Balanced K-Means Clustering on an Adiabatic Quantum Computer

Applied Quantum Machine Learning Project

Politecnico di Milano

May 19, 2021

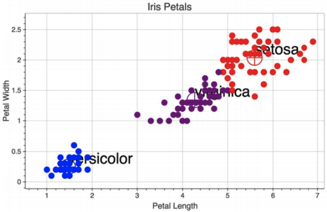
Authors:
Pierriccardo Olivieri
Francesco Piro
Matteo Sacco

Professors:
Cremonesi Paolo
Alessandro Luongo
Maurizio Ferrari Dacrema

Outline

Advantages over classical approaches

- Better targets the global solution of the training problem
- Better theoretic scalability on large datasets



Outline

- QUBO formulation and theoretical analysis
- Empirical Analysis
- Conclusions and considerations

Lloyd's algorithm

- Complexity $O(Nkdi)$ [13]
  - N number of data points
  - k number of clusters
  - d number of features
  - i number of iterations before the algorithm converges

Scikit-learn implementation

- Complexity $O(Nkd)$ [18]

[13] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A K-Means clustering algorithm" Applied Statistics

[18] "Scikit-learn: Machine learning in python," J. Mach. Learn. Res.

Malinen et al.

- Complexity $O(N^3)$ [13]

Algorithm 1. Balanced k-means
Input: dataset X, number of clusters k
Output: partitioning of dataset.

```
Initialize centroid locations C^0.
t ← 0
repeat
   Assignment step:
         Calculate edge weights.
         Solve an Assignment problem.
   Update step:
         Calculate new centroid locations C^{t+1}
   t ← t + 1
until centroid locations do not change.
Output partitioning.
```

[21] Malinen, Mikko. (2014). Balanced K-Means for Clustering.

QUBO formulation

$$\min_{z \in \mathbb{B}^M} z^T A z$$

$$X = \{x_1, x_2, \ldots, x_N\}$$

$$\Phi = \{\phi_1, \phi_2, \ldots, \phi_k\} \, .$$

$$\min_{\Phi} \sum_{j=1}^{k} \sum_{x \in \phi_j} \|x - \mu_j\|^2$$

$$\min_{\Phi} \sum_{j=1}^{k} \frac{1}{2\,|\phi_j|} \sum_{x,y \in \phi_j} \|x - y\|^2$$

$$\min_{\Phi} \sum_{j=1}^{k} \sum_{x,y \in \phi_j} \|x - y\|^2$$

Distance matrix: D

Assignment matrix: $\hat{W}$

$$\sum_{x,y \in \phi_j} \|x - y\|^2 = \hat{w}_j'^T D \hat{w}_j'$$

$$\min_{\hat{w}} \hat{w}^T \left(I_k \otimes D\right) \hat{w}$$

$$\alpha\left(\hat{w}_j'^T\hat{w}_j' - N/k\right)^2$$

$$\hat{w}_j'^T\alpha F\hat{w}_j'$$

$$F = 1_N - \frac{2N}{k}I_N$$

$$\beta\left(\hat{w}_i^T\hat{w}_i - 1\right)^2$$

$$\hat{w}_i^T\beta G\hat{w}_i$$

$$G = 1_k - 2I_k$$

$$\min_{\hat{w}} \hat{w}^T\left(I_k \otimes (D + \alpha F)\right)\hat{w}$$

$$\hat{w}^T Q^T\left(I_N \otimes \beta G\right)Q\hat{w}$$

$$\min_{\hat{w}} \hat{w}^T\left(I_k \otimes (D + \alpha F) + Q^T\left(I_N \otimes \beta G\right)Q\right)\hat{w}$$

Hyper-parameters tuning

$$\min_{\hat{w}} \hat{w}^T \left(I_k \otimes (D + \alpha F) + Q^T (I_N \otimes \beta G) Q \right) \hat{w}$$

$$\alpha = \frac{\max(D)}{2(N/k) - 1} \qquad \beta = \max(D)$$

Converting equation into a QUBO

$$\min_{\hat{w}} \hat{w}^T \left(I_k \otimes (D + \alpha F) + Q^T (I_N \otimes \beta G) Q \right) \hat{w}$$

$$\min_W \sum_{l=1}^{k} \sum_{j=1}^{N} \sum_{i=1}^{N} \sum_{m=1}^{d} w_{il} \left(x_{im} - x_{jm} \right)^2 w_{jl}$$

$$+\alpha \sum_{l=1}^{k} \sum_{j=1}^{N} \sum_{i=1}^{N} w_{il} f_{ij} w_{jl} + \beta \sum_{l=1}^{N} \sum_{j=1}^{k} \sum_{i=1}^{k} w_{li} g_{ij} w_{lj}$$

- Complexity $O(N^2 kd)$

Malinen et al.

