Kepler Data Analysis

Richard Anderson

Fork updates:
https://github.com/Rander417/KeplerExoplanet
- *New presentation*
- *Raw data pulled from CSV initially*
- *Pickled key data sets*
- *Refactored all notebooks*

Forked from team project:
https://github.com/tom-jj-G/KeplerExoplanets

Columbia University – Fu School of Engineering
- *Data Analytics 6mo Program*

My role was building the ML pipeline
- *Data Cleaning*
- *EDA*
- *Preprocessing*
- *Building the ML models*

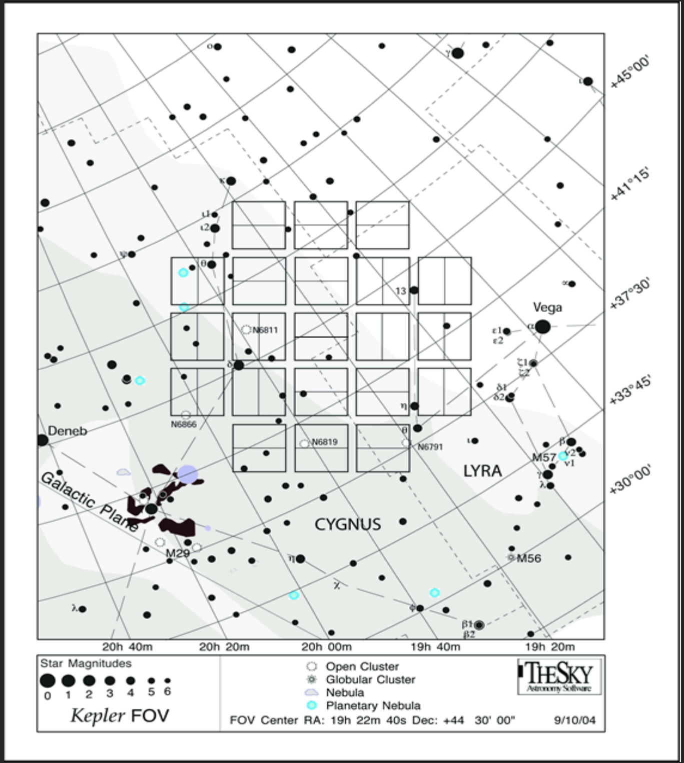
Teammates:
Damien Corr, Priscilla Lin, Tom Greff

Kepler Mission Overview

The Kepler Telescope photometer consists of 21 CCD modules, each with two 2200x1024 pixel CCDs for a grand total of 94.6 million active pixels.

Source - https://keplerscience.arc.nasa.gov/the-kepler-space-telescope.html

https://www.nasa.gov/kepler/faq

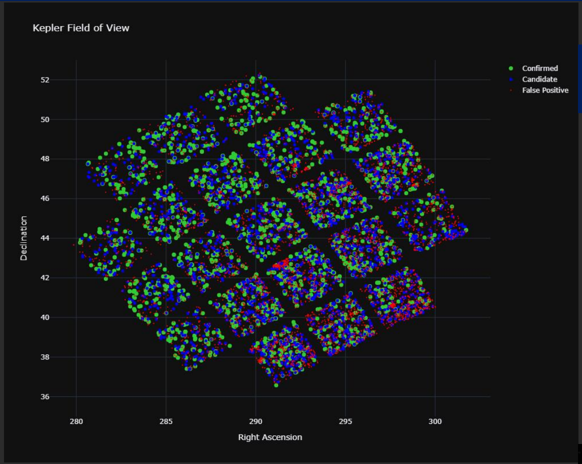


Kepler Field of View (FOV)

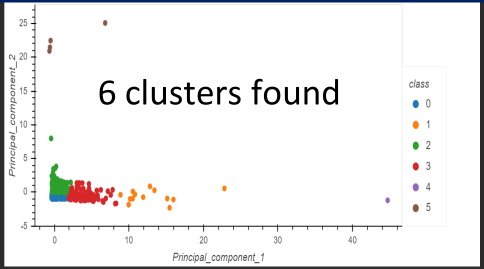
https://www.nasa.gov/mission_pages/kepler/overview/index.html

4 Big Questions

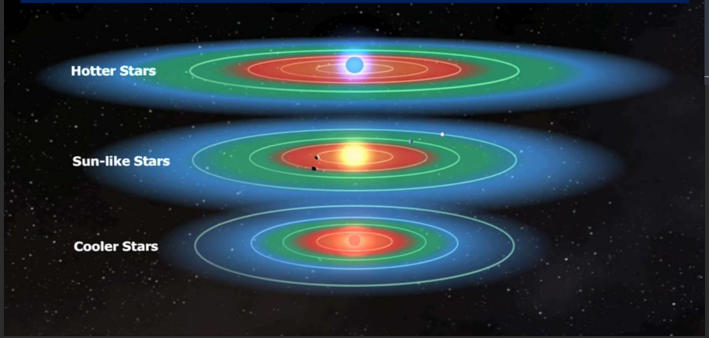
Is the Object of Interest an Exoplanet?



