Report for ex3 - 10136960

1) What are those “parameters” that need learning in discrete and continuous naïve Bayes?
2) How do you store the learned “Parameter List” in your programme for Parts 1 & 2?

3) What are test results on all the given data sets described in Parts 1 & 2?

4) What are the motivation and setting(s) in your cross-validation experiments?

5) Based on your observation and analysis on experimental results achieved in Parts 1 & 2, can you grasp any non-trivial implication? If any, in your report, you must explicitly describe your experimental evidence or theoretical justification that leads to such an implication.

1. Discrete - need to learn feature values and class values.
   Continuous - need to learn means and stdevs for each feature value given a class.
   Also must learn the prior probabilities for the original label set.

2. Part 1: there are 57 attribute_value x class_number matrices, where each one of the 57 represents the probability of an attribute having a value, given a class number.
   Part 2: the standard deviations for each of the 57 features are in two columns (one for each class).
   The means are also represented in the same way.

3.

Dataset	Accuracy (%, 3sf)
av2_c2	89.1
av3_c2	89.4
av7_c3	86.3
avc_c2	78.5
spambase	Avg 81.4, stdev 14.7

4. My cross validation is 10-fold, because it allows for rigourous training and a more reliable accuracy result due to more a high number of distinct testing samples, being tested over all 10 partitionings, exactly once.

5. From my experiments, I made the following implications
   a. The accuracy of classification for a continuous dataset (avc_c2, for example) was lower than that of the discrete datasets on average (78.5% as compared to 89.4% respectively).
      This would most likely be due to the discrete datasets having far less attribute values than that of a continuous dataset, thus making it easier for a label to have a certain feature value.
   b. During the 10-fold cross validation, the accuracy for each k value generally

decreased, as I got the following in my accuracy matrix:

Accuracy (%)	K-value
97.4	1
96.3	2
95.2	3
90.0	4
77.8	5
80.0	6
79.6	7
73.9	8
75.9	9
48.3	10

This trend is due to the way the spambase data is labelled - it starts off with yes/1 values but it is somehow worse when it comes to classifying no/0 values.