

KLASIFIKASI KEBUTUHAN JUMLAH PRODUK MAKANAN *CUSTOMER* MENGUNAKAN *K-MEANS CLUSTERING* DENGAN OPTIMASI PUSAT AWAL *CLUSTER* ALGORITMA GENETIKA

Yudi Istianto^{*1}, Shofwatul 'Uyun²

^{1,2} Universitas Islam Negeri Sunan Kalijaga Yogyakarta
Email: ¹yudiistantoo@gmail.com, ²shofwatul.uyun@uin-suka.ac.id
^{*}Penulis Korespondensi

(Naskah masuk: 01 Januari 2020, diterima untuk diterbitkan: 15 Oktober 2021)

Abstrak

PT. Harum Bakery adalah salah satu perusahaan di Yogyakarta yang bergerak pada bidang produksi dan distribusi produk makanan roti. Setiap konsumen memiliki jumlah kebutuhan roti yang tidak teratur, sedangkan roti hanya dapat bertahan dalam waktu dua hari. Roti yang sudah berusia lebih dari dua hari akan diganti dengan yang baru oleh distributor, sehingga dapat menimbulkan kerugian bagi perusahaan. Penelitian ini mencoba untuk melakukan data mining dengan tujuan mengklasifikasikan jumlah produk makanan kepada *customer* menggunakan *k-means clustering* dengan optimasi pusat awal *cluster* algoritma genetika. Pada penelitian ini digunakan 210 data dari penjualan produk selama tiga minggu. Data tersebut akan diproses dengan menerapkan metode data mining melalui tahap *preprocessing* kemudian tahap klasifikasi. *Preprocessing* yang dilakukan antara lain, data *transformation* dan *k-means clustering*. Hasil dari *clustering* yang membutuhkan aturan tertentu lebih efektif dengan optimasi karena dari 210 data terdapat 200 data yang layak masuk tahap klasifikasi. Hasil dari pengujian mendapatkan akurasi terbaik sebesar 58.50 % dan *crossvalidation* untuk lima *fold* berhasil mendapatkan rata-rata akurasi sebesar 50.58% lebih besar 2.51 % dari KNN tanpa *preprocessing*.

Kata kunci: *K-means, Clustering, Algoritma Genetika*

CLASSIFICATION OF NEEDS FOR THE NUMBER OF CUSTOMER FOOD PRODUCTS USING *K-MEANS CLUSTERING* WITH OPTIMIZATION INITIAL CLUSTER CENTER GENETIC ALGORITHM

Abstract

PT. Harum Bakery is one of the companies in Yogyakarta engaged in the production and distribution of bakery food products. Every consumer has an irregular amount of bread needs while bread can only last for two days. Bread that is more than two days old will be replaced by a new one by the distributor which causes losses for the company. This study tries to apply data mining to classify the number of customer needs for food products using *k-means clustering* with optimization initial cluster center genetic algorithm. In this study used 210 data from product sales for three weeks. Data will be processed by applying data mining method with *preprocessing* before going through classification. *Preprocessing* includes data *transformation* and *k-means clustering*. The results of *clustering* that require certain rules are more effective with optimization because 210 data have 200 data that are worth entering the classification stage. The results of the test get the best accuracy of 58.50% and *crossvalidation* for five fold managed to get an average accuracy of 50.58% greater than 2.51% of KNN without *preprocessing*.

Keywords: *K-means, Clustering, Genetik Algoritma*

1. PENDAHULUAN

Perkembangan teknologi telah membawa masyarakat dalam perubahan hidup yang lebih baik. Teknologi tidak hanya sekedar pelengkap namun telah menjadi kebutuhan hidup manusia yang tidak dapat terpisahkan. Komputer adalah salah satu teknologi yang digunakan untuk mempermudah

kehidupan manusia dalam semua bidang tidak terkecuali bidang industri. Pemanfaatan ilmu komputer sangat berperan dalam penggunaan komputer yang lebih baik. Salah satu ilmu komputer yang sangat berperan ini adalah data mining. Data mining merupakan serangkaian proses untuk menggali nilai tambah berupa informasi yang selama

ini tidak diketahui secara manual dari suatu basisdata dengan melakukan penggalian pola-pola dari data dengan tujuan untuk memanipulasi data menjadi informasi yang lebih berharga yang diperoleh dengan cara mengekstraksi dan mengenali pola yang penting atau menarik dari data yang terdapat dalam basis data.

Salah satu metode yang digunakan dalam pengolahan data adalah pengelompokan/*clustering* (Rustiyan & Mustakim, 2018). Metode algoritma K-means adalah metode *clustering* yang paling populer digunakan karena beberapa kelebihan yang dimiliki antara lain algoritma ini sederhana dan mudah diterapkan (Widiarina, 2013). Selain itu algoritma K-means tidak terpengaruh dengan adanya urutan objek (Aranda & Natasya, 2016). Namun, K-means mempunyai kelemahan yang diakibatkan oleh penentuan pusat awal *cluster*. Hal ini menyebabkan hasil *cluster*-nya berupa solusi yang sifatnya *local optimal*. Menggunakan metode hierarki tertentu dapat dikolaborasikan untuk optimasi pusat awal *cluster* K-means. Pada penelitian yang dilakukan oleh Asroni, dkk dengan K-means *clustering* data penerimaan calon mahasiswa baru fakultas kedokteran dan ilmu keperawatan mampu mengelompokkan data menjadi empat kelompok dengan jumlah iterasi terbaik sebanyak tiga kali disimpulkan bahwa pendidikan dokter dan ilmu hubungan internasional menjadi jurusan pilihan calon mahasiswa baru Universitas Muhammadiyah Yogyakarta (Asroni, Fitri, & Prasetyo, 2018). Pada penelitian yang dilakukan Aranda dan Natasya dengan K-means clustering pada sistem pendukung keputusan pemilihan konsentrasi untuk mahasiswa international class STMIK Amikom Yogyakarta berhasil diterapkan dengan jumlah iterasi terbaik sebanyak tiga kali (Aranda & Natasya, 2016).

Salah satu algoritma yang dapat digunakan untuk optimasi adalah algoritma genetika. Algoritma Genetika merupakan metode pencarian solusi yang sesuai dengan kriteria dari banyak kombinasi solusi yang ada tanpa harus menguji satu-persatu untuk mendapatkannya. Tahap proses siklus genetika diawali dengan membangkitkan populasi awal yaitu pembangkitan sejumlah individu secara acak. Pada kasus penentuan komposisi bahan pangan untuk diet penyakit ginjal dan saluran kemih jumlah kromosom yang dibangkitkan sebanyak 20 kromosom ('Uyun & Hartati, 2011). Pada optimasi pengantaran barang dalam perdagangan online dibangkitkan dalam satu populasi sebanyak 8 kromosom dengan panjang kromosom sebanyak 40 gen (Pratama, Djamal, & Komarudin, 2017). Penelitian sebelumnya dalam optimasi neural network dengan algoritma genetika untuk prediksi hasil pemilukada pengujian dengan menggunakan neural network berbasis algoritma genetika didapatkan nilai accuracy 93.03 % dengan nilai precision 91.28 % dan nilai AUC adalah 0.971 (Badrul, 2016). Algoritma genetika telah terbukti menjadi metode yang efektif untuk digunakan dalam

data mining dan pengenalan pola, Sebagian besar aplikasi algoritma genetika dalam pengenalan pola mengoptimalkan beberapa parameter dalam proses klasifikasi, algoritma genetika telah diterapkan untuk menemukan set optimal bobot fitur yang meningkatkan akurasi klasifikasi (Cantú-Paz, 2003).

