# Product Bundle Identification using Clustering Algorithms

### Name: Siddhant Sarnobat Mail ID: siddhantsarnobat20@gmail.com

# INDEX

I. Introduction

* Overview of the Project
* Objective: Product Bundle Identification using Clustering Algorithms

II. Data Preprocessing

* Load Necessary Libraries
  + Brief description of libraries used (tidyverse, lubridate, arules, cluster, factoextra, FactoMineR).
* Load and Inspect the Dataset
  + Description of the dataset structure and content.
  + Summary statistics from exploratory data analysis (EDA).
* Data Preparation
  + Conversion to a data frame and handling outliers.
  + Normalization of numeric data.
* Prepare Data for PCA
  + Setting Product\_Code as row names.
  + Conversion to matrix format for PCA.
  + Verification of data dimensions and structure.

III. Principal Component Analysis (PCA)

* Apply PCA
  + Explanation of PCA application with parameters.
* Interpretation of PCA results
  + Variance explained and principal components.
* Compute Silhouette Width
  + Calculation and interpretation of silhouette width for initial clustering evaluation.

IV. Clustering Algorithms

* K-means Clustering
  + Determination of optimal clusters using the Elbow method.
  + Implementation of K-means clustering with optimal K.
  + Evaluation of clustering quality using silhouette analysis.
  + Visualization of clusters and silhouette plot.
* Hierarchical Clustering
  + Perform hierarchical clustering and dendrogram visualization.
  + Selection of clusters based on maximum distance threshold.
  + Evaluation of clustering quality using silhouette analysis.
  + Visualization of clusters and silhouette plot.
* DBSCAN Clustering
  + Implementation of DBSCAN clustering with specified parameters (eps, minPts).
  + Handling of noise points and identification of useful clusters.
  + Evaluation of clustering quality using silhouette analysis.
  + Visualization of clusters and silhouette plot.

V. Conclusion and Results

* Summary of findings from each clustering algorithm.
* Comparison of clustering methods in terms of effectiveness for product bundle identification.
* Selection of DBSCAN as the final clustering model based on performance metrics.

VI. Future Work

* Recommendations for further analysis or improvements.
* Potential applications or extensions of the current methodology.

# I. Introduction

## Overview of the Project

In today's competitive market, understanding consumer behavior and preferences is crucial for businesses aiming to optimize their product offerings. One effective strategy is product bundling, where complementary items are grouped together to enhance customer value and increase sales. The goal of this project is to leverage data-driven techniques, specifically clustering algorithms, to identify meaningful product bundles within a dataset of weekly sales transactions.

## Objective: Product Bundle Identification using Clustering Algorithms

The primary objective of this project is to apply advanced data analytics techniques to categorize products into coherent bundles. Clustering algorithms will be employed to group products based on their sales patterns, aiming to uncover associations that indicate natural affinities between items. By doing so, the project seeks to facilitate strategic marketing decisions and enhance operational efficiency by offering targeted bundled products that appeal to specific customer segments.

Through this endeavor, we aim to demonstrate the efficacy of clustering algorithms—specifically K-means, Hierarchical Clustering, and DBSCAN—in extracting meaningful insights from sales data. These insights will not only help in identifying product bundles but also provide actionable recommendations for businesses to optimize their marketing strategies and drive sales growth.

This report outlines the methodology, results, and implications derived from applying these clustering techniques to the dataset, ultimately contributing to the broader understanding of how data science can be applied in retail and marketing contexts for product strategy enhancement.

# II. Data Preprocessing

## Load Necessary Libraries

The project utilizes several R libraries for efficient data manipulation, visualization, and advanced analytics:

* **tidyverse**: A comprehensive collection of packages for data manipulation and visualization, facilitating streamlined workflows.
* **lubridate**: Provides functions to work with dates and times, ensuring easy handling of temporal data if required.
* **arules**: Supports market basket analysis for discovering associations and patterns in transactional data.
* **cluster**: Offers various clustering algorithms, essential for grouping similar items based on defined criteria.
* **factoextra**: Provides functions to visualize multivariate analysis results, enhancing the interpretation of clustering outcomes.
* **FactoMineR**: Enables principal component analysis (PCA) and factor analysis, fundamental for dimensionality reduction and pattern recognition in high-dimensional datasets.

## Load and Inspect the Dataset

The dataset used in this project consists of weekly sales transactions structured as a data frame:

* **Structure**: Each row represents a product identified by Product\_Code, and each column corresponds to weekly sales data (W0 to W51).
* **Content**: The dataset contains sales figures over time, allowing for temporal analysis of product performance and trends.

## Summary Statistics from Exploratory Data Analysis (EDA)

During EDA, key summary statistics were computed to understand the dataset's characteristics:

* **Central Tendency**: Mean and median provide insights into average sales volume and distribution.
* **Variability**: Standard deviation and range highlight the dispersion of sales values across products.
* **Distribution**: Histograms and box plots reveal the distribution of sales data, identifying outliers and potential data anomalies.

