Support Vector Machine

1. Problem Statement

The two feature data set should be classified using Support Vector Machine algorithm. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on. The data should be trained and tested using the SVM. Experiment on the kernel trick on the SVM algorithm.

2. Problem Solution

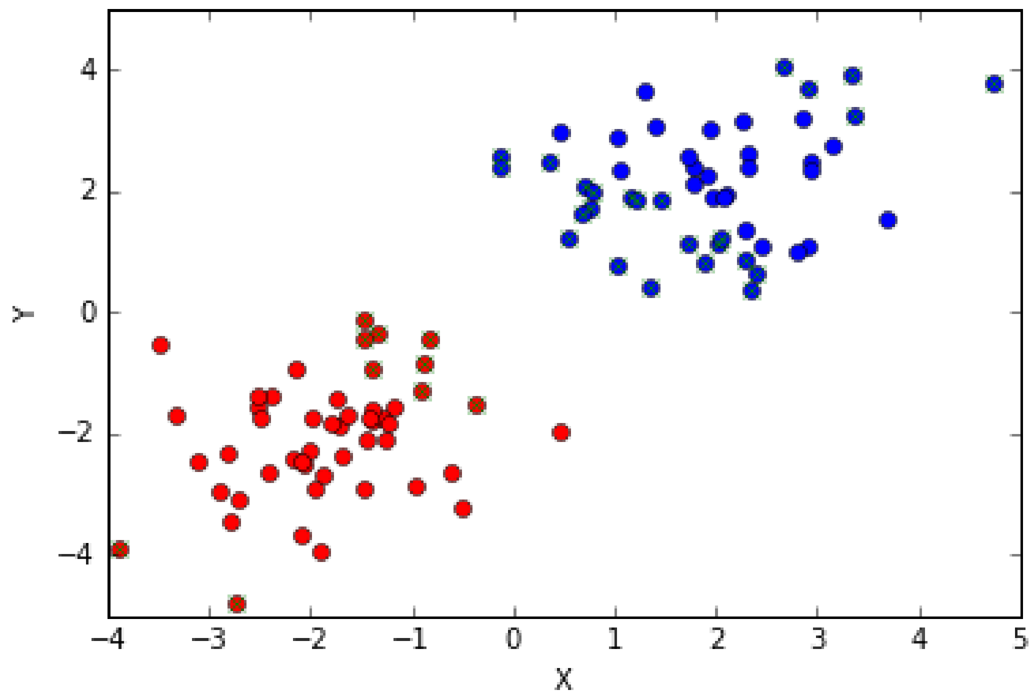
The solution is to find the parameters of SVM. The first step is to create the data which is linearly separable and non separable select the model. To find alpha using the CVXOPT function in python solve for alpha. By using the alpha find the weights of the features and bias. Implementation of kernel trick using polynomial and Gaussian implementation.

3. Implementation Details

I have implemented the code in ipython notebook. The filename has to be mentioned and in the tool bar option “Cell” -> “Run All” will implemented the whole file and the results will be printed.

4. Results and Discussion

1. Creation of Linearly separable data and Hard Margin Classification

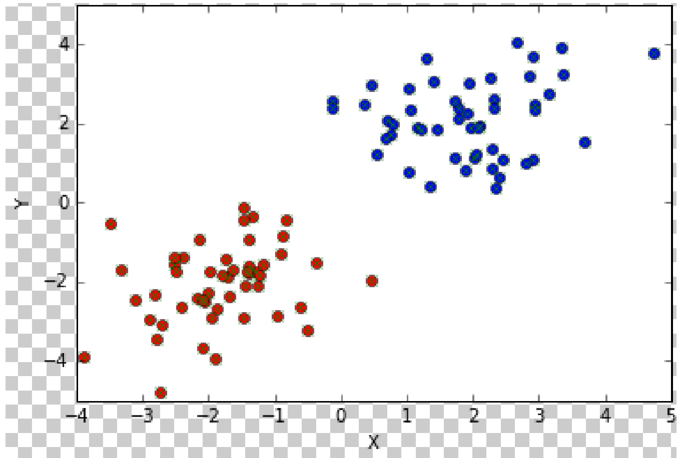


The data point with the cross over them is the support vectors.

