Introduction to ML (CS771), Autumn 2023
Indian Institute of Technology Kanpur
Homework Assignment Number 1

Student Name: Akash Shivaji Varude
Roll Number: 231110006
Date: September 15, 2023

QUESTION 1

We can find optimal values of $\boldsymbol{w}_c$ and $\boldsymbol{M}_c$,by finding the first-order derivative of our objective function and equating it with zero.

Our cost function is

$$(\hat{\boldsymbol{w}}_c, \hat{\boldsymbol{M}}_c) = arg \min_{\boldsymbol{w}_c, \boldsymbol{M}_c} \sum_{(\boldsymbol{x}_n : y_n = c)} \frac{1}{N_c} (\boldsymbol{x}_n - \boldsymbol{w}_c)^T \boldsymbol{M}_c (\boldsymbol{x}_n - \boldsymbol{w}_c) - \log |\boldsymbol{M}_c| \tag{1}$$

Partially differentiating above function with respect to $\boldsymbol{w}_c$ we get,

$$\frac{\partial L}{\partial \boldsymbol{w}_c} = -\frac{2}{N_c} \sum_{(\boldsymbol{x}_n : y_n = c)} \boldsymbol{M}_c (\boldsymbol{x}_n - \boldsymbol{w}_c)$$

Equating it with zero to get optimal value of $\boldsymbol{w}_c$

$$-\frac{2\boldsymbol{M}_c}{N_c} \sum_{(\boldsymbol{x}_n : y_n = c)} (\boldsymbol{x}_n - \boldsymbol{w}_c) = 0$$

$$\sum_{(\boldsymbol{x}_n : y_n = c)} (\boldsymbol{x}_n - \boldsymbol{w}_c) = 0$$

$$\sum_{(\boldsymbol{x}_n : y_n = c)} \boldsymbol{x}_n - \sum_{(\boldsymbol{x}_n : y_n = c)} \boldsymbol{w}_c = 0$$

$$\sum_{(\boldsymbol{x}_n : y_n = c)} \boldsymbol{x}_n = \sum_{(\boldsymbol{x}_n : y_n = c)} \boldsymbol{w}_c$$

$$\sum_{(\boldsymbol{x}_n : y_n = c)} \boldsymbol{x}_n = \boldsymbol{w}_c N_c$$

$$\boldsymbol{w}_c = \frac{1}{N_c} \sum_{(\boldsymbol{x}_n : y_n = c)} \boldsymbol{x}_n \tag{2}$$

Partially differentiating equation (1) with respect to $\boldsymbol{M}_c$ we get,

$$\frac{\partial L}{\partial \boldsymbol{M}_c} = \frac{1}{N_c} \sum_{(\boldsymbol{x}_n : y_n = c)} (\boldsymbol{x}_n - \boldsymbol{w}_c)^T (\boldsymbol{x}_n - \boldsymbol{w}_c) - \boldsymbol{M}_c^{-T}$$

Equating it with zero to get optimal value of $\boldsymbol{M}_c$

$$\frac{1}{Nc}\left(\sum_{(\boldsymbol{x}_n:y_n=c)}(\boldsymbol{x}_n-\boldsymbol{w}_c)^T(\boldsymbol{x}_n-\boldsymbol{w}_c)\right)-\boldsymbol{M}_c^{-T}=0$$

$$\frac{1}{Nc}\sum_{(\boldsymbol{x}_n:y_n=c)}(\boldsymbol{x}_n-\boldsymbol{w}_c)^T(\boldsymbol{x}_n-\boldsymbol{w}_c)=\boldsymbol{M}_c^{-T}$$

$$\boldsymbol{M}_c=N_c\sum_{(\boldsymbol{x}_n:y_n=c)}[(\boldsymbol{x}_n-\boldsymbol{w}_c)\cdot(\boldsymbol{x}_n-\boldsymbol{w}_c)^T]^{-1} \tag{3}$$

Special Case: (When $\boldsymbol{M}_c$ is an Identity matrix)
In this case our cost function becomes

$$L_{\hat{\boldsymbol{w}}_c}=\frac{1}{N_c}\sum_{(\boldsymbol{x}_n:y_n=c)}\|\boldsymbol{x}_n-\boldsymbol{w}_c\|^2 \tag{4}$$

This means we are minimizing squared distance between data points and class specific $\boldsymbol{w}_c$. Hence optimal value of $\boldsymbol{w}_c$ remains same i.e.

$$\boldsymbol{w}_c=\frac{1}{N_c}\sum_{(\boldsymbol{x}_n:y_n=c)}\boldsymbol{x}_n$$

and as $\boldsymbol{M}_c$ is Identity matrix, it is fixed and it doesn't require optimization

Introduction to ML (CS771), Autumn 2023
Indian Institute of Technology Kanpur
Homework Assignment Number 1

Student Name: Akash Shivaji Varude
Roll Number: 231110006
Date: September 15, 2023

QUESTION 2

Yes, the one-nearest-neighbor (1-NN) algorithm will be consistant in the noise-free setting where all training inputs are correctly labeled.

