TECHNEX '26

13 MAR- 15 MAR 2026

STELLAR ANALYTICS

INTRODUCTION

Prelude: Signals from the Stars

The Era of Discovery

The year is 2236. Humanity no longer searches the cosmos blindly.

Over the past two centuries, advances in space telescopes and observational astronomy have transformed how we explore the universe. Missions like Kepler and its successors have continuously monitored distant stars, detecting tiny dips in starlight—possible signs of planets passing in front of their host stars.

These detections, known as Kepler Objects of Interest (KOIs), have revolutionized exoplanet science. However, not every signal tells the truth.

The Challenge of False Signals

Not all detected signals correspond to real planets.

Binary stars, stellar noise, instrumental errors, and background eclipsing systems often mimic planetary transits. As a result, astronomers face a crucial challenge:

Which signals represent real exoplanets, and which are false positives?

Even after confirmation, many planetary characteristics—such as planetary radius—must be estimated from incomplete and noisy data, relying heavily on stellar parameters and transit behavior.

To push the boundaries of discovery, astronomers now rely on data-driven intelligence to separate truth from illusion.

Mission Brief: The Stellar Verification Program

You are part of the Stellar Verification Program, a global initiative tasked with improving the reliability of exoplanet discovery.

Using a merged dataset of Kepler Objects of Interest (KOIs) and host star parameters, your mission is twofold:

- Distinguish confirmed exoplanets from false positives
- Accurately predict the planetary radius of confirmed exoplanets

Your work will help refine detection pipelines and bring us closer to understanding the diversity of worlds beyond our solar system.

GENERAL INSTRUCTIONS

Team Size: Upto a maximum of 3 members

Dataset Access:
The dataset provided contains:
- KOI-level observational parameters
- Host star physical properties

Dataset links:
- KOI–Host Star Merged Dataset

Primary Data Source:
- NASA Exoplanet Archive

(Participants may refer to official documentation for additional context.)

Additional Resources:
- Kepler Objects of Interest (KOIs) – NASA Exoplanet Archive
- Exoplanet Detection Methods
- Transit Photometry and Light Curve Analysis
- False Positive Scenarios in Exoplanet Detection

(All external resources are for reference and conceptual understanding only.)

Problem Breakdown:
- Build an end-to-end system that accepts user inputs through a frontend interface
- Perform robust input validation and apply a consistent preprocessing pipeline
- Implement machine learning models for:
  - Classification of confirmed exoplanets vs false positives
  - Regression for planetary radius prediction
- Enable real-time inference using trained models via backend services
- Present predictions with interpretable outputs and simple visual insights

Programming Environment:
- Participants are free to use any programming language, framework, or library.
- Preferred programming environments: Google Colab, Jupyter Notebook

Dataset Description:
The dataset contains one row per Kepler Object of Interest (KOI) and includes:

Transit-related features:
- Orbital period, Transit depth and duration, Impact parameter, Signal-to-noise metrics, Number of observed transits

Host-star properties:
- Effective temperature, Surface gravity, Metallicity, Stellar mass, radius, and density, Associated measurement uncertainties

All features are provided in raw form, without normalization, imputation, or feature synthesis.
.

PROBLEM STATEMENT

Round 1

Required System Architecture

1. Frontend Layer (User Interface)
   - Provide a clean and intuitive interface for users to enter input features required by the model
   - Display brief descriptions or tooltips explaining each input feature and relevant ML terms
   - Allow users to submit input data for prediction
   - Display prediction results and visualizations clearly
2. Frontend Validation
   - Perform client-side validation to ensure:
   - Required fields are not empty
   - Input values are within acceptable ranges
   - Data types (numeric, categorical, etc.) are correct
   - Provide immediate feedback to users for invalid inputs before making API calls
3. API Communication with Backend
   - Send validated input data from the frontend to the backend via RESTful API calls
   - Ensure secure and structured data transfer (e.g., JSON format)
   - Handle API responses, errors, and timeouts gracefully
4. Build ML Pipeline
   - Reproducible preprocessing pipeline
   - Feature selection informed by EDA
   - Support for both classification and regression inference
5. Model Inference
   - Load a pre-trained machine learning model in the backend
   - Perform real-time inference on preprocessed input data
   - Support extensibility for future model updates
6. Prediction Output and Visualization
   - Return prediction results to the frontend in a structured format
   - Display:
   - Final prediction values (e.g., class label, probability, regression output)
   - Confidence scores if applicable
   - Generate simple visualizations (e.g., bar charts, probability distributions, trend graphs) to improve interpretability
7. Bonus Implementations
   - Store user inputs, timestamps, and prediction outputs in a database
   - Enable tracking of past predictions for auditing, debugging, or analytics
   - Ensure data privacy and secure storage practices
   - Build low latency pipeline and reliable predictions