- Complexity $O(N^3)$

Scikit-learn implementation

- Complexity $O(Nkd)$

Algorithms used for comparisons

- balanced quantum k-means (case study)
- balanced classical k-means (authors implementation)
- classical k-means scikit-learn implementation

classical version of the k-means (non balanced) is used since, due to the structure of the dataset, constitute a valid comparison.

Adjusted Rand Index (ARI)

- compare the similarity of two partitions of a dataset
- range from −1 to 1 (high values indicates the two partitions are similar)
- used to compare the target partitions to the partitions produced by clustering

Total Computing time in quantum approach

$$t = t_{QUBO_convertion} + t_e + t_a + t_{postprocessing} \tag{1}$$

- $t_{QUBOconvertion}$ time to convert the problem in QUBO
- t_e time to embed the QUBO on hardware
- t_a time to solve the QUBO (anealing time)
- $t_{postprocessing}$ extract clustering from binary solution

synthetic classification datasets created with *make_classification* (Scikit-learn)

Datasets structure

- **N** points
- **k** classes
- **1** cluster per class
- **d** features
- clusters centered on a d-dimensional hypercube (with side length 2.0)
- points generated from a normal dist. about their cluster center (std 1.0)
- each class made of $\frac{N}{k}$ points

Classical Machine

- 2.7 GHz Dual-Core Intel i5
- 8 GB 1.867 MHz DDR3 memory

Quantum Machine

- D-Wave 2000Q quantum computer
- 2048 qubits, 5600 inter-qubit connections

Technical Aspects

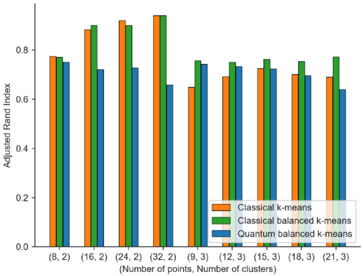
- quantum approach pre/post-processing done via the above classical machine
- quantum anealing operation perfomed 100 times for each experiment
- only ground state is used

Experiments Setup

- clustering quality of the 3 algorithms is compared
- each algorithm evaluated on different **problem types**
  - total of 9 problem types
  - defined by *(num. of points, num. of clusters)*
- for each problem type:
  - all the 3 algorithm evaluated on 50 **synthetic datasets**

Commenting Results for Quantum Approach

- performances drop for $k = 2$
  - less way to cluster means a local solution is more likely to be the correct one
- performances drop as the problem size increase
  - reflection of the quantum hardware



Limitations and Approximations I

Limitations faced

- **Variable limitation** D-Wave 2000Q qubit limitation for problems $Nk > 64$ var.
- **Qubit connectivity** limitation => higher embedding time

Approximations

- Quantum run time for larger problems $(Nk > 64)$
    - used to evaluate scalability of the Quantum Approach
    - measure $t_{QUBO_convertion}$ (measurable)
    - estimate embedding time t_e (from smaller problems)
    - estimate anealing time t_a (costant, averaging smaller problems)
    - measure $t_{postprocessing}$

Limitations and Approximations II

According to the embedding algorithms choosed by authors which scales quadratically in the number of binary variables of the QUBO

$$t_e = 1.887 \times 10^{-6}(Nk)^2 + 4.632 \times 10^{-6}(Nk) + 4.022 \times 10^{-4} \quad (2)$$

$$t_a = 0.03481 \pm 0.00008 \quad (3)$$

Experiments to assess scalability

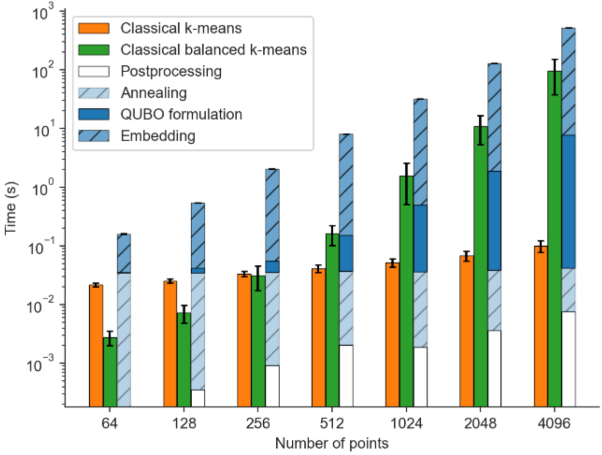
- baselines evaluated on the three variables:
  - N data points
  - k clusters
  - d features
- $\forall$ **problem type** baselines runned on 50 **synthetic datasets**

Scalability w.r.t. points (N) I

Setup and Considerations

- baselines evaluated on increasing **data points**
- fixed cluster $k = 4$ and features $d = 2$
- considerations:
  - quantum is outperformed (due to embedding time)
  - future embedding time improvements may surpass classical balanced ($N \geq 1024$)
  - classical k-means scales the best expected since its **time complexity** $O(Nkd)$ vs quantum balanced $O(N^2kd)$

