CSE512 Fall 2019 - Machine Learning - Homework 4

November 2, 2019

Name: Anmol Shukla

Solar ID: 112551470

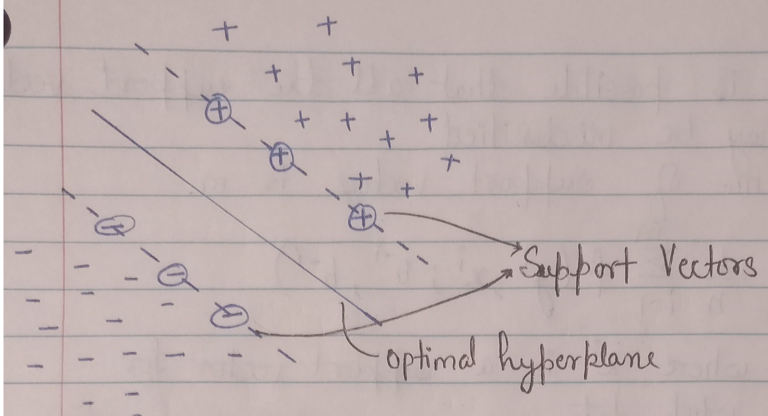
NetID: anmshukla

Email address: Anmol.shukla@stonybrook.edu

Names of people whom I have discussed the homework with: None

1 Question 1

1.1 Linear Case



The main objective of SVM is to maximize the margin i.e. maximize the distance of closest points from the optimal seperating plane.

LOOCV error for SVM = $\sum_{i=1}^{n} f(y^i, x_i; \theta^{-i} b^{-i})$

y^i expected = $x_i^T \theta^{-i} + b^{-i}$

where θ^{-i} is the value of weights leaving i^{th} example
b^i is the bias after leaving out the ith example.
if $y^i \times y^i$expected $= 1$ (i.e. pt. is correctly classified)
else missclassified
and value of $f(x) = 0$ if $x = 1$, else 0
Now, considering 2 cases when leaving out a point:-

(i) If a point is not a support vector :- the margins do not change since the margins are solely defined by support vectors. $\therefore f(x) = 0$.

(ii) If a point is a support vector, then the margins will shift giving a new classifier plane. This may lead to the case where this point is misclassified.

$f(x) = 0$ if pt. is correctly classifed

$= 1$ if misclassified.

<u>Worst case</u>: it is possible that all the support vectors may be misclassified

$\therefore$ no. of support vectors is m.

$$\text{LOOCV error} = \frac{1}{n} \sum_{i=1}^{m} f(y^i, x^i; \theta^{-i}, b^{-i})$$

where x^i is a support vector for whole data.

For worst case scenario $f(x) = 1 \quad \forall x \in$ Support Vectors

$\therefore$ the maximum LOOCV error $= \frac{1}{n} \sum_{i=1}^{m} (1)$

$$\Rightarrow \frac{m}{n}$$

1.2 General Case

1.2. If we use a general kernel that separates the data in high-dimensional feature space, then, the bound on LOOCV will still hold.

Similar to the previous answer, if the data is linearly separable in higher dimension. Thus, removing non-support vector won't change the SVM margin & thus won't add any error. On the other hand, if in the higher dimension, a support vector is removed, then, margin will change. This will add one to the LOOCV error. In the worst case, all the 'm' support vectors in the higher dimensions will lie on the wrong side of the plane. & add "m" to the error.

$$\therefore \text{LOOCV error} = \frac{1}{n} \sum_{i=1}^{n} 1 \leq \frac{m}{n}$$

2 Question 2

2.1 SVM Dual Objective

2.1. SVM dual Objective $= \max_{\alpha} \sum_{j=1}^{n} \alpha_j - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i \alpha_i y_j \alpha_j \cdot k(x_i, x_j)$

$$\text{s.t. } \sum_{j=1}^{n} y_j \alpha_j = 0$$

$$0 \leq \alpha_j \leq C \quad \forall j$$

this is same as:

$$\underset{\alpha}{\text{minimize}} \ \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i \alpha_i y_j \alpha_j \cdot k(x_i, x_j) - \sum_{j=1}^{n} \alpha_j$$

Quadprog in matlab:

$$\min_{x} \frac{1}{2} x^T H x + f^T x \quad . \text{ s.t. } \begin{cases} A \cdot x \leq b \\ Aeq \cdot x = beq \\ lb \leq x \leq ub \end{cases}$$

Since we are using Linear Kernel

$H = (YY^T) .* (X^T X)$

$f = -1 \times \text{ones}(\text{size}(Y,1),1) \equiv \begin{bmatrix} -1 \\ -1 \\ \vdots \\ -1 \end{bmatrix}_{n \times 1}$

