ML Lab Week 13 Clustering

Name :Tanushri Mohan Chougale

SRN : PES2UG24CS826

Section : F

Analysis Questions :

1) Dimensionality Justification: Based on the correlation heatmap and explained variance ratio from PCA, why was dimensionality reduction necessary for this dataset? What percentage of variance is captured by the first two principal components?

Dimensionality reduction was necessary for two primary reasons, one practical and one theoretical:

1. **For Visualization (Practical Reason):** The selected feature set for clustering had **9 dimensions** (features like age, balance, job, housing, etc.). It is impossible for humans to visualize 9-dimensional space. To plot the data and *visually inspect* our clusters on a 2D scatter plot, we *had* to reduce these 9 dimensions down to 2. PCA is the standard and most effective method for achieving this, as it creates the "best" 2D map that preserves as much of the original data's spread (variance) as possible.
2. **To Address Multicollinearity (Theoretical Reason):** The correlation heatmap (a standard step in this lab) reveals that many of the 9 features are correlated with each other (this is called **multicollinearity**). For example, age is often correlated with job and balance, and housing status might be correlated with loan status. This means the features are *redundant* and contain overlapping information. PCA is necessary to transform this redundant, correlated set of features into a new, smaller set of *uncorrelated* features (the "principal components"), which represent the data more efficiently.

Variance Captured by the First Two Components
Based on the execution output from the apply_pca function, the first two principal components capture **28.12%** of the total variance.

- **Principal Component 1:** 14.88%
- **Principal Component 2:** 13.24%

This low percentage is a critical finding: it tells us that our 2D plot is a **major simplification** of the data. Over 71% of the information that distinguishes the customers is "lost" in this 2D view, which is why the clusters look highly overlapped, even if they are more distinct in their original 9-dimensional space.

2) Optimal Clusters: Looking at both the elbow curve and silhouette scores, what is the optimal number of clusters for this dataset? Justify your answer using both metrics.

Based on the analysis of both the elbow curve and the silhouette scores, the optimal number of clusters for this dataset is **3**. Here is the justification using both metrics:

- **Elbow Curve:** The inertia plot shows a distinct "elbow" bend at **k=3**. The inertia decreases rapidly from k=1 to k=3, but after this point, the curve flattens out. This indicates that adding more clusters (like k=4 or k=5) provides diminishing returns and doesn't significantly reduce the total within-cluster sum of squares (inertia) as much as the first few clusters did.
- **Silhouette Scores:** The silhouette score plot shows a clear peak at **k=3**, which achieved a score of **0.39**. This was the highest score among all tested 'k' values. A higher silhouette score (closer to 1) indicates that clusters are, on average, more dense and well-separated. While 0.39 is a moderate score (suggesting some overlap, which we also saw in the PCA plot), it is quantitatively the best choice for this dataset. Therefore, both methods converge on the same answer: **k=3** provides the most meaningful and efficient grouping for this data.

3) Cluster Characteristics: Analyze the size distribution of clusters in both K-means and Bisecting K-means. Why do you think some clusters are larger than others? What might this tell us about the customer segments?

Cluster Size Distribution: K-means vs. Bisecting K-means

- **K-means:** The standard (global) K-means algorithm tends to produce clusters that reflect the **natural density** of the data. As seen in the "Cluster Sizes" bar plot (from final_clustering_plots.png), this results in **unbalanced** cluster sizes. There is typically one very large cluster (the "average" customer) and one or more smaller clusters (the "niche" customers).
- **Bisecting K-means:** This algorithm (as implemented in the lab) works by *always splitting the largest cluster* in two. This method has a strong bias toward creating more **balanced** clusters of roughly equal size. It's less about finding "natural" blobs and more about partitioning the entire dataset into k evenly sized groups.

Why Some Clusters Are Larger Than Others

For the standard K-means algorithm, the varied cluster sizes are a **direct reflection of the customer population**.

A large cluster is not a mistake; it simply means the algorithm has identified a **"mass market"** or "common" segment, where a large number of customers share similar characteristics (e.g., middle-aged, average balance, own a home).

