

Dokumentasi

Implementasi Eksperimen

Analisis Pola Pembelian Pelanggan dalam Industri E-commerce Menggunakan K-means Clustering

1. Eksplorasi Data:

Pada tahap ini, memuat dataset "Online Retail Data Set" dari Kaggle menggunakan library pandas. Dataset tersebut berisi informasi tentang transaksi pembelian pelanggan di industri e-commerce, seperti nomor faktur, kode produk, deskripsi produk, jumlah barang yang dibeli, tanggal dan waktu transaksi, harga satuan produk, nomor identifikasi pelanggan, dan negara asal transaksi. Setelah memuat dataset, beberapa baris pertama dan informasi tentang tipe data serta nilai-nilai yang hilang ditampilkan untuk memahami karakteristik dataset.

```
# Import library yang diperlukan
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import silhouette_score

# Load Datasets
file_path = "OnlineRetail.csv"

# Memuat dataset dari file CSV menggunakan library pandas
df = pd.read_csv(file_path, encoding="unicode_escape")

# Eksplorasi Data
print(df.head()) # Melihat beberapa baris pertama dataset
print(df.info()) # Informasi tentang tipe data dan nilai-nilai yang hilang
```

```

InvoiceNo StockCode Description Quantity \
0 536365 85123A WHITE HANGING HEART T-LIGHT HOLDER 6
1 536365 71053 WHITE METAL LANTERN 6
2 536365 84406B CREAM CUPID HEARTS COAT HANGER 8
3 536365 84029G KNITTED UNION FLAG HOT WATER BOTTLE 6
4 536365 84029E RED WOOLLY HOTTIE WHITE HEART. 6

InvoiceDate UnitPrice CustomerID Country
0 12/1/2010 8:26 2.55 17850.0 United Kingdom
1 12/1/2010 8:26 3.39 17850.0 United Kingdom
2 12/1/2010 8:26 2.75 17850.0 United Kingdom
3 12/1/2010 8:26 3.39 17850.0 United Kingdom
4 12/1/2010 8:26 3.39 17850.0 United Kingdom
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
# Column Non-Null Count Dtype
---
0 InvoiceNo 541909 non-null object
1 StockCode 541909 non-null object
2 Description 540455 non-null object
3 Quantity 541909 non-null int64
4 InvoiceDate 541909 non-null object
5 UnitPrice 541909 non-null float64
6 CustomerID 406829 non-null float64
7 Country 541909 non-null object
dtypes: float64(2), int64(1), object(5)
memory usage: 33.1+ MB
None

```

Gambar 1 Hasil Dari Eskplorasi Data

2. Pembersihan Data:

Setelah memuat dataset, tahap berikutnya adalah membersihkan data dari data duplikat dan nilai-nilai yang hilang. Data yang bersih diperlukan agar analisis dapat dilakukan secara akurat dan menghindari bias dalam hasil.

```

# Pembersihan Data
df.drop_duplicates(inplace=True) # Menghapus data duplikat
df.dropna(inplace=True) # Menghapus baris yang mengandung nilai-nilai
yang hilang

# Preprocessing Data
df['InvoiceDate'] = pd.to_datetime(df['InvoiceDate']) # Mengonversi
InvoiceDate ke tipe datetime

# Karena K-means sensitif terhadap skala data, kita perlu melakukan
standarisasi
scaler = StandardScaler()
data_scaled = scaler.fit_transform(data[['Quantity', 'UnitPrice']])

```

3. Analisis Pola Pembelian

Pada tahap ini, akan melakukan analisis pola pembelian pelanggan berdasarkan dataset yang telah dibersihkan. Beberapa analisis yang dilakukan adalah:

- Identifikasi produk yang paling sering dibeli oleh pelanggan.
- Eksplorasi keterkaitan antara produk berdasarkan pola pembelian.
- Visualisasi tren belanja berdasarkan waktu untuk melihat bagaimana perilaku pembelian pelanggan berubah dari waktu ke waktu.

```
# Analisis Pola Pembelian
# Mencari tren belanja berdasarkan waktu
df['YearMonth'] = df['InvoiceDate'].dt.to_period('M')
shopping_trends = df.groupby('YearMonth')['Quantity'].sum()

# Konversi YearMonth dari Period ke tanggal (tanggal awal setiap bulan)
shopping_trends.index = shopping_trends.index.to_timestamp()

# Visualisasi tren belanja berdasarkan waktu
plt.figure(figsize=(10, 6))
plt.plot(shopping_trends.index, shopping_trends, marker='o', linestyle='-'
        ')
plt.xlabel('Bulan')
plt.ylabel('Jumlah Barang Terbeli')
plt.title('Tren Belanja Pelanggan Berdasarkan Waktu')
plt.xticks(rotation=45)
plt.grid(True)

plt.show()
```



Gambar 2 Visualiasi Pola Pembelian

4. Segmentasi Pelanggan

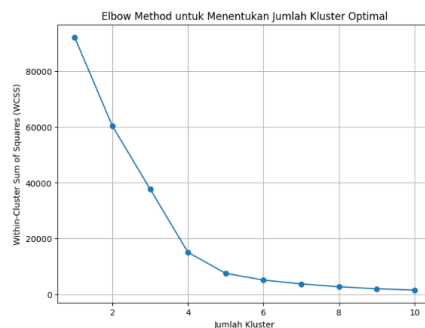
Tahap selanjutnya adalah melakukan segmentasi pelanggan berdasarkan pola pembelian mereka. Untuk melakukan hal ini, saya akan menggunakan metode K-means Clustering. Langkah-langkahnya adalah:

- Preprocessing data dengan menskalakan fitur-fitur yang relevan.
- Menentukan jumlah kluster optimal menggunakan Elbow Method.
- Menerapkan K-means Clustering dengan jumlah kluster yang optimal yang telah ditentukan sebelumnya.

```
# Segmentasi Pelanggan dengan K-means Clustering
# Menentukan jumlah kluster yang optimal dengan menggunakan Elbow Method
wcss = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters=i, init='k-means++', max_iter=300,
n_init=10, random_state=42)
    kmeans.fit(data_scaled)
    wcss.append(kmeans.inertia_)

# Plot Elbow Method untuk menentukan jumlah kluster yang optimal
plt.figure(figsize=(8, 6))
plt.plot(range(1, 11), wcss, marker='o')
plt.xlabel('Jumlah Kluster')
plt.ylabel('Within-Cluster Sum of Squares (WCSS)')
plt.title('Elbow Method untuk Menentukan Jumlah Kluster Optimal')
plt.grid(True)
plt.show()

# Berdasarkan plot Elbow Method, pilih jumlah kluster yang optimal dan
lakukan clustering
n_clusters = 4
kmeans = KMeans(n_clusters=n_clusters, init='k-means++', max_iter=300,
n_init=10, random_state=42)
data['Cluster'] = kmeans.fit_predict(data_scaled)
```



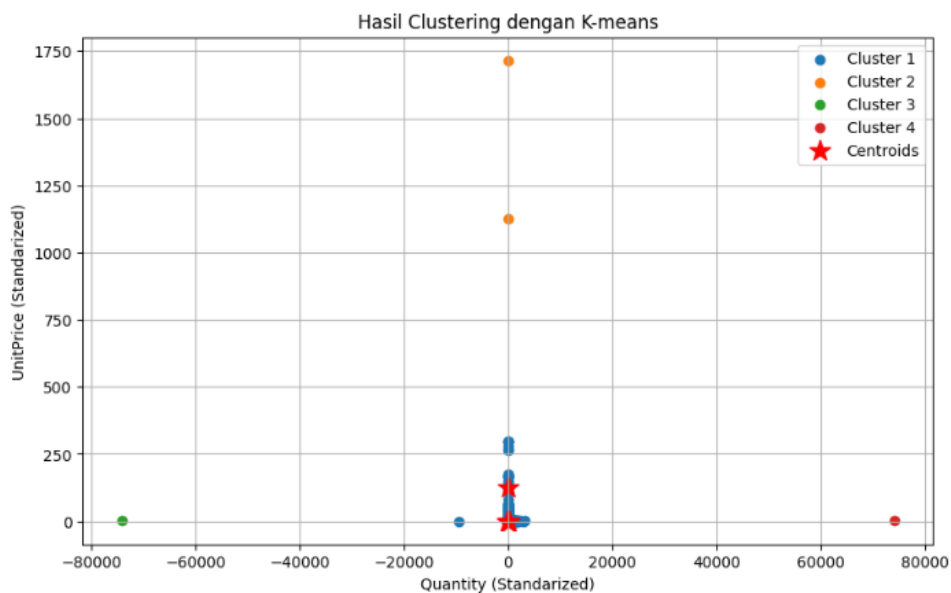
Gambar 3 Visualisasi Jumlah Cluster Optimal

5. Evaluasi dan Interpretasi

Pada tahap ini, saya akan mengevaluasi kualitas segmentasi yang dilakukan dengan menggunakan metrik evaluasi, yaitu Silhouette Score. Nilai Silhouette Score akan membantu kita menentukan seberapa baik kualitas dari kluster yang telah dipisahkan. Selanjutnya, hasil clustering akan divisualisasikan untuk menganalisis karakteristik masing-masing kelompok pelanggan.

```
# Evaluasi hasil clustering
silhouette_avg = silhouette_score(data_scaled, data['Cluster'])
print(f"Silhouette Score untuk jumlah kluster {n_clusters}:
{silhouette_avg}")

# Visualisasi hasil clustering
plt.figure(figsize=(10, 6))
for i in range(n_clusters):
    cluster_data = data[data['Cluster'] == i]
    plt.scatter(cluster_data['Quantity'], cluster_data['UnitPrice'],
label=f'Cluster {i+1}')
plt.scatter(kmeans.cluster_centers[:, 0], kmeans.cluster_centers[:, 1],
s=200, c='red', marker='*', label='Centroids')
plt.xlabel('Quantity (Standarized)')
plt.ylabel('UnitPrice (Standarized)')
plt.title('Hasil Clustering dengan K-means')
plt.legend()
plt.grid(True)
plt.show()
```



Gambar 4 Hasil Clustering dengan K-Means

6. Kesimpulan

Kesimpulannya, eksperimen ini memberikan wawasan berharga tentang perilaku pembelian pelanggan dalam industri e-commerce. Segmentasi pelanggan dengan menggunakan K-means Clustering dapat membantu perusahaan untuk mengoptimalkan strategi pemasaran, meningkatkan kepuasan pelanggan, dan meraih kesuksesan dalam bisnis e-commerce. Namun, eksperimen ini masih dapat diperluas dengan penambahan analisis asosiasi atau mencoba metode clustering lainnya untuk memperoleh informasi lebih mendalam dan akurat tentang perilaku pembelian pelanggan.