Joshua Samuel P. Siy
the answers of the discussion related to KNN(except the extra) is based on the scikit's library of the knn. The difference found between the code-from-scratch KNN and scikit is that the scikit library is slightly more accurate.

2. Test your classifier implementations on the provided data set several times with different parameter settings and using *cross validation*. Provide a PDF entitled hw02.pdf that discussed the following items:

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

Discussions:
1)
The full covariance matrix compared to the diagonal covariance varies slightly from each other. There are some tests that the diagonal covariance is equal to the full covariance in terms of accuracy, but most of the time the diagonal covariance matrix is often a little bit lower than the full covariance matrix. depending on the M values eg. M=90.

```
The accuracy of Probabilistic Generative classifier is:  97.87878787878788 %

The accuracy of KNN classifier is:  96.96969696969697 %

The accuracy of Probabilistic Generative classifier(diagonal) is:  98.18181818181819 %
```

However when it comes to dealing with data sets with alot of features and numbers like the hyperspectral dataset. The diagonal covariance matrix proves to be less accurate than the full covariance matrix.

```
The accuracy of Probabilistic Generative classifier is:  53.333333333333336 %

The accuracy of KNN classifier is:  81.21212121212122 %

The accuracy of Probabilistic Generative classifier(diagonal) is:  31.515151515151512 %
```

Which most of the time. i believe that the full covariance matrix is better than the diagonal covariance matrix when it comes to accuracy of the given data sets.

2) When it comes to selecting the value of "k" it means the number of closest neighbors around the input data. So considering you put a data set and you tried the KNN classification method. you end up selecting the closest possible point/points depending on your value of k. So if i have smaller k, it will select a few neighbors and the distance will not be probably as large compared to having a big value of k so we can say that the data will not be as noisy for prediction. Having bigger k means bigger noise and probably more basis of comparison that could influence that

accuracy of the classifier. So varying k from small to large, means the input data tries to get more neighbors for comparison and the data results become noisier at the same time depending on the gathered information from the neighbor points the accuracy of the classifier could vary. Normally observed is that a good amount of features will help the KNN classify it better while having too much classes at the same time too much data (which some of the features could be identical differing from classes to classes) can make it less accurate compared to using other classifiers such as PGC or regression. Also having k=1 overfits the data.

3. 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.

For the data set of 2D i would choose Probabilistic generative classifier. because the accuracy is bigger compared to PGC diagonal and KNN.For the 7D data set i'd choose KNN because PGC and PGC diag produces 100% accuracy which heavily overfits the data set. aside from Knn having 98% accuracy.

```
The accuracy of Probabilistic Generative classifier is:  100.0 %

The accuracy of KNN classifier is:  98.7878787878788 %

The accuracy of Probabilistic Generative classifier(diagonal) is:  100.0 %
```

For hyperspectral datasets i'd also choose KNN because the PGCand PGC diagonal's accuracy seems to say that the data results are underfitted and is not very accurate

```
The accuracy of Probabilistic Generative classifier is:  53.333333333333336 %

The accuracy of KNN classifier is:  81.21212121212122 %

The accuracy of Probabilistic Generative classifier(diagonal) is:  31.515151515151512 %
```

Extra:

The assignment taker coded the KNN from scratch and the difference between the real library and the scratch coded KNN is: the accuracy of the scratch coded KNN is lower by 0.030% compared to the actual library. The assignment coder suspects that the prediction value may be slightly inaccurate since it looks for the highest amount of repetitions in the matrix generated . What if the matrix is around [1,1,2,2]? how would you say that the class of the data point 2 or 1