Banking Customers Segmentation using K-means Clustering

Adnane Deroui
Master in Data Science and Business Analytics
International University of Rabat
Rabat, Morocco

***Abstract*—Customer Segmentation is a vital approach for banks to tailor their products and services to specific customer needs. Using K-means clustering, customers are grouped based on features such as age, balance, credit history, and loan status. This clustering method enables the identification of key segments, such as high-balance customers who are suitable for premium services, credit card offers, and personal loans, as well as customers with a history of defaults, who may benefit from debt management programs. By leveraging these clusters, banks can enhance customer targeting, delivering personalized banking products and services more effectively**

I. Introduction

Banks play a pivotal role in the economic framework of any nation, not only by facilitating the flow of funds between borrowers and lenders but also by contributing to the stability and growth of the economy through investments and regulatory influence. Beyond their domestic responsibilities, banks are key players in international trade, foreign investments, and maintaining global financial stability.

As financial institutions face rapid technological advancements, the challenge of offering personalized services while ensuring security has become increasingly important. Customer segmentation, particularly through the use of advanced techniques like K-means clustering, allows banks to target customers more effectively by grouping them based on shared characteristics such as balance, credit history, and loan needs. This segmentation enhances the ability of banks to offer tailored financial products, from premium services and personal loans to debt management programs, thereby improving customer satisfaction and loyalty.

In this context, K-means clustering serves as a powerful tool for identifying diverse customer groups and optimizing the marketing of banking products and services. By leveraging data analytics, banks can not only increase operational efficiency but also align their offerings with customer needs, fostering a more secure and personalized banking experience.

II. Literature Review

A. What is Machine Learning?

The study of computer algorithms that provide systems the capacity to learn from experience automatically is known as machine learning. Machine learning algorithms allow computers to make decisions independently without outside assistance. Such choices can be made by identifying significant underlying patterns in complex data. [1] Supervised, unsupervised, and reinforcement learning are the different kinds of machine learning algorithms. A few hybrid strategies and other widely used techniques provide organic expansion of machinelearning issue types.

B. Unsupervised Learning

In unsupervised learning, the model is trained on unlabeled data, meaning that the algorithm must find patterns and structure without any predefined categories or outputs. Unlike supervised learning, where the model learns from tagged data, unsupervised learning is focused on discovering the underlying structure in the data through methods such as clustering or dimensionality reduction. The algorithm identifies relationships within the input data and groups similar instances together, often used in applications like customer segmentation, market analysis, and anomaly detection. Popular algorithms for unsupervised learning include K-means clustering, hierarchical clustering, and principal component analysis (PCA) [2]. Unsupervised learning is crucial in situations where labeling data is costly or impractical, enabling models to derive valuable insights from raw datasets [3].

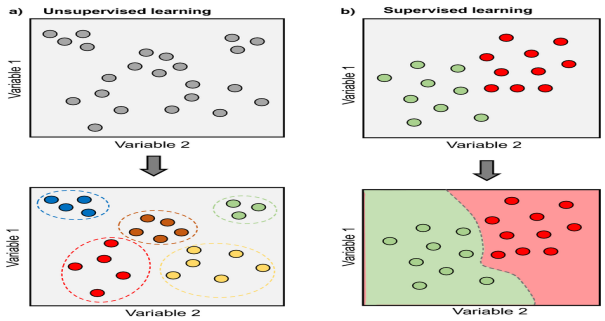

Fig. 1. Supervised and Unsupervised Learning

C. K-means Algorithm

The K-means algorithm is one of the most widely used clustering techniques in unsupervised learning due to its simplicity and efficiency. It works by partitioning a dataset into *K clusters*, where each cluster is represented by its *centroid*. The algorithm operates iteratively, beginning with the *initialization* phase, where K centroids are randomly selected. This is followed by the *assignment step*, where each data point is assigned to the closest centroid based on a distance metric, usually Euclidean distance. After the assignment, the centroids are recalculated as the *mean of all the points* in the respective clusters (*the update step*) [4].

