1 Technical Background

K-means clustering is an unsupervised Machine Learning (ML) protocol whose main goal is to classify datasets into K clusters. The term "k-means" was first used by James MacQueen in 1967, and published as a journal article in 1982. K-means clustering can minimize within cluster variances, the problem is computationally difficult (NP-hard) and the time complexity of the algorithm is specified by O(LNpK), where L = number of iterations taken to form clusters, N = number of datapoints, p = dataset dimension, K = centroids.

Clustering is the process of dividing the datasets into groups such that points in the same group are as similar as possible, and points in different groups are as dissimilar as possible. The k-means algorithm finds groups with the number of groups represented by the variable K. The algorithm works in an iterative manner to assign each data point to one of the k groups based on the features that are provided.

Fig.1 give an example of how K-means to divide the similar datasets into one group, in this case K=3

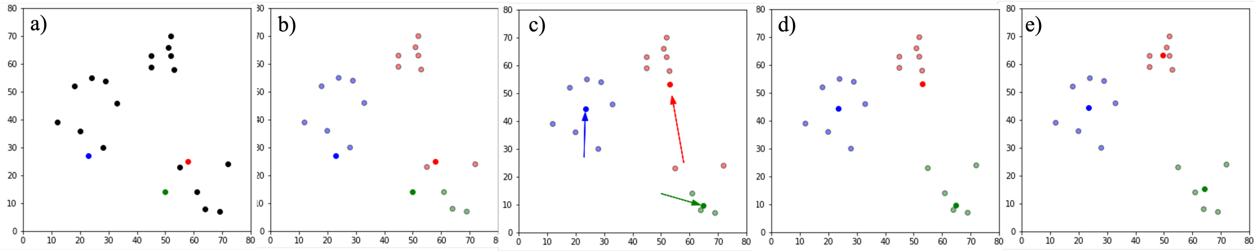

Figure 1: Some random 2D training data points are plotted with black dots in a), three random points are chosen to be the cluster centroids. b) Assign the training data points to the selected clusters. c) Calculate average and update the cluster centroids. d)Repeat data assignment in figure b). e) Repeat figure b), c) and d) until final clusters e) are reached.

A general way of doing the K-means algorithm is usually summarized in the following steps:

- Initialize uniformly distributed cluster centroids c_j which are randomly selected from the data setsInitialize uniformly distributed cluster centroids c_j which are randomly selected from the data sets.
- For each c_j, the chosen metric is evaluated. For example, in simpler cases, the distance between each training data point and selected cluster centroids c_j is evaluated.
- New clusters C_j are defined by assigning training data to the nearest cluster centroids c_j.
- New cluster centroids are updated
- Repeat steps 3 to 5 are repeated depending on the resource constraints

In the second step, the Euclidean distance between cluster centroids and training data point need to be efficiently sampling and estimating, however it becomes exponentially difficult as number of variables (features) of each data point increase. Therefore, implement a computationally cheap quantum algorithm to calculate Euclidean distance is helpful.

In this project, our team proposed a hybrid algorithm which contains classical calculation to calculate cluster centroids and assigns features and using quantum circuit to compute Euclidean distance.

2 Hybrid K-means clustering

The hybrid K-means clustering algorithm contains quantum circuit to calculate the Euclidean distance between the cluster centroids and the training data points, and classical method to assign and update cluster centroids. The work flow of the hybrid K-means clustering is presented in Fig. 2

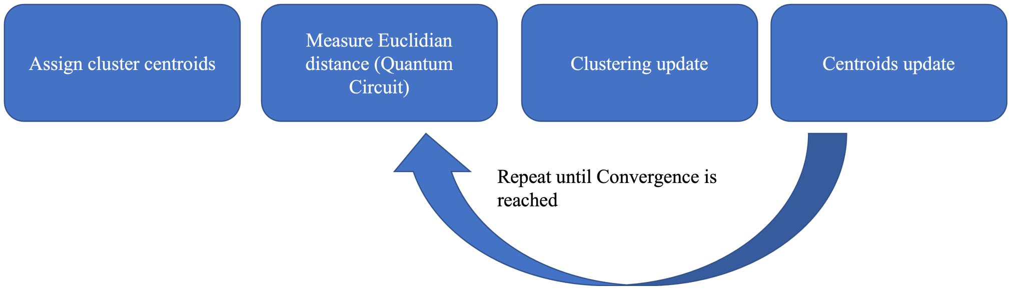
Assign cluster centroids
Measure Euclidian distance (Quantum Circuit)
Clustering update
Centroids update
Repeat until Convergence is reached

Figure 2: : A flow chart of how the hybrid K-means clustering algorithm works in our system.

