TÉCNICO LISBOA

Aprendizagem 2021/22

Homework I – Group G047

I. Pen-and-paper

1) $X_{train} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 5 \\ 0 & 2 & 4 \\ 1 & 2 & 3 \\ 2 & 0 & 7 \\ 1 & 1 & 1 \\ 2 & 0 & 2 \\ 0 & 2 & 9 \end{bmatrix}$ $\|X_{train}\| = \begin{bmatrix} 1.4142 \\ 5.1962 \\ 4.4721 \\ 3.7417 \\ 7.2801 \\ 1.7321 \\ 2.8284 \\ 9.2195 \end{bmatrix}$ $\phi_{train} = \begin{bmatrix} 1.0000 & 1.4142 & 2.0000 & 2.8283 \\ 1.0000 & 5.1962 & 27.0005 & 140.3000 \\ 1.0000 & 4.4721 & 19.9997 & 89.4406 \\ 1.0000 & 3.7417 & 14.0003 & 52.3850 \\ 1.0000 & 7.2801 & 52.9999 & 385.8443 \\ 1.0000 & 1.7321 & 3.0002 & 5.1966 \\ 1.0000 & 2.8284 & 7.9998 & 22.6268 \\ 1.0000 & 9.2195 & 84.9992 & 783.6499 \end{bmatrix}$ $Z_{train} = \begin{bmatrix} 1 \\ 3 \\ 2 \\ 0 \\ 6 \\ 4 \\ 5 \\ 7 \end{bmatrix}$

$$W = (\phi_{train}^T \times \phi_{train})^{-1} \times \phi_{train}^T \times Z = \begin{bmatrix} 4.5831 \\ -1.6868 \\ 0.3376 \\ -0.0133 \end{bmatrix}$$

2) $X_{test} = \begin{bmatrix} 2 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$ $\|X_{test}\| = \begin{bmatrix} 2.0000 \\ 2.4495 \end{bmatrix}$ $\phi_{test} = \begin{bmatrix} 1.0000 & 2.0000 & 4.0000 & 8.0000 \\ 1.0000 & 2.4495 & 6.0001 & 14.6971 \end{bmatrix}$ $Z_{test} = \begin{bmatrix} 2 \\ 4 \end{bmatrix}$

$$\hat{Z}_{test} = \phi_{test} \times W = \begin{bmatrix} 2.4535 \\ 2.2814 \end{bmatrix} \quad RMSE(\hat{Z}_{test}, Z_{test}) = \sqrt{\frac{1}{2}\sum_{i=1}^{2}(Z_{test_i} - \hat{Z}_{test_i})^2} = 1.2568$$

3) $median(y_3) = 3.5$ $y_3 = \begin{cases} 0, & y < 3.5 \\ 1, & y > 3.5 \end{cases}$ $E(y_i) = -\sum_{v \in y_i} p(v) \log_2 p(v)$ $E(Output|y_i) = \sum_{v \in y_i} p(v) E(Output|v)$

$IG(Output|y_i) = E(Output) - E(Output|y_i)$

	y_1	y_2	y_3	Output
x_1	1	1	0	N
x_2	1	1	1	N
x_3	0	2	1	N
x_4	1	2	0	N
x_5	2	0	1	P
x_6	1	1	0	P
x_7	2	0	0	P
x_8	0	2	1	P

$$E(Output) = E\left(\frac{1}{2}, \frac{1}{2}\right) = 1$$

$$IG(Output|y_1) = 1 - \left(\frac{1}{4}E\left(\frac{1}{2}, \frac{1}{2}\right) + \frac{1}{2}E\left(\frac{3}{4}, \frac{1}{4}\right) + \frac{1}{4}E(1)\right) = 0.3444$$

$$IG(Output|y_2) = 1 - \left(\frac{1}{4}E(1) + \frac{3}{8}E\left(\frac{2}{3}, \frac{1}{3}\right) + \frac{3}{8}E\left(\frac{2}{3}, \frac{1}{3}\right)\right) = 0.3113$$

$$IG(Output|y_3) = 1 - \left(\frac{1}{2}E\left(\frac{1}{2}, \frac{1}{2}\right) + \frac{1}{2}E\left(\frac{1}{2}, \frac{1}{2}\right)\right) = 0$$

Choose y_1 as the root of the decision tree. When $y_1 = 2$ the leaf of the tree is P.

