Assignment: Homework 5

How to Hand It In

1. Put all your solutions in one folder. Compress this folder and name it <firstname>_<lastname>_HW5.zip. For example, “Zhiyao_Duan_HW5.zip”.
2. Make sure you have a report to show graphs and explain your answers, in addition to source code.
3. Submit to the corresponding entry on Blackboard.

When to Hand It In

It is due at 11:59 PM on the date specified on the course calendar. **Late assignments will receive a 20% deduction of the full grade each day.**

Problems (10 points in total)

Please note a notation difference in this assignment from lecture slides: here vectors are row vectors. Adapting to notation differences is an important skill in machine learning.

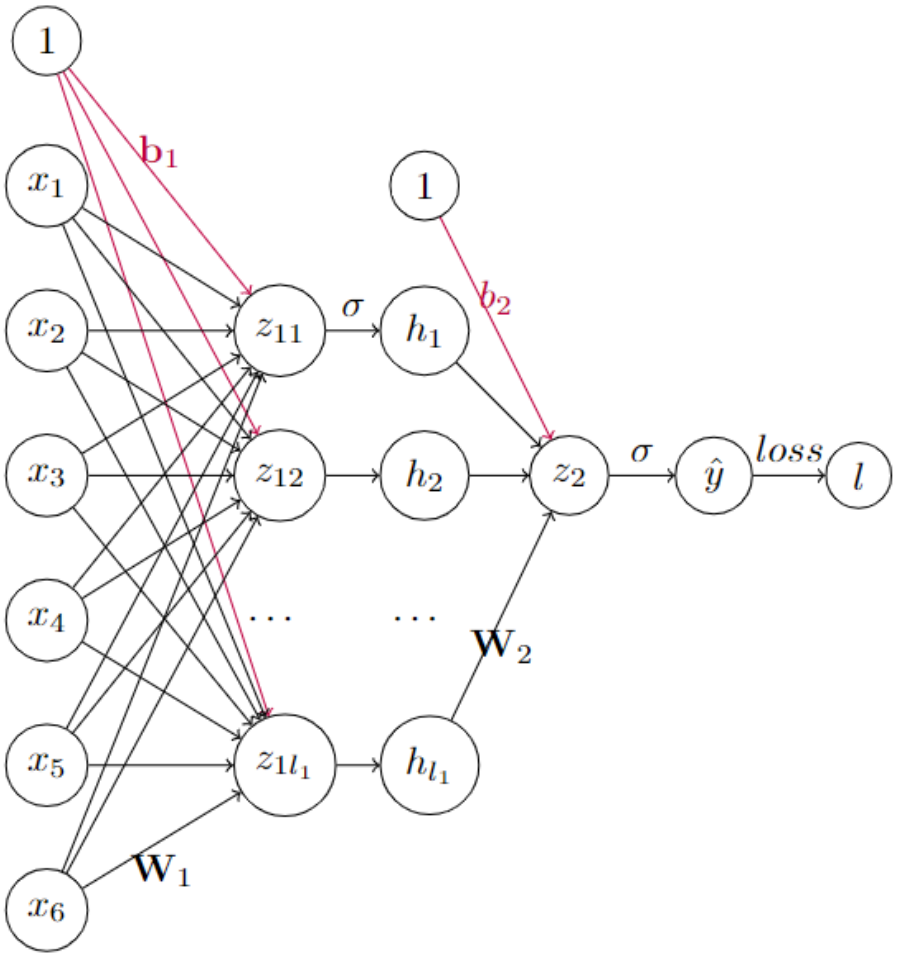

Figure 1. Structure of MLP network.

1. (10 points) Implement a multi-layer perceptron (MLP) classifier and the backpropagation algorithm to train it on the "Maternal Health Risk Data" dataset. The structure of the network is shown in Figure 1. The dataset was collected from medical institutions to identify health conditions that pose the greatest risks during pregnancy. The dataset can be accessed at https://www.kaggle.com/datasets/drmbsharma/maternal-health-risk-data-set.

