Klasfikasi Anime dengan KMeans

Sabtu, 5 Agustus 2023

Vektor Team

Latar Belakang

1. Sumber data yang digunakan dalam projek ini berasal dari https://www.kaggle.com/datasets/crxxom/all-animes-in-mal
2. Melakukan klasifikasi dengan menggunakan K-Means

Tujuan

1. Mengidentifikasi anime terpopuler berdasarkan cluster yang sudah dimodelkan menggunakan algoritma K-Means

Alur Pengerjaan

DATA UNDERSTANDING

EXPLORATORY DATA ANALYSIS

VISUALIZATION

FEATURE ENGINEERING

MODELING

DATA UNDERSTANDING

Latar Belakang

Anime adalah animasi asal Jepang yang digambar dengan tangan maupun menggunakan teknologi komputer. Kata anime merupakan singkatan dari animation dalam bahasa Inggris, yang merujuk pada semua jenis animasi Di luar Jepang, istilah ini digunakan secara spesifik untuk menyebutkan segala animasi yang diproduksi di Jepang. Meskipun demikian, tidak menutup kemungkinan bahwa anime dapat diproduksi di luar Jepang.

source : https://id.wikipedia.org/wiki/Anime

Feature

Kumpulan data ini berisi informasi mendetail lebih dari 20k+ anime yang terdaftar di myanimelist dengan fitur-fitur berikut:

1. judul : judul anime
2. episode : jumlah episode
3. status : apakah anime masih tayang atau sudah selesai tayang
4. tema : tema anime
5. demografi: demografi anime (misalnya shonen, shojo, seinen dan josei)
6. genre: genre anime
7. ketik: apakah anime itu acara tv atau film dll
8. favorit: jumlah pengguna terautentikasi yang memfavoritkan anime
9. Popularitas: peringkat anime berdasarkan jumlah anggota.
10. rank : peringkat anime berdasarkan skor dibandingkan dengan anime lainnya
11. skor: skor rata-rata dari semua pengguna.
12. member : jumlah total orang yang menambahkan anime ke daftar.
13. sinopsis : plot anime
14. tayang : saat anime ditayangkan
15.durasi: durasi anime misalnya. durasi per episode
16. premiered: musim dimana anime tersebut ditayangkan
17. studio : studio yang memproduksi anime tersebut

EXPLORATORY DATA ANALYS

Data Set View

```
[19] df.head()
```

	Unnamed: 0	title	episodes	status	theme	demographic	genres	type	favorites	popularity	rank	score	members
0	0	Fullmetal Alchemist: Brotherhood	64	Finished Airing	Military	Shounen	Action,Adventure,Drama,Fantasy,	TV	218,277	#3	#1	9.10	3,190,961
1	1	Steins;Gate	24	Finished Airing	Unknown	Unknown	Drama,Sci-Fi,Suspense,	TV	183,696	#13	#2	9.07	2,452,142
2	2	Bleach: Sennen Kessen-hen	13	Finished Airing	Unknown	Shounen	Action,Adventure,Fantasy,	TV	18,421	#458	#3	9.06	455,428
3	3	Gintama°	51	Finished Airing	Unknown	Shounen	Action,Comedy,Sci-Fi,	TV	16,042	#332	#4	9.06	599,235
4	4	Kaguya-sama wa Kokurasetai: Ultra Romantic	13	Finished Airing	School	Seinen	Comedy,Romance,	TV	29,397	#193	#5	9.05	832,346

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 24262 entries, 0 to 24261
Data columns (total 18 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Unnamed: 0    24262 non-null  int64
 1   title         24262 non-null  object
 2   episodes      24262 non-null  object
 3   status        24262 non-null  object
 4   theme         24262 non-null  object
 5   demographic   24262 non-null  object
 6   genres        24262 non-null  object
 7   type          24262 non-null  object
 8   favorites     24262 non-null  object
 9   popularity    24262 non-null  object
 10  rank          20197 non-null  object
 11  score         15294 non-null  float64
 12  members       24262 non-null  object
 13  synopsis      24262 non-null  object
 14  aired         24262 non-null  object
 15  duration      24262 non-null  object
 16  premiered     24262 non-null  object
 17  studios       24262 non-null  object
dtypes: float64(1), int64(1), object(16)
memory usage: 3.3+ MB
```

DATA PROCESSING

Hapus Kolom

Penghapusan kolom yang tidak memberikan informasi atau memiliki deskrispi atau nilai yang sama dengan kolom lainnya.

```
# Menghapus kolom-kolom tertentu dari DataFrame
df = df.drop(['Unnamed: 0', 'status', 'synopsis', 'episodes', 'premiered', 'members', 'aired'], axis=1)

# Menampilkan DataFrame setelah menghapus kolom
print(df)
```

Pengecekan apakah ada duplikat. Jika **ADA** maka di hapus.

```
[27] df.duplicated().sum()

     27

[28] df.drop_duplicates(inplace=True)
```

Ubah Tipe Data

Mayoritas tipe data adalah **Object.**

Ubah **favorites, popularity, rank dan duration**

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 24262 entries, 0 to 24261
Data columns (total 18 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   Unnamed: 0   24262 non-null  int64
 1   title        24262 non-null  object
 2   episodes     24262 non-null  object
 3   status       24262 non-null  object
 4   theme        24262 non-null  object
 5   demographic  24262 non-null  object
 6   genres       24262 non-null  object
 7   type         24262 non-null  object
 8   favorites    24262 non-null  object
 9   popularity   24262 non-null  object
 10  rank         20197 non-null  object
 11  score        15294 non-null  float64
 12  members      24262 non-null  object
 13  synopsis     24262 non-null  object
 14  aired        24262 non-null  object
 15  duration     24262 non-null  object
 16  premiered    24262 non-null  object
 17  studios      24262 non-null  object
dtypes: float64(1), int64(1), object(16)
memory usage: 3.3+ MB
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 12803 entries, 0 to 12812
Data columns (total 11 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   title        12803 non-null  object
 1   theme        12803 non-null  object
 2   demographic  12803 non-null  object
 3   genres       12803 non-null  object
 4   type         12803 non-null  object
 5   favorites    12803 non-null  int64
 6   popularity   12803 non-null  int64
 7   rank         12803 non-null  int64
 8   score        12803 non-null  float64
 9   duration     12803 non-null  object
 10  studios      10544 non-null  object
dtypes: float64(1), int64(3), object(7)
memory usage: 1.2+ MB
```

