Evaluation Model of Students Learning Outcome Using K-Means Algorithm

Muhammad Iqbal Wijonarko Lado Rayhan Najib

1806186736 1806141265
Fakultas Ilmu Komputer Fakultas Ilmu Komputer
Universitas Indonesia Universitas Indonesia
muhammad.iqbal815@ui.ac.id lado.rayhan@ui.ac.id

June 24, 2021

Abstrak - K-means merupakan salah satu metode yang digunakan untuk melakukan proses clustering data. Metode ini cukup populer untuk digunakan karena cukup sederhana. Namun di sisi lain, k-means mempunyai permasalahan seperti performa hasil clustering sangat ditentukan oleh pemilihan jumlah cluster serta metode inisialiasi yang digunakan. Pada penelitian ini, kami menguji sampel data nilai akhir dari 134 siswa pada suatu sekolah. Data tersebut dianalisis dengan membagi menjadi 3 cluster. Hasil analisanya adalah terdapat 40.3% siswa yang memiliki predikat nilai memuaskan, 25.4% siswa yang memiliki predikat baik, dan 34.3% siswa yang memiliki predikat nilai buruk. Dari hasil ini, dapat dibandingkan nilainya dengan hasil pada paper [1] yang menggunakan algoritma built-in python dan mencari mana yang terbaik untuk dijadikan acuan untuk mendapatkan keputusan terbaik dalam meningkatkan hasil pembelajaran siswa yang lebih baik.

Kata Kunci: Clustering, Algoritma K-Means, Data Mining

1 Pendahuluan

1.1 Latar Belakang

Semakin meningkatnya pertumbuhan jumlah data yang tersedia di berbagai bidang ilmu, membuatnya semakin sulit untuk dimanipulasi dan dianalisis menjadi informasi penting. K-Means merupakan algoritma yang paling populer mengenai *clustering method* untuk kumpulan data yang besar. [2]. K-means *Clustering* adalah salah satu *"unsupervised machine learning algorithms"*. *Unsupervised learning* adalah data yang tidak memiliki label secara eksplisit

dan mampu belajar dari data dengan pola yang implisit.*Unsupervised learning* merupakan jenis *learning* yang hanya mempunyai variabel input tapi tidak mempunyai variabel output yang berhubungan.

Dari pernyataan sebelumnya didapatkan bahwa K-Means *Clustering* adalah suatu metode penganalisaan data atau metode *Data Mining* yang melakukan proses pemodelan tanpa supervisi *(unsupervised)* dan merupakan salah satu metode yang melakukan pengelompokan data dengan sistem partisi. Pengelompokkan ini dapat menggunakan beberapa parameter sebagai acuan pengelompokkan. Proses pengelompokkannya dengan melihat kemiripan pengukuran jarak (*distance measure*) suatu titik data dengan titik lainnya. Suatu *clustering* dapat dikatakan baik bilamana memiliki *high intra-class similarity* dan *low inter-class similarity*.

