CO553 - Introduction to Machine Learning: Neural networks

Prepared by Marek Rei

Autumn 2020/2021

1 Questions

Here is a set of various questions to improve your understanding of neural networks.

1. Remember that coursework 2 part 1 covers much of the neural network material – feedforward networks, activation functions, loss functions, backpropagation, gradient descent and data normalisation. Even if your teammates implemented some of it, make sure you go over all of it and understand how it works.

2. Given a network with the following output activations, write 1) what type of task are we likely solving with this network, 2) what would be the most common loss function for this activation and task combination, and 3) one example of this type of task.

   (a) One neuron with sigmoid
   Answer: Binary classification. Binary crossentropy. Classifying whether an image contains a cat or not.

   (b) Multiple neurons with softmax
   Answer: Multi-class classification. Categorical crossentropy. Detecting the native language of a speaker based on their accent.

   (c) One neuron with linear output
   Answer: Regression. Mean squared error. Predicting how many goals a team will score in the next game.

   (d) Multiple neurons with sigmoid
   Answer: Multi-label classification. One binary crossentropy for each sigmoid output neuron. Predicting which products a given customer will buy.

3. Write out the formula for linear regression with two input features.
   Answer: $\hat{y}^{(i)} = w_1 x_1^{(i)} + w_2 x_2^{(i)} + b$

4. Given the linear regression from the previous question, derive the gradient descent updates for each of the parameters.
   Answer:
   $w_1 \leftarrow w_1 - \alpha \cdot 2 \cdot \sum_{i=1}^{N} (\hat{y}^{(i)} - y^{(i)}) x_1^{(i)}$
   $w_2 \leftarrow w_2 - \alpha \cdot 2 \cdot \sum_{i=1}^{N} (\hat{y}^{(i)} - y^{(i)}) x_2^{(i)}$
   $b \leftarrow b - \alpha \cdot 2 \cdot \sum_{i=1}^{N} (\hat{y}^{(i)} - y^{(i)})$
   Note that including the constant $\frac{1}{2}$ in the MSE loss is optional and can be thought of as part of the chosen learning rate. We didn't use it here.

5. You have a multi-layer neural network for regression. Input layer with 2 neurons; hidden layer with 2 neurons and sigmoid activation; output layer with 1 neuron and linear activation.

   The weights between the input layer and hidden layer look like this:

$$W_1 = \begin{bmatrix} 0.1 & 0.2 \\ 0.3 & 0.4 \end{bmatrix}$$

$$b_1 = \begin{bmatrix} 0.2 & -0.1 \end{bmatrix}$$

The weights between the hidden layer and the output layer look like this:

$$W_2 = \begin{bmatrix} 0.5 \\ 0.6 \end{bmatrix}$$

$$b_2 = \begin{bmatrix} 0.1 \end{bmatrix}$$

Given the following input:

$$X = \begin{bmatrix} 0.2 & -0.4 \end{bmatrix}$$

calculate the output of the network.

Answer: $\hat{y} = 0.629622$

6. Calculate the mean squared error (MSE) for the network in the previous question, given the target value $y = 1.5$.

   Answer: $MSE = 0.757558$

7. Calculate the gradients for the weight matrix W_1 and bias vector b_1 of that network.

   Answer:

$$\frac{\partial MSE}{\partial \hat{y}} = -1.74075589$$

$$\frac{\partial MSE}{\partial W_2} = \begin{bmatrix} -0.91386062 \\ -0.77502067 \end{bmatrix}$$

$$\frac{\partial MSE}{\partial A_1} = \begin{bmatrix} -0.87037795 & -1.04445354 \end{bmatrix}$$

(where A_1 is the activation output of the hidden layer)

$$\frac{\partial MSE}{\partial b_2} = -1.74075589$$

$$\frac{\partial MSE}{\partial W_1} = \begin{bmatrix} -0.04341028 & -0.05159584 \\ 0.08682056 & 0.10319169 \end{bmatrix}$$

$$\frac{\partial MSE}{\partial b_1} = \begin{bmatrix} -0.21705141 & -0.25797922 \end{bmatrix}$$

8. Calculate the updated values for W_1 and b_1 after doing 1 step of stochastic gradient descent with learning rate 0.1.

   Answer:

$$W_1 = \begin{bmatrix} 0.10434103 & 0.20515958 \\ 0.29131794 & 0.38968083 \end{bmatrix}$$

$$b_1 = \begin{bmatrix} 0.22170514 & -0.07420208 \end{bmatrix}$$