$A = [\ ]$

$b = [\ ]$

$Aeq = y' \quad (1 \times n)$

$beq = 0$

$lb = \text{zeros}(n,1)$

$ub = C * \text{ones}(n,1)$

2.4 $C = 0.1$

```
Validation accuracy: [90.735695]
Objective Value: [24.764818]
Support Vectors: [339]
Validation Confusion Matrix -
   152    32
     2   181
```

2.5 $C = 10$

```
Validation accuracy: [97.820163]
Objective Value: [112.146132]
Support Vectors: [123]
Validation Confusion Matrix -
   180     4
     4   179
```

2.6

For this challenge, I used the SVM code that I created for Q 2.2 and Q 2.3. Additionally, I used one-vs-all SVM classifier with Linear Kernel and for predicting the class for a data point, I assigned the class with the maximum confidence which can be calculated as $((X' * w) + b)$. I also tried Polynomial Kernel but it did not give me better results than a Linear Kernel. To find the best value for hyper-parameters, I did a search on the input space of hyper-parameters and evaluated the accuracy of classifier on validation set. I did not include the validation set in the training set or else it would give us incorrect answers. Then, based on the value of C found using cross-validation, I trained the model for which I got a Kaggle score (on 30% test data) of 0.50666 with $C = 0.0056$.

3 Question 3

3.4.1

AP: 0.638

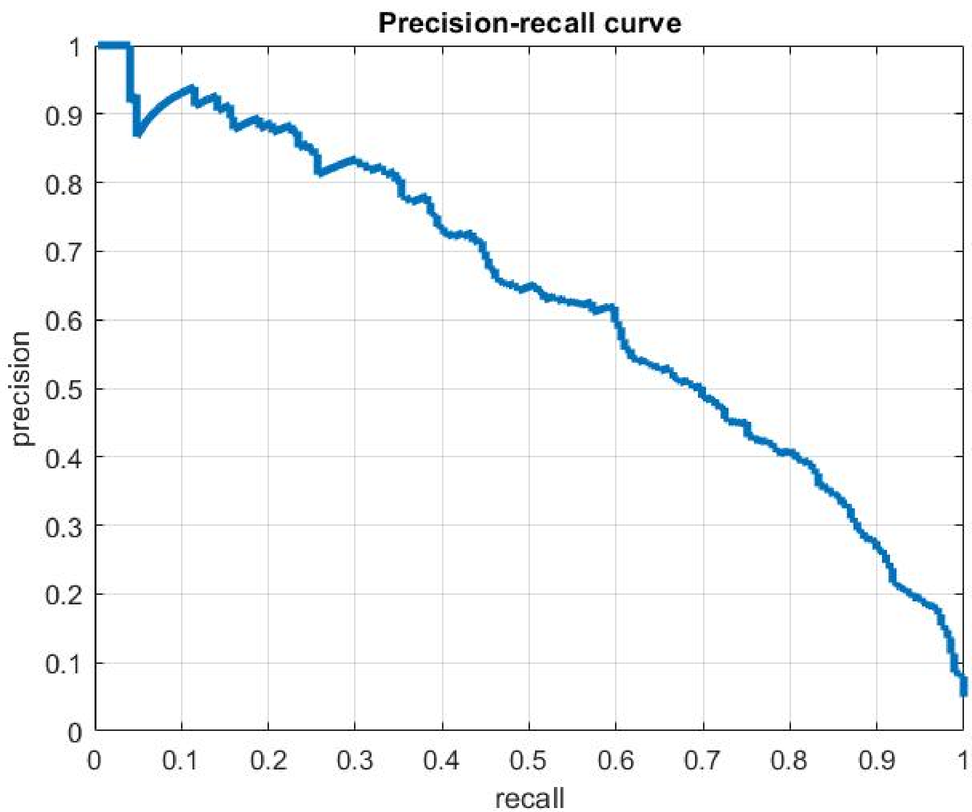


Figure 1: Precision recall curve for 3.4.1

3.4.3

NOTE: Please see the last page for the curves.
Objective values:

Iteration	**Objective Value**
1	0.6398
2	0.9156
3	0.9943
4	1.0102
5	1.0121
6	1.0122
7	1.0122
8	1.0122
9	1.0122
10	1.0122

AP Values:

Iteration	**AP**
1	0.8275
2	0.8620
3	0.8637
4	0.8582
5	0.8591
6	0.8587
7	0.8587
8	0.8587
9	0.8587
10	0.8587