Klasifikasi merupakan salah satu teknik dalam data mining. Klasifikasi adalah pemrosesan untuk menemukan sebuah model atau fungsi yang menjelaskan dan mencirikan konsep atau kelas data, untuk kepentingan tertentu (Java, Wisdayani, Nur, Wasono, & Semarang, n.d.). Salah satu algoritma klasifikasi yang cukup populer adalah *k-nearest neighbor* (KNN). Dalam pengenalan pola, algoritma *k-nearest neighbor* adalah salah satu metode non-parametrik yang paling terkenal dan berguna untuk mengelompokkan objek berdasarkan fitur-fitur yang dekat. *k-nearest neighbor* dirancang dengan konsep bahwa label atau kelas ditentukan oleh suara mayoritas tetangganya (Won Yoon & Friel, 2015).

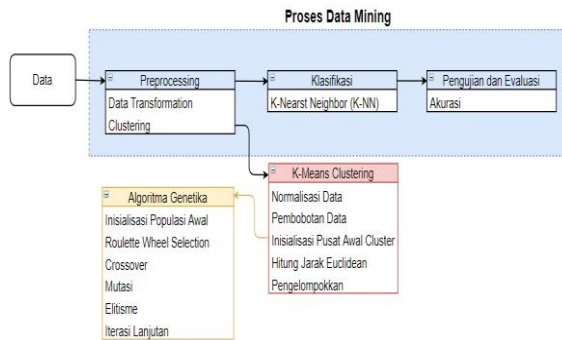
PT. Harum Bakery adalah salah satu perusahaan di Yogyakarta yang bergerak pada bidang produksi dan distribusi makanan roti. PT. Harum Bakery pada saat ini mampu memproduksi kurang lebih 1000 roti dalam sehari, sebelum hari selanjutnya mereka distribusikan ke toko-toko dan pedagang jajanan pasar. Pada saat ini jumlah konsumen telah mencapai lebih dari 50 konsumen. Setiap konsumen memiliki jumlah kebutuhan roti yang tidak teratur setiap hari sedangkan roti hanya dapat bertahan dalam waktu dua hari. Roti yang sudah berusia lebih dari dua hari akan diganti dengan yang baru oleh distributor yang bisa menimbulkan kerugian bagi perusahaan. Berdasarkan latar belakang di atas maka peneliti tertarik untuk melakukan penelitian mengenai data mining untuk klasifikasi kebutuhan jumlah produk makanan *customer* menggunakan *k-means clustering* dengan optimasi pusat awal *cluster* algoritma genetika yang bermanfaat untuk mempermudah prediksi jumlah kebutuhan *customer* dimasa mendatang berdasarkan data penjualan masa lampau pada PT. Harum Bakery.

2. METODE PENELITIAN

2.1. Jenis dan Desain Penelitian

Metode penelitian yang digunakan adalah metode analisis deskriptif dengan pendekatan kuantitatif yang menekankan analisisnya terhadap data-data numerik (angka). Masalah yang diteliti dalam penelitian ini berkaitan dengan data mining data penjualan produk. Data penjualan tersebut dianalisis dan diolah terlebih dahulu sehingga dapat dimengerti oleh komputer. Selanjutnya dilakukan proses data *transformation* dan *clustering* menggunakan metode K-means dengan optimasi penentuan pusat awal cluster dengan menggunakan algoritma genetika. Data penjualan yang telah diklaster selanjutnya diklasifikasi dengan algoritma *k-nearest neighbors* (KNN). Terakhir untuk mengetahui seberapa baik algoritma yang digunakan

dilakukan pengujian dan evaluasi. Tahapan yang dilakukan pada penelitian disajikan pada Gambar 2. Proses Data Mining.



Gambar 1. Proses Data Mining

2.2. Data

Data merupakan penjualan produk dari distributor PT. Harum Bakery ke *customer*. Data tersebut merupakan penjualan produk selama tiga minggu di bulan Juli 2019. Data ini memiliki atribut antara lain disajikan pada Tabel 1. Atribut Data.

Tabel 1. Atribut Data

No	Atribut	Penjelasan
1	Kode <i>Customer</i>	Kode dari <i>customer</i> yang diposisikan sebagai kode yang mendeskripsikan identitas <i>customer</i> .
2	Jenis Hari	Jenis dari hari apakah merupakan hari kerja atau hari libur.
3	Cuaca	Kondisi cuaca ketika transaksi.
4	Hari	Nama hari ketika transaksi.
5	Jual	Jumlah penjualan produk ketika transaksi
6	Retur	Jumlah pengembalian produk ketika transaksi
7	Optimal	Jual dikurangi dengan retur.

2.3. Preprocessing

Pada tahap ini dilakukan untuk menyeleksi data sebelum dilakukan klasifikasi dengan tujuan meningkatkan akurasi pada tahap klasifikasi. Pada tahapan ini dilakukan antara lain data *transformation* dan *clustering* menggunakan *k-means* untuk mendapatkan data latih yang baik (layak).

2.3.1. Data Transformation

Pada tahap ini dilakukan pengkodean dan transformasi kode atribut sesuai dengan Tabel 2. Kode dan Transformasi Atribut.

Tabel 2. Kode dan Transformasi Atribut

Atribut	Kode	Transform
Kode <i>Customer</i>	C01	1
	C02	2
	C03	3
	C04	4
	C05	5
	C06	6
	C07	7
	C08	8
	C09	9
	C10	10
Jenis Hari	Hari Libur	1
	Bukan Hari Libur	2
Cuaca	Hujan	1

Gerimis	2
Cerah	3
Berawan	4

Atribut	Kode	Transform.
Hari	Senin	1
	Selasa	2
	Rabu	3
	Kamis	4
	Jumat	5
	Sabtu	6
	Minggu	7
Jumlah Jual	Tetap	Tetap
Jumlah Retur	Tetap	Tetap
Jumlah Optimal	Tetap	Tetap

2.3.2. K-means Clustering

Pada penelitian ini menggunakan *k-means* untuk melakukan *clustering* data. Pendekatan untuk membuat *cluster* mendikte keanggotaan dalam grup berdasarkan tingkat persamaan. Setiap *cluster* haruslah memiliki rentang nilai tertentu berdasarkan aturan (*rule*) dari Tabel 3. *Rule Cluster*.

