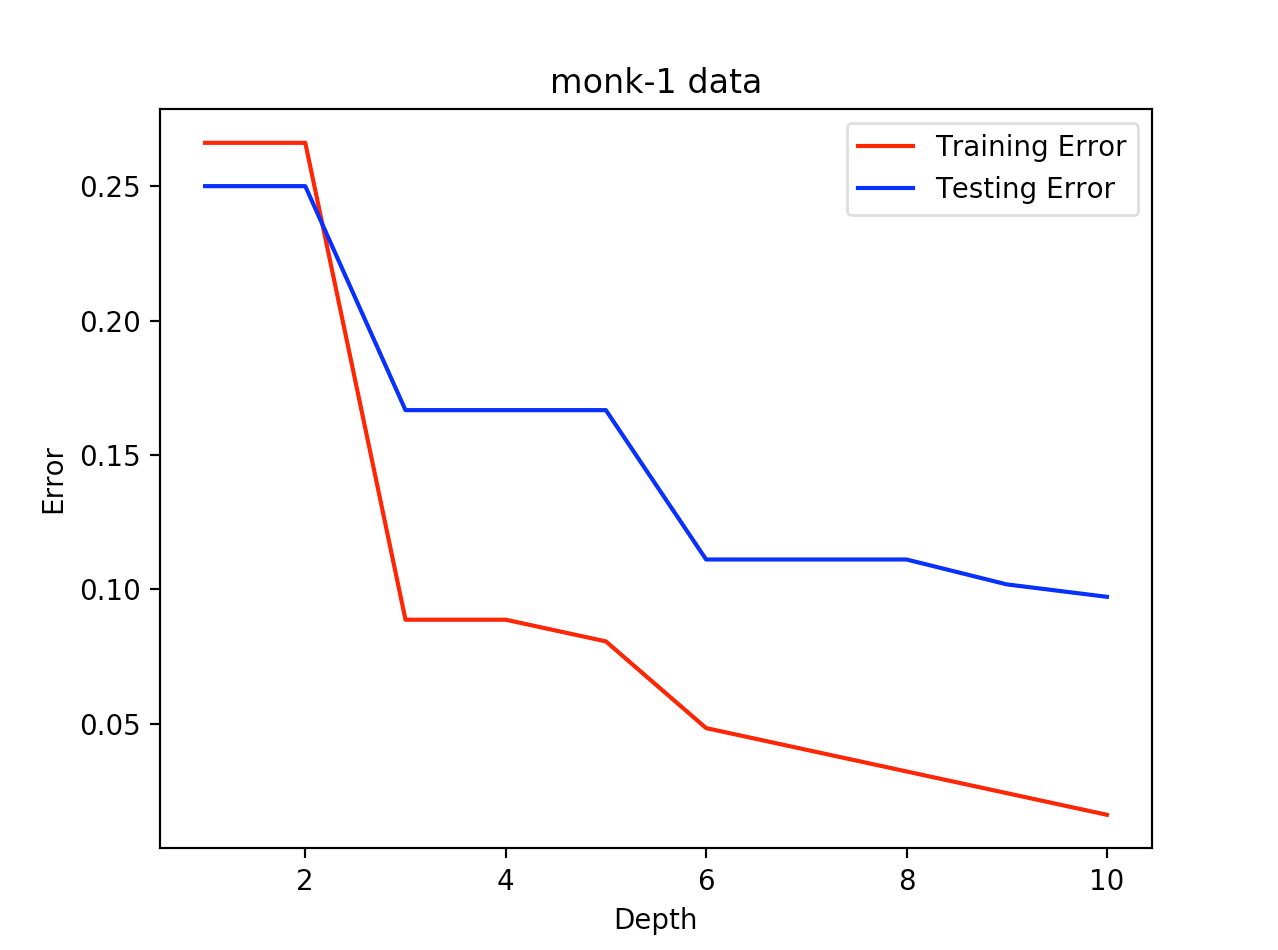
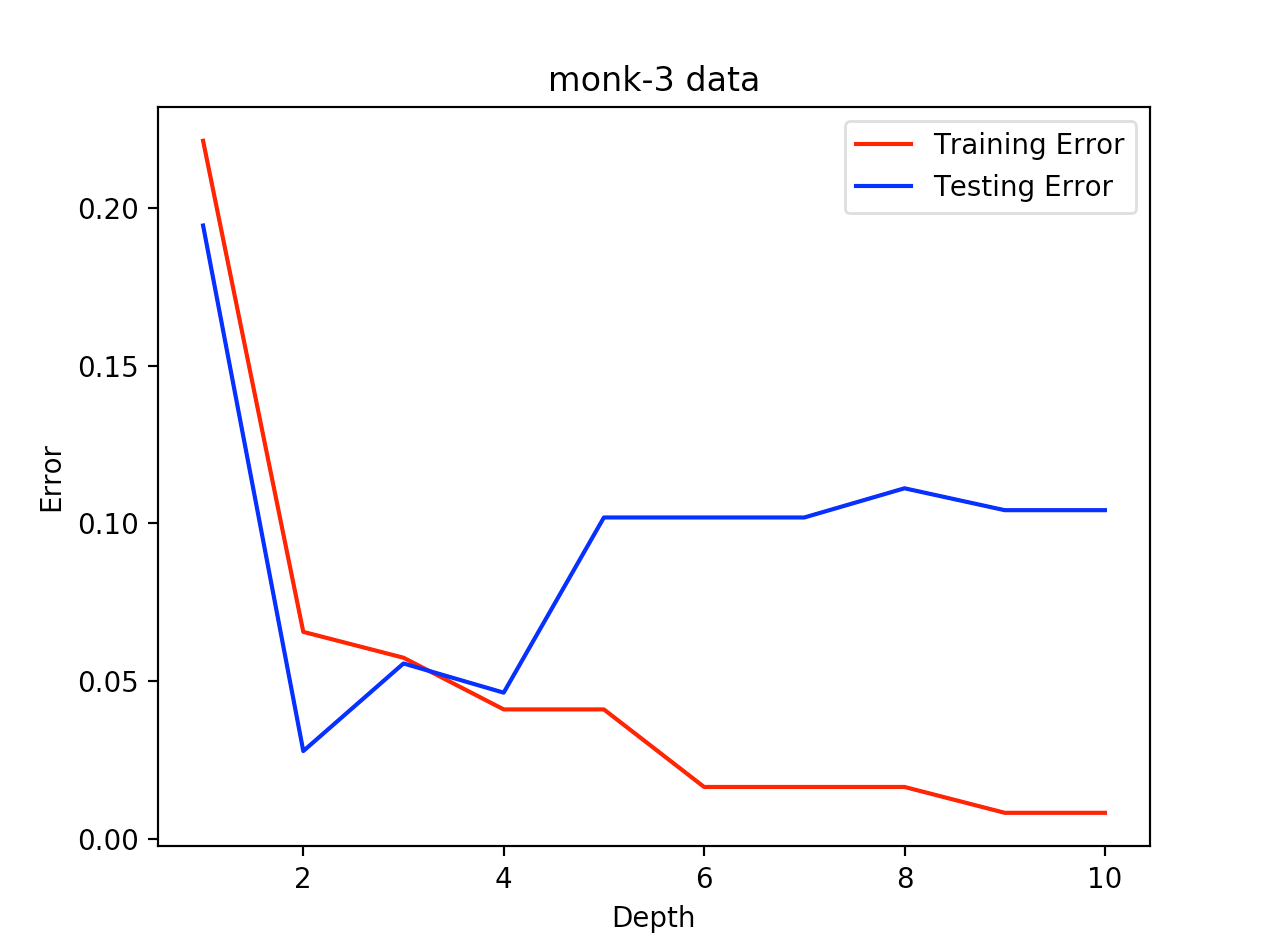
**b.** The following are the graphs obtained by the binary id3 over the three datasets with varying depth from 1-10.







