Chairul Imam
1608107010040

Deskripsi Dataset

Saya menggunakan dataset **Seeds** dari Repository UCI Machine Learning. Dataset terdiri dari **7 feature predictor**, **1 feature target** dan **210 instance**. Feature class terdiri dari tiga jenis varietas gandum yaitu Kama, Rosa dan Canadian, masing-masing 70 instance.

Persebaran varietas:

- instance 1-70 atau varieties **1** untuk **Kama**
- instance 71-140 atau varieties **2** untuk **Rosa**
- instance 141-210 atau varieties **3** untuk **Canadian**

Untuk informasi detail mengenai dataset dan mengunduh dataset dapat diakses pada URL berikut: Seeds

Langkah yang dilakukan

Langkah dan screenshot yang dimuat pada dokumen ini adalah snippet utama dari notebook yang lebih lengkap. Source code yang lebih lengkap dapat diakses pada bagian **Source Code**.

Scaling Dataset

Scaling dataset bertujuan agar rentang nilai setiap feature berada pada rentang yang sama. Saya menggunakan *MinMaxScaler* dengan rentang nilai (0, 1).

```
from sklearn.preprocessing import MinMaxScaler
```

```
# atur rentang nilai
scaler = MinMaxScaler(feature_range=(0, 1))

# tranformasi nilai feature ke range (0, 1)
X_scaled = scaler.fit_transform(X)
```

```
# konversi ke Dataframe
X_scaled = pd.DataFrame(data=X_scaled,
                        columns=colnames[:-1])

X_scaled.head()
```

Dimension Reduction dengan PCA

Penggunaan PCA bertujuan untuk kemudahan visualisasi cluster yang akan dibentuk. Hasil dari PCA juga akan digunakan sebagai input dataset pada clustering KMeans dan Spectral.

Plot actual cluster ini akan digunakan untuk melihat perbedaan antara cluster actual (cluster sebenarnya) dan cluster yang dihasilkan oleh metode clustering **KMeans** dan **Apa**. Feature prediktor dataset seeds berdimensi 7 sehingga tidak mungkin untuk memvisualisasikan persebaran data point. Untuk itu, kita akan menggunakan **PCA (Principal Component Analysis)** untuk mereduksi dimensi dataset menjadi 2.

```
from sklearn.decomposition import PCA
```

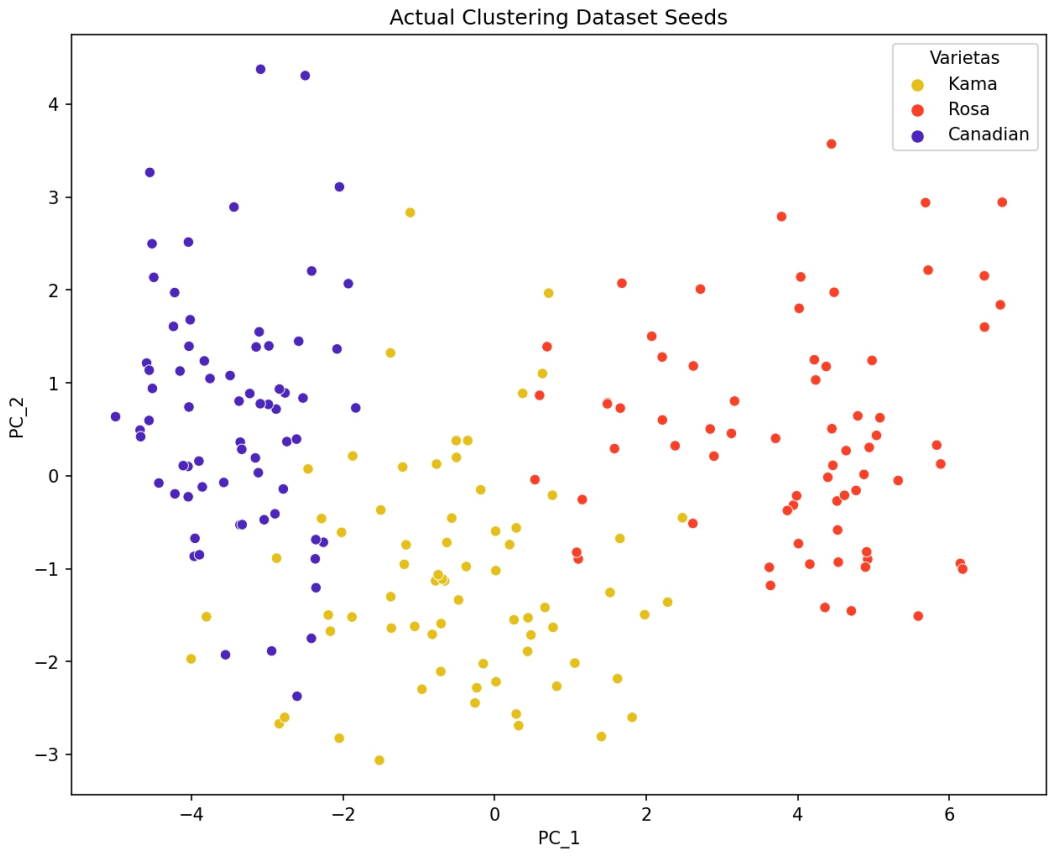
```
# atur dimensi
pca = PCA(n_components=2)

# reduksi X_scale menjadi 2 dimensi
X_reduced = pca.fit_transform(X)
```

```
# konversi X_reduced ke DataFrame
X_reduced = pd.DataFrame(data=X_reduced,
                         columns=['PC_1', 'PC_2'])
```

