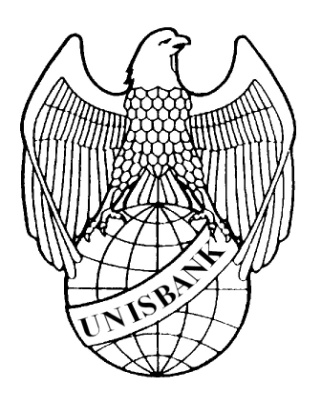
LAPORAN PROJECT MACHINE LEARNING

CLUSTERING DRAMA ASIA MENGGUNAKAN METODE K-MEANS CLUSTERING



DISUSUN OLEH:

AHMAD FUAD KAHFI 17.01.53.0122

THONY KHIKMATULLAH 17.01.53.0136

MIFTAH KHUL JANAH 17.01.53.0137

MOCH GALUH RAGA PRAMESWARA 17.01.53.0200

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS STIKUBANK SEMARANG**

**2020**

1. **Deskripsi Masalah**

Drama menggambarkan realitas kehidupan, karakter dan perilaku manusia melalui partisipasi dan dialog yang dipentaskan. Drama asia merupakan sebuah drama yang berasal dari asia, ada banyak drama yang berasal dari asia contoh nya ada drama cina, jepang dan korea. Drama asia banyak disukai banyak orang dari kalangan remaja sampai dewasa.

Namun dibutuhkan pengelompokan drama asia dengan metode yang tepat yaitu dengan metode K-Means Clustering, agar drama asia tersebut dapat tersusun sesuai variabel yang ditentukan.

1. **Tujuan**

Membuat Machine Learning yang dapat pengelompokan drama asia yang sudah ada dengan metode K-Means Clustering, agar drama asia tersebut dapat tersusun sesuai variabel yang ditentukan.

1. **Dataset**

Sumber data: Doramaku.org

Data diambil dari website drama asia yang lengkap kemudian dipilih 200 data untuk dipakai dan dari website tersebut diambil 4 variabel untuk proses clustering dapat dilihat pada tabel dibawah ini:

|  |  |  |
| --- | --- | --- |
| **No** | **Nama variable** | **Type** |
| 1 | Judul | string |
| 2 | Episode | int |
| 3 | Rating | bolean |
| 4 | Views | int |

Sampel data:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Judul** | **Episode** | **Rating** | **Views** |
|  | A beautiful mind 2016 | 18 | 8.4 | 177 |
|  | A Gentleman's Dignity | 20 | 8.3 | 156 |
|  | A Korean Odyssey | 20 | 8.3 | 136 |
|  | A Love To Kill | 16 | 7.4 | 133 |
|  | A Piece of Your Mind | 12 | 8.2 | 146 |
|  | A Pledge to God | 48 | 7.8 | 236 |
|  | A Poem a Day | 16 | 8.0 | 142 |
|  | A-Teen 2 | 20 | 8.2 | 125 |
|  | A-Teen | 24 | 8.3 | 203 |
|  | About Is Love | 30 | 8.0 | 156 |

1. **Metode ML**

Metode Machine Learning: K-Means Clustering

Tools :deepnote.com

K- Means Clustering merupakan sesuatu tata cara penganalisaan informasi ataupun tata cara Data Mining yang melaksanakan proses pemodelan tanpa supervisi( unsupervised) serta merupakan salah satu tata cara yang melaksanakan pengelompokan informasi dengan sistem partisi.

Data clustering menggunakan metode K-Means Clustering ini secara umum dilakukan dengan algoritma dasar sebagai berikut:

1. Tentukan jumlah cluster
2. Alokasikan data ke dalam cluster secara random
3. Hitung centroid/rata-rata dari data yang ada di masing-masing cluster
4. Alokasikan masing-masing data ke centroid/rata-rata terdekat
5. Kembali ke Step 3, apabila masih ada data yang berpindah cluster atau apabila perubahan nilai centroid, ada yang di atas nilai threshold yang ditentukan atau apabila perubahan nilai pada objective function yang digunakan di atas nilai threshold yang ditentukan.

