## Project Report: Unsupervised Learning Using K-Means Clustering

**Dataset**: [Facebook Live Sellers in Thailand (UCI ML Repository)](https://archive.ics.uci.edu/ml/datasets/Facebook+Live+Sellers+in+Thailand" \t "_new)  
**Tool**: Google Colab (Python)

# **Unsupervised Learning**

Machine learning algorithms can be broadly classified into two categories : supervised and unsupervised learning. There are other categories also like semi-supervised learning and reinforcement learning. But, most of the algorithms are classified as supervised or unsupervised learning. The difference between them happens because of presence of target variable. In unsupervised learning, there is no target variable. The dataset only has input variables which describe the data. This is called unsupervised learning.

1. ****Means clustering**** is the most popular unsupervised learning algorithm. It is used when we have unlabelled data which is data without defined categories or groups. The algorithm follows an easy or simple way to classify a given data set through a certain number of clusters, fixed apriori. K-Means algorithm works iteratively to assign each data point to one of K groups based on the features that are provided. Data points are clustered based on feature similarity.

K-Means clustering can be represented diagrammatically as follows:-

# ****K-Means****

# ****K-Means Clustering with Python****

****K-Means clustering**** is used to find intrinsic groups within the unlabelled dataset and draw inferences from them. In this kernel, I implement K-Means clustering to find intrinsic groups within the dataset that display the same status\_type behaviour. The status\_type behaviour variable consists of posts of a different nature (video, photos, statuses and links).

# ****Applications of clustering****

K-Means clustering is widely used for many applications which include:

1. Image segmentation
2. Customer segmentation
3. Species clustering
4. Anomaly detection
5. Clustering languages

# ****K-Means Clustering intuition****

K-Means clustering is used to find intrinsic groups within the unlabelled dataset and draw inferences from them. It is based on centroid-based clustering.

****Centroid**** - A centroid is a data point at the centre of a cluster. In centroid-based clustering, clusters are represented by a centroid. It is an iterative algorithm in which the notion of similarity is derived by how close a data point is to the centroid of the cluster. K-Means clustering works as follows:- The K-Means clustering algorithm uses an iterative procedure to deliver a final result. The algorithm requires number of clusters K and the data set as input. The data set is a collection of features for each data point. The algorithm starts with initial estimates for the K centroids. The algorithm then iterates between two steps:-

## ****Data assignment step****

Each centroid defines one of the clusters. In this step, each data point is assigned to its nearest centroid, which is based on the squared Euclidean distance. So, if ci is the collection of centroids in set C, then each data point is assigned to a cluster based on minimum Euclidean distance.

## ****Centroid update step****

In this step, the centroids are recomputed and updated. This is done by taking the mean of all data points assigned to that centroid’s cluster.

The algorithm then iterates between step 1 and step 2 until a stopping criteria is met. Stopping criteria means no data points change the clusters, the sum of the distances is minimized or some maximum number of iterations is reached. This algorithm is guaranteed to converge to a result. The result may be a local optimum meaning that assessing more than one run of the algorithm with randomized starting centroids may give a better outcome.

The K-Means intuition can be represented with the help of following diagram:-

# ****k-means institution****

# ****Choosing the value of K****

The K-Means algorithm depends upon finding the number of clusters and data labels for a pre-defined value of K. To find the number of clusters in the data, we need to run the K-Means clustering algorithm for different values of K and compare the results. So, the performance of K-Means algorithm depends upon the value of K. We should choose the optimal value of K that gives us best performance. There are different techniques available to find the optimal value of K. The most common technique is the ****elbow method**** which is described below.

# ****The elbow method****

The elbow method is used to determine the optimal number of clusters in K-means clustering. The elbow method plots the value of the cost function produced by different

We can see that if K increases, average distortion will decrease. Then each cluster will have fewer constituent instances, and the instances will be closer to their respective centroids. However, the improvements in average distortion will decline as K increases. The value of K at which improvement in distortion declines the most is called the elbow, at which we should stop dividing the data into further clusters.

**INTRODUCTION**

**Objective:**To explore and group similar Facebook live videos using K-Means clustering based on features such as likes, shares, comments, and other engagement metrics.

**Background**: Live streaming has become a powerful tool for small businesses to promote and sell products directly to audiences. Understanding patterns in engagement can help improve strategy.

**Problem Statement**: The dataset lacks predefined labels, so we aim to identify distinct clusters of Facebook Live sessions that exhibit similar viewer behavior.

