PCA-guided search for K-means

Geometric Data Analysis - Project Report

Borachhun YOU
École Normale Supérieure Paris-Saclay
borachhun.you@ens-paris-saclay.fr

ABSTRACT

The chosen topic for this project revolves around an initialization method for K-means clustering algorithm that involves with principal component analysis (PCA). This report primarily focuses on a technique proposed by Xu et al. in their paper titled "PCA-guided search for K-means", in which the technique was shown to yield a better solution than the classic approach with random initialization. In addition, an experiment is conducted in this report to compare the results obtained from the proposed technique and other methods of K-means, and an extension of the studied method is also discussed.

1 INTRODUCTION

In the field of unsupervised learning, clustering techniques play an essential role in grouping complex, unlabeled, high-dimensional data. Among the techniques, K-means is one of the most widely used algorithms, mainly because it is simple to understand and implement. However, the classic approach of K-means clustering, which was first proposed by Stuart Lloyd, has been shown to converge quickly and easily from the initialized cluster centroids to one of the many local solutions due to the non-convexity of the clustering problem.

One way to achieve a better solution from the algorithm is to improve the technique of initializing the cluster centroids. There have been numerous research papers demonstrating different techniques of such that yield better results compared to the original approach.

This report focuses on one initialization method proposed by Xu et al. in their paper titled "PCA-guided search for K-means" [3]. The technique, which involves with the concept of principal component analysis (PCA), was proven to be effective by showing a close relationship between the global solution of the K-means problem and the PCA subspace.

This report begins by recalling the K-means algorithm and PCA (section 2 and 3), before proving how K-means solution can be found in the PCA subspace (section 4). Then, the algorithm of the proposed technique is given in section 5, and after that an experiment is conducted on the technique as well as other versions of K-means with various datasets in order to compare the performances (section 6). Finally, an extension to the proposed method is discussed, which leads to an improvement of the method (section 7).

2 K-MEANS

With the K-means clustering problem, we are given a set of n data points $x_1, ..., x_n \in \mathbb{R}^p$ and an integer K. We wish to choose K centers $\mu_1, ..., \mu_K \in \mathbb{R}^p$ of cluster $C_1, ..., C_K$ respectively so as to minimize the following function:

$$J_{\text{K-means}} = \sum_{j=1}^{K} \sum_{x_i \in C_j} \|x_i - \mu_j\|^2 \qquad (1)$$

The most common algorithm used to solve the given problem is the Lloyd's algorithm, which is as follows:

- Choose K data points arbitrarily as the initial centers $\mu_1, ..., \mu_K$
- For each data point, assign x_i to cluster C_j that has the closest center μ_j
- Find the new center of cluster C_j by computing

$$\mu_j = \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i$$

- Repeat the second and third step until convergence.

Here, in the first step, the initial centers are chosen uniformly at random from $x_1, ..., x_n$, which has been shown to converge easily to a local solution. We will see a better initialization method in the next sections.

3 PRINCIPAL COMPONENT ANALYSIS (PCA)

In the area of dimensionality reduction, Principal Component Analysis or PCA is one of the well-known techniques for reducing the dimension of datasets. It is done by applying a linear transformation to the data into a new low-dimensional space, called PCA subspace, where most of the variation in the data can be described with the fewer dimensions.

Let $X = (x_1, ..., x_n) \in \mathbb{R}^{p \times n}$ be a column matrix of n data points of dimension p. We apply PCA to the data points by first calculating the covariance matrix of X, which is given by:

$$\text{cov}(X) = \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})(x_i - \bar{x})^T$$

We then compute for the eigenvectors corresponding to k largest eigenvalues $\lambda_1, ..., \lambda_k$ of the covariance matrix. We denote $U = (u_1, ..., u_k)$ a column matrix of the eigenvectors. The transformed dataset in the PCA subspace is given by

$$Y = (y_1, ..., y_n) = U^T X \in \mathbb{R}^{k \times n} \qquad (2)$$

4 K-MEANS AND PCA RELATIONSHIP

From the works of [2] and [4], Xu et al. showed a close relationship between K-means and PCA. Specifically, they demonstrated how continuous solution to K-means problem is related to PCA and the existence of K-means solution in PCA subspace.

