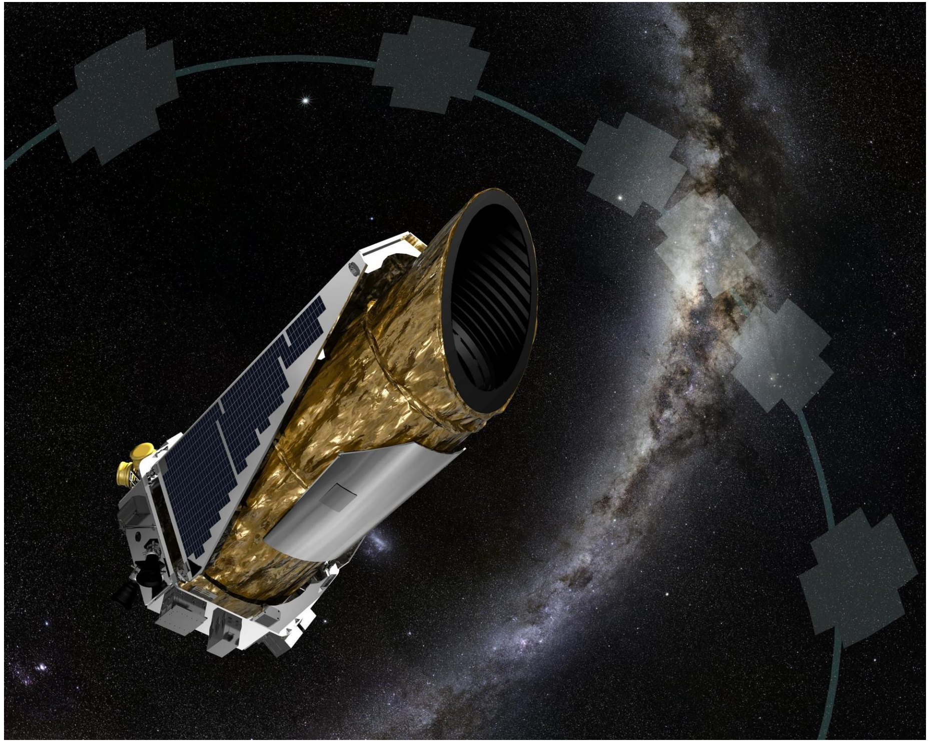
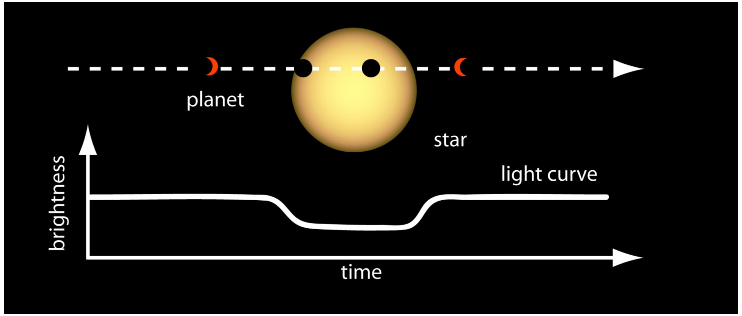
Project Report: A Machine Learning Model for Classifying Potential Exoplanets from Kepler Data



The Kepler Space Telescope. Source: NASA

Johannes Kepler (1571 - 1630)

Background



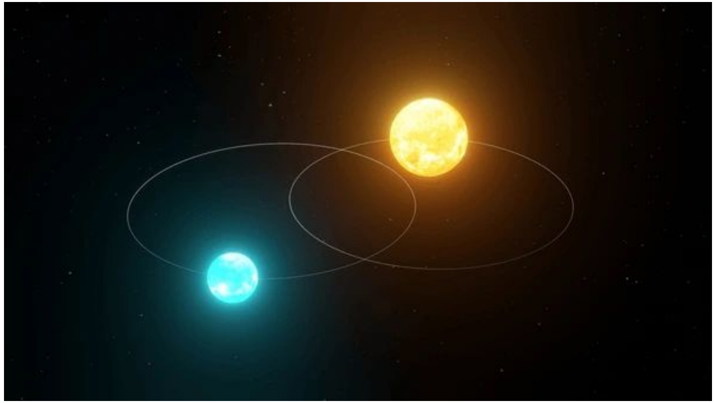
- NASA's Kepler Space Telescope has detected thousands of exoplanets (planets orbiting stars outside of our solar system) by measuring dips in stellar brightness caused by planets passing in front of the star.
- Not all changes in brightness are caused by exoplanets; some are caused by other stars or measurement noise.
- Exoplanet candidates are called Kepler Objects of Interest (KOIs), and are characterized by many measured variables and categorical features.
- The NASA Exoplanet Archive contains a table of past KOIs which have been marked as **CONFIRMED** (true exoplanet) or **FALSE POSITIVE** based on follow-up observations.

Project Goal: Accelerate the process of exoplanet discovery by training a machine learning model to classify KOIs from the NASA Exoplanet Archive as CONFIRMED or FALSE POSITIVE

- Utilize tabulated data from the KOI Cumulative Table.
- Try both supervised and unsupervised methods.
- Produce an accurate, *interpretable* model with visualizations of results.
- Discover insights about which features of the dataset are most useful.