The first step is to randomly assign the cluster centroids, however in order to perform quantum computing, we need to convert the classical centroids into quantum states and here we use the amplitude embedded method. Suppose we have a centroid x with n dimensions (features) which can be written in the form of a vector $x = [x_1, x_2, x_3, ...x_n]$ where $x_j (j \in n)$ is the feature. With the the amplitude embedding, the classical information can be mapped into Hilbert Space in a form of

$$|a\rangle = \frac{1}{|X|}\sum_{i=1}^{n} x_i |x_i\rangle \tag{1}$$

where $|X|$ is the normalization factor, $|x_i\rangle$ is the quantum state which is in bit string configuration and x_i is the features of the cluster centroids. We can also apply the similar transformation technique to the training data points y and we have

$$|b\rangle = \frac{1}{|Y|}\sum_{i=1}^{n} y_i |y_i\rangle \tag{2}$$

After forming the quantum states the question of finding Euclidian distance between two classical points are mapped into finding the distance between two quantum states $|a\rangle$ and $|b\rangle$. Firstly, we generate two with $|a\rangle$, $|b\rangle$ and an ancillary qubit

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle \otimes |a\rangle + |1\rangle \otimes |b\rangle) \tag{3}$$

$$|\phi\rangle = \frac{1}{Z}(|a| |0\rangle + |b| |1\rangle) \tag{4}$$

where $Z = \sqrt{|a|^2 + |b|^2}$ is the normalization constant. Next step, we introduce the **swap test** [2] to computing the distance. The swap test aims to apply a controlled swap gate to $|\psi\rangle$ and $|\phi\rangle$ which allows us to experimentally determine the overlap between the two states $|\langle\psi|\phi\rangle|^2$, this will be integral in calculating the Euclidean distance. The initial state for quantum computing is

$$|S_0\rangle = |0\rangle \otimes |\psi\rangle \otimes |\phi\rangle \tag{5}$$

Then we apply a Hadamard gate on the ancillary qubit to make a superposition state

$$|S_1\rangle = \frac{1}{\sqrt{2}}(|0\rangle \otimes |\psi\rangle \otimes |\phi\rangle + |1\rangle \otimes |\psi\rangle \otimes |\phi\rangle) \tag{6}$$

The controlled SWAP gate is then applied to swap $|\psi\rangle$ and $|\phi\rangle$ conditioned on the ancillary qubit, this result in

$$|S_2\rangle = \frac{1}{\sqrt{2}}(|0\rangle \otimes |\psi\rangle \otimes |\phi\rangle + |1\rangle \otimes |\phi\rangle \otimes |\psi\rangle) \tag{7}$$

Another Hadmard gate is placed on the ancillary qubit at the end gives the result

$$|S_3\rangle = \frac{1}{2}|0\rangle \otimes (|\psi\rangle \otimes |\phi\rangle + |\phi\rangle \otimes |\psi\rangle) + \frac{1}{2}|1\rangle \otimes (|\phi\rangle \otimes |\psi\rangle - |\psi\rangle \otimes |\phi\rangle) \tag{8}$$

Finally, we measure state $|S_3\rangle$ with the ancillary qubit $|0\rangle$, the probability $P(0)$ is

$$P(0) = |\langle 0|S_3\rangle|^2 = \frac{1}{2} + \frac{1}{2}|\langle \psi|\phi\rangle|^2 \tag{9}$$

The final Euclidean distance is

$$|a-b|^2 = Z(4P(0)-2) \tag{10}$$

Reference

[1] A. Sarma, R. Chatterjee, K. Gili, and T. Yu, "Quantum unsupervised and supervised learning on superconducting processors," Quantum Information and Computation 20 (2019), 10.48550/arXiv.1909.04226.

[2] M. Kang and J. Heo and S. Choi and S. Moon and S. Han (2019), Implementation of SWAP test for two unknown states in photons via cross-Kerr nonlinearities under decoherence effect, Scientific Reports Vol. 9., 10.1038/s41598-019-42662-4.