Decision Trees

Maciej Małecki

August 24, 2019

Introduction

The purpose of my project is to analyze the impact of different hyperparameters on obtained clustering solutions using Decision Trees from sklearn python package. The dataset was 'Social Network Ads', with attributes such as UserID, Gender, Age, EstimatedSalary, Purchased (if advertised item was purchased or not). Table presents first 5 of 400 observations of dataset and table shows basic descriptive statistics.

UserID	Gender	Age	EstimatedSalary	Purchased
15624510	Male	19	19000	0
15810944	Male	35	20000	0
15668575	Female	26	43000	0
15603246	Female	27	57000	0
15804002	Male	19	76000	0

Table 1: First 5 observations of dataset

	Age	EstimatedSalary	Purchased
mean	37.655	69742.5	0.3575
std	10.4829	34096.9603	0.4799
min	18	15000	0
25%	29.75	43000	0
median	37	70000	0
75%	46	88000	1
max	60	150000	1

Table 2: Summary of basic statistics dataset

Comparing of influence of different hyperparameters

max_depth=2

Firstly, we have chosen the max_depth = 2. Classification is based on two attributes – on Age and EstimatedSalary, and the criterion is 'enthropy'. The accuracy for this classification on the test set was 94% and actually it was the best result we have got. The decision tree we can see on 1 and the outputs on figures 2. As we state from these, The clusterisation is simple but yet very effective for data.

