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

Student Name: Jhanvi A Zanje
Roll Number: 231110021
Date: September 15, 2023

QUESTION 1

Our cost function is

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

To find the optimal values of $\mathbf{w}_c$ and $\mathbf{M}_c$ for the given objective/loss function, we'll need to differentiate the objective with respect to $\mathbf{w}_c$ and $\mathbf{M}_c$, set the derivatives equal to zero, and solve for the optimal values. Let's start by finding the derivatives:

1) Partially differentiating the above function with respect to $\mathbf{w}_c$ we get,

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

Setting this derivative to zero:

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

Now, let's solve for $\mathbf{w}_c$:

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

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

Now, we can simplify this equation:

The left side, $\sum_{(x_n:y_n=c)}\mathbf{x}_n$, represents the sum of all data points $\mathbf{x}_n$ where y_n is equal to c. In other words, it's the sum of all data points in class c.

The right side, $\sum_{(x_n:y_n=c)}\mathbf{w}_c$, represents the sum of $\mathbf{w}_c$ repeated N_c times, where N_c is the number of data points in class c.

So, we have:

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

Now, to isolate $\mathbf{w}_c$, divide both sides by N_c:

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

This equation states that the optimal $\mathbf{w}_c$ is equal to the mean (average) of all data points in class c.

In summary, $\mathbf{w}_c$ represents a central reference point for class c, and it plays a key role in classification tasks by helping to define class boundaries and make predictions based on the characteristics of the data points in that class.

2) Partially differentiating the cost function with respect to $\mathbf{M}_c$ we get,

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

Equating it with zero to get the optimal value of $\mathbf{M}_c$:

$$\frac{1}{N_c} \sum_{(x_n : y_n = c)} (\mathbf{x}_n - \mathbf{w}_c)^T (\mathbf{x}_n - \mathbf{w}_c) - (\mathbf{M}_c^{-1})^{-T} = 0$$

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

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

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

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

3) Now for the special case where we take $\mathbf{M}_c = \mathbf{I}$:

$$L_{(\hat{w}_c, \hat{M}_c)} = \arg\min_{\mathbf{w}_c, \mathbf{M}_c} \sum_{(x_n : y_n = c)} \frac{1}{N_c} (\mathbf{x}_n - \mathbf{w}_c)^T \mathbf{I} (\mathbf{x}_n - \mathbf{w}_c) - \log |\mathbf{I}| \tag{3}$$

$$L_{(\hat{w}_c, \hat{M}_c)} = \arg\min_{\mathbf{w}_c, \mathbf{M}_c} \sum_{(x_n : y_n = c)} \frac{1}{N_c} (\mathbf{x}_n - \mathbf{w}_c)^T (\mathbf{x}_n - \mathbf{w}_c) - 0 \tag{4}$$

$$(\hat{w}_c, \mathbf{I}) = \arg\min \left(\frac{1}{n} \sum_{n=1}^{n} (\mathbf{x}_n - \mathbf{w}_c)^T (\mathbf{x}_n - \mathbf{w}_c) \right) \tag{5}$$

The optimization issue is reduced to finding $\mathbf{W}_c$ that minimizes the average squared Euclidean distance between each $\mathbf{x}_n$ and $\mathbf{w}_c$ when $\mathbf{M}$ is an identity matrix.

Student Name: Jhanvi A Zanje
Roll Number: 231110021
Date: September 15, 2023

QUESTION

2

Question Number 2

Yes, In this instance, 1NN will be consistent. Since there exists an endless amount of training data sets, each with a clean, accurate label. That means that anytime you receive test data, you can always discover training data that completes within a certain distance of it. The probability of finding such a point tends to be 1 as the number of training data increases. As a result, you can categorize them without making a mistake. In other words, your training set already contains the test data.

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

Student Name: Jhanvi A Zanje
Roll Number: 231110021
Date: September 15, 2023

QUESTION 3

Our objective when partitioning nodes in a decision tree is to reduce variability. However, conventional metrics like the Gini Index or the information gain/entropy gain are inappropriate for regression problems where the data is continuous. We choose node splits based on variance reduction in such cases. We use the formula below to get the variance at each node.

Formula for Variance: The variance of a dataset containing values $x_1, x_2, \ldots, x_n$ is defined as:

$$\mathrm{Var}(X) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

Here: - n is the number of data points. - x_i represents individual data points. - $\bar{x}$ is the mean (average) of the data points.

Formula for Variance Reduction: To calculate the reduction in variance for a potential split in a decision tree, follow these steps:

1. Compute the variance of the target variable for all data points within the current node before the split, denoted as $\mathrm{Var}_{\mathrm{par}}$:

$$\mathrm{Var}_{\mathrm{par}} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y})^2$$

Where: - n is the number of data points in the current node. - y_i represents the target values in the current node. - $\bar{y}$ is the mean (average) of the target values in the current node.

2. Calculate the weighted average of the variances of the two resulting child nodes for each potential split based on a feature and a split threshold. The weight assigned to each child node corresponds to the proportion of data points it contains. Denote these variances as $\mathrm{Var}_{\mathrm{left}}$ and $\mathrm{Var}_{\mathrm{right}}$ for the left and right child nodes, respectively.