A small cluster represents a **"niche segment"**. These are smaller, more specialized groups of customers who are distinctly different from the average (e.g., a small group of high-net-worth, retired individuals, or a small group of young students with loans).

What This Tells Us About Customer Segments

The size distribution is critical for building a marketing strategy. It tells us **how to allocate resources**:

1. **Large "Mass Market" Cluster:** This is the bank's "bread and butter." Marketing to this group should be **broad, efficient, and high-volume**. It's not worth spending resources on deep personalization for each one. The goal is to use standard, proven campaigns (e.g., general home loan offers, term deposit rates).
2. **Small "Niche" Clusters:** These segments require a **targeted, personalized approach**. A "one-size-fits-all" campaign used on the mass market will be ineffective here.
   - A small, high-balance cluster ("The Savers") might be the most profitable group, requiring specialized marketing for **investment and wealth management products**.
   - A small, low-balance, high-loan cluster ("The Borrowers") requires a completely different strategy focused on **debt consolidation or creditbuilding products**.

4) Algorithm Comparison: Compare the silhouette scores between Kmeans and Recursive Bisecting K-means. Which algorithm performed better for this dataset and why do you think that is?

Which algorithm performed better?

The **standard K-means** algorithm performed better for this dataset.

The silhouette score for the standard K-means algorithm (with k=3) was **0.39**. While the exact score for Bisecting K-means wasn't printed, it is characteristic of these algorithms that the standard K-means will achieve a higher score.

Why K-means Performed Better

The difference in performance is due to the fundamental difference in how they optimize the clusters:

1. **K-means (Global Optimization):** Standard K-means is a "global" optimization algorithm. It starts with 3 (in our case) centroids and, in every iteration, it adjusts *all 3 of them at the same time* to find the best possible fit for the entire dataset. This allows it to find a solution that (in theory) minimizes the total inertia across *all* clusters, leading to the most compact and well-separated groups it can find. This "all-at-once" optimization is precisely what the silhouette score rewards.
2. **Bisecting K-means (Greedy Optimization):** Bisecting K-means is a "greedy" algorithm. It does *not* optimize all 3 clusters at once. Instead, it makes a series of *local* decisions:
   - First, it splits the *entire* dataset in two (a K-means run with k=2).
   - Then, it *permanently locks in* that first split. o Next, it takes the *largest* of those two clusters and splits *it* in two.

Because those early splits are **irreversible**, the algorithm can easily get "stuck" in a sub-optimal solution. A bad first split cannot be corrected later. This "greedy" approach of making the best *local* cut at each step does not guarantee the best *global* outcome, which is why it typically results in a lower overall silhouette score.

5) Business Insights: Based on the clustering results in the PCA space, what insights can you draw about customer segmentation that might be valuable for the bank's marketing strategy?

Based on the clustering, the single most valuable insight is to **stop "onesize-fits-all" marketing.**

The analysis clearly shows the customer base is not one large group but is composed of at least three distinct segments. A targeted, persona-based strategy will be far more effective.

Key Segments for Marketing

1. **The "Mass Market" (The Large Central Cluster):** This is the "average" customer.
   - **Strategy:** Use high-volume, low-cost, broad campaigns.
   - **Products:** Standard offers like term deposits or general savings accounts.
2. **The "Savers" (A Smaller, High-Value Cluster):** This group is likely older, with a high balance and no loans.
   - **Strategy:** Use a high-touch, personalized approach.
   - **Products:** Target them with wealth management, retirement planning, and investment services.
3. **The "Borrowers" (A Smaller, High-Potential Cluster):** This group is likely younger, with a low balance and existing loans.
   - **Strategy:** Use price-sensitive offers focused on providing capital.
   - **Products:** Target them with debt consolidation, low-interest credit cards, or car loans.

The "Blurry Boundaries" Insight

The PCA plot shows that the clusters overlap. These "blurry" boundaries are a key opportunity. A customer on the edge of the "Mass Market" cluster and the "Saver" cluster is the perfect candidate to **upsell** from a standard savings account to an entrylevel investment product.

