Int 247

Roll no 4

Registration number 11701145

Pulkit matta

Project Report

Question: Predict the grade of student using different classification techniques Bachelor of Science (Forensic Science)...

Dataset: Basic Insight

Dataset provided by faculty was a real life data of marks of different branches of students including many other useful information (for my question) and other less useful information. Dataset also included missing values. And required preprocessing (I have included dataset with my project submission).

Compiler used: Spyder

Step Wise Strategy Opted For Solution

1) **Extracting (Forensic Science) data from complete dataset**
   a) Getting Basic Insight
      i) Type of features
      ii) Shape
      iii) Description
      iv) Number of missing values
2) **Preprocessing** -:
   a) Null value handing
   b) Finding categorical data(nunique)
   c) Label encoding (only categorical data)
   d) Standard Scaling (only Remaining features)

- **e)** Finding Correlation between all features and grades(for pruning non-essential features)
- **f)** Splitting set into target and data

3) Classification

- **a)** Splitting data into train and test
- **b)** Trying different classification algorithm while documenting effects of change in hyper parameters of each
- **c)** Keeping record of accuracy of all the algorithms
- **d)** Graphical representation of the best algorithm with best hyper parameter settings according to the question.

Library + Classes Used For Solution

- **Pandas** –: for handling dataset/ analysis/ deletion

- **Numpy**-: for in built features like sum (), array (), arrange () …etc.

- **Sklearn** :- for classes like SimpleImputer, LabelEncoder, StandardScaler, train_test_split(metrics), naive_bayes, metrics(onfusion_matrix, accuracy_score), tree, neighbors(KNeighborsClassifier), ensemble(BaggingClassifier, AdaBoostClassifier, RandomForestClassifier), linear_model(LogisticRegression), svm(SVC)

- **Matplotlib**:- for representing data graphically for analysis.

- **Seaborn**:- for making complex graphical plot for decision regarding data

Code explanation with reasons / output screenshot

And

Result visualization

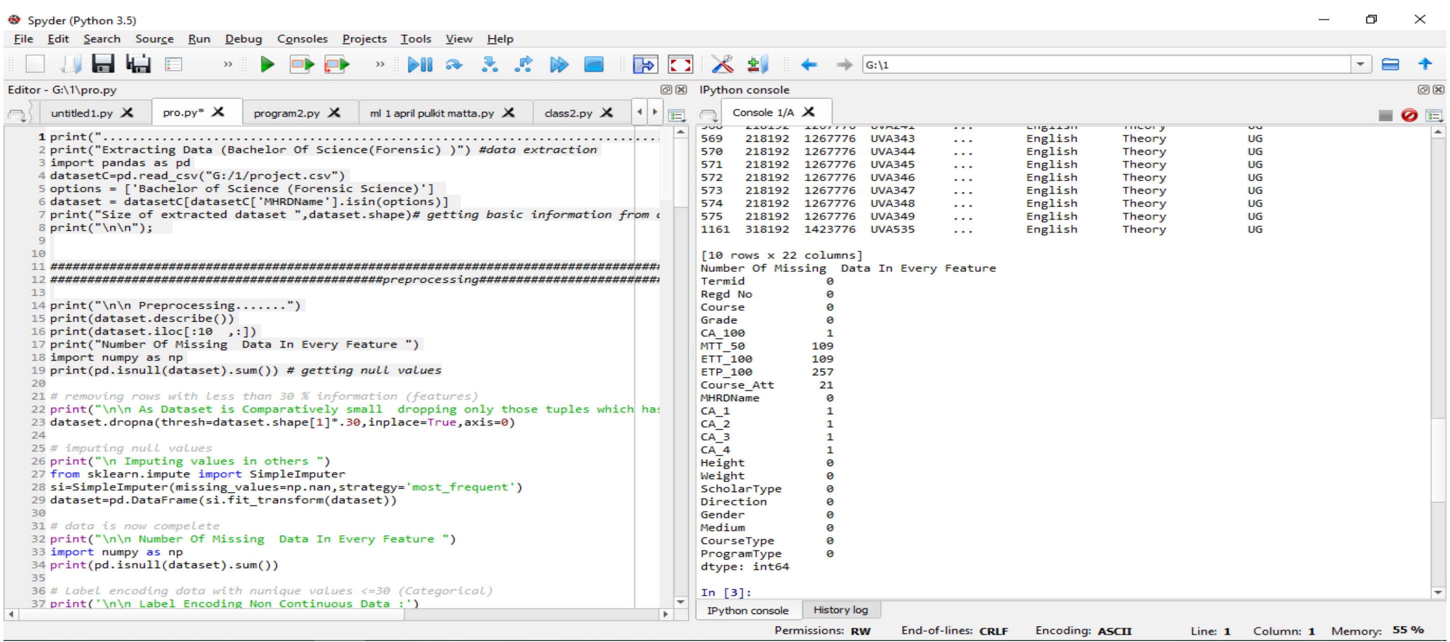
Key: hyper parameter(value)->(accuracy) ,

hyperparameterSetting1>> hyperparameterSetting2:   means 1 performed better with much higher accuracy)

- ## Basic features of data

Dataset consisted of different branches from which I extracted useful data Bachelor of Science (Forensic Science)

Extracting data basic features



- ## Preprocessing data

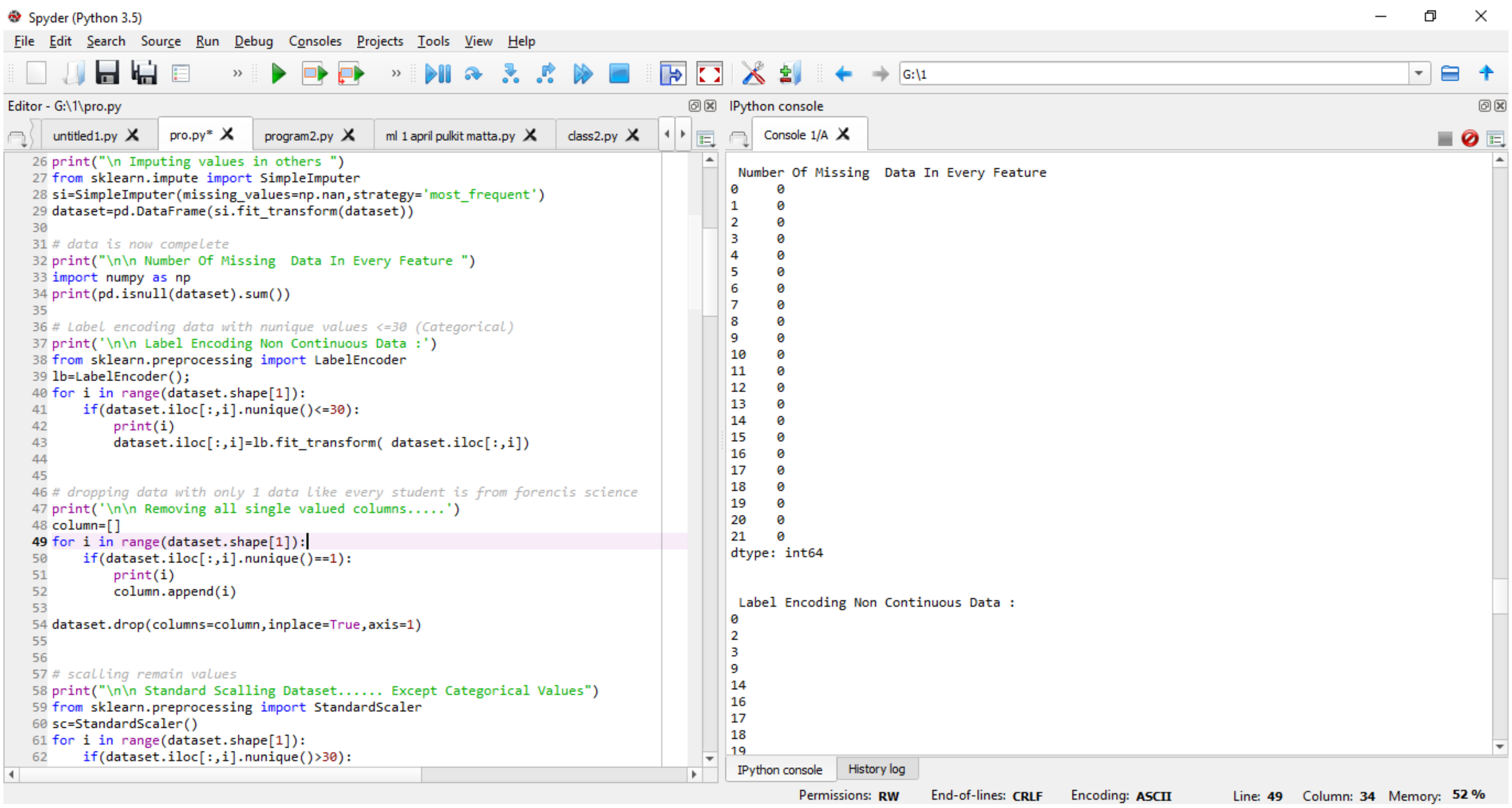
removing rows with less than 30 % information (features)

imputing null values

Label encoding data with nunique values <=30 (Categorical)

dropping data with only 1 data like every student is from forensic science

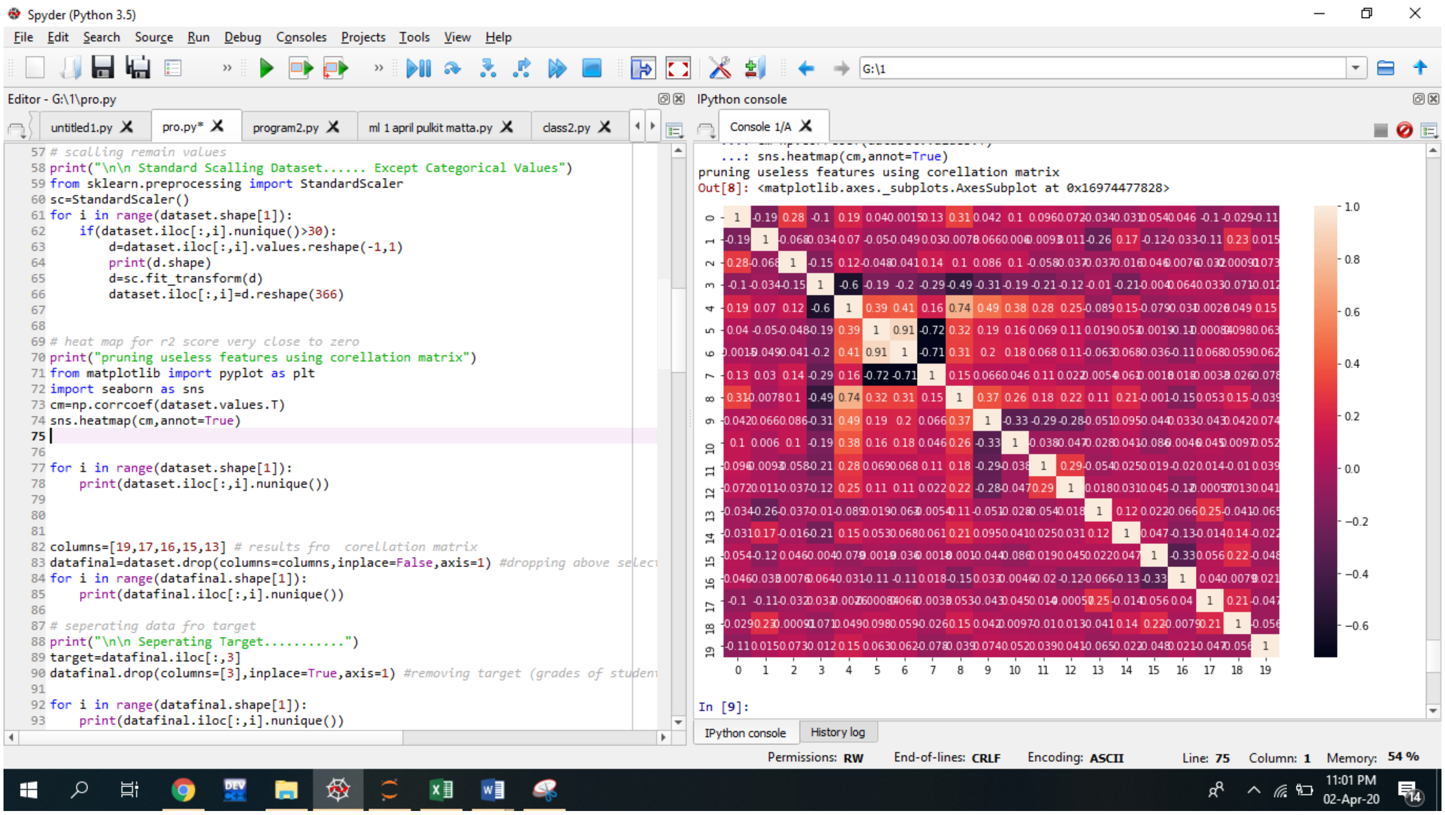
scaling remain values (Non Categorical Features)




```
[8 rows x 20 columns]
   0          1   2  3         4         5  ...        15  16  17  18  19  20
0  1  -1.976381   4  2 -0.253121  0.458265  ... -0.374231   1   2   0   0   1
1  1  -1.976381   8  0  0.167216  1.198773  ... -0.374231   1   2   0   0   1
2  1  -1.976381  12  0  1.049925  1.075355  ... -0.374231   1   2   0   0   1
3  1  -1.976381  13  0  0.419419 -1.207877  ... -0.374231   1   2   0   0   1
4  1  -1.976381  14  0 -0.127020 -1.207877  ... -0.374231   1   2   0   0   1
5  1  -1.976381  15  0  0.545520  1.137064  ... -0.374231   1   2   0   0   1
6  1  -1.976381  16  0  0.377385  0.951937  ... -0.374231   1   2   0   0   1
7  1  -1.976381  17  0  0.755689  0.766810  ... -0.374231   1   2   0   0   1
8  1  -1.976381  18  1  0.755689  1.383900  ... -0.374231   1   2   0   0   1
9  2  -1.937173  20  0  0.503486 -1.207877  ... -1.640019   1   1   0   0   1
```

removing redundant features

Heat map for correlation coefficient to find redundant features



As I discovered from heat map correlation coefficient of features like height, program type, medium gender, direction, scholar type are very close to zero when compared with feature number 3 which are grades which means

- they contribute very little in classification
- but increase over fitting
- therefore I removed them from dataset
- interesting fact: weight of the person was actually related to grades more than above features