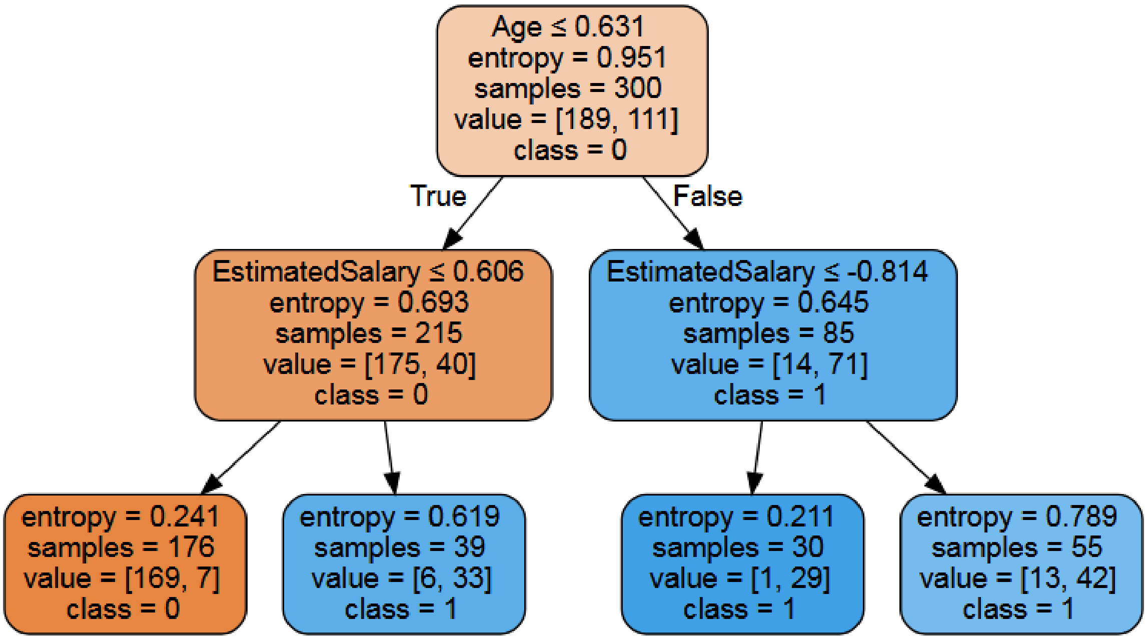


Figure 1: Image of decision tree generated using Graphviz, max_depth=2

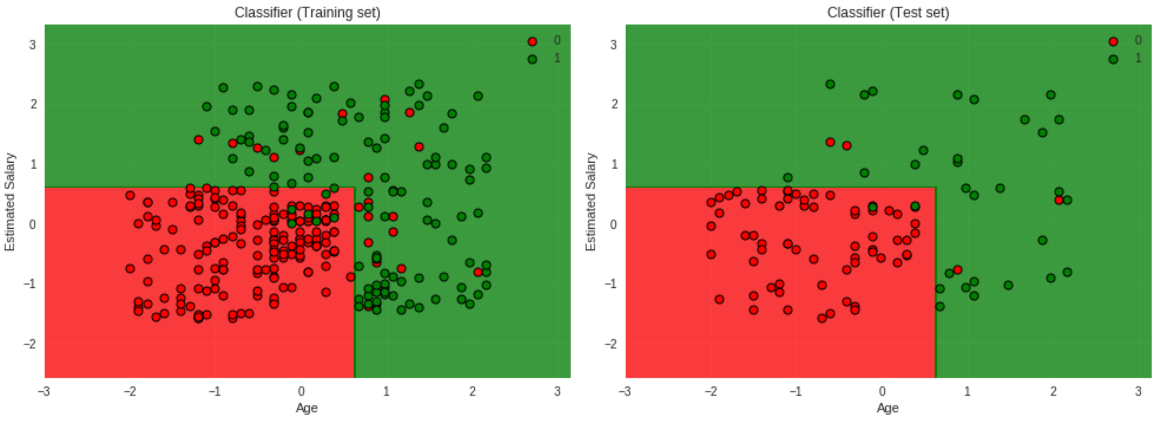


Figure 2: Training and validation visualization, max_depth=2

max_depth=4

In case of max_depth=4, the accuracy was 93%. The decision tree from fig. 3 is starting to be complicated, but as we can see on fig. 4 it is maybe better suited to training data but not necessarily to test data, but accuracy is still satisfying.

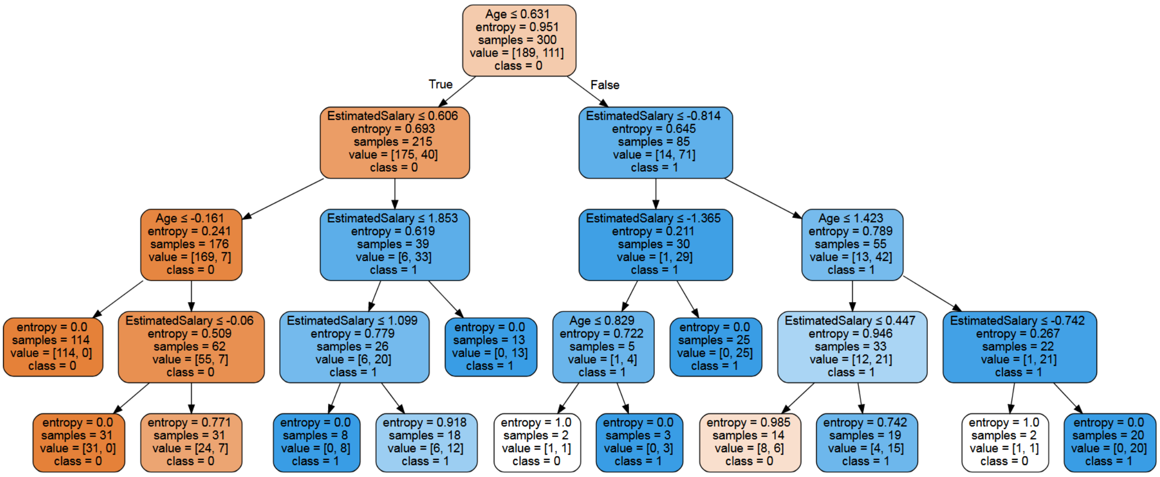


Figure 3: Image of decision tree generated using Graphviz, max_depth=4

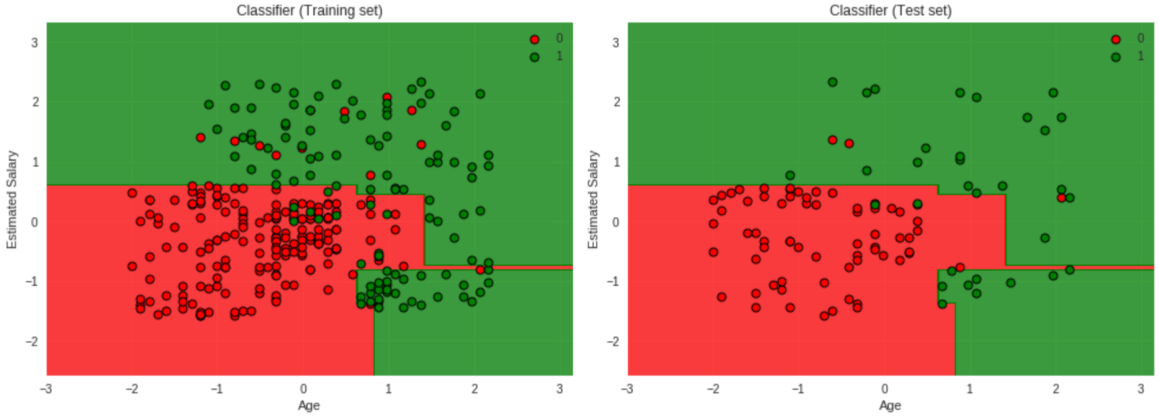


Figure 4: Training and validation visualization, max_depth=4

max_depth=6

The decision tree from fig. 5 is very, maybe even unnecessarily, complicated. With only two attributes it starts to have the characteristics of overfitting. The accuracy is much lower than in previous cases and is equal 90%. In the fig. 6 we can see rectangular-shaped areas which badly classify both training set points and test points.

