Stellar Classification

UML 501 Machine Learning Project Report

Submitted by:

(102116099) Aditi Binjola

(102116117) Sonali Jindal

BE Third Year, CSE

Group No: 3CS12

Submitted to: Dr. Anjula Mehto

Computer Science and Engineering Department

TIET, Patiala

November 2023

Introduction :

Project Overview:

Stellar classification is a fundamental task in astronomy aimed at categorizing celestial entities such as stars, galaxies, and quasars based on their observable features. The process involves analysing and interpreting data obtained through observations, allowing astronomers to understand the characteristics and behaviours of various cosmic bodies. This project revolves around the classification of celestial entities using the Stellar Classification Dataset - SDSS17 sourced from Kaggle. The dataset encapsulates observations made by the Sloan Digital Sky Survey (SDSS), offering a rich repository of information on stars, galaxies, and quasars.

Objectives:

- Construct a robust classification model for predicting the class of cosmic entities.
- Employ machine learning techniques, including feature engineering and exploratory data analysis.
- Implement various classification algorithms, such as Logistic Regression, K-Nearest Neighbours (KNN), Decision Tree, Random Forest, and XGBoost.
- Compare the performance of these algorithms based on accuracy scores.
- Fine-tune hyperparameters to optimize the models.
- Identify the algorithm that delivers the highest accuracy on the provided dataset.

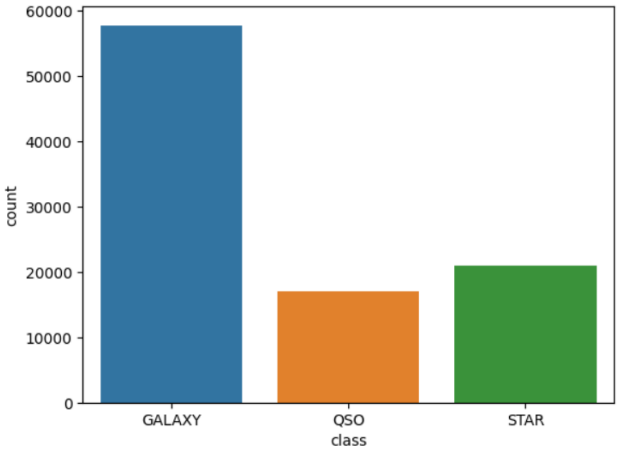
Problem Statement:

This project aims at classifying cosmic entities into their appropriate classes (stars, galaxies and quasars by using various classification algorithms such as Logistic Regression, K-Nearest Neighbours (KNN), Decision Tree, Gaussian Naïve Bayes, Random Forest, XGBoost and ANN. It is a multi class classification problem. Then the algorithms are compared on the basis of their accuracy score like precision, recall and F1 score to find the best one. In this way, we can identify the algorithm that provides highest accuracy to the given dataset.

Dataset :

[Dataset Link]

The data consists of 100,000 observations of space taken by the SDSS (Sloan Digital Sky Survey). Every data point is described by 17 feature columns and 1 class column which identifies it to be either. a star, galaxy, or quasar Among them, there are 59,445 examples of galaxies, 21,594 examples of stars and 18,691 examples of quasars, as shown in figure.



Features In the dataset -

1. object_ID = Object Identifier is the unique value that identifies the object in the image catalog used by SDSS.
2. alpha = Right Ascension angle (at J2000 epoch).
3. delta = Declination angle (at J2000 epoch).
4. UV_filter = Ultraviolet filter in the photometric system.
5. green_filter = Green filter in the photometric system.
6. red_filter = Red filter in the photometric system.
7. near_IR_filter = Near Infrared filter in the photometric system.
8. IR_filter = Infrared filter in the photometric system.
9. run_ID = Run Number used to identify the specific scan. Each run typically covers a specific area of the sky.
10. rereun_ID = Rerun Number to specify how the image was processed.
11. cam_col = Camera column to identify the scanline within the run.
12. field_ID = Field number to identify each field.
13. spec_obj_ID = Unique ID used for optical spectroscopic objects (this means that 2 different observations with the same spec_obj_ID must share the output class).
14. class = Object class (galaxy, star, or quasar object).
15. red_shift = Redshift value based on the increase in wavelength.
16. plate_ID = Plate ID, identifies each plate in SDSS.
17. MJD = Modified Julian Date, used to indicate when a given piece of SDSS data was taken.

18. fiber_ID = Fiber ID that identifies the fiber that pointed the light at the focal plane in each observation.

