Lab 11 Report

Nakkina Vinay (B21AI023)

Question 1

Subpart 1:

- Import the necessary libraries and read the data using read_csv()

	variance	skewness	curtosis	entropy	class
0	3.62160	8.66610	-2.8073	-0.44699	0
1	4.54590	8.16740	-2.4586	-1.46210	0
2	3.86600	-2.63830	1.9242	0.10645	0
3	3.45660	9.52280	-4.0112	-3.59440	0
4	0.32924	-4.45520	4.5718	-0.98880	0
...	...	...	...	...	...
1367	0.40614	1.34920	-1.4501	-0.55949	1
1368	-1.38870	-4.87730	6.4774	0.34179	1
1369	-3.75030	-13.45860	17.5932	-2.77710	1
1370	-3.56370	-8.38270	12.3930	-1.28230	1
1371	-2.54190	-0.65804	2.6842	1.19520	1

1372 rows × 5 columns

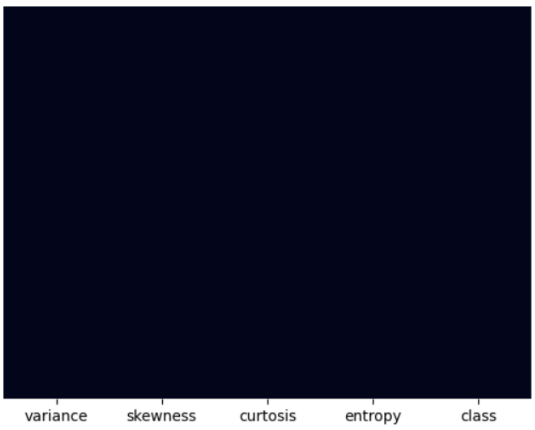
- Performing info() and describe() on the dataset

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1372 entries, 0 to 1371
Data columns (total 5 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   variance  1372 non-null   float64
 1   skewness  1372 non-null   float64
 2   curtosis  1372 non-null   float64
 3   entropy   1372 non-null   float64
 4   class     1372 non-null   int64  
dtypes: float64(4), int64(1)
memory usage: 53.7 KB
```

	variance	skewness	curtosis	entropy	class
count	1372.000000	1372.000000	1372.000000	1372.000000	1372.000000
mean	0.433735	1.922353	1.397627	-1.191657	0.444606
std	2.842763	5.869047	4.310030	2.101013	0.497103
min	-7.042100	-13.773100	-5.286100	-8.548200	0.000000
25%	-1.773000	-1.708200	-1.574975	-2.413450	0.000000
50%	0.496180	2.319650	0.616630	-0.586650	0.000000
75%	2.821475	6.814625	3.179250	0.394810	1.000000
max	6.824800	12.951600	17.927400	2.449500	1.000000

- Checking for any empty values in the dataset

'Cream Lines in the graph indicates the empty values'



- Since there are no cream lines the dataset has no missing values

- Splitting the data into X and y and standardizing the data with StandardScaler
- Now splitting the data into train, test and validation sets in the ratio of 70:20:10
- Printing the shapes of the sets

```
Training set shape: (960, 4) (960,)
Validation set shape: (136, 4) (136,)
Testing set shape: (276, 4) (276,)
```

Subpart 2:

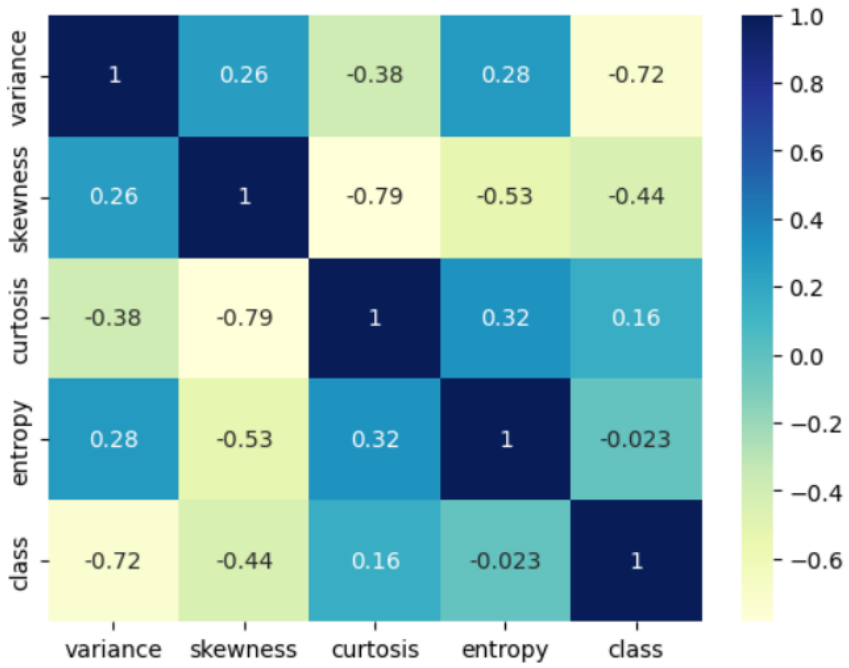
- Defining a list of values for C