**c.** Decision tree on monk-1.data with depths 1,3,5 are shown below along with their confusion matrix and test error

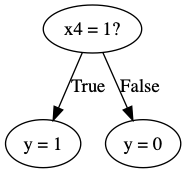
**DEPTH= 1**

**Confusion matrix**

[[216 0]

[108 108]]

Test Error = 25.00%.



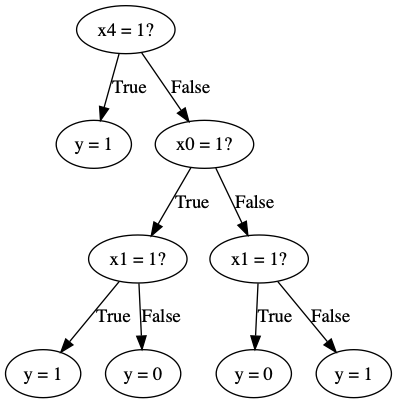
**DEPTH= 3**

**Confusion matrix**

[[144 72]

[ 0 216]]

Test Error = 16.67%.



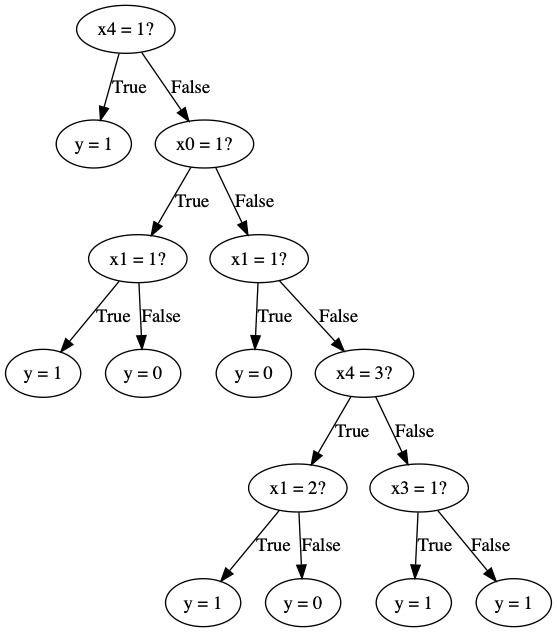
**DEPTH= 5**

**Confusion matrix**

[[156 60]

[ 12 204]]

Test Error = 16.67%.



**d.** We can import sklearn and use the DecisionTreeClassifier to train a decision tree using the open machine learning library. The code to import and for the model is given below. We have also used predict and confusion matrix functions from Scikit learn.

*from* sklearn.metrics *import* confusion\_matrix

*from* sklearn *import* tree

*for* dept in range(1,6,2):

clf = tree.DecisionTreeClassifier(

                criterion='entropy', max\_depth=dept)

clf.fit(Xtrn, ytrn)

  y\_pred = clf.predict(Xtst)

  test\_err = compute\_error(ytst, y\_pred)\*100

  print("Test Error = {0:.2f}%".format(test\_err))

print(confusion\_matrix(ytst, y\_pred))

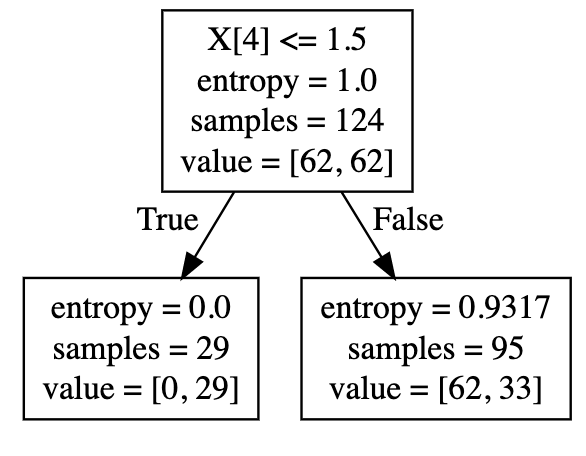
Using scikit learn we get the following error, confusion matrix and the decision tree.

**Depth = 1**

Test Error = 25.00%

[[216 0]

[108 108]]