Cek Anomali Pada Data

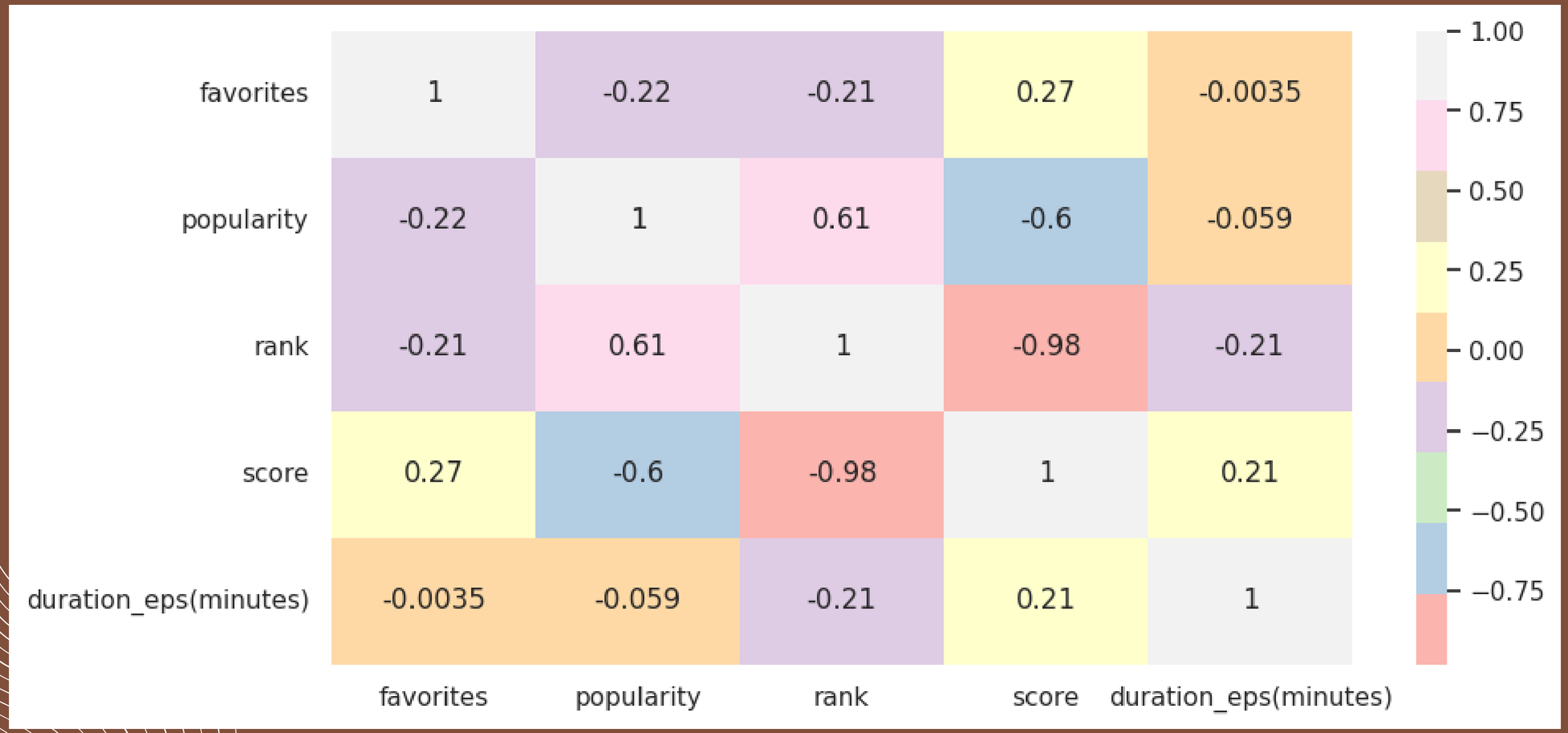
Melakukan pengecekan apakah setiap kolom memiliki data kosong atau tidak sesuai

```
title                       0
theme                       0
demographic                 0
genres                      0
type                        0
favorites                2056
popularity                  0
rank                        0
score                       0
duration_eps(minutes)     384
studios                  2259
dtype: int64
```

