Supplementary Material: Probabilistic Sampling of Balanced K-Means using Adiabatic Quantum Computing

Jan-Nico Zaech
ETH Zurich
INSAIT, Sofia University
Sofia, Bulgaria

Martin Danelljan
ETH Zurich

Tolga Birdal
Imperial College
London

Luc Van Gool
ETH Zurich
INSAIT, Sofia University
Sofia, Bulgaria

1. Introduction

The main manuscript presents a novel approach for probabilistic clustering based on exploiting the probabilistic nature of adiabatic quantum computing (AQC). In the supplementary material, we provide additional details and that complement the main manuscript. Section 2 presents the detailed derivation of the energy function used in the manuscript. Section 3 discusses the optimization of the inference parameters to increase the AQC solver performance. Section 4 provides information on data generation for the synthetic and real-world datasets used in the experiments. Sections 5, 6 and 7 extend the clustering performance and calibration evaluation on synthetic data and the IRIS dataset respectively. In Section 8, we discuss failure cases encountered during the experiments. Finally, we outline the limitations of our approach in Section 9.

Overall, the supplementary material aims to clarify open questions from the main manuscript, provides additional insights into the proposed method and discusses its current limitations.

2. Energy Function Derivation

The following section shows the step-by-step derivation of the energy function used in this paper. As clusters are independent, the energy for each cluster can be computed separately $E(X|Z) = \sum_k E_k(X|Z)$, where X represents the data-points and Z is the assignment matrix with entry Z_{ki} assigning point x_i to cluster c_k. For a single cluster c_k, the energy can be further extended into the quadratic and linear terms as follows

$$\begin{aligned} E_k(X|Z) &= \sum_i Z_{ki}(\mathbf{x}_i - \mu_k)^\intercal \mathbf{I}(\mathbf{x}_i - \mu_k) \\ &= \sum_i Z_{ki}(\mathbf{x}_i^\intercal \mathbf{I}\mathbf{x}_i - 2\mathbf{x}_i^\intercal \mathbf{I}\mu_k + \mu_k^\intercal \mathbf{I}\mu_k) \\ &= \sum_i Z_{ki}\mathbf{x}_i^\intercal \mathbf{I}\mathbf{x}_i - 2\sum_i Z_{ki}\mathbf{x}_i^\intercal \mathbf{I}\mu_k + \sum_i Z_{ki}\mu_k^\intercal \mathbf{I}\mu_k \\ &= \sum_i Z_{ki}\mathbf{x}_i^\intercal \mathbf{I}\mathbf{x}_i - 2\sum_i Z_{ki}\mathbf{x}_i^\intercal \mathbf{I}\mu_k + s_k\mu_k^\intercal \mathbf{I}\mu_k, \end{aligned} \tag{1}$$

with the cluster mean $\mu_\mathbf{k}$ and identity matrix $\mathbf{I}$. By using the maximum likelihood (ML) estimator of the cluster mean

$$\mu_k = \frac{1}{s_k}\sum_j Z_{kj}\mathbf{x}_j, \tag{2}$$

with cluster size s_k, the energy formulation only depends on the data and the cluster assignment

$$\begin{aligned} &\sum_i Z_{ki}\mathbf{x}_i^\intercal \mathbf{I}\mathbf{x}_i - 2\sum_i Z_{ki}\mathbf{x}_i^\intercal \mathbf{I}\mu_k + s_k\mu_k^\intercal \mathbf{I}\mu_k \\ &= \sum_i Z_{ki}\mathbf{x}_i^\intercal \mathbf{I}\mathbf{x}_i - 2\sum_i Z_{ki}\mathbf{x}_i^\intercal \mathbf{I}\frac{1}{s_k}\sum_j Z_{kj}\mathbf{x}_j \\ &+ s_k\frac{1}{s_k}\sum_i Z_{ki}\mathbf{x}_i^\intercal \mathbf{I}\frac{1}{s_k}\sum_j Z_{kj}\mathbf{x}_j \\ &= \sum_i Z_{ki}\mathbf{x}_i^\intercal \mathbf{I}\mathbf{x}_i - \frac{1}{s_k}\sum_i\sum_j Z_{ki}Z_{kj}\mathbf{x}_i^\intercal \mathbf{I}\mathbf{x}_j. \end{aligned} \tag{3}$$

