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

Student Name: Yashvir Singh Nathawat
Roll Number: 231110059
Date: September 15, 2023

QUESTION 1

Consider a classification model where we are given training data $\{x_n, y_n\}_{n=1}^N$ from K classes. Each input $x_n \in \mathbb{R}^D$ and each class c is defined by two parameters, $w_c \in \mathbb{R}^D$ and a $D \times D$ positive definite (PD) matrix M_c, $c = 1, 2, \ldots, K$.

Given Objective function:

$$(\hat{w}_c, \hat{M}_c) = \arg\min_{w_c, M_c} \sum_{xn:yn=c} \frac{1}{N_c} \left((xn - w_c)^T M_c (xn - w_c) - \log |M_c| \right)$$

We are giving a minimization problem with w_c and M_c and to find minima we are equating first order derivative equals to 0.

Derivation with respect to w_c:

Take the derivative of the objective with respect to w_c:

$$\frac{\partial}{\partial w_c} \left(\sum_{x_n:yn=c} \frac{1}{N_c} \left((x_n - w_c)^T M_c (x_n - w_c) - \log |M_c| \right) \right) = 0$$

Now, let's simplify this expression:

$$\sum_{x_n:y_n=c} \frac{1}{N_c} \left(-2 M_c (x_n - w_c) \right) = 0$$

Multiply both sides by $\frac{2}{N_c}$ to get:

$$\sum_{x_n:yn=c} M_c (x_n - w_c) = 0$$

Now, sum over all training examples x_n for class c:

$$M_c \left(\sum_{x_n:y_n=c} (x_n - w_c) \right) = 0$$

$$\sum_{x_n:y_n=c} (x_n) - N_c (w_c) = 0$$

Rearranging the values to find the optimal w_c:

$$\hat{w}_c = \frac{1}{N_c} \sum_{xn:yn=c} x_n$$

This implies that the optimal value of w_c is the mean of all training examples xn for class c.

Optimizing with respect to M_c:

Take the derivative of the objective with respect to M_c. First, let's find the derivative of the log determinant term $\log|M_c|$:

$$\frac{\partial}{\partial M_c}\left(\log|M_c|\right) = \frac{1}{|M_c|}\text{adj}(M_c)$$

Now, let's find the derivative of the entire objective function with respect to M_c:

$$\frac{\partial}{\partial M_c}\left(\sum_{x_n:y_n=c}\frac{1}{N_c}\left((x_n-w_c)^T M_c(x_n-w_c)-\log|M_c|\right)\right)=0$$

This simplifies to:

$$\sum_{x_n:y_n=c}\left(\frac{1}{N_c}(x_n-w_c)(x_n-w_c)^T-\frac{1}{|M_c|}\text{adj}(M_c)\right)=0$$

Rearranging the equation to get:

$$\sum_{x_n:y_n=c}\left(\frac{1}{N_c}(x_n-w_c)(x_n-w_c)^T\right)=\frac{1}{|M_c|}\text{adj}(M_c)$$

Now as we know:

$$\text{adj}(A)=|A|\cdot A^{-1}$$

Therefor we get:

$$\sum_{x_n:y_n=c}\left(\frac{1}{N_c}(x_n-w_c)(x_n-w_c)^T\right)=|M_c|^{-1}$$

Now, to get the optimal $\hat{M}_c$ multiplying both sides by $\hat{M}_c$ and multiplying both sides by $\left(\frac{1}{N_c}\sum_{x_n:y_n=c}\left((x_n-w_c)(x_n-w_c)^T\right)\right)^{-1}$:

$$\hat{M}_c=\left(\frac{1}{N_c}\sum_{x_n:y_n=c}\left((x_n-w_c)(x_n-w_c)^T\right)\right)^{-1}$$

Special Case: M_c is an Identity Matrix (I):

When the covariance matrix M_c is an identity matrix ($M_c = I$).

In this case, the optimization problem simplifies as follows:

$$(w_c, I)=\arg\min_{w_c}\sum_{x_n:y_n=c}\frac{1}{N_c}||x_n-w_c||^2-\log|I|$$

Since the determinant of the identity matrix is always 1 ($|I| = 1$), the $\log|I|$ term becomes zero.

$$(w_c)=\arg\min_{w_c}\sum_{x_n:y_n=c}\frac{1}{N_c}||x_n-w_c||^2$$

Above equation is just standard mean square error of input points x_n and parameter w_c for all data points belonging to class c.

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

Student Name: Yashvir Singh Nathawat
Roll Number: 231110059
Date: September 15, 2023

QUESTION 2

Yes, the one-nearest-neighbor algorithm is consistent. Let us consider the statement by Cover and Hart (1967):

"As $N \to \infty$, the 1-NN error is no more than twice the error of the Bayes Optimal classifier. Similar guarantees hold for $k > 1$." The Bayes optimal error rate is given by:

$$\epsilon_{\text{BayesOpt}} = 1 - \max_i P(y_i|x)$$

Let us consider:
- N: The number of data points in the dataset.
- x_q: The test point for which we want to make a prediction.
- x_{NN}: The nearest neighbor of x_q in the dataset.
- y_i: The true label or class of a data point.
- $P(y_i|x_q)$: The conditional probability that y_i is the true label of x_q.
- $P(y_i|x_{\text{NN}})$: The conditional probability that y_i is the true label of x_n.