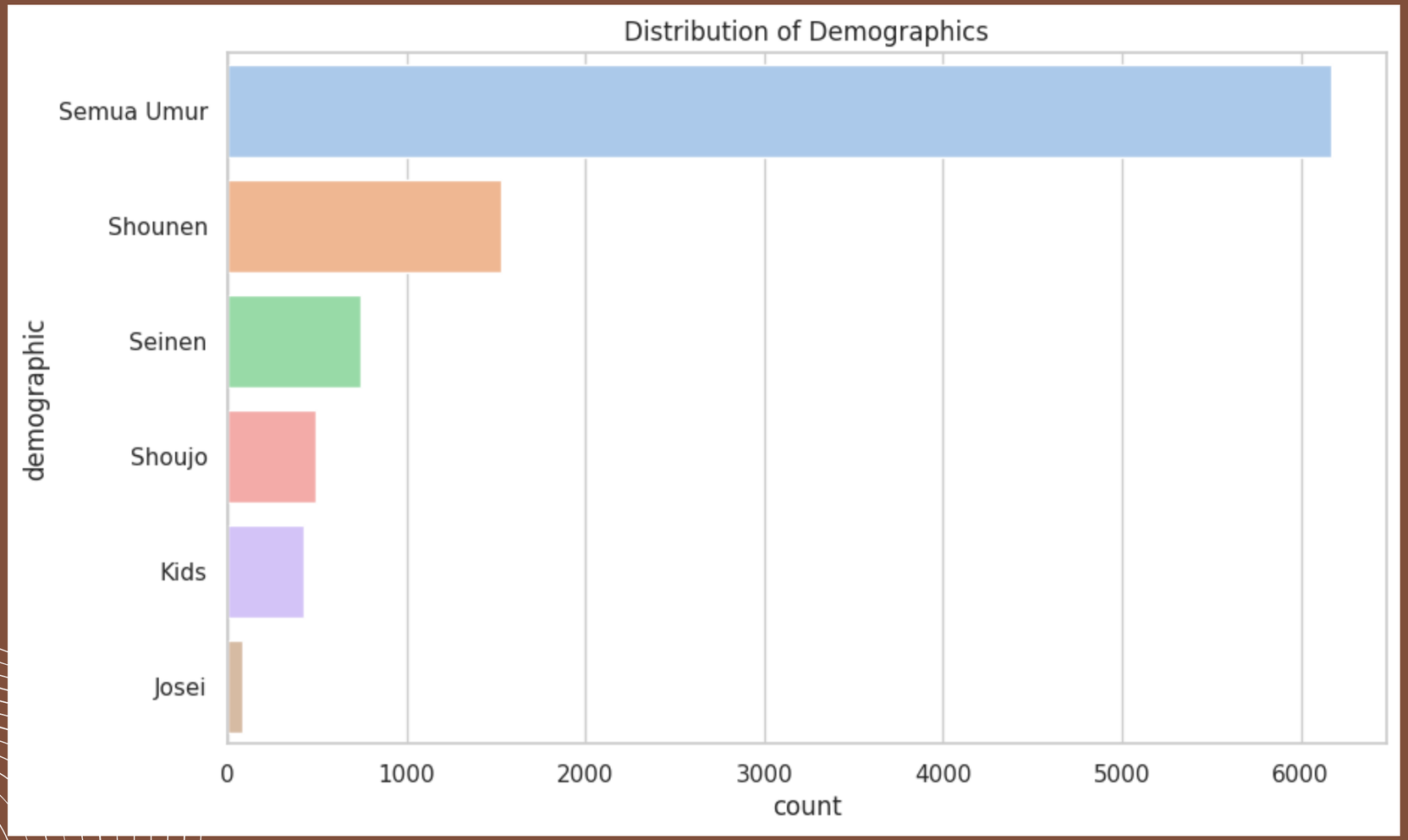
```
title                    0
theme                    0
demographic              0
genres                   0
type                     0
favorites                0
popularity               0
rank                     0
score                    0
duration_eps(minutes)    0
studios                  0
dtype: int64
```

