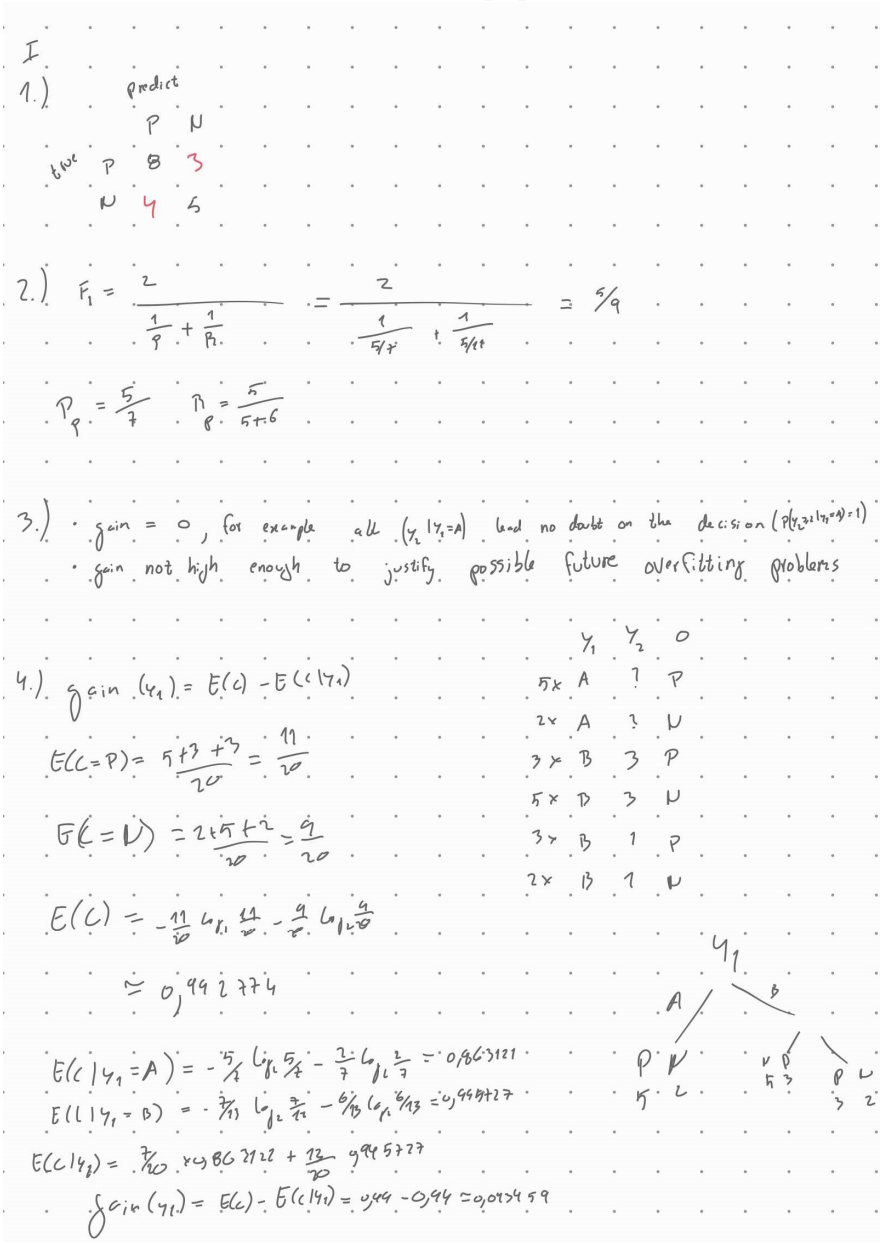
Aprendizagem 2021/22

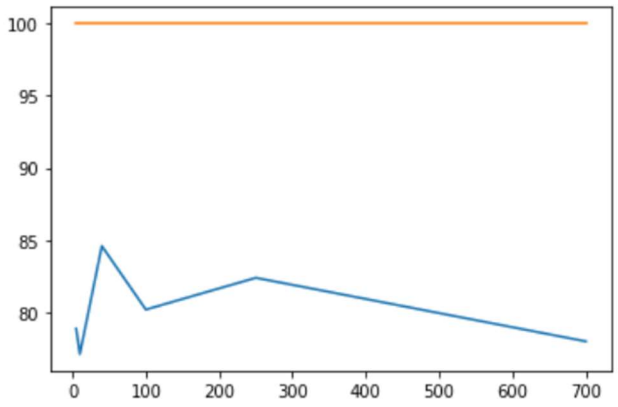
Homework I – Group 102

I. Pen-and-paper



II. Programming and critical analysis

1)



2) While when testing, the algorithm is guessing the answer, when evaluating on already trained instances it is certain about its class. So, unless it undervalues any train line, it will guess its class correctly. Which could also show some signs to overfitting.

III. APPENDIX

```
import pandas as pd, matplotlib.pyplot as plt
from scipy.io.arff import loadarff
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import train_test_split
from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import mutual_info_classif

data = loadarff('drive/MyDrive/ML/pd_speech.arff')
df = pd.DataFrame(data[0])
df['class'] = df['class'].str.decode('utf-8')
X = df.drop('class', axis=1)
y = df['class']

Kvalues = [5,10,40,100,250,700]
test_scores = []
train_scores = []

for i in Kvalues:
  X_new = SelectKBest(mutual_info_classif, k=i).fit_transform(X, y)
  X_train, X_test, y_train, y_test = train_test_split(X_new, y, train_size = 0.7, random_state = 1)

  predictor = DecisionTreeClassifier(random_state=1)
  predictor.fit(X_train, y_train)

  train_scores.append(predictor.score(X_train, y_train)*100)
  test_scores.append(predictor.score(X_test, y_test)*100)

plt.plot(Kvalues, test_scores)
plt.plot(Kvalues, train_scores)
plt.show()
```

END