The evaluation is as below

```
Confusion Matrix
[[50  0]
 [ 0 50]]
Accuracy 1.0
Precision [1.0, 1.0]
Recall [1.0, 1.0]
F_score [1.0, 1.0]
```

2. Soft Margin Classification Evaluation for Linearly Separable Data

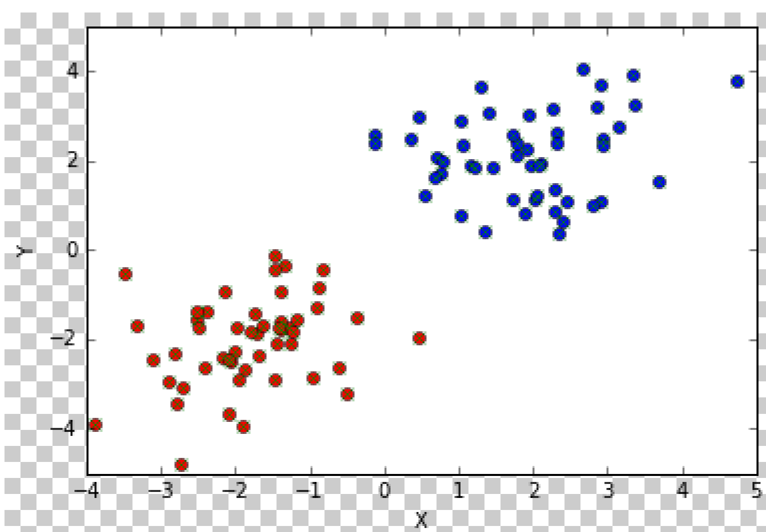


The data point with the cross over them is the support vectors.

The evaluation is as follows:

```
Confusion Matrix
[[50  0]
 [ 0 50]]
Accuracy 1.0
Precision [1.0, 1.0]
Recall [1.0, 1.0]
F_score [1.0, 1.0]
```

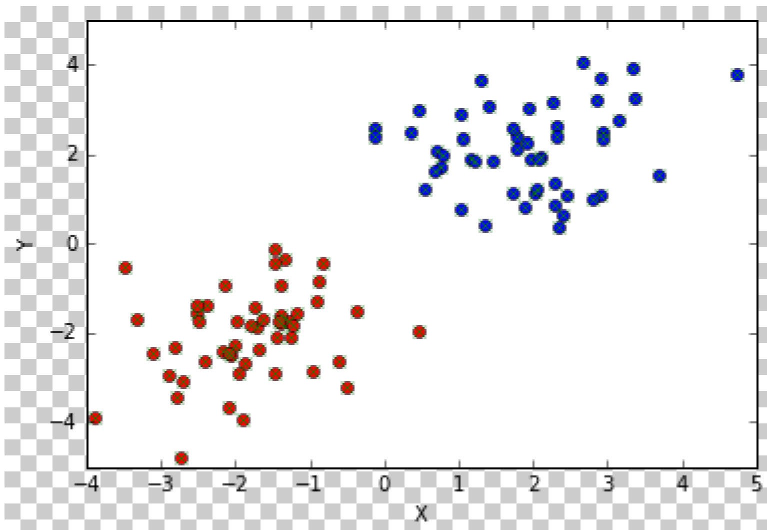
3. Gaussian Kernel: Hard margin; Linearly Separable Data:



The evaluation is as follows:

```
Confusion Matrix
[[50  0]
 [ 0 50]]
Accuracy 1.0
Precision [1.0, 1.0]
Recall [1.0, 1.0]
F_score [1.0, 1.0]
```

4. Gaussian Kernel: Soft Margin; Linearly Separable Data:

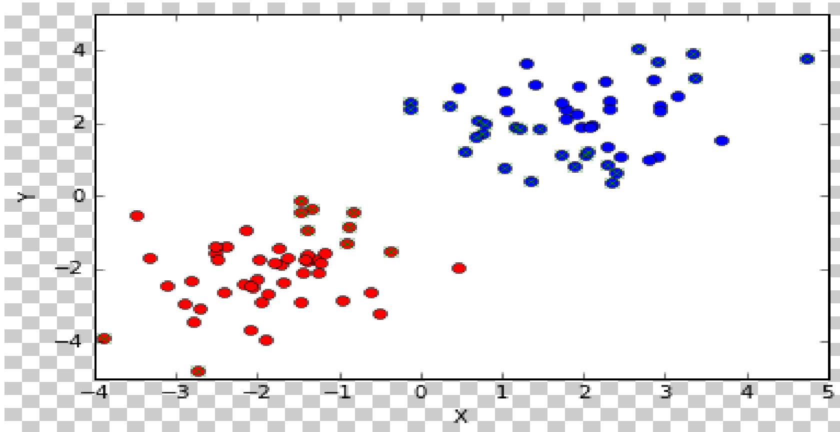


The data point with the cross over them is the support vectors.