$y_1 = 0$	y_2	y_3	Output
x_3	2	1	N
x_8	2	1	P

$$E(Output) = E\left(\frac{1}{2}, \frac{1}{2}\right) = 1$$

$$IG(Output|y_2) = 1 - \left(E\left(\frac{1}{2}, \frac{1}{2}\right)\right) = 0 \quad IG(Output|y_3) = 1 - \left(E\left(\frac{1}{2}, \frac{1}{2}\right)\right) = 0$$

When $y_1 = 0$ the leaf of the tree is uncertain.

TÉCNICO LISBOA

Aprendizagem 2021/22

Homework I – Group G047

$y_1 = 1$	y_2	y_3	Output
x_1	1	0	N
x_2	1	1	N
x_4	2	0	N
x_6	1	0	P

$$E(Output) = E\left(\frac{3}{4}, \frac{1}{4}\right) = 0.8113$$

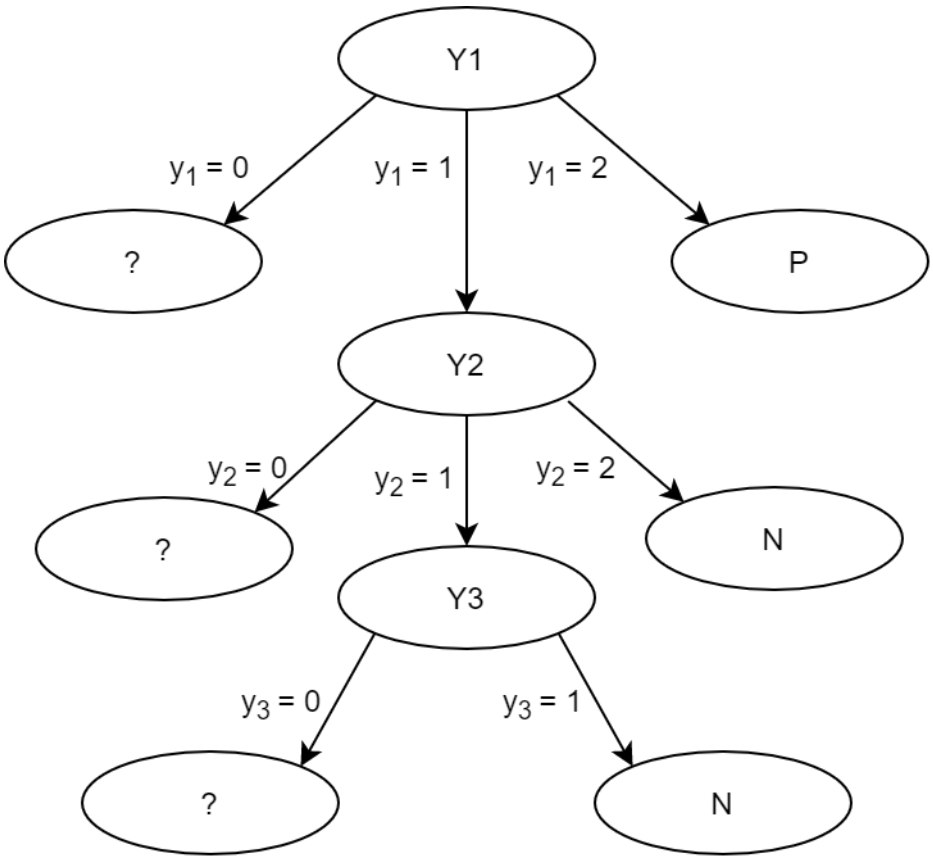
$$IG(Output|y_2) = 0.8113 - \left(\frac{3}{4}E\left(\frac{2}{3}, \frac{1}{3}\right) + \frac{1}{4}E(1)\right) = 0.1226$$

$$IG(Output|y_3) = 0.8113 - \left(\frac{3}{4}E\left(\frac{2}{3}, \frac{1}{3}\right) + \frac{1}{4}E(1)\right) = 0.1226$$

Choose $y_2$as the leaf for $y_1 = 1$ on the decision tree. When $y_2 = 2$ the leaf of the tree is N. There is no information about $y_2 = 0$, therefore that leaf is uncertain.