Tabel 3. Rule Cluster

Jumlah Optimal	Cluster
optimal <= 10	C1
optimal > 10 & optimal <= 20	C2
optimal > 20 & optimal <= 30	C3
optimal > 30 & optimal <= 40	C4
optimal > 40 & optimal <= 50	C5

2.3.2.1. Normalisasi Data

Suatu data nilai atributnya haruslah bernilai setara untuk dilakukan *clustering*, hal ini akan sangat mempengaruhi hasil pengelompokkan data nantinya. Pada tahap normalisasi data pada penelitian ini rasio normalisasi atribut diusulkan persamaan sebagai berikut Tabel 4. Rasio Normalisasi Atribut.

Tabel 4. Rasio Normalisasi Atribut

Atribut	Normalisasi	Kode
Kode <i>Customer</i>	Jumlah optimal per <i>customer</i> / jumlah transaksi per <i>customer</i>	A1
Jenis Hari	Jumlah optimal per jenis hari / jumlah transaksi per jenis hari	A2
Cuaca	Jumlah optimal per cuaca / jumlah transaksi per cuaca	A3
Hari	Jumlah optimal per hari / jumlah transaksi per hari	A4
Jual	Tetap	A5
Retur	Tetap	A6
Optimal	Tetap	A7

2.3.2.2. Pembobotan Data

Pembobotan data digunakan untuk mendapatkan bobot yang nantinya akan berguna ketika proses inisialisasi pusat awal *cluster*. Bobot ini berasal dari nilai atribut-atribut pada record suatu data. Berikut ini adalah persamaan yang (1):

$$\text{Bobot} = \sqrt{A1^2 + A2^2 + \dots + An^2} \quad (1)$$

2.3.2.3. Inisialisasi Pusat Awal Cluster

Penelitian ini mencoba melakukan optimasi pada langkah pembangkitan pusat awal *cluster* menggunakan algoritma genetika. Dipilih sebanyak lima data random yang berbeda selanjutnya akan dihitung *fitness* terbaik yang akan menjadi lima pusat *cluster* terbaik.

Populasi awal terdiri dari empat kromosom dengan lima gen yang mewakili anggota kelompok *cluster*. Berdasarkan *rule* sesuai Tabel 3. *Rule Cluster* dibuatlah populasi awal dengan usulan Tabel 5. Pembentukan Kromosom.

Tabel 5. Pembentukan Kromosom

No	Krom.	Gen 1	Gen 2	Gen 3	Gen 4	Gen 5
1	Krom.1	G11	G21	G31	G41	G51
2	Krom.2	G12	G22	G32	G42	G52
...
n	Krom.n	G1n	G2n	G3n	G4n	G5n

Evaluasi digunakan untuk menghitung *fitness* pada tiap kromosom. Evaluasi nilai *fitness* dalam penelitian ini menggunakan rumus aritmatika dengan tetap memperhatikan *rule*. Tujuannya adalah mencari deret angka yang hilang berdasarkan dengan angka yang telah diketahui. Berikut ini adalah persamaan yang diusulkan untuk menghitung *fitness* (2):

1. Inisialisasi nilai N : (2)

$$N = \frac{(Gen_n - Gen_1)}{(Jumlah\ Gen - 1)}$$

2. Inisialisasi nilai sejumlah Gen :

$$C_2 = Gen_2 + (N * (2 - 1))$$

...

$$C_{(n-1)} = Gen_{(n-1)} + (N * (n - 1))$$

3. Menentukan *fitness* kromosom:

$$\frac{1}{abs(Gen_2 - C_2) + \dots + abs(Gen_{(n-1)} - C_{(n-1)}) + 0.01}$$

2.3.2.4. Hitung Jarak Euclidean

Berikut adalah persamaan dalam menghitung jarak *euclidean* pada tahap *k-means clustering* (3).

$$d(P, Q) = \sqrt{\sum_{j=1}^p (x_j(P) - x_j(Q))^2} \quad (3)$$

2.3.2.5. Pengelompokkan

Pada tahap ini setelah proses hitung jarak data dikelompokkan sesuai dengan jarak terdekat dengan pusat *cluster*. Setelah data dikelompokkan selanjutnya adalah menilai apakah *cluster* yang dihasilnya sesuai dengan *rule cluster* Tabel 3. Data *cluster* yang tidak sesuai dengan *rule* dianggap tidak layak untuk dijadikan data latih pada tahap klasifikasi.

3.4. K-Nearest Neighbor (K-NN) Classification

Algoritma klasifikasi yang digunakan pada penelitian ini adalah *k-nearest neighbors* (KNN).

Prinsip kerja algoritma KNN adalah mencari jarak terdekat antara data yang dievaluasi dengan *k* tetangga terdekatnya. Berikut adalah persamaan dalam menghitung jarak *euclidean* klasifikasi (4).

$$d_i = \sqrt{\sum_{i=1}^p (x_{1i} - x_{2i})^2} \quad (4)$$

2.5. Pengujian dan Evaluasi

Di dalam mengukur kinerja suatu algoritma terhadap kasus tertentu umumnya dilakukan pengujian dan evaluasi. Pengujian dilakukan dengan data uji yang telah disiapkan. Pada klasifikasi data kinerja model yang dihasilkan akan menggambarkan sejauh mana model tersebut dapat mengklasifikasikan data ke dalam kelas-kelas tertentu. Pada penelitian ini untuk mengukur kinerja model yang digunakan adalah *confusion matrix*.

Confusion matrix adalah suatu metode yang digunakan untuk melakukan perhitungan akurasi pada konsep data mining. Akurasi merupakan suatu cara yang biasa digunakan untuk mengukur kinerja sistem klasifikasi. Perhitungan akurasi bertujuan untuk memperkirakan seberapa efektif algoritma tersebut dengan menunjukkan probabilitas nilai sebenarnya (*actual*) dan keseluruhan label kelas. Semakin tinggi nilai akurasi menandakan semakin baik model klasifikasi yang digunakan. Akurasi didefinisikan melalui persamaan (5) (Sokolova, Japkowicz, & Szpakowicz, 2006).

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (5)$$

Pada pengukuran kinerja menggunakan *confusion matrix* terdapat empat istilah yaitu tersebut adalah *True Positive (TP)*, *True Negative (TN)*, *False Positive (FP)*, dan *False Negative (FN)*. Nilai *True Negative (TN)* merupakan jumlah data negatif yang terdeteksi dengan benar, sedangkan *False Positive (FP)* merupakan data negatif namun terdeteksi sebagai data positif. Sementara itu, *True Positive (TP)* merupakan data positif yang terdeteksi benar. *False Negative (FN)* merupakan kebalikan dari *True Positive*, sehingga data positif namun terdeteksi sebagai data negatif.

3. LANDASAN TEORI

3.1. K-Means

K-means merupakan salah satu algoritma dalam data mining yang bisa digunakan untuk melakukan *clustering* suatu data. Terdapat banyak pendekatan untuk membuat *cluster*, di antaranya adalah membuat aturan yang mendikte keanggotaan dalam grup yang sama berdasarkan tingkat persamaan di antara anggota-anggotanya. Pendekatan lainnya adalah dengan membuat sekumpulan fungsi yang mengukur beberapa properti dari pengelompokan tersebut sebagai fungsi dari beberapa parameter dari sebuah *clustering*. Metode *k-means* adalah metode yang termasuk dalam algoritma *clustering* berbasis

jarak yang membagi data ke dalam sejumlah *cluster* dan algoritma ini hanya bekerja pada atribut numerik. Dasar algoritma *k-means* adalah sebagai berikut (Xu, 2010):

1. Tentukan nilai *k* sebagai jumlah klaster yang ingin dibentuk.
2. Inisialisasi *k* sebagai *centroid* yang dapat dibangkitkan secara random.
3. Hitung jarak setiap data ke masing-masing *centroid* menggunakan persamaan *Euclidean Distance*.
4. Kelompokkan setiap data berdasarkan jarak terdekat antara data dengan *centroidnya*.
5. Tentukan posisi *centroid* baru.

Kembali ke langkah 3 jika posisi *centroid* baru dengan *centroid* lama tidak sama.

3.2. Algoritma Genetika

Algoritma Genetika sebagai cabang dari algoritma evolusi merupakan metode yang digunakan untuk memecahkan suatu pencarian nilai dalam permasalahan-permasalahan optimasi nonlinier (Gen & Cheng, 1999). Algoritma Genetika direpresentasikan oleh urutan langkah-langkah prosedur kromosom buatan yang bergerak dari satu populasi ke populasi baru menggunakan seleksi alami. Algoritma Genetika berbeda dengan metode konvergensi umum yang bersifat deterministik (Gen & Cheng, 1999). Algoritma Genetika tidak berbeda dengan seleksi alam dan ilmu genetika dikarenakan algoritma ini menggunakan seleksi individu terbaik. Menurut Haupt dan Haupt (Randy & Sue, 2004), struktur dasar algoritma genetika terdiri atas beberapa langkah yaitu sebagai berikut:

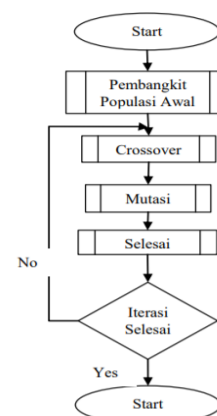
1. Inisialisasi Populasi.
2. Evaluasi Populasi.
3. Seleksi Populasi yang akan dikenai operator genetika.
4. Proses penyilangan pasangan kromosom tertentu.
5. Proses mutasi kromosom tertentu.
6. Evaluasi populasi baru (*k*).
7. Ulangi dari langkah 3 selama syarat berhenti belum terpenuhi.

Populasi awal pada Algoritma Genetika dibangkitkan secara acak, sedangkan populasi selanjutnya dibangkit oleh operator-operator Algoritma Genetika dari proses-proses pada generasi sebelumnya. Generasi berikutnya dibentuk dengan melalui serangkaian proses yang mirip dengan seleksi proses alamiah. Sebagian dari generasi berikutnya dibentuk dari kromosom-kromosom yang bertahan dari generasi sebelumnya, dan sebagian lainnya dilahirkan dari kromosom generasi sebelumnya.

Seperti halnya proses seleksi alam, maka dalam Algoritma Genetika juga terdapat proses yang mirip dengan proses seleksi alam yang terjadi pada setiap generasi. Jika dalam proses alamiah, hanya individu

yang bugar saja yang bertahan hidup dalam populasinya, maka demikian juga dalam proses komputasi. Pada setiap generasi, kromosom-kromosom akan mengalami evaluasi. Evaluasi digunakan untuk menghitung *fitness* pada tiap kromosom (Mu'asyaroh & Mahmudy, 2016). Nilai *fitness* merupakan suatu ukuran kualitas dari tiap kromosom (Mu'asyaroh & Mahmudy, 2016). Bentuk fungsi *fitness* bergantung pada masalah yang akan dioptimalisasi dan akan dibahas secara tersendiri. Nilai *fitness* dari suatu kromosom akan menunjukkan kualitas kromosom dalam populasi. Kromosom yang memiliki nilai *fitness* lebih besar akan memiliki peluang lebih besar juga dibandingkan dengan kromosom yang memiliki nilai *fitness* lebih kecil ('Uyun & Hartati, 2011). Hal ini bertujuan agar populasi yang baru dibentuk dapat menghasilkan solusi yang lebih baik daripada kumpulan solusi dari populasi yang lama (Wicaksono, 2019).

Populasi generasi yang baru dibentuk dengan cara menyeleksi nilai *fitness* dari kromosom anak (*offspring*) (Badrul, 2016). Generasi berikutnya dikenal dengan istilah anak (*offspring*) terbentuk dari gabungan dua kromosom generasi sekarang yang bertindak sebagai induk (*parent*) dengan menggunakan operator penyilangan (*crossover*) (Badrul, 2016). Beberapa operator genetika yang paling populer antara lain: (1) *selection*, (2) *crossover* dan (3) *mutation* ('Uyun & Hartati, 2011). Setelah melewati sekian iterasi, akan didapatkan individu terbaik. Individu terbaik ini akan mempunyai susunan kromosom yang bisa dikonversi menjadi solusi yang paling baik atau mendekati optimum (Mu'asyaroh & Mahmudy, 2016). Berikut adalah cara kerja Algoritma Genetika dapat dilihat pada Gambar 2. *Flowchart* Algoritma Genetika.



Gambar 2. *Flowchart* Algoritma Genetika

4. HASIL DAN PEMBAHASAN

4.1. Data

Sumber data utama yang digunakan dalam penelitian ini adalah data penjualan PT. Harum Bakery dipilih sejumlah sepuluh customer yang mempunyai konsisten melakukan transaksi selama tiga minggu. Berikut disajikan 210 data beserta

atribut yang melekat sesuai dengan Tabel 1. Atribut Data, disajikan pada Tabel 6. Data Latih dan Uji.

Tabel 6. Kode Data Latih dan Uji

No	Cust omer	Jenis Hari	Cu aca	Ha ri	Ju al	Re tur	Opt imal
1	C10	Bukan Hari Libur	Cle ar	Mi ng gu	20	4	16
2	C09	Bukan Hari Libur	Cle ar	Mi ng gu	30	3	27
...
210	C01	Bukan Hari Libur	Clo uds	Se nin	50	5	45

4.2. Preprocessing

Pada tahap ini dilakukan preprocessing untuk menyeleksi data sebelum dilakukan klasifikasi dengan tujuan meningkatkan akurasi pada tahap klasifikasi. Pada tahapan dilakukan antara lain data tranformation dan clustering menggunakan K-means untuk mendapatkan data latih yang baik (layak).

4.2.1. Data Transformation

Pada tahapan ini atribut data penjualan dikodekan sehingga menjadi data numerik sesuai dengan persamaan Tabel 2. Kode dan Transformasi Atribut. Berikut merupakan hasil dari pengodean atribut disajikan pada Tabel 7. Kode Data Latih dan Uji.

Tabel 7. Kode Data Latih dan Uji

ID	A1	A2	A3	A4	A5	A6	A7
1	10	2	3	7	20	4	16
2	9	2	3	7	30	3	27
...
210	1	2	4	1	50	5	45

4.2.2. K-Means Clustering

Pada tahapan *clustering* algoritma yang digunakan adalah K-means. Tahapan K-means *clustering* yang digunakan pada penelitian ini antara lain adalah normalisasi data, pembobotan data, inisialisasi pusat awal *cluster*, hitung jarak *euclidean*, dan pengelompokkan.

