

BAB II

LANDASAN TEORI

Bab II dijelaskan mengenai tinjauan pustaka serta teori-teori untuk penunjang dalam pembuatan laporan. Teori sebagai penunjang tersebut ialah seperti Angkutan Umum, Pengelompokan Data, Algoritma *K-Means*, Algoritma Genetika, *Elbow*, *Silhouette Coefficient* dan *Davies-Bouldin Index*.

2.1. Tinjauan Pustaka

Tinjauan pustaka ini berisi tentang berbagai penelitian yang berkaitan dengan penelitian yang akan dilakukan. Berbagai penelitian yang menjadi dasar diantaranya:

(Zheng, Bai, & Chan, 2019), Penelitian dengan judul *Optimization Of Savonius Turbine Clusters Using An Evolutionary Based Genetic Algorithm* Penelitian ini bertujuan untuk mengoptimasi tata letak turbin angina dengan algoritma genetika.. Pada penelitian ini memiliki masalah yaitu kekhawatiran terhadap emisi gas rumah kaca yang meningkat. Oleh sebab itu penelitian ini memanfaatkan energi angin dari turbin. Metode yang digunakan dalam penelitian ini adalah algoritma genetika untuk mengoptimasi tata letak turbin yang dapat dipindahkan. Hasil dari penelitian ini adalah dengan meningkatkan optimasi dalam tata letak turbin angin dapat meningkatkan sebesar 30% angin dibandingkan dengan turbin yang terisolasi atau turbin yang tidak dapat dipindahkan. Dengan potensi besar hasil optimalisasi menggunakan algoritma genetika dapat membangun peternakan angin yang efisien (Zheng, Bai, & Chan, 2019).

(Cherif, 2018), Penelitian dengan judul *Optimization Of K-NN Algorithm By Clustering And Reliability Coefficients: Application To Breast-Cancer Diagnosis*. Penelitian ini bertujuan untuk mengoptimasi diagnosis penyakit kanker payudara dengan pengelompokan menggunakan algoritma K-NN. Pada penelitian ini memiliki masalah dalam mendiagnosis penyakit payudara dengan metode K-NN karena pada metode K-NN memiliki kelemahan seperti lambat karena meninjau semua *instance*, rentan terhadap dimensi, *sensitive* terhadap atribut yang tidak relevan dan berkorelasi, pilihan jarak yang salah atau nilai K menurunkan kinerja. Untuk mengoptimalkan

dalam mendiagnosis penyakit kanker payudara dapat memadukan metode antara algoritma K-NN dan algoritma CNN karena algoritma CNN dapat menghilangkan data duplikat, menghapus instance yang tidak relevan karena tidak memberikan informasi tambahan dan menunjukkan kesamaan dengan dataset pelatihan lainnya. Hasil dari memadukan algoritma K-NN dan algoritma CNN dihasilkan bahwa algoritma yang diusulkan mengungguli KNN, naïve bayes, SVM dengan ukuran f sedikit melibih 94% pada dataset yang dipertimbangkan (Cherif, 2018).

(Ghezelbash, Maghsoud, & Carranza, 2019), Penelitian dengan judul *Optimization Of Geochemical Anomaly Detection Using A Novel Genetic K-Means Clustering (GKMC) Algorithm*. Penelitian ini bertujuan untuk mengoptimalkan dalam mendeteksi *anomaly* dengan menggunakan *Novel Genetic K-Means Clustering (GKMC) Algorithm*. Masalah pada penelitian ini adalah bagaimana mengoptimalkan pendektsian *anomaly* dalam mencari geokimia seperti deposit mineral. Pada penelitian ini menggunakan metode algoritma GKMC dan TKMC untuk banding. Hasil dari penelitian ini dalam mencari penyimpangan atau anomaly menggunakan metode GKMC dan TKMC terdapat perbedaan yang signifikan. Jika melakukan pencarian anomali menggunakan GKMC hasil yang diberikan sebesar 83% sedangkan menggunakan metode TKMC hasil yang diberikan sebesar 66%. Dari hasil tersebut dibuktikan mengoptimalkan dengan metode GKMC dapat diandalkan karena efisien dan kuat untuk mengenali anomaly (Ghezelbash, Maghsoudi, & Carranza, 2020).

(Praja, Kusuma, Setianingsih, 2019), Penelitian dengan judul *Penerapan Metode K-Means Clustering Dalam Pengelompokan Data Penumpang Dan Kapal Angkutan Laut Di Indonesia*. Penelitian ini bertujuan untuk meninjau laju pertumbuhan jumlah angkutan laut di setiap provinsi di Indonesia. Pada penelitian ini menggunakan metode pengelompokan *K-Means* karena tidak terpengaruh terhadap urutan objek yang digunakan dan juga pusat cluster ditentukan secara acak dari salah satu objek pada permulaan perhitungan. Hasil dari pengelompokan data menggunakan K-Means menghasilkan pengelompokan rendah, sedang, tinggi dan sangat tinggi. Perubahan

hasil dari pengelompokan bisa disebabkan beberapa faktor seperti kondisi cuaca yang buruk dalam melakukan pelayaran (Praja, Kusuma, & Setianingsih, 2019).

(Fajrianti, Bustan, & Tiro, 2019), Penelitian dengan judul *Penggunaan Analisis Cluster K-Means Dan Analisis Diskriminan Dalam Pengelompokan Desa Miskin Di Kabupaten Pangkep*. Penelitian ini bertujuan untuk melakukan pembuktian terhadap tingkat pendidikan yang rendah. Pada penelitian ini menggunakan metode pengelompokan algoritma K-Means dan analisis diskriminan dengan parameter tingkat Pendidikan, angka kelahiran kasar, angka kematian dan rata-rata banyaknya anggota keluarga. Dari hasil penelitian diperoleh bahwa dengan membentuk pengelompokan dengan 3 *cluster* yang terbentuk diperoleh sebesar 98,06% maka hasil akurasi lebih baik dari *cluster* lain yang dibentuk (Dalam, Desa, & Pangkep, n.d.).

(Akbar, Marisa, & Wijaya, 2019), Penelitian berjudul *Sistem Informasi Pemberian Bonus Upah Dan Penjadwalan Karyawan Menggunakan Metode Algoritma Genetika*. Penelitian ini bertujuan untuk melakukan pembuatan jadwal karyawan dan pemberian upah bonus karyawan. Serta memberikan bukti gaji karyawan secara benar dan akurat. Pada penelitian ini menggunakan metode algoritma genetika dengan parameter kendaraan, kuli dan waktu dalam proses penjadwalan. Hasil dari penelitian tersebut sebesar 87,37% membuktikan dengan mengoptimalkan penggunaan algoritma genetika tergantung pada nilai parameter pembangkitan kromosom yang dimasukan, semakin besar jumlah kromosom yang dimasukan maka semakin baik hasil dari penjadwalan karyawan dan pemberian upah bonus secara benar dan akurat (Akbar et al., 2019).

(Suryani, & Priyanti, 2019), Penelitian berjudul *Optimasi Naïve Bayes Dan Algoritma Genetika Untuk Prediksi Penerimaan Beasiswa Pendidikan Pada SMP Utama*. Penelitian ini bertujuan untuk menentukan tingkat akurasi yang tinggi untuk mengklasifikasi penerima beasiswa berdasarkan kriteria sekolah agar mendapatkan penerima beasiswa yang benar-benar membutuhkan. Penelitian ini menggunakan metode algoritma naïve bayes dengan parameter status orangtua, pekerjaan orangtua,

status orangtua, pekerjaan orangtua, rumah sewaan, peralatan rumah tangga, kendaraan, tabungan orangtua, perhiasan orangtua, ponsel dan uang saku sebesar 77,50%. Untuk mendapatkan tingkat akurasi yang lebih optimal digunakanlah algoritma genetika menghasilkan akurasi sebesar 83,33% (Wahyudi, Bahri, & Handayani, 2019).