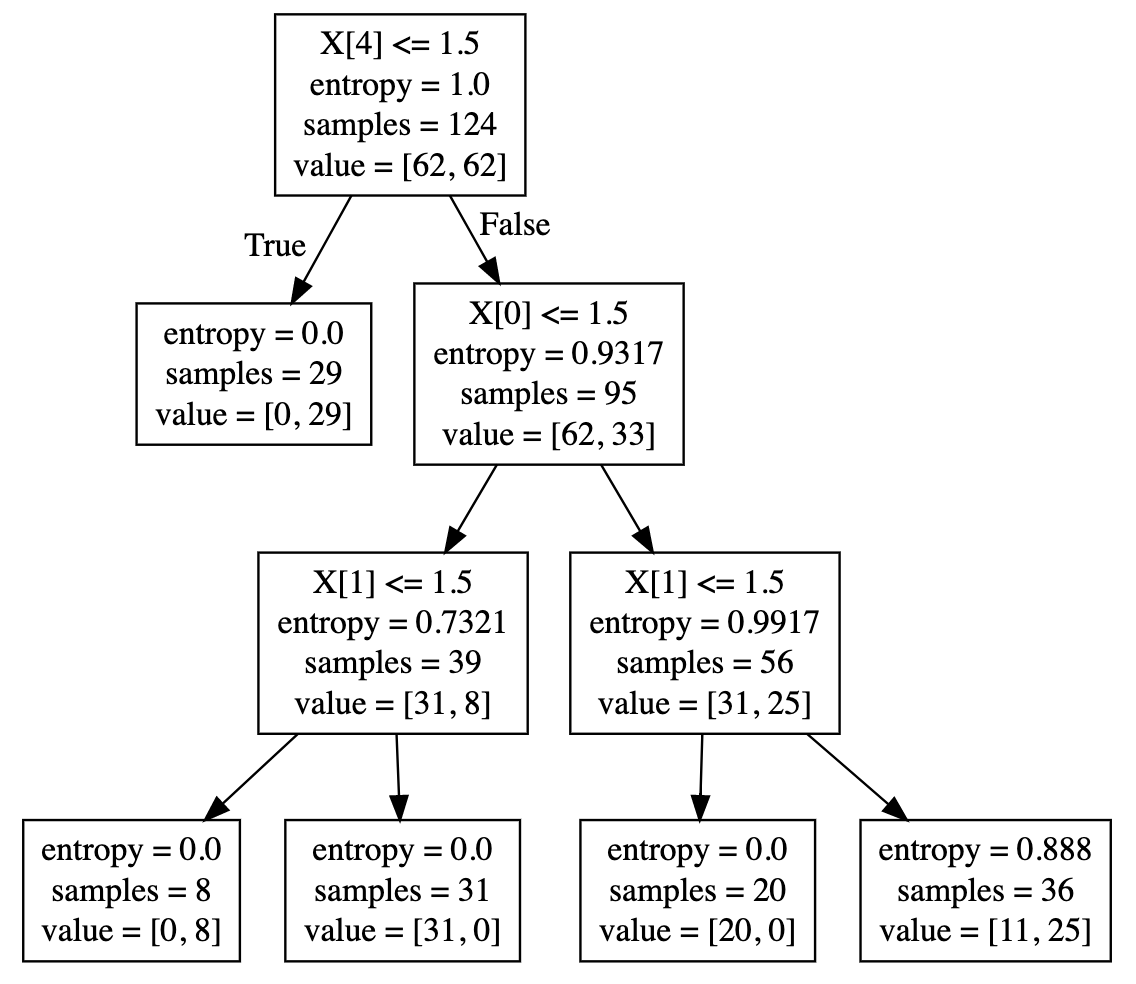


**Depth = 3**

Test Error = 16.67%

[[144 72]

[ 0 216]]

