CSE512 Spring 2021 - Machine Learning - Homework 4

Your Name: Sagar Onkar Toshniwal

Solar ID: 113260061

NetID email address: sagaronkar.toshniwal@stonybrook.edu

Names of people whom you discussed the homework with: None but I have taken reference from hw4_utils function, stackoverflow, medium, towardsdatascience, etc.

Que 1]

1.1] Linear case

→ The main objective of SVM is to maximize the margin which is the distance between the closest point from the separating hyperplane.

$$\max_{\gamma, \bar{w}, \bar{b}} \gamma$$

$$\text{s.t.} \quad y^j (\bar{w} \cdot x^j + \bar{b}) \geq \gamma \quad \forall j$$

$\gamma \Rightarrow$ margin.

$$\gamma = \frac{\|\bar{w}\|}{\|w\|} = \frac{1}{\|w\|}$$

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

$$\text{s.t.} \quad y^j (w \cdot x^j + b) \geq 1 \quad \forall j$$

∴ The LOOCV error (for SVM):-

$$(LOOCV)_{error} = \frac{1}{n} \sum_{i=1}^{n} e_i \quad ; \quad$$ ~~error~~ e_i is the error when e_i th data point is removed.

Since the data is linearly separable there are two cases:-

1) <u>Removing</u> a <u>support</u> <u>vector</u> data point :-

If a point is support vector, then the removal of a support vector will change or shift to a new classifier plane. This leads to case where the point is misclassified.

h

error when point is correctly Classified = 0 ⇐ e_i
error when point is incorrectly Classified = 1 ⇐ e_i

Let us consider worst case scenario :-
number of Support Vectors is m. & All of them are incorrectly classified i.e. they are misclassified.

$$\text{LOOCV error} = \frac{1}{n} \sum_{i=1}^{m} e_i \qquad (\because e_i \text{ is } 1 \text{ for incorrectly classified})$$

$$= \frac{1}{n} \times m \qquad \because \text{ Since all are misclassified.}$$

$$\boxed{\text{LOOCV error} = \frac{m}{n}}$$

Here $n \Rightarrow$ no. of data points.

2) Removing a Non-Support Vector :-

If a point is non-support vector, then the decision boundary doesn't change. & the data points won't be misclassified.

$\therefore e_i = 0$ in this scenario.

$$\boxed{\text{LOOCV error} = 0}$$

Que 1.2]

1.2] General Case :-

→ If we use a General Kernel that Separates the data in high dimensional feature space, then the bound on the training error will still hold.

Case 1] Removing a Support Vector point :-

Since the data is linearly Separable, Removing a Support vector will change the decision boundary, adding 1 (for every misclassified point) to the LOOCV error

In worst case for all m Support vector points which lie on the wrong side will add m errors to the LOOCV error.

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

$$= \frac{m}{n} \quad \ldots \text{worst Case scenario.}$$

Case 2] Removing a Non-Support Vector point :-

Since the data is linearly separable, removing a Non-Support vector won't Change the decision boundary & thus won't add any error.

$$\text{LOOCV error} = 0$$

Que 2.1]

Accuracy and Confusion Matrix on Train and Test Data

```
[32]: print (" Accuracy on test : ", acc)
      print ("Confusion Matrix of test : \n ", confusion_matrix(ytest,ypred))

 Accuracy on test :  0.8700939745715865
Confusion Matrix of test :
  [[11643   792]
 [ 1323  2523]]

[33]: ypred_train = m1.predict(Xtrain)
      print (" Accuracy on train : ",accuracy_score(ytrain,ypred_train))
      print ("Confusion Matrix of train :\n ", confusion_matrix(ytrain,ypred_train))

 Accuracy on train :  0.9043334049937041
Confusion Matrix of train :
  [[23667  1053]
 [ 2062  5779]]
```

Que 2.2]

Best Cross validation Accuracy is 88.2678 % for eta(learning_rate) of 0.2.

I have tried learning rate – (0.3, 0.2, 0.1, 0.01, 0.001) and

got the following Accuracy - [0.8808353808353808, 0.8826781326781327, 0.8808353808353808, 0.8639434889434889, 0.8627149877149877]

Accuracy on test data – 87.279%

```
[30]: for i in range (5) :
          print ("learning rate and accuracy : ", (m[i],acc_hp[i]))

learning rate and accuracy :  (0.3, 0.8808353808353808)
learning rate and accuracy :  (0.2, 0.8826781326781327)
learning rate and accuracy :  (0.1, 0.8808353808353808)
learning rate and accuracy :  (0.01, 0.8639434889434889)
learning rate and accuracy :  (0.001, 0.8627149877149877)

[34]: mf = XGBClassifier(use_label_encoder =False, eta = 0.2 )
      mf.fit(Xtrain,ytrain)
      ypred2 = mf.predict(Xtest)

      acc2 = accuracy_score(ytest,ypred2)
      acc2

[34]: 0.8727965112708065

[35]: print (confusion_matrix(ytest,ypred2))

[[11679   756]
 [ 1315  2531]]
```