Scalability w.r.t. points (N) II

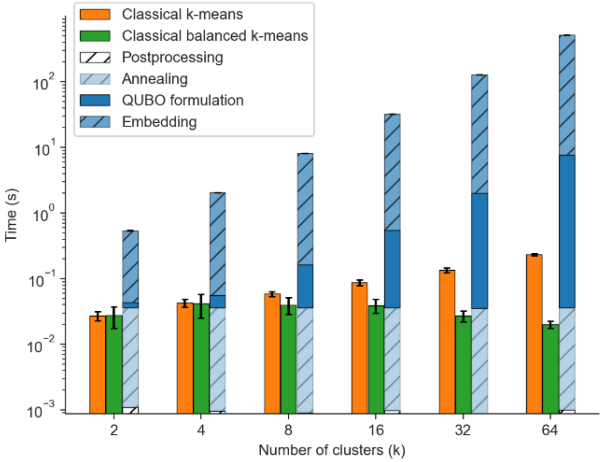


Scalability w.r.t. clusters (k) I

Setup and Considerations

- baselines evaluated on increasing **cluster size**
- fixed data points $N = 256$ and features $d = 8$
- considerations:
  - quantum scales worse on cluster size w.r.t. to other approaches
  - expected: third term on QUBO has $O(Nk^2)$ time complexity

Scalability w.r.t. clusters (k) II

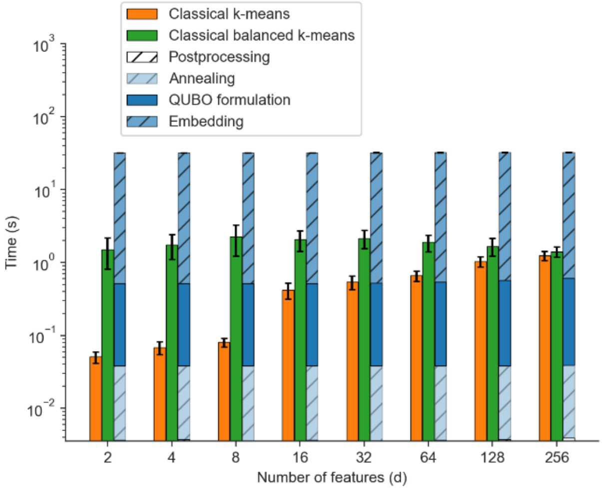


Scalability w.r.t. features (d) I

Setup and Considerations

- baselines evaluated on increasing **features number**
- fixed data points $N = 1024$ and cluster $k = 4$
- considerations:
  - quantum is the worse on time
  - quantum is promising in a future perspective, depending on embedding process optimizations
  - quantum approach scales better w.r.t. to classical *k-means* on d
  - expected: QUBO formulation only requires one comput. related to the dimension of the dataset
  - *classical balanced k-means* scales better in d w.r.t. to quantum approach
  - expected: *quantum balanced* $O(N^2kd)$ vs *classical balanced* $O(N^3)$

Scalability w.r.t. features (d) II



The Iris Dataset

- Reduced due to qubit limitations on modern hardware
- Pick N/k points from $2 \leq k \leq 3$ of the data set's classes

Experiments Run

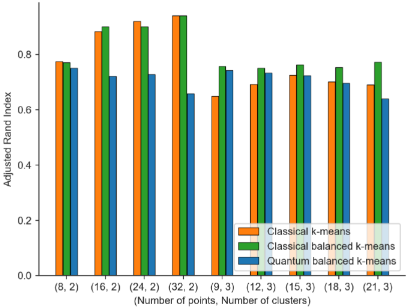
- All the 3 clustering algorithms were tested
- Experiments are run on 50 subsets of the dataset

Results

- $k = 2$
  - Trivial case, points are linearly separable
  - Classical algorithms perform better than quantum
  - Evident as the number of binary variables (Nk) increases

Clustering a Benchmark Data Set II

- $k = 3$
  - time to extract rforms **Scikit-Learn** implementation
  - Performance of the QA degrades as the problem size increases



Author's Conclusions

- Enhancements provided by adiabatic computers for solving **NP**-Hard or **NP**-Complete problems
- Promising result for Quantum Machine Learning
- The approach targets the global solution of the training problem **better** than the classic alternatives
- The **D-Wave 2000Q** machine
- Quantum approach partitions data with similar accuracy to the classical approaches
- The approach assumes viability as the quantum hardware improves

Future Works

- Bring the QUBO formulation to the generic k-means training problem
- Use elements of the approach to formulate quantum algorithms for similar clustering models
  - k-medoids clustering
  - fuzzy C-means clustering
- Cluster larger datasets

Can we cluster larger datasets on Advantage?

D-Wave 2000Q

- 2048 qubits
- 6,016 couplers
- 128,472 JJs

Advantage

- 5640 qubits
- 40,484 couplers
- 1,030,000 JJs

Thanks for your Attention