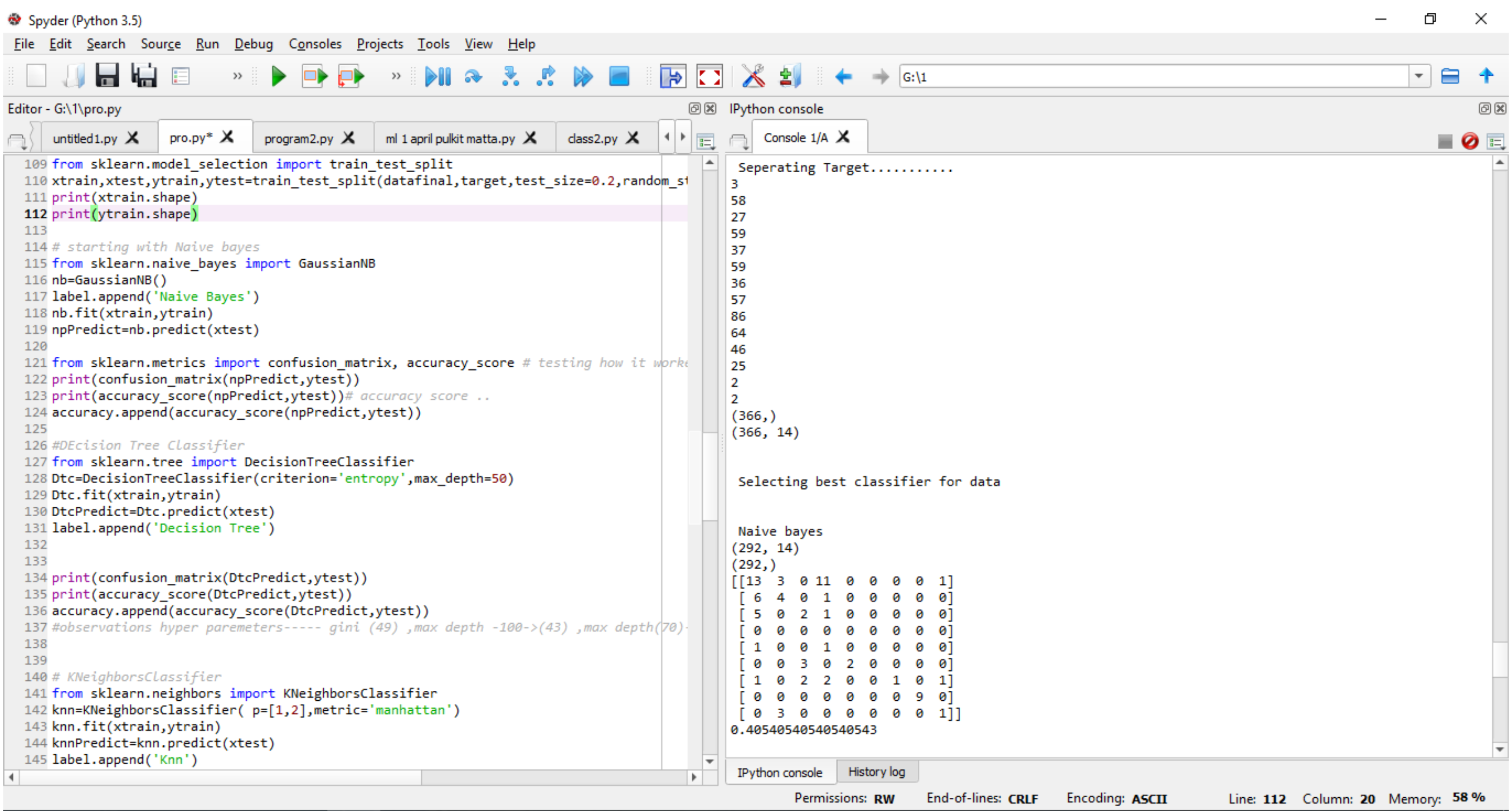
separating data from target

testing Different classification algorithms

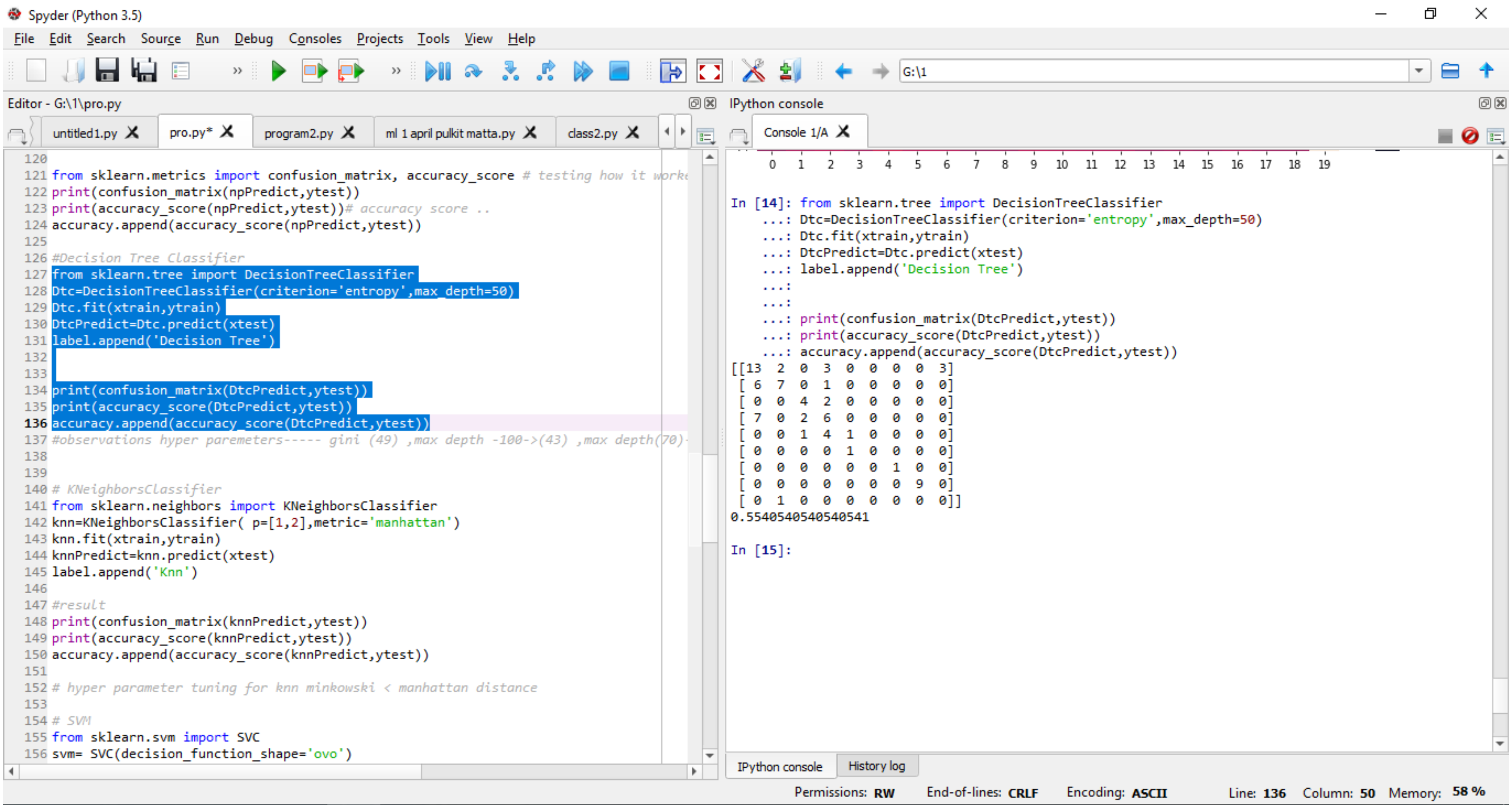
#splitting data

- **Classification**

1. **Starting with Naive Bayes**



