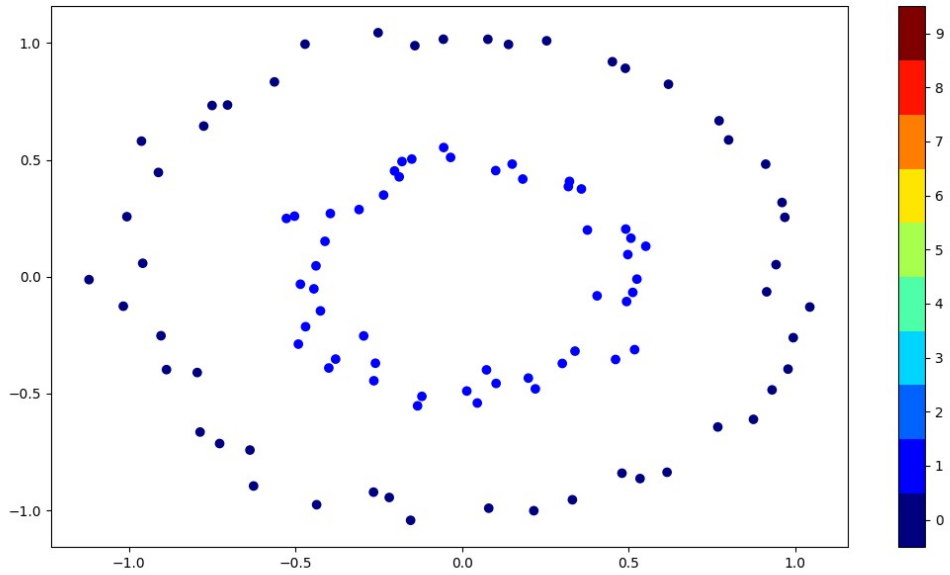
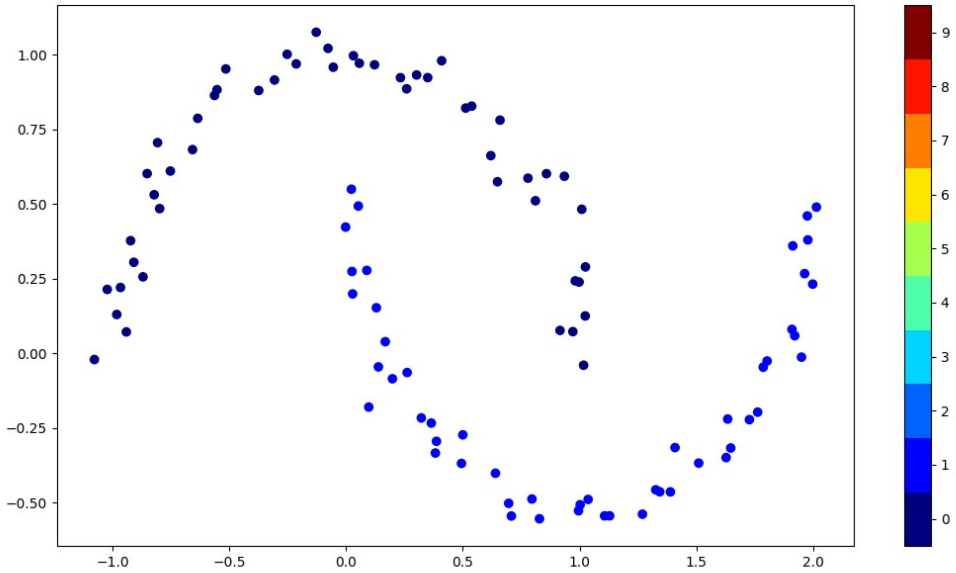
Exploring Datasets and Kernels:

1.
Dataset 1



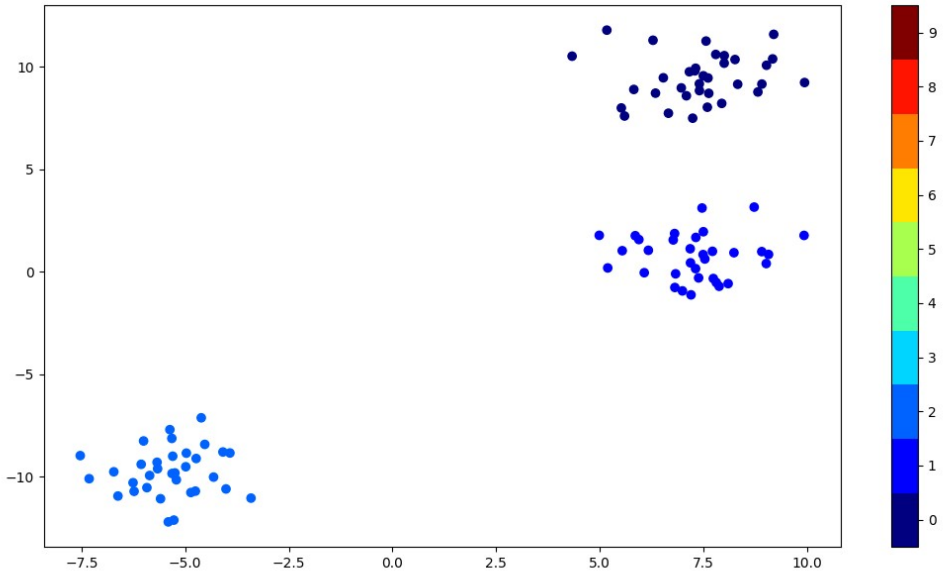

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python explore_1.py --data data_1.h5
{0: 50, 1: 50}
```

Dataset 2



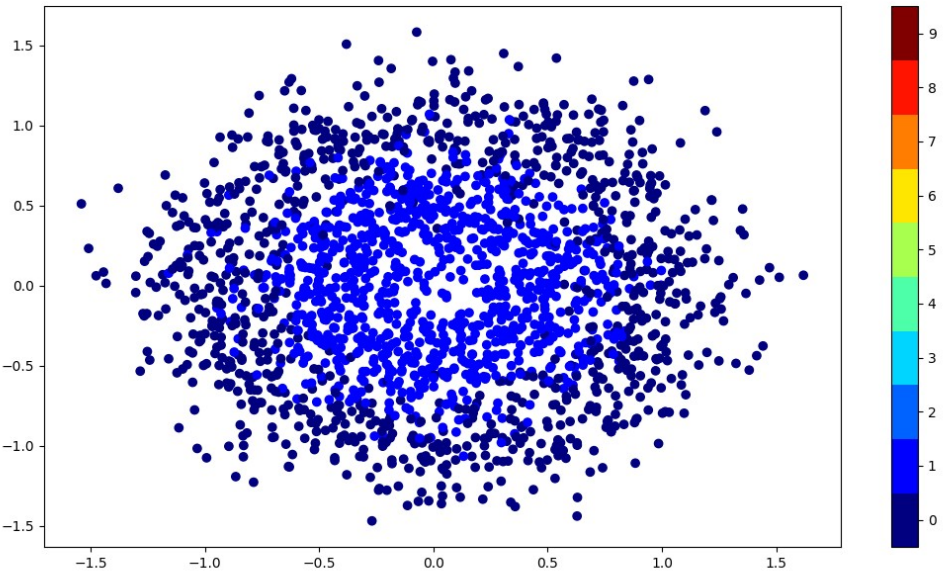

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python explore_1.py --data data_2.h5
{0: 50, 1: 50}
```

Dataset 3



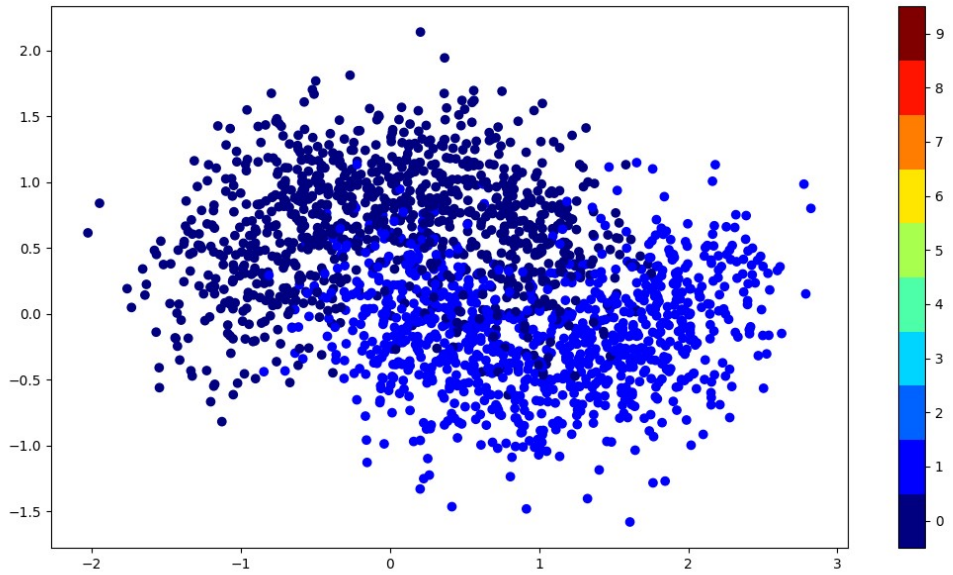

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python explore_1.py --data data_3.h5
{0: 34, 1: 33, 2: 33}
```

Dataset 4



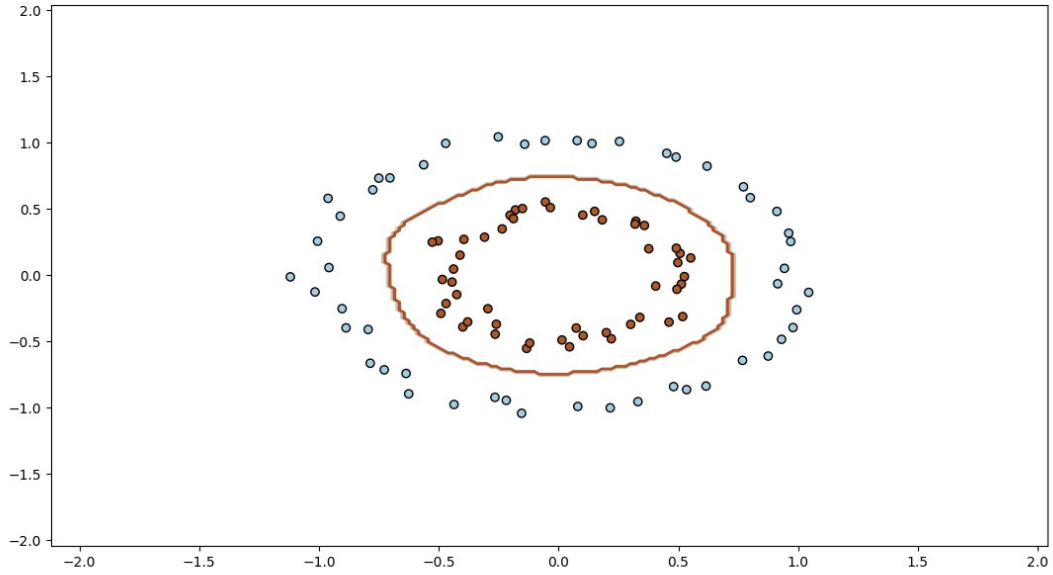

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python explore_1.py --data data_4.h5
{0: 1000, 1: 1000}
```

Dataset 5



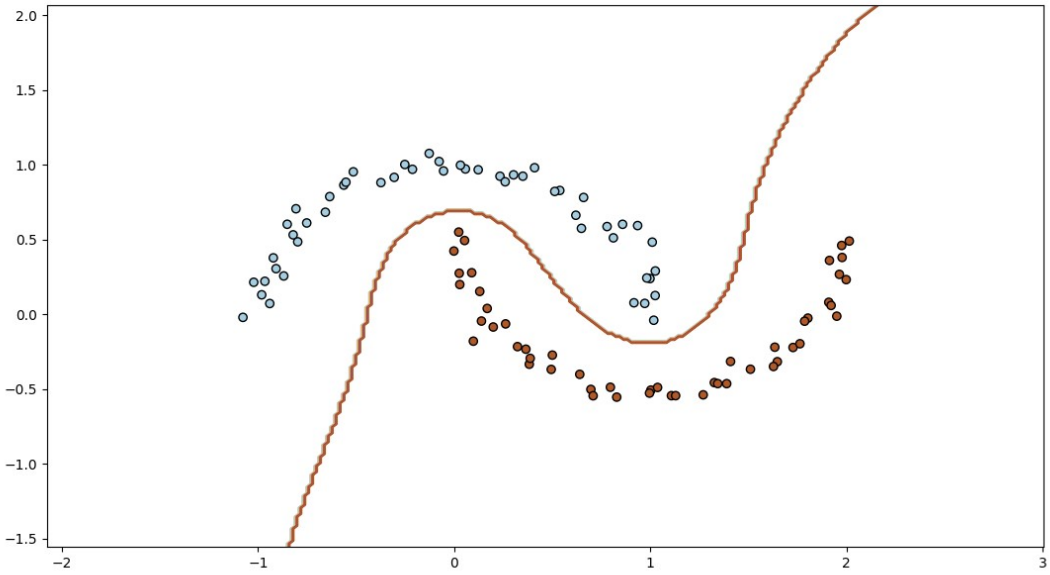
```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python explore_1.py --data data_5.h5
{0: 1000, 1: 1000}
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$
```

Q2:
Dataset 1



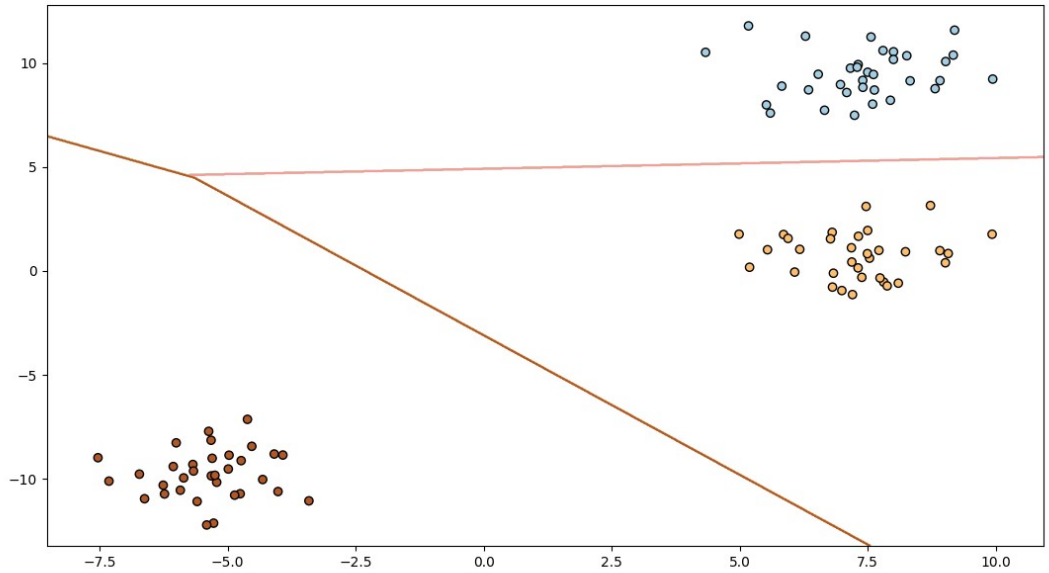
Choice of Kernel: RBF as it is not linearly seperable