Related Work

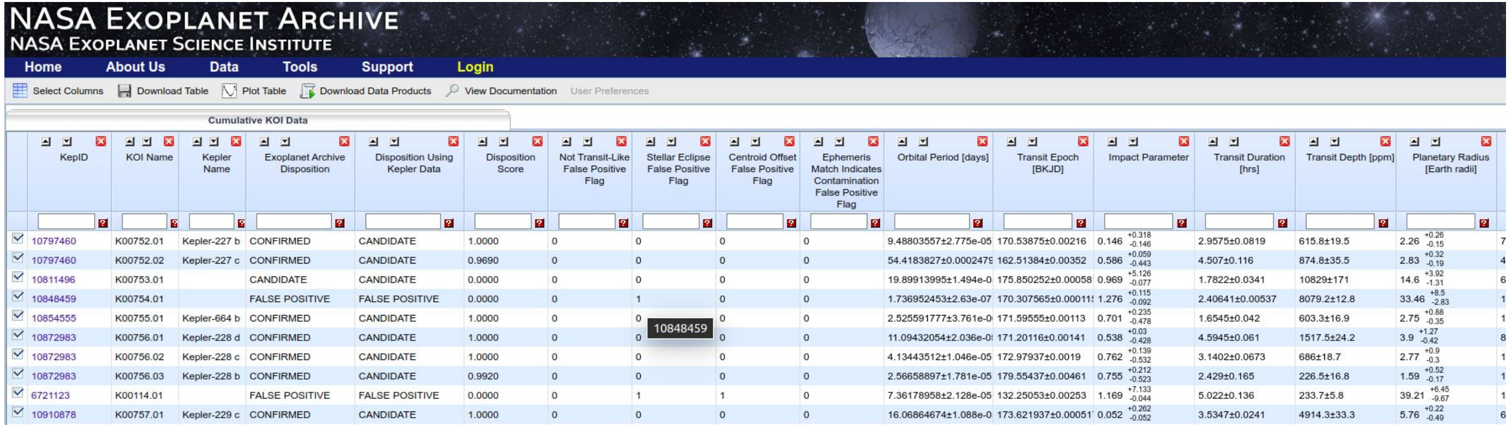
- Morton et. al (2016) developed a statistical modeling tool called VESPA for calculating false positive probabilities from several combined data sources [1].
- Shallue & Vanderburg (2018) achieved high accuracy (>98%) using convolutional neural networks (CNNs) trained on time-series Kepler light curve data [2].
- Our project complements these approaches with a more interpretable classifier based on tabular data.
- The Kepler False Positive Working Group gained insight into some of the most common causes of false positives:
  - Orbiting binary star
  - Fluctuation in brightness
  - Electronic crosstalk



1. Morton, T.D. et al. *False positive probabilities for all Kepler objects of interest.* (2016). The Astrophysical Journal.
2. Shallue, Christopher J and Vanderburg, Andrew. *Identifying Exoplanets with Deep Learning.* (2018). The Astronomical Journal.

Data Understanding

- Studied the features of the KOI Cumulative Table and selected the most useful ones, omitting irrelevant or redundant features.
- Removed features that gave away information about the target variable, such as “false positive flags” indicating the reason for a false positive.
- Removed features that were derived from combinations of other features, such as “planetary radius”, which was estimated based on star and telescope data already in the table.



KepID	KOI Name	Kepler Name	Exoplanet Archive Disposition	Disposition Using Kepler Data	Disposition Score	Not Transit-Like False Positive Flag	Stellar Eclipse False Positive Flag	Centroid Offset False Positive Flag	Ephemeris Match Indicates Contamination False Positive Flag	Orbital Period [days]	Transit Epoch [BKJD]	Impact Parameter	Transit Duration [hrs]	Transit Depth [ppm]	Planetary Radius [Earth radii]	
10797460	K00752.01	Kepler-227 b	CONFIRMED	CANDIDATE	1.0000	0	0	0	0	9.48803557±2.775e-05	170.53875±0.00216	$0.146^{+0.318}_{-0.146}$	2.9575±0.0819	615.8±19.5	$2.26^{+0.26}_{-0.15}$	7
10797460	K00752.02	Kepler-227 c	CONFIRMED	CANDIDATE	0.9690	0	0	0	0	54.4183827±0.0002479	162.51384±0.00352	$0.586^{+0.059}_{-0.443}$	4.507±0.116	874.8±35.5	$2.83^{+0.32}_{-0.19}$	4
10811496	K00753.01		CANDIDATE	CANDIDATE	0.0000	0	0	0	0	19.89913995±1.494e-0	175.850252±0.00058	$0.969^{+5.126}_{-0.077}$	1.7822±0.0341	10829±171	$14.6^{+3.92}_{-1.31}$	6
10848459	K00754.01		FALSE POSITIVE	FALSE POSITIVE	0.0000	0	1	0	0	1.736952453±2.63e-07	170.307565±0.00011	$1.276^{+0.115}_{-0.092}$	2.40641±0.00537	8079.2±12.8	$33.46^{+8.5}_{-2.83}$	1
10854555	K00755.01	Kepler-664 b	CONFIRMED	CANDIDATE	1.0000	0	0	0	0	2.525591777±3.761e-0	171.59555±0.00113	$0.701^{+0.235}_{-0.478}$	1.6545±0.042	603.3±16.9	$2.75^{+0.88}_{-0.35}$	1
10872983	K00756.01	Kepler-228 d	CONFIRMED	CANDIDATE	1.0000	0	0	0	0	11.09432054±2.036e-0	171.20116±0.00141	$0.538^{+0.03}_{-0.428}$	4.5945±0.061	1517.5±24.2	$3.9^{+1.27}_{-0.42}$	8
10872983	K00756.02	Kepler-228 c	CONFIRMED	CANDIDATE	1.0000	0	0	0	0	4.13443512±1.046e-05	172.97937±0.0019	$0.762^{+0.139}_{-0.532}$	3.1402±0.0673	686±18.7	$2.77^{+0.9}_{-0.3}$	1
10872983	K00756.03	Kepler-228 b	CONFIRMED	CANDIDATE	0.9920	0	0	0	0	2.56658897±1.781e-05	179.55437±0.00461	$0.755^{+0.212}_{-0.523}$	2.429±0.165	226.5±16.8	$1.59^{+0.52}_{-0.17}$	1
6721123	K00114.01		FALSE POSITIVE	FALSE POSITIVE	0.0000	0	1	1	0	7.36178958±2.128e-05	132.25053±0.00253	$1.169^{+7.133}_{-0.044}$	5.022±0.136	233.7±5.8	$39.21^{+6.45}_{-9.67}$	1
10910878	K00757.01	Kepler-229 c	CONFIRMED	CANDIDATE	1.0000	0	0	0	0	16.06864674±1.088e-0	173.621937±0.00051	$0.052^{+0.262}_{-0.052}$	3.5347±0.0241	4914.3±33.3	$5.76^{+0.22}_{-0.49}$	6

