Instituto Politécnico de Viseu
Escola Superior de Tecnologia e Gestão de Viseu
Departamento de Informática

Docente: Filipe Cabral Pinto

Final assignment

Naive Bayes, Decision Tree, Random Forest and kNN algorithm

Análise e Exploração de Dados

Mestrado em Sistemas e Tecnologias de Informação para as Organizações

16095 – Manuel Augusto Tarouca Martins

19164– Elodie Morin

Viseu, janeiro 2020

Instituto Politécnico de Viseu
Escola Superior de Tecnologia e Gestão de Viseu
Departamento de Informática

Final assignment
Naive Bayes, Decision Tree, Random Forest and kNN algorithm

Mestrado em Sistemas e Tecnologias de Informação para as Organizações

Análise e Exploração de Dados

Ano Letivo 2019/2020

Viseu, janeiro 2020

1.1. Índice

1.2. Índice de figuras

2. Introduction

In this assignment the students are going to explore the naive bayes, decision tree, random forest and kNN algorithm taught in class. The students are going to apply these algorithms to the datasets provided by the teacher and based on the output of these algorithms they are going to analyse the metrics, infer results and then export them into a spreadsheet in Excel.

3. Development

3.1. Context

The programming language used to code this project was python and the integrated development environment used was PyCharm with the Jupiter Notebook plugin.

In order to open the files provided by the students, it is necessary to have an IDE compatible with Jupiter Notebooks.

3.2. Default behaviour

```
#%%

import pandas as pd
db = pd.read_csv('adult.csv', names=["age","workclass","fnlwgt","education",...
descriptive = db.iloc[:,0:14].values
target = db.iloc[:,14].values

```

Ilustração 1 – Dataset importation

First the datasets are imported into the program and split between the descriptive and target values.

```
#%%

#Label encoder
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
descriptive[:,0] = le.fit_transform(descriptive[:,0])
descriptive[:,1] = le.fit_transform(descriptive[:,1])
descriptive[:,2] = le.fit_transform(descriptive[:,2])
descriptive[:,3] = le.fit_transform(descriptive[:,3])
descriptive[:,4] = le.fit_transform(descriptive[:,4])
descriptive[:,5] = le.fit_transform(descriptive[:,5])
descriptive[:,6] = le.fit_transform(descriptive[:,6])
descriptive[:,7] = le.fit_transform(descriptive[:,7])
descriptive[:,8] = le.fit_transform(descriptive[:,8])
descriptive[:,9] = le.fit_transform(descriptive[:,9])
descriptive[:,10] = le.fit_transform(descriptive[:,10])
descriptive[:,11] = le.fit_transform(descriptive[:,11])
descriptive[:,12] = le.fit_transform(descriptive[:,12])
descriptive[:,13] = le.fit_transform(descriptive[:,13])

```

Ilustração 2 - Encoding

The values are then pre-processed by a label encoder. The label encoder will transform every categorical value into a discrete value.

```
#%%

#Split Dataset
percentagem = 0.5
from sklearn.model_selection import train_test_split
#change this line for 0.15, 0.30, 0.50
descriptiveTraining, descriptiveTest, targetTraining, targetTest = train_test_split(descriptive,target,test_size = percentagem, random_state = 2)

#print(descriptiveTraining)
#print(descriptiveTest)

```

Ilustração 3 – Training and Test Dataset

Then the dataset is split into the training and the test parts according to a given percentage which will take the value of 15%, 30% or 50% here. The random state is set to 2 to allow the test set to have all the possible output, especially with the lymphography dataset when the test set rate is set to 15%.

```
#%%

#Naive Bayes Algorithm & Test
from sklearn.naive_bayes import GaussianNB
classifier = GaussianNB()
classifier.fit(descriptiveTraining, targetTraining)
prediction = classifier.predict(descriptiveTest)

```

Ilustração 4 - Classifier

The training dataset go through the classifier in order to correctly fit the algorithm to the dataset, and then we do a prediction of the test data.

```
#%%

#Metrics
from sklearn.metrics import confusion_matrix, accuracy_score
accuracy = accuracy_score(targetTest,prediction)
matrix = confusion_matrix(targetTest,prediction)
print("Accuracy:")
print(accuracy )
print("Confusion matrix:")
print(matrix)
```

Ilustração 5 - Metrics

From this prediction we can get metrics to see the accuracy of each algorithm, and a confusion matrix.

4. Results

4.1. Naive Bayes

The worst results are with the use of the three functions. The encoder does make the results a little worst. The presence or not of the scaler doesn't seems to affect the results.
The fact of transforming the column of categorical values to multiple columns with Boolean values doesn't help to get better results, it seems to be the opposite.
Use a one-hot encoder just complexify the algorithm to predict the data. As the Naive Bayes is based on probabilities, scaling the data won't affect the results.

a) Lymphography

The accuracy is better with half of the dataset used for test.

b) Adult

The test size doesn't affect the accuracy.

4.2. Decision tree

The results are pretty much the same, wherever we use the encoder and/or the scaler or not

a) Lymphography

The accuracy is better with a small test size.
With or without the scaler we have the same results for a test size of 30% and 50%. The 15% is better without it.

b) Adult

The accuracy it quite the same regarding the test size. The worst is 30%.

4.3. Random forest

The results are quite the same regardless of the use of an encoder and/or a scaler. Which is logic

a) Lymphography

We also have better result with a small test size.
The results are the same for all the algorithms, except the one with only the scaler but the results are worse than with the other algorithms.

b) Adult

The results are globally the same for each algorithms and percentages. The best results are with the smaller test set, as the algorithm is more fitted.

4.4. kNN

The results are better using the scaler, and even better without the encoder.
As the scaler will normalize the values, the variables that have bigger range for their values won't be minimize in the calculation of the nearest neighbour.
The results are a little better without the use of the one-hot encoder.

c) Lymphography

The results are better using a scaler and a small test size.

d) Adult

Same, better results are using scaler.