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

QUESTION 1

Student Name: Sanjeev Kumar
Roll Number: 231110044
Date: September 15, 2023

Training data:

$$\{\mathbf{x_n}, \mathbf{y_n}\}_{n=1}^{N};$$

Optimization Problem:

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

To find the optimal value of $\mathbf{w_c}$ and $\mathbf{M_c}$, we can differentiate the loss function partially w.r.t. each of the variable and equate to zero. (Since loss function is a convex function, it will have only global minima).

1. For Optimal $\mathbf{w_c}$: Taking partial derivative of the objective function with respect to $\mathbf{w_c}$ and setting it to zero:

$$\frac{\partial L}{\partial \mathbf{w_c}} = 0$$

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

$$\mathbf{w_c} = \frac{1}{N_c} \sum_{\mathbf{x_n}:\mathbf{y_n}=c} (\mathbf{x_n})$$

2. For Optimal $\mathbf{M_c}$: Taking partial derivative of the objective function with respect to $\mathbf{M_c}$ and setting it to zero:

$$\frac{\partial L}{\partial \mathbf{M_c}} = 0$$

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

Solving for the optimal $\mathbf{M_c}$ is more complex and involves matrix calculus. The optimal $\mathbf{M_c}$ would depend on the data and the specific values of $\mathbf{w_c}$ that minimize the objective function.

Regarding the special case when $\mathbf{M_c}$ is an identity matrix ($\mathbf{I}$), this simplifies the optimization problem. In this case, the quadratic term $(\mathbf{x_n} - \mathbf{w_c})^T \mathbf{M_c} (\mathbf{x_n} - \mathbf{w_c})$ becomes $(\mathbf{x_n} - \mathbf{w_c})^T (\mathbf{x_n} - \mathbf{w_c})$, which is the Euclidean distance between $\mathbf{x_n}$ and $\mathbf{w_c}$. The optimization problem then becomes:

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

The term $-\log|\mathbf{I}|$ is a constant and doesn't affect the optimization. So, the model reduces to finding the optimal $\mathbf{w_c}$ that minimizes the sum of squared distances between the data points $\mathbf{x_n}$ and $\mathbf{w_c}$, which is a common form of k-means clustering, where $\mathbf{w_c}$ represents the cluster centroids.

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

Student Name: Sanjeev Kumar
Roll Number: 231110044
Date: September 15, 2023

QUESTION 2

The one-nearest-neighbor (1-NN) algorithm is consistent in the noise-free setting, where every training input is labeled correctly. In this scenario, the Bayes optimal error rate is zero because all training data is correctly classified. The 1-NN algorithm, which classifies a test point based on the class of its closest neighbor in the training data, will also achieve zero error in this noise-free setting because it will select the training point such that it perfectly matches the test input.

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

Student Name: Sanjeev Kumar
Roll Number: 231110044
Date: September 15, 2023

QUESTION 3

When constructing Decision Trees for regression, we want to split on features which minimizes the variance of the real-valued labels within each child node. This quantifies the homogeneity or similarity of the labels in each node.

$$Variance = \frac{\sum(X - \mu)^2}{N}$$

Lower the value of Variance, higher the purity or homogeneity of the node and higher the variance more impure node, we will get.

Steps to calculate Variance:

1. Calculate variance of each child node using the above formula.
2. Calculate variance of each split which will work as a weighted average variance of child nodes.
3. Here we will select the split which has the lowest variance value.

These steps will continue until the homogeneous node is achieved.

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

Student Name: Sanjeev Kumar
Roll Number: 231110044
Date: September 15, 2023

QUESTION 4

For unregularized linear regression model, weight vector can be given as:

$$\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

where: $\mathbf{X}$ is the matrix having N training points each having D features.
y is the column vector of all the responses of the training input.

