Direct Partitioning Clustering

Predictive Modeling & Statistical Learning

Gaston Sanchez

Introduction

Direct Partitioning Methods

In these slides we'll discuss direct partitioning methods, which can be labeled under the umbrella term of *prototype methods*.

Direct Partitioning Methods

With direct partitioning methods, two clusters are always separated; and the number of clusters is generally defined a priori. The most common methods of this kind of clustering are:

- K-means
- K-medoids
- other flavors of K-prototypes
- Kohonen networks (or maps)

Clustering Idea

Like all clustering methods, direct partitioning methods around prototypes involves dividing a set of n objects in K groups such that:

- each group is as much homogeneous as possible
  i.e. *within-groups homogeneity*.
- groups are as distinct as possible among them
  i.e. *between-groups heterogeneity*

K-Means

How does K-Means work?

We would like to partition a data set of n objects into K non-overlapping clusters

$$C_1, C_2, \ldots, C_K$$

For instance, if the i-th observation is in the k-th cluster, then $i \in C_k$.

Preliminary Concepts

Let $C_1, C_2, \ldots, C_K$ denote sets containing the indices of the observations in each cluster.

- $C_1 \cup C_2 \cup \cdots \cup C_K = \{1, \ldots, n\}$. In other words, each observation belongs to at least one of the K clusters.
- $C_k \cap C_h = \emptyset$ for all $k \neq h$. In other words, the clusters are non-overlapping: no observation belongs to more than one cluster.

Clustering Criterion

The goal is to have a minimal "within-cluster variation", i.e. the elements within a cluster should be as similar as possible.

One way of achieving this is to minimize the sum of all the pair-wise squared distances between the observations in each cluster:

$$min \left\{ \sum_{k=1}^{K} \frac{1}{|C_k|} \sum_{i,l \in C_k} d^2(\mathbf{x_i} - \mathbf{x_l}) \right\}$$

Clustering Criterion

The goal is to have a minimal "within-cluster variation", i.e. the elements within a cluster should be as similar as possible.

An equivalent criterion is to minimize WSSD: the within-cluster sum of squared distances, that is, distances of the observations in each cluster and their corresponding centroids:

$$min \left\{ \sum_{k=1}^{K} \sum_{i \in C_k} d^2(\mathbf{x_i} - \mathbf{g_k}) \right\}$$

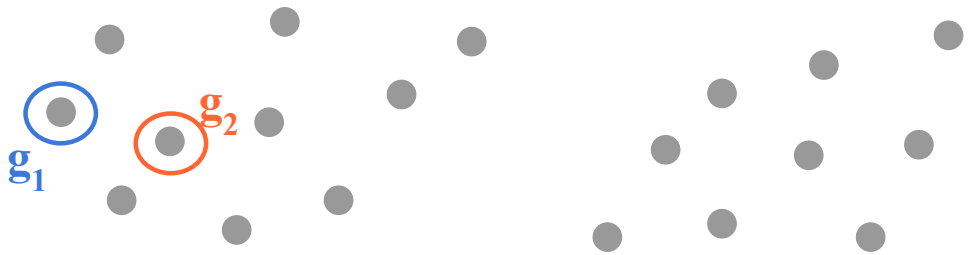
K-Means Algorithm

Initialization: Randomly select K centers $\mathbf{g_k}$

Do for 1:max_iter

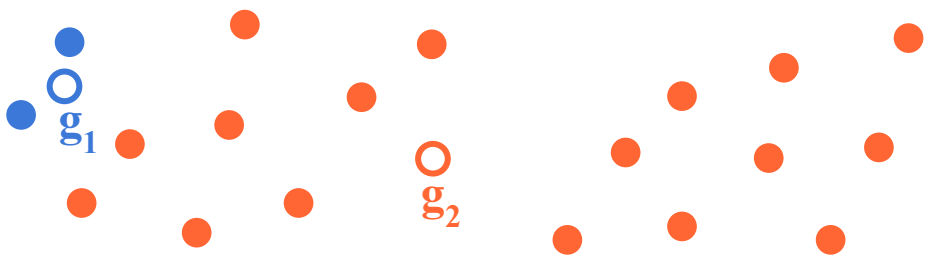
- Assign every individual to the closest center
- Definition of the new partition
- Update centroids of every class
- Stop when:
  - the old centroids and the new centroids are sufficiently similar
  - or the configuration of clusters does not change
  - or criterion (e.g. WSSD) doesn't decrease anymore
  - or the maximum number of iterations is reached