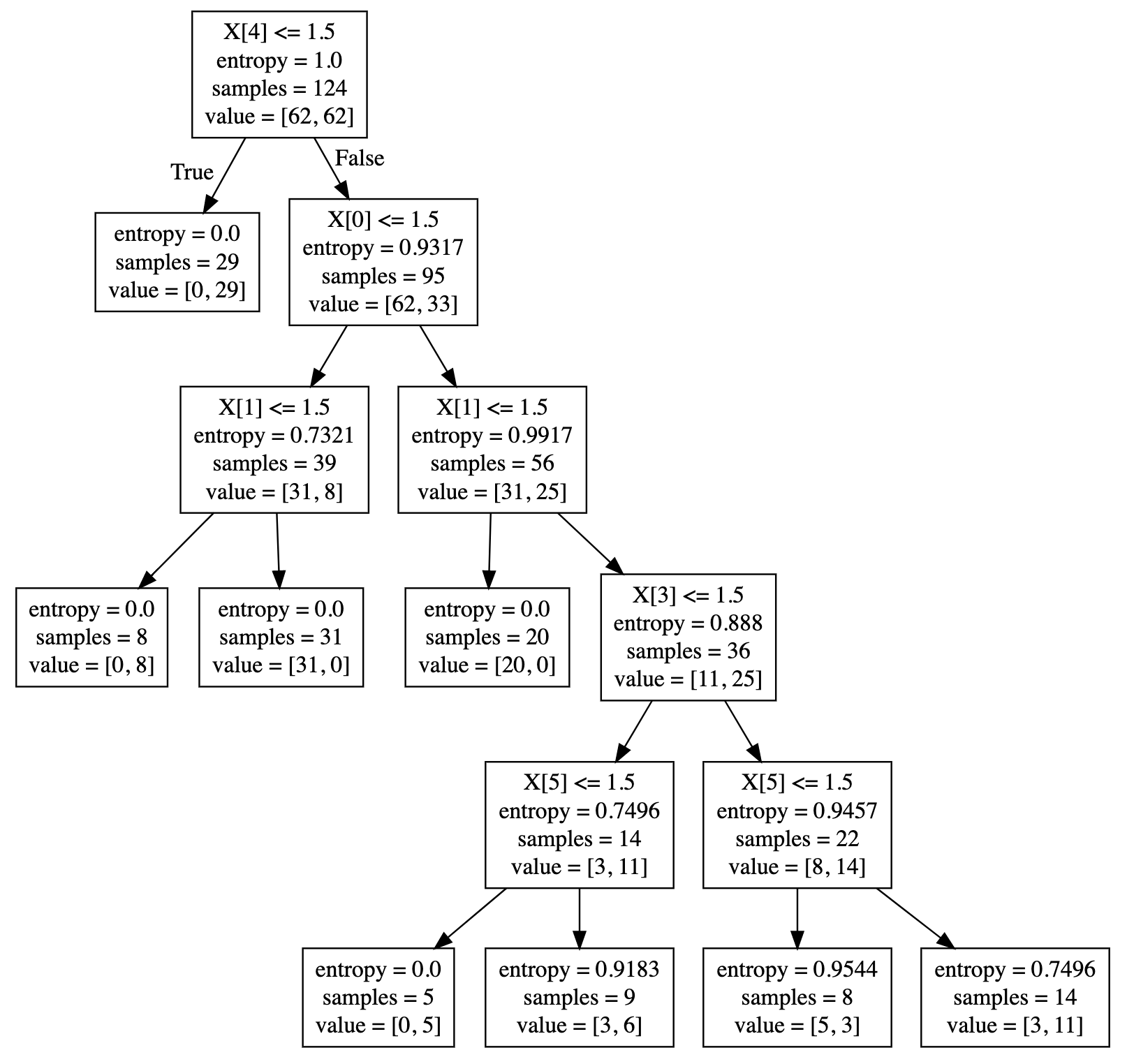


**Depth = 5**

Test Error = 16.67%

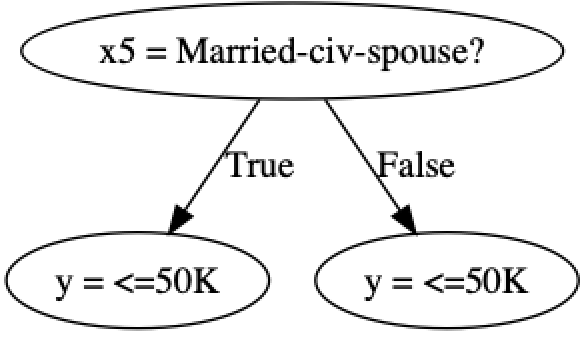
[[168 48]

[ 24 192]]



e.

