

Machine Learning Assignment-1 2019-20

SUBMITTED BY

Akshit Khanna	2017A7PS0023P
Jhaveri Ayush Rajesh	2017A7PS0215P
Vitthal Bhandari	2017A7PS0136P

1. Introduction

A multilayer perceptron (MLP) is a feedforward artificial neural network that generates a set of outputs from a set of inputs using non-linear activation functions. MLP is composed of several layers of input nodes connected as a directed graph between the input and output layers. MLP uses backpropagation for training the network by adjusting weights every iteration called epochs.

We have implemented a generalized MLP network with any number of layers with different types of activation functions and gradient descent algorithms to compare the accuracy of the methods over two types of problem.

The types of Activation functions used are Sigmoid and Tanh.

The types of gradient descent algorithms used are regular gradient descent and gradient descent with momentum.

The learning rate of the network is 0.05 and momentum rate is 0.9.

The error used is quadratic error.

2. Dataset

The two dataset consists of Binary Classification and Multiple Classification problems. Blood transfusion service center data set is taken for binary classification and Contraceptive method choice dataset is taken for multiple classification.

2.1 Binary Classification

Blood Transfusion dataset has adopted the donor database of Blood Transfusion Service Center in Hsin-Chu City in Taiwan. The dataset consists of 4 attributes and 1 binary output :-

- R (Recency - months since last donation) Numerical Value
- F (Frequency - total number of donations) Numerical Value
- M (Monetary - total blood donated in c.c.) Numerical Value
- T (Time - months since first donation) Numerical Value
- A binary variable representing whether he/she donated blood in March 2007 (1 stand for donating blood; 0 stands for not donating blood).

The dataset was cleaned as 2 attributes were totally correlated. Hence only 3 attributes were considered for training. The input values were normalized for better training of the network. The Train set consisted of 500 values and the test set consists of 248 values. The link of the dataset is

<https://archive.ics.uci.edu/ml/datasets/Blood+Transfusion+Service+Center>

The network used a hidden layer of 4 neurons and output layer of 2 neurons. The last is softmaxed for classification.

Results table on Test Set

Activation	Gradient Descent	Epochs	Accuracy
Sigmoid	Regular	10	98.38
	Momentum	10	98.78
	Regular	8	80.56
	Momentum	8	96.45
Tanh	Regular	10	100
	Momentum	10	100

2.2 Multiple Classification

Contraceptive Method Choice is a subset of the 1987 National Indonesia Contraceptive Prevalence Survey. The samples are married women who were either not pregnant or do not know if they were at the time of interview. The problem is to predict the current contraceptive method choice (no use, long-term methods, or short-term methods) of a woman based on her demographic and socio-economic characteristics.

The dataset consists of 9 attributes and 1 output class as follows :-

- Wife's age (numerical)
- Wife's education (categorical)
- Husband's education (categorical)
- Number of children ever born (numerical)
- Wife's religion (categorical)
- Wife's now working? (categorical)
- Husband's occupation (categorical)
- Standard-of-living index (categorical)
- Media exposure (categorical)
- Contraceptive method used (class attribute) 1=No-use, 2=Long-term, 3=Short-term

The dataset was used categorical values and converted them to numerical and then normalised the input data for better training. The train set consists of 900 samples and test set consists of 573 samples .

The link of the dataset is

<https://archive.ics.uci.edu/ml/datasets/Contraceptive+Method+Choice>

The network used a hidden layer of 5 neurons and output layer of 3 neurons. The output was softmaxed for classification.

Results table on Test Set

Activation	Gradient Descent	Epochs	Accuracy
Sigmoid	Regular	15	45.4
	Momentum	15	52.27
	Regular	20	79.9
	Momentum	20	81.46
Tanh	Regular	25	44.05
	Momentum	25	81.46
	Regular	35	81.4

The accuracy over test set is not able to exceed 81.5 percent accuracy after any number of epochs.

3.Design of the program

The interface of the program is as follows :-

```
Enter the choice of Dataset to be tested:
    1: Blood Transfusion Binary Classification
    2: Contraceptive Multi-Classification
1

Enter the number of layers in the network : 2
Enter the number of neurons in Layer 2 :4
Enter the number of neurons in Layer 3 :2

Enter the number of epochs to be run : 10

Enter your choice for gradient descent :
    1: Regular Gradient Descent
    2: Gradient Decent with Momentum
1

Enter the choice of activation function :
    1: Sigmoid
    2: Tanh
1
```

The structures used in the program are as follows :-

```
typedef struct
{
    int size ;
    double *weights, *prev_weights ;
} Neuron ;

typedef struct
{
    int num_neurons ;
    Neuron *arr_neuron ;
} Layer ;

typedef struct
{
    int num_layers, num_inputs ;
    Layer *arr_layer ;
} Network ;
```

Functions used in the program :-

1.

```
double * layer_output(Network *mlp, double *input_data_single, int
layer, int act)
```

This function is used to get output of any layer with or without applying activation to the calculated output. It is used to calculate the result, error as well as delta values for adjusting weights.

2.

```
double avg_quad_err(Network *mlp, double **input_data, double
**output_data, int num_data)
```

This function is used to calculate the average quadratic error over all the training samples.

3.

```
double * delta_calc(Network *mlp, int layer, double *input_data_single,
double *output_data_single)
```

This function is used to calculate the delta for each layer required in the gradient descent step of backpropagation.

4.

```
void grad_desc_momentum(Network *mlp, int layer, double
*input_data_single, double *output_data_single, double *delta, double
*prev_layer_act)
```

```
void grad_desc_regular(Network *mlp, int layer, double
*input_data_single, double *output_data_single, double *delta, double
*prev_layer_act)
```

Both these functions are the variations of gradient descent applied based on user inputs for training the network.

5.

```
double activation(double x) ;
double deriv_act(double x) ;
```

These functions are used to apply activation to layer output and derivative of the activation function respectively.

6.

```
void act_softmax(double *output, int num_out)
```

This function is used to apply softmax activation to the output of the final layer for classification.

7.

```
void print_acc(Network *mlp, double **input_data, double **output_data,
int num_data, int index)
```

This function is used to calculate the accuracy of the network on Test set .

4. Appendix

The code consists of a C file and a C header file. The link to the code with dataset is at the following drive link :-

<https://drive.google.com/open?id=1TFJt3A8ok9zYMzjlCr0M8EqT6ydSXDBM>