Pulsars - Will it classify?

Patrick Coffey

15/08/2019

Abstract

Can pulsars be classified using common machine learning techniques? This report intends to evaluate the effectiveness of 4 machine learning models. Logistic Regression, Naive Bayes, Decision Tree, and Random Forest models are trained and then validated on data-set comprised of statistical measures of the High Time Resolution Universe Study (Bates et al. 2011). Data is standardised to a normal distribution and PCA is used to visualise and assess the ability to cluster or identify outliers. Classifiers are trained on the data-set and show that even with a limited number of samples most classifiers are quite effective. This has distinct and strong implications for automating the search for pulsars in increasing the efficiency of current methods.

Introduction

Due to the utility of pulsars in astrophysics there is significant research going into identifying Pulsars. Any advances in the classification and automation of searches is greatly welcomed by the scientific community.

Pulsars are stars comprised entirely of neutrons. Neutron stars are the most compact stars in the observed universe (Y Potekhin 2010). Pulsars are specifically spinning neutron stars. This is apparent by the oscillating bursts of Radiowaves and light emited from either end of the neutron star. Since the spin rate is mostly constant it allows researchers to calculate some very interesting values. The distance to a pulsar can be calculated within a margin of error or 30 meters (Technology 2007b). Due to this fact, pulsars are one of the most useful scientific bodies in the universe, providing a similar "yard stick" function to Cephid Variable binary star systems. Informally they are referred to as "Cosmic Lighthouses".

This paper intends to assess multiple binary classification techniques to ascertain the effectiveness of machine learning in the search for pulsars. As part of this investigation, the most important features of the dataset will be identified for future works in this area.

Data

The data ultilised by this report comes from Kaggle, it was uploaded by the user Pavan Raj and was originally collated by Dr. Robert Lyon from the University of Manchester's Jodrell Bank Center for Astrophysics. Initially this data was collected as part of the High Time Resolution Universe Study (HTRU2) (Bates et al. 2011). It was collected using the multibeam receptor on the Parkes Radio Telescope. The sample period was 64 micro seconds meaning this study had the ability to identify very short period pulsars.

This data-set consists of 17898 samples labelled samples, of which 16259 are not pulsars and 1639 are. The last column of each sample is a Boolean representation of whether the observation represents a Pulsar. The remaining columns consist of 8 continuous variables representing statistical values (Mean, Standard Deviation, Kurtosis, Skewness) of the Integrated Pulse (IP) profile (first 4) and the same statistical measures of the Dispersion Measure-Signal to Noise Ratio (DM-SNR) curve.

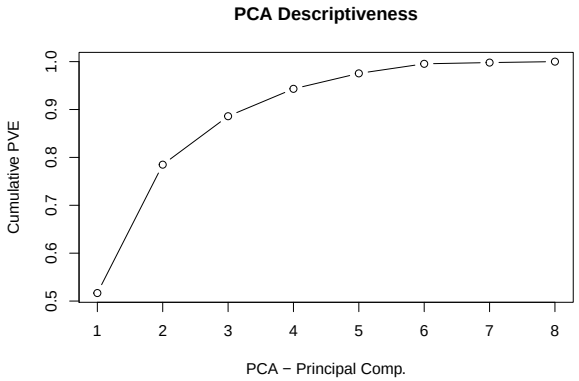
The Dispersion Measure represents the "integrated column density of free electrons between an observer and a pulsar", which equates more simply to the number of electrons between the observing device and the pulsar inside a 1 centimeter diameter column (Technology 2007a). The Dispersion Measure manifests to the observer as a delay in the pulse from the Pulsar due to the interaction of the free electrons on the electromagnetic waves that comprise the light/radio bursts Pulsars are known for. RAther than using the raw measures from teh telescope the aforementioned statistical measures have been taken to reduce the complexity of this data-set by effectively aggregating each sample longitudinally and averaged in both temporal and frequency domains.

Since the aim of this paper is to ascertain the best way to identify the pulsars in this data-set and the features themselves are actually quite complicated in exactly what they represent it is easier to think of them simply as statistical measures of stars that can be used to classify the samples. Deeper Explanations can be found in the referenced paper.

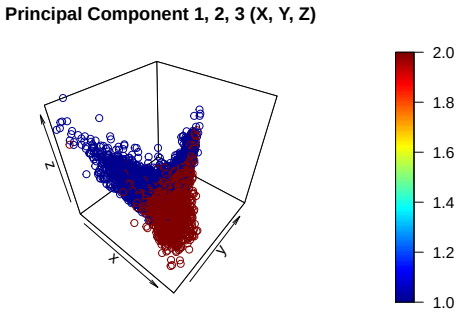
Methods

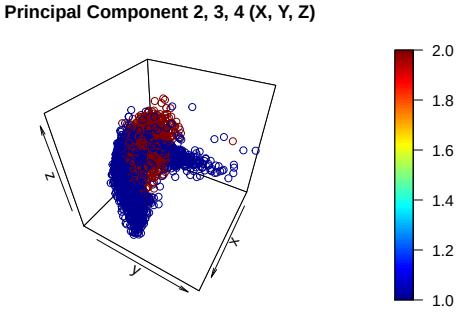
All analysis and modelling was completed using R version 3.6.1 (R Core Team 2019). Initially the data-set

was viewed using the labels and Principal Component Analysis to ascertain whether there were any distinct clusters or outliers. the R package plot3D (Soetaert 2017) was used to visualise the first 4 Primary Components over two separate three dimensional charts. The first four Primary Components represent over 90% of the variance in this data-set as seen in the PCA Descriptiveness plot chart below.



It is clear that the targets of this investigation (Pulsars) are not separated into there own clusters or represent outliers in this data-set. Further analysis of the PCA Descriptiveness chart showed classifiers should produce reasonable results since there are clear decision boundaries that could be drawn through the two Principal Component charts in the 3 dimensional spaces of the first 4 Principal Components. Since this data-set is almost linearly separable simple classification methods like Naive Bayes Classifiers and Logistic Regression should yield reasonable results.





Four models are Assessed for accuracy using 40% of the data-set as a holdout validation data-set. These models are Logistic Regression, provided by the stats (R Core Team 2019) package; Naive Bayes, provided by the naivebayes (Majka 2019) package; Decision Trees, provided by the tree (Ripley 2019) package; and Random Forests, provided by the randomForest (Liaw and Wiener 2002) package.

All models are trained against to Train data-set and evaluated on the validation data-set. There is a 60%/40% split leaving a moderate amount of data for validation. This allows for a more precise evaluation of the data since we are using a large portion for validation. Since there is a strong class imbalance present in this data-set with positive results only comprising 9.16% of the total samples. A baseline classifier score of 90.84% can be obtained by simply predicting negative for all samples. Using this baseline, the aforementioned classifiers will be assessed.

Logistic Regression

Logistic Regression is an extension of Linear Regression that produces a classifier (McCullagh and Nelder 1989). This is the most simple classifier tested in this paper and produces spectacular results. Logistic Regression is very sensitive to outliers just like Linear regression (McCullagh and Nelder 1989), this model is trained on the scaled and normalised data rather than the raw data-set.

```
# build  logistic regresion model
lrm <- stats::glm(
  formula=IsPulsar~.,
  data=as.data.frame(
    scaled_pulsar_df[train_idx,]),
  family=binomial
)
```

All models will be validated using the same process; this is shown in the code block below. If the classifier

Table 1: Logistic Reg. Confusion Matrix

	0	1
0	6482	102
1	34	541

Table 2: Naive Bayes Confusion Matrix

	0	1
0	4235	388
1	2281	255

Table 3: Decision Tree Confusion Matrix

	0	1
0	6468	100
1	48	543

produces a probability of the class being positive this is "hardened" to a zero or one using 0.5 as the cut off. This is achieved using the `ifelse` function of R as follows: `ifelse(probability_predictions > 0.5, 1, 0)`.

```
# get predictions
lrm_probs <- predict(
  object=lrm,
  newdata=scaled_pulsar_df[test_idx,],
  type='response')

# hard sigmoid the probabilities
lrm_preds <- ifelse(lrm_probs > 0.5, 1, 0)

# produce the cross tab
lrm_tab <- table(
  lrm_preds,
  pulsar_df[test_idx,"IsPulsar"])

# calculate accuracy
lrm_accuracy <-sum(
  diag(lrm_tab))/sum(lrm_tab)

# show accuracy
print(paste0(
  'Acc: ', lrm_accuracy*100, '%'))
```

```
## [1] "Acc: 98.1002933370582%"
```

```
# show confusion matrix
knitr::kable(
  lrm_tab,
  caption = "Logistic Reg. Confusion Matrix")
```

Naive Bayes Classifier

Naive Bayes Classifiers tackle the classification problem from a different approach to the Logistic Regression Classifier. Instead of looking to seperate the data directly, they instead attempt to look at the probability of a given feature appearing in a given class. This can produce state of the art results on highly dimensional data sets (often in the NLP realm) since the resulting model is very light on computing resources.

```
# create naive bayes classifier
nbm <- naivebayes::naive_bayes(
  formula=IsPulsar~.,
  data=as.data.frame(
    scaled_pulsar_df[train_idx,])
)
```

```
## [1] "Acc: 62.7182567397681%"
```

Decision Tree Classifier

Decision Tree Classifiers attpempt to discriminate classes by using a feature and picking a split point to reduce the classification space as much as possible (Quinlan 1986). Multiple levels of "Decisions" are stacked into "Trees". Decision Trees are often a very solid choice when first tackling a classification problem since they are simple and often perform quite well.

```
# create the decision tree model
dtm <- tree::tree(
  formula=IsPulsar~.,
  data=as.data.frame(
    scaled_pulsar_df[train_idx,])
)
```

```
## [1] "Acc: 97.9326721609163%"
```

Random Forest Classifier

Random Forest Classifiers are an extension of decision trees, where numerous shallow Decision Tree's are combined through a voting mechanism to produce a well rounded and generalised classifier (Breiman 2001).

```
# create the random forest model
rfm <- randomForest::randomForest(
  formula=IsPulsar~.,
  data=as.data.frame(
    scaled_pulsar_df[train_idx,]),
  importance=TRUE
)
```

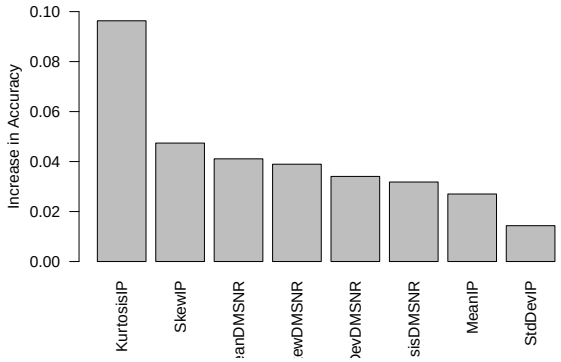
```
## [1] "Acc: 98.1002933370582%"
```

```
# get feature importance
# from Random Forest
```

Table 4: Random Forest Confusion Matrix

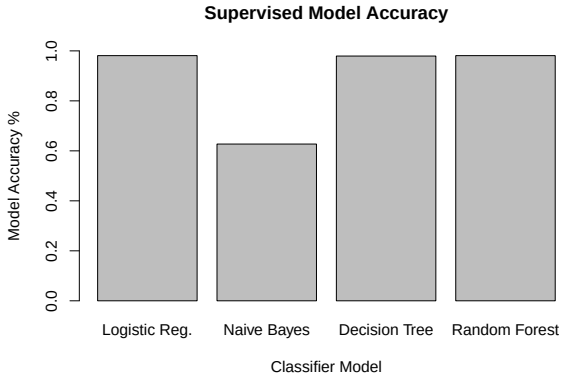
	0	1
0	6472	92
1	44	551

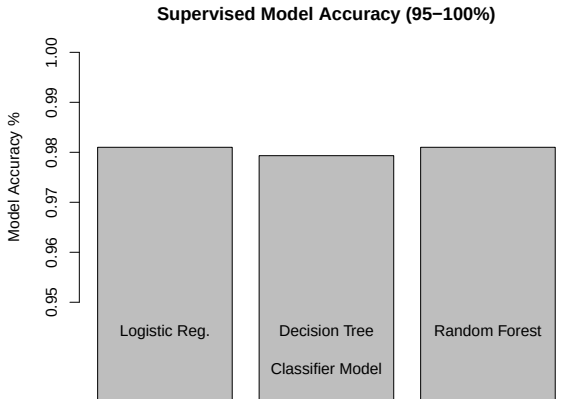
```
feat_importance <- rfm$importance
important_feats <- names(
  sort(feat_importance[,3],
       decreasing = TRUE)[1:4])

# sort and plot them
barplot(
  height = sort(feat_importance[,3],
                decreasing = TRUE),
  names.arg = names(sort(
    feat_importance[,3],
    decreasing = TRUE)),
  las=2,
  ylim = c(0.0, 0.1),
  ylab="Increase in Accuracy"
)
```



Kurtosis of the Integrated Pulse (KurtosisIP) is clearly the most important feature as defined by the Random Forest Classifier trained for this paper. Its effect on accuracy was over 2x stronger than any other feature. Analysis of this shows that much lighter classifiers could be constructed using just the first 4 features as ranked by the Random Forest Classifier. In the case of Naive Bayes Classifier this may increase the classification accuracy since there would be less probability distributions muddying the waters. Naive Bayes Classifiers treat all features independantly and equally, thus they benefit strongly from feature selection.