```
C_values = [0.1, 1, 10, 100, 1000]
```

- Training the SVM classifier for each value of C and evaluating its accuracy on thr entire dataset and printing them

```
C = 0.1, Accuracy = 0.9818840579710145
C = 1, Accuracy = 0.9855072463768116
C = 10, Accuracy = 0.9927536231884058
C = 100, Accuracy = 0.9927536231884058
C = 1000, Accuracy = 0.9927536231884058
```

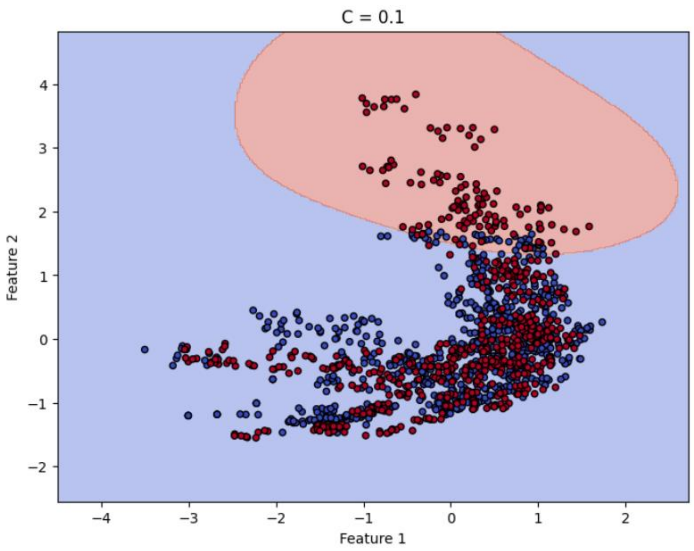
- Now finding the covariance of the dataset and using the heatmap printing it