1NN assigns the label based on nearest neighbour in training data. In a noise-free environment, where all training inputs are accurately labeled, the nearest neighbor will share the same class label as the input being classified. Hence, as the training dataset size increases infinitely, 1-NN will consistently make zero errors, converging to the Bayes optimal error rate of zero.

Introduction to ML (CS771), Autumn 2023
Indian Institute of Technology Kanpur
Homework Assignment Number 1

Student Name: Akash Shivaji Varude
Roll Number: 231110006
Date: September 15, 2023

QUESTION 3

If the decision trees are used for regression problem, 'Reduction in Variance' would be great criteria for choosing a feature to split nodes. So variance can be measure of homogeneity. If a node is entirely homogeneous, then the variance is zero.

At first we will calculate variance of target variable in each subset after the split. We will choose that feature which results in the greatest reduction in variance (i.e., the difference between the variance before and after the split).

Formula for Reduction in Variance given by:

$$RV = V(D) - \left(\frac{|D|}{|D_L|} \cdot V(D_L) + \frac{|D|}{|D_R|} \cdot V(D_R) \right)$$

where,
$V(D)$: Variance of the target variable in the current group of data
$|D_L|$:Number of data points that end up in the left child node after the split
$|D_R|$:Number of data points that end up in the right child node after the split
$V(D_L)$:Variance of the target variable in the left child node after the split
$V(D_R)$:Variance of the target variable in the right child node after the split

So,we find the split that will maximize this reduction in variance as it will result in more homogeneous groups. Hence tree will be better at predicting values within each group.

Introduction to ML (CS771), Autumn 2023
Indian Institute of Technology Kanpur
Homework Assignment Number 1

Student Name: Akash Shivaji Varude
Roll Number: 231110006
Date: September 15, 2023

QUESTION 4

In unregularized linear regression, the prediction at a test input $\boldsymbol{x}_*$ is given by, $f(\boldsymbol{x}_*) = \hat{\boldsymbol{w}}^\top \boldsymbol{x}_*$ where $\hat{\boldsymbol{w}} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{y}$.

Substituting this expression of $\hat{\boldsymbol{w}}$ into the prediction formula we get,

$$f(\boldsymbol{x}_*) = \left(\left(\boldsymbol{X}^\top \boldsymbol{X} \right)^{-1} \boldsymbol{X}^\top \boldsymbol{y} \right)^\top \boldsymbol{x}_*$$

$$f(\boldsymbol{x}_*) = (\boldsymbol{X}^\top \boldsymbol{y})^\top ((\boldsymbol{X}^\top \boldsymbol{X})^{-1})^\top \boldsymbol{x}_*$$

$$f(\boldsymbol{x}_*) = (\boldsymbol{X}^\top \boldsymbol{y})^\top ((\boldsymbol{X}^\top \boldsymbol{X})^\top)^{-1} \boldsymbol{x}_*$$

as $\boldsymbol{X^T X}$ is symmetric we can write

$$f(\boldsymbol{x}_*) = (\boldsymbol{X}^\top \boldsymbol{y})^\top (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{x}_*$$

$$f(\boldsymbol{x}_*) = \boldsymbol{y}^\top \boldsymbol{X} (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{x}_*$$

$$f(\boldsymbol{x}_*) = \boldsymbol{y}^\top \hat{\boldsymbol{w}}_{\text{new}}$$

where $\hat{\boldsymbol{w}_{\text{new}}} = \boldsymbol{X} (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{x}_*$ and it is N x 1 vector.

as we can see, $\boldsymbol{y}^\top$ is 1 x N vector and $\hat{\boldsymbol{w}_{\text{new}}}$ is N x 1 vector, the expression $\boldsymbol{y}^\top \hat{\boldsymbol{w}_{\text{new}}}$ represents their inner product. Hence we can write it as

$$f(\boldsymbol{x}^*) = \sum_{n=1}^{N} w_n y_n$$

Discussion about weights:

In our problem, each weight w_n represents importance of each feature for modeling linear relationship between features and the training responses. So, value w_n is higher for more important feature than others. While in KNN, each weight w_n emphasize the importance of nearby training examples, giving more importance to nearby training example. hence value of w_n for training example nearer to x_* will be more than farther training example.