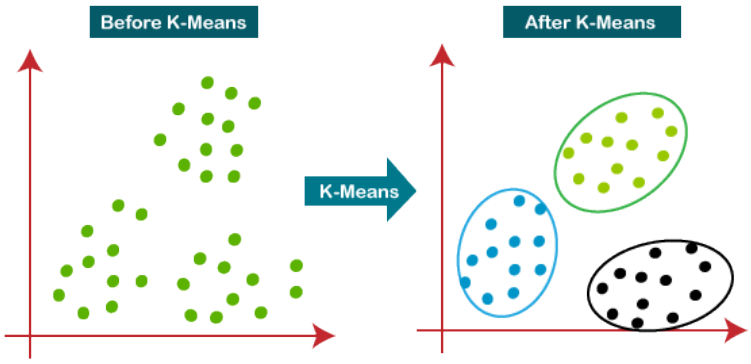

Figure 1: Sumber : https://static.javatpoint.com/tutorial/machine-learning

1.2 Tujuan Penelitian

Clustering merupakan sebuah solusi dari masalah *unsupervised learning*, dimana untuk setiap masalah yang terkait terdapat kumpulan data yang tidak diketahui. Metode *K-Means Clustering* memiliki tujuan untuk mengklasifikasikan data, dengan cara menentukan pengelompokan dalam satu set data yang tidak diketahui. Tujuan dari penulisan ini adalah untuk memberikan contoh penerapan algoritma k-Means untuk suatu set data. Untuk melakukan penerapan algoritma ini akan dibutuhkan sebuah eksperimen pada suatu set data kemudian setelah melihat hasil setelah digunakannya algoritma k-Means ini.

2 Tinjauan Pustaka

K-Means Clustering Algorithm digunakan untuk kumpulan data yang besar. Proses yang dilakukan untuk menggali data yang besar tersebut dan menghasilkan informasi atau pengetahuan baru dapat dilakukan dengan teknologi basis data yang dikenal dengan *Data Mining*. *Data Mining* diterapkan untuk mengumpulkan dan menghasikan suatu informasi baru yang selama ini tidak diketahui secara manual yang didapat dari sekumpulan data yang besar.

Teknik yang dapat dilakukan untuk melakukan *data mining* adalah *clustering*. *Clustering* terbagi menjadi 2 yaitu *hierarchical clustering* dan *non-hierarchical clustering*. *K-*

Means Clustering adalah salah satu metode pada *non-hierarchical clustering* dengan melakukan pembagian data kedalam suatu kelompok dimana data pada satu kelompok memiliki karakteristik yang sama dan berbeda dengan kelompok lain. Pengelompokkan ini akan membantu dalam proses pengambilan keputusan atau saat menentukan kesimpulan dari suatu masalah yang membutuhkan interpretasi data yang besar.

2.1 Tahapan Sebelum Implementasi Algoritma k-Means

Menurut Agus Nur Khomarudin, terdapat beberapa tahapan sebelum mengimplementasi *K-Means Algorithm*. Tahapan pertama adalah KDD atau *Knowledge Discovery in Database*. KDD merupakan sebuah proses dalam menggali dan menganalisis data yang berukuran besar hingga menemukan *knowledge* baru. *Data Mining* merupakan inti dari proses ini. Berikut merupakan proses dari KDD:

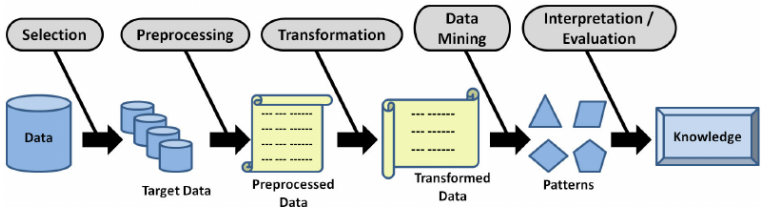

Figure 2: Sumber : https://www.researchgate.net

Pada KDD, alur proses terdapat 5 langkah yaitu[1, 3]:

1 *Data Selection and Collection*

2 *Cleaning and Preparing Data / PreProcessing*

3 *Transformation*

4 *Data Mining*

5 *Interpretation/Evaluation*

Secara general *approach* ini merupakan alur proses yang sudah dijelaskan pada bagian KDD atau pada *Figure 2*. Data yang sangat besar diseleksi hingga menemukan data yang cukup relevan dengan tujuan analisis data. Hasil seleksi data tersebut dinamakan *target data*. Setelah itu masuk ke tahap *preprocessing* dimana *target data* diubah menjadi *preprocessed data* yang selanjutnya akan ditransformasikan menjadi bentuk dengan format standar yang sesuai. Setelah itu *Data Mining* berperan dalam mengekstrak *transformed data* menggunakan metode cerdas. *Pattern* yang didapat dari proses *Data Mining* akan dievaluasi dengan mengidentifikasi pola yang menarik dan merepresentasikan pengetahuan yang menghasilkan *knowledge*.

