Scalable K-means++ Problem Set

Zack Nawrocki, Ziping Song, and Jeff Lehmann

December 2020

1 Conceptual Questions

1) In the presentation, we calculated the distances for clustering using the Euclidean distance algorithm. It is also possible to calculate distances, using the Manhattan distance algorithm (you can learn about the formula here https://xlinux.nist.gov/dads/HTML/manhattanDistance.html). When could the Manhattan distance be useful?
(Hint: Page 1 - Abstract: https://bib.dbvis.de/uploadedFiles/155.pdf)

2) Give an example, different than one discussed in the presentation, of what k-means clustering can be useful for, in real-world situations.

3) With k-means algorithms, we utilize averaging. Similar algorithms exist, such as the k-medians algorithm (utilizing medians). Give one situation where this algorithm could be a better approach.

4) Why would oversampling, in the Scalable k-means++ algorithm, improve initialization?

2 Proof Question

1) As discussed in the presentation Theorem 1 is as follows:

If an α approximation algorithm is used in Step 8, then Algorithm k-means// obtains a solution that is an $O(\alpha)$-approximation to k-means.

To prove this Bahmani proves that the expected cost of adding new centers at each iteration, where ϕX(C) is the cost of the current centers in C, ϕX(C $\cup$ C') is the cost of adding new centers, C', to C, ϕ^* is a constant factor that shows the drop in solution cost after each iteration (Hint: does not exist in first iteration) and α is approximately $e^{\frac{-l}{2k}}$, in Theorem 2:

Theorem 2

$\mathbb{E}[\phi X(C \cup C')] \leq 8\phi^* + \frac{1+\alpha}{2}\phi X(C)$

To prove Theorem 1, Bahmani proposes the following Corollary which would prove that there is a constant factor approximation to k-means after O(log ψ) rounds if we prove that the cost of clustering at the ith iteration is:

Corollary 3

$\mathbb{E}[\phi^{(i)}] \leq (\frac{1+\alpha}{2})^{(i)}\psi + \frac{16}{1-\alpha}\phi^*$

Prove Corollary 3 using induction which therefore proves the approximation guarantee of Theorem 1. (Hint use Theorem 2 in your inductive step)