CUSTOMER SEGMENTATION USING DATASCIENCE

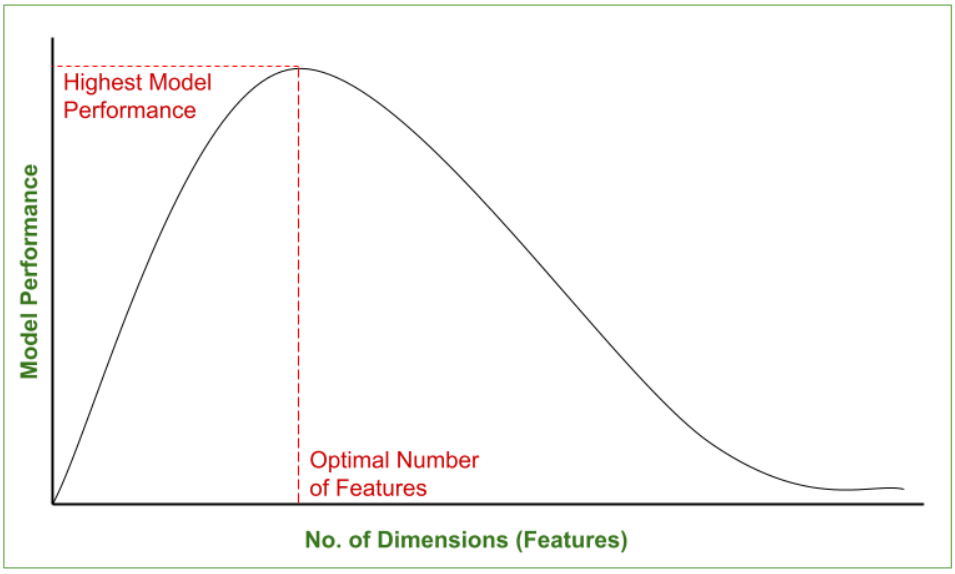
Phase-2 Project

Introduction

The advancements in Data Science and Machine Learning have made it possible for us to solve several complex regression and classification problems. However, the performance of all these ML models depends on the data fed to them. Thus, it is imperative that we provide our ML models with an optimal dataset. Now, one might think that the more data we provide to our model, the better it becomes – however, it is not the case. If we feed our model with an excessively large dataset (with a large no. of features/columns), it gives rise to the problem of where in the model starts getting influenced by outlier values and noise. This is called the **Curse of Dimensionality**.

Curse of Dimensionality.

The following graph represents the change in model performance with the increase in the number of dimensions of the dataset. It can be observed that the model performance is best only at an option dimension, beyond which it starts decreasing.



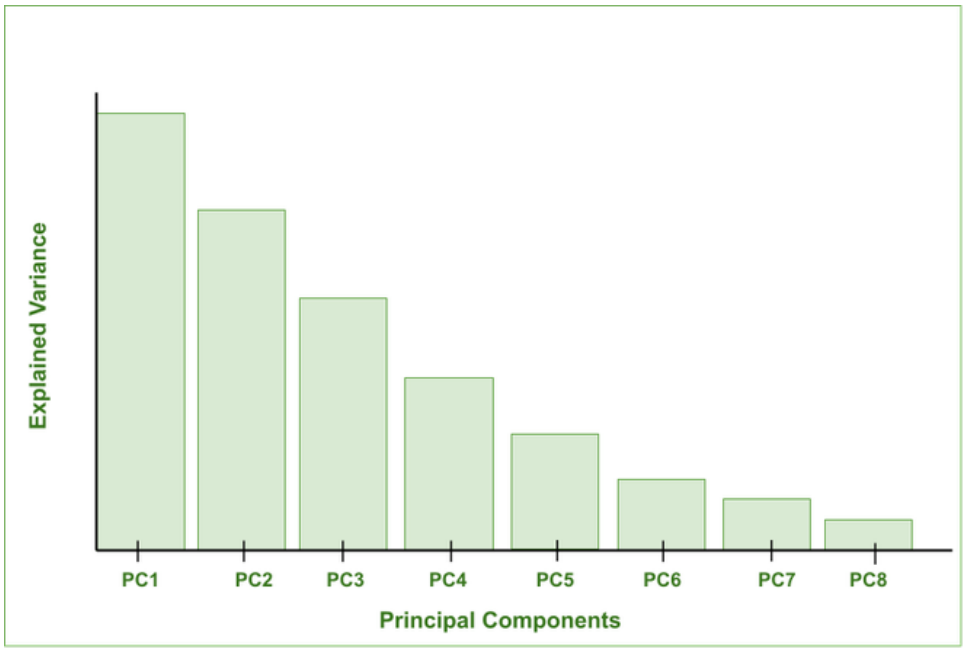
Dimensionality Reduction is a statistical/ML-based technique wherein we try to reduce the number of features in our dataset and obtain a dataset with an optimal number of dimensions.