Dataset 2



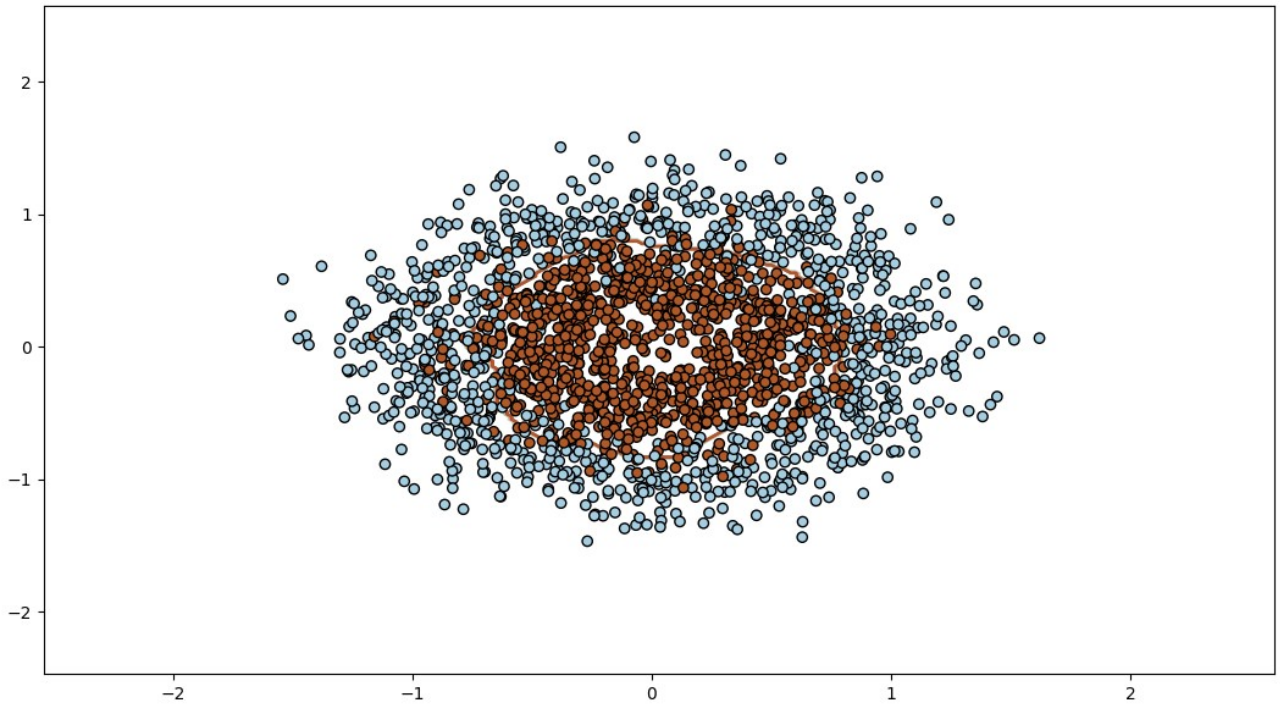
Choice of Kernel: RBF as it is not linearly seperable

Dataset 3



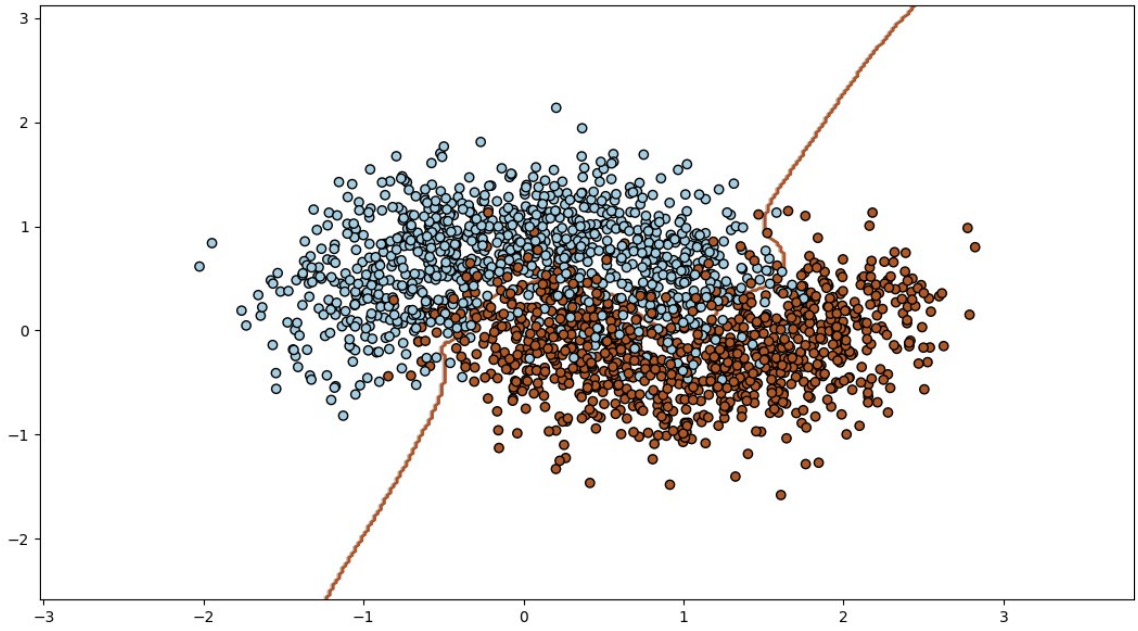
Choice of Kernel: Linear as they can be linearly seperated

Dataset 4



Choice of Kernel: RBF as it is not linearly seperable

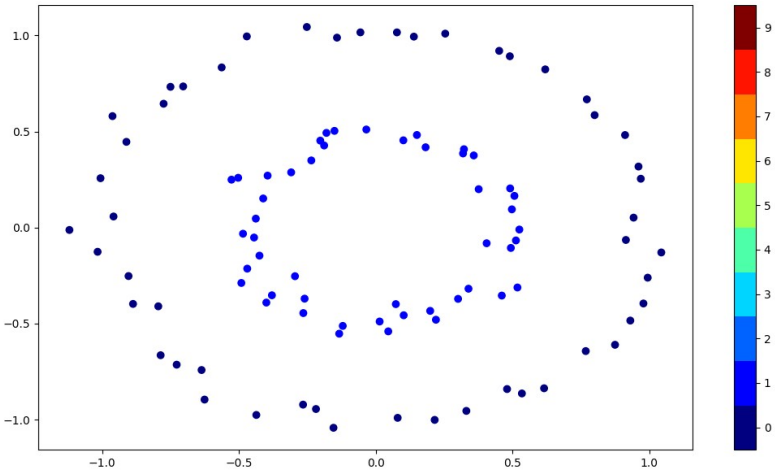
Dataset 5



Choice of Kernel: RBF as it is not linearly seperable

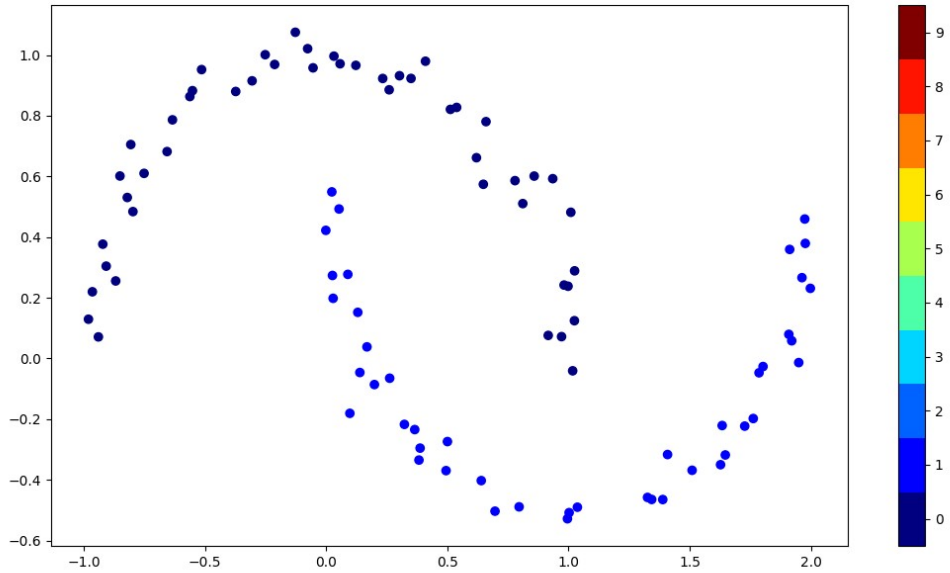
Q3
Used Technique : modified z_score:
modified_z_scores = [0.6745 * (y - median) / median_absolute_deviation_y]
Dataset 1

