Assignment 6: Support Vector Machine

Prajwal Dinesh Sahu
ME18B114
Dept. of Mechanical Engineering
Indian Institute of Technology, Madras
me18b114@smail.iitm.ac.in

***Abstract*—This paper gives a brief introduction of Support Vector Machines and the basic mathematical theory behind it. We also look at some metrics to evaluate the performance of our model. Further, we apply the classifier on the Neutron Star Database, containing the various parameters of the star emissions. The prediction task is to classify the star as Pulsars and not Pulsars. Through exploratory data analysis and support vector machines, we investigate trends in data to predict what characteristics make a star a pulsar. Then we evaluate our model using confusion matrix and F1 score. Our findings reveal some interesting trends.**

I. Introduction

There are multiple types of algorithm methods used in machine learning. One such popular and commonly used supervised machine learning method is Support Vector Machine.

The SVM algorithm's purpose is to find the optimum line or decision boundary for categorising n-dimensional space into classes so that additional data points may be readily placed in the proper category in the future. This best decision boundary is called a hyperplane. The extreme points/vectors that assist create the hyperplane are chosen via SVM. Support vectors are the extreme instances, and the method is called a Support Vector Machine.

In this paper, we apply the SVM algorithm to predict whether a neutron star is a pulsar star or not based on the various given features of the star. The features are obtained from the integrated pulse profile and the DM-SNR curve.

The paper will first give a brief explanation of support vector machines. Then, we will try to solve the pulsar prediction and model our data-set by
1. Cleaning the data.
2. Exploratory and visual analysis.
3. Scaling the data.
4. Checking for Correlation.
5. Training the model.
6. Evaluating the model.
7. Interpreting the results.

II. Support Vector Machines Theory

A. *Classification*

Classification is a process of categorizing a given set of data into classes. Predicting a qualitative response for an observation can be referred to as classifying that observation, since it involves assigning the observation to a category, label or class.

Many real world problems require classification rather than regression. In the classification setting we have a set of training observations (x1, y1), . . . , (xn, yn) that we can use to build a classifier.

Regression is not useful in problems like if we are trying to predict the medical condition of a patient in the hospital and we have 3 possible diagnoses- stroke, drug overdose, and epileptic seizure based on the symptoms of the patient. We can label these diseases at 1,2 and 3. In this case, even if we create a metric to accurately convert the symptoms to numeric data, regression won't be able to classify the data as it cannot give the output as 1,2 or 3. Alternatively, we can create a threshold that if predicted value is between 0.5 to 1.5, we will assign it to label 1. This brings us to the basic idea behind classification algorithms which can give qualitative response.

B. *Support Vector Machines*

*1) **Hyperplanes**:* A hyperplane is a subspace whose dimension is one less than that of its ambient space. For example, if a space is 3-dimensional then its hyperplanes are the 2-dimensional planes, while if the space is 2-dimensional, its hyperplanes are the 1-dimensional lines.

In 2D Cartesian coordinates, such a hyperplane can be described with a single linear equation of the following form (where at least one of the a_i is non-zero and b is an arbitrary constant):

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n = b. \tag{1}$$

Any hyperplane of a Euclidean space has exactly two unit normal vectors.

Affine hyperplanes are used to define decision boundaries in many machine learning algorithms such as linear-combination (oblique) decision trees, and perceptrons.

*2) **Linear SVM:*** We are given a training dataset of n points of the form $(x_1, y_1)....(x_n, y_n)$ where the y_i are either 1 or 1, each indicating the class to which the point x_i belongs. We want to find the ”maximum-margin hyperplane” that divides the group of points x_i for which $y_i = 1$ from the group of points for which $y_i = -1$, which is defined so that the distance between the hyperplane and the nearest point x_i from either group is maximized.

Any hyperplane can be written as the set of points $\mathbf{x}\mathbf{x}satisfying$

$$w^T x - b = 0 \tag{2}$$

*3) **The cost function:*** To compute the SVM classifier, we have to minimize the following expression -

$$\left[\frac{1}{n}\sum_{i=1}^{n}\max\left(0, 1 - y_i(\mathbf{w}^T\mathbf{x}_i - b)\right)\right] + \lambda\|\mathbf{w}\|^2. \tag{3}$$

To solve it we can use either quadratic programming, gradient descent etc.

