

I. Pen-and-paper

Answer 1

HOMEWORK II

Bernardo Castiço ist196845
Hugo Rita ist196870

1-Passo 1 - Design Matrix

$$\phi = \begin{bmatrix} 1 & \phi_1(x_1) & \phi_2(x_2) & \phi_3(x_3) \\ 1 & \phi_1(x_1) & \phi_2(x_2) & \phi_3(x_3) \\ 1 & \phi_1(x_1) & \phi_2(x_2) & \phi_3(x_3) \\ 1 & \phi_1(x_1) & \phi_2(x_2) & \phi_3(x_3) \\ 1 & \phi_1(x_1) & \phi_2(x_2) & \phi_3(x_3) \\ 1 & \phi_1(x_1) & \phi_2(x_2) & \phi_3(x_3) \\ 1 & \phi_1(x_1) & \phi_2(x_2) & \phi_3(x_3) \\ 1 & \phi_1(x_1) & \phi_2(x_2) & \phi_3(x_3) \end{bmatrix} = \begin{bmatrix} 1 & \sqrt{2} & (\sqrt{2})^2 & (\sqrt{2})^3 \\ 1 & \sqrt{27} & (\sqrt{27})^2 & (\sqrt{27})^3 \\ 1 & \sqrt{20} & (\sqrt{20})^2 & (\sqrt{20})^3 \\ 1 & \sqrt{14} & (\sqrt{14})^2 & (\sqrt{14})^3 \\ 1 & \sqrt{53} & (\sqrt{53})^2 & (\sqrt{53})^3 \\ 1 & \sqrt{3} & (\sqrt{3})^2 & (\sqrt{3})^3 \\ 1 & \sqrt{8} & (\sqrt{8})^2 & (\sqrt{8})^3 \\ 1 & \sqrt{85} & (\sqrt{85})^2 & (\sqrt{85})^3 \end{bmatrix} = \begin{bmatrix} 1 & \sqrt{2} & 2 & 2\sqrt{2} \\ 1 & \sqrt{27} & 27 & 27\sqrt{27} \\ 1 & \sqrt{20} & 20 & 20\sqrt{20} \\ 1 & \sqrt{14} & 14 & 14\sqrt{14} \\ 1 & \sqrt{53} & 53 & 53\sqrt{53} \\ 1 & \sqrt{3} & 3 & 3\sqrt{3} \\ 1 & \sqrt{8} & 8 & 2\sqrt{8} \\ 1 & \sqrt{85} & 85 & 85\sqrt{85} \end{bmatrix}$$

Passo 2 - Target

$$T = \begin{bmatrix} 1 \\ 3 \\ 2 \\ 0 \\ 6 \\ 4 \\ 5 \\ 7 \end{bmatrix}$$

Passo 3 - Error

$$E(w) = \sum_{i=1}^n (t^{(i)} - \sigma^{(i)})^2 = (T - \phi w)^T (T - \phi w)$$

$$\frac{\delta E(w)}{\delta w} = 0 \Leftrightarrow \frac{\delta ((T - \phi w)^T (T - \phi w))}{\delta w} = 0 \Leftrightarrow$$

$$\Leftrightarrow (-\phi)^T (T - \phi w) + (T - \phi w)^T (-\phi) = 0 \Leftrightarrow$$

$$\Leftrightarrow w = (\phi^T \phi)^{-1} \phi^T T$$

$\phi^T = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \sqrt{2} & \sqrt{27} & \sqrt{20} & \sqrt{14} & \sqrt{53} & \sqrt{3} & \sqrt{8} & \sqrt{85} \\ 2 & 27 & 20 & 14 & 53 & 3 & 8 & 85 \\ \sqrt{2}^3 & \sqrt{27}^3 & \sqrt{20}^3 & \sqrt{14}^3 & \sqrt{53}^3 & \sqrt{3}^3 & \sqrt{8}^3 & \sqrt{85}^3 \end{bmatrix}$
 $\phi^T \phi = \begin{bmatrix} 8 & 35,8843 & 212 & 1482,2811 \\ 35,8843 & 212 & 1482,2811 & 11436 \\ 212 & 1482,2811 & 11436 & 93573,5164 \\ 1482,2811 & 11436 & 93573,5164 & 793976 \end{bmatrix}$