The evaluation is as follows:

```
Confusion Matrix
[[48  2]
 [ 0 50]]
Accuracy 0.98
Precision [1.0, 0.96153846153846156]
Recall [0.95999999999999996, 1.0]
F_score [0.97959183673469385,
0.98039215686274506]
```

5. Polynomial Kernel: Hard Margin; Linearly Separable Data:

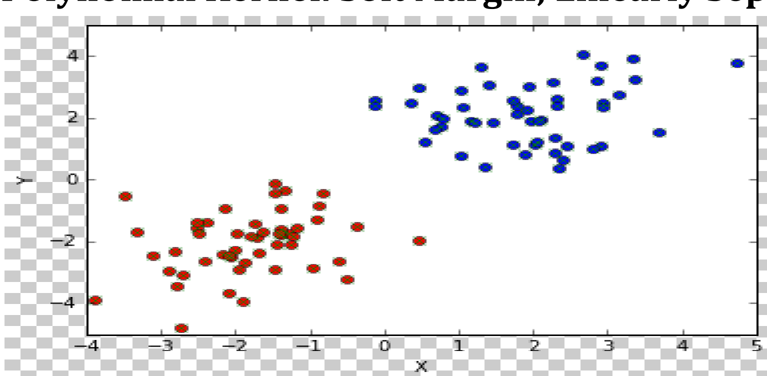


The data point with the cross over them is the support vectors.

The evaluation is as follows:

```
Confusion Matrix
[[50  0]
 [ 0 50]]
Accuracy 1.0
Precision [1.0, 1.0]
Recall [1.0, 1.0]
F_score [1.0, 1.0]
```

6. Polynomial Kernel: Soft Margin; Linearly Separable Data:



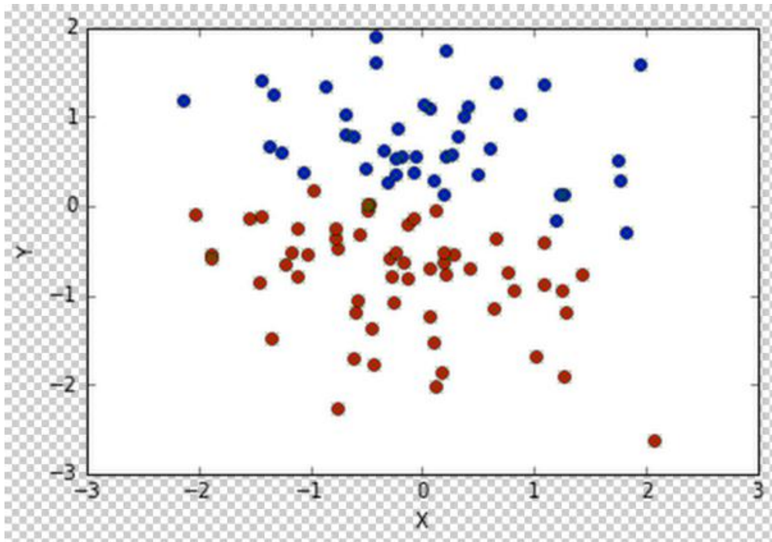
The data point with the cross over them is the support vectors.

