**Deep Learning Assignment-01**

*Submitted by*

***Samahit Mahapatra***

**A124009**

**Master of Technology**

**In**

**Computer Science and Engineering**

******

**INTERNATIONAL INSTITUTE OF INFORMATION**

**TECHNOLOGY, BHUBANESWAR**

**ODISHA, 751003 INDIA**

**Solutions of Deep Learning Assignment 01**

**Q1. For a D -dimensional input vector, show that the optimal weights can be represented by the expression: l**

**w = (XTX)−1XTt**

**What is the possible estimation of w?**

**Solution:**

This expression represents the optimal weights (or coefficients) in a linear regression model when using the ordinary least squares (OLS) method.

Given:

Input matrix X of size N×D, where:

N = number of samples

D = number of features

Target vector t of size N×1, containing the true output values

Weight vector w of size D×1, which we want to estimate

Optimal Weight Solution:

The optimal weights w that minimizes the sum of squared errors (SSE) between the predicted outputs Xw and the true outputs t are given by the normal equation:

w = (XTX)−1XTt

Derivation:

1. The linear regression model predicts:

= Xw

1. The error (residual) is:

Error = t−Xw

1. The sum of squared errors (SSE) is:

SSE = (t−Xw)T (t−Xw)

1. To minimize SSE, taking the gradient w.r.t. w and set it to zero:

∇w SSE = −2XT(t−Xw) = 0

1. Solving for w, we get:

XTXw = XTt

1. If XTX is invertible, the solution is:

w=(XTX)−1XTt

Possible Estimation of w:

* If N≥D and X has full rank, then w is uniquely determined.
* If N<D (fewer samples than features), XTX is singular (non-invertible), and infinitely many solutions exist.

If XTX is nearly singular, small perturbations in X can lead to large changes in w. Regularization helps stabilize the solution.

Hence, the optimal weight vector w is estimated as:

w=(XTX)−1XTt

This is the best linear unbiased estimator (BLUE) under the Gauss-Markov assumptions. If the assumptions are violated, alternative methods (e.g., weighted least squares, robust regression) may be needed.

**Q2. OR Gate in single neural network**

**Solution:**

The OR gate is a fundamental logic gate that outputs 1 if at least one of its inputs is 1. We can model this using a single-layer perceptron (SLP), which is the simplest form of a neural network.

Truth Table for OR Gate:

|  |  |  |
| --- | --- | --- |
| X1 | X2 | Output(t) |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

Perceptron Model for OR Gate:

A single-layer perceptron consists of:

* Inputs: x1, x2 (binary values: 0 or 1)
* Weights: w1,w2
* Bias: b
* Activation Function:

Step function (Heaviside function):

Output =

where z =w1x1 + w2x2 + b

Finding Weights and Bias:

We need to find w1, w2, b such that the perceptron correctly classifies all OR gate inputs.

Possible Solution:

One valid set of weights and bias is:

w1 = 1

w2 = 1

b = −0.5

Verification:

* Input (0, 0): z = (1)(0)+(1)(0)−0.5= −0.5 Output = 0 (Correct)
* Input (0, 1):

z = (1)(0)+(1)(1)−0.5 = 0.5 Output = 1 (Correct)

* Input (1, 0):

z = (1)(1)+(1)(0)−0.5 = 0.5 Output = 1 (Correct)

* Input (1, 1):

z = (1)(1)+(1)(1)−0.5 = 1.5 Output = 1 (Correct)

Therefore, this perceptron correctly implements the OR gate

**Q3**. :**for given graph give the following solutions,**

1. **Generalized Point of Intersection for Shallow Neural Networks for input space parameterized by spherical coordinates θ and ϕ**

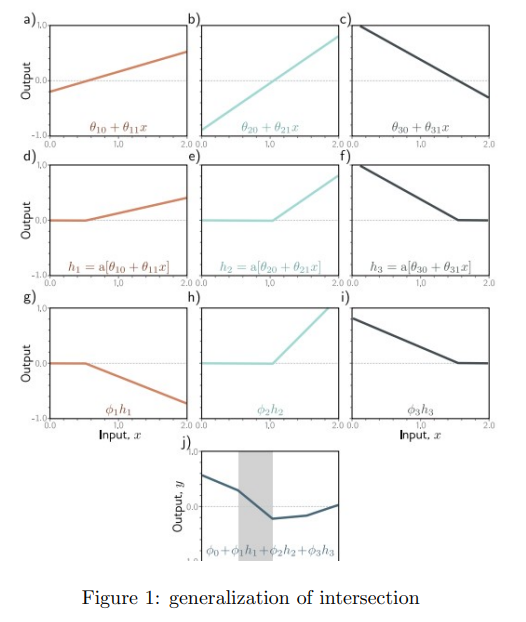
Solution:

Generalizing the Point of Intersection in Terms of θ and ϕ for Shallow Neural Networks

Step 1: Structure of a Shallow Neural Network

Let us consider a shallow neural network with:

* Input dimension: d
* Number of hidden neurons: m
* Activation function: σ
* Weight vectors: wi ∈
* Bias terms: bi ∈
* Output weights: ai ∈



The output of the network is given by:

Step 2: Weight Vectors in Angular Coordinates

In spherical coordinates:

Step 3: Decision Boundary Condition

For each neuron, the decision boundary satisfies:

which in spherical coordinates becomes:

Step 4: Intersection of Decision Boundaries

If two neurons intersect, we solve the system:

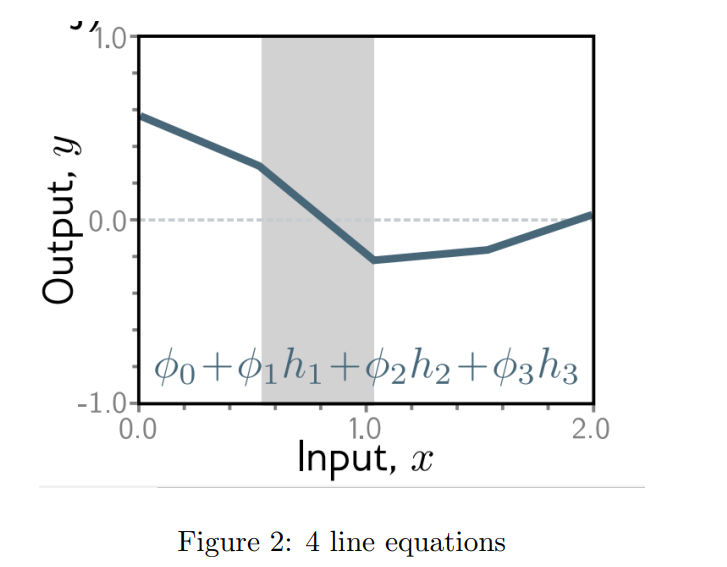
which translates to:

Step 5: General Solution

The point of intersection x can be computed by solving the linear system:

where A is the matrix formed by the weight directions in spherical coordinates and b is the bias vector.

1. Give the equation of 4-line segment in graph in, etc. for figure



Solution:

Let us consider a shallow neural network with three hidden units and ReLU activations. Let the output y of the network be defined by the following equation:

where each hidden unit is given by the ReLU activation function:

The output y(x) is composed of four linear segments, which can be written as:

Explicitly, the four-line segments are:

* First segment: y = ϕ0
* Second segment: y = ϕ0 + ϕ1(θ10 + θ11x)
* Third segment: y = ϕ0 + ϕ1(θ10 + θ11x) + ϕ2(θ20 + θ21x)
* Fourth segment: y = ϕ0 + ϕ1(θ10 + θ11x) + ϕ2(θ20 + θ21x) + ϕ3(θ30 + θ31x)

The activation thresholds x1, x2, and x3 where each hidden unit is activated are given by:

, for each neuron.

The output function combines all the active hidden units according to their weights and it is expressed in the above form.

**Q4. :Let x1, x2, . . . , xn be independent and identically distributed (i.i.d.) vectors from a multivariate normal distribution:**

**where µ is the unknown mean vector and Σ is the known covariance matrix.**

Solution:

Maximum Likelihood Estimate of Unknown Mean Vector

Let x1, x2, . . . , xn be independent and identically distributed (i.i.d.) vectors from a multivariate normal distribution:

where µ is the unknown mean vector and Σ is the known covariance matrix.

The probability density function (PDF) of xi is given by:

Likelihood Function:

Given the independence of the samples, the likelihood function is the product of the individual densities:

Taking the natural logarithm of the likelihood function (log-likelihood):

Maximizing the Log-Likelihood:

To find the MLE of µ, we differentiate log L(µ) with respect to µ and set the result to zero:

Simplifying:

Result: MLE of Mean Vector

Hence, the maximum likelihood estimate of the unknown mean vector is the sample mean.