### ****Dataset Overview****

### ****Source****: UCI Machine Learning Repository

### ****Data Description****: The dataset contains 7050 Facebook live session records with the following key features:

### status\_type: Type of status (live, photo, video, etc.)

* num\_reactions, num\_comments, num\_shares: Engagement metrics
* num\_views: Number of views
* num\_likes, num\_loves, num\_wows, etc.: Reaction breakdown

**Target**: Unsupervised (no target column)

**EXPLORATORY DATA ANALYSIS**

We can see that there are 7050 instances and 16 attributes in the dataset. In the dataset description, it is given that there are 7051 instances and 12 attributes in the dataset.So, we can infer that the first instance is the row header and there are 4 extra attributes in the dataset. Analysis the following to prepare the data:

* Handled missing values
* Drop redundant columns
* Trimmed dataset by droping status\_id and status\_published
* Normalized/standardized numerical features
* Convert category variable into integers
* One-hot encoding for categorical data

**METHODOLOGY**

* **Unsupervised Learning Technique**: K-Means Clustering
* **Reason for Choice**:
* Simple and effective for discovering groupings in unlabeled data
* Works well with numerical and engagement-based features
* **Steps Taken**:

1. Declare feature vector and target variable
2. Feature scaling
3. Fitting K-means model
4. K-mean model parameters study(inertial)
5. Check quality of quality of weak classification by the model
6. Used elbow method to find optimal number of cluster
7. K-mean model with different cluster to achieve more relatively high accuracy.

* The KMeans algorithm clusters data by trying to separate samples in n groups of equal variances, minimizing a criterion known as ****inertia****, or within-cluster sum-of-squares Inertia, or the within-cluster sum of squares criterion, can be recognized as a measure of how internally coherent clusters are.
* The k-means algorithm divides a set of N samples X into K disjoint clusters C, each described by the mean j of the samples in the cluster. The means are commonly called the cluster ****centroids****.
* The K-means algorithm aims to choose centroids that minimize the inertia, or within-cluster sum of squared criterion.

### Inertia

****Inertia**** is not a normalized metric.

* The lower values of inertia are better and zero is optimal.
* But in very high-dimensional spaces, euclidean distances tend to become inflated (this is an instance of curse of dimensionality).
* Running a dimensionality reduction algorithm such as PCA prior to k-means clustering can alleviate this problem and speed up the computations.
* The lesser the model inertia, the better the model fit.
* We can see that the model has very high inertia. So, this is not a good model fit to the data.

# ****Use elbow method to find optimal number of clusters****

* So, our weak unsupervised classification model achieved a very weak classification accuracy of 1%.
* I will check the model accuracy with different number of clusters.

# ****K-Means model with different clusters****

Result: 4288 out of 7050 samples were correctly labeled.

Accuracy score: 0.61

We have achieved a relatively high accuracy of 61% with k=4.

# ****Results and conclusion****

* In this project, I have implemented the most popular unsupervised clustering technique called ****K-Means Clustering****.
* I have applied the elbow method and find that k=1 (k is number of clusters) can be considered a good number of cluster to cluster this data.
* I have find that the model has very high inertia of 899.1609. So, this is not a good model fit to the data.
* I have achieved a weak classification accuracy of 1% with k=1 by our unsupervised model.
* So, I have changed the value of k and find relatively higher classification accuracy of 61% with k=2.
* I have find that the model has a relatively low inertia of 237.7572. So, it’s a good model fit to the data.
* Hence, we can conclude that k=2 being the optimal number of clusters.

### ****Limitations****

* The dataset may not represent all small businesses
* Only Facebook Live is covered – no multichannel analysis
* K-Means assumes spherical clusters – may not always match real-world data structure

### ****Recommendations****

* Further analysis using other clustering algorithms (DBSCAN, Hierarchical)
* Combine with sentiment analysis of comments
* Develop a dashboard to track cluster performance over time

### ****Appendix****

**colab link:** https://colab.research.google.com/drive/1gfxkizpq6OQ4lvbaThCZWuJv\_UkDftOu?usp=sharing

**Key libraries used**: pandas, matplotlib, seaborn, scikit-learn

# ****References****

The work done in this project is inspired from following books and websites:-

Udemy course – Machine Learning – A Z by Kirill Eremenko and Hadelin de Ponteves

<https://en.wikipedia.org/wiki/K-means_clustering>

<https://jakevdp.github.io/PythonDataScienceHandbook/05.11-k-means.html>

<https://www.datacamp.com/community/tutorials/k-means-clustering-python>

<https://www.datascience.com/blog/k-means-clustering>

<https://acadgild.com/blog/k-means-clustering-algorithm>