4.2.2.1. Normalisasi Data

Dilakukan normalisasi data untuk setiap atribut sesuai dengan rasio persamaan sesuai dengan Tabel 4. Rasio Normalisasi Atribut. Pada atribut yang nilai normalisasinya tidak “tetap” akan dihitung nilai normalisasinya. Nilai normalisasi ini akan digunakan untuk mencari bobot suatu *record* data. Setelah setiap rasio atribut telah dihitung maka selanjutnya setiap rasio tersebut diterapkan pada setiap *record* data. Berikut hasil dari normalisasi data dapat dilihat pada Tabel 8. Normalisasi Data Latih dan Uji.

Tabel 8. Normalisasi Data Latih dan Uji

ID	A1	A2	A3	A4	A5	A6	A7
1	15.14	20.79	22.93	25.85	20	4	16
2	15.90	20.79	22.93	25.85	30	3	27

...
210	25.57	20.79	23.16	23.61	50	5	45

4.2.2.2. Pembobotan Data

Pada tahapan ini dilakukan pembobotan untuk mengetahui bobot dari data penjualan berdasarkan usulan persamaan (1). Hasil dari pembobotan data disajikan pada Tabel 9. Bobot Data Latih dan Uji selengkapanya terdapat pada Lampiran Bobot Data Latih dan Uji. Berikut adalah detail dari perhitungan bobot yang dilakukan :

$$\begin{aligned} \text{Bobot}_{(1)} &= \sqrt{15.14^2 + 20.79^2 + 22.93^2 + 25.85^2 + 20^2 + 4^2 + 16^2} \\ &= 50.27 \\ \text{Bobot}_{(2)} &= \sqrt{15.90^2 + 20.79^2 + 22.93^2 + 25.85^2 + 30^2 + 3^2 + 27^2} \\ &= 59.30 \\ &\dots \\ \text{Bobot}_{(210)} &= \sqrt{25.57^2 + 20.79^2 + 23.16^2 + 25.85^2 + 50^2 + 5^2 + 16^2} \\ &= 82.04 \end{aligned}$$

Tabel 9. Bobot Data Latih dan Uji

ID	A1	A2	A3	A4	A5	A6	A7	Bo bot
1	15.	20.	22.	25.	20	4	16	50.
	14	79	93	85				27
2	15.	20.	22.	25.	30	3	27	59.
	90	79	93	85				30
...
210	25.	20.	23.	23.	50	5	45	82.
	57	79	16	61				04

4.2.2.3. Inisialisasi Pusat Awal Cluster

Pada penelitian ini tahapan dari proses pembangkitan pusat awal cluster menggunakan algoritma genetika antara lain inisialisasi populasi awal *roulette wheel selection*, *crossover*, mutasi, elitisme, dan iterasi lanjutan.

4.2.2.3.1. Inisialisasi Populasi Awal

Berdasarkan aturan pembangkitan populasi awal pada Tabel 5. Pembentukan Kromosom maka terbentuk populasi awal. Pada Tabel 10 kromosom-1 gen-1 nilai “125” berasal dari pembangkitan nilai random dari salah satu ID kromosom yang jumlah optimal ≤ 10 sesuai dengan persamaan pada *rule* Tabel 3. *Rule Cluster*. Hasil dari proses pembentukan populasi awal disajikan pada Tabel 10. Populasi Awal dan *fitness* populasi awal pada Tabel 11. *Fitness* Populasi Awal.

Tabel 10. Populasi Awal

No	Krom.	Gen1	Gen2	Gen3	Gen4	Gen5
1	Krom.1	125	192	12	28	169
2	Krom.2	118	41	134	87	205
3	Krom.3	131	174	69	149	179
4	Krom.4	173	49	7	88	210

Kromosom 1 :

$$N = \frac{(82.47 - 41.27)}{(5 - 1)} = 10.29$$

$$C_2 = 41.27 + (10.29 * 1) = 51.57$$

$$C_3 = 41.27 + (10.29 * 2) = 61.87$$

$$C_4 = 41.27 + (10.29 * 3) = 72.17$$

$$= \frac{1}{abs(48.04 - 51.57) + abs(53.85 - 61.87) + abs(71.08 - 72.17) + 0.01}$$

$$f = \frac{1}{12.64} = 0.079$$

...

Kromosom 4 :

$$N = \frac{(82.040 - 41.895)}{(5 - 1)} = 10.036$$

$$= \frac{1}{abs(46.94 - 51.93) + abs(60.61 - 61.96) + abs(72.37 - 72.00) + 0.01}$$

$$f = \frac{1}{6.7223} = 0.148$$

Tabel 11. Fitness Populasi Awal

No	Krom.	Ge n1	Ge n2	Ge n3	Ge n4	Ge n5	fitness
1	Krom.1	41.	48.	53.	71.	82.	0.07
		27	04	85	08	47	
2	Krom.2	44.	42.	56.	71.	79.	0.06
		73	91	91	47	41	
3	Krom.3	40.	47.	60.	70.	78.	0.22
		66	32	19	36	81	
4	Krom.4	41.	46.	60.	72.	82.	0.14
		89	94	61	37	04	

4.2.2.3.2. Roulette Wheel Selection

Pada tahap ini populasi awal akan dilakukan *roulette wheel selection* berdasarkan persamaan fitness (3). Probabilitas kromosom pada Tabel 12. *Fitness* dan Probabilitas dan hasil dari proses *roulette wheel selection* pada Tabel 13. *Roulette Wheel Selection*.

Tabel 12. Fitness dan Probabilitas

No	Krom.	fitness	Probabilitas
1	Krom.1	0.079	0.153
2	Krom.2	0.061	0.118
3	Krom.3	0.226	0.438
4	Krom.4	0.149	0.289

Probabilitas :

$$\text{Jumlah fitness} = 0.079 + 0.061 + 0.226 + 0.149 = 0.514$$

$$\text{Kromosom1 : } 0.079 / 0.514 = 0.153$$

$$\text{Kromosom2 : } 0.061 / 0.514 = 0.118$$

$$\text{Kromosom3 : } 0.226 / 0.514 = 0.438$$

$$\text{Kromosom4 : } 0.149 / 0.514 = 0.289$$

Hitung q :

$$q1 = 0 + 0.153 = 0.153$$

$$q2 = 0.153 + 0.118 = 0.271$$

$$q3 = 0.271 + 0.438 = 0.710$$

$$q4 = 0.710 + 0.289 = 1$$

Bangkitkan random :

$$r1 = 0.163$$

$$r2 = 0.253$$

$$r3 = 0.652$$

$$r4 = 0.459$$

Bandingkan :

$$r1 > q1 \mid 0.163 > 0.153 \quad \text{kromosom1 dan 2 tukar}$$

$$r2 < q2 \mid 0.253 < 0.271 \quad \text{kromosom 2 tetap}$$

$$r3 < q3 \mid 0.652 < 0.710 \quad \text{kromosom 3 tetap}$$

$$r4 < q4 \mid 0.459 < 1 \quad \text{kromosom 4 tetap}$$

Tabel 13. Roulette Wheel Selection

No	Krom.	Gen 1	Gen 2	Gen 3	Gen 4	Gen 5
1	Krom.1	118	41	134	87	205
2	Krom.2	125	192	12	28	169
3	Krom.3	131	174	69	149	179
4	Krom.4	173	49	7	88	210

4.2.2.3.3. Crossover

Pada tahap *crossover* akan dibangkitkan nilai random untuk setiap gen dari kromosom ganjil dan akan dibandingkan dengan *probabilitas crossover*. Apabila nilai random lebih kecil dari *probabilitas crossover* maka gen akan ditukar dengan gen dari kromosom genap dibawahnya dengan urutan gen pada kromosom sama. Hasil dari proses *crossover* dapat dilihat pada Tabel 14. *Crossover*.