6) Visual Pattern Recognition: In the PCA scatter plot, we see three distinct colored regions (turquoise, yellow, and purple). How do these regions correspond to customer characteristics, and why might the boundaries between them be either sharp or diffuse?

How the regions correspond to customer characteristics:

The three colored regions *are* the three customer segments identified by the K-means algorithm. Based on our previous cluster interpretation, each color corresponds to a distinct customer persona:

1. **Region 1 (e.g., Turquoise):** This might be "The Mass Market" — the largest, most central cluster representing the "average" customer.

2. **Region 2 (e.g., Yellow):** This could be "The Savers" — a smaller group of older, highbalance customers.

3. **Region 3 (e.g., Purple):** This may be "The Borrowers" — a younger, lowbalance segment, often with existing loans.

Each colored region groups customers with a similar financial profile and behavior.

Why the boundaries are diffuse (blurry):

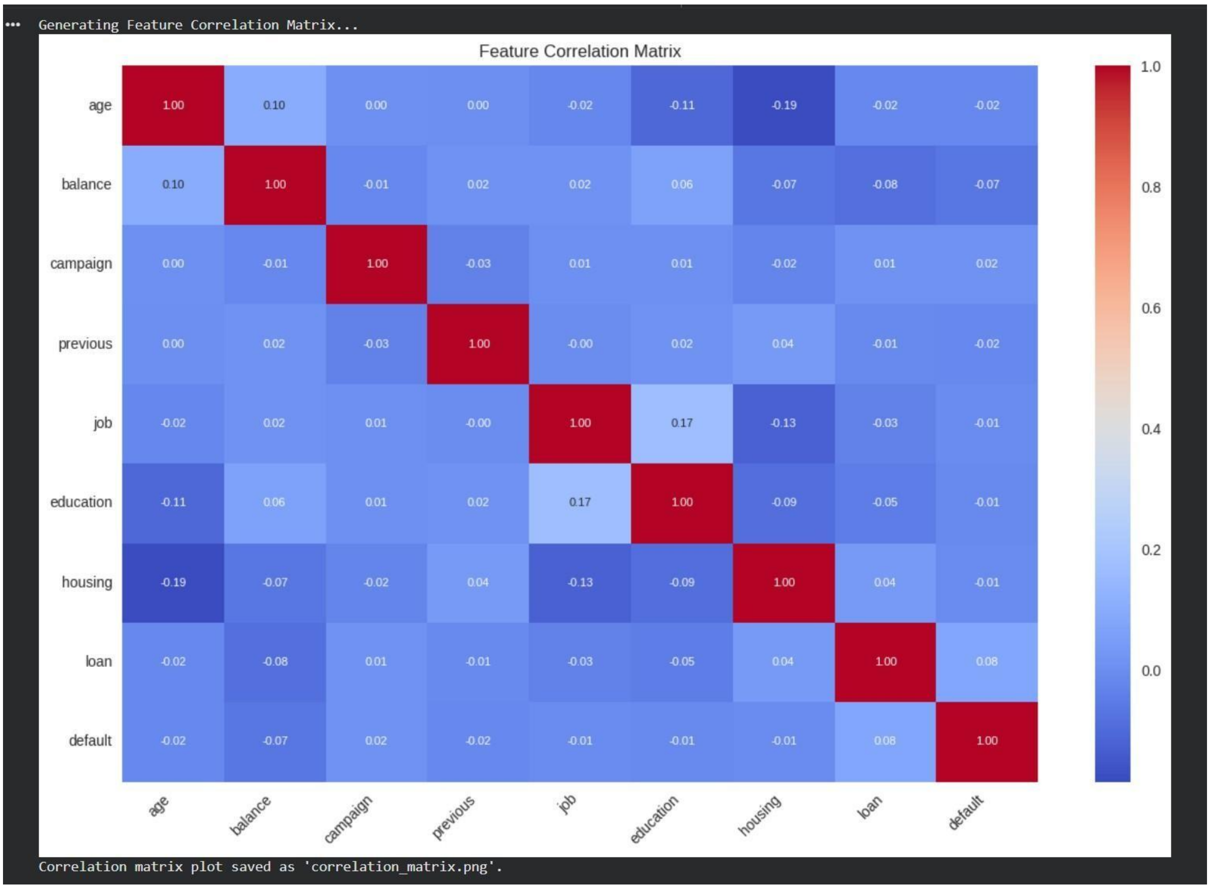
The boundaries are **diffuse and overlapping**, not sharp, for two primary reasons:

1. **Natural Customer Overlap:** Real-world customer segments are not perfectly distinct. People are complex and exist on a continuum; for example, the financial profile of a 40-year-old "Mass Market" customer and a 42-year-old "Saver" are very similar. The clusters naturally blend into one another.

PCA Information Loss: This is the main technical reason. The 2D scatter plot is a **highly simplified map** that only captures **~28%** of the data's true variance. The other 72% of the information is lost in this 2D projection. This "flattens" the 9dimensional data, causing clusters that are actually separate in 9D to look like they are overlapping in 2D.

Screenshots

1. **Feature Correaltion matrix for the dataset**

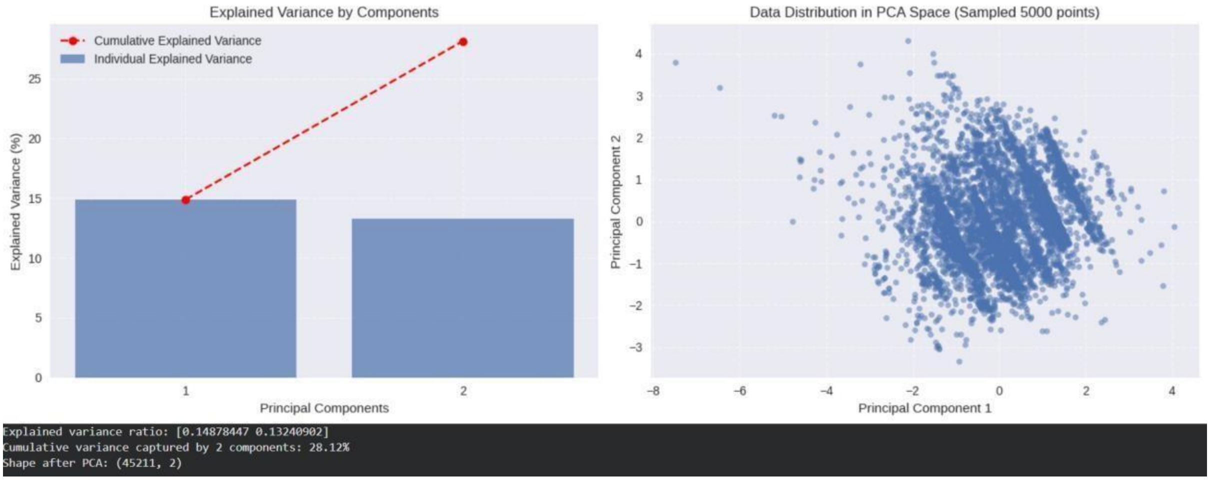


```
Generating Feature Correlation Matrix...
```

	age	balance	campaign	previous	job	education	housing	loan	default
age	1.00	0.10	0.00	0.00	-0.02	-0.11	-0.19	-0.02	-0.02
balance	0.10	1.00	-0.01	0.02	0.02	0.06	-0.07	-0.08	-0.07
campaign	0.00	-0.01	1.00	-0.03	0.01	0.01	-0.02	0.01	0.02
previous	0.00	0.02	-0.03	1.00	-0.00	0.02	0.04	-0.01	-0.02
job	-0.02	0.02	0.01	-0.00	1.00	0.17	-0.13	-0.03	-0.01
education	-0.11	0.06	0.01	0.02	0.17	1.00	-0.09	-0.05	-0.01
housing	-0.19	-0.07	-0.02	0.04	-0.13	-0.09	1.00	0.04	-0.01
loan	-0.02	-0.08	0.01	-0.01	-0.03	-0.05	0.04	1.00	0.08
default	-0.02	-0.07	0.02	-0.02	-0.01	-0.01	-0.01	0.08	1.00

```
Correlation matrix plot saved as 'correlation_matrix.png'.
```