Let $\mathbf{x}_*$ be the test point then prediction y_* can be given as:

$$y_* = \mathbf{w}^T\mathbf{x}_* = {\mathbf{x}_*}^T\mathbf{w}$$

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

$$y_* = \mathbf{W}\mathbf{y}$$

Where, $\mathbf{W} = {\mathbf{x}_*}^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$. Therefore, $\mathbf{W} = (\mathbf{w_1 w_2}....\mathbf{w_n})$ comes out to be a 1xN matrix. We can also write $\mathbf{y} = (y_1 y_2 ... y_n)^T$

y_* can be re-written as:

$$y_* = \sum_{n=1}^{N} \mathbf{w_n} y_n$$

where $\mathbf{w}_n$ is a n^{th} index of 1xN matrix $\mathbf{W}$, Since $\mathbf{X}$ is a matrix having N training data inputs, $\mathbf{w_n}$ can be written as:

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

In linear regression, the weight vector depends on the full dataset $\mathbf{X}^T\mathbf{X}$, whereas in weighted kNN, the weights typically depend only on the nearest neighbors.
The role of $\mathbf{x}_*$ in the expression differs between the two methods; it appears in the numerator for linear regression and in the denominator for kNN.
The representation of weights also differs: in linear regression, they are products of $\mathbf{x}_*$ and training data, while in kNN, they are typically expressed as a sum in the denominator.

Student Name: Sanjeev Kumar
Roll Number: 231110044
Date: September 15, 2023

QUESTION

5

$$L(\mathbf{M}) = \sum_{n=1}^{N} \left(y_n - \mathbf{w}^T \tilde{\mathbf{x}}_n\right)^2$$
$$L(\mathbf{M}) = \mathbf{k}y - \tilde{\mathbf{X}}\mathbf{w}\mathbf{k}^2$$
$$L(\mathbf{M}) = \mathbf{k}y - (\mathbf{R} \circ \mathbf{X})\,\mathbf{w}\mathbf{k}^2$$

When the input $\mathbf{X}$ is dropped out such that any input dimension is retained with probability p, then the expected value of $L(M)$ is given by:

$$\mathbb{E}_{R\sim\text{Bernoulli}(n;p)}\left[L(M)\right]$$

To minimize this with respect to $\mathbf{w}$, a new objective function $L(\mathbf{w})$ is defined as:

$$L(\mathbf{w}) = \arg\min_{\mathbf{w}} \left[\mathbb{E}_{R\sim\text{Bernoulli}(n;p)}\left[\mathbf{k}^T\mathbf{k}\right]\right]$$

This can be further expressed as:

$$L(\mathbf{w}) = \arg\min_{\mathbf{w}} \left[\mathbb{E}_{R\sim\text{Bernoulli}(n;p)}\left[(y - (\mathbf{R}\circ\mathbf{X})\mathbf{w})^T(y - (\mathbf{R}\circ\mathbf{X})\mathbf{w}) + p(1-p)\text{TRACE}\left((\mathbf{X}\mathbf{w})(\mathbf{X}\mathbf{w})^T\right)\right]\right]$$

Simplifying the expression:

$$L(\mathbf{w}) = \arg\min_{\mathbf{w}} \left[\mathbf{k}y - p\mathbf{X}\mathbf{w}\mathbf{k}^2 + p(1-p)\text{TRACE}\left(\mathbf{X}\mathbf{w}\mathbf{w}^T\mathbf{X}^T\right)\right]$$

Comparing $L(\mathbf{w})$ with the ridge regression objective function:

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

It's clear that the objective $L(\mathbf{w})$ resembles the objective function of ridge regression, where $p(1-p)\text{diag}(\mathbf{X}^T\mathbf{X})$ acts as a regularizer and $\mathbf{k}y - p\mathbf{X}\mathbf{w}\mathbf{k}^2$ is like the squared loss.

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

Student Name: Sanjeev Kumar
Roll Number: 231110044
Date: September 15, 2023

QUESTION 6

Method 1: Convex method -
Accuracy of the model on given test cases is: 46.89320388349515

Method 2: Linear regression -
Accuracy of model on different λ values are:
Test accuracy for $\lambda = 0.1$ is 59.54692556634305
Test accuracy for $\lambda = 1$ is 67.39482200647248
Test accuracy for $\lambda = 4$ is 73.52750809061487
Test accuracy for $\lambda = 8$ is 73.43042071197411
Test accuracy for $\lambda = 12$ is 73.23624595469256
Test accuracy for $\lambda = 15$ is 72.79935275080906
Test accuracy for $\lambda = 20$ is 71.68284789644012
Test accuracy for $\lambda = 30$ is 69.51456310679612

$\lambda = 4$ gives the best test case accuracy over different values of λ, we tried.