Validation Accuracy Compared





Results and Discussion

Analysis of the charts produced using PCA provide enough reason to exclude investigating unsupervised clustering since there isn't distinct enough clusters that correspond the the target variable present in this data-set. Random Forest Classifiers appear to be the best choice for classifying this data-set however, Logistic Regression and Decision Trees and the very slight decrease in accuracy make these classifiers a superb choice if increased sample size and/or processing time became an issue (Pranckevicius and Marcinkevičius (2017)).

The results of these classifiers is very promising and shows there is a strong case for their use in the search for Pulsars. The use of Random Forests allows ranking of feature importances based on the increase provided to accuracy of the classifier. It is clear that the Integrated Pulse Kurtosis, DM/SNR Skewness, DM/SNR Mean and integrated Pulse Skewness are the four most important factors in this dataset.

The resource efficiency of Logistic Regression classifiers could mean this Pulsar detection algorithm could be built directly into radio telescope firmware allowing

much more effective and automated sky surveys to take place in the future. Interestingly Logistic Regression performed very well giving its significant less resource usage when compared to the other algorithms tested (Pranckevicius and Marcinkevičius (2017)).

Conclusions

This paper shows there is a strong case for Machine Learning (ML) to be used in the search for "Cosmic Lighthouses" (Pulsars). This has huge implications for the Astrophysics community since it proves that Pulsars are seperate from regular samples to a high degree.

This paper did not address the class imbalance problem in this dataset however scored state of the art scores for classification of these pulsars. This head room for improvement warrants serious consideration for future works.

Feature selection techniques were not evaluated in this paper even though features were ranked on importance. The Naive Bayes Classifier performed the worst on this task however it is the authors belief that there is much room for improvement when given only a subset of the features. This is the work of a future paper on Machine Learnings effectivity on classifying Pulsars.

The work in this paper is limited by the level of preprocessing required for producing the dataset however, the preprocessing is standard and usual for this type of data in the astrophysics community.

References

Bates, S. D., M. Bailes, N. D. R. Bhat, M. Burgay, S. Burke-Spolaor, N. D'Amico, A. Jameson, et al. 2011. "The High Time Resolution Universe Pulsar Survey - Ii. Discovery of Five Millisecond Pulsars" 416 (4): 2455–64. https://doi.org/10.1111/j.1365-2966.2011.18416.x.

Breiman, Leo. 2001. "Random Forests." *Mach. Learn.* 45 (1): 5–32. https://doi.org/10.1023/A:1010933404324.

Liaw, Andy, and Matthew Wiener. 2002. "Classification and Regression by randomForest." *R News* 2 (3): 18–22. https://CRAN.R-project.org/doc/Rnews/.

Majka, Michal. 2019. *Naivebayes: High Performance Implementation of the Naive Bayes Algorithm in R.* https://CRAN.R-project.org/package=naivebayes.

McCullagh, P., and J. A. Nelder. 1989. *Generalized Linear Models, Second Edition.* Chapman and Hall/Crc Monographs on Statistics and Applied Probability Series. Chapman & Hall. http://books.google.com/books?id=h9kFH2/_FfBkC.

Pranckevicius, Tomas, and Virginijus Marcinkevičius. 2017. "Comparison of Naive Bayes, Random Forest, Decision Tree, Support Vector Machines, and Logistic Regression Classifiers for Text Reviews Classification." *Baltic Journal of Modern Computing* 5 (January). https://doi.org/10.22364/bjmc.2017.5.2.05.

Quinlan, J. R. 1986. "Induction of Decision Trees." *Mach. Learn.* 1 (1): 81–106. https://doi.org/10.1023/A:1022643204877.

R Core Team. 2019. *R: A Language and Environment for Statistical Computing.* Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Ripley, Brian. 2019. *Tree: Classification and Regression Trees.* https://CRAN.R-project.org/package=tree.

Soetaert, Karline. 2017. *Plot3D: Plotting Multi-Dimensional Data.* https://CRAN.R-project.org/package=plot3D.

Technology, Swinburne University of. 2007a. "Pulsar Dispersion Measure." 2007. https://astronomy.swin.edu.au/cms/astro/cosmos/p/pulsar+dispersion+measure.

———. 2007b. "Pulsar Timing." 2007. http://astronomy.swin.edu.au/cosmos/P/Pulsar+Timing.

Y Potekhin, Aleksandr. 2010. "The physics of neutron stars." *Physics Uspekhi* 53 (12): 1235–56. https://doi.org/10.3367/UFNe.0180.201012c.1279.