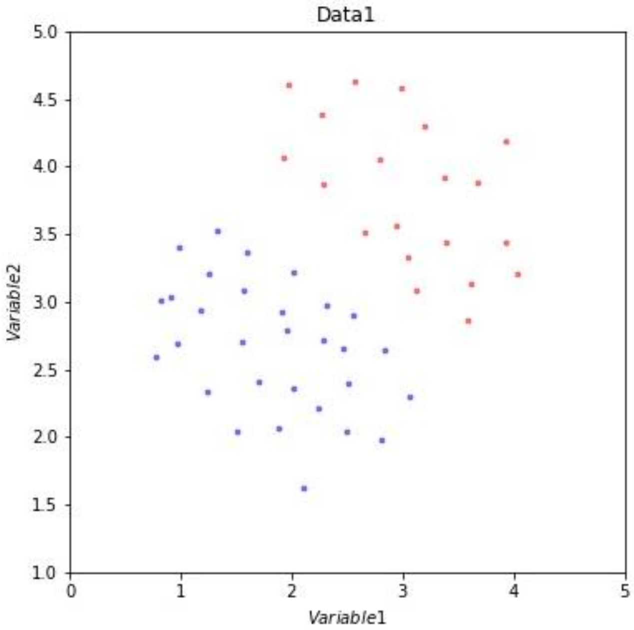
SVM using CVXOPT

1. **SCATTER PLOT FOR DATA 1 –**
   Blue points belong to class -1 and red points belong to class 1



2. **SVM ON DATA 1 –**
   Dual Problem –

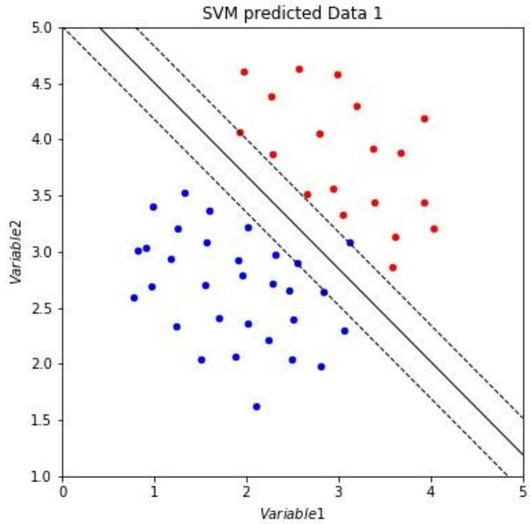
$$\max[\sum_{n=1}^{N} \lambda_n - \frac{1}{2}\{\sum_{n=1}^{N}\sum_{m=1}^{N} \lambda_n \lambda_m t_n t_m x_n^T x_m\}], \text{sub. to } \lambda \geq 0 \ \& \sum_{n=1}^{N} \lambda_n t_n = 0 \ \Rightarrow (1)$$

I used python package named CVXOPT to solve the above given optimization problem.
The CVXOPT solves the following problem –

$$\min[\frac{1}{2}x^T P x + q^T x], \text{subject to } Gx \leq h \text{ and } Ax = b \Rightarrow (2)$$

Equation (1) can be converted to equation (2) by multiplying it by (-1); after this we have,
G => Identity matrix with containing -1
h = > Zero matrix
P => H matrix => H = t_n*t_m*x_n^T *x_m
q = Matrix containing all values of -1
A = t_n (class values, i.e. -1 and 1)
b => Zero Matrix

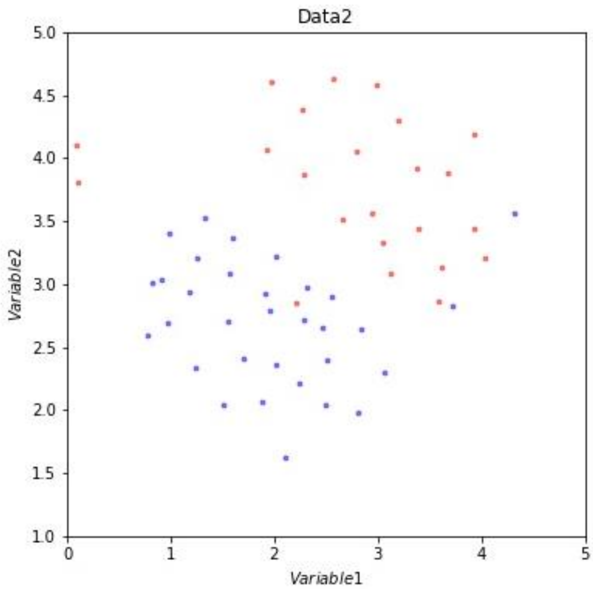
Using this we obtain the following graph for predicted values with decision boundary –



