**I. Pen-and-paper**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  | Output |
|  | 1 | 1 | 0 | N |
|  | 1 | 1 | 1 | N |
|  | 0 | 2 | 1 | N |
|  | 1 | 2 | 0 | N |
|  | 2 | 0 | 1 | P |
|  | 1 | 1 | 0 | P |
|  | 2 | 0 | 0 | P |
|  | 0 | 2 | 1 | P |

Choose as the root of the decision tree. When the leaf of the tree is P.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | Output |
|  | 2 | 1 | N |
|  | 2 | 1 | P |

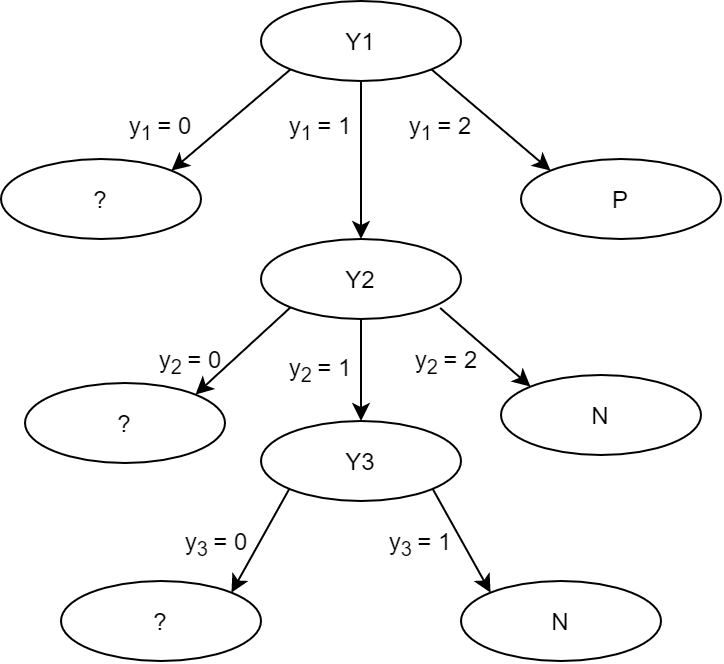
When the leaf of the tree is uncertain.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | Output |
|  | 1 | 0 | N |
|  | 1 | 1 | N |
|  | 2 | 0 | N |
|  | 1 | 0 | P |

Choose as the leaf for on the decision tree. When the leaf of the tree is N. There is no information about , therefore that leaf is uncertain.

|  |  |  |
| --- | --- | --- |
|  |  | Output |
|  | 0 | N |
|  | 1 | N |
|  | 0 | P |

When the leaf of the tree is N. When the leaf of the tree is uncertain.



Some of the leaves are uncertain because we don’t have enough training data to complete the tree.