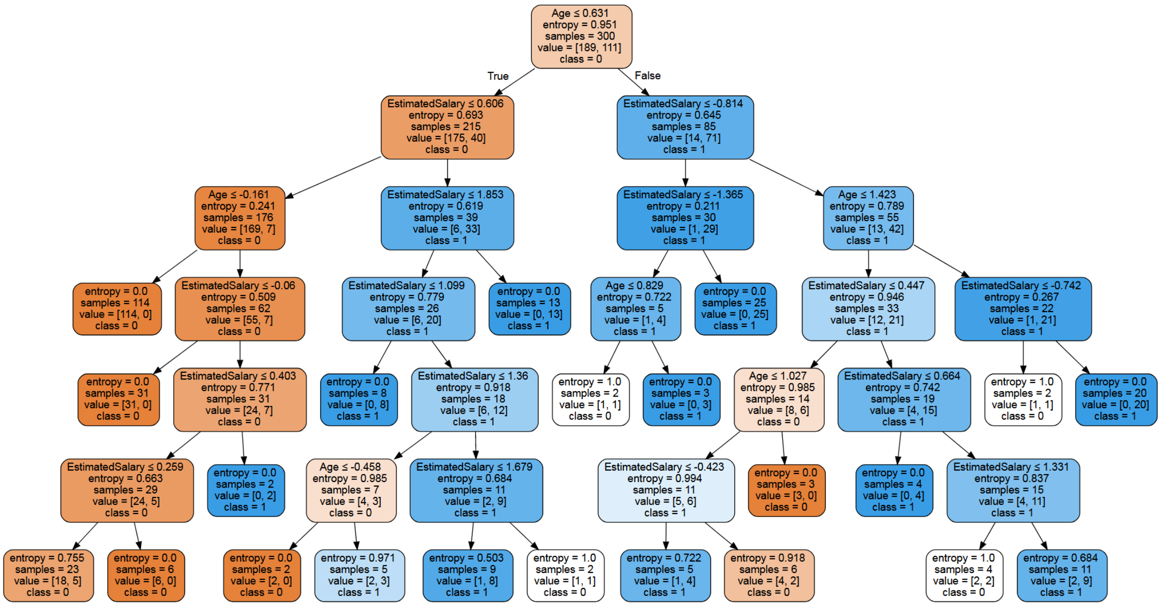


Figure 5: Image of decision tree generated using Graphviz, max_depth=6

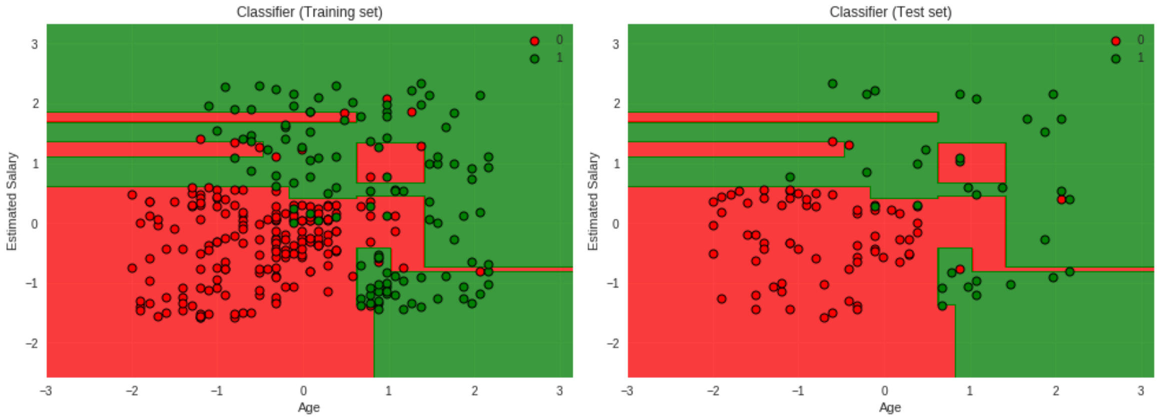
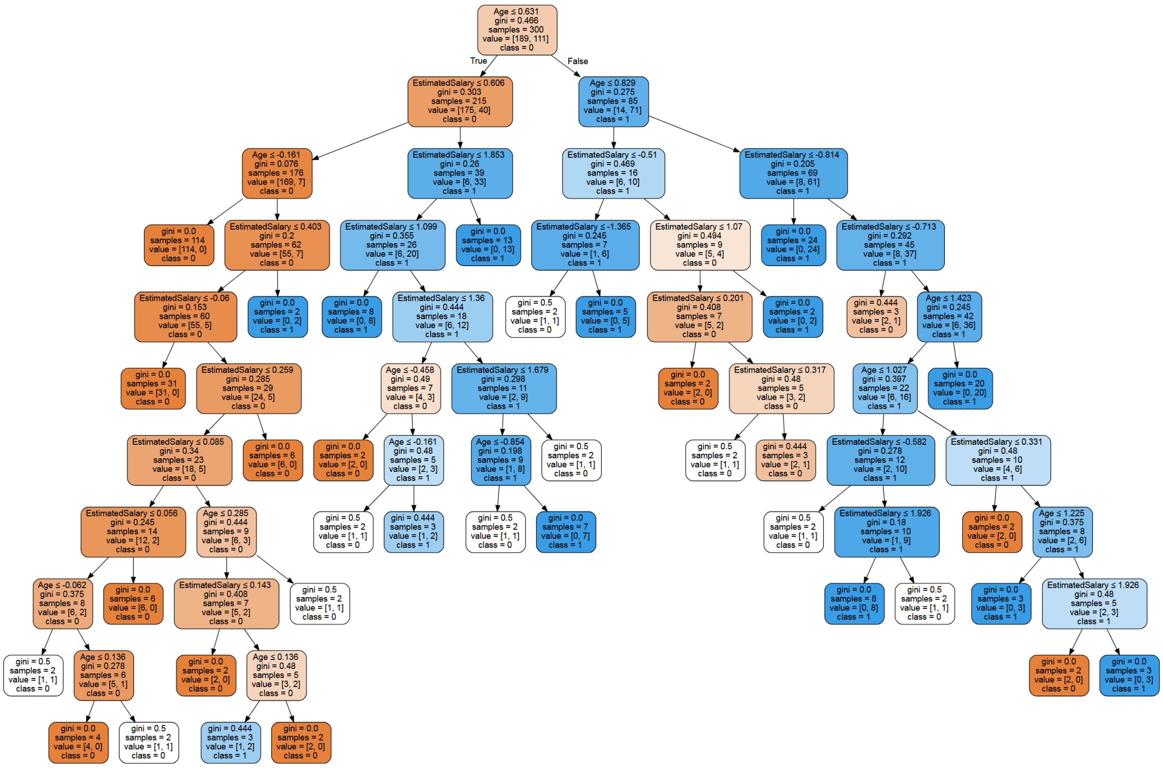


Figure 6: Training and validation visualization max_depth=6

criterion = entropy

For this and the next section tree we have let the sklearn library function to pick the maximal depth of a tree and we wanted to check the influence of of criterion on the output. So the first one is 'entropy' and the tree can be seen on fig. 7. It is even more complicated than for previous trials. The accuracy was also equal 90% for this case and we see that actually the classifier tends to be overfitted – it is good for the training set, but have issues with test data.