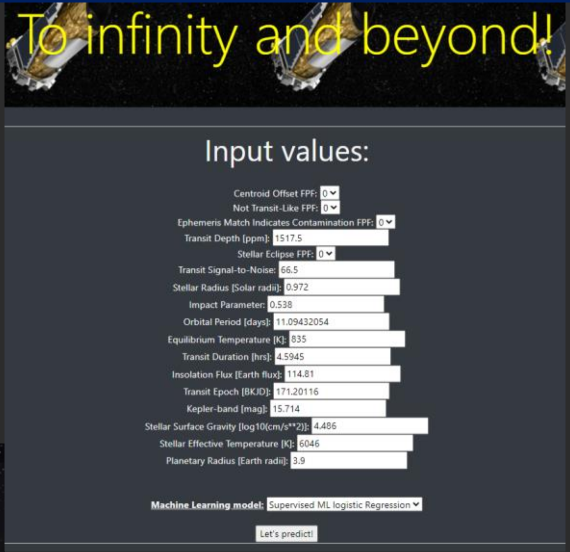
Does EDA reveal interesting groupings?



Is the exoplanet in the habitable zone?



Can future observers use our models?



https://kepler-groupa.herokuapp.com/

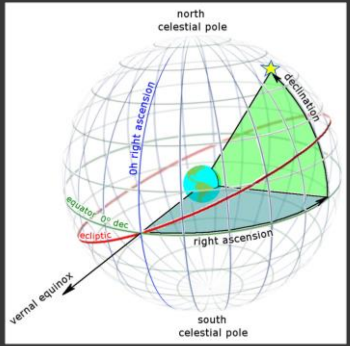
Domain knowledge

CHALLENGES

- Astrophysics terminology
- Cryptic acronyms & abbreviations
  - Koi_tce_delivname, koi_fpflag_nt...
- Dense reference material
  - 382 pages!

KEPLER
DATA PROCESSING
HANDBOOK
KSCI-19081-003

CELESTIAL COORDINATES

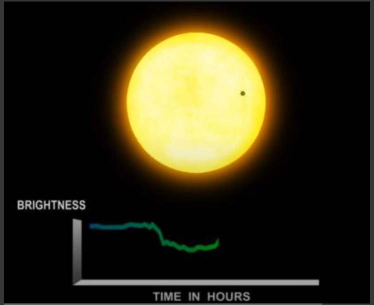


Declination corresponds to latitude & ***Right Ascension*** to longitude

https://skyandtelescope.org/astronomy-resources/right-ascension-declination-celestial-coordinates/

TRANSIT

When one object crosses in front of another in space

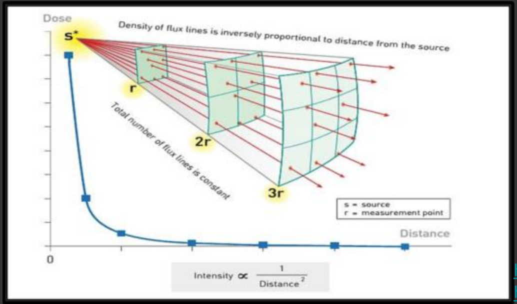


Transits by terrestrial planets produce a small change in a star's brightness of about 1/10,000 (100 parts per million, ppm), lasting for 2 to 16 hours.

https://exoplanets.nasa.gov/resources/1022/kepler-transit-graph/

FLUX

A star's apparent brightness



https://bit.ly/2HBZvM3
https://bit.ly/34yWwNi

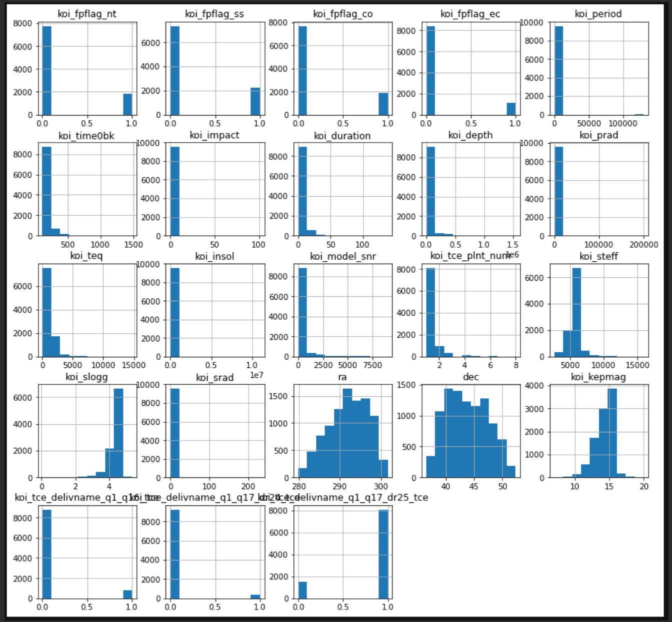
EDA & Preprocessing

Generally clean combination of numerical and categorical

Large number of nulls

Two Ys?!?

