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

Student Name: B Anshuman
Roll Number: 200259
Date: September 15, 2023

QUESTION 1

Consider a single class c. We can find optimal w_c and M_c which generalizes for all classes.

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

$$(\hat{w_c}, \hat{M_c}) = \arg\min_{w_c, M_c} \mathcal{L}(w_c, M_c)$$

By first order optimization, $\frac{\partial \mathcal{L}}{\partial w_c} = 0$ would yield optimum w_c, similarly for M_c.

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial w_c} &= \sum_{x_n : y_n = c} \frac{1}{N_c} \frac{\partial}{\partial (x_n - w_c)} (x_n - w_c)^T M_c (x_n - w_c) \frac{\partial}{\partial w_c}(x_n - w_c) \\
&= \sum_{x_n : y_n = c} (M_c + M_c^T)(x_n - w_c)(-1) \\
&= \sum_{x_n : y_n = c} x_n - N_c w_c \\
&= 0
\end{aligned}$$

$$w_c = \frac{1}{N_c} \sum_{x_n : y_n = c} x_n \tag{2}$$

Similarly for M_c

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial w_c} &= \sum_{x_n : y_n = c} \frac{1}{N_c}(x_n - w_c)(x_n - w_c)^T - (M_c^{-1})^T \\
&= 0
\end{aligned}$$

$$M_c = \left(\left(\frac{1}{N_c} \sum_{x_n : y_n = c} (x_n - w_c)(x_n - w_c)^T \right)^{-1} \right)^T \tag{3}$$

Suppose if M_c was identity; 1 would simplify to euclidean squared loss:

$$\mathcal{L} = \frac{1}{N_c} \sum_{x_n : y_n = c} (x_n - w_c)^T (x_n - w_c) \tag{4}$$

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

Student Name: B Anshuman
Roll Number: 200259
Date: September 15, 2023

QUESTION 2

Given a classification algorithm is said to be *consistent* if, when it has access to infinite training data, it's error rate approaches optimal error rate (called *Bayes optimal*).

One Nearest-Neighbour algorithm with infinite correctly labelled training data implies all the input space and it's labels is provided to the algorithm already, which means it's error rate **would approach zero** for test data, because the test set would belong to the input space, which already is present in the training set. Therefore 1NN is Bayes Optimal.

Note that algorithms that use representation/prototype based learning will on the other hand not reach Bayes optimal error rate.

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

Student Name: B Anshuman
Roll Number: 200259
Date: September 15, 2023

QUESTION 3

First defining *input set* of a node, is the set of data points received by the node during training that has to be split into multiple more pure sets, as child nodes.

At every decision stump of the decision tree, we have maximum information gain that is lowest possible average entropy of it's child nodes' sets.

Given, labels are real-valued, with features being classes or real-valued or both. We'll be using variance of label value to quantify the homogenity of a set.

Let (X, y) be the input set at root. For splitting at the root (recursively, root is every subtree's top), let's choose x feature out of X to consider for the decision stump. Here arise 2 cases for a feature x:

- x is categorical, $x \in \{A_1, A_2, \cdots, A_N\}$: Measure the variance of input set, $\frac{1}{N_r - 1}\sum_{n=1}^{N_r}(y_n - \bar{y}_r)^2$, where $\bar{y}_r = \frac{1}{N_r}\sum_{n=1}^{N_r} y_n$ We can split the input space according to each class A_i with N_{a_i} number of elements (note $\sum_{i=1}^{N} N_{a_i} = N_r$) and measure the variance of each child node set, $\frac{1}{N_{a_i} - 1}\sum_{n=1}^{N_{a_i}}(y - \bar{y}_{a_i})^2$ where y_{a_i} is mean similarly defined as y_r.
  We would require the following averaged variance of child sets to be minimized:

$$\sum_{i=1}^{N} \frac{N_{a_i}}{N_r} \left(\frac{1}{N_{a_i} - 1} \sum_{n=1}^{N_{a_i}} (y - \bar{y}_{a_i})^2 \right) - \frac{1}{N_r - 1} \sum_{n=1}^{N_r} (y_n - \bar{y}_r)^2 \tag{5}$$

  For minimizing, we can iterate through other features of X

- x is real valued: We can split the input space to 2 categories $\{A_+, A_-\}$ on basis of $x < a$. We can decide a value by iterating through all data points (sorted according to x), choose a as average of 2 consecutive data points's x values, effectively splitting the input data set into those less than the average (A_+) and those greater than/equal (A_-). Now for all these a, calculate average variance as in 5 and take the particular a which gives the minimum average variance.
  We can extend this idea by iterating through all features, and choosing that particular x and corresponding a which has the minimum variance!