Accuracy obtained was 98%.
Weights – [2.54140955, 3.0693502]
Bias – [-16.372343517060063]

3. **MODIFIED SVM ON DATA 2 –**

Scatter Plot for Data2 –
Blue points belong to class -1 and red points belong to class 1



Modified optimization problem –

$\max[\sum_{n=1}^{N} \lambda_n \; - \frac{1}{2}\{\sum_{n=1}^{N}\sum_{m=1}^{N} \lambda_n \, \lambda_m t_n t_m x_n^T x_m\}]$, sub. to $0 \leq \lambda \leq \mathrm{C}$ & $\sum_{n=1}^{N} \lambda_n t_n = 0$ **=> (3)**

For this as well, I followed the same strategy of using CVXOPT solver to solve the above given optimization problem; the minor changes which are necessary include –

$$-\lambda \leq 0$$
$$\lambda \leq C$$

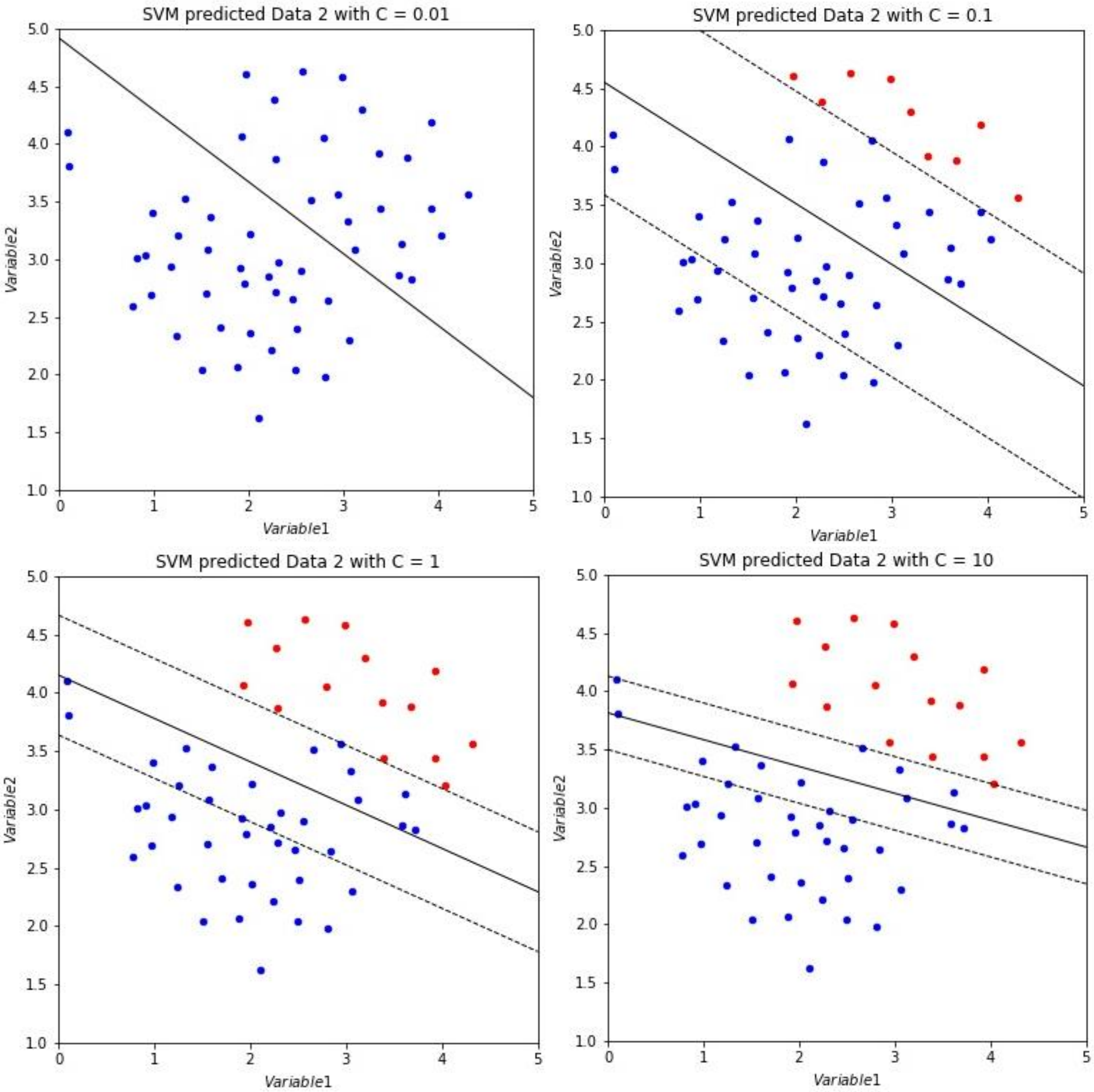
Thus h in equation (2) will contain zeros in the first half and C in the second half. Also, G will have an identity matrix of -1 in the first half and identity matrix of 1 in the second half.

