We thank all reviewers for their comments and we will make every effort to address all comments in the revision.

R1 and R5: Rates/finite sample bounds. One of main difficulties in finding a rate of convergence of our estimator is precisely characterizing how convergence of the mixture components (l. 192) depends on the convergence of the mixture distribution (l. 191) in Theorem 3. Our algorithm can be seen as approximating a distribution with a low-rank symmetric tensor. While matrix convergence implies convergence of spectral decompositions and is well characterized, general tensors suffer from a "lack of closeness[1]." E.g. it is possible to construct a sequence of symmetric rank 3 tensors that converges to a symmetric rank 4 tensor[1]. This sort of phenomenon makes it difficult to characterize convergence, and this is only exacerbated since we are working in tensor spaces of functions rather than Euclidean vectors. Interestingly the full version of Theorem 3 (Supp.) implies that for general product spaces "lack of closeness" is solved by assuming densities of the form in (l. 117) identifiable. We will look into this convergence in future work.

R1 and R5: Nonconvex Optimization The reviewers have correctly assessed that the suggested algorithm is not guaranteed to solve (6) globally. We note that the class of nonconvex optimization problems known to be solvable by descent methods is extremely limited, e.g., phase retrieval, PCA, matrix completion. The objective function we consider is a polynomial of degree six, while the constraint set is convex. While general polynomial optimization is NP-hard[2], we expect and observe that the proposed optimization problem is no more difficult than other common nonconvex optimization problems in machine learning, e.g., training neural networks. Projected SGD is widely used in practice, and there exists some convergence analysis in the literature [3]. Specifically, convergence to a stationary point can be shown under Lipschitz-type assumptions. In the revision, we will add some discussion of the technical details to the main text, making it clear that we do not claim the proposed algorithm finds a global minimum.

R1, R3, and R5: Additional Discussion It was suggested that the work would benefit from more motivation and discussion of applications. As the reviewers have pointed out, the grouped sample setting is not commonly considered in the literature. One exception is clustering with must-link constraints. There is quite a bit of work in this area with applications in interactive visual clustering and computer vision[4]. Additionally our method can be seen as a continuous version of multinomial mixture modelling, which is used in psychometrics where measurements over time are collected for a group of, for example, bipolar disorder patients and used to identify subgroups within that population whose condition is only evident with repeated temporal measurements [5]. We will add more discussion of applications and assumptions to the revised paper by folding section 3 into section 4 and removing one or more proof sketches.

R1: "NDIGO (the algorithm introduced by this work) outperforms other methods on the training sample, but not for out-of-sample setting (where NPMIX performs the best)." This is a typo. NDIGO performs best (see supp.).

R1: "In the current version, it was not immediately clear to me why (6) and (7) are equivalent." In the main text, we will add more detailed pointers to appropriate sections in the supplement.

R1: "It would be nice to see a little more experiments on synthetic data." Before submission, we removed some synthetic data experiments (e.g., variations on the moons dataset, mixtures of mixtures of Gaussians) to keep the paper within eight pages, but we will add these results to the supplemental in the revision.

R5: "How practical is this approach on large datasets?" The coreset approach appears to scale very well to large datasets. For example, the Twitter experiment we have $2n > 3 \cdot 10^6$.

R5: "What tradeoffs and limitations exist in practice, esp. compared to related algorithms?" In the optimization problem we must tradeoff between computational complexity and clustering performance by adjusting the coreset size. Another limitation of our approach is when mixture components are technically mutually irreducible but their supports only differ on a set of very small measure, but any algorithm may fail in this case. It is hard to point to an algorithm that will work when our's does not since other approaches rely on notions of cluster separability. If the data is well fit by some parametric mixture or if components are well separated it could be better to use existing techniques.

R5: "The assumption that M is known should be mentioned up front and discussed early on" We will be sure to emphasize this point in the revision. Although we did not try it, our spectral initialization should offer a heuristic for selecting M by looking for the knee in the eigenvalue curve. AIC/BIC/MDL should also be applicable.

R5: "In the experiments, which of the algorithms compared make use of the paired observations, and which do not?" On line 257 we specify that NPMIX does not utilize pair information, but we will reword things to clarify this.

[1] See "Symmetric tensors and symmetric tensor rank" by Comon et al.

[2] See "Complete solutions and extremality criteria to polynomial optimization problems" by Gao

[3] See "Convergence of a multi-agent projected stochastic gradient algorithm for non-convex optimization" by Bianchi and Jakubowicz, as well as "Zeroth-order Stochastic Projected Gradient Descent For Nonconvex Optimization" by Liu et al.

[4] See "Constrained Clustering: Advances in Algorithms, Theory, and Applications" by Basu et al.

[5] For example "A mixed-binomial model for Likert-type personality measures" Allik 2014 or "Cognitive psychometrics: Using multinomial processing tree models as measurement tools" Batchelder 2010