Exercises
Introduction to Machine Learning
FS 2018

Institute for Machine Learning
Dept. of Computer Science, ETH Zürich
Prof. Dr. Andreas Krause
Web: `https://las.inf.ethz.ch/teaching/introml-s18`
Email questions to:
Esfandiar Mohammadi, `mohammadi@inf.ethz.ch`

Series 4, Apr 10th, 2018 (ANNs)

Solutions will be published on Friday, April 20th 2018.

Problem 1 (Recurrent Neural Networks):

In the lecture so far, we saw feedforward artificial neural networks, which do not contain any cycles and for which the nodes do not maintain a persistent state over several runs. This exercise considers artificial neural networks with nodes that maintain a persistent state that can be updated. This kind of neural network is called a recurrent neural networks (RNN). As an example, consider the following RNN with

$$y_t = Wx_t + Vs_t$$
$$s_{t+1} = y_t$$

from some initial state s_0, where t denotes the tth call of the RNN, i.e., x_t is the tth input.

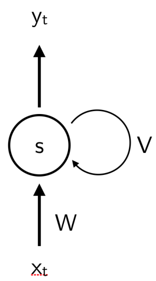


Figure 1

(a) What is the recurrent state in the RNN from Figure 1? Name one example that can be more naturally modeled with RNNs than with feedforward neural networks?

(b) As the state of an RNN changes over different runs of the RNN, the loss functions that we use for feedforward neural networks do not yield consistent results. For a dataset $X := (x_t, y_t)_1^k$, please propose a loss function (based on the mean square loss function) for RNNs and justify why you chose this loss function. (Hint: order is important).

(c) For a dataset $X := (x_t, y_t)_1^k$ (for some $k \in \mathbb{N}$), show how information is propagated by drawing a feedforward neural network that corresponds to the RNN from Figure 1 for $k = 3$. Recall that a feedforward neural network does not contain nodes with a persistent state. (Hint: unfold the RNN.)

Problem 2 (Expressiveness of Neural Networks):

In this question we will consider neural networks with sigmoid activation functions of the form

$$\varphi(z) = \frac{1}{1 + \exp(-z)}.$$

If we denote by v_j^l the value of neuron j at layer l its value is computed as

$$v_j^l = \varphi\left(w_0 + \sum_{i \in \text{Layer}_{l-1}} w_{j,i} v_i^{l-1}\right).$$

In the following questions you will have to design neural networks that compute functions of two Boolean inputs X_1 and X_2. Given that the outputs of the sigmoid units are real numbers $Y \in (0,1)$, we will treat the final output as Boolean by considering it as 1 if greater than 0.5 and 0 otherwise.

(a) Give 3 weights w_0, w_1, w_2 for a single unit with two inputs X_1 and X_2 that implements the logical OR function $Y = X_1 \vee X_2$.

(b) Can you implement the logical AND function $Y = X_1 \wedge X_2$ using a single unit? If so, give weights that achieve this. If not, explain the problem.

(c) It is impossible to implement the XOR function $Y = X_1 \oplus X_2$ using a single unit. However, you can do it using a multi-layer neural network. Use the smallest number of units you can to implement XOR function. Draw your network and show all the weights.

(d) Create a neural network with only one hidden layer (of any number of units) that implements

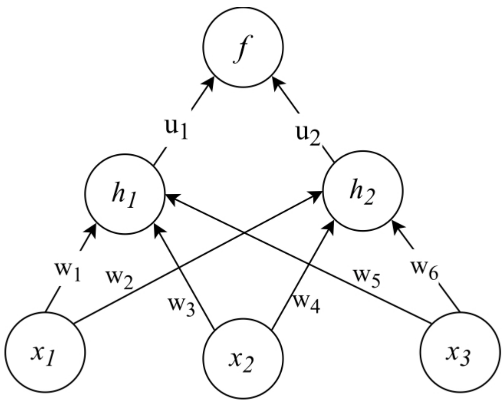
$$(A \vee \neg B) \oplus (\neg C \vee \neg D).$$

Draw your network and show all the weights.

Problem 3 (Exam question: Artificial Neural Networks):

Consider the following neural network with two logistic hidden units h_1, h_2, and three inputs x_1, x_2, x_3. The output neuron f is a linear unit, and we are using the squared error cost function $E = (y - f)^2$. The logistic function is defined as $\rho(x) = 1/\left(1 + e^{-x}\right)$.

[*Note: You can solve part (c) without using the solution for part (b).*]



(a) Consider a single training example $\boldsymbol{x} = [x_1, x_2, x_3]$ with target output (label) y. Write down the sequence of calculations required to compute the squared error cost (called forward propagation).

(b) A way to reduce the number of parameters to avoid overfitting is to tie certain weights together, so that they share a parameter. Suppose we decide to tie the weights w_1 and w_4, so that $w_1 = w_4 = w_{\text{tied}}$. What is the derivative of the error E with respect to w_{tied}, i.e. $\nabla_{w_{\text{tied}}} E$?

(c) For a data set $D = \{(\boldsymbol{x}^{(1)}, y^{(1)}), \cdots, (\boldsymbol{x}^{(n)}, y^{(n)})\}$ consisting of n labeled examples, write the pseudocode of the stochastic gradient descent algorithm with learning rate η_t for optimizing the weight w_{tied} (assume all the other parameters are fixed).