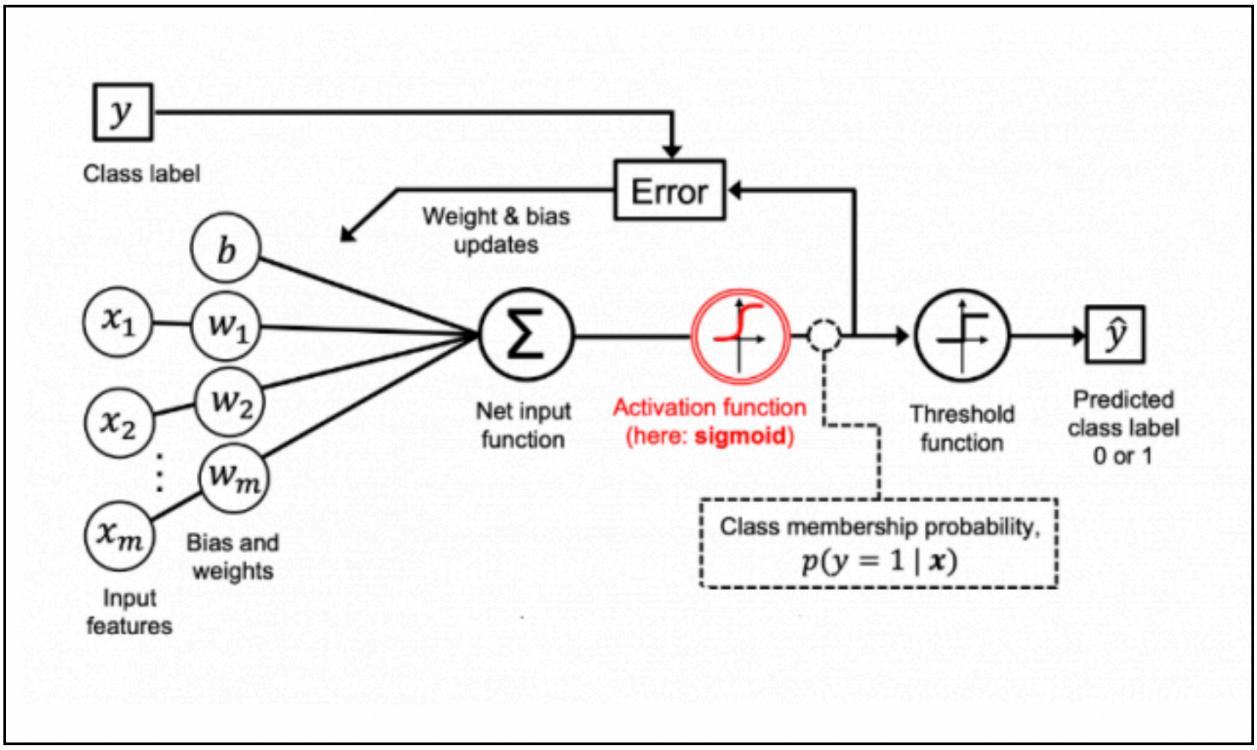
Problem 1. (30 points)

Consider a logistic regression problem where $\mathcal{X} = \mathbb{R}^d$ and $\mathcal{Y} = \{0, 1\}$. Derive the weight update rule that maximizes the conditional likelihood assuming that a data set $\mathcal{D} = \{(\boldsymbol{x} *_i, y_* i)\}_{i=1}^n$ is given. Add explanation of each step.

What is Logistic Regression ?

Logistic Regression is used for classification and prediction analysis tasks. It estimates the probability of an event, which is always between 0 and 1 , unlike Linear Regression which predicts a continuous dependent variable like “price” of a house .



Sigmoid Function in Logistic Regression

Logistic Regression uses the sigmoid function for predicting the probability - which is an S shaped curve which transforms any value to a number between 0 and 1 . Sigmoid function can be defined by :

$$\sigma(x) = \frac{1}{1 + \exp^{-x}}$$

Definitions used in the Answer

- **X is the design matrix** which represents the input variables and features. where m is the number of features and n is the number of training examples.

$$X = \begin{bmatrix} x_1 & \dots & x_{1m} \\ x_2 & \dots & x_{2m} \\ \dots & \dots & \dots \\ x_N & \dots & x_{Nm} \end{bmatrix}$$

- **W is the weights** - parameters for which an algorithm tries to find the optimal values to minimize error bw predicted and actual values. W = (w0, w1,…wm)
- **w*** is the optimal weight we want
- α **- it is a hyper parameter called learning rate** being used in the Gradient Descent algorithm. In layman language it is called “step size” used to reach the next point .
- **Y is the target variable**
- $\hat{y}^{(i)}$ **is the predicted value**

