Homework Assignment 4

Part-1: Math and Concepts

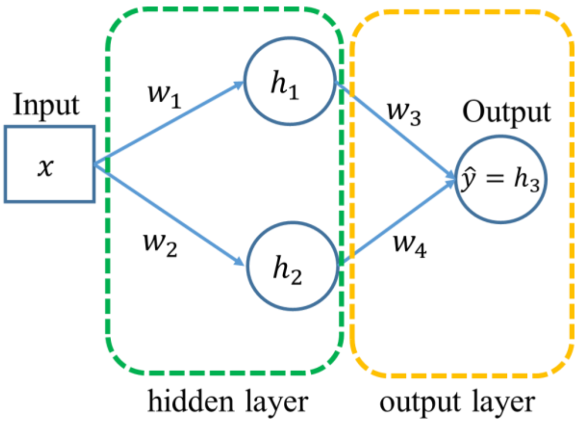
1. Handle class-imbalance

We have a class-imbalanced dataset, and the task is to build a classifier on this dataset. From the perspective of PDF, there are two types/scenarios of class-imbalance (see lecture notes). Now, assume we have scenario-1.

(1) Why do we use weighted-accuracy to measure the performance of a classifier?

(2) When class-weight is not an option for a classifier, what other options do we have to handle class-imbalance?

2. Backpropagation In A Neural Network



$$h_1 = f_1(w_1 x + b_1)$$

$$h_2 = f_2(w_2 x + b_2)$$

$$\hat{y} = f_3(w_3 h_1 + w_4 h_2 + b_3)$$

$$f_n' = \frac{\partial f_n(v)}{\partial v}, \text{ n} = 1,2,3$$

$x, w_1, w_2, w_3, w_4, b_1, b_2, b_3, h_1, h_2, h_3$ are scalars

(1) Compute the derivatives of the loss L with respect to parameters and input, assuming $\frac{\partial L}{\partial h_3}$ is known.

$\frac{\partial L}{\partial w_1}$

$\frac{\partial L}{\partial w_2}$

$\frac{\partial L}{\partial w_3}$

$\frac{\partial L}{\partial w_4}$

$\frac{\partial L}{\partial b_1}$

$\frac{\partial L}{\partial b_2}$

$\frac{\partial L}{\partial b_3}$

$\frac{\partial L}{\partial x}$

3. Computational Graph

$h = 2x + 1$

$z = x^2$

$$y = \frac{1}{1 + e^{-h}}$$

(1) Draw the computational graph based on the above three equations

(2) What is $\frac{\partial y}{\partial z}$ from the graph?

4. Classification using Neural Networks

Assume in an application, there are two classes. We can use binary-cross-entropy loss or cross-entropy loss.

Assume we have two networks, Net-1 and Net-2.

Net-1 is trained with binary-cross-entropy loss

Net-2 is trained with cross-entropy loss

The structure of Net-1 is almost the same as that of Net-2, except the output layer.

What is the difference in the output layers of Net-1 and Net-2?

5. Input and Output Normalization for a Neural Network

Usually, we need to apply normalization/standardization to the inputs for classification and regression tasks. For example, if the input is an image, then every pixel value is divided by 255, so that the pixel values of the normalized image are in the range of 0 to 1. Input normalization facilitates the convergence of training algorithms.

We may also need to apply normalization to the output. Assume the input is an image of a person, the output vector has two components, $\hat{y}_{(1)}$ and $\hat{y}_{(2)}$: $\hat{y}_{(1)}$ is the monthly income (in the range of 0 to 10,000), and $\hat{y}_{(2)}$ is the age (in the range of 0 to 100). The MSE loss for a single data sample is

$$L = (\hat{y}_{(1)} - y_{(1)})^2 + (\hat{y}_{(2)} - y_{(2)})^2$$

where $y_{(1)}$ and $y_{(2)}$ are ground truth values of an input data sample.

Question: Is output normalization necessary or not for this regression task? Why?

If it is necessary, what normalization can be applied?

6. Activation Functions for Regression (for Graduate Students)

Neural networks can be used for regression. To model nonlinear input-output relationship, a neural network needs nonlinear activation functions in the hidden layers. Usually, the output layer does not need nonlinear activation functions. However, sometimes, there are requirements for outputs. For example, if the output is the sale price of a house, then the output should be nonnegative.

Assume $\boldsymbol{z}$ is the scalar output of a network, and the network does not have nonlinear activation function in the output layer. Now, there is some requirement for output, and you decide to add a nonlinear activation function.

You design nonlinear activation functions for three different requirements:

(1) the final output y should be nonnegative ($y \geq 0$), then what is the activation function $y = f(z)$?

(2) the final output y should be nonpositive ($y \leq 0$), then what is the activation function $y = f(z)$?

(3) the final output y should be $a \leq y \leq b$, then what is the activation function $y = f(z)$?

(4) Could we add the nonlinear activation function to the network **after** the network is trained without the nonlinear activation function?

7. Normalization inside a Neural Network (for Graduate Students)

It is often necessary to normalize the input (and output if the task is regression) for training a neural network. To facilitate the convergence of training a deep neural network, it is necessary to normalize the output of each layer of a neural network.

Read the paper https://arxiv.org/abs/1803.08494 and answer the questions:

(1) Batch Normalization will be highly unstable if batch_size is very small. Why?

(2) Why is Layer Normalization independent of batch_size?

Grading

The number of points for each question/task

	Undergraduate Student	Graduate Student
1. Handle class-imbalance	5	5
2. Backpropagation	10	10
3. Computational Graph	10	10
4. Classification	5	5
5. Output Normalization	10	10
6. Activations for Regression	N.A. (extra 5 points)	5
7. Normalization inside Network	N.A. (extra 5 points)	5
H3P2T1 (classification)	30	25
H3P2T2 (regression)	30	25

For programming tasks, H3P2T1 and H3P2T2, you can use either Keras or Pytorch.

If you use both Keras and Pytorch, you will get additional 10 points (if there are no bugs in your code).

I highly suggest that you use both packages, so that you can put this line in your resume for a job:

“familiar with deep learning packages (Keras and Pytorch)”