threshold x1=1 and x2=1



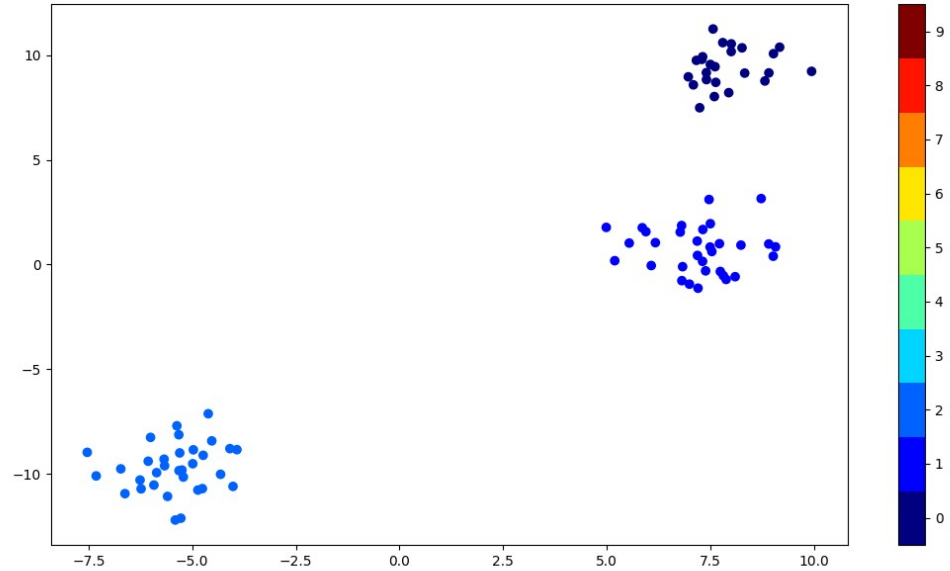
Dataset 2

threshold x1=2.4 and x2=1



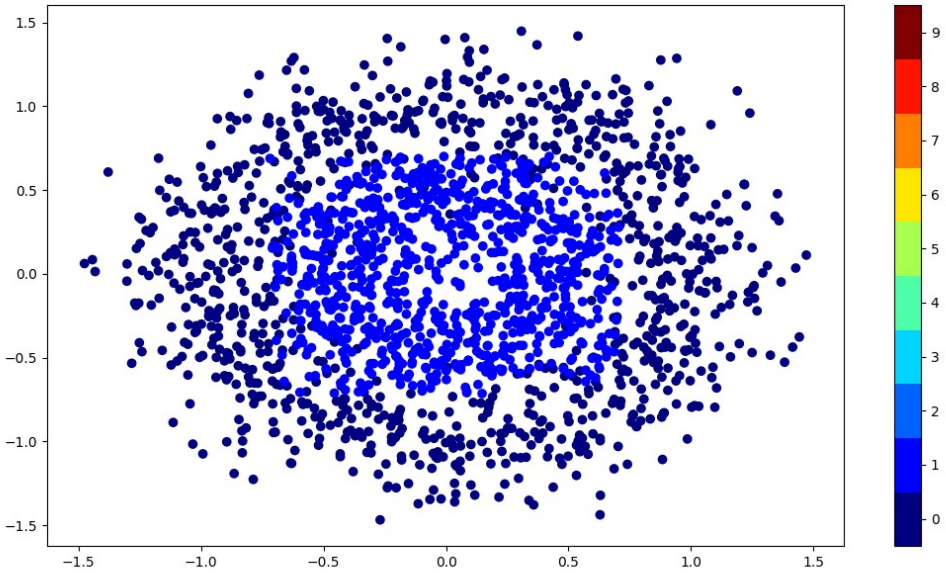
Dataset 3

threshold x1=1.1 and x2=2



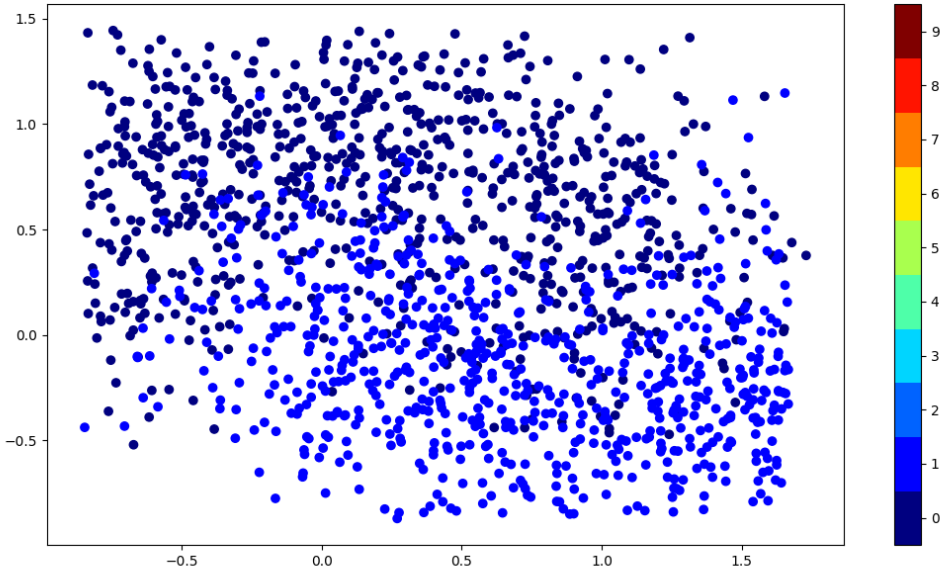
Dataset 4

threshold x1=1.5 and x2=1.5



Dataset 5

threshold x1=1.8 and x2=1.4



SVM

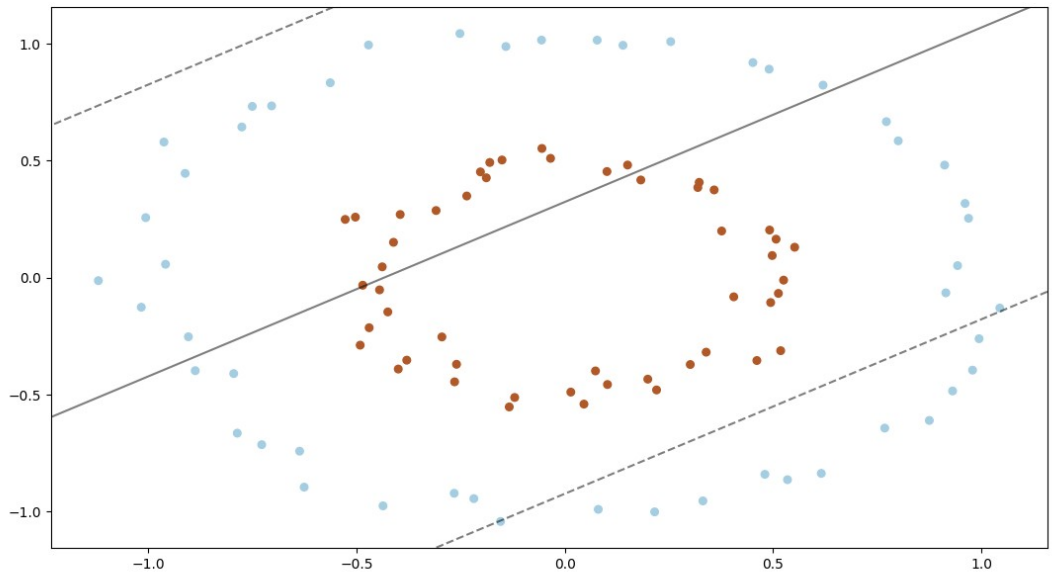
1. Linear kernel

Dataset 1:
One vs One

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1.p
y --classifier "ovo"
file is  /home/garvita/Desktop/Assignment2/data_1.h5
(1, -1, 0.001)
0.44
```

- Choice of hyperparameter C=1,max_iter=default,tol=.001
- Method to choice: grid search
- Linear SVM should not be preffered in this case ,either kernel svm or logistic regression should be used.
- Metrics used to compare is : accuracy which is coming as .44 in case of linear kernel which is very low

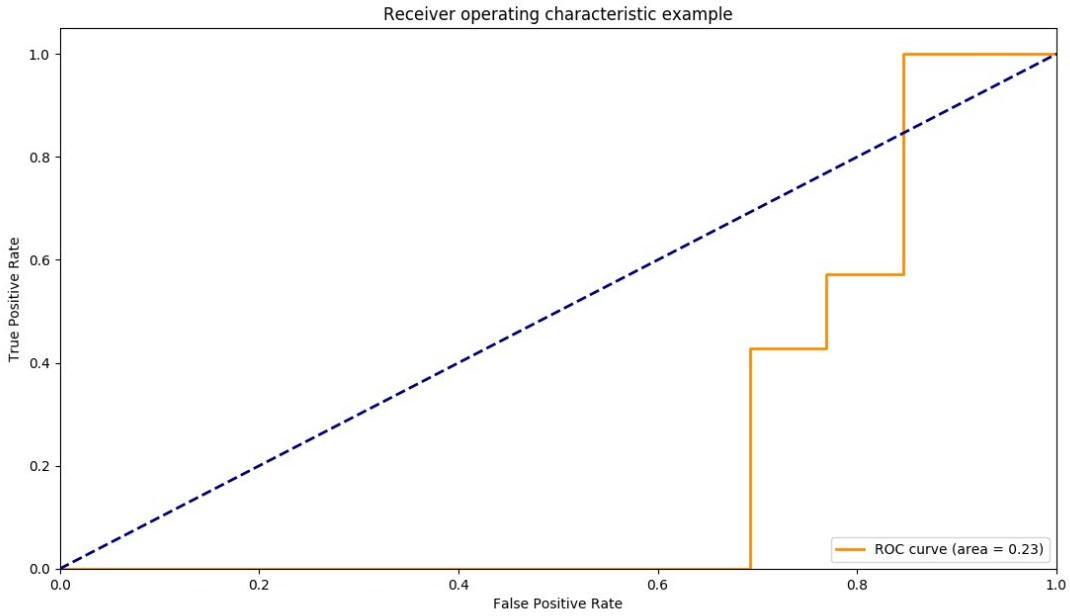
- Margin Seperating hyperplane and support vector



- Confusion Matrix:

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1_c
onfusion_roc.py  --classifier "ovo"
/home/garvita/Desktop/Assignment2/data_1.h5
[[  2.  11.]
 [  3.   4.]]
```

- ROC curve

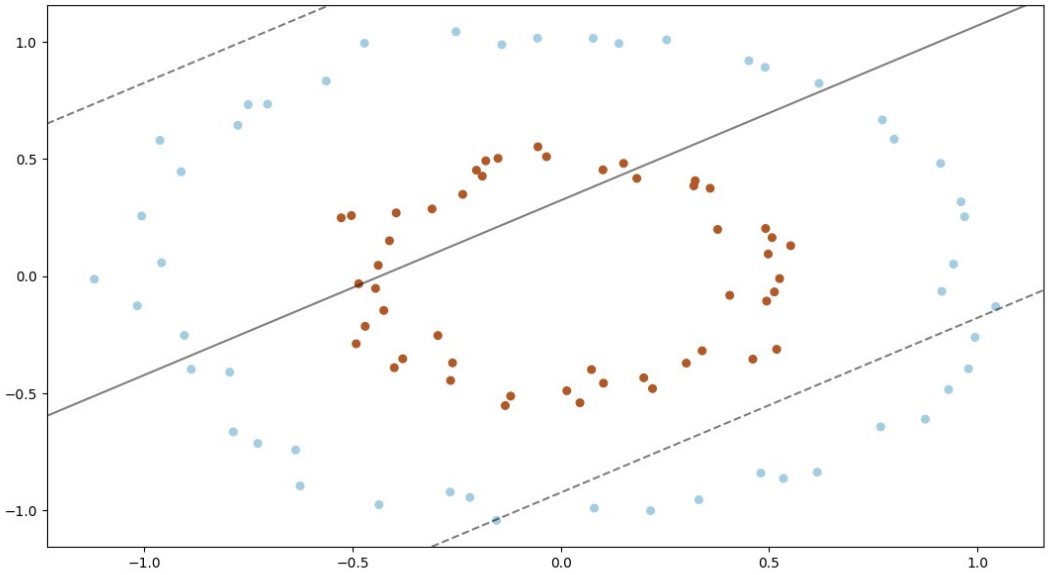


One vs Rest

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1.p
y --classifier "ovr"
file is  /home/garvita/Desktop/Assignment2/data_1.h5
(1, -1, 0.001)
0.44
```

- Choice of hyperparameter C=1,max_iter=default,tol=.001
- Method to choice: grid search
- Linear SVM should not be preffered in this case ,either kernel svm or logistic regression should be used.
- Metrics used to compare is : accuracy which is coming as .44 in case of linear kernel which is very low

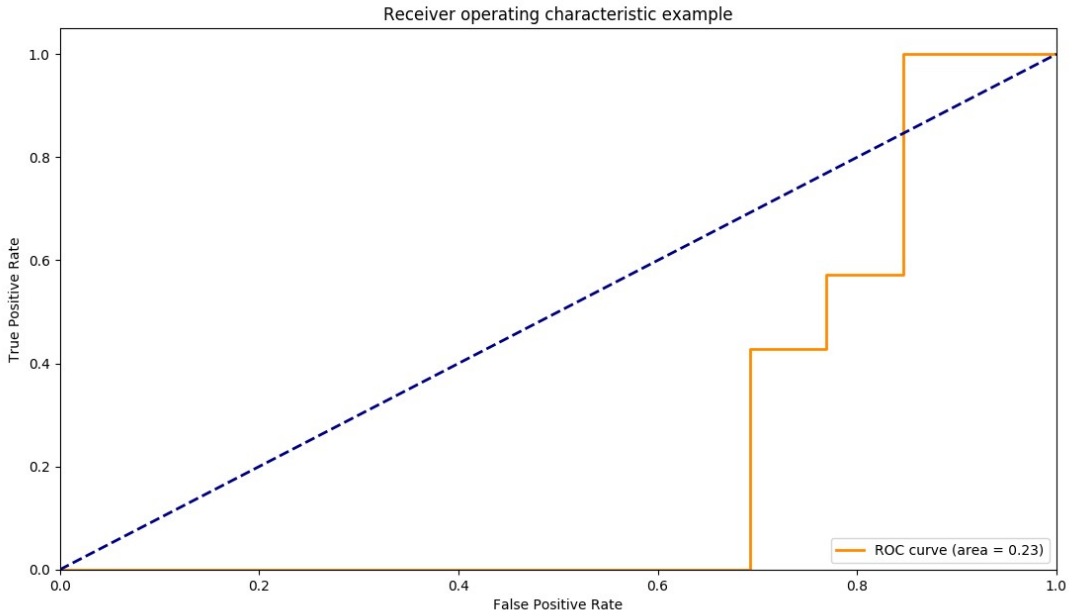
- Margin Seperating hyperplane and support vector



- Confusion Matrix:

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1_c
onfusion_roc.py  --classifier "ovr"
/home/garvita/Desktop/Assignment2/data_1.h5
{'1 1': 4, '0 1': 11, '0 0': 2, '1 0': 3}
[[  2.  11.]
 [  3.   4.]]
```

- ROC curve



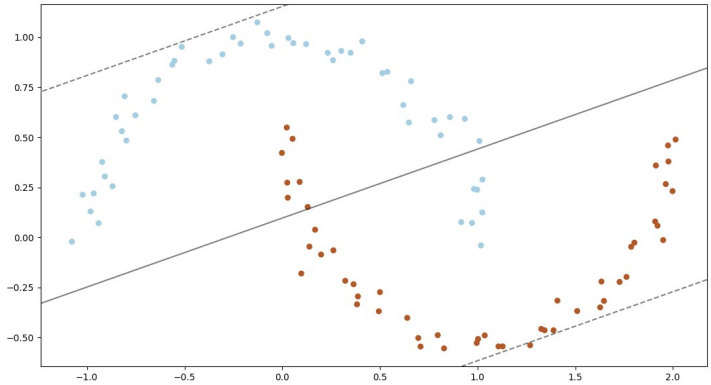
Dataset 2

One vs One

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1.p
y --classifier "ovo"
file is  /home/garvita/Desktop/Assignment2/data_2.h5
(1, -1, 0.001)
0.86
```

- Choice of hyperparameter C=1,max_iter=default,tol=.001
- Method to choice: grid search
- Linear SVM should not be preffered in this case ,either kernel (polynomial or rbf)svm or logistic regression should be used.
- Metrics used to compare is : accuracy score

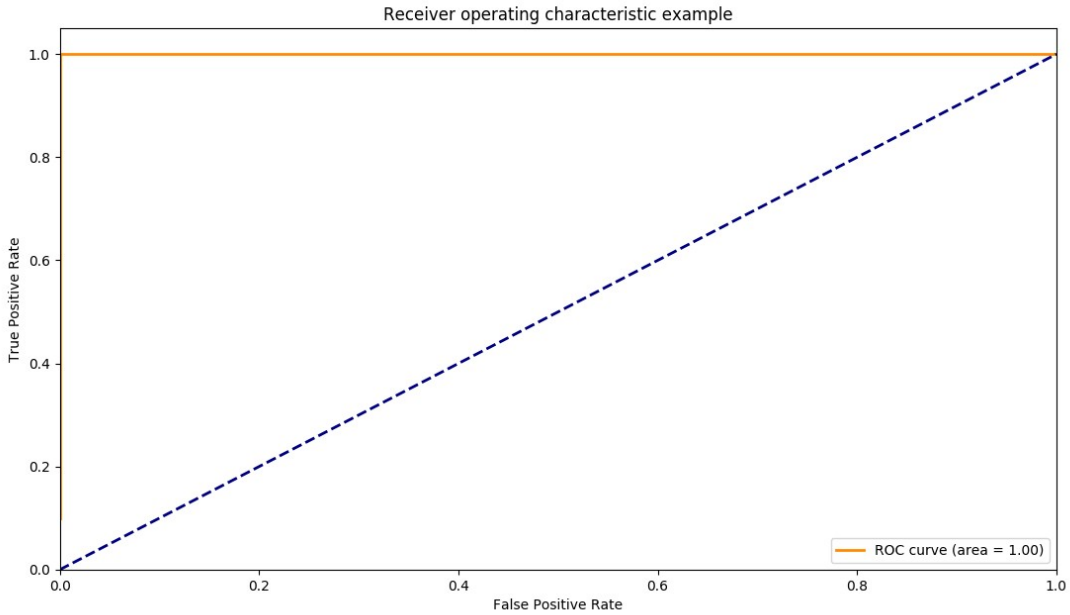
- Margin Seperating hyperplane and support vector



- Confusion Matrix:

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1_c
onfusion_roc.py  --classifier "ovo"
/home/garvita/Desktop/Assignment2/data_2.h5
{'1 1': 7, '0 1': 0, '0 0': 10, '1 0': 3}
[[ 10.   0.]
 [  3.   7.]]
```

- ROC curve

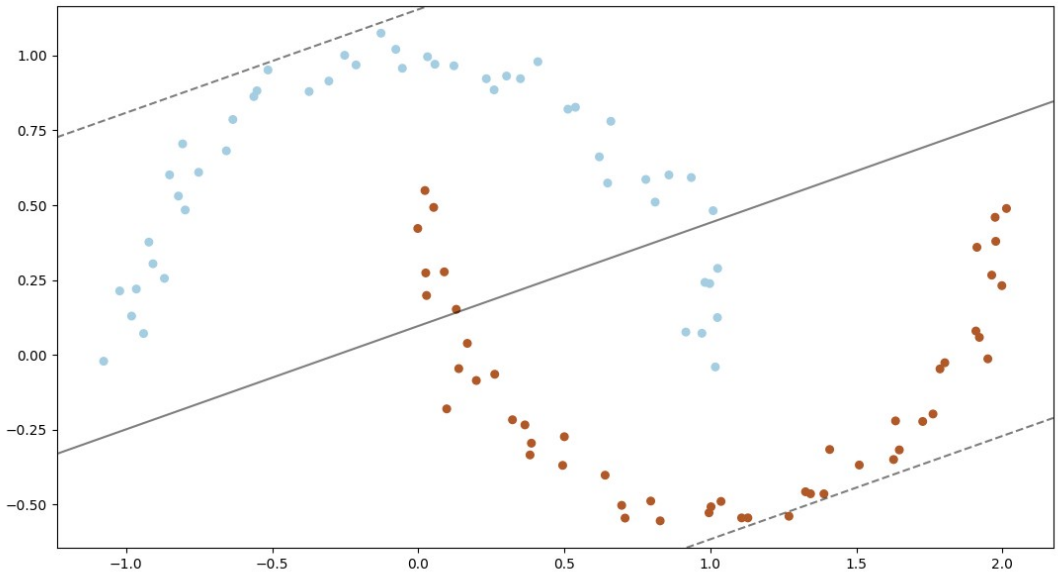


One vs Rest

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1.p
y --classifier "ovr"
file is  /home/garvita/Desktop/Assignment2/data_2.h5
(1, -1, 0.001)
0.86
```

- Choice of hyperparameter C=1,max_iter=default,tol=.001
- Method to choice: grid search
- Linear SVM should not be preffered in this case ,either kernel (polynomial or rbf)svm or logistic regression should be used.
- Metrics used to compare is : accuracy score

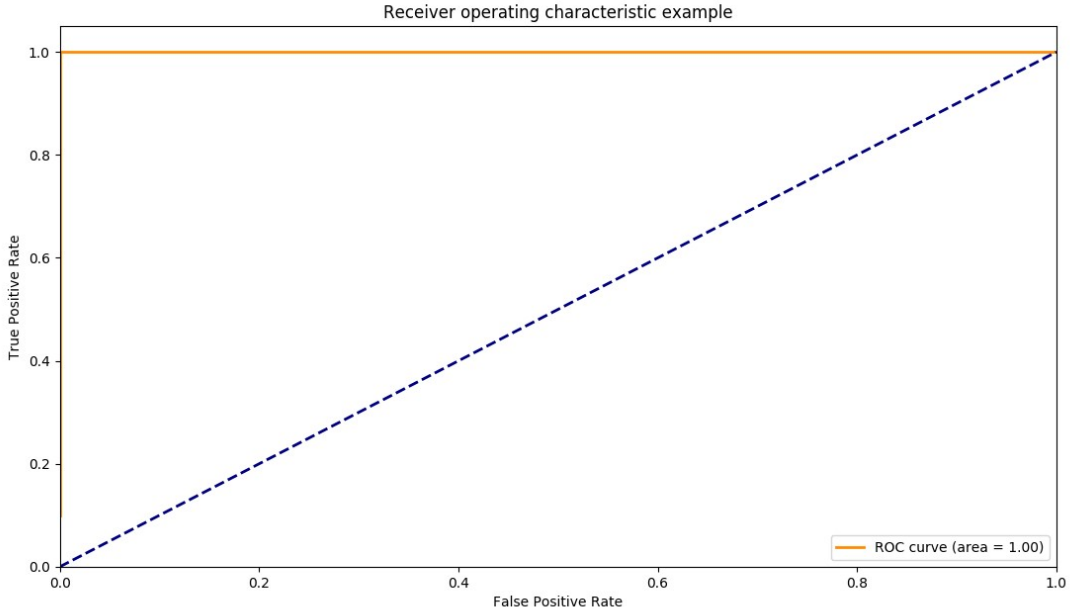
- Margin Seperating hyperplane and support vector



- Confusion Matrix:

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1_
onfusion_roc.py  --classifier "ovr"
/home/garvita/Desktop/Assignment2/data_2.h5
{'1 1': 7, '0 1': 0, '0 0': 10, '1 0': 3}
[[ 10.   0.]
 [  3.   7.]]
```