VISUALIZATION

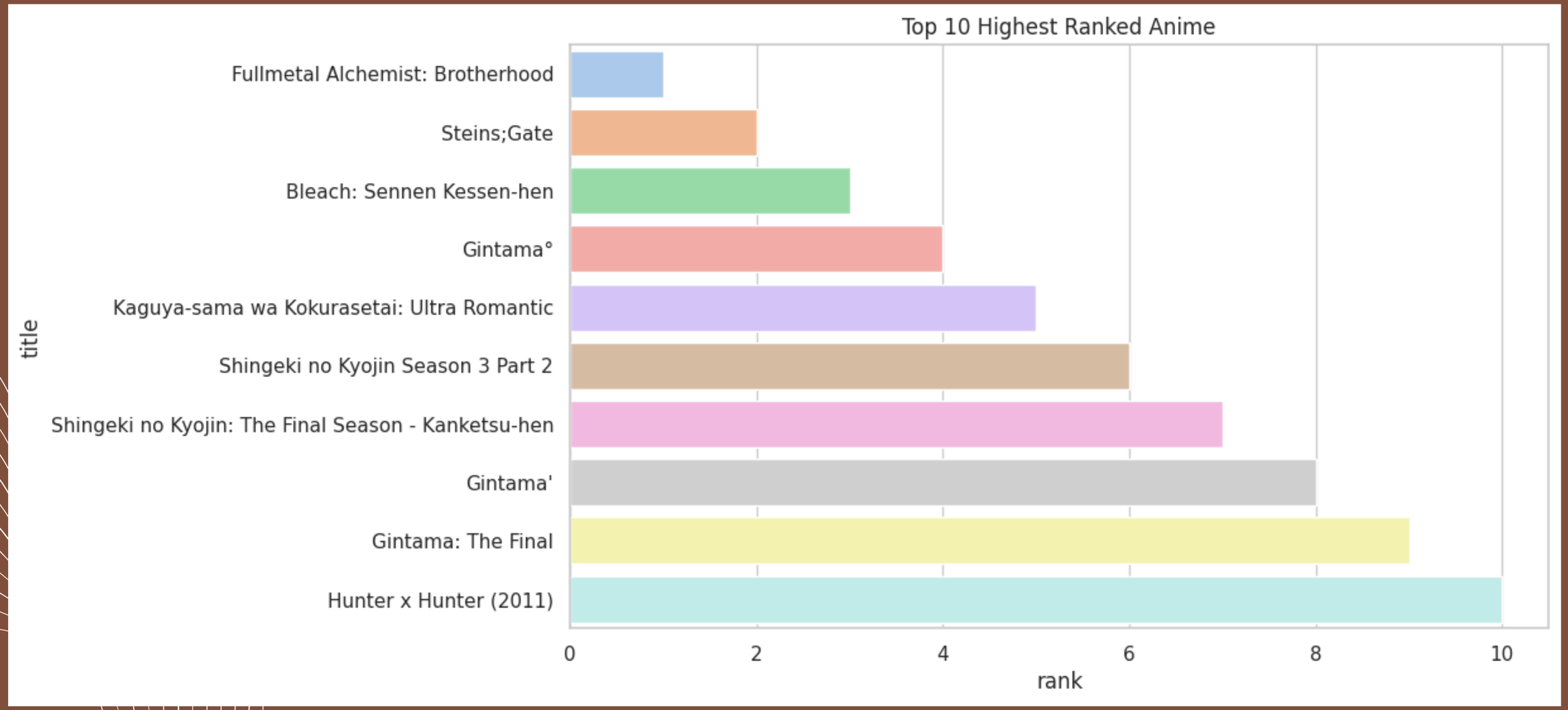
Heat Map


favorites
popularity
rank
score
duration_eps(minutes)
1
-0.22
-0.21
0.27
-0.0035
-0.22
1
0.61
-0.6
-0.059
-0.21
0.61
1
-0.98
-0.21
0.27
-0.6
-0.98
1
0.21
-0.0035
-0.059
-0.21
0.21
1
favorites
popularity
rank
score
duration_eps(minutes)
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75