Tabel 14. Crossover

No	Krom.	Gen 1	Gen 2	Gen 3	Gen 4	Gen 5
1	Krom.1	125	192	12	28	169
2	Krom.2	118	41	134	87	205
3	Krom.3	173	49	7	88	179
4	Krom.4	131	174	69	149	210

$$\text{Probabilitas Crossover} = 0.9 \quad PC > r = \text{tukar}$$

$$r \text{ kromosom1 gen1} = 0.669 \text{ (tukar)}$$

$$r \text{ kromosom1 gen2} = 0.334 \text{ (tukar)}$$

$$r \text{ kromosom1 gen3} = 0.485 \text{ (tukar)}$$

$$r \text{ kromosom1 gen4} = 0.195 \text{ (tukar)}$$

$$r \text{ kromosom1 gen5} = 0.145 \text{ (tukar)}$$

$$r \text{ kromosom3 gen1} = 0.222 \text{ (tukar)}$$

$$r \text{ kromosom3 gen2} = 0.097 \text{ (tukar)}$$

$$r \text{ kromosom3 gen3} = 0.379 \text{ (tukar)}$$

$$r \text{ kromosom3 gen4} = 0.509 \text{ (tukar)}$$

$$r \text{ kromosom3 gen5} = 0.976 \text{ (tetap)}$$

4.2.2.3.4. Mutasi

Pada tahap mutasi akan dibangkitkan nilai random untuk setiap gen dari kromosom apabila nilai random lebih kecil dari probabilitas mutasi maka gen akan diganti dengan gen baru sesuai dengan aturan Tabel 5. Pembentukan Kromosom. Hasil dari proses mutasi dapat dilihat pada Tabel 15. Mutasi.

Tabel 15. Mutasi

No	Krom.	Gen1	Gen2	Gen3	Gen4	Gen5
1	Krom.1	132	192	139	28	169
2	Krom.2	118	41	134	87	205
3	Krom.3	173	49	7	88	205
4	Krom.4	131	174	69	28	210

$$\text{Probabilitas Mutasi} = 0.1 \mid PM > r = \text{ganti}$$

$$r \text{ kromosom1 gen1} = 0.018 \text{ (ganti)}$$

$$r \text{ kromosom1 gen2} = 0.727 \text{ (tetap)}$$

$$r \text{ kromosom1 gen3} = 0.016 \text{ (ganti)}$$

$$r \text{ kromosom1 gen4} = 0.377 \text{ (tetap)}$$

r kromosom1 gen5 = 0.845 (tetap)
 r kromosom2 gen1 = 0.945 (tetap)
 r kromosom2 gen2 = 0.768 (tetap)
 r kromosom2 gen3 = 0.909 (tetap)
 r kromosom2 gen4 = 0.104 (tetap)
 r kromosom2 gen5 = 0.202 (tetap)
 r kromosom3 gen1 = 0.615 (tetap)
 r kromosom3 gen2 = 0.853 (tetap)
 r kromosom3 gen3 = 0.282 (tetap)
 r kromosom3 gen4 = 0.731 (tetap)
 r kromosom3 gen5 = 0.055 (ganti)
 r kromosom4 gen1 = 0.852 (tetap)
 r kromosom4 gen2 = 0.774 (tetap)
 r kromosom4 gen3 = 0.478 (tetap)
 r kromosom4 gen4 = 0.084 (ganti)
 r kromosom4 gen5 = 0.884 (tetap)

4.2.2.3.5. Seleksi

Pada tahap ini dilakukan seleksi dengan cara elitisme. Jika individu hasil seleksi akan diproses lanjut dengan *crossover* dan mutasi, sedangkan dengan elitisme maka individu dengan kromosom terbaik dijaga. Jika menggunakan elitisme, nilai *fitness* individu terbaik selalu terjaga dan nilainya cenderung naik, jumlah generasi yang dipakai untuk mendapat individu terbaik yang diinginkan juga lebih sedikit. Hasil dari proses elitisme dapat dilihat pada Tabel 16. Elitisme.

Tabel 16. Elitisme.

No	Gen1	Gen2	Gen3	Gen4	Gen5	fitness
1	131	174	69	149	179	0.226
2	131	174	69	28	210	0.183
3	131	174	69	149	210	0.161
4	173	49	7	88	219	0.149
5	173	49	7	88	205	0.149
6	173	49	7	88	179	0.138
7	132	192	139	28	169	0.113
8	125	192	12	28	169	0.079
9	125	192	12	28	169	0.079
10	118	41	134	87	205	0.061
11	118	41	134	87	205	0.061
12	118	41	134	87	205	0.061

4.2.2.3.6. Iterasi Lanjutan

Pada tahap iterasi lanjutan dilakukan pengulangan tahap algoritma genetika dari proses *roulette wheel selection* hingga tahap seleksi elitisme sampai ditemukan solusi terbaik dengan *fitness* mendekati 1 atau telah mencapai generasi maksimal. Pada penelitian ini didapatkan solusi terbaik dengan *fitness* 0.979 dapat dilihat pada Tabel 17. Hasil Algoritma Genetika dan Tabel 18. Hasil Percobaan Algoritma Genetika.

Tabel 17. Hasil Algoritma Genetika

N	Gen	Gen	Gen	Gen	Gen	fitness
o	1	2	3	4	5	s
1	181	128	64	87	70	0.979
2	131	146	9	88	70	0.978
...

Tabel 18. Hasil Percobaan Algoritma Genetika

No	Generasi						fitness
1	18						0.979
2	15						0.978
3	30						0.979
4	13						0.979
5	37						0.978
Gen 1 / cluster_id = 181							
Cluster	A1	A2	A3	A4	A5	A6	A7
C1	10	2	2	3	10	3	7
Gen 2 / cluster_id = 128							
C2	3	2	1	2	20	6	14
Gen 3 / cluster_id = 64							
C3	7	2	4	1	30	4	26
Gen 4 / cluster_id = 87							
C4	4	2	4	6	40	0	40
Gen 5 / cluster_id = 70							
C5	1	2	4	1	50	4	46

4.2.2.4. Hitung Jarak Euclidean

Berdasarkan solusi terbaik yang telah dihitung menggunakan algoritma genetika pada Tabel 17. Hasil Algoritma Genetika maka pusat *cluster* terbaik merupakan terjemahan gen dari kromosom terbaiknya. Hasil dari pusat awal *cluster* terbaik dapat dilihat Tabel 19. Pusat Awal *Cluster* dan Tabel 20. Normalisasi Pusat Awal *Cluster*.