Methodology:

Data Exploration and Preprocessing:

- Firstly the exploratory data analysis is conducted to understand the structure and characteristics of the dataset.
- Missing or anomalous data are addressed through appropriate preprocessing techniques.
- Then outliers are detected and removed from the dataset

Feature Engineering:

- Identify and extract relevant features from the dataset that contribute to accurate classification and remove all the irrelevant features.
- Potential correlations between features are explored to enhance the model's predictive capabilities.

Algorithm Selection and Implementation:

- Evaluate and implement various machine learning algorithms like Logistic Regression, K-Nearest Neighbours (KNN), Decision Tree, Gaussian Naïve Bayes, Random Forest, XGBoost and ANN suitable for multiclass classification.
- Fine-tune hyperparameters to optimize the performance of selected algorithms.

Model Comparison and Evaluation:

- Assess the accuracy and efficiency of each implemented algorithm through various accuracy methods like precision, recall and f1 score.
- Compare models to identify the most effective algorithm for cosmic entity classification.

Data Visualization:

- It is the graphical representation of the data to cover insights, patterns and trends by the creation of visual elements like charts, graphs and maps to convey the information effectively.

Results:

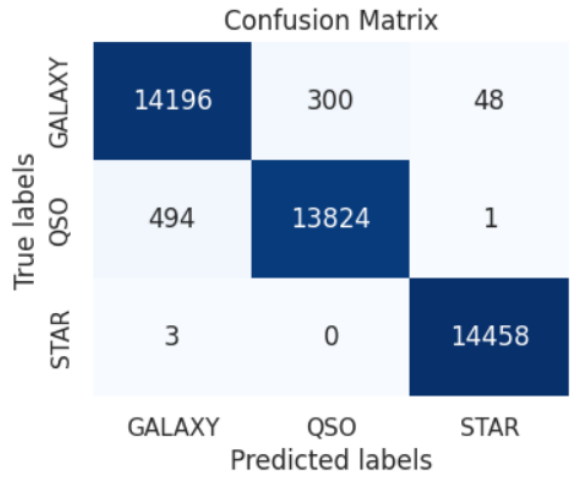
Precision, Recall score and F1 score of all the algorithms

	Algorithm	Recall score	Precision	F1 score
0	LogisticRegression	0.940218	0.940326	0.939692
1	KNN	0.929254	0.929069	0.929020
2	Decision Tree	0.969901	0.969896	0.969896
3	GaussianNB	0.803042	0.804517	0.797129
4	Random Forest	0.980473	0.980497	0.980460
5	XGBoost	0.978834	0.978898	0.978829
6	ANN	0.952128	0.952992	0.951990

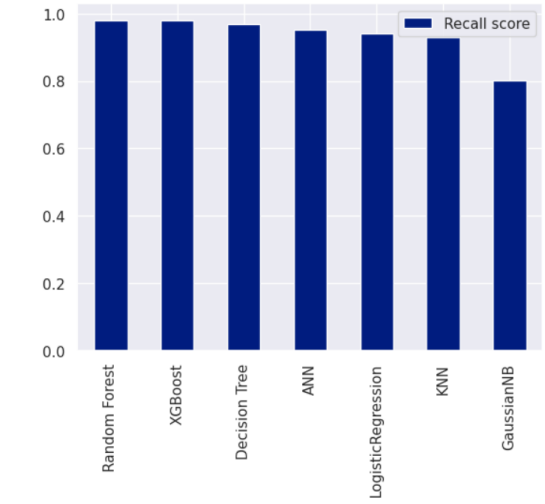
```
[139] #Random forest accuracy
      from sklearn.metrics import accuracy_score
      accuracy_score(y_test, y_pred5)
```

