We thank the reviewers for their insightful comments, and their remarks concerning typos, notation, references. All these will be corrected in the final paper. We answer some of the specific questions raised in the reviews.

**Referee 1**

*On the quantum advantage.* The classical time for a single iteration of $k$-means is $O(kNd)$, we will add this to the main paper. One of the main advantages of $q$-means is to provably reduce this to $O(poly(k, d, \log N))$. Such provable speedups are quite hard to obtain. Recent breakthroughs in quantum-inspired classical algorithms show similar speedups, however these algorithms are strongly impractical (see [arXiv:1905.10415 (2019)] and our Section 5.2, 433-454). Note also that classically, the noisy $k$-means has asymptotically the same running time as $k$-means, in particular the complexity with respect to the number of points remains $O(N)$ and does not drop to $O(\log N)$.

*The well-clusterability assumption.* Like $k$-means, the $q$-means algorithm and its runtime are compatible with any dataset and we do not need the well-clusterability assumption for our analysis. We only consider this assumption at a second stage in order to provide a second, more interpretable, running time for these datasets in terms of $k, d$ rather than $\kappa$ and $\mu$. We do not know of a specialized classical algorithm for well clustered datasets, but a quantum algorithm would probably still offer an advantage due to its logarithmic dependence on the number of data points.

*Comparison with [WKS14].* The $q$-means algorithm is substantially different from [WKS 14], which works on sparse datasets. The first step of $q$-means indeed assigns points to the closest cluster using methods similar to [WKS14]. However, the centroid update step uses quantum linear algebra and tomography techniques that have been developed more recently. This is also the reason that WKS obtains only a quadratic speedup for $k$-nearest neighbors.

*On $\delta$-$k$-means.* The introduction of $\delta$-$k$-means is necessary to have a rigorous classical analogue of our quantum algorithm. Our quantum algorithm has some inherent approximation error due to quantum subroutines like distance estimation and tomography that output estimates with error $\delta$. Therefore a quantum algorithm using these procedures cannot simulate the exact $k$-means. Despite this, $q$-means is still useful for machine learning purposes. Noise is also practically motivated and appears even in some classical implementations to achieve robust clustering and avoid overfitting. This is also confirmed by our experiments with the MNIST and synthetic dataset.

*More comments.* Indeed, the running time of the quantum algorithm is "hairy" but we find it important to include all parameters explicitly, including $\kappa(V)$ (condition number) and $\mu(V)$ (Frobenius norm/spectral norm). A discussion bounding these parameters for well clustered datasets is in section 5.3, and for 'low-rank' data one can think of both parameters as $\sqrt{k}$. For lines 201-207, we measure only the second register, thus the state obtained is one of the $k$ cluster centers almost uniformly at random to which the coupon collector argument can be applied.

**Referee 2**

*Improvement: 1.* A gaussian distribution would indeed improve the empirical performances of the $\delta$-$k$-means algorithm. However, the uniform distribution comes as a consequence of the current quantum algorithm, thus one would need a different variant to work with the gaussian distribution.

*2.* The probability of failure does not affect significantly the running time: for the median it appears within a logarithmic factor; the tomography works with probability $1 - 1/poly(d)$ with the stated running time.

*3.* The distance estimation error is $\epsilon_1$, hence the difference between two distances can err by $2\epsilon_1$, needing $\epsilon_1 = \delta/2$.

*4.* Our work also includes the $q$-means++ algorithm for initialization which follows exactly the $k$-means++ algorithm by creating a probability distribution proportional to the squares of the distances. These distances are indeed estimated with a noise addition of order $\epsilon_1$ that can be tuned. This will be stated more clearly in the main paper.

*5.* The classical $k$-means can be sped up to complexity $O(kdN/p)$ when using $p$ parallel processors, thus one would need an infeasibly large number of parallel processors to match the quantum running time. Moreover, the complexity of the quantum algorithm refers to the size of the circuit and we do not assume parallel processors. Note also, that we could apply our quantum algorithm on each one of these processors and reduce the complexity to $O(kd\log(N/p))$. Quantum algorithms can be more powerful than just parallel algorithms, they involve a completely different set of techniques and can offer in certain cases unrivalled speedups.

*Bottleneck.* Indeed, $k$-means/$q$-means can get stuck in local minima, but $k$-means++/$q$-means++ finds a local optimum that is provably within $\log(k)$ of the true optimal value, and it rarely takes many iterations. In most large data sets, the bottleneck is the cost of each iteration, since one needs to go over all points. This is the great advantage of our algorithm. Even with a considerable number of parallel processors, large data sets still cannot be classically clustered efficiently.

**Referee 3.** We will make our best effort to present clearly all core results, previous work, and ideas within the main paper page limits, and make it easier for the classical ML community to understand the potential of quantum computing.

Finally, some compelling advantages of $q$-means: It is directly comparable to classical algorithms as it has the same input and output. The algorithm is simple enough to be implemented on small quantum computers and could provide an experimental test for quantum machine learning. Despite its flaws, $k$-means is fundamental to classical ML and we think $q$-means will also be fundamental for quantum ML as it provides strong evidence for the relevance of quantum computing for clustering. It is also a great introduction for the classical ML community to the power of quantum, in line with NeurIPS. Last, concerning impact, we note that our techniques can be extended to other problems such as Gaussian Mixture Models, Expectation Maximization, spectral clustering, Neural Networks and such work is under way.