Data Inspection and Cleaning

- After feature selection, data was imported to Pandas dataframe containing 9564 rows and 18 columns.
- Surprisingly, features are all numerical.
- Rows without a positive or negative target label (marked as CANDIDATE) were removed, leaving 7585 entries with a 36%-64% positive-negative split.
- Most columns had some missing values.
- Missing values were imputed using the median of the non-null values.

```
df['koi_disposition'].value_counts()
```

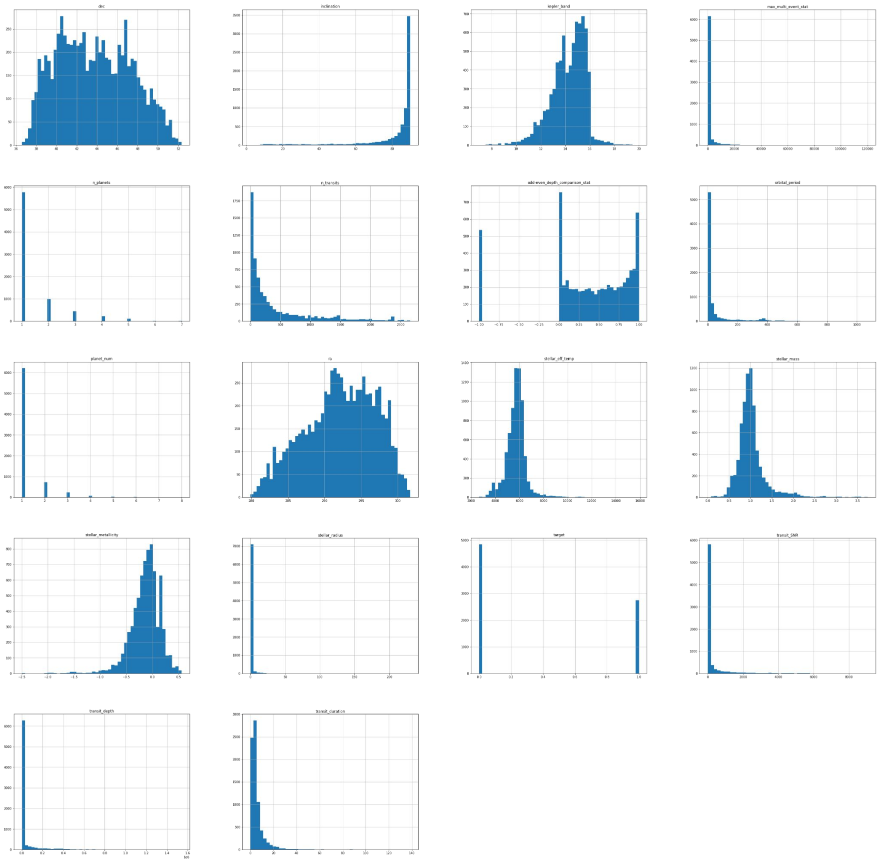
```
FALSE POSITIVE    4839
CONFIRMED         2746
CANDIDATE         1979
Name: koi_disposition, dtype: int64
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 7585 entries, 0 to 9563
Data columns (total 18 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   orbital_period                  7585 non-null   float64
 1   transit_duration                7585 non-null   float64
 2   transit_depth                   7326 non-null   float64
 3   inclination                     7325 non-null   float64
 4   max_multi_event_stat            6904 non-null   float64
 5   transit_SNR                     7326 non-null   float64
 6   n_planets                       7585 non-null   int64
 7   n_transits                      6904 non-null   float64
 8   planet_num                      7299 non-null   float64
 9   odd-even_depth_comparison_stat  6675 non-null   float64
 10  stellar_eff_temp                7326 non-null   float64
 11  stellar_metallicity             7306 non-null   float64
 12  stellar_radius                  7326 non-null   float64
 13  stellar_mass                    7326 non-null   float64
 14  ra                              7585 non-null   float64
 15  dec                             7585 non-null   float64
 16  kepler_band                     7584 non-null   float64
 17  target                          7585 non-null   int64
dtypes: float64(16), int64(2)
memory usage: 1.1 MB
```