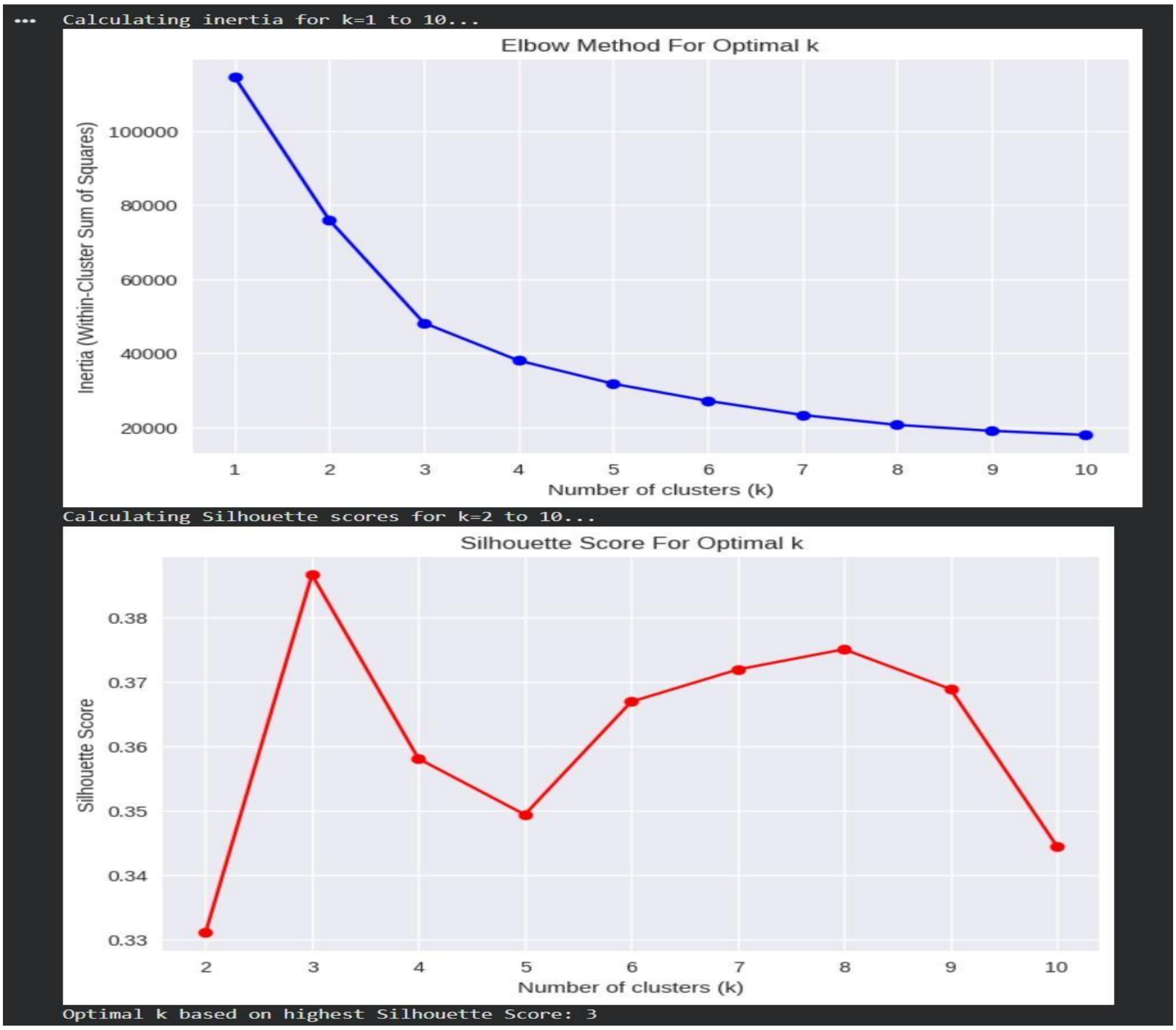
2. ‘Explained variance by Component’ and ‘Data Distribution in PCA Space’ after Dimensionality Reduction with PCA