$(\phi^T \phi)^{-1} = \begin{bmatrix} 8,1955 & -6,2313 & 1,3049 & -0,0793 \\ -6,2313 & 5,0781 & -1,1044 & 0,0686 \\ 1,3049 & -1,1043 & 0,2472 & -0,0157 \\ -0,0793 & 0,0687 & -0,0157 & 0,0010 \end{bmatrix}$

$(\phi^T \phi)^{-1} \phi^T = \begin{bmatrix} 1,7626 & -0,0312 & -0,6695 & -1,0069 & 1,3794 & 0,1905 & -0,7751 & -0,5107 \\ -1,0643 & -0,6319 & 0,5312 & 0,9640 & -1,3070 & -0,1922 & 0,8501 & 0,5101 \\ 0,1933 & 0,0436 & -0,0907 & -0,1868 & 0,3232 & 0,0524 & -0,1954 & -0,1145 \\ -0,0107 & -0,0044 & 0,0044 & 0,01094 & -0,0116 & -0,0022 & 0,0123 & 0,0110 \end{bmatrix}$

sendo assim $w = \begin{bmatrix} 4,584 \\ -1,687 \\ 0,338 \\ -0,013 \end{bmatrix} = (\phi^T \phi)^{-1} \phi^T T$

Answer 2

2- $\theta^{(q)} = w_0 + w_1 \phi_1(x_q) + w_2 \phi_2(x_q) + w_3 \phi_3(x_q)$

$\|x_9\| = 2 \quad \theta^{(9)} = 4,584 - 1,687 \times 2 + 0,338 \times 2^2 - 0,013 \times 2^3 = 2,458$

$\theta^{(10)} = w_0 + w_1 \phi_1(x_{10}) + w_2 \phi_2(x_{10}) + w_3 \phi_3(x_{10})$

$\|x_{10}\| = \sqrt{6} \quad \theta^{(10)} = 4,584 - 1,687 \times \sqrt{6} + 0,338 \times \sqrt{6}^2 - 0,013 \times \sqrt{6}^3 = 2,289$

$RMS E = \sqrt{\frac{1}{m} \sum_{i=1}^m (t^{(i)} - \theta^{(i)})^2} = \sqrt{\frac{1}{2} [(2 - \theta^{(9)})^2 + (4 - \theta^{(10)})^2]} =$

$= \sqrt{\frac{1}{2} [(2 - 2,458)^2 + (4 - 2,289)^2]} = 1,2525$

Answer 3

3- Após a binarização

	y_3	t_i
x_1	0	N
x_2	1	N
x_3	1	N
x_4	1	N
x_5	1	P
x_6	0	P
x_7	0	P
x_8	1	P
x_9	0	N
x_{10}	0	P

$\rightarrow E_{start}(y_{out}) = E\left(\frac{1}{2}, \frac{1}{2}\right) = -\left(\frac{1}{2} \log_2\left(\frac{1}{2}\right) + \frac{1}{2} \log_2\left(\frac{1}{2}\right)\right) = 1$

$\rightarrow E(y_{out}/y_3) = \frac{3}{8} E(y_{out}/y_3=0) + \frac{5}{8} E(y_{out}/y_3=1) =$

$= \frac{3}{8} E\left(\frac{1}{3}, \frac{2}{3}\right) + \frac{5}{8} E\left(\frac{3}{5}, \frac{2}{5}\right) = \frac{3}{8} \left[-\left(\frac{1}{3} \log_2\left(\frac{1}{3}\right) + \frac{2}{3} \log_2\left(\frac{2}{3}\right)\right)\right] +$

