Machine Learning Methodologies for Detecting Pulsar Candidates

EMSE 6992: Data Science Introduction and Practicum

Final Project: Fall 2019

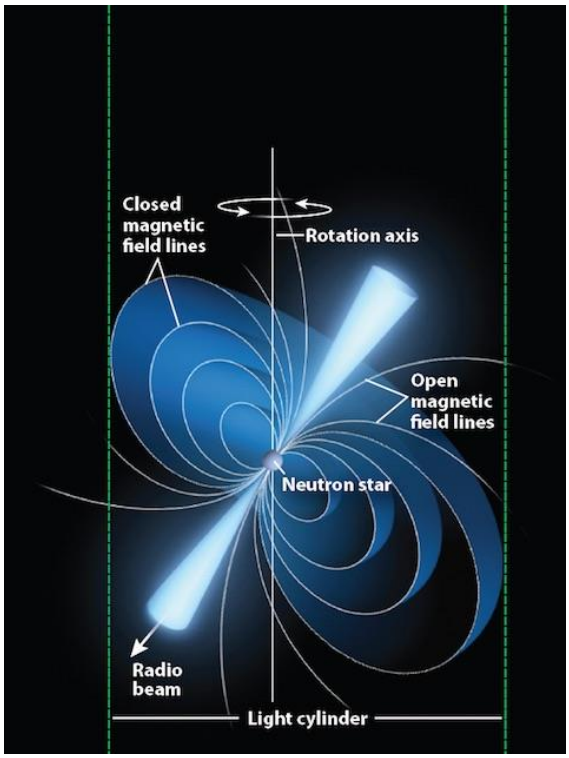
David DeBruce

Contents

Abstract

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. While pulsars have helped to provide great insight into the fundamental nature of the universe, they are also quite challenging to detect. This project will employ data visualization methods to display the results of an exploratory data analysis and the increase in classification accuracy gained with machine learning. The data used in this project is the "Predicting a Pulsar Star" dataset from Kaggle, which describes a sample of pulsar candidates and builds on work developed by Frank Ceballos. Specifically, normalization and feature engineering were incorporated before training to determine impact on accuracy metrics. Since this project seeks to employ a wide range of models using varying hyperparameters, we use the 18 models listed below. Normalization is performed to scale the features in the data between 0 and 1. Features with a high correlation (above 90%) are more linearly dependent and have almost the same effect on the target class. This project trained 18 different classifiers using the Scikit-learn library using the models listed below. Parameter grid search was also performed for each classifier to determine the optimal hyperparameter space to maximize recall scores.

Introduction



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 turn, their emission beam crosses our line of sight and produces a pattern of broadband radio waves detectable from Earth. This regular periodicity has proven to be an invaluable tool for astronomers to understand the universe, including in their discovery of extrasolar planets and confirmation of the existence of gravitational radiation.

While pulsars have helped to provide great insight into the fundamental nature of the universe, they are also quite challenging to detect. Experts argue that this is due to two main reasons, (1) the increasing volume of data that astronomers must search; and (2) the corresponding growing number of pulsar candidates arising from that data requiring analysis.[1] Pulsar detection is a big data challenge, specifically in terms of volume and velocity, and is well suited for machine learning methodologies. 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. This project will also employ data visualization methods to display the results of an exploratory data analysis and the increase in classification accuracy gained with machine learning.

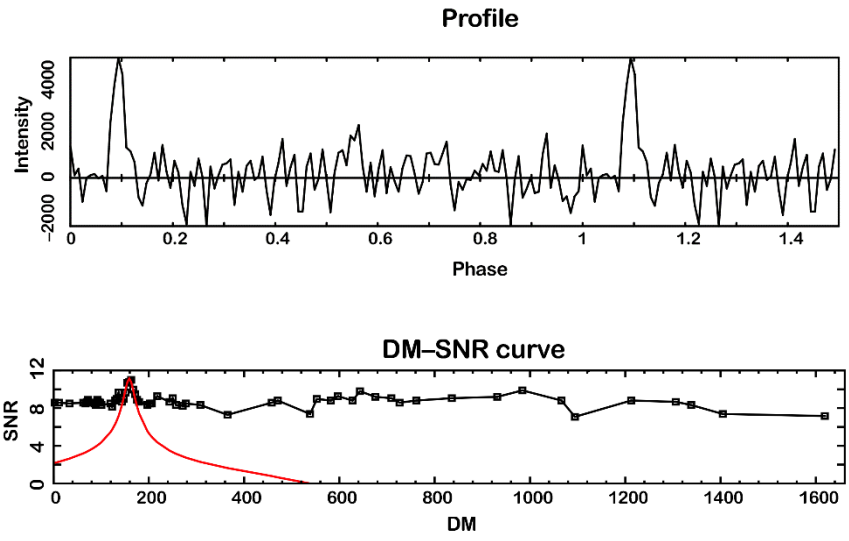
[1] Lyon, R. J. (2016). Why Are Pulsars Hard to Find? University of Manchester, School of Computer Science.

