# COMP 7745/8745: Machine Learning

Instructor: Deepak Venugopal

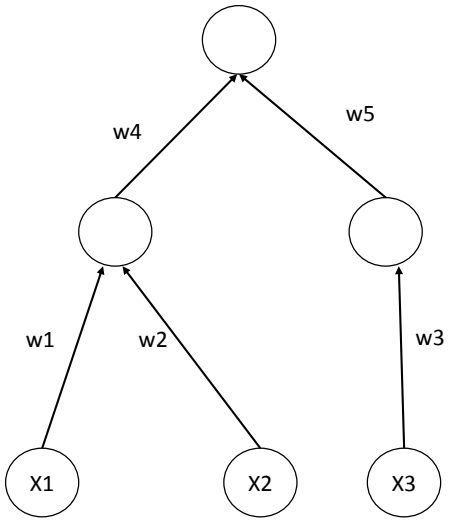
Spring 2017: Homework 1

Due Dates: February 21, 2016 (Hard copy due before the start of class), code on ecourseware

1. Can the following functions be represented using decision trees? If your answer is yes, draw the corresponding tree, if your answer is no, briefly state why. (9 points)
   - $A \wedge \neg B$
   - $A$ XOR $B$
   - $A \vee (B \wedge C)$
2. Can the following functions be represented using perceptrons. If your answer is yes, compute the weight vector for the perceptron such that it can classify all instances of the functions correctly. Give a one line justification if your answer is No. (9 points)
   - $A \vee B$
   - $\neg A \vee B$
   - $A$ XOR $B$
3. For each of the following, state whether true of false with a brief justification. (10 points)
   (a) A 2-layered neural network where each perceptron is a linear unit has the same expresiveness as a 2-layered neural network where each perceptron is a sigmoid unit.
   (b) Given a training dataset with N features, the number of nodes in any decision tree is guaranteed to be lesser than or equal to N.
4. Run the ID3 algorithm (manually) for the following dataset to classify whether students like a restaurant or not. (10 points)

| **Price** | **Fast** | **On Campus** | **Like** |
|---|---|---|---|
| $ | No | No | No |
| $ | Yes | Yes | No |
| $$ | No | Yes | No |
| $$ | Yes | Yes | Yes |

5. Compute the backpropagation update equations for each weight in below figure given the input $X1 = -1$, $X2 = 1$, $X3 = -1$ with the desired response $y = 0$. Let the weights be initialized as follows: $w_1 = -1$, $w_2 = 2$, $w_3 = 1$, $w_4 = 1$ and $w_5 = 2$. (You don't need to explicitly compute the sigmoid values, just leaving them symbolically as S(.) is sufficient). (12 points)




6. In this question, you will experiment with the perceptron weight learning rule. You have been given 3 datasets d-10.csv, d-100.csv and d-500.csv. The first line in each of these datasets is just a column identifier and each subsequent line corresponds to one instance where the last attribute is the class value. Implement the perceptron weight learning algorithm (not the sigmoid gradient descent rule). Initialize all the weights randomly between 0 and 1. Remember to add a bias term. Present results as in the following table for each dataset.

| **Learning rate** | **Error after 100 iterations** | **Error after 500 iterations** | **Error after 1000 iterations** |
|---|---|---|---|

   Consider an iteration as a complete pass over the entire dataset. Compute the error as the % of mis-classifications. Very briefly explain your results, e.g., did the error go down on all three datasets with increased iterations, what was the effect of the learning rate, what was the effect of the increase in dimensionality (number of attributes) with the error, etc. (25 points)

7. (25 points) Here, you will use Weka to experiment with ID3, J48 and neural networks (multilayer perceptron) using the wines.csv and iris.csv datasets. In both datasets, the first line is just the attribute names and each subsequent line specifies one instance where the final attribute is the class value. (Starter code for loading files and filtering them (discretizing/nominal-class) is provided)

   For both datasets, you will analyze the performance of the three algorithms using the average F-1 score obtained using cross-validation.

   1. For ID3, report the weighted F-1 scores for each dataset (You may need to use the filtering functions provided in the starter code for the datasets)

2. For J48, for each dataset, (you may need to use the filtering functions provided in the starter code for the datasets), vary the confidence factor in J48 between 0.1 and 0.5 (smaller confidence factor results in larger pruning of the tree) and report the F-1 for each case.

3. For Neural networks, for each dataset, vary the number of hidden layers (1,2) and the learning rate (0.01,0.1,0.2). Report the cross-validated weighted F1-scores for each of the 6 cases.

Briefly describe your overall results (which one performed well for which dataset, etc.)