This finally shows that the total energy only depends on the distance between each pair of points

$$
\begin{aligned}
&\sum_i Z_{ki}\mathbf{x}_i^\intercal \mathbf{I}\mathbf{x}_i - \frac{1}{s_k}\sum_i\sum_j Z_{ki}Z_{kj}\mathbf{x}_i^\intercal \mathbf{I}\mathbf{x}_j \\
&= \frac{1}{s_k}\sum_i\sum_j Z_{ki}Z_{kj}\ \mathbf{x}_i^\intercal \mathbf{I}\mathbf{x}_i - \mathbf{x}_i^\intercal \mathbf{I}\mathbf{x}_j \\
&= \frac{1}{s_k}\sum_i\sum_j Z_{ki}Z_{kj}\ \mathbf{x}_i^\intercal \mathbf{I}(\mathbf{x}_i - \mathbf{x}_j) \\
&= \frac{1}{s_k}\sum_i\sum_j Z_{ki}Z_{kj}\ \frac{1}{2}[\mathbf{x}_i^\intercal \mathbf{I}(\mathbf{x}_i - \mathbf{x}_j) + \mathbf{x}_j^\intercal \mathbf{I}(\mathbf{x}_j - \mathbf{x}_i)] \\
&= \frac{1}{s_k}\sum_i\sum_j Z_{ki}Z_{kj}\ (\mathbf{x}_i - \mathbf{x}_j)^\intercal \mathbf{I}(\mathbf{x}_i - \mathbf{x}_j).
\end{aligned}
\tag{4}
$$

3. Inference Parameter Optimization

Using the quadratic penalty method to include constraints in the Quadratic Binary Optimization (QUBO) formulation requires finding suitable Lagrangian multipliers λ. Even though a very high multiplier theoretically guarantees to find a feasible solution, it also deteriorates the conditioning of the optimization problem. Therefore, a suitable Lagrangian multiplier lifts the cost of any constraint violation above all relevant solutions of the clustering problem, while keeping them low enough to avoid scaling the total energy of the problem up considerably. To estimate the multipliers for each constraint, we follow an iterative procedure.

In an initial step, balanced k-means[2] is used to find a feasible clustering solution. This is used to offset the distance terms for each point such that the total clustering solution has an energy of 0. For the next steps, the constraints are separated into 3 components:

The cluster size constraint is defined by $\sum_i Z_{ki} = s_k\ \forall k$. Due to its strong diagonal term in its quadratic form using Lagrange multipliers, it quickly degrades the energy scaling of the problem. The Lagrangian multiplier corresponding to this constraint is estimated from the maximum cost improvement that can be achieved by switching one point between clusters.

The constraint $\sum_k Z_{ki} = 1\ \forall i$, which ensures the matching of every point to exactly one cluster, is further segmented into two parts. One part contains the positive off-diagonal elements and penalizes assigning a single point to multiple clusters. Its corresponding Lagrangian is computed from the maximum cost improvement that can be achieved by assigning an additional point to any cluster, compared to the k-means solution. The other term contains negative diagonal elements and adds an incentive to assign every point to one cluster. The corresponding multiplier is estimated by the maximum cost improvement by removing one point from a cluster and thus violating the constraint.

In five subsequent optimization steps the clustering problem is solved using simulated annealing and the Lagrangian multipliers are increased for constraints that are not fulfilled.

In the last step, which is only performed for simulated annealing, measurements at low temperatures of the Boltzmann distribution are handled. In scenarios where only a single valid clustering solution is returned, the Lagrangian multiplier of the cluster size constraint is increased, which results in sampling the problem at a higher temperature.

Finding well-suited Lagrangian multipliers is crucial due to the low fidelity of the current generation of AQCs, which requires careful engineering of the problem energy. Therefore, we expect this procedure to become of reduced importance in future generations of lower noise AQCs.

