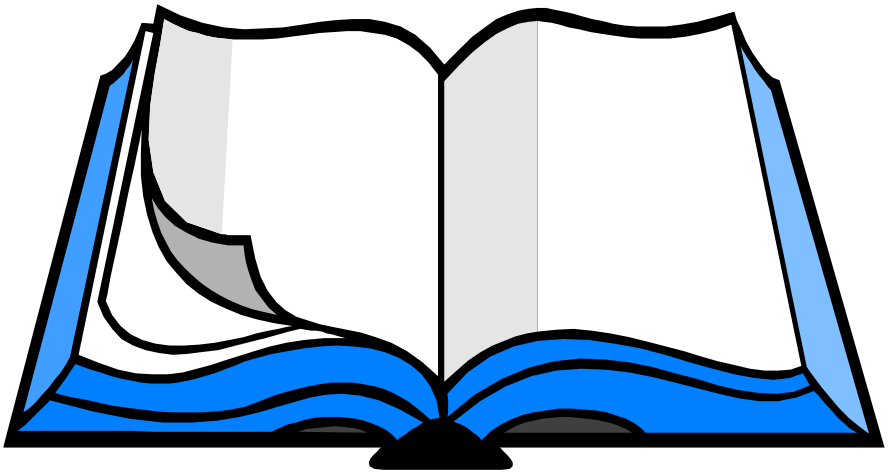
University of Science

Information Technology Department

LAB02: Decision tree



***Instructors**: Nguyễn Ngọc Thảo, Lê Ngọc Thành*

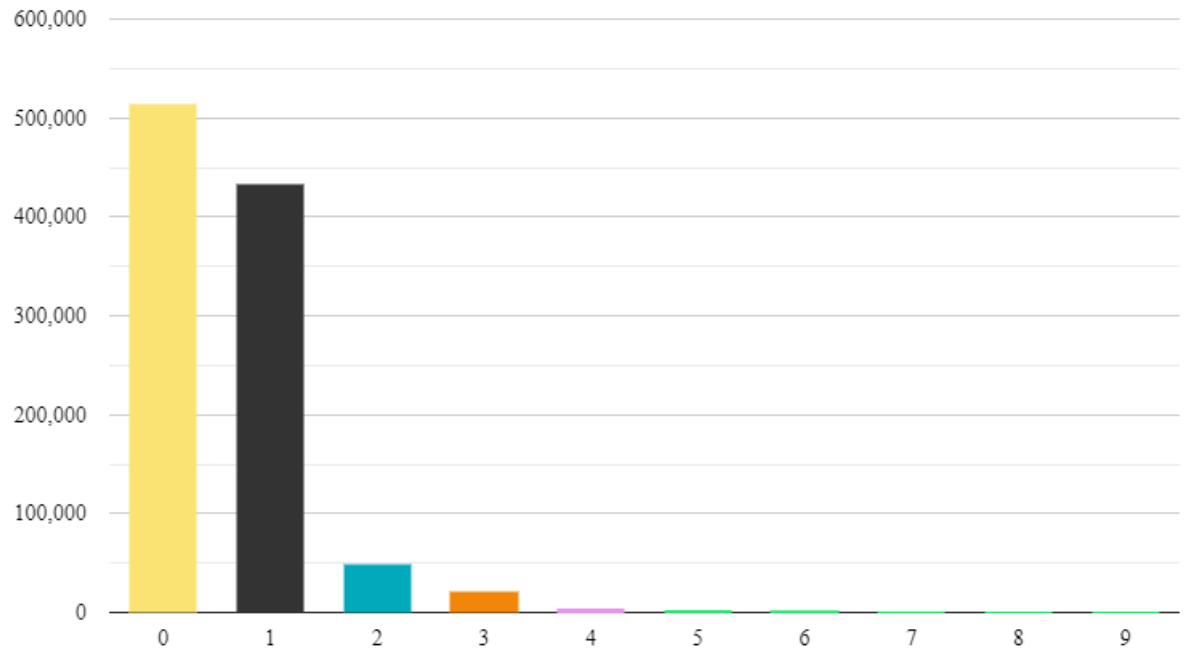
Students: *21127667 – Trương Công Gia Phát*

Introductions

The lab’s goal is to write a Python program and use the scikit learn function to build a decision tree on the UCI Poker Hand Data Set.

