HW6 Starter

Problem 1: Principal Component Analysis (PCA)

1a

1ai) True. Finding eigenvectors are required to decompose matrix X into its constituents and minimize reconstruction error.

1aii) False. Selecting K solely by minimizing error will always yield K = N (perfect reconstruction). This could however, be computationally exhaustive and defeats the purpose of performing dimension reduction. We should also look at memory constraints and use cumulative variance plot to decide K.

1aiii) True. PCA is a nested model so a PCA of K = 10 will have overlaps with PCA where K = 9, reducing the computation required.

1b

1bi) No, PCA components learned will not be the same as PCA is sensitive to input scaling.

1bii) I would recommend standardizing the input data. This ensures that input data is centered as PCA is sensitive to input scaling, seen in 1bi).

1c

The formula used for computing the mean is wrong. The correct procedure is to calculate mean using training data x(n) instead of test data x(t).

Problem 2: K-Means Clustering

2a

2ai) True. When K = 1, the centroid will always contain all samples as coordinate descent here is similar to finding the mean for each feature set over all samples.

2aii) False. When k = 2, coordinate descent many contain local minima which yields different clustering results depending on where centroids are initialized

2aiii) False. Any other gradient descent algorithms that is able to minimize the cost function while yielding an updated final centroid position may work.

2aiv) False. Any other method that uses the derivative of the cost function (i.e: SGD) may potentially work

2b

I would initialize the first 5 centroids based on the locations obtained from the K = 5 solutionn. I would then use kmeans++ to initialize the 6th centroid probabilistically. This ensures that the cost for the K = 6 solution is always at least equal to the K = 5 solution.

2c

I would change:

1. Init = 'random' to init = 'k-means++'. This increeases the chance of obtaining the global minimum via a smarter centroid initialization
2. n_init = 1 to n_init = 10. This increases the chance that the global minima is reached with each solution and is also the default used by sklearn.

2d

2di) O(knf)

In order to assign any sample to a cluster, we need to compare all of its features (O(f)) each of the k clusters (O(k). This makes runtime of one sample assignment O(kf). Over n samples, total runtime will be O(knf)

2dii) O(nf + k)

Updating the location of one cluster requires calculating the mean of all the features (O(f)) of k/n samples. Runtime for one cluster is O(kf/n).

Total runtime over k clusters is thus O(kf + k).