CS 422: Data Mining

Department of Computer Science
Illinois Institute of Technology
Vijay K. Gurbani, Ph.D.

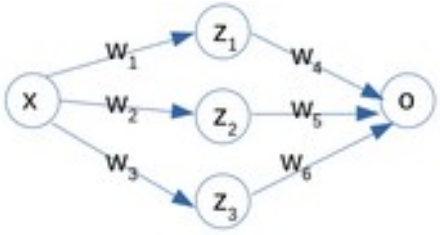
Fall 2022: Homework 7

Due date: Monday, November 21, 2022, 11:59:59 PM Chicago Time

Please read all of the parts of the homework carefully before attempting any question. If you detect any ambiguities in the instructions, please let me know right away instead of waiting until after the homework has been graded.

1. Exercises (Please submit a PDF file containing answers to these questions. Any other file format will lead to a loss of 0.5 point. Non-PDF files that cannot be opened by the TAs will lead to a loss of all points.)

1.1 (2 points) Consider the neural network shown in the figure below.



The weight matrix, W, is: $[1, 1, -1, 0.5, 1, 2]^T$. Assume that the hidden layer uses RelU and the output layer uses Sigmoid activation function. Assume squared error. The input x = 4, and the output y = 0.

Recall that RelU is defined as a function, f(x) = max(0, x). Its derivative is $f'(x) = \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{otherwise} \end{cases}$

Squared error is defined as $E(y, \hat{y}) = (y - \hat{y})^2$.

The partial derivative of E with respect to $\hat{y}$ is $-2(y - \hat{y})$.

Using this information, answer the following questions:
(Show all work, and all answers should be rounded to 3 decimal places OR POINTS WILL BE TAKEN OFF!)
(a) Use forward propagation to compute the predicted output.
(b) What is the loss or error value?
(c) Using backpropagation, compute the gradient of the weight vector, that is, compute the partial derivative of the error with respect to all of the weights.
(d) Using a learning rate of 1.0, compute new weights from the gradient. With the new weights, use forward propagation to compute the new predicted output, and the loss (error).
(e) Comment on the difference between the loss values you observe in (b) and (d).

1.2 [1 point] Tan Chapter 4, questions 14, 15.

1.3 [1 point] Consider a dataset that has 8 predictors. You train a neural network with 3 hidden layers and an output layer that predicts a continuous value (a regression problem). The first hidden layer has 16 neurons, the second has 8 neurons, and the third has 4 neurons. In this network, how many total parameters will you have?

2 Practicum problems

2.1 Feed Forward Neural Networks

Please use the template file (Problem-2-1-Template.Rmd) provided with the homework to bootstrap your code.

The dataset available for this problem on Blackboard (wifi_localization.csv) is described in detail at https://archive.ics.uci.edu/ml/datasets/Wireless+Indoor+Localization. It contains signal for 7 WiFi access points, which are used to predict a user's location in one of four rooms. The response variable, therefore, is a room number ranging from 1-4. Using this dataset, you will train models to predict the user's location. 80% of the data will be used for training and 20% for testing. Use the seed value of **1122** before you divide the data into a train and test set.

(a) [1 point] Train a decision tree model on the training set, and fit the test set to the model. Print the confusion matrix, and summarize the confusion matrix in the format shown below:

```
Decision Tree Model
   Overall accuarcy: XX.XX
   Sensitivity Class 1: XX.XX  Class 2: XX.XX
               Class 3: XX.XX  Class 4: XX.XX
   Specificity Class 1: XX.XX  Class 2: XX.XX
               Class 3: XX.XX  Class 4: XX.XX
   PPV         Class 1: XX.XX  Class 2: XX.XX
               Class 3: XX.XX  Class 4: XX.XX
   Bal. Acc.   Class 1: XX.XX  Class 2: XX.XX
               Class 3: XX.XX  Class 4: XX.XX
```

You will now train a neural network to predict the user's location in one of the four rooms. For the output layer, use the softmax activation function. Loss is measured in terms of accuracy, using 'categorical_crossentropy' on the 'adam' optimizer (see the compile() function). For the fit() method, set the batch_size to 32 and the validation_split to 0.20.

(b) [2 points] In the first stage, use one hidden layer, and only one neuron in the hidden layer. Use the relu activation function for the neuron in the hidden layer. Train the network for 100 epochs. Fit the held out test dataset to the model and create (and print) the confusion matrix. Based on the training and the fitting of the model, answer the following questions:

(i) Print out the loss and accuracy from fitting the test dataset. The output should be in the following form:

```
For one neuron in hidden layer, loss: X.XX, Accuracy: X.XX
```

(ii) Look at the plot for accuracy; why do you think the accuracy is low? (1-3 sentences.)

(iii) Examine the predicted labels, and print these out. What pattern do you see in the predicted labels?

(iii) Is the bias of the model high or low or just about right? Why?

(iv) Do you think we will get better results if we increase the training to 200 epochs? Why?

Before starting the second stage below, either use a new variable to store the model, or null out the variable that stored the model in (a) if you want to reuse that variable. The reason is that if you don't null it out, the model in (b) will have residual information left over from (a) and your results will not be quite accurate.

(c) [2 points] In the second stage, improve your neural network such that it predicts with high accuracy. Start increasing the number of neurons in the hidden layer until the accuracy improves (should be in the high 0.90's). Create the best model you can with one hidden layer by varying the number of neurons in the hidden layer. Fit

the held out test dataset to the model. Based on the training and the fitting of the model, answer the following questions:

(i) Print out the loss and accuracy from fitting the test dataset. The output should be in the following form:

```
Best model has XX neurons in the hidden layer.
In this model, loss: X.XX, Accuracy: X.XX
```

(ii) Is the bias of the model high or low or just about right? Why?

(iii) Based on the plots of accuracy and validation, at what epoch do you think we should stop the training to minimize over-fitting?

(d) [1 point] For the best model in (c), print the confusion matrix, and summarize the confusion matrix in the format shown below:

```
Best Neural Network Model
   Overall accuarcy: XX.XX
   Sensitivity Class 1: XX.XX  Class 2: XX.XX
               Class 3: XX.XX  Class 4: XX.XX
   Specificity Class 1: XX.XX  Class 2: XX.XX
               Class 3: XX.XX  Class 4: XX.XX
   PPV         Class 1: XX.XX  Class 2: XX.XX
               Class 3: XX.XX  Class 4: XX.XX
   Bal. Acc.   Class 1: XX.XX  Class 2: XX.XX
               Class 3: XX.XX  Class 4: XX.XX
```

(i) Compare the above output to similar output of a decision tree model in (a). Comment on what you observe.

(ii) If you had to deploy one of these two models in production, which one would you choose and why?