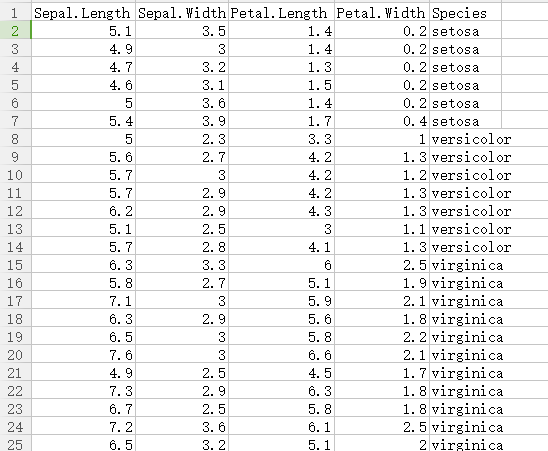
**Classification of Iris Flower Datasets Using Decision Trees**

**1. Background of the dataset**

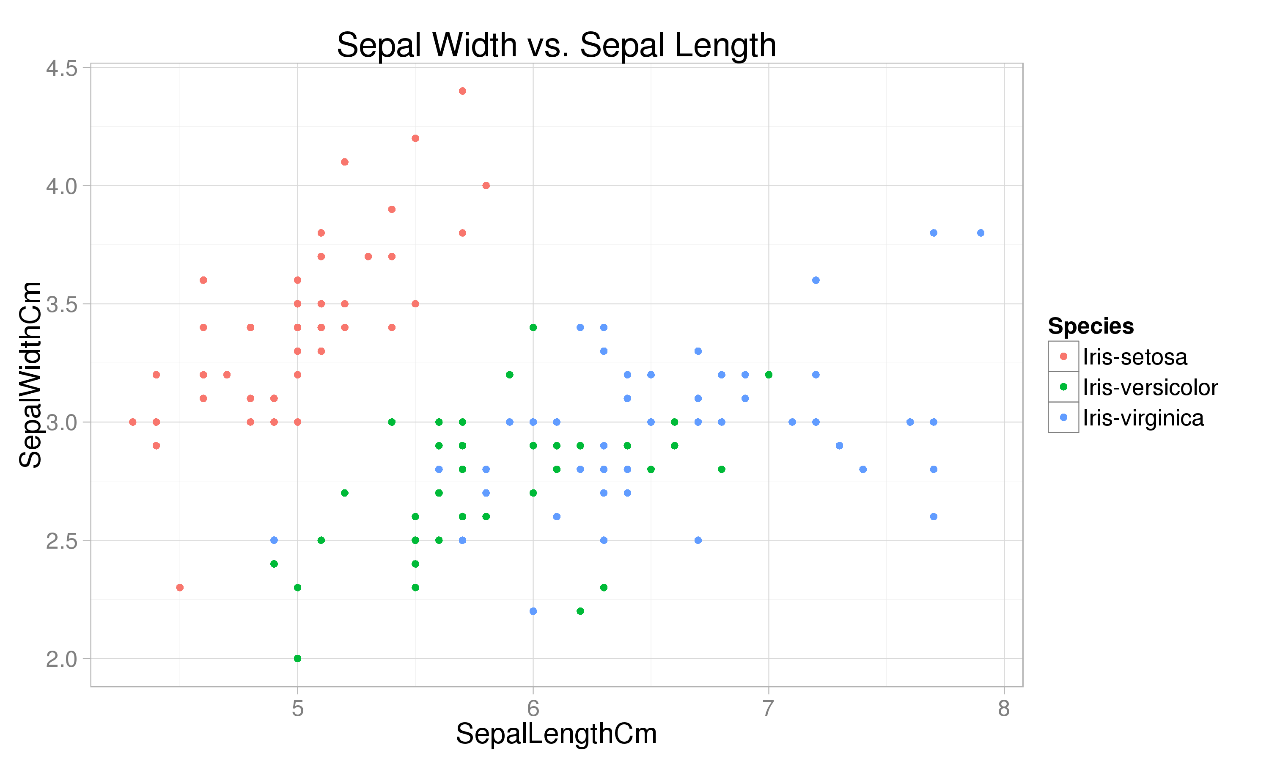
The Iris dataset was used in R.A. Fisher's classic 1936 paper ‘[The Use of Multiple Measurements in Taxonomic Problems](http://rcs.chemometrics.ru/Tutorials/classification/Fisher.pdf)’. It can also be found on the ‘[UCI Machine Learning Repository](http://archive.ics.uci.edu/ml/)’.

It includes three iris species with 50 samples each as well as some properties about each flower. In the data set, one flower species is linearly separable from the other two, but the other two are not linearly separable from each other.



There’s a figure (Sepal width as the Y-axis and Sepal length as the X-axis) apparently shows the separation of the data in the dataset.

From this figure, we can find that points represent iris-setosa separates form the other two species while the points of these two species merges and cannot be separated easily.



**2. Problem**

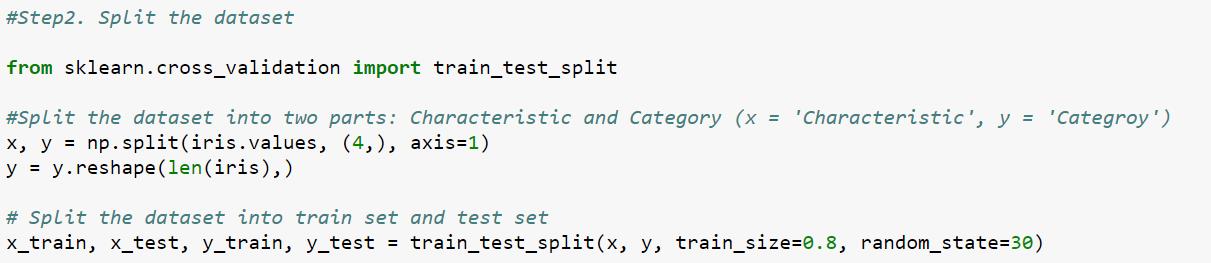
So, here comes the problem. One species, Iris-Setosa, is "linearly separable" from the other two which means that we can draw a line between Iris-Setosa samples and samples corresponding to the other two species. We can see this in the graph above.

However, Iris-Versicolor and Iris-Virginica are not linearly separable from each other: there is no line that we can draw to differentiate them on any subset of the feature. So we have to work out a solution to separate these two species

**3. Solution**

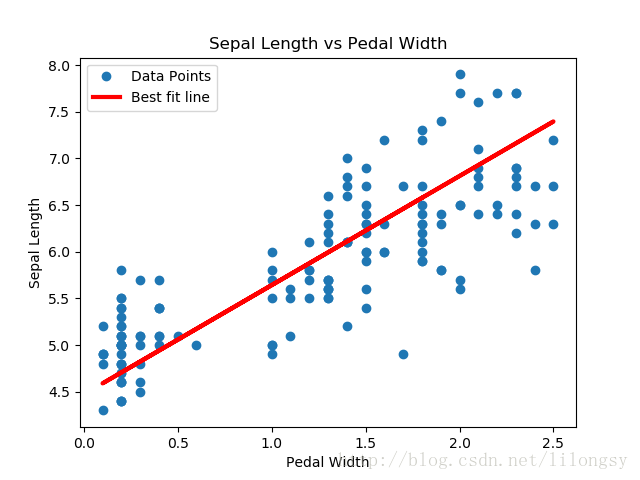
**Pretreatment of the dataset**

From the flow chart, we first read the data set as well as split the data set into the characteristic and category of train set as well as test set.

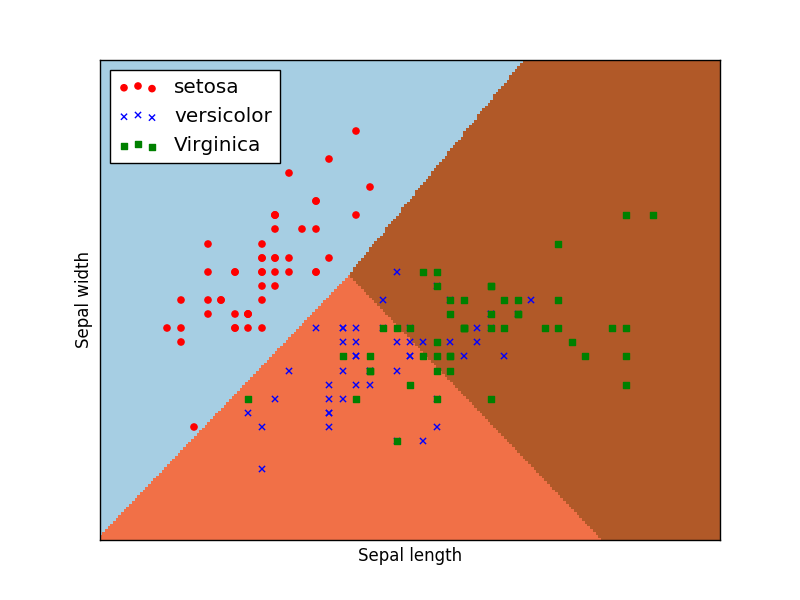
****

In the experiment, we first use liner and logistic regression to separate the iris species. However, we find that the results are far from satisfaction.

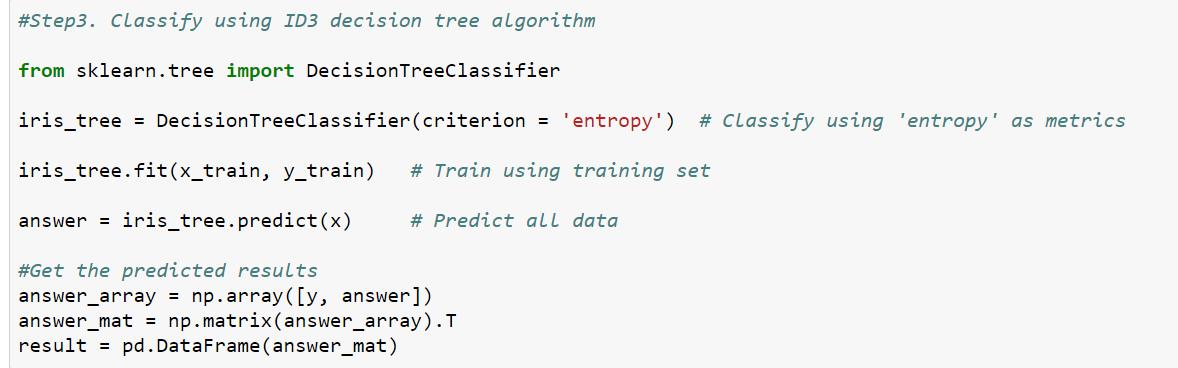
**Liner regression:**



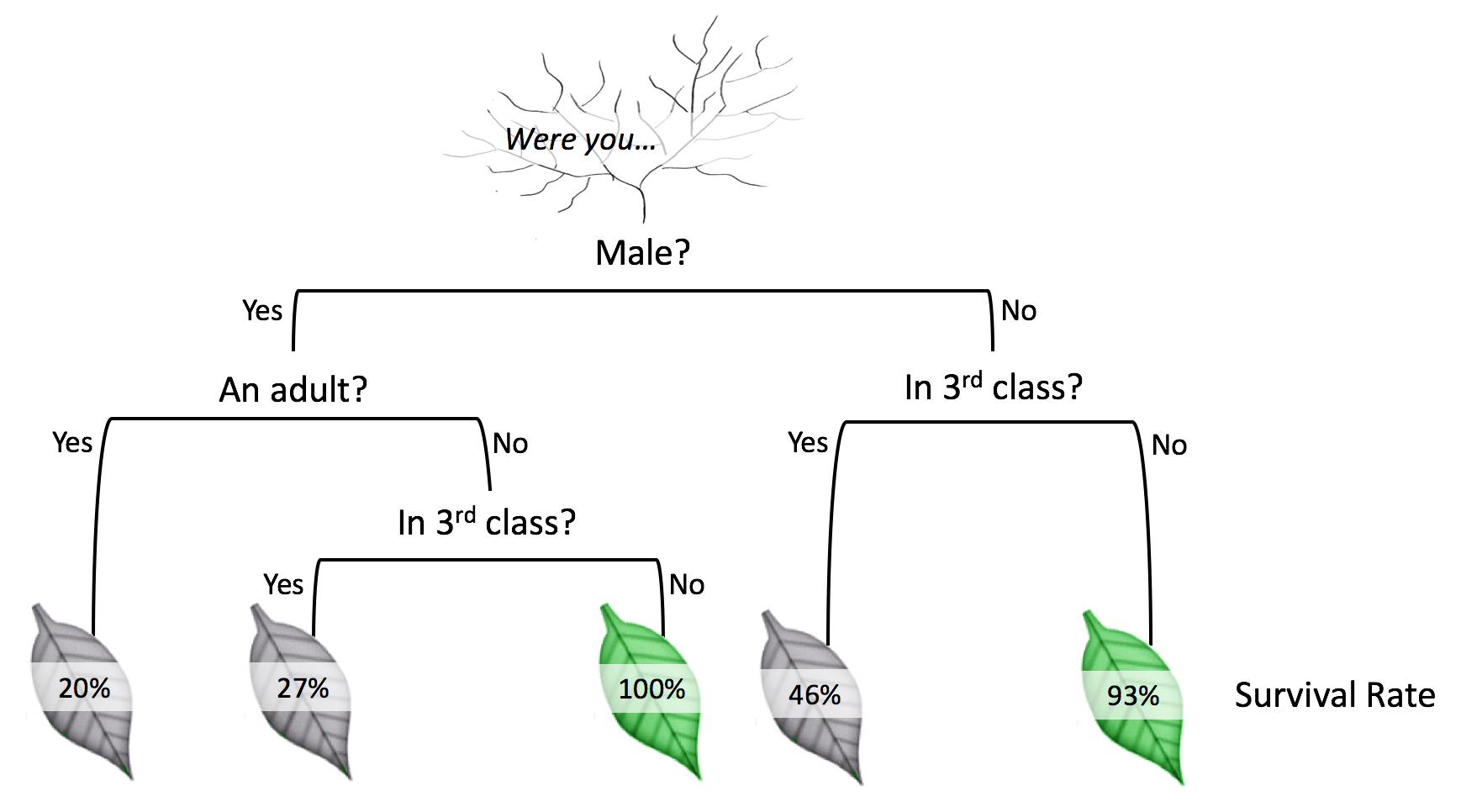
**Logistic regression:**



**Decision tree algorithm:**

****

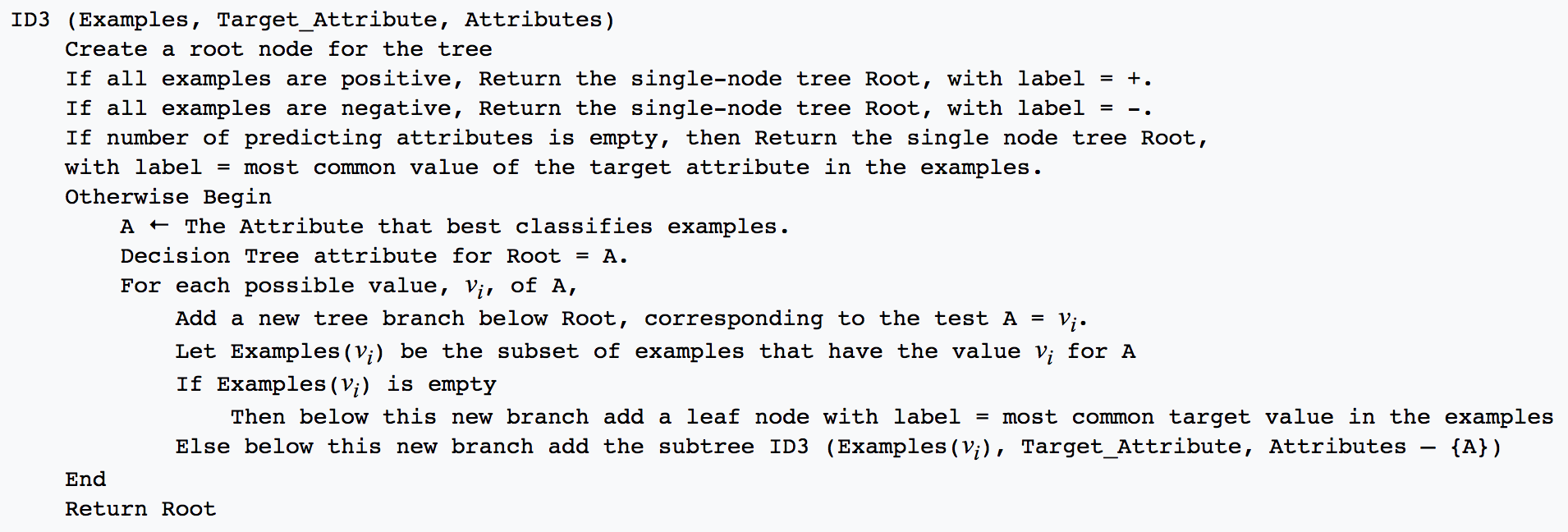
Since the liner and logistic regression cannot meet our expectation, we employed ID3 decision tree algorithm to tackle this classification problem. So, what exactly is a decision tree? A decision tree is a tree-like model, where each node corresponds to an input variable and each of its children represents a specific subset of this input variable’s domain. The leaves of a decision tree are potential target values. Given the values of these input variables, we can trace a path to a leaf and the value of this leaf is our prediction.



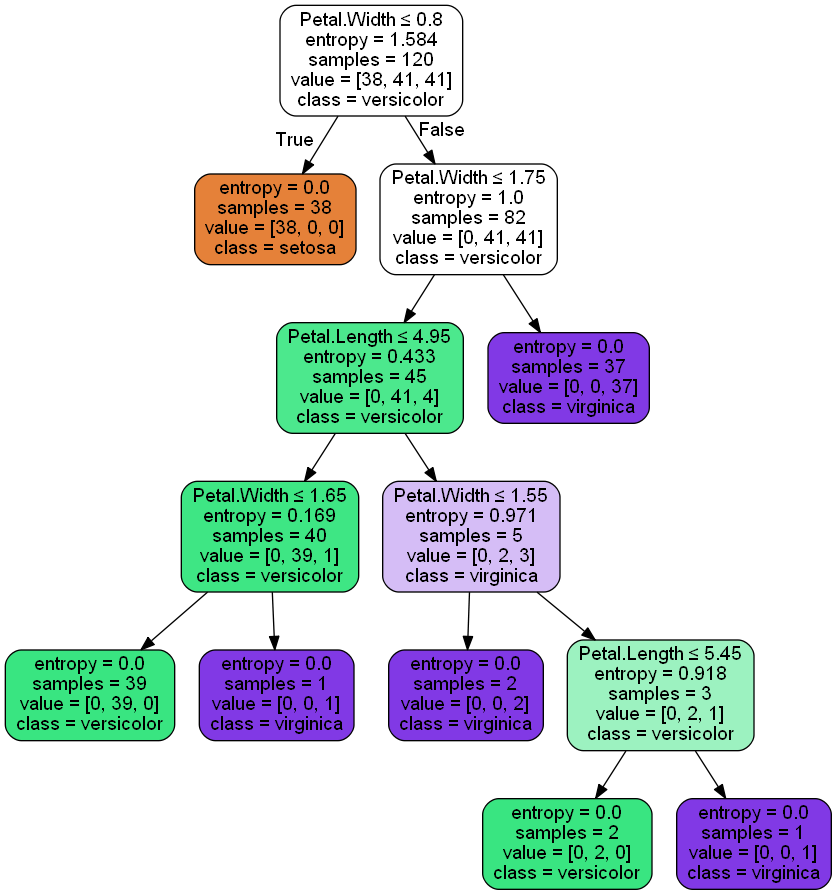
Let’s look at an example above. Here we are solving a survival problem. The nodes “male”, “an adult?” and “in 3rd class” are three input variables. If the value of “male” is “Yes”, then we proceed to its left children. Otherwise, we go to right. For an observation with input values being “Not male” and “Not in 3rd class”, we can easily find the answer to be “93% survival rate”.

**ID3 decision tree algorithm:**

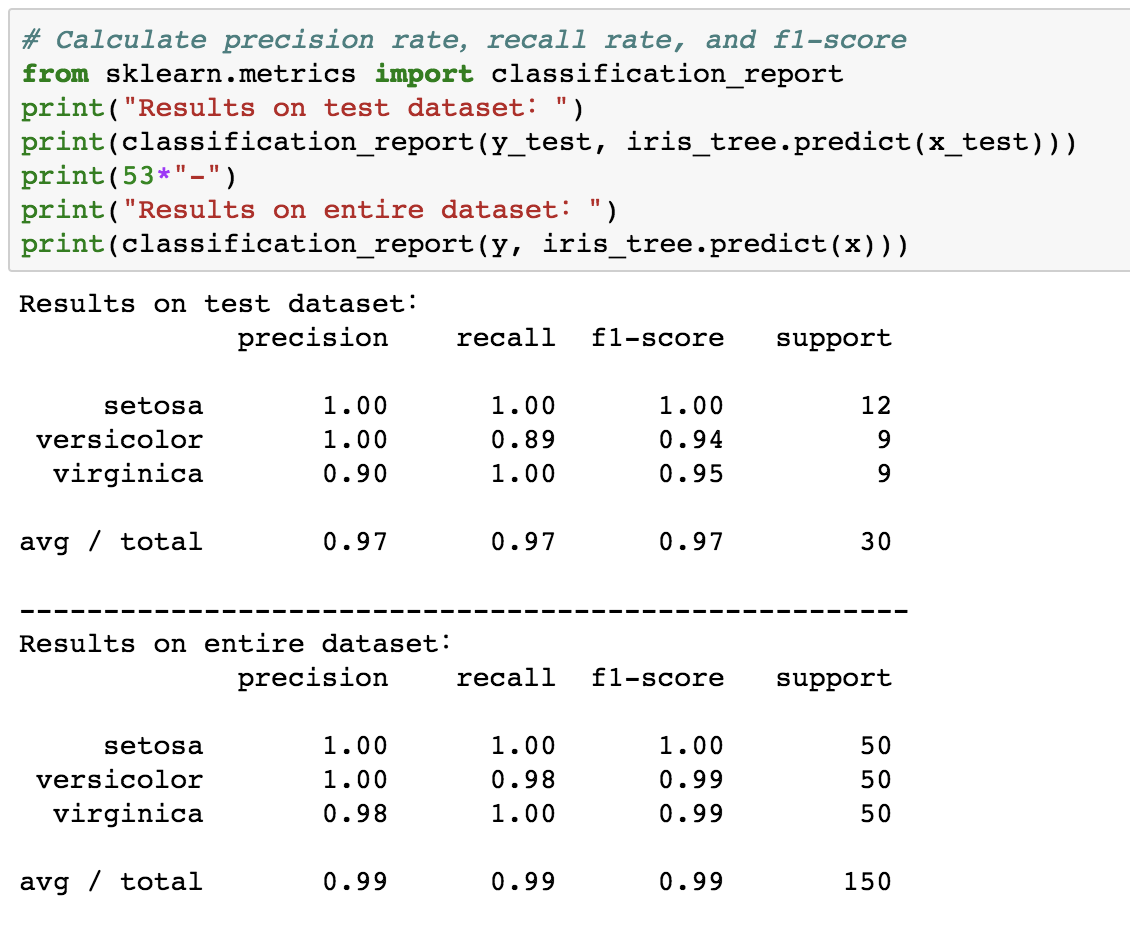
ID3 is a particular type of decision tree. It uses entropy as the metrics. To summarize the algorithm, there’s a pseudocode below. In each iteration, we calculate the entropy for each unused input variables and select the one with the smallest entropy as our cut-off variable. Then we split our training data based on this cut-off variable. The algorithm terminates when data can no longer be split, or when every input variable is used.



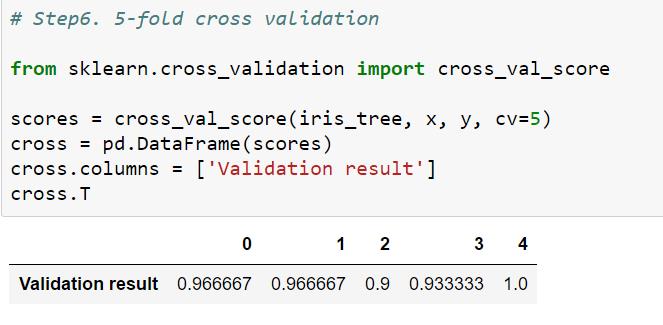
To better explain our classifier, we plotted our trained decision tree. From the graph, we can see that if a petal width is less than or equal to 0.8, then it belongs to setosa class. And if a petal width is greater than 1.75, then it’s virginica class, and so on so forth.



**4. Results and validation**

To generalize our results, we used precision rate, recall rate, and f1 score as our metrics. Precision is True Positive divided by the sum of True Positive and False Positive, while recall rate is True Positive divided by the sum of True Positive and False Negative. Both metrics reflect the performance of our classifier. F-1 score is simply the average of precision and recall rate. On our test dataset, these three results are 0.97, 0.97, and 0.97 respectively, which are fairly satisfactory.   


In addition, we also performed 5-fold cross-validation. And as we can see from the slides, the lowest accuracy is 0.9, which is also satisfactory.



**5. Summary**

From the above results, we can see that ID3 algorithm is quite good for Iris dataset classification, decision tree has two advantages: First, the decision tree model can be read very well, descriptive and helpful for manual analysis; second, it has high efficiency. The decision tree only needs to be constructed once and used repeatedly. The maximum number of calculations per prediction does not exceed the depth of the decision tree.

And there are some other things worth talking about for this project.

The main difficulty in this project is model selection. At first, we used linear regression model analysis, and later switched to logistic regression model, but the classification effect is not very good. Finally, we chose decision tree classification, and the results are satisfactory.