

Assignment 3 - Designing Artificial Neural Networks for classification of LBW Cases from Scratch

Problem Statement:

Low Birth weight (LBW) acts as an indicator of sickness in newborn babies. LBW is closely associated with infant mortality as well as various health outcomes later in life. Various studies show strong correlation between maternal health during pregnancy and the child's birth weight. We use health indicators of pregnant women such as age, height, weight, community etc in order for early detection of potential LBW cases. This detection is treated as a classification problem between LBW and not-LBW classes

You have been provided with a Dataset consisting of data collected from a hospital which classifies the patient as cases of LBW and cases of non-LBW.

This is a design assignment that requires you to design a neural network from scratch using only numpy. You are expected to submit a working code on the dataset provided to you

Details on Implementation:

Use the python file given to you as the template

1. **Input** to the neural network is a data sample of 9 features (More about the dataset explained in the next section)
2. **Output** is a binary classification of LBW cases
3. You are free to choose any number of layers and any number of neurons in your network
4. Pick appropriate activation functions and loss functions for your implementation
For example :
Activation functions can be Sigmoid, Tanh, Relu, Softmax, etc.
Loss functions can be MSE, Binary Cross Entropy, etc.
5. A good design with out of the box implementation will fetch you higher points in the assignment
6. Please put appropriate comments in the beginning of your code explaining the parameters used such as -
 - a. number of layers
 - b. number of neurons per layer
 - c. dimensions of weight and bias matrices

- d. activation functions used in each layer
 - e. loss function used
 - f. any additional components added to the neural network
7. The dataset provided needs pre-processing to be done (More about it is given in the below section)

About the Dataset :

The Dataset consists of 10 columns

1. **Community** - Information about the patient's community
 - a. SC - 1
 - b. ST - 2
 - c. BC - 3
 - d. OC - 4
2. **Age** - Patients age in years
3. **Weight** - Weight in Kg during Trimester
4. **Delivery Phase** -
 - a. 1 - Before 37 weeks
 - b. 2 - After 42 weeks
5. **HB** - Haemoglobin content
6. **IFA** - determines if the patient took Folic acid or not
 - a. 1 - patient consumed Folic acid
 - b. 0 - patient did not consume Folic acid
7. **BP** - Blood Pressure during Trimester
8. **Education** - Educational Qualification of the patient on a scale of 0-10
9. **Residence** - indicates whether the patient is resident of the town or village in which the study was conducted (indicated by 1) or if the patient is a non-resident (indicated by 2)
10. **Result** - Label 1 indicates case of LBW, Label 0 indicates non LBW case

Split up of marks (Total 10 Marks)

- 1) 2 marks - training and testing accuracy above 85%
- 2) 2 marks - Data pre-processing
- 3) 4 marks - Complexity of Design
- 4) 1 mark - Well documented code with comments

5) 1 mark - Out of the box implementations (Doing more than what is mentioned)

Helpful Points

- The above explanation of the dataset is just for your reference and must be interpreted as factual data.
- **Account for the missing values in the dataset. Clean the dataset before any usage**
- Consider each column of the dataset as a feature that the neural network tries to learn
- Consider one portion of the dataset as training data and the other portion of the dataset as testing data. **You are free to choose any test-train split of your choice but keep it practical.** (Preferably use the standard 70-30 split i.e 70% of your data will be training data and 30% of your data will be testing data)

Important Instructions

- You are **not allowed** to use any built-in functions for the Implementation. Use of libraries such as Scikit is strictly only allowed for generating test-train splits. Usage of any additional libraries is strictly restricted to cleaning of the dataset.
- Pandas can be used for pre-processing the data
- **Every other implementation must be using Numpy only. No keras/ scikit/ Tensorflow code will be accepted.**