**Applied Data Mining (UN3106)**
Spring 2018
http://stat.columbia.edu/∼porbanz/UN3106S18.html

**Peter Orbanz**
porbanz@stat.columbia.edu
**Phyllis Wan**
pw2348@columbia.edu

# Homework 4

Due: 19 April 2018

**Homework submission:** We will collect your homework **at the beginning of class** on the due date. If you cannot attend class that day, you can leave your solution in Phyllis Wan's postbox in the Department of Statistics, 10th floor SSW, at any time before then.

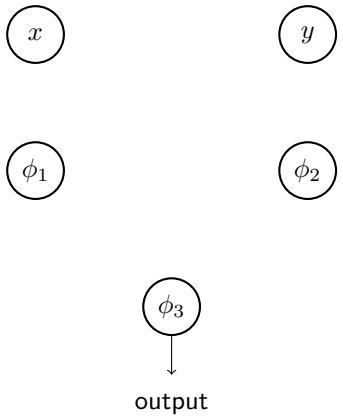
We do not accept homework submitted late. There will be no exceptions.

### Problem 1

Complete the neural network below so that it computes the function $f(x, y) := \mathbb{I}\{\sigma(x) > \sigma(y)\}$. Assume that:

- $x$ and $y$ are the input nodes.
- Each function $\phi_1$, $\phi_2$, $\phi_3$ can be chosen as any of the functions $\phi(x) = c$ (where $c$ is a constant), $\phi(x) = x$, $\phi(x) = \sigma(x)$ (the sigmoid function), or $\phi(x) = \mathbb{I}\{x > 0\}$.

Please add edges and weights as required, and specify which functions are used for $\phi_1$, $\phi_2$ and $\phi_3$.

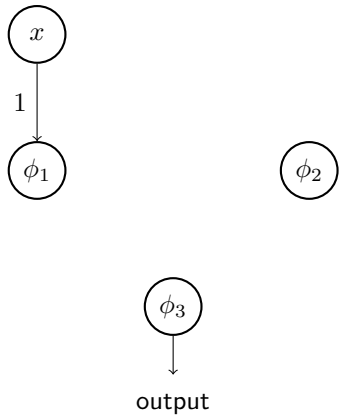



### Problem 2

Complete the neural network below so that it computes the function $f(x) := -\sigma(-x)$, where $\sigma$ is the sigmoid function. Assume that:

- $x$ and $y$ are the input nodes.
- There is no edge from the input node $x$ to $\phi_2$.
- Each function $\phi_1$, $\phi_2$, $\phi_3$ can be chosen as any of the functions $\phi(x) = c$ (where $c$ is a constant), $\phi(x) = x$, $\phi(x) = \sigma(x)$, or $\phi(x) = \mathbb{I}\{x > 0\}$.

Please add edges and weights as required, and specify which functions are used for $\phi_1$, $\phi_2$ and $\phi_3$.




**Problem 3: XOR perceptron**

Consider a neural network which:

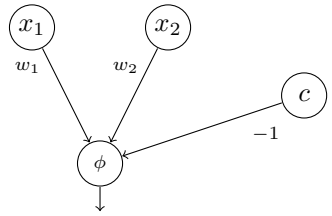
- Takes inputs in $\mathbf{x} = (x_1, x_2) \in \{-1, 1\}^2$.
- Has a single hidden layer with two vertices.
- Represents the function

$$f(\mathbf{x}) := \begin{cases} 1 & \text{if } x_1 = x_2 \\ 0 & \text{otherwise} \end{cases}$$

There several possible networks with these properties. Please draw such a network such that all weights and biases (=constant input units, if you need any) only take values $1$ or $-1$.

**Problem 4: Neural network classifier**

Consider the classifier on $\mathbb{R}^2$, given by the following neural network with $\phi(x) := \mathbb{I}\{x \geq 0\}$:




Suppose that

$$\mathbf{w} := \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \qquad c := \frac{1}{2\sqrt{2}} \qquad \mathbf{x} := \begin{pmatrix} -3 \\ 0 \end{pmatrix} \qquad \mathbf{x}' := \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix} .$$

Compute the classification result for $\mathbf{x}$ and for $\mathbf{x}'$.

**Problem 5: Vector multiplication network**

Design a network with four input units $x_1, y_1, x_2, y_2$ that computes the scalar product of the vectors

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \quad \text{and} \quad \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} .$$

You can assume for simplicity that $x_1, x_2, y_1, y_2 > 0$.

**Hint**: Recall that $\log(a) + \log(b) = \log(ab)$ and $a = \exp(\log(a))$.