- We can clearly see that Curtosis and Entropy has high correlation with the class column

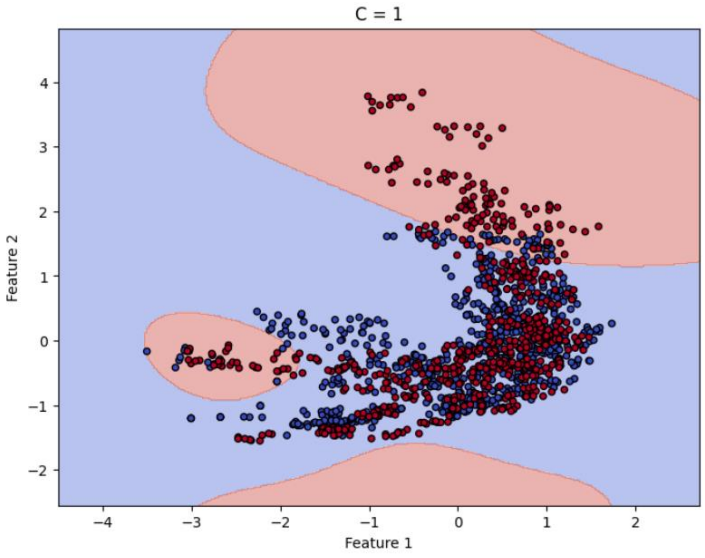


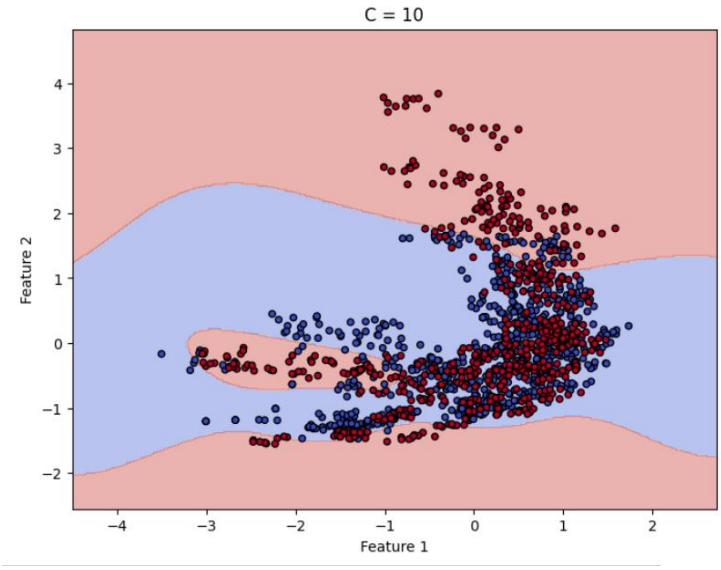
- Now selecting those two features having the highest correlation with the target and extracting them from the data

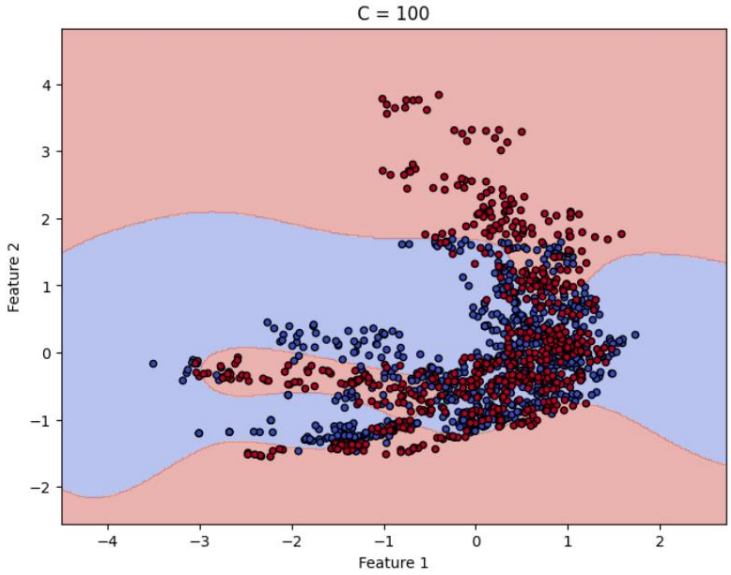
```
Selected features: Index(['entropy', 'curtosis'], dtype='object')
```

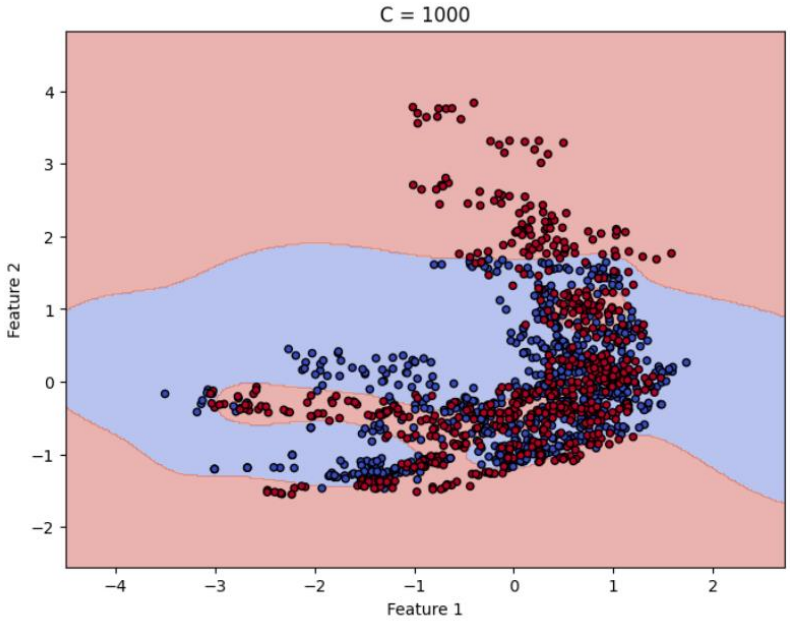
- Plotting the decision boundaries for each value of C using the 2 features from the dataset, who has the highest correlation with the target.