When $N \to \infty$, the distance between x_q and x_{NN} tends to 0 and they will likely overlap.

$$P(y_i|x_q) = P(y_i|x_{NN})$$

Now, we define ϵ_{NN} as the probability of wrongly classifying the test point x_q:

$$\epsilon_{\text{NN}} = P(y_i|x_q)(1 - P(y_i|x_{\text{NN}})) + P(y_i|x_{\text{NN}})(1 - P(y_i|x_q))$$

$$\epsilon_{\text{NN}} \leq (1 - P(y_i|x_{\text{NN}})) + (1 - P(y_i|x_q)) \leq 2(1 - P(y_i|x_q)) \leq 2\epsilon_{\text{BayesOpt}}$$

where $\epsilon_{\text{BayesOpt}}$ signifies the error of the Bayes Optimal classifier also the inequality follows from the observations that $P(y_i|x_q) \leq 1$, $P(y_i|x_{\text{NN}}) \leq 1$.

A classification algorithm is said to be consistent if, whenever it has access to infinite amounts of training data, its error rate approaches the optimal error rate (a.k.a. Bayes optimal).

$$\epsilon_{\text{BayesOpt}} \leq \epsilon_{\text{NN}} \leq 2\epsilon_{\text{BayesOpt}}$$

Given the Bayes optimal error rate is zero ($\epsilon_{\text{BayesOpt}}$=0) and the noise-free setting (i.e., every training input is labeled correctly), the one-nearest-neighbor algorithm is **CONSISTENT!** as its error rate approaches the optimal error rate (a.k.a. Bayes optimal).

$$\epsilon_{\text{NN}} \to 0$$

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

Student Name: Yashvir Singh Nathawat
Roll Number: 231110059
Date: September 15, 2023

QUESTION 3

In regression tasks, where the labels are real-valued, a good criterion to choose a feature for splitting is to maximize the **Standard Deviation Reduction (SDR)**.

1. **Standard Deviation (S) for Tree Building (Branching)**

   The formula to calculate the standard deviation for a sample of data is:

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

$$S = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n}}$$

   Where:

n : Number of examples of parent node.

x_i : target value of of ith example in parent node.

$\bar{x}$: Mean of the target values of parent node.

2. **Standard deviation for two attributes (target and predictor)**

   For target feature (T) and value (X) of selected feature, we are computing standard deviation of the combined variables S(T,X)

$$S(T, X) = \sum_{i=1}^{k} \left(\frac{n_i}{n} \times SD_i \right)$$

   Where:

k : Number of child nodes after the split

n_i : Number of data points in the i-th child node

n : Total number of data points in the parent node

SD_i : Standard deviation of the target variable in the i-th child node

3. **Splitting Criteria**

   The attribute with the largest standard deviation reduction is chosen for the decision node.

   A higher SDR value suggests that the split results in more homogeneous child nodes, making it a suitable criterion for selecting the best attribute to split on in decision tree regression when the goal is to minimize the variability of the target variable.

$$SDR = SD_{\text{parent}} - S(T, X)$$

   Where:

SD_{parent} : Standard deviation of the target variable in the parent node

   The dataset is divided based on the values of the selected attribute. This process is run recursively on the non-leaf branches, until all data is processed.

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

Student Name: Yashvir Singh Nathawat
Roll Number: 231110059
Date: September 15, 2023

QUESTION 4

In the unregularized linear regression model, the prediction for a test input x_* is given by:

$$y_* = \hat{w}^T x_* = x_*^T \hat{w}$$

where $\hat{w} = (X^T X)^{-1} X^T y$ is the solution.
Now,let's put the value of w in prediction rule:

$$y_* = x_*^T (X^T X)^{-1} X^T y$$

It can be written as:

$$y_* = Wy$$

where $\mathbf{W} = [w_1, w_2, w_3, \ldots, w_N]$ is row vector with N elements and also we can say $\mathbf{y} = [y_1, y_2, y_3, \ldots, y_n]^T$

The equation $y_* = Wy$ can be represented in summation form as:

$$y^* = \sum_{i=1}^{N} w_i \cdot y_i$$

The relationship between w_n (the n-th column of matrix $\mathbf{W}$) and the input vector $\mathbf{x}^*$, as well as the training data $\mathbf{x}_1$ to $\mathbf{x}_n$, can be expressed as:

$$w_n = \mathbf{x}^* \cdot \left(\mathbf{X}^T \mathbf{X}\right)^{-1} \cdot \mathbf{x}_n$$

Here, w_n represents the n-th index of the $1 \times N$ matrix $\mathbf{W}$. $\mathbf{X}$ is a matrix whose rows are the N training vectors $\mathbf{x}_1$ to $\mathbf{x}_n$. The expression $\mathbf{x}^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{x}_n$ signifies that w_n depends on the input $\mathbf{x}^*$ and all the training data from $\mathbf{x}_1$ to $\mathbf{x}_n$.