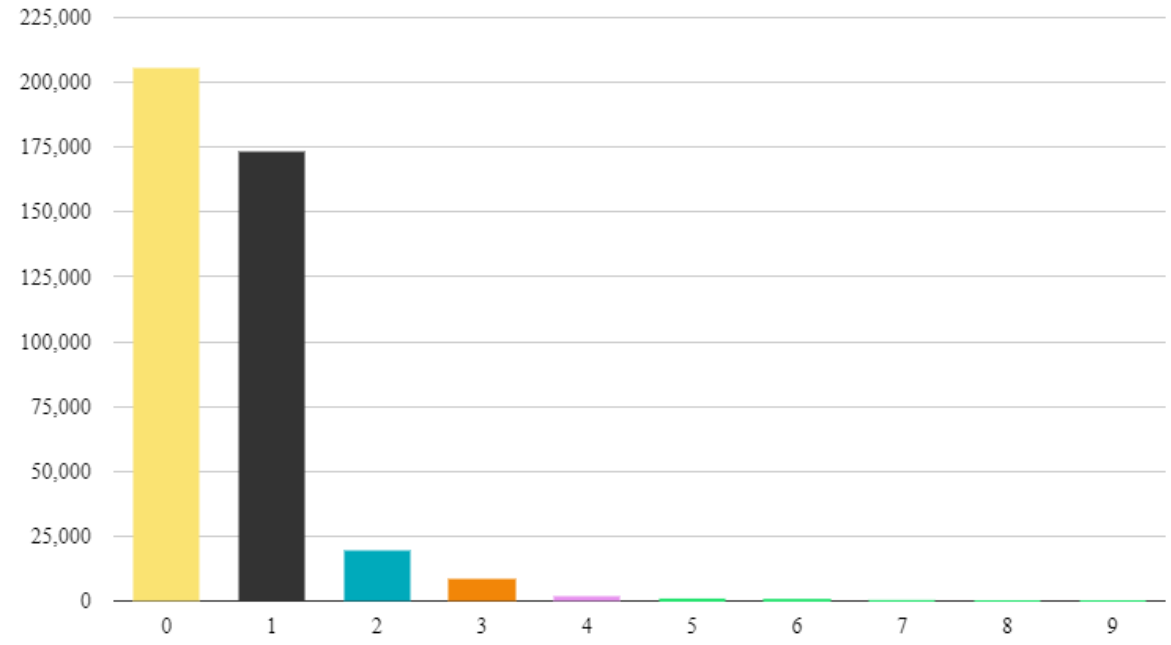
I. Preparing the dataset

I use classification_report() to count the number of labels on each set then visualize it as follows

