Machine Learning Methodologies for Detecting Pulsar Candidates

EMSE 6992: Data Science Introduction and Practicum

Final Project: Fall 2019

David DeBruce

Overview


Introduction


Methodology

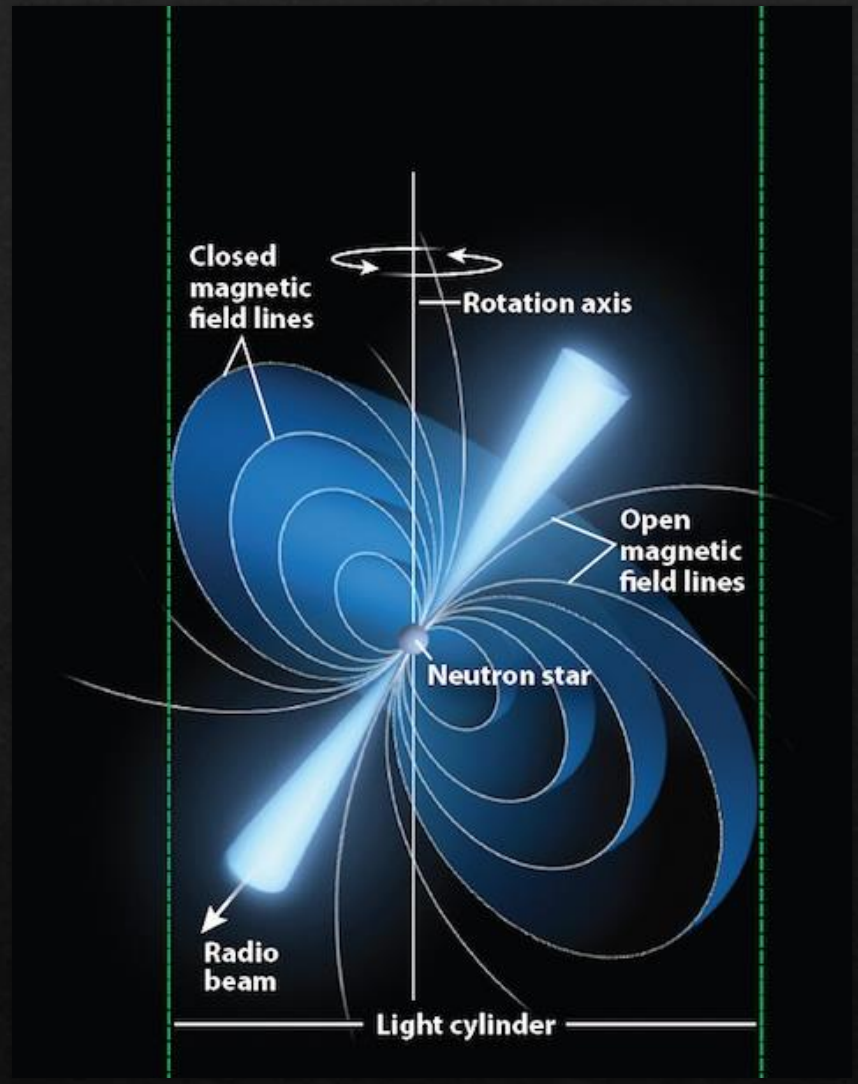

Results


Conclusion


References

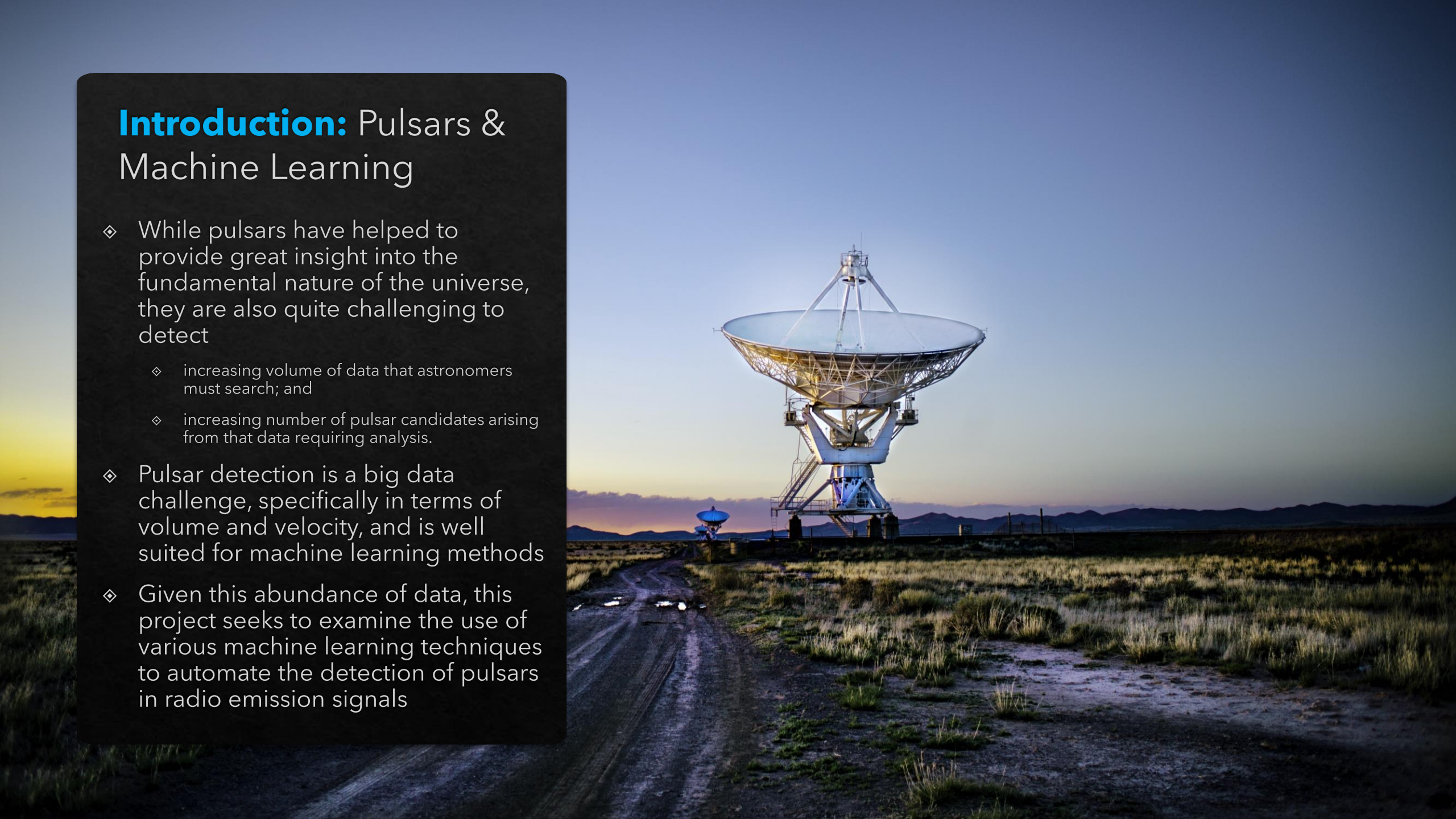
Introduction: What are Pulsars?

- Pulsars are rapidly rotating neutron stars formed after the collapse of a massive star during a supernova that emits beams of electromagnetic radiation at regular intervals
- As they rotate, their emission beam crosses our line of sight and produces a pattern of broadband radio waves detectable from Earth
- This regularity has proven to be an invaluable tool for astronomers to understand the universe, including in the discovery of extrasolar planets and confirmation of the existence of gravitational radiation