*4) **Quadratic Programming for SVM:*** The cost function can be rewritten as a constrained optimization problem with a differentiable objective function in the following way.

For each $i \in \{1, \ldots, n\}$ we introduce a variable $\zeta_i = \max\left(0, 1 - y_i(\mathbf{w}^T\mathbf{x}_i - b)\right)$. Note that ζ_i is the smallest nonnegative number satisfying $y_i(\mathbf{w}^T\mathbf{x}_i - b) \geq 1 - \zeta_i$.

Thus we can rewrite the optimization problem as follows

$$\text{minimize } \frac{1}{n}\sum_{i=1}^{n}\zeta_i + \lambda\|\mathbf{w}\|^2$$

$$\text{subject to } \mathrm{y}_i(\mathbf{w}^T\mathbf{x}_i - b) \geq 1 - \zeta_i \text{ and } \zeta_i \geq 0, \text{for all } i \tag{4}$$

*5) **Sub-gradient descent:*** Sub-gradient descent algorithms for the SVM work directly with the expression.

$$f(\mathbf{w}, b) = \left[\frac{1}{n}\sum_{i=1}^{n}\max\left(0, 1 - y_i(\mathbf{w}^T\mathbf{x}_i - b)\right)\right] + \lambda\|\mathbf{w}\|^2 \tag{5}$$

The traditional gradient descent (or SGD) methods can be adapted, where instead of taking a step in the direction of the function’s gradient, a step is taken in the direction of a vector selected from the function’s sub-gradient. This approach has the advantage that, for certain implementations, the number of iterations does not scale with n, the number of data points.

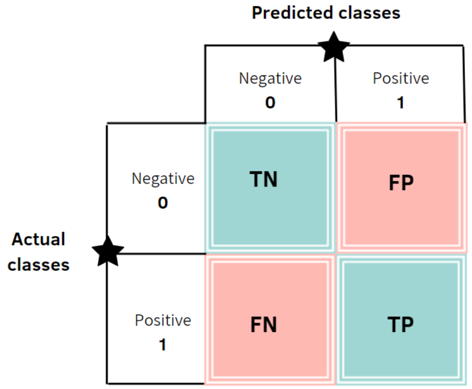
C. Advantages of SVM

- SVM has L2 Regularization feature. So, it has good generalization capabilities which prevent it from over-fitting.
- SVM can be used to solve both classification and regression problems.
- A small change to the data does not greatly affect the hyperplane and hence the SVM. So the SVM model is stable
- SVM can efficiently handle non-linear data.

D. Evaluating the model

In model evaluation, we measure the extent to which the model fits the data.

*1) **Confusion Matrix:*** A confusion matrix is a table that shows the performance of a machine learning model. It displays how many of the model’s predictions were right and how many were incorrect and which classes did the model succeed in and which did it fail in. It offers us an understanding of how the model works. If a model fails for a certain class, we may investigate it, figure out why, and try to improve the model.



		Predicted classes	
		Negative 0	Positive 1
Actual classes	Negative 0	TN	FP
	Positive 1	FN	TP

Fig. 1. Sample Confusion Matrix

There are various metrics based on confusion matrix. Some of them are listed below

*2) **Precision and Recall:*** Precision is the ratio between the True Positives and all the Positives. For our model, it will be the measure of number of cars for which we correctly identified its safety category.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive + False Positive}} \tag{6}$$

Recall is the measure of our model correctly identifying True Positives. For all the cars in the given safety label, recall tells us how many we correctly predicted.

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive + False Negative}} \tag{7}$$

*3) **F1 Score:*** F1 score is the measure which combines both precision and recall. It is calculated as the harmonic mean of precision and recall

-

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{8}$$

III. The problem

Pulsars are a rare type of Neutron star that produces radio emissions detectable here on Earth. They are of considerable scientific interest as probes of space-time, the inter-stellar medium, and states of matter. Machine learning tools are now being used to automatically label pulsar candidates to facilitate rapid analysis. The key task is to Predict if a star is a pulsar start or not. Each candidate is described by 8 continuous variables and a single class variable. The first four are simple statistics obtained from the integrated pulse profile (folded profile). This is an array of continuous variables that describe a longitude-resolved version of the signal that has been averaged in both time and frequency. The remaining four variables are similarly obtained from the DM-SNR curve.