Unbalanced




```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9564 entries, 0 to 9563
Data columns (total 50 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   rowid               9564 non-null   int64
 1   kepid               9564 non-null   int64
 2   kepoi_name          9564 non-null   object
 3   kepler_name         2294 non-null   object
 4   koi_disposition     9564 non-null   object
 5   koi_pdisposition    9564 non-null   object
 6   koi_score           8054 non-null   float64
 7   koi_fpflag_nt       9564 non-null   int64
 8   koi_fpflag_ss       9564 non-null   int64
 9   koi_fpflag_co       9564 non-null   int64
 10  koi_fpflag_ec       9564 non-null   int64
 11  koi_period          9564 non-null   float64
 12  koi_period_err1     9110 non-null   float64
 13  koi_period_err2     9110 non-null   float64
 14  koi_time0bk         9564 non-null   float64
 15  koi_time0bk_err1    9110 non-null   float64
 16  koi_time0bk_err2    9110 non-null   float64
 17  koi_impact          9201 non-null   float64
 18  koi_impact_err1     9110 non-null   float64
 19  koi_impact_err2     9110 non-null   float64
 20  koi_duration        9564 non-null   float64
 21  koi_duration_err1   9110 non-null   float64
 22  koi_duration_err2   9110 non-null   float64
 23  koi_depth           9201 non-null   float64
 24  koi_depth_err1      9110 non-null   float64
 25  koi_depth_err2      9110 non-null   float64
 26  koi_prad            9201 non-null   float64
 27  koi_prad_err1       9201 non-null   float64
 28  koi_prad_err2       9201 non-null   float64
 29  koi_teq             9201 non-null   float64
 30  koi_teq_err1        0 non-null      float64
 31  koi_teq_err2        0 non-null      float64
 32  koi_insol           9243 non-null   float64
 33  koi_insol_err1      9243 non-null   float64
 34  koi_insol_err2      9243 non-null   float64
 35  koi_model_snr       9201 non-null   float64
 36  koi_tce_plnt_num    9218 non-null   float64
 37  koi_tce_delivname   9218 non-null   object
 38  koi_steff           9201 non-null   float64
 39  koi_steff_err1      9096 non-null   float64
 40  koi_steff_err2      9081 non-null   float64
 41  koi_slogg           9201 non-null   float64
 42  koi_slogg_err1      9096 non-null   float64
 43  koi_slogg_err2      9096 non-null   float64
 44  koi_srad            9201 non-null   float64
 45  koi_srad_err1       9096 non-null   float64
 46  koi_srad_err2       9096 non-null   float64
 47  ra                  9564 non-null   float64
 48  dec                 9564 non-null   float64
 49  koi_kepmag          9563 non-null   float64
dtypes: float64(39), int64(6), object(5)
memory usage: 3.6+ MB
```

Handling Null Values

40k+ Null cells across 10k rows & 50 columns of data (500k cells)

363 rows with nulls after cleaning (including dropping +/- error columns)

We decided to drop the nulls due to their small volume & results of imputing

```
kelperRAW_df.isnull().sum().sum()
```

40557

```
keplerProcessed_df.isnull().sum().sum()
```

3572

Exoplanet_Archive_Disposition	0
Not_Transit-Like_FPF	0
Stellar_Eclipse_FPF	0
Centroid_Offset_FPF	0
Ephemeris_Match_Indicates_Contamination_FPF	0
Orbital_Period_[days]	0
Transit_Epoch_[BKJD]	0
Impact_Parameter	363
Transit_Duration_[hrs]	0
Transit_Depth_[ppm]	363
Planetary_Radius_[Earth radii]	363
Equilibrium_Temperature_[K]	363
Insolation_Flux_[Earth flux]	321
Transit_Signal-to-Noise	363
TCE_Planet_Number	346
koi_steff	363
koi_slogg	363
koi_srad	363
right_ascension	0
declination	0
Kepler_band [mag]	1
TCE_Delivery_q1_q16_tce	0
TCE_Delivery_q1_q17_dr24_tce	0
TCE_Delivery_q1_q17_dr25_tce	0
dtype: int64	

Impute methods evaluated:

Mean

```
# Impute NaNs via Mean
imputer_mean = SimpleImputer(missing_values=np.nan, strategy='mean')

keplerProcesse
keplerProcesse
```

Median

```
# Impute NaNs via Median
imputer_median = SimpleImputer(missing_values=np.nan, strategy='median')

keplerProcessedMedianImpute_df = kep
keplerProcessedMedianImpute_df.iloc[
```

Mode

```
# Impute NaNs via Mode
imputer_mode = SimpleImputer(missing_values=np.nan, strategy='most_frequent')

keplerProcessedModeImpute_df = keplerProcessed_df.copy(deep=True)
keplerProcessedModeImpute_df.iloc[:,:] = imputer_mode.fit_transform(keplerProcessedMeanImpute_df)
```

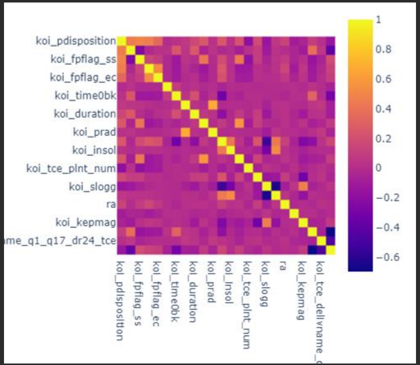
Mode had a negative f1 impact while Mean & Median had no discernable impact

Feature Analysis & A Tale of Two Ys

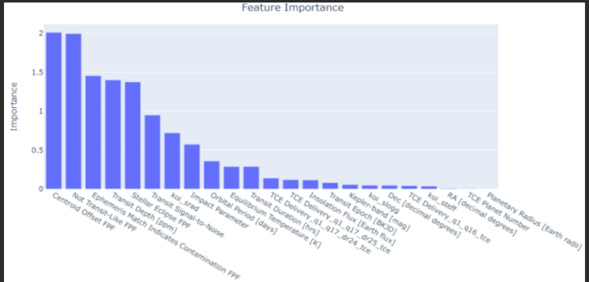
- **Which target?** Kepler Disposition vs Exoplanet Archive Disposition (EAD)
  - **We chose EAD as it was the result of the most recent NASA analysis**
  - ML models had a 99% f1 with kepler disposition

- **We analyzed our features using the methods below**
  - "Sequential Feature Selection” is part of the mlxtend library - programmatically analyzes feature combinations

Feature Correlation



Feature Importance



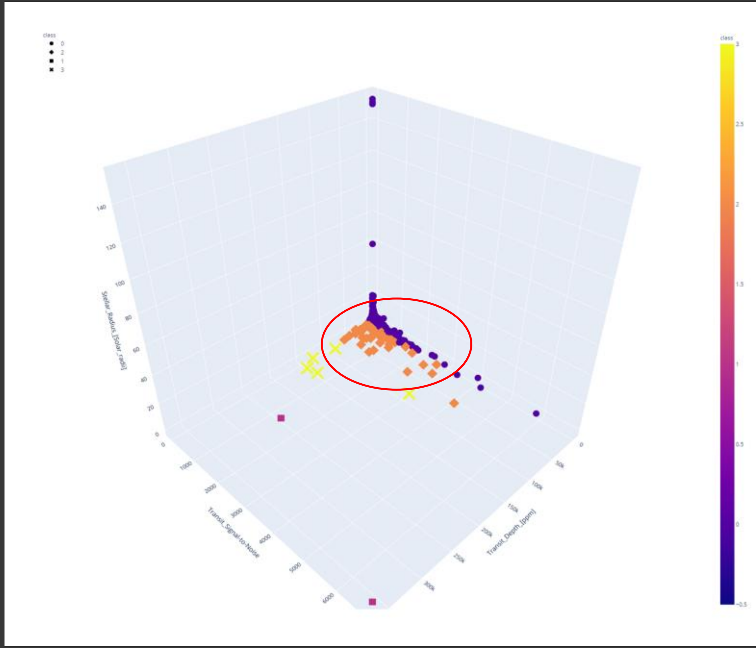
Sequential Feature Selection

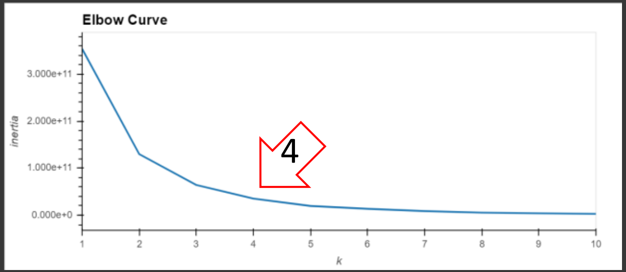
```
8: {'feature_idx': (0, 1, 2, 3, 9, 20, 21, 22),
 'cv_scores': array([-0.01962382, -0.01745037, -0.00757576, -0.00960293, -0.00605649,
        -0.01017782, -0.01898536, -0.01070084, -0.00808577, -0.00282427]),
 'avg_score': -0.011108341749627281,
 'feature_names': ('koi_fpflag_nt',
  'koi_fpflag_ss',
  'koi_fpflag_co',
  'koi_fpflag_ec',
  'koi_prad',
  'koi_tce_delivname_q1_q16_tce',
  'koi_tce_delivname_q1_q17_dr24_tce',
  'koi_tce_delivname_q1_q17_dr25_tce')},
```

Future options

- *Variance Inflation Factor*
- *Multicollinearity*
- *Lasso regression*

K-Means Clustering

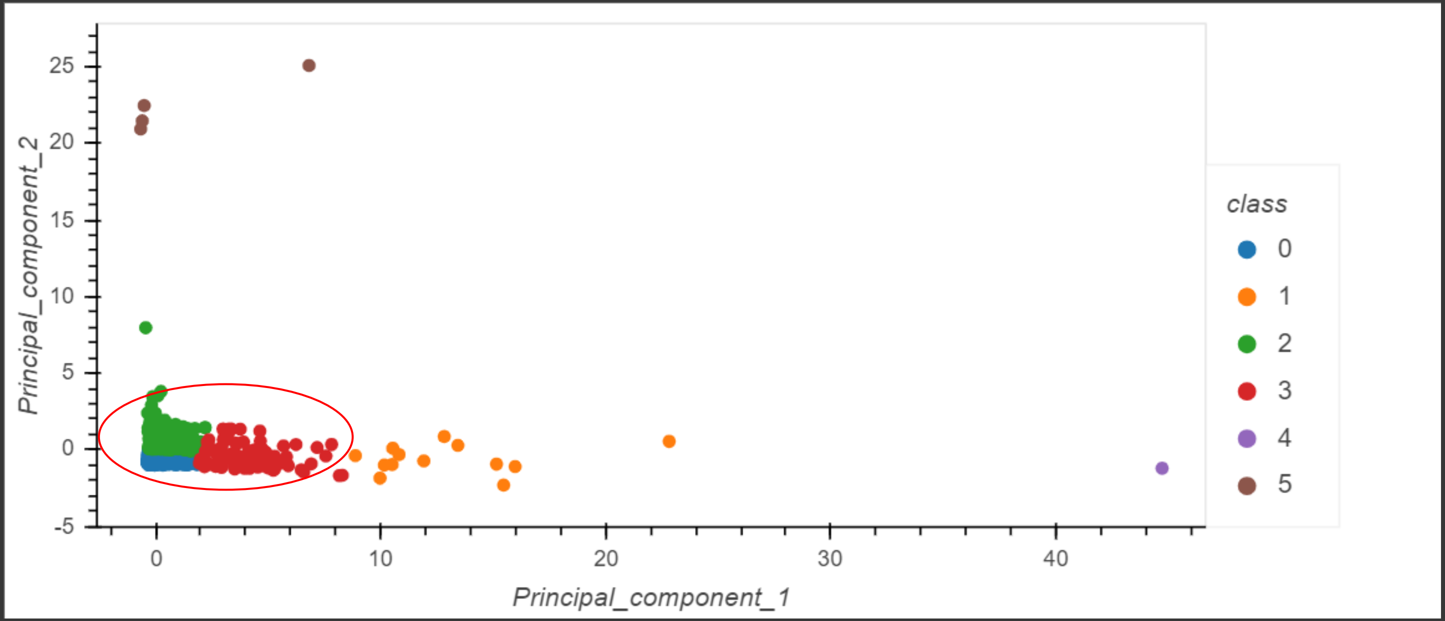




⇒ For the three most important features, the major part of the object has a low transit depth (<20,000 parts/million), transit signal-to-lose (<2000) and stellar radius (<20 solar radii)

⇒ Object of interest are gathered for these important features

K-Means Clustering w/PCA



⇒ 6 clusters after using *a new* elbow curve

⇒ 36% of the information is lost when the four-dimension data were reduced to a two one

⇒ Confirmation that most of the data is consistent/homogeneous

Supervised Machine Learning

Logistic Regression - 83% f1

- Chosen since our questions are categorical
- Weaker results due to unbalanced data

	precision	recall	f1-score	support
0	0.68	0.57	0.62	534
1	0.66	0.74	0.69	572
2	0.98	1.00	0.99	1131
accuracy			0.83	2237
macro avg	0.77	0.77	0.77	2237
weighted avg	0.83	0.83	0.83	2237

Random Forest - 90% f1

- Alternative to better handle the data imbalance

	pre	rec	spe	f1	geo	iba	sup
0	0.81	0.78	0.94	0.79	0.86	0.72	534
1	0.82	0.81	0.94	0.81	0.87	0.75	572
2	0.98	1.00	0.98	0.99	0.99	0.98	1131
avg / total	0.90	0.90	0.96	0.90	0.93	0.86	2237

Gradient Boosted Trees - 90% f1

- Chosen to better handle the data imbalance

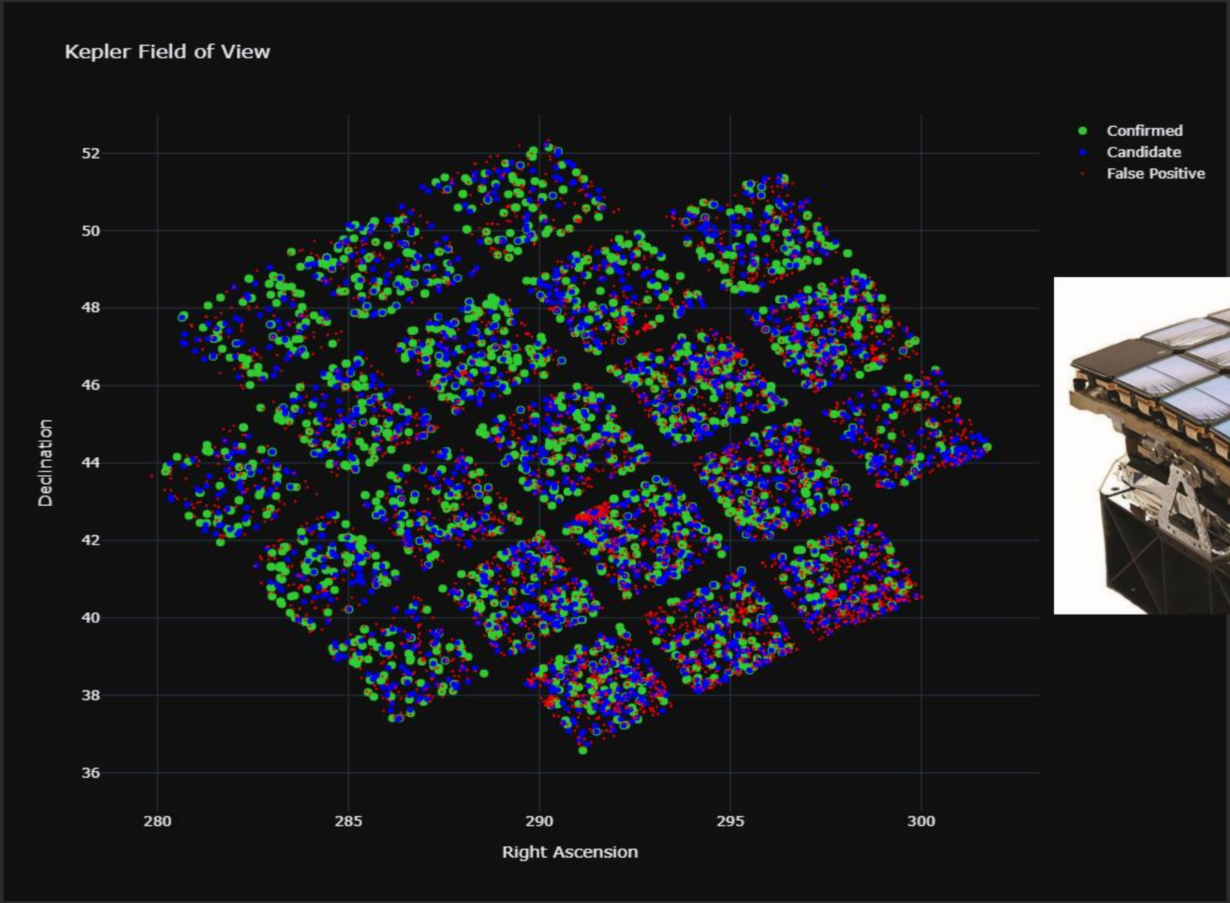
	precision	recall	f1-score	support
0	0.82	0.78	0.80	534
1	0.81	0.83	0.82	572
2	0.98	1.00	0.99	1131
accuracy			0.90	2237
macro avg	0.87	0.87	0.87	2237
weighted avg	0.90	0.90	0.90	2237

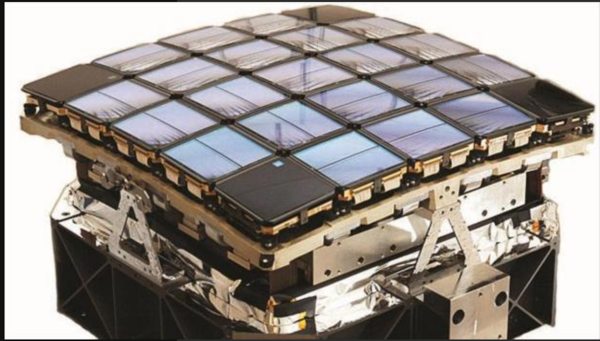
Deep Neural Network - 84% f1

- Uses relu and softmax

	precision	recall	f1-score	support
0	0.67	0.70	0.69	534
1	0.71	0.68	0.70	572
2	0.98	0.99	0.98	1131
accuracy			0.84	2237
macro avg	0.79	0.79	0.79	2237
weighted avg	0.84	0.84	0.84	2237