Euclidian distance between point $P(p_1, p_2)$ *and* $Q(q_1, q_2)$:

$$d(P,Q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2}$$

The process repeats until the centroids stabilize, meaning that the assignments no longer change significantly, or a stopping criterion, such as minimal change in the error function, is met [5]. This error function measures the sum of squared distances between points and their assigned centroids, and the algorithm halts when the improvement between iterations is below a pre-defined threshold.

Despite its efficiency, K-means can be sensitive to the initial placement of centroids, which may lead to suboptimal clustering, and its performance can degrade with high-dimensional data. Nevertheless, variants such as K-means++ address some of these limitations by improving the initialization step [6]

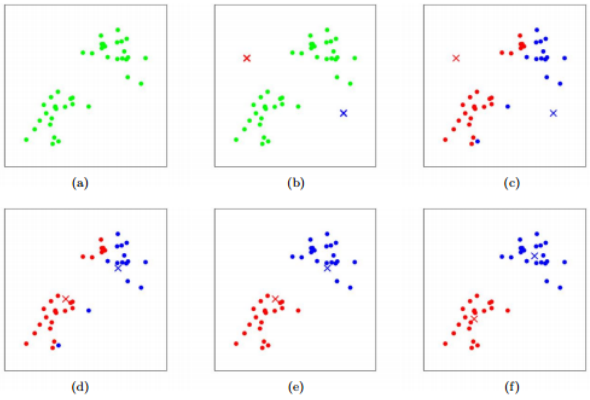

Fig. 2. K-means iterative process

D. Finding Optimal Number of Clusters

Finding the optimal number of clusters, denoted as *K*, is a crucial step in the K-means clustering process. An improper choice of K can lead to underfitting or overfitting, affecting the quality of the resulting clusters. One widely used technique to determine the optimal K is the *Elbow Method*, which involves plotting the *sum of squared errors (SSE)* for different values of K and identifying the "elbow point" where the reduction in SSE becomes marginal. This indicates the point at which adding more clusters does not significantly improve the model's performance [7].

$$\text{Distortion or Inertia} = \sum_{i=1}^{N} (x_i - C_k)^2$$

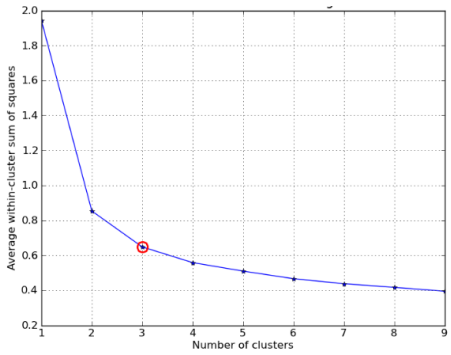

Fig. 3. Overview of Elbow Method

Another popular approach is the *Silhouette Score*, which measures how similar a data point is to its own cluster compared to others. A higher silhouette score indicates better-defined clusters, and this metric can be used to evaluate various values of K to find the optimal number [8].

$$\text{Silhouette score} = s(i) = \frac{b_i - a_i}{max\{a_i, b_i\}}$$

a: *average intracluster distance*

b: *average nearestcluster distance*

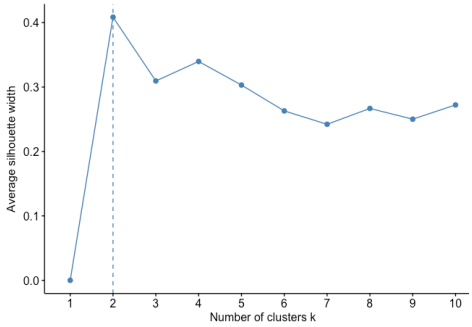

Fig. 4. Overview of The Silhouette Method

Additionally, *Gap Statistics* compares the within-cluster dispersion for different values of K against a null reference distribution of the data, helping to identify the best clustering structure [9]. In high-dimensional datasets, *Principal Component Analysis (PCA)* is often applied to reduce the dimensionality before clustering, helping mitigate noise and irrelevant features that could skew the determination of *K* [10]. Regardless of the method, finding the optimal number of clusters is crucial for ensuring meaningful segmentation and improving the overall effectiveness of the K-means algorithm.