(Hadi, Putra, & Kumara, 2016), Penelitian berjudul *Penentuan Kompetensi Mahasiswa Dengan Algoritma Genetik Dan Metode Fuzzy Cmeans*. Penelitian ini bertujuan untuk menentukan kompetensi untuk mahasiswa. Pada penelitian ini menggunakan metode pengelompokan Algoritma Genetika dan Algoritma *Fuzzy Cmeans*. Pada Algoritma Genetika digunakan dalam menentukan fitur yang relevan dan Algoritma *Fuzzy Cmeans* digunakan dalam melakukan klasterisasi data secara optimal. Dari kedua algoritma menghasilkan 77 fitur dan hanya 61 fitur yang relevan dan valid memberikan hasil persentasae rata-rata kesesuaian sebesar 88,89% (Hadi, Gede Darma Putra, & Satya Kumara, 2016).

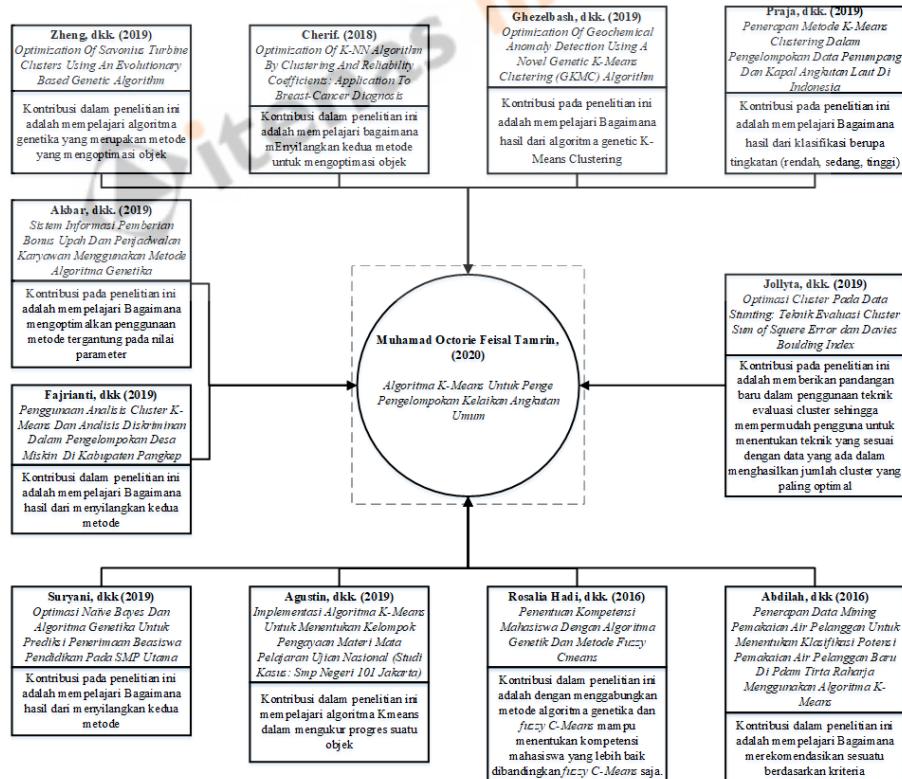
(Agustin, Fitria, Hanifah, 2015), Penelitian berjudul *Implementasi Algoritma K-Means Untuk Menentukan Kelompok Pengayaan Materi Mata Pelajaran Ujian Nasional (Studi Kasus: Smp Negeri 101 Jakarta)*. Penelitian bertujuan untuk menentukan kelompok pengayaan materi, dapat juga digunakan untuk mengukur progres siswa terhadap materi yang sudah dikuasai. Penelitian ini menggunakan metode pengelompokan Algoritma K-Means dengan parameter sesuai dengan jumlah mata pelajaran UN. Dari penelitian ini menghasilkan kemampuan siswa secara keseluruhan dengan mata pelajaran yang harus diperlukan (Agustin, 2015).

(Abdillah, Putra, & Renaldi, 2016), Penelitian berjudul *Penerapan Data Mining Pemakaian Air Pelanggan Untuk Menentukan Klasifikasi Potensi Pemakaian Air Pelanggan Baru Di Pdam Tirta Raharja Menggunakan Algoritma K-Means*. Penelitian ini bertujuan untuk memberikan rekomendasi pengevaluasian potensi pendapatan dan pemakaian air calon pelanggan baru PDAM disesuaikan dengan kriteria pemakaian air pelanggan baru. Pada penelitian ini menggunakan metode

pengelompokan K-Means. Hasil dari penelitian tersebut sebesar 95,80% (Abdillah et al., 2016).

(Jollyta, Efendiz, Zarlis & Mawengkang, 2019), Penelitian berjudul *Optimasi Cluster Pada Data Stunting: Teknik Evaluasi Cluster Sum of Squere Error dan Davies Boulding Index*. Penelitian ini bertujuan untuk mengelompokan data stunting. Pada penelitian ini menggunakan Algoritma K-Means, metode Sum of Squere Error (SSE) dan Davies Boulding Index (DBI). Hasil penelitian ini diharapkan dapat memberikan pandangan baru dalam penggunaan teknik evaluasi cluster sehingga mempermudah pengguna untuk menentukan teknik yang sesuai dengan data yang ada dalam menghasilkan jumlah cluster yang paling optimal dan informasi yang benar (Jollyta, Efendi, Zarlis, & Mawengkang, 2019).

Berdasarkan tinjauan pustaka dilakukan pemetaan tinjauan pustaka yang menunjang penelitian ini seperti pada Gambar 2.1 berikut.



Gambar 2. 1 Peta Tinjauan Pustaka

2.2. Angkutan Umum

Angkutan adalah suatu sarana yang disediakan untuk mengangkut barang atau penumpang dari suatu tempat ke tempat yang dituju. Proses dalam angkutan ini dapat berupa kendaraan seperti bus, mobil dan motor. Selain kendaraan dapat juga tanpa kendaraan seperti becak, sepedah dan kereta kuda.



Gambar 2. 2 Angkutan Umum
(Sumber : Liputan6.com)

Angkutan umum adalah suatu kendaraan seperti bus, minibus, kereta api dan kapal termasuk angkutan air dan udara yang dapat mengangkut penumpang serta barang dari suatu tempat ke tempat yang lain dengan sistem sewa atau bayar (Warpani, 2002). Angkutan umum disediakan oleh pemerintah dan suatu perusahaan sebagai salah satu pelayanan transportasi publik dengan tarif yang telah ditentukan.

Menurut undang-undang nomor 22 tahun 2009 tentang angkutan orang dengan kendaraan bermotor umum dalam trayek terdapat beberapa jenis pelayanan angkutan dengan kendaraan bermotor terdiri dari:

1. Angkutan lintas batas negara, merupakan pemindahan orang dari suatu negara ke negara lain yang berbatasan dengan asalnya.
2. Angkutan antar kota antar provinsi, merupakan pemindahan orang dari suatu provinsi ke provinsi lain.
3. Angkutan antar kota dalam provinsi, merupakan pemindahan orang dari suatu kota ke kota lain yang masih satu provinsi.

4. Angkutan perkotaan, merupakan pemindahan orang dari suatu tempat ke tempat lain yang masih dalam satu kota.
5. Angkutan pedesaan, merupakan pemindahaan orang dari suatu tempat ke tempat lain yang masih dalam satu kabupaten.

Menurut Warpani (Warpani, 2002) kinerja angkutan umum adalah hasil kerja angkutan umum dalam beroperasi selama ini untuk melayani kegiatan masyarakat. Angkutan umum dikatakan memiliki kinerja yang baik jika angkutan tersebut menghasilkan pelayanan yang aman, cepat,murah, dan nyaman bagi penumpang. Parameter kinerja angkutan umum dalam penelitian ini diantaranya adalah :

1. Kecepatan perjalanan
2. Faktor muat (*load factor*)
3. Waktu antara (*time headway*)

2.3. Pengelompokan Data

Klasterisasi atau pengelompokan data merupakan sebuah pendekatan untuk mencari kesamaan atau keselarasan dalam data dan menempatkan data yang sama ke dalam suatu kelompok. Tujuan dari pengelompokan data adalah untuk menempatkan objek yang mirip dalam satu *cluster* dan membuat jarak *cluster* sejauh mungkin agar terlihat obyek pada satu *cluster* berbeda dengan *cluster* lain (Susanto & Suryani, 2010).

