2020_1127_Exploratory_Data_Analysis

November 27, 2020

1 Classifying Pulsars from the High Time Resolution Universe Survey (HTRU2) - Exploratory Data Analysis

1.1 Overview & Citation

The purpose of this project is to analyze telescope data to correctly separate actual pulsar data from radio frequency noise. The dataset was retrieved from the UC Irvine Machine Learning Repository at the following link: https://archive.ics.uci.edu/ml/datasets/HTRU2#

The data comes from the High Time Resolution Universe Survey (South), known as HTRU2. The dataset was donated to the UCI Repository by Dr. Robert Lyon of The University of Manchester, United Kingdom. The two papers requested for citation in the description are listed below: * 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 * R. J. Lyon, HTRU2, DOI: 10.6084/m9.figshare.3080389.v1.

Below is an edited description of the dataset, taken from the website. The citation numbers refer to the publications listed on the website:

HTRU2 is a data set which describes a sample of pulsar candidates collected during the High Time Resolution Universe Survey (South) [1]. Pulsars are a rare type of Neutron star that produce radio emission detectable here on Earth. They are of considerable scientific interest as probes of space-time, the inter-stellar medium, and states of matter (see [2] for more uses).

As pulsars rotate, their emission beam sweeps across the sky, and when this crosses our line of sight, produces a detectable pattern of broadband radio emission. As pulsars rotate rapidly, this pattern repeats periodically. Thus pulsar search involves looking for periodic radio signals with large radio telescopes. Each pulsar produces a slightly different emission pattern, which varies slightly with each rotation (see [2] for an introduction to pulsar astrophysics to find out why). Thus a potential signal detection known as a 'candidate', is averaged over many rotations of the pulsar, as determined by the length of an observation. In the absence of additional info, each candidate could potentially describe a real pulsar. However in practice almost all detections are caused by radio frequency interference (RFI) and noise, making legitimate signals hard to find.

Machine learning tools are now being used to automatically label pulsar candidates to facilitate rapid analysis. Classification systems in particular are being widely adopted, (see [4,5,6,7,8,9]) which treat the candidate data sets as binary classification problems. Here the legitimate pulsar examples are a minority positive class, and spurious examples the majority negative class. At present multi-class labels are unavailable, given the costs associated with data annotation. The data set shared here

contains 16,259 spurious examples caused by RFI/noise, and 1,639 real pulsar examples. These examples have all been checked by human annotators.

1.2 Feature Names

The dataset was downloaded from the UC Irvine Repository as a CSV file. Column names were not provided in the original dataset. The features are listed below according to the feature names listed on the website, with the annotated column names used in this project listed in parentheses:

1. Mean of the integrated profile. (IP_Mean)
2. Standard deviation of the integrated profile. (IP_StdDev)
3. Excess kurtosis of the integrated profile. (IP_Kurtosis)
4. Skewness of the integrated profile. (IP_Skewness)
5. Mean of the DM-SNR curve. (DM_Mean)
6. Standard deviation of the DM-SNR curve. (DM_StdDev)
7. Excess kurtosis of the DM-SNR curve. (DM_Kurtosis)
8. Skewness of the DM-SNR curve. (DM_Skewness)
9. Class (Class)

The XLSX and CSV files with the added column names are provided in the GitHub repository. We will import and explore the data in the following sections.

1.3 Exploratory Data Analysis

1.3.1 Import the Relevant Libraries

[1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set()
```

1.3.2 Import & Check the Data

[3]:
```
df = pd.read_csv('2020_1125_Pulsar_Data.csv')
pulsar_data = df.copy()
```

[4]:
```
pulsar_data.head()
```

[4]:

	IP_Mean	IP_StdDev	IP_Kurtosis	IP_Skewness	DM_Mean	DM_StdDev	\
0	140.562500	55.683782	-0.234571	-0.699648	3.199833	19.110426	
1	102.507812	58.882430	0.465318	-0.515088	1.677258	14.860146	
2	103.015625	39.341649	0.323328	1.051164	3.121237	21.744669	
3	136.750000	57.178449	-0.068415	-0.636238	3.642977	20.959280	
4	88.726562	40.672225	0.600866	1.123492	1.178930	11.468720	

	DM_Kurtosis	DM_Skewness	Class
0	7.975532	74.242225	0