The evaluation is as follows:

```
Confusion Matrix
[[50  0]
 [ 0 50]]
Accuracy 1.0
Precision [1.0, 1.0]
Recall [1.0, 1.0]
```

```
F_score [1.0, 1.0]
```

7. Data Generation and Hard Margin Classification; Non-Linearly Separable Data:

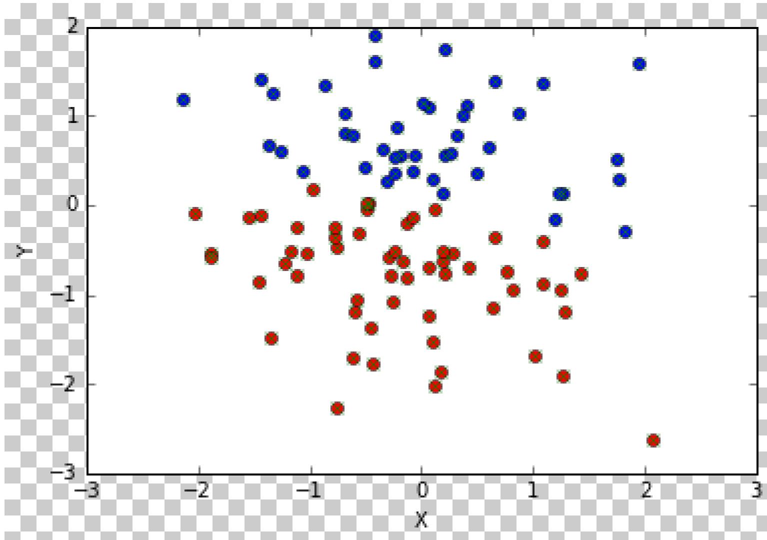


The data point with the cross over them is the support vectors.

The evaluation is as follows:

```
Confusion Matrix
[[57  0]
 [ 5 38]]
Accuracy 0.95
Precision [0.919354838709677 38, 1.0]
Recall [1.0, 0.8837209302325 5816]
F_score [0.957983193277310 85,
0.938271604938271 66]
```

8. Data Generation and Soft Margin Classification; Non-Linearly Separable Data:

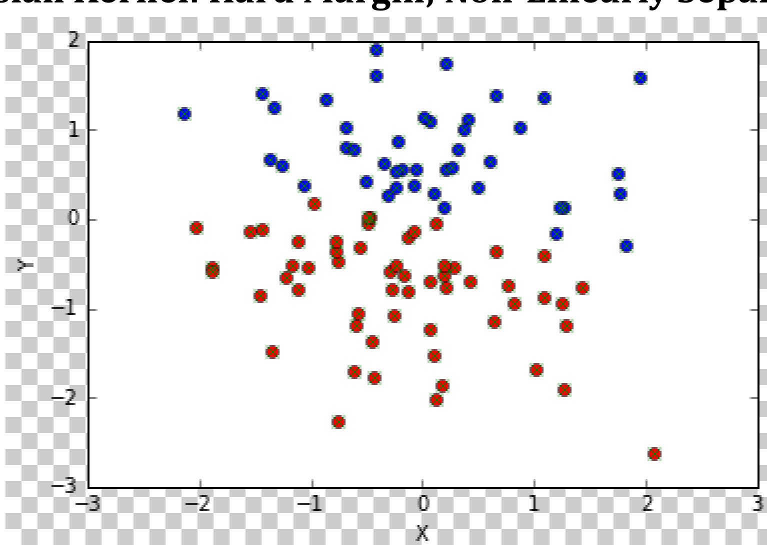


The data point with the cross over them is the support vectors.

The evaluation is as follows:

```
Confusion Matrix
[[57  0]
 [ 0 43]]
Accuracy 1.0
Precision [1.0, 1.0]
Recall [1.0, 1.0]
F_score [1.0, 1.0]
```

9. Gaussian Kernel: Hard Margin; Non-Linearly Separable Data:

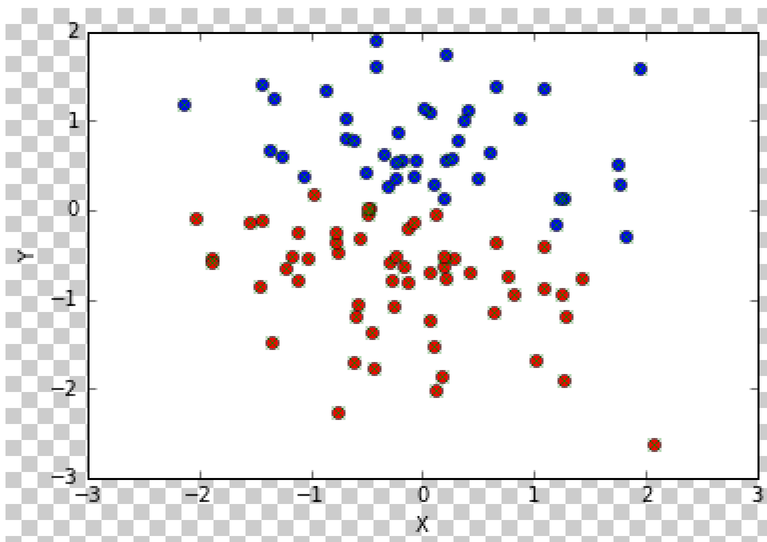


The data point with the cross over them is the support vectors.

