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

Student Name: Vikram Kumar
Roll Number: 23128003
Date: September 15, 2023

QUESTION 1

To find the optimal values of w_c and M_c for the given objective/loss function, we can take the derivative of the objective with respect to w_c and M_c, set the derivatives to zero, and solve for w_c and M_c.

The objective function is as follows:

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

Optimal w_c:

To find the optimal w_c, we'll take the derivative of the objective with respect to w_c, set it to zero, and solve for w_c:

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

Now, let's solve for w_c:

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

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

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

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

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

$$\sum_{x_n:y_n=c} x_n = N_c w_c$$

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

So, the optimal value for w_c is the mean of all the training examples in class c:

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

Optimal M_c:

Now, let's find the optimal M_c. To do this, we'll take the derivative of the objective with respect to M_c, set it to zero, and solve for M_c:

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

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

$$\left[\frac{\partial det(X)}{\partial X} = det(X)\left(X^{-1}\right)^T\right]$$

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

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

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

$$M_c^{-1} = \left(\frac{1}{N_c}\sum_{x_n:y_n=c}(x_n - w_c)(x_n - w_c)^T\right)^T \quad \left[(X.X^T)^T = (X^T)^T.X^T = X.X^T\right]$$

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

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

Special Case when M_c is an Identity Matrix:

$$M_c = I$$

When M_c is an identity matrix, the model simplifies. In this case, the optimization problem reduces to finding the optimal w_c only, and the regularization term ($\log|M_c|$) becomes constant. The model becomes a linear classifier with the Mahalanobis distance metric reduced to the Euclidean distance. The optimal w_c would still be the mean of the training examples in class c:

$$L = \sum_{x_n:y_n=c}\frac{1}{N_c}(x_n - w_c)^T I(x_n - w_c) - \log|I|$$

$$L = \sum_{x_n:y_n=c}\frac{1}{N_c}(x_n - w_c)^T(x_n - w_c) - \log|I|$$

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

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

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

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

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

$$\sum_{x_n:y_n=c} x_n = N_c w_c$$

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

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

This corresponds to a simple centroid-based classifier for class c, where the mean vector w_c is used to classify new data points based on their proximity to the class centroid.

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

Student Name: Vikram Kumar
Roll Number: 23128003
Date: September 15, 2023

QUESTION 2

Yes, 1NN will be consistent in this case!

Yes,1-nearest neighbor (1NN) algorithm will consistently perform well in this scenario. This is because there is are infinite amount of training data, and each data point is accurately labeled without any errors. So when we receive a test data point, you can always find a training data point very close to it. As the number of training data points is infinity, the probability of finding a nearby training point is closer to 1. So you can classify test data with absolute accuracy, essentially because you already have those test data points within your extensive training set.

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

Student Name: Vikram Kumar
Roll Number: 23128003
Date: September 15, 2023

QUESTION 3

When constructing Decision Trees for regression, the goal is to split on features that minimize the variance of the real-valued labels within each child node. The criterion that quantifies the homogeneity/diversity of the set of real-valued labels is typically the mean squared error (MSE) or variance.

Mean Squared Error (MSE) is a common measure used for regression in Decision Trees. It quantifies the average squared difference between the predicted values and the actual values for a set of examples at a node. Lower MSE indicates that the labels are more homogeneous (closer to a single value), which is desirable for regression.

Mathematically, for a node N with n examples and their corresponding labels $y_1, y_2, \ldots, y_n$, the MSE is calculated as:

$$\text{MSE}(N) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y})^2$$

where $\bar{y}$ is the mean of the labels $y_1, y_2, \ldots, y_n$.

When deciding which feature to split on, the one that minimizes the weighted sum of the MSE in the resulting child nodes after the split is chosen. The split that leads to the lowest overall MSE (highest reduction in MSE) is considered the best choice.