Pengelompokan data terbagi menjadi dua jenis yaitu *hierarchical clustering* (hierarki) dan *non-hierarchical clustering* (partisi). Dalam proses pengelompokan data *hierarchical clustering* satu data tunggal dapat dianggap sebuah *cluster*, dua atau lebih *cluster* kecil dapat bergabung menjadi sebuah cluster besar dan begitu seterusnya hingga semua data dapat bergabung menjadi sebuah *cluster*. Sedangkan pada pengelompokan data jenis *non-hierarchical clustering* membagi set data ke dalam sejumlah *cluster* sehingga tidak bertumpang-tindih antara satu *cluster* dengan *cluster* lain dimana setiap *cluster* memiliki pusat *cluster* masing-masing (Prasetyo, 2014).

2.4. Normalisasi Data

Terdapat beberapa cara untuk menghasilkan hasil yang efektif salah satunya adalah melakukan normalisasi data. Normalisasi data merupakan suatu cara untuk memberikan bobot yang sama pada semua parameter atau variabel dimana bobot yang berlebihan pada suatu variabel akan dihilangkan sehingga dapat meningkatkan akurasi pada proses yang dilakukan (Virmani, Taneja, & Malhotra, 2015). Pada penelitian ini normalisasi yang digunakan adalah *Min-Max* normalisasi dimana normalisasi tersebut merubah data menjadi data yang berjarak antara 0,0 sampai 1,0.

Rumus *Min-Max* normalisasi yang digunakan seperti pada Persamaan 2.1 sebagai berikut (Virmani et al., 2015).

$$v' = \frac{v - \min_A}{\max_A - \min_A} \quad (2.1)$$

Dimana:

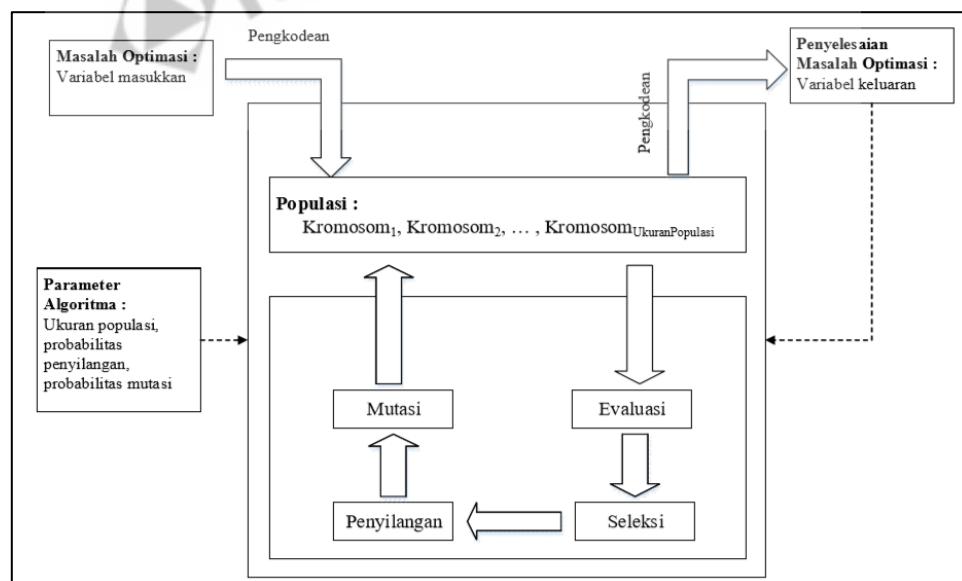
- | | | |
|----------|---|----------------------------------|
| v' | = | Nilai yang telah dinormalisasi |
| v | = | Nilai yang belum dinormalisasi |
| \max_A | = | Nilai maksimum dari atribut ke A |
| \min_A | = | Nilai minimum dari atribut ke A |

2.5. Algoritma Genetika

Algoritma genetika adalah salah satu metode yang meniru konsep pada teori Darwin yaitu proses seleksi alam dimana individu-individu saling bersaing untuk mempertahankan hidup (Arkeman, Seminar, & Gundawan, 2012). Individu-individu tersebut bersaing dimana yang kurang fit akan punah dan yang fit akan terus hidup. Algoritma ini dapat melakukan pencarian dan optimasi dengan baik. Algoritma ini bekerja tidak secara langsung kepada parameter-parameter permasalahan yang dipilih tetapi dimulai dengan bekerja pada calon-calon solusi yang dikodekan kedalam kromosom yang isinya terdiri dari unit-unit kecil atau gen.

Menurut Haupt , Algoritma Genetika merupakan suatu metode heuristik yang dikembangkan berdasarkan prinsip genetika dan proses seleksi alamiah Teori Evolusi Darwin. Metode optimasi dikembangkan oleh John Holland sekitar tahun 1960-an dan dipopulerkan oleh seorang mahasiswanya, David Goldberg, pada tahun 1980-an (Zukhri, 2014). Demikian juga dalam proses pencarian yang berlangsung dalam Algoritma Genetika. Pencarian dimulai dengan pembangkitan sejumlah individu secara acak yang disebut dengan kromosom. Kromosom-kromosom ini merupakan representasi calon penyelesaian yang akan diperiksa nilai yang sebenarnya. Seperti halnya proses evolusi alamiah, kromosom-kromosom akan dinilai tingkat kebugarannya. Hanya kromosom dengan tingkat kebugaran yang tinggi saja yang terpilih untuk bertahan dalam populasi.

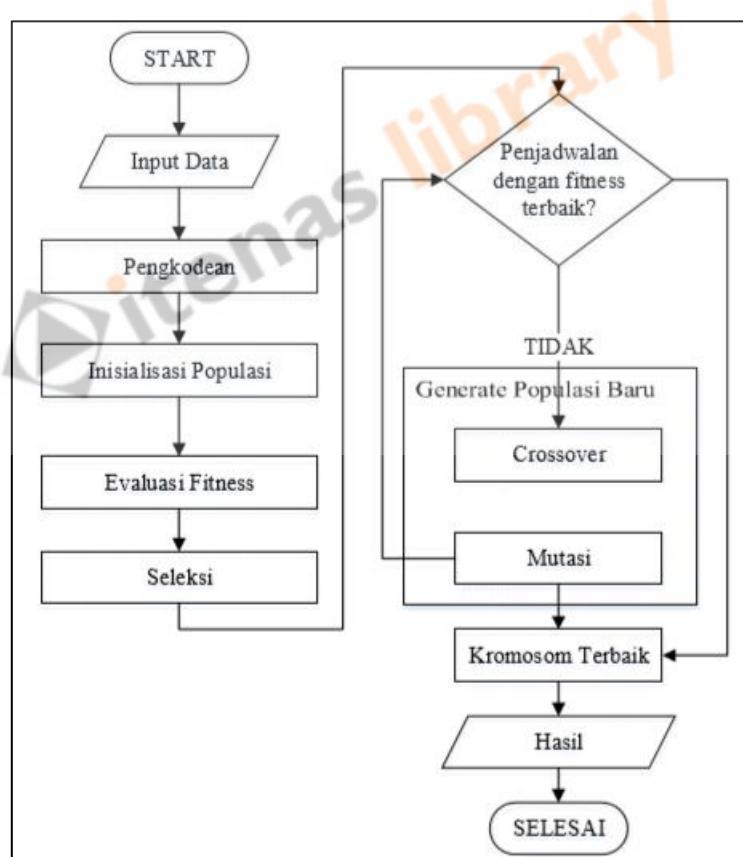
Kerangka kerja yang biasa digunakan dalam penerapan Algoritma Genetika untuk menyelesaikan suatu masalah optimasi ditunjukkan pada Gambar 2.2. Keberhasilan penggunaan Algoritma Genetika sangat ditentukan oleh penentuan pernyataan masalah ke dalam bentuk titik-titik pencarian yang disebut dengan kromosom, serta pemilihan operator-operator yang digunakan (Zukhri, 2014).



Gambar 2. 3 Kerangka Kerja Algoritma Genetika
Sumber : (Zukhri, 2014)

Dalam Algoritma Genetika, pemrosesan kromosom-kromosom sebagai sebuah populasi oleh operator genetika terjadi secara berulang. Pada mulanya, populasi awal dibangkitkan secara acak sesuai dengan representasi masalah yang akan dikembangkan. Selanjutnya, operatator-genetika akan menggabungkan informasi genetis dari unsur-unsur populasi untuk membentuk populasi generasi berikutnya. Setiap kromosom mempunyai nilai *fitness* yang akan setara dengan nilai penyelesaian-penyelesaian masalah, diharapkan bertambah semakin bagus (Zukhri, 2014).