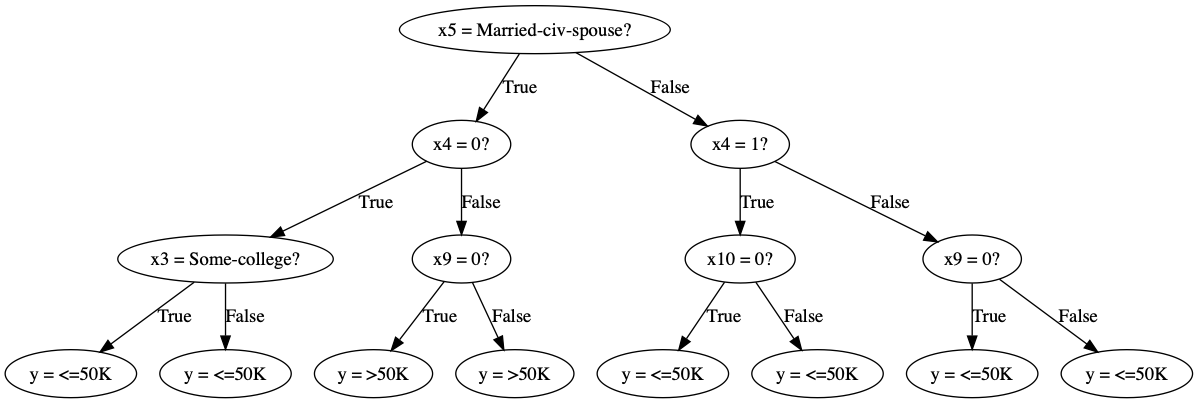
**Depth = 1**

[[4945 0] 

[1568 0]]

Test Error = 24.07%.

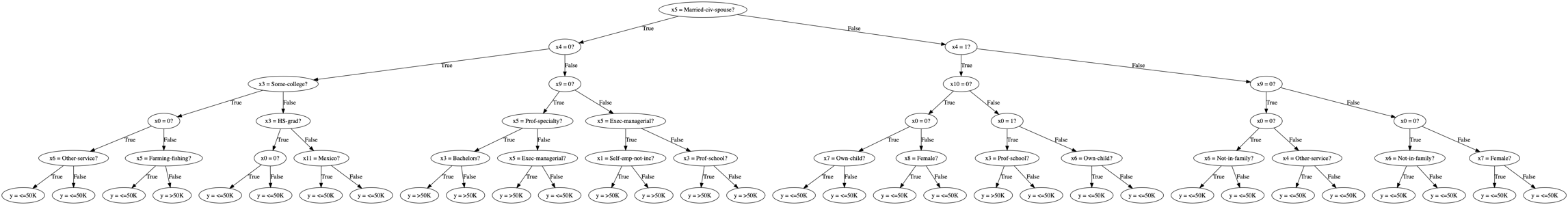
**Depth = 2**



[[4578 367]

[ 844 724]]

Test Error = 18.59%.

**Depth = 5**

[[4397 548]

[ 651 917]]

Test Error = 18.41%.

**USING SKLEARN**

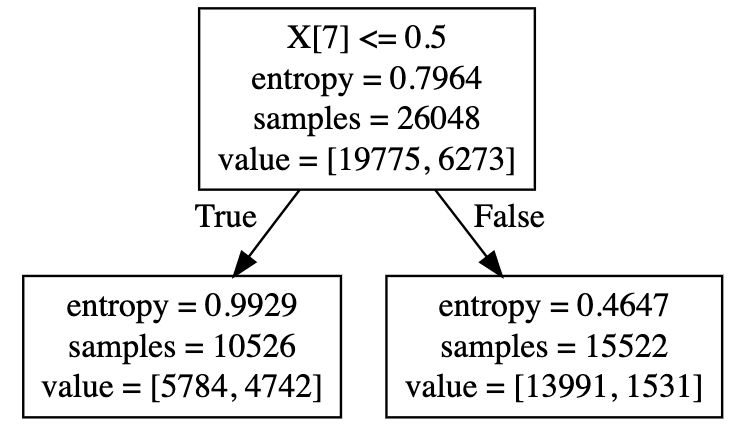
Sklearn doesn’t work on string values, wo we use LabelEncoder to encode the data before sending it to the classifier

**Depth = 1**

Test Error = 24.07%

[[4945 0]

[1568 0]]