In addition to MSE, we can use the R-squared (R2) score to assess the performance of the regression model. R2 measures the proportion of the variance in the dependent variable that is predictable from the independent variables. It is defined as:

$$R^2 = 1 - \frac{\text{MSE}}{\text{Var}(y)}$$

where $\text{Var}(y)$ is the variance of the labels.

In summary, for regression in Decision Trees, our aim to minimize the Mean Squared Error (MSE) or variance of the labels within each node to determine the best feature to split on and can use the R-squared (R2) score to evaluate the model's performance.

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

Student Name: Vikram Kumar
Roll Number: 23128003
Date: September 15, 2023

QUESTION 4

To show that $\hat{w} = (X^T X)^{-1} X^T y$
We want to find the value of $\hat{w}$ that minimizes the least squares loss function:

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

Where: - $L(\hat{w})$ is the least squares loss.

in the matrix form:

$$L(\hat{w}) = (X\hat{w} - y)^T (X\hat{w} - y) = (\hat{w}^T X^T - y^T)(X\hat{w} - y)$$

$$L(\hat{w}) = (\hat{w}^T X^T X\hat{w} - \hat{w}^T X^T y - y^T X\hat{w} + y^T y)$$

To find the minimum value of loss function first step is to differentiate the loss function with respect to $\hat{w}$ and set it equal to zero :

$$\frac{\partial L}{\partial \hat{w}} = 0$$

Let's compute the derivative:

$$\frac{\partial L}{\partial \hat{w}} = \frac{\partial}{\partial \hat{w}}(\hat{w}^T X^T X\hat{w} - \hat{w}^T X^T y - y^T X\hat{w} + y^T y)$$

$$\frac{\partial L}{\partial \hat{w}} = 2X^T X\hat{w} - X^T y - X^T y + 0$$

$$\frac{\partial L}{\partial \hat{w}} = 2X^T X\hat{w} - 2X^T y$$

Setting this equal to zero:

$$2(X^T X\hat{w} - X^T y) = 0$$

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

$$X^T X\hat{w} - X^T y = 0$$

$$X^T X\hat{w} = X^T y$$

Now, to isolate $\hat{w}$, we can multiply both sides by $(X^T X)^{-1}$:

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

$\hat{w} = (X^T X)^{-1} X^T y$. Therefore, a prediction at test input x_* can be written as

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

. Consequently,

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

This can also be expressed as $y_* = Wy$, where $W = {x_*}^T (X^T X)^{-1} X^T$. Hence, $W = (w_1, w_2, \ldots, w_N)$ forms a $1 \times N$ matrix. We can represent $y = (y_1, y_2, \ldots, y_N)^T$. As a result,

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

Here, w_n corresponds to the n-th index of the $1 \times N$ matrix W. It's worth noting that W depends on the input x_* and all the training data from x_1 to x_N. This is due to the presence of the term $X^T X$ in the expression for w_n.

These weights w_n are different from the weights in a weighted version of k-nearest neighbors in the following way:

- In linear regression, the weights w_n are determined by the dot product between the test input and each training input, weighted by the inverse covariance matrix of the training data. These weights are based on the relationship between the input features and the response variable.
- In a weighted version of k-nearest neighbors, the weights are typically based on the inverse distance of the test input from the training inputs. These weights depend solely on the spatial proximity between the test input and the training data points, without considering the relationships between input features and response variables explicitly.

Student Name: Vikram Kumar
Roll Number: 23128003
Date: September 15, 2023

QUESTION 5

To show that minimizing the expected value of the new loss function, which involves masked inputs, is equivalent to minimizing a regularized loss function, we need to derive the expression of this regularized loss function.

Let's start by defining the new loss function with masked inputs:

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

Where:

- $L(w)$ is the new loss function.
- y_n is the target or observed output for the n-th input.
- w is the weight vector.
- $\tilde{x_n}$ represents the input with masked features.