III. Methodology

A. Problem Statement

The primary problem this study aims to solve is the effective segmentation of banking customers to enhance the targeting and personalization of financial products and services. Banks typically manage large, diverse customer bases with varying needs, behaviors, and financial backgrounds. Without proper segmentation, offering the right products to the right customers becomes a challenge, potentially leading to inefficient marketing, lower customer satisfaction, and missed revenue opportunities.

B. Importing Libraries

Python offers a wide range of libraries that make data analysis, visualization, and machine learning more accessible and efficient. Below are the libraries used in this project

Pandas	*Python library for data manipulation and analysis*
Numpy	*Python library for linear algebra using arrays and matrices*
Matplotlib	*Python library for basic data visualizations*
Seaborn	*Python library for advanced data visualizations*
Scipy	*Python library for scientific and technical computing*
Scikit-learn	*Python library for machine learning algorithms*

Table 1. Libraries Used for Customer Clustering

C. Overview of the Banking Customer Dataset

The dataset contains comprehensive records of customer profiles, capturing a variety of attributes related to demographics, financial behavior, and product engagement. It includes information on customers from different geographic regions and job sectors, with varying credit scores, account balances, and engagement levels with banking products. By analyzing patterns in customer behavior and financial habits, the dataset enables deeper insights into customer needs, allowing for more targeted marketing strategies and personalized banking services. The dataset comprises *10,000 rows* and *16 columns*, capturing diverse aspects of customer profiles and reflecting a significant volume of data for analysis.

```
data.dtypes
```

```
Credit Score         int64
Geography           object
Gender              object
Age                  int64
Fidelity             int64
Balance            float64
Num of Products      int64
Salary             float64
Job                 object
Marital             object
Education           object
Credit Default      object
Housing Loan        object
Personal Loan       object
Active              object
Credit Card         object
dtype: object
```

Fig. 5. Checking Data Types

This blend of data types offers a rich foundation for segmentation through clustering helping to uncover patterns in customer behavior and preferences.

D. Exploratory Data Analysis

In this section, we perform an Exploratory Data Analysis (EDA) to better understand the structure and key characteristics of the customer dataset. EDA is essential for identifying trends, spotting anomalies, and providing a general overview of the data before diving into any machine learning or clustering algorithms.

1) Statistical Summary for Numerical Variables

```
data[['Credit Score','Age',"Fidelity","Balance","Num of Products","Salary"]].describe()
```

	Credit Score	Age	Fidelity	Balance	Num of Products	Salary
count	10000.00000	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000
mean	650.50980	38.924900	5.013100	76472.875009	1.530300	100102.655508
std	96.64416	10.489159	2.892225	62399.784768	0.581649	57510.530054
min	350.00000	18.000000	0.000000	0.000000	1.000000	11.580000
25%	584.00000	32.000000	3.000000	0.000000	1.000000	51014.837500
50%	652.00000	37.000000	5.000000	97173.290000	1.000000	100218.210000
75%	717.25000	44.000000	7.000000	127639.372500	2.000000	149400.107500
max	850.00000	92.000000	10.000000	250898.090000	4.000000	199992.480000

Fig. 6. Statistical Summary

This statistical summary provides a comprehensive snapshot of the dataset's key numerical features related to customer profiles. The average *Credit Score* of approximately 650 with a standard deviation of 96.64 indicates a varied range of creditworthiness among customers, spanning from 350 to 850. The average *Age* of approximately 38.92 years with a standard deviation of 10.49 highlights a diverse demographic, ranging from 18 to 92 years old. *Fidelity*, likely representing customer tenure, shows an average of about 5.01 years with a standard deviation of 2.89, suggesting moderate longevity with the bank. *Balance* varies significantly with an average of 76,472.88 units and a standard deviation of 62,399.78, ranging from 0 to 250,898.09 units, indicating a wide spectrum of financial engagement. The average *Number of Products* per customer is 1.53, ranging from 1 to 4, indicating varying levels of banking product usage. Finally, the average *Salary* of approximately 100,102.66 units with a standard deviation of 57,510.53 reflects diverse income levels across the customer base. These insights provide a foundational understanding of customer demographics and financial behaviors, essential for further analysis and segmentation in banking and financial services.