Methodology

Data Identification & Acquisition

The data used in this project is the "Predicting a Pulsar Star" dataset from Kaggle, which describes a sample of pulsar candidates that were collected during the High Time Resolution Universe survey.[2] [3] The dataset is a table of comma-separated values that consists of 16,259 false examples, caused by noise and radio frequency interference, and 1,639 positive pulsar examples. Each pulsar candidate is characterized by eight numerical features and one class label as described below. The integrated profile can be thought of as a pulsar's fingerprint and describes the unique pulse profile shape of the radio signal. DM-SNR is a 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.

1.	Mean of the integrated profile	`IP_mean`
2.	Standard deviation of the integrated profile	`IP_std`
3.	Excess kurtosis of the integrated profile	`IP_kurt`
4.	Skewness of the integrated profile	`IP_skew`
5.	Mean of the DM-SNR curve	`DMSNR_mean`
6.	Standard deviation of the DM-SNR curve	`DMSNR_std`
7.	Excess kurtosis of the DM-SNR curve	`DMSNR_kurt`
8.	Skewness of the DM-SNR curve	`DMSNR_skew`
9.	Class	`target`

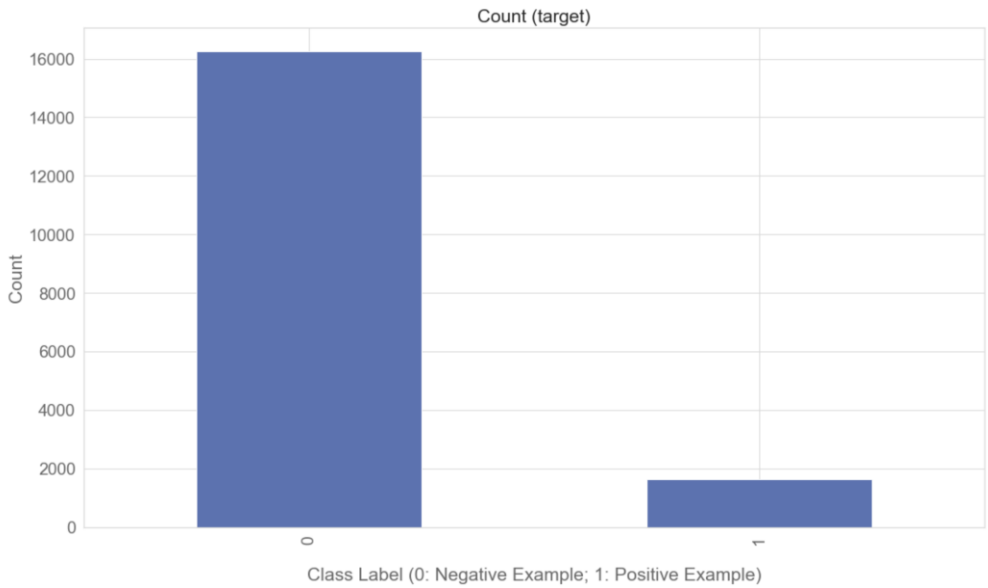


[2] kaggle. (2018, May 9). Predicting a Pulsar Star. Retrieved from https://www.kaggle.com/pavanraj159/predicting-a-pulsar-star

[3] Lyon, D. R. (2017, February 14). HTRU2 Data Set. Retrieved from https://archive.ics.uci.edu/ml/datasets/HTRU2