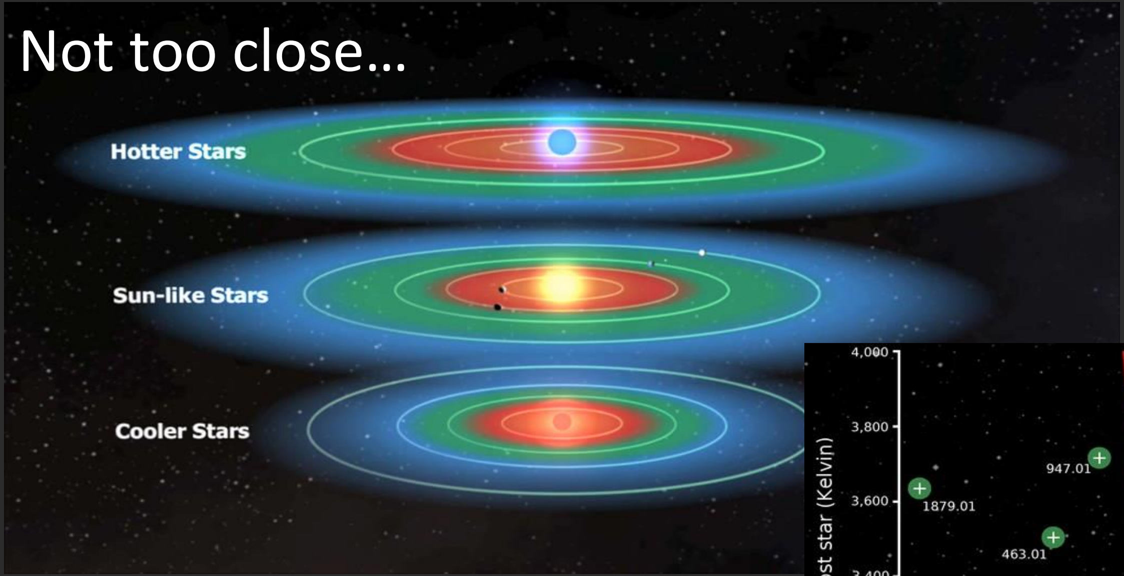
Graphing the results

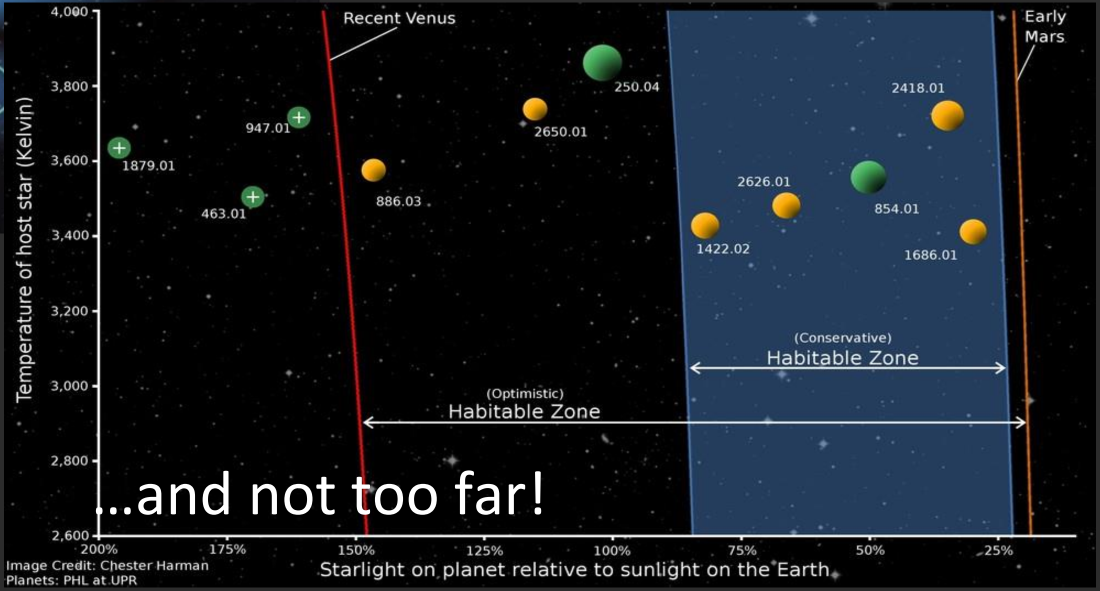




Kepler Array

Habitable or Not? The Goldilocks Zone





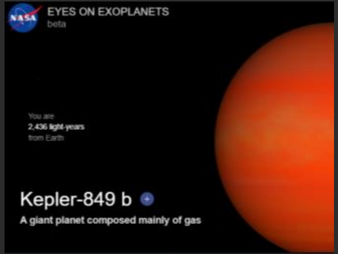
Is the Exoplanet in the Goldilocks Zone?