Student Name: Akash Shivaji Varude
Roll Number: 231110006
Date: September 15, 2023

QUESTION

5

New loss function is defined as:

$$L_{\text{new}} = \sum_{n=1}^{N} (y_n - \boldsymbol{w}^\top \widetilde{\boldsymbol{x}}_n)^2 \tag{1}$$

Expectation of this function with respect to the random mask vectors m_n is:

$$E[L_{\text{new}}] = E\left[\sum_{n=1}^{N} (y_n - \boldsymbol{w}^\top \widetilde{\boldsymbol{x}}_n)^2\right]$$

$$E[L_{\text{new}}] = \sum_{n=1}^{N} \left[E(y_n - \boldsymbol{w}^\top \widetilde{\boldsymbol{x}}_n)^2\right]$$

$$E[L_{\text{new}}] = \sum_{n=1}^{N} \left[E(y_n - \boldsymbol{w}^\top (\boldsymbol{x}_n \circ \boldsymbol{m}_n))^2\right]$$

$$E[L_{\text{new}}] = \sum_{n=1}^{N} \left[E\left(y_n^2 - 2y_n(\boldsymbol{w}^\top (\boldsymbol{x}_n \circ \boldsymbol{m}_n)) + (\boldsymbol{w}^\top (\boldsymbol{x}_n \circ \boldsymbol{m}_n))^2\right)\right]$$

$$E[L_{\text{new}}] = \sum_{n=1}^{N} \left[y_n^2 - 2y_n \boldsymbol{w}^\top E\left[(\boldsymbol{x}_n \circ \boldsymbol{m}_n)\right] + (\boldsymbol{w}^\top)^2 E\left[(\boldsymbol{x}_n \circ \boldsymbol{m}_n)^2\right]\right] \tag{2}$$

Now, Each element of $x_n \circ m_n$ is either 0 with probability $(1-p)$ or 1 with probability p. Hence every element of m_n also follows a Bernoulli distribution with probability p. Hence

$$E\left[(\boldsymbol{x}_n \circ \boldsymbol{m}_n)\right] = \boldsymbol{x}_n p \tag{3}$$

Also, for $E\left[(\boldsymbol{x}_n \circ \boldsymbol{m}_n)^2\right]$, elements of $\boldsymbol{x}_n \circ \boldsymbol{m}_n$ are either 0 or $\boldsymbol{x}_n$. Hence,

$$E\left[(\boldsymbol{x}_n \circ \boldsymbol{m}_n)^2\right] = E\left[(\boldsymbol{x}_n \circ \boldsymbol{m}_n)(\boldsymbol{x}_n \circ \boldsymbol{m}_n)\right]$$

$$E\left[(\boldsymbol{x}_n \circ \boldsymbol{m}_n)^2\right] = E\left[(\boldsymbol{x}_n^2 \circ \boldsymbol{m}_n^2)\right] \tag{4}$$

Hence substituting (3), (4) in equation (2) we get,

$$E[L_{\text{new}}] = \sum_{n=1}^{N} \left[y_n^2 - 2y_n \boldsymbol{w}^\top \boldsymbol{x}_n p + (\boldsymbol{w}^\top)^2 E\left[(\boldsymbol{x}_n^2 \circ \boldsymbol{m}_n^2) \right] \right]$$

$$E[L_{\text{new}}] = \sum_{n=1}^{N} \left[y_n^2 - 2py_n \boldsymbol{w}^\top \boldsymbol{x}_n + (\boldsymbol{w}^\top)^2 E\left[(\boldsymbol{x}_n^2 \circ \boldsymbol{m}_n^2) \right] \right] \tag{5}$$

So for converting it in regularized form, we can see, y_n^2 represents square of target values. So it is constant with respect to model weights w. $2py_n \boldsymbol{w}^\top \boldsymbol{x}_n$ this term represents the contribution of the data to expected loss. Because $\boldsymbol{w}^\top \boldsymbol{x_n}$ is predicted output and y_n is actual output. So, it's the mean squared error (MSE) term. While the term, $(\boldsymbol{w}^\top)^2 E\left[(\boldsymbol{x}_n^2 \circ \boldsymbol{m}_n^2) \right]$ penalizes large values of the model's weights $\boldsymbol{w}$ using L2 regularization.

Hence we can write equation (5) in regularized form as,

$$E[L_{\text{new}}] = \sum_{n=1}^{N} (y_n - \boldsymbol{w}^\top \boldsymbol{x}_n)^2 + \lambda \sum_{d=1}^{D} \boldsymbol{w}_d^2 \tag{6}$$

Introduction to ML (CS771), Autumn 2023
Indian Institute of Technology Kanpur
Homework Assignment Number 1

Student Name: Akash Shivaji Varude
Roll Number: 231110006
Date: September 15, 2023

QUESTION 6

Method 1(Convex):
Test Accuracy = 46.89320388349515
Method 2(Regress):
for λ=[0.01, 0.1, 1, 10, 20, 50, 100]
Corresponding Test Accuracies are
For λ =0.01 Test accuracy = 58.090614886731395 %
For λ =0.1 Test accuracy = 59.54692556634305 %
For λ =1 Test accuracy = 67.39482200647248 %
For λ =10 Test accuracy = 73.28478964401295 %
For λ =20 Test accuracy = 71.68284789644012 %
For λ =50 Test accuracy = 65.08090614886731 %
For λ =100 Test accuracy = 56.47249190938511 %