The evaluation is as follows:

```
Confusion Matrix
[[57  0]
 [ 5 38]]
Accuracy 0.95
Precision [0.91935483870967738, 1.0]
Recall [1.0, 0.88372093023255816]
F_score [0.95798319327731085,
0.93827160493827166]
```

10. Gaussian Kernel: Soft Margin; Linearly Non-Linearly Separable Data:

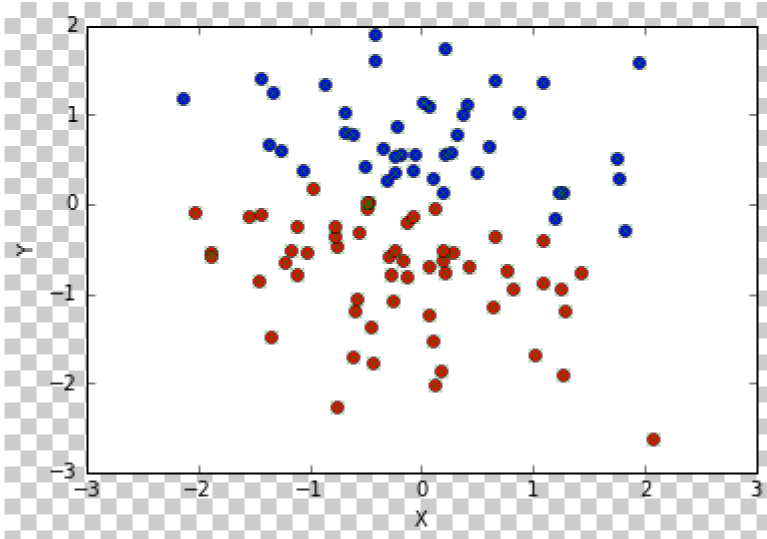


The data point with the cross over them is the support vectors.

The evaluation is as follows:

```
Confusion Matrix
[[57  0]
 [ 6 37]]
Accuracy 0.94
Precision [0.90476190476190477, 1.0]
Recall [1.0, 0.86046511627906974]
F_score [0.95000000000000007,
0.92499999999999993]
```

11. Polynomial Kernel: Hard Margin; Non-Linearly Separable Data:

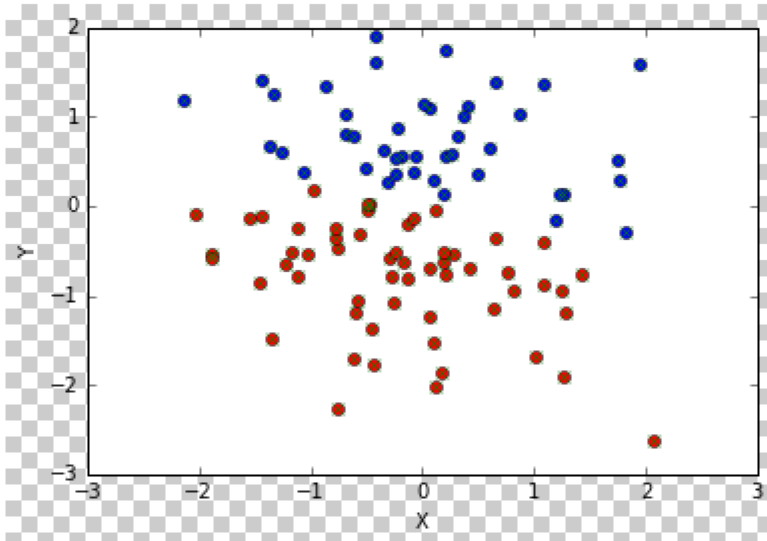


The data point with the cross over them is the support vectors.