- ROC curve



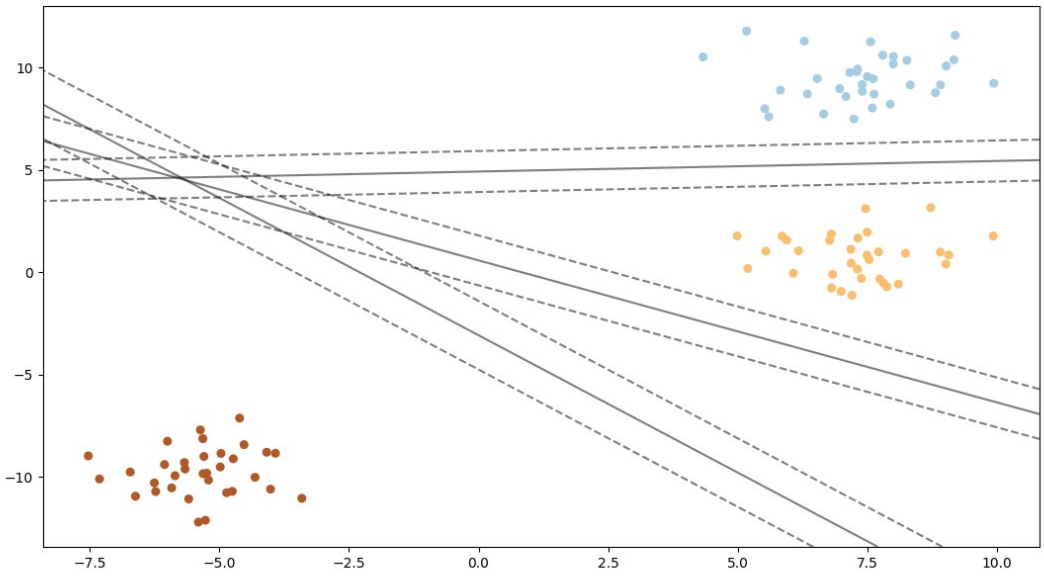
Dataset 3

One vs One

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1.p
y --classifier "ovo"
file is  /home/garvita/Desktop/Assignment2/data_3.h5
(1, -1, 0.001)
1.0
```

- Choice of hyperparameter C=1,max_iter=default,tol=.001
- Method to choice: grid search
- Linear SVM should be preffered in this case
- Metrics used to compare is : accuracy score

- Margin Seperating hyperplane and support vector



- Confusion Matrix:

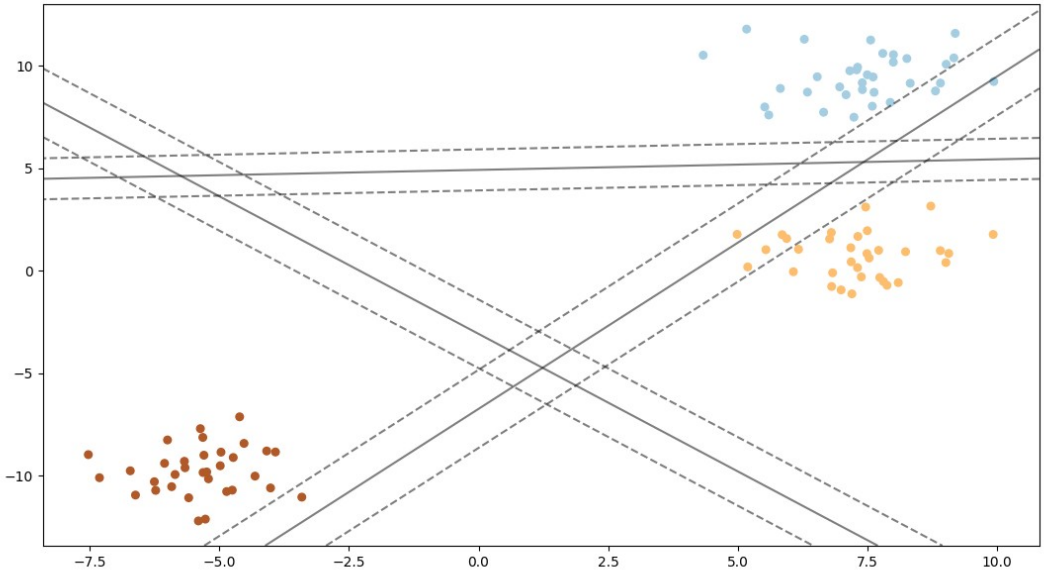
```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1_c
onfusion_roc.py  --classifier "ovo"
/home/garvita/Desktop/Assignment2/data_3.h5
{'0 2': 0, '0 1': 0, '0 0': 4, '1 2': 0, '1 0': 0, '1 1': 8, '2 1': 0, '2 0': 0,
 '2 2': 8}
[[ 4.  0.  0.]
 [ 0.  8.  0.]
 [ 0.  0.  8.]]
```

One vs Rest

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1.p
y --classifier "ovr"
file is  /home/garvita/Desktop/Assignment2/data_3.h5
(1, -1, 0.001)
0.98
```

- Choice of hyperparameter C=1,max_iter=default,tol=.001
- Method to choice: grid search
- Linear SVM should be preffered in this case
- Metrics used to compare is : accuracy score

- Margin Seperating hyperplane and support vector



- Confusion Matrix:

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1_c
onfusion_roc.py  --classifier "ovr"
/home/garvita/Desktop/Assignment2/data_3.h5
{'0 2': 0, '0 1': 0, '0 0': 4, '1 2': 0, '1 0': 0, '1 1': 8, '2 1': 0, '2 0': 0,
 '2 2': 8}
[[ 4.  0.  0.]
 [ 0.  8.  0.]
 [ 0.  0.  8.]]
```

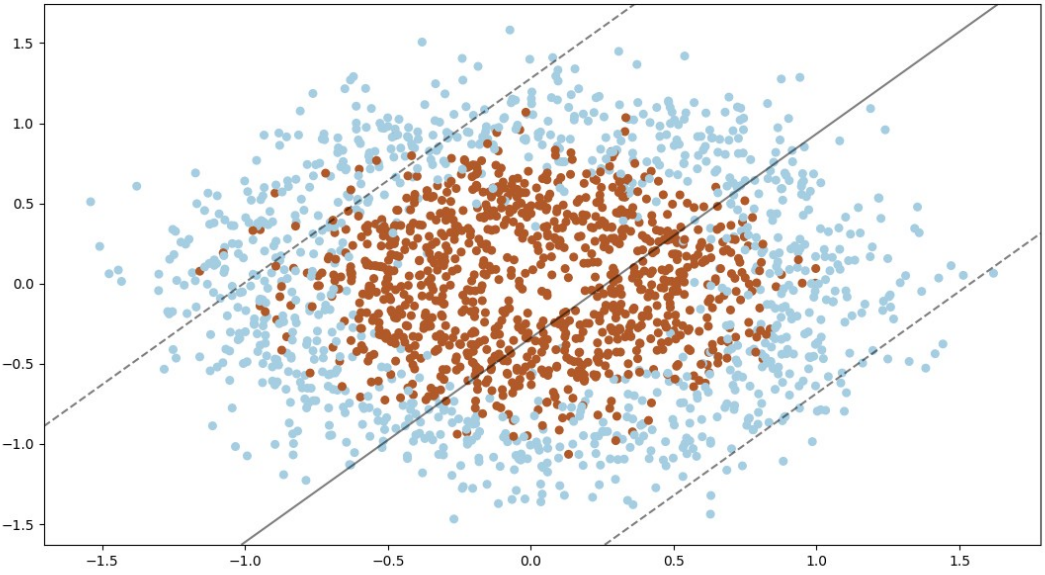
Dataset 4:

One vs One

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1.
y --classifier "ovo"
file is  /home/garvita/Desktop/Assignment2/data_4.h5
(1, -1, 0.001)
0.5445
```

- Choice of hyperparameter C=1,max_iter=default,tol=.001
- Method to choice: grid search
- Linear SVM should not be preffered in this case ,either kernel (rbf) svm or logistic regression should be used.
- Metrics used to compare is : accuracy score

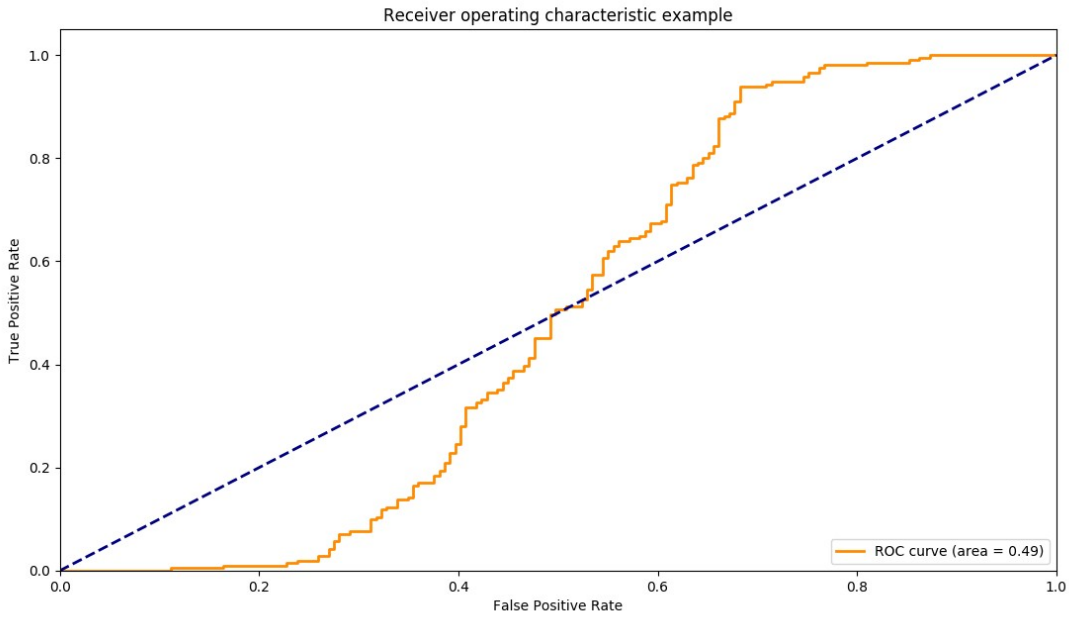
- Margin Seperating hyperplane and support vector



- Confusion Matrix

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1_c
onfusion_roc.py  --classifier "ovo"
/home/garvita/Desktop/Assignment2/data_4.h5
{'1 1': 0, '0 1': 0, '0 0': 189, '1 0': 211}
[[ 189.    0.]
 [ 211.    0.]]
```

- ROC curve

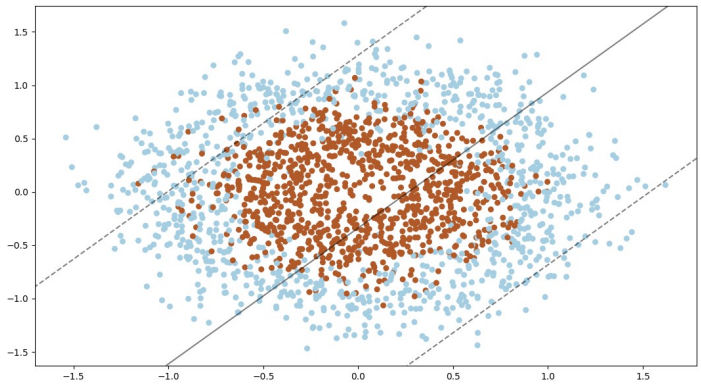


One vs Rest

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1.p
y --classifier "ovr"
file is  /home/garvita/Desktop/Assignment2/data_4.h5
(1, -1, 0.001)
0.5445
```

- Choice of hyperparameter C=1,max_iter=default,tol=.001
- Method to choice: grid search
- Linear SVM should not be prefered in this case ,either kernel (rbf) svm or logistic regression should be used.
- Metrics used to compare is : accuracy score

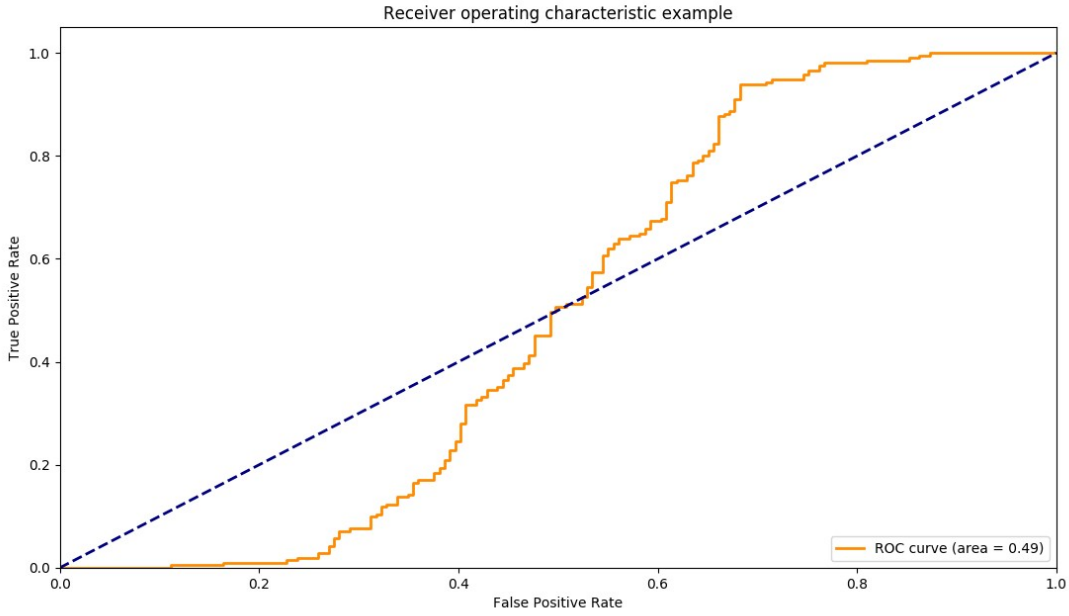
- Margin Seperating hyperplane and support vector



