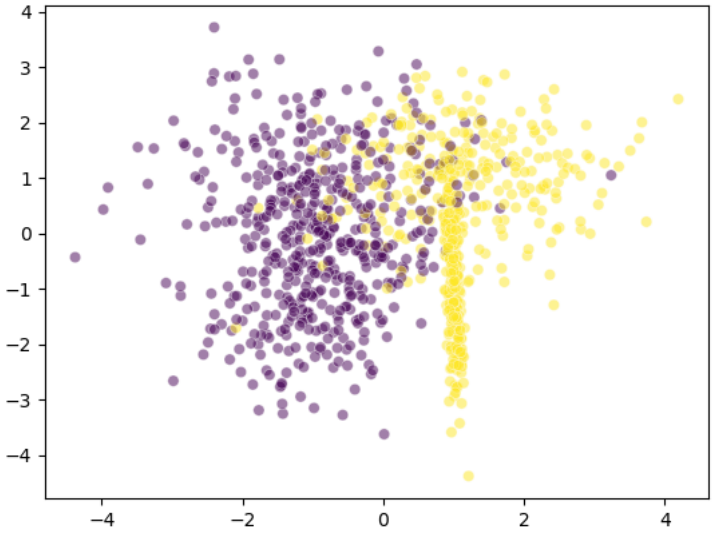
Adaboost

Saurabh Burewar (B18CSE050)

Dataset

A dataset is randomly generated with 1000 examples in a 2D space. Each example is assigned a value of 1 or -1. This dataset can be found in the file 'B18CSE050_Data.csv' and a plot of this data can be found in 'B18CSE050_Plots/samples_plot.png'. In this experiment, the following dataset is used -



To create a new dataset, just delete the 'B18CSE050_Data.csv' file and run the program again.

Adaboost

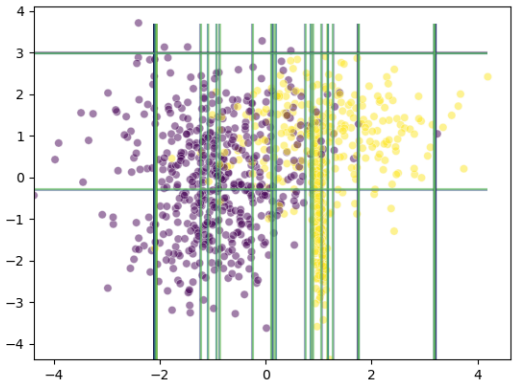
Adaboost is a boosting algorithm which uses decision stumps which is a Decision tree classifier. Adaboost is trained for n stages and in each stage data is classified using a stump and weights are updated. So, increasing the number of stages will increase the number of stumps used in boosting and thus we will get better results.

In this implementation, the *Adaboost* class consists of 3 main functions - fit(), predict() and plot().

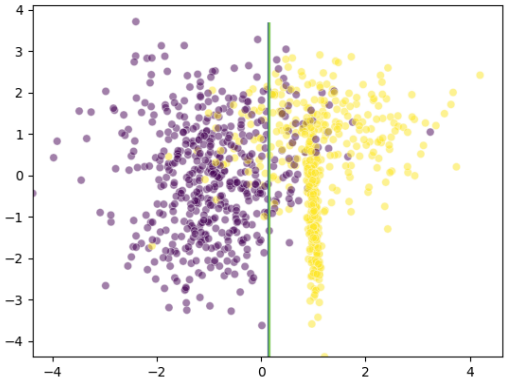
Experiments

Since we are saving our created dataset in 'B18CSE050_Data.csv', we use this data with different number of stages in Adaboost and look at the performance.

For 100 stages, plotting the decision boundary of each stump, we get the plot on the left below, and Adaboost gives the boundary as the figure on the right -

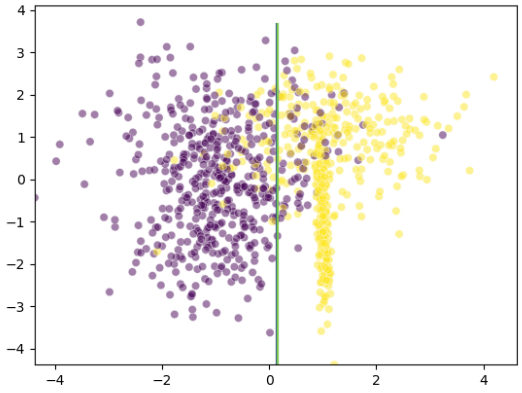


Stages=100 (Individual stumps boundary)