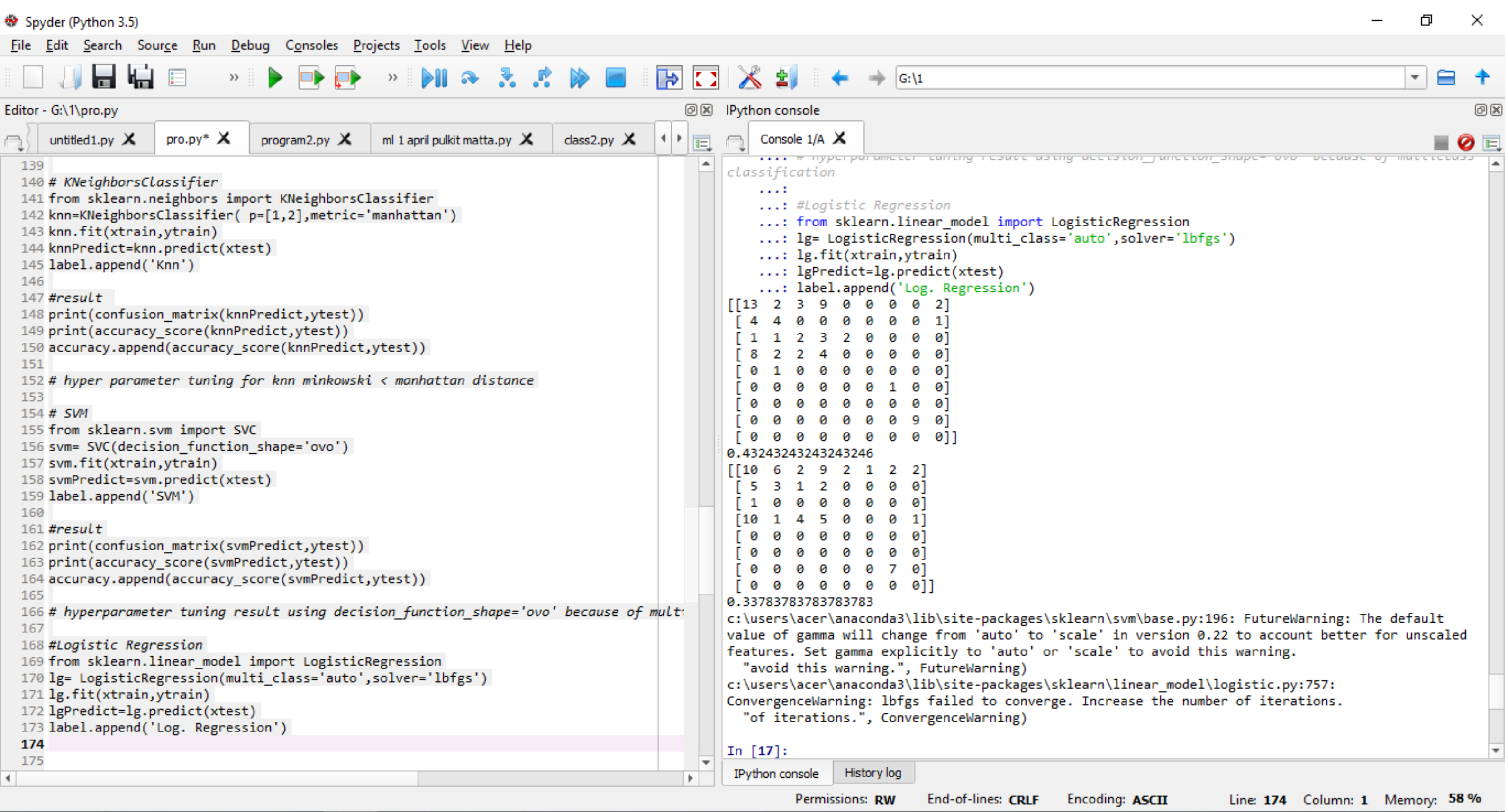
2. **Decision Tree Classifier**



#observations hyper parameters----- gini ->(49) ,max depth -100->(43) , max depth(70)->44 , max depth(49), using entropy ->(50)