- Confusion Matrix

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1_c
onfusion_roc.py  --classifier "ovr"
/home/garvita/Desktop/Assignment2/data_4.h5
{'1 1': 0, '0 1': 0, '0 0': 189, '1 0': 211}
[[ 189.    0.]
 [ 211.    0.]]
```

- ROC curve



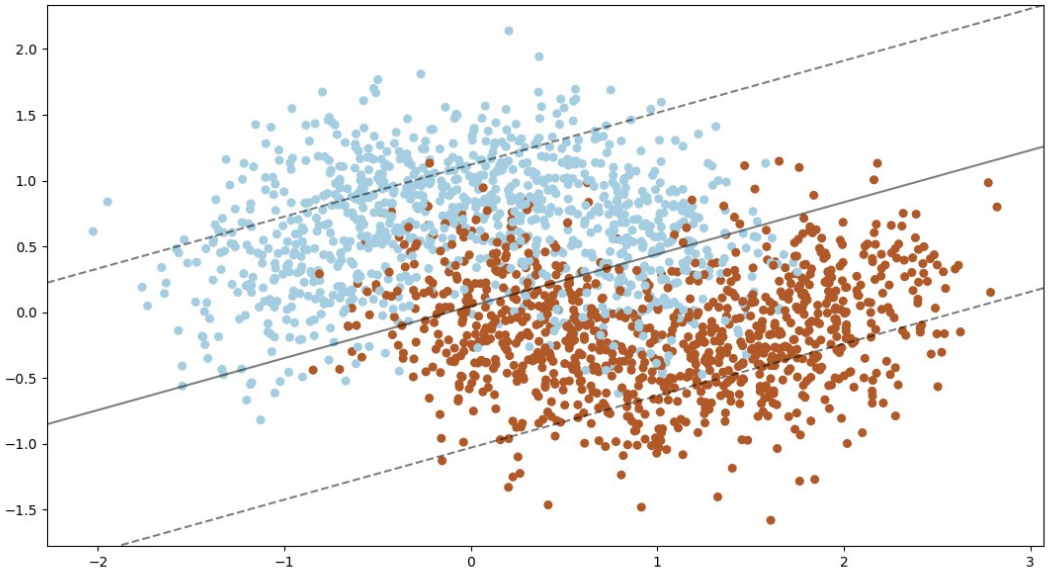
Dataset 5

One vs One

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1.p
y --classifier "ovo"
file is  /home/garvita/Desktop/Assignment2/data_5.h5
(1, -1, 0.001)
0.846
```

- Choice of hyperparameter C=1,max_iter=default,tol=.001
- Method to choice: grid search
- Linear SVM can be prefered in this case(but not a good choice) ,either kernel (polynomial) svm or logistic regression should be used.
- Metrics used to compare is : accuracy score

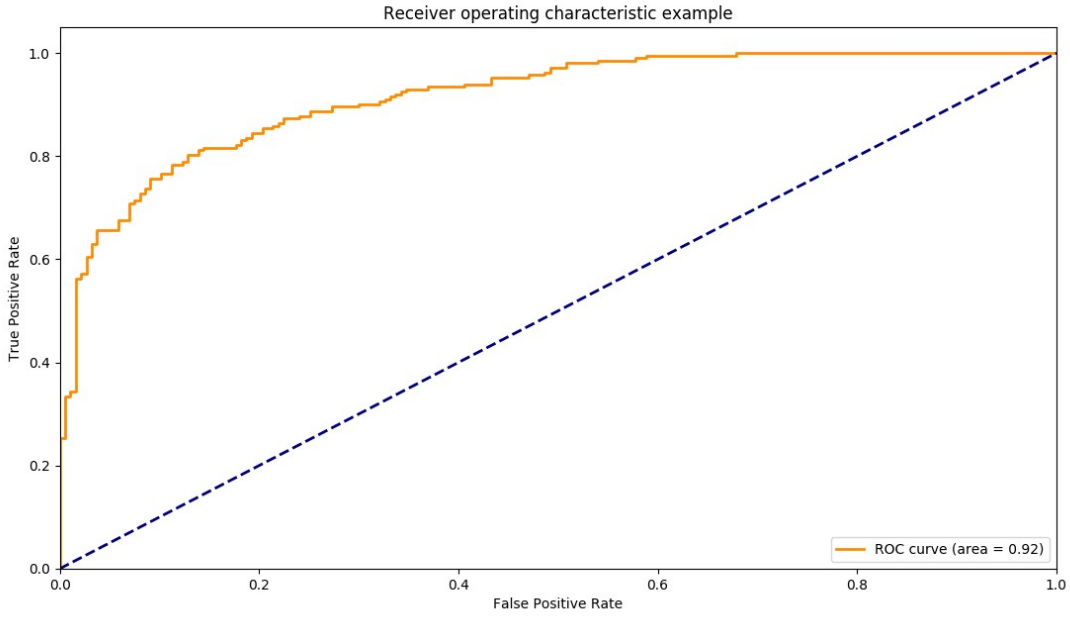
- Margin Seperating hyperplane and support vector



- Confusion metrics

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1_c
onfusion_roc.py  --classifier  "ovo"
/home/garvita/Desktop/Assignment2/data_5.h5
{'1 1': 182, '0 1': 38, '0 0': 149, '1 0': 31}
[[ 149.   38.]
 [  31.  182.]]
```

- ROC Curve

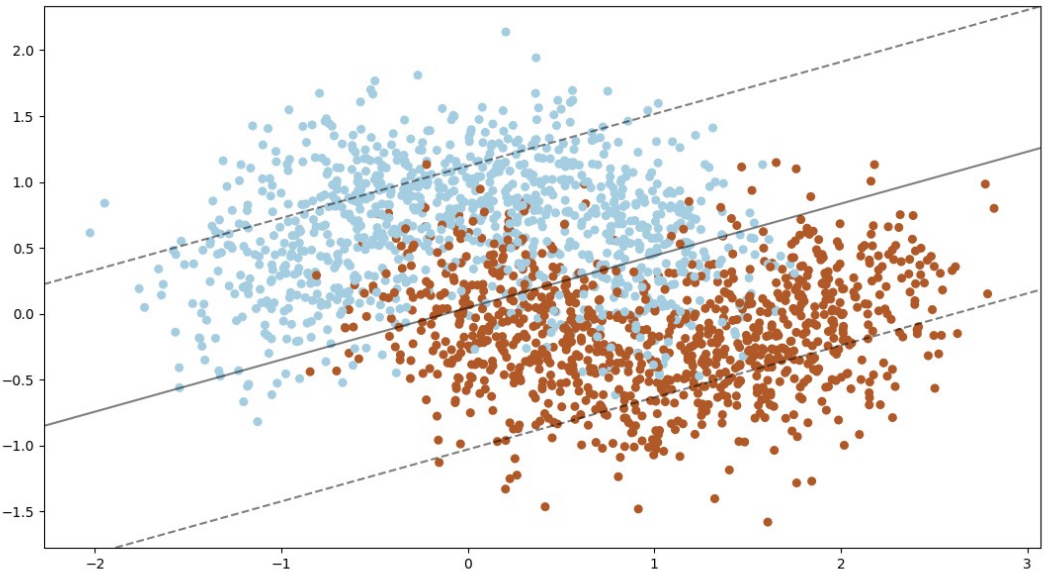


One vs Rest

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1.
y --classifier "ovr"
file is  /home/garvita/Desktop/Assignment2/data_5.h5
(1, -1, 0.001)
0.846
```

- Choice of hyperparameter C=1,max_iter=default,tol=.001
- Method to choice: grid search
- Linear SVM can be prefered in this case(but not a good choice) ,either kernel (polynomial) svm or logistic regression should be used.
- Metrics used to compare is : accuracy score

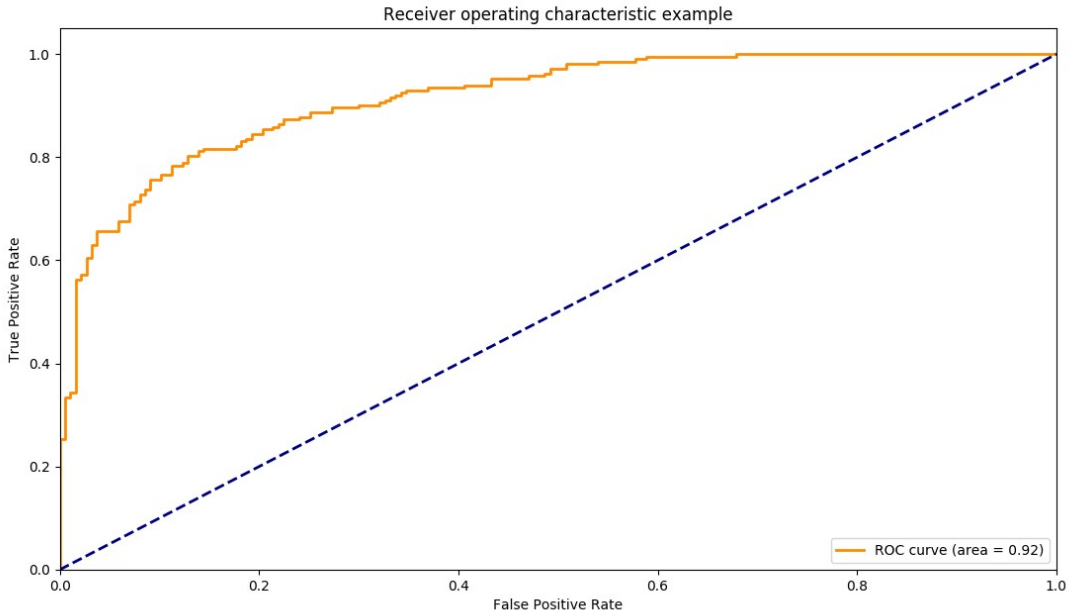
- Margin Seperating hyperplane and support vector



- Confusion Metrics

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1_c
onfusion_roc.py  --classifier  "ovo"
/home/garvita/Desktop/Assignment2/data_5.h5
{'1 1': 182, '0 1': 38, '0 0': 149, '1 0': 31}
[[ 149.   38.]
 [  31.  182.]]
```

- ROC curve

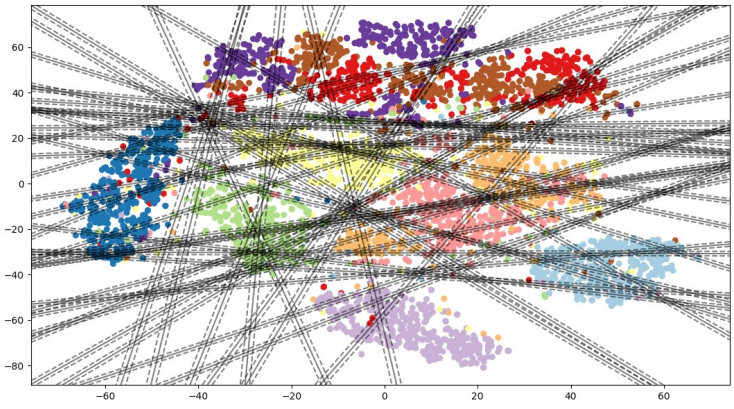


Previous datasets:
Part A

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1.p
y --classifier "ovo"
file is  /home/garvita/Desktop/ml/hw/Template/Data/part_A_train.h5
(1, -1, 0.001)
0.755238095238
```

- Choice of hyperparameter C=1,max_iter=default,tol=.001
- Method to choice: grid search
- Logistic Regression should be used over linear svm as in LR accuracy obtained was .877(in previous assignment).
- Metrics used to compare is : accuracy score