Que 3.3.1]

mAP = 6.69e-05

AP = 0.68

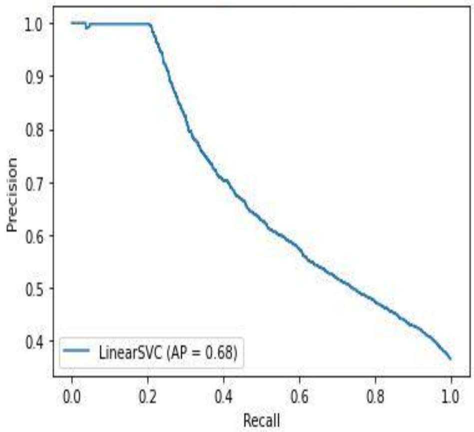
Precision recall curve given below

```
[ ] ap =hw.compute_mAP(result_file= 'result.npy',dataset='validation')

  2%|█          | 23/1000 [00:00<00:04, 222.71it/s]Loading annotations and predictions
100%|██████████| 1000/1000 [00:41<00:00, 24.31it/s]
mAP: 6.695950287394226e-05
```

```
from sklearn.metrics import plot_precision_recall_curve
import matplotlib.pyplot as plt
plot_precision_recall_curve(clf,D_val,lb_val)
```

```
<sklearn.metrics._plot.precision_recall_curve.PrecisionRecallDisplay at 0x7f5265daea10>
```



Que 3.3.3]

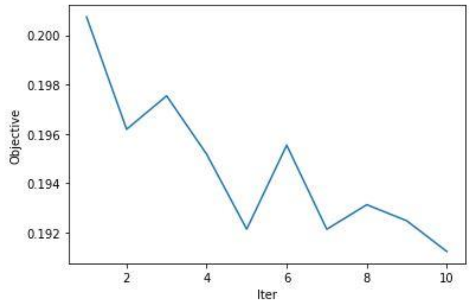
Values of Objective function for 10 iterations are - [0.20073592, 0.1961869, 0.19754075, 0.19517744, 0.19214153, 0.19554298, 0.19213783, 0.19313292, 0.19247874, 0.19123771]

Values of Average Precison for 10 iterations are - [0.67794486, 0.40120352, 0.45223088, 0.64249706, 0.57081924, 0.44464972, 0.5360358, 0.48870624, 0.48647239, 0.58637103]

```
Objective :  [0.20073592 0.1961869  0.19754075 0.19517744 0.19214153 0.19554298
 0.19213783 0.19313292 0.19247874 0.19123771]
Average Precision :  [0.67794486 0.40120352 0.45223088 0.64249706 0.57081924 0.44464972
 0.5360358  0.48870624 0.48647239 0.58637103]
```

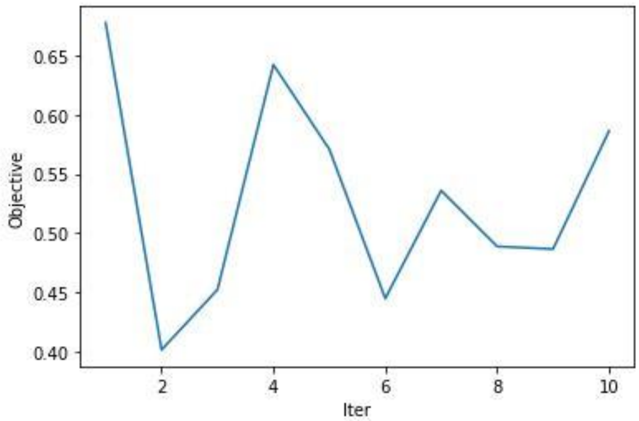
Objective vs Iteration plot :

```python
x = [1,2,3,4,5,6,7,8,9,10]
plt.plot (x, obj_array)
plt.xlabel ("Iter")
plt.ylabel ("Objective")
plt.show()
```



Average Precision vs Iteration Plot :

```python
plt.plot (x, ap_array)
plt.xlabel ("Iter")
plt.ylabel ("Objective")
plt.show()
```



(Minor error, by mistake I wrote y-label as objective. It should be Average Precision. While plotting I plotted correctly but while giving label name I forgot. So kindly ignore this minor. Also to cross check I have printed accuracy table for every iteration to remove any confusion.)

Que 3.3.4]

AP from spreadsheet :- 0.0000678

17	113260061	0.0000678