3. Compute the reduction in variance as the difference between the variance of the current node before the split and the weighted average of the variances of the child nodes after the split:

$$\text{Reduction in Variance} = \mathrm{Var}_{\mathrm{par}} - \left(\frac{n_{\mathrm{left}}}{n}\cdot \mathrm{Var}_{\mathrm{left}} + \frac{n_{\mathrm{right}}}{n}\cdot \mathrm{Var}_{\mathrm{right}}\right)$$

Where: - n_{left} is the number of data points in the left child node. - n_{right} is the number of data points in the right child node.

The split that optimizes the variance reduction is chosen via the decision tree algorithm. The variation of the target variable within the resulting child nodes is significantly reduced as a result of this split.

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

Student Name: Jhanvi A Zanje
Roll Number: 231110021
Date: September 15, 2023

QUESTION 4

Here: $\hat{w} = (\mathbf{X^T X})^{-1}\mathbf{X^T y}$
Predicting for test input x_* can be expressed as $\mathbf{y}_* = \mathbf{\hat{w}^T x}_* = \mathbf{x}_*^\mathbf{T}\mathbf{\hat{w}}$.
Hence,

$$\mathbf{y}_* = \mathbf{x}_*^\mathbf{T}(\mathbf{X^T X})^{-\mathbf{1}}\mathbf{X^T y}$$

$$\Rightarrow y_* = \mathbf{Wy}$$

where $\mathbf{W} = \mathbf{x}_*^\mathbf{T}(\mathbf{X^T X})^{-\mathbf{1}}\mathbf{X^T}$
here,
$\mathbf{W} = (w_1, w_2, \ldots, w_N)$ is a $1 \times N$ matrix.
We can represent $\mathbf{y}$ as $(y_1, y_2, \ldots, y_N)^T$.
Therefore,

$$\mathbf{y}_* = \mathbf{Wy} = \sum_{n=1}^{N} w_n y_n$$

w_n corresponds to the nth index of the $1 \times N$ matrix W.
X is a matrix with rows representing the N training vectors x_n.
w_n can be expressed as:

$$w_n = \mathbf{x}_*^\mathbf{T}(\mathbf{X^T X})^{-\mathbf{1}}\mathbf{x_n}$$

Since the term $X^T X$ appears in the expression for w_n, it follows that w_n depends on the input x_* and all of the training data from x_1 to x_n. This contrasts with the weighted K-nearest neighbors (KNN) method, where each weight is completely dependent on x_* and x_n.

Additionally, x_* is in the numerator in this case, whereas it is in the denominator in KNN. In contrast to KNN, where it is stated as a sum in the denominator, w_n is expressed here as a product involving x_*. This is another difference.

Student Name: Jhanvi A Zanje
Roll Number: 231110021
Date: September 15, 2023

QUESTION 5

The new loss function is defined as:

$$\sum_{n=1}^{N}(y_n - w^T\tilde{x})^2$$

Here, $\tilde{x}_n = x_n \cdot m_n$ and m_n is a binary mask vector with $m_{nd} \sim \text{Bernoulli}(p)$.
Now, let's compute the expected value of this new loss function:

$$E[P] = E\left[\sum_{n=1}^{N}(y_n - w^T\tilde{x})^2\right]$$

We can utilize the linearity of expectation:

$$E[P] = \sum_{n=1}^{N} E\left[(y_n - w^T\tilde{x})^2\right]$$

$$E\left[(y_n - w^T\tilde{x})^2\right] = E\left[(y_n - w^T x_n m_n)^2\right]$$

$$E\left[-2y_n w^T x_n m_n\right] = -2py_n w^T x_n$$

$$E\left[(y_n - w^T x_n m_n)^2\right] = E\left[y_n^2 - 2py_n w^T x_n + (w^T x_n m_n)^2\right]$$

$$E[P] = \sum_{n=1}^{N}\left(E[y_n^2] - 2py_n w^T x_n + p(w^T x_n)^2\right)$$

Now, consider the term $E[(w^T x_n m_n)^2]$. Since m_{nd} is binary, we have:

$$E\left[(w^T x_n m_n)^2\right] = p(w^T x_n)^2$$

Substitute this back into the expression for $E[P]$:

$$E[P] = \sum_{n=1}^{N}\left(E[y_n^2] - 2py_n w^T x_n + p(w^T x_n)^2\right)$$

This expression is equivalent to minimizing the following regularized loss function:

$$L(w) = \sum_{n=1}^{N}\left(E[y_n^2] - 2py_n w^T x_n + p(w^T x_n)^2\right)$$

Our Regularized loss function will be:

$$L(w) = \frac{1}{n}\sum_{n=1}^{N}(y_n - w^T\tilde{x})^2 + \lambda w^T w$$

This is comparable to the L2 regularization method, where the loss function is modified by the addition of a penalty term whose square is proportionate to the weights.

Term $p(w^T x_n)^2$ in the loss function, which corresponds to L2 regularization Therefore, by minimizing the anticipated value of the new loss function, similar to L2 regularization, you are essentially determining the weights w that strike a balance between fitting the data (the first two terms) and regulating the amount of the weights (the third term).

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

Student Name: Jhanvi A Zanje
Roll Number: 231110021
Date: September 15, 2023

QUESTION 6

Method 1: The model's accuracy on the provided dataset is reported as 46.89% (46.893203883495144%).
Method 2: The maximum accuracy observed is 73.28% (73.28478964401295%) for $\lambda = 10$.
Here are the accuracy values corresponding to different λ settings:

λ	Accuracy (in %)
0.01	58.0906148867314%
0.1	59.54692556634305%
1	67.39482200647249%
10	73.28478964401295%
20	71.68284789644013%
50	65.08090614886731%
100	56.47249190938511%