2) *Distribution of Customer Balances by Geography*

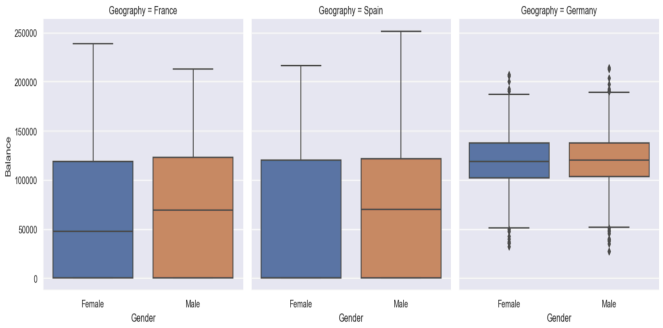

Fig. 7. Boxplot Representation of Customer Balances

The median customer balance varies significantly by both gender and geography. For female customers, the median balance is notably higher in Germany at 118907.60€ compared to France at 47450.43€, and is notably lower in Spain at €0.00, suggesting a possible absence of female customers or inactive accounts in Spain.

Among male customers, Germany again shows the highest median balance at 120120.49€, surpassing France at 69278.68€ and Spain at 69857.01€. These figures highlight a regional disparity in customer balances and may indicate differing financial behaviors or economic conditions across these countries.

3) *Distribution of Customers by Credit Default*

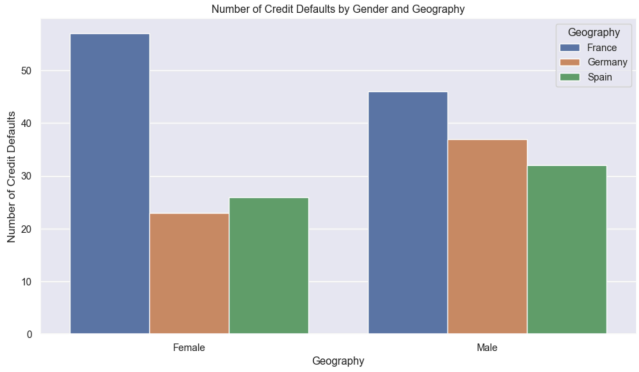

Fig. 8. Credit Default by Gender and Country

The analysis of credit defaults by geography and gender reveals interesting patterns. In France, the number of female customers who defaulted on credit is slightly higher *(57)* than their male counterparts *(46)*. Conversely, in Germany, more males *(37)* defaulted compared to females *(23)*, indicating a gender-based variation in credit default behavior. Spain presents a more balanced distribution, with *26* female defaulters and *32* male defaulters. These findings highlight potential geographic and gender-based differences in financial behavior and credit risk, suggesting that both factors may play a role in determining default rates across different regions.

4) *Distribution of Defaulted Customers by Job*

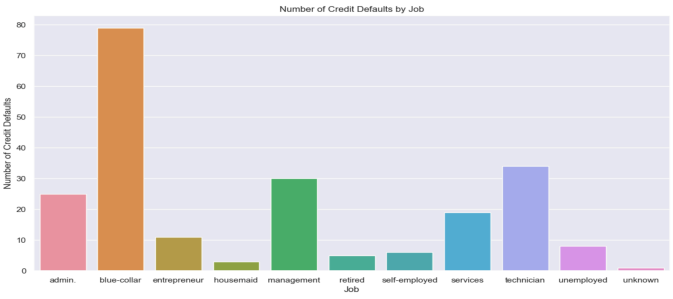

Fig. 9. Credit Default by Type of Job

The distribution of credit defaults across different job categories reveals notable trends. Blue-collar workers have the highest number of defaults, with *79* instances, suggesting a potential correlation between this job type and financial vulnerability. Technicians *(34)* and individuals in management positions *(30)* also show a significant number of defaults, reflecting that even higher-skilled or more stable professions are not immune to credit risk. Administrative roles *(25)* and service workers *(19)* follow, while entrepreneurs *(11),* retired individuals *(5),* and the self-employed *(6)* exhibit lower default counts. The lowest figures are found among housemaids *(3),* unemployed individuals *(8),* and those with unknown occupations *(1),* indicating varying levels of financial stress across job sectors. These results suggest that occupation plays a key role in predicting credit default risk.

