**Stellar Object Classification**

Develop comprehensive machine learning models to accurately classify celestial objects as stars, galaxies, or quasars. By utilizing these models, the institute aims to improve astronomical cataloging, research directions, and our understanding of the universe.

[**Click here for Dataset Link**](https://bostoninstituteofanalyti399-my.sharepoint.com/:f:/g/personal/projects_bostoninstituteofanalytics_org/EiE7MCiCw-9LlW3QJvIBB90BQtmT0oO4ido9mz53BiYNmg?e=aaqyg8)

**Project Overview:**

This project aims to develop machine learning models to classify celestial objects into stars, galaxies, and quasars. This project aims to faciltate efficient automated classification, insights into distinguishing features, and optimized data processing for large-scale surveys through the predictive models, data visualizations, and a comparative analysis of various algorithms.

**Project Benefits:**

* Automated Classification: The models will help efficiently categorize large numbers of celestial objects from survey data.
* Research Insights: Understanding the key features that distinguish different celestial objects can guide future astronomical research.
* Data Processing Optimization: Identifying the most effective classification algorithms can improve data processing pipelines for large-scale astronomical surveys.

**Deliverables:**

* A set of machine learning models designed for classifying celestial objects into stars, galaxies, and quasars.
* Comprehensive data visualizations illustrating relationships between various astronomical features and object classifications.
* A comparative analysis of different machine learning algorithms for the classification task.

**Project Guidelines:**

* Data Exploration: Conduct comprehensive EDA using visualization techniques to understand feature distributions and relationships with object classifications.
* Data Preprocessing: Apply appropriate scaling techniques for numerical variables and handle any necessary feature engineering.
* Model Selection: Evaluate various classification algorithms including Decision Trees, Random Forests, K-Nearest Neighbors, Support Vector Machines, Gradient Boosting, AdaBoost, and XGBoost.
* Performance Evaluation: Use appropriate metrics such as accuracy, precision, recall, and F1-score to assess model performance.
* Visualization: Create clear and informative visualizations using libraries like matplotlib and seaborn to communicate data insights and model performance.
* Code Quality: Ensure well-structured, commented Python code following best practices.

**Submission:**

* Submit Jupyter notebook (.ipynb) or Python script (.py) containing data analysis, preprocessing steps, and model training.
* Submit final report as .docx or PowerPoint Presentation.
* (Optional) Submit interactive dashboard for exploring classification results.
* NOTE: Create a zip file of the above-mentioned items for the final submission.

**Project assigned on: 7th October, 2024**