$+ \frac{5}{8} \left[-\left(\frac{3}{5} \log_2\left(\frac{3}{5}\right) + \frac{2}{5} \log_2\left(\frac{2}{5}\right)\right)\right] = 0,9512$

$\rightarrow E(y_{out}/y_2) = \frac{2}{8} E(1) + \frac{3}{8} E\left(\frac{2}{3}, \frac{1}{3}\right) + \frac{3}{8} E\left(\frac{2}{3}, \frac{1}{3}\right) =$

$= \frac{2}{8} [- (1 \log_2(1))] + \frac{3}{8} \left[-\left(\frac{2}{3} \log_2\left(\frac{2}{3}\right) + \frac{1}{3} \log_2\left(\frac{1}{3}\right)\right)\right] +$

$+ \frac{3}{8} \left[-\left(\frac{2}{3} \log_2\left(\frac{2}{3}\right) + \frac{1}{3} \log_2\left(\frac{1}{3}\right)\right)\right] = 0,6887$

$\rightarrow E(y_{out}/y_1) = \frac{2}{8} E\left(\frac{1}{2}, \frac{1}{2}\right) + \frac{4}{8} E\left(\frac{3}{4}, \frac{1}{4}\right) + \frac{2}{8} E(1) = \frac{2}{8} \left[-\left(\frac{1}{2} \log_2\left(\frac{1}{2}\right) + \frac{1}{2} \log_2\left(\frac{1}{2}\right)\right)\right] +$

$+ \frac{4}{8} \left[-\left(\frac{3}{4} \log_2\left(\frac{3}{4}\right) + \frac{1}{4} \log_2\left(\frac{1}{4}\right)\right)\right] + \frac{2}{8} [- (1 \log_2(1))] = 0,6556$

	Information Gain
y_3	$E(y_{out}) - E(y_{out}/y_3) =$ $= 1 - 0,9512 = 0,0488$
y_2	$E(y_{out}) - E(y_{out}/y_2) =$ $= 1 - 0,6887 = 0,3113$
y_1	$E(y_{out}) - E(y_{out}/y_1) =$ $= 1 - 0,6556 = 0,3444$

O y_1 é o que possui mais information gain.

Answer 3 - continuação

$E(y_{out}|y_1=0) = -\left(\frac{1}{2}\log_2\left(\frac{1}{2}\right) + \frac{1}{2}\log_2\left(\frac{1}{2}\right)\right) = 1$
 $\rightarrow y_2$ para $y_1=0$, y_2 é sempre 2
 $E_2 = E\left(\frac{1}{2}, \frac{1}{2}\right) = -\left(\frac{1}{2}\log_2\left(\frac{1}{2}\right) + \frac{1}{2}\log_2\left(\frac{1}{2}\right)\right) = 1$
 $\rightarrow y_3$ para $y_1=0$, y_3 é sempre 1 dada a sua bimanha
 $E_3 = E\left(\frac{1}{2}, \frac{1}{2}\right) = -\left(\frac{1}{2}\log_2\left(\frac{1}{2}\right) + \frac{1}{2}\log_2\left(\frac{1}{2}\right)\right) = 1$

Dada a information gain
 são ambos 0 temos de
 tomar uma decisão
 e optamos que $y_1=0$
 seja N.

