Class Notes on K-means++ Clustering and Related Algorithms

Instructor - Amit Awekar

March 3 - 7, 2025

Contents

1 K-means++ Clustering

1.1 Introduction

K-means++ is an enhanced initialization algorithm for the standard k-means clustering method. By selecting initial centers with a probabilistic scheme based on squared distances, it improves both the convergence speed and the quality of the final clusters.

A good initialization can avoid poor local minima and reduce iterations.

1.2 Algorithm Steps

1. **Random Initialization:** Choose the first center c_1 uniformly at random from the data points.
2. **Distance Calculation:** For each data point x, compute its distance $D(x)$ to the nearest center already chosen.
3. **Probabilistic Selection:** Select the next center from the data points with probability proportional to $D(x)^2$, i.e.,

$$\Pr(x \text{ is chosen}) = \frac{D(x)^2}{\sum_{x' \in X} D(x')^2}.$$

4. **Repeat:** Continue Steps 2 and 3 until k centers are chosen.
5. **Standard k-means:** Run the standard k-means algorithm using the selected centers as initialization.

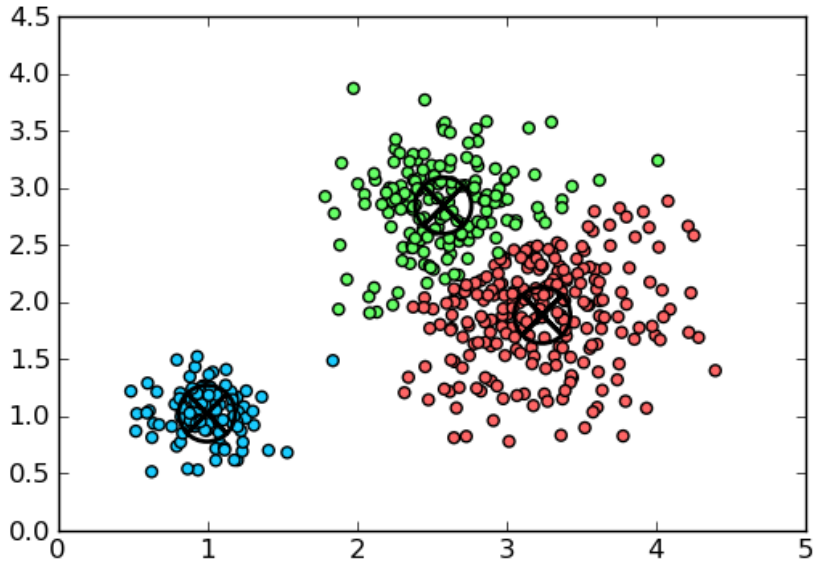

Figure 1: Visualization of the K-means++ initialization process.

1.3 Mathematical Formulation

The clustering loss is measured by the Sum of Squared Errors (SSE):

$$\text{SSE} = \sum_{i=1}^{n} \|x_i - c_{(x_i)}\|^2,$$

where $c_{(x_i)}$ is the center assigned to data point x_i. [1]

The probabilistic update in K-means++ plays a critical role by giving preference to points that are far from current centers, thereby promoting diversity among the chosen centers.

SSE Bound in K-means++

For a non-optimal representative a_0, it can be shown that:

$$\mathrm{SSE}(a_0) \leq 2\,\mathrm{SSE}_{\mathrm{opt}}(A).$$

Furthermore, the analysis of K-means++ shows that the expected SSE after initialization is bounded by:

$$\text{Expected SSE} \leq 8\,\mathrm{SSE}_{\mathrm{opt}}(A).$$

Due to the probabilistic selection, the approximation factor is no worse than $O(\log k)$.

1.4 Implications of the Probabilistic Update

The design of the probabilistic update ensures that:

- Centers are well spread out, reducing the chance of poor clustering.
- The algorithm is less sensitive to outliers compared to purely random initialization.
- There is a theoretical guarantee on the quality of the initialization relative to the optimal clustering.

2 Privacy Preserving K-means Clustering

2.1 Overview

Privacy preserving k-means is adapted for distributed settings where sensitive data cannot be shared openly. Consider a scenario with:

- n data points, each with r attributes.
- $r + 1$ trusted parties.

