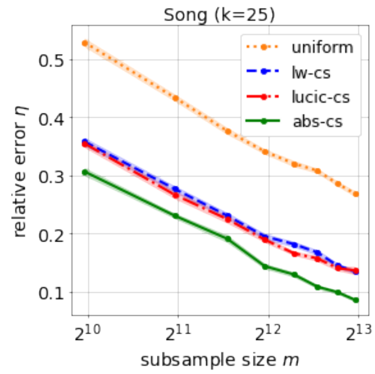
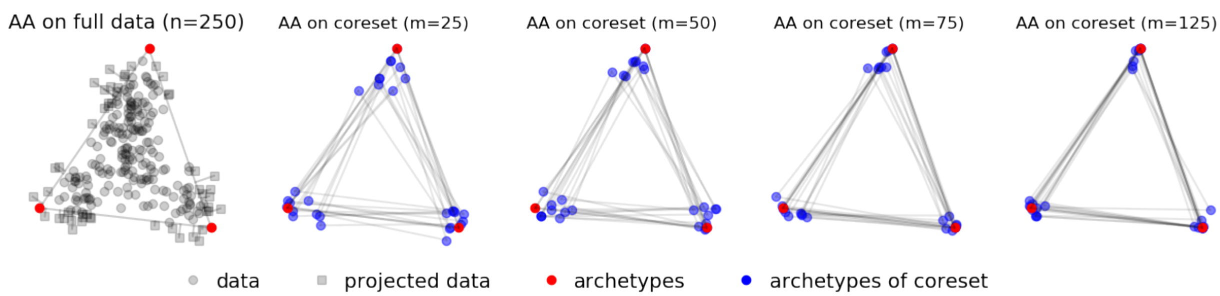
We thank all reviewers for their careful reading and their valuable comments. In the following, we answer the main questions and comment on the points raised. R1: *[..] I wonder why the original kmeans coreset was not used in experiments [..]* The experimental analysis of Bachem et al. (2018) shows that the lightweight-coreset performs very similar to the one in Lucic et al. (2016) for k-means. As seen in the figure on the right, the performance of Lucic et al. (2016) (lucic-cs) is indeed very similar to the lightweight-coreset, even for AA. We now included this baseline in the paper. R1: *The dimension of B is stated wrongly [..]* Thank you for pointing us to the typo in the dimensionalities of the matrices $\mathbf{A}$ and $\mathbf{B}$. We revised the manuscript accordingly. R1: *I did not understand the comment that "removing $\frac{\varepsilon}{2}\phi_X(Q)$ causes problem in clustering" [..] Also, the statement "In contrast to k-means, we assume that the mean .." is not clear to me.* Thank you for raising this issue. Reviewer 3 also pointed this out. We revised these paragraphs accordingly. R2: *[..] However, it is not quite clear to me how the archetype positions are updated after initialisation. [..]* After initialization, the matrices $\mathbf{A}$ and $\mathbf{B}$ ($\mathbf{Z} = \mathbf{BX}$) are optimized such that the residual sum of squares (RSS in Eq. (2)) is minimal. The standard procedure of the alternating optimization over $\mathbf{A}$ for fixed $\mathbf{B}$ and vice versa is also outlined in Algorithm 1 in the supplementary material. R2: *[..] Table 1 reporting the relative errors suggests that there might be a substantial deviation between coreset and full dataset archetypes. [...]* Note that the "large" relative errors may be due to a too small coreset size m for this data set and that we chose the same coreset sizes for all data sets. By increasing m, the relative error is expected to drop further. In practice, one does usually not choose ε and δ and compute the correct m but rather uses the largest m suitable for the infrastructure at hand. Our theoretical results ensure that the errors are bounded and that the approach is better than a naive uniform subsample. R2: *[..] Is the archetypal interpretation of identifying (more or less) stable prototypes whose convex combinations describe the data still applicable?* R3: *[..] However, the archetypes found by the coreset and the original dataset can be different, and it will be interesting to have theoretical properties for $|Q_C - Q_X|$ [..]* The archetypes found by the coreset and the full data set are indeed different. Otherwise, the relative error was zero since we always report error on the full data set. Measuring $\|Q_C - Q_X\|_F$ as suggested is not trivial since the archetypes in the $\mathbf{Z}$ matrices might be permuted. Hence, we would have to rely on something like optimal transport on empirical distributions. However, even if we computed those, the errors would drop for increasing coreset sizes m. As m approaches n, the archetypes on the coresets converge towards the archetypes on the full data set. It is also not clear how to measure interpretability. The reviewers are totally right by stating that AA is naturally more sensitive to points on the boundary of the data. This is one reason why we dropped the uniform part within the sampling distribution (compare Eq. (5) with line 148) to put more focus on points far away of the center of data. By increasing m, more of those points are discovered and the archetypes can be placed closer to the real boundary. However, the directions in which the archetypes lie should be approximatively preserved. We conducted a simple experiment on toy data with $n = 250$ points and sampled 10 coresets for each $m \in \{25, 50, 75, 125\}$. The archetypes learned on the coresets (blue) converge to the archetypes learned on all data (red). The larger the coreset the better the approximation.





R2: *Practically, the number of archetypes k is of interest. In the presented framework, is there a way to perform model selection in order to identify an appropriate k?* The standard way of model selection in AA is to compute the RSS for various values of k, then plot the error against k and finally choosing the k according to the elbow criterion. This can be also done on a coreset. R2: *The work in [3] might be worth to mention as a related approach. [..]* Thank you for pointing us to this related literature; the new version now includes a discussion in the related work section. R3: *Since the author are reporting relative error, the y axis of the figures should start from 0* For most of the plots it won't make a difference, but for covertype it would add a lot of white space and render differences in performance very small. R3: *in line 97-98, please state with respect to what the expectation is taken over* Thank you. We updated this part.