D212 Task 1

Kesselly Kamara

011006868

Mya Bass

K-mean Clustering Techniques

2024

Contents

[I. Research Questions 3](#_Toc181824496)

[A. Purpose of data mining 3](#_Toc181824497)

[II. Technique Justification 3](#_Toc181824501)

[B. Choosing Clustering Technique 3](#_Toc181824502)

[III. Data Preparation 3](#_Toc181824504)

[IV. Analysis 6](#_Toc181824505)

[D. Perform the data analysis 6](#_Toc181824506)

[V. Data Summary and Implications 8](#_Toc181824509)

[VI. Demonstration 9](#_Toc181824514)

# Research Questions

## Purpose of data mining

### What are the distinct patient profiles based on demographics, medical conditions, and hospitalization details, and which profiles show a higher likelihood of readmission using **k - mean clustering**?

### To classify patients into distinct groups based on key health metrics to inform more effective health interventions. This question is relevant for hospitals looking to enhance patient care and resource allocation. By clustering patients based on continuous health metrics, the hospital can identify groups with similar health profiles. This information can be used to tailor health programs, preventive measures, or treatment plans aimed at improving outcomes for specific patient segments (Ankita, 2024).

### 

# Technique Justification

## B. Choosing Clustering Technique

### 1. K-means clustering is an unsupervised learning algorithm used to group data. It efficiently groups unlabeled data points into clusters. The K-means clustering algorithm is an example of an exclusionary or "hard" clustering method (Kavlakoglu, 2024). This form of clustering stipulates that a data point can only exist in a single cluster. (Kavlakoglu, 2024). It is an iterative process of assigning each data point to the groups and slowly data points get clustered based on similar features (Dabbura, 2022). Each data point is assigned to the nearest centroid based on the Euclidean distance. This forms k clusters. The centroids of each cluster are recalculated as the mean of all data points assigned to that cluster (Dabbura, 2022). The primary outcome of K-means clustering is the identification of distinct patient groups based on selected features. Each cluster reflects a set of patients with similar characteristics, such as demographic information, medical history, and health metrics (Dabbura, 2022).

2. K-means makes the assumption that every cluster is the same size or has the same variance. In a data set with higher measurement variation in one cluster than another, this assumption would not hold true. For instance, this assumption would be broken in a medical data set if the weight of a person with a severe illness fluctuated more than that of a person with a less severe illness (Winn, Model-Based Machine Learning).

3. **Data Manipulation and Analysis**:

I used python for my analysis:

* pandas: For data manipulation and analysis, especially for handling data in DataFrame format.
* numpy: For numerical operations; arrays and mathematical functions.
* matplotlib: For creating static, interactive, and animated visualizations in Python.
  + Seaborn: A Matplotlib-based statistical data visualization package that offers a sophisticated interface for creating eye-catching visuals.
* scikit-learn: For implementations of k-means clustering and tools for preprocessing data, model evaluation, and more.
* scipy: Useful for functions for clustering analysis and mathematical operations.
* scikit-learn.preprocessing: Contains functions for standardization, normalization, and encoding categorical variables.
* missingno: A library for visualizing missing data, which can help you identify and handle null values in your dataset.

# Data Preparation

1. One data preprocessing goal is featuring scaling. To ensure that each independent variable (feature) contributes equally to the distance computations used in clustering, feature scaling attempts to equalize the range of features. K-means clustering is sensitive to the size of the features as it depends on figuring out the distances between data points and cluster centroids.

2. **Continuous Variables**

* **Initial\_days**: Continuous (measured in days)
* **TotalCharge**: Continuous (measured in currency)

**Categorical Variables**

* **ReAdmis:** Categorical (values indicating readmission status: "Yes" or "No")

3. Data Preparation Steps

* Load the dataset into pandas

import pandas as pd

medical\_clean\_path = r'C:\Users\mbray\OneDrive\Documents\212\medical\_clean.csv'

* Read and name datafile

df = pd.read\_csv(medical\_clean\_path)

* Inspect data types: Check the data types of each column to identify numeric and non-numeric columns

# Check the data types of each column

print(df.dtypes)

* Identify nulls

print(df.isnull().sum())

* Check for duplicates

# Check for duplicates

duplicates = df.duplicated()

# Display the total number of duplicate rows

num\_duplicates = duplicates.sum()

print(f'Total number of duplicate rows: {num\_duplicates}')

if num\_duplicates > 0:

    duplicate\_rows = df[duplicates]

    print(duplicate\_rows)

* Identify outliers

# Function to identify outliers using IQR for specified columns

def find\_outliers\_iqr(df, columns):

    outlier\_indices = []

    for col in columns:

        Q1 = df[col].quantile(0.25)

        Q3 = df[col].quantile(0.75)

        IQR = Q3 - Q1

        lower\_bound = Q1 - 1.5 \* IQR

        upper\_bound = Q3 + 1.5 \* IQR

        # Identify outliers

        outliers = df[(df[col] < lower\_bound) | (df[col] > upper\_bound)]

        outlier\_indices.extend(outliers.index.tolist())

    return set(outlier\_indices)

# Get unique outlier indices

outlier\_indices\_iqr = find\_outliers\_iqr(df, categorical)

print(f'Total number of outliers found using IQR: {len(outlier\_indices\_iqr)}')

* Drop unused columns

dataset = dataset.drop(columns=[‘CaseOrder','Customer\_id','Interaction', 'UID', 'Lat', 'Lng','City', 'Zip','State', 'County', 'Job', 'Marital', 'Gender', 'vitD\_supp', 'Soft\_drink', 'Initial\_admin', 'HighBlood', 'Stroke', 'Complication\_risk', 'Overweight', 'Arthritis', 'Diabetes', 'Hyperlipidemia', 'BackPain', 'Anxiety', 'Allergic\_rhinitis', 'Reflux\_esophagitis', 'Asthma', 'Services', 'Item1', 'Item2', 'Item3', 'Item4', 'Item5', 'Item6', 'Item7', 'Item8'])]

* Feature Selection and Feature Standardizing

# Select continuous variables for clustering

features = dataset[[ 'Initial\_days', 'TotalCharge']]

from sklearn.preprocessing import StandardScaler

# Standardizing the features

scaler = StandardScaler()

scaled\_features = scaler.fit\_transform(features)

# Convert the scaled features back to a DataFrame for easier interpretation

scaled\_features\_df = pd.DataFrame(scaled\_features, columns=features.columns)

# Save the cleaned dataset to a CSV file

dataset.to\_csv('C:/Users/mbray/OneDrive/Documents/212/new\_cleaned\_medical\_data.csv', index=False)

The cleaned data is attached as new\_cleaned\_data.csv

# Analysis

## D. Perform the data analysis

### The optimal number of clusters is k=2. To determine the optimal number of clusters for my dataset I used the Elbow Method. The Elbow Method involves calculating the Within-Cluster Sum of Squares (WCSS) for various values of k, ranging from 1 to 10 clusters. As the number of clusters grows, the within-cluster sum of squares (WCSS) tends to decrease. This is because adding more clusters inherently brings points closer to their respective centroids, thus diminishing the overall distance. In order to effectively use the Elbow Method, it is crucial to pinpoint the moment on the graph known as the "elbow point," where there is a significant slowdown in the rate of decrease. (Ankita, 2024).

A black and white symbol

Description automatically generated

A graph with a blue line

Description automatically generated

### Code used to perform clustering analysis (Ankita, 2024).

# Specify the features for clustering

features = dataset[[ 'Initial\_days', 'TotalCharge']]

# Define the range of k values to test

k\_values = range(1, 11)

sse = []

# Run K-means clustering for each value of k and calculate SSE

for k in k\_values:

    kmeans = KMeans(n\_clusters=k, random\_state=42)

    kmeans.fit(scaled\_features\_df)

    sse.append(kmeans.inertia\_)  # SSE is stored in inertia\_

# Plot the Elbow Curve

plt.figure(figsize=(10, 6))

plt.plot(k\_values, sse, marker='o')

plt.title('Elbow Method for Optimal k')

plt.xlabel('Number of Clusters (k)')

plt.ylabel('Sum of Squared Errors (SSE)')

plt.xticks(k\_values)

plt.grid()

plt.show()

kmeans\_optimal = KMeans(n\_clusters=2, random\_state=42)

cluster\_labels = kmeans\_optimal.fit\_predict(features\_scaled\_df)

# Add the cluster labels to the scaled DataFrame

features\_scaled\_df['Cluster'] = cluster\_labels

# Display the first few rows of the DataFrame with cluster labels

print(features\_scaled\_df.head())

# Optionally, count the number of samples per cluster

cluster\_counts = features\_scaled\_df['Cluster'].value\_counts()

print(f"Cluster counts:\n{cluster\_counts}")

Initial\_days TotalCharge Cluster

0 -0.907310 -0.727185 0

1 -0.734595 -0.513228 0

2 -1.128292 -1.319983 0

3 -1.244503 -1.460517 0

4 -1.261991 -1.467285 0

Cluster counts:

Cluster

0 5002

1 4998

Name: count, dtype: int64

# Analyze the characteristics of each cluster

cluster\_analysis = features\_scaled\_df.groupby('Cluster')[['Initial\_days', 'TotalCharge']].mean()

print("Cluster Characteristics:\n", cluster\_analysis)

