# A mathematical essay of support vector machines

Devansh Sanghvi
*Department Of Biotechnology*
*Indian Institute of Technology, Madras*
Chennai, India
be19b002@smail.iitm.ac.in

***Abstract*—This paper aims to develop an understanding of the mathematical properties of support vector machines and showcase its application in predicting if the given data points are pulsar stars.**

## I. Introduction

**Support Vector Machines** (SVMs in short) are machine learning algorithms that are used for classification and regression purposes. SVMs are one of the powerful machine learning algorithms for classification, regression and outlier detection purposes. An SVM classifier builds a model that assigns new data points to one of the given categories. Thus, it can be viewed as a non-probabilistic binary linear classifier.

Now, we should be familiar with some SVM terminology.

- Hyperplane: A hyperplane is a decision boundary which separates between given set of data points having different class labels. The SVM classifier separates data points using a hyperplane with the maximum amount of margin. This hyperplane is known as the maximum margin hyperplane and the linear classifier it defines is known as the maximum margin classifier.
- Support Vectors: Support vectors are the sample data points, which are closest to the hyperplane. These data points will define the separating line or hyperplane better by calculating margins.
- Margin: A margin is a separation gap between the two lines on the closest data points. It is calculated as the perpendicular distance from the line to support vectors or closest data points. In SVMs, we try to maximize this separation gap so that we get maximum margin.

The following diagram illustrates these concepts visually.

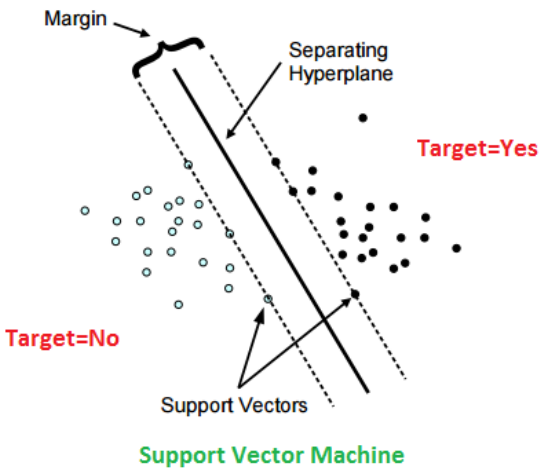


Fig. 1. Terminologies in SVMs

This paper will serve as a part of the requirements to be fulfilled for receiving credits in the course **EE4708: Data Analytics Laboratory**.

## II. Support Vector Machines

In the section above, discussed some terminologies that are used in support vector machines. We will now look at support vector machines with the terms mentioned.

In SVMs, our main objective is to select a hyperplane with the maximum possible margin between support vectors in the given dataset. SVM searches for the maximum margin hyperplane in the following 2 step process –

- Generate hyperplanes which segregates the classes in the best possible way. There are many hyperplanes that might classify the data. We should look for the best hyperplane that represents the largest separation, or margin, between the two classes.
- So, we choose the hyperplane so that distance from it to the support vectors on each side is maximized. If such a hyperplane exists, it is known as the maximum margin hyperplane and the linear classifier it defines is known as a maximum margin classifier.

The following diagram illustrates the concept of maximum margin and maximum margin hyperplane in a clear manner.

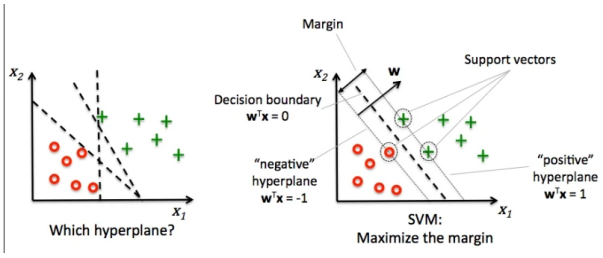


Fig. 2. Maximum margin and maximum margin hyperplane

The advantages, disadvantages and metrics for evaluation for SVMs are given below:

- **Advantages:**
  1) SVM works relatively well when there is a clear margin of separation between classes.
  2) SVM is more effective in high dimensional spaces.
  3) SVM is effective in cases where the number of dimensions is greater than the number of samples.
  4) SVM is relatively memory efficient
- **Disadvantages:**
  1) SVM algorithm is not suitable for large data sets.
  2) SVM does not perform very well when the data set has more noise i.e. target classes are overlapping.
  3) In cases where the number of features for each data point exceeds the number of training data samples, the SVM will underperform.
  4) As the support vector classifier works by putting data points, above and below the classifying hyperplane there is no probabilistic explanation for the classification.
- **Metrics for evaluation:** As for many other machine learning algorithms, we can extract the classification report for our classification. The main metrics from the classification report that are of our interest are:
  1) Precision: How many values predicted to be in a certain class are in that class
  2) Recall: How many values in each class were given the correct label
  3) F1-score: weighted average of precision and recall
  4) Accuracy: This score measures how many labels the model got right out of the total number of predictions. Accuracy is not a great measure of classifier performance when the classes are imbalanced. We need more information to understand how well the model really performed.

  The metric which you give the most importance to depends on your interest in false positives/ false negatives.

## III. Pre-processing and Exploratory Analysis on the Data

In this section, we will first pre-process the data to extract the important features in the data provided, then we will visualize the data to understand the effects of certain features and modify the data-set to optimize the results for our classification models.

### A. Pre-processing the data

The initial data-set has a training and test data-set. The training data set has 12528 samples and 8 independent features, namely: 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 target_class. Pre-processing the data-set includes (Note that the changes are made in both the training and testing data-sets):

1) Missing value assessment: There are missing data in three features. Since the number of missing values in each of the rows are very low, we replace the missing values with the respective column means.
2) Additional Variables: No features seem to depend on each other and hence, it is safe to include every feature in our model.

### B. Visualization

After preprocessing the data, we visualize the data, to better understand the correlation between different features:

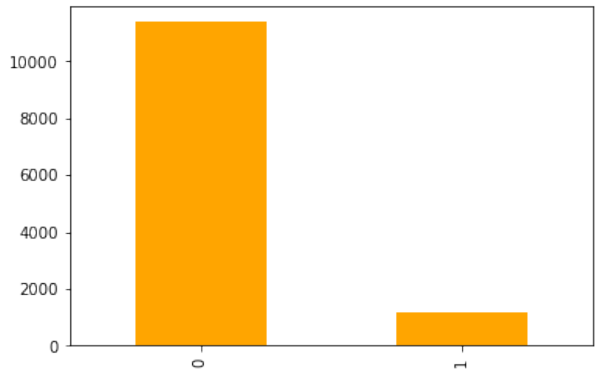


Fig. 3. Count plot of the target class

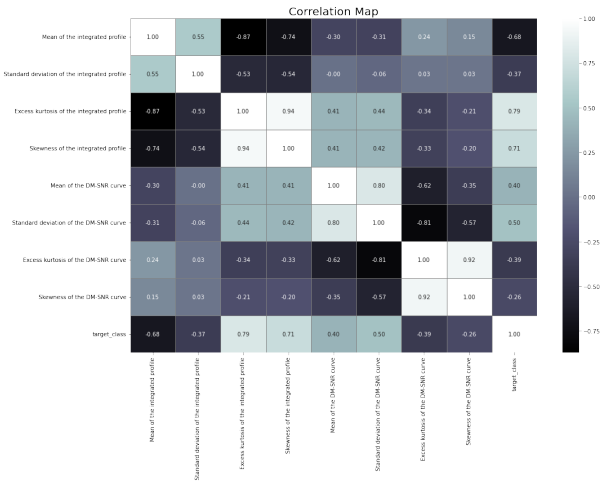


Fig. 4. Visualization of the correlation matrix for the data.

