Assignment 3

Homework assignments will be done individually: each student must hand in their own answers. Use of partial or entire solutions obtained from others or online is strictly prohibited. Electronic submission on Canvas is mandatory.

Table 1: Data set for question 2

data	x_{i1}	x_{i2}	y_i	α_i
$\mathbf{x}_1$	4	2.9	1	0.414
$\mathbf{x}_2$	4	4	1	0
$\mathbf{x}_3$	1	2.5	-1	0
$\mathbf{x}_4$	2.5	1	-1	0.018
$\mathbf{x}_5$	4.9	4.5	1	0
$\mathbf{x}_6$	1.9	1.9	-1	0
$\mathbf{x}_7$	3.5	4	1	0.018
$\mathbf{x}_8$	0.5	1.5	-1	0
$\mathbf{x}_9$	2	2.1	-1	0.414
$\mathbf{x}_{10}$	4.5	2.5	1	0

1. **Support Vector Machines** (20 points) Given 10 points in Table 1, along with their classes and their Lagranian multipliers (α_i), answer the following questions:

   (a) (7 pts) What is the equation of the SVM hyperplane $h(x)$? Draw the hyperplane with the 10 points.

   (b) (8 pts)What is the distance of x_6 from the hyperplane? Is it within the margin of the classifier?

   (c) (5 pts)Classify the point $z = (3, 3)^\top$ using $h(x)$ from above.

   Solution:

   a) The Equation of a SVM hyper-plane is given by the following equation,

   $h(x) = w^T x + b$ ...equation(1)

   The points that have $\alpha = 0$ are not support vectors. Points that have $\alpha > 0$ are support considered as support vectors. In the above inputs, we have data points x_1, x_4, x_7 and x_8 as support vectors. These points helps us to identify the equation of hyper-plane.

   The weight vector w is computed by the following equation,

   $w = \sum_{i,\alpha_i>0} \alpha_i y_i x_i$

   $$= \{0.414 \times 1 \times \begin{bmatrix} 4 \\ 2.9 \end{bmatrix} + 0.018 \times -1 \times \begin{bmatrix} 2.5 \\ 1 \end{bmatrix} + 0.018 \times 1 \times \begin{bmatrix} 3.5 \\ 4 \end{bmatrix} + 0.414 \times -1 \times \begin{bmatrix} 2 \\ 2.1 \end{bmatrix}\}$$

   $$= \begin{bmatrix} 4 \times 0.414 \\ 2.9 \times 0.414 \end{bmatrix} - \begin{bmatrix} 2.5 \times 0.018 \\ 1 \times 0.018 \end{bmatrix} + \begin{bmatrix} 3.5 \times 0.018 \\ 4 \times 0.018 \end{bmatrix} - \begin{bmatrix} 2 \times 0.414 \\ 2.1 \times 0.414 \end{bmatrix}$$

   $$= \begin{bmatrix} 1.656 \\ 1.2006 \end{bmatrix} - \begin{bmatrix} 0.045 \\ 0.018 \end{bmatrix} + \begin{bmatrix} 0.065 \\ 0.072 \end{bmatrix} - \begin{bmatrix} 0.828 \\ 0.8694 \end{bmatrix}$$

$w = \begin{bmatrix} 0.846 \\ 0.3852 \end{bmatrix}$

The bias in the equation(1) is the average bias for each support vector. These vectors have $\xi_i = 0$ and have $0 < \alpha_i < c$

We have vectors, $x1, x_4, x_7$ and x_9 that satisfy these conditions.

The bias for these vectors is give by the following equation,

$b_1 = y_1 - w^T x_1$

$= 1 - \begin{bmatrix} 0.846 & 0.3852 \end{bmatrix} \times \begin{bmatrix} 4 \\ 2.9 \end{bmatrix}$

$= 1 - \{0.846 \times 4 + 0.3852 \times 2.9\}$

$= 1 - 4.50108$

$b_1 = -3.50108$

$b_4 = y_4 - w^T x_4$

$= -1 - \begin{bmatrix} 0.846 & 0.3852 \end{bmatrix} \times \begin{bmatrix} 2.5 \\ 1 \end{bmatrix}$