K-Means



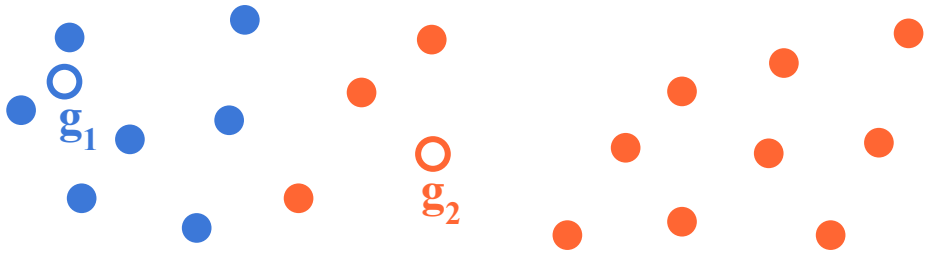
Start with random centroids

Assign observations to closest centroid

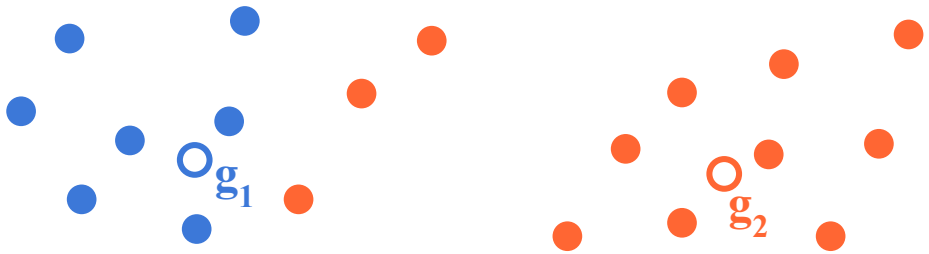


Recalculate centroids

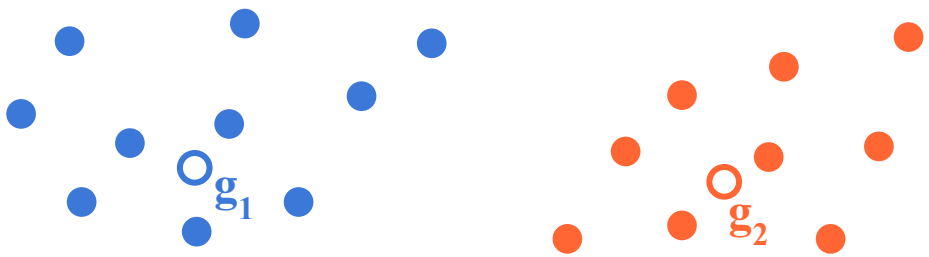
K-Means



Reassign observations to closest centroid



Recalculate centroids



Reassign observations to closest centroid

MacQueen's K-Means (1967)

Initialization: Randomly select K centers $\mathbf{g_k}$

Do for 1:max_iter

- Assign an individual to the closest center
- Update center for each new individual introduced into the cluster
- Stop when:
  - the old centroids and the new centroids are sufficiently similar
  - or the configuration of clusters does not change
  - or criterion (e.g. WSSD) doesn't decrease anymore
  - or the maximum number of iterations is reached

MacQueen's K-Means (1967)

In MacQueen's K-means, the centroid of each group is recalculated for each new object introduced into the group, instead of waiting for the assignation of all the objects before recalculating the centroids.

The convergence is faster and may even be completed in a single iteration, but the result depends on the order of the objects in the data set.

More about K-Means

What does K-Means optimize?

Assume that the n observations have masses m_i

Let $d^2(\mathbf{x_i}, \mathbf{g_k^s})$ be the squared distance between observation i and the centroid of the group k, at step s.

We focus on the within-cluster sum of squared distances:

$$\sum_{k=1}^{K} \left\{ \sum_{i \in C_k^s} m_i d^2(\mathbf{x_i}, \mathbf{g_k^s}) \right\}$$

What does K-Means optimize?