Age ≤ 0.631 gini = 0.466 samples = 300 value = [189, 111] class = 0
True
False
EstimatedSalary ≤ 0.606 gini = 0.303 samples = 215 value = [175, 40] class = 0
Age ≤ 0.829 gini = 0.275 samples = 85 value = [14, 71] class = 1
Age ≤ -0.161 gini = 0.076 samples = 176 value = [169, 7] class = 0
EstimatedSalary ≤ 1.853 gini = 0.26 samples = 39 value = [6, 33] class = 1
EstimatedSalary ≤ -0.51 gini = 0.469 samples = 16 value = [6, 10] class = 1
EstimatedSalary ≤ -0.814 gini = 0.205 samples = 69 value = [8, 61] class = 1
gini = 0.0 samples = 114 value = [114, 0] class = 0
EstimatedSalary ≤ 0.403 gini = 0.2 samples = 62 value = [55, 7] class = 0
EstimatedSalary ≤ 1.099 gini = 0.355 samples = 26 value = [6, 20] class = 1
gini = 0.0 samples = 13 value = [0, 13] class = 1
EstimatedSalary ≤ -1.365 gini = 0.245 samples = 7 value = [1, 6] class = 1
EstimatedSalary ≤ 1.07 gini = 0.494 samples = 9 value = [5, 4] class = 0
gini = 0.0 samples = 24 value = [0, 24] class = 1
EstimatedSalary ≤ -0.713 gini = 0.292 samples = 45 value = [8, 37] class = 1
EstimatedSalary ≤ -0.06 gini = 0.153 samples = 60 value = [55, 5] class = 0
gini = 0.0 samples = 2 value = [0, 2] class = 1
gini = 0.0 samples = 8 value = [0, 8] class = 1
EstimatedSalary ≤ 1.36 gini = 0.444 samples = 18 value = [6, 12] class = 1
gini = 0.5 samples = 2 value = [1, 1] class = 0
gini = 0.0 samples = 5 value = [0, 5] class = 1
EstimatedSalary ≤ 0.201 gini = 0.408 samples = 7 value = [5, 2] class = 0
gini = 0.0 samples = 2 value = [0, 2] class = 1
gini = 0.444 samples = 3 value = [2, 1] class = 0
Age ≤ 1.423 gini = 0.245 samples = 42 value = [6, 36] class = 1
gini = 0.0 samples = 31 value = [31, 0] class = 0
EstimatedSalary ≤ 0.259 gini = 0.285 samples = 29 value = [24, 5] class = 0
Age ≤ -0.458 gini = 0.49 samples = 7 value = [4, 3] class = 0
EstimatedSalary ≤ 1.679 gini = 0.298 samples = 11 value = [2, 9] class = 1
gini = 0.0 samples = 2 value = [2, 0] class = 0
EstimatedSalary ≤ 0.317 gini = 0.48 samples = 5 value = [3, 2] class = 0
Age ≤ 1.027 gini = 0.397 samples = 22 value = [6, 16] class = 1
gini = 0.0 samples = 20 value = [0, 20] class = 1
EstimatedSalary ≤ 0.085 gini = 0.34 samples = 23 value = [18, 5] class = 0
gini = 0.0 samples = 6 value = [6, 0] class = 0
gini = 0.0 samples = 2 value = [2, 0] class = 0
Age ≤ -0.161 gini = 0.48 samples = 5 value = [2, 3] class = 1
Age ≤ -0.854 gini = 0.198 samples = 9 value = [1, 8] class = 1
gini = 0.5 samples = 2 value = [1, 1] class = 0
gini = 0.5 samples = 2 value = [1, 1] class = 0
gini = 0.444 samples = 3 value = [2, 1] class = 0
EstimatedSalary ≤ -0.582 gini = 0.278 samples = 12 value = [2, 10] class = 1
EstimatedSalary ≤ 0.331 gini = 0.48 samples = 10 value = [4, 6] class = 1
EstimatedSalary ≤ 0.056 gini = 0.245 samples = 14 value = [12, 2] class = 0
Age ≤ 0.285 gini = 0.444 samples = 9 value = [6, 3] class = 0
gini = 0.5 samples = 2 value = [1, 1] class = 0
gini = 0.444 samples = 3 value = [1, 2] class = 1
gini = 0.5 samples = 2 value = [1, 1] class = 0
gini = 0.0 samples = 7 value = [0, 7] class = 1
gini = 0.5 samples = 2 value = [1, 1] class = 0
EstimatedSalary ≤ 1.926 gini = 0.18 samples = 10 value = [1, 9] class = 1
gini = 0.0 samples = 2 value = [2, 0] class = 0
Age ≤ 1.225 gini = 0.375 samples = 8 value = [2, 6] class = 1
Age ≤ -0.062 gini = 0.375 samples = 8 value = [6, 2] class = 0
gini = 0.0 samples = 6 value = [6, 0] class = 0
EstimatedSalary ≤ 0.143 gini = 0.408 samples = 7 value = [5, 2] class = 0
gini = 0.5 samples = 2 value = [1, 1] class = 0
gini = 0.0 samples = 8 value = [0, 8] class = 1
gini = 0.5 samples = 2 value = [1, 1] class = 0
gini = 0.0 samples = 3 value = [0, 3] class = 1
EstimatedSalary ≤ 1.926 gini = 0.48 samples = 5 value = [2, 3] class = 1
gini = 0.5 samples = 2 value = [1, 1] class = 0
Age ≤ 0.136 gini = 0.278 samples = 6 value = [5, 1] class = 0
gini = 0.0 samples = 2 value = [2, 0] class = 0
Age ≤ 0.136 gini = 0.48 samples = 5 value = [3, 2] class = 0
gini = 0.0 samples = 2 value = [2, 0] class = 0
gini = 0.0 samples = 3 value = [0, 3] class = 1
gini = 0.0 samples = 4 value = [4, 0] class = 0
gini = 0.5 samples = 2 value = [1, 1] class = 0
gini = 0.444 samples = 3 value = [1, 2] class = 1
gini = 0.0 samples = 2 value = [2, 0] class = 0