2.2 Algorithm Steps and Data Flow

1. **Local Representative Selection:** Each of the first r parties selects k representatives from their local data.
2. **Distance Matrix Computation:** Each party computes a distance matrix D_i between its data points and the k representatives. Here, D_i is of dimension $n_i \times k$, where n_i is the number of data points at party i.
3. **Secure Aggregation:** The computed distance matrices are securely transmitted to the $(r+1)^{\text{th}}$ party.
4. **Global Clustering:** The $(r+1)^{\text{th}}$ party aggregates the information and performs clustering to update the representatives.
5. **Feedback:** The updated representatives are sent back to the original parties for further local processing.

[1] A lower SSE indicates more compact and well-defined clusters.

This distributed approach ensures that sensitive raw data remains local while still enabling a collaborative clustering process.

Each local distance matrix D_i has dimensions $n_i \times k$.

3 Global K-means Clustering

3.1 Overview

Global k-means is an incremental algorithm that adds one cluster center at a time. Unlike standard k-means, which can be sensitive to initialization, Global k-means evaluates all data points as potential candidates for the new center.

3.2 Methodology

At each incremental step, the algorithm solves the following optimization problem:

$$\min_{c_{k+1}} \sum_{i=1}^{n} \min \left\{ \|x_i - c_j\|^2, \|x_i - c_{k+1}\|^2 \right\},$$

where c_j are the already chosen centers. This exhaustive evaluation ensures that each new center minimizes the overall clustering error optimally.

4 SSE Loss Calculations and Analysis in K-means++

4.1 SSE Loss for a Non-optimal Representative

When considering a representative a_0 that is not the true optimal center, the SSE loss can be approximated as:

$$\mathrm{SSE}(a_0) \approx 2\,\mathrm{SSE}_{\mathrm{opt}}(A).$$

4.2 Analysis for K-means++

In the context of K-means++ initialization, the analysis reveals:

$$4\,\mathrm{SSE}_{\mathrm{opt}}(A) + 4\,\mathrm{SSE}_{\mathrm{opt}}(A) = 8\,\mathrm{SSE}_{\mathrm{opt}}(A).$$

Nonetheless, the probabilistic approach ensures that the expected loss is within a $O(\log k)$ factor of the optimal SSE, providing a strong theoretical guarantee.

The logarithmic bound is crucial for understanding the efficiency gains of K-means++.

5 Elkan's K-means

5.1 Overview

Elkan's k-means algorithm introduces efficiency improvements to the standard k-means by leveraging the triangle inequality to reduce the number of distance calculations.

5.2 Mathematical Underpinnings

Elkan's method maintains upper and lower bounds for distances between data points and cluster centers:

- Let $u(x) = \|x - c_{(x)}\|$ denote the distance of point x to its assigned center.

- For any other center c_j, a lower bound $l(x, j)$ is maintained such that:

$$l(x, j) \leq \|x - c_j\|.$$

Using the triangle inequality, these bounds are updated when centers move, which in turn reduces the number of explicit distance calculations required during each iteration.

Efficiency Insight

Elkan's algorithm can greatly reduce computation time, particularly in high-dimensional spaces where distance calculations are expensive.

6 Anti-clustering Problem Definition

6.1 Definition

Anti-clustering is concerned with partitioning a dataset into groups that are as heterogeneous as possible, contrary to standard clustering where homogeneity is the goal. Formally, given a dataset $X = \{x_1, x_2, \ldots, x_n\}$, the objective is to partition X into k groups $\{G_1, G_2, \ldots, G_k\}$ so as to:

$$\max_{G_1,\ldots,G_k} \sum_{j=1}^{k} \text{Var}(G_j)$$

subject to:

$$\bigcup_{j=1}^{k} G_j = X \quad \text{and} \quad G_i \cap G_j = \emptyset \text{ for } i \neq j.$$

6.2 Applications

Anti-clustering is useful in contexts such as:

- Experimental design, where diverse groups are required.
- Portfolio diversification, ensuring broad representation across groups.
- Forming balanced groups in educational or organizational settings.

7 Summing Up the Week

This week, we have covered:

- The K-means++ algorithm, its probabilistic initialization, mathematical formulation, and theoretical guarantees.
- The framework for Privacy Preserving K-means clustering, detailing the data flow, matrix dimensions, and secure aggregation.
- A brief overview of Global k-means and its incremental approach.
- Detailed SSE loss analyses for K-means++ and the significance of the $O(\log k)$ guarantee.
- Elkan's K-means algorithm and its efficient use of the triangle inequality.
- The problem definition of Anti-clustering, highlighting its objectives and applications.

These detailed notes are intended to serve as a comprehensive reference for exam preparation and further study into advanced clustering methodologies.