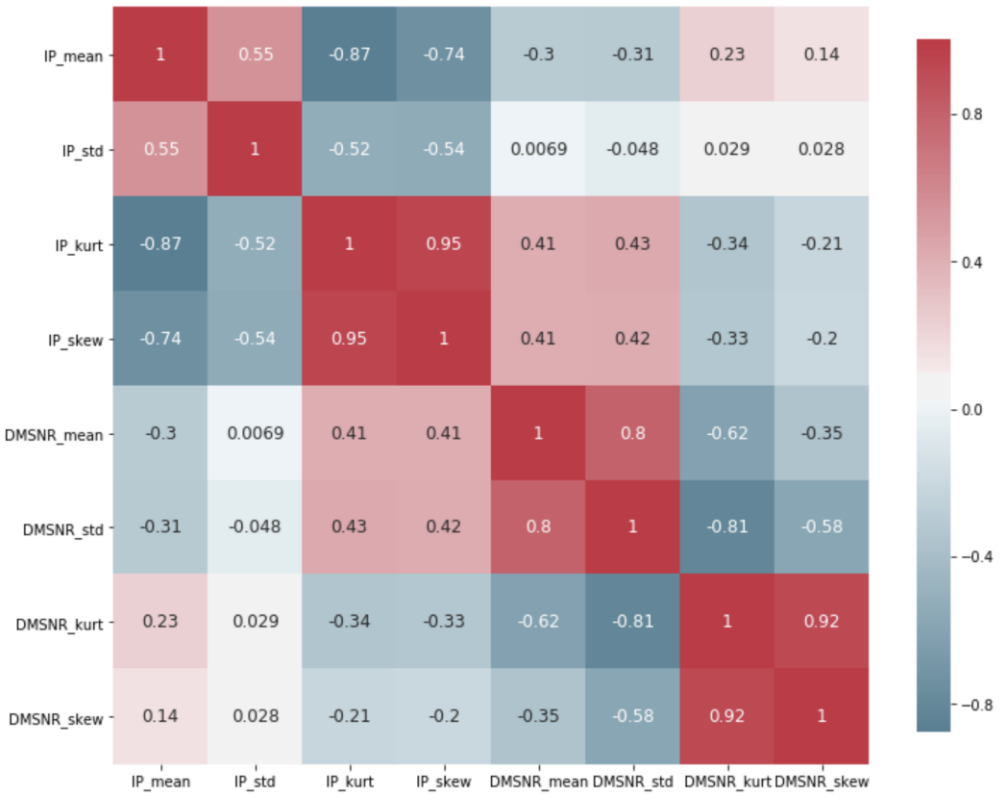
Data Representation

Due to the class imbalance, as discussed in the previous section, 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 merely predict the most common class label. The figure below shows that the negative example class comprises over 90% of the dataset.