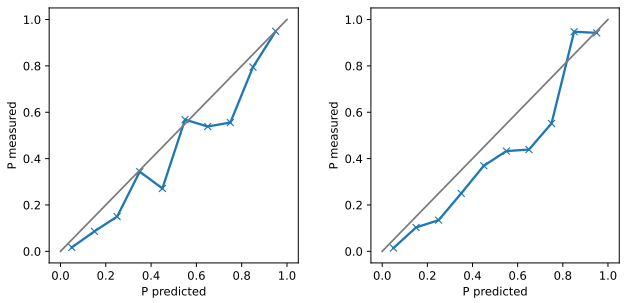
For experiments performed on the D-Wave AQC, all Lagrangian optimization steps are performed with SIM, before measuring the final results on the AQC due to the strong compute-time limitations.

4. Data Generation

Synthetic Data is generated by sampling a total of I points from separate normal distributions for each of K clusters. The cluster centers are selected as the corners of a simplex with uniformly drawn edge length, such that the distance between each pair of clusters lies within a predefined range $[d_{\text{min}}, d_{\text{max}}]$. The feature-space needs to be at least $K-1$ dimensional for each clustering problem. Sampling the edge-length randomly allows to generate a wide range of clustering problems with a different degree of ambiguity, due to the changing degree of overlap between distributions. This allows to evaluate the whole range of predicted posterior probability values. For each experiment a total of L clustering tasks is generated to evaluate the clustering metrics.

IRIS [1] is subsampled to generate quantitative results over different clustering scenarios. The whole IRIS dataset contains 3 classes, 50 samples for each class and 4 features forming a 4-dimensional space. According to the experiment parameters we randomly select a subset of classes, samples and features without replacement to allow running the tasks on a D-wave AQC. This generates different clustering problems, while keeping the general structure of the data in IRIS.

Image data is used to demonstrate the applicability of our method to computer vision tasks. We collected 8 images each for cars and boats and two images of cars towing boats. Visual embeddings for each image are extracted after the last layer of VGG16 [3] pretrained on Imagenet. Subsequently, the high-dimensional features are clustered using our approach, and the first coreset is computed to identify the ambiguous samples as demonstrated in the main paper.



(a) 3 clusters, 30 points. (b) 3 clusters, 45 points.

Figure 1. Evaluation of the calibration for simulated annealing in clustering scenarios with 3 clusters and 30/45 points respectively. All results generated with 1,000 problems in each scenario and 20,000 measurements for each clustering problem.

5. Cluster Calibration Evaluation

This section extends the analysis of the calibration of our method to additional synthetic scenarios with more variation and increased problem size. The plots provided in this section are generated similarly to the main manuscript where first all clustering solutions Z are accumulated in bins according to their estimated posterior probability $P(Z|\mathbf{X})$, including all sampled but non-optimal solutions. After accumulation, the ratio of correct solutions in each bin is evaluated and plotted over the probability range of each bin. In the plots the diagonal represents the desired calibration.

Further calibration plots for the scenario with 3 clusters and an increasing number of total points are provided in Figure 1. The scenarios are solved using simulated annealing with 20,000 measurements for each problem. The experiment with a total of 45 points in Figure 1b shows an overestimation of the posterior probability of the respective solutions. This can be attributed to two possible scenarios, where 1) the best solution is found, but not all relevant high-energy solutions are found during annealing and 2) the lowest-energy solution is not found and thus, the probability of all other solutions is overestimated. As the optimization problem becomes harder with an increasing number of points, the behavior is stronger in Figure 1b than in Figure 1a.

6. Coreset Sparsification Performance

The set of feasible solutions can be merged by using the calibrated confidence scores in Algorithm 1 introduced in the main manuscript. It sequentially removes uncertain points from the solution thus, increasing the solution probability. In Figure 2, we show the probability of the best merged solution being correctly evaluated over the minimum solution probability of the sparsified coreset. It shows that our approach of removing single points can considerably increase

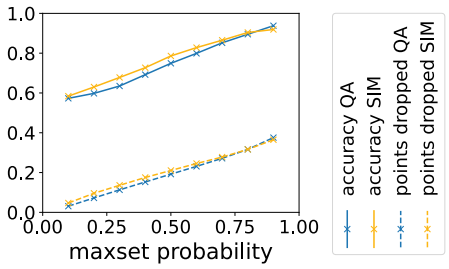


Figure 2. Clustering accuracy wrt. removing uncertain points by merging coresets.

the solution probability, thus highlighting the quality of the found coresets.

