *elbow* adalah salah satu metode yang digunakan untuk menentukan jumlah *cluster* yang optimal, yaitu dengan cara melihat presentase setiap *cluster* yang akan membentuk siku pada suatu titik tertentu.



Gambar 3.5 Metode Elbow

3.3.7 Algoritma *K-Means*



Gambar 3.6 Algoritma *K-Means*

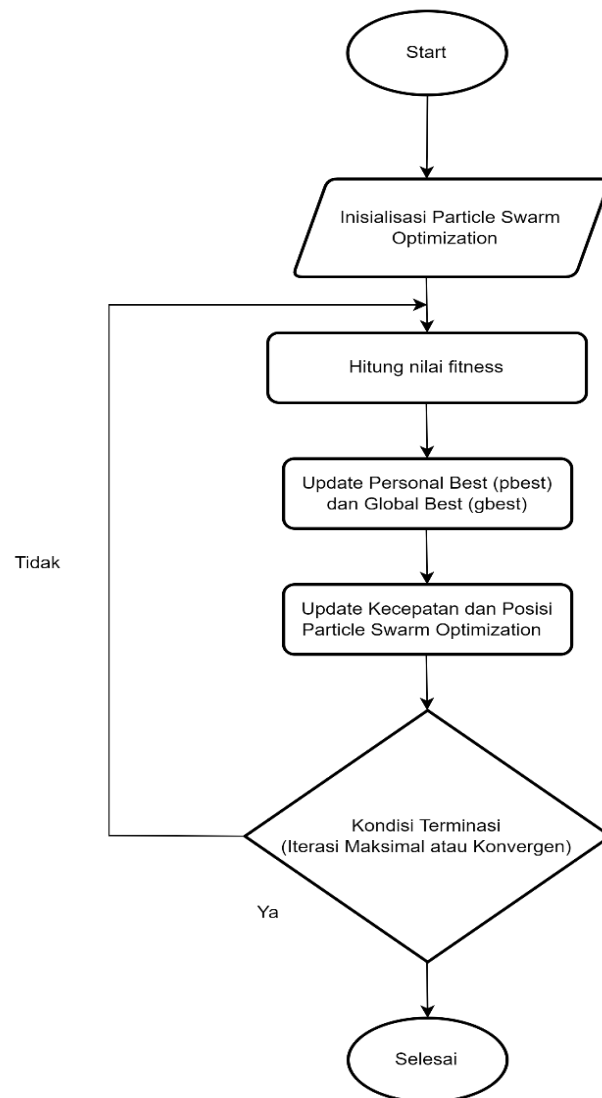
Berikut interpretasi dari diagram alir tersebut berdasarkan Gambar 3.6:

1. Input data penjualan, banyak *cluster*, dan pusat *cluster* secara acak: input data penjualan dengan memasukkan data penjualan yang akan dikelompokkan, kemudian tentukan jumlah *cluster* dan inisialisasi pusat *cluster* secara acak dalam ruang pencarian data.
2. Hitung jarak antara titik data dan pusat *cluster*.
3. Alokasikan titik data ke *cluster* terdekat: metapkan setiap titik data ke *cluster* dengan pusat *cluster* terdekat berdasarkan jarak yang telah

dihitung.

4. Hitung nilai *fitness*
5. Perbarui pusat *cluster*: menghitung ulang pusat *cluster* sebagai rata-rata dari semua titik data yang dialokasikan ke *cluster* tersebut.
6. Apakah pusat *cluster* tidak berubah: dengan memeriksa apakah pusat *cluster* yang baru sama dengan pusat *cluster* sebelumnya. Jika pusat *cluster* tidak berubah (Ya), lanjutkan ke langkah berikutnya. Jika pusat *cluster* berubah (Tidak), ulangi dari langkah hitung jarak antara titik data dan pusat *cluster*.
7. Output data penjualan hasil *clustering*: memberikan hasil akhir dari proses *clustering*.