Introduction: Pulsars & Machine Learning

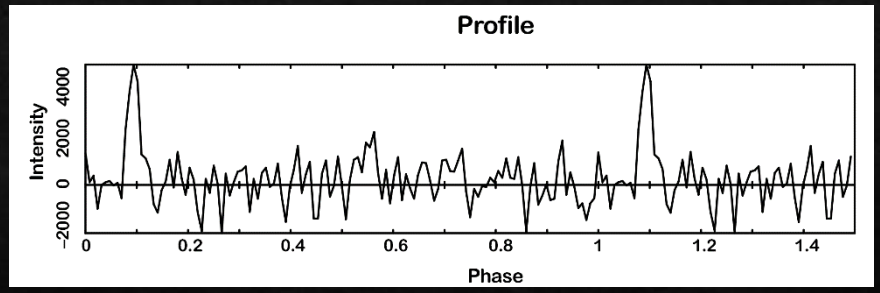
- While pulsars have helped to provide great insight into the fundamental nature of the universe, they are also quite challenging to detect
  - increasing volume of data that astronomers must search; and
  - increasing number of pulsar candidates arising from that data requiring analysis.
- Pulsar detection is a big data challenge, specifically in terms of volume and velocity, and is well suited for machine learning methods
- Given this abundance of data, this project seeks to examine the use of various machine learning techniques to automate the detection of pulsars in radio emission signals

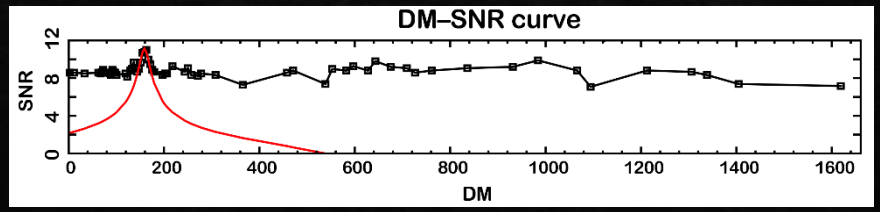


Methodology: Data Identification & Acquisition

- "Predicting a Pulsar Star" dataset from Kaggle which describes a sample of pulsar candidates that were collected during the High Time Resolution Universe survey
- 16,259 false examples (caused by noise & radio frequency interference) & 1,639 positive examples
- Each pulsar candidate is characterized by eight numerical features, and one class label
  - **Integrated Profile (IP)** – pulsar's fingerprint and describes the unique pulse profile shape of the radio signal
  - **DM-SNR** – plot of the dispersion measure, which is a pulse broadening effect that occurs over a finite bandwidth caused by free electrons, versus the measure signal-to-noise ratio

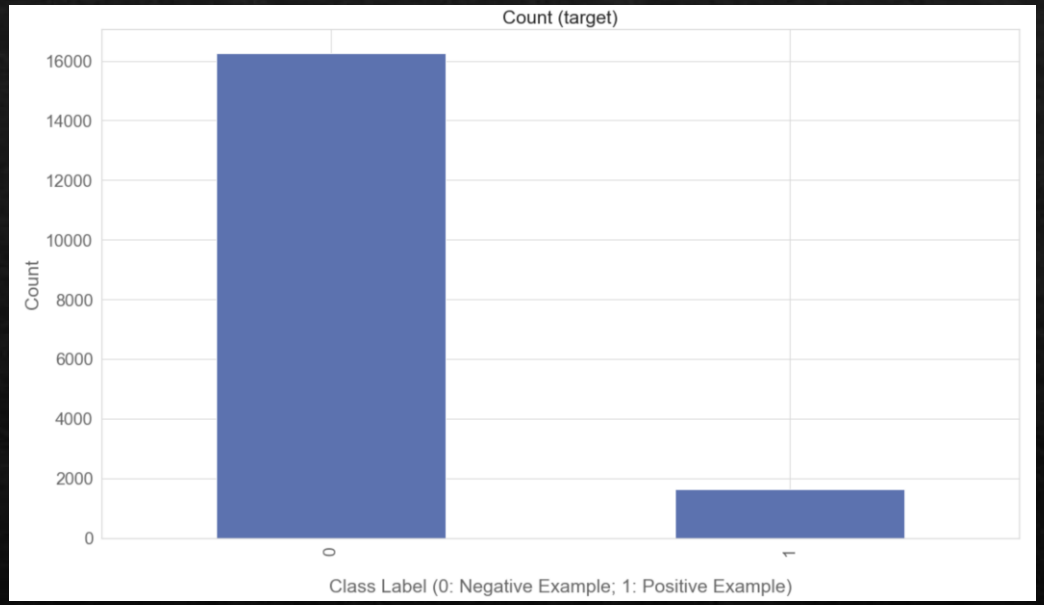
Mean of the integrated profile	`IP_mean`
Standard deviation of the integrated profile	`IP_std`
Excess kurtosis of the integrated profile	`IP_kurt`
Skewness of the integrated profile	`IP_skew`
Mean of the DM-SNR curve	`DMSNR_mean`
Standard deviation of the DM-SNR curve	`DMSNR_std`
Excess kurtosis of the DM-SNR curve	`DMSNR_kurt`
Skewness of the DM-SNR curve	`DMSNR_skew`
Class	`target`