Cluster Characteristics:

Initial\_days TotalCharge

Cluster

0 -0.958450 -0.946704

1 0.959217 0.947462

# Randomly sample 1000 rows from the dataset

sampled\_df = features\_scaled\_df.sample(n=1000, random\_state=42)

# Fit KMeans with the optimal k (2)

kmeans\_optimal = KMeans(n\_clusters=2, random\_state=42)

cluster\_labels = kmeans\_optimal.fit\_predict(sampled\_df.drop('Cluster', axis=1))

# Add the cluster labels to the sampled DataFrame

sampled\_df['Cluster'] = cluster\_labels

# Calculate the silhouette score for the optimal k using the sampled DataFrame

silhouette\_avg\_optimal = silhouette\_score(sampled\_df.drop('Cluster', axis=1), cluster\_labels)

print(f'Average Silhouette Score for optimal k: {silhouette\_avg\_optimal:.3f}')

Average Silhouette Score for optimal k: 0.773

# Data Summary and Implications

## E. Data Analysis

### Internal validity and homogeneity verify the quality of the clusters created. The presence of clear differences in total charges and initial days hospitalized suggests that the clustering algorithm has successfully identified distinct groups of patients that may require different management strategies or interventions. The presence of clear differences in total charges and initial days hospitalized suggests that the clustering algorithm has successfully identified distinct groups of patients that may require different management strategies or interventions. I also used the silhouette score. The silhouette score of **0.773** indicates a moderate-high level of separation between the clusters (Ankita, 2024).

### The cluster analysis reveals distinct characteristics between the two groups, with Cluster 0 showing closer-to-average values for certain metrics, while Cluster 1 shows more extreme values. Cluster 0 represents patients who tend to have shorter hospital stays, with an average Initial\_days score of approximately -0.96, indicating that these patients likely require less intensive care or have less severe conditions. Similarly, the TotalCharge value for Cluster 0 is also lower, around -0.95, suggesting that patients in this group incur lower medical charges, which may reflect routine or less complex treatments. On the other hand, Cluster 1 consists of patients with longer hospital stays, as indicated by a Initial\_days score of around 0.96, meaning these patients likely have more serious health conditions requiring extended care. Additionally, the TotalCharge value for this cluster is approximately 0.95, indicating higher charges, possibly due to more complex or resource driven treatments required for these patients. Meanwhile, patients in Cluster 0, with shorter stays and lower charges, could potentially benefit from less intensive management and routine care. Understanding the characteristics of each cluster allows healthcare professionals to make more informed decisions and allocate resources more efficiently. These insights could guide policy decisions aimed at improving care and optimizing the use of hospital resources, ensuring that more severe cases receive the attention they need, while routine cases are managed appropriately to avoid unnecessary resource expenditure.

### One limitation of this clustering analysis is that it doesn't directly incorporate causal relationships between features and readmission. One limitation of this data analysis is that it only considers two continuous variables (TotalCharge and Initial\_days) for clustering. While these features provide valuable insights, they may not fully capture the complexity of patient profiles.

### Establish a follow-up protocol specifically for patients with characteristics aligned with Cluster 1. Increase the number of nutritional support to reduce initial days for patients with higher complication risks and those with longer expected hospital stays. In turn total charge will decrease.

# Demonstration

## F. Panapto link

<https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=c9f762a4-8d8b-4abb-a848-b2210040ae4e>

## G. Web coding source

<https://www.analyticsvidhya.com/blog/2021/05/k-mean-getting-the-optimal-number-of-clusters/>

<https://www.ibm.com/topics/k-means-clustering>

## H. Sources

Ankita. (2024, October 17). *K-means: Getting the optimal number of clusters*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/05/k-mean-getting-the-optimal-number-of-clusters/

Dabbura, I. (2022, September 27). *K-means clustering: Algorithm, applications, evaluation methods, and drawbacks*. Medium. https://towardsdatascience.com/k-means-clustering-algorithm-applications-evaluation-methods-and-drawbacks-aa03e644b48a

Kavlakoglu, E., & Winland, V. (2024, August 27). *What is K-means clustering?*. IBM. https://www.ibm.com/topics/k-means-clustering

Winn, J. (n.d.). Model-Based Machine Learning: 8.4 K-means Clustering . In *K means clustering*. essay. Retrieved November 4, 2024, from https://www.mbmlbook.com/ModelAnalysis\_K-means\_Clustering.html#:~:text=Some%20hidden%20assumptions%20in%20k%2Dmeans&text=For%20example%2C%20because%20the%20variance,clusters%20are%20the%20same%20size.