23 May – 5 June 2022

WEB-BASED AUTOMATIC CLASSIFICATION OF EXOPLANETS

COSMOSOC,
SPACE DATA SCIENCE,
IIT DHARWAD

Problem Statement

Build a stand-alone web-based application using open-source software(s) to identify and categorize The Threshold crossing events (TCE's). From the data given TCE's must be classified into different classes. False detection should be minimum, and TCE's must be Detected.

The problem can be broadly categorized into two parts:

1. Developing a statistical/machine learning model to cover the mentioned parameters.
2. Deploying it efficiently to a stand-alone application and web-based tool (no additional APIs used).

Basic Requirements

1. The web-based tool should accept the data in CSV file format. It should be able to analyze data and display plots picturing your findings from the data, the accuracy of your model, etc.

2. The statistical/machine learning model should classify the data of exoplanets, and the parameters used for the classification should be properly mentioned. Classification criteria are open for innovation.

3. The web-based tool should be developed completely using open source (Python/Perl etc.) software.

4. The teams must bring out the limitations of the method(s) used in the tool and possible enhancement later through research.

Data Source

You may find the dataset for this problem statement on this GitHub repository: i-was-here/SpaceDataSci-SoI (github.com)

This dataset was produced by the NASA Exoplanet Archive
http://exoplanetarchive.ipac.caltech.edu

If the participating team requires more data to train the model, they may look for more data over the internet to train their models

*The evaluation of model will be based on real data gathered from valid sources, like:
http://exoplanetarchive.ipac.caltech.edu

Software Specifications

1.

Web-Application

The final product must be deployed web-app. It must be a stand-alone application.

2.

Liscence

The tool must be shared under GPL or any other open source license.

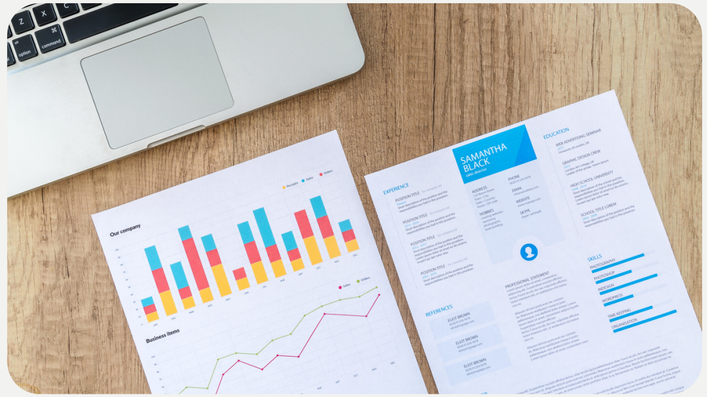
3.

Open Source

The app should use only open source, python is desirable.

CODES AND DOCUMENTATION

Submissions



Report/Documentation

The teams are supposed to make a brief report which should include the working and results(performance of model).

ReadMe

All teams must create a ReadMe file which will have all the necessary details related to getting the project working. Improper instructions written in the file will lead to penelty.

Submission

The complete project must be submitted as a GitHub repository, which will include all the codes, report, readme file or any other files.

Evaluation Criteria

Machine learning model:

01

Reporting multiple metrics like precision/recall, AUC ,etc. along with diagnostics and error analysis to provide insight into model performance will be judged

02

Overall creativity will be judged. This applies not only to your use of a unique machine learning technique, but could also be a unique problem formulation, visualization of the data, evaluation metric, or use of existing tools

Web-based application:

01

Ease of use of the GUI as well as the Python codes

02

Concise and clear documentation to ensure ease of use by a third party not knowing computational aspects

03

The tool should be compatible enough to get installed in all Linux environments without any patch up

Team size for this event is maximum 6 participants.
Participation awards shall be awarded to all participant**s.**