Question 1

This homework has only one question, which is a bit longer than the questions in previous assignments, however it also builds on the previous assignments. So, at the end of the day, you should be able to “borrow” a lot of your own code from before and finish this assignment quite easily.

In this experiment, you are to compare various different classification approaches:

1. using complete knowledge of the statistics of the data and computing optimum discriminant functions for the classification (Chapter 2);
2. assuming that you know only the model of the distributions, but not their parameters (Chapter 3);
3. by first reducing the dimensionality of the data set and then classifying it (Chapter 3);
4. assuming that you do not know the underlying distributions and employing Parzen Window (Chapter 4);
   and finally
5. assuming that you do not know the underlying distributions and employing k-NN (Chapter 4).
6. assuming that you do not know anything, but the class labels (Chapter 5)

Initially, you must create four data sets, but later you will be given the data sets to be actually used in your final reports. The data sets that you will create are ‘trivial’ and will be useful for debugging your programs. The ones that I will provide will be more challenging.

While using your data sets, the four testing/training data sets described here MUST be kept the same throughout all parts below. That is, you should create your data points ONCE, save them, and use them for all parts and subparts below. Do NOT create different data for each question or you may end up with strange results!!

The four data sets will be referred to as: 1) Training Data I, with 50 samples in each class; 2) Training Data II, with 500 samples in each class; 3) Testing Data I, also with 500 samples in each class; and finally 4) Testing Data II, with 10000 samples in each class.

All data sets must consist of 5-d points divided in 3-classes with the underlying normal distributions \( p(\mathbf{x}|\omega_i) = N(\mu_i, \Sigma_i) \), where:

\[
\mu_1 = [2, 3, 1, 5.5, 8.7]'
\mu_2 = [-4.5, 6, -1, 3, 10]'
\mu_3 = [1.2, -2.3, 1.5, -0.5, 2.7]'
\Sigma_1 = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 0.5 & 0 & 0 & 0 \\
0 & 0 & 2.5 & 0 & 0 \\
0 & 0 & 0 & 0.7 & 0 \\
0 & 0 & 0 & 0 & 3.5 \\
\end{bmatrix}
\Sigma_2 = \begin{bmatrix}
2 & 0 & 1 & 0.5 & 0 \\
0 & 3.5 & 0 & 0 & 0.6 \\
1 & 0 & 4.5 & 1.2 & 0 \\
0.5 & 0 & 1.2 & 1.6 & 0 \\
0 & 0.6 & 0 & 0 & 2.5 \\
\end{bmatrix}
\Sigma_3 = \begin{bmatrix}
4.2 & 0 & 1.3 & 2.5 & 1.4 \\
0 & 5 & 0 & 0 & 3.6 \\
1.3 & 0 & 4.5 & 4.2 & 0 \\
2.5 & 0 & 4.2 & 5.6 & 0 \\
1.4 & 3.6 & 0 & 0 & 7.5 \\
\end{bmatrix}
\]

You will assume that all states of nature are equally probable.

---

1The sizes of the data sets above are not typical to real-life classification problems. The choices above were made solely to create interesting situations for discussion in your report.
Part I

Here, you will first (in\ a\ and\ b) use complete knowledge about the data. Then (in\ c\ and\ d), you will “forget” that you know the means and covariances, and use ML to estimate them.

a) Classify the Testing Data I, using the given statistics above and the Bayes decision rule. Compute the confusion matrix.

b) Repeat part a) for the Testing Data II.

c) Compute the ML estimates (\hat{\mu}_i\ and\ \hat{\Sigma}_i) for each class using the Training Data I and classify the Testing Data II using Bayes decision rule. Compute the confusion matrix.

d) Repeat part b), but this time use the Training Data II for the ML and then classify the Testing Data II again.

e) Make comments on each of the results above and then compare them (‘compare’ does not mean to say “this was better than that”, but to say why that was the case).

Part II

In this part, you will reduce the dimensionality of the data by applying MDA. That is,

a) Using MDA on the Training Data I, find the matrix W such that \( \bar{y} = W \bar{x} \). (What is the expected dimension of \( \bar{y} \)?) For each class, compute the \( \hat{\mu}_i \) and \( \hat{\Sigma}_i \) of the new reduced-space variable \( \bar{y} \).

b) Apply the transformation W above to the Testing Data II and classify it using Bayes decision rule. Compute the confusion matrix.

c) Repeat parts a) above, but this time use the Training Data II to compute the matrix W and the \( \hat{\mu}_i \) and \( \hat{\Sigma}_i \) for each class of the new reduced-space variable \( \bar{y} \). Then classify the Testing Data II again. Compute the confusion matrix.

d) Comment on the results from this Part II and then compare these results with the results from the previous Part I.

Part III

Now, you will completely forget that you know anything about any of the distributions and/or their parameters and apply a non-parametric approach to classification.

a) First, you will apply Parzen Window to each class in the Testing Data II using the Training Data I and a hypercube window function with \( h_n = 0.7 \). You must classify the Testing Data II according to the maximum posterior probability. Compute the confusion matrix. Repeat the classification for \( h_n = 0.1 \) and \( h_n = 5 \).

b) Repeat part a), but this time use the Training Data II for the Parzen Window and classify the same Testing Data II with, again, \( h_n = 0.1 \), \( h_n = 0.7 \), and \( h_n = 5 \).

c) Repeat part b) using a Gaussian kernel (a Gaussian window function) with \( \sigma = 0.1 \). Then repeat this part with \( \sigma = 0.7 \) and \( \sigma = 5 \).

d) Comment on the results above. Compare them with the results from the previous Parts above (I and II).

Part IV

Once again, you will forget that you know anything about any of the distributions and/or their parameters and apply another non-parametric approach.

a) Using the Training Data I, you will classify the Testing Data II using k-Nearest-Neighbour. Use \( k_n = \sqrt{n} \).

b) Repeat part a), but this time use the Training Data II to classify the Testing Data II.
c) Repeat part b) using your choice for $k_n = f(n)$.

d) Comment on the results above. Compare them with the results from the previous Parts above, in special Part III.

**Part V**

Once again, you will forget that you know anything about any of the distributions and/or their parameters and apply another non-parametric approach. This time, repeat Part IV above using a linear classifier and the Perceptron criterion – for part a) and c), you obviously cannot pick a value for $k_n$, so, instead, use $\eta = 1/2$ for part a) and then use $\eta = 1/\sqrt{k}$ for part c), where $k$ is the iteration.