Logistic Regression Model

Using conditional likelihood function, our Logistic Regression model can be described as :

$$p(y_i = 1 \mid x_i, w) = \hat{y} = \sigma(a) = \frac{1}{1 + \exp^{-a}} \qquad \text{..... (1)}$$

where, $a = w^T x_i$

Now that we have the logistic Regression Model , **our task is to get the best weights w** . This can be done by maximising the likelihood . We call our Likelihood function as the **objective function.** And we know value of target variable can lie between 0 and 1 . So we get -

$$\begin{aligned} p(y_i = 0 \mid x_i) &= 1 - p(y_i = 1 \mid x_i) \\ &= 1 - \sigma(a) \qquad \text{.... (2)} \end{aligned}$$

Using eq 1 and eq 2 ,

$$p(y \mid x_i) = \sigma(a)^{y_i} . (1 - \sigma(a))^{1-y_i}$$

Objective Function

We call our Likelihood function as the objective function. **Our aim is to maximum the value of objective function.** And we know value of target variable can lie between 0 and 1 . So we get objective function as below

$$p(y \mid x_i) = \sigma(a)^{y_i} . (1 - \sigma(a))^{1-y_i}$$

Since our input is a matrix X, we know every training example is an independant event so we use x_i to indicate ith example.

Likelihood Function can be defined by :

$$\text{Likelihood } L(w) = \prod_{i=1}^{n} \sigma(a)^{y_i} . (1 - \sigma(a))^{1-y_i} \qquad \dots (3)$$

Maximising the likelihood (by maximising we mean to reducing the distance of 2 distributions to the least)

$$w^* = \arg \textbf{max} \prod_{i=1}^{n} \sigma(a)^{y_i} . (1 - \sigma(a))^{1-y_i} \qquad \dots (4)$$

The above equation L(w) has product of all the terms, we usually avoid products in computation because it costs more . Moreover **summations are easier to handle and numerically stable**. Therefore we convert eq (4) to summation . We will use "log" for this transformation. This will also reduce the magnitude of the likelihoods.

$$\log L(w) = \sum_{i=1}^{n} y_i \log \sigma(a) + (1 - y_i) \log \sigma(a)) \qquad \dots (5)$$

In optimisations, **we usually prefer Minimizing rather than maximising**.
Maximising the log likelihood is same as minimising the negative of log likelihood. We transform the objective func to **Negative Log Likelihood.** So we introduce a (-) & eq (5) becomes :

$$w^* = \text{arg } \textbf{min} \sum_{i=1}^{n} - y_i \log \sigma(a) + (1 - y_i)\log \sigma(a))$$

$$J(w) = - \sum_{i=1}^{n} y_i \log \sigma(a) + (1 - y_i)\log \sigma(a)) \quad \ldots(6)$$

(optimal weight) w^* = arg min J(w)

Weight Update Rule

For weight updates , we use the concept of Gradient Descent which will help in optimising the Objective Function . Gradient Descent is an iterative algorithm which we use to find the minimum of a function which can be differentiated.

Weight Update Rule is : $w_{NEW} = w_{OLD} - \alpha \Delta_w J(w_{OLD})$

we can represent $\Delta_w J(w) = \frac{\delta J(w)}{d x}$

Using Conditional Likelihood Function, lets modify cost funciton as :

$$J(w) = - \sum_{i=1}^{n} y \log a + (1 - y)\log a$$

where , a= $w^T x_i$

$$\hat{y} = a = \sigma(w^T x + b) = \sigma(z)$$

J(w) is the cost for all observations and L is loss for individual observations $J(w) = \sum_{i=1}^{n} L\text{oss}$

Lets derivate it using chain rule :

$$\frac{\delta L}{dw} = \frac{\delta L}{da} * \frac{\delta a}{dw}$$

$$= \frac{\delta L}{da} * \frac{\delta a}{dz} * \frac{\delta z}{dw}$$

lets compute all these 3 products individually -

$$\frac{\delta L}{da} = -y\frac{1}{a} + (1-y)\frac{1}{1-a} \qquad ...(7)$$

$$\frac{\delta a}{dz} = \frac{\delta(1+e^{-z})^{-1}}{dz} = e^{-z}/(1+e^{-z})^2$$

= ((1-a) * a^2) / (a)
= a (1- a) ... (8)

$$\frac{\delta z}{dw} = \frac{\delta}{dw}(w^T x) = x \qquad ...(9)$$

using T =1 for w1 and differentiating

Using eq (7) , (8) , (9) and solving it , we get :

$$\frac{\delta L}{dw} = x(a-y)$$

since $\hat{y} = a = \sigma(w^T x + b) = \sigma(z)$

$$\frac{\delta L}{dw} = x(\hat{y}-y)$$

$$\frac{\delta L}{dw} = \begin{cases} -(1-\hat{y}).x^i & \text{if y=1} \\ \hat{y}.x^i & \text{if y =0} \end{cases}$$

combining the both terms for value of y=0 and y=1 & using it in Cost Function , we get -

$$\frac{\delta}{dw} \text{J(w)} = -\frac{1}{n}[\sum_{i=1}^{n}(y^i - \hat{y}(x^i)]x$$

Problem 2. (30 points)

The sigmoid function is given as: $\sigma(a) = \frac{1}{1+e^{-a}}$ Solve the following questions.