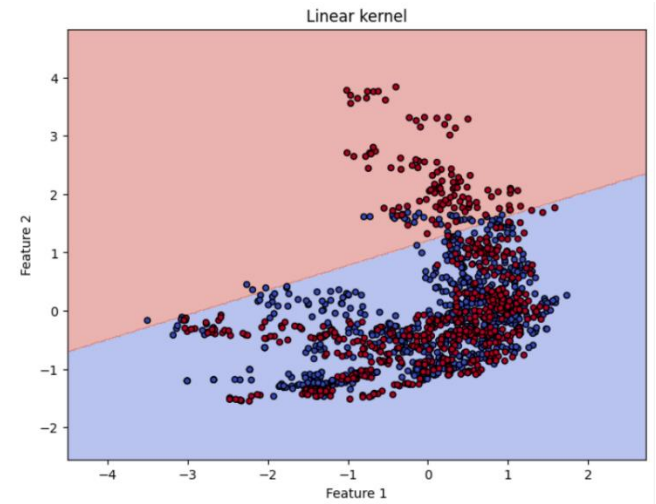
Subpart 3:

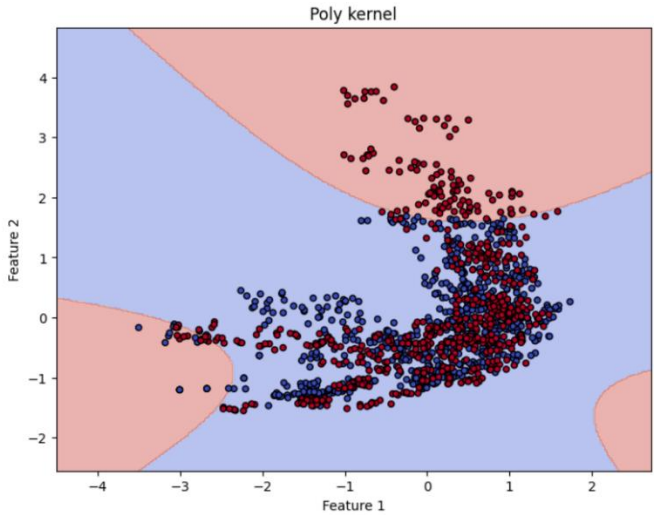
- Taking four types of kernels

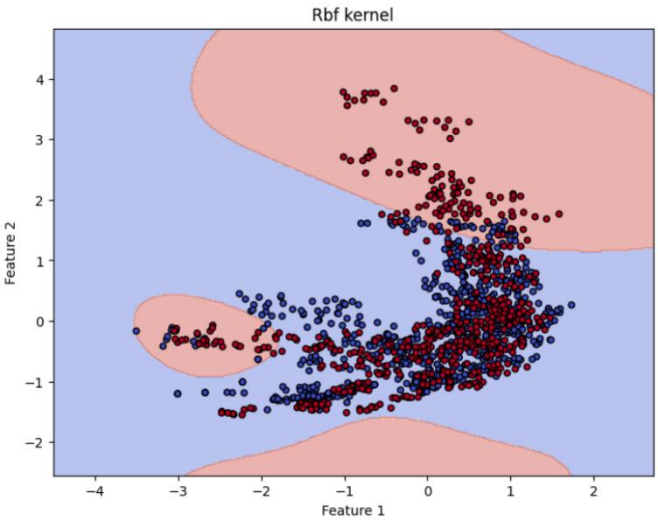
```
kernels = ['linear', 'poly', 'rbf', 'sigmoid']
```

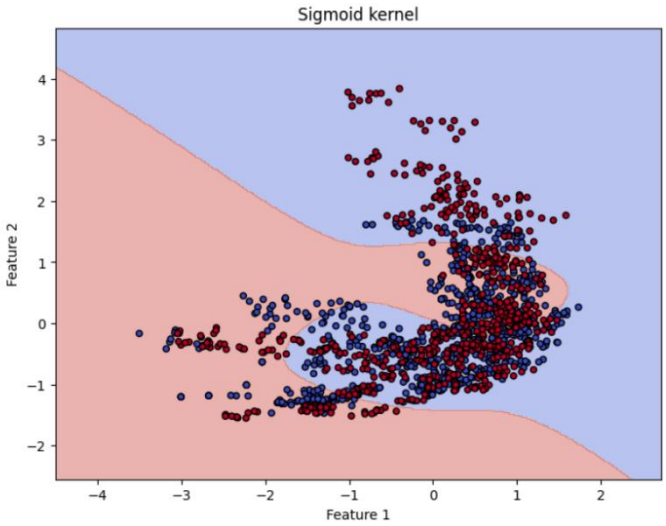
- Training the SVM models using these four different kernels and predicting the accuracy and plotting the decision boundaries using the 2 features from the dataset, who have the highest correlation with the target.