Struktur umum Algoritma Genetika dapat dijelaskan pada diagram alir (flowchart) Gambar 2.3 berikut.



Gambar 2. 4 Flowchart Algoritma Genetika
Sumber : (Suhartono, 2015)

Pada proses pengkodean algoritma genetika kode biner dapat digunakan untuk merepresentasikan bilangan bulat dengan mudah jika banyaknya kemungkinan bilangan bulat yang akan direpresentasikan (p) sebesar pangkat dua bilangan tertentu, misalnya p adalah 2, 4, 8, 16, dan seterusnya. Hal ini mudah dimengerti karena setiap kode biner pasti dapat dipetakan satu-satu dengan semua kemungkinan bilangan bulat yang akan direpresentasikan (Suhartono, 2015).

Pada fungsi evaluasi ini yaitu nilai *fitness* yang menyatakan nilai dari fungsi tujuan. Tujuan dari algoritma Genetika adalah memaksimalkan nilai *fitness*. Jika yang dicari nilai maksimal, maka nilai *fitness* adalah nilai fungsi itu sendiri. Tetapi jika yang dibutuhkan adalah nilai minimal, maka nilai *fitness* merupakan *versi* dari nilai fungsi itu sendiri (T. Sutojo, Edy Mulyanto, 2011).

Rumus *fitness* yang digunakan seperti pada Persamaan 2.2 sebagai berikut (Suhartono, 2015).

$$Fitness = \frac{1}{1 + (F_1 B_1 + F_2 B_2 + \dots)} \quad (2.2)$$

Dimana:

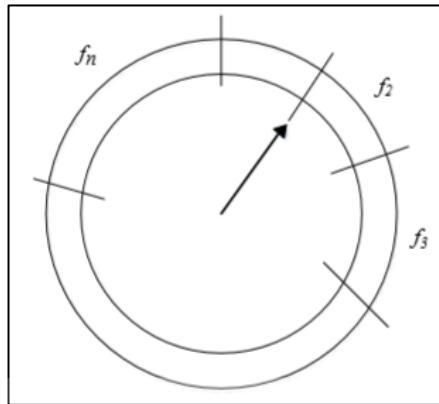
B_n = Bobot Pelanggaran

F_n = Banyaknya Pelanggaran

n = 1 ... n

Fungsi seleksi sebanding nilai *fitness* biasanya diimplementasikan dengan model roda rolet (*roulette wheel*). Dalam model ini, keliling lingkaran roda rolet dibentuk dari busur-busur sebanyak N. Perbandingan besar busur sana dengan perbandingan nilai *fitness* setiap kromosom jika perbandingan nilai *fitness* setiap kromosom adalah $f_1:f_2:f_3:\dots:f_n$, maka roda rolet untuk proses seleksi ini seperti contoh pada Gambar 2.4 proses seleksi disarankan pada posisi jarum roda rolet berhenti. Jika jarum diputar secara acak maka dapat dikatakan bahwa busur yang lebih besar

mempunyai kemungkinan yang lebih besar pula sebagai tempat jarum berhenti (Suhartono, 2015).



Gambar 2. 5 Ilustrasi Seleksi Roda Rolet
Sumber : (Zukhri, 2014)

Pada hardianti dan purwanto juga dijelaskan cara kerja seleksi dengan roda rolet (*roulette wheel*) sebagai berikut (Sucitas & Nft, n.d.):

1. Menghitung *fitness* relative $p[k]$ masing – masing kromosom. Rumus untuk menentukan *fitness* relative kromosom ke-k adalah *fitness* kromosom ke-k dibagi dengan total *fitness* seperti pada Persamaan 2.3 berikut.

$$p[k] = \frac{\text{fitness}[k]}{\text{total fitness}} \quad (2.3)$$

2. Menghitung *fitness* komulatif $q[k]$. dengan ketentuan sebagai berikut :

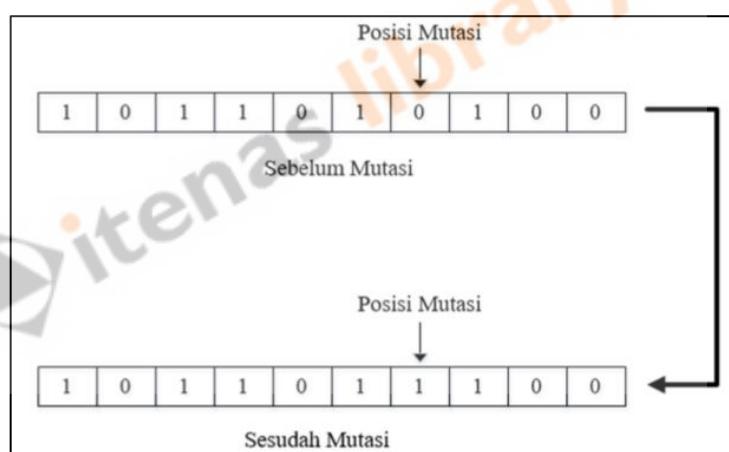
Untuk $k = 1$ maka $q[1] = p[1]$

Untuk $k \leq 1$ maka $q[k] = p[k-1] + p[k]$

3. Memilih kromosom yang akan dijadikan rekomendasi untuk ikut serta dalam tahap selanjutnya dengan cara membangkitkan bilangan r sebanyak *pop size*. Menentukan populasi baru yang terbentuk, dengan ketentuan jika bilangan acak $r[k]$ kurang dari komulatif $q[k]$, maka kromosom ke-k diganti dengan kromosom yang mempunyai nilai *fitness* kumulatif $q[k]$ tersebut

Penyilangan merupakan operator dalam Algoritma Genetika yang bertujuan untuk melahirkan kromosom baru yang mewarisi sifat-sifat induknya sebagaimana proses reproduksi yang terjadi dalam kehidupan alam. Kromosom baru hasil penyilangan disebut sebagai kromosom keturunan. Oleh karena populasi dalam Algoritma Genetika dimodelkan agar berukuran tetap, maka kromosom keturunan harus memungkinkan untuk dimasukkan dalam populasi baru (Suhartono, 2015).

Dalam representasi kromosom dengan kode biner, operasi mutasi dilakukan secara mudah, yaitu dengan mengubah nilai gen pada operasi tertentu. Hal ini berarti, jika sebuah gen terpilih secara acak untuk dikenakan operasi mutasi, maka nilai gen tersebut akan berubah dari nol menjadi satu atau dari satu menjadi nol (Suhartono, 2015) seperti pada Gambar 2.5 berikut.



Gambar 2. 6 Ilustrasi Mutasi Kode Biner
Sumber : (Zukhri, 2014)

2.6. Euclidean Distance

Euclidian Distance diperkenalkan oleh Euclid yang merupakan seorang matematikawan Yunani sekitar tahun 300 B.C.E yang mempelajari hubungan antara sudut dan jarak. Metode ini berkaitan dengan teorema *Pythagoras* dan biasa diterapkan pada 1, 2 dan 3 dimensi. *Euclidian Distance* dapat mengukur jarak suatu data dengan data lain atau 2 titik yang berbeda. Berikut Persamaan 2.4 pada metode *Euclidian Distance*.

$$d(x, y) = |x - y| = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.4)$$

Dimana:

- $d(x, y)$ = Jarak antara x dan y
- x = Data pusat klaster
- y = Data pada atribut
- x_i = Nilai pusat klaster ke i
- y_i = Nilai setiap data ke i

2.7. Algoritma *K-Means*

Algoritma *K-Means* adalah suatu metode untuk melakukan analisis klaster *unsupervised* atau *non hirarki* yang memisahkan data ke dalam kelompok berdasarkan karakteristik sehingga jika terdapat data yang mempunyai karakteristik yang sama maka akan dikelompokan ke dalam satu klaster (Maguitman, 2010). Pada dasarnya penggunaan algoritma *K-Means* dalam melakukan proses klasterisasi tergantung dari data yang ada dan konklusi yang ingin dicapai. Parameter yang harus dimasukan ketika menggunakan algoritma ini adalah nilai K atau jumlah klaster yang biasanya ditentukan berdasarkan keinginan.