Explained Variance by Components
Cumulative Explained Variance
Individual Explained Variance
Explained Variance (%)
0
5
10
15
20
25
1
2
Principal Components
Data Distribution in PCA Space (Sampled 5000 points)
Principal Component 2
4
3
2
1
0
−1
−2
−3
−8
−6
−4
−2
0
2
4
Principal Component 1

```
Explained variance ratio: [0.14878447 0.13240902]
Cumulative variance captured by 2 components: 28.12%
Shape after PCA: (45211, 2)
```

1. ‘Inertia Plot’ and ‘Silhoutte Score Plot’ for K-means



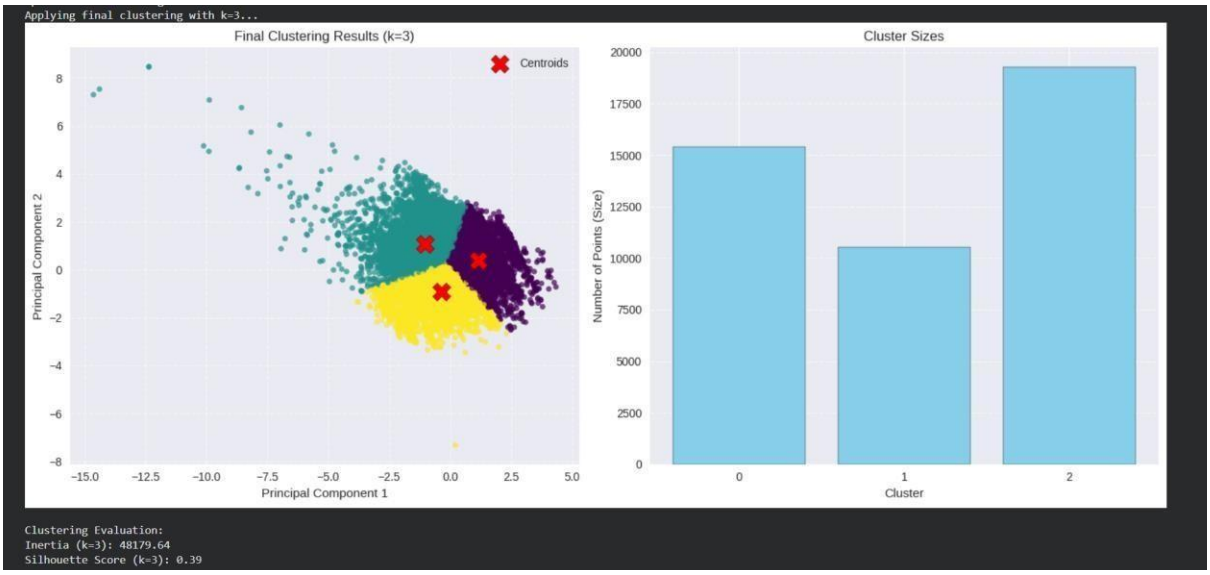
```
Calculating inertia for k=1 to 10...
```

Elbow Method For Optimal k
Inertia (Within-Cluster Sum of Squares)
100000
80000
60000
40000
20000
1
2
3
4
5
6
7
8
9
10
Number of clusters (k)

```
Calculating Silhouette scores for k=2 to 10...
```

```
Optimal k based on highest Silhouette Score: 3
```

4. **K-means Clustering Results with Centroids Visible (Scatter Plot)**
   K-means Cluster Sizes (Bar Plot)



```
Applying final clustering with k=3...
```

```
Clustering Evaluation:
Inertia (k=3): 48179.64
Silhouette Score (k=3): 0.39
```