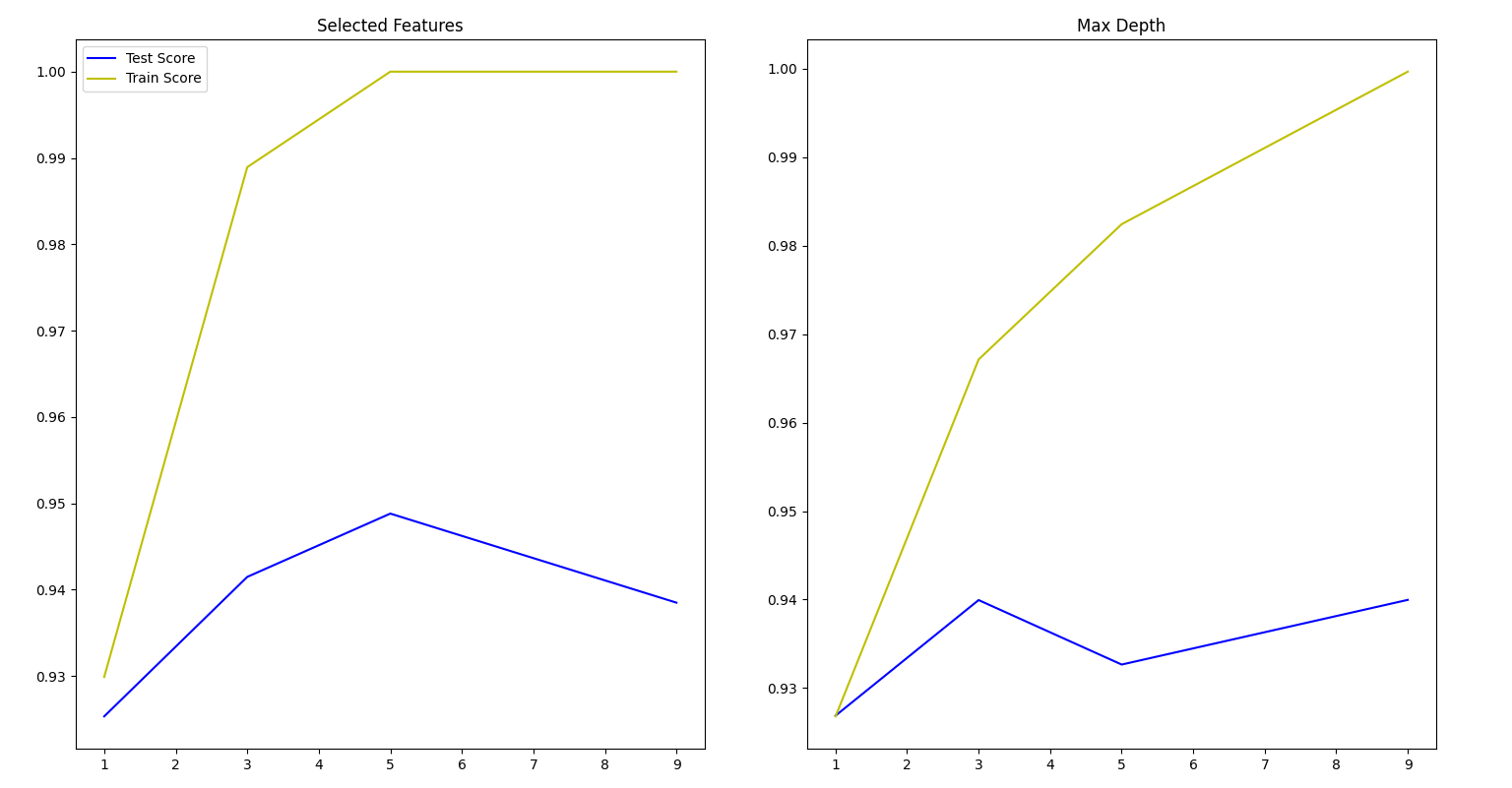
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  | Output |
|  | 2 | 0 | 0 | N |
|  | 1 | 2 | 0 | P |

Decision Tree for : , is classified as P

Decision Tree for : , is classified as N

The decision tree has an accuracy of

**II. Programming and critical analysis**

1. 
2. Two reasons for the correlation between graphs is:
3. The increase in features and depth both cause overfitting in order to better fit the training data, and therefore increases in training scores and decreases in testing scores;
4. The depth and features are both related to the number of nodes in the tree, a node corresponds to a feature and belongs to a specific depth, therefore they both behave similarly.
5. Looking at the graph with varying max depth, we can see that maximum depth = 3 is the one that gives a better testing score. This happens because the more depth we give the more overfitting the model is going to have in order to better fit the training data, which can be seen by the increase in training score with bigger depth, and a reduced testing score.

**III. APPENDIX**

from sklearn import tree

from scipy.io import arff

import numpy as num

import matplotlib.pyplot as plt

from sklearn.feature\_selection import SelectKBest, mutual\_info\_classif

from sklearn.model\_selection import cross\_validate, KFold

file = open("breast.w.arff", "r")

data, meta = arff.loadarff(file)

input = data[meta.names()[:-1]].tolist()

output = data["Class"].tolist()

**#i**

slFeatures = [[],[]]

kFol = KFold(n\_splits=10, shuffle=True, random\_state=47)

for i in [1,3,5,9]:

    input\_new = SelectKBest(mutual\_info\_classif, k=i).fit\_transform(input,output)

    classifier = tree.DecisionTreeClassifier(criterion='entropy')

    crossRes = cross\_validate(classifier, input\_new, output, scoring = 'accuracy', cv = kFol, return\_train\_score=True)

    slFeatures[0].append(num.average(crossRes['test\_score']))

    slFeatures[1].append(num.average(crossRes['train\_score']))

**#ii**

slDepth=[[],[]]

for i in [1,3,5,9]:

    classifier = tree.DecisionTreeClassifier(criterion='entropy', max\_depth=i)

    crossRes = cross\_validate(classifier, input, output, scoring = 'accuracy', cv = kFol, return\_train\_score=True)

    slDepth[0].append(num.average(crossRes['test\_score']))

    slDepth[1].append(num.average(crossRes['train\_score']))

plt.subplot(1,2,1)

plt.plot([1,3,5,9], slFeatures[0], 'b-', label="Test Score")

plt.plot([1,3,5,9], slFeatures[1], 'y-', label="Train Score")

plt.title("Selected Features")

plt.legend()

plt.subplot(1,2,2)

plt.plot([1,3,5,9], slDepth[0], 'b-')

plt.plot([1,3,5,9], slDepth[1], 'y-')

plt.title("Max Depth")

plt.show()

**END**