3.4.4

AP: 0.8018

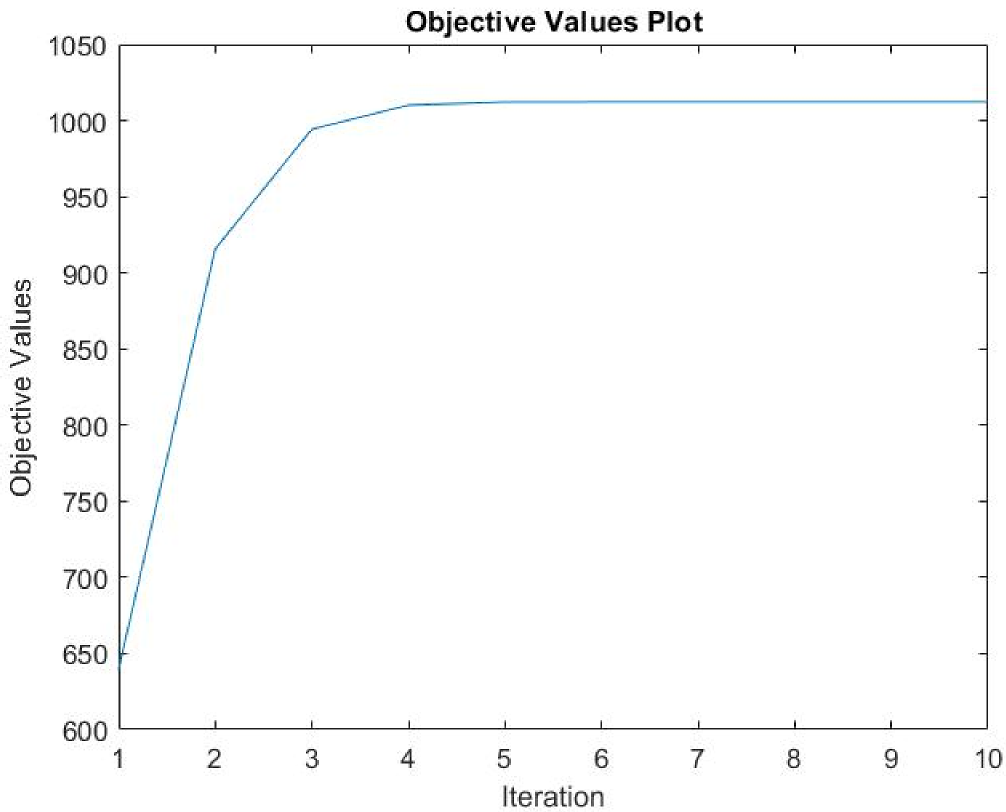


Figure 2: Objective values for 3.4.3

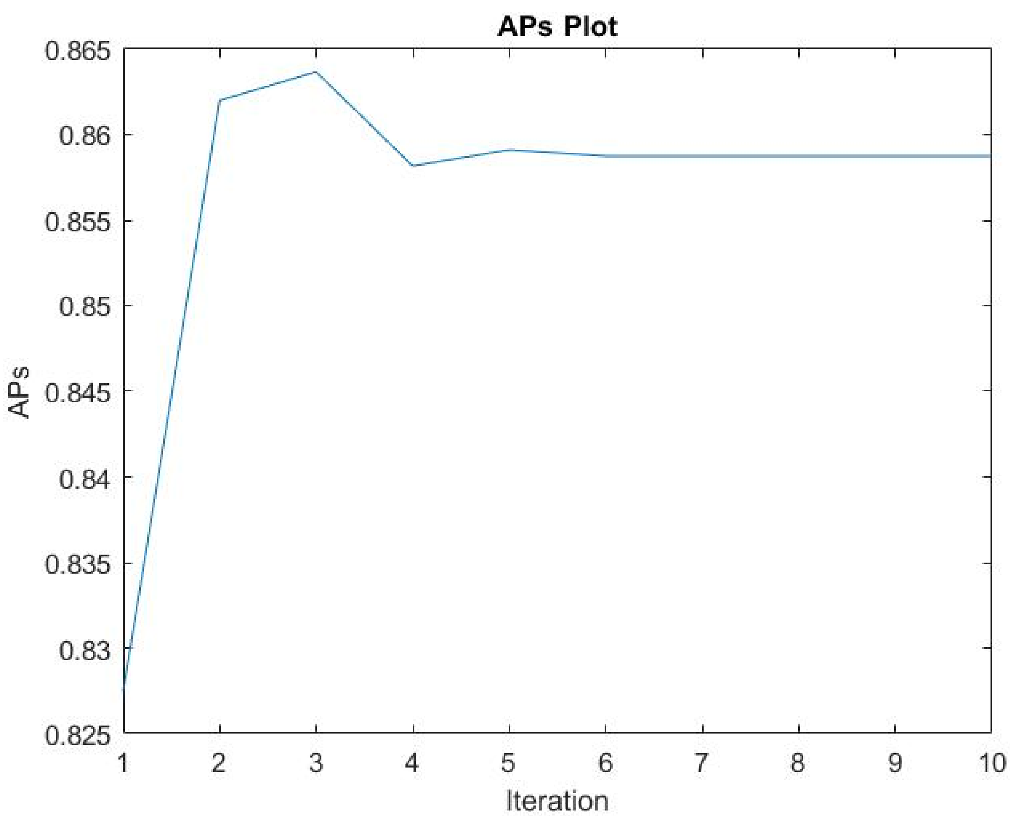


Figure 3: APs for 3.4.3