Figure 7: Image of decision tree generated using Graphviz, criterion = entropy

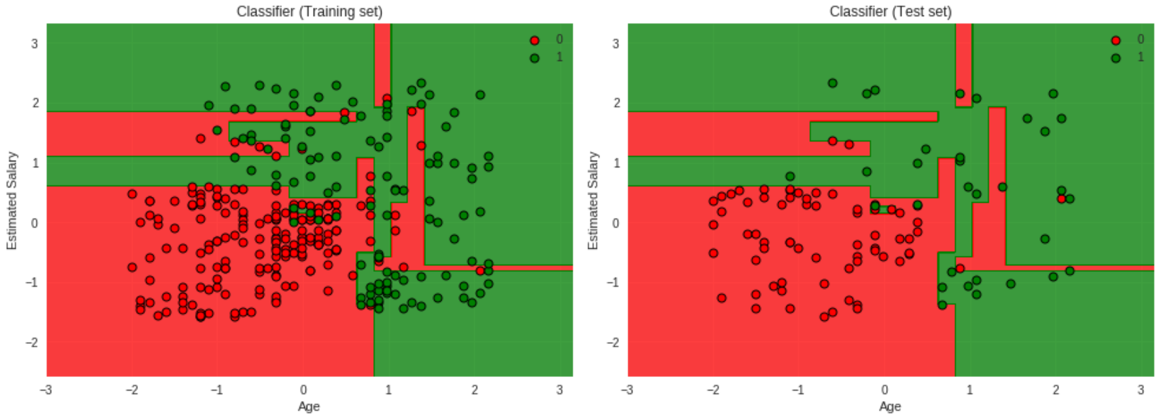


Figure 8: Training and validation visualization, criterion = entropy

criterion = gini

For this case we also let the library to choose depth of the tree with is also more complicated like we can see no fig. 9. Comparing 11 and plot 6 we can state that differences are actually not visible. The accuracy was, the same like in previous cases, 90%.