Methodology: Data Representation

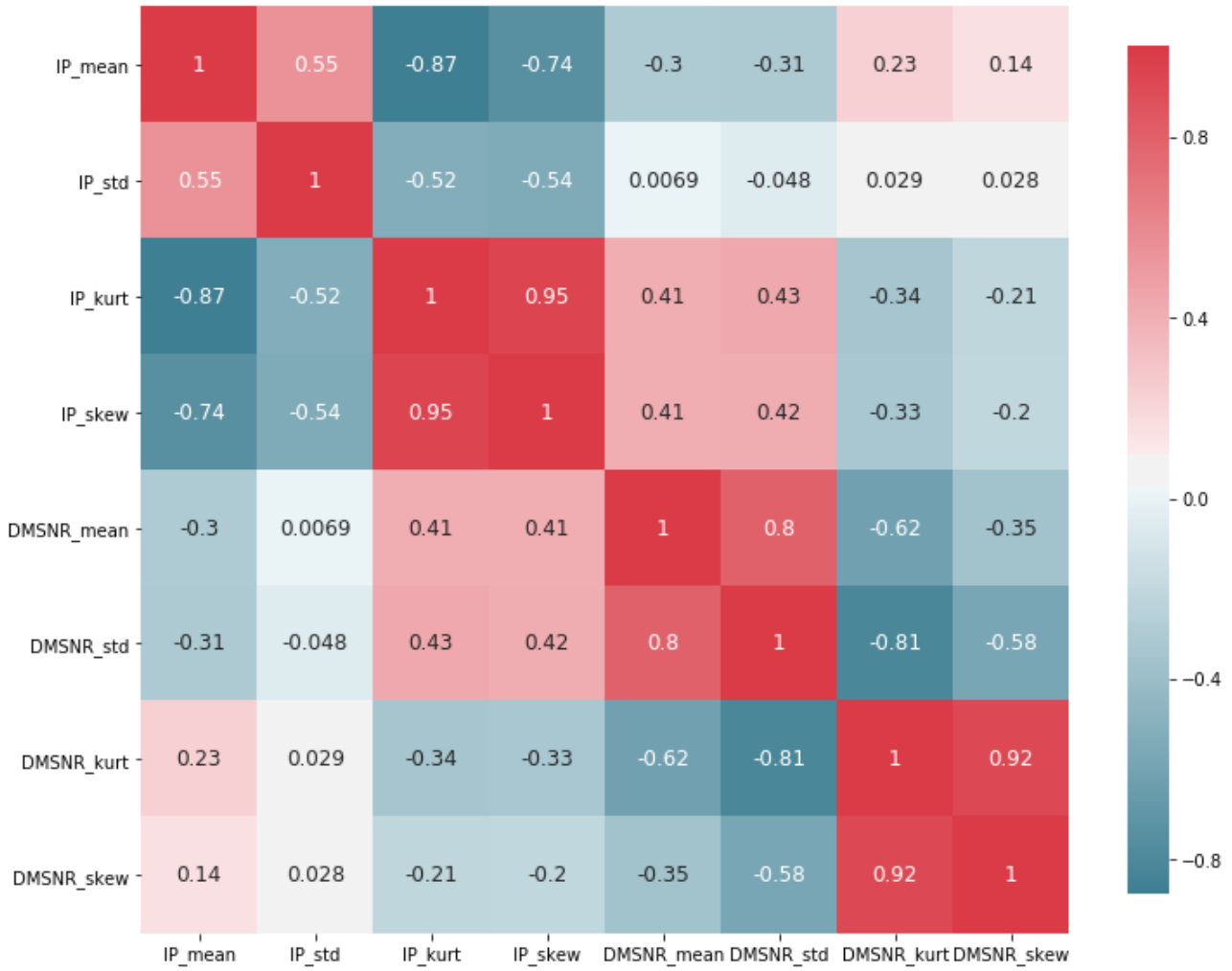
- Due to the class imbalance, random under sampling of the larger negative pulsar example class was used to ensure a more accurate representation of the machine learning models.
- If this technique was not used, the associated accuracy metrics might be misleading since it would simply predict the most common class label.
- The figure shows that the negative example class comprises over 90% of the dataset.

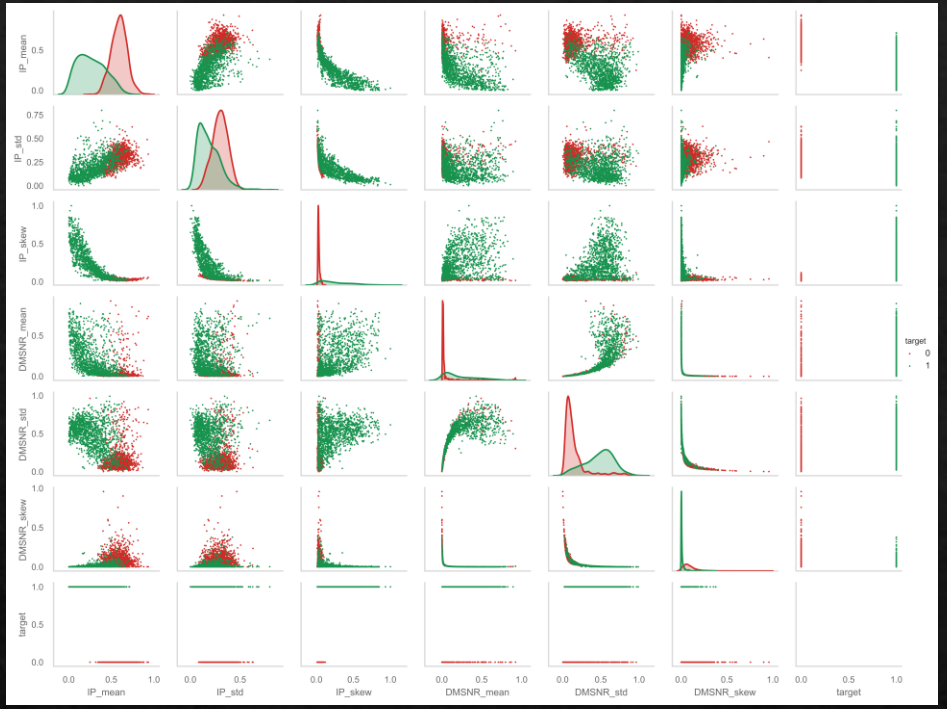


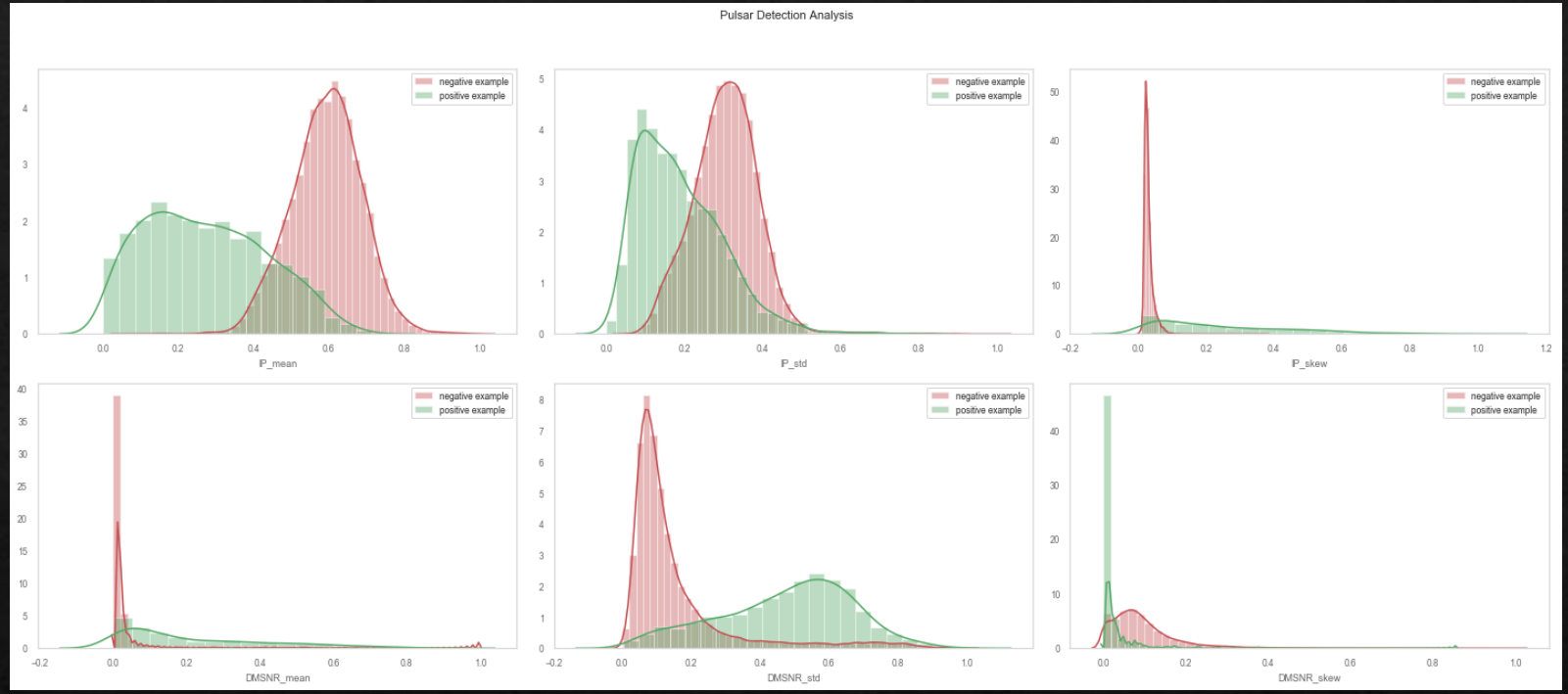
	Mean of the integrated profile	Standard deviation of the integrated profile	Excess kurtosis of the integrated profile	Skewness of the integrated profile	Mean of the DM-SNR curve	Standard deviation of the DM-SNR curve	Excess kurtosis of the DM-SNR curve	Skewness of the DM-SNR curve	target_class
0	140.562500	55.683782	-0.234571	-0.699648	3.199833	19.110426	7.975532	74.242225	0
1	102.507812	58.882430	0.465318	-0.515088	1.677258	14.860146	10.576487	127.393580	0
2	103.015625	39.341649	0.323328	1.051164	3.121237	21.744669	7.735822	63.171909	0
3	136.750000	57.178449	-0.068415	-0.636238	3.642977	20.959280	6.896499	53.593661	0
4	88.726562	40.672225	0.600866	1.123492	1.178930	11.468720	14.269573	252.567306	0

Methodology:
Exploratory Data Analysis

- Determined the basic characteristics of the data such as patterns/anomalies and feature correlation
- The correlation heatmap numerically shows how each feature relates to every other feature in the dataset
- The plot shows that some features have a positive correlation, such as `DMSNR_kurt` and `DMSNR_skew`, while some features are negatively correlated, such as `IP_kurt` and `IP_mean`







Methodology: Exploratory Data Analysis

- Pair plot shows the distribution of each feature
- The diagonal axes show the univariate distribution of the data for the feature in that column

Methodology: Normalization & Feature Engineering

- Since many ML models are employed, normalization is performed to scale the features between 0 and 1
- Features with high correlation (> 90%) are more linearly dependent and have almost the same effect on the dependent variable. Therefore, `IP_kurt` & `DMSNR_kurt` were dropped due to their high correlation

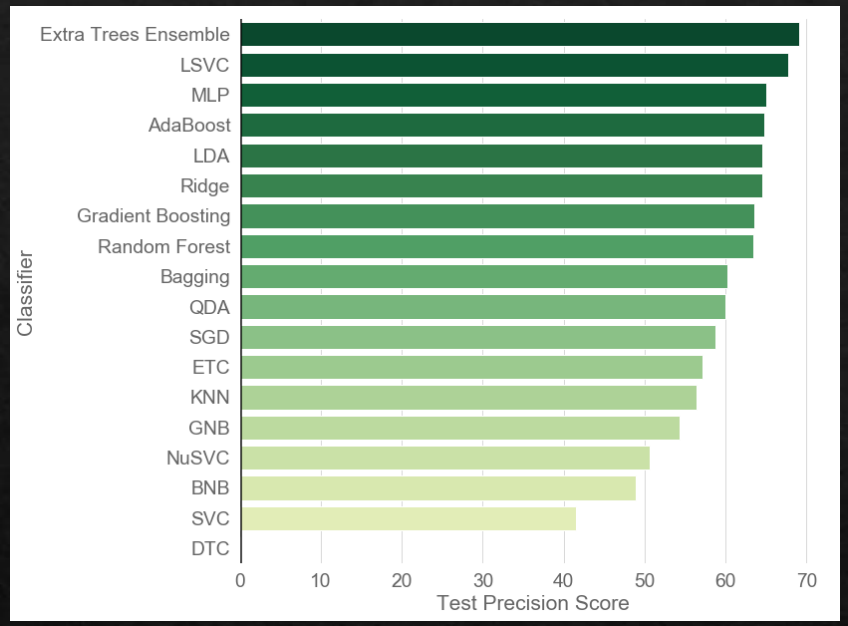
	IP_mean	IP_std	IP_skew	DMSNR_mean	DMSNR_std	DMSNR_skew	target
0	0.721342	0.417687	0.015627	0.013382	0.113681	0.063890	0
1	0.517628	0.460908	0.018268	0.006560	0.072524	0.108443	0
2	0.520346	0.196868	0.040677	0.013030	0.139188	0.054610	0
3	0.700933	0.437884	0.016534	0.015368	0.131583	0.046581	0
4	0.443854	0.214847	0.041712	0.004327	0.039684	0.213369	0

Methodology: ML Model Development

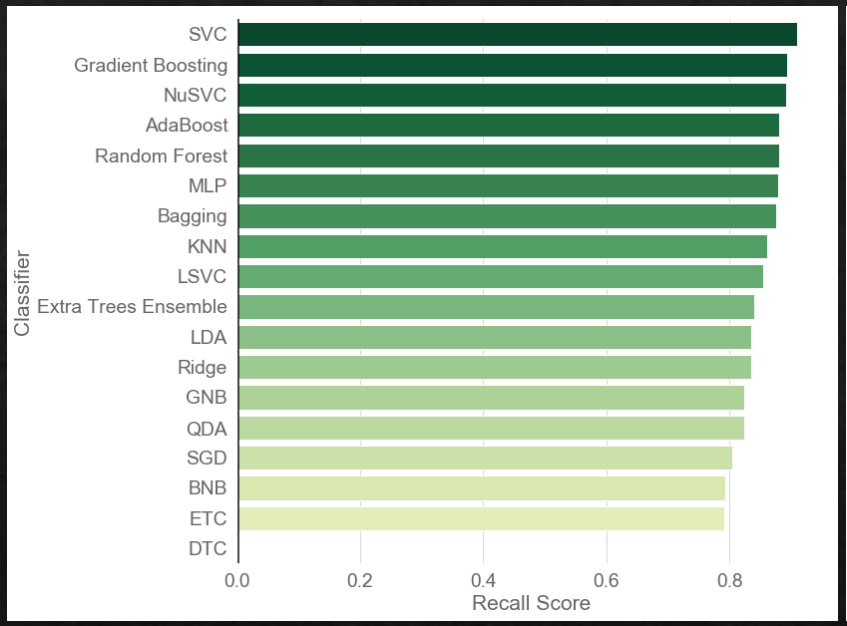
- Trained 18 different classifiers using Scikit-learn library
- Parameter Grid Search for each classifier to determine optimal hyperparameter space that maximizes recall score

Models	
Latent Dirichlet Allocation	BernoulliNB Classifier
Quadratic Discriminant Analysis	GaussianNB Classifier
AdaBoost Classifier	K Nearest Neighbors Classifier
Bagging Classifier	Multi-layer Perceptron Classifier
Gradient Boosting	Linear Support Vector Classification
Extra Trees Ensemble	Nu-Support Vector Classification
Random Forest	C-Support Vector Classification
Ridge	Decision Tree Classifier
SGD Classifier	Extra Tree Classifier

Results

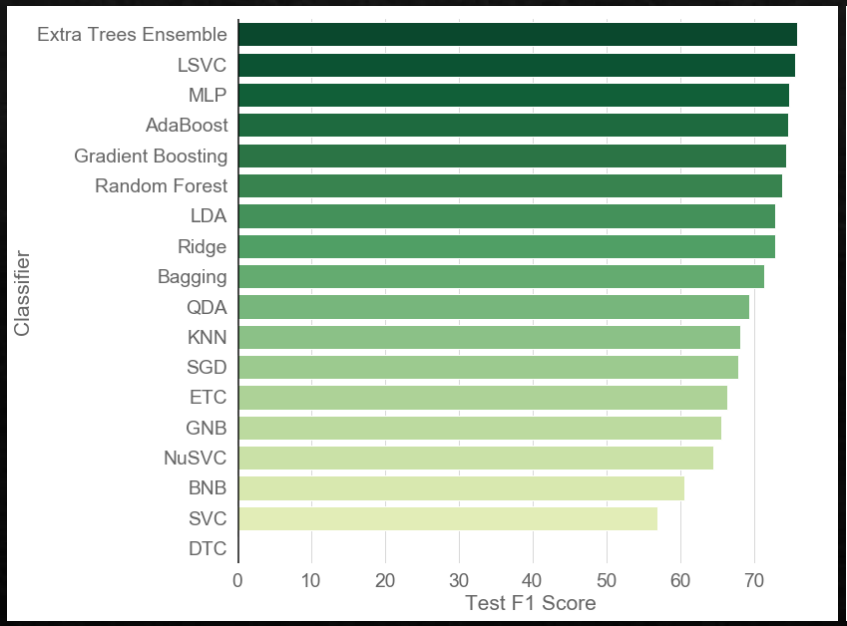


	Classifier	Test Precision Score
13	Extra Trees Ensemble	69.067797
4	LSVC	67.755102
5	MLP	65.076336
15	AdaBoost	64.772727
17	LDA	64.541833
10	Ridge	64.541833
12	Gradient Boosting	63.553114
11	Random Forest	63.450835
14	Bagging	60.176991
16	QDA	59.925094
9	SGD	58.757062
0	ETC	57.169460
6	KNN	56.418919
7	GNB	54.329372
3	NuSVC	50.584795
8	BNB	48.966614
2	SVC	41.480611
1	DTC	0.000000