Tahapan selanjutnya adalah *Data Mining*. *Data Mining* adalah proses meng-ekstrak atau menggali knowledge yang ada pada sekumpulan data (Agus Nur Khomarudin). Data yang digali berasal dari sekumpulan data besar dari berbagai banyak bidang. *Data Mining* adalah kumpulan teknik untuk penemuan otomatis yang efisien dari pola yang sebelumnya

tidak diketahui, valid, baru, berguna, dan dapat dipahami dalam database besar (G.K. Gupta). *Data Mining* adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan machine learning untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari database yang besar (Tacbir Hendro P.). Penggunaan *Data Mining* dapat dilakukan untuk data kecil namun akan lebih baik digunakan pada data yang besar jika ingin menemukan sesuatu yang menarik dan baru.

Selain KDD, terdapat pendekatan proses dalam melakukan *Data Mining*, yaitu *CRISP-DM Approach*. Untuk alur proses dengan menggunakan *CRISP-DM Approach* terdapat beberapa perbedaan. Pendekatan proses dengan *CRISP-DM* terlihat pada gambar di *Figure 3*. alur

Figure 3: Sumber : commons.wikimedia.org

proses *CRISP-DM Approach* terdapat 6 langkah yaitu:

1 *Business Understanding* : Pemahaman tentang substansi bisnis yang akan dilakukan atau tujuan bisnis

2 *Data Understanding* : Fase mengumpulkan data awal, mempelajari data untuk mendapat *insight* mengenai data tersebut.

3 *Data Preparation* : Memilih *table* dan *field* yang akan ditransformasikan ke dalam *database* baru untuk bahan *data mining*

4 *Modeling* : Fase menentukan teknik *data mining* yang digunakan

5 *Evaluation* : Fase interpretasi terhadap hasil *data mining* yang dilakukan sesuai dengan langkah sebelumnya

6 *Deployment* : Fase penyusunan laporan dari pengetahuan yang didapat dari evaluasi pada proses data mining

Perbedaan antara keduanya terdapat pada tahap 1 dan 6 pada *CRISP-DM* sedangkan langkah lainnya serupa dengan KDD dimana tahap *Data Understanding* merupakan gabungan dari

tahap *Selection* dan *Pre Processing* di KDD. Tahap *Data Preparation* merupakan tahap *Transformation*. Tahap *Modeling* merupakan tahap *Data Mining* dan yang terakhir tahap *Evaluation* merupakan tahap *Interpretation/Evaluation*. Tahap *Business Understanding* merupakan pemahaman mengenai bisnis dan scope bisnis tersebut hingga tujuan bisnis yang sesuai dengan harapan pengguna. Lalu tahap *Deployment* merupakan langkah penyusunan hasil pengetahuan yang didapat dari pemodelan. Kedua langkah ini tidak dijelaskan pada KDD.

2.2 *Data Mining*

Pada KDD, terdapat satu proses yang merupakan bagian implementasi dari *K-Means Algorithm* yaitu adalah *Data Mining*. Data Mining merupakan langkah yang paling penting pada algoritma yang akan diimplementasikan dalam proses untuk mengekstrak pola atau aturan pada sekumpulan data yang memiliki manfaat untuk ditafsirkan menjadi informasi yang baru dan berguna. Informasi tersebut bisa dimanfaatkan pada banyak bidang. *Data Mining* menggunakan pendekatan *discovery-based* dalam melakukan pencocokan pola (*pattern-matching* dan algoritma-algoritma lainnya dimanfaatkan dalama menentukan relasi-relasi kunci yang terdapat di kumpulan data yang dieksplorasi. *Data Mining* memiliki akar yang panjang dari bidang ilmu seperti kecerdasan buatan (*artificial intelligent*, *machine learning*, statistik dan basis data. *Data Mining* terbagi menjadi dua kategori yaitu: prediksi dan deskripsi (Ahmed). Teknik prediksi menggunakan data historis dalam menyimpulkan sesuatu kejadian di masa depan. Teknik deskripsi memiliki tujuan untuk menemukan pola dalam data yang menyediakan beberapa informasi tentang hubungan interval yang tersembunyi. Pada teknik deskripsi terdapat