As given, $\tilde{x_n} = x_n \circ m_n$, where $\circ$ denotes elementwise product (Hadamard product) and m_n is the binary mask vector. Specifically, $m_{nd} = 1$ with probability p (feature retained) and $m_{nd} = 0$ with probability $1-p$ (feature masked).

Now, let's consider the expectation with respect to the mask vectors m_n:

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

Using linearity of expectation:

$$\mathbb{E}[L(w)] = \frac{1}{N}\sum_{n=1}^{N}\mathbb{E}[(y_n - w^T\tilde{x_n})^2]$$

Now, we'll expand the square term and consider the expectations separately:

$$\begin{aligned}\mathbb{E}[L(w)] &= \frac{1}{N}\sum_{n=1}^{N}\mathbb{E}[y_n^2 - 2y_n w^T\tilde{x_n} + (w^T\tilde{x_n})^2] \\ &= \frac{1}{N}\sum_{n=1}^{N}\Big(\mathbb{E}[y_n^2] - 2\mathbb{E}[y_n w^T\tilde{x_n}] + \mathbb{E}[(w^T\tilde{x_n})^2]\Big)\end{aligned}$$

Now, we'll use the fact that $\mathbb{E}[y_n w^T\tilde{x_n}] = \mathbb{E}[y_n]\mathbb{E}[w^T\tilde{x_n}]$ because the mask vectors m_n are independent of the other variables:

$$\mathbb{E}[L(w)] = \frac{1}{N}\sum_{n=1}^{N}\Big(\mathbb{E}[y_n^2] - 2\mathbb{E}[y_n]\mathbb{E}[w^T(x_n \circ m_n)] + \mathbb{E}[(w^T\tilde{x_n})^2]\Big)$$

Now, consider that $\mathbb{E}[y_n]$ and $\mathbb{E}[w^T \tilde{x_n}]$ are constants with respect to w since they do not depend on it. Also, note that $\mathbb{E}[(w^T \tilde{x_n})^2]$ is a constant term with respect to w. Therefore, we can rewrite the above expression as:

$$\mathbb{E}[L(w)] = \frac{1}{N}\sum_{n=1}^{N}\Big(\mathbb{E}[y_n^2] - 2\mathbb{E}[y_n]\mathbb{E}[w^T(x_n \circ m_n)] + \mathbb{E}[(w^T \tilde{x_n})^2]\Big) = C - 2\sum_{n=1}^{N}\mathbb{E}[y_n]\mathbb{E}[w^T(x_n \circ m_n)]$$

Now, we can see that minimizing the expected value of $L(w)$ is equivalent to minimizing a regularized loss function, where the regularization term is given by:

$$\Omega(w) = \sum_{n=1}^{N}\mathbb{E}[y_n]\mathbb{E}[w^T(x_n \circ m_n)]$$

Therefore, the regularized loss function is:

$$\text{Regularized Loss}(w) = \frac{1}{N}\sum_{n=1}^{N}(y_n - w^T \tilde{x_n})^2 + \lambda\sum_{n=1}^{N}\mathbb{E}[y_n]\mathbb{E}[w^T(x_n \circ m_n)]$$

Where λ is a regularization parameter that controls the importance of the regularization term. This regularization term encourages the model to have smaller weights, effectively acting as a form of feature selection.

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

Student Name: Vikram Kumar
Roll Number: 23128003
Date: September 15, 2023

QUESTION 6

Method 1
Test accuracy with Euclidean distance is 46.89320388349515$.
Method 2
Test accuracy for lambda = 0.01 is: 58.090614886731395
Test accuracy for lambda = 0.1 is: 59.54692556634305
Test accuracy for lambda = 1 is: 67.39482200647248
Test accuracy for lambda = 10 is: 73.28478964401295
Test accuracy for lambda = 20 is: 71.68284789644012
Test accuracy for lambda = 50 is: 65.08090614886731
Test accuracy for lambda = 100 is: 56.47249190938511