**Performance Assessment**

OFM4 — OFM4 Task 1: Clustering Techniques

Bader Ale

Department of Information Technology, Western Governors University

D212 Data Mining II

2024

Table of Contents

[Part I: Research Question 3](#_Toc179017286)

[Part II: Technique Justification 3](#_Toc179017287)

[Part III: Data Preparation 4](#_Toc179017288)

[Part IV: Analysis 4](#_Toc179017289)

[Part V: Data Summary and Implications 4](#_Toc179017290)

# Part I: Research Question

The purpose of the analysis using k-means clustering is to answer the following “*In the context of a major hospital chain, "Can we identify distinct patient clusters based on demographic and clinical factors that are more likely to experience hospital readmission within 30 days?".*  For this task, k-means clustering was used on the given medical data set. This unsupervised learning algorithm was used to group patients according to the following common features: Age, Vitamin D levels during hospital stay and Readmission status. These clusters can help medical administrators understand patterns in these clusters that show them a high risk for readmission, which would form a proper basis on which effective targeted management interventions could be based.

# Part II: Technique Justification

The K-means algorithm is a kind of partitioning method where clusters are formed by the similarities of data points, in this case, patients, based on some predefined features such as age, comorbidities, length of stay, and location of discharge. First, the algorithm randomly picks some centroids and then assigns patients to the nearest centroid according to Euclidean distance. The centroids are recalculated when all the patients have been assigned to their respective clusters. The process repeats until the centroids no longer move significantly-when, in other words, the clusters are stable (scikit-learn developers, n.d.).

For this this k-means model, three variables were chosen for the clusters: Age, Vitamin D levels, and the target variable ReAdmission. The purpose of this analysis was to group patients into well-defined clusters that possess similar characteristics, hence allowing an analysis of interactions and influences these factors have on readmission.  
The k-means algorithm represents each patient as a data point in three-dimensional space, whose dimensions are Age, Vitamin D levels, and ReAdmission status. In an iterative process, it assigns each patient to a cluster by computing a distance measure between the data point representing a patient and the centroid of each cluster. It starts with the centroids chosen arbitrarily, but in every subsequent step, the centroid is updated to actually represent an 'mean' value of patients for a cluster. This process is repeated until convergence occurs; the centroids are not changing significantly anymore. We would expect the result of this to be the clear formation of patient clusters that reflect distinct relationships between Age, Vitamin D levels, and ReAdmission. For instance, one cluster may consist of a higher group of aged patients with low Vitamin D levels who are more predisposed to readmission, while another cluster can represent the younger patients who have higher Vitamin D levels and thus will present less likelihood of readmission. The information shall allow the administrators to understand how such factors as age and Vitamin D level result in readmissions to elaborate more focused preventive measures for high-risk groups.

One of the important assumptions of the k-means clustering algorithm is that clusters are spherical and equally sized. What this means is that the algorithm takes on the assumption that data points inside each cluster are evenly distributed around a central point, called a centroid, and the distance between points and their centroid is relatively uniform across all clusters. This works well when the clusters of data are near symmetric. The algorithm can sometimes fail to capture the exact underlying structure of the data set for oddly shaped data or of varying cluster size (scikit-learn developers, n.d.).

The packages used for this assessment include pandas for handling the data structures, seaborn to visualize the data and SKLearn which was used as the main package for the K-means learning algorithm. SKLearn was also used to scale the data before running K-means as well as using the Label Encoder function to encode the ‘ReAdmis’ variable from Yes/No to numerical values.

# 

# Part III: Data Preparation

# One important goal in preparing the data for k-means clustering is to ensure that the variables are on a similar scale. Since k-means calculates distances between data points, variables with larger ranges can disproportionately influence the clustering results. For example, if "Age" values range from 20 to 90, and "Vitamin D levels" vary between 10 and 50, the algorithm may give more weight to Age simply because its numerical range is larger. To address this, we need to standardize the continuous variables, so they all contribute equally to the analysis. This step ensures that the clustering results are not skewed by differences in measurement scales.

For this assessment, three variables were used: TotalCharge, Initial\_days and ReAdmission. TotalCharge is a continuous variable representing the total cost associated with a patient's initial hospital stay, offering insight into the financial aspect of care. Initial Days, another continuous variable, captures the length of the initial hospital stay, providing valuable information about the duration of treatment. The third variable, Readmission, is categorical and indicates whether a patient was readmitted after discharge. To ensure that the k-means clustering algorithm can process this variable effectively, LabelEncoder was used to convert Readmission from categorical (Yes/No) into a continuous numerical format.

To prepare the data for k-means clustering, the first step involved importing the dataset using pandas. This allowed for an initial inspection of the data and identification of relevant features for the analysis. The dataset was loaded with the following code:

1. df = pd.read\_csv('D:\GitHub Repos\WGU\_MSDA\D212\_Data Mining\medical\_clean.csv')

2. df.head()

Once the data was loaded, the next step was data cleaning. This involved removing unnecessary columns, such as customer identifiers and geographic details, to focus on the key variables relevant to clustering. These irrelevant columns were dropped using the following code:

1. df\_kmeans = df.copy()

2. df\_kmeans.drop(['CaseOrder', 'Customer\_id', 'Interaction', 'UID', 'City', 'Zip', 'Lat', 'Lng',

3. 'Population', 'Area', 'TimeZone', 'State', 'County', 'Job', 'Marital',

4. 'Gender', 'Soft\_drink', 'Initial\_admin', 'HighBlood', 'Stroke', 'Complication\_risk',

5. 'Overweight', 'Arthritis', 'Diabetes', 'Hyperlipidemia', 'BackPain', 'Anxiety',

6. 'Allergic\_rhinitis', 'Reflux\_esophagitis', 'Asthma', 'Services', 'Item1', 'Item2'], axis=1, inplace=True)

7. df\_kmeans.head()

8.

Next, the categorical variable ReAdmis (representing whether a patient was readmitted) was converted into a numerical format using LabelEncoder. This step was necessary to include categorical data in the k-means clustering algorithm, which only works with numerical inputs. The encoding was done as follows:

1. from sklearn.preprocessing import LabelEncoder

2.

3. enc = LabelEncoder()

4. df\_kmeans['ReAdmis'] = enc.fit\_transform(df\_kmeans['ReAdmis'])

Following this, the continuous variables like TotalCharge and Initial Days were standardized using StandardScaler. Standardization ensures that each feature contributes equally to the distance calculations used in the k-means algorithm, which is sensitive to scale. The code for this step was:

1. from sklearn.preprocessing import StandardScaler

2.

3. scaler = StandardScaler()

4. df\_means\_columns = df\_kmeans.columns

5. df\_kmeans\_scaled = pd.DataFrame(scaler.fit\_transform(df\_kmeans), columns=df\_means\_columns)

6. df\_kmeans\_scaled.head()

These preprocessing steps—data cleaning, encoding categorical variables, and standardizing continuous variables—ensured the dataset was prepared for effective clustering using the k-means algorithm. A copy of the cleaned dataset is provided for review.

# Part IV: Analysis

# Part V: Data Summary and Implications

Works Cited

scikit-learn developers. (n.d.). *2.3. Clustering*. Retrieved October 2024, from Unsupervised Learning: https://scikit-learn.org/stable/modules/clustering.html#k-means