You can keep on splitting until each child node only has one data point, or you can stop splitting and label that particular node with the *average* label of the input set of that node.

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

Student Name: B Anshuman
Roll Number: 200259
Date: September 15, 2023

QUESTION 4

Linear Regression Model is:

$$y = w^T x$$

We know that Linear Regression Loss can be minimized to yield:

$$w_{LS} = \arg\min_w \mathcal{L}(w) = \arg\min_w \frac{1}{N}\sum_{n=1}^{N}(y_n - w^T x_n)^2$$

$$w_{LS} = (X^T X)^{-1} X^T y \tag{6}$$

$$= (X^T X)^{-1} \sum_{n=1}^{N} y_n x_n \tag{7}$$

During inference, $y_* = w_{LS}^T x_*$ can thus be written as

$$y_* = ((X^T X)^{-1} \sum_{n=1}^{N} y_n x_n) \cdot x_*$$
$$= \sum_{n=1}^{N} \left(((X^T X)^{-1} x_n)^T x_*\right) y_n$$

Therefore, $w_n = ((X^T X)^{-1} x_n)^T x_*$, where $(X^T X)^{-1}$ is the inverse of Gramian matrix (or covariance matrix of feature space), and is a constant during inference. Denote it as G. Therefore w_n can be seen as dot-product similarity between x_n and x_* projected to some space by G, more similar is x_* to x_n, higher weightage is given to y_n for $f(x_*)$.

In **K-nearest neighbours**, for a new input x_*, sort the distances from all N training samples, and choose nearest K samples. One way of deciding the label y_* can be:

$$d_n = \|x_* - x_n\|_2^2$$
$$= (x_* - x_n)^T (x_* - x_n)$$
$$y_* = \frac{1}{\sum_{i=1}^{K} \frac{1}{d_i}} \sum_{n=1}^{K} \frac{1}{d_n}\, y_n$$
$$\therefore w_n = \eta \frac{1}{(x_* - x_n)^T (x_* - x_n)}$$

w_n^{KNN} and w_n^{LS} are different in terms that *KNN* derieves similarity between features x_n and x_* in terms of inverse of distances, while *LS* derieves similarity in terms of inner product between them. Some other differences include *LS* depending on the entire dataset as Gramian matrix to calculate a single w_n, unlike that of *KNN*.

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

Student Name: B Anshuman
Roll Number: 200259
Date: September 15, 2023

QUESTION 5

Let's denote n^{th} sample as $x^{(n)}$ and i^{th} feature as x_i.

For a sample $x^{(n)}$, each feature can be masked with a probability p. Define $D \times D$ diagonal matrix $M^{(n)}$ with each (d,d) entry $M^{(n)}_{d,d} \sim Bernoulli(p)$, with $\tilde{x}^{(n)} = M^{(n)}x^{(n)}$.

Modified Loss function:

$$\mathcal{L}(w) = \sum_{n=1}^{N}(y^{(n)} - w^T M^{(n)}x^{(n)})^2$$

Expectation of $\mathcal{L}$:

$$\begin{aligned}
\mathbb{E}_M[\mathcal{L}(w)] &= \sum_{n=1}^{N}\mathbb{E}[(y^{(n)} - w^T M^{(n)}x^{(n)})^2] \\
&= \sum_{n=1}^{N}\mathbb{E}[(y^{(n)})^2 + (w^T M^{(n)}x^{(n)})^2 - 2y^{(n)}w^T M^{(n)}x^{(n)}] \\
&= \sum_{n=1}^{N}(y^{(n)})^2 + \mathbb{E}[(w^T M^{(n)}x^{(n)})^2] - 2y^{(n)}w^T px^{(n)} \qquad (8)
\end{aligned}$$

Second term for a particular n (denoting $x^{(n)}$ as x) is equivalent to $(\sum_{k=1}^{D} w_k m_k x_k)^2 = \sum_{j=1}^{D}\alpha_j^2 + \sum\sum_{k\neq l}\alpha_k\alpha_l$, where $\alpha_i = w_i m_i x_i$, and note that m_i and $m_j, i \neq j$ are independent.