X_reduce adalah dataframe dengan shape (210, 2). X_reduce akan digunakan sebagai data yang akan diklustering.

Actual Cluster

Gambar dibawah ini adalah cluster sebenarnya yang diplot dari dataset.

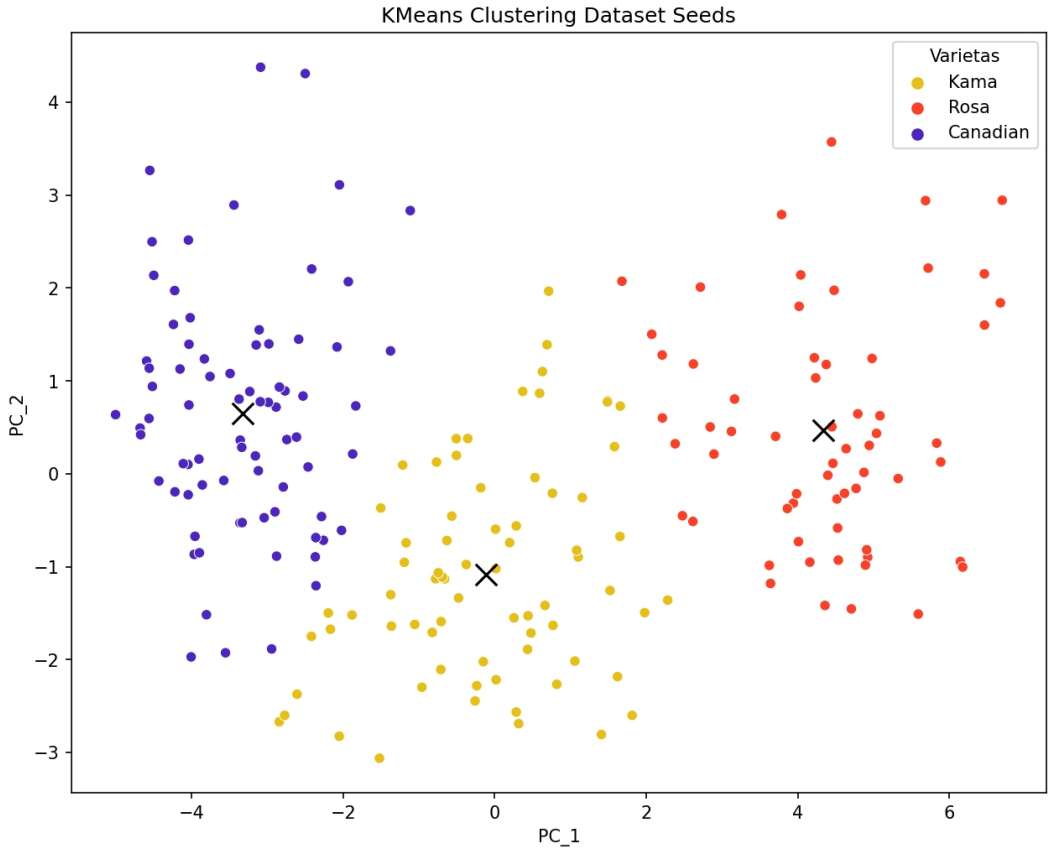


Clustering

KMeans

Dilakukan clustering dengan algoritma KMeans, dengan snippet dan plot cluster sebagai berikut:

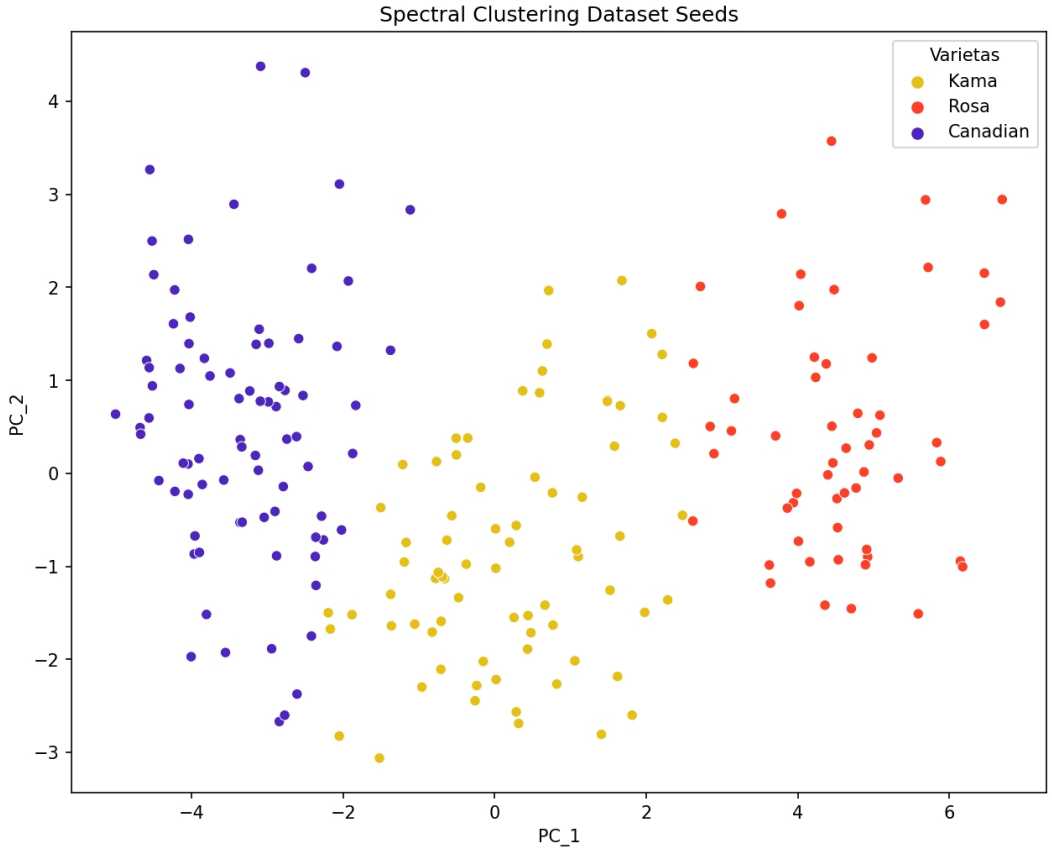
```
fig = plt.figure(figsize=(10, 8),
                 dpi=150,
                 facecolor='white')

ax = sns.scatterplot(x=df_kmeans['PC_1'],
                     y=df_kmeans['PC_2'],
                     hue=df_kmeans['varieties'],
                     hue_order=['Kama', 'Rosa', 'Canadian'],
                     palette=sns.color_palette('CMRmap_r', 3))

# plot centroids
plt.scatter(x=km_centroids[:, 0],
            y=km_centroids[:, 1],
            marker='x',
            c='k',
            s=150)

# set title legend
ax.legend(title="Varietas")

plt.title('KMeans Clustering Dataset Seeds')

# simpan figure
plt.savefig('../output/figures/kmeans_cluster.png',
            bbox_inches='tight')
```



Spectral Clustering

Dilakukan clustering dengan algoritma Spectral, dengan snippet dan plot cluster sebagai berikut:

```
fig = plt.figure(figsize=(10, 8),
                 dpi=150,
                 facecolor='white')

ax = sns.scatterplot(x=df_spec['PC_1'],
                     y=df_spec['PC_2'],
                     hue=df_spec['varieties'],
                     hue_order=['Kama', 'Rosa', 'Canadian'],
                     palette=sns.color_palette('CMRmap_r', 3))

# set title legend
ax.legend(title="Varietas")

plt.title('Spectral Clustering Dataset Seeds')

# simpan figure
plt.savefig('../output/figures/spectral_cluster.png',
            bbox_inches='tight')
```



Source Code

Untuk mengakses dan mencoba source code, silakan mengakses pada URL berikut:

https://github.com/chairul-imam/Data-Mining-and-Machine-Learning/blob/main/Seeds.ipynb

Referensi

Complete Gradient Clustering Algorithm for Features Analysis of X-ray Images

7. Unsupervised learning: PCA and clustering

https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_digits.html#sphx-glr-auto-examples-cluster-plot-kmeans-digits-py

Clustering Performance Evaluation | Tutorialspoint