At the end of step s, the group C_k^s is formed of those observations that are closest to the centroid $\mathbf{g_k^s}$

The within-gropus spread at step s is given by:

$$\text{WSSD}(s) = \sum_{k=1}^{K} \left\{ \sum_{i \in C_k^s} m_i d^2(\mathbf{x_i}, \mathbf{g_k^s}) \right\}$$

where $\mathbf{g_k^s}$ is the centroid of cluster C_k

What does K-Means optimize?

From step s to $s+1$, we calculate new centroids $\mathbf{g_k^{s+1}}$

The **intermediate** within-gropus spread from step s to $s+1$ is given by:

$$\text{WSSD}^*(s \to s+1) = \sum_{k=1}^{K} \left\{ \sum_{i \in C_k^s} m_i d^2(\mathbf{x_i}, \mathbf{g_k^{s+1}}) \right\}$$

where $\mathbf{g_k^{s+1}}$ is the updated centroid of clsuter C_k

What does K-Means optimize?

At the end of step $s+1$, the group C_k^{s+1} is formed of those observations that are closest to the centroid $\mathbf{g_k^{s+1}}$

The within-gropus spread of step $s+1$ is given by:

$$\text{WSSD}(s+1) = \sum_{k=1}^{K} \left\{ \sum_{i \in C_k^{s+1}} m_i d^2(\mathbf{x_i}, \mathbf{g_k^{s+1}}) \right\}$$

where $\mathbf{g_k^{s+1}}$ is the centroid of cluster C_k

What does K-Means optimize?

It can be shown that:

$$\text{WSSD}(s) \geq \text{WSSD}^*(s) + \text{WSSD}(s+1)$$

- The objective criterion decreases (i.e. non-decreasing function).
- This guarantees convergence of the K-Means algorithm.
- It usually converges fast.

Local Optima

The K-means algorithm can get stuck in "local optima" and not find the best partition.

Hence, it is important to run the algorithm multiple times with random starting points to find a good solution.

- Choose the seeds "wisely"
- Extensions: K-medoids, Kohonen maps, ...

K-Means: local optima

Fast K-Means Algorithm

Quick and dirty

- Randomly select $\mathbf{g_k}$ centers
- Assign the first individual to the closest center
- Update the new center of the affected class
- Assign the second individual to its closes center
- Update the new center of the affected class
- ...

K-Prototypes Summary

K-Prototypes

- Assume there are K prototypes denoted by $\mathbf{p_k}$, $k = 1, \ldots, K$.
- Each object is assigned to one of the prototypes.
- Minimize a total within-cluster measure of spread.

Pros and Cons

Pros and Cons

Pros of partitioning methods:

- The main advantage of the partitioning methods is that their complexity is linear.
- Their execution time is proportional to the number of n objects (since the nK distances between objects are calculated at each step)
- The number of iterations needed to minimize the within-cluster sum of squares is generally small.
- These pros make partitioning methods attractive for large volumes of data.

Pros and Cons

Pros of partitioning methods:

- The final partition depends greatly on the initial choice of centroids
- Consequently, you don't have a global optimum, but a local optimum.
- The other major drawback is that the number of clusters K is fixed, and must be predetermined.
- Also, these methods are only good at detecting spherical forms. Even convex forms such as ellipses cannot be detected well if they are not sufficiently separated.

Bibliography

- **Modern Multivariate Statistical Techniques** by Julian Izenman (2008). *Chapter 12: Cluster Analysis*. Springer.
- **Data Mining and Statistics for Decision Making** by Stephane Tuffery (2011). *Chapter 9: Cluster Analysis*. Wiley.

French literature

- **Probabilites, analyse des donnees et statistique** by Gilbert Saporta (2006). *Chapter 11: Methodes de classification*. Editions Technip, Paris.
- **Statistique Exploratoire Multidimensionnelle** by Lebart et al (2004). *Chapter 2, section 2.1: Agregation autour des centres mobiles*. Dunod, Paris.
- **Approche pragmatique de la classification** by Nakache and Confais (2005). *Chapter 4: Classification par partition*. Editions Technip, Paris.
- **Statistique: Methodes pour decrire, expliquer et prevoir** by Michel Tenenhaus (2008). *Chapter 9: Analyses des proximites, des preferences et typologie*. Editions Technip, Paris.