A. *Cleaning the Data*

We check for null values in all columns and find that the around 20 percent data is missing in 'Excess kurtosis of the integrated profile' and 'Standard deviation of the DM-SNR curve' variables and 10 percent data is missing in the 'Skewness of the DM-SNR curve' variable, as shown in the figure below -

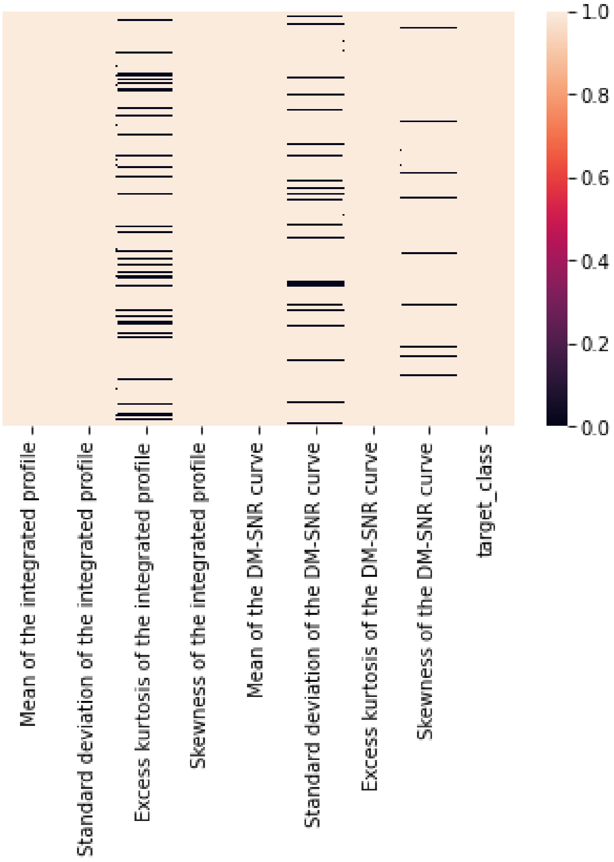


Fig. 2. Missing Values Heatmap

We impute the missing values with the respective column medians.

B. *The Variables*

- 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

C. *The Target Variable*

We have to predict whether a neutron star is a pulsar or not. In the given data, there is an high imbalance. A countplot is shown below -

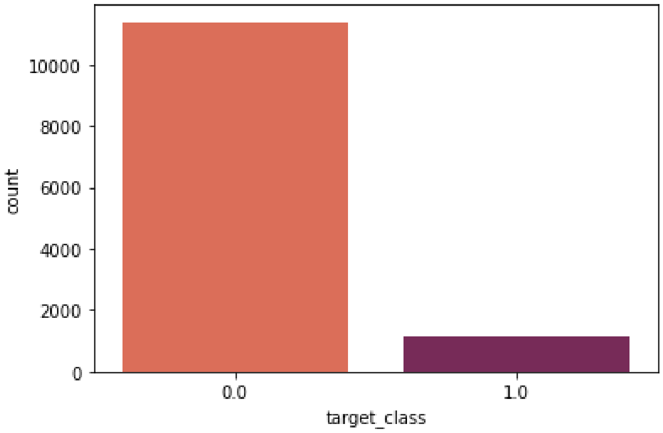


Fig. 3. Target Countplot

D. *Numerical Variables*

*1) **Mean of the integrated profile**:* This variable indicates the mean of the integrated profile of the neutron stars. To visualize its relationship with the target variable, we draw a boxplot -

From the plot, we can see that the pulsar stars' mean of the integrated profile is lesser than non-pulsars.

*2) **Standard deviation of the integrated profile**:* This variable indicates the std. deviation of the integrated profile of the neutron stars. To visualize its relationship with the target variable, we draw a boxplot -

From the plot, we can see that the pulsar stars' std. deviation of the integrated profile is lesser than non-pulsars. This implies the integrated profile of pulsars is less spread out.