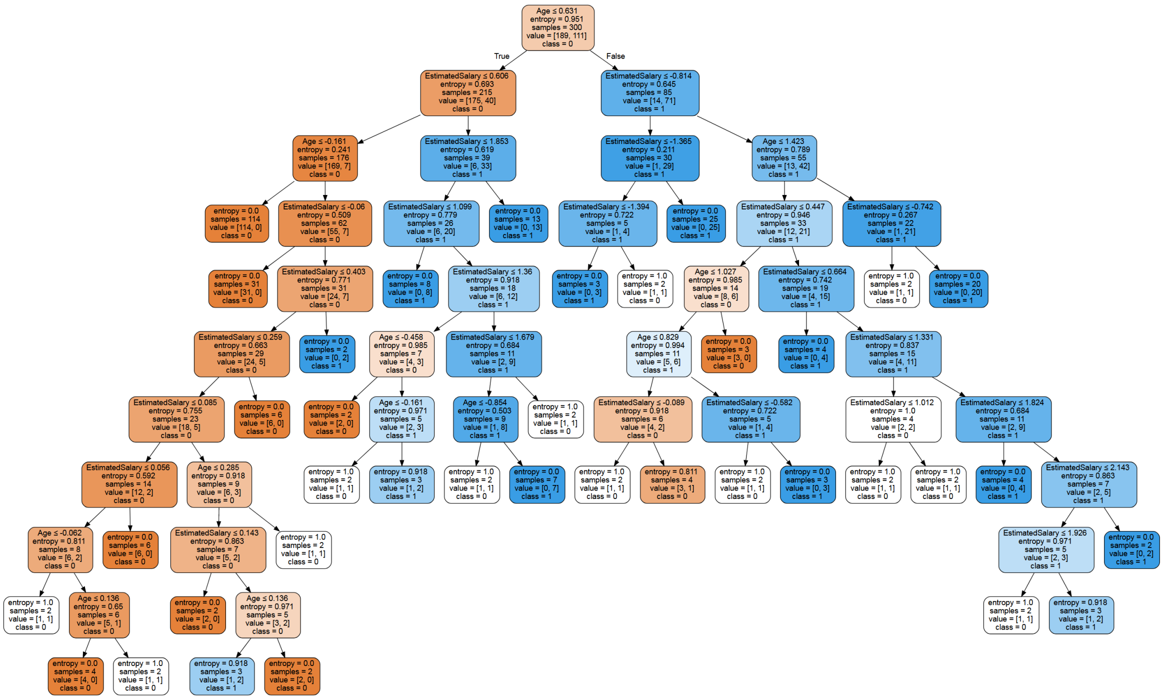

Age ≤ 0.631 entropy = 0.951 samples = 300 value = [189, 111] class = 0
True
False
EstimatedSalary ≤ 0.606 entropy = 0.693 samples = 215 value = [175, 40] class = 0
EstimatedSalary ≤ -0.814 entropy = 0.645 samples = 85 value = [14, 71] class = 1
Age ≤ -0.161 entropy = 0.241 samples = 176 value = [169, 7] class = 0
EstimatedSalary ≤ 1.853 entropy = 0.619 samples = 39 value = [6, 33] class = 1
EstimatedSalary ≤ -1.365 entropy = 0.211 samples = 30 value = [1, 29] class = 1
Age ≤ 1.423 entropy = 0.789 samples = 55 value = [13, 42] class = 1
entropy = 0.0 samples = 114 value = [114, 0] class = 0
EstimatedSalary ≤ -0.06 entropy = 0.509 samples = 62 value = [55, 7] class = 0
EstimatedSalary ≤ 1.099 entropy = 0.779 samples = 26 value = [6, 20] class = 1
entropy = 0.0 samples = 13 value = [0, 13] class = 1
EstimatedSalary ≤ -1.394 entropy = 0.722 samples = 5 value = [1, 4] class = 1
entropy = 0.0 samples = 25 value = [0, 25] class = 1
EstimatedSalary ≤ 0.447 entropy = 0.946 samples = 33 value = [12, 21] class = 1
EstimatedSalary ≤ -0.742 entropy = 0.267 samples = 22 value = [1, 21] class = 1
entropy = 0.0 samples = 31 value = [31, 0] class = 0
EstimatedSalary ≤ 0.403 entropy = 0.771 samples = 31 value = [24, 7] class = 0
entropy = 0.0 samples = 8 value = [0, 8] class = 1
EstimatedSalary ≤ 1.36 entropy = 0.918 samples = 18 value = [6, 12] class = 1
entropy = 0.0 samples = 3 value = [0, 3] class = 1
entropy = 1.0 samples = 2 value = [1, 1] class = 0
Age ≤ 1.027 entropy = 0.985 samples = 14 value = [8, 6] class = 0
EstimatedSalary ≤ 0.664 entropy = 0.742 samples = 19 value = [4, 15] class = 1
entropy = 1.0 samples = 2 value = [1, 1] class = 0
entropy = 0.0 samples = 20 value = [0, 20] class = 1
EstimatedSalary ≤ 0.259 entropy = 0.663 samples = 29 value = [24, 5] class = 0
entropy = 0.0 samples = 2 value = [0, 2] class = 1
Age ≤ -0.458 entropy = 0.985 samples = 7 value = [4, 3] class = 0
EstimatedSalary ≤ 1.679 entropy = 0.684 samples = 11 value = [2, 9] class = 1
Age ≤ 0.829 entropy = 0.994 samples = 11 value = [5, 6] class = 1
entropy = 0.0 samples = 3 value = [3, 0] class = 0
entropy = 0.0 samples = 4 value = [0, 4] class = 1
EstimatedSalary ≤ 1.331 entropy = 0.837 samples = 15 value = [4, 11] class = 1
EstimatedSalary ≤ 0.085 entropy = 0.755 samples = 23 value = [18, 5] class = 0
