**ANALYSIS:**

* + Classification using Support vector machines :
    - A new kind of method used for classifying non-probabilistically but entirely using the distance metric between features to classify a new datapoint. According to this technique, datapoints get mapped into the feature space based on their features, and datapoints of the dame class tend to be associated in proximity.
    - 2 classes can be distinguished by cutting this feature space by a hyper plane, which is basically a line in a 2 dimensional feature space. Finding the optimal equation of the hyperplane is done based on the fact that the hyper plane should have the maximum margin between the two classes.
    - This hyperplane is decided by a set of datapoints of either of these classes that are at the boundaries of their clusters. These datapoints are called ‘support vectors’. Vectors, as each data point is basically a vector of its dimensional features, and support, because these are the vectors that help in deciding the optimal hyperplane.
    - Sometimes, a linear hyperplane will not be able to draw the distinction between the 2 classes. In this case we increase the dimension of the feature space using a function of these lower dimensional vectors. This is called the ‘kernel trick’.
    - Once the optimal hyperplane or ‘perceptron of optimal stability’ is found, classification of further data points becomes direct.
    - For multi-class classification, the above mentioned method is repeated for every class abstracting the other class, and finally the closest class is chosen. This is called the ‘one versus all’ method.
    - To run the model, we first import the svm from sklearn and use the svm.SVC() to an instance of the SVC object.
    - The fitting and prediction is done as usual using the general fit(), predict() and score() functions of the estimators.
    - There are different model variations in the svm module that use different kernel functions to do the kernel trick so as to account for non linear hyperplane estimation. The kernels used are linear, Gaussian radial basis function & polynomial function of order 3. These provide different boundaries and the efficiency of a particular model depends on the case study and the data set. The LinearSVM() produces different results from the SVM() using the linear kernel as the former minimises the squared hinge loss while the latter minimises the normal hinge loss. Moreover the former uses one-vs-all for multi class classification while the latter uses one-vs-one multi class classification.
    - Plots for each of these variations are done. For the purpose of simplicity and ability to visualise in 2 dimensions, the number of features /dimensions is limited to 2.