*3) **Excess kurtosis of the integrated profile**:* This variable indicates the excess kurtosis of the integrated profile of the neutron stars. To visualize its relationship with the target variable, we draw a boxplot -

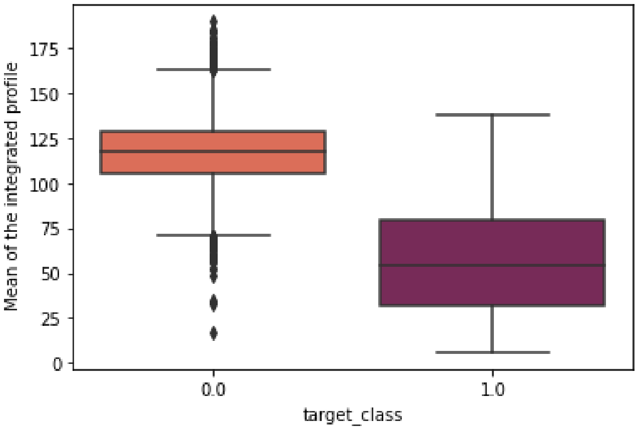


Fig. 4. Mean of the integrated profile vs target

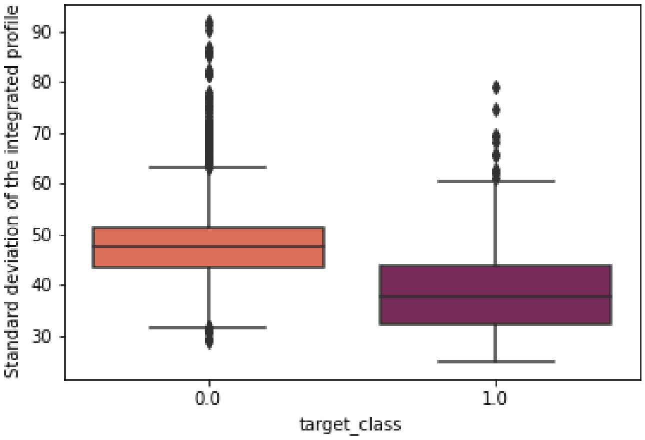


Fig. 5. Std. Deviation of the integrated profile vs target

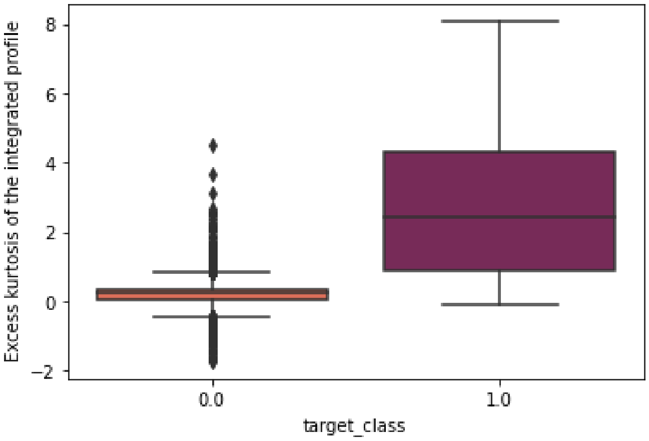


Fig. 6. Excess kurtosis of the integrated profile vs target

From the plot, we can see that the pulsar stars' excess kurtosis of the integrated profile is more than the non-pulsars.

*4) **Skewness of the integrated profile**:* This variable indicates the skewness of the integrated profile of the neutron stars. To visualize its relationship with the target variable, we draw a boxplot -

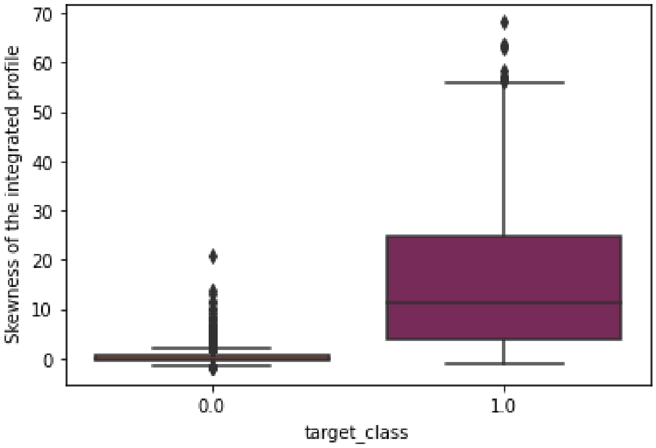


Fig. 7. Skewness of the integrated profile vs target

From the plot, we can see that the pulsar stars' skewness of the integrated profile is more than the non-pulsars.