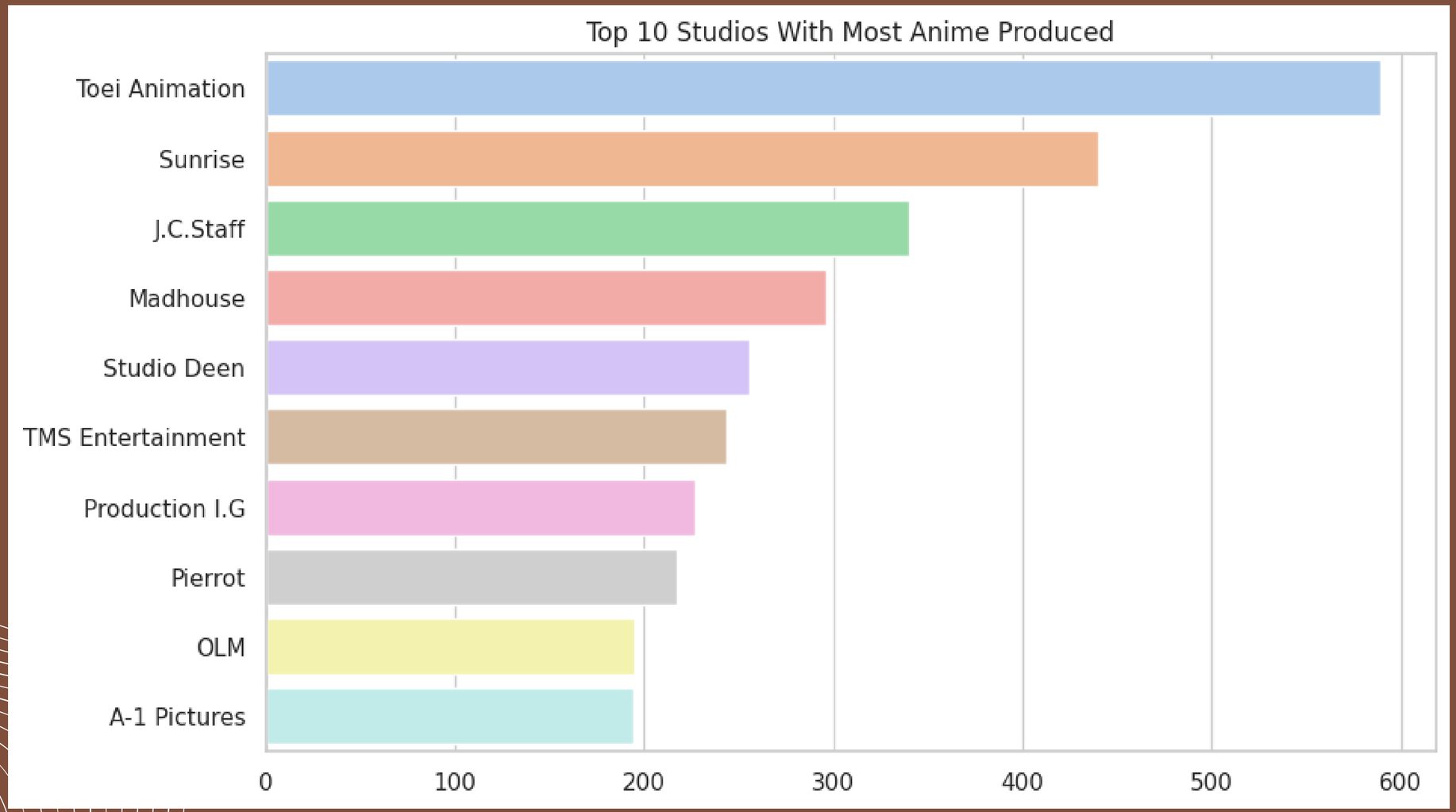
Distribusi dari Demographics



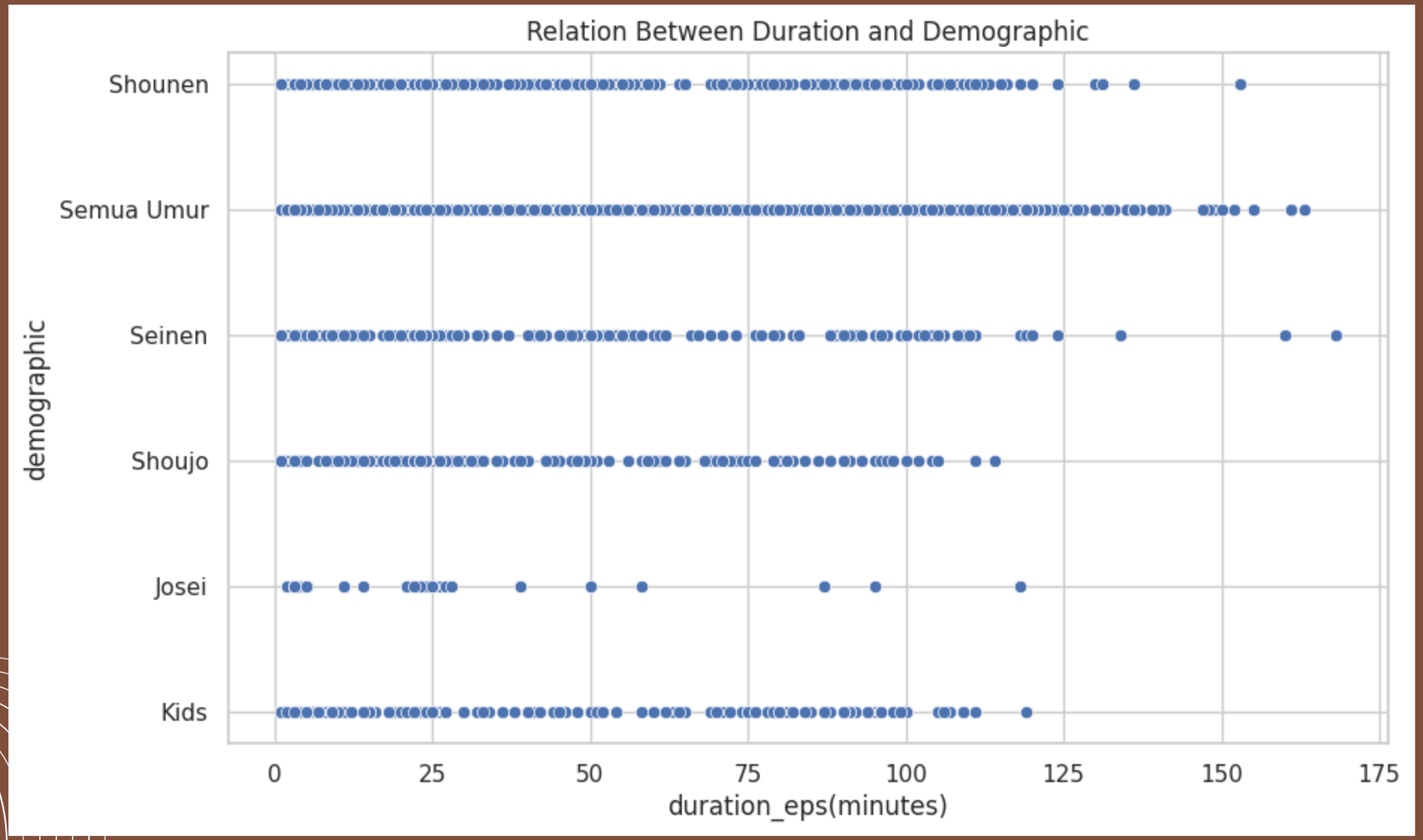
10 Anime Tertinggi Berdasarkan Ranking



10 Studio Tertinggi



Relasi Antara Durasi Dan Demographic



FEATURE ENGINEERING

Binning

Melakukan pengelompokkan (binning) data dalam sebuah DataFrame menggunakan nilai dari kolom ke dalam kategori berdasarkan rentang nilai tertentu.

```python
df['rank_range'] = pd.cut(
    x=df['rank'],
    bins=[0, 4271, 8542, 12813],
    labels=["low", "moderate", "high"]
)

df['score_range'] = pd.cut(
    x=df['score'],
    bins=[0, 4.2, 6.6, 9.1],
    labels=["bad", "good", "excellent"]
)

df['popularity_range'] = pd.cut(
    x=df['popularity'],
    bins=[0, 6177, 12353, 18529],
    labels=["unpopular", "popular", "very popular"]
)
```

rank_range	score_range	popularity_range	favorites_range
low	excellent	unpopular	yes
low	excellent	unpopular	yes
low	excellent	unpopular	no
low	excellent	unpopular	no
low	excellent	unpopular	no