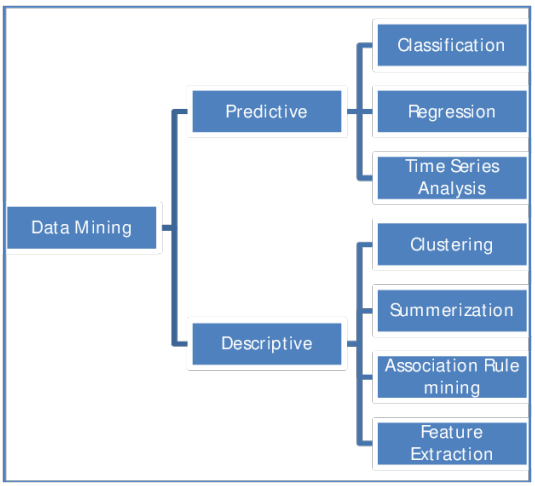

Figure 4: Sumber : researchgate.net/figure/Data-Mining-Techniques

metode yang bernama *clustering*.

Clustering merupakan proses mengatur objek menjadi anggota kelompok yang hampir sama dalam beberapa cara (Nur Wakhidah). Sebuah *cluster* adalah kumpulan dari objek-objek yang ”mirip” diantara mereka dan ”berbeda” dengan objek dari *cluster* lainnya. *Clustering algorithms* mengelompokkan *data points* ke dalam *cluster* menggunakan beberapa gagasan tentang ’*similarity*’ yang bisa digambarkan dengan sesederhana jarak *Euclidean* [4].

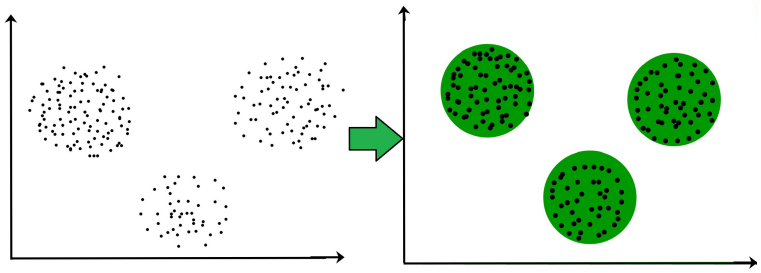

Figure 5: Sumber : https://www.geeksforgeeks.org/clustering-in-machine-learning/

Dari gambar diatas, kita bisa melakukan identifikasi dari 3 kelompok menjadi data yang bisa dibagi dengan kesamaan kriteria jarak dari dua atau lebih objek di dalam *cluster* yang sama jika mereka dekat dan sesuai dengan jarak yang diberikan. Hal ini disebut dengan *distance-based clustering*. *Clustering* memiliki tujuan untuk melakukan pengklasifikasikan data dengan menggunakan pengelompokan di dalam satu set data yang tidak diketahui. *Clustering* algoritma bisa diklasifikan menjadi empat kelompok yaitu: *Exclusive Clustering*, *Overlapping Clustering*, *Hierarchical Clustering*, dan *Probabilistic Clustering*. *K-means* merupakan bagian dari *Exclusive Clustering* karena data dikelompokkan ke dalam suatu cara yang eksklusif, dimana jika suatu fakta kepemilikan suatu *cluster* tidak bisa digunakan (menjadi anggota) di *cluster* lain.

3 Permasalahan

Pada masalah *mixed data clustering* yang akan dilakukan analisanya dengan menggunakan algoritma k-means. *Datasets* yang akan dilakukan analisa merupakan data nilai akhir siswa di suatu kelas dengan jumlah frekuensinya.