Tabel 19. Pusat Awal *Cluster*

Cluster	A1	A2	A3	A4	A5	A6	A7
C1	10	2	2	3	10	3	7
C2	3	2	1	2	20	6	14
C3	7	2	4	1	30	4	26
C4	4	2	4	6	40	0	40
C5	1	2	4	1	50	4	46

Tabel 20. Normalisasi Pusat Awal *Cluster*

Clus ter	A1	A2	A3	A4	A5	A6	A7
C1	15.14	20.79	18.16	16.56	10	3	7
C2	24.19	20.79	17.42	19.83	20	6	14
C3	23.33	20.79	23.61	23.16	30	4	26
C4	21.33	20.79	23.61	21.53	40	0	40
C5	25.57	20.79	23.61	23.16	50	4	46

Setelah diketahui pusat *cluster* selanjutnya adalah menghitung jarak data uji ke pusat *cluster*. Hasil hitung jarak iterasi I pada disajikan pada Tabel 21. Jarak *Cluster* Iterasi I dan jarak iterasi II disajikan pada Tabel 24. Jarak *Cluster* Iterasi II.

$$\begin{aligned}
 d(1,1) &= \sqrt{(15.14 - 15.14)^2 + (20.79 - 20.79)^2 + (25.85 - 18.16)^2 + (22.93 - 16.56)^2} \\
 &= \sqrt{(20 - 10)^2 + (4 - 3)^2 + (16 - 7)^2} \\
 &= 16.7799 \\
 d(1,2) &= \sqrt{(15.14 - 24.19)^2 + (20.79 - 20.79)^2 + (25.85 - 17.42)^2 + (22.93 - 19.83)^2 + (20 - 20)^2 + (4 - 6)^2 + (16 - 14)^2} \\
 &= 13.0589 \\
 &\dots
 \end{aligned}$$

Tabel 21. Jarak *Cluster* Iterasi I

ID	C1	C2	C3	C4	C5	Hasil
1	16.77	13.05	16.49	32.20	43.74	C2
2	30.00	20.67	7.88	17.73	29.33	C3
3	30.95	18.90	2.70	16.94	27.83	C3
4	19.71	10.64	13.15	29.98	41.20	C2
5	17.47	11.93	14.84	30.96	42.55	C2
...
210	56.83	43.74	27.69	13.06	1.41	C5

Tabel 22. Jumlah Data *Cluster* Iterasi I

No	Cluster	Jumlah
1	C1	20
2	C2	82
3	C3	81
4	C4	22
5	C5	5

Setelah diketahui anggota tiap-tiap *cluster* kemudian pusat *cluster* baru dapat dihitung berdasarkan data anggota tiap-tiap *cluster* baru tersebut didapat dari rumus = nilai hasil / jumlah hasil. Berikut adalah detail dari perhitungan pusat *cluster* baru yang dilakukan.

$$\begin{aligned}\text{Cluster 1 (A1)} &= 355.71 / 20 = 17.78 \\ \text{Cluster 1 (A2)} &= 415.90 / 20 = 20.79 \\ &\dots \\ \text{Cluster 5 (A7)} &= 225 / 5 = 45\end{aligned}$$

Tabel 23. Pusat *Cluster* Iterasi II

Cluster	A1	A2	A3	A4	A5	A6	A7
C1	17.	20.	19.	19.	10.	4.5	6.3
	78	79	61	18	18		
C2	19.	20.	20.	20.	20.	5.5	14.71
	59	79	17	33	30	8	
C3	21.	20.	21.	21.	29.	4.7	25.12
	84	79	14	37	91	9	
C4	23.	20.	22.	21.	40	4.8	35.18
	52	79	39	54		1	
C5	23.	20.	22.	22.	50	5	45
	41	79	97	20			

Tabel 24. Jarak *Cluster* Iterasi II

ID	C1	C2	C3	C4	C5	Hasil
1	15.45	7.94	15.86	29.20	42.65	C2
2	29.25	17.44	8.15	15.55	28.16	C3
...
210	55.91	43.06	28.66	14.30	2.44	C5

Pada data perhitungan jarak iterasi II dilakukan perhitungan yang sama dengan perhitungan jarak terasi I. Hasil dari kelompok *cluster* iterasi I dan terasi II tidak berubah, hal ini menunjukkan bahwa pengelompokkan telah relevan.

4.4. Pengelompokkan

Pada tahap ini setelah proses hitung jarak data dikelompokkan sesuai dengan jarak terdekat dengan pusat *cluster*. Setelah data dikelompokkan selanjutnya adalah menilai apakah *cluster* yang dihasilnya sesuai dengan Tabel 3. *Rule Cluster*. Apabila data masuk kelompok dengan tidak melanggar *rule* maka data tersebut layak dijadikan data latih pada tahap klasifikasi. Berikut adalah hasil dari pengelompokkan dengan *rule* yang diberikan oleh pihak PT. Harum Bakery pada Tabel 25. Hasil *Preprocessing*.

Tabel 25 Hasil *Preprocessing*

No	Customer	Jenis Hari	Cuaca	Hari
1	C10	Bukan Hari Libur	Clear	Minggu
...
14	C07	Bukan Hari Libur	Rain	Sabtu
...
210	C01	Bukan Hari Libur	Clouds	Senin

Tabel 26 Hasil *Preprocessing*

Jual	Retur	Optimal	Cluster	Layak
20	4	16	C2	Ya
...
20	10	10	C2	Tidak
...
50	5	45	C5	Ya

4.5. Pengujian dan Evaluasi

Berdasarkan proses *preprocessing* dari 210 data ternyata hanya 200 data yang layak untuk menjadi data uji pada tahap klasifikasi. Berdasarkan 200 data sebagai data uji yang terbagi menjadi lima kelas, yaitu A(19), B(75), C(79), D(22), dan E(5). Pengujian yang dilakukan menggunakan *confusion matrix* dengan hasil pengujian disajikan pada Tabel 27. Pengujian *Confusion Matrix*.

Tabel 27. Pengujian *Confusion Matrix*

	True e A	True B	True e C	True e D	True e E	Precision
Pred.A	6	6	6	0	0	33.33 %
Pred.B	13	52	11	2	0	66.66 %
Pred.C	0	9	51	13	1	68.91 %
Pred.D	0	6	8	6	2	27.27 %
Pred.E	0	2	3	1	2	25.00 %
Recall	31.5 7 %	69.3 3 %	64.5 5 %	27.2 7 %	40.0 0 %	

Akurasi model klasifikasi yang terbentuk dapat dihitung berdasarkan persamaan (5). Hasil dari pengujian dengan variasi *k* pada tiap *fold* disajikan pada Tabel 28. Pengujian *Crossvalidation*.