   Denote the input of the network as $x = [x_1, x_2, x_3, x_4, x_5, x_6] \in \mathbb{R}^{1\times 6}$. According to the definition of MLP, we know that the linear projection and the sigmoid activation of the first layer can be written as:

$$z_{1j} = b_{1j} + \sum_{i=1}^{6} x_i w_{(1)ij}, \qquad j = \{1, 2, \dots, l_1\}$$
$$h_j = \sigma(z_{1j}), \qquad j = \{1, 2, \dots, l_1\}$$

   Here, $z_1 \in \mathbb{R}^{1\times l_1}$, $h \in \mathbb{R}^{1\times l_1}$, $W_{(1)} \in \mathbb{R}^{6\times l_1}$, $b_1 \in \mathbb{R}^{1\times l_1}$, and $w_{(1)ij}$ denotes the entry at row i and column j in $W_{(1)}$. If we write the equations as matrices, it will be:

$$z_1 = b_1 + xW_{(1)}$$
$$h = \sigma(z_1)$$

   Then, the second layer calculates the prediction output:

$$z_2 = b_2 + \sum_{j=1}^{l_1} h_j w_{(2)j}$$
$$\hat{y} = \sigma(z_2)$$

   Here, $z_2 \in \mathbb{R}$, $\hat{y} \in \mathbb{R}$, $W_{(2)} \in \mathbb{R}^{l_1\times 1}$, $b_2 \in \mathbb{R}$. The matrix format is:

$$z_2 = b_2 + hW_{(2)}$$
$$\hat{y} = \sigma(z_2)$$

   Finally, we calculate the cross-entropy loss for binary classification:

$$l = -(y \log \hat{y} + (1 - y) \log(1 - \hat{y}))$$

   Theory Questions: We are going to use the chain rule to derive the partial derivatives of each component to prepare for backpropagation. Please use word/latex to format the equations, or you can take pictures of your handwritten derivations. Please combine them into a pdf file and make them readable.

a. (0.5 points) What is $\frac{\partial l}{\partial \hat{y}}$?

b. (0.5 points) What is $\frac{\partial l}{\partial z_2}$? Please express in terms of $\frac{\partial l}{\partial \hat{y}}$ and z_2.

c. (0.5 points) What is $\frac{\partial l}{\partial b_2}$?

d. (0.5 points) What is $\frac{\partial l}{\partial w_{(2)j}}$ for the entry j in $W_{(2)}$? What is $\frac{\partial l}{\partial W_{(2)}}$? Please express in terms of $\frac{\partial l}{\partial z_2}$ and h.

e. (0.5 points) What is $\frac{\partial l}{\partial h_j}$ for the entry j in h? What is $\frac{\partial l}{\partial h}$? Please express in terms of $\frac{\partial l}{\partial z_2}$ and $W_{(2)}$.

f. (0.5 points) What is $\frac{\partial l}{\partial z_1}$ in terms of $\frac{\partial l}{\partial h}$ and h?

g. (1 point) What is $\frac{\partial l}{\partial b_1}$ in terms of $\frac{\partial l}{\partial z_1}$? What is $\frac{\partial l}{\partial W_{(1)}}$ in terms of $\frac{\partial l}{\partial z_1}$ and x ?

Programing Questions: Skeleton code is provided which includes data preprocessing and the basic structure of the code. You need to fill the TODOs in the skeleton code.

a. (1 point) Implement the forward pass of a two-layer MLP (i.e., with a single hidden layer) classifier that uses sigmoid activation for both the hidden layer and output layer. The output layer has a single node. - Randomly initialize weights and biases in the range of [-1, 1).

b. (2 points) Implement the backpropagation algorithm using cross-entropy loss.

c. (1.5 points) Implement a training function. Train your model on the training set with default hyperparameters provided in the skeleton code. Use the entire training set as a batch, i.e., computing the gradient on the entire dataset in each iteration. Plot the error versus the number of epochs and report classification accuracies on subsets.
   Note: When we have a very large dataset, it would be computationally expensive to compute the gradient on the entire dataset in each iteration, and people often compute a stochastic gradient using a small subset (called a batch) in each iteration. You are welcome to try it, but there is no extra credit for this exploration.

d. (1.5 points) Experiment with hyperparameters such as number of iterations, learning rate, and hidden layer size to evaluate how accuracy is affected. Use validation data to identify the best set of hyperparameters. If you are not confident about your own implementation, you may use sklearn.neural_network.MLPClassifier instead for this question. Record the changes made, the reasoning behind the changes, and the results in your report.