ECE523 Engineering Applications of Machine Learning and Data Analytics

Homework 3

1 Support Vector Machines

$$Ł(\boldsymbol{W}, b, \xi, \alpha) = \frac{1}{2}\|\boldsymbol{W}\|^2 + \frac{C}{2}\sum_{i=1}^{n}\xi_i^2 - \sum_{i=1}^{n} a_i[y_i(\boldsymbol{W}^T\boldsymbol{X}_i + b) + \xi_i - 1]$$

$$\frac{\partial Ł}{\partial \boldsymbol{W}} = 0 \quad \Rightarrow \quad \boldsymbol{W} = \sum_{i=1}^{n}\alpha_i y_i \boldsymbol{X}_i \tag{1}$$

$$\frac{\partial Ł}{\partial b} = 0 \quad \Rightarrow \quad \sum_{i=1}^{n}\alpha_i y_i = 0 \tag{2}$$

$$\frac{\partial Ł}{\partial \xi_i} = 0 \quad \Rightarrow \quad \xi_i = \frac{\alpha_i}{C} \tag{3}$$

Introduce (1), (2), (3) into Ł, We will get:

$$\arg\min_{\alpha} \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_t\alpha_j y_i y_j \boldsymbol{X}_i^T\boldsymbol{X}_j - \frac{1}{2C}\sum_{i=1}^{n}\alpha_i^2$$

$$s.t. \quad \alpha_i \geq 0 \quad \forall i \in [n] \quad and \quad \sum_{i=1}^{n}\alpha_i y_i = 0$$

2 Multi-Layer Perceptron

	Classification Error	
	trainging	testing
50HLN+no regularization	0.0247	0.0495
50HLN+L2 regularization	0.0670	0.0649
250HLN+no regularization	0.0031	0.0419
250HLN+L2 regularization	0.0631	0.0612

Parameters:
Learing Rounds: 20000
Regularization Coefficient : 0.01
Learning Rate: 0.001