- Margin Seperating hyperplane and support vector

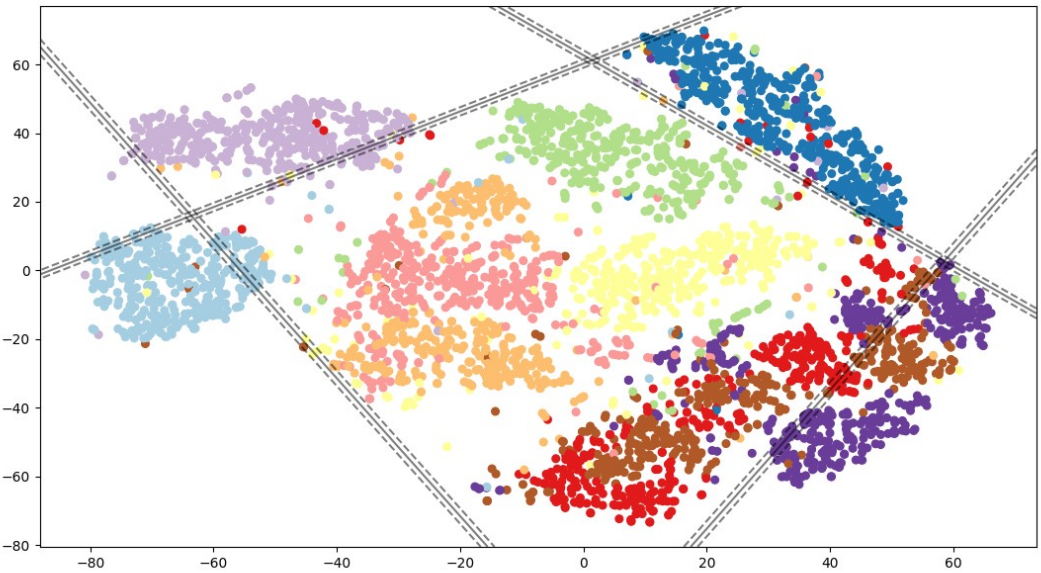


- Confusion Metrics

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1_confusion_roc.py  --classifier  "ovo"
/home/garvita/Desktop/ml/hw/Template/Data/part_A_train.h5
[[ 72.   0.   1.   1.   0.   2.   0.   0.   2.   0.]
 [  0.  85.   0.   0.   0.   0.   0.   0.   2.   0.]
 [  1.   2.  79.   3.   3.   1.   1.   0.   1.   0.]
 [  1.   1.   2.  65.   0.   2.   0.   2.   1.   1.]
 [  1.   1.   0.   0.  79.   0.   1.   1.   0.   4.]
 [  0.   0.   1.   3.   2.  60.   1.   1.   1.   0.]
 [  3.   1.   1.   0.   0.   2.  95.   0.   0.   0.]
 [  0.   1.   1.   3.   1.   0.   0.  89.   0.   0.]
 [  2.   1.   3.   1.   0.   4.   1.   0.  62.   2.]
 [  0.   0.   0.   0.   4.   0.   0.   4.   2.  70.]]
```

One vs rest

- Margin Seperating hyperplane and support vector

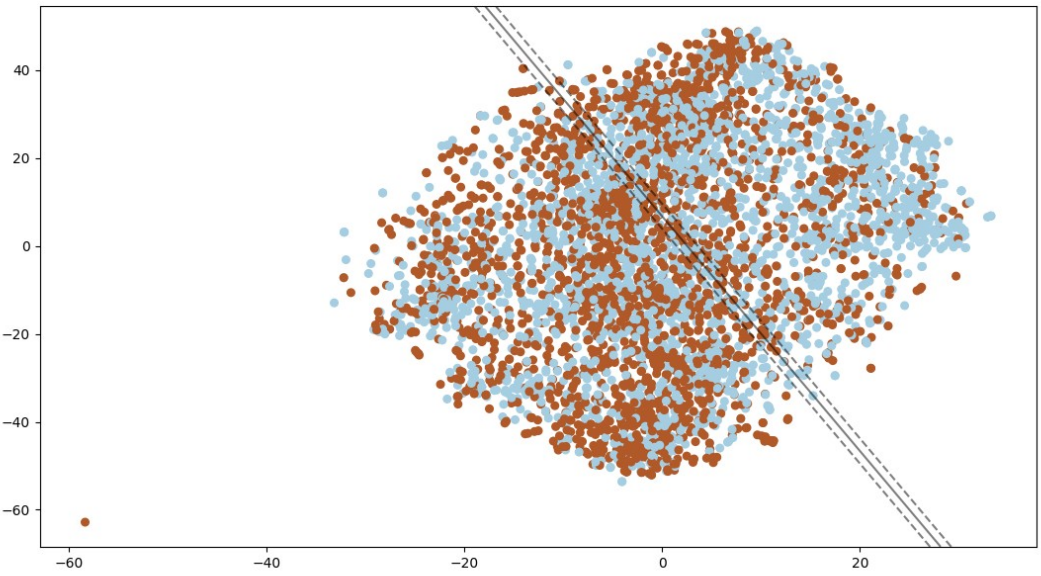


- Confusion Matrix

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1_confusion_roc.py  --classifier  "ovr"
/home/garvita/Desktop/ml/hw/Template/Data/part_A_train.h5
[[ 72.   0.   1.   1.   0.   2.   0.   0.   2.   0.]
 [  0.  85.   0.   0.   0.   0.   0.   0.   2.   0.]
 [  1.   2.  79.   3.   3.   1.   1.   0.   1.   0.]
 [  1.   1.   2.  65.   0.   2.   0.   2.   1.   1.]
 [  1.   1.   0.   0.  79.   0.   1.   1.   0.   4.]
 [  0.   0.   1.   3.   2.  60.   1.   1.   1.   0.]
 [  3.   1.   1.   0.   0.   2.  95.   0.   0.   0.]
 [  0.   1.   1.   3.   1.   0.   0.  89.   0.   0.]
 [  2.   1.   3.   1.   0.   4.   1.   0.  62.   2.]
 [  0.   0.   0.   0.   4.   0.   0.   4.   2.  70.]]
```

Part B
One vs One

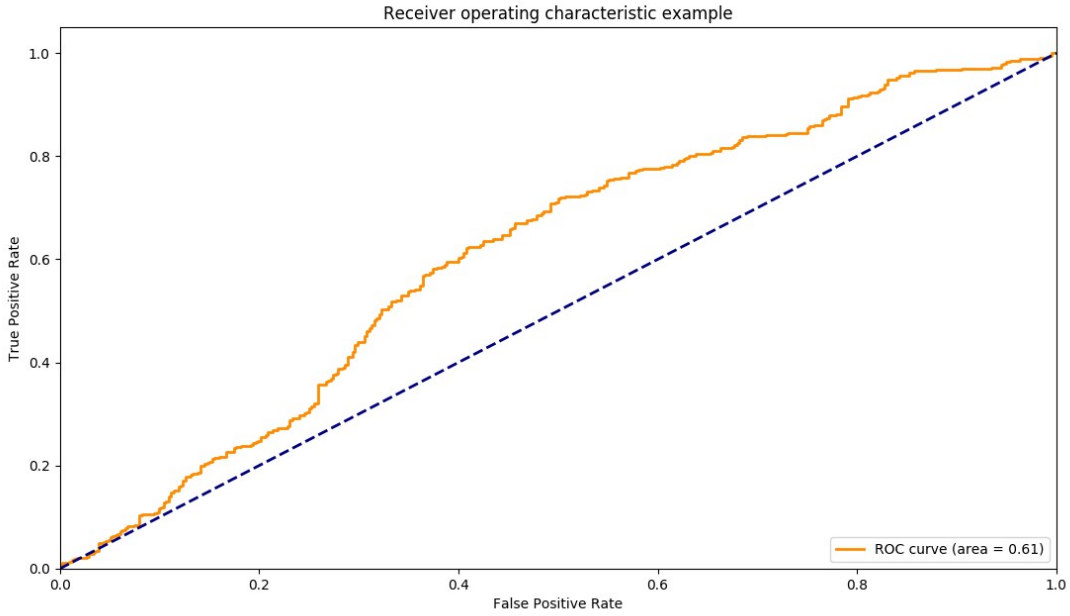
- Margin Seperating hyperplane and support vector



- Confusion matrix

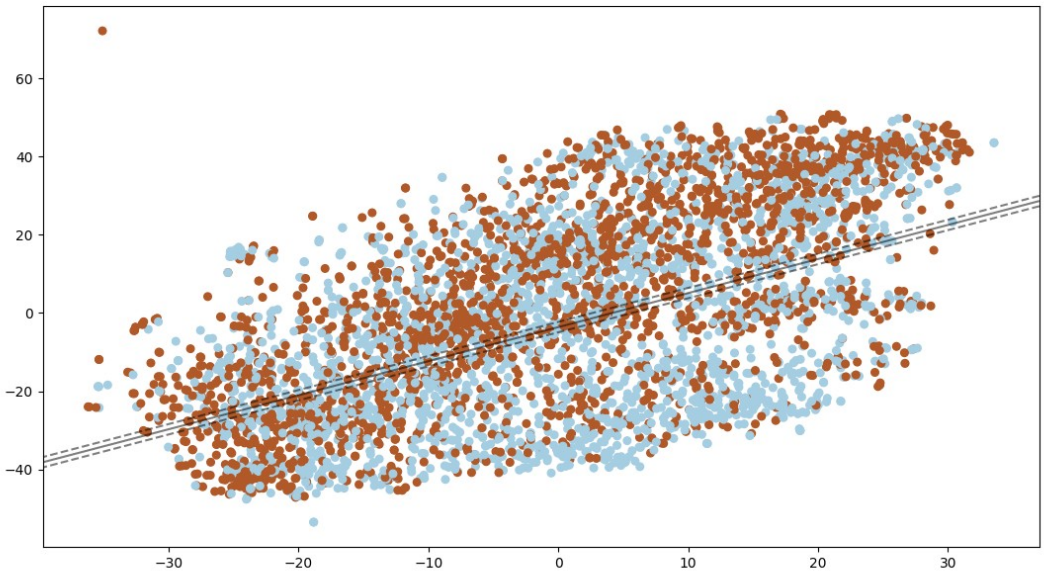
```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1_confusion_roc.py  --classifier  "ovo"
/home/garvita/Desktop/ml/hw/Template/Data/part_B_train.h5
[[ 224.  188.]
 [ 198.  230.]]
```

- ROC curve

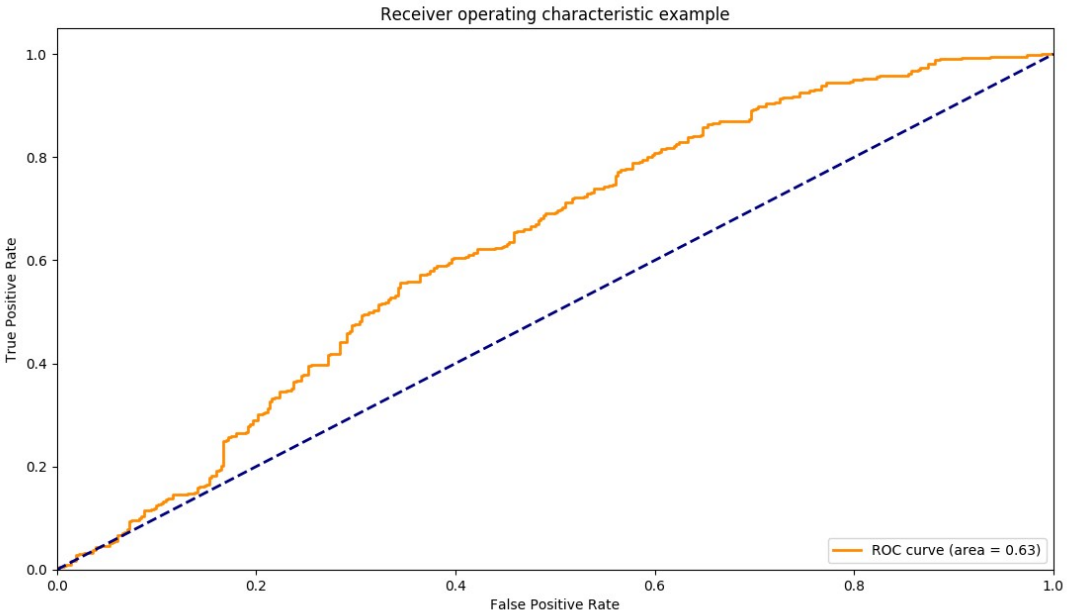


One vs Rest

- Margin Seperating hyperplane and support vector



- Confusion Matrix:
- ROC curve:



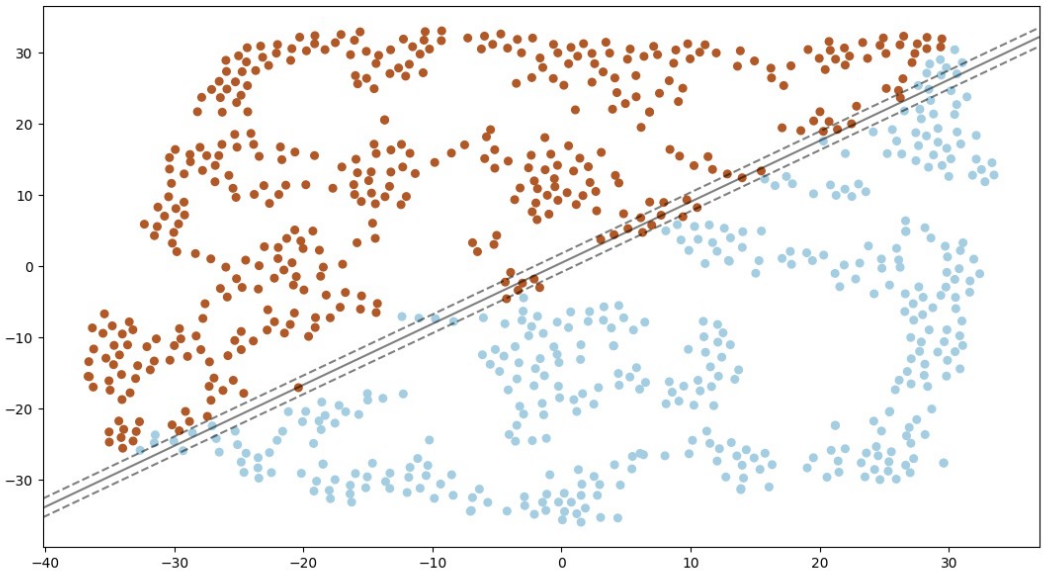
Part C

One vs One

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1.p
y --classifier "ovo"
file is  /home/garvita/Desktop/ml/hw/Template/Data/part_C_train.h5
(1, -1, 0.001)
0.99875
```

- Choice of hyperparameter C=1,max_iter=default,tol=.001
- Method to choice: grid search
- Logistic Regression should be used as data is linearly seperable
- Metrics used to compare is : accuracy score

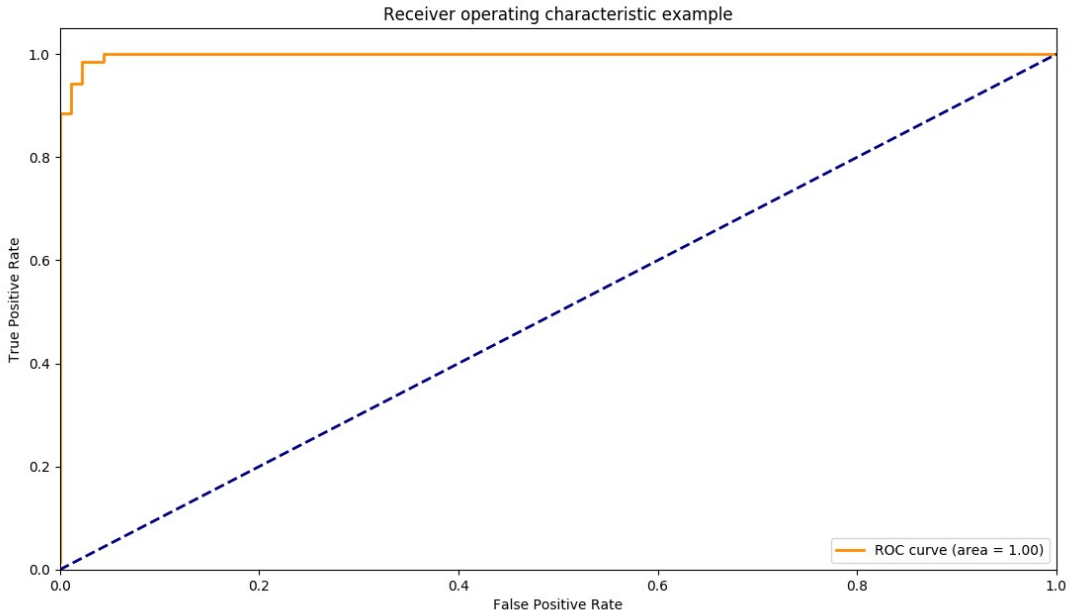
- Margin Seperating hyperplane and support vector



- Confusion Matrix:

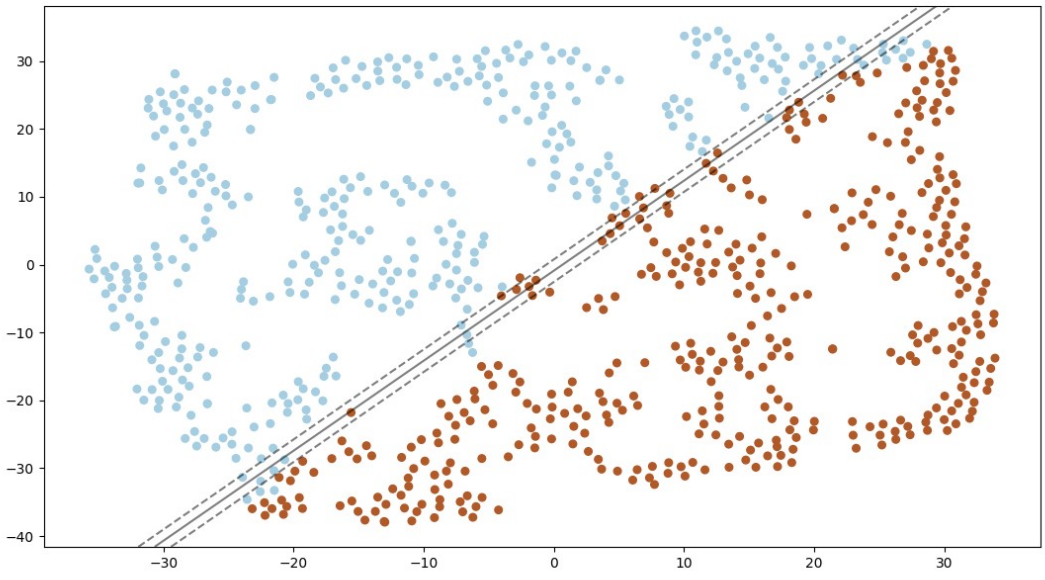
```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1_confusion_roc.py  --classifier  "ovo"
/home/garvita/Desktop/ml/hw/Template/Data/part_C_train.h5
[[ 90.   0.]
 [  0.  70.]]
```

- ROC curve



One vs Rest

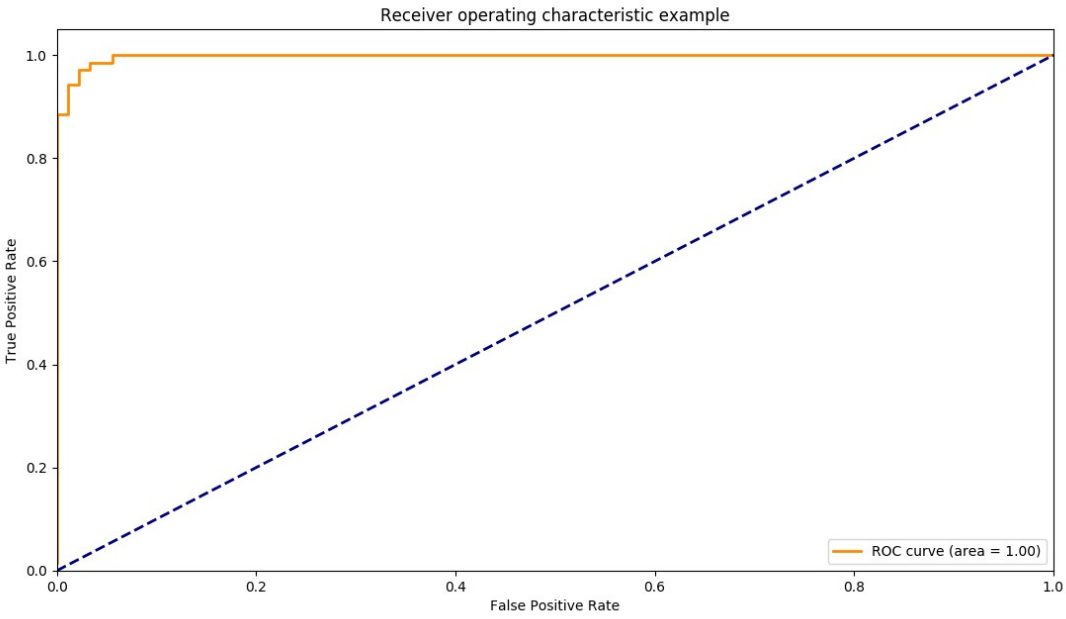
- Margin Seperating hyperplane and support vector



- Confusion Matrix:

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_1_confusion_roc.py  --classifier  "ovr"
/home/garvita/Desktop/ml/hw/Template/Data/part_C_train.h5
[[ 90.   0.]
 [  0.  70.]]
```

