ESILV - Python for data analysis

SALHI Nassim Djamil - DIA5 - A4

Project description

This project will cover how to analyse the QSAR biodegradation Dataset which is open source and how to create and train a machine learning model to predict if an entry in the Dataset ready biodegradable (RB) and not ready biodegradable (NRB).

Dataset Description

- The Dataset that we are working on represent

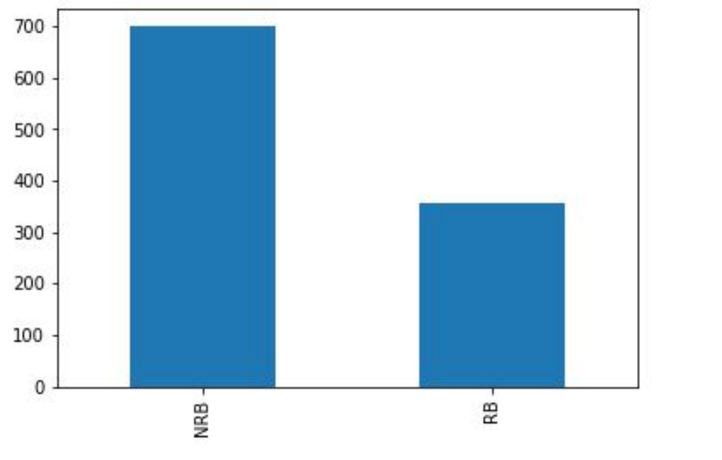
- The Dataset contains 42 columns and 1055 rows.

- Each entry is classified to one of these to classes (RB orNRB).

Dataset Exploration

- We can see that the dataset contains numerical values and categorical values that are represented by numbers from 0 to the number of categories.

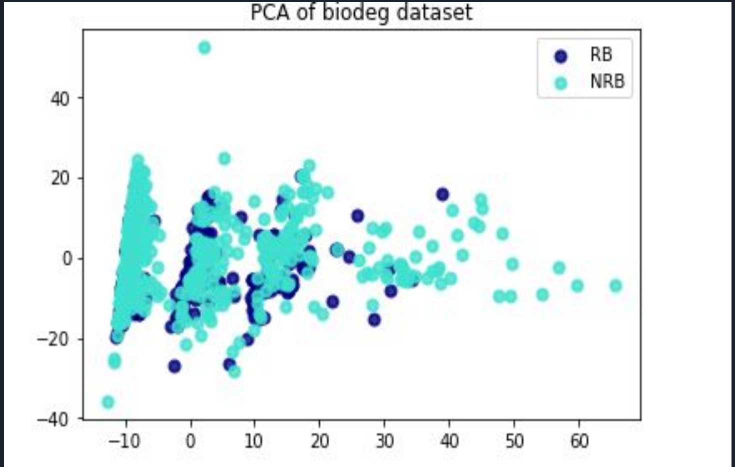
- We can see that the classes are not equally distributed since we have 699 entries that are NRB and 356 entries are RB .



Dataset Visualisation

- For the data visualisation we choose to use a correlation matrix, a scatter matrix and a PCA (Principal Component Analysis).

- Using The PCA we move from a 41D Dataset to a 2D one which can help see if the dataset is linearly separable and as we can see from this graphic is that the problem is most likely not linearly separable.



Machine learning model training

- We choose to train 5 different models and compare them to find the most accurate one for our dataset

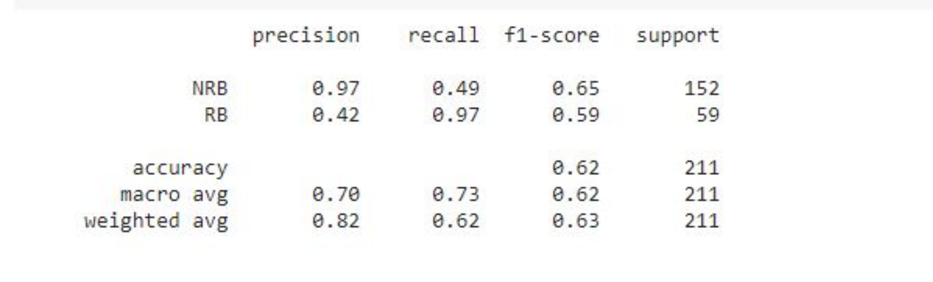
- The 5 models are : Logistic Regression - Decision Tree - KNN - Linear Discriminant Analysis - GaussianNB and SVC.

Machine learning result

- The best two models are the decision tree and logistic regression with 83% accuracy.

- for more insight about the accuracy values we choose to use a classification report that did show the accuracy by classes and we can see that we have a better accuracy for the NRB classes then the RB class which can be caused the fact that the classes are not equally distributed.

  GaussianNB accuracy :

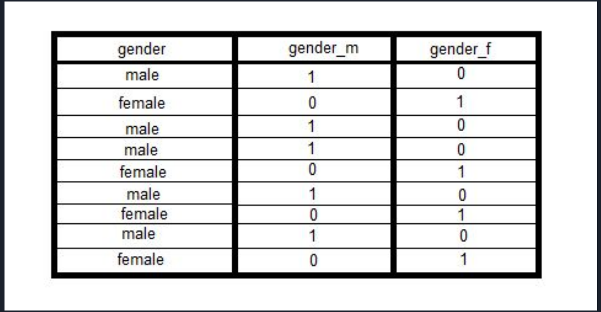


```
              precision    recall  f1-score   support

         NRB       0.97      0.49      0.65       152
          RB       0.42      0.97      0.59        59

    accuracy                           0.62       211
   macro avg       0.70      0.73      0.62       211
weighted avg       0.82      0.62      0.63       211
```

Some propositions

- Since the classes are not equally distributed we can try to train a model using a sub-dataset using the same number of entries for both classes.

- Since we have some categorical values a solution to increase the accuracy we can create dummy variables for each column ex:



gender	gender_m	gender_f
male	1	0
female	0	1
male	1	0
male	1	0
female	0	1
male	1	0
female	0	1
male	1	0
female	0	1

Thanks you !