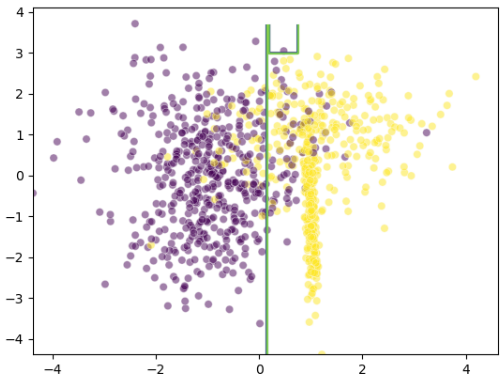


Stages=100 (Adaboost boundary)

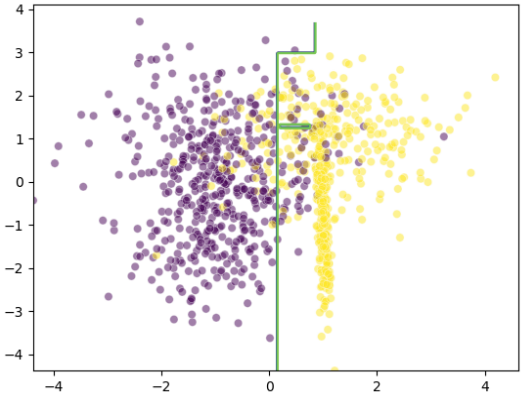
Increasing the number of stages gives us an increase in the performance of the classifier -



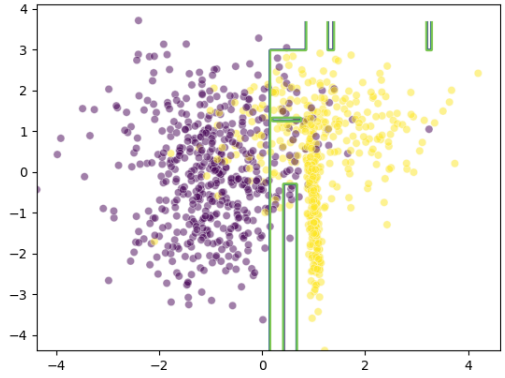
Stages=200



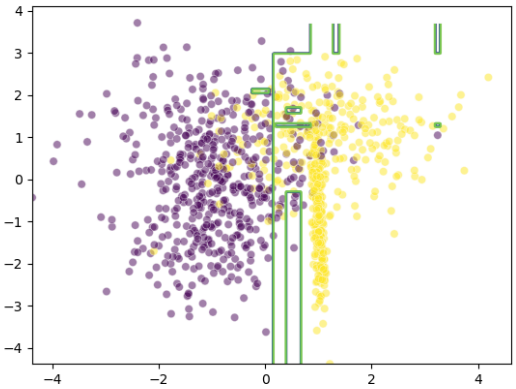
Stages=300



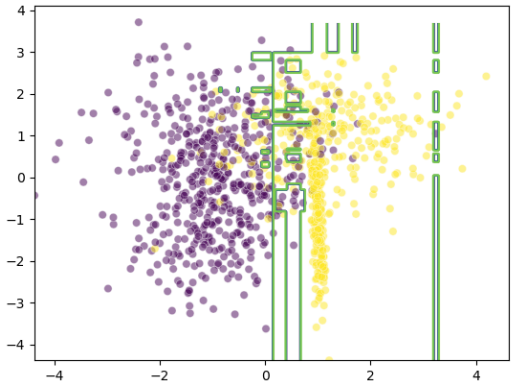
Stages=500



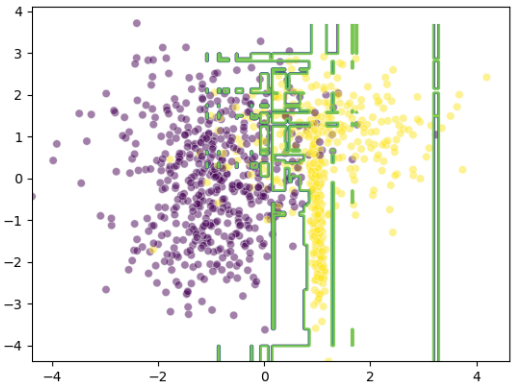
Stages=700



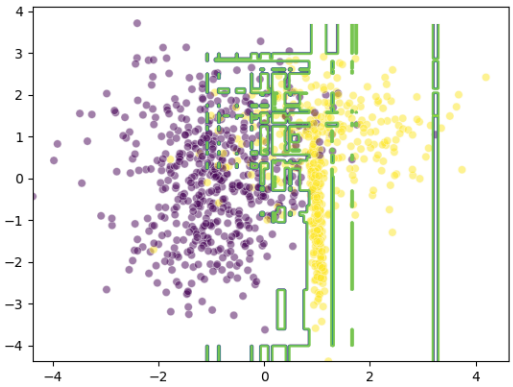
Stages=1000