Library yang digunakan adalah:

1. numpy
2. matplotlib.pyplot
3. pandas
4. **Penjelasan Cara Kerja Code**

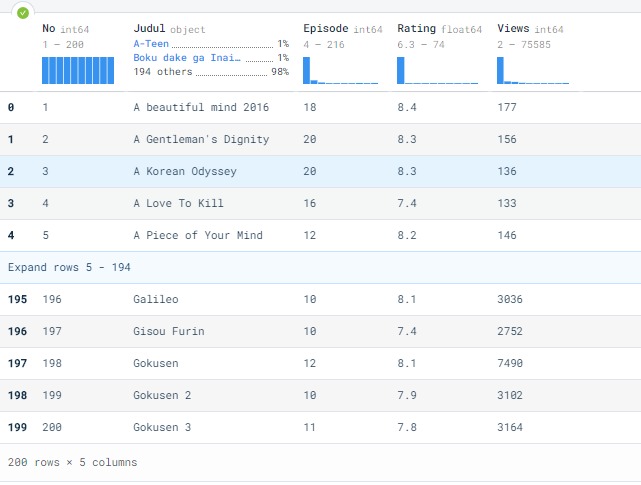
**Program Pyton**

1. # Mengimpor library
2. import numpy as np
3. import matplotlib.pyplot as plt
4. import pandas as pd
5. #Mengimpor dataset
6. dataset = pd.read\_csv('DataDramaAsia.csv')
7. X = dataset.iloc[:, [2,4]].values
8. dataset
9. # Menggunakan metode elbow untuk menentukan angka cluster yang tepat
10. from sklearn.cluster import KMeans
11. wcss = []
12. for i in range(1, 11):
13. kmeans = KMeans(n\_clusters = i, init = 'k-means++', random\_state = 42)
14. kmeans.fit(X)
15. wcss.append(kmeans.inertia\_)
16. plt.plot(range(1, 11), wcss)
17. plt.title('Metode Elbow')
18. plt.xlabel('Jumlah clusters')
19. plt.ylabel('WCSS')
20. plt.show()
21. # Menjalankan K-Means Clustering ke dataset
22. kmeans = KMeans(n\_clusters = 3, init = 'k-means++', random\_state = 42)
23. y\_kmeans = kmeans.fit\_predict(X)
24. # Visualisasi hasil clusters
25. plt.scatter(X[y\_kmeans == 0, 0], X[y\_kmeans == 0, 1], s = 100, c = 'red', label = 'Cluster 1')
26. plt.scatter(X[y\_kmeans == 1, 0], X[y\_kmeans == 1, 1], s = 100, c = 'blue', label = 'Cluster 2')
27. plt.scatter(X[y\_kmeans == 2, 0], X[y\_kmeans == 2, 1], s = 100, c = 'green', label = 'Cluster 3')
28. plt.scatter(kmeans.cluster\_centers\_[:, 0], kmeans.cluster\_centers\_[:, 1], s = 100, c = 'yellow', label = 'Centroids')
29. plt.title('Clusters drama asia')
30. plt.xlabel('Episode (1-200')
31. plt.ylabel('Views(1-100000)')
32. plt.legend()
33. plt.show()

**Penjelasan Program**

**Line 1-4** Berguna untuk mengimpor library yang dibutuhkan

**Line 6** Mengimpor dataset DataDramaAsia.csv yang isinya kurang lebih sebagai berikut:

****

**Line 7**, melakukan slicing, dari dataset yang dimiliki. Kita hanya memerlukan kolom ke 2 (episode) dan 4 (views) saja.

**Line 10**, mengimpor library K-Means.

**Line 11**, membuat list WCSS (mempersiapkan perhitungan WCSS).

**Line 12** adalah perintah looping, perlu diingat bahwa kita ingin melakukan looping 10 kali. Oleh karena itu di python ditulis range(1,11), karena angka 11 tidak diikutkan oleh python.

**Line 13** adalah menuliskan objek kmeans untuk melakukan algoritma K-Means. Selanjutnya perintah pertama adalah KMeans (kapital K dan M), yang merupakan class dari library K-Means yang diimpor di line 10, dengan beberapa parameter n\_clusters yang merupakan jumlah kluster, diikuti dengan parameter kedua init yang merupakan pemilihan jumlah K di awal (kali ini kita gunakan K++, agar tidak terkena jebakan centroid. Kemudian parameter yang terakhir adalah random\_state = 42. Random state ini seperti seed pada R, yang jika dipilih 42, maka ketika kita memilih 42 di kesempatan yang berbeda, maka bilangan random yang dihasilkan akan sama.

