s4. a) **SVM(Support Vector Machine)** gave best accuracy of around **98** %.

SVM is Non-probabilistic, i.e., it does not assume the Gaussian distribution or independence of input features. We have used svm.LinearSVC - Linear Support Vector Classification, which divides the clusters of and takes the minimum distance of test sample from that cluster which works well with lesser number of features.

b)

SVM’s performance highly depends on the 2 fold strategy that we used and also on the features and number of samples. It is not the case that it is the best algorithm as it depends on those parameters also. Below are the individual reasons of the algorithms which could have affected their overall accuracy.

**LDA** gave an accuracy of 97.3% because it assumes the data is Gaussian and each feature has the same variance. Also, it can only learn linear boundaries. It also depends on the lesser number of features and use of 2 fold cross validation.

**Naïve Baysian** gave an accuracy of 96%, because it assumes a gaussian distribution and assumes features are independent. It predicts the output based on the conditional probability which couldn’t perform better than the LDA. Also, it can’t perform best when we use half the data as training set.

**Linear regression** gave an accuracy of **96**.67% when used with linear and polynomial of 2 degree and gave an accuracy of **92**% when used with polynomial of degree 3. This algorithm establishes a linear relationship among the features based on the statistical analysis which performs better with higher number of features. **The polynomial of degree 2/3** are not naturally flexible enough to capture more complex patterns. Fitting polynomials and adding the right interaction terms is time-consuming. Data might not fit a polynomial curve.

**kNN** gave an accuracy of **94**% because it finds the nearest neighbours on the plot for the prediction and can’t predict accurately when there is an overlapping of value of features for different classes.

**QDA** had an accuracy of **96**% because it assumes a gaussian distribution and assumes features are independent. Also, I think QDA perform better when data is high dimensional.

**Decision Tree Classifier** gave an accuracy of **95.3**% because it is based on heuristic algorithms such as the greedy algorithm where locally optimal decisions are made at each node and therefore it cannot guarantee to return the globally optimal decision tree. Also, Decision trees can be unstable because small variations in the data might result in a completely different tree being generated.

**Random Forest** and **Extra tree** classifiers gave accuracy of **94**% and **95.3**% respectively. Both can have same weakness as Decision Tree but it creates multiple decision trees where chances of having these weaknesses are less

**Nueral Network** gave an accuracy of **96**% because different random weight initializations can lead to different test results and it is sensitive to feature scaling.

7.

a. K fold cross validation is **not** used.

b. 80% of data is used as training set.

c. 360 samples in test set

d. 1437 samples in training set

e. 64

f. 64

g. 10

h. classes [0, 1, 2, 3, 4, 5, 6, 7, 8, 9 ]