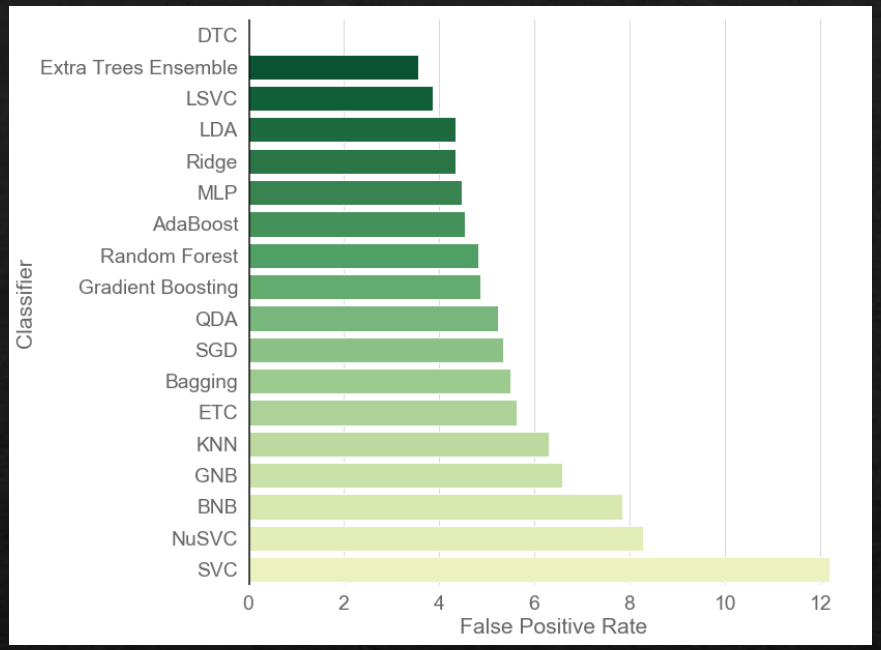
	Classifier	Recall Score
2	SVC	0.909794
12	Gradient Boosting	0.894330
3	NuSVC	0.891753
15	AdaBoost	0.881443
11	Random Forest	0.881443
5	MLP	0.878866
14	Bagging	0.876289
6	KNN	0.860825
4	LSVC	0.855670
13	Extra Trees Ensemble	0.840206
17	LDA	0.835052
10	Ridge	0.835052
7	GNB	0.824742
16	QDA	0.824742
9	SGD	0.804124
8	BNB	0.793814
0	ETC	0.791237
1	DTC	0.000000

- **Precision** is the number of True Positives divided by the number of True Positives and False Positives
- **Recall** is the number of True Positives divided by the number of True Positives and the number of False Negatives
- **F1 Score** conveys the balance between the precision and the recall



	Classifier	Test F1 Score
13	Extra Trees Ensemble	75.813953
4	LSVC	75.626424
5	MLP	74.780702
15	AdaBoost	74.672489
12	Gradient Boosting	74.304069
11	Random Forest	73.786408
10	Ridge	72.808989
17	LDA	72.808989
14	Bagging	71.353620
16	QDA	69.414317
6	KNN	68.163265
9	SGD	67.899891
0	ETC	66.378378
7	GNB	65.506653
3	NuSVC	64.552239
8	BNB	60.570305
2	SVC	56.981437
1	DTC	0.000000

	Classifier	False Positive Rate
1	DTC	0.000000
13	Extra Trees Ensemble	3.572302
4	LSVC	3.865916
17	LDA	4.355273
10	Ridge	4.355273
5	MLP	4.477612
15	AdaBoost	4.551015
11	Random Forest	4.820161
12	Gradient Boosting	4.869097
16	QDA	5.236115
9	SGD	5.358454
14	Bagging	5.505261
0	ETC	5.627600
6	KNN	6.312699
7	GNB	6.581845
8	BNB	7.854172
3	NuSVC	8.270125
2	SVC	12.184977



Conclusion

- **Extra Trees Ensemble** was the best performing model, and had the lowest false positive rate
- The model implements a meta estimator that fits a number of randomized decision trees (a.k.a. extra-trees) on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control overfitting

References

- Ceballos, F. (2019, July 27). Searching for Pulsars with Machine Learning. Retrieved from https://medium.com/i-want-to-be-the-very-best/searching-for-pulsars-with-machine-learning-f4db5fa58b3c
- kaggle. (2018, May 9). Predicting a Pulsar Star. Retrieved from https://www.kaggle.com/pavanraj159/predicting-a-pulsar-star
- Lyon, D. R. (2017, February 14). HTRU2 Data Set. Retrieved from https://archive.ics.uci.edu/ml/datasets/HTRU2
- Lyon, R. J. (2016). Why Are Pulsars Hard To Find? University of Manchester, School of Computer Science.


Questions?