Label Encoding

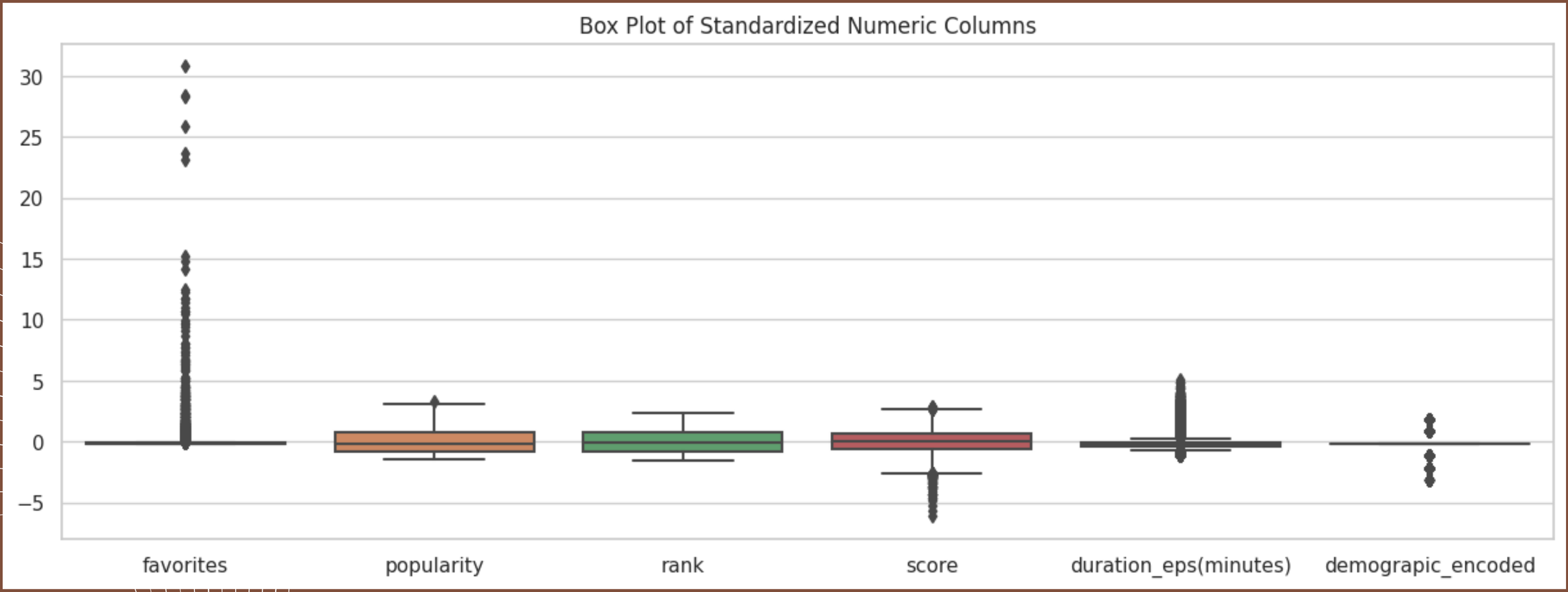
Melakukan proses mengubah nilai-nilai dalam sebuah kolom (sering kali berisi data kategori atau label) menjadi bilangan bulat.

```
# Membuat objek LabelEncoder
encoder = LabelEncoder()

# Menggunakan LabelEncoder untuk mengubah nilai kategori menjadi bilangan bulat
df_new['demograpic_encoded'] = encoder.fit_transform(df_new['demographic'])

# Menampilkan DataFrame hasil
print(df_new)
```

SCALLING

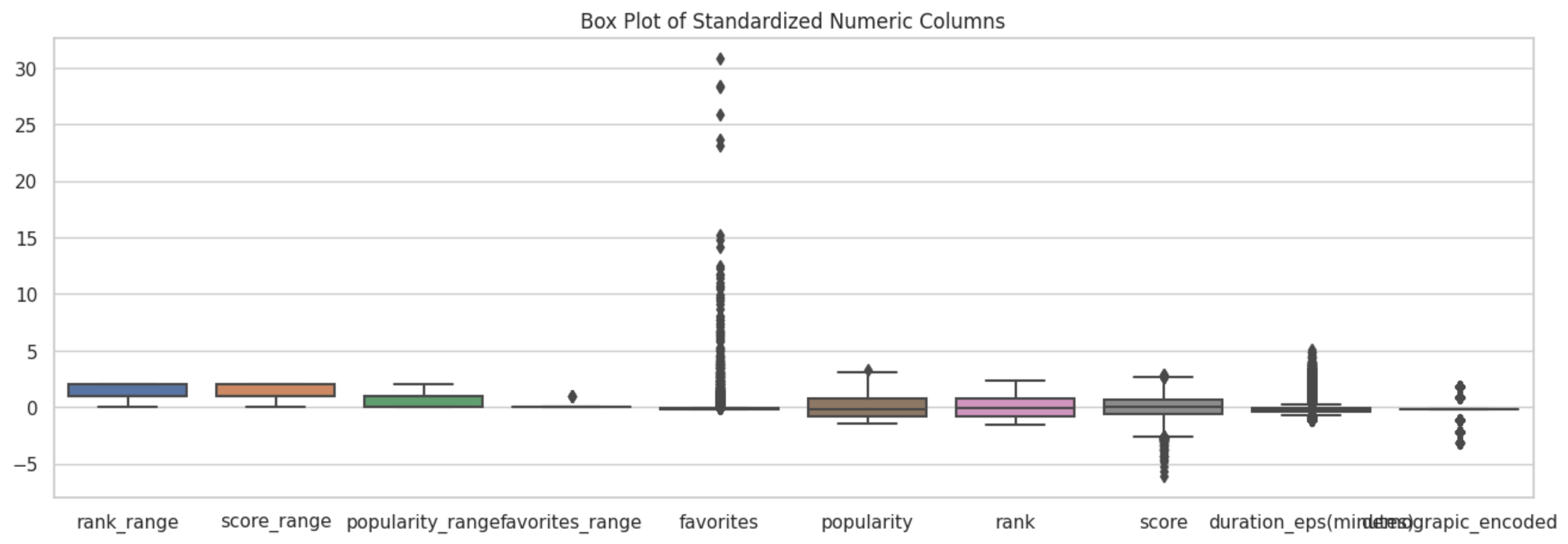
Scalling Pada Numeric Coloums

Melakukan proses untuk mengubah rentang nilai dari fitur (features) dalam dataset ke dalam rentang yang lebih spesifik atau lebih terstandarisasi.



Scalling Pada Object Coloums

Melakukan proses untuk mengubah rentang nilai dari fitur (features) dalam dataset ke dalam rentang yang lebih spesifik atau lebih terstandarisasi.