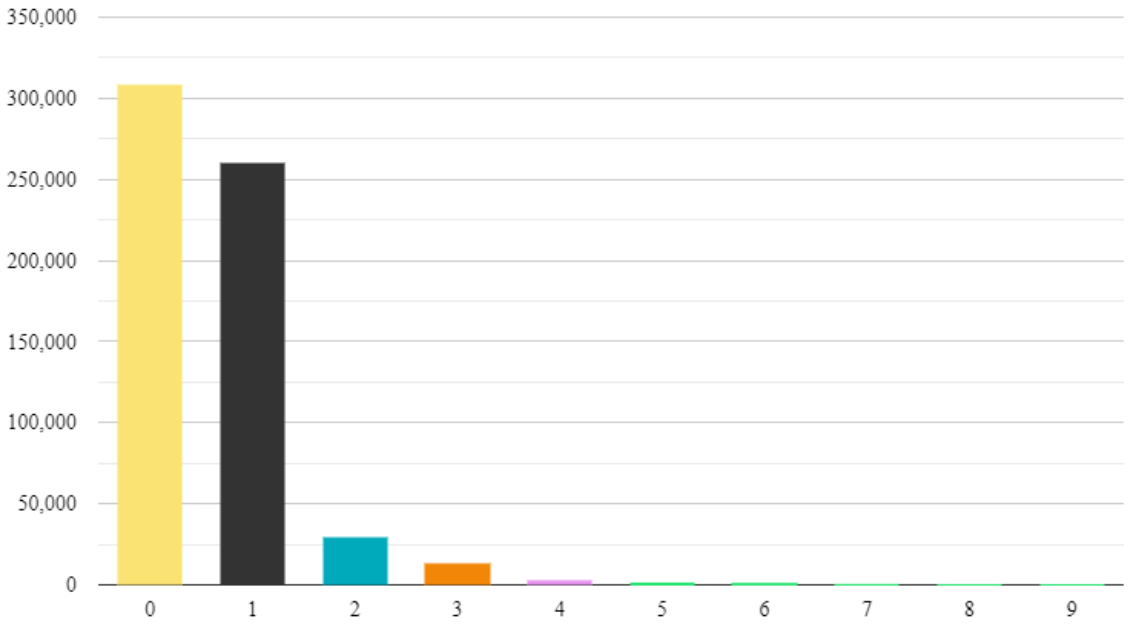


Classes distribution in original set

40/60 dataset

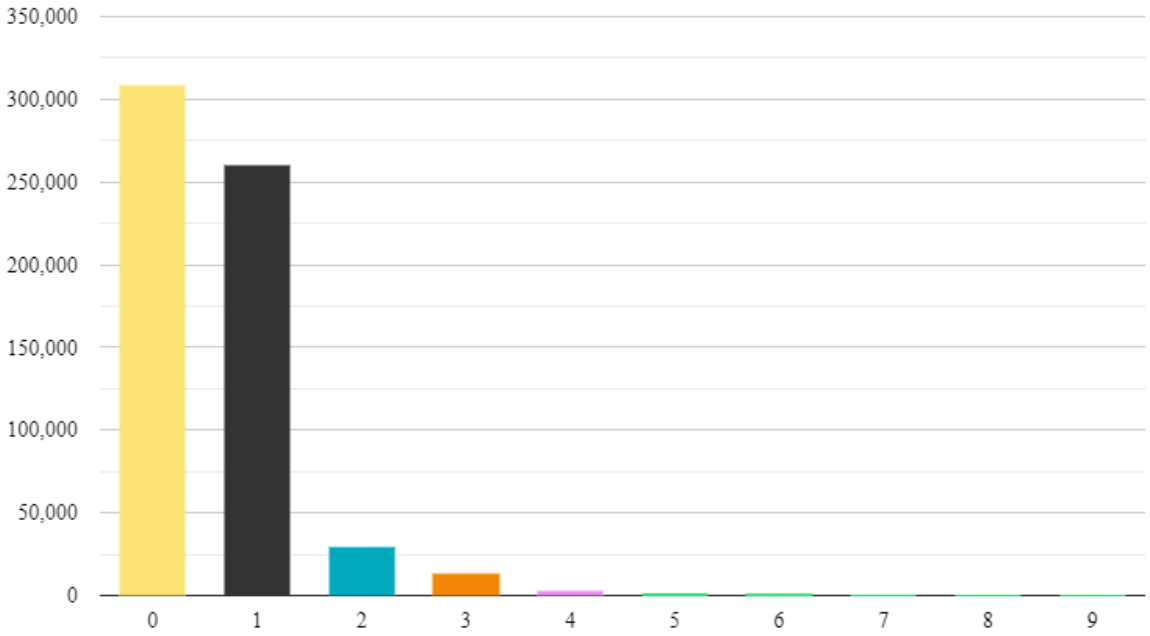


Classes distribution in training set

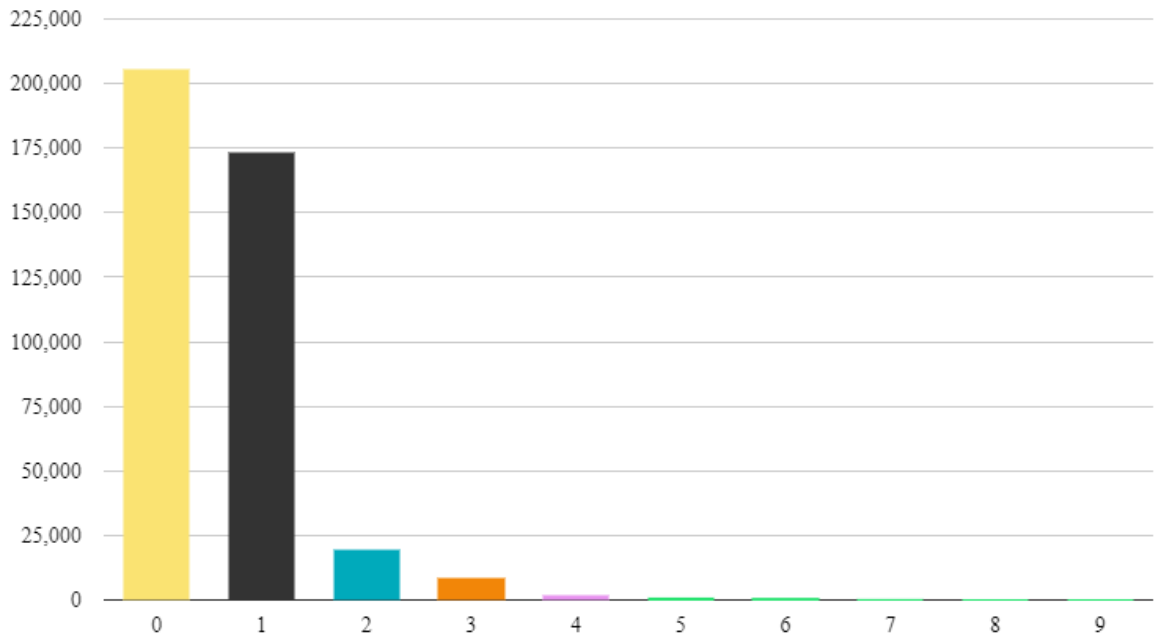


Classes distribution in test set

60/40 dataset

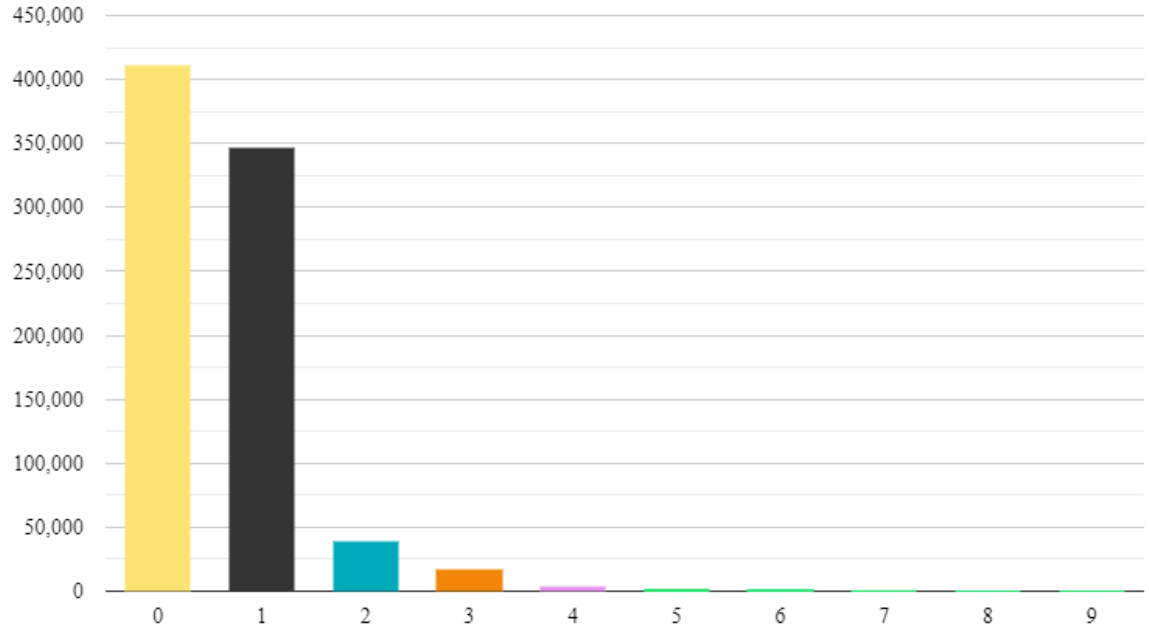


Classes distribution in training set

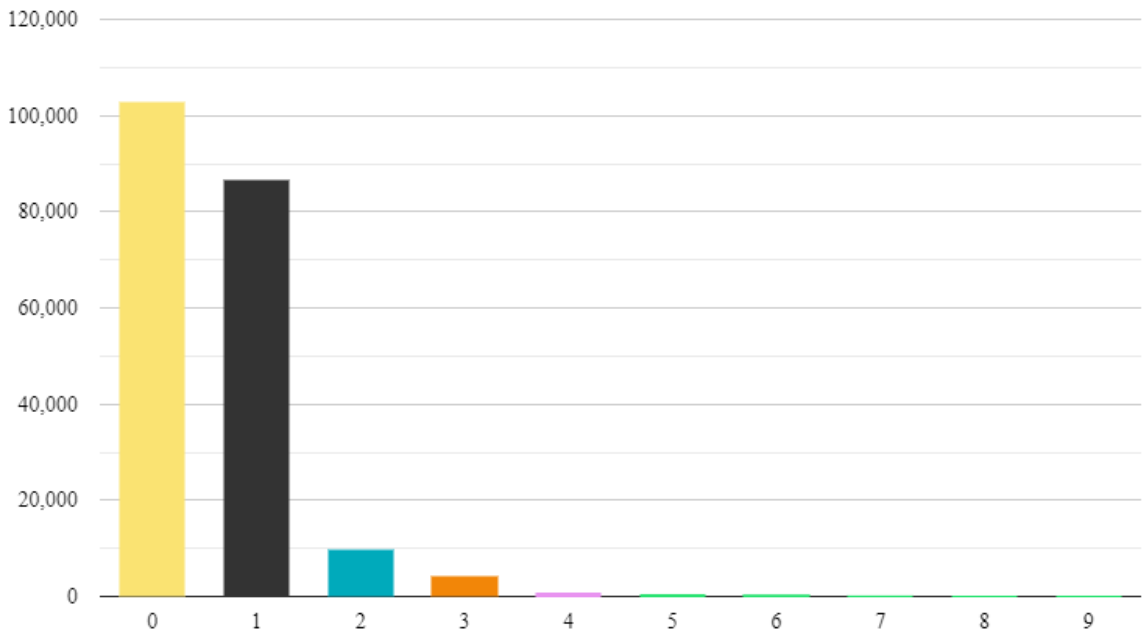


Classes distribution in test set

80/20 dataset

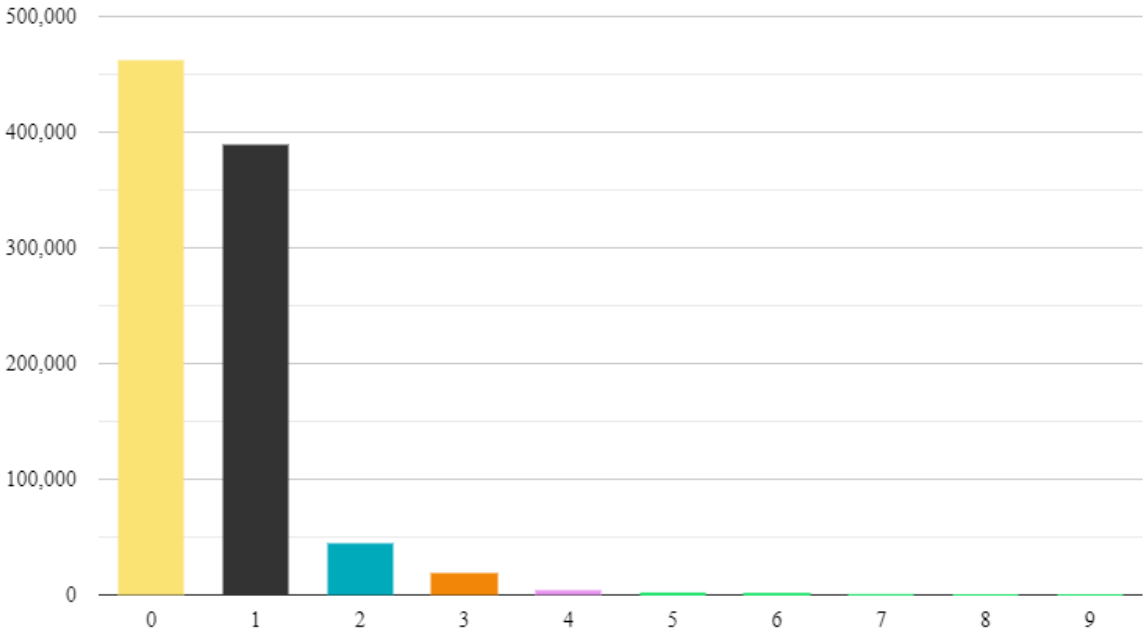


Classes distribution in training set

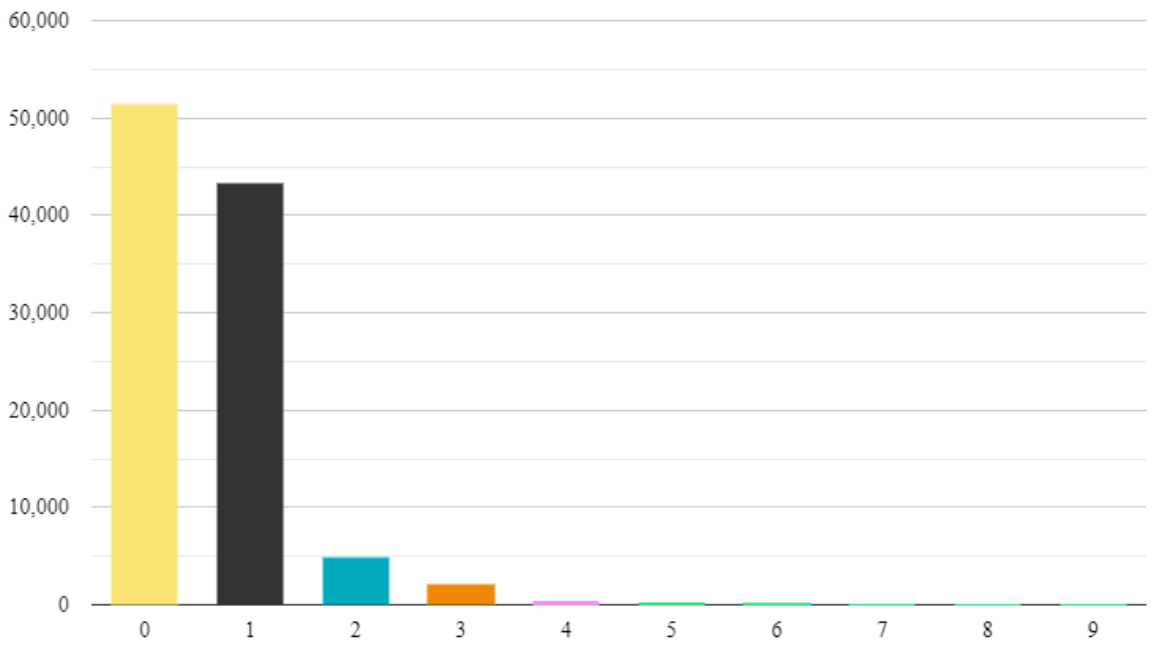


Classes distribution in test set

90/10



Classes distribution in training set



Classes distribution in test set