5) *Correlation Matrix*

The correlation matrix reveals weak relationships between the variables, indicating that most features have little to no linear correlation with each other. The strongest correlation is a negative one between the number of products and balance *(-0.304)*, suggesting that customers with more products tend to have lower balances. Other variables, such as credit score, age, fidelity, and salary, show minimal correlations with each other or with other features, indicating that these factors may be relatively independent in the dataset. Overall, the matrix suggests no strong multicollinearity, which means the features are not strongly related and can contribute independently to modeling efforts.

Fig. 10. Correlation Matrix of Numerical Variables

E. *Modeling*

In our approach to customer segmentation, we utilized *K-means clustering* to group banking clients based on similar financial and demographic characteristics. This technique is well-suited for our dataset, as it allows the bank to uncover natural groupings of customers, which can then be targeted with personalized products and services. Before applying the K-means algorithm, we handled the categorical variables in our dataset using Label Encoding Since the dataset includes non-numeric fields such as *Geography, Gender, Job, Marital status, and Education,* it was essential to convert these categories into numerical values to be compatible with the algorithm.

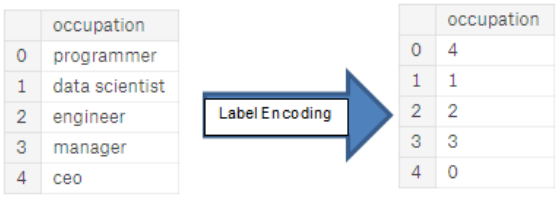

Fig. 11. Label Encoding Process

Label encoding assigned a unique integer to each category, enabling us to effectively incorporate these categorical features into the clustering process. By preparing the data in this way, we ensured that the K-means algorithm could analyze both the numeric and categorical aspects of customer profiles, leading to more accurate and meaningful segmentation.

```
categorical_var = ["Geography","Gender","Job","Marital","Education",
          "Credit Default","Housing Loan","Personal Loan","Active","Credit Card"]

from sklearn import preprocessing
label_encoder = preprocessing.LabelEncoder()
for i in cat_var:
    data[i] = label_encoder.fit_transform(data[i])
```

Fig. 12. Label Encoding with Sklearn and Python

In order to determine the optimal number of clusters, we iterated through a range of cluster values from 1 to 30. For each number of clusters, we calculated two key metrics: *Distortion and Silhouette score.*

Distortion measures the sum of the squared distances between each data point and its closest cluster center, helping us assess the compactness of clusters. As we increased the number of clusters, we observed a decrease in distortion, reflecting tighter clusters. However, after a certain point, the reduction becomes marginal, signaling the potential optimal number of clusters.

Additionally, we calculated the *Silhouette score*, which measures how similar an object is to its own cluster compared to other clusters. A higher silhouette score indicates better-defined clusters. By combining the evaluation of both distortion and silhouette score, we can effectively choose the optimal number of clusters that balance compactness and separation.

This approach allowed us to identify the best cluster configuration, which is crucial for accurate segmentation and, subsequently, for developing tailored banking products and services for different customer groups.

```
from sklearn.cluster import KMeans
from sklearn import metrics
from sklearn.metrics import silhouette_score

distortions = []
silhouette = []

for k in range(1, 31):
    if k == 1 :
        model = KMeans(n_clusters=k).fit(data)
        model.fit(data)
        model.inertia_
        distortions.append(model.inertia_)
    elif k > 1 :
        model = KMeans(n_clusters=k).fit(data)
        model.fit(data)
        model.inertia_
        distortions.append(model.inertia_)
        silhouette.append(silhouette_score(data, model.fit_predict(data)))
```

Fig. 13. Training process for Different Clusters

The distortion scores generated for the K-means clustering model reflect the sum of squared distances between each data point and its closest cluster center across different numbers of clusters, ranging from 1 to 30. To determine the optimal number of clusters, we can apply the elbow method, which involves plotting the distortion scores against the number of clusters and identifying the point where the rate of decline sharply decreases, resembling an "elbow."