```
1     10.576487   127.393580      0
2      7.735822    63.171909      0
3      6.896499    53.593661      0
4     14.269573   252.567306      0
```

[9]:
```
# Showing only data for input variables, since the statistics for binary␣
 ↪variables in Class are meaningless
pulsar_data.drop('Class',axis=1,inplace=False).describe()
```

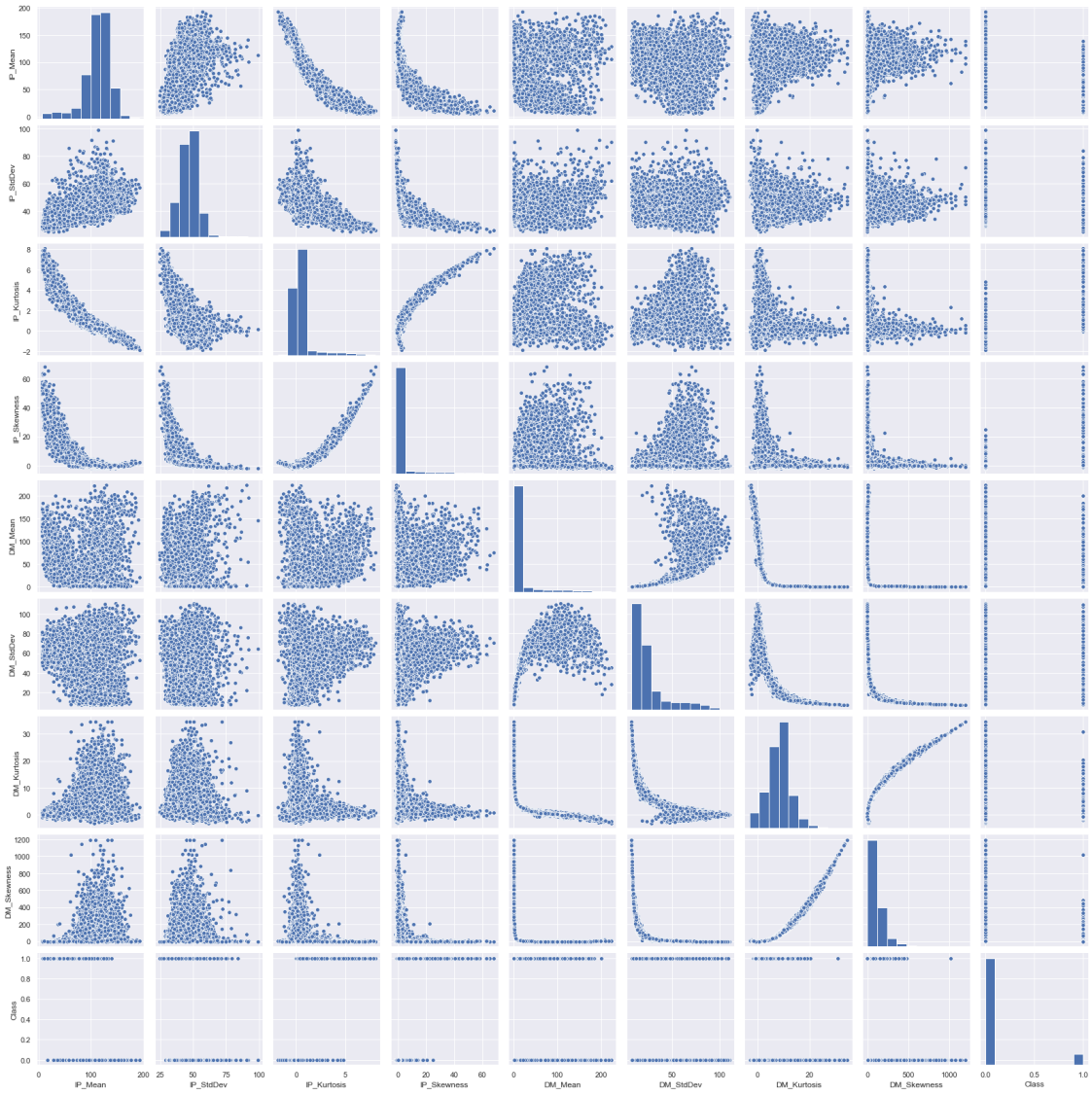
[9]:
```
            IP_Mean     IP_StdDev   IP_Kurtosis   IP_Skewness       DM_Mean  \
count  17898.000000  17898.000000  17898.000000  17898.000000  17898.000000
mean     111.079968     46.549532      0.477857      1.770279     12.614400
std       25.652935      6.843189      1.064040      6.167913     29.472897
min        5.812500     24.772042     -1.876011     -1.791886      0.213211
25%      100.929688     42.376018      0.027098     -0.188572      1.923077
50%      115.078125     46.947479      0.223240      0.198710      2.801839
75%      127.085938     51.023202      0.473325      0.927783      5.464256
max      192.617188     98.778911      8.069522     68.101622    223.392140

          DM_StdDev   DM_Kurtosis   DM_Skewness
count  17898.000000  17898.000000  17898.000000
mean      26.326515      8.303556    104.857709
std       19.470572      4.506092    106.514540
min        7.370432     -3.139270     -1.976976
25%       14.437332      5.781506     34.960504
50%       18.461316      8.433515     83.064556
75%       28.428104     10.702959    139.309331
max      110.642211     34.539844   1191.000837
```

1.3.3 Data Visualization

The visualization below shows paired plots for every input and outut variable.

[10]:
```
sns.pairplot(pulsar_data)
```

[10]:
```
<seaborn.axisgrid.PairGrid at 0x7fda9cf54df0>
```



1.4 Modeling Recommendations

This dataset poses a binary classification problem. All input values are numerical, and the output variable is categorical. Since there are no null values in the dataset, no additional preprocessing is requred prior to modeling. The following classification models should be explored and compared for performance: * Multiple Logistic Regression * Decision Tree Classification * Random Forest Classification * Support Vector Machines (SVM) * Deep Artificial Neural Networks (DNN)