3 Support Vector Machines

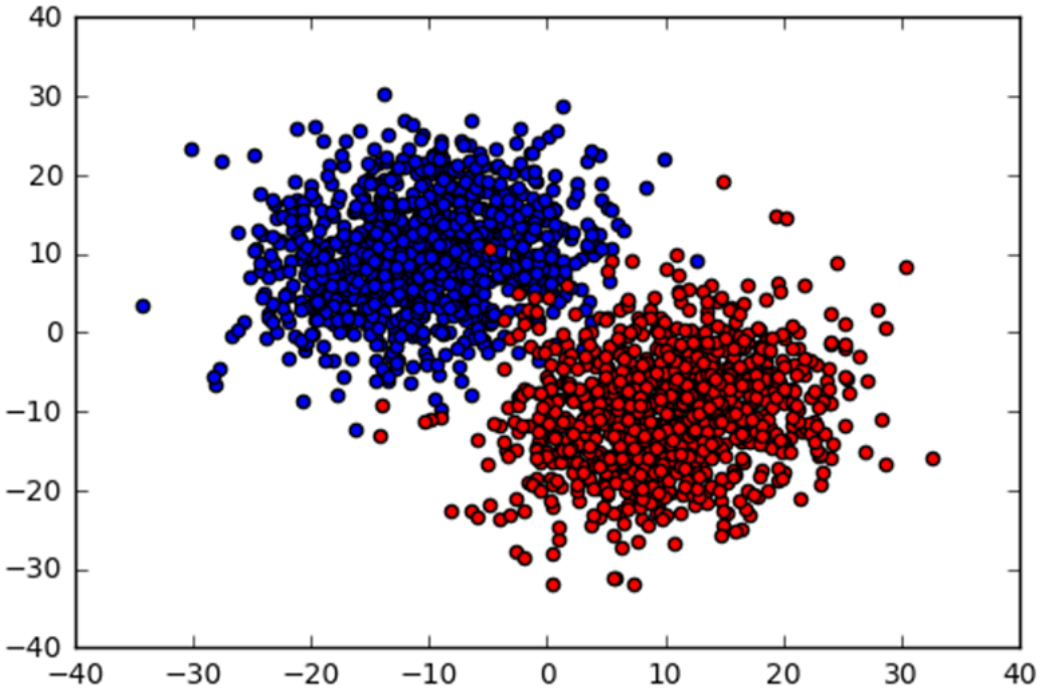

Fig.1 Traing data

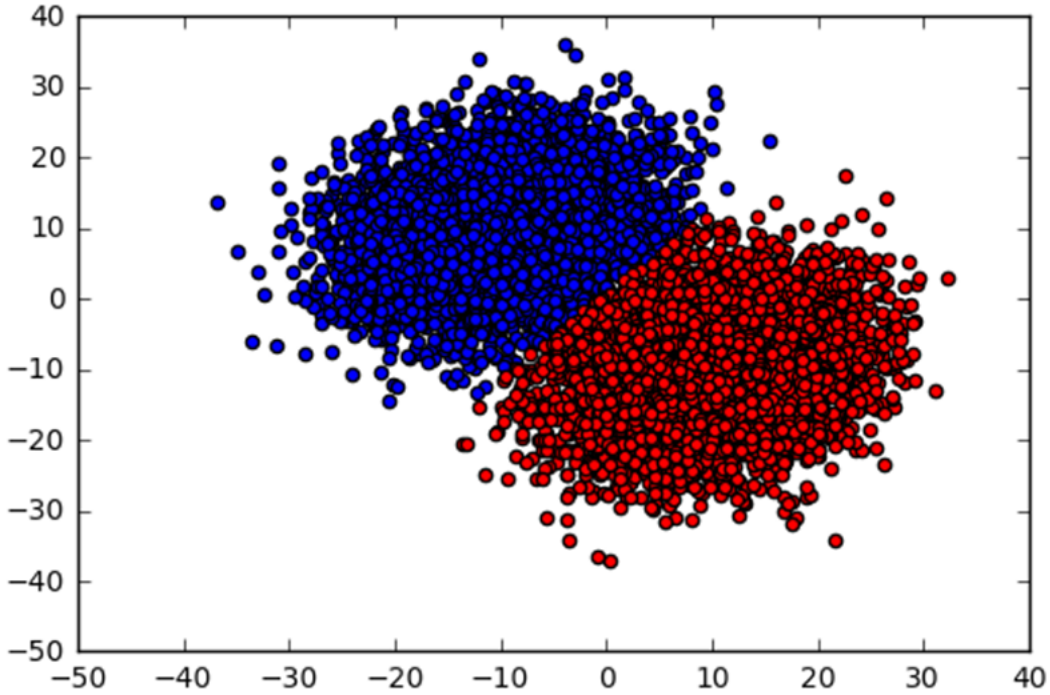

Fig.2 Test data with linear kernel

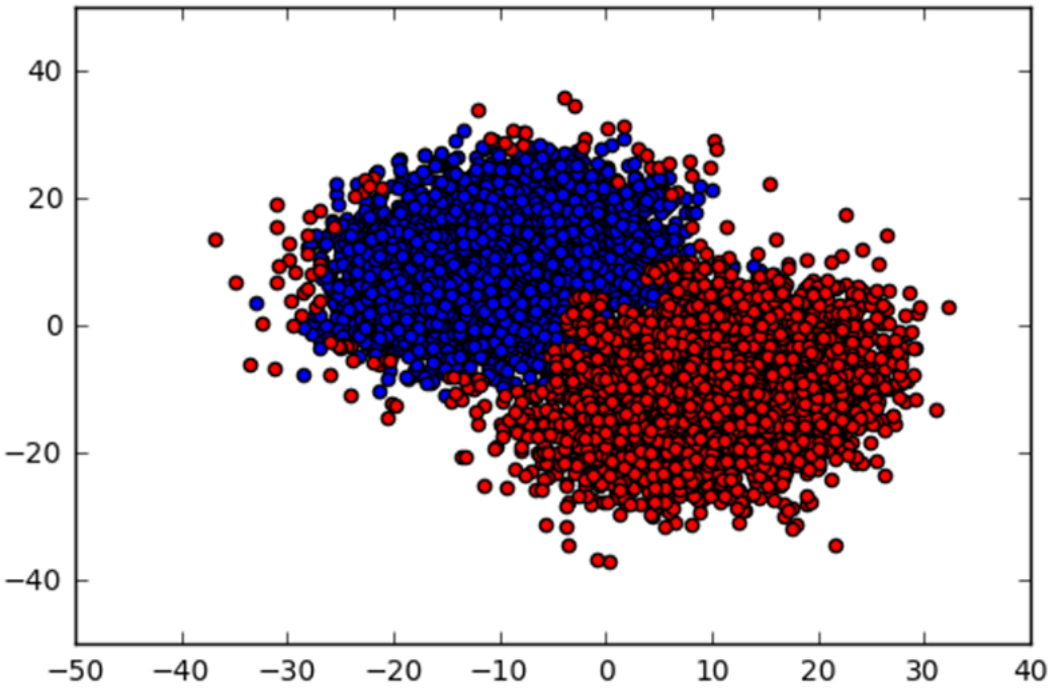

Fig.3 Test data with gaussian kernel

Table 1 Classification error with different kernels

	Classification Error	
	trainging	testing
Linear Kernel	0.0150	0.0131
Guassian Kernel	0.0035	0.0287

4 Deep Learning (Bonus)

Table 2 Classification error - Deep network VS Shallow network

		Classification Error	
		trainging	testing
1 hidden layer	250HLN+no regularization	0.0031	0.0419
	250HLN+L2 regularization	0.0631	0.0612
3 hidden layer	250HLN+no regularization	0.0024	0.0423
	250HLN+L2 regularization	0.0105	0.0442

From the table, I find that the deep network can improve the accuracy, and regularization is a good way to prevent overfitting.