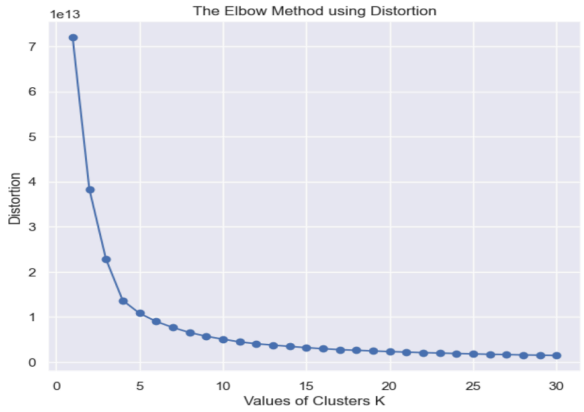

Fig. 14. Distortion Representation for Different Clusters

From the provided distortion scores, we observe a steep reduction in distortion as the number of clusters increases from 1 to around 5, after which the rate of reduction slows down significantly. This indicates that: *after around 5 clusters, adding more clusters does not substantially improve the compactness of the clusters*. This point of diminishing returns marks the "elbow," suggesting that 5 clusters might be the optimal number for segmenting the customers.

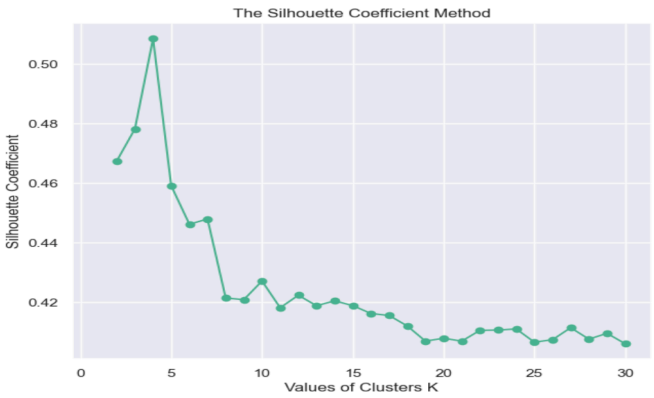

Fig. 15. Average Silhouette Score for Different Clusters

The silhouette scores provide a measure of how well each point fits within its assigned cluster compared to other clusters. Scores close to 1 indicate that data points are well-matched to their own cluster and poorly matched to neighboring clusters, while scores near 0 suggest overlapping or poorly defined clusters. We observe a peak at *4 clusters with a score of 0.508*, which represents the highest level of cluster separation and coherence. After this point, the silhouette scores generally decline or fluctuate with additional clusters, indicating diminishing returns in cluster quality as the number of clusters increases. This suggests that 4 clusters may be the optimal number for our segmentation.