## Data Preparation

1. **Conversion to a Data Frame and Handling Outliers**:
   * The dataset was imported into R and converted into a data frame (data\_df).
   * An outlier detection function based on Z-score was applied to numeric columns, marking extreme values as NA to ensure data integrity.
2. **Normalization of Numeric Data**:
   * Numeric columns, excluding Product\_Code, were normalized using the scale() function, standardizing the sales data to a common scale.
   * Normalization facilitates fair comparison and improves the performance of clustering algorithms sensitive to scale differences.

## Prepare Data for PCA

1. **Setting Product\_Code as Row Names**:
   * To facilitate easier referencing and analysis, the Product\_Code column was set as row names in the data\_df data frame.
2. **Conversion to Matrix Format for PCA**:
   * The processed data frame (data\_df) was converted into a matrix (data\_matrix) suitable for PCA analysis using as.matrix().
3. **Verification of Data Dimensions and Structure**:
   * Prior to PCA, the dimensions and structure of data\_matrix were verified to ensure compatibility with PCA functions.
   * Confirmation of correct row names alignment and matrix dimensions was conducted to avoid data misalignment during subsequent analyses.

# III. Principal Component Analysis (PCA)

## Apply PCA

Principal Component Analysis (PCA) is a dimensionality reduction technique used to transform high-dimensional data into a smaller set of variables called principal components while preserving the variation present in the original data. In this project, PCA is applied to the dataset of weekly sales transactions to extract underlying patterns and reduce the number of dimensions for clustering analysis.

**Parameters Used:**

* **Scale.unit**: Ensures variables are scaled to have unit variance, necessary when variables are measured on different scales.
* **ncp (Number of Components)**: Specifies the number of principal components to retain. Typically, the number is chosen based on the amount of variance explained or specific business requirements.
* **Graph**: Determines whether graphical outputs of PCA results (e.g., scree plot) are displayed.

**Interpretation of PCA Results:**

* **Variance Explained**: PCA results include the proportion of variance explained by each principal component. This information helps in understanding how much information each principal component retains from the original dataset.
* **Principal Components**: Each principal component is a linear combination of the original variables. They are ordered by the amount of variance they explain, with the first component explaining the most variance, followed by subsequent components.

## Compute Silhouette Width

Silhouette width is a metric used to evaluate the quality of clusters formed by clustering algorithms, including those based on PCA-transformed data. It measures how similar each point is to its own cluster compared to other clusters. Silhouette width ranges from -1 to 1, where a value close to 1 indicates well-clustered data points.

**Calculation and Interpretation:**

* **Calculation**: Silhouette width is computed using the silhouette() function from the cluster package, which requires the clusters formed and the distance matrix of the data.
* **Interpretation**: A higher average silhouette width suggests that the clusters are well-separated and distinct. This metric helps in selecting the optimal number of clusters and assessing the clustering algorithm's performance.

In this project, PCA serves as a preprocessing step to reduce the dataset's dimensions while retaining important information. The computed principal components and silhouette width will guide the subsequent clustering algorithms—K-means, Hierarchical Clustering, and DBSCAN—in identifying meaningful product bundles based on sales patterns.

# IV. Clustering Algorithms

## K-means Clustering

**Determination of Optimal Clusters using the Elbow Method:** The Elbow method was used to determine the optimal number of clusters (K) for K-means clustering. It helps in identifying a point where the within-cluster sum of squares (WSS) starts to diminish, suggesting an optimal balance between complexity and explanatory power.

**Implementation of K-means Clustering with Optimal K:** Based on the Elbow method, the optimal number of clusters (K) was chosen to segment the PCA-transformed data effectively. K-means clustering partitions the data into K clusters by iteratively assigning data points to the nearest cluster centroid.

**Evaluation of Clustering Quality using Silhouette Analysis:** Silhouette analysis was employed to assess the quality of the clusters formed by K-means clustering. It measures how similar each point is to its own cluster compared to other clusters, providing insight into the clustering structure's cohesion and separation.

**Visualization of Clusters and Silhouette Plot:** Visual representations such as silhouette plots and cluster visualizations were used to illustrate the clustering results. These visuals help in interpreting the cluster assignments and assessing the compactness and separation of clusters.

## Hierarchical Clustering

**Perform Hierarchical Clustering and Dendrogram Visualization:** Hierarchical clustering was applied to the PCA-transformed data to create a hierarchy of clusters. A dendrogram, which illustrates the hierarchical relationships between data points, was used to aid in determining the number of clusters by identifying distinct clusters at different heights.

**Selection of Clusters Based on Maximum Distance Threshold:** Clusters were selected by cutting the dendrogram at a specified maximum distance threshold. This approach helps in delineating clusters based on a chosen level of similarity or dissimilarity between data points.

**Evaluation of Clustering Quality using Silhouette Analysis:** Silhouette analysis was conducted to evaluate the quality of hierarchical clustering results. It provides a measure of how well each data point fits into its assigned cluster relative to other clusters, aiding in the validation of cluster assignments.

