LAB5 REPORT

AMAN BHANSALI(B20ME010)

Task1:

1. It is clear from the data that our drugs column is what we are trying to predict and hence it is the label and rest all are our features.
2. Searching for missing values is a part of data preprocessing to check if our data contains all values or not so it can directly be checked by the function 'data.isnull()'.
3. Categorical data as the name suggests are data which can be categorized so for our data set they are Sex-Nominal, BP-Ordinal, Cholesterol-Ordinal, Drug-Nominal, Categorical- Sex, BP, Cholesterol, Drug.
4. For splitting the data into train and test I used the function train_test_split which takes parameters like data name, test size (which we choose) and random_state(ensures that the generated are reproducible, the value we provide is called a seed)
5. To set the seed value equal to 55 which I did using 'random_state' function.
6. We needed to use entropy information gain and Gini index. For this we have inbuilt functions in the classifier called as DecisionTreeClassifier() which has one of its parameters as criterion(which takes 'entropy' or 'gini')
7. For training the model using decision tree classifier the above written function is used also a new parameter called as min_depth is used. It is the value/no. after which we want our stop diverging into further branches. High value of this parameter may cause our model to overfit the data and give bad results on test data, while low value may cause underfitting. An optimal value needs to be used like I used 3 in my classifier. After this the functions such as '.fit(), .predict()' were used to implement the classifier completely.

   For accuracy I used the library function of sklearn directly.
   Talking about the accuracy of the model I found that all the three models corresponding to three different values of splitting gave a good accuracy on the test data set.
   For the three cases the model and test accuracies are:

Test Accuracy for 1st case-0.8833333333333333

Test Accuracy for 2nd case- 0.85

Test Accuracy for 3rd case- 0.9

8. Preparing a confusion matrix for this again I used python metrics library function which takes input two values y_test, y_pred and gives the confusion matrix directly and the classification values such as precision, F1-score, etc.
9. For graphical visualisation of the tree it can be done using the library of graphviz and tree. It takes parameter as our asisigned classifier and returns the tree structure as per that classifier. Similarly can be done for the three splitting values.
10. For overfitting part it can be observed from the values we get for model and test accuracy. A new parameter called as min_depth is used in classifier. It is the value/no. after which we want our stop diverging into further branches. High value of this parameter may cause our model to overfit the data and give bad results on test data, while low value may cause underfitting. An optimal value needs to be used like I used 3 in my classifier. And checking on the train data also our model is having good accuracy i.e not equal to 1(which may be a case of overfitting). It can clearly we observed that all the models have a small gap between the model's accuracy on train data and on test data which means the model is not overfitting the data in a huge amount

Model Accuracy 0.9214285714285714
Test Accuracy 0.8833333333333333

Model Accuracy 0.925
Test Accuracy 0.85

Model Accuracy 0.9111111111111111
Test Accuracy 0.9

Task 2:

The value of Concrete compressive strength (MPa, megapascals) is the label or the target value and the rest are the parameters/features whose values determine the value of the label.

My roll no. is B20ME010 the last three digits are even so the splitting would be in 80:20. For splitting the dataset into train and test we use
For splitting the data into train and test I used the function train_test_split which takes parameters like data name, test size (which we choose here for my case to be 0.2) and

random_state(ensures that the generated are reproducible, the value we provide is called a seed)

For reproducibility we will set the seed value (random_state value)equal to 2021 here.

For my case since the last three digits are even so I will be using nod selection strategy as ‘best’ in my decision tree regressor which is done by a parameter called splitter.
Now we need to implement the Decision Tree Regressor on our model. For this we use the function of scikitlearn library called the ‘DecisionTreeRegressor’. It takes certain parameters like criteretion, random state, splitter, max. depth.
Criterion- ‘mse’, ‘mae’ mse means mean squared error which minimizes the L2 loss using the mean of each terminal node while mae means mean absolute error which minimizes the L1 loss using the median of each terminal node.
Max. depth- The height of the tree like what no. of steps we want our branches to diverge further.
Splitter- It has two values ‘best’ and ‘random’. For our case I used best as per my roll no. (even). Best means to choose the best split at each node.

Now we cannot calculate the model accuracy and the confusion matrix for this dataset because for a set of features we are getting here an output which is a random number, it could either be any integer or decimal and not a kind of output that classifies something. Accuracy score, confusion matrix and classification report as the name suggests are for classification problems only.

Graphical visualisation can be performed using python’s inbuilt library graphviz and tree which creates graphical visualisation showing the splitting. It takes parameter as classifier which we assign while implementing regressor.

https://colab.research.google.com/drive/1Bu5XKQTtlxj01cp2M-frr-icJjMy7Bix?usp=sharing

https://colab.research.google.com/drive/1Gmr3HV-_31TMX_NlPYZh4XeXqvT91u8E?usp=sharing