The evaluation is as follows:

```
Confusion Matrix
[[57  0]
 [ 5 38]]
Accuracy 0.95
Precision [0.91935483870967738, 1.0]
Recall [1.0, 0.88372093023255816]
F_score [0.95798319327731085,
0.93827160493827166]
```

12. Polynomial Kernel: Soft Margin; Non-Linearly Separable Data:



The data point with the cross over them is the support vectors.

The evaluation is as follows:

```
Confusion Matrix
[[57  0]
 [ 0 43]]
Accuracy 1.0
Precision [1.0, 1.0]
Recall [1.0, 1.0]
F_score [1.0, 1.0]
```

13. External Data : "breast-cancer-wisconsin.data.txt"

Soft Margin

```
Confusion Matrix
[[458   0]
 [ 90 151]]
Accuracy 0.871244635193
Precision [0.83576642335766427, 1.0]
Recall [1.0, 0.62655601659751037]
F_score [0.91053677932405563, 0.77040816326530615]
```

Hard Margin

```
Confusion Matrix
[[458   0]
 [ 90 151]]
Accuracy 0.871244635193
Precision [0.83576642335766427, 1.0]
```

```
Recall [1.0, 0.62655601659751037]
F_score [0.91053677932405563, 0.77040816326530615]
```

Soft Margin; Polynomial Kernel

```
Confusion Matrix
[[458   0]
 [ 90 151]]
Accuracy 0.871244635193
Precision [0.83576642335766427, 1.0]
Recall [1.0, 0.62655601659751037]
F_score [0.91053677932405563, 0.77040816326530615]
```

Soft Margin; Gram Matrix Kernel

```
Confusion Matrix
[[458   0]
 [ 90 151]]
Accuracy 0.871244635193
Precision [0.83576642335766427, 1.0]
Recall [1.0, 0.62655601659751037]
F_score [0.91053677932405563, 0.77040816326530615]
```

Hard Margin; Polynomial Kernel

```
Confusion Matrix
[[458   0]
 [ 90 151]]
Accuracy 0.871244635193
Precision [0.83576642335766427, 1.0]
Recall [1.0, 0.62655601659751037]
F_score [0.91053677932405563, 0.77040816326530615]
```

Soft Margin; Polynomial Kernel

```
Confusion Matrix
[[458   0]
 [ 90 151]]
Accuracy 0.871244635193
Precision [0.83576642335766427, 1.0]
Recall [1.0, 0.62655601659751037]
F_score [0.91053677932405563, 0.77040816326530615]
```

Question number 3

$$L_p = \frac{1}{2}\|W\|^2 + C\sum_{i=1}^{m} \xi_i - \sum_{i=1}^{m} \alpha_i \left(y^{(i)}(W^T x_i + w_0) - 1 + \xi_i\right) - \sum_{i=1}^{m} \beta_i \xi_i$$

$$\nabla L_p \Rightarrow \frac{\partial L_p}{\partial w} = 0 \qquad \frac{\partial L_p}{\partial w_0} = 0 \qquad \frac{\partial L_p}{\partial \xi_i} = 0$$

$$\frac{\partial L_p}{\partial w_i} = \frac{1}{2} \times 2(W) + 0 - \sum_{i=1}^{m} \alpha_i (y^{(i)} x^i) = 0$$

$$w_r = \sum_{i=1}^{m} \alpha_i y^{(i)} x_i$$

$$\frac{\partial L_p}{\partial w_0} = -\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

$$\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

$$\frac{\partial L_p}{\partial \xi_i} = C - \alpha_i - \beta_i = 0$$

$$L_p = \frac{1}{2}\left(\sum_{i=1}^{m} \alpha_i y^{(i)} x_i\right)^2 + C\sum_{i=1}^{m} \xi_i - \sum_{i=1}^{m} \alpha_i \left(y^{(i)}\left(\sum_{i=1}^{m} \alpha_i y^{(i)} x_i\right)^T x_i\right) + \sum_{i=1}^{m} \alpha_i y$$

$$+ \sum_{i=1}^{m} \alpha_i - \sum_{i=1}^{m} \alpha_i \xi_i + \sum_{i=1}^{m} \beta_i \xi_i$$

$$L_D = -\frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y^{(i)} y^{(j)} \cdot x^{(i)T} \cdot x^{(j)} + \sum_{i=1}^{m} \alpha_i$$

5. References

https://en.wikipedia.org/wiki/Support_vector_machine