I am participating in the NASASpaceApps Challenge. My challenge is A World Away: Hunting for Exoplanets with AI.  
  
This is the information about the challenge.   
SUMMARY  
Data from several different space-based exoplanet surveying missions have enabled discovery of thousands of new planets outside our solar system, but most of these exoplanets were identified manually. With advances in artificial intelligence and machine learning (AI/ML), it is possible to automatically analyze large sets of data collected by these missions to identify exoplanets. Your challenge is to create an AI/ML model that is trained on one or more of the open-source exoplanet datasets offered by NASA and that can analyze new data to accurately identify exoplanets.  
  
BACKGROUND  
Exoplanetary identification is becoming an increasingly popular area of astronomical exploration. Several survey missions have been launched with the primary objective of identifying exoplanets. Utilizing the “transit method” for exoplanet detection, scientists are able to detect a decrease in light when a planetary body passes between a star and the surveying satellite. Kepler is one of the more well-known transit-method satellites, and provided data for nearly a decade. Kepler was followed by its successor mission, K2, which utilized the same hardware and transit method, but maintained a different path for surveying. During both of these missions, much of the work to identify exoplanets was done manually by astrophysicists at NASA and research institutions that sponsored the missions. After the retirement of Kepler, the Transiting Exoplanet Survey Satellite (TESS), which has a similar mission of exoplanetary surveying, launched and has been collecting data since 2018.  
  
For each of these missions (Kepler, K2, and TESS), publicly available datasets exist that include data for all confirmed exoplanets, planetary candidates, and false positives obtained by the mission (see Resources tab). For each data point, these spreadsheets also include variables such as the orbital period, transit duration, planetary radius, and much more. As this data has become public, many individuals have researched methods to automatically identify exoplanets using machine learning. But despite the availability of new technology and previous research in automated classification of exoplanetary data, much of this exoplanetary transit data is still analyzed manually. Promising research studies have shown great results can be achieved when data is automatically analyzed to identify exoplanets. Much of the research has proven that preprocessing of data, as well as the choice of model, can result in high-accuracy identification. Utilizing the Kepler, K2, TESS, and other NASA-created, open-source datasets can help lead to discoveries of new exoplanets hiding in the data these satellites have provided.  
  
OBJECTIVES  
Your challenge is to create an artificial intelligence/machine learning model that is trained on one or more of NASA’s open-source exoplanet datasets, and not only analyzes data to identify new exoplanets, but includes a web interface to facilitate user interaction. A number of exoplanet datasets from NASA’s Kepler, K2, and TESS missions are available (see Resources tab). Feel free to utilize any open-source programming language, machine learning libraries, or software solutions that you think would fit into this project well. Think about the different ways that each data variable (e.g., orbital period, transit duration, planetary radius, etc.) might impact the final decision to classify the data point as a confirmed exoplanet, planetary candidate, or false positive. Processing, removing, or incorporating specific data in different ways could mean the difference between higher-accuracy and lower-accuracy models. Think about how scientists and researchers may interact with the project you create. Will you allow users to upload new data or manually enter data via the user interface? Will you utilize the data users provide to update your model? The choices are endless!  
  
POTENTIAL CONSIDERATIONS  
You may (but are not required to) consider the following:  
Your project could be aimed at researchers wanting to classify new data or novices in the field who want to interact with exoplanet data and do not know where to start.  
Your interface could enable your tool to ingest new data and train the models as it does so.  
Your interface could show statistics about the accuracy of the current model.  
Your model could allow hyperparameter tweaking from the interface.  
  
The datasets I am planning to use are as follows:  
@https://exoplanetarchive.ipac.caltech.edu/cgi-bin/TblView/nph-tblView?app=ExoTbls&config=cumulative  
@https://exoplanetarchive.ipac.caltech.edu/cgi-bin/TblView/nph-tblView?app=ExoTbls&config=k2pandc  
@https://exoplanetarchive.ipac.caltech.edu/cgi-bin/TblView/nph-tblView?app=ExoTbls&config=TOI

Features:

**Tier 1: Critical Features (Must Include)**

critical\_features = [

'pl\_orbper', *# Orbital period*

'pl\_rade', *# Planetary radius*

'pl\_bmasse', *# Planetary mass*

'pl\_ratdor', *# Semi-major axis*

'pl\_eqt', *# Equilibrium temperature*

'pl\_trandep', *# Transit depth*

'pl\_trandur', *# Transit duration*

'st\_rad', *# Stellar radius*

'st\_mass', *# Stellar mass*

'st\_teff', *# Stellar temperature*

'discoverymethod', *# Detection method*

'pl\_status' *# Confirmation status*

]

**Tier 2: Important Features (Should Include)**

important\_features = [

'pl\_eccen', *# Orbital eccentricity*

'pl\_orbincl', *# Orbital inclination*

'pl\_dens', *# Planetary density*

'pl\_insol', *# Insolation flux*

'pl\_imppar', *# Impact parameter*

'pl\_ratror', *# Planet-to-star radius ratio*

'st\_met', *# Stellar metallicity*

'st\_logg', *# Stellar surface gravity*

'sy\_pnum', *# Number of planets in system*

'pl\_facility' *# Discovery facility*

]

**Tier 3: Additional Features (Nice to Have)**

additional\_features = [

'pl\_tranmid', *# Transit midpoint*

'sy\_dist', *# System distance*

'pl\_telescope', *# Discovery telescope*

'pl\_orbpererr1', *# Orbital period error*

'pl\_radeerr1', *# Radius error*

'pl\_bmasseerr1' *# Mass error*

]