$= -1 - \{0.846 \times 2.5 + 0.3852 \times 1\}$

$= -1 - 2.5002$

$b_4 = -3.5002$

$b_7 = y_7 - w^T x_7$

$= 1 - \begin{bmatrix} 0.846 & 0.3852 \end{bmatrix} \times \begin{bmatrix} 3.5 \\ 4 \end{bmatrix}$

$= 1 - \{0.846 \times 3.5 + 0.3852 \times 4\}$

$= 1 - 4.50108$

$b_7 = -3.50108$

$b_9 = y_9 - w^T x_9$

$= -1 - \begin{bmatrix} 0.846 & 0.3852 \end{bmatrix} \times \begin{bmatrix} 2 \\ 2.1 \end{bmatrix}$

$= -1 - \{0.846 \times 2 + 0.3852 \times 2.1\}$

$= -1 - 2.50092$

$b_9 = -3.50092$

$b = \frac{b_1 + b_4 + b_7 + b_9}{4}$

$b = \frac{(-3.50108 - 3.5002 - 3.50108 - 3.50092)}{4}$

$b = -3.50082$

Equation of hyper-plane,

$h(x) = \begin{bmatrix} 0.846 \\ 0.3852 \end{bmatrix}^T x - 3.50082$

The diagram for the hyper-plane is included in the pdf Diagram_Q1.

b) Distance of point $x_6 \begin{bmatrix} 1.9 \\ 1.9 \end{bmatrix}$ from hyper-plane h(x) is given as follows,

The slack variable,

$\xi_i = 1 - y_i(w^T x_i + b)$

$\xi_6 = 1 - y_6(w^T x_6 + b)$

$= 1 - (-1) \times \begin{bmatrix} 0.846 & 0.3852 \end{bmatrix} \times \begin{bmatrix} 1.9 \\ 1.9 \end{bmatrix} - 3.50082$

$= 1 + 2.33928 - 3.50082$

$= 1 + (-1.16154)$

$\xi_6 = -0.16154$

The signed distance of point from hyper-plane is given by the following equation,

$D_i = \frac{w^T \times x_i + b}{||w||}$

$D_6 = \frac{w^T \times x_6 + b}{||w||}$

$= \frac{\begin{bmatrix} 0.846 \\ 0.3852 \end{bmatrix}^T \times \begin{bmatrix} 1.9 \\ 1.9 \end{bmatrix} - 3.5008}{\sqrt{w_1^2 + _w 2^2}}$

$= \frac{(0.846 \times 1.9 + 0.3852 \times 1.9 - 3.5008)}{\sqrt{0.846^2 + 0.3852^2}}$

$= \frac{1.6074 + 0.73188 - 3.5008}{\sqrt{0.86409504}}$

$= \frac{-1.16152}{0.9295671251}$

$D_6 = -1.249527838$

The value of $\xi_6 < 0$, the point is infeasible. It is within the margin of the classifier and will be considered as 0 during the optimization problem.

c) We have the equation of hyper-plane as,

$h(x) = \begin{bmatrix} 0.846 \\ 0.3852 \end{bmatrix}^T x - 3.50082$

Therefore,

$h(z) = \begin{bmatrix} 0.846 \\ 0.3852 \end{bmatrix}^T z - 3.50082$

We have $z = \begin{bmatrix} 3 & 3 \end{bmatrix}^T$

Therefore,

$h(z) = \begin{bmatrix} 0.846 & 0.3852 \end{bmatrix} \times \begin{bmatrix} 3 \\ 3 \end{bmatrix} - 3.50082$

$= \{0.846 \times 3 + 0.3852 \times 3 - 3.50082\}$

$= 3.6936 - 3.50082$

$h(z) = 0.19278$

Since, we have $h(z) > 0$, therefore it will be classified into class $y_z = +1$

2. **Support Vector Machines** (20 points) The SVM loss function with slack variables can be viewed as:

$$\min_{\mathbf{w},b} \frac{||\mathbf{w}||^2}{2} + \gamma \sum_{i=1}^{N} \underbrace{\max(0, 1 - y_i f(\mathbf{x}_i))}_{\text{Hinge loss}} \tag{1}$$

The hinge loss provides a way of dealing with datasets that are not separable.

