CCA-7 Determine the optimal number of clusters

Overview

This task includes the determination of optimal number of cluster by using elbow method and silhouette Analysis. After using this method on both min-max and standard scaled datasets we will analyze the outcomes plot. After analysing the plot for both datasets, we will able to determine the optimal number of clusters for each scaled datasets

Learning Objectives

As we discussed above we will learn how to determine the optimal number of clusters by using elbow method and silhouette analysis. We will use this method for our both scaled datasets. Throughout the process we will figure it out the reasons of choosing the optimal number of clusters. We will mentioned the reason behind the choosing the number of clusters. And finally we outcomes or visualize these methods to make decisions

Imports libraries

This step includes the import of necessary libraries for our project phase:

```
import os
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.metrics import silhoutte_score

```

Steps:

Step 1: Ensure the correct path for visualizations

This step includes the paths for the obtained visualizations from both methods.

```
# ensure the correct paths for saving visualizations
project_root = os.path.dirname(os.path.dirname(os.path.abspath('')))
optimal_clusters_path = os.path.join(project_root,'Clustering_Analysis', 'optimal_clusters')
os.makedirs(optimal_clusters_path, exist_ok=True)
```

```

```

Step 2: Load the required scaled datasets (min-max scaled And standard scaled dataset)

This step includes the loading of both min-max scaled and standard scaled datasets.

```
#paths to the datasets
min_max_scaled_data_path = os.path.join(project_root,'data_preparation', 'scaling_techniques', 'min_max_scaled_dataset.csv')
standard_scaled_data_path = os.path.join(project_root,'data_preparation', 'scaling_techniques', 'min_max_scaled_dataset.csv')

# load both scaled dataset
df_min_max_scaled = pd.read_csv(min_max_scaled_data_path)
df_standard_scaled = pd.read_csv(standard_scaled_data_path)


```

Step 3: Determining the optimal number of cluster (elbow method)

At first, we we use elbow method to determine the optimal number of cluster for both min-max and standard scaled datasets.

```
# Function to determine optimal number of clusters using the Elbow Method
def determine_optimal_clusters(df, scaling_label):
    features = df[['tenure', 'MonthlyCharges']]
    wcss = []
    for i in range(1, 11):
        kmeans = KMeans(n_clusters=i, init='k-means++', n_init=10, random_state=42)
        kmeans.fit(features)
        wcss.append(kmeans.inertia_)

    plt.figure(figsize=(10, 6))
    plt.plot(range(1, 11), wcss, marker='o')
    plt.title(f'Elbow Method ({scaling_label})')
    plt.xlabel('Number of Clusters')
    plt.ylabel('WCSS')
    plt.savefig(os.path.join(optimal_clusters_path, f'elbow_method_{scaling_label.lower().replace(" ", "_")}.png'))
    plt.show()

```

Step 4 : Determine the optimal number of cluster (silhouette analysis)

In this step we will use silhouette analysis for both scaled datasets to determine the optimal number of clusters.

```
# Function to determine optimal number of clusters using Silhouette Analysis
def determine_optimal_clusters_with_silhouette(df, scaling_label):
    features = df[['tenure', 'MonthlyCharges']]
    silhouette_scores = []
    for i in range(2, 11):
```

```
        kmeans = KMeans(n_clusters=i, init='k-means++', n_init=10, random_state=42)
        kmeans.fit(features)
        silhouette_scores.append(silhouette_score(features, kmeans.labels_))

    plt.figure(figsize=(10, 6))
    plt.plot(range(2, 11), silhouette_scores, marker='o')
    plt.title(f'Silhouette Analysis ({scaling_label})')
    plt.xlabel('Number of Clusters')
    plt.ylabel('Silhouette Score')
    plt.savefig(os.path.join(optimal_clusters_path, f'silhouette_analysis_{scaling_label.lower().replace(" ", "_")}.png'))
    plt.show()

```

Step 5 : Apply both methods

In this part we will apply elbow method and silhouette analysis to determine the optimal number of clusters.

```
# Apply the Elbow Method and Silhouette Analysis to determine the optimal number of clusters for both datasets
determine_optimal_clusters(df_min_max_scaled, 'Min-Max Scaled')
determine_optimal_clusters(df_standard_scaled, 'Standard Scaled')
determine_optimal_clusters_with_silhouette(df_min_max_scaled, 'Min-Max Scaled')
determine_optimal_clusters_with_silhouette(df_standard_scaled, 'Standard Scaled')

```

Step 6 : Choosing the optimal number of clusters

This steps includes the interpretation of the plot we obtained from both methods to determine the optimal number of clusters

Elbow method: From the both plot elbow appears at 4 clusters, that suggests that addind more clusters offers diminishing results

Silhouette Analysis: From the plot we gained from silhouette analysis suggests that the highest silhouette score occurs at 4 for both scaled datasets.

Conclusion

After analysing both methods for both scaled datasets, we can conclude that the number of clusters for both datasets are 4. The number of clusters we obtained from both methods will help us to train the clusters and provides a robust nature grouping within the data. These information can be used for future analysis and decision-making in the area of customer segmentation.

Next Steps:

After determining the optimal number of cluster, we will now train the clustering model by that determined clusters. That will helps us to interpret the characteristics of each cluster and helps us to generate the visualizations.