3. KNeighborsClassifier

hyper parameter tuning for knn minkowski < manhattan distance



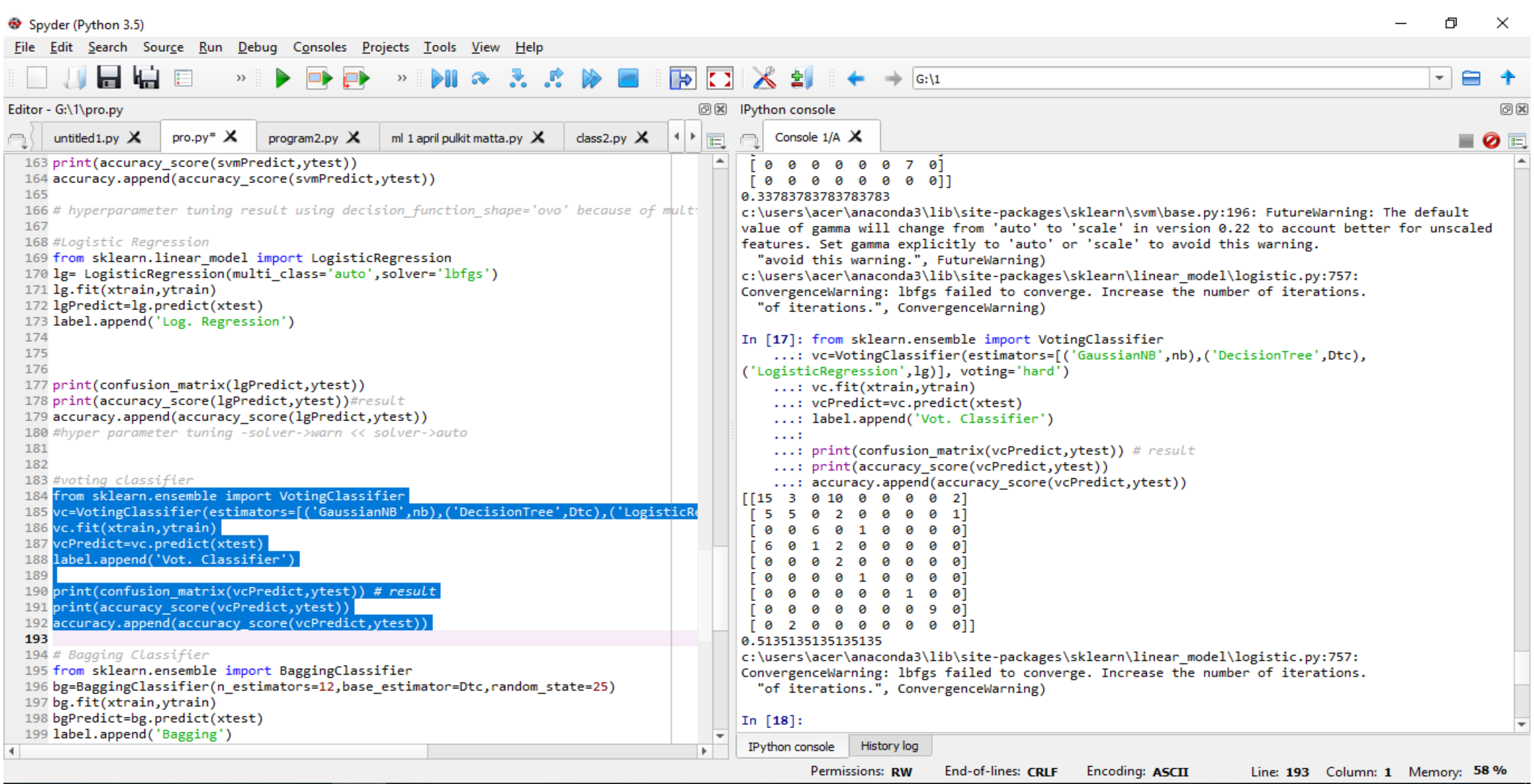
4. SVM

hyperparameter tuning result using decision_function_shape='ovo' because of multiclass classification