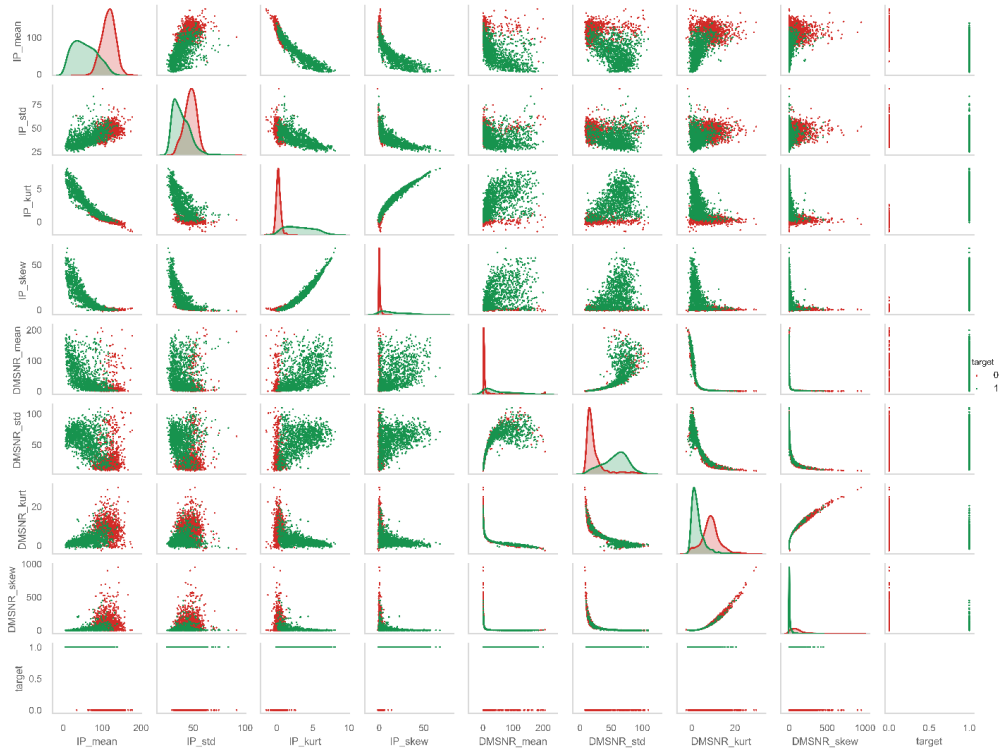


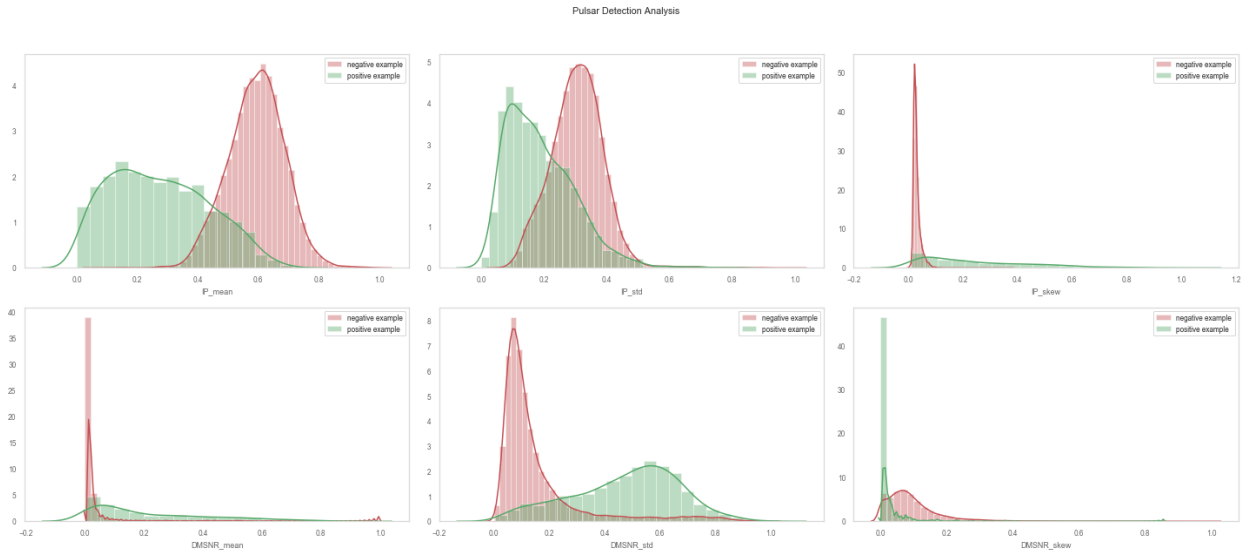
Data Analysis

First, we performed exploratory data analysis to determine the basic characteristics of the data, such as patterns and anomalies, and understand feature correlation. The correlation heatmap below numerically shows how each feature relates to every other feature in the dataset where red represents the negative pulsar example class, and green represents a positive pulsar example. 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`.



This information is also represented graphically in the plots below.





The pair plot above shows the distribution of each feature, while the diagonal axes show the univariate distribution of the data for the feature in that column.

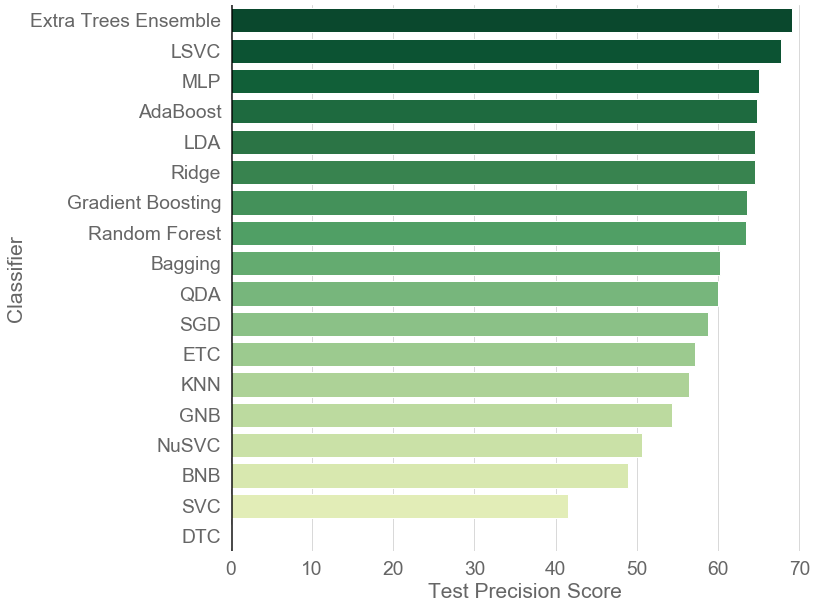
The next step is the development of machine learning models. Since this project seeks to employ a wide range of models using varying hyperparameters, we use the 18 models listed below. Normalization is performed to scale the features in the data between 0 and 1. Features with a high correlation (above 90%) are more linearly dependent and have almost the same effect on the target class. This project trained 18 different classifiers using the Scikit-learn library using the models listed below. Parameter grid search was also performed for each classifier to determine the optimal hyperparameter space to maximize recall scores.