(a) (8 pts)Argue that $l = \max(0, 1 - y\mathbf{w}^\top \mathbf{x})$ is convex as a function of $\mathbf{w}$

(b) (5 pts) Suppose that for some $\mathbf{w}$ we have a correct prediction of f with $\mathbf{x}_i$, i.e. $f(\mathbf{x}_i) = \text{sgn}(\mathbf{w}^\top \mathbf{x}_i)$. For binary classifications ($y_i = +1/-1$), what range of values can the hinge loss, l, take on this correctly classified example? Points which are classified correctly and which have non-zero hinge loss are referred to as margin mistakes.

(c) (7 pts) Let $M(\mathbf{w})$ be the number of mistakes made by $\mathbf{w}$ on our dataset (in terms of classification loss). Show that:

$$\frac{1}{n} M(\mathbf{w}) \le \frac{1}{n} \sum_{i=1}^{n} \max(0, 1 - y_i \mathbf{w}^\top \mathbf{x}_i)$$

In other words, the average hinge loss on our dataset is an upper bound on the average number of mistakes we make on our dataset.

Solution:

a)We can state that the maximum of two convex functions is convex.

To prove $l = \max(0, 1 - yw^T x)$ is convex, we need to prove that the 0 and $1 - yw^T x$ are covex function.

Convexity is defined by the following equation:

$\forall x_1, x_2 \in X, \forall t \in [0,1] : f(tx_1 + (1-t)x_2) \le tf(x_1) + (1-t)f(x_1)$

Let $f((x,y), w_1) = 1 - yw^T x$

f is said to be convex if the following equation prevails $tf((x,y),w_1)+(1-t)f((x,y),w_2) \ge f((x,y),tw_1+(1-t)w_2$

We have,

$tf((x,y),w_1) + (1-t)f((x,y),w_2) \ge t(1 - yw_1^T x) + (1-t)(1 - yw_2^T x)$

$tf((x,y),w_1) + (1-t)f((x,y),w_2) \ge t - tyw_1^T x + 1 - yw_2^T x - t - tyw_2^T x$

$tf((x,y),w_1) + (1-t)f((x,y),w_2) \ge 1 - yw_1^T x - yw_2^T x - tyw_2^T x$

$tf((x,y),w_1) + (1-t)f((x,y),w_2) \ge 1 - y(w_1^T x + w_2^T x + tw_2^T x)$

$tf((x,y),w_1) + (1-t)f((x,y),w_2) \ge 1 - y(w_1^T x + (1-t)w_2^T x)$

$tf((x,y),w_1) + (1-t)f((x,y),w_2) \ge f(x,y), tw_1 + (1-t)w_2$

Since, 0 and $1 - yw^T x$ both are convex so we can say that function l is convex.

b)The hyper-plane separates the correctly classified points.

There are two types of error,

$y_i = 1$ and $w^T x_i + b < 0$

$y_i = -1$ and $w^T x_i + b > 0$

$y_i w^T x_i$ is always positive given that a point is correctly classified.

For all miss classified points,

$y_i(w^T x_i) < 0$

This contributes to the loss as the points violates margin constraints.

When,

$y_i(w^T x_i) = 1$

This does not contributes to the loss as the points lie on the margins.

When $y_i(w^T x_i) = 0$, the value of $1 - y_i(w^T x_i)$ is maximum, equal to 1. This max argument also gives us positive values.

Thus we can say that the range of hinge loss is [0,1] for correctly classified points.

c) To prove:

$$\frac{1}{n} M(\mathbf{w}) \leq \frac{1}{n} \sum_{i=1}^{n} \max(0, 1 - y_i \mathbf{w}^\top \mathbf{x}_i)$$

Assume,

$r = y_i w^\top x_i$

Assuming a new function l such that $l(r) = 0$ for $r > 0$, and $l(r) = 1$ for $r < 0$.

Also , $h(r) = max\{0, 1 - r\}$

Here , $l(r) \leq h(r)$

If $y_i w^T x_i$ is negative then we can confirm an error has occurred and $l(y_i w^T x_i) = 1$

Therefore,

$M(\mathbf{w}) = \sum_i l(y_i w^T x_i) \leq \sum_i h(y_i w^T x_i) = \sum_{i=1}^{n} \max(0, 1 - y_i \mathbf{w}^\top \mathbf{x}_i)$

Therefor we can state that,

$\frac{1}{n} M(\mathbf{w}) \leq \frac{1}{n} \sum_{i=1}^{n} \max(0, 1 - y_i \mathbf{w}^\top \mathbf{x}_i)$

3. **Decision Trees** (20 points) Implement a Decision Tree model for the Titanic data set.