One of the most common ways to accomplish Dimensionality Reduction is Feature Extraction, wherein we reduce the number of dimensions by mapping a higher dimensional feature space to a lower-dimensional feature space. The most popular technique of Feature Extraction is Principal Component Analysis (PCA)

Principal Component Analysis (PCA)

As stated earlier, Principal Component Analysis is a technique of feature extraction that maps a higher dimensional feature space to a lower-dimensional feature space. While reducing the number of dimensions, PCA ensures that maximum information of the original dataset is retained in the dataset with the reduced no. of dimensions and the co-relation between the newly obtained Principal Components is minimum. The new features obtained after applying PCA are called Principal Components and are denoted as PCi (i=1,2,3.....n). Here, (Principal Component-1) PC1 captures the maximum information of the original dataset, followed by PC2, then PC3 and so on.

The following bar graph depicts the amount of Explained Variance captured by various Principal Components. (The Explained Variance defines the amount of information captured by the Principal Components).



Explained Variance Vs Principal Components

n order to understand the mathematical aspects involved in Principal Component Analysis do check out **Mathematical Approach to PCA.** In this article, we will focus on how to use PCA in Python for Dimensionality Reduction.

Data Collection and Preprocessing:

- Start by gathering relevant customer data, which can include demographics, purchase history, website activity, and more.
- Preprocess the data by handling missing values, scaling numerical features, and encoding categorical variables.

Feature Selection:

- Carefully choose the features that you believe are relevant to segment customers. High-dimensional data can lead to increased computational complexity and overfitting.

Dimensionality Reduction:

a. **Principal Component Analysis (PCA):**

- PCA is a linear dimensionality reduction technique that can help reduce the number of features while preserving as much variance in the data as possible.
- Apply PCA to the feature matrix to create a set of principal components. These components are linear combinations of the original features.
- You can choose to retain a certain percentage of variance (e.g., 95%) and keep the corresponding principal components.

b. **t-Distributed Stochastic Neighbor Embedding (t-SNE):**

- t-SNE is a non-linear dimensionality reduction technique that is especially useful for visualizing high-dimensional data.
- It aims to maintain the pairwise distances between data points, making it effective for preserving local structure in the data.
- Apply t-SNE to create a lower-dimensional representation of your customer data. Typically, you'd reduce it to 2 or 3 dimensions for visualization.

Visualization:

- After dimensionality reduction, you can create visualizations to explore customer segments.
- Plot the reduced-dimensional data using scatter plots or other visualization techniques. Points that are close in the reduced space are likely to be similar customers.
- You can color-code or label the points according to the segments generated by clustering algorithms (e.g., K-Means) or other segmentation techniques.

Clustering:

- Use clustering algorithms like K-Means, hierarchical clustering, or DBSCAN on the reduced-dimensional data to create customer segments.
- The clusters should represent groups of customers with similar behaviors or characteristics.

Interpretation and Action:

- Analyze the characteristics of each customer segment to understand their unique attributes and behaviors.
- Develop targeted marketing strategies, product recommendations, or personalized experiences for each segment.
- Continuously monitor and refine your segmentation approach as new data becomes available.

Evaluation:

- Evaluate the effectiveness of your segmentation by measuring key performance metrics such as conversion rates, retention, and revenue for each segment.
- Use A/B testing or other methods to validate the impact of your personalized strategies on each customer segment.

Conclusion:

Incorporating dimensionality reduction techniques like PCA and t-SNE into your customer segmentation workflow can help you gain insights from high-dimensional data, identify hidden patterns, and make more informed decisions to improve customer engagement and satisfaction.