II. Building the decision tree classifiers

I use the sklearn.tree.DecisionTreeClassifier to build the decision tree in file ipynb for each data set with entropy as the main criterion.

III. Evaluating the decision tree classifiers

40/60 dataset with max_depth of 2

Out of all hands that the model predicted would be nothing, only 51% actually is.

Out of all the hands that are nothing, the model only predicted this outcome correctly for 89% of those hands.

Out of all hands that the model predicted would be one pair, only 47% actually is and only 15% is correct.

The confusion matrix generally predicts how off was the nothing hand in details.

60/40 dataset with max_depth of 2

Out of all hands that the model predicted would be nothing, only 51% actually is and only 96% is correct.

Out of all hands that the model predicted would be one pair, only 48% actually is and only 6% is correct.

80/20 dataset with the max_depth of 7

Out of all hands that the model predicted would be nothing, only 58% actually is and only 76% is correct.

Out of all hands that the model predicted would be one pair, only 52% actually is and only 42% is correct.

Out of all hands that the model predicted would be two pairs, only 44% actually is and none is correct.

Out of all hands that the model predicted would be three of a kind, only 29% actually is and none is correct.

90/10 dataset with the max_depth of 3

Out of all hands that the model predicted would be nothing, only 51% actually is and only 89% is correct.

Out of all hands that the model predicted would be one pair, only 47% actually is and only 15% is correct.

IV. The depth and accuracy of a decision tree

Max_depth	None	2	3	4	5	6	7
accuracy	untracable	0.506341	0.5253	0.525302	0.557731	0.55773	0.558282

So as we see, if we increase the max_depth, the accuracy_score we get will be higher which means the machine will predict the poker hand more accurately.

V. References

https://www.youtube.com/watch?v=wxS5P7yDHRA&t=204s

https://www.youtube.com/watch?v=Kf8cS7ygtUc&t=213s

https://www.youtube.com/watch?v=mpjl4-S_Mho