   - Explain how you preprocess the features.
   - Build a tree on the training data and evaluate the performance on the test data.
   - Compare Gini index and Information Gain.
   - Report your best accuracy on the test data set.
   - Give a brief description of your observations.

Solution:

Data Cleaning:

1) The attributes "SibSp" and "Parch" are used to check if the passenger was travelling alone or not. A new attribute "individual" is created based on this.

2) The attribute "Cabin" is converted into numerical value with the "Deck_Pos" as position of decks in the ship determine the survival.

3) The attribute "Age" is an important characteristic for survival. However this attribute contains a large number of distinct value, so this attribute is divided into age groups.

4) The attribute "Embarked" is converted into numeric value attribute.

5) The attribute "Sex" is converted into numeric value attribute.

6) The attribute "Fare" consists of various distinct value. The higher Fare determines the more importance of that person which determines that safety of such passenger is more guaranteed. The distinct value cannot be used as it is, therefore it is divided into groups.

7) The attributes "Name", "Ticket" and "PassengerId" are dropped as their values do not determine survival. There is no co-relation between these attributes and the target variable.

Observations:

1) The gini index for the attributes are greater than information gain.

2) The accuracy achieved is greater than any other algorithm so far, however there is trade-off as this can be due to over-fitting of the model.

3) The creation of the root node takes a lot of time as recursion is involved which requires lots of space along with the data-frames.

4) The categorical data needs to be converted to numerical data and high Cardinality numerical data also needs to be grouped as such data will lead to over-fitting.

4. **Boosting** (20 points) Implement AdaBoost for the Titanic data set. You can use package/tools to implement your decision tree classifiers. The fit function of DecisionTreeClassifier in sklearn has a parameter: sample weight, which you can use to weigh training examples differently during various rounds of AdaBoost.

   - Plot the train and test errors as a function of the number of rounds from 1 through 500.
   - Report your best accuracy on the test data set.
   - Give a brief description of your observations.

   Solution:

   Data Cleaning:

   1) The attributes "SibSp" and "Parch" are used to check if the passenger was travelling alone or not. A new attribute "individual" is created based on this.

   2) The attribute "Cabin" is converted into numerical value with the "Deck_Pos" as position of decks in the ship determine the survival.

   3) The attribute "Age" is an important characteristic for survival. However this attribute contains a large number of distinct value, so this attribute is divided into age groups.

   4) The attribute "Embarked" is converted into numeric value attribute.

   5) The attribute "Sex" is converted into numeric value attribute.

   6) The attribute "Fare" consists of various distinct value. The higher Fare determines the more importance of that person which determines that safety of such passenger is more guaranteed. The distinct value cannot be used as it is, therefore it is divided into groups.

   7) The attributes "Name", "Ticket" and "PassengerId" are dropped as their values do not determine survival. There is no co-relation between these attributes and the target variable.

   Observation

   1) The training error is more than testing error, which indicate over-fitting.

   2) The accuracy is very high which supports the above argument as well.

   3) The training error starts with a higher value and eventually goes to a lower value as the iterations progress in the Adaboost.

   4) The same behaviour is expected in the testing errors, however it remains constant after a certain value as the iterations progress in the Adaboost.

5. **Neural Networks** (20 points) Develop a Neural Network (NN) model to predict a handwritten digit images into 0 to 9. The pickled file represents a tuple of 3 lists : the training set, the validation set and the testing set. Each of the three lists is a pair formed from a list of images and a list of class labels for each of the images. An image is represented as numpy 1-dimensional array of 784 (28 x 28) float values between 0 and 1 (0 stands for black, 1 for white). The labels are numbers between 0 and 9 indicating which digit the image represents. The code block below shows how to load the dataset.

```
import cPickle, gzip, numpy

# Load the dataset
f = gzip.open('mnist.pkl.gz', 'rb')
train_set, valid_set, test_set = cPickle.load(f)
f.close()
```

- Plot the train, validation, and test errors as a function of the epoches.
- Report the best accuracy on the validation and test data sets.
- Apply early stopping using the validation set to avoid overfitting.
- Give a brief description of your observations.

Solution: