Pulsar Classification - Proposal

Anil Kumar Pallekonda

Contents

Abstract

Classifying a pulsar from noise is quite challenging. To automate the process of identifying pulsars from the observed emission patterns through the radio telescopes, we are going to build a classification model which classifies the input into pulsars and noise groups.

Proposal

About Data

The data set consists of pulsar candidate profile information which is collected through High Time Resolution Universe Survey(HTRU2). A neutron star which produces radio emissions and can be identified from the earth is called pulsar. Pulsars are of scientific interest as probes of their interstellar medium, space-time, and states of matter.
In general, pulsars rotate; so their emission beams sweep across the sky. As pulsars rotate and produce radio emissions, we can see patterns periodically. These periodic patterns can be identified using radio telescopes. Each pulsar has different patterns in emitting radio signals, and this pattern will change with each rotation. Identifying right/possible signal pattern is known as the candidate of the pulsar. The candidate is the average over many pulsar rotations determined by the observation length(time). In practice, the real pulsar signal is hard to find as patterns are detected by the RFI(Radio Frequency Interface).There will also be noise in the signals. Identifying a right pulsar pattern manually is tough and consumes much time. Machine learning tools will help automate the process of classifying the observed patterns into noise or pulsar emissions.

Data Description

This data set has nine variables and 17,898 observations in it. First four variables describe the longitude resolved version of signal that has been averaged in both time and frequency. The next four variables describe averaged time and frequency measures of the DM-SNR curve. This data set is already classified into two groups; those are positive and negative. Positive means the identified pattern is pulsar and negative means the identified pattern is noise. This dataset consists of 1,639 positive and 16,259 negative examples. Around 90% of the data is classified as noise, and only 10% of the data is classified as positive. All the data in this data set is annotated by humans.

Question of Interest

The goal of this project is to provide a machine learning tool which can differentiate pulsars and noise based on candidate profile measures. When we consider identifying a pulsar is chain process using their emissions, this machine learning model will suit at the end of the chain process. As humans annotated this data, expecting accuracy for the predictive model is more than 90%.

Methods Planning to Use

Before building a model, we should understand the data well. As a part of this process, first, we would like to perform the data preparation which consists of identifying & handling missing values and outliers. Next, we need to perform range normalization across the data set if variable scaling measures are different. Apart from these, as part of the exploratory analysis, we would like to achieve few tasks like finding central tendency of the data, underlying distribution of the data, finding out if there is any correlation between the variables, identifying the relevant variables, etc. If the data set contains imbalanced data, we should use sampling techniques to balance the data.

Once the exploratory analysis is completed, we would build a classification model which can classify the given input into pulsar or noise groups. To accomplish this task, we will divide the given data into two sets, i.e., training set which is used to train the model and test set which is used to validate the model performance. To check the accuracy of the model, we will be using any of the available methods, i.e., confusion matrix, precision, recall, f-score, area under the curve, etc.
As part of this modeling task, we would like to develop models using naïve Bayes classifier and decision trees. The accuracy of the model will be measured using the above tools and using this we will compare the performance of two models. The most accurate model will be selected for a universal survey to classify the pulsars from noise.

Data Preparation

The data set has 17,988 observations with nine variables. When we look at the data set, 90.8% data is classified as noise, and 9.2% data is classified as pulsars. This data is imbalanced, and if we use this data to create a model, we will end up creating biased classification using developed model. To avoid this, we will use sampling methods to balance the imbalance dataset. We can use any of the methods i.e. "Under-Sampling," "Over Sampling," "Synthetic Data Generation" and "Cost-Sensitive Learning." In our scenario, we will be using "Synthetic data Generation" approach to balancing the data.

Data Loading

I have used `read_csv` function from `readr` package to load the data into R. I have mentioned that `class` column should read; as a factor datatype with the values 0 and 1.

```
HTRU_2 <- read_csv("D:/Academics/ML1Project/HTRU2/HTRU_2.csv",
    col_types = cols(Class = col_factor(levels = c("0", "1"))),
    na = "NA")
```

Missing Values

When we check for the missing values in the dataset, there are no missing values in any of the nine variables.

```
sapply(unique(HTRU_2), function(x) any(is.na(x)))
```

```
##    MeanIG     SDIGP     EKIGP    SkeIGP MeanDMSNR   SDDMSNR   EKDMSNR
##     FALSE     FALSE     FALSE     FALSE     FALSE     FALSE     FALSE
##  SkeDMSNR     Class
##     FALSE     FALSE
```

Shuffling the data

As data set is sorted with `Class` variable, before applying the above-mentioned sampling techniques, we are going to shuffle all the observations to select random values for the training set and test set. 80% of the given data is split into a training set, and rest of the data is treated as the test set. When we look at the proportion of the pulsars and noises, the pulsars are only 9.04%, and rest of the data is noise. At the same time when we look at the test set, 90.8% of data classified as noise and rest of the data is classified as pulsars.

```
HTshuf <- HTRU_2[sample(nrow(HTRU_2)), ]

# creating test, validation and train sets
size <- round(nrow(HTshuf) * 0.8)
```

```
testset <- HTshuf[size:nrow(HTshuf), ]
trainset <- HTshuf[1:size, ]

str(trainset)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    14318 obs. of  9 variables:
##  $ MeanIG   : num  103.6 120.4 106 110.4 98.4 ...
##  $ SDIGP    : num  45.2 46.9 44.2 45.8 40.2 ...
##  $ EKIGP    : num  0.552 0.107 -0.225 0.379 0.625 ...
##  $ SkeIGP   : num  0.778 0.373 0.337 0.486 1.169 ...
##  $ MeanDMSNR: num  21.821 3.787 1.217 0.762 1.96 ...
##  $ SDDMSNR  : num  51.2 25.1 11.8 10.3 13.7 ...
##  $ EKDMSNR  : num  2.4 7.03 13.34 18.88 10.41 ...
##  $ SkeDMSNR : num  4.76 50.68 222.52 405.29 137.25 ...
##  $ Class    : Factor w/ 2 levels "0","1": 2 1 1 1 1 1 1 1 2 1 ...
```

```
str(testset)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    3581 obs. of  9 variables:
##  $ MeanIG   : num  116 113 103 132 105 ...
##  $ SDIGP    : num  40.6 41.7 44.1 37.5 43.5 ...
##  $ EKIGP    : num  0.0907 0.4531 0.5343 0.3215 0.4264 ...
##  $ SkeIGP   : num  1.244 1.129 0.688 1.135 0.876 ...
##  $ MeanDMSNR: num  0.987 5.154 2.796 9.81 5.371 ...
##  $ SDDMSNR  : num  9.32 29.27 18.11 35.19 25.13 ...
##  $ EKDMSNR  : num  19.04 5.97 8.64 3.64 5.24 ...
##  $ SkeDMSNR : num  471.7 35.7 90 12.5 29.7 ...
##  $ Class    : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

```
# identifying the noise and pulsar proportions
prop.table(table(trainset$Class))
```

```
##
##          0          1
## 0.90648135 0.09351865
```

```
prop.table(table(testset$Class))
```

```
##
##          0          1
## 0.91622452 0.08377548
```

Converting Imbalanced data to Balanced data

In the previous section, we have identified that information in the pulsar dataset is imbalanced. To avoid biased prediction, we can balance the data set using sampling rules. We are using `Synthetic Data Generation` method to reconcile the data. To accomplish that, we are using `ROSE` function from `ROSE` library.

After applying sampling techniques, when we look at the proportion of training set and test values, we can say that data set is now pretty much balanced.

```
trainRose <- ROSE(Class ~ ., data = trainset, seed = 1)$data
testRose <- ROSE(Class ~ ., data = testset, seed = 1)$data

str(trainRose)
```

```
## 'data.frame':    14318 obs. of  9 variables:
```

```
##  $ MeanIG   : num  133.6 131.9 139.2 95.5 129.7 ...
##  $ SDIGP    : num  48.6 49.8 43.5 40.8 49.4 ...
##  $ EKIGP    : num  0.275 0.15 -0.183 0.396 0.195 ...
##  $ SkeIGP   : num  0.3548 0.0805 -0.1692 0.8896 -0.0172 ...
##  $ MeanDMSNR: num  7.55 8.91 10.8 8.74 8.33 ...
##  $ SDDMSNR  : num  14.3 45.7 36.7 11 20.1 ...
##  $ EKDMSNR  : num  10.05 6.63 6.38 11.11 8.62 ...
##  $ SkeDMSNR : num  70.5 36.3 50.3 85.5 140.3 ...
##  $ Class    : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

```
str(testRose)
```

```
## 'data.frame':    3581 obs. of  9 variables:
##  $ MeanIG   : num  95.9 108.8 104.1 117.8 133.5 ...
##  $ SDIGP    : num  55.6 45.5 37.3 65.6 46.7 ...
##  $ EKIGP    : num  0.45563 0.51614 0.43292 0.00276 0.34197 ...
##  $ SkeIGP   : num  -0.40888 0.28321 1.67682 -0.9978 -0.00639 ...
##  $ MeanDMSNR: num  6.74 4.74 -5.12 -11.28 16.23 ...
##  $ SDDMSNR  : num  18.1 19.1 19.8 24.8 21.1 ...
##  $ EKDMSNR  : num  7.84 13.45 12.05 7.26 6.04 ...
##  $ SkeDMSNR : num  102.4 147.7 105.7 69.7 37.9 ...
##  $ Class    : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

```
# Identifying the noise and pulsar proportions
prop.table(table(trainRose$Class))
```

```
##
##         0         1
## 0.5032826 0.4967174
```

```
prop.table(table(testRose$Class))
```

```
##
##         0         1
## 0.5143815 0.4856185
```

Exploratory Analysis

As a part of the exploratory analysis, first, we will look at basic statistics of the data set. After that, we will see the underlying distributions and correlations among the variables.

Basic Statistics

Basic statistics like mean, median, standard deviation, min, max, etc. can be found below. First four variables are related to the pattern of radio emission, and the next four variables are related DM-SNR curve. The factor variable is a classified variable which represents observed pattern as a pulsar or noise. In this data set, the pulsar is denoted as 1, and noise is denoted as 0.

```
str(trainRose)
```

```
## 'data.frame':    14318 obs. of  9 variables:
##  $ MeanIG   : num  133.6 131.9 139.2 95.5 129.7 ...
##  $ SDIGP    : num  48.6 49.8 43.5 40.8 49.4 ...
##  $ EKIGP    : num  0.275 0.15 -0.183 0.396 0.195 ...
##  $ SkeIGP   : num  0.3548 0.0805 -0.1692 0.8896 -0.0172 ...
```

```
##  $ MeanDMSNR: num  7.55 8.91 10.8 8.74 8.33 ...
##  $ SDDMSNR  : num  14.3 45.7 36.7 11 20.1 ...
##  $ EKDMSNR  : num  10.05 6.63 6.38 11.11 8.62 ...
##  $ SkeDMSNR : num  70.5 36.3 50.3 85.5 140.3 ...
##  $ Class    : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

```
str(testRose)
```

```
## 'data.frame':    3581 obs. of  9 variables:
##  $ MeanIG   : num  95.9 108.8 104.1 117.8 133.5 ...
##  $ SDIGP    : num  55.6 45.5 37.3 65.6 46.7 ...
##  $ EKIGP    : num  0.45563 0.51614 0.43292 0.00276 0.34197 ...
##  $ SkeIGP   : num  -0.40888 0.28321 1.67682 -0.9978 -0.00639 ...
##  $ MeanDMSNR: num  6.74 4.74 -5.12 -11.28 16.23 ...
##  $ SDDMSNR  : num  18.1 19.1 19.8 24.8 21.1 ...
##  $ EKDMSNR  : num  7.84 13.45 12.05 7.26 6.04 ...
##  $ SkeDMSNR : num  102.4 147.7 105.7 69.7 37.9 ...
##  $ Class    : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

```
summary(trainRose)
```

```
##      MeanIG           SDIGP           EKIGP             SkeIGP
##  Min.   :-43.02   Min.   :14.55   Min.   :-3.2326   Min.   :-24.1251
##  1st Qu.: 54.14   1st Qu.:36.58   1st Qu.: 0.1406   1st Qu.: -0.1133
##  Median : 96.57   Median :43.56   Median : 0.6001   Median :  1.0754
##  Mean   : 86.91   Mean   :43.04   Mean   : 1.6734   Mean   :  8.0130
##  3rd Qu.:118.90   3rd Qu.:49.36   3rd Qu.: 3.0189   3rd Qu.: 12.7998
##  Max.   :202.08   Max.   :99.14   Max.   :10.0119   Max.   : 81.9938
##    MeanDMSNR          SDDMSNR          EKDMSNR          SkeDMSNR
##  Min.   :-76.7541   Min.   :-10.85   Min.   :-6.109   Min.   :-160.312
##  1st Qu.: -0.1876   1st Qu.: 17.60   1st Qu.: 1.769   1st Qu.:   1.897
##  Median : 12.2973   Median : 33.35   Median : 5.233   Median :  35.862
##  Mean   : 29.2533   Mean   : 39.83   Mean   : 5.860   Mean   :  66.177
##  3rd Qu.: 45.8530   3rd Qu.: 61.82   3rd Qu.: 9.305   3rd Qu.: 104.681
##  Max.   :242.5332   Max.   :128.29   Max.   :36.548   Max.   :1173.364
##  Class
##  0:7206
##  1:7112
##
##
##
##
```

```
summary(testRose)
```

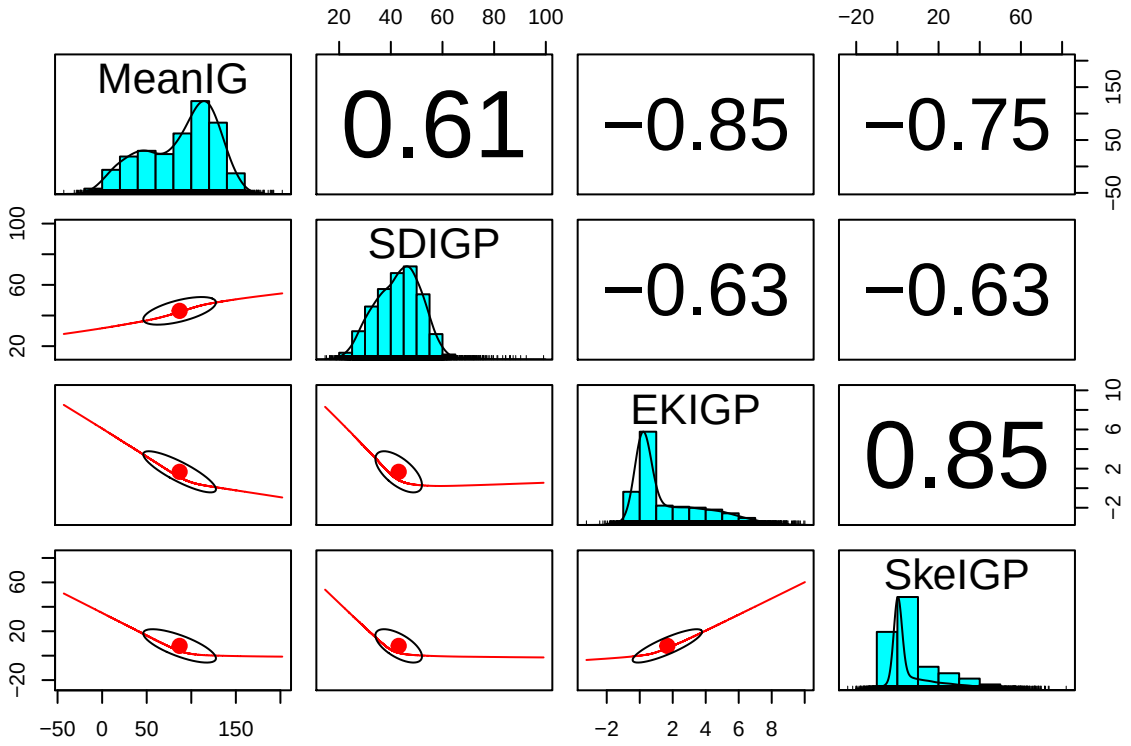
```
##      MeanIG           SDIGP           EKIGP             SkeIGP
##  Min.   :-30.94   Min.   :17.79   Min.   :-2.7018   Min.   :-20.83122
##  1st Qu.: 59.36   1st Qu.:36.79   1st Qu.: 0.1363   1st Qu.: -0.09561
##  Median : 97.06   Median :43.80   Median : 0.5547   Median :  0.99228
##  Mean   : 87.87   Mean   :43.25   Mean   : 1.5903   Mean   :  7.64225
##  3rd Qu.:119.02   3rd Qu.:49.50   3rd Qu.: 2.7575   3rd Qu.: 12.32253
##  Max.   :190.23   Max.   :77.70   Max.   :10.2942   Max.   : 66.11788
##    MeanDMSNR          SDDMSNR          EKDMSNR          SkeDMSNR
##  Min.   :-74.534   Min.   :-24.14   Min.   :-4.465   Min.   :-147.966
##  1st Qu.: -1.008   1st Qu.: 17.83   1st Qu.: 1.783   1st Qu.:   2.634
##  Median : 11.982   Median : 33.59   Median : 5.193   Median :  31.989
```

```
##  Mean   : 28.477   Mean   : 39.96   Mean   : 5.752   Mean   :  65.387
##  3rd Qu.: 45.313   3rd Qu.: 61.82   3rd Qu.: 9.172   3rd Qu.: 104.274
##  Max.   :244.045   Max.   :122.45   Max.   :31.007   Max.   :1155.268
##  Class
##  0:1842
##  1:1739
##
##
##
##
```

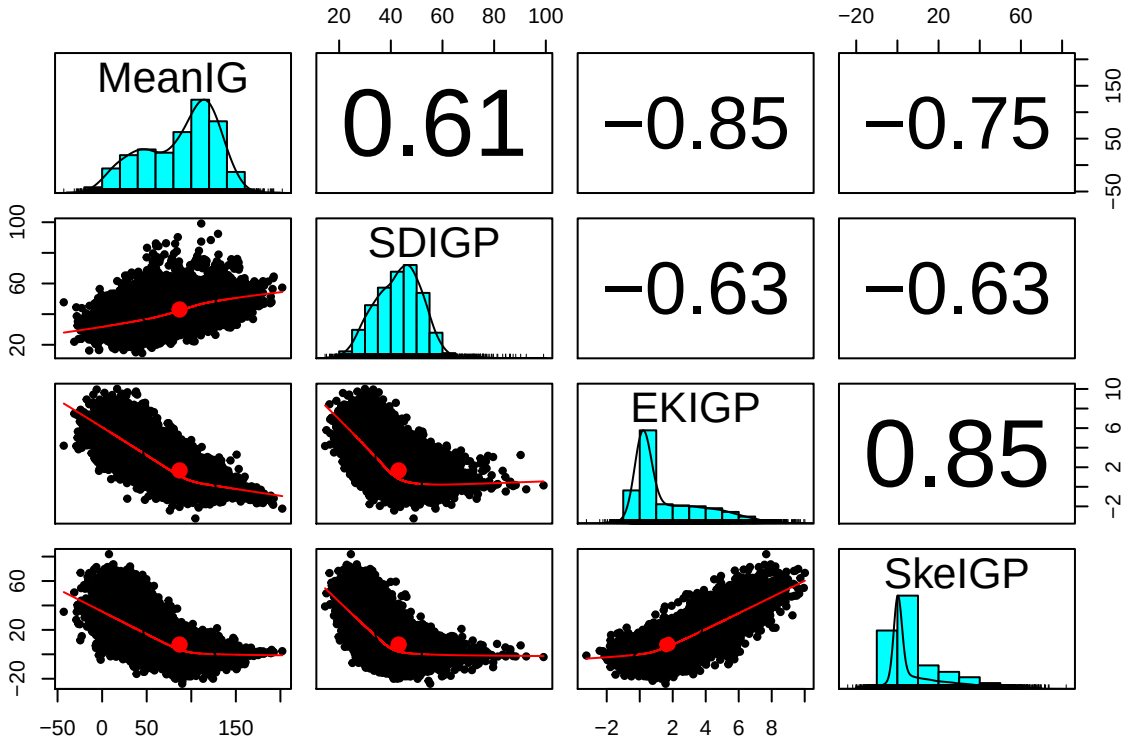
Visualizations

To check the distributions and correlations in a single graph, `paris.pannel` function has been used from `psych` package.

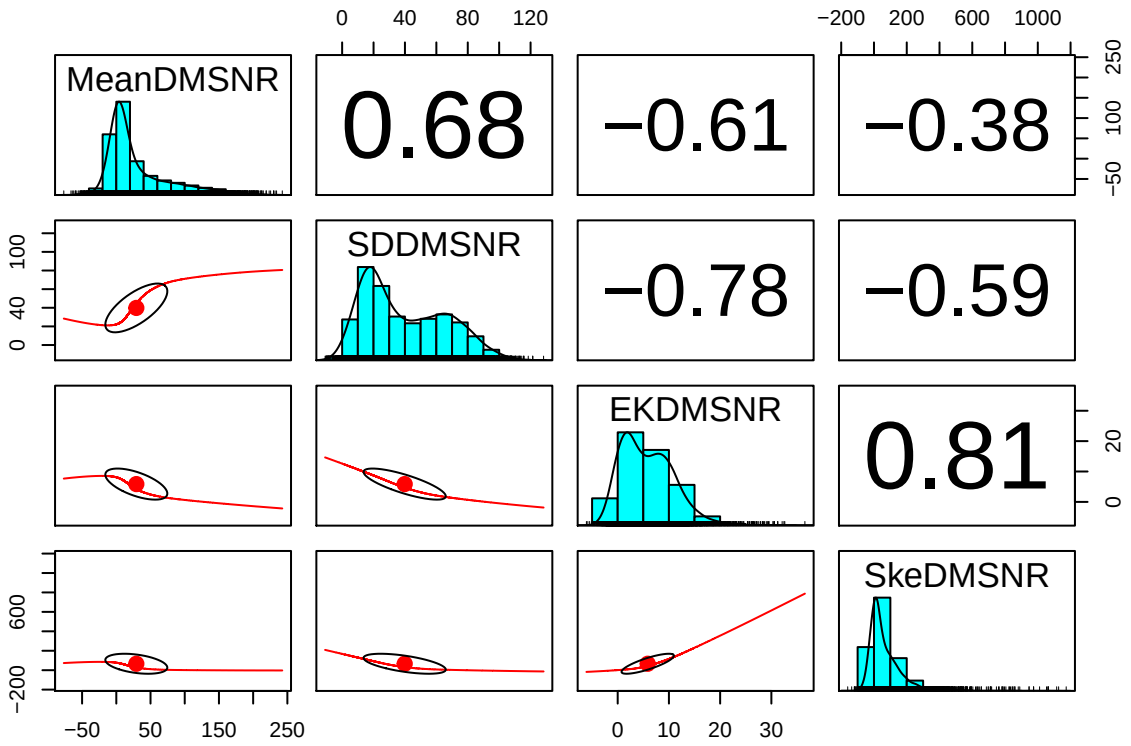
```
pairs.panels(trainRose[c("MeanIG", "SDIGP", "EKIGP", "SkeIGP")],
    show.points = FALSE)
```



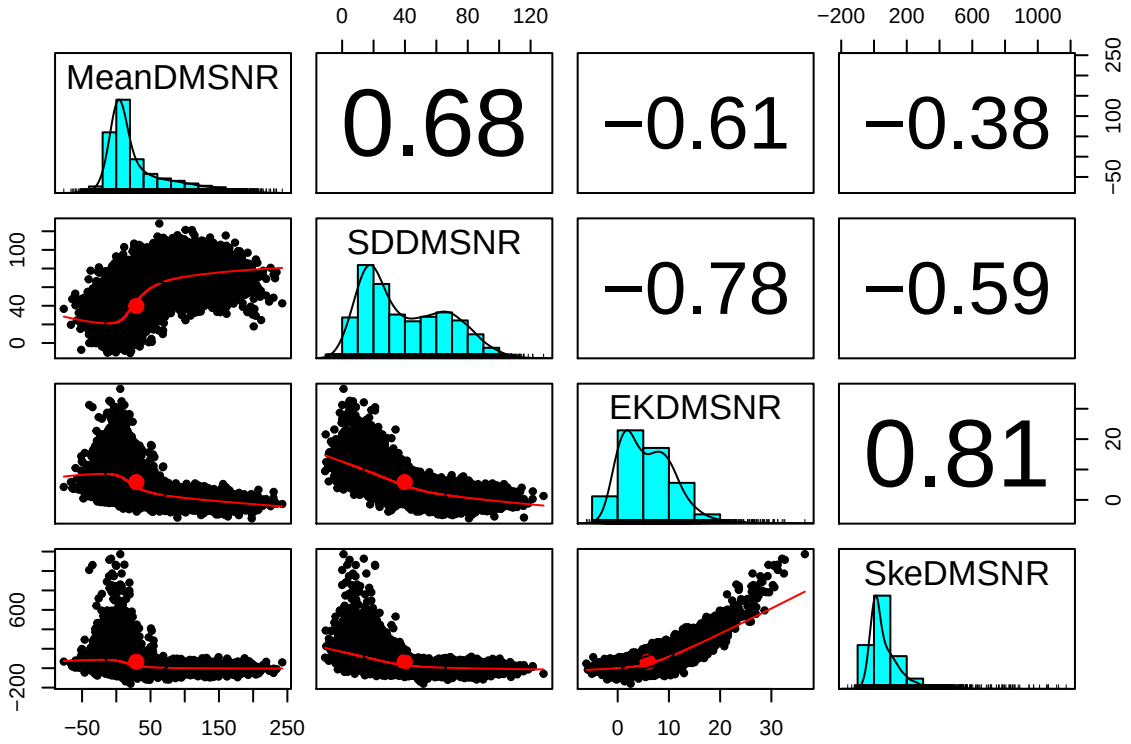

```
pairs.panels(trainRose[c("MeanIG", "SDIGP", "EKIGP", "SkeIGP")])
```




```
pairs.panels(trainRose[c("MeanDMSNR", "SDDMSNR", "EKDMSNR", "SkeDMSNR")],
    show.points = FALSE)
```




```
pairs.panels(trainRose[c("MeanDMSNR", "SDDMSNR", "EKDMSNR", "SkeDMSNR")])
```



When we look at the correlation matrix plot, we can say that there is moderate to strong relation between the candidate profile measures and a moderate relationship between DM-SNR curve metrics. In the plots, central diagonal histograms represent the distribution of each feature in the plot. In few plots, we can see the single line with dot and eclipse. The eclipse in each plot represents the strength of the relationship between x and y variables of the plot. Stretched eclipse denotes a strong correlation between x and y variables. The dot in the eclipse represents point mean value of x and y variables. The line is called **loess curve** which represents the general relationship between x and y variables.

In candidate profile measures, mean values are skewed to the left; standard deviation follows approximately normal distribution and the rest two measures are skewed to the right. The numbers in the plot represent the relationship between the two values. The positive number accounts for a positive correlation, and a negative number represents the negative correlation between two feature variables on the scale 0 to 1 where 0 denotes weak correlation, and 1 denotes strong correlation.

Limitations

As this is a proposal, few methods and processes mentioned in this document may vary as work progresses.

References

[1] M. J. Keith et al., 'The High Time Resolution Universe Pulsar Survey - I. System Configuration and Initial Discoveries',2010, Monthly Notices of the Royal Astronomical Society, vol. 409, pp. 619-627. DOI: 10.1111/j.1365-2966.2010.17325.x

[2] D. R. Lorimer and M. Kramer, 'Handbook of Pulsar Astronomy', Cambridge University Press, 2005.

[3] R. J. Lyon, 'Why Are Pulsars Hard To Find?', PhD Thesis, University of Manchester, 2016.

[4] R. J. Lyon et al., 'Fifty Years of Pulsar Candidate Selection: From simple filters to a new principled real-time classification approach', Monthly Notices of the Royal Astronomical Society 459 (1), 1104-1123, DOI: 10.1093/mnras/stw656

[5] R. P. Eatough et al., 'Selection of radio pulsar candidates using artificial neural networks', Monthly Notices of the Royal Astronomical Society, vol. 407, no. 4, pp. 2443-2450, 2010.

[6] S. D. Bates et al., 'The high time resolution universe pulsar survey vi. an artificial neural network and timing of 75 pulsars', Monthly Notices of the Royal Astronomical Society, vol. 427, no. 2, pp. 1052-1065, 2012.

[7] D. Thornton, 'The High Time Resolution Radio Sky', PhD thesis, University of Manchester, Jodrell Bank Centre for Astrophysics School of Physics and Astronomy, 2013.

[8] K. J. Lee et al., 'PEACE: pulsar evaluation algorithm for candidate extraction a software package for post-analysis processing of pulsar survey candidates', Monthly Notices of the Royal Astronomical Society, vol. 433, no. 1, pp. 688-694, 2013.

[9] V. Morello et al., 'SPINN: a straightforward machine learning solution to the pulsar candidate selection problem', Monthly Notices of the Royal Astronomical Society, vol. 443, no. 2, pp. 1651-1662, 2014.

[10] R. J. Lyon, 'PulsarFeatureLab', 2015, [Web Link].

[11] R. J. Lyon, B. W. Stappers, S. Cooper, J. M. Brooke, J. D. Knowles, Fifty Years of Pulsar Candidate Selection: From simple filters to a new principled real-time classification approach, Monthly Notices of the Royal Astronomical Society 459 (1), 1104-1123, DOI: 10.1093/mnras/stw656.

[12] R. J. Lyon, HTRU2, DOI: 10.6084/m9.figshare.3080389.v1.

[13] Analytic Vidhya Content Team, ' Practical Guide to deal with Imbalanced Classification Problems in R', Analytics Vidhya, 2016.