4.1 Continuous solution to K-means

The solution of K-means is described by the cluster membership indicator matrix $H = (h_1, ..., h_K) \in \mathbb{R}^{n \times K}$ where $H_{ij} = \frac{1}{\sqrt{n_j}}$ (with column normalization) if x_i belongs to cluster C_j that has n_j elements, or $H_{ij} = 0$ otherwise. Analysis cannot be done on this discrete solution of H. However, it is possible to analyze a continuous relaxed version of the solution, i.e. the elements of the indicator matrix are relaxed from discrete values into continuous values in $(-1, 1)$.

From the result of PCA in (2), we define

$$\tilde{Y} = \left(U\Sigma^{-1}\right)^T X$$

where $\Sigma = \text{diag}(\sigma_1, ..., \sigma_k)$ and $\sigma_k = \sqrt{\lambda_k}$. The difference between Y and $\tilde{Y}$ is that $\text{cov}(Y) = \text{diag}(\lambda_1, ..., \lambda_k)$ and $\text{cov}(\tilde{Y}) = I$

Theorem 4.1. *The optimal solution of the relaxed cluster indicators H for K-means clustering on data X is given by $\tilde{Y}^T R^T$, where R is an unknown rotation matrix of size $K \times K$ satisfying $RR^T = R^T R = I$.*

Proof. The objective function of the K-means problem (1) can be written as:

$$\begin{aligned} J_{\text{K-means}} &= \sum_{j=1}^{K} \sum_{x_i \in C_j} \left(\|x_i\|^2 - 2\mu_j^T x_i + \|\mu_j\|^2 \right) \\ &= \sum_{i=1}^{n} \|x_i\|^2 - \sum_{j=1}^{K} \mu_j^T \sum_{x_i \in C_j} (2x_i - \mu_j) \end{aligned}$$

Also, the cluster centroid can be written in terms of cluster indicators:

$$\mu_j = \frac{1}{n_j} \sum_{x_i \in C_j} x_i = \frac{1}{\sqrt{n_j}} X h_j \tag{3}$$

Therefore, we obtain:

$$\begin{aligned} J_{\text{K-means}} &= \sum_{i=1}^{n} \|x_i\|^2 - \sum_{j=1}^{K} \mu_j^T (2n_j\mu_j - n_j\mu_j) \\ &= \sum_{i=1}^{n} \|x_i\|^2 - \sum_{j=1}^{K} n_j \mu_j^T \mu_j \\ &= \sum_{i=1}^{n} \|x_i\|^2 - \sum_{j=1}^{K} \sqrt{n_j} \mu_j^T \mu_j \sqrt{n_j} \\ &= \sum_{i=1}^{n} \|x_i\|^2 - \sum_{j=1}^{K} h_j^T X^T X h_j \end{aligned}$$

Since the first term is constant, minimizing $J_{\text{K-means}}$ is equivalent to maximizing:

$$\begin{aligned} \sum_{j=1}^{K} h_j^T X^T X h_j &= \text{Tr}(H^T X^T X H) = \text{Tr}(H^T X^T X H R R^T) \\ &= \text{Tr}(R^T H^T X^T X H R) = \text{Tr}(\tilde{H}^T X^T X \tilde{H}) \end{aligned}$$

where $\tilde{H} = HR$ and R is a rotation matrix satisfying $RR^T = I$. Additionally, it is possible to verify that $H^T H = \tilde{H}^T \tilde{H} = I$, and thus the K-means problem becomes

$$\begin{aligned} &\max_{\tilde{H}} Tr(\tilde{H}^T X^T X \tilde{H}) \\ &\text{s.t. } \tilde{H}^T \tilde{H} = I \end{aligned}$$

By relaxing the elements of H to be continuous real numbers in $(0, 1)$, the spectral relaxation of the discrete optimization is obtained [4]. With this relaxation, the optimal solution $\tilde{H}^*$ of the maximization problem is given by the K eigenvectors $v1, ..., v_K$ of the Gram matrix $X^T X$ associated with the K largest eigenvalues $\xi_1, ..., \xi_K$. Let $V = (v1, ..., v_K)$, $\tilde{H}^* = V$ or $H^* = VR^T$.