```
Accuracy is  :  58.08823529411765 when kernel is  linear

Accuracy is  :  58.82352941176471 when kernel is  poly

Accuracy is  :  59.55882352941176 when kernel is  rbf

Accuracy is  :  39.705882352941174 when kernel is  sigmoid
```







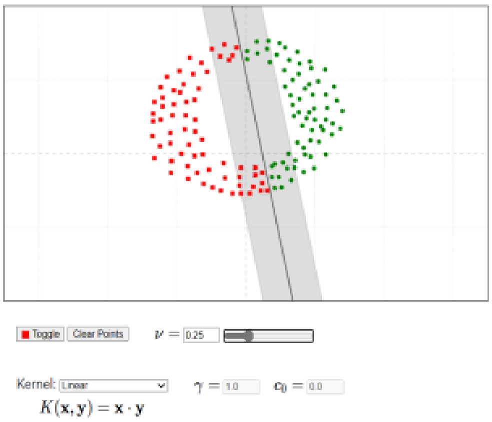


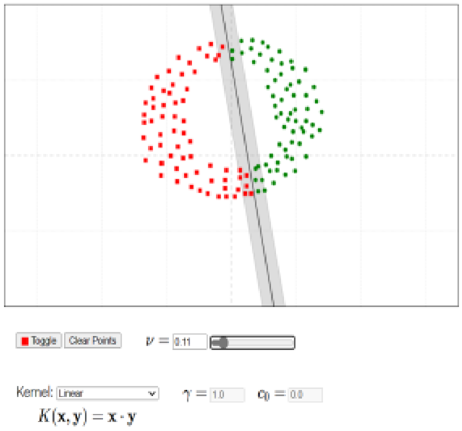
From this we can say on the above dataset model with the kernel =’rbf’ performed well when compared to other kernels and sigmoid is performing worst for the above dataset

Subpart 4:

First dataset (linearly separable):

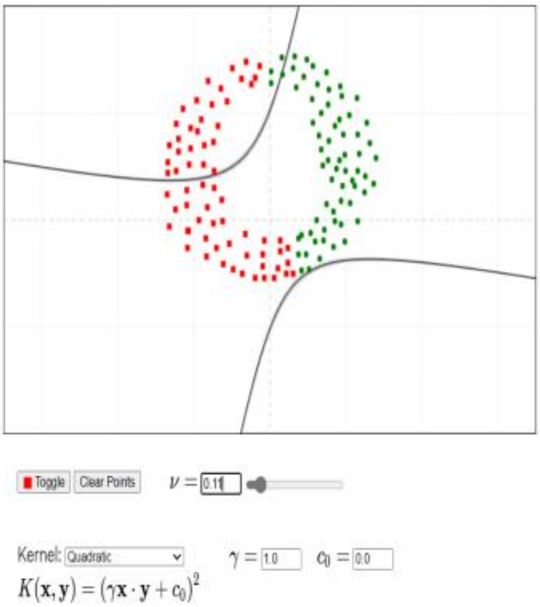
Linear kernel: this is a linearly separable data and therefore linear kernel can separate the kernels well.

