SGN-13006 Introduction to Pattern Recognition and Machine Learning
Department of Signal Processing, Tampere University of Technology
Joni Kamarainen and Katariina Mahkonen
Exercise 3

These problems are discussed in exercise sessions on Friday 11.9 and Monday 14.9 Start preparing your solutions before the session. During the exercise session you can consult the teacher, and once finished your work, show your solutions to the teacher to get the exercise points.

1. **Decision trees** (1 point)

   Give decision trees to represent the following boolean functions:

   (a) $A \wedge \neg B$

   (b) $A \vee [B \wedge C]$

   (c) $A \quad XOR \quad B$

   (d) $[A \wedge B] \vee [C \wedge D]$

2. **Entropy measures** (1 point)

   Consider the following set of training examples:

Instance	Classification	a_1	a_2
1	+	T	T
2	+	T	T
3	-	T	F
4	+	F	F
5	-	F	T
6	-	F	T

   (a) What is the entropy of this collection of training examples with respect to the target function classification?

   (b) What is the information gain of a_2 relative to these training examples?

   (c) What is the information gain of a_1 relative to these training examples?

3. **Decision Tree Classifier** (2 points)

   Data: We are using the tiny subset of Imagenet dataset. Use the Matlab template `imagenet_tiny5_process.m` to handle the data.

   Feature extraction: For decision tree classification we need nominal features. This means that there can be only a finite number of possible values for each feature. The color value as an integer between 0 and 255 is such indeed, but let's consider something with smaller number of possible values. Let the feature vector be of length 64 - one feature per pixel - where each feature gets one of the three values $1, 2, 3$ depending on which color channel value at that pixel is the biggest.

   Classifier training: Use the given (in course POP-page) tree learning algorithm `ID3_train` and

   Classification: Classify each evaluation image, with the classification function `ID3_classify` (from POP).

   Calculate the classification accuracy.

   Test with different values for parameters maximalDepth and minimumNodeSize of ID3-algorithm.

4. **Implementation of the Random Forest algorithm (using ID3)** (2 points)

   (a) Write a Matlab function to construct a random forest, where each tree is made with ID3-algorithm. The function *ID3_learn* to make one tree with ID3 is given at the course POP-page. The function call of the function implementing random forest should be:

   ```
   forest = random_forest_learn(traindata,trainclass,N)
   ```

The input *traindata* has to be categorical as in previous exercise for ID3-algorithm. The other input *trainclass* contains the class labels. The third input N is the number of trees in the forest. You can use for example $2/(N+1)\%$ of data and attributes for each tree. The function should output the forest as a structure containing all the trees of the forest as well as information about which attributes each tree uses.

(b) Implement also a function to do classification with the forest. The function call is as follows:

```
classes = random_forest_classify(data,forest)
```

The inputs are the categorical test vectors and the structure *forest*. The output is a vector containing one class label for each test data vector. The function *ID3_test* to get a classification with one ID3-tree is given in the course POP-page.