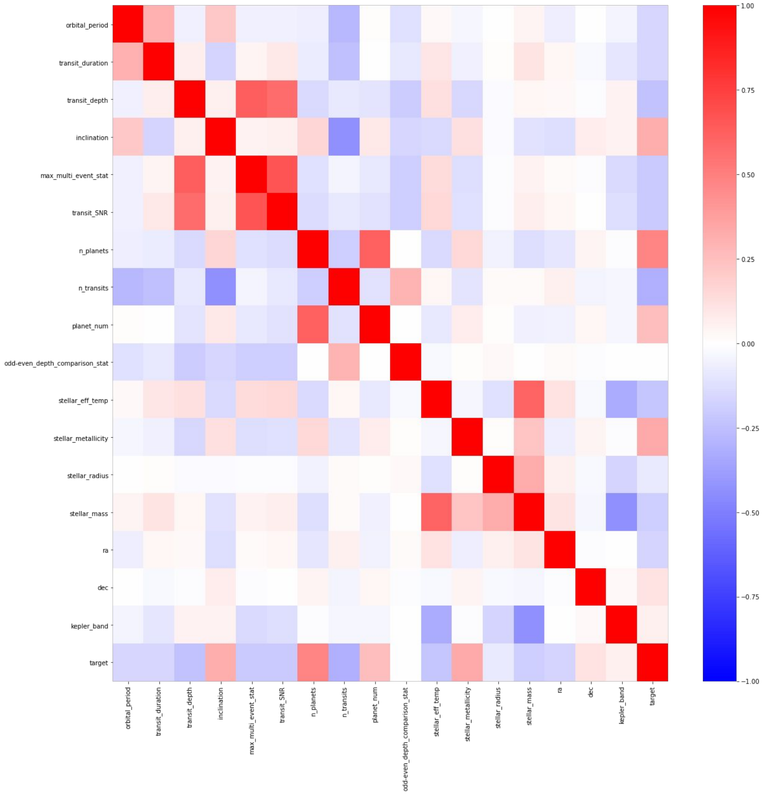
Exploratory Data Analysis

- Plotted distributions of each variable using histograms.
- Some variables are normally distributed, others exponentially distributed.
- One variable codes missing data as -1, treated same as null



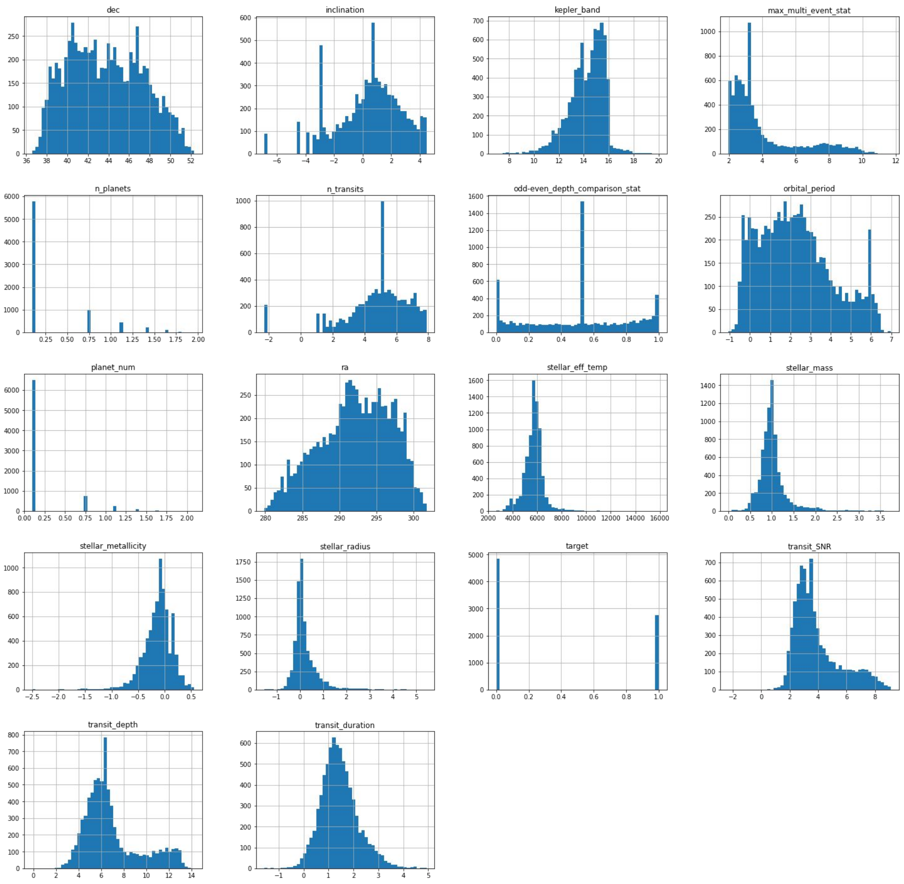
Exploratory Data Analysis

- Plotted correlation matrix using Seaborn heatmap function.
- Not many strong correlations between variables.
- Statistical significance variable strongly correlated with magnitude of signal - perhaps redundant
- Star mass and temperature strongly correlated



Data Preprocessing

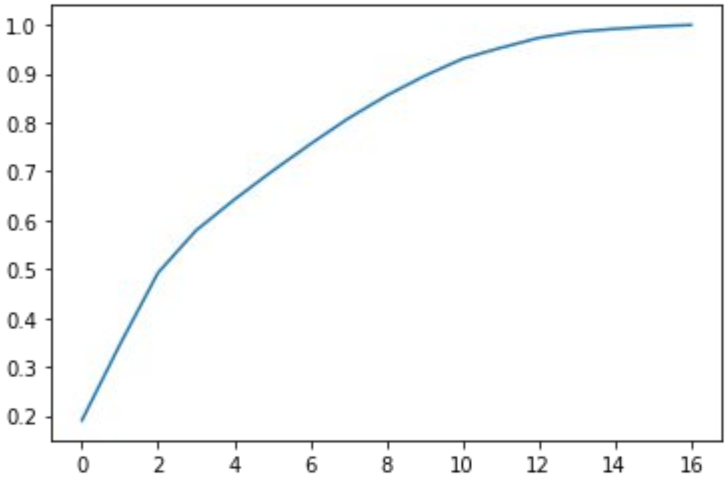
- Normally-distributed variables transformed by Z-score.
- Log transformation for exponentially-distributed variables, followed by normalization.
- Principal Component Analysis (PCA) further transformed data

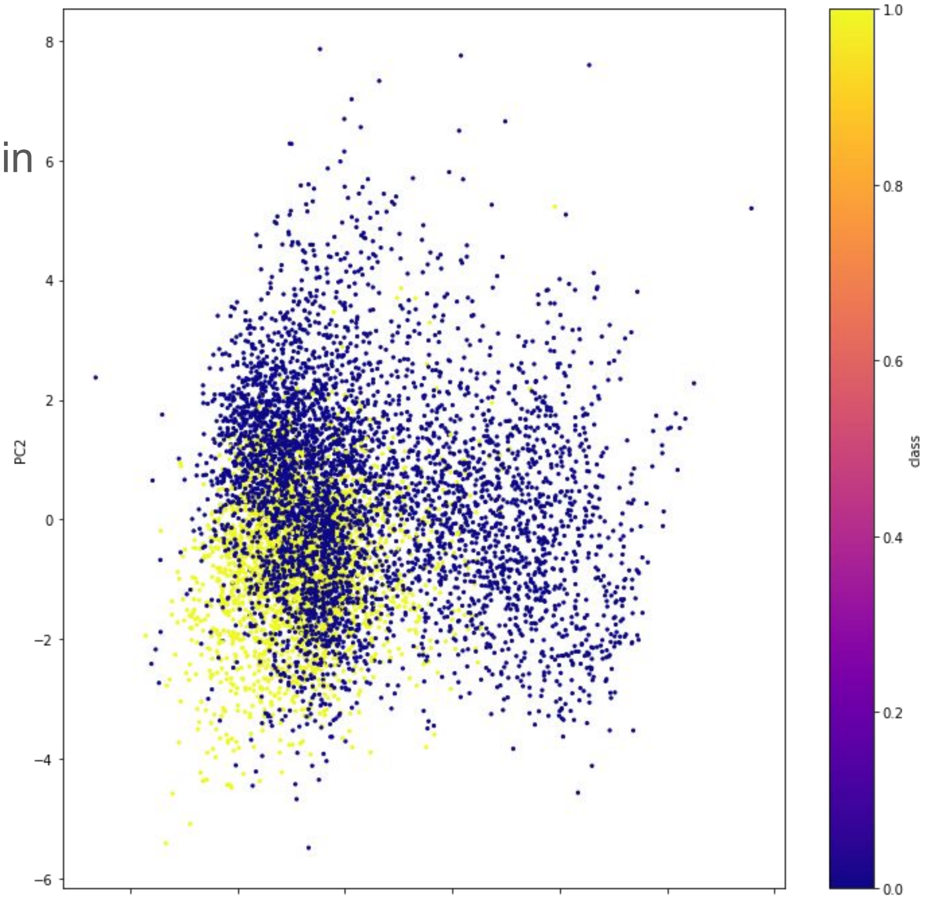


Exploratory Data Analysis

- Scatterplot of first two PCA components.
- True exoplanets in yellow, false positives in blue.
- 49% of explained variance

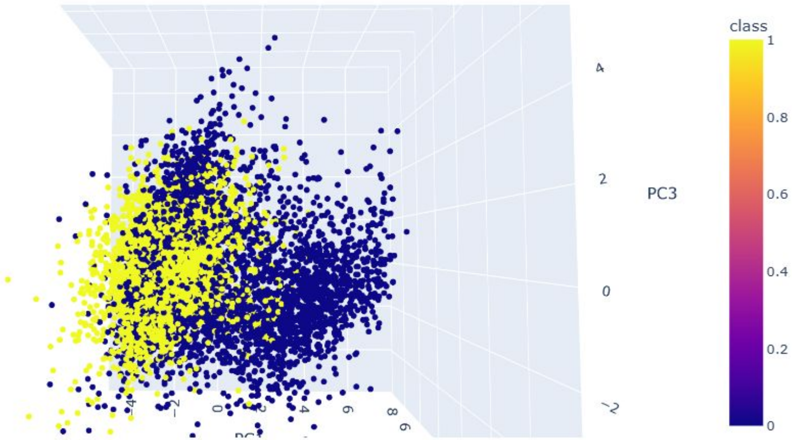
Plot of explained variance vs. number of components:

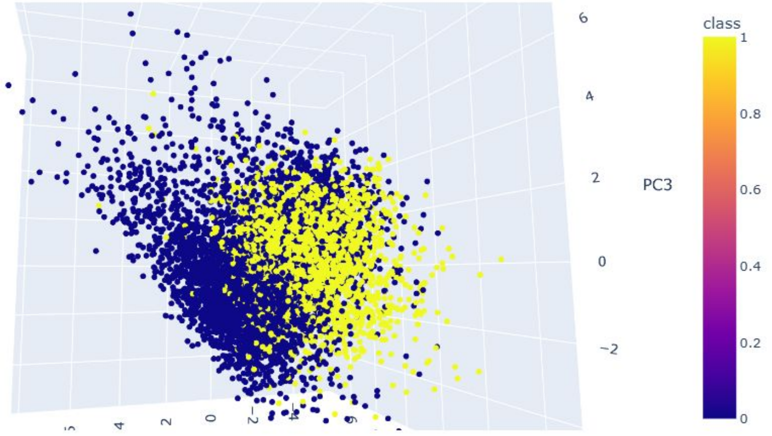




Exploratory Data Analysis

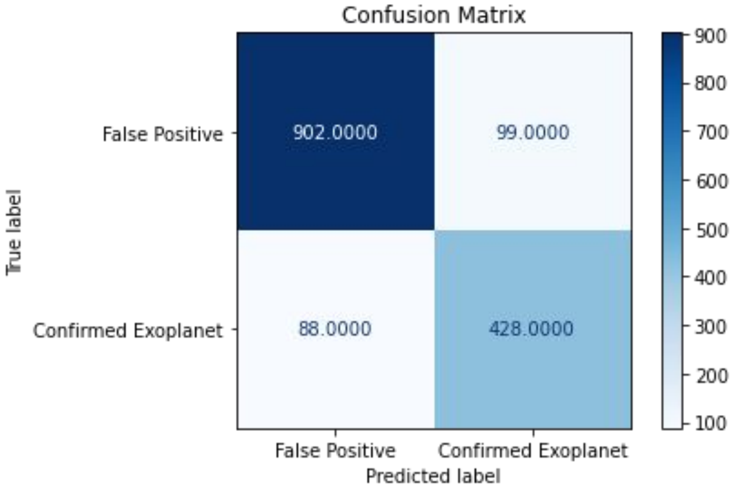
- 3-D Plotly Scatterplot of first three PCA components.
- 58% of explained variance
- Less overlap than 2-D plot
- Looks promising for clustering model.





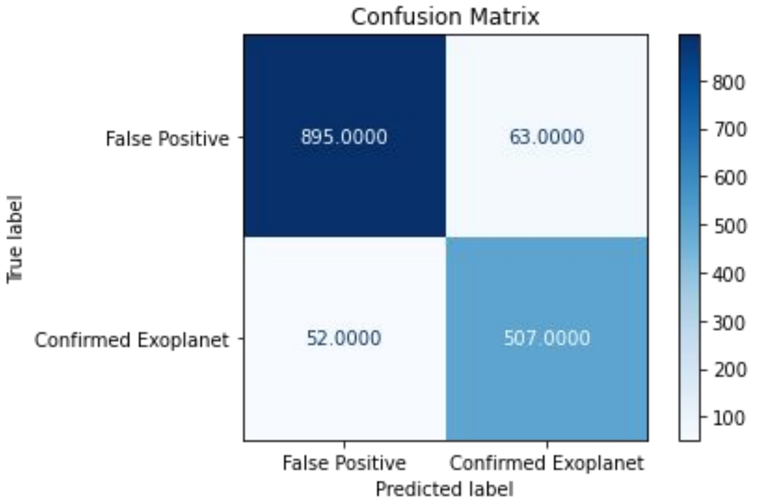
Data Modeling - Decision Tree

- Split data into training set and testing set with 80%-20% split.
- Used non-transformed data, since decision tree uses only thresholds.
- Trained a baseline decision tree using SciKit-learn library with default parameters
  - No limit for number of variables or leaves in the tree
- Achieved baseline testing accuracy of 88%.
- Precision (accuracy on predicted positives) and recall (percentage of true positives identified) were lower, at 81% and 83%, respectively.



Data Modeling - Decision Tree

- Optimized decision tree model with SciKit-learn GridSearchCV() function
  - Hyperparameter sweep with 5-fold cross-validation
  - Optimized for ROC-AUC score to try to improve recall
- Accuracy: 92.4%
- Precision: 88.9%
- Recall: 90.7%

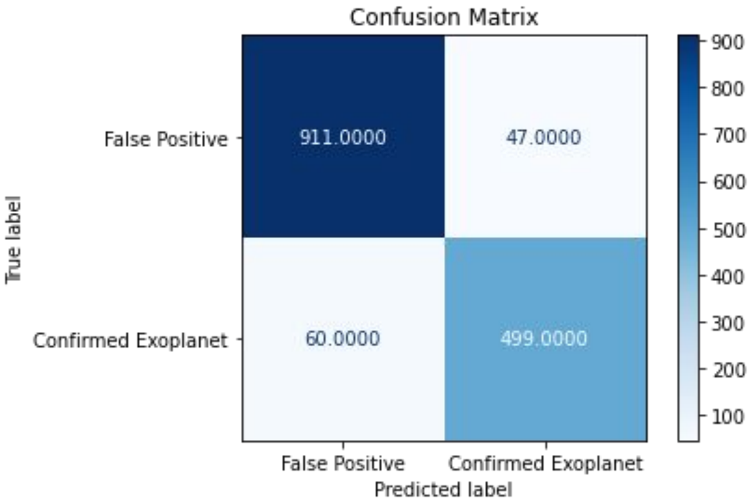



```
print("Best Parameters:", grid_search.best_params_)
print("Best Cross-Validation Score:", grid_search.best_score_)
```

```
Best Parameters: {'criterion': 'entropy', 'max_depth': 10, 'min_samples_leaf': 4, 'min_samples_split': 10}
Best Cross-Validation Score: 0.9374086565335495
```