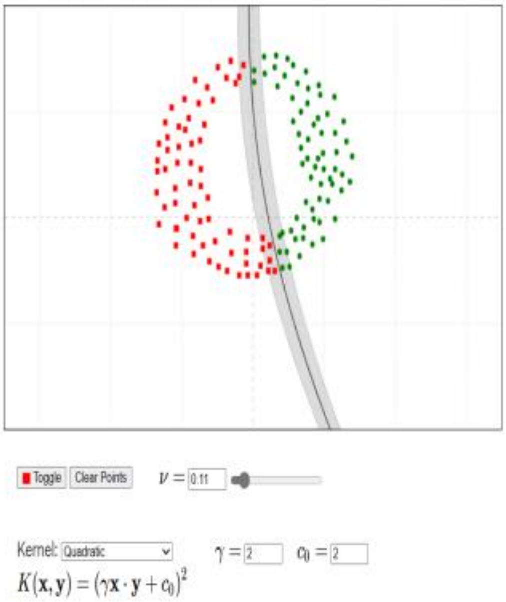


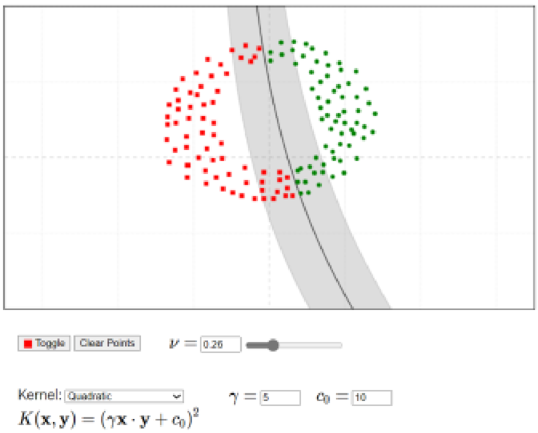


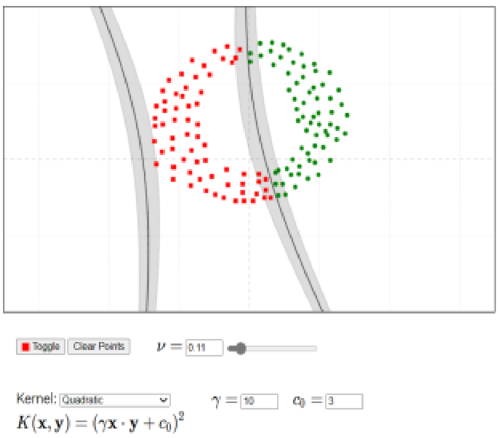
Increasing the regularization leads to the model paying more attention to the distance of separability between points from different classes

Quadratic kernel:



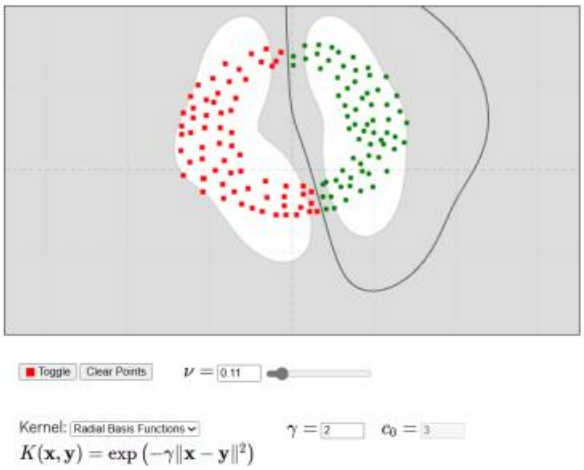


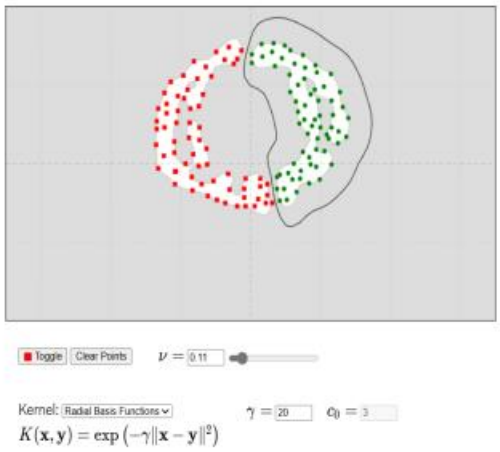




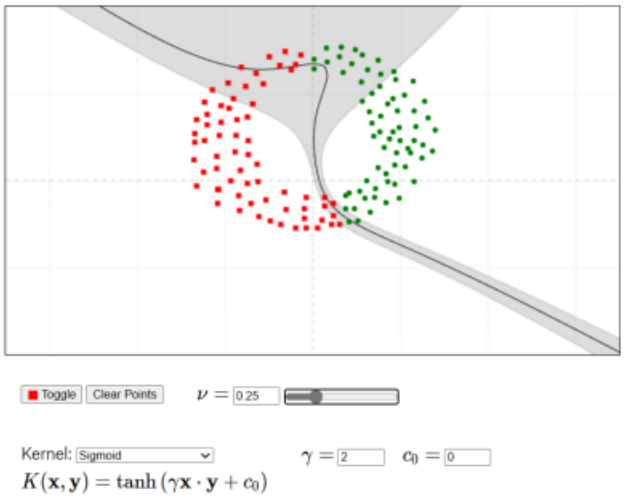
Increasing the value of co leads to a better decision boundary in case of linearly separable data.

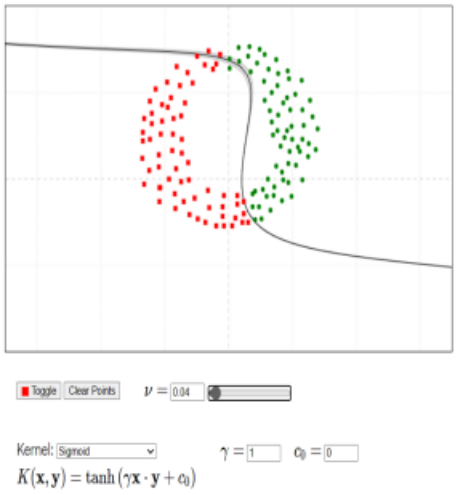
RBF kernel: this is able to classify very well as it projects the data points into infinite dimension space which leads the model to learn all the complexities.

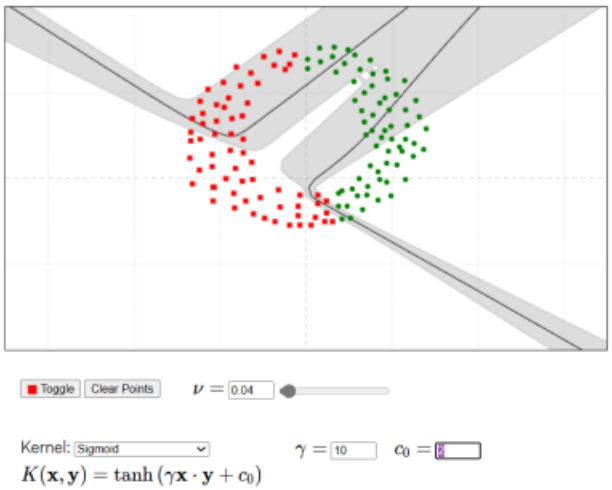


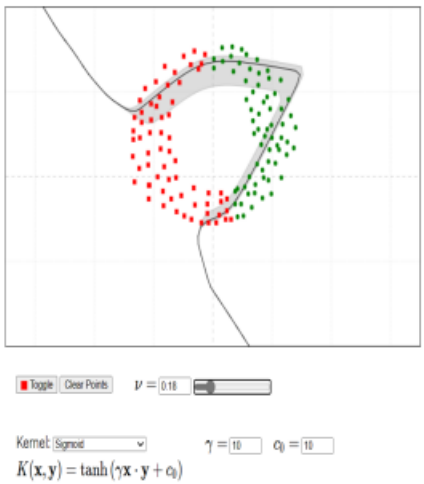


Sigmoid kernel





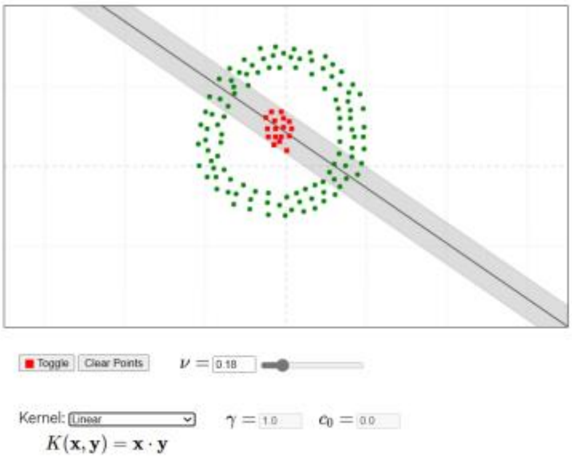


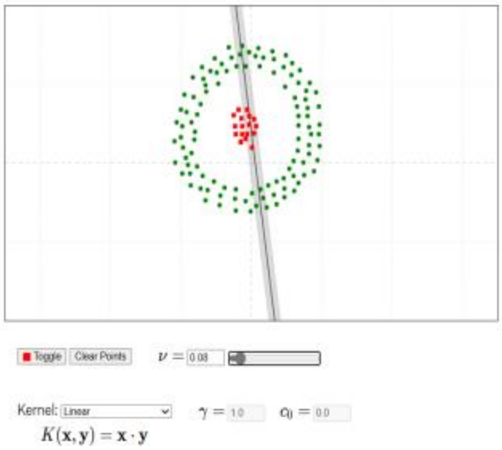


The decision boundaries of the sigmoid kernel are very unstable and change very much even with minimal fluctuations in parameters. It performs the best with low margin, co=0 and low value of gamma.

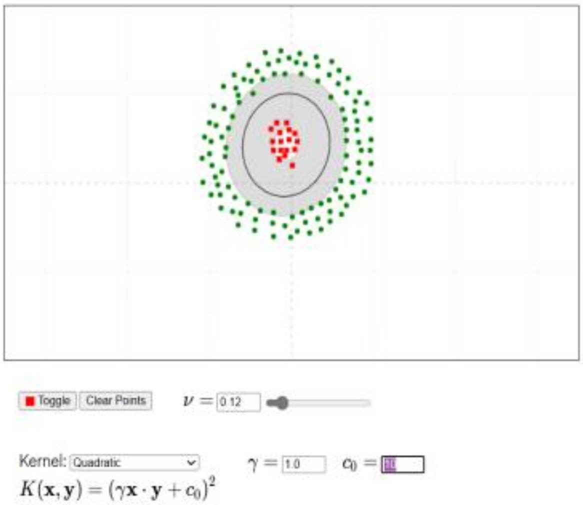
Second dataset (non-linearly separable):

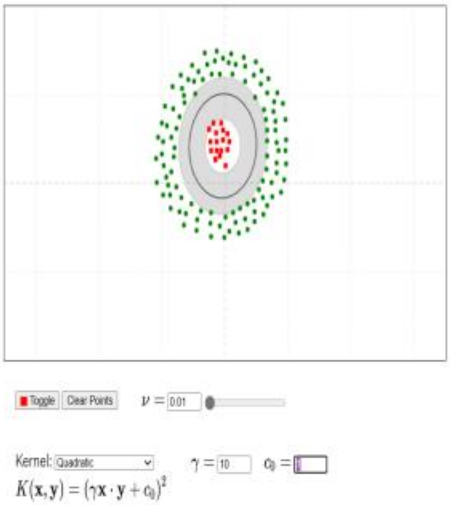
Linear: this kernel is not able to correctly classify the cluster as the data points are not linearly separable

