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

Student Name: Divyaksh Shukla
Roll Number: 231110603
Date: September 14, 2023

QUESTION 1

We need to minimise the below objective function with respect to w_c and $\mathbf{M}_c$. We start off by optmising the function with respect to w_c and equate it to 0

$$\begin{aligned}
&\frac{\partial}{\partial w_c}\left(\frac{1}{N_c}\sum_{x_n:y_n=c}(x_n-w_c)^T\mathbf{M}_c(x_n-w_c)-log|\mathbf{M}_c|\right)=0\\
&\implies \frac{-2}{N_c}\mathbf{M}_c\left(\sum_{x_n:y_n=c}x_n-w_c\right)=0\\
&\implies w_c=\sum_{x_n:y_n=c}x_n \qquad (1)
\end{aligned}$$

Now with respect to $\mathbf{M}_c$ and equating it to 0 we get:

$$\begin{aligned}
&\frac{\partial}{\partial \mathbf{M}_c}\left(\frac{1}{N_c}\sum_{x_n:y_n=c}(x_n-w_c)^T\mathbf{M}_c(x_n-w_c)-log|\mathbf{M}_c|\right)=0\\
&\implies \frac{1}{N_c}\sum_{x_n:y_n=c}(x_n-w_c)^T(x_n-w_c)-\frac{-1}{\cancel{|\mathbf{M}_c|}}\cancel{|\mathbf{M}_c|}(\mathbf{M}_c^{-1})^T=0\\
&\implies \mathbf{M}_c^{-1}=\frac{1}{N_c}\sum_{x_n:y_n=c}(x_n-w_c)^T(x_n-w_c) \qquad (2)
\end{aligned}$$

Here the RHS of (2) is the covarianace matrix of x_n and w_c (assuming a uniform probability distribution of x_n). Thus,

$$\begin{aligned}
&\implies \mathbf{M}_c^{-1}=Cov_{\mathbf{x}\sim P(\mathbf{x})}[\mathbf{x};w_c]\\
&\implies \mathbf{M}_c=Cov^{-1}_{\mathbf{x}\sim P(\mathbf{x})}[\mathbf{x};w_c]\ \text{, where } \mathbf{x}\in c \qquad (3)
\end{aligned}$$

(3) shows that $\mathbf{M}_c$ is inverse of the covariance matrix.

In the special case that $\mathbf{M}_c$ is an identity matrix then the equation in the question boils down to finding the euclidean distance between x_n and w_c.

$$(\hat{w_c})=\arg\min_{w_c}\frac{1}{N_c}\sum_{x_n:y_n=c}(x_n-w_c)^T(x_n-w_c) \qquad (4)$$

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

Student Name: Divyaksh Shukla
Roll Number: 231110603
Date: September 14, 2023

QUESTION 2

Consistency is defined when the error rate while testing is also at Bayes' optimal. Thus, we have also got a definite decision boundary.

In a noise-free setting, with a sampled set of the population used for training, the decision boundary is decided by the sampled points and even if the training shows 0 error rate the decision boundary is not necessarily optimal. Thus, in a noise-free setting one-nearest neighbour algorithm is not consistent.

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

Student Name: Divyaksh Shukla
Roll Number: 231110603
Date: September 14, 2023

QUESTION 3

We can use the same Information Gain formula but replace Entropy with a cost function which measures co-linearity between points say Cosine similarity. Which means:

$$IG = \frac{|S_1|}{|S|} Sim(S_1) + \frac{|S_2|}{|S|} Sim(S_2) - Sim(S)$$

where

$$Sim(D) = \sum_{i,j}^{|D|} (y_i . y_j)$$

The node with the highest information gain will then be selected as a splitting attribute.

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

Student Name: Divyaksh Shukla
Roll Number: 231110603
Date: September 14, 2023

QUESTION 4

Let's start with the basic formula for linear regression to find $f(x_*)$

$$\begin{aligned} f(x_*) &= w^T x_* \\ &= \left[(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T y\right] x_* \\ &= (\mathbf{X}^T y)^T \left[(\mathbf{X}^T\mathbf{X})^{-1}\right]^T x_* \\ &= y^T\mathbf{X}\left[(\mathbf{X}^T\mathbf{X})^{-1}\right]^T x_* \end{aligned}$$

Transposing on both sides. $f(x_*)$ is a scalar so its transpose will yield the same value.

$$f(x_*) = \left[x_*^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\right] y \tag{5}$$

In (5), $\left[x_*^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\right]$ corresponds to w_n of the equation given in the question. This is quite different from the typical style of obtaining the test response by weighing the training responses based on the inverse of distance between training points and test point. Here we are taking a dot product similarity between the test data point and all the training data points.

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

Student Name: Divyaksh Shukla
Roll Number: 231110603
Date: September 14, 2023

QUESTION 5

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

now let's write this equation in vector and matrix form to make the algebra easier. So $\sum y_n = \mathbf{y}$ and $\sum x_n = \mathbf{X}$.

Let's also assume that $\tilde{\mathbf{X}} = \mathbf{X}.\mathbf{M}$ where $\mathbf{M}$ is the matrix obtained by putting in 0 and 1 based on the Bernoulli distribution, with p defining the probability of 1 (include the datapoint) and $(1-p)$ defining the probability of 0 (excluding the datapoint).