With singular value decomposition,

$$X = \sum_{i=1}^{r} \sigma_i u_i v_i^T = U\Sigma V^T \tag{4}$$

where r is the rank of X. It can then be verified

$$\tilde{Y} = \left(U\Sigma^{-1}\right)^T X = V^T = \tilde{H}^{*T} = R^T H^{*T}$$

and $\sigma_k = \sqrt{\lambda_k} = \sqrt{\xi_k}$.

□

4.2 K-means solution in PCA subspace

Suppose that $c_1, ..., c_K$ are the optimal cluster centroids obtained from K-means. These centroids span a subspace called *cluster centroid subspace*, where the global optimal solution of K-means can be found. As for the PCA subspace, it is spanned by the eigenvectors $u_1, ..., u_k$.

Theorem 4.2. *Cluster centroid subspace is identical to PCA subspace.*

Proof. Let $C = (c_1, ..., c_K)$. The orthonormal basis of the cluster centroid space is $\tilde{C} = C(C^T C)^{-\frac{1}{2}}$, and the projection operator of the space is $\tilde{C}\tilde{C}^T$. Now we would like to prove that $\tilde{C}\tilde{C}^T = UU^T$.

From (3), we can write

$$C = (c_1, ..., c_K) = X\left(\frac{h_1}{\sqrt{n_1}}, ..., \frac{h_K}{\sqrt{n_K}}\right) = XHN^{-\frac{1}{2}}$$

where $N = \text{diag}(n_1, ..., n_K)$. Therefore,

$$\begin{aligned} \tilde{C}\tilde{C}^T &= C(C^T C)^{-1} C^T \\ &= XHN^{-\frac{1}{2}}(N^{-\frac{1}{2}} H^T X^T X H N^{-\frac{1}{2}})^{-1}(XHN^{-\frac{1}{2}})^T \\ &= XH(H^T X^T X H)^{-1}(XH)^T \\ &= X\tilde{H}(\tilde{H}^T X^T X \tilde{H})^{-1}(X\tilde{H})^T \end{aligned}$$

From theorem 4.1, the solution of K-means is $HR = \tilde{H} = V$, and thus $\tilde{C}\tilde{C}^T = XV(V^T X^T XV)^{-1}(XV)^T$.

From (4), since $V^T V = I$, $XV = U\Sigma V^T V = U\Sigma$, and as a result

$$\tilde{C}\tilde{C}^T = U\Sigma(\Sigma^T U^T U\Sigma)^{-1}(U\Sigma)^T = UU^T$$

□

Theorem 4.1 and 4.2 explain that the optimal solution of K-means lies in the k-dimensional PCA subspace. The main idea here is that we can search for the optimal solution in the much smaller k-dimensional PCA subspace rather than the original p-dimensional full space, and it is certain that the correct solution exists within this smaller space. Although it is not guaranteed to find the true global solution, the obtained solutions are usually better compared to the solutions from the full data space, measured with the objective function of the K-means problem.

5 PCA-GUIDED SEARCH FOR K-MEANS

The algorithm of the PCA-guided method of K-means proposed by Xu et al. is as follows:

- Apply PCA on the original dataset
- Perform K-means clustering with random initialization in the PCA subspace
- Use the cluster membership obtained in the previous step to construct initial cluster centroids in the full space
- Perform K-means clustering in the full space.

