HOMEWORK 3
CSSM502- ADVANCED DATA ANALYSIS IN PYTHON

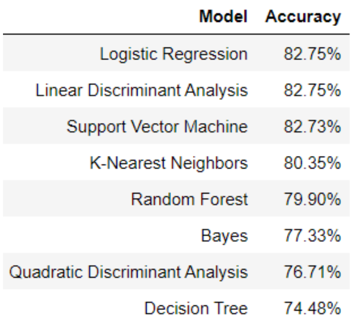
İZEL YAZICI - 0077549

1. Introduction

The purpose of this homework is to review and practice fundamental machine learning concepts. The idea is to build a predictive model of whether a respondent likely voted in their last presidential election. For this purpose, I used “cses4_cut.csv” file which containing a subset of the CSES Wave Four data set.

2. Trying multiple approaches:

I tested different classifiers and regressors to see their behavior without any pre-processing or dimensionality-reduction operation.
Results are as follows:



Model	Accuracy
Logistic Regression	82.75%
Linear Discriminant Analysis	82.75%
Support Vector Machine	82.73%
K-Nearest Neighbors	80.35%
Random Forest	79.90%
Bayes	77.33%
Quadratic Discriminant Analysis	76.71%
Decision Tree	74.48%

3. Dimensionality Reduction with Feature Selection

Feature selection is a technique where we choose those features in our data that contribute most to the target variable. In other words, we choose the best predictors for the target variable. With feature selection we can reduce overfitting, improve accuracy, and reduce training time. For this purpose I used **sklearn.feature_selection.SelectKBest** function and took 10 features with the highest score which are:

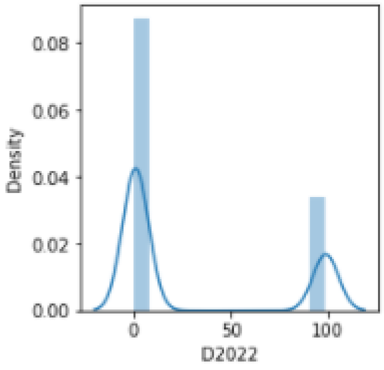
'D2011', 'D2021', 'D2022', 'D2023', 'D2026', 'D2027', 'D2028', 'D2029', 'D2030'

4. Pre-processing:

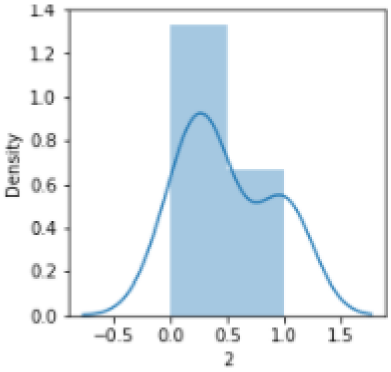
I used quantile transformer method **(sklearn.preprocessing.QuantileTransformer**) to solve unwanted (like unneccary, missing and outliers) data problem. This method transforms the features to follow a uniform or a normal distribution. Therefore, for a given feature, this transformation tends to spread out the most frequent values. It also reduces the impact of outliers.

For instance: You can see one of the feature’s distibution graph befora and after transformation

Before Transformation

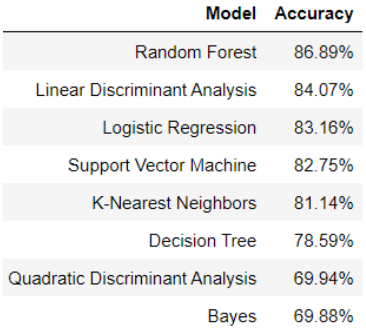


After Transformation



5. Classifiers with Dimensionality-Reduction And Pre-Processing:

After pre-processing and feature selection, I re-trained the models. Results are as follows:



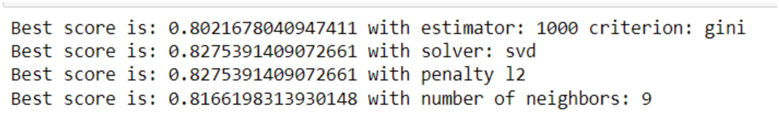
Model	Accuracy
Random Forest	86.89%
Linear Discriminant Analysis	84.07%
Logistic Regression	83.16%
Support Vector Machine	82.75%
K-Nearest Neighbors	81.14%
Decision Tree	78.59%
Quadratic Discriminant Analysis	69.94%
Bayes	69.88%

6. Hyperparameter Tuning :

After I have tried GridSearchCV, I realized that doesnt gives us better accuracy if I compare with some hyper parameter tuning codes that I wrote from scracth as a bunch of parameter loop, thats why I used them for this part.

I took the top 5 classifiers and regressors and looped them until I found the best hyperparameters

Best results were achieved with these parameters:



```
Best score is: 0.8021678040947411 with estimator: 1000 criterion: gini
Best score is: 0.8275391409072661 with solver: svd
Best score is: 0.8275391409072661 with penalty l2
Best score is: 0.8166198313930148 with number of neighbors: 9
```