Algoritma *K-Means* pada awalnya mengambil sebagian dari banyaknya komponen dari populasi untuk dijadikan pusat kluster awal. Pada step ini pusat klaster dipilih secara acak dari sekumpulan populasi data. Berikutnya *K-Means* menguji masing-masing komponen didalam populasi data dan menandai komponen tersebut ke salah satu pusat klaster yang telah di definisikan tergantung dari jarak minimum antar komponen dengan tiap-tiap pusat cluster. Posisi pusat klaster akan dihitung kembali sampai semua komponen data digolongkan kedalam tiap-tiap klaster dan terakhir akan terbentuk posisi kluster baru.

Untuk itu digunakan algoritma *K-Means* yang didalamnya memuat proses dasar sebagai berikut :

1. Tentukan k sebagai jumlah klaster yang ingin dibentuk. Tetapkan pusat klaster.
2. Hitung jarak setiap data ke pusat klaster menggunakan metode *Euclidian Distance* seperti pada Persamaan 2.4 pada metode *Euclidian Distance*.
3. Kelompokkan data ke dalam cluster yang dengan jarak yang paling pendek menggunakan Persamaan 2.5 berikut.

$$\text{Min } \sum_k^k = 1d_{ik} = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.5)$$

4. Hitung pusat cluster yang baru menggunakan Persamaan 2.6 berikut.

$$C_{kj} = \frac{\sum_i^p = 1 X_{ij}}{P} \quad (2.6)$$

Dimana :

$$X_{ij} \in \text{kluster ke } k$$

P = Banyaknya anggota cluster ke k

5. Ulangi langkah 2 sampai dengan 4 hingga sudah tidak ada lagi data yang berpindah ke kluster yang lain.

2.8. Metode *Elbow*

Metode *Elbow* merupakan suatu metode untuk menghasilkan informasi dalam menentukan jumlah klaster terbaik dari hasil presentase antara jumlah klaster yang membentuk siku pada suatu titik. Pada metode *elbow* memberikan gagasan dengan cara memilih nilai klaster lalu menambah nilai klaster tersebut untuk dijadikan model data dalam penentuan klaster terbaik dan presentase perhitungan yang dihasilkan menjadi pembanding dengan jumlah klaster yang ditambah.

Hasil presentase yang berbeda dari setiap nilai klaster dapat ditunjukan dengan menggunakan grafik, jika nilai klaster pertama dengan nilai klaster kedua memberikan sudut dalam grafik atau mengalami penurunan nilai paling besar maka nilai klaster tersebut yang terbaik (Omran, Engelbrecht, & Salman, 2007). Dalam proses perbandingannya dapat dihitung menggunakan *SSE* (*Sum of Square Error*) dari setiap klaster sehingga semakin nilai K atau jumlah klaster maka nilai *SSE* semakin kecil. Terdapat rumus *SSE* pada *K-Means* seperti Persamaan 2.7 sebagai berikut (Publication, 2016).

$$SSE = \sum_{i=1}^k \sum_{x_j \in S_i} \|X_j - \mu_i\|^2 \quad (2.7)$$

Dimana:

X_j = Data pada klaster Si

μ_i = Centroid dari klaster atau nilai mean dari total dataset

Setelah dilakukan terdapat nilai atau jumlah k yang mengalami penurunan paling besar dan nilai k yang mengalami penurunan paling besar tersebut akan perlahan mulai stabil. Terdapat langkah-langkah menentukan jumlah klaster dengan metode *Elbow* sebagai berikut (Purnima & Arvind, 2014):

1. Inisialisasi nilai awal k .
2. Naikan nilai k .
3. Hitung hasil menggunakan *SSE* (*Sum of Square Error*) dari tiap nilai k .
4. Lihat hasil *SSE* (*Sum of Square Error*) dari nilai k yang turun secara signifikan.
5. Tetapkan nilai k yang membentuk siku.

2.9. Metode *Silhouette Coefficient*

Metode *Silhouette Coefficient* merupakan suatu metode untuk melakukan evaluasi atau menguji kualitas dari klaster yang dihasilkan (Azuri, Zulhanif, & Pontoh, 2016). Metode *Silhouette Coefficient* menggabungkan antara metode *cohesion* dan

separation. Hasil dari perhitungan *Silhouette Coefficient* terdapat pada angka antara -1 sampai 1, jika nilai *Silhouette Coefficient* mendekati angka 1 maka pengelompokan pada suatu klaster semakin membaik. Sebaliknya jika mendekati angka -1 maka pengelompokan pada suatu klaster semakin memburuk.

Terdapat beberapa tahapan dalam perhitungan *Silhouette Coefficient* diantaranya sebagai berikut:

1. Menghitung rata-rata jarak antara suatu data dengan data lain pada suatu klaster seperti pada Persamaan 2.8 dengan cara memisahkan i dengan semua data lain yang berada pada satu klaster.

$$a(i) = \frac{1}{|A| - 1} \sum_{j \in A, j \neq i} d(i, j) \quad (2.8)$$

Dimana:

- $a(i)$ = Perbedaan rata-rata objek (i) ke semua objek lain pada klaster A
- $d(i, j)$ = Jarak antara data i dengan data j
- A = Klaster

2. Menghitung rata-rata jarak dari data tersebut dengan semua data di klaster lain seperti pada Persamaan 2.9 berikut.

$$d(i, C) = \frac{1}{|C|} \sum_{j \in C} d(i, j) \quad (2.9)$$

Dimana:

- $d(i, C)$ = Perbedaan rata-rata objek i ke semua objek lain pada C
 - C = Klaster lain selain klaster A atau klaster C tidak sama dengan klaster A
3. Menghitung $d(i, C)$ untuk semua C, maka diambil nilai terkecil menggunakan Persamaan 2.10 berikut.

$$b(i) = \min_{c \neq A} d(i, C) \quad (2.10)$$

Dimana:

$b(i) =$ Klaster tetangga dari objek (i) yang mencapai nilai minimum.

Klaster B yang mencapai minimum yaitu ($d(i, B) = b(i)$) disebut tetangga dari objek (i) dan ini adalah klaster terbaik kedua untuk objek (i).

4. Menghitung nilai *Silhouette Coefficient* untuk setiap i seperti pada Persamaan 2.11 berikut.

$$s(i) = \frac{(bi - ai)}{\max(ai, bi)} \quad (2.11)$$

Dimana:

$s(i) =$ Nilai Silhouette Coefficient.

Untuk menilai suatu nilai *Silhouette Coefficient* dapat dilihat pada Tabel 2.1 yang telah dibuat Kaufman dan Rousseeuw berikut (Azuri et al., 2016).

Tabel 2. 1 Interpretasi Nilai Silhouette Coefficient

Nilai Silhouette Coefficient	Interpretasi
0,71 - 1,00	Klaster yang kuat
0,51 - 0,70	Klaster telah laik atau sesuai
0,26 - 0,50	Klaster yang lemah
$\leq 0,25$	Tidak dapat dikatakan sebagai klaster

2.10. Metode Davies-Bouldin Index

Metode *Davies Bouldin-Index* merupakan suatu metode untuk mengevaluasi klaster yang ditemukan oleh David L. Davies dan Donald W. Bouldin. Evaluasi menggunakan metode *Davies Bouldin Index* dapat dilihat dari kohesi untuk mengukur berapa besar kedekatan atau jarak antara data-data yang berada pada klaster yang sama. Selain itu dilihat juga dari separasi untuk mengukur perbedaan jarak data-data yang berbeda dengan klaster lain (Saikhu & Gita, 2013).