$E(y_{out}|y_1=1) = -\left(\frac{3}{4}\log_2\left(\frac{3}{4}\right) + \frac{1}{4}\log_2\left(\frac{1}{4}\right)\right) = 0,8113$
 $\rightarrow y_2: E_2 = \frac{3}{4}E\left(\frac{2}{3}, \frac{1}{3}\right) + \frac{1}{4}E(1) =$
 $= \frac{3}{4}\left[-\left(\frac{2}{3}\log_2\left(\frac{2}{3}\right) + \frac{1}{3}\log_2\left(\frac{1}{3}\right)\right)\right] = 0,6887$
 $\rightarrow y_3: E_3 = \frac{1}{2}E\left(\frac{1}{2}, \frac{1}{2}\right) + \frac{1}{2}E(1) =$
 $= \frac{1}{2}\left[-\left(\frac{1}{2}\log_2\left(\frac{1}{2}\right) + \frac{1}{2}\log_2\left(\frac{1}{2}\right)\right)\right] = 0,5$

Information gain
 $y_2 = 1 - 0,6887 = 0,3113$
 $y_3 = 1 - 0,5 = 0,5 \checkmark$ e y_3 é o que tem maior information gain.

Quando $y_1=1$ e $y_3=0$, y_2 é 1 em ambos os casos
 sendo num caso o output P e no outro N.
 Faz a isto temos de tomar uma decisão
 e optamos por escolher P para $y_3=0$.

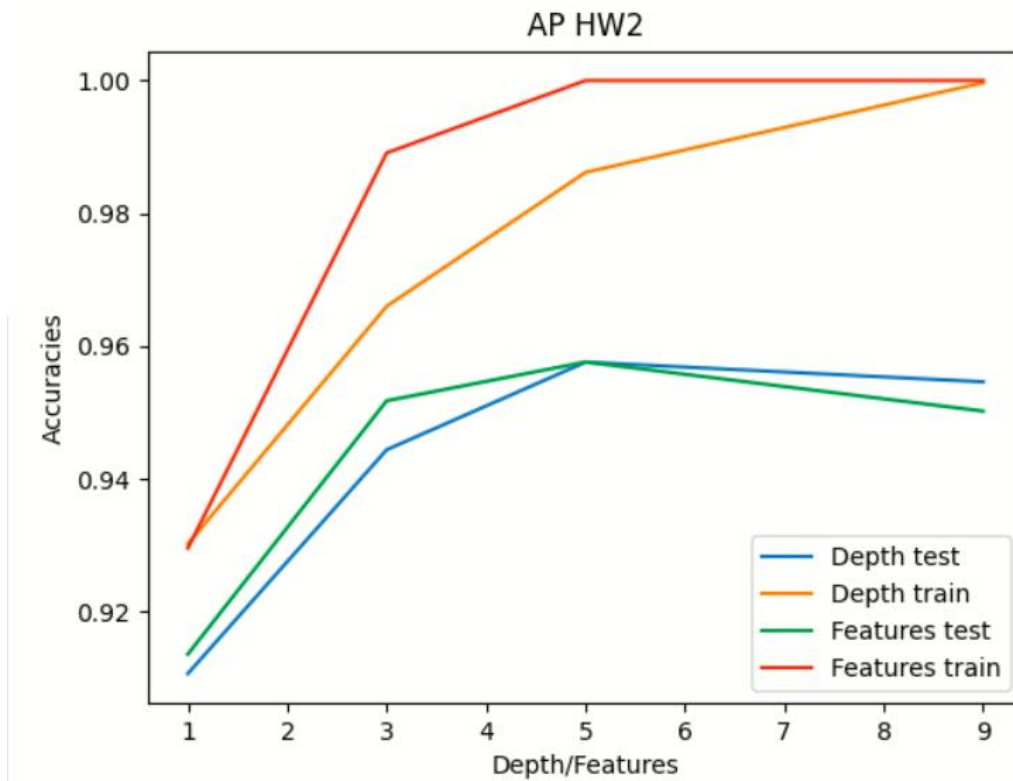
Portanto a decision tree final é esta

Answer 4

4- Para x_9 , segundo a decision tree output = P. O output real é N.
 Para x_{10} , segundo a decision tree output = . O output real é P.
 Accuracy = $\frac{\text{output certos}}{\text{casos totais}} = \frac{1}{2} = 0,5$

II. Programming and critical analysis

Answer 5



Para a obtenção deste gráfico começámos por separar a data do ficheiro fornecido usando um 10-fold cross validation. Posto isto, seleccionámos as melhores features usando a função do python SelectKBest sendo o resultado desta função usado para a alínea i. Assim, para calcular as decision trees usamos a função DecisionTreeClassifier com os parâmetros default, excepto o max depth na alínea ii.