*5) **Mean of the DM-SNR curve**:* This variable indicates the Mean of the DM-SNR curve of the neutron stars. To visualize its relationship with the target variable, we draw a boxplot -

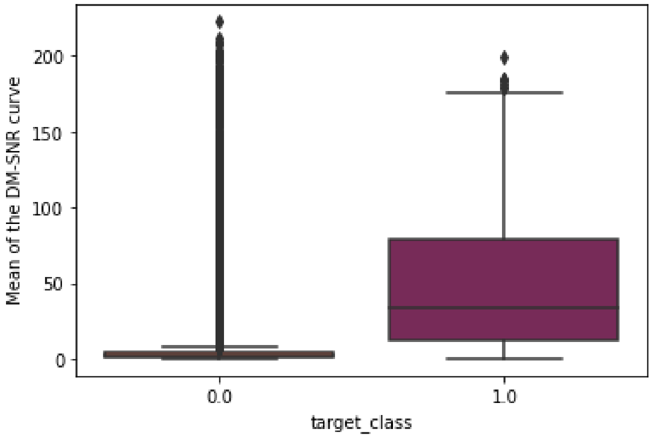


Fig. 8. Mean of the DM-SNR curve vs target

From the plot, we can see that the pulsar stars' mean of the DM-SNR curve is more than the non-pulsars.

*6) **Standard deviation of the DM-SNR curve:*** This variable indicates the std. deviation of the DM-SNR curve of the neutron stars. To visualize its relationship with the target variable, we draw a boxplot -

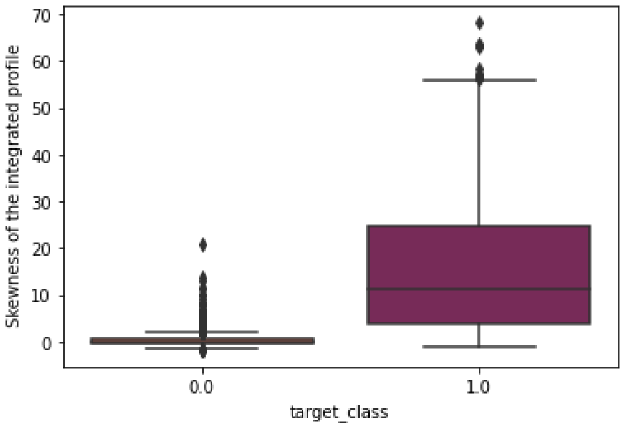


Fig. 9. Std. deviation of the DM-SNR curve vs target

From the plot, we can see that the pulsar stars' std. deviation of the DM-SNR curve is more than the non-pulsars. It implies that the DM-SNR curve is more spread out.

*7) **Excess kurtosis of the DM-SNR curve:*** This variable indicates the excess kurtosis of the DM-SNR curve of the neutron stars. To visualize its relationship with the target variable, we draw a boxplot -

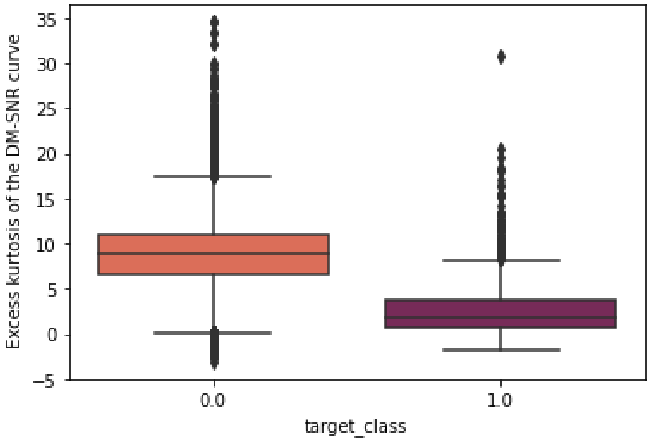


Fig. 10. Excess kurtosis of the DM-SNR curve vs target

From the plot, we can see that the pulsar stars' excess kurtosis of the DM-SNR curve is lesser than the non-pulsars.