Input:

- Himpunan *integer* yang merupakan nilai dari siswa dalam range 1-100 yang didapatkan dari randomize seeder.
- Jumlah *cluster* yang diinginkan.

Output:

- Persentase jumlah anggota dari masing - masing *cluster*

4 Metodologi

Pada paper Sunarya [1], penulis melakukan implementasi pencarian *cluster* dari *datasets* yang berupa nilai hasil belajar 50 siswa dan data tersebut di proses dengan menggunakan algoritma K-means *built-in* dari python. Pada paper ini akan dilakukan langkah - langkah yang

sama dengan [1], tetapi menggunakan algoritma dan *datasets* yang berbeda. Kami menggunakan *datasets* yang berbeda dikarenakan pada [1] tidak diberikan sample data yang akan dilakukan ujicoba. Algoritma yang pada analisis ini dijalankan tanpa pemanggilan *built-in function K-Means*.

4.1 Penjelasan algoritma K-Means

K-Means adalah algoritma untuk mempartisi objek yang ada dalam kedalam satu atau lebih *cluster* atau kelompok objek berdasarkan karakteristiknya, dimana objek yang memiliki karakteristik yang sama akan dikelompokan ke dalam satu *cluster* yang sama dan objek yang mempunyai karakteristik yang berbeda akan dikelompokan kedalam *cluster* yang lain [3]. Langkah-langkah dalam algoritma k-Means *Clustering* adalah [3, 5, 6]:

1 Terima jumlah *cluster* untuk mengelompokkan data dan dataset untuk *cluster* sebagai nilai input.

2 Inisialisasi pusat centroid dengan acak untuk awal program menggunakan titik data.

3 Melakukan penghitungan jarak antara dua data dan cari kemiripan suatu data untuk menentukan *cluster*-nya dengan mencari jarak terdekat tiap data tersebut dengan pusat *cluster*

4 Melakukan penghitungan ulang antara pusat *cluster* dengan anggota clusternya. Pusat cluster merupakan rata - rata dari semua data yang merupakan anggota dari cluster tersebut.

5 Ulangi langkah mencari anggota *cluster* pada langkah sebelumnya sampai tidak ada perubahan lagi.

Rumus yang digunakan dalam mencari pusat *cluster* / *cetroid* adalah berikut ini:

$$v_{ij} = \frac{\sum_{k=1}^{N_i} X_{kj}}{N_i}$$

4.2 *Code* K-means

```
import matplotlib.pyplot as plt
import numpy as np

np.random.seed(0)

def euclidean_distance(x1, x2):
    return np.sqrt(np.sum((x1 - x2) ** 2))
```

```
class KMeans:
    def __init__(self, K=5, max_iters=100, plot_steps=False):
        self.K = K
        self.max_iters = max_iters
        self.plot_steps = plot_steps
        self.iter = 0
        self.clusters = [[] for _ in range(self.K)]
        self.centroids = []

    def predict(self, X):
        self.X = X
        self.n_samples, self.n_features = X.shape

        random_sample_idxs = np.random.choice(self.n_samples, self.K,
            replace=False)
        self.centroids = [self.X[idx] for idx in random_sample_idxs]

        for _ in range(self.max_iters):
            self.iter += 1
            self.clusters = self._create_clusters(self.centroids)

            if self.plot_steps:
                self.plot()

            centroids_old = self.centroids
            self.centroids = self._get_centroids(self.clusters)
            if self._is_converged(centroids_old, self.centroids):
                break

            if self.plot_steps:
                self.plot()

        return self._get_cluster_labels(self.clusters)

    def _get_cluster_labels(self, clusters):
        labels = np.empty(self.n_samples)

        for cluster_idx, cluster in enumerate(clusters):
            for sample_index in cluster:
                labels[sample_index] = cluster_idx
        return labels

    def _create_clusters(self, centroids):
        clusters = [[] for _ in range(self.K)]
        for idx, sample in enumerate(self.X):
```

```
            centroid_idx = self._closest_centroid(sample, centroids)
            clusters[centroid_idx].append(idx)
        return clusters

    def _closest_centroid(self, sample, centroids):
        distances = [euclidean_distance(sample, point) for point in
            centroids]
        closest_index = np.argmin(distances)
        return closest_index

    def _get_centroids(self, clusters):
        centroids = np.zeros((self.K, self.n_features))
        for cluster_idx, cluster in enumerate(clusters):
            cluster_mean = np.mean(self.X[cluster], axis=0)
            centroids[cluster_idx] = cluster_mean
        return centroids

    def _is_converged(self, centroids_old, centroids):
        distances = [
            euclidean_distance(centroids_old[i], centroids[i]) for i
                in range(self.K)
        ]
        return sum(distances) == 0

    def plot(self):
        fig, ax = plt.subplots(figsize=(5, 5))

        for i, index in enumerate(self.clusters):
            point = self.X[index].T
            ax.scatter(*point)

        for point in self.centroids:
            ax.scatter(*point, marker="x", color="black", linewidth=2)

        plt.show()
```