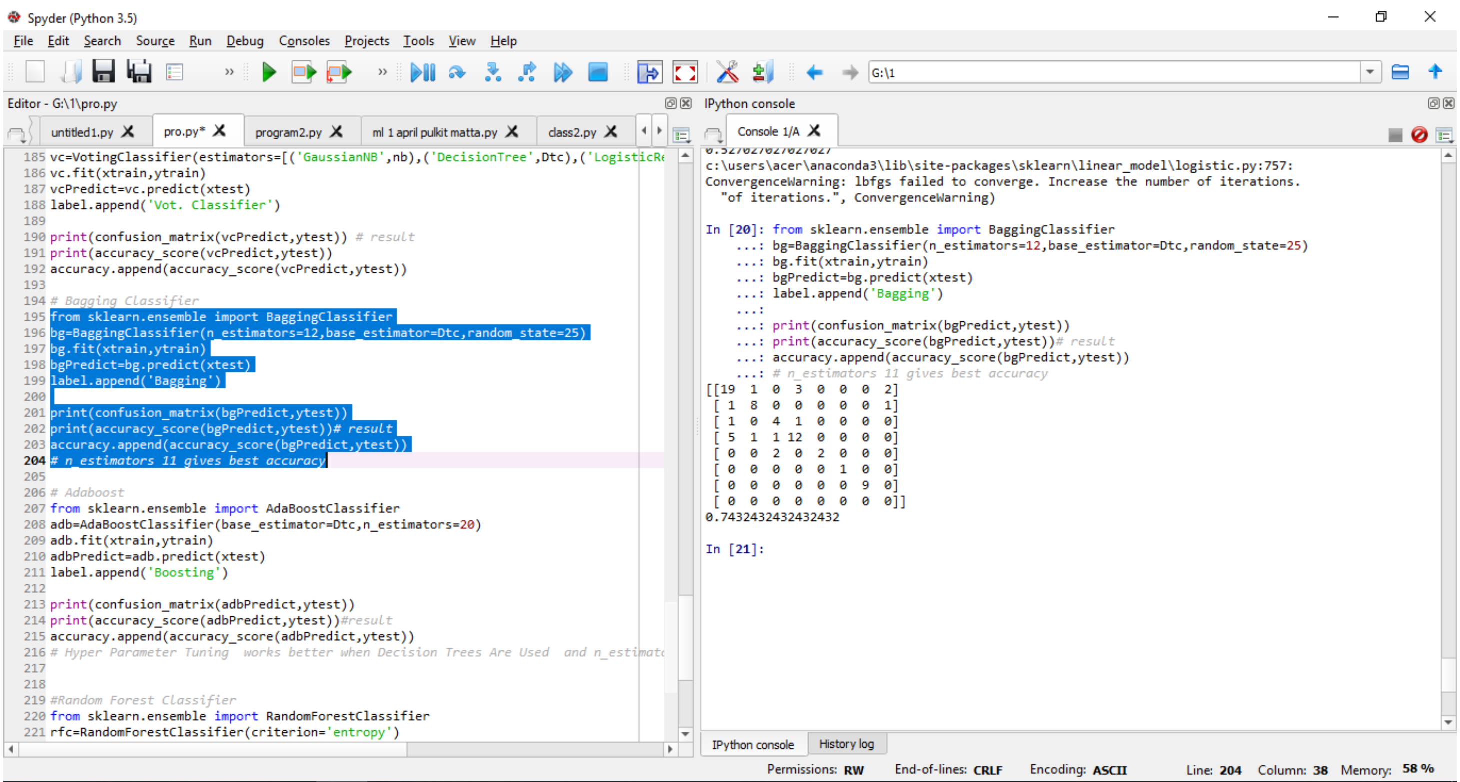
#Logistic Regression

#hyper parameter tuning -solver->warn << solver->auto

1) voting classifier

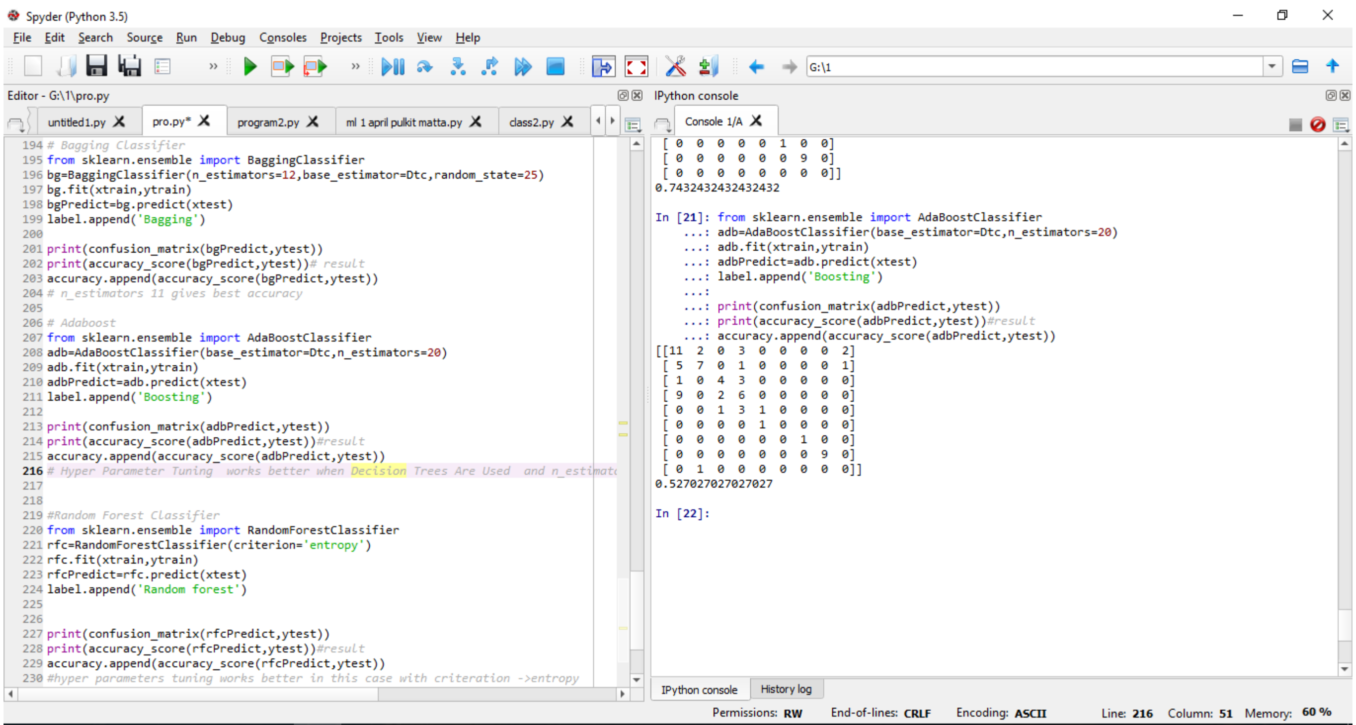


5. Bagging Classifier



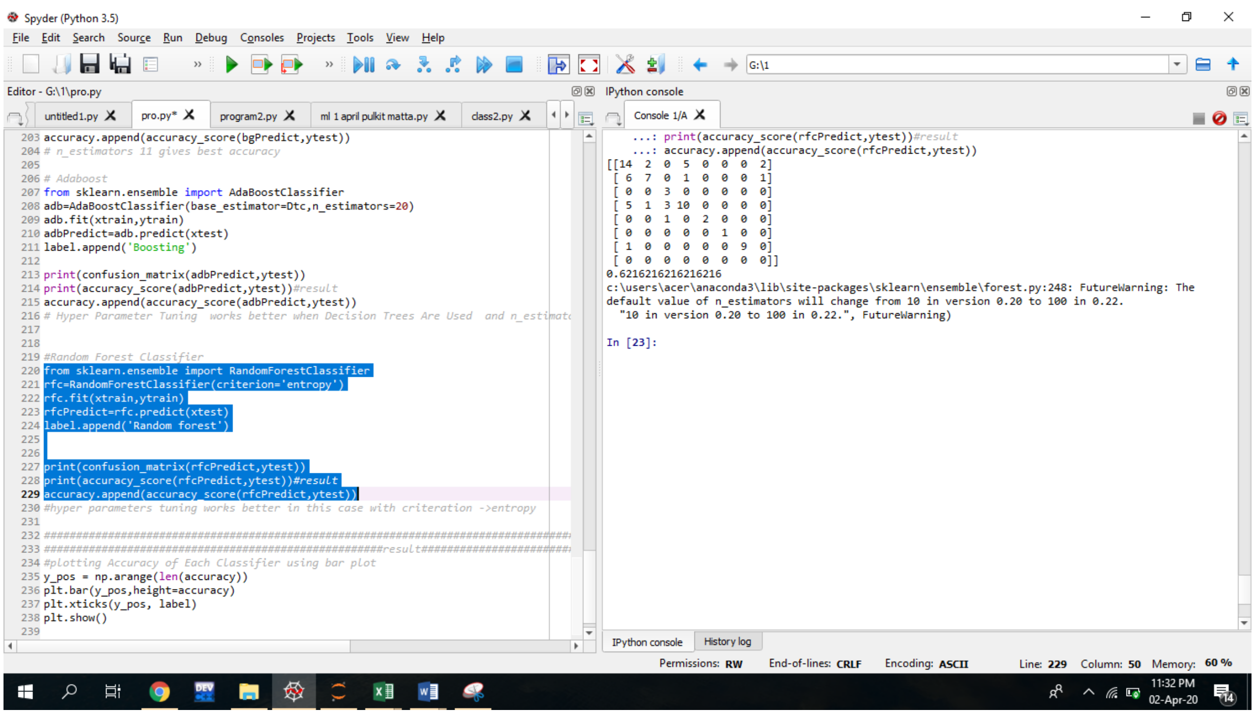
#hyperparameter tuning result: - n_estimators (11) ->(74.32)

6. Adaboost



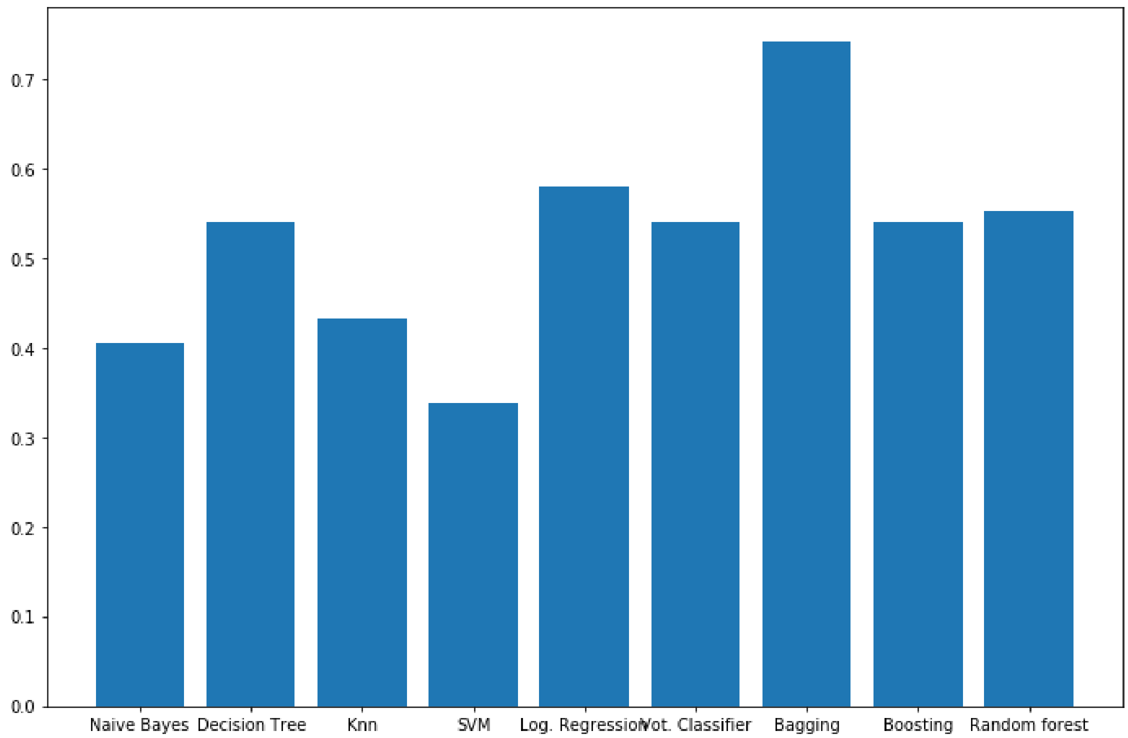
 # Hyper Parameter Tuning works better when Decision Trees Are Used and n_estimators (20)->52.75

7. Random Forest Classifier



#hyper parameters tuning works better in this case with criterion ->entropy as compared to gini

```
...: plt.show()
```



Accuracy graph

Final result + Observations

- According to the no free lunch theorem there is no universal classifier that can classify every problem as we observed that for this problem bagging is a better option .
- as I saw this in my project by iterating it many times with different setting of different models that minor settings can have a deep impact on results.
- This graphical result depicts that bagging performed better that every other classifier when (decision tree was used as an estimator for it)
- After tuning dataset I discovered that height, program type, medium gender, direction, scholar type, were responsible for over fitting for result as they were have correlation coefficient ~0 with grade of student . performance was much better after their removal
- From study of different truth of algorithm I observed that dataset was a little biased towards the grade A+.

- Scaling of non-categorical data helped graph like heat map faster