CS 539: Final Project - Plasticc Challenge

Nathan Hsu, Manasee Godsay, Yao-Chun Hsieh, Wei Zhao

Outline

The Problem

The Large Synoptic Survey Telescope (LSST)

https://www.lsst.org/

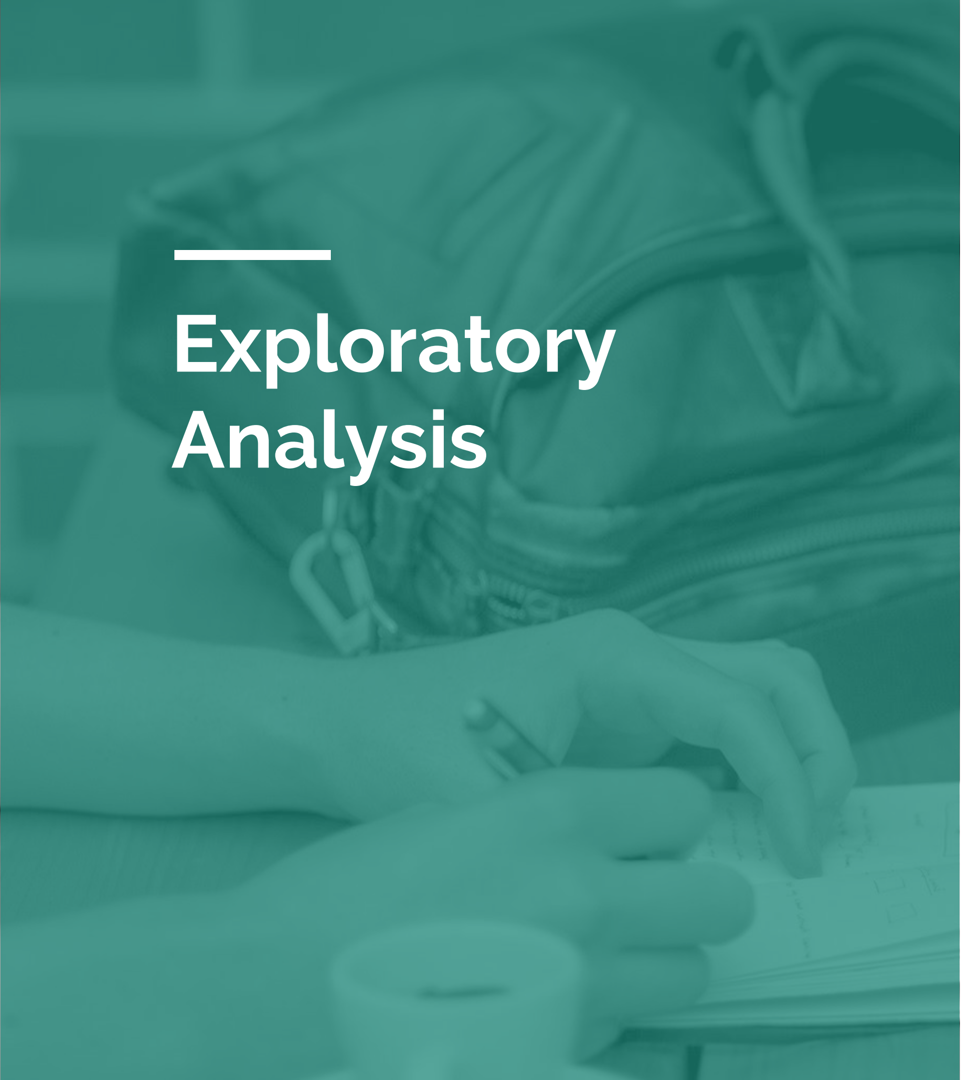
We want to know the status (14 classes) of stars

- Secular, Pulsating and Eruptive Variable Stars
- Tidal Disruption Events
- Kilonovae
- Supernovae of different types
- Active Galactic Nucleii
- Microlensing Events
- Eclipsing Binaries

2 ways to observe stars

Spectroscopy	**Photometry**
Expensive	Much cheaper
Very detailed	Less detailed
Hard to schedule	Going on live in 2019!