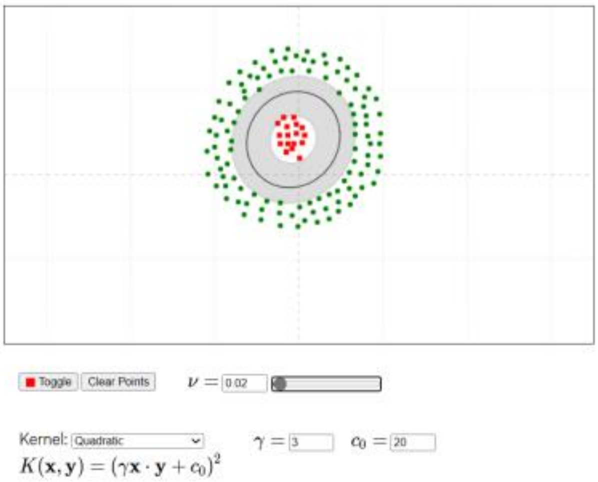


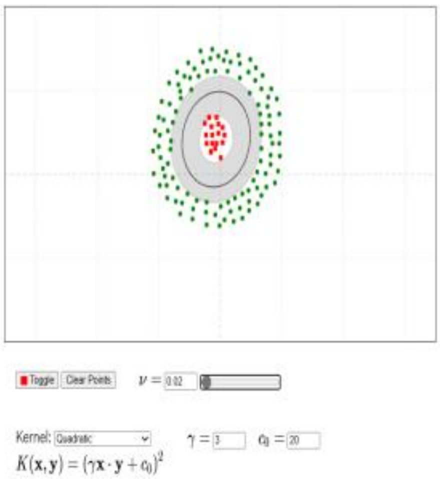


Quadratic kernel: this kernel works very well in case of the kind of data that we have taken and is able to classify the clusters without errors



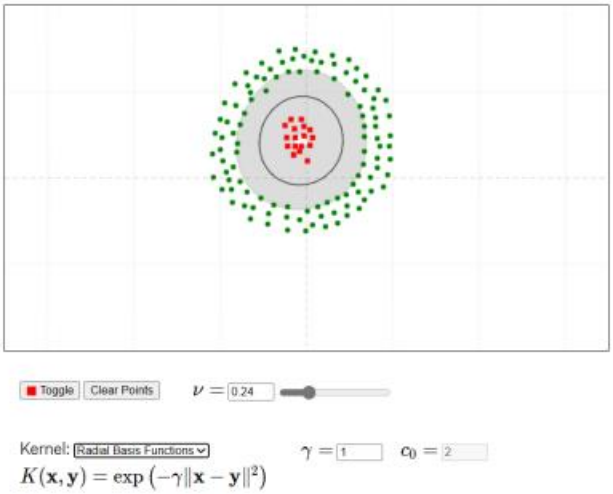


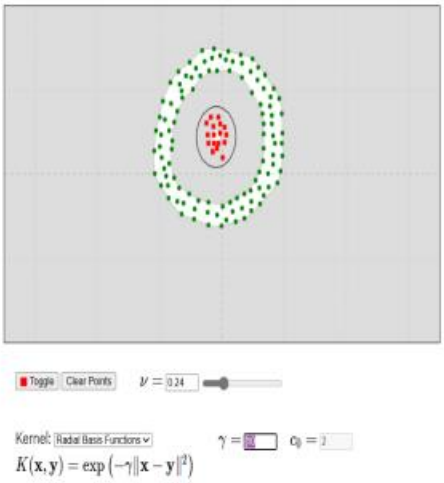




Changing the gamma value doesn't affect much on the decision bounday and the regularization value and the margin need to be less for obtaining a better decision boundary.

RBF kernel: this kernel is also able to provide us with good decision boundaries with very low errors





Sigmoid kernel