7. Evaluation on IRIS

Table 2 in the main manuscript provides performance metrics for randomly subsampled versions of the IRIS dataset [1]. In this section, we further evaluate the performance on the whole IRIS dataset, which contains 3 classes, 50 samples for each class and 4 features. We use simulated annealing with 20,000 measurements and balanced k-means to solve the IRIS clustering task, which both provide the same solution. The qualitative results are depicted in Figure 3, where all pairs of features are visualized. The shape of each sample represents the ground truth class and the color the result of the clustering algorithm. While the different feature pairs are plotted separately, the problem is solved as a single 4-dimensional clustering task. As results are identical with simulated annealing and balanced k-means, clustering metrics are also identical with a Completeness of 77.7%, Adjusted Rand index 78.6% and Fowlkes-Mallows Score of 85.6%.

8. Failure Cases

Analyzing the failure cases of our method provides valuable insight into the current state of quantum computing in computer vision, which aids to identify areas that need to be further investigated.

8.1. K-means

The analysis of synthetic problems in Table 1 in the main manuscript shows an advantage of our approach compared to the balanced k-means algorithm [2] for the smaller clustering scenarios. These cases can be traced back to the k-means algorithm finding local minima where switching points given the last cluster means does not improve the data fit. This scenario is avoided in our formulation by jointly optimizing for the assignment and cluster centers. Two examples of such failure cases of k-means clustering are provided in Figure 4.

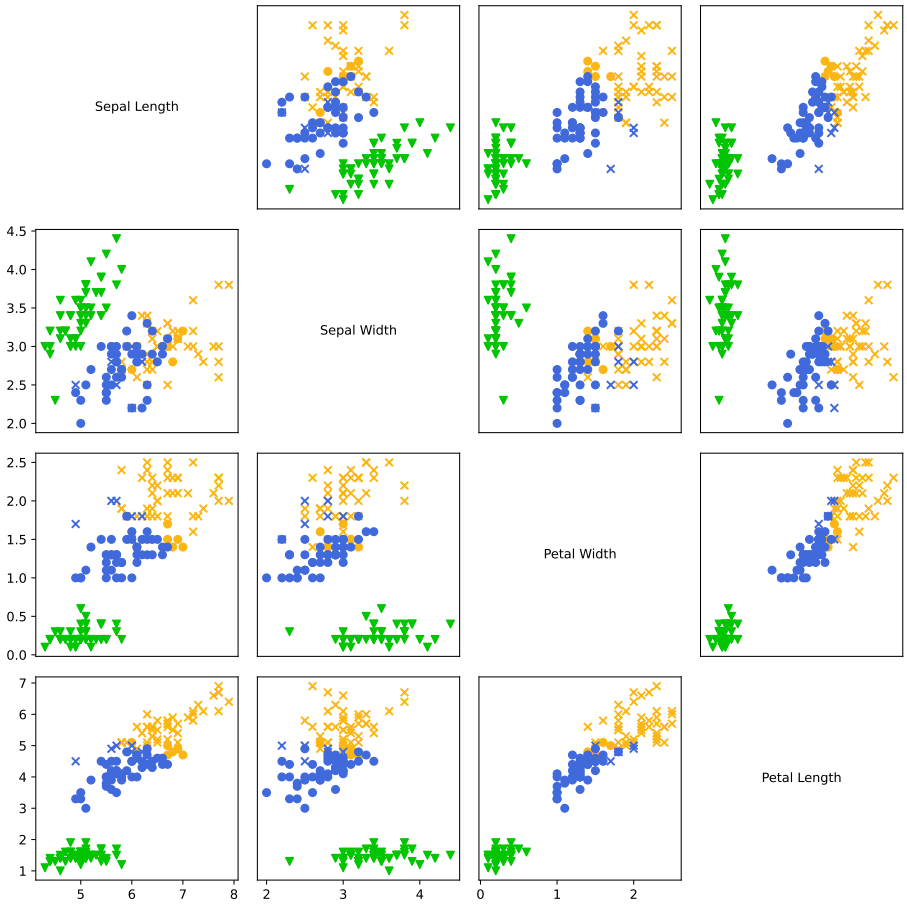


Figure 3. Clustering results on the IRIS dataset for simulated annealing and k-means.