*8) **Skewness of the DM-SNR curve:*** This variable indicates the skewness of the DM-SNR curve of the neutron stars. To visualize its relationship with the target variable, we draw a boxplot -

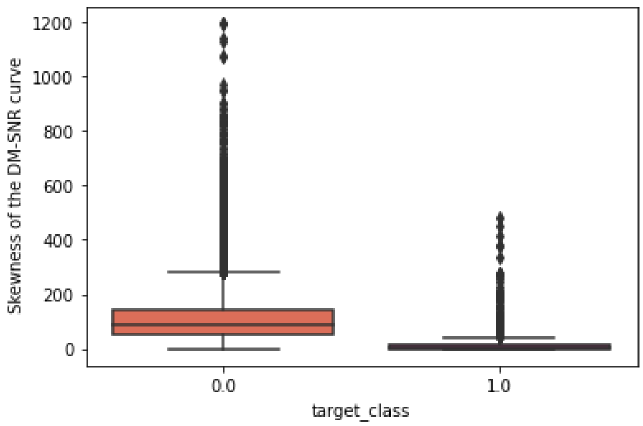


Fig. 11. Skewness of the DM-SNR curve vs target

From the plot, we can see that the pulsar stars' skewness of the DM-SNR curve is klesser than the non-pulsars.

E. Pairplot and Correlation

Through the below pairplot and correlation heatmap, we can see the values of correlation among our numeric variables-

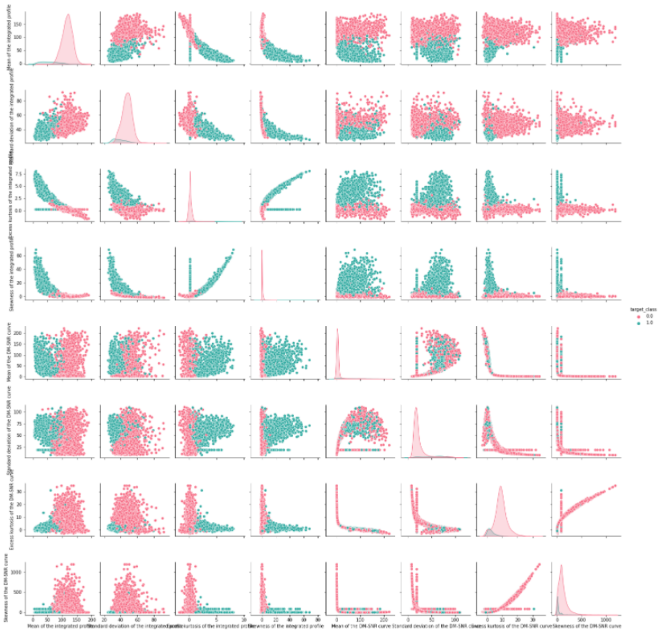


Fig. 12. Pairplot

There is a high positive correlation between following features:

Excess kurtosis of the integrated profile - Skewness of the integrated profile (0.87)

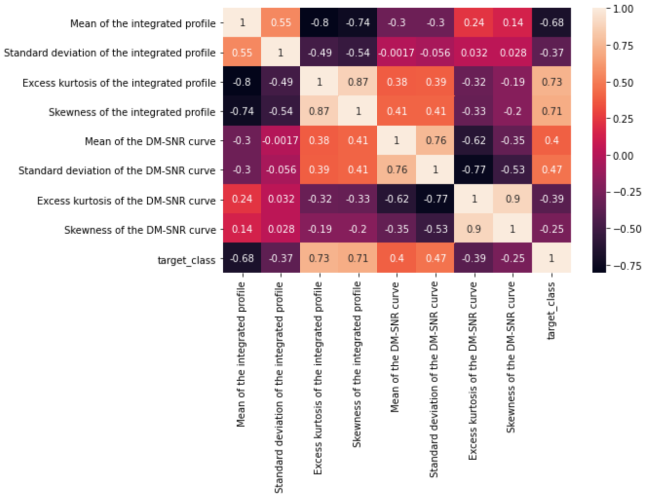


Fig. 13. Correlation Heatmap

Mean of the DM-SNR curve - Standard deviation of the DM-SNR curve(0.76)
Excess kurtosis of the DM-SNR curve - Skewness of the DM-SNR curve (0.9)

There is a high negative correlation between following features:

Mean of the integrated profile - Excess kurtosis of the integrated profile (-0.8)
Mean of the integrated profile - Skewness of the integrated profile (-0.74)
Standard deviation of the DM-SNR curve - Excess kurtosis of the DM-SNR curve (-0.77)

IV. The Model

A. *Train Test Split*

We have the following X and y for our final model, which we divide our data into training and testing data..
X = [' 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']
Y = Target class

B. *Training the Model and Grid Search*

Using scikit-learn.ensemble's SVC() function on our X and y variables, we develop a support vector classifier model.

