# Executive Summary– Build and Evaluate Tree Models and Participate in the Multi-Class Prediction of Obesity Risk Kaggle Competition (Late Submission).

By Samuel Mbah Nde – DDS 8555 – Assignment 6

## Project Overview.

This week, I performed implemented hierarchical and k-means clustering models and answered conceptual questions 1 and 7 from chapter 12 of ISLP (James, Witten, Hastie, Tibshirani, & Taylor, 2023). This executive summary is divided into 2 sections. In section 1, I present the solutions to the conceptual questions and in section 2, I implement k-means and hierarchical clustering models using the wines dataset from Kaggle.com (Wang, 2020). The code used for this analysis can be found in my GitHub repository https://github.com/ndesamuelmbah/DDS-8555/tree/main/AssignmentFiles/Week7 (Nde, 2024).

**Section 1: Conceptual Questions.**

**Question 1**: This is question 1 extracted from page 552 of ISLP (James, Witten, Hastie, Tibshirani, & Taylor, 2023) and is displayed in the screenshot below.

A black text on a white background

AI-generated content may be incorrect.  
**Solution**

Equation (12.18) states:

The left-hand side represents the sum of squared pairwise differences between all data points in cluster . We know that When we take the part that sums over the :

But we know that summation over *i* and *i’* covers all pairs symmetrically and we know (by definition of the centroid ) that

So, **(1) simplifying** we get.

With **(3)** because multiplying both sides of **(2)** by :

Applying this, we transform the last term of equation **(3)** to be:

Thus,

And applying the outer sum over the p features, we get.

And this completes the proof.

**(b) Why the K-Means Algorithm Decreases the Objective Function**

The K-Means has the following objective function.

If we substitute the right-hand side of equation **12.18**, then, we can equation **12.17** rewrite it as.

**Step 1: Computing Centroids Minimizes Squared Deviations**

In Step 2(a) of Algorithm 12.2, the centroids are computed as the means of the points in each cluster. The mean minimizes the sum-of-squared deviations, ensuring that for fixed cluster assignments, the objective function (12.17) is minimized.

**Step 2: Reassigning Points Further Reduces the Objective**

In Step 2(b), each observation is reassigned to the closest centroid. This guarantees that every observation is in the best cluster to minimize the squared deviation from the centroid, reducing the total sum of squared deviations.

**Step 3: Monotonic Decrease Until Convergence**

Each iteration of K-means (Steps 2(a) and 2(b)) either decreases or keeps the objective function the same. Since the function is lower-bounded by 0, the process must eventually converge to a local minimum.

**Question 2**: This is question 7 extracted from page 554 of ISLP (James, Witten, Hastie, Tibshirani, & Taylor, 2023) and is displayed in the screenshot below.

A text on a white background

AI-generated content may be incorrect.

**Solution**

I downloaded the data from Kaggle and performed the analysis in python. My goal was to find the proportionality constant or slop of the line that is fit on the correlation distances and the squared pairwise Euclidean distances. As shown in Figure 1 below, the proportion of constant produced a value of 1.13 which supports the hypothesis.

**Figure 1**  
*Proportionality of correlation distances with Euclidean distances.*  
A screenshot of a computer screen

AI-generated content may be incorrect.

**Section 2.** Clustering with Wines Dataset from Kaggle.com(Wang, 2020)**.**I explored the data and confirmed that it does not contain any missing values. Then I computed a correlation matrix shown in Figure 2 below to confirm and quantify some of the visible trends that I saw in a Pairplot of the data before moving on the performed principal component analysis on the dataset.

**Figure 2**  
*Correlation matrix of columns in the wines dataset.*   
A screenshot of a chart

AI-generated content may be incorrect.

In the principal component analysis, I set the number of components to the float value of 0.83 which is slightly higher than the value requested in the exercise so that it will be easy to see the trend in the dataset. Figure 4 below shows the scree plot I obtained from the PCA implementation.  
**Figure 4***Scree plot of the PCA implementation on the dataset.*  
A comparison of a graph

AI-generated content may be incorrect.  
**Note:** *The plot on the left shows the proportion of variation explained (PVE) in each component of the PCA for example more than 35% of variation in the dataset is captured in the first component. The plot on the left is cumulation of the PVE up to the current number of component (on the horizontal - axis). The red line at 0.8 shows happens to coincide at component 5 indicating that using 5 component is enough to capture 80% of the variation in the dataset.*

After determining the number of components needed to capture 80% of the variation in the dataset, I used the resulting dataset to fit clustering models, starting with K-means model. To get the optimal k (number of clusters) from the k-means model, I incremented k from 1 to 11 and plotted the within class sum of squares to get the elbow plot shown in Figure 5 below (Josse & Husson, 2012).