In the given algorithm, it can be observed that the majority of iterations of the algorithm are spent in the PCA subspace (second step) because in the third step the initially defined cluster centroids are already close to a local minima, and therefore a relatively small number of iterations take place in the full space. This is beneficial since computations are faster in the low-dimensional PCA subspace than in the full space.

6 EXPERIMENT

This section demonstrates the performance of the proposed algorithm by applying it on multiple datasets against other versions of K-means. Here, 3 datasets are used to compare the performances, two of which were also used in the original paper with slightly different configurations:

- *AT&T face dataset* contains grayscale images of 40 distinct people, each with 10 different images. The images are downscaled from the original size of 92×112 pixels to 64×64 pixels, and then flattened into vectors of 4096 dimensions.
- *Coil-20 dataset* has two available versions, and the one that is used here is the processed version where backgrounds are removed from objects. It consists of grayscale images of 20 objects, and for each object there are 72 images taken at different rotation angles. Each image has a size of 128×128 pixels, which is then flattened into a vector.
- *EMNIST dataset*, specifically the *ByClass* split, contains 28-by-28-pixel images of handwritten digits and characters. For this experiment, only the 26 capital letters of English are picked, and for each character the first 80 images are chosen.

In addition to the datasets, different versions of K-means that are used to compare with the proposed method are the original K-means and one of its variants called K-means++. In the initialization step, the original approach assigns K random data points as initial centers of the K clusters. In contrast, K-means++, which was proposed by Arthur and Vassilvitskii [1], generates the initial cluster centroids as follows:

- Randomly assign a data point as the first center
- Choose the next center from the data points where each data point has a probability of $\frac{D(x_j)^2}{\sum_{i=1}^{n} D(x_i)^2}$, $D(x_j)$ being the shortest distance from x_j to the closest center that has already been chosen
- Repeat the second step until a total of K centers have been chosen.

As for the implementation of the algorithms, the K-means function from Scikit-learn library is used, which can take initialized centers as inputs. The random initialization has already been implemented in the function, whereas for K-means++, the initialization step is programmed from scratch. The same is true for the proposed method but with the help of the PCA function from the same library. Additionally, the dimension of PCA is set to equal the number of clusters.

To compare the performances, all methods are run 1000 times, and the objective function of the K-means problem (or inertia) (1) is computed at the end of each run. Figure 1, 2 and 3 show the comparison of the 3 methods on the 3 datasets in terms of inertia, which is sorted in descending order. Table 1, 2 and 3 provide further information, including execution time and average inertia of the runs.

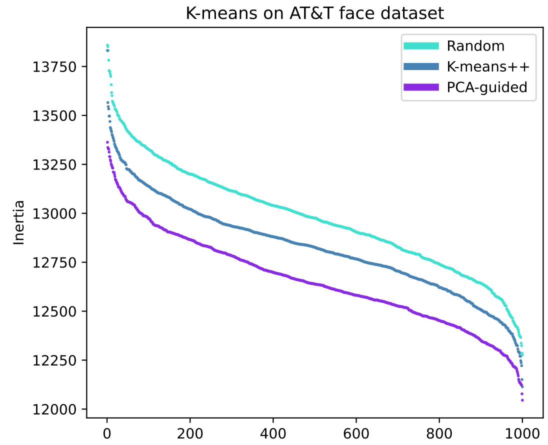

Figure 1: Results of K-means on AT&T face dataset

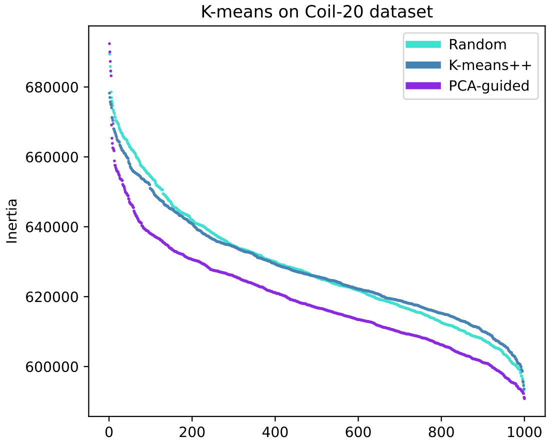

Figure 2: Results of K-means on Coil-20 dataset

Based on figure 1, 2 and 3, the PCA-guided algorithm outperforms the other 2 algorithms on all 3 datasets by giving the smallest inertia in all runs. Meanwhile, K-means++, which is an improved variant of the regular K-means, struggles to beat the original version with the Coil-20 and EMNIST letter dataset.