entropy = 0.0 samples = 6 value = [6, 0] class = 0
entropy = 0.0 samples = 2 value = [2, 0] class = 0
Age ≤ -0.161 entropy = 0.971 samples = 5 value = [2, 3] class = 1
Age ≤ -0.854 entropy = 0.503 samples = 9 value = [1, 8] class = 1
entropy = 1.0 samples = 2 value = [1, 1] class = 0
EstimatedSalary ≤ -0.089 entropy = 0.918 samples = 6 value = [4, 2] class = 0
EstimatedSalary ≤ -0.582 entropy = 0.722 samples = 5 value = [1, 4] class = 1
EstimatedSalary ≤ 1.012 entropy = 1.0 samples = 4 value = [2, 2] class = 0
EstimatedSalary ≤ 1.824 entropy = 0.684 samples = 11 value = [2, 9] class = 1
EstimatedSalary ≤ 0.056 entropy = 0.592 samples = 14 value = [12, 2] class = 0
Age ≤ 0.285 entropy = 0.918 samples = 9 value = [6, 3] class = 0
entropy = 1.0 samples = 2 value = [1, 1] class = 0
entropy = 0.918 samples = 3 value = [1, 2] class = 1
entropy = 1.0 samples = 2 value = [1, 1] class = 0
entropy = 0.0 samples = 7 value = [0, 7] class = 1
entropy = 1.0 samples = 2 value = [1, 1] class = 0
entropy = 0.811 samples = 4 value = [3, 1] class = 0
entropy = 1.0 samples = 2 value = [1, 1] class = 0
entropy = 0.0 samples = 3 value = [0, 3] class = 1
entropy = 1.0 samples = 2 value = [1, 1] class = 0
entropy = 1.0 samples = 2 value = [1, 1] class = 0
entropy = 0.0 samples = 4 value = [0, 4] class = 1
EstimatedSalary ≤ 2.143 entropy = 0.863 samples = 7 value = [2, 5] class = 1
Age ≤ -0.062 entropy = 0.811 samples = 8 value = [6, 2] class = 0
entropy = 0.0 samples = 6 value = [6, 0] class = 0
EstimatedSalary ≤ 0.143 entropy = 0.863 samples = 7 value = [5, 2] class = 0
entropy = 1.0 samples = 2 value = [1, 1] class = 0
EstimatedSalary ≤ 1.926 entropy = 0.971 samples = 5 value = [2, 3] class = 1
entropy = 0.0 samples = 2 value = [0, 2] class = 1
entropy = 1.0 samples = 2 value = [1, 1] class = 0
Age ≤ 0.136 entropy = 0.65 samples = 6 value = [5, 1] class = 0
entropy = 0.0 samples = 2 value = [2, 0] class = 0
Age ≤ 0.136 entropy = 0.971 samples = 5 value = [3, 2] class = 0
entropy = 1.0 samples = 2 value = [1, 1] class = 0
entropy = 0.918 samples = 3 value = [1, 2] class = 1
entropy = 0.0 samples = 4 value = [4, 0] class = 0
entropy = 1.0 samples = 2 value = [1, 1] class = 0
entropy = 0.918 samples = 3 value = [1, 2] class = 1
entropy = 0.0 samples = 2 value = [2, 0] class = 0