- ROC curve



2. RBF Kernel

Dataset 1

One vs One

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2.p
y --classifier "ovo"
/home/garvita/.local/lib/python2.7/site-packages/sklearn/svm/base.py:218: Conver
genceWarning: Solver terminated early (max_iter=10).  Consider pre-processing yo
ur data with StandardScaler or MinMaxScaler.
  % self.max_iter, ConvergenceWarning)
/home/garvita/.local/lib/python2.7/site-packages/sklearn/svm/base.py:218: Conver
genceWarning: Solver terminated early (max_iter=100).  Consider pre-processing y
our data with StandardScaler or MinMaxScaler.
  % self.max_iter, ConvergenceWarning)
(1, -1, 0.001)
0.5
```

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2_c
onfusion.py --classifier "ovo"
/home/garvita/Desktop/Assignment2/data_1.h5
[[ 13.   0.]
 [  7.   0.]]
```

One vs Rest

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2.p
y --classifier "ovr"
/home/garvita/.local/lib/python2.7/site-packages/sklearn/svm/base.py:218: Conver
genceWarning: Solver terminated early (max_iter=10).  Consider pre-processing yo
ur data with StandardScaler or MinMaxScaler.
  % self.max_iter, ConvergenceWarning)
/home/garvita/.local/lib/python2.7/site-packages/sklearn/svm/base.py:218: Conver
genceWarning: Solver terminated early (max_iter=100).  Consider pre-processing y
our data with StandardScaler or MinMaxScaler.
  % self.max_iter, ConvergenceWarning)
(0.0001, 10, 0.001)
0.5
```

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2_c
onfusion.py --classifier "ovr"
/home/garvita/Desktop/Assignment2/data_1.h5
[[ 13.   0.]
 [  7.   0.]]
```

Dataset 2

One vs One

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2.p
y --classifier "ovo"
/home/garvita/.local/lib/python2.7/site-packages/sklearn/svm/base.py:218: Conver
genceWarning: Solver terminated early (max_iter=10).  Consider pre-processing yo
ur data with StandardScaler or MinMaxScaler.
  % self.max_iter, ConvergenceWarning)
(0.0001, 10, 0.001)
0.83
```

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2_c
onfusion.py --classifier "ovo"
/home/garvita/Desktop/Assignment2/data_2.h5
[[ 9.  1.]
 [ 2.  8.]]
```

One vs Rest

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2.p
y --classifier "ovr"
/home/garvita/.local/lib/python2.7/site-packages/sklearn/svm/base.py:218: Conver
genceWarning: Solver terminated early (max_iter=10).  Consider pre-processing yo
ur data with StandardScaler or MinMaxScaler.
  % self.max_iter, ConvergenceWarning)
/home/garvita/.local/lib/python2.7/site-packages/sklearn/svm/base.py:218: Conver
genceWarning: Solver terminated early (max_iter=100).  Consider pre-processing y
our data with StandardScaler or MinMaxScaler.
  % self.max_iter, ConvergenceWarning)
(0.0001, 10, 0.001)
0.83
```

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2_c
onfusion.py --classifier "ovr"
/home/garvita/Desktop/Assignment2/data_2.h5
[[ 9.  1.]
 [ 2.  8.]]
```

Dataset 3
One vs One

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2.p
y --classifier "ovo"
/home/garvita/.local/lib/python2.7/site-packages/sklearn/svm/base.py:218: Conver
genceWarning: Solver terminated early (max_iter=10).  Consider pre-processing yo
ur data with StandardScaler or MinMaxScaler.
  % self.max_iter, ConvergenceWarning)
/home/garvita/.local/lib/python2.7/site-packages/sklearn/svm/base.py:218: Conver
genceWarning: Solver terminated early (max_iter=100).  Consider pre-processing y
our data with StandardScaler or MinMaxScaler.
  % self.max_iter, ConvergenceWarning)
(0.0001, 10, 0.001)
0.79
```

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2_c
onfusion.py --classifier "ovo"
/home/garvita/Desktop/Assignment2/data_3.h5
[[ 4.  0.  0.]
 [ 6.  2.  0.]
 [ 0.  0.  8.]]
```

One vs Rest

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2.p
y --classifier "ovr"
/home/garvita/.local/lib/python2.7/site-packages/sklearn/svm/base.py:218: Conver
genceWarning: Solver terminated early (max_iter=10).  Consider pre-processing yo
ur data with StandardScaler or MinMaxScaler.
  % self.max_iter, ConvergenceWarning)
/home/garvita/.local/lib/python2.7/site-packages/sklearn/svm/base.py:218: Conver
genceWarning: Solver terminated early (max_iter=100).  Consider pre-processing y
our data with StandardScaler or MinMaxScaler.
  % self.max_iter, ConvergenceWarning)
(0.0001, 10, 0.001)
0.67
```

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2_c
onfusion.py --classifier "ovr"
/home/garvita/Desktop/Assignment2/data_3.h5
[[ 4.  0.  0.]
 [ 8.  0.  0.]
 [ 0.  0.  8.]]
```

Dataset 4

One vs One

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2.p
y --classifier "ovo"
/home/garvita/.local/lib/python2.7/site-packages/sklearn/svm/base.py:218: Conver
genceWarning: Solver terminated early (max_iter=10).  Consider pre-processing yo
ur data with StandardScaler or MinMaxScaler.
  % self.max_iter, ConvergenceWarning)
/home/garvita/.local/lib/python2.7/site-packages/sklearn/svm/base.py:218: Conver
genceWarning: Solver terminated early (max_iter=100).  Consider pre-processing y
our data with StandardScaler or MinMaxScaler.
  % self.max_iter, ConvergenceWarning)
(0.0001, 10, 0.001)
0.509
```

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2_c
onfusion.py --classifier "ovo"
/home/garvita/Desktop/Assignment2/data_4.h5
[[ 189.    0.]
 [ 211.    0.]]
```

One vs Rest

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2.p
y --classifier "ovr"
/home/garvita/Desktop/Assignment2/data_4.h5
(1, -1, 0.001)
0.5
```

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2_c
onfusion.py --classifier "ovr"
/home/garvita/Desktop/Assignment2/data_4.h5
[[ 189.    0.]
 [ 211.    0.]]
```

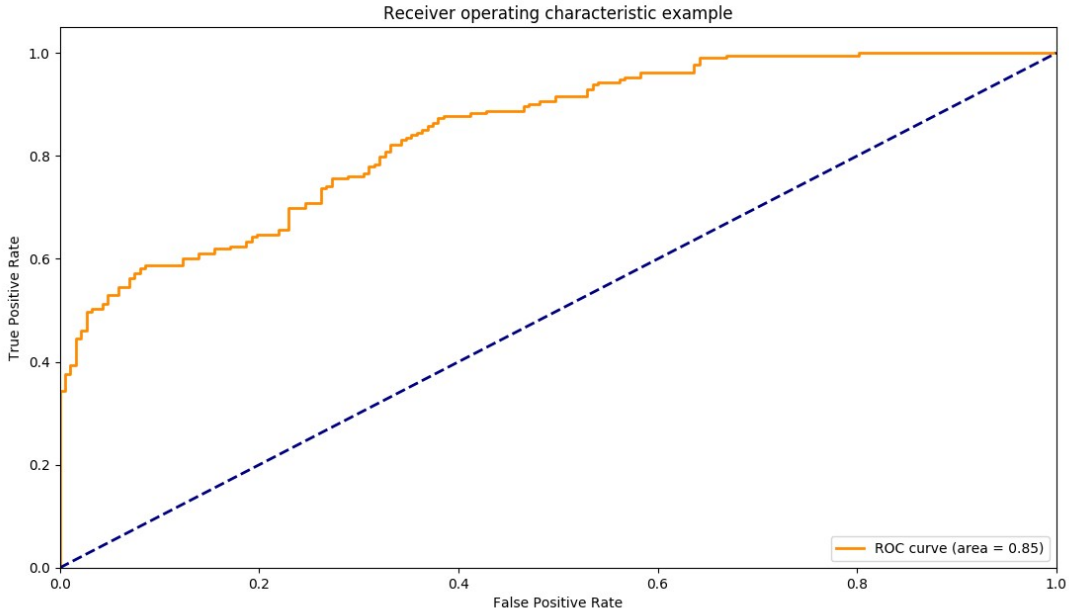
Dataset 5

One vs One

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2.p
y --classifier "ovo"
/home/garvita/.local/lib/python2.7/site-packages/sklearn/svm/base.py:218: Conver
genceWarning: Solver terminated early (max_iter=10).  Consider pre-processing yo
ur data with StandardScaler or MinMaxScaler.
  % self.max_iter, ConvergenceWarning)
/home/garvita/.local/lib/python2.7/site-packages/sklearn/svm/base.py:218: Conver
genceWarning: Solver terminated early (max_iter=100).  Consider pre-processing y
our data with StandardScaler or MinMaxScaler.
  % self.max_iter, ConvergenceWarning)
(0.01, -1, 0.01)
0.7795
```

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2_c
onfusion.py --classifier "ovo"
/home/garvita/Desktop/Assignment2/data_5.h5
[[  99.   88.]
 [  21.  192.]]
```

ROC Curve:



One vs Rest

```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2.p
y --classifier "ovr"
/home/garvita/Desktop/Assignment2/data_5.h5
(1, -1, 0.001)
0.736
```

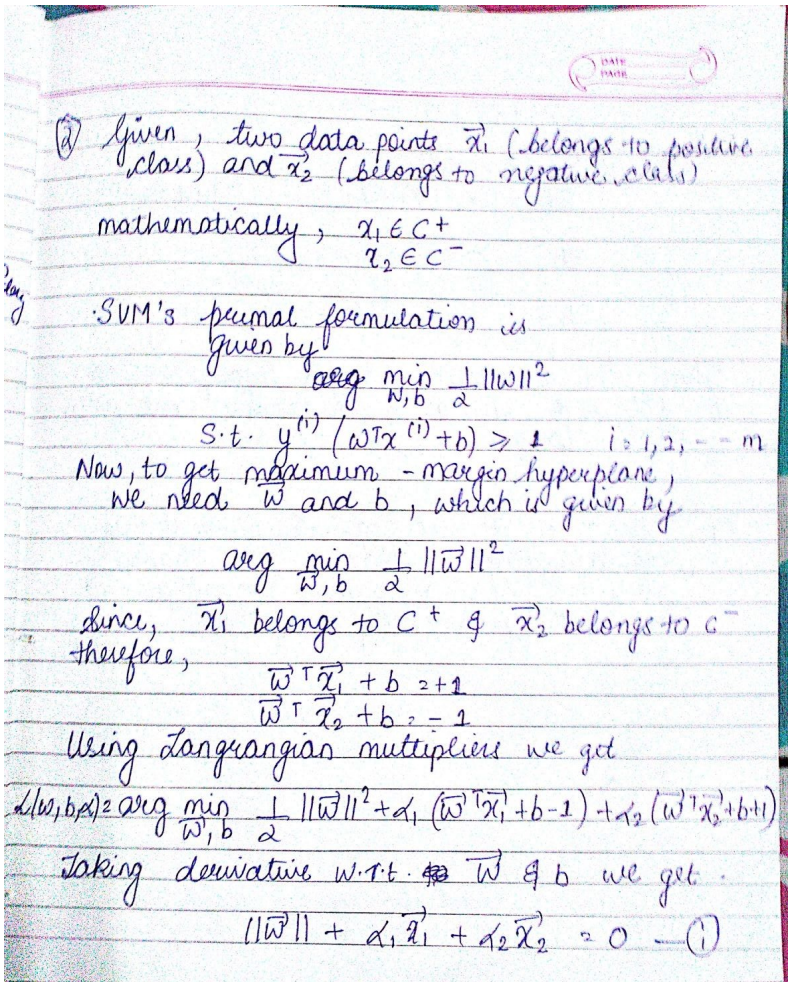
```
garvita@garvita-HP-Pavilion-15-Notebook-PC:~/Desktop/Assignment2$ python SVM_2_c
onfusion.py --classifier "ovr"
/home/garvita/Desktop/Assignment2/data_5.h5
[[  99.   88.]
 [  21.  192.]]
```

Theory Questions

1.
Consider the RBF kernel: it can map the given data into a higher dimensional space, with possibly infinite dimensions. There are two ways to look at it: one way is that every data points gets its own dimension, which would lead to overfitting. However, in reality, this generally does not happen. Why?