With table 1, 2 and 3, the same can be said about the performance of the PCA-guided algorithm as it beats the other algorithms

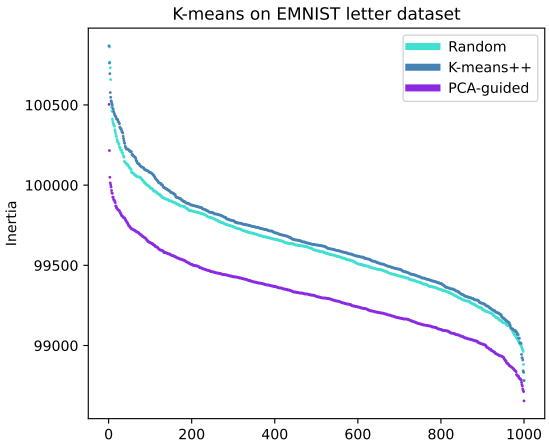

Figure 3: Results of K-means on EMNIST letter dataset

Table 1: Results of K-means on AT&T face dataset

Method	Time (s)	Lowest inertia	Average inertia
Random	74.98	12276.11	12977.00
K-means++	2250.73	12113.56	12825.43
PCA-guided	143.50	12045.48	12656.14

Table 2: Results of K-means on Coil-20 dataset

Method	Time (s)	Lowest inertia	Average inertia
Random	1717.96	591033.69	628099.14
K-means++	6668.07	591132.63	628046.88
PCA-guided	2974.25	590793.00	619006.93

Table 3: Results of K-means on EMNIST letter dataset

Method	Time (s)	Lowest inertia	Average inertia
Random	190.46	98780.47	99602.93
K-means++	6447.41	98781.89	99645.65
PCA-guided	316.62	98654.73	99311.96

with the smallest lowest inertia and average inertia. In terms of the computation time, the classic K-means runs the fastest, which contrasts the results given in the original paper which stated that the PCA-guided method was the fastest. This is mainly due to the usage of existing K-means function from Scikit-learn which may have been optimized, and the classic K-means does not have any computation at the initialization step unlike the other two.

7 EXTENSION

The proposed PCA-guided search K-means algorithm performs K-means in the PCA subspace in order to initialize the centroids in the full space. Specifically, the K-means that is performed is the standard K-means with random initialization in the subspace. As mentioned in previous sections, there exist many different initialization methods for K-means that improve the solution. By using such methods instead of the standard random initialization, it is possible to obtain a better solution in the PCA subspace and subsequently a better solution in the full data space.

This section of the report explores the possibility of improving the result from the proposed algorithm by replacing the random initialization of K-means in the PCA subspace with the already mentioned K-means++, a more careful method of choosing initial centroids. By combining the existing implementations, the new method is applied on the 3 previous datasets the same way as in section 6 to compare the performance with the original proposed technique. The obtained results and comparisons are shown in figure 4, 5, 6 and table 4.

