Project Name:
To build a binary classifier based on SVM (Support Vector Machine) and submit the predictions on the testing data.

Specifications:
Both the training and testing datasets have 200 rows and 2 columns each. The training dataset has the corresponding labels attached with it whether they belong to class 0 or 1.

Demonstration:
I used K-fold cross-validation on the training data (K =10) and tried out three different kernels (linear, polynomial and radial) SVM with degree = 3 and different values of gamma ranging from 0.5 to 100 . Here, 0.5 is the value of 'auto' where auto = 1/ number of features in the training data = ½ = 0.5
I used 10-fold cross-validation technique so as to prevent any overfitting on the training data.
I have reported the different values of accuracy which I got for the 3 kernels below:

Linear SVM:

```
[ 0.8   0.85  0.7   0.75  0.85  0.85  0.8   0.95  0.85  0.85]
```

Polynomial SVM:(Degree =3 and Gamma = 100)

```
[ 0.8   0.75  0.75  0.8   0.85  0.8   0.85  0.85  0.85  0.7 ]
```

Radial SVM: (Degree = 3 and Gamma = 100)

```
[ 0.75  0.75  0.9   0.85  0.8   0.8   0.8   0.95  0.85  0.95]
```

Using K-fold cross-validation technique, we can go on to conclude that Radial SVM goes on to give the highest accuracy as we report 4 values of cross-validation accuracy which are greater than or equal to 90%.

Conclusion:
Therefore, I decided to train my model with **radial kernel SVM (gamma = 100) and degree of polynomial =3** and went on to achieve an accuracy of more than **98%** on the training dataset.

I used this kernel with all the same specifications to report the results on the testing data.
I have plotted the kernels (each for linear SVM, polynomial SVM and radial SVM) juxtaposed with the training data in the Jupyter notebook which I have submitted .

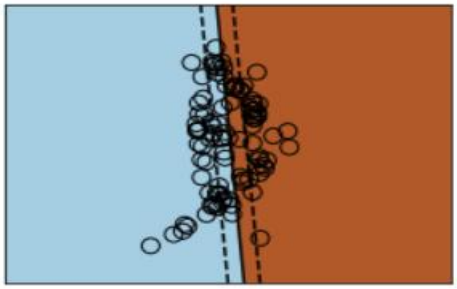

Fig 1: Linear SVM juxtaposed with the training set

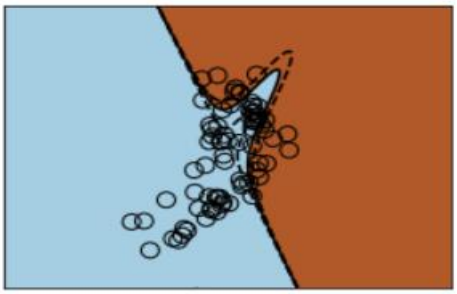

Fig 2: Polynomial SVM juxtaposed with the training set

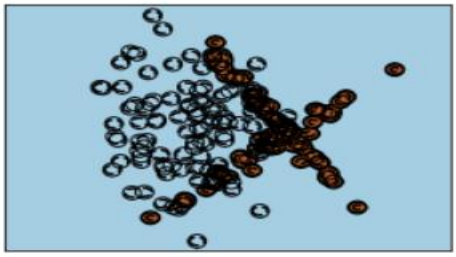

Fig 3: Radial SVM juxtaposed with the training set (Optimal Classifier which went on to achieve 98% accuracy on the training data)