8.1.1 Annealing based clustering

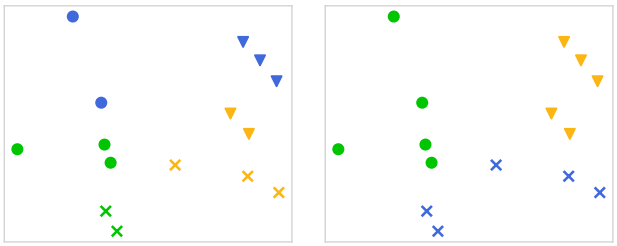
The scenario with a total of 45 Points in 3 Clusters shows an advantage of k-means in the number of correctly solved problems compared to our approach using simulated annealing. The main source for this behavior lies in not finding the lowest energy and thus, the optimal solution of the clustering problem, as depicted in Figures 5a and 5b. Another source of error in this scenario is shown if Figures 5c and 5d, where the local k-means solution corresponds to the ground truth, even though it has a higher energy. As the solutions returned by our approach are still dense clusters, the clustering metrics remain competitive with the balanced k-means approach.

9. Limitations

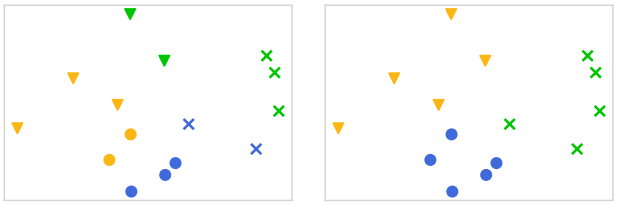
Our work aims at demonstrating the potential of using a quantum computer as a sampler for k-means clustering, in order to find multiple likely solutions and their associated calibrated posterior probabilities. Given the novelty of applying quantum computing to computer vision, it's natural that many works in this area, including ours, still come with limitations.

Current quantum computers are still limited in their fidelity of qubit couplings, which represent the terms of the quadratic cost function. This requires a careful selection of Lagrangian multipliers, which adds additional computational cost in the current formulation. With improving AQCs, this problem can be reduced and help to increase the problem size, as well as the robustness of the formulation. Another hardware limitation is the restricted connectivity between qubits. In the D-Wave Advantage 2 prototype used in this work, each qubit is coupled to up to 20 neighbors. This requires to build chains of qubits to represent a dense cost-matrix. Therefore, investigating sparse representations for clustering that reduce the required chain length can help to embed larger problems on the AQC.

Finally, our clustering approach is following the k-means cost function, with an identity covariance matrix. While this can model a range of practical problems, where the distribution of the data can directly be influenced, future work should investigate formulations of higher-order terms.

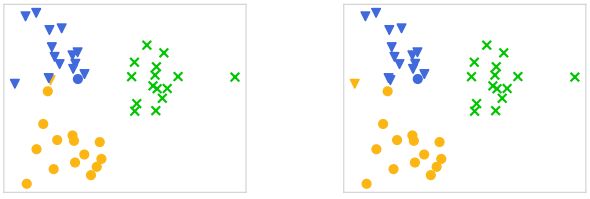


(a) K-means. (b) Simulated annealing.

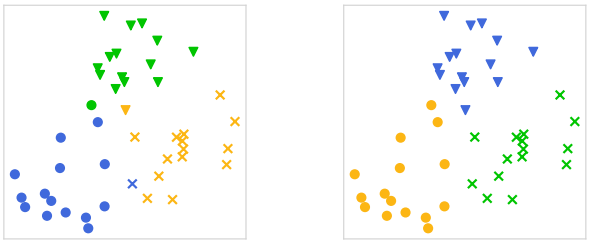


(c) K-means. (d) Simulated annealing.

Figure 4. Failure cases for k-means clustering. While our formulation finds the correct solution, k-means returns a local minimum.



(a) Simulated annealing. (b) K-means.



(c) Simulated annealing. (d) K-means.

Figure 5. Failure cases for simulated annealing clustering.

References

[1] R. A. Fisher. The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics*, 7(2):179–188, 1936. 2, 3

[2] Mikko I. Malinen and Pasi Fränti. Balanced K-Means for Clustering. In *Structural, Syntactic, and Statistical Pattern Recognition*, pages 32–41. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014. 2, 3

[3] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv:1409.1556 [cs]*, 2014. 2