Terdapat Langkah-langkah perhitungan *Davies Bouldin-Index* adalah sebagai berikut:

1. Sum of Squere Within Cluster (SSW)

Pada perhitungan *Sum of Squere Within Cluster* dapat mengetahui nilai kohesi dalam sebuah klaster ke I. Berikut Persamaan 2.12 untuk memperoleh nilai *Sum of Squere Within Cluster*.

$$SSW_i = \frac{i}{m_i} \sum_{j=i}^{m_i} d(x_j, c_j) \quad (2.12)$$

Dimana:

m_i = Jumlah data dalam klaster ke- i

c_i = Centroid klaster ke- i

$d(x_j, c_j)$ = Jarak setiap data ke centroid i yang dihitung menggunakan euclidean

2. Sum of Squere Between Cluster (SSB)

Pada perhitungan *Sum of Squere Between Cluster* dapat mengetahui nilai separasi antar klaster. Berikut Persamaan 2.13 untuk memperoleh nilai *Sum of Squere Between Cluster*.

$$SSB_{i,j} = d(x_i, x_j) \quad (2.13)$$

Dimana:

$d(x_i, x_j)$ = Jarak antara data ke- i dengan data ke- j pada klaster yang lain.

3. Ratio

Pada perhitungan *Ratio* ($R_{i,j}$) agar dapat mengetahui nilai perbandingan klaster ke-I dan klaster ke-j untuk menghitung nilai rasio yang dimiliki oleh masing-masing klaster. Berikut Persamaan 2.14 untuk memperoleh nilai *Ratio*.

$$R_{i,j,\dots,n} = \frac{SSW_i + SSW_j + \dots + SSW_n}{SSB_{i,j} + \dots + SSB_{n_i,n_j}} \quad (2.14)$$

Dimana:

$SSWi$ = *Sum of Square Within Cluster* pada centroid i

$SSBi,j$ = *Sum of Square Between Cluster* data ke- i dengan j pada cluster yang berbeda

4. *Davies-Bouldin Index* (DBI)

Setelah menghitung nilai ratio selanjutnya melakukan perhitungan *Davies-Bouldin Index* (DBI) dimana nilai ratio tersebut dihunakan dalam perhitungan ini. Berikut Persamaan 2.15 untuk memperoleh nilai *Davies-Bouldin Index* (DBI).

$$DBI = \frac{1}{k} \sum_{i=1}^k \max_{i \neq j} (R_{i,j,\dots,k}) \quad (2.15)$$

Dimana:

$R_{i,j}$ = Nilai ratio dari nilai SSW dan SSB

2.11. Studi Kasus

Pada subbab ini dijelaskan mengenai studi kasus dari normalisasi data, penentuan pusat klaster menggunakan metode *Elbow*, *Silhouette Coefficient* dan *Davies-Bouldin Index*. Penentuan pusat klaster menggunakan Algoritma Genetika dan klasterisasi menggunakan Algoritma *K-Means*.

2.10.1. Normalisasi Data

Proses normalisasi data ini berfungsi untuk menyamakan rentang nilai dari masing-masing atribut. Normalisasi data tersebut menggunakan Persamaan 2.1 diilustrasikan sebagai berikut.

Tabel 2. 2 Ilustrasi Normalisasi Data

No	X	Y
1	1,5	640
2	2,6	398
3	1,3	200

Pada proses normalisasi data menggunakan *Min-Max Normalization* harus mengetahui nilai terbesar dan terkecil dari masing-masing atribut. Seperti pada atribut

X nilai terbesarnya adalah 2,6 dan nilai terkecil adalah 1,3. Selanjutnya melakukan perhitungan dimana data X 1,1 merupakan data yang tepilih dikurangi nilai minimal pada atribut dibagi nilai masimal dikurangi nilai minimal. Perhitungan tersebut dapat dilihat sebagai berikut.

$$X1' = \frac{1.5 - 1.3}{2.6 - 1.3} = 0,153846154$$

$$X2' = \frac{2.6 - 1.3}{2.6 - 1.3} = 1$$

$$X3' = \frac{1.3 - 1.3}{2.6 - 1.3} = 0$$

Sama halnya dengan atribut X, pada atribut Y juga dilakukan normalisasi seperti perhitungan berikut.

$$Y1' = \frac{640 - 200}{640 - 200} = 1$$

$$Y2' = \frac{398 - 1.3}{640 - 200} = 0.45$$

$$Y3' = \frac{200 - 1.3}{640 - 200} = 0$$

Hasil dari normalisasi menggunakan *Min-Max Normalizarion* dapat dilihat seperti Tabel 2.3 berikut.

Tabel 2. 3 Ilustrasi Data Hasil Normalisasi

No	X	Y
1	0.15	1
2	1	0.45
3	0	0

2.10.2. Elbow

Metode *Elbow* pada penelitian ini berfungsi untuk menentukan jumlah klaster dari dataset yang digunakan. Terdapat beberapa tahap dalam proses penentuannya seperti ilustrasi dengan data pada Tabel 2.4 berikut.

Tabel 2. 4 Ilustrasi Data Elbow

No	X	Y
1	1,5	640
2	2,6	398
3	1,3	200
4	2,1	220
5	3,0	222

Pada proses awal ini data akan dikelompokan terlebih dahulu untuk menghasilkan nilai *Elbow* dari masing-masing kelompok. Seperti pada ilustrasi pada Tabel 2.4 akan di kelompokan menjadi 2 dengan proses pengelompokan menggunakan penentuan jarak yang lebih dekat dengan pusat klaster yang ditentukan. Pusat klaster ditentukan secara acak seperti pada Tabel 2.4 pusat klaster yang dipilih adalah data nomor 3 mewakili pusat klaster 1 dan data nomor 5 mewakili pusat klaster 2. Hasil proses pengelompokan seperti pada Tabel 2.5 berikut.

Tabel 2. 5 Ilustrasi Data Elbow Dikelompokan

No	X	Y	Cluster	Jumlah
1	1,5	640	2	641,5
2	2,6	398	2	400,6
3	1,3	200	1	201,3
4	2,1	220	2	222,1
5	3,0	222	2	225

Setelah proses pengelompokan pada metode *Elbow* selanjutnya melakukan proses pencarian nilai yang digunakan pada proses *Sum of Square Error* (SSE) pada Persamaan 2.7 seperti berikut.

$$\begin{aligned}
 \mu_1 &= (201,3)/\text{total dataset klaster} \\
 &= 201,3/1 \\
 &= 201,3 \\
 \mu_2 &= (641,5 + 400,6 + 222,1 + 225)/\text{total dataset klaster} \\
 &= 1489,2/4 \\
 &= 372,3
 \end{aligned}$$

$$\begin{aligned}
 SSE &= ((201,3 - 201,3)^2) + ((641,5 - 372,3)^2 + (400,6 - 372,3)^2 \\
 &\quad + (222,1 - 372,3)^2 + (225 - 372,3)^2) \\
 &= 117526,86
 \end{aligned}$$

Dari dataset yang dikelompokan menjadi 2 klaster didapat nilai *Sum of Square Error* (SSE) sebesar 117526,86. Pada proses ini dilakukan secara berulang dengan jumlah klaster yang bertambah sampai dengan nilai n klaster.

2.10.3. Silhouette Coefficient

Metode *Silhouette Coefficient* pada penelitian ini berfungsi untuk menentukan jumlah klaster dari dataset yang digunakan. Terdapat beberapa tahap dalam proses penentuannya seperti ilustrasi dengan data pada Tabel 2.4. Sama halnya dengan proses *Elbow*, pada proses *Silhouette Coefficient* data dikelompokan terlebih dahulu seperti pada Tabel 2.5 lalu melakukan proses menghitung rata-rata jarak antara suatu data dengan data lain pada suatu klaster seperti pada Persamaan 2.8. Proses tersebut sebagai berikut.

$$\begin{aligned}
 a(1) &= \frac{1}{4-1} \sqrt{((2,6 - 1,5)^2 + (398 - 640)^2) + ((2,1 - 1,5)^2 + (220 - 640)^2) + \\
 &\quad ((3 - 1,5)^2 + (222 - 640)^2)} \\
 &= 801,3372
 \end{aligned}$$

Pada proses menghitung rata-rata jarak antara data nomor 1 dengan data lain pada Tabel 2.4 menghasilkan nilai 801,3372. Berikut nilai rata-rata jarak pada data-data yang lain seperti pada Tabel 2.6 berikut.