1. (5 points) Compute $\frac{\delta\sigma(a)}{\delta w}$ when $a = w^T x$, where $w, x \in R^m$
2. (10 points) For logistic regression with target variable $y \in \{0, 1\}$. Show the posterior of y with given x and w, i.e. $P(y|x, w)$:
3. (15 points) Show the loss function for logistic regression and explain how do we learn w.

Problem 2, Part 1 -

Logistic Regression uses the sigmoid function for predicting the probability - which is an S shaped curve which transforms any value to a number between 0 and 1 . Sigmoid function can be defined by :

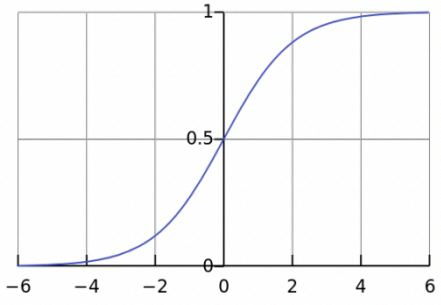


Figure 1: the sigmoid function

$$\sigma(a) = \frac{1}{1 + \exp^{-a}}$$

Derivation of Sigmoid is as follows :

$$\sigma'(a) = \frac{d}{dw}\sigma(a) \qquad , \quad \text{where } a = w^T . x$$

$$\frac{d}{dw}\sigma(w^T x) = \frac{d}{dw}\left(\frac{1}{1+e^{-w^T x}}\right)$$

Using Reciprocal Rule we know $: \frac{d}{dx}[1/u] = \frac{-d}{dx} v . (v)^{-2}$

$$\sigma'(a) = - \frac{d}{dw}(e^{-w^T x}) . (1+e^{-w^T x})^{-2}$$

$$\sigma'(a) = -(-e^{-w^T x})(1+e^{-w^T x})^{-2} . (-x)$$

$$\sigma'(a) = e^{-w^T x} . (1+e^{-w^T x})^{-2} . (x)$$

Rewriting the equation :

$$\sigma'(a) = \frac{1.e^{-w^T x}}{(1+e^{-w^T x}) . (1+e^{-w^T x})} . x$$

$$\sigma'(a) = \frac{1}{(1+e^{-w^T x})} . \frac{e^{-w^T x}}{(1+e^{-w^T x})} . x$$

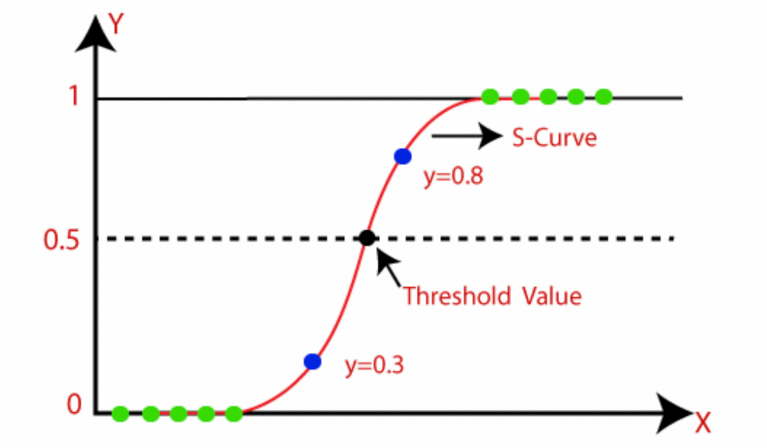
Rearranging the equation a bit to simplify it :

$$\sigma'(a) = \frac{1}{(1 + e^{-w^T x})} \cdot [\frac{e^{-w^T x}}{(1 + e^{-w^T x})}] \cdot x$$

$$\sigma'(a) = \sigma(a)(1 - \sigma(a)) \cdot x$$

Problem 2, Part 2 -

Logistic Regression is a discriminative model which can be used to predict labels and classifications, it models the posterior probability directly from training data. which is an S shaped curve which transforms any value to a number between 0 and 1 .



$$\sigma(a) = \frac{1}{1 + \exp^{-a}}$$

Posterior probability for y=1 :

$$p(y_i = 1 \mid x_i, w) = \sigma(a) = \frac{1}{1 + \exp^{-a}} \qquad \text{where, } a = w^T x_i$$