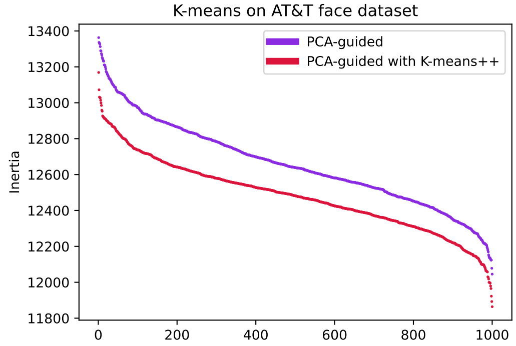

Figure 4: Results of K-means on AT&T face dataset

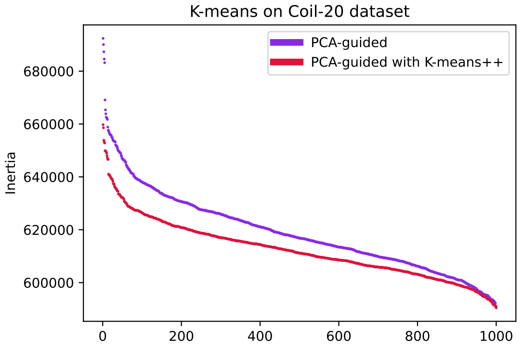

Figure 5: Results of K-means on Coil-20 dataset

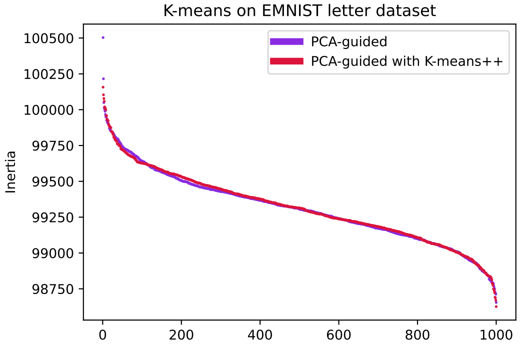

Figure 6: Results of K-means on EMNIST letter dataset

Table 4: Results of PCA-guided method with K-means++

Dataset	Time (s)	Lowest inertia	Average inertia
AT&T face	3087.46	11864.09	12480.19
Coil-20	5305.18	590459.63	612388.30
EMNIST letter	6683.76	98625.78	99316.70

According to figure 4, 5 and 6, the modified method outperforms the original proposed method in all 1000 runs for AT&T face dataset and Coil-20 dataset. With the EMNIST letter dataset, both methods have approximately the same performance. The same can be said based on table 4 in terms of the number of lowest inertia and average inertia. It can also be observed that the new method takes longer to execute compared to the original proposed method.

8 CONCLUSION

Throughout this report, the PCA-guided method of K-means proposed by Xu et al. [3] has been studied and demonstrated to lead to a better solution by showing a close relationship between K-means and PCA. A practical experiment has also been conducted to verify the improved result given by the proposed method. Additionally, an extension to the method has been looked into which lead to an improvement over the original proposed algorithm.

REFERENCES

[1] David Arthur and Sergei Vassilvitskii. 2007. K-Means++: The Advantages of Careful Seeding. *Proc. of the Annu. ACM-SIAM Symp. on Discrete Algorithms* 8, 1027–1035. https://doi.org/10.1145/1283383.1283494

[2] Chris Ding and Xiaofeng He. 2004. K-Means Clustering via Principal Component Analysis. In *Proceedings of the Twenty-First International Conference on Machine Learning* (Banff, Alberta, Canada) *(ICML '04)*. Association for Computing Machinery, New York, NY, USA, 29. https://doi.org/10.1145/1015330.1015408

[3] Qin Xu, Chris Ding, Jinpei Liu, and Bin Luo. 2015. PCA-guided search for K-means. *Pattern Recognition Letters* 54 (2015), 50–55. https://doi.org/10.1016/j.patrec.2014.11.017

[4] Hongyuan Zha, Xiaofeng He, Chris Ding, Ming Gu, and Horst Simon. 2001. Spectral Relaxation for K-means Clustering. In *Advances in Neural Information Processing Systems*, T. Dietterich, S. Becker, and Z. Ghahramani (Eds.), Vol. 14. MIT Press. https://proceedings.neurips.cc/paper_files/paper/2001/file/d5c186983b52c4551ee00f72316c6eaa-Paper.pdf