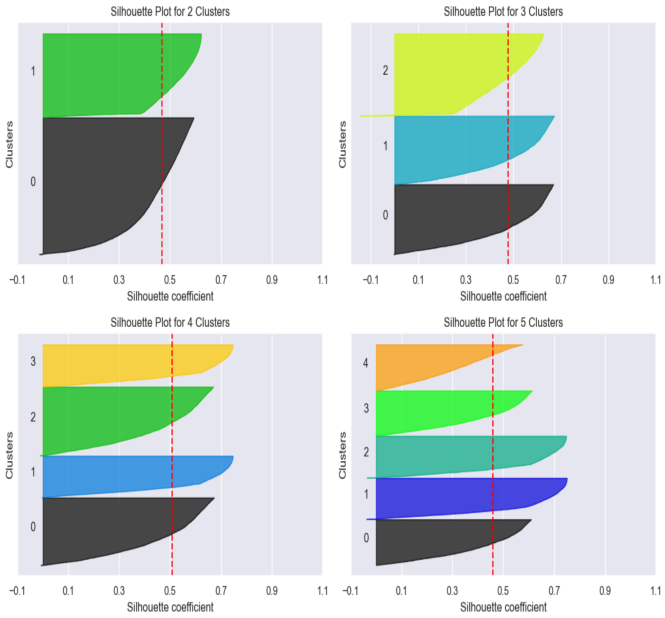

Fig. 16. Silhouette Graphs for Different Clusters

IV. Results and Experiments

After creating the four clusters, it is evident that the main differences between them are based on *Balance and Salary*. The clusters are distinctly categorized according to the amounts of balance and income, suggesting that these two characteristics are crucial for customer segmentation. For instance, clusters with higher balances and greater salaries might include customers with greater financial stability and higher purchasing power. Conversely, clusters with lower balances and salaries could represent customers with more limited financial resources.

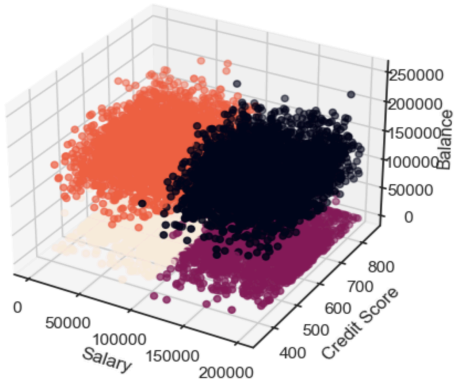

Fig. 17. Visualization of Customer Clusters

This observation indicates that balance and salary are key indicators of customer financial profiles and can be used to design tailored marketing and management strategies for each group. For *targeting* these clusters, banks could develop customized products and services. For example:

- ***Customers with high balances and high salaries*** might benefit from premium offers such as private wealth management accounts, advantageous loan conditions, or sophisticated investment products.
- ***Customers with high balances but low salaries*** could be offered high-yield savings accounts, flexible investment solutions, or personalized financial advisory services to optimize their substantial assets.
- ***Customers with low balances but high salaries*** might benefit from high-interest savings accounts, customized credit solutions, or comprehensive financial planning services to better manage their income and grow their savings.
- ***Customer with low balances and salaries*** could be targeted with accounts with reduced fees, or credit options suited to their needs.

This targeted approach would allow for more precise responses to the financial needs of each segment, thereby enhancing customer satisfaction and optimizing sales opportunities for the bank.

V. Conclusion

Leveraging the Next Best Offer principle in conjunction with customer segmentation allows for highly targeted and effective banking strategies. By analyzing customer clusters based on attributes such as balance and salary, financial institutions can tailor their offerings to meet the specific needs and preferences of each group. This approach not only enhances customer satisfaction and loyalty by providing relevant solutions but also optimizes the institution's engagement and profitability by aligning offers with individual financial profiles and needs.

REFERENCES

[1] B. Mahesh, "Machine learning algorithms-a review," *International Journal of Science and Research (IJSR).*vol. 9, pp. 381–386, 2020.
[2] Han, J., Pei, J. & Kamber, M. "Data Mining: Concepts and Techniques" Elsevier, 2011
[3] Goodfellow, I., Bengio, Y. & Courville, A. "Deep Learning" MIT Press, 2016
[4] Lloyd, S. "Least Square Quantization in PCM", IEEE, 1982
[5] Jain, A. K. "Data Clustering: 50 years beyond K-means", 2010
[6] Halkidi, M., Batistakis et al. "On Clustering Validation Techniques" Journal of Intelligent Information Systems, 2001
[7] Lloyd, S. "Least Square Quantization in PCM", IEEE, 1982
[8] Rousseuw, P. J. "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis" Journal of Computational and Applied Mathematics, 1987
[9] Tibshirani, R., Walther, Hastie, T. "Estimating the number of clusters in a dataset via the gap statistic" Journal of the Royal Statistical Society, 2001
[10] Jolliffe, I. T. "Principal Component Analysis" Springer Series in Statistics