Tabel 2. 6 Ilustrasi Data Hasil Rata-rata Jarak Dalam Satu Klaster

X	A(i)
1	801,337159
2	478,6700204
3	0
4	598,7321879
5	478,8626808

Setelah melakukan perhitungan rata-rata jarak antara suatu data dengan data lain pada suatu klaster, selanjutnya melakukan perhitungan rata-rata jarak suatu data dengan data lain yang klasternya berbeda seperti Persamaan 2.9 seperti berikut.

$$\begin{aligned} d(i, 1) &= \frac{1}{4} \sqrt{((1.5 - 1.3)^2 + (640 - 200)^2)} \\ &= 110,0000114 \end{aligned}$$

Hasil dari rata-rata jarak data nomor 1 yang merupakan klaster 2 terhadap data nomor 3 yang merupakan klaster 1 sebesar 110,0000114. Berikut nilai rata-rata jarak pada data-data yang lain seperti pada Tabel 2.7 berikut.

Tabel 2. 7 Ilustrasi Data Hasil Rata-rata Jarak Klaster Yang Berbeda

d(i,1)	d(i,2)
110,0000114	-
49,50106691	-
-	680,0858908
5,003998401	-
5,516396016	-

Selanjutnya proses mencari nilai minimum seperti pada Persamaan 2.10 dimana nilai jarak antara klaster 1 dan klaster 2 diambil nilai yang paling kecil seperti pada Tabel.28 berikut.

Tabel 2. 8 Ilustrasi Data Minimum

X	b(i)
1	110,0000114
2	49,50106691
3	680,0858908
4	5,003998401
5	5,516396016

Proses selanjutnya menghitung nilai *Silhouette Coefficient* seperti pada persamaan 2.11 seperti berikut.

$$\begin{aligned} S(1) &= \frac{(110 - 801.34)}{\max(110 - 801.34)} \\ &= 0,862729427 \end{aligned}$$

Dari hasil proses menghitung *Silhouette Coefficient* didapatkan nilai pada data nomor 1 adalah sebesar 0,862729427. Untuk nilai keseluruhan dilakukan rata-rata pada data hasil perhitungan *Silhouette Coefficient* yang lain seperti pada Tabel 2.9 berikut.

Tabel 2. 9 Ilustrasi Hasil Silhouette Coefficient

Data ke	s(i)
1	0,862729427
2	0,896586239
3	-1
4	0,991642343
5	0,988480213
Rata-Rata	0,547887644

Hasil perhitungan *Silhouette Coefficient* dengan klaster berjumlah 2 adalah 0,547887644. Pada proses ini dilakukan secara berulang dengan jumlah klaster yang bertambah sampai dengan nilai n klaster.

2.10.4. Davies-Bouldin Index

Metode *Davies-Bouldin Index* pada penelitian ini berfungsi untuk menentukan jumlah klaster dari dataset yang digunakan. Sama halnya dengan proses *Elbow & Silhouette Coefficient* data dikelompokan terlebih dahulu seperti pada Tabel 2.5 lalu menghitung jarak data ke pusat klaster seperti pada Tabel 10 berikut.

Tabel 2. 10 Data Ilustrasi DBI

No	X	Y	Centroid X	Centroid Y	Cluster	Jarak
1	1,5	640	2,1	220	2	420,0004286
2	2,6	398	2,1	220	2	178,0007022
3	1,3	200	1,3	200	1	0
4	2,1	220	2,1	220	2	0
5	3,0	222	2,1	220	2	2,19317122

Proses selanjutnya menghitung *Sum of Squere Within Cluster* agar mengetahui nilai kohesi dalam sebuah klaster ke i seperti pada Persamaan 2.12 berikut.

$$\begin{aligned}
 SSW1 &= (0)/\text{total data cluster} \\
 &= 0/1 \\
 &= 0
 \end{aligned}$$

$$\begin{aligned}SSW2 &= (420,0004286 + 178,0007022 + 0 + 2,19317122)/4 \\&= 150,0485755\end{aligned}$$

Selanjutnya melakukan proses mencari nilai *Sum of Squere Between Cluster* seperti pada Persamaan 2.13 berikut.

$$\begin{aligned}SSB1,2 &= \sqrt{(1,3 - 2,1)^2 + (200 - 220)^2} \\&= 20,01599361\end{aligned}$$

Setelah mendapatkan nilai *Sum of Squere Within Cluster* dan *Sum of Squere Between Cluster* selanjutnya melakukan perhitungan ratio seperti pada Persamaan 2.14 berikut.

$$\begin{aligned}Ratio &= (0 + 150,0485755)/20,01599361 \\&= 7,496434025\end{aligned}$$

Proses perhitungan selanjutnya adalah perhitungan *Davies-Bouldin Index* seperti pada Persamaan 2.15 berikut.

$$\begin{aligned}DBI &= (1 \times 7,496434025)/2 \\&= 3,748217013\end{aligned}$$

Hasil perhitungan *Davies-Bouldin Index* dengan jumlah klaster sebanyak 2 dihasilkan nilai DBI sebesar 3,748217013.

2.10.5. Algoritma Genetika

Pada penelitian ini Algoritma Genetika merupakan suatu metode yang dapat megoptimalkan penentuan pusat klaster untuk digunakan pada proses klasterisasi di Algoritma *K-Means*. Terdapat beberapa masukan dalam proses Algoritma Genetika diantaranya nilai probabilitas *crossover*, nilai probabilitas mutasi, jumlah individual, jumlah iterasi yang dilakukan dan data yang digunakan pada proses penentuan pusat klaster seperti ilustrasi pada Tabel 2.11 berikut.

Tabel 2. 11 Ilustrasi Data Algoritma Genetika

No	CO	HC
1	0,8	1
2	0,22	0
3	0,02	0,142857143
4	1	0,428571429
5	0	0,107142857

Tahap pertama dalam proses penentuan pusat klaster yaitu melakukan evaluasi *fitness* dimana tiap gen pada satu kromosom dijumlahkan seperti berikut.

- $Fitness$ Kromosom Ke-1 = $0,8 + 1 = 1,8$
- $Fitness$ Kromosom Ke-2 = $0,22 + 0 = 0,22$
- $Fitness$ Kromosom Ke-3 = $0,02 + 0,142857143 = 0,162857143$
- $Fitness$ Kromosom Ke-4 = $1 + 0,428571429 = 1,428571429$
- $Fitness$ Kromosom Ke-5 = $0 + 0,107142857 = 0,107142857$

Setelah melakukan proses evaluasi *fitness* selanjutnya melakukan seleksi dimana proses ini untuk mendapatkan calon induk yang baik agar mendapatkan keturunan yang baik. Seleksi pada penelitian ini menggunakan seleksi *ranking* (*rank based selection*) dimana pada seleksi ini memberikan nilai *fitness* baru untuk masing-masing kromosom berdasarkan *ranking fitnessnya*. Ilustrasi pada proses seleksi adalah sebagai berikut.

- Kromosom 1 = 1,8
- Kromosom 2 = 0,22
- Kromosom 3 = 0,162857143
- Kromosom 4 = 1,428571429
- Kromosom 5 = 0,107142857

Dari hasil melakukan evaluasi *fitness* dilakukan pengurutan atau *ranking* dimana pada proses ini mendapatkan nilai *fitness* baru untuk masing-masing kromosom. Kromosom dengan nilai *fitness* terkecil mendapatkan nilai *fitness* baru sebesar 1 lalu pada nilai *fitness* terkecil ke 2 mendapatkan nilai *fitness* baru sebesar 2 sampai nilai *fitness* terbesar mendapatkan nilai sebesar nilai n seperti berikut.

- *Rangking 1* – Kromosom 1 = 1,8 (Nilai *fitness* terbaru 5)
- *Rangking 2* – Kromosom 4 = 1,428571429 (Nilai *fitness* terbaru 4)
- *Rangking 3* – Kromosom 2 = 0,22 (Nilai *fitness* terbaru 3)
- *Rangking 4* – Kromosom 3 = 0,162857143 (Nilai *fitness* terbaru 2)
- *Rangking 5* – Kromosom 5 = 0,107142857 (Nilai *fitness* terbaru 1)

Setelah melakukan pengurutan atau *rangking* selanjutnya mencari nilai probabilitas seperti berikut.

- $P[i] = Fitness[i]/Total\ Fitness$
- $P[1] = 5/(5+4+3+2+1) = 0,333333333$
- $P[2] = 4/(5+4+3+2+1) = 0,266666667$
- $P[3] = 3/(5+4+3+2+1) = 0,2$
- $P[4] = 2/(5+4+3+2+1) = 0,133333333$
- $P[5] = 1/(5+4+3+2+1) = 0,066666667$

Selanjutnya melakukan proses pencarian *fitness* komulatif dari nilai probabilitas sebelumnya yang telah ditentukan.