Models	
Latent Dirichlet Allocation	BernoulliNB Classifier
Quadratic Discriminant Analysis	GaussianNB Classifier
AdaBoost Classifier	K Nearest Neighbors Classifier
Bagging Classifier	MLP Classifier
Gradient Boosting	Linear Support Vector Classification
Extra Trees Ensemble	NuSVC
Random Forest	SVC
Ridge	Decision Tree Classifier
SGD Classifier	Extra Tree Classifier

Results

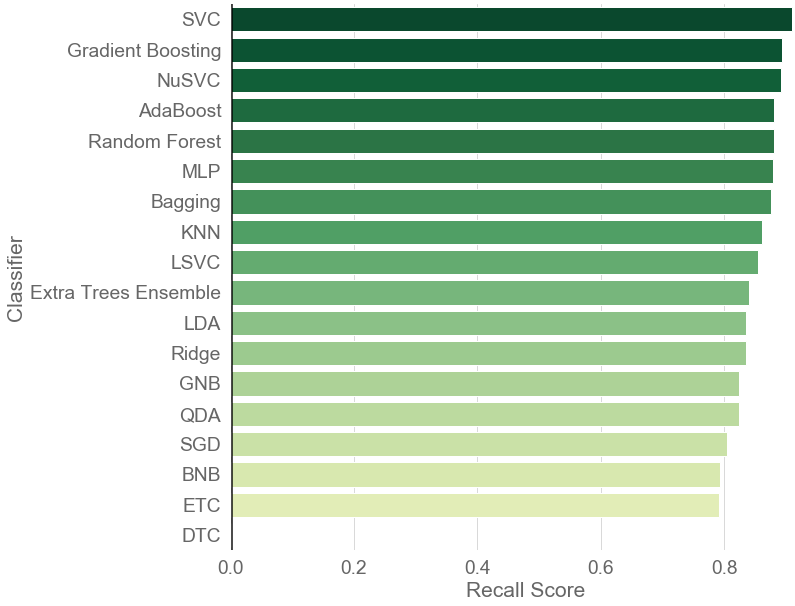
To identify the most accurate machine learning model, this project uses recall, precision, and F1 scores as shown in the visualizations below.

Precision Scores:



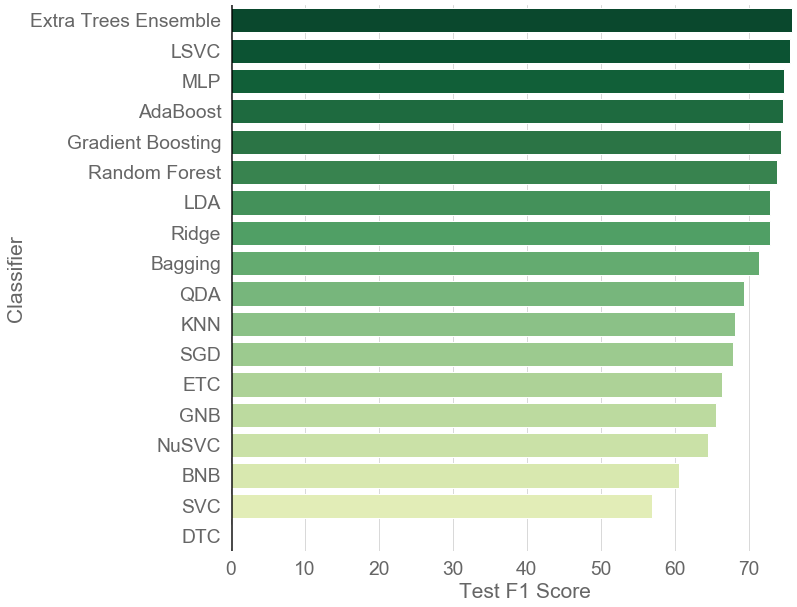
	Classifier	Test Precision Score
13	Extra Trees Ensemble	69.067797
4	LSVC	67.755102
5	MLP	65.076336
15	AdaBoost	64.772727
17	LDA	64.541833
10	Ridge	64.541833
12	Gradient Boosting	63.553114
11	Random Forest	63.450835
14	Bagging	60.176991
16	QDA	59.925094
9	SGD	58.757062
0	ETC	57.169460
6	KNN	56.418919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

Recall Scores:



	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

F1 Scores:



	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

Conclusion

The Extra Trees Ensemble model was the best performing model, which also had the lowest false positive rate. This decision tree model fits a specified number of randomized decision trees (i.e., extra-trees) on various sub-samples and uses averaging to improve the prediction 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.