With results that $\mathbb{E}[\alpha_k^2] = (w_k x_k)^2 p$ and $\mathbb{E}[\alpha_k]\mathbb{E}[\alpha_l] = (w_k x_k w_l x_l)p^2$ for $k \neq l$

$$\begin{aligned}
\mathbb{E}[(w^T M^{(n)}x^{(n)})^2] &= \sum_{j=1}^{D}\mathbb{E}[\alpha_j^2] + \sum_{k\neq l,k=1,l=1}^{D}\mathbb{E}[\alpha_k]\mathbb{E}[\alpha_l] \\
&= \big(p((w_1x_1)^2 + (w_2x_2)^2 + \cdots) + p^2(w_1x_1w_2x_2 + \cdots)\big) \\
&= p\sum_{j=1}^{D}(w_jx_j)^2 + p^2\sum_{k\neq l}w_kx_kw_lx_l + \left(p^2\sum_{j=1}^{D}(w_jx_j)^2 - p^2\sum_{j=1}^{D}(w_jx_j)^2\right) \\
&= p^2\left(\sum_{j=1}^{D}(w_jx_j)^2 + \sum_{k\neq l}w_kx_kw_lx_l\right) + p\sum_{j=1}^{D}(w_jx_j)^2 - p^2\sum_{j=1}^{D}(w_jx_j)^2 \\
&= p^2(\sum_{d=1}^{D}w_dx_d)^2 + p(1-p)\sum_{j=1}^{D}(w_jx_j)^2
\end{aligned}$$

Putting this at 8

$$\begin{aligned}
\mathbb{E}_M[\mathcal{L}(w)] &= \sum_{n=1}^{N}(y^{(n)})^2 + p^2\left(\sum_{d=1}^{D} w_d x_d^{(n)}\right)^2 + p(1-p)\sum_{d=1}^{D}(w_d x_d^{(n)})^2 - 2py^{(n)}w^T x^{(n)} \\
&= \sum_{n=1}^{N}(y^{(n)} - pw^T x^{(n)})^2 + p(1-p)\sum_{n=1}^{N}\sum_{d=1}^{D}(w_d x_d^{(n)})^2 \\
&= \sum_{n=1}^{N}(y^{(n)} - pw^T x^{(n)})^2 + p(1-p)\sum_{d=1}^{D} w_d^2 \sum_{n=1}^{N}(x_d^{(n)})^2 \\
&= \sum_{n=1}^{N}(y^{(n)} - pw^T x^{(n)})^2 + p(1-p)\sum_{d=1}^{D} w_d x_d^T x_d w_d \\
&= \sum_{n=1}^{N}(y^{(n)} - pw^T x^{(n)})^2 + p(1-p)w^T \Lambda w \qquad (9)
\end{aligned}$$

Where Λ is a $D \times D$ matrix,

$$\Lambda = \begin{cases} x_d^T x_d & i = d, j = d \\ 0 & else \end{cases}$$

Therefore 9 is the regularised loss form of expectation of masked linear regression model, and minimization of that would lead to minimization of regularised square loss.

Student Name: B Anshuman
Roll Number: 200259
Date: September 15, 2023

QUESTION
6

- Using *convex* combination method, test dataset accuracy obtained is 46.89%.
- *regress* over class attributes to get class means, yields the following results

Accuracy (%)	Lambda
58.09	0.01
59.55	0.1
67.39	1
73.28	10
71.68	20
65.08	50
56.47	100

Best value of λ is 10, giving 73.28% accuracy.