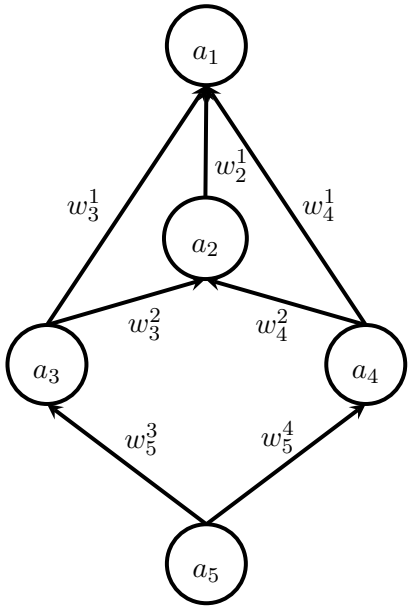
Assignment 4: Neural Networks and Deep Learning

Submission: November 10th
2 students per group

Prof. Fabio A. González
Machine Learning - 2015-II
Maestría en Ing. de Sistemas y Computación

---

1. Consider the following neural network:



where $a_i = \sum_j w_j^i z_j$, $z_i = f_i(a_i)$ for $i = 1, 2, 3, 4$, $z_5 = a_5$ (an input neuron), $f_2(x) = \text{relu}(x)$, and $f_1(x) = f_3(x) = f_4(x) = \text{sigmoid}(x)$. relu$(x)$ corresponds to a rectifier linear unit transfer function defined as:

$$\text{relu}(x) = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

   (a) Write a function to simulate the neural network.

   (b) Deduce the equations to calculate δ_i (the error value per neuron) for all the neurons. Write a function that given a training sample and the weights of the network calculate δ_i for each neuron.

   (c) Assuming that the weight matrix is:

	1	2	3	4
2	3			
3	-4	1		
4	-1	-3		
5			2	-10

use the functions from items (a) and (b) to calculate the output of each neuron, z_i, and the error, δ_i, for the following training samples:

x	y
0.0	0.5
1.0	0.1

(d) Write a function to train the neural network using stochastic gradient descent.

(e) Use the function to train the network with the following training samples:

x	y
-3.0	0.7312
-2.0	0.7339
-1.5	0.7438
-1.0	0.7832
-0.5	0.8903
0.0	0.9820
0.5	0.8114
1.0	0.5937
1.5	0.5219
2.0	0.5049
3.0	0.5002

Plot the evolution of the error and the predictions of the trained network. Write down the weights of the trained network.

2. Bird classification.

(a) Download the dataset *birds* from http://www-cvr.ai.uiuc.edu/ponce_grp/data/.

(b) Use Caffe and the AlexNet pre-trained model, to classify the images in the birds dataset. Construct a confusion matrix that relates the bird classes with the 10 most frequent classes from ImageNet predicted by the model.

(c) Use Caffe and the AlexNet pre-trained model to extract features for all the images in the bird dataset. Use the output of the 'fc6' layer. Train a linear classifier (logistic regression or linear svm) and evaluate it, using the train, validation and test partitions suggested for the dataset.

(d) Repeat the previous step, but this time using as features the output of the 'fc7' layer. Compare and discuss.

3. RICA autoencoder.

(a) Download the code from `https://github.com/jatinshah/ufldl_tutorial` and run both `train.py` and `softmax_exercise.py`, which solve the exercises in `http://ufldl.stanford.edu/wiki/index.php/Exercise:Vectorization` and `http://ufldl.stanford.edu/wiki/index.php/Exercise:Softmax_Regression` respectively. Discuss the results.

(b) Modify the code to train a Restricted Independent Component Analysis (RICA) model instead of a sparse autoencoder. The objective function of RICA is as follows:

$$\min_{W} \quad \lambda \left\| Wx \right\|_1 + \frac{1}{2} \left\| W^T W x - x \right\|_2^2 .$$

Use the approximation $\|y\|_1 \approx \sqrt{y^2 + \epsilon}$ for an ϵ small enough.

(c) Repeat item (a) with the new code. Compare and discuss.

4. The assignment must be submitted as an IPython notebook through the following Dropbox file request, before midnight of the deadline date. The file must be named as `ml-assign4-unalusername1-unalusername2.ipynb`, where `unalusername` is the user name assigned by the university (include the usernames of all the members of the group). If there are several files, please put them in a zip file.