**Figure 5**  
*Using the elbow method of choosing optimal number of clusters (k) for k-means model and a plot of the dendrogram from hierarchical clustering.*  
A graph of a number of clusters

AI-generated content may be incorrect.A diagram of a city

AI-generated content may be incorrect.  
**Note:** *Left: The red line at k=3 indicates the elbow, the point from which increasing k does not lead to significant declines in within class sum of square (WCSS). Right: Dendrogram showing the 3 clusters from a hierarchical clustering model. A cut of the tree anywhere between15 and 25 will result in 3 clusters.*

Figure 6 below shows the predicted clusters for both k-means and hierarchical clustering colored by the predicted cluster.

**Figure 6**  
*Plot of first 2 principal components in the dataset colored by predicted cluster.*

A screenshot of a graph

AI-generated content may be incorrect.

The plot shows that the 2 clustering techniques produced very similar results. A few points did get different clusters. To visualize these, I plotted a confusion table and treated the predictions as though they were true and predicted values from a classification problem as suggested in (Görtler, et al., 2022).

**Figure 7**  
Confusion table and “classification report” comparing results of k-means clustering to hierarchical clustering.

A diagram of a cluster

AI-generated content may be incorrect.A screenshot of a computer screen

AI-generated content may be incorrect.

**Note:** *This confusion table (Left) shows that only 5 of the 178 values were classified as belonging to different classes in both models. The classification report (Right) quantifies this in a more consumable manner. If we were to assume that the true classes were produced by the K-means clustering technique and use a hierarchical clustering technique to attempt to predict the values, then we would get a high accuracy score of 97%.*

Finally, to evaluate the quality of the clusters, I computed the Silhouette score and Cophenetic Correlation Coefficient of the predictions from the k-means model and the hierarchical clustering model and got values of 0.35 and 0.67 respectively. These values indicate that the clustering techniques do indeed fit the data quite well (Batool & Hennig, 2021).

**How I tested/verified model assumptions**[¶](http://localhost:8888/notebooks/JupyterNotebooks/NdeSDDS8555-7.ipynb?#how-to-test-these-assumptions)

1. **PCA Assumption (Linear Relationships & Standardization)**
   * **Check correlations** between features to confirm linear relationships. This was done using the correlation plot in Figure 2.
   * **Use a scree plot** to ensure enough variance is retained. (Figure 4).
2. **K-Means Assumption (Spherical Clusters & Centroid Sensitivity)**
   * **Visualize clusters** in 2D using PCA (Figure 6).
   * **Use silhouette scores** to evaluate clustering quality.
   * **Run K-Means multiple times** with different centroid initializations (was done with *init* parameter set to 20 to ensure that the algorithm runs multiple times and returns the best results.
3. **Hierarchical Clustering Assumption (Nested Structure)**
   * **Use cophenetic correlation coefficient** to measure how well the clustering preserves pairwise distances.

# Bibliography

Batool, F., & Hennig, C. (2021). Clustering with the Average Silhouette Width. *Computational Statistics & Data Analysis, 158*, 107-190. doi:10.1016/j.csda.2021.107190

Görtler, J. a., Wongsuphasawat, K., Ren, D., Nair, R., Kirchner, M., & Patel, K. (2022). Neo: Generalizing Confusion Matrix Visualization to Hierarchical and Multi-Output Labels., (pp. 1-13). doi:10.1145/3491102.3501823

James, G., Witten, D., Hastie, T., Tibshirani, R., & Taylor, J. (2023). *An Introduction to Statistical Learning: with Applications in Python (1st ed.).* Springer Nature.

Josse, J., & Husson, F. (2012). Selecting the number of components in principal component analysis using cross-validation approximations. *Computational Statistics & Data Analysis, 56*(6), 1869-1879. doi:10.1016/j.csda.2011.11.012

Nde, S. M. (2024, 02). *DDS-8555*. Retrieved from Github: https://github.com/ndesamuelmbah/DDS-8555.git

Reade, W., & Chow, A. (2024). *Multi-Class Prediction of Obesity Risk*. Retrieved from Kaggle: https://kaggle.com/competitions/playground-series-s4e2

Wang, H. (2020). *Wine Dataset for Clustering*. Retrieved from Kaggle: https://www.kaggle.com/datasets/harrywang/wine-dataset-for-clustering