0: Summary

In the recitation we discussed the following topics:

1. The view of Neural Networks as nonlinear statistical models.
2. Classification using a single neuron.
3. The Perceptron algorithm.
4. Few exercises on Neural Nets, as a warm-up for the midterm.

1. Neural Networks as Projection Pursuit Regression


2. Single Neuron as a Classifier

In the recitation we went through chapter 39 from David MacKay’s book, [http://www.inference.phy.cam.ac.uk/mackay/itila/](http://www.inference.phy.cam.ac.uk/mackay/itila/).

3. Perceptron

The proof of the theorem we did in the class can be found on page 2 of this document [http://www.cs.cmu.edu/~avrim/ML04/lect0122.ps](http://www.cs.cmu.edu/~avrim/ML04/lect0122.ps).

4. Exercises

Suppose that you have two types of activation functions at hand:
Figure 1. Neural net.

- identity
  
  \[ g_I(x) = x, \]

- step function
  
  \[
g_s(x) = \begin{cases} 
1 & \text{if } x \geq 0, \\
0 & \text{otherwise.}
\end{cases}
\]

So, for example, the output of a neural network with one input \( x \), a single hidden layer with \( K \) units having step function activations, and a single output with identity activation can be written as

\[
\text{out}(x) = g_I(w_0 + \sum_{k=1}^{K} w_k g_s(w_0^{(k)} + w_1^{(k)} x)),
\]

and can be drawn as in Figure 1.

4.1. Representation.

(1) Consider the step function \( u(x) \) in Figure 2. Construct a neural network with one input \( x \) and one hidden layer whose response is \( u(x) \). That is, if \( x < a \), the output of your network should be 0, whereas if \( x \geq a \), the output should be \( y \). Draw the structure of the neural network, specify the activation function for each unit (either \( g_I \) or \( g_s \)), and specify the values for all weights (in terms of \( a \) and \( y \)).

**SOLUTION:** One solution is given in Figure 3. We can set \( w_0 = 0, w_1 = y, w_0^{(1)} = -a, w_1^{(1)} = 1 \).

(2) Now consider the indicator function \( \mathbb{1}_{[a,b]}(x) \):

\[
\mathbb{1}_{[a,b]}(x) = \begin{cases} 
1, & \text{if } x \in [a,b); \\
0, & \text{otherwise.}
\end{cases}
\]

Construct a neural network with one input \( x \) and one hidden layer whose response is \( y \mathbb{1}_{[a,b]}(x) \), for given real values \( y, a, \) and \( b \); that is, its output is \( y \) if \( x \in [a,b) \), and 0 otherwise. Draw the structure of the neural network, specify the activation function for each unit (either \( g_I \) or \( g_s \)), and specify the values for all weights (in terms of \( a, b, \) and \( y \)).
4.2. Boolean Functions. We implemented using a single neuron with a sigmoid actication the boolean function \( f(x_1, x_2, x_3) = (x_1 \text{ AND } x_2) \text{ OR } x_3 \). The key is to understand that the classification boundary defined by the vector \( w = (w_0, w_1, w_2, w_3) \) as \( w^T x = 0 \) is linear, i.e., a hyperplane. For each row of the truth table by substituting the specific values for \( x_1, x_2, x_3 \) we obtain an expression containing the weights. E.g., for all variables equal to 0 the expression which defines the hyperplane reduces to \( w_0 \). Since we know the output, i.e. the value of the function \( f \), we have a set of inequalities, one per row. For example when all variables are 0 then \( f(0,0,0)=0 \) which implies that \( w_0 < 0 \) in order to classify the point \((0,0,0)\) as 0. We finally pick the weights in such way to satisfy all the inequalities.
Figure 4.