MODELING

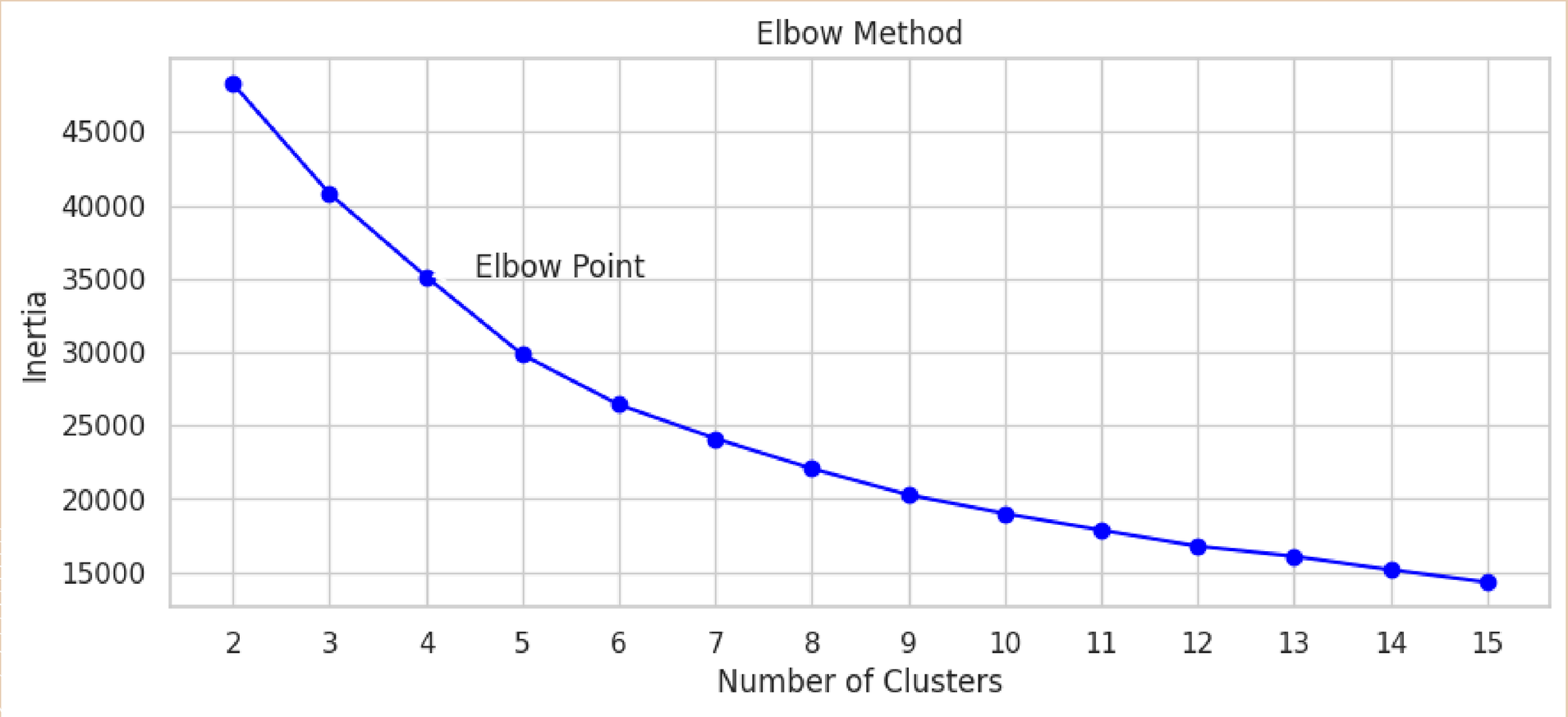
Modeling K-Means

Melakukan proses pembelajaran mesin yang digunakan untuk tugas clustering atau pengelompokan data.

```
# Modeling
range_clusters = list(range(2, 16))
inertia = []

for k in range_clusters:
    kmeans = KMeans(n_clusters=k)
    kmeans.fit(df_model)
    inertia.append(kmeans.inertia_)
```