TECHNEX'26
13 MAR- 15 MAR 2026

Predictive Tasks

Task A — Classification

Goal:

Distinguish CONFIRMED exoplanets from FALSE POSITIVE transit signals.

- Target column: koi_disposition

Participants are expected to carefully inspect the dataset and frame the problem accordingly.

Task B — Regression

Goal:

Predict the planetary radius (in Earth radii) for exoplanets.

- Target column: koi_prad
-

Important Notes

- Some features may not be available at prediction time in real observational scenarios.
- Certain columns may be appropriate for one task but not the other.
- Participants are expected to perform exploratory data analysis and make informed feature-selection decisions.
- Care should be taken to avoid target leakage.
- You can choose any tech stack you want.
- Suggestions for beginners -
- You can use
- React.js for Frontend
- Firestore firebase or PostgreSql for Databases
- Node/ Django/ Flask for backend
- Whichever you find easier to work with
- A deployed web application would gain more points than a locally hosted application.

Evaluation

Classification

- Metrics: F1-score, ROC–AUC
- Evaluation focuses on the binary distinction defined in Task A.

Regression

- Metrics: RMSE, MAE
- Evaluation is performed on physically meaningful predictions.

System Development Performance

- Correctness and consistency of input validation and preprocessing
- Reliability and latency of real-time inference
- Robust API communication and error handling
- Clarity and interpretability of prediction outputs

Submission Format

PDF Document

The PDF should include:

- Title Page:
- Project title, team name/members, date, and contact information

- Problem Statement:
- Brief summary of the problem, objectives, and predictive tasks
- Assumptions:
- Key assumptions made regarding data availability, feature usage, and system design
- ML Tasks Description:
  - Classification task: distinguishing confirmed exoplanets and false positives
  - Regression task: planetary radius prediction
  - Choice of targets and framing of the problem
- Approach:
  - Data preprocessing and feature engineering (including SDE)
  - Model selection and training methodology for both tasks
  - Overview of system architecture (frontend, backend, inference flow)
- Results & Insights:
  - Summary of classification and regression performance
  - Key observations, feature importance, and limitations
- Appendix (Optional):
  - Additional plots, dashboards, references, or system diagrams

ROUND 2

Dashboard Requirements – Stellar Analytics 2026

The Round 2 dashboard should be visually clear, scientifically informative, and interactive, effectively communicating analysis, model results, and insights.

Key Elements to Include

- Introduction & Context
  - Mission brief and objectives
  - Dataset summary (number of KOIs, confirmed vs false positives, key features, preprocessing challenges)
- Visualizations & Insights
  - Distributions of key transit and stellar features
  - Correlation plots and insights from engineered features
- Classification Results
  - Visualization of predicted classes (confirmed vs false positives)
  - Model performance metrics (e.g., F1-score, ROC–AUC, confusion matrix)
- Regression Results
  - Predicted vs actual planetary radius
  - Error or residual analysis
- Bonus (Optional)
  - Interactive tables/cards for selected objects
  - Hover-based insights for detailed exploration

Note

An element of surprise may be introduced in Round 2. Qualified teams may be required to incorporate additional features and adapt their pipelines accordingly.

EVENT COORDINATORS

- Shubham Khetan: 93285 24215
- Tapish Samaiya: 80814 69994