Por fim, fizemos fit da data e predict usando as decision trees para poder comparar resultados.

Answer 6

Optámos por calcular a correlação de Pearsson com o objetivo de obter um valor para melhor esclarecer a existência de tal correlação, sendo o valor obtido = 0.98768982.

Uma das razões para a correlação observada deve-se ao facto de apesar de no caso i. se seleccionar as max features e no caso ii. se seleccionar a max depth. Pode-se verificar que ao seleccionar max features = $i \in [1, 3, 5, 9]$ estamos a seleccionar também uma max depth correspondente ao valor de i que seleccionamos.

A outra razão passa pelo facto de ao limitarmos tanto a max depth como a max features com valores baixos, teremos pouca informação para testar a data, o que se reflete em accuracies mais baixas. O mesmo efeito acontece para valores altos como $k = 9$, sendo que nestes casos o facto de existir uma data bastante vasta também leva a uma ligeira diminuição na accuracy.

Answer 7

A depth que seleccionamos é $k = 5$, uma vez que para uma tree com max depth igual a 5 ao testarmos a nossa test data é aí que se atinge um valor máximo, ocorrendo para valores superiores a 5 overfit.

III. APPENDIX

Nota: Falamos com o professor das teóricas do tagus ([Andreas Miroslaus Wichert](#)) e ele disse-nos que não havia qualquer problema em exceder o numero de páginas relativamente às referenciadas no enunciado.

Paste your programming code here using Consolas 9pt or 10pt.

Use **highlighting** or **colored** text to facilitate the analysis by your faculty hosts.

```
# Grupo 117 Aprendizagem HomeWork 2
# Bernardo Castico ist196845
# Hugo Rita ist196870
from sklearn import tree
from sklearn.model_selection import KFold
from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import mutual_info_classif
import matplotlib.pyplot as plt

#Res = the 10-fold cross validation with our group number (117)
Res = KFold(n_splits=10, random_state=117, shuffle=True)

def getDataToMatrix(lines):
    reallines = []
    data = []
    toDelete = []
    for i in range(len(lines)):
        if i > 11:
            reallines += [lines[i]]
    for i in range(len(reallines)):
        for j in range(len(reallines[i])):
            if reallines[i][j] == "benign\n":
                reallines[i][j] = 1
            elif reallines[i][j] == "malignant\n":
                reallines[i][j] = 0
            elif reallines[i][j] == '?':
                toDelete += [i]
            else:
                reallines[i][j] = int(reallines[i][j])
    for i in range(len(reallines)):
        if i not in toDelete:
            data += [reallines[i]]
    return data

def splitData(list):
    a = []
    b = []
    for i in list:
        a.append(i[:-1])
        b.append(i[-1])
    return [a,b]
```

```
def main():
    depthTestX, finalAccuraciesAllFeatures, finalAccuraciesDepth, res , AllFeaturesTrainY,
    AllFeaturesTrainX = [],[],[],[],[],[]
    depthTestY , AllFeaturesTestY, AllFeaturesTestX, depthTrainY, depthTrainX = [],[],[],[],[]

    with open("HW2.txt") as f:
        lines = f.readlines()
    for line in lines:
        tmp = line.split(',')
        res.append(tmp)
    data = getDataToMatrix(res)

    for i in [1,3,5,9]:
        counter11, counter12, counter21, counter22 = 0,0,0,0
        accuraciesDepth, accuraciesAllFeatures = [],[]

        for train, test in Res.split(data):
            testData, trainData = [],[]
            accuracyAuxDepthTest, accuracyAuxDepthTrain = 0,0
            accuracyAuxAllFeaturesTest, accuracyAuxAllFeaturesTrain = 0,0

            for j in test:
                testData += [data[j]]
            for j in train:
                trainData += [data[j]]
            trainDataSplit = splitData(trainData)
            testDataSplit = splitData(testData)

            decision = SelectKBest(mutual_info_classif, k=i).fit(trainDataSplit[0],
trainDataSplit[1])
            decisionTrainData = decision.transform(trainDataSplit[0])
            decisionTestData = decision.transform(testDataSplit[0])

            resultDepth = tree.DecisionTreeClassifier(max_depth=i, criterion="gini",
max_features=None)
            resultAllFeatures = tree.DecisionTreeClassifier(max_depth=None, criterion="gini",
max_features=None)

            resultDepth.fit(trainDataSplit[0], trainDataSplit[1])
            resultAllFeatures.fit(decisionTrainData, trainDataSplit[1])

            predictionsTest = resultDepth.predict(testDataSplit[0])
            predictionsTrain = resultDepth.predict(trainDataSplit[0])

            predictionsTestFeatures = resultAllFeatures.predict(decisionTestData)
            predictionsTrainFeatures = resultAllFeatures.predict(decisionTrainData)
```