Exploratory Analysis



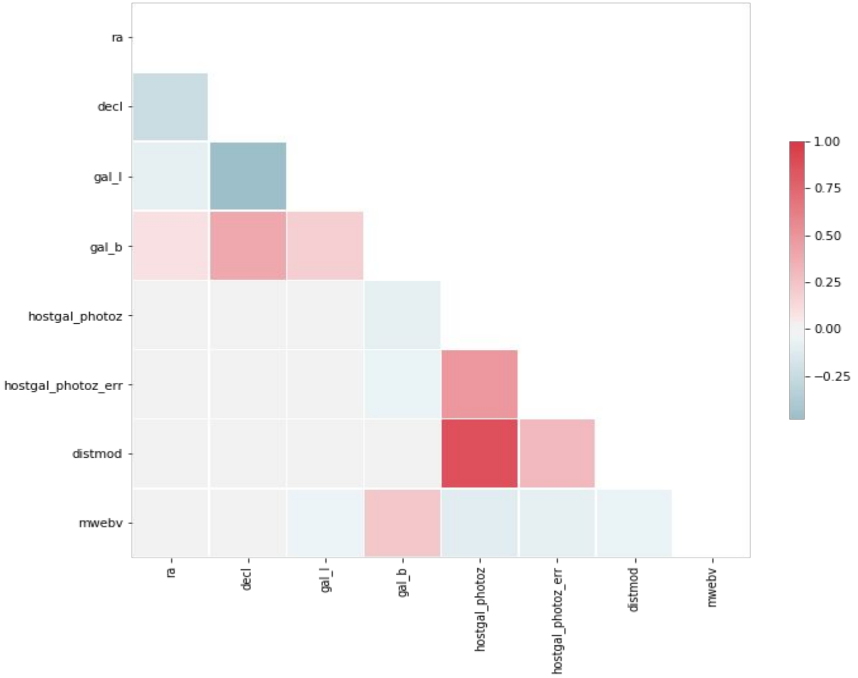
Exploratory Analysis

Trying to understand the data, we tried out looking at the two kinds of data provided:

- Meta Data
- Time Series Data

Pearson Correlation Matrix - Meta Data

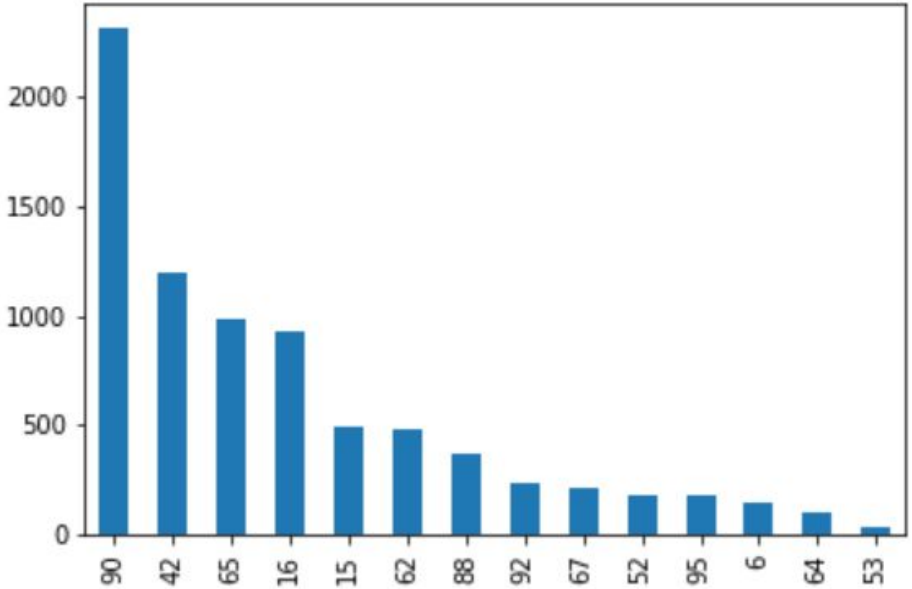
- We see that distmod and hostal_photoz are highly correlated.
- Limitations of Meta Data - Domain knowledge



Some insights..

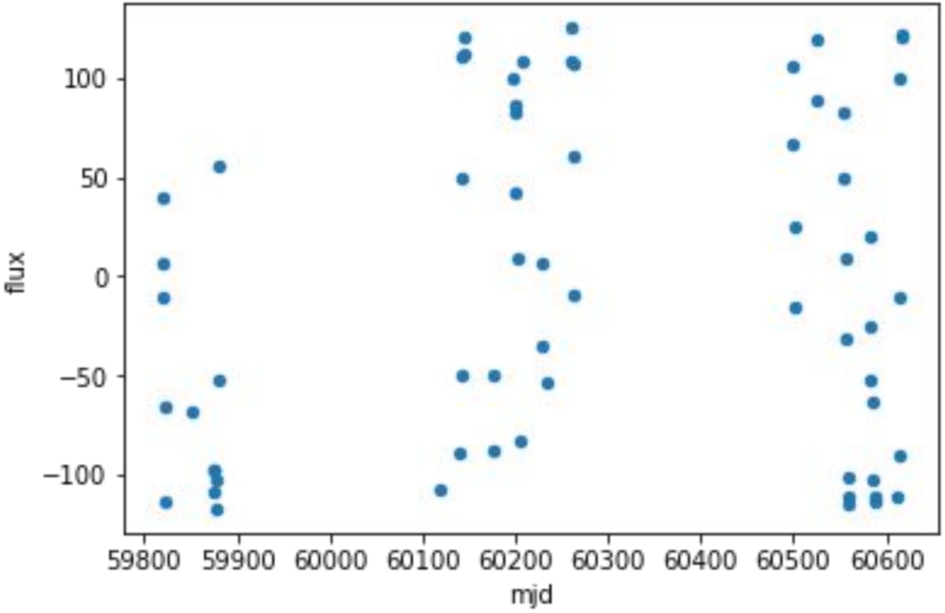
- How many values in each class?
- Unbalanced Data

X-axis: Count , Y-axis: Class ID



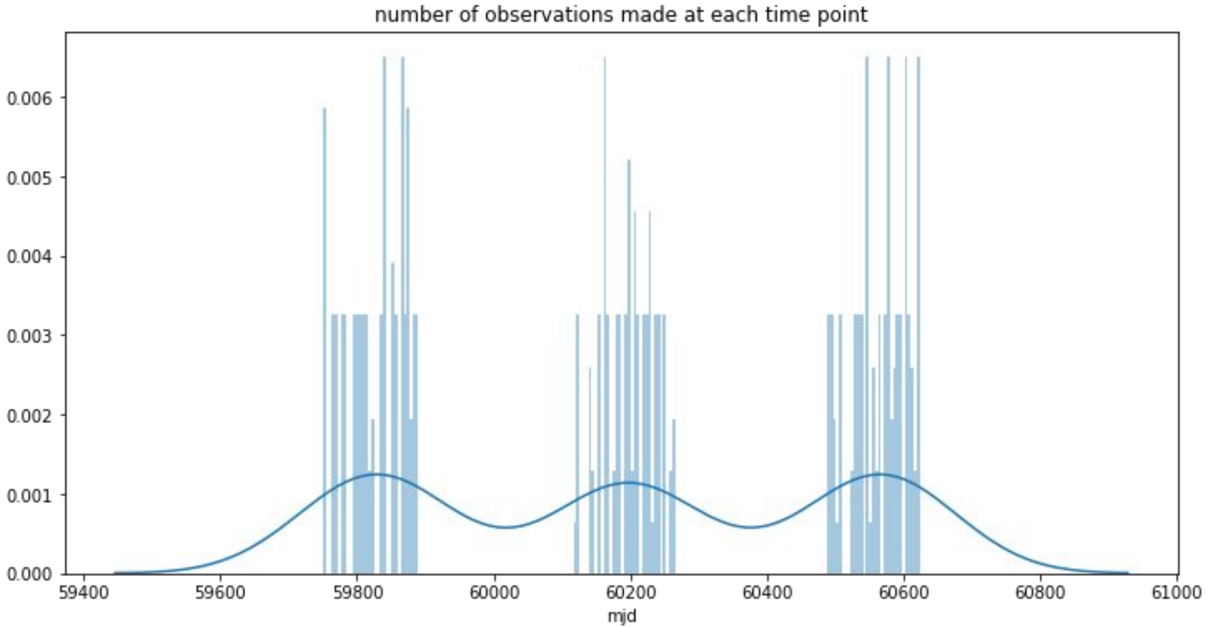
Time-Series Data

- What data do we have?“Time-Series” Flux
- What is Passband?



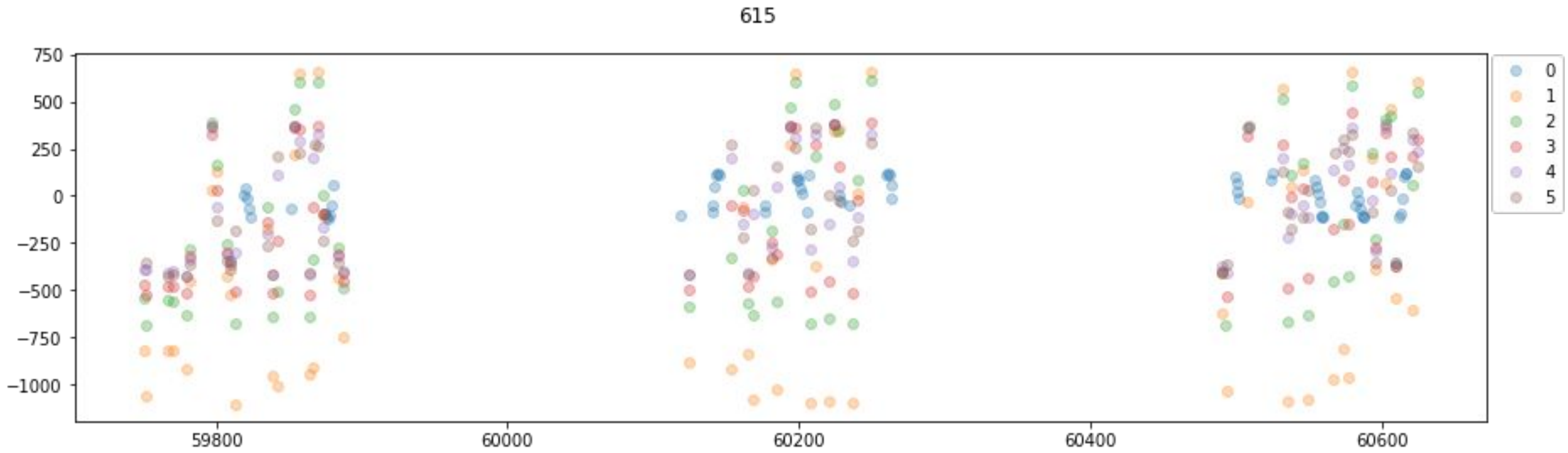
Number of observations at each time point

There is irregularity in the recorded observations.



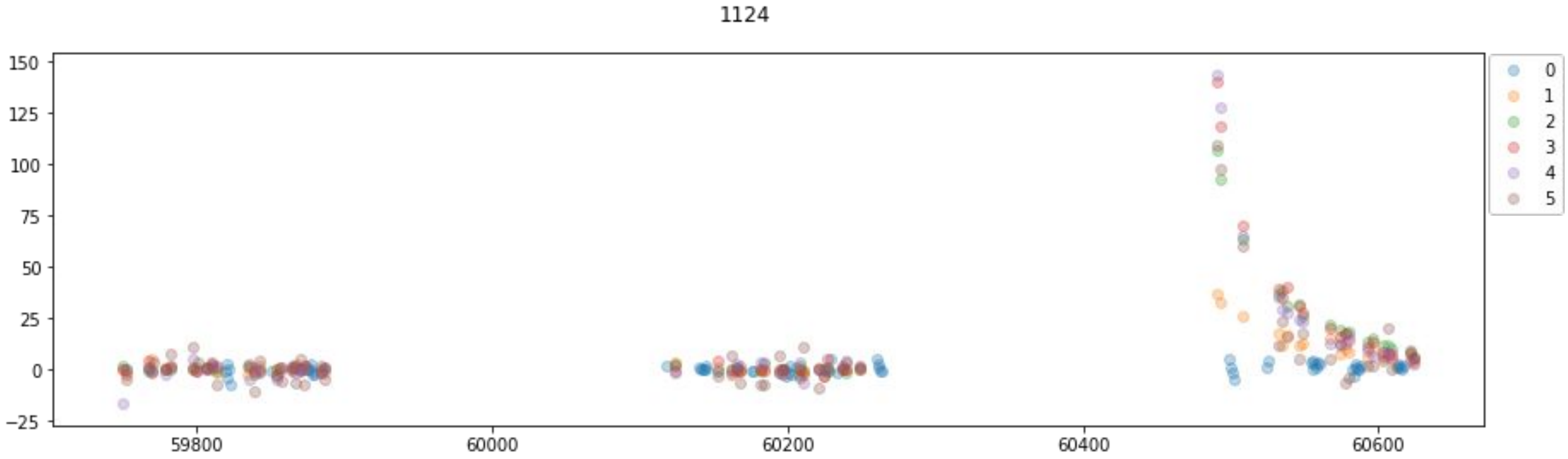
Scatter plots of different objects

For a particular object: 615

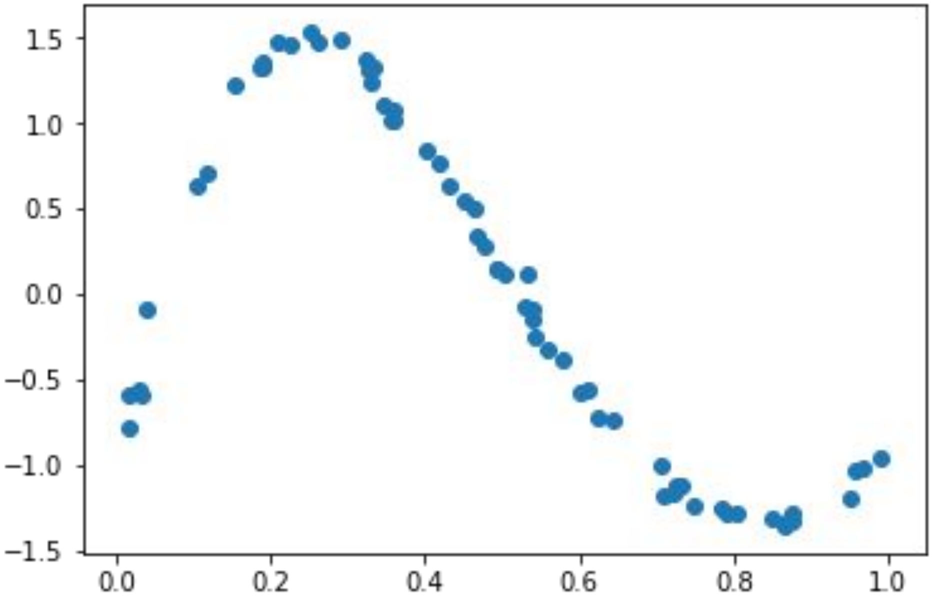


Scatter plots of different objects

For a particular object : 1124



After “Time-smoothing and normalization”



Feature Engineering

Feature Engineering

WHY:

- Get more useful features
- Reduce Computational complexity
- Reduce noise(useless features)
- Improving the model performance

HOW:

- Feature Extraction
  - Statistical Methods
- Feature Normalization
- Feature Selection
  - Select with Feature Importance
- Feature Transformation (D.R.)

Feature Extraction with Statistical Methods

- Using pandas.dataframe.agg()

customer_id	value
1	12
1	17
1	10
2	20
2	22
2	26
2	18
3	16
3	19
3	14

aggs = {‘value’: [‘min’, ‘max’, ‘mean’, ‘std’]}

df.groupby(‘customer_id’).agg(aggs)

customer_id	value_min	value_max	value_mean	value_std
1	10	17	13.0	3.6
2	18	26	21.5	3.4
3	14	19	16.3	2.5

Feature Selection with Feature Importance

- **Removing features with low variance** (dependent variable not needed)
- **Select K-Best with chi2 score function**
- **Select From Model (LR, RF)**

customer_id	value_min	value_max	value_mean	value_std
1	10	17	13.0	3.6
2	18	26	21.5	3.4
3	14	19	16.3	2.5

category
A
B
A

Feature Transformation

- **Principal Component Analysis** (dependent variable not needed)
- **Linear Discriminant Analysis**

Feature Engineering in our case

- Feature Extraction with Statistical Methods
- Feature Extraction with customized function
- Feature Normalization with MinMaxScaler
- Feature Transformation with LDA

The Models Used

The models used

Binary Classification:

- Logistic Regression
- Decision Tree
- Random Forest
- Gradient boosting
- SVM

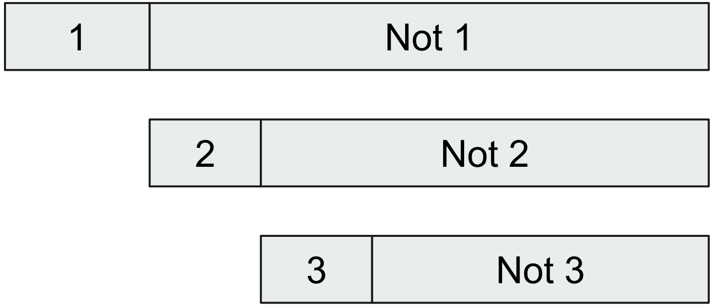
Multiple Classification

- Logistic Regression
- Random Forest
- Gradient boosting
- SVM

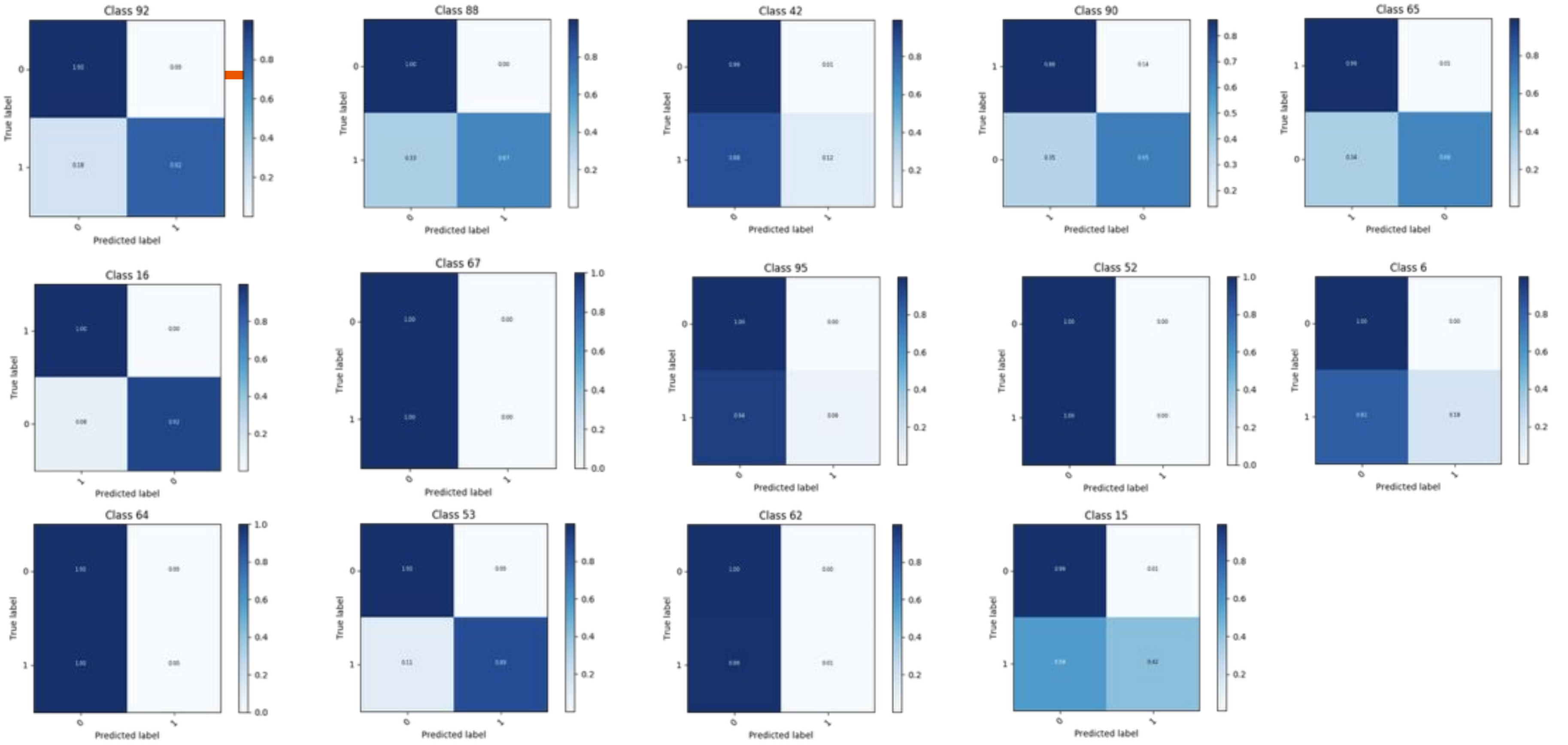
Models Comparison

Binary Classification

- 14 binary classification
- Giving a specific model to each class
- Create features: Peak_frequency
- Order is based on confusion matrix
- Higher TP and TN



Binary Classification



Binary Classification

- Logistic Regression
- Decision Tree
- Random Forest
- Gradient boosting
- SVM

Binary Classification

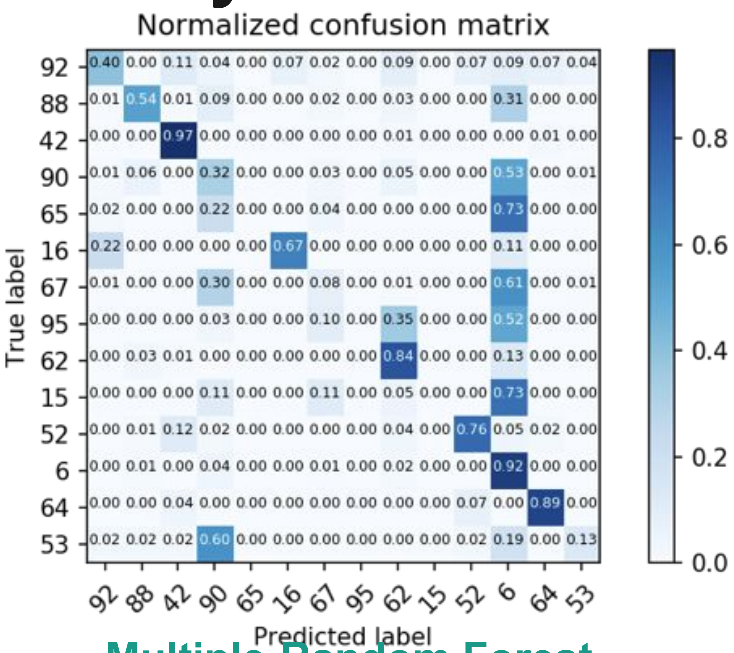
- Logistic Regression
- Decision Tree
- Random Forest
- Gradient boosting
- SVM