$y_1 = 1 \cap y_2 = 1$	y_3	Output
x_1	0	N
x_2	1	N
x_6	0	P

When $y_3 = 1$ the leaf of the tree is N. When $y_3 = 0$ 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.

4)

	y_1	y_2	y_3	Output
x_9	2	0	0	N
x_{10}	1	2	0	P

Decision Tree for x_9: $y_1 = 2$, x_9 is classified as P

Decision Tree for x_{10}: $y_1 = 1 \cap y_2 = 2$, x_{10} is classified as N

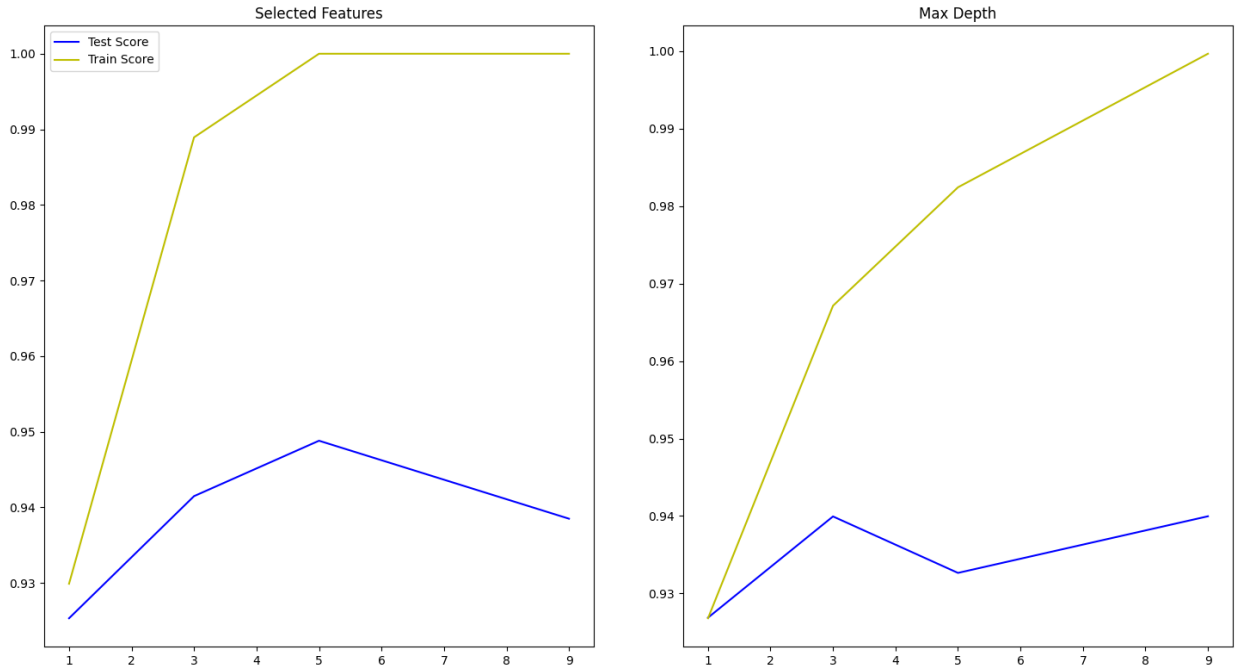
The decision tree has an accuracy of $\frac{0}{2} = 0$

Aprendizagem 2021/22

Homework I – Group G047

II. Programming and critical analysis

5)



6) Two reasons for the correlation between graphs is:

1. 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;
2. 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.

7) 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