Listing 1: Sumber: github.com/python-engineer/

5 Analisis

Pada bagian ini dijelaskan hasil analisis pada Algoritma K-means. Analisis dilakukan berdasarkan *code* yang telah tersedia pada *section* 3.2 .

5.1 Analisis kompleksitas

Time Complexity yang terdapat dari algoritma tersebut dapat ditinjau dari beberapa parameter. Kita dapat menentukan variable **L** sebagai waktu untuk menghitung jarak antar dua objek pada *euclidean function*. Variable **K** menandakan jumlah cluster yang digunakan atau centroid. Terakhir **n** yang merupakan banyaknya objek. Terdapat variable lain yang dapat menentukan *time complexity*. Variable **m** yang merepresentasikan jumlah dimensi vektor, namun dalam analisa masalah ini kami menggunakan 2 dimensi vektor (x,y) sehingga dapat dihiraukan. Terakhir variable **I** yang menunjukan iterasi saat melakukan *clustering*. Dari uraian mengenai variable tersebut, kami dapat menentukan *time complexity* dari algoritma tersebut yaitu $O(KnL)$. Jika iterasi yang digunakan besar dan sangat mempengaruhi waktu eksekusi maka *time complexity* algoritma adalah $O(IKnL)$. Terlebih jika terdapat m-dimensi yang besar dan centroid yang tidak dalam sparse maka *time complexity* algoritma tersebut menjadi $O(IKnLm)$.

6 Eksperimen

6.1 *Dataset*

Pada eksperimen disini, kami menggunakan *dataset* yang berupa *integer* yang terdiri dari nilai akhir siswa beserta jumlah frekuensi nilainya. Data pada Table 1 ini akan menjadi input pada algoritma K-Means dan nantinya akan dilakukan *plotting point* dengan Frekuensi Nilai sebagai x dan Range Nilai sebagai y. Kemudian dari data ini akan dibagi menjadi 3 cluster yang terdiri dari kategori memuaskan, baik, dan buruk.

No.	Frekuensi Nilai	Range nilai
1	3	100
2	7	95
3	10	90
4	23	85
5	3	80
6	8	75
7	9	70
8	3	65
9	4	60

10	5	55
11	6	50
12	3	45
13	4	40
14	10	35
15	3	30
16	3	25
17	5	20
18	8	15
19	3	10
20	5	5
21	9	0

Table 1: Daftar nilai siswa

6.2 Hasil eksperimen

Clustering data tersebut diperlukan 6 iterasi hingga centroid baru sudah mengalami konvergensi. Figure 6,7,8 menjelaskan bagaimana perubahan data pada cluster dan juga perubahan centroid yang ditandai dengan **x**. Pada gambar pertama menunjukan nilai centroid terdapat pada titik data yang menunjukkan pemilihan centroid pertama yaitu pemilihan secara random dari titik-titik data. Selanjutnya centroid akan dihitung berdasarkan mean dari tiap sumbu x dan sumbu y.