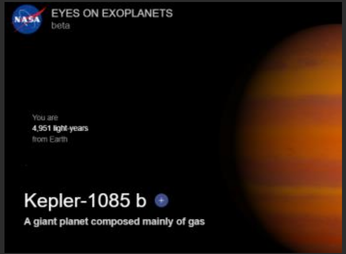
Habitable Criteria:

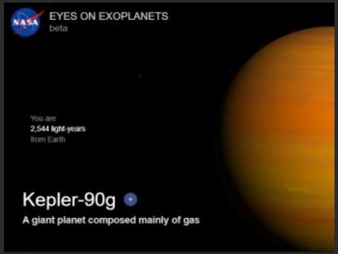
- Orbital_period[days]: 200 ~ 400
- Stellar_effective_temperature > 5500 ~ 6500
- Stellar_radius[solar_radii]: 1 ~ 2
- Stellar_surface_gravity[log10(cm/s**2)]: > 4
- Stellar_metallicity: > 0

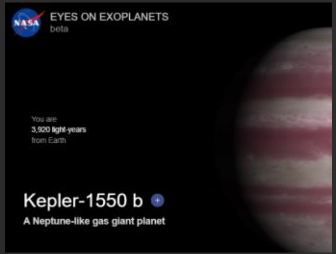
Confirmed & Candidates: **2248**

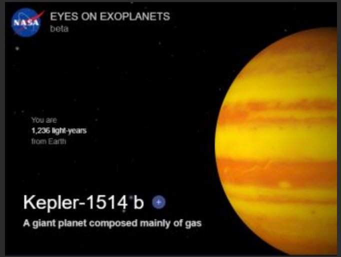
Habitable: **10**

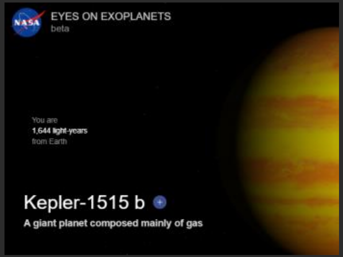












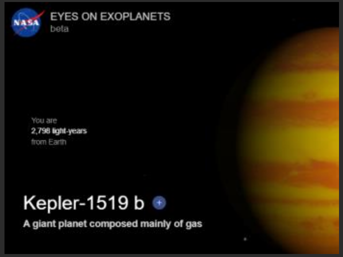


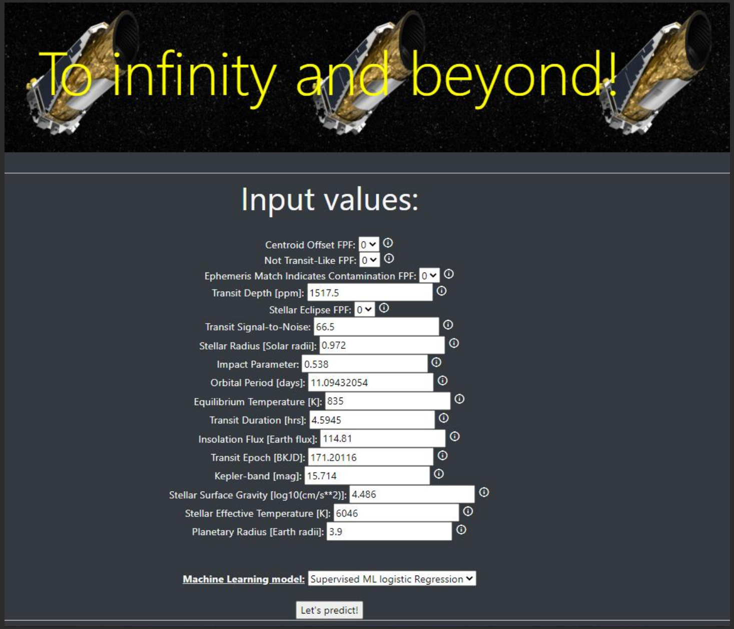
Image sources: https://exoplanets.nasa.gov/exoplanet-catalog/

Prediction web-app

Takes new observations and predicts if the object is an exoplanet using our ML models

- Built with ES6/HTML
- Hosted online

https://kepler-groupa.herokuapp.com/



Technologies

NASA
scikit learn
TensorFlow
plotly
jupyter
SQLAlchemy
seaborn
HEROKU
python™
Amazon RDS
PostgreSQL
Flask
GitHub
kaggle
Google docs
+ab|eau