Ans: It doesn't lead to overfitting because it uses **regularization** and it reduces the chance of overfitting.**Choosing right value of C(should be low) and gamma** prevent data from being overfitted.

2. Show that, irrespective of the dimensionality of the data space, a data set consisting of just two data points, one from each class, is sufficient to determine the location of the maximum-margin hyperplane. You are expected to give a mathematical proof.



(2) Given, two data points $\vec{x_1}$ (belongs to positive class) and $\vec{x_2}$ (belongs to negative class)

mathematically, $x_1 \in C^+$
$x_2 \in C^-$

SVM's primal formulation is given by

$$\arg \min_{w,b} \frac{1}{2}\|w\|^2$$

$$s.t.\ y^{(i)}(w^T x^{(i)} + b) \geq 1 \quad i = 1, 2, \ldots m$$

Now, to get maximum-margin hyperplane, we need $\vec{w}$ and b, which is given by

$$\arg \min_{\vec{w},b} \frac{1}{2}\|\vec{w}\|^2$$

since, $\vec{x_1}$ belongs to C^+ & $\vec{x_2}$ belongs to C^-
therefore,

$$\vec{w}^T \vec{x_1} + b = +1$$
$$\vec{w}^T \vec{x_2} + b = -1$$

Using Langrangian multipliers we get

$$L(w,b,\alpha) = \arg \min_{\vec{w},b} \frac{1}{2}\|\vec{w}\|^2 + \alpha_1(\vec{w}^T\vec{x_1} + b - 1) + \alpha_2(\vec{w}^T\vec{x_2} + b + 1)$$

Taking derivative w.r.t. $\vec{w}$ & b we get.

$$\|\vec{w}\| + \alpha_1\vec{x_1} + \alpha_2\vec{x_2} = 0 \quad \text{--- (1)}$$

$\alpha_1 + \alpha_2 = 0$ —(ii)

we get, $\alpha_1 = -\alpha_2$

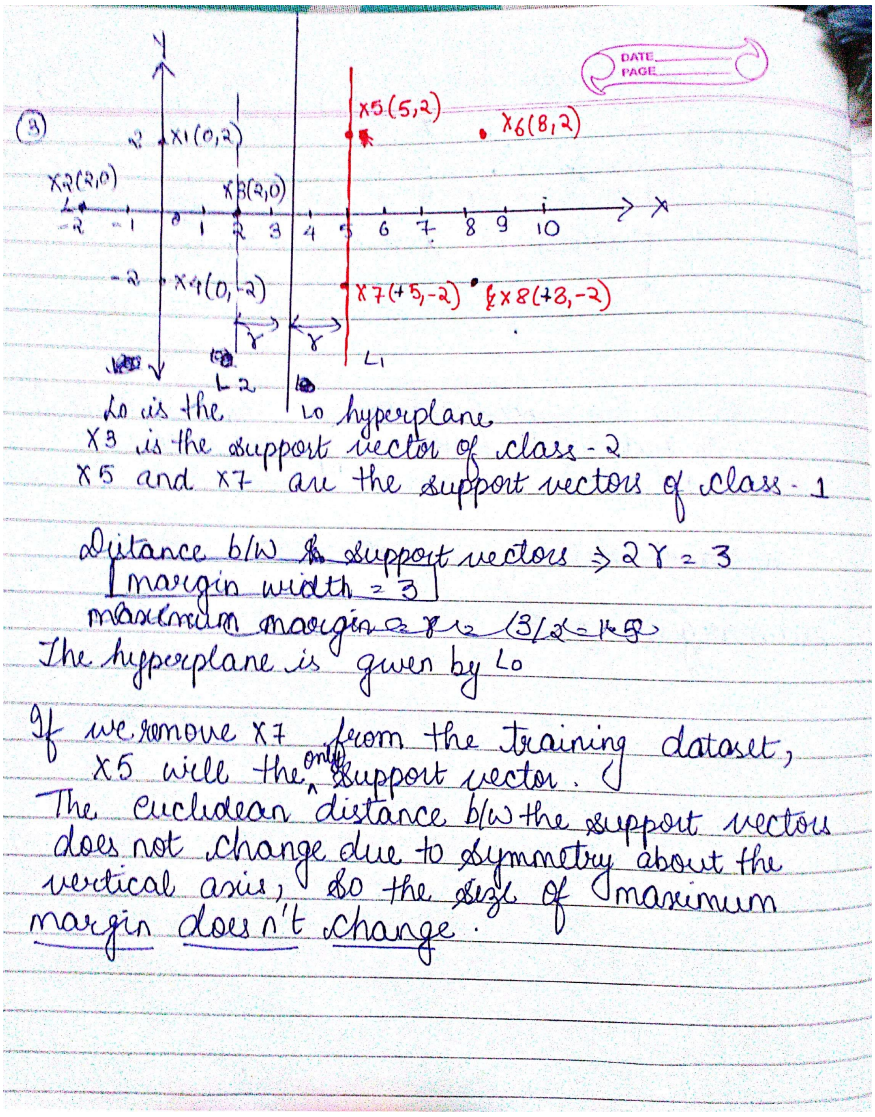
$\vec{w_e} = \alpha_1 (x_1 - x_2)$ [using (i) & (ii)]

$2b = -\vec{w}^T (x_1 + x_2)$

$b = -\frac{1}{2} (\vec{w}^T (x_1 + x_2))$ [using (i) & (ii)]

Thus, the maximum margin hyperplane can be obtained using 2 datapoints, one from each class.

3. Consider the hard-margin SVM. Find the maximum margin for the following case. How does the size of maximum margin changes if we remove X7 from the training dataset.



L_0 is the L_0 hyperplane.

X_3 is the support vector of class -2

X_5 and X_7 are the support vectors of class -1

Distance b/w 2 support vectors $\Rightarrow 2\gamma = 3$

[margin width = 3]

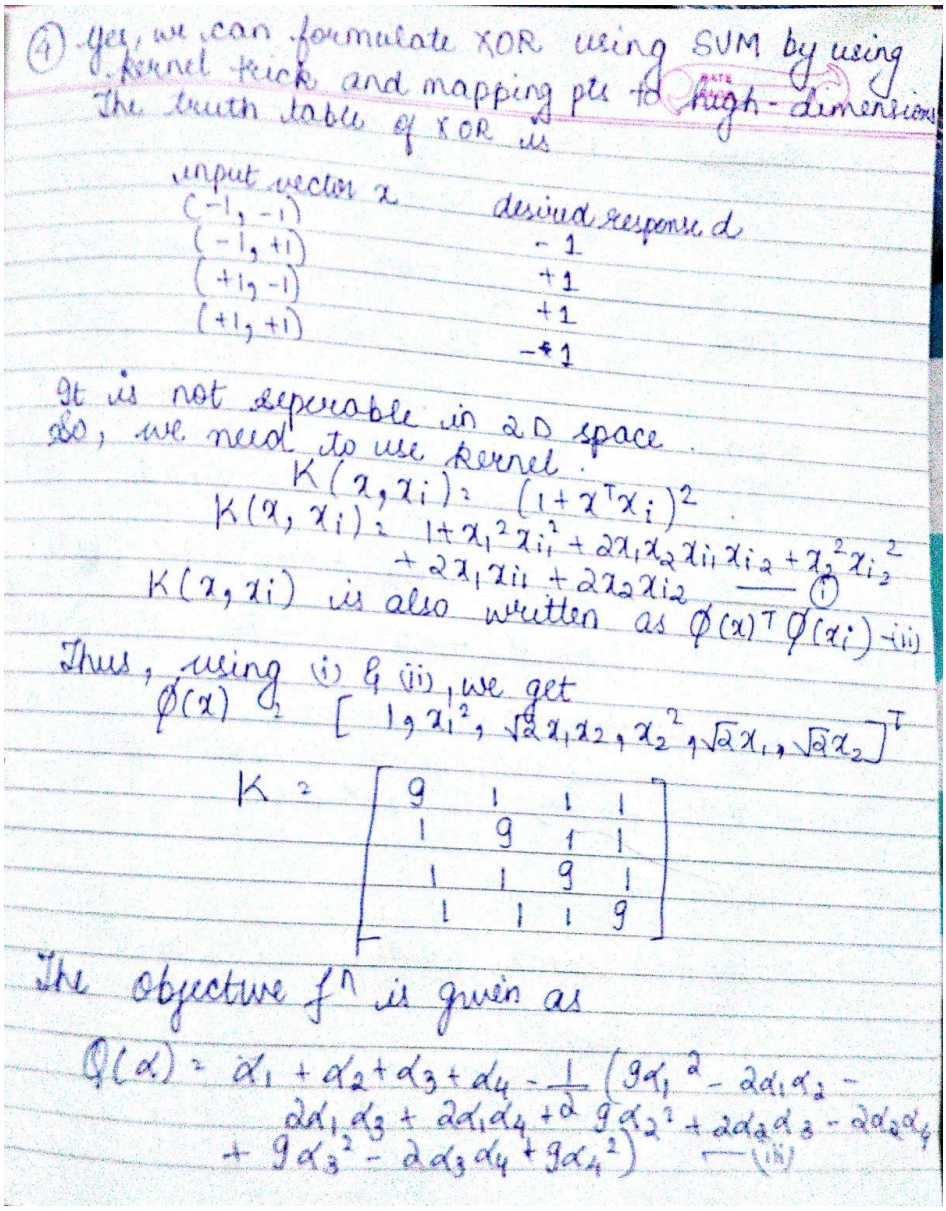
maximum margin $= \gamma = (3/2 = 1.5)$

The hyperplane is given by L_0

If we remove X_7 from the training dataset, X_5 will be the only support vector.

The euclidean distance b/w the support vectors does not change due to symmetry about the vertical axis, so the size of maximum margin doesn't change.

4. Can you model the XOR operator using an SVM? Justify.



① Yes, we can formulate XOR using SVM by using kernel trick and mapping pts to high-dimensions.

The truth table of XOR is

input vector x	desired response d
$(-1, -1)$	-1
$(-1, +1)$	$+1$
$(+1, -1)$	$+1$
$(+1, +1)$	-1

It is not seperable in 2D space.
So, we need to use kernel.

$$K(x, x_i) = (1 + x^T x_i)^2$$

$$K(x, x_i) = 1 + x_1^2 x_{i1}^2 + 2x_1 x_2 x_{i1} x_{i2} + x_2^2 x_{i2}^2 + 2x_1 x_{i1} + 2x_2 x_{i2} \quad \text{— (i)}$$

$K(x, x_i)$ is also written as $\phi(x)^T \phi(x_i)$ — (ii)

Thus, using (i) & (ii), we get

$$\phi(x) = [1, x_1^2, \sqrt{2}x_1 x_2, x_2^2, \sqrt{2}x_1, \sqrt{2}x_2]^T$$

$$K = \begin{bmatrix} 9 & 1 & 1 & 1 \\ 1 & 9 & 1 & 1 \\ 1 & 1 & 9 & 1 \\ 1 & 1 & 1 & 9 \end{bmatrix}$$

The objective fn is given as

$$Q(\alpha) = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 - \frac{1}{2}(9\alpha_1^2 - 2\alpha_1\alpha_2 - 2\alpha_1\alpha_3 + 2\alpha_1\alpha_4 + 9\alpha_2^2 + 2\alpha_2\alpha_3 - 2\alpha_2\alpha_4 + 9\alpha_3^2 - 2\alpha_3\alpha_4 + 9\alpha_4^2) \quad \text{— (iii)}$$

$$\frac{\partial Q(\alpha)}{\partial \alpha_1} = 1 - \frac{1}{2}(18\alpha_1 - 2\alpha_2 - 2\alpha_3 + 2\alpha_4) = 0$$

$$\Rightarrow 9\alpha_1 - \alpha_2 - \alpha_3 + \alpha_4 = 1 \quad \text{— (iv)}$$

$$\frac{\partial Q(\alpha)}{\partial \alpha_2} = 1 - \frac{1}{2}(-2\alpha_1 + 18\alpha_2 + 2\alpha_3 - 2\alpha_4)$$

$$\Rightarrow -\alpha_1 + 9\alpha_2 + \alpha_3 - \alpha_4 = 1 \quad \text{— (v)}$$

$$\frac{\partial Q(\alpha)}{\partial \alpha_3} = 1 - \frac{1}{2}(-2\alpha_1 + 2\alpha_2 + 18\alpha_3 - 2\alpha_4)$$

$$\Rightarrow -\alpha_1 + \alpha_2 + 9\alpha_3 - \alpha_4 = 1 \quad \text{— (vi)}$$

$$\frac{\partial Q(\alpha)}{\partial \alpha_4} = 1 - \frac{1}{2}(2\alpha_1 - 2\alpha_2 - 2\alpha_3 + 18\alpha_4)$$

$$\Rightarrow \alpha_1 - \alpha_2 - \alpha_3 + 9\alpha_4 = 1 \quad \text{— (vii)}$$

Solving (iv), (v), (vi) & (vii), we get

$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0.125 \quad \text{— (viii)}$$

Putting (viii) in eq (ii), we get

$$\Rightarrow Q_0(\alpha) = 1/4$$

$Q_0(\alpha)$ can be written as $\frac{1}{2}\|w_0\|^2$

$$\frac{1}{2}\|w_0\|^2 = \frac{1}{4}$$

$$\|w_0\| = \frac{1}{\sqrt{2}}$$

Optimum weight vector w_0

$$\Rightarrow w_0 = \frac{1}{8}[-\phi(x_1) + \phi(x_2) + \phi(x_3) - \phi(x_4)]$$

$$\frac{1}{8}\left[-\begin{bmatrix}1\\1\\\sqrt{2}\\1\\-\sqrt{2}\\-\sqrt{2}\end{bmatrix} + \begin{bmatrix}1\\1\\-\sqrt{2}\\1\\-\sqrt{2}\\\sqrt{2}\end{bmatrix} + \begin{bmatrix}1\\1\\-\sqrt{2}\\1\\\sqrt{2}\\-\sqrt{2}\end{bmatrix} - \begin{bmatrix}1\\1\\\sqrt{2}\\1\\\sqrt{2}\\\sqrt{2}\end{bmatrix}\right]$$

$$= \begin{bmatrix}0\\0\\-1/\sqrt{2}\\0\\0\\0\end{bmatrix}$$

Hyperplane is $w_0^T \phi(x) = 0$

$$\begin{bmatrix}0 & 0 & \frac{-1}{\sqrt{2}} & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}1\\x_1^2\\\sqrt{2}x_1x_2\\x_2^2\\\sqrt{2}x_1\\\sqrt{2}x_2\end{bmatrix} = 0$$

$$\Rightarrow -x_1x_2 = 0$$

XOR can be depicted as