Comparison of Unregularized Linear Regression Model with Weighted K-Nearest Neighbors

The weighted K-Nearest Neighbors (WKNN) algorithm can be represented with the following formula:

$$\hat{y} = \frac{1}{\sum_{i=1}^{K} w_i} \sum_{i=1}^{K} w_i \cdot y_i$$

where

$$w_n = \frac{1}{\|\mathbf{x}^* - \mathbf{x}_n\|_2}$$

while Unregularized Linear Regression Mode:

$$w_n = \mathbf{x}_* \cdot \left(\mathbf{X}^T \mathbf{X}\right)^{-1} \cdot \mathbf{x}_n$$

The weights, denoted as w_n, in this regression model depend on both the test input $\mathbf{x}^*$ and the entire set of training data $\mathbf{x}_1$ to $\mathbf{x}_N$. These weights are determined by the inner product between the test input $\mathbf{x}^*$ and each training input $\mathbf{x}_n$, with adjustments made based on the characteristics of the training data matrix $\mathbf{X}$.

In a weighted variation of the K-Nearest Neighbors (KNN) algorithm, w_n is commonly defined as the reciprocal of the distance between the test input $\mathbf{x}^*$ and the training input $\mathbf{x}_n$. This implies that in KNN, the weights are predominantly influenced by the spatial separation of data points. In contrast, in linear regression, the weights take into account a more intricate relationship, which encompasses inner products and the training data matrix.

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

Student Name: Yashvir Singh Nathawat
Roll Number: 231110059
Date: September 15, 2023

QUESTION 5

Given Loss function using masked input be

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

In matrix form :

$$L(M) = \|y - \tilde{X}w\|^2$$

To represent the masked input $\tilde{x}_n = x_n \odot m_n$ in matrix form, we can use the following:

$$\tilde{X} = X \odot M$$

Where:
- X is the original input matrix with dimensions $N \times D$.
- M is a matrix of the same size as X where each element M_{ij} follows a Bernoulli distribution with probability p, i.e., $M_{ij} \sim \text{Bernoulli}(p)$. This matrix M is the binary mask matrix. - $\odot$ represents the element-wise product (Hadamard product).

$$L(M) = \|y - (M \odot X)w\|^2$$

New loss function with masked input is as follows:

$$L(w) = \arg\min_w \left\{ \mathbb{E}\left[\|y - (M \odot X)w\|^2 \right] \right\}$$

$$L(w) = \arg\min_w \mathbb{E}\left[((y - (M \odot X))^T (y - (M \odot X)w) \right]$$

Using result :

$$E[a^T a] = \text{Tr}(\Sigma) + c^T c$$

where-

$$c = E[a], \quad \Sigma = \text{Var}[a]$$

$$a = y - (M \odot X), E(a) = y - pXw$$

Also,

$$\begin{aligned} \text{Tr}(\Sigma) = \text{Tr}(\text{Var}[a]) &= \text{Tr}(\text{Var}[(y - (M \cdot X)w]) \\ &= p(1-p)\text{Tr}[(Xw)(Xw)^T] \end{aligned}$$

and

$$c^T c = E[a]^T E[a] = (y - pXw)^T (y - pXw)$$

Using above all result, we get:

$$L(w) = \arg\min_w \left\{ (y - pXw)^T (y - pXw) + p(1-p)\,\text{Tr}\left((Xw)(Xw)^T \right) \right\}$$

$$\Rightarrow L(w) = \arg\min_w \left\{ \|y - pXw\|^2 + p(1-p)\text{Tr}\left(Xww^T X^T \right) \right\}$$

$$\Rightarrow L(w) = \arg\min_{w} \left\{ \|y - pXw\|^2 + p(1-p)w^T(X^TX)w \right\}$$

$$L(w) = \arg\min_{w} \left\{ \|y - pXw\|^2 + p(1-p)\|\sqrt{\text{diag}(X^TX)}w\|^2 \right\}$$

Comparing $L(w)$ with ridge regression objective function,

$$L_{\text{ridge}}(w) = \arg\min_{w} \left\{ (y - Xw)^T(y - Xw) + \lambda w^T w \right\}$$

It is obvious that the objective $L(w)$ resembles the objective function of ridge regression and is therefore identical to minimizing a regularized loss function, where $p(1-p)\|\sqrt{\text{diag}(X^TX)}w\|^2$ acts as a regularizer and $\|y - pXw\|^2$ is equivalent to squared loss.

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

Student Name: Yashvir Singh Nathawat
Roll Number: 231110059
Date: September 15, 2023

QUESTION 6

Method 1: The convex.ipynb file's first method's accuracy was 46.89320388349515
Method 2:

Lambda	Accuracy (%)
0.01	58.090614886731395
0.1	59.54692556634305
1	67.39482200647248
10	73.28478964401295
20	71.68284789644012
50	65.08090614886731
100	56.47249190938511

Table 1: Accuracy Results

Best accuracy comes at lambda = 10.