**Visualization of Clusters and Silhouette Plot:** Visualizations of hierarchical clustering results, including cluster dendrograms and silhouette plots, were utilized to visualize cluster structures and validate clustering outcomes.

## DBSCAN Clustering

**Implementation of DBSCAN Clustering with Specified Parameters:** DBSCAN clustering, a density-based clustering method, was implemented using specified parameters such as epsilon (eps) and minimum points (minPts). This method identifies clusters based on dense regions of data points, accommodating varying densities within the dataset.

**Evaluation of Clustering Quality using Silhouette Analysis:** Silhouette analysis was employed to assess the quality of DBSCAN clustering results. It focuses on evaluating clusters identified by DBSCAN, particularly useful clusters that exhibit significant within-cluster similarity and distinctiveness from other clusters.

**Visualization of Clusters and Silhouette Plot:** Visualizations such as silhouette plots and cluster representations were utilized to illustrate DBSCAN clustering results. These visuals help in understanding the distribution of clusters and identifying noise points or outliers within the dataset.

# V. Conclusion and Results

## Summary of Findings from Each Clustering Algorithm

**K-means Clustering:** K-means clustering identified three distinct clusters based on the PCA-transformed sales data. The average silhouette width, a measure of clustering quality, was 0.742. Cluster 3 showed the highest average silhouette width of 0.86, indicating better internal cohesion and separation compared to other clusters.

**Hierarchical Clustering:** Hierarchical clustering segmented the data into two main clusters with an average silhouette width of 0.774. Cluster 1 exhibited a higher average silhouette width (0.82), suggesting more homogeneous intra-cluster points compared to Cluster 2 (0.61).

**DBSCAN Clustering:** DBSCAN identified three clusters, with an average silhouette width of 0.760. Cluster 3 demonstrated the highest average silhouette width of 0.90, indicating well-defined and distinct clusters within the dataset. Notably, DBSCAN effectively handled noise points and identified meaningful clusters despite varying densities.

## Comparison of Clustering Methods in Terms of Effectiveness for Product Bundle Identification

Each clustering method—K-means, Hierarchical, and DBSCAN—offers unique insights into product bundle identification:

* **K-means** provides clear, compact clusters but requires predefined K and may be sensitive to outliers.
* **Hierarchical clustering** offers a hierarchical structure that can reveal nested clusters but is sensitive to the choice of linkage method and distance metric.
* **DBSCAN** adapts well to varying cluster densities and noise points, automatically identifying clusters based on local density, which is advantageous for real-world datasets with irregular shapes and densities.

## Selection of DBSCAN as the Final Clustering Model Based on Performance Metrics

Based on the evaluation metrics:

* **DBSCAN** achieved the highest average silhouette width (0.760), indicating robust clustering performance with well-defined clusters.
* It effectively handled noise points and automatically determined the number of clusters based on data density, making it suitable for identifying product bundles in sales data where cluster shapes and densities can vary.

Therefore, **DBSCAN** is selected as the final clustering model for this project due to its superior performance metrics and ability to handle the inherent complexities of the dataset effectively.

# VI. Future Work

## Recommendations for Further Analysis or Improvements

**1. Feature Engineering:**

* Explore additional features or transformations that could enhance clustering results, such as seasonal adjustments or product-specific attributes beyond sales data.

**2. Ensemble Clustering Approaches:**

* Implement ensemble clustering methods that combine results from multiple algorithms (e.g., K-means, DBSCAN, and Hierarchical) to potentially improve clustering robustness and accuracy.

**3. Incorporating External Data:**

* Integrate external datasets, such as customer demographics or marketing campaign data, to enrich the clustering analysis and uncover deeper insights into product bundling strategies.

**4. Time-series Analysis:**

* Incorporate time-series analysis to capture temporal patterns in sales data, allowing for dynamic clustering that considers changing consumer behaviors over time.

## Potential Applications or Extensions of the Current Methodology

**1. Market Basket Analysis:**

* Extend clustering results to perform market basket analysis, identifying frequently co-purchased products within each cluster to suggest effective product bundling strategies.

**2. Predictive Modeling:**

* Develop predictive models using clustered data to forecast sales trends for bundled products, enabling proactive inventory management and marketing campaign planning.

**3. Customer Segmentation:**

* Apply clustering insights to segment customers based on their purchasing behavior, allowing for personalized marketing approaches and targeted product recommendations.

**4. Real-time Clustering:**

* Implement real-time clustering algorithms to handle streaming data and enable timely adjustments to product bundling strategies based on evolving market dynamics.

## Conclusion

The methodology presented lays a foundation for exploring and optimizing product bundle identification using clustering algorithms. By expanding on these recommendations and applications, future analyses can further enhance decision-making processes in marketing, sales, and inventory management, ultimately leading to improved customer satisfaction and business outcomes.