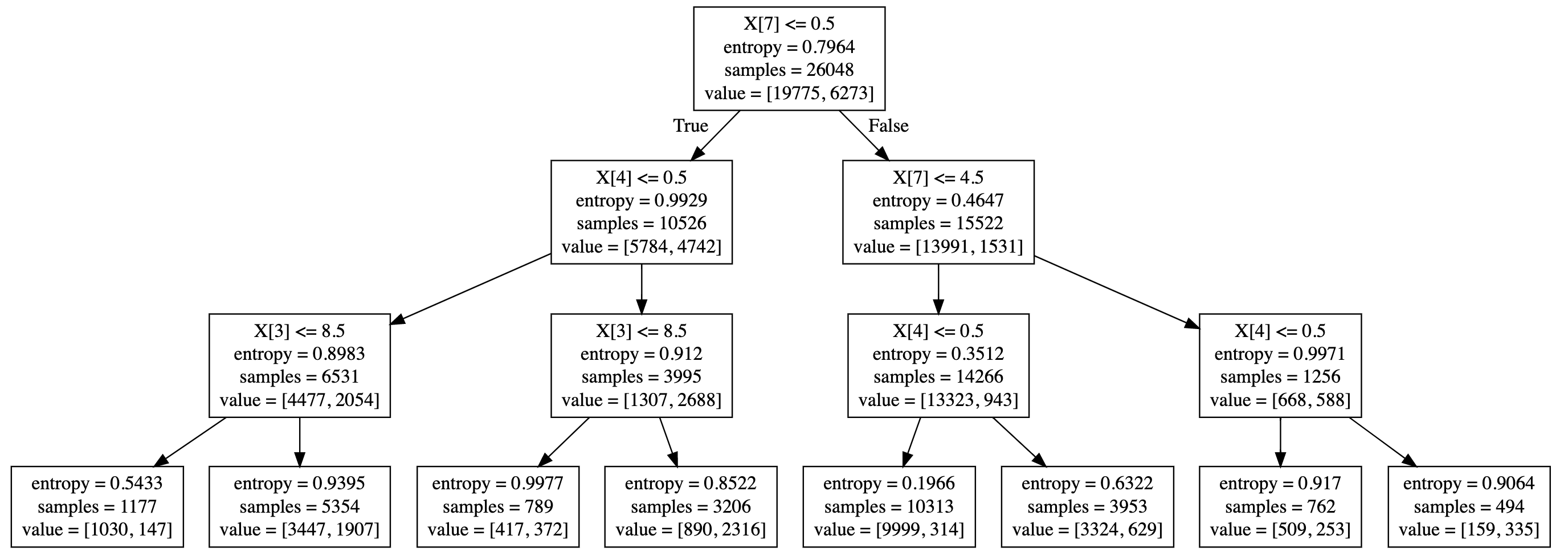


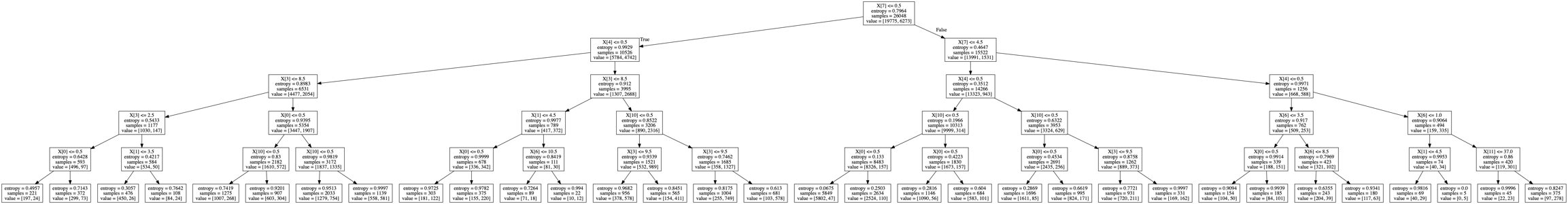
**Depth = 3**

Test Error = 18.01%

[[4697 248]

[ 925 643]]



**Depth = 5**

Test Error = 17.89%

[[4479 466]

[ 699 869]]

**Conclusion**

On a brief look at the results above we can see that on the monk’s data , our id3 model and sklearn model gave the exact same error and even trees. On using this with Income data obtained from UCI repository, the results are slightly different with sklearn giving a lesser error than our id3 model.

Comparing the models with various depths, it is apparent that for monk’s data a depth of 3 seems optimal. Even if we increase the depth to 5, the reduction in testing error is insignificant and any value more than that just overfits the data.