Aprendizagem 2021/22
Homework II – Group 117

```
for j in range(len(predictionsTestFeatures)):
    if predictionsTestFeatures[j] == testDataSplit[1][j]:
        accuracyAuxAllFeaturesTest += 1
    if predictionsTest[j] == testDataSplit[1][j]:
        accuracyAuxDepthTest += 1
for j in range(len(predictionsTrainFeatures)):
    if predictionsTrainFeatures[j] == trainDataSplit[1][j]:
        accuracyAuxAllFeaturesTrain += 1
    if predictionsTrain[j] == trainDataSplit[1][j]:
        accuracyAuxDepthTrain += 1
accuraciesAllFeatures += [[accuracyAuxAllFeaturesTest/len(predictionsTestFeatures),
accuracyAuxAllFeaturesTrain/len(predictionsTrainFeatures)]]
accuraciesDepth += [[accuracyAuxDepthTest / len(predictionsTest), accuracyAuxDepthTrain
/ len(predictionsTrain)]]

for k in range(len(accuraciesDepth)):
    counter11 += accuraciesDepth[k][0]
    counter12 += accuraciesDepth[k][1]
    counter21 += accuraciesAllFeatures[k][0]
    counter22 += accuraciesAllFeatures[k][1]

finalAccuraciesDepth += [[counter11 / 10, counter12 / 10]]
finalAccuraciesAllFeatures += [[counter21 / 10, counter22 / 10]]

#Plot
for i in range(4):
    depthTestX = [1,3,5,9]
    depthTestY += [finalAccuraciesDepth[i][0]]
    depthTrainX = [1, 3, 5, 9]
    depthTrainY += [finalAccuraciesDepth[i][1]]
    AllFeaturesTestX = [1,3,5,9]
    AllFeaturesTestY += [finalAccuraciesAllFeatures[i][0]]
    AllFeaturesTrainX = [1, 3, 5, 9]
    AllFeaturesTrainY += [finalAccuraciesAllFeatures[i][1]]

plt.xlabel('Depth/AllFeatures')
plt.ylabel('Accuracies')
plt.title('AP HW2')
plt.plot(depthTestX, depthTestY, label = "depth test")
plt.plot(depthTrainX, depthTrainY, label = "depth train")
plt.plot(AllFeaturesTestX, AllFeaturesTestY, label = "All Features test")
plt.plot(AllFeaturesTrainX, AllFeaturesTrainY, label = "All Features train")
plt.legend()
plt.show()
main()
```

Aprendizagem 2021/22
Homework II – Group 117

END