**Line 14** merupakan perintah agar objek kmeans di line 13, digunakan untuk mengolah data X yang sudah kita definisikan di line 7.

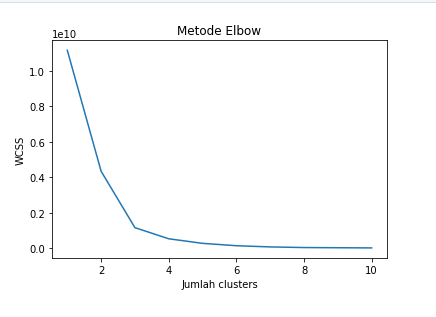
**Line 15** merupakan perintah untuk menghitung WCSS dengan menuliskan perintah append setelah wcss. Append merupakan method di python untuk menambahkan objek. Algoritma wcss dituliskan dengan perintah kmeans.inertia\_ (dengan underscore).

**Line 16** merupakan perintah untuk menampilkan plot. Sumbu x pada plot adalah jumlah kluster dari 1-10, maka ditulis range(1,11). Sumbu y nya adalah skor wcss yang dihitung di line 15.

**Line 17-19** adalah perintah plot untuk estetika, seperti nama sumbu x, sumbu y dll.

**Line 20** adalah perintah menampilkan plotnya.

Jika benar, maka tampilan plotnya akan tampak sebagai berikut:



Hasil perhitungan WCSS dari K=1 sampai K=10

Melalui gambar di atas, dapat dilihat bahwa bentuk elbow (siku) terlihat saat jumlah kluster adalah 3. Oleh karena itu, kita tentukan bahwa jumlah K yang baik adalah 3.

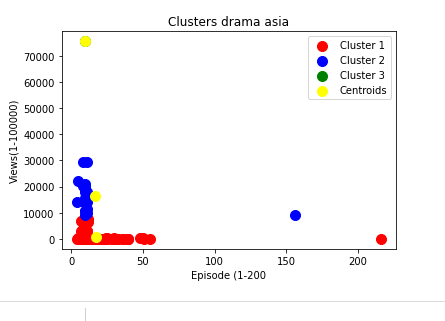
**Line 22** adalah melakukan prediksi seperti apa pengelompokan klusternya jika kita pilih K=3. Kita siapkan objek y\_kmeans (tentu saja pemilihan nama ini bebas) dengan method bukan fit melainkan fit\_predict terhadap variabel X yang sudah didefinisikan di line 7.

**Line 25-28** menampilkan hasil clusteringnya.

**Line 29-33** adalah perintah untuk menampilkan plotnya.

1. **Grafik Hasil**

Tampilan model dari clusters drama asia. Gambar ini adalah hasil , agar drama asia tersebut dapat tersusun sesuai variabel yang ditentukan berdasarkan view dan episode drama asia.



Dari hasil grafik diatas memiliki jarak yang berdekatan denngan satu clusternya sehingga sudah cukup baik untuk digunakan sebagai pengelompokan namun ada cluster yang terpisah sangat jauh dan memiliki data yang sedikit. Untuk mengatasi masalah tersebut bisa dilakukan proses normalisasi terlebih dahulu agar dapat mengatasi masalah tersebut.

1. **Kesimpulan dan Saran**

**Kesimpulan:**

Dari grafik clustering diatas dapat disimpulkan bahwa dataset drama asia yang dikelompokkan menjadi empat berdasarkan variabel episode dan viewers yang telah terbagi data poin kedalam yang cukup rapi. Dapat dilihat semua data poin masuk kedalam cluster masing-masing. Pembagian clusternya sudah cukup baik dapat dilihat dari jarak data poin yang tidak saling berdekatan.

**Saran:**

Untuk penelitian selanjutnya disarankan untuk mencari data yang memiliki jarak yang tidak terlalu jauh agar titik clustering saling berdekatan dan juga dapat menambahkan pada programnya fitur untuk melihat setiap cluster terdiri dari data apa saja.

**LINK YOUTUBE:** <https://youtu.be/DXhn_nogn1A>