Data Modeling - Random Forest

- Optimized with GridSearchCV()
  - Best AUC: ensemble of 200 decision trees randomly assigned 4 features each
- Required 15 minutes to run grid search, vs. 30 seconds for decision tree
- Accuracy: 92.9%
- Precision: 91.4%
- Recall: 89.3%

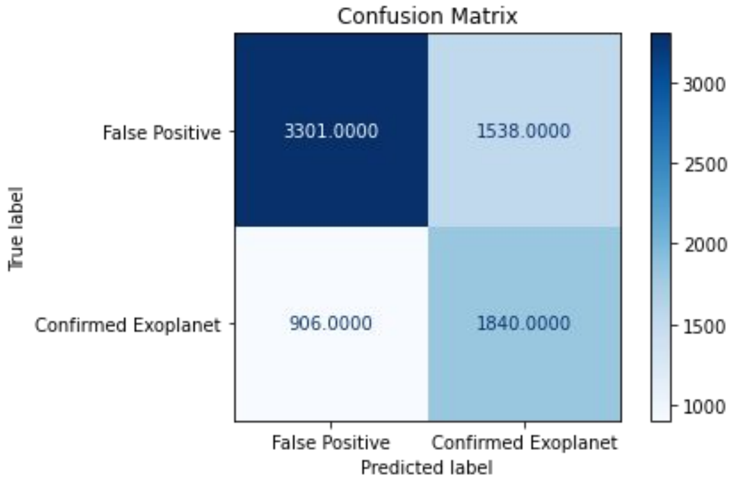



```
print("Best Parameters:", grid_search.best_params_)
print("Best Cross-Validation Score:", grid_search.best_score_)
```

```
Best Parameters: {'bootstrap': True, 'max_features': 'sqrt', 'min_samples_leaf': 2, 'min_samples_spli
t': 5, 'n_estimators': 200}
Best Cross-Validation Score: 0.9730332920978869
```

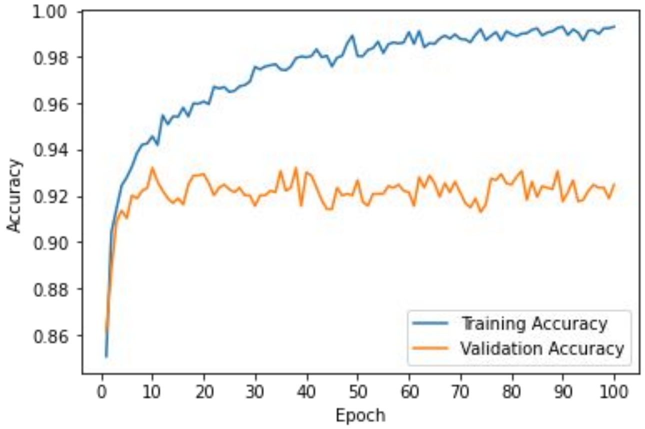
Data Modeling - Agglomerative Clustering

- Unsupervised learning model - no train-test split
- Used PCA-transformed data.
- Performed parameter sweep of distance metrics, cluster-linkage metrics, and number of PCA components included.
- Best model used 8 PCA components
    - About 85% of explained variance
- Only 68% accurate



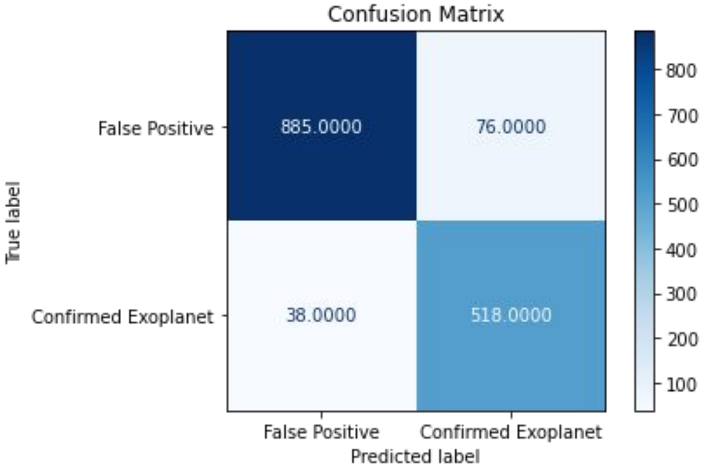
Data Modeling - Dense Neural Network

- Used all components of PCA-transformed data.
- 3 dense layers plus dropout layer
- Accuracy: 92.5%
- Precision: 87.2%
- Recall: 93.2%




```
model = Sequential([
    Dense(256, activation='relu', input_shape=(17,)),
    BatchNormalization(),
    Dense(128, activation='relu'),
    BatchNormalization(),
    Dense(64, activation='relu'),
    BatchNormalization(),
    Dropout(0.3),
    Dense(1, activation='sigmoid')
])
```

```
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
```