Tabel 28. Pengujian *Crossvalidation*

k	fold 1	fold 2	fold 3	fold 4	fold 5	Total	Accuracy
1	25. 00	42. 50	35. 00	37. 50	40. 00	180	36 %
2	60. 00	67. 50	57. 50	60. 00	65. 00	310	62 %
3	55. 00	40. 00	50. 00	55. 00	47. 50	247.5	49.5 %
4	47. 50	42. 50	42. 50	52. 50	55. 00	240	48 %
5	55. 00	45. 00	50. 00	62. 50	52. 50	265	53 %
6	70. 00	42. 50	42. 50	62. 50	57. 50	275	55 %
Rata-rata						50.58%	

Berdasarkan pengujian menggunakan *confusion matrix* dapat disimpulkan bahwa akurasi adalah 58.5 %. Berikut ini merupakan hasil perbandingan klasifikasi KNN dan klasifikasi KNN dengan *preprocessing k-means* yang berbeda pada Tabel 29. Hasil Perbandingan Pengujian.

Tabel 29. Hasil Perbandingan Pengujian

<i>k</i>	KNN	KNN <i>Preprocessing</i>
1	33.33 %	34.5 %
2	54.76 %	58.5 %
3	46.19 %	48.00 %
4	50.47 %	46.00 %
5	45.23 %	45.5 %
6	48.09 %	53.5 %
Rata-rata	46.34 %	47.67 %

5. KESIMPULAN

Berdasarkan pengujian menggunakan variasi *k* yang berbeda pada Tabel 28. Hasil Perbandingan Pengujian, maka didapatkan hasil rata-rata akurasi klasifikasi menggunakan *preprocessing* mengalami peningkatan dibandingkan tanpa *preprocessing*. Berdasarkan hasil penelitian yang telah dilakukan maka dapat disimpulkan sebagai berikut :

1. Algoritma genetika mampu mengatasi kelemahan algoritma *k-means clustering* yang diakibatkan oleh penentuan pusat awal *cluster*-nya berupa solusi yang sifatnya *local optimal*.
2. Kinerja algoritma klasifikasi *k-nearest neighbors* yang menggunakan *preprocessing k-means clustering* cukup baik karena hasil pengujian berhasil mendapatkan akurasi terbaik 58.50 % dan menggunakan *crossvalidation* berhasil mendapatkan rata-rata akurasi 50.58% lebih besar 2.51 % dari KNN tanpa *preprocessing*.
3. Akurasi yang didapat masih belum maksimal. Hal ini sangat dipengaruhi oleh jumlah data yang masih kurang dan memiliki jumlah kelas yang kurang merata, sehingga perlu dilakukan penambahan jumlah data dan pemerataan kelas untuk mendapatkan akurasi yang lebih baik.

6. DAFTAR PUSTAKA

- UYUN, S., & HARTATI, S. 2011. Penentuan Komposisi Bahan Pangan Untuk Diet Penyakit Ginjal dan Saluran Kemih dengan Algoritma Genetika. *Seminar Nasional*. Retrieved from <http://journal.uui.ac.id/index.php/Snati/article/view/2196>
- ARANDA, J., & NATASYA, W. A. G. 2016. Penerapan Metode K-Means Cluster Analysis Pada Sistem Pendukung Keputusan Pemilihan Konsentrasi Untuk Mahasiswa International Class Stmik Amikom Yogyakarta. *Semnasteknomedia Online*, 4(1), 4-2-1.
- ASRONI, A., FITRI, H., & PRASETYO, E. 2018. Penerapan Metode Clustering dengan Algoritma K-Means pada Pengelompokan Data Calon Mahasiswa Baru di Universitas Muhammadiyah Yogyakarta (Studi Kasus: Fakultas Kedokteran dan Ilmu Kesehatan, dan Fakultas Ilmu Sosial dan Ilmu Politik). *Semesta Teknika*, 21(1), 60-64. <https://doi.org/10.18196/st.211211>
- BADRUL, M. 2016. Optimasi Neural Network Dengan Algoritma Genetika Untuk Prediksi Hasil Pemilu. *Bina Insani ICT Journal*, 3421, 229-242.
- CANTÚ-PAZ, E. 2003. Preface. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2724(May 2014), VII-IX. <https://doi.org/10.1007/3-540-45110-2>
- GEN, M., & CHENG, R. 1999. Genetic Algorithms and Engineering Optimization. *Genetic Algorithms and Engineering Optimization*. <https://doi.org/10.1002/9780470172261>
- JAVA, C., WISDAYANI, D. S., NUR, I. M., WASONO, R., & SEMARANG, U. M. (n.d.). Penerapan Algoritma K-Nearest Neighbor dalam Klasifikasi Tingkat Keparahan Korban Kecelakaan Lalu Lintas di Kabupaten Jawa Tengah. 373-380.
- MU'ASYAROH, F. L., & MAHMUDY, W. F. 2016. Implementasi Algoritma Genetika Dalam Optimasi Model AHP dan Topsis Untuk Penentuan Kelayakan Pengisian Bibit Ayam Broiler di Kandang Peternak. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 3(4), 226. <https://doi.org/10.25126/jtiik.201634206>
- PRATAMA, R. A., DJAMAL, E. C., & KOMARUDIN, A. 2017. Optimalisasi Pengantaran Barang dalam Perdagangan Online Menggunakan Algoritma Genetika. *Seminar Nasional Aplikasi Teknologi Informasi (SNATi)*, pp. 11-15.
- RANDY, L. H., & SUE, E. H. 2004. Practical Genetic Algorithms. *New York: Wiley & Sons, Inc*, 50, 62.
- RUSTIYAN, R., & MUSTAKIM, M. 2018. Penerapan Algoritma Fuzzy C Means untuk Analisis Permasalahan Simpanan Wajib Anggota Koperasi. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 5(2), 171. <https://doi.org/10.25126/jtiik.201852605>
- SOKOLOVA, M., JAPKOWICZ, N., & SZPAKOWICZ, S. 2006. Beyond accuracy, F-score and ROC: A family of discriminant measures for performance evaluation. *Australasian Joint Conference on Artificial Intelligence*, WS-06-06(c), 24-29.
- WICAKSONO, S. A. 2019. Optimasi Sistem Penempatan Magang Menerapkan Algoritma Genetika. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 6(1), 17. <https://doi.org/10.25126/jtiik.201961950>
- WIDIARINA. 2013. Algoritma Cluster Dinamik Untuk Optimasi Cluster Pada Algoritma K-Means Dalam Pemetaan Nasabah Potensial Algoritma Cluster Dinamik Untuk Optimasi Cluster Pada Algoritma K-Means Dalam. *Tesis Magister Ilmu Komputer, Nusa Mandiri*, 1(1), 33-36.
- WON YOON, J., & FRIEL, N. 2015. Efficient model selection for probabilistic K nearest neighbour classification. *Neurocomputing*, 149(PB), 1098-1108.

- <https://doi.org/10.1016/j.neucom.2014.07.023>
XU, S. 2010. An expectation-maximization algorithm for the Lasso estimation of quantitative trait locus effects. *Heredity*, 105(5), 483–494.
<https://doi.org/10.1038/hdy.2009.180>