EECS 16ML Introduction to Machine Learning Skills
Fall 2020 Quiz Solutions

Test your Understanding of K-Means

1. What is the difference between supervised and unsupervised learning? Which category does K-Means fall under?
   Solution: The main difference between supervised and unsupervised learning is that supervised learning is done using ground truth knowledge. In other words, we have prior knowledge of what our output samples should be for any input in our training set. Common examples of supervised learning techniques include linear regression and neural networks. On the other hand, unsupervised learning does not have any associated labels or outputs so the goal is to uncover the natural structure of the data. In the case of K-Means, we do not have access to the true clusters, so it would fall under unsupervised learning.

2. Cost is the counterpart to error for unsupervised, clustering models.

   (a) Explain why the sum of squared distance is used to measure cost for K-Means.
   Solution: Because we do not have access to the true labels, we can't use standard misclassficiation error like we would do in a classification problem. Instead, we need to have a measure of cost that accounts for how well the data is clustered. In K-Means, we have two objectives: minimizing intra-cluster distance and maximizing inter-cluster distance. Because a good model should have low cost, we pick the intra-cluster distance as a measure of cost, therefore yielding the sum of squared distance.

   (b) What are other potential measures of cost for K-Means?
   Solution: Other potential measures of cost for K-Means could include the inverse of the intercluster distance, which decreases as the model performance increases. You could also look at how similar the data points in the same cluster are to each other, rather than just how close the data point is to the centroid of the cluster.

3. Is K-Means always guaranteed to converge? Why or why not?
   Solution: Yes, K-Means is always guaranteed to converge to a solution. By converge, it will eventually find clustering assignments and centroids that will not change between iterations. This is because the cost function is always guaranteed to decrease monotonically in each iteration. Therefore, K-Means is guaranteed to converge in a finite number of iterations.

4. Will K-Means always produce the same final clusters each time it's run? Why or why not?
   Solution: No, even though K-Means is guaranteed to converge, it is not guaranteed to always converge to the same solution. The final clusters are highly dependent on where the centroids are first initialized to be. This is why random restarts are very helpful to find an ideal K-Means clustering assignment.

5. What is the big-O runtime of Lloyd's Algorithm? Let n = number of data points, d = dimension of each data point, k = number of clusters, and i = number of iterations needed till convergence.
   Solution: The big-O runtime of Lloyd's algorithm is $O(nkdi)$.

6. Why are traditional hyperparameter tuning techniques, like grid search, ineffective for picking k?
Solution: As k (the number of clusters) increases, the cost is going to monotonically decrease. When k = n (the number of data points), the total cost will be 0 because the centroid of each cluster will be the 1 data point assigned to that cluster. However, we have to be careful to account for noise in our dataset and for the fact that there may be other data points in the cluster that are not present in the dataset. In order to get clusters that are actually representative of the underlying structure, we want to pick the k that represents the sharp drop off in cost, also known as the elbow of the cost vs k graph. This way, we picking the k that gives us low cost without overfitting to the dataset.

7. In Project Part 2, Example 1: What explains the phenomenon occurring with the clustering? What steps can you take to prevent this?
Solution: The blue and green clusters are much larger than the orange cluster. The K-means algorithm optimizes its clustering to minimize the intra-cluster distance. In cases like this example, where there are cluster size imbalances, the K-means algorithm achieves a solution where 2 cluster centroids are assigned to what is actually a single cluster. This allows the algorithm to reduce distances to the cluster center within a larger cluster, which outweighs the larger cost incurred on the orange datapoints.

8. In Project Part 2, Example 3: What explains the performance of the clustering on the given dataset? What solves the issue and clusters the data as intended?
Solution: The K-means algorithm does not correctly cluster the data points in this example. It forces a linear boundary and places the centroids on opposite sides between the inner and outer circles to minimize cluster distances. As we've seen with regression techniques, we can try lifting the feature space. In this case, a natural thought might be to lift the feature space using the radius-squared feature (summing the squares of the coordinates). However, we also need to have a systematic way of removing the features (i.e. (x,y) coordinates) that don't actually help us with the clustering problem we're trying to solve. We can manually transform the dataset into points that are clusterable for the K-means algorithm using the radius-squared feature. The inner circle all correspond to smaller positive values and the outer circle all correspond to larger positive values. In this case, we were able to manually construct the feature that helps separate our data and fix the issue.

9. What are the benefits of Kernelized K-Means? Describe an appropriate situation where you would use Kernelized K-Means over regular K-Means.
Solution: Kernels are a much more efficient way of calculating similarity between two data points compared to the Euclidean distance. We can also implicitly encode features into the kernel function, especially in the cases where it is difficult to create manual features. Kernelized K-Means would make sense to use in situations where we need to work with implicit features like in Natural Language Processing work, or if we ever want to use a measure of distance that is different from Euclidean distance.

10. How would you modify K-Means if you knew the measurement of certain features were more accurate than other features?
Solution: We can modify our cost function to take into account the more accurate features. Either we can discount the inaccurate features completely or we can add weights to the more accurate features so that they play a larger role in determining the cost. This way, our cost will be more representative of the more accurate features which we can trust more.

11. Another variation of K-Means is called K-Medoids. A medoid is defined as the object of a cluster whose average dissimilarity to all objects in the cluster is minimal - the most centrally located point in the cluster. In this variation, instead of updating the centroids to be the average of all the points

in the cluster, the centroid is chosen to be the mediod of the cluster. What are the advantages and disadvantages of this method?

Solution:

- Advantages: It is more robust to noise and outliers because it minimizes a sum of pairwise dissimilarities compared to the sum of squared Euclidian distance. In addition, because the dissimilarity measure used to find the medoid can be arbitrarily chosen, the K-Medoids is flexible to different types of data where it may be hard to define distance (i.e. between words).
- Disadvantages: K-Medoids can be much more computationally expensive than K-Means because it is more difficult to find the centroid. In addition, the initial structuring of the data may actually not have any data points that are central to the cluster. For example, if all the data points in a cluster seem to lie in a circle, then K-Means would pick the center of the circle as the centroid. However, K-Medoids would be forced to pick one of the data points on the outskirts of the circle, which could skew the results.