Figure 9: Image of decision tree generated using Graphviz, criterion = gini

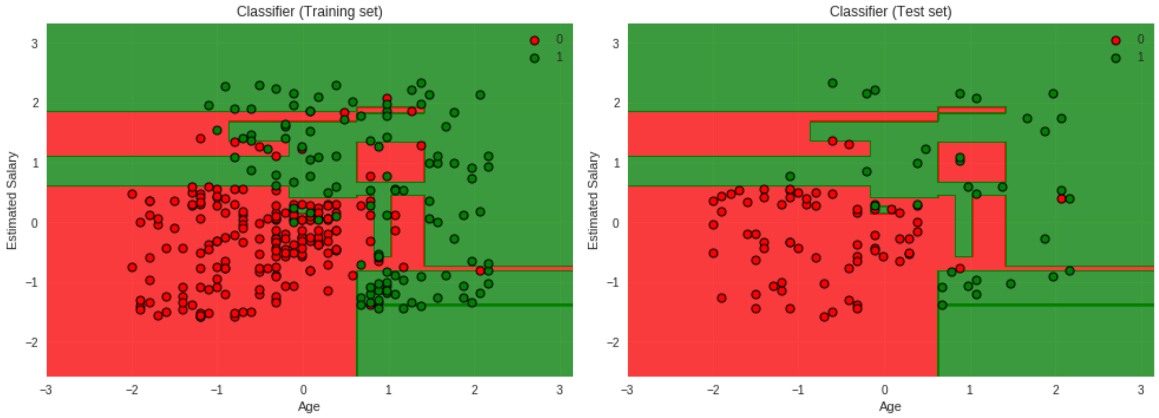


Figure 10: Training and validation visualization, criterion = gini

splitter='random'

This time we wanted to check what is the influence of the splitter, because previously we always had splitter set to best. Now accuracy was accuracy = 87.0% for max_depth=4, comparing to 93% for the earlier tree for max_depth=4.

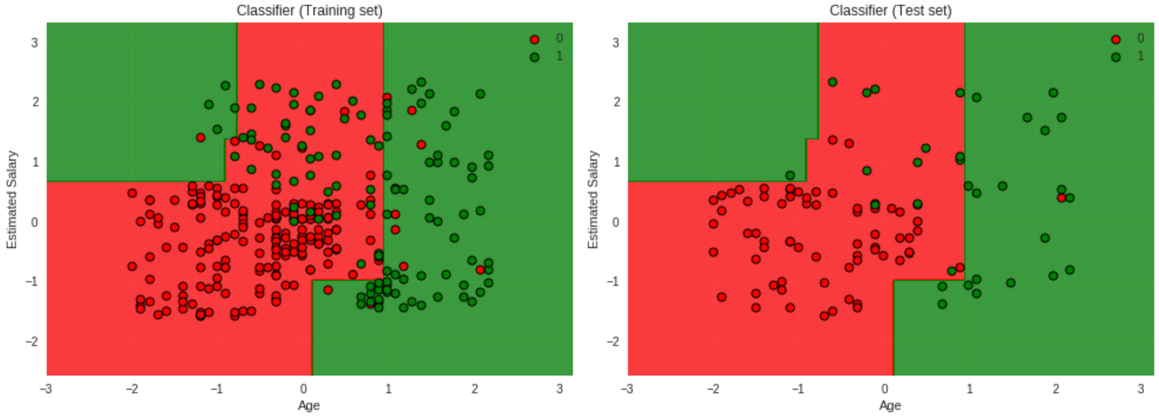


Figure 11: Training and validation visualization, splitter='random'

Conclusions

As we could see on my examples not always bigger depth of a tree gives better results. The trees with too big depth can have better results on training sets, but they tend to be overfitted and gave poorer results on test set. It is also important to check available criterion functions which measure the quality of a split, because sometimes for the same depth, they can us give different results as everything depends on data that we have.