**SECOND ASSIGNMENT. MACHINE LEARNING WITH SCIKIT-LEARN.**[**¶**](https://ocw.uc3m.es/ingenieria-informatica/machine-learning-i/secondassignmentpart1.html#SECOND-ASSIGNMENT.-MACHINE-LEARNING-WITH-SCIKIT-LEARN.)

**PART I (1.5 POINTS)**[**¶**](https://ocw.uc3m.es/ingenieria-informatica/machine-learning-i/secondassignmentpart1.html#PART-I-(1.5-POINTS))

The aim of part I of the Scikit-learn assignment is for you to self-learn and get used to this Machine Learning tool. The main part (part II) of the assignment will be explained next week (11/12).

Here, you will learn to:

* Perform a crossvalidation on the iris classification problem with decision trees (so far, we have only done regression)
* Perform a crossvalidation on the iris classification problem **with KNN** (I haven't explained this, you will have to learn how to use it from the web)
* Perform grid search in order to determine the best value for hyper-parameter K

You will also have to go through two notebooks I have prepared for you in order to see how crossvalidation and hyper-parameter tuning are used in Scikit-learn

**0. Carry out the "DECISION TREES WITH A TRAINING AND A TESTING SET AND CROSSVALIDATION" notebook and understand the main ideas**[**¶**](https://ocw.uc3m.es/ingenieria-informatica/machine-learning-i/secondassignmentpart1.html#0.-Carry-out-the-)

**1. Perform a crossvalidation on the iris classification problem with decision trees:**[**¶**](https://ocw.uc3m.es/ingenieria-informatica/machine-learning-i/secondassignmentpart1.html#1.-Perform-a-crossvalidation-on-the-iris-classification-problem-with-decision-trees:)

**It is important to remember that for classification, you have to use**

* clf = tree.DecisionTreeClassifier() # for constructing the classifier
* metrics.accuracy # for computing error

In [7]:

# Write code here

**2. Perform a crossvalidation on the iris classification problem with KNN**[**¶**](https://ocw.uc3m.es/ingenieria-informatica/machine-learning-i/secondassignmentpart1.html#2.-Perform-a-crossvalidation-on-the-iris-classification-problem-with-KNN)

I haven't explained how to use KNN in Scikit-learn. You will have to read and obtain the relevant information [here](http://scikit-learn.org/stable/modules/neighbors.html)

In [8]:

# Write code here

**3. Try different values for K (KNN) - change them by hand- and see if you obtain a better result than with KNN default value. Always use crossvalidation.**[**¶**](https://ocw.uc3m.es/ingenieria-informatica/machine-learning-i/secondassignmentpart1.html#3.-Try-different-values-for-K-(KNN)---change-them-by-hand--and-see-if-you-obtain-a-better-result-than-with-KNN-default-value.-Always-use-crossvalidation.)

In [9]:

# Write code here

**4. Carry out THE "DECISION TREE HYPER-PARAMETERS. TUNING DECISION TREES" notebook and understand the main ideas**[**¶**](https://ocw.uc3m.es/ingenieria-informatica/machine-learning-i/secondassignmentpart1.html#4.-Carry-out-THE-)

**5. USE GRID SEARCH AND RANDOMIZED SEARCH TO FIND THE OPTIMAL VALUE FOR K**[**¶**](https://ocw.uc3m.es/ingenieria-informatica/machine-learning-i/secondassignmentpart1.html#5.-USE-GRID-SEARCH-AND-RANDOMIZED-SEARCH-TO-FIND-THE-OPTIMAL-VALUE-FOR-K)

In [10]:

# Write code here

**6. OPTIONAL (you may get 0.25 extra points if you decide to do this).**[**¶**](https://ocw.uc3m.es/ingenieria-informatica/machine-learning-i/secondassignmentpart1.html#6.-OPTIONAL-(you-may-get-0.25-extra-points-if-you-decide-to-do-this).)

K is the main hyper-parameter of KNN. Find another hyper-parameter that you consider relevant, and try to optimize both K and the other parameter using grid-search. Are you able to improve on previous results?

In [11]:

# Write code here (optional)

In [ ]:

