Homework 2

Discussion of Question 1:

See the code for the results.

Discussion of Question 2:

For the question,

When training the probabilistic generative classifier, how does the *full covariance* compare to *diagonal covariance* in performance for each of the data sets? Why?

When the data dimension is small, like 2D dataset and 7D dataset, the *full covariance* and *diagonal covariance* don't have sensible difference in performance. While the random state M is 20, for 2D dataset, their prediction accuracies are 97.27%, 97.27%, respectively. For 7D dataset, their prediction accuracies are 100.0%, 100.0%, respectively. While the random state M is 10, for 2D dataset, their prediction accuracies are 98.79%, 99.09%, respectively. For 7D dataset, their prediction accuracies are 100.0%, 100.0%, respectively. Thus, the results show that prediction accuracies fluctuate in a small range with different random states. However, for hyperspectral dataset, the *diagonal covariance* has better performance than *full covariance*. While the random state M is 20, their prediction accuracies are 71.21%, 58.18%, respectively. While the random state M is 10, for 2D dataset, their prediction accuracies are 66.97%, 54.84%, respectively. The reason of these phenomena cannot be totally illustrated, but we can conjecture that data points of every dimension could have more effect on each other when the dimension of datasets is growing, and it may cause prediction errors. Thus, if we ignore the covariance of datasets and only consider standard deviation of the datasets for every dimension, we can get more accurate prediction.

For the question,

When training KNN classifier, what happens as you vary *k* from small to large? Why?

When *k* grows from small to large, the KNN prediction accuracy grows as well firstly, and then decreases. That's because *k* represents the number train data samples, between which a test point considers distance. If the number is large, the test can involve more data samples that belong to the same class. This means the test data is more likely to have the same attribution with data samples from the same class, and it can help the test point to be correctly classified into the right class. However, if *k* is too large, it can involve data points from other classes, and it will cause prediction errors. Thus, the KNN prediction accuracy grows firstly, and then decreases.

Discussion of Question 3:

For the question,

Determine which classifier(s) you would use for each data set and give an explanation of your reasoning. *Hint*: This should incorporate some discussion based on results from cross-validation.

I'd choose KNN classifier for all data sets because with same random state M=20, for 2D dataset, the prediction accuracies for PG classifier with full covariance, PG classifier with diagonal covariance and KNN classifier, are 97.27%, 97.27% and 96.97%. For 7D dataset, the prediction accuracies are

100%, 100% and 100%. For hyperspectral dataset, the prediction accuracies are 58.18%, 71.21% and 81.82%. The accuracy could even be higher if I make *k* an appropriate value, higher than PG classifier for 2D dataset. When *k* is set to 5 rather than 3, the prediction accuracy of KNN classifier for 2D dataset is 97.27%. Thus, I prefer KNN classifier.

Discussion of Question 4:

See .txt files.