Binary Classification

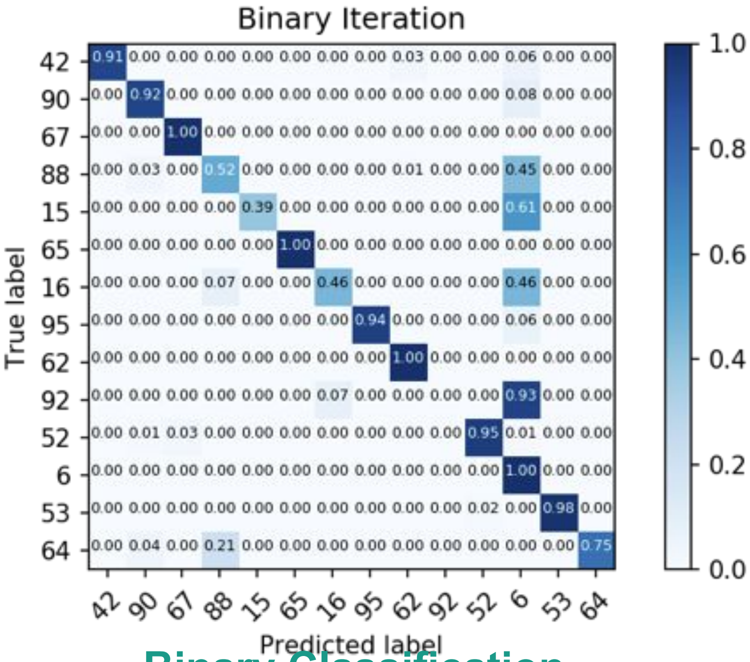
- Logistic Regression
- Random Forest

16	92	53	65	88	90	15	6	42	95	62	67	52	64

Binary Classification



Multiple-Random Forest

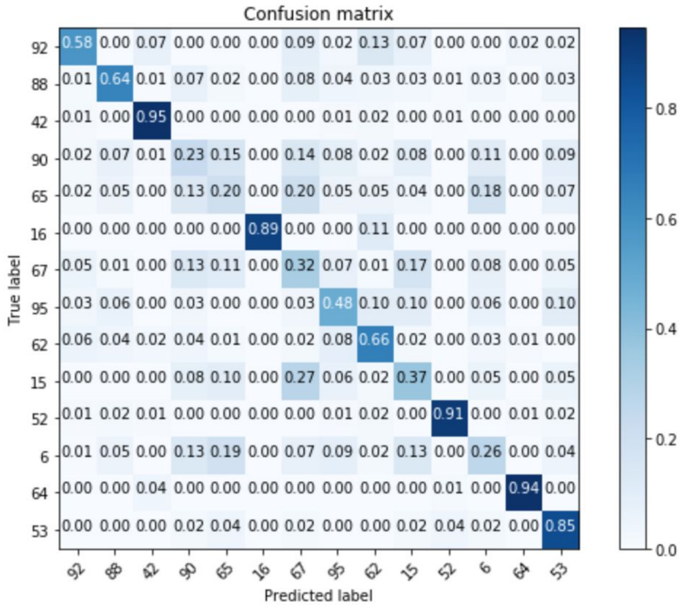


Binary Classification

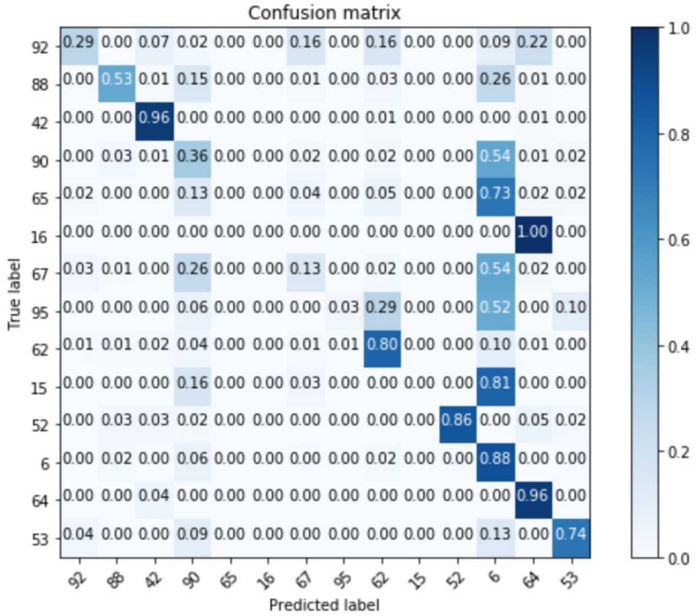
Multiple Classification

- Logistic Regression
- Random Forest
- Gradient Boosting
- SVM

Multiple Classification



Logistic Regression



SVM

Models Comparison

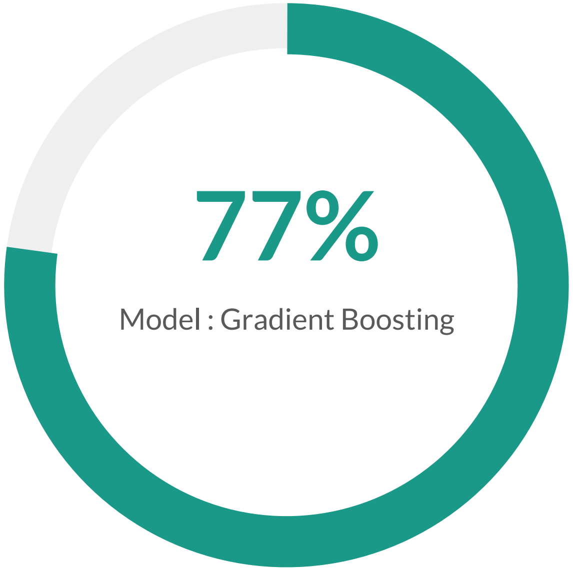
Method	Model	Accuracy	Features Used
Binary	Logistic Regression +Random Forest	0.61	All Features
Multiple	Logistic Regression	0.49	LDA Features
	SVM	0.66	LDA Features
	Random Forest	0.76	All Features
	Gradient Boosting	0.77	All Features

Conclusion and Future Scope

Conclusion

The best results were obtained by using Gradient Boosting, followed closely by Random Forest - 76%

Project website: https://manaseegodsay.github.io/MLProjectPlasticc/



What next?

- Trying out CNN on scatter plots of Time series data
- Trying out the XGBoost model
- Creating new features and running models on them

Questions?

References

https://www.kaggle.com/c/PLAsTiCC-2018

https://www.kaggle.com/mithrillion/strategies-for-flux-time-series-preprocessing

https://www.lsst.org/

https://www.kaggle.com/ashishpatel26/beginner-baseline-of-lgb-plotly

Thank you!