3.3.8. Algoritma *Particle Swarm Optimization*



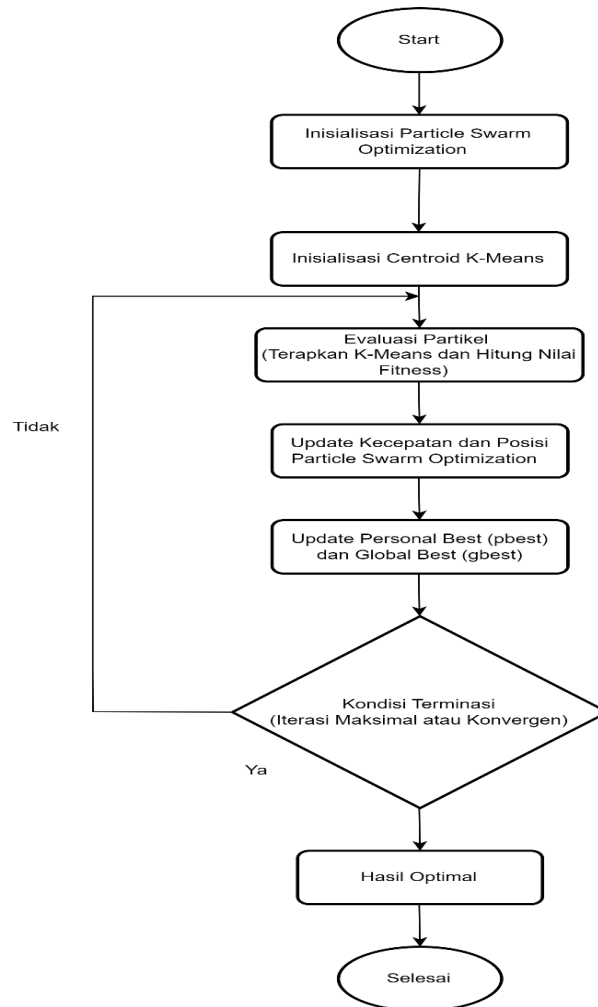
Gambar 3.7 *Particle Swarm Optimization*

Berikut interpretasi dari diagram alur tersebut berdasarkan Gambar 3.7:

1. Inisialisasi *Particle Swarm Optimization* yaitu dengan menentukan jumlah partikel, menginisialisasi posisi awal dari setiap partikel secara acak dalam ruang pencarian, dan menginisialisasi kecepatan awal dari setiap partikel secara acak.
2. Hitung nilai *fitness*
3. Update kecepatan dan posisi *Particle Swarm Optimization*.

4. Update Personal Best (*pbest*) dan Global Best (*gbest*).
5. Kondisi terminasi (iterasi maksimal atau konvergensi) yaitu jika iterasimaksimal artinya algoritma berhenti setelah mencapai jumlah iterasi yang ditentukan, sedangkan jika konvergensi artinya algoritma berhenti jika perubahan nilai *fitness* antar iterasi sangat kecil atau tidak signifikan.
6. Output nilai terbaik yaitu setelah mencapai kondisi terminasi, hasil yang baik diperoleh dari Global Best (*gbest*) yang dioptimalkan.

3.3.9. Kombinasi *K-Means* dengan *Particle Swarm Optimization*



Gambar 3.8 Kombinasi *K-Means* dengan *Particle Swarm Optimization*

Berikut interpretasi dari diagram alur tersebut berdasarkan Gambar 3.8:

1. Inisialisasi *Particle Swarm Optimization*: menginisialisasi populasi awal partikel. Setiap partikel mewakili satu nilai dalam ruang pencarian. Lokasi awal dan kecepatan setiap partikel ditetapkan secara acak.
2. Inisialisasi *centroid K-Means*: menentukan jumlah *cluster* agar dapat mengetahui posisi *centroid* yang diinisialisasi secara acak.
3. Evaluasi partikel (terapkan *K-Means* dan hitung nilai *fitness*): menerapkan *K-Means* pada setiap data ke *centroid* terdekat, kemudian perbarui posisi *centroid* berdasarkan rata-rata data yang ditetapkan ke setiap *centroid* dan kemudian hitung nilai *fitness* menggunakan *Sum of Squared Errors* (SSE).
4. Update kecepatan dan posisi *Particle Swarm Optimization*.
5. Update Personal Best (*pbest*) dan Global Best (*gbest*).
6. Kondisi terminasi (iterasi maksimal atau konvergensi): jika iterasi maksimal artinya algoritma berhenti setelah mencapai jumlah iterasi yang ditentukan, sedangkan jika konvergensi artinya algoritma berhenti jika perubahan nilai *fitness* antar iterasi sangat kecil atau tidak signifikan.
7. Hasil optimal: setelah mencapai kondisi terminasi, hasil *clustering* yang baik diperoleh dari Global Best (*gbest*) yang dioptimalkan.

3.3.10. Model Evaluation

Pada tahap ini performa model dievaluasi menggunakan *Davies Bouildin Index* (DBI). Pada tahap ini model dievaluasi menggunakan *Davies Bouildin Index* (DBI). *Davies Bouildin Index* (DBI) digunakan untuk melakukan proses evaluasi dari hasil model *clustering*. *Davies Bouildin Index* (DBI) melakukan proses pengukuran suatu hasil pengelompokan berdasarkan pada nilai kohesi dan separasi. Hasil pengelompokan kohesi merupakan jumlah dari kedekatan data terhadap suatu titik pusat (*centroid*) dari suatu

hasil pengelompokan dan hasil pengelompokan separasi berdasarkan pada jarak antar *centroid* dari *cluster* nya.