Therefore, $\mathbb{E}[\tilde{X}] = pX$ and $cov(\tilde{X}, \tilde{X}) = p(1-p)\Gamma^2$, where Γ is the covariance matrix of M, but as all the values are independent so it is just a diagonal matrix only carrying variance values.

$$\begin{aligned}
L(w) &= (\mathbf{y}^T - w^T\tilde{\mathbf{X}}^T)(\mathbf{y} - \mathbf{X}w) \\
&= \mathbb{E}[(\mathbf{y}^T - w^T\tilde{\mathbf{X}}^T)(\mathbf{y} - \mathbf{X}w)] \\
&= \mathbb{E}[\mathbf{y}^T\mathbf{y} - 2\mathbf{y}^T\tilde{\mathbf{X}}w + w^T\tilde{\mathbf{X}}^T\tilde{\mathbf{X}}w] \\
&= \mathbf{y}^T\mathbf{y} - 2p\mathbf{y}^T\mathbf{X}w + w^T\mathbb{E}[\tilde{\mathbf{X}}^T\tilde{\mathbf{X}}]w
\end{aligned}$$

From $cov(\mathbf{M}, \mathbf{M}) = \mathbb{E}[\mathbf{M}\mathbf{M}^T] - \mathbb{E}[\mathbf{M}]\mathbb{E}[\mathbf{M}]^T$ we get

$$\begin{aligned}
\implies \mathbb{E}[L(w)] &= \mathbf{y}^T\mathbf{y} - 2p\mathbf{y}^T\mathbf{X}w + p^2w^T\mathbf{X}^T\mathbf{X}w + p(1-p)w^T\Gamma^2w \\
&= (\mathbf{y}^T - pw^T\mathbf{X}^T)(\mathbf{y} - p\mathbf{X}w) + p(1-p)||\Gamma w||^2 \\
&= \sum_{n=1}^{N}(y_n - w^T x_n)^2 + p(1-p)||\Gamma w||^2
\end{aligned}$$

This is similar to L_2 regularized loss function where $\lambda = p(1-p)\Gamma^2$.

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

Student Name: Divyaksh Shukla
Roll Number: 231110603
Date: September 14, 2023

QUESTION
6

1 Learning with Prototypes

We have a data of 4096 features (extracted from a deep learning model). Our task is to predict the class labels of the unseen datapoints.

1. We start off by first looking at the data
2. Then we compute the means of the 40 seen classes as μ_k

1.1 Method 1 - Using class attributes similarity

We compute the similarity between seen and unseen classes by taking the dot product between the class attribute vectors of the seen and unseen classes like below

$$similarity = A_s A_u^T$$

Then the similarity values were normalised by dividing each value in the above similarity matrix by the sum of its row.

$$similarity_i = \frac{similarity_{i,j}}{\sum_{j=1}^{40} similarity_{i,j}}$$

Then we can estimate the means of the unseen classes by:

$$unseen\ means = similarity * means\ seen$$

This method gives an accuracy of 46.893%.

1.2 Method 2 - Linear regression

This method starts similar to Method 1 by computing the means of the seen classes. Then we perform linear regression between the class attribute values and the means to get a model to map class attributes to means. We then use this model to compute means of the unseen classes and give predictions.

In the linear regression we can regularize the model with a parameter λ. In this case the chosen values of λ are: 0.01, 0.1, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0, 6.5, 7.0, 7.5, 8.0, 8.5, 9.0, 9.5, 10, 20, 50, 100.

Then prediction accuracy is compared with values of λ to get the optimal λ value with the highest accuracy.

This model got highest accuracy of 73.722% with an optimal λ of 6.50.

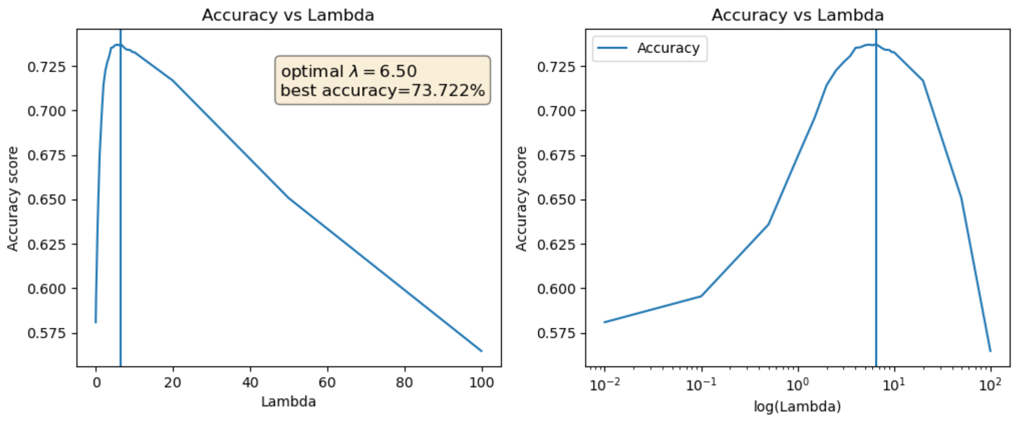

Figure 1: Accuracy vs lambda. The left chart shows the accuracy score compared to each value of lambda on a linear scale, while the right chart shows lambda on a log scale. The optimal lambda is marked with a vertical blue line