Q&A session – Week 41: Practice Questions with solutions

Q1: Consider a trained Artificial Neural Network with **5 input values** and a **single** neuron. The weights of the input edges are given by the following vector: **w** = (1, 1, 0, -1, -1). The neuron uses the standard linear summation function, i.e., the function that computes the dot product between the input vector **v** and the weight vector **w**. Note that the bias weight is ignored.

Recall that the dot product of two vectors **a** and **b** is the sum of the pairwise products of their corresponding coordinates:

---

according to Wikipedia:

The dot product of two vectors $\mathbf{a} = [a_1, a_2, ..., a_n]$ and $\mathbf{b} = [b_1, b_2, ..., b_n]$ is defined as:

$$\mathbf{a} \cdot \mathbf{b} = \sum_{i=1}^{n} a_i b_i = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$$

The activation function of the neuron is the unit step-wise function:

$$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$$

The classification problem is binary and the class labels are 0 or 1. The activation function ***f*** above directly produces the class labels.

A new (test) example **v** = (-1, -1, 0, 1, 1) is given to the network for classification.

Assuming that the bias weight is always zero (i.e., $w_0 = 0$), answer the following questions by providing clear and sufficient motivations for your answers:

(i) What is the predicted class label for example **v**?

We need to compute the dot product of the weights against the new test example:

(1, 1, 0, -1, -1) dot (-1, -1, 0, 1, 1) = -4 < 0. Hence, class 0.

(ii) Consider a test set with 100 examples, for which the first two features are negative (non-zero), the third feature is a very high positive number, while the remaining features are positive (non-zero). If the true class of all examples is 0, what is the accuracy of the network on the given test set?

The first two features will be multiplied by 1 each, hence they will remain negative in the dot product. The third feature will be multiplied by 0, so we can safely ignore it. Finally, the last two features will be multiplied by -1, hence they will be negative eventually. So the sum of these multiplications, i.e., the dot product, will be a negative number. This means all test examples will be assigned to class 0. Therefore, accuracy=100%.

(iii) Consider the same test set as in (ii) but now with 30 examples of class 1 and 70 examples of class 0. What is the accuracy of the network on the test set?

70 examples will be classified correctly. Hence, accuracy=70%.

Q2: For each of the following statements about deep learning and ranking indicate whether it is correct or not:

- The output of an auto-encoder is much larger than the input.

False. It should typically be equal to the input size.

- Back propagation refers to the transmission of error through the neural network to allow the weights of the network to be adjusted so that the network can learn.

True.

Q3: Prof. Birdfreak wants to use an artificial neural network (ANN) to automatically determine the species of Galapagos finches (birds of the subfamily Geospizinae) in images using the following measurements: (i) beak length, (ii) beak height, (iii) eye diameter, (iv) head length, and (v) body length. Given the location where the pictures were taken, the possible species are: (i) Large Ground Finch Geospiza magnirostris, (ii) Medium Ground Finch Geospiza fortis, (iii) Small Tree Finch Camarhynchus (formerly Geospiza) parvulus, and (iv) Green Warbler-Finch Certhidea olivacea. He has a database of a few hundred labelled images of individuals of these species on which to train his ANN. The are several design aspects Prof. Birdfreak needs to take into account:

(a) How many input and how many output units should the ANN have?

Input: as many as the features + bias = 5 + 1 = 6

Output: as many as the class labels = 4

(b) Should the ANN have hidden neurons?

Of course since without any intermediate neurons it will be a simple perceptron, hence able to simply solve binary and linear classification problems.

(c) What activation functions should the ANN use?

Any activation function here is acceptable, but you could propose sigmoid or relu.

(d) How should the ANN weights be trained?

Using error backpropagation.