1) Exploration of the target class countplot: 11375 out of the 12528 samples are classified as not
2) Correlation matrix: Most of our columns are already related or derived from one or another. And we can see it clearly in the correlation matrix.
3) Pairplot of the features: we can see that our data is quite separable on most of the columns


Fig. 5. Pairplot of the features

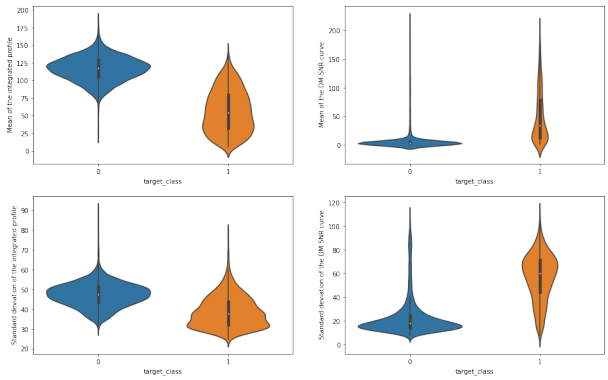


Fig. 6. Violin Plots for the standard deviation and the means

4) Violin plot of the features: We can see that our data has different kind of distributions which is helpful for training our models.

## IV. The Support Vector Machines Model

After preprocessing the data, we move to the heart of the problem: the prediction of whether an star (not in the training data-set) is a pulsar star. Here are the steps for the same:

- Appropriate Data type: Before applying a classification model to our data, we will analyze the data types for all variables, and change them, when necessary. Initially, all variables are of the continuous category. We can use the max-min scaler to normalize the ranges of different columns.
- Different Classification models and results: We have applied the various classification models we have learnt till now to our data set and we compare the results shown by the SVM with the other models. The results are shown in the confusion matrices for the different classification models, accompanied by a bar chart comparison of the models.

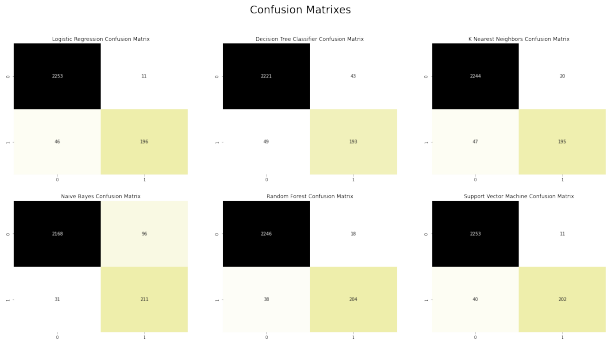


Fig. 7. Confusion matrices for the different classification models

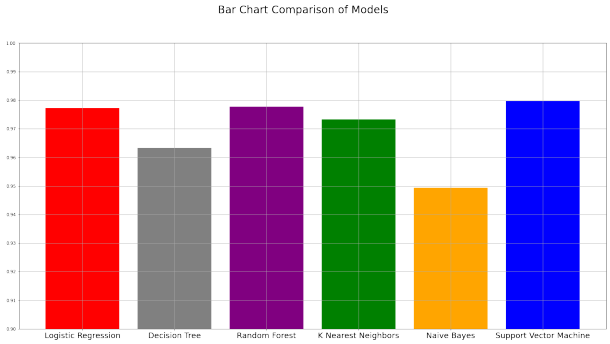


Fig. 8. Bar chart comparing the different classification models based on the model scores

If we compare total mistakes: RandomForests, SVM, KNN seem to be best for this dataset. As shown by the results, random forests performs the classification excellently and delivers great results. If we look at the bar chart and check the scores, LogisticRegression, RandomForest and SVM are better than the others.
SVMs are clearly the winners for the best classification model.

## V. Conclusion

The SVM model was succesfully demonstrated, both visually and numerically. While the accuracy of the model is good enough, we could perform Kernel tricks to further improve the SVM score.

## References

[1] "HTRU2 Dataset", archive.ics.uci.edu, 1990 [Online]. Available: **Link**.