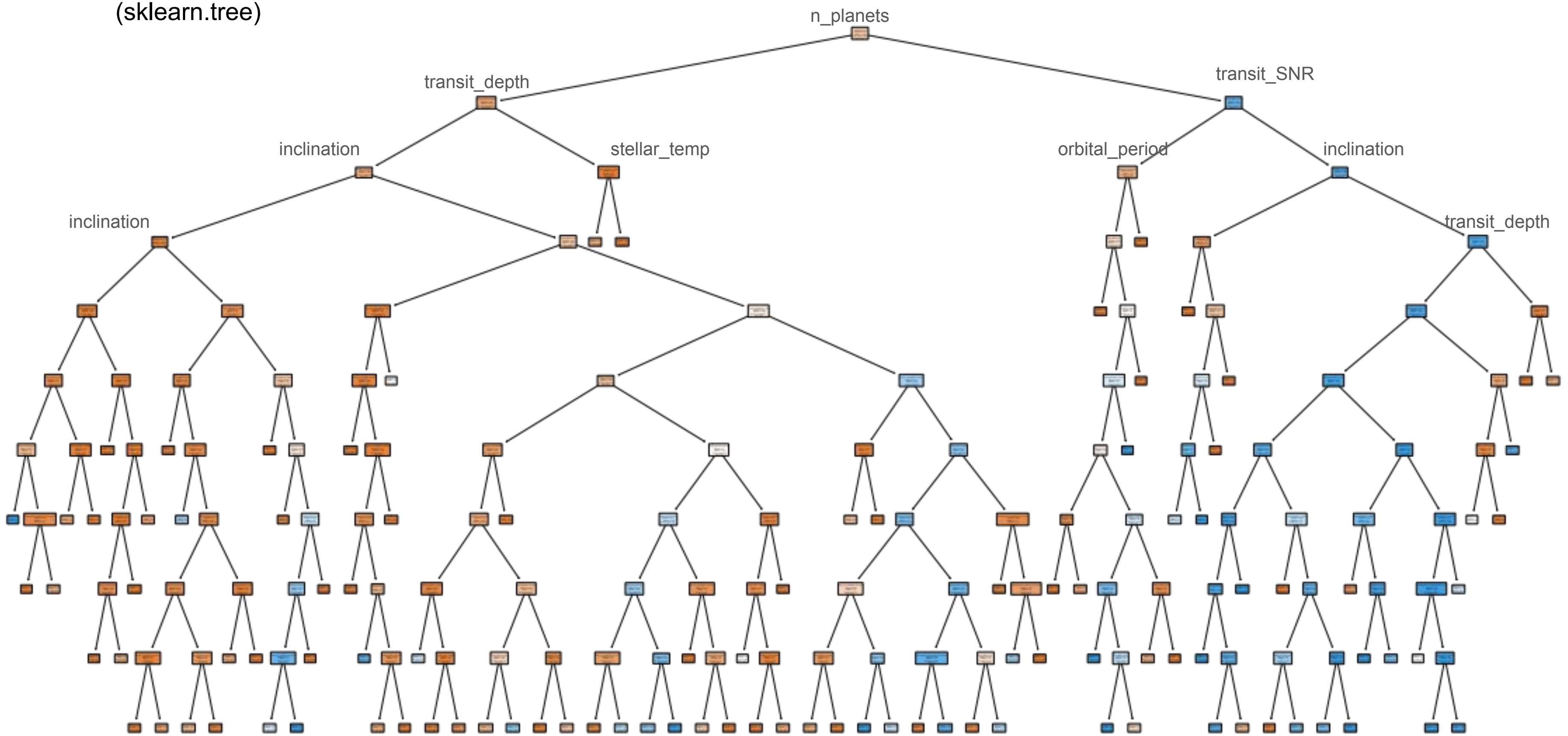


Evaluation Summary

Model	Accuracy	Precision	Recall	F1 Score
Decision Tree	92.4%	88.9%	90.7%	0.898
Random Forest	92.9%	91.4%	89.3%	0.903
Agglomerative Clustering	67.8%	54.5%	67.0%	0.601
Dense Neural Network	92.5%	87.2%	93.2%	0.901

- Supervised models (Decision Tree, Random Forest, Dense Neural Network) greatly outperformed the unsupervised model (Agglomerative Clustering).
- A single optimized decision tree performed just about as well as more powerful methods.
- The interpretability of the decision tree model makes it the ideal choice for this project.

Decision Tree Visualization
(sklearn.tree)



Project Timeline

- Phase 1 - Data Cleaning and Understanding, including EDA Visualizations (2 days)
- Phase 2 - Data Preprocessing and Feature Selection (1 day)
- Phase 3 - Baseline Model Development (1 day)
- Phase 4 - Model Evaluation and Hyperparameter Tuning (2 days)
- Phase 5 - Final Presentation of Results (1 day)

Future Work

- Try to understand why the clustering approach didn't work.
    - Try different feature selection, data transformation, feature engineering
- Optimize a model for subset of data with no other discovered planets in the star system
    - Number of planets in the system is highly predictive - could be a limitation of the model

Key Takeaways

- The project succeeded in building an accurate, efficient, interpretable classifier for detecting exoplanets from the NASA Exoplanet Archive's KOI data.
- Data Understanding phase proved to be crucial.
- The feature space ended up being smaller and more numerical than expected.
    - Single decision tree model was sufficient for this feature space.
- Evaluation plan shift - less emphasis on accuracy due to class imbalance.
    - High recall ensures potential exoplanets don't often get missed.