

HW 2 – Machine Learning Introduction

All code is present in the GitHub repo

<https://github.com/somesh-bagadiya/CMPE257-Fall23-Somesh-Bagadiya.git>

Folder Name is HW 2

Task 1 HP 1

Task 1, Hp1

→ $\prod_{n=1}^N P(y_n | x_n)$ is the likelihood for Linear Regⁿ

Now, Let's maximize it.

Maximizing the likelihood is equivalent to maximizing the below mentioned equation.

$$\frac{1}{N} \ln \left(\prod_{n=1}^N \theta(y_n w^T x_n) \right) \quad [\max]$$

We are using $\frac{1}{N} \ln$ because maximizing

the ~~above~~ Likelihood will have monotonically increasing graph. and the above mentioned eqⁿ will also have monotonically increasing graph

But we don't want to maximize the in sample error, we want to minimize it.

∴ New eqⁿ will be.

$$-\frac{1}{N} \ln \left(\prod_{n=1}^N \theta(y_n w^T x_n) \right)$$

$$\rightarrow \frac{1}{N} \ln \left[\prod_{n=1}^N \left(\frac{1}{\theta(y_n w^T x_n)} \right) \right]$$

Since its Log/ln we can write above mentioned equation as

$$\rightarrow \frac{1}{N} \sum_{n=1}^N \ln \left(\frac{1}{\theta(y_n w^T x_n)} \right)$$

Now, $\theta(y_n w^T x_n) = \frac{1}{1 + e^{-y_n w^T x_n}}$
(Signal)

thus,

$$\Rightarrow \frac{1}{N} \sum_{n=1}^N \ln(1 + e^{-y_n w^T x_n}) = E_{in}(w)$$

We know that likelihood is directly proportional to signal while E_{in} is inversely proportional to signal.

Hence, we can state that maximizing likelihood will minimize in sample error.

Task 1 HP 2

PAGE No.	
DATE	/ /

Task 1 , HP 2

→ $\nabla E_{in}(w(t)) = ?$

$$E_{in}(w) = \frac{1}{N} \sum_{n=1}^N \ln(1 + e^{-y_n w^T x_n})$$

$$\nabla E_{in} = \frac{1}{N} \sum_{n=1}^N \frac{1}{(1 + e^{-y_n w^T x_n})} \cdot (1 + e^{-y_n w^T x_n})'$$

$$= \frac{1}{N} \sum_{n=1}^N \frac{1}{(1 + e^{-y_n w^T x_n})} \left[0 + (-y_n x_n) e^{-y_n w^T x_n} \right]$$

$$= \frac{1}{N} \sum_{n=1}^N \frac{1}{1 + e^{-y_n w^T x_n}} \cdot (-y_n x_n e^{-y_n w^T x_n})$$

$$= -\frac{1}{N} \sum_{n=1}^N y_n x_n \left(\frac{e^{-y_n w^T x_n}}{1 + e^{-y_n w^T x_n}} \right)$$

$$\sigma(s) = \frac{e^s}{1 + e^s} = \frac{1}{1 + e^{-s}}$$

$$\therefore \nabla E_{in}(w) = -\frac{1}{N} \sum_{n=1}^N y_n x_n \sigma(y_n w^T x_n)$$

Task 4

PAGE No.	
DATE	/ /

	+1	-1
+1	0	c_a
-1	c_r	0

a) i) Cost Accept = $(1 - g(n)) c_a$

complete classificatⁿ for Accept
is True positive and False positive
= $g(n) \times TP + (1 - g(n)) \times FP$

$$= g(n) \times 0 + (1 - g(n)) \times c_a = \underline{\underline{(1 - g(n)) c_a}}$$

ii) Similarly

$$\begin{aligned} \text{Cost Reject} &= g(n) \times (r + (1 - g(n)) \times 0) \\ &= \underline{\underline{g(n) \cdot (r)}} \end{aligned}$$

b) Threshold is κ , and for the system to be fair Cost Accept = Cost Reject

$$(1 - g(n)) \cdot c_a = g(n) \cdot (r)$$

$$\left(\frac{1}{g(n)} - 1 \right) = \frac{cr}{c_a}$$

$$\frac{1}{g(n)} = \frac{r + c_a}{c_a} \Rightarrow g(n) = \frac{c_a}{r + c_a} = \kappa$$

c) Super market, $\alpha = 1$, $\gamma = 10$

$$K = \frac{1}{1+10} = \frac{1}{11}$$

Super market will easily accept fingerprints of the customer

CIA, $\alpha = 1000$, $\gamma = 1$

$$K = \frac{1000}{1000+1} = \frac{1000}{1001}$$

CIA will reject almost every fingerprint until it is very confident about the agent

Citations:

- References from Abu-Mostafa Yaser Video lectures on youtube.
- Discussions with Arielle Shnayder and Mahima Suresh.
- Code snippets from Mahima Suresh.