A Guide To The K-means Clustering Algorithm

Parker Gibbons ‘25

Abstract

Data in the real-world could be presented with natural groupings and patterns that are not immediately obvious. Thus we need a method for working with unlabeled, unclassified data (unsupervised learning). There is also the chance that datasets could be higher dimensional with multiple features, which could be difficult to directly interpret. As a result, the method also needs to be able to reduce the dimension of the data or observation space. Moreover, the method needs to be appropriate to a wide range of applications where unsupervised learning is required to identify natural groupings or patterns within data. [4] [5]

Goal: To partition a dataset into separate and distinct groups by minimizing the variance within each group and maximizing the difference between groups [5]

Introduction

K-means clustering algorithm: An unsupervised machine learning algorithm that is widely used in the field of data analytics. [5]

The K-means clustering algorithm provides a powerful method for identifying patterns within data and partitioning data points into a set of k groups (*k* clusters). Since the method is unsupervised, it uses a pre-specified number of clusters which can be determined through various methods such as the elbow method. The algorithm seeks to create clusters that are distinct and non-overlapping while also being defined by their centroid, the average of all data points in the cluster. Steps are taken during this process to refine each centroid to ensure that the data points in each cluster are as similar as possible, while also making sure each cluster is distinct from the others. This is done using a distance measure such as Euclidean distance , Manhattan distance, or correlation-based distances. [4] [5]

Definitions

- **Cluster:** A group of data points that have similar characteristics or are closer to one another than to data points of other groups (clusters) [5]
- **Centroid:** The mean of the observation values assigned to a cluster [5]
- **Sum of squares (SS) (Statistics):** How spread out a set of data points is from the mean [5]
- **Total within-cluster variation (SS_{within}):** Measures the compactness (i.e., goodness) of each cluster, which we want to be as small as possible [5]
- **Euclidean distance (Linear Algebra):** The straight line distance between two points [5] $d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$

Project Goals

- Effectively explain the steps of the k-means clustering algorithm as well as the math and theory behind it
- Show the various applications and scenarios in which the algorithm can be used
- Show how the algorithm can be used in R with larger datasets, using visual interpretations of the clusters which makes it easier to assess the variation within each cluster and the magnitude of difference between each cluster

Formal Mathematical Statement

$$\min_{C_1, C_2, \ldots, C_k} \sum_{i=1}^{k} \sum_{x_j \in C_i} \|x_j - \mu_i\|^2$$

Where:

- μ_i is the centroid of each cluster C_i
- $\|x_j - \mu_i\|^2$ is the Euclidean distance between data point x_j and cluster centroid μ_i

[5]

Guiding Example

An example from STAT306: Multivariate Sports Analytics will be used throughout the explanation of the algorithm steps.

Goal: We want to determine whether MLB teams of different market sizes depend on different variables to win and have playoff success

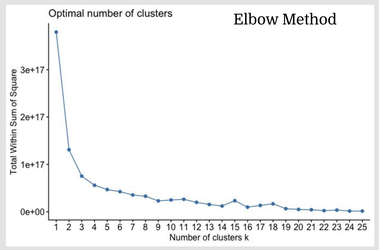
Hypothesis: Teams of different market sizes depend on different variables to win and have playoff success

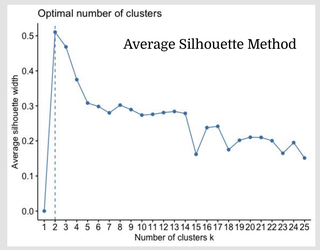
Variables used for clustering: Total payroll from the past 4 seasons, City population, Media market size, Team valuation, Attendance per game from the past 3 seasons

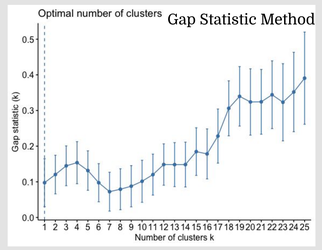
Algorithm Steps

Step 1: Specifying number of clusters (k)

- Common rule of thumb: $k = \sqrt{n/2}$ [5]
- Logical number based on context and goal of analysis (e.g. sports markets, categories) [2]
- Elbow method
  - Looks at total within-cluster sum of squares as a function of the number of clusters [2]
- Average silhouette method
  - Measures quality of clustering and how well each point lies within its cluster [2]
- Gap statistic method
  - Aims to maximize gap statistic under null reference distribution [2]







Example

- Use context
  - e.g. Sports markets are usually classified in three groups (small, medium, large)
- Elbow method suggests three or four clusters could be optimal
  - Determined by looking for knee in the plot (adding another cluster does not decrease the within-cluster sum of squares much)

Step 2: Selecting initial centroids

- Select *k* random observations at random from the data set to use as the initial cluster centroids [5]

Step 3: Cluster assignment

- Assign each observation to their closest centroid based on the distance measure selected [3]
  - **Euclidean distance**
    - Optimal for normal distributed features, very sensitive to outliers, could skew cluster results and give false confidence in the compactness of the cluster [5]
  - **Manhattan distance** (sum of the absolute differences between coordinates of two points)
    - Optimal for features that slightly differ from normality [5]
  - **Correlation-based distances** (widely used for gene expression data) [5]
    - Minkowski or Gower distances

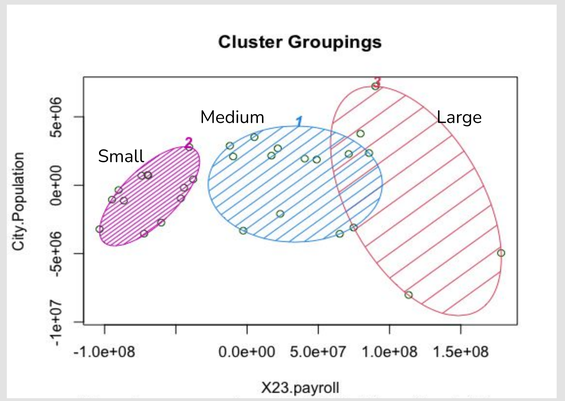
Step 4: Updating cluster centroids

- For each of the *k* clusters, update the cluster centroid by calculating the new mean values of all the data points in the cluster
  - The centroid for the *i*-th cluster is a vector of length *p* containing the means of all *p* features for the observations in cluster *i*

Step 5: Minimize within-cluster variation

- Using new recalculated centroids, every observation is checked to see if it might be closer to a different cluster than it’s current cluster
- All observations are reassigned again to their nearest cluster using the updated centroids
- Iteratively minimize SS_{within}
  - Cluster assignment and centroid update steps are iteratively repeated until the cluster assignments and centroids stop changing (the clusters formed in the current iteration are the same as those obtained in the previous iteration) [5]
  - A good rule of thumb is between 10 and 20 iterations

Clustering Results



*Each axis scale shows distance from mean

Small	Medium	Large

Annotated R-Code For Example

[1] [2] [4] [5]