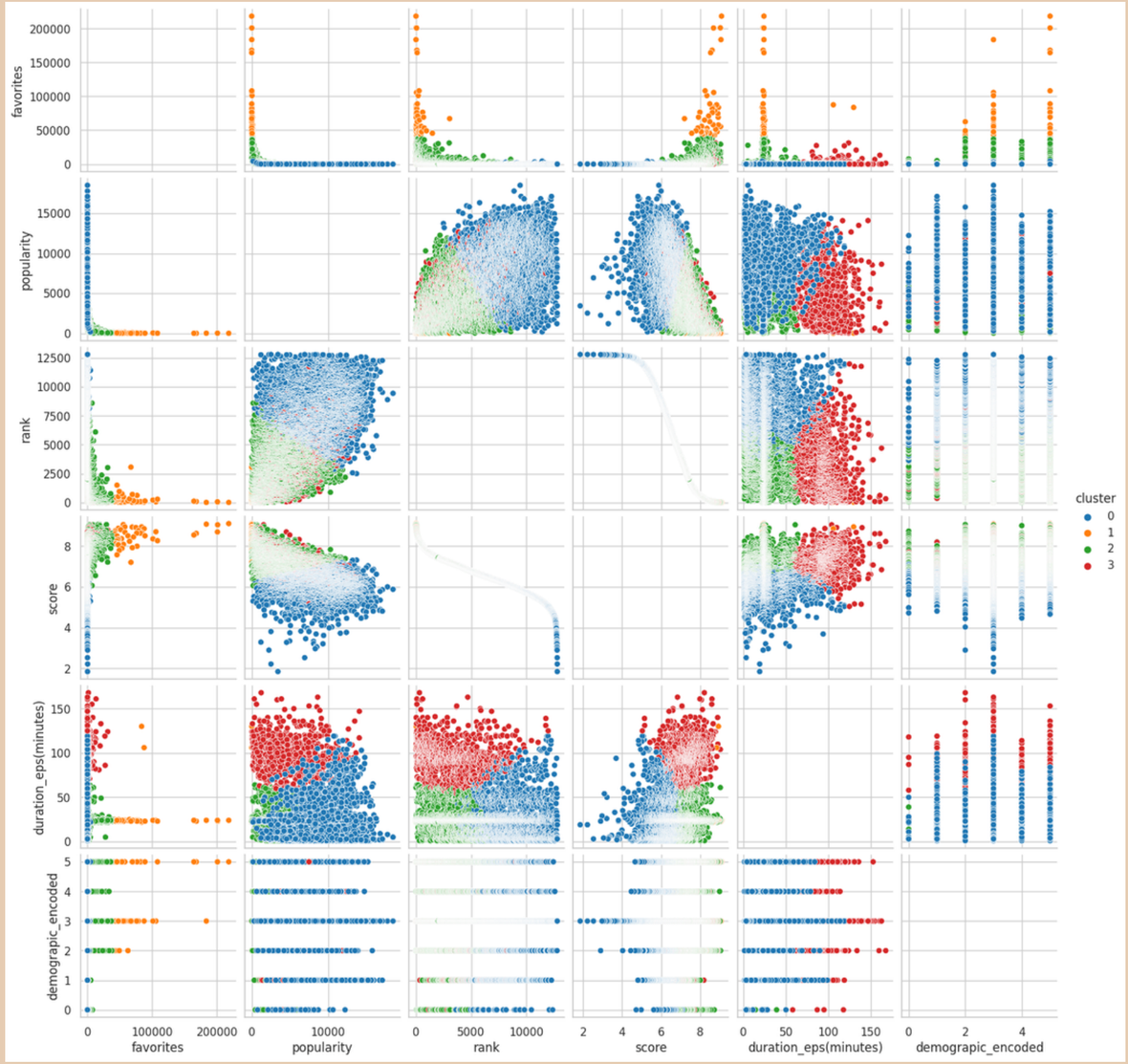
Elbow Method

Melakukan proses visual yang digunakan dalam analisis kluster untuk membantu menentukan jumlah optimal kluster (clusters) dalam suatu datase



Scatter Plot

menjalankan algoritma K-Means clustering pada dataset yang sudah di-standardisasi (df_std) dengan jumlah kluster yang ditentukan sebanyak 4



Hasil Statistik

cluster		0	1	2	3
favorites	mean	26.136869	80231.977778	1416.113320	586.763780
	std	134.746419	47245.879340	4149.618936	2330.076264
	min	1.000000	41929.000000	1.000000	1.000000
	q25	2.000000	47775.000000	34.000000	9.000000
	median	5.000000	65050.000000	139.000000	36.000000
	q75	14.000000	83739.000000	751.500000	201.250000
	max	4509.000000	218277.000000	38182.000000	31310.000000
popularity	mean	8723.638292	39.822222	3295.546720	4945.575787
	std	3301.279504	36.447527	2392.327702	3167.859515
	min	175.000000	1.000000	12.000000	44.000000
	q25	6278.000000	13.000000	1368.500000	2300.500000
	median	8958.000000	28.000000	2858.000000	4496.500000
	q75	11222.000000	50.000000	4736.500000	7309.250000
	max	18529.000000	140.000000	12277.000000	14116.000000
rank	mean	8271.778266	293.444444	2969.000663	3159.451772
	std	2094.508017	515.969412	1775.816770	2326.271896
	min	2491.000000	1.000000	3.000000	9.000000
	q25	6682.000000	42.000000	1487.500000	1217.750000
	median	8111.000000	109.000000	2879.000000	2713.500000
	q75	9819.000000	307.000000	4327.500000	4644.000000
	max	12813.000000	3056.000000	8629.000000	11990.000000
score	mean	6.065612	8.471111	7.287020	7.266752
	std	0.552802	0.421713	0.492966	0.613129
	min	1.850000	7.200000	6.060000	5.040000
	q25	5.780000	8.230000	6.920000	6.850000
	median	6.160000	8.540000	7.230000	7.270000
	q75	6.450000	8.750000	7.570000	7.670000
	max	7.310000	9.100000	9.060000	9.040000
duration_eps(minutes)	mean	22.009573	27.888889	23.025403	96.610236
	std	17.420454	19.829068	9.336472	17.870074
	min	1.000000	22.000000	1.000000	54.000000
	q25	8.000000	23.000000	23.000000	87.000000
	median	23.000000	24.000000	24.000000	95.000000
	q75	25.000000	24.000000	24.000000	107.000000
	max	119.000000	130.000000	64.000000	168.000000

KESIMPULAN

Modeling dilakukan dengan menggunakan K-Means dengan mempertimbangkan semua fiture yang diperlukan. Dengan menggunakan metode Elbow di tentukan cluster terbagi menjadi 4 kelas.

Kemudian setelah ditentukan cluster terbagi menjadi 4 cluster, dilakukan perhitungan anal;isa dengan menggunakan **'mean', 'std', 'min', q25, 'median', q75, 'max'** ditemukan bahwa cluster 0 menjadi cluster terbaik.

KONTRIBUSI

Tugas dan %kontribusi

Pembagian tugas diberikan berdasarkan kesiapan anggota. Adapun perbedaan% dikarenakan setiap anggota memiliki tanggung jawab dan kontibusi di setiap tugas.

Tugas	Agus A.M	Fadhil J.V	Mulia P.S	Zakaria F	Zulfaikar F
Data understanding	20 %	20 %	20 %	20 %	20 %
Exploration Data Analys	25 %	-	25 %	25 %	25 %
Visualization	50%	-	-	50%	-
Feature Engineering	-	-	100%	-	-
Modeling	-	-	50%	-	50%

Thank you for listening!

Vektor Team

Sabtu, 05 Agustus 2023