Since the value of y lies between 1 and 0 , we can calculate posterior probability for y=0 as :

$$p(y_i = 0 \mid x_i) = 1 - p(y_i = 1 \mid x_i)$$
$$= 1 - \sigma(a)$$

Since $\sigma(a) = \hat{y}$, we can say :

$$p(y_i = y \mid x_i, w) = \begin{cases} \hat{y} & \text{when y= 1} \\ 1 - \hat{y} & \text{when y= 0} \end{cases}$$

Combining these 2 equations we get the posterior probability for y

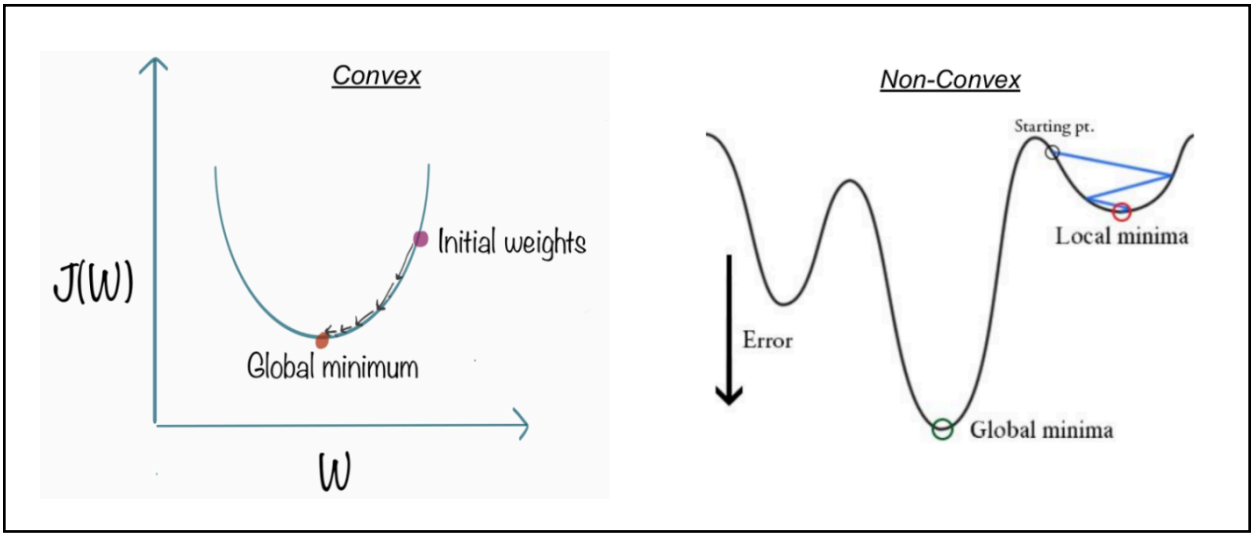
$$p(y_i = y \mid x_i) = (\hat{y})^y \,.\, (1 - \hat{y}^{1-y})$$

OR

$$p(y_i = y \mid x_i) = \sigma(a)^y \,.\, (1 - \sigma(a))^{1-y}$$

PROBLEM 2 - Part 3 - Show the loss function for logistic regression and explain how do we learn w

Maximum Squared Error does not work with Logistic Regression as it does with Linear Regression. The reason for this is, for Logistic Regression, MSE **does not produce a convex graph** , as a result of which its **difficult to find the global minimum**. This is the reason we use log graph because the output of Logisitc Regression is between 0 and 1 and log loss function helps us in heavily penalising the samples which are far from the desired value.



$$p(y_i = y \mid x_i, w) = \begin{cases} \hat{y} & \text{when y= 1} \\ 1 - \hat{y} & \text{when y= 0} \end{cases}$$

We can combine these 2 terms together for y=1 and y=0 to & scaling it for n samples , we get :

$$J(w) = -1/n \cdot \sum_{i=1}^{n} y_i \log \hat{y} + (1 - y_i)\log \hat{y})$$

NOW, for weight updates , we use the concept of Gradient Descent. Gradient Descent is an iterative algorithm which we use to find the minimum of a function which can be differentiated.

Weight Update Rule is : $w^{NEW} = w^{OLD} - \alpha \Delta_w J(w^{OLD})$

we can represent $\Delta_w J(w) = \frac{\delta J(w)}{d x}$

Algorithm :

1. Initialise the Weights Vector W with random or predefined values
2. Set the α **(learning rate)** for the weight updates
3. Repeat the above steps until convergence is met

What is “convergence” ?
It is a condition used to determine when the solution is reached . Its basically used to stop the optimization process.

Criteria for Convergence
1. Target value for Loss Function is reached
2. Maximum number of iterations have reached.
3. objective function is improved relatively between iterations.