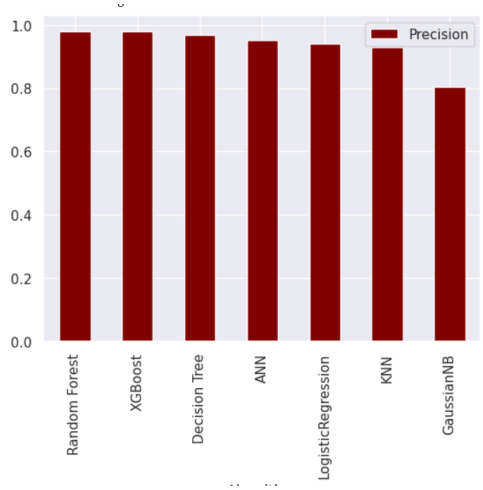
```
0.9804727172006278
```

Confusion Matrix

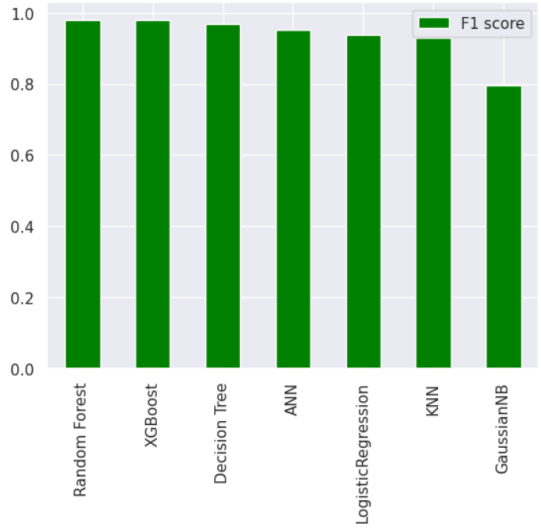


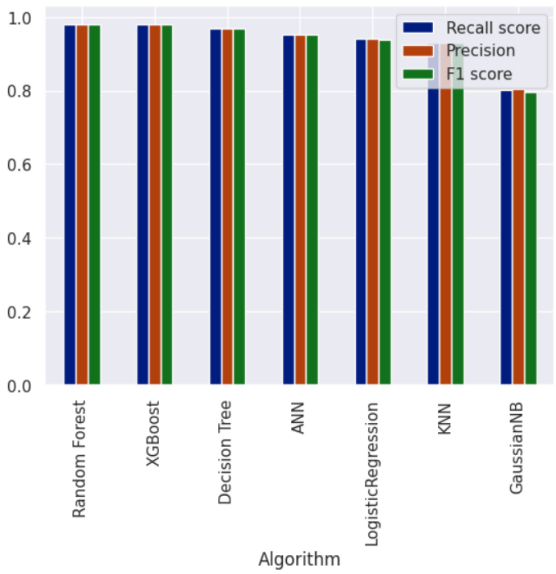
Bar Graphs

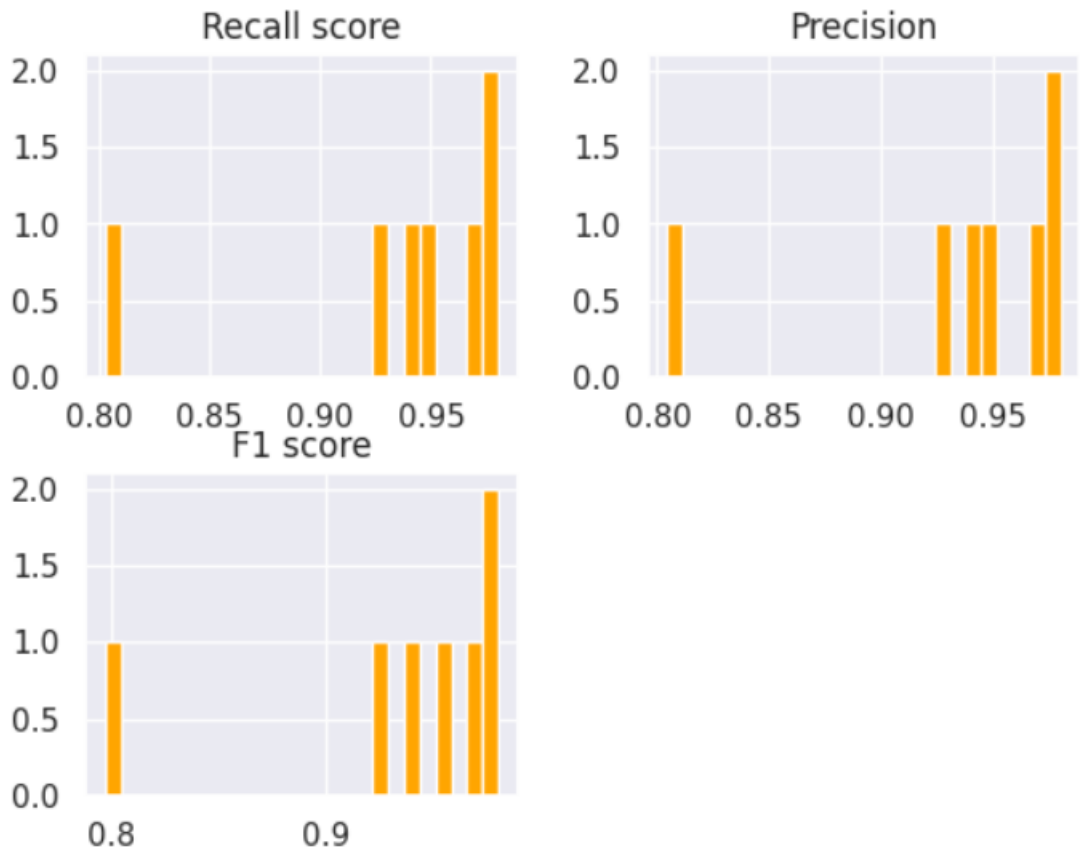

Recall score
1.0
0.8
0.6
0.4
0.2
0.0
Random Forest
XGBoost
Decision Tree
ANN
LogisticRegression
KNN
GaussianNB