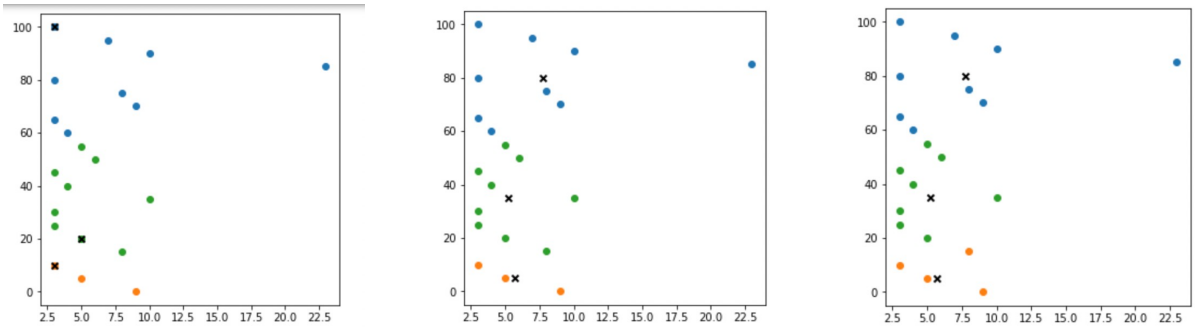

Figure 6: Perubahan *cluster* dan *centroid* 1-3

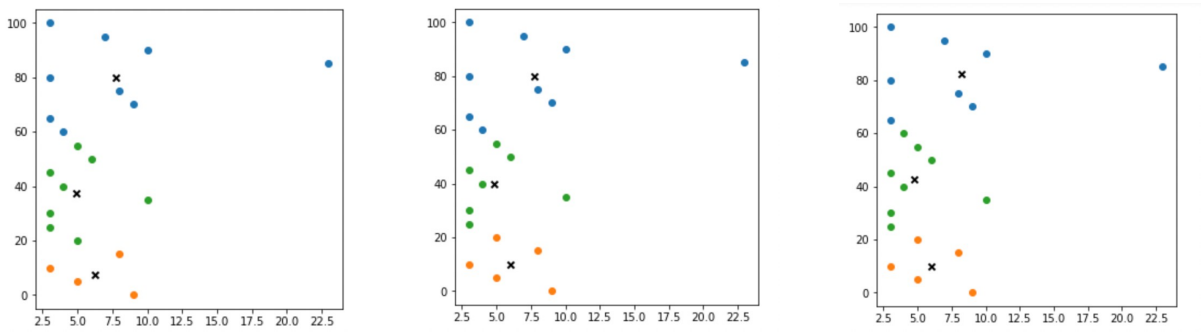

Figure 7: Perubahan *cluster* dan *centroid* 4-6

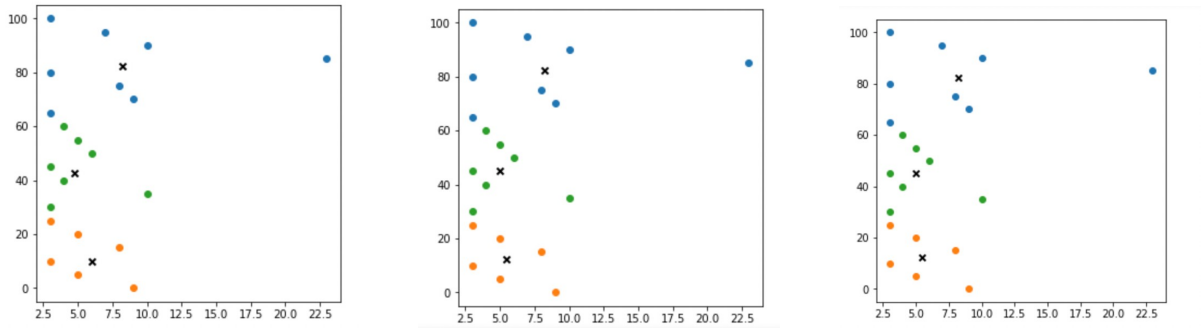

Figure 8: Perubahan *cluster* dan *centroid* 7-9

Table 2 dibawah ini menunjukkan hasil *clustering* data dengan membaginya menjadi 3 clustering. Kemudian dari algoritma tersebut didapatkan 3 *cluster* dengan *range* nilai yang berbeda dan juga frekuensi data yang terdapat pada *cluster* tersebut. *Cluster* 1 terdiri dari rentang nilai 70 hingga 100 dengan 6 data didalamnya. Kemudian *Cluster* 2 terdiri dari rentang nilai 40 - 65 dengan 7 data. Terakhir *Cluster* 3 terdiri dari rentang nilai 0 - 35 dengan 8 data.

No.	**Cluster**	**Frekuensi Data**	**Range nilai**
1	Memuaskan	6	70 - 100
2	Baik	7	40 - 65
3	Buruk	8	0 - 35

Table 2: Daftar nilai siswa

7 Penutup

7.1 Kesimpulan

Algoritma K-Means *Clustering* dapat membantu dalam permasalahan untuk mengelompokkan suatu data pada kelompok yang sesuai. Dalam eksperimen yang kami lakukan yaitu dengan menerapkan algoritma K-Means pada dataset yang kami gunakan didapatkan 3 *cluster* dengan range nilai yang berbeda dan frekuensi data yang berbeda pula. Data-data ini kemudian dapat direpresentasikan pada kelompok *cluster* tersebut. Dengan ini akan dengan mudah menyatakan bahwa untuk siswa yang memiliki nilai 70 - 100 dapat dikategorikan sebagai siswa yang mendapatkan nilai memuaskan. Kemudian untuk siswa