Decision Tree Classifier: SkySurvey

Mauro Sozio

We are going to use a dataset from the Sloan Digital Sky Survey (SDSS) which contains several images of the sky collected with a wide-angle optical telescope in New Mexico, United States. The final data release covers over 35% of the sky and it is publicly available. More information can be found on Wikipedia: `https://en.wikipedia.org/wiki/Sloan_Digital_Sky_Survey`. We are going to use a small excerpt of that dataset containing data related to approximately 10000 objects in the sky. We are going to focus on the task of classifying sky objects as stars, galaxies or quasars. In our dataset, the class value 0 corresponds to star, 1 corresponds to galaxy and 2 to quasar. The feature names are (respectively) 'ra', 'dec', 'u', 'g', 'r', 'i', 'z', 'run', 'rerun', 'camcol', 'field', 'specobjid', 'redshift', 'plate', 'mjd', 'fiberid'.

This lab will not be evaluated, however, a few questions related to the lab might be asked at the final exam. Questions:

1. (10/100) Given the dataset we provided to you, build a decision tree using the parameter $min\_sample\_leaf = 0.01$. Such a parameter value specifies that the training data per leaf is 1% of all training data which allows us to get statistically significant results. Set also $random\_state = RandomState(2018)$, which makes the algorithm deterministic. All other parameters should have their default values.

2. (15/100) compute the generalization error of the decision tree you built. To this end, you might use the array $clf.tree\_.children\_left$ where $clf.tree\_.children\_left[i] = -1$ if i is a leaf while clf is the tree you built with $DecisionTreeClassifier$ in sci-kit learn.

3. (15/100) The decision tree you built in the first part of the question might not be ideal for our task. You should try to change the input parameters of $DecisionTreeClassifier$, so as to build a decision tree with *minimum generalization error*. Here we consider the parameter max_depth. Determine the best value for max_depth so as to minimize the generalization error. You should maintain $min\_sample\_leaf = 0.01$ so as to make sure to obtain results that are statistically significant. Do not change $random\_state$ either. What could be a good value for max_depth? Visualize the decision tree.

4. (10/100) Compare the decision trees you built in point 1 and the best one you obtained in point 3. Which one would you recommend to use to classify sky objects?

5. (10/100) Consider the decision tree you considered to be best in the previous point. Predict the class value of an object of your choice. Which feature is most relevant when classifying sky objects?

6. (10/100) Do you think that the best decision tree you built could be pruned so as to improve the generalization error?

7. (30/100) The library we recommend (sci-kit learn) does not support post-pruning, yet. However this could be implemented by using the variables of the $tree\_$ object computed by the DecisionTreeClassifier in sci-kit learn. See [1] to see some examples. In particular, $clf.tree\_.children\_left[i]$ specifies the index of the left children of i, $clf.tree\_.children\_right[i]$ specifies the index of the right children of i, while $clf.tree\_.value[i]$ specifies the class distribution of i. Implement a post-pruning strategy (among the ones we considered in our course) and run it on the best decision tree so far. Does this improve the generalization error? In case you cannot modify your instance of the DecisionTreeClassifier, you can use another data structure to store the pruned tree.

[1] `http://scikit-learn.org/stable/auto_examples/tree/plot_unveil_tree_structure.html#sphx-glr-auto-examples-tree-plot-unveil-tree-structure-py`