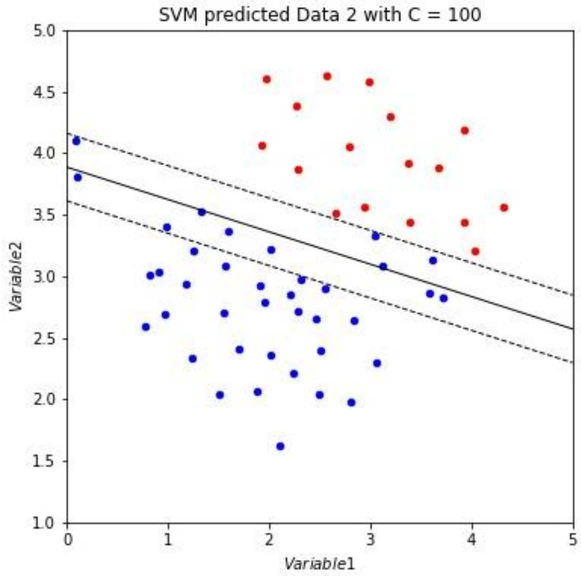
The following values of C were used for the analysis –

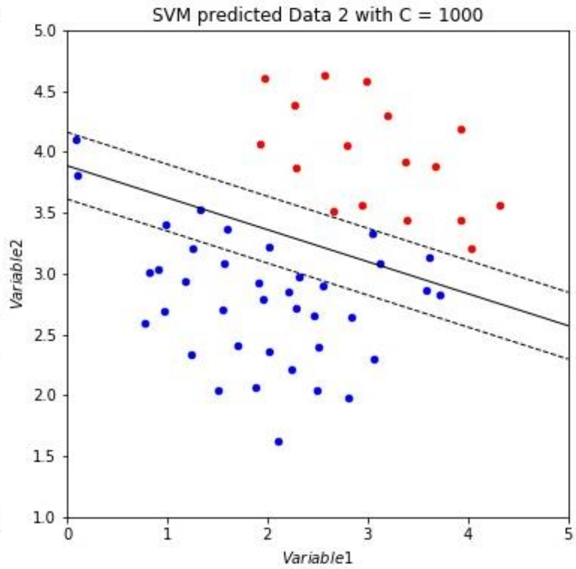
[0.01, 0.1, 1, 10, 100, 1000, 10000, 100000, 1000000, 10000000]

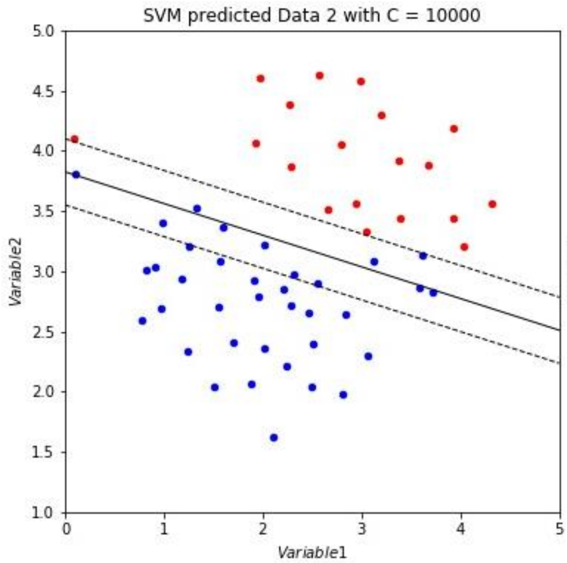
Using this we obtain the following graphs for predicted values with decision boundary for different values of C –

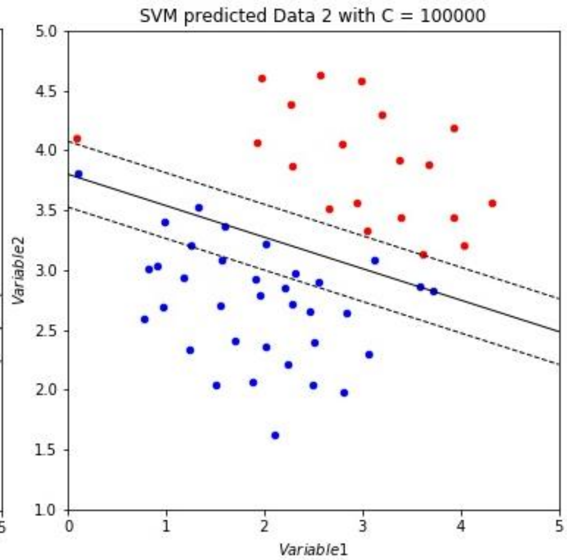
Blue points belong to class -1 and red points belong to class 1

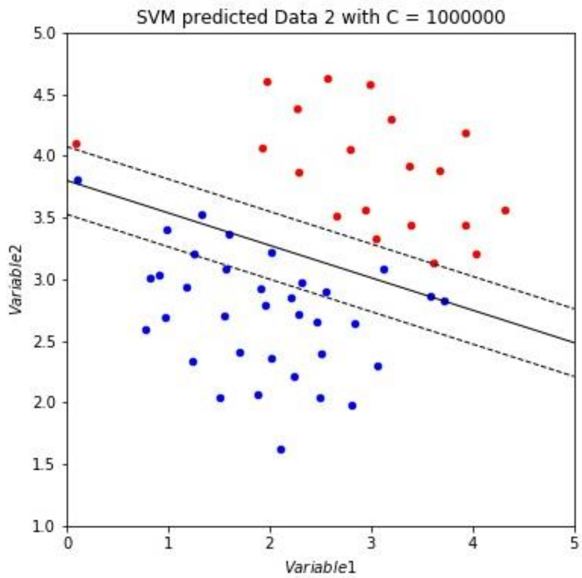


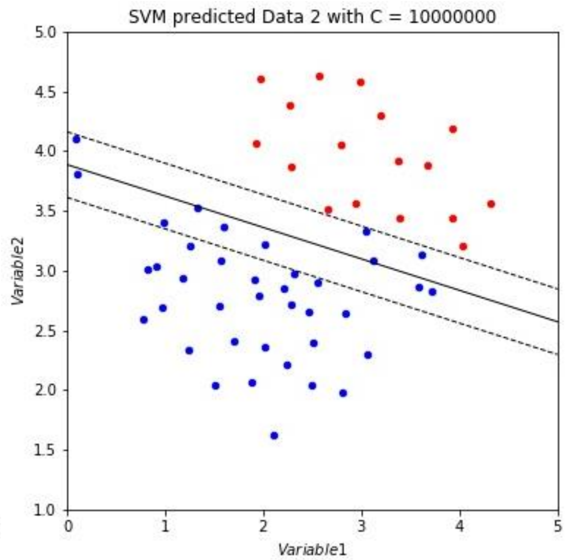

SVM predicted Data 2 with C = 100
Variable1
Variable2


SVM predicted Data 2 with C = 1000
Variable1
Variable2


SVM predicted Data 2 with C = 10000
Variable1
Variable2


SVM predicted Data 2 with C = 100000
Variable1
Variable2


SVM predicted Data 2 with C = 1000000
Variable1
Variable2


SVM predicted Data 2 with C = 10000000
Variable1
Variable2

The following accuracies were obtained for different values of C –

```
accuracy for 0.01 = 58.18181818181818
accuracy for 0.1 = 70.9090909090909
accuracy for 1 = 81.81818181818183
accuracy for 10 = 83.63636363636363
accuracy for 100 = 85.45454545454545
accuracy for 1000 = 85.45454545454545
accuracy for 10000 = 89.0909090909091
accuracy for 100000 = 90.9090909090909
accuracy for 1000000 = 90.9090909090909
accuracy for 10000000 = 85.45454545454545
```

The models with highest accuracies had C values equal to 10^5 and 10^6 respectively.