Feed-Forward Neural Networks II

COSC 410: Applied Machine Learning
Fall 2025
Prof. Forrest Davis

October 21, 2025

Warm-up

1. Discuss with your neighbor the highlight of your fall break.
2. Consider the neural network in Figure 1, each node (A2, A3, A4) uses a sigmoid activation function. Calculate A2, A3, and A4.

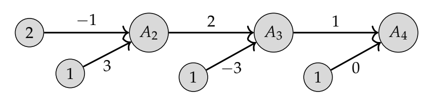

Figure 1: Three-layer neural network

Logistics

- Codelet 4 on Feed-Forward Neural Networks out, due Friday Oct 31
- Midterm Exam grades posted

Learning Objectives

- Understand neural network backward computation
- Give the motivation for backpropagation
- Describe some techniques for over-fitting with neural networks

Summary: We complete the basic neural network pipeline with backward and backpropagation and talk a bit about over-fitting in neural networks.

Backward Computation with Feed-Forward Neural Networks

WHILE THE FORWARD COMPUTATION computes the output (or prediction) of a neural network, the **backward** computation is the calculation of the gradient of the weights based on a cost (or loss) function. A naive implementation of this can be computationally expensive. Let's see why using Figure 2.

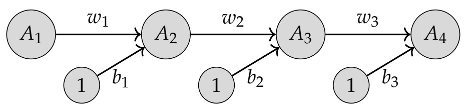

Figure 2: Schematized three layer neural network

For a regression problem, we would compare the output (A4; e.g., 0.6) and compare it to some continuous target value (e.g., 1.0). We use mean-squared error to quantify the quality of that output (in this case, that it is an error of 0.4^2).

Let's quantify the effect w_3 has on this error, $\frac{\partial \text{MSE}}{\partial w_3}$. It is helpful to add a variable representing a step of the computation: $z_4 = w_3 a_3 + b_3$. Then,

$$\frac{\partial \text{MSE}}{\partial w_3} = \frac{\partial z_4}{\partial w_3} \frac{\partial A_4}{\partial z_4} \frac{\partial \text{MSE}}{\partial A_4} \tag{1}$$

Question

Give the expression (similar to the one in Equation 1) for the following:

1. $\frac{\partial \text{MSE}}{\partial w_2}$

2. $\frac{\partial \text{MSE}}{\partial w_1}$

Backpropagation

THERE IS A MORE COMPUTATIONALLY effective way of calculating backward, called **backpropagation**. Let's consider ways we could calculate the loss for Figure 2.

- Each weight randomly chosen
- From the first weight (w_1) to the last weight (w_3)
- From the last weight (w_3) to the first weight (w_1)

Question

Which one is faster and why? What information has to be stored?

Application Considerations

IN USING NEURAL NETWORKS, it is important to keep in mind issues of over-fitting. In essence, over-fitting occurs when a model does well on training data but fails to generalize to data from the same distribution (e.g., validation data). This is often because a model is more complex than the underlying tasks requires.

Question

You can diagnosis over-fitting by observing the loss for training and validation data during training. How will training loss compare to validation loss if the model is over-fitting?

There are some standard ways to diminish over-fitting in neural networks:

- Try a simpler model
  - Less layers
  - Less nodes per layer
- Early-stopping
  - Stop training early if the validation loss stays the same (or goes higher or only decreases by some small amount)
- Dilution and Dropout
  - Randomly weaken/remove connections or remove whole nodes from the network during part of training

Practice Problems

1. You want to design a neural network to predict the amount of sunshine tomorrow based on the last three day's max temperature, min temperature, amount of rain fall, pressure, and humidity level. What will be the input to your neural network (how many features will there be).

Before Next Class

- Reading and pre-class quiz
- Look at Codelet 4