yang memiliki nilai 40 - 65 dikategorikan sebagai siswa yang mendapatkan nilai yang baik. Terakhir siswa yang memiliki nilai 0 - 35 dikategorikan sebagai siswa dengan nilai yang buruk. Jumlah mahasiswa yang memiliki predikat nilai memuaskan yaitu sebesar 40.3%.Lalu jumlah mahasiswa dengan predikat nilai baik sebesar 25.4% dan jumlah mahasiswa dengan predikat nilai buruk sebesar 34.3%.

8 Tabel Kontribusi Kerja

Nama Anggota	**Persentase Pekerjaan**
Muhammad Iqbal Wijonarko	50%
Lado Rayhan Najib	50%

References

[1] A. Sunarya, S. L. Nurmika, N. Asmainah *et al.*, "Evaluation model of students learning outcome using k-means algorithm," in *Journal of Physics: Conference Series*, vol. 1477, no. 2. IOP Publishing, 2020, p. 022027.

[2] M. Capó, A. Pérez, and J. A. Lozano, "An efficient approximation to the k-means clustering for massive data," *Knowledge-Based Systems*, vol. 117, pp. 56–69, 2017.

[3] A. N. Khomarudin, "Teknik data mining: Algoritma k-means clustering," *Ilmu Komputer*, 2016.

[4] A. Ahmad and S. S. Khan, "Survey of state-of-the-art mixed data clustering algorithms," *Ieee Access*, vol. 7, pp. 31 883–31 902, 2019.

[5] N. Wakhidah, "Clustering menggunakan k-means algorithm," *Jurnal Transformatika*, vol. 8, no. 1, pp. 33–39, 2010.

[6] J. Oyelade, O. Oladipupo, and I. Obagbuwa, "Application of k means clustering algorithm for prediction of students academic performance," *International Journal of Computer Science and Information Security*, vol. 7, 02 2010.