To find out the best hyper-parameters for SVC model, we employ scikit-learn's RandomSearchCV method to do a grid search with the below given parameters -

'C': [0.01,0.1, 1, 10, 100, 1000],
'gamma': [1, 0.1, 0.01, 0.001, 0.0001],
'kernel': ['rbf'],
'tol':[0.01,0.001,0.0001],
'degree': [2,3,4,5]

C. *Best Model*

The grid search yielded the following results for the best model -
SVC(C=100, degree=5, gamma=0.001, random state=1)

D. *Predictions and Metrics*

The confusion matrix for our testing set in shown below

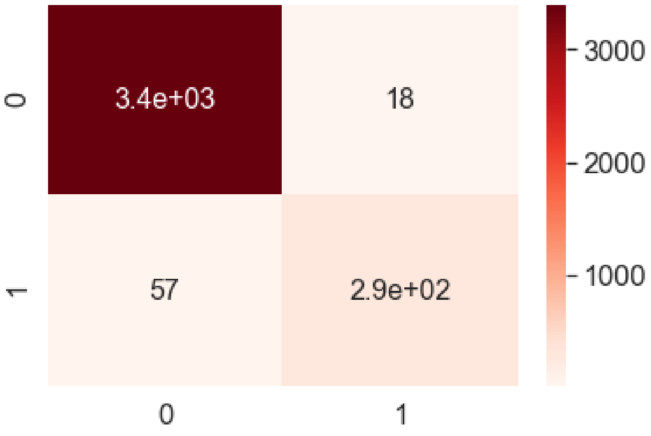
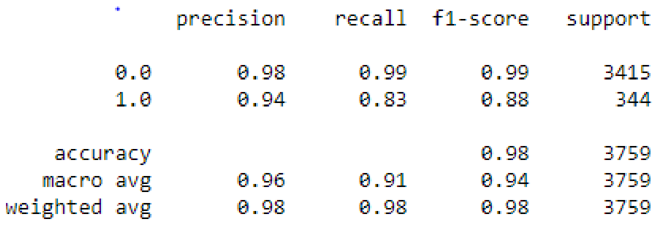


Fig. 14. Confusion Matrix for testing data

Based on the matrix, we can calculate the following (Fig 12) -



```
              precision    recall  f1-score   support

         0.0       0.98      0.99      0.99      3415
         1.0       0.94      0.83      0.88       344

    accuracy                           0.98      3759
   macro avg       0.96      0.91      0.94      3759
weighted avg       0.98      0.98      0.98      3759
```

Fig. 15. Scores for testing data

The f1 score for our best fit model after grid search is 0.98.

V. Conclusions

We were able to achieve a F1 score of 0.98 with a max depth of 11 by fitting Support Vector Classifier on our data set. The parameters for the best model were C=100, degree=5, gamma=0.001

From our observations, we can conclude the following about the significant features-

- **Mean of the integrated profile** - The mean of the integrated profile is lesser for pulsar stars than non-pulsars.
- **Excess kurtosis and skewness of the integrated profile** - For pulsars, the excess kurtosis and skewness is higher and more spread out as compared to non-pulsars.
- **Mean of the DM-SNR curve** - The mean of the integrated profile is higher and spread out for pulsar stars than non-pulsars.
- **Excess kurtosis and skewness of the DM-SNR** - For pulsars, the excess kurtosis and skewness is lower and less spread out as compared to non-pulsars.

Thus, as shown by both the qualitative and the quantitative analysis, the pulsar stars can be classified depending on its various features.

REFERENCES

[1] G. James, D. Witten, T. Hastie, and R. Tibshirani, An introduction to statistical learning: With applications in r. New York, NY: Springer, 2021.

[2] T. Hastie, J. Friedman, and R. Tisbshirani, The elements of Statistical Learning: Data Mining, Inference, and prediction. New York: Springer, 2017.

[3] Medium. 2021. Support Vector Machine — Introduction to Machine Learning Algorithms. [online] Available at: ¡https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47¿ [Accessed 26 November 2021].

[4] En.wikipedia.org. 2021. Support-vector machine - Wikipedia. [online] Available at: ¡https://en.wikipedia.org/wiki/Support-vector-machine¿ [Accessed 26 November 2021].

[5] scikit-learn. 2021. sklearn.svm.SVC. [online] Available at: ¡https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html¿ [Accessed 26 November 2021].