- $Q[i] = \text{Komulatif } [i] + \text{Probabilitas } [i]$
- $Q[1] = 0 + 0,333333333 = 0,333333333$
- $Q[2] = 0,333333333 + 0,266666667 = 0,6$
- $Q[3] = 0,6 + 0,2 = 0,8$
- $Q[4] = 0,8 + 0,133333333 = 0,933333333$
- $Q[5] = 0,933333333 + 0,066666667 = 1$

Proses pencarian *fitness* komulatif merupakan proses mencari jumlah dari setiap nilai probabilitas seleksi. Setelah mencari *fitness* komulatif selanjutnya membangkitkan nilai acak dari *range* 0 sampai 1 seperti berikut.

- $R[1] = 0,476815501$
- $R[2] = 0,280548705$
- $R[3] = 0,548259517$
- $R[4] = 0,95683984$
- $R[5] = 0,074286614$

Nilai acak digunakan untuk membandingkan dengan nilai komulatif dimana dipilih nilai komulatif yang mendekati nilai acak ($Q > R$) untuk menentukan kromosom nilai baru seperti pada Tabel 2.12 berikut.

Tabel 2. 12 Ilustrasi Membandingkan Nilai Komulatif dan Nilai Acak

No	Bilangan Acak	Q>R
1	0,476815501	0,6
2	0,280548705	0,333333333
3	0,548259517	0,6
4	0,95683984	1
5	0,074286614	0,333333333

Populasi baru yang terbentuk dapat dilihat sebagai berikut.

- Kromosom 1' = Kromosom 2
- Kromosom 2' = Kromosom 1
- Kromosom 3' = Kromosom 2
- Kromosom 4' = Kromosom 5
- Kromosom 5' = Kromosom 1

Sehingga hasil seleksi dan kromosom baru dapat dilihat seperti pada Tabel 2.13 berikut.

Tabel 2. 13 Ilustrasi Hasil Seleksi

No	X	Y	Nilai Fitness	Asal
1	0,22	0	0,22	K2
2	0,8	1	1,8	K1
3	0,22	0	0,22	K2
4	0	0,107142857	0,107142857	K5
5	0,8	1	1,8	K1

Selanjutnya melakukan proses penyilangan atau *crossover* dimana jika nilai probabilitas *crossover* sebesar 0,5 maka diharapkan 50% dari total akan mengalami *crossover*. Tahap pertama yaitu membangkitkan nilai acak sebanyak jumlah populasi yang ditentukan yaitu 5.

- R1 = 0,092138348
- R2 = 0,325572148
- R3 = 0,736209531
- R4 = 0,668950222
- R5 = 0,349119073

Data Kromosom yang terpilih untuk *crossover*.

- $R1 = 0,342838348$
- $R3 = 0,036209531$

Dari proses pembangkitan nilai acak, terpilih kromosom 1 dan 3 yang dilakukan *crossover*. Selanjutnya variabel dilakukan perubahan kepada biner dengan *encoding* lalu melakukan *crossover* satu titik seperti pada Tabel 2.14 dan hasil *crossover* seperti pada Tabel 2.15, proses *crossover* tersebut sebagai berikut.

Tabel 2. 14 Ilustrasi Crossover

Parent	X								Y							
Parent 1	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0
Parent 2	1	1	0	0	1	1	0	0	1	1	1	1	1	1	1	1

Tabel 2. 15 Ilustrasi Hasil Crossover

Child	X								Y							
Child 1	0	0	1	1	1	1	0	0	1	1	1	1	1	1	1	1
Child 2	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0

Setelah melakukan *crossover* selanjutnya melakukan mutasi dimana nilai mutasi didapat saat proses masukan, jika probabilitas mutasi sebesar 0,1 maka diharapkan 1% dari total gen akan dilakukan mutasi. Diketahui Panjang total gen sebesar 80 lalu melakukan mutasi sebesar 0,1 atau 10% dari 80 yaitu sebanyak 8 data yang termutasi dengan penentuan gen yang terkena mutasi dilakukan secara acak melalui bilangan acak yang telah dibangkitkan. Ilustrasi tersebut seperti Tabel 2.16 berikut.

Tabel 2. 16 Ilustrasi Mutasi

Parent/Child	X								Y							
Parent 1	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0
Parent 2	1	1	0	0	1	1	0	0	1	1	1	1	1	1	1	1

Child 1	0	0	1	1	1	1	0	0	1	1	1	1	1	1	1	1
Child 2	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0

Pada gen yang terkena mutasi dilakukan perubahan dimana jika nilai awalnya ‘1’ maka diubah menjadi ‘0’ dan sebaliknya jika nilai awalnya ‘0’ diubah menjadi ‘1’. Ilustrasi tersebut seperti 2.17 berikut.

Tabel 2. 17 Ilustrasi Hasil Mutasi

Parent/Child	X										Y						
	0	1	1	1	1	0	0	0	0	0	0	0	0	0	1	0	0
Parent 1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	1	0	0
Parent 2	1	1	0	0	1	0	0	0	1	0	1	1	1	1	1	1	1
Child 1	0	0	1	0	1	1	0	0	1	1	1	1	1	1	1	0	1
Child 2	1	1	0	0	1	0	0	0	0	0	1	0	0	0	1	0	0

Selanjutnya melakukan perubahan nilai *decoding* yang sebelumnya dari biner menjadi desimal dengan hasil seperti pada Tabel 2.18 sebagai berikut.

Tabel 2. 18 Ilustrasi Hasil Evaluasi Fitness

No	X	Y	Nilai Fitness
P1	0,470588235	0,015686275	0,48627451
P2	0,784313725	0,749019608	1,533333333
C1	0,17254902	0,992156863	1,164705882
C2	0,784313725	0,133333333	0,917647059

Setelah melakukan mutasi dan hasilnya seperti pada Tabel 2.18 selanjutnya melakukan evaluasi *fitness* dimana pada penelitian ini adalah nilai *child* yang memiliki nilai *fitness* paling besar untuk digunakan menjadi pusat klaster menggunakan Algoritma K-Means sesuai jumlah K yang dimasukan saat proses awal Algoritma Genetika.

2.10.6. Algoritma K-Means

Pada penelitian ini Algoritma *K-Means* bertujuan untuk melakukan klasterisasi dengan pendekatan menggunakan *Euclidian Distance* dalam proses penentuan jarak antar data dengan pusat klaster. Proses penentuan pusat klaster pada Algoritma *K-Means* dilakukan secara acak tetapi pada penelitian ini pusat klaster ditentukan oleh

Algoritma Genetika. Data yang digunakan pada ilustrasi adalah Tabel 2.11 proses menghitung jarak antara suatu data dengan pusat klaster menggunakan Persamaan 2.5 sebagai.

Tabel 2. 19 Ilustrasi Pusat Klaster

C1	0,17254902	0,992156863
C2	0,784313725	0,133333333

$$d(1,1) = \sqrt{(0,8 - 0,17254902)^2 + (1 - 0,992156863)^2}$$

$$= 0,627499998$$

$$d(1,2) = \sqrt{(0,8 - 0,784313725)^2 + (1 - 0,133333333)^2}$$

$$= 0,866808612$$

Selanjutnya melakukan pencarian nilai jarak terkecil antara data dengan pusat klaster seperti Persamaan 2.6 dengan hasil seperti pada Tabel 2.20 berikut.

Tabel 2. 20 Ilustrasi Hasil Klasterisasi

No	Jarak Ke klaster		Hasil	Klaster
	C1	C2		
1	0,627499998	0,866808612	0,627499998	1
2	0,993290912	0,579851497	0,579851497	2
3	0,862891197	0,764373059	0,764373059	2
4	1,001151171	0,365631101	0,365631101	2
5	0,901677855	0,784750891	0,784750891	2

Hasil dari klasterisasi menggunakan Algoritma *K-Means* dengan pendekatan penentuan jarak data dengan pusat klaster menggunakan *Euclidian Distance* seperti pada Tabel 2.20 dimana dari hasil klasterisasi tersebut mengambil nilai jarak data minimum.