Precision
1.0
0.8
0.6
0.4
0.2
0.0
Random Forest
XGBoost
Decision Tree
ANN
LogisticRegression
KNN
GaussianNB

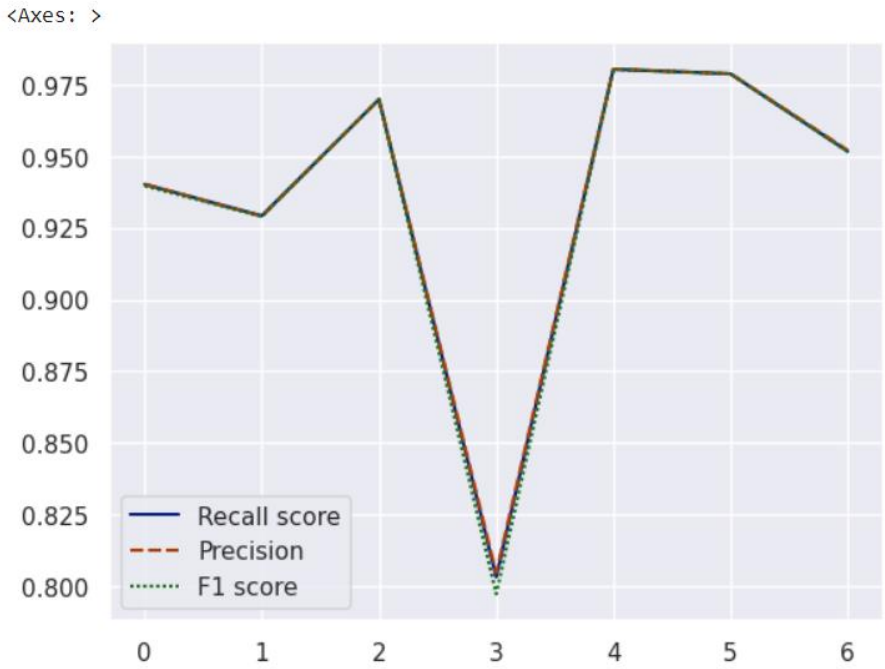
<Axes: xlabel='Algorithm'>


F1 score
1.0
0.8
0.6
0.4
0.2
0.0
Random Forest
XGBoost
Decision Tree
ANN
LogisticRegression
KNN
GaussianNB


Recall score
Precision
F1 score
1.0
0.8
0.6
0.4
0.2
0.0
Random Forest
XGBoost
Decision Tree
ANN
LogisticRegression
KNN
GaussianNB
Algorithm


Recall score
Precision
F1 score
2.0
1.5
1.0
0.5
0.0
0.80
0.85
0.90
0.95
0.8
0.9

Line plot


<Axes: >
0.975
0.950
0.925
0.900
0.875
0.850
0.825
0.800
Recall score
Precision
F1 score
0
1
2
3
4
5
6

Conclusion:

Random forest is the top performing model according to all the 3 parameters Precision, Recall and F1 score with an accuracy of 0.980472.

Precision	Algorithm	Recall score	F1 score
0.980497	Random Forest	0.980473	0.980460
0.978898	XGBoost	0.978834	0.978829
0.969896	Decision Tree	0.969901	0.969896
0.944223	ANN	0.943726	0.943438
0.940326	LogisticRegression	0.940218	0.939692
0.929069	KNN	0.929254	0.929020
0.804517	GaussianNB	0.803042	0.797129

F1 score	Algorithm	Recall score
0.980460	Random Forest	0.980473
0.978829	XGBoost	0.978834
0.969896	Decision Tree	0.969901
0.943438	ANN	0.943726
0.939692	LogisticRegression	0.940218
0.929020	KNN	0.929254
0.797129	GaussianNB	0.803042

F1 score	Algorithm	Recall score
0.980460	Random Forest	0.980473
0.978829	XGBoost	0.978834
0.969896	Decision Tree	0.969901
0.943438	ANN	0.943726
0.939692	LogisticRegression	0.940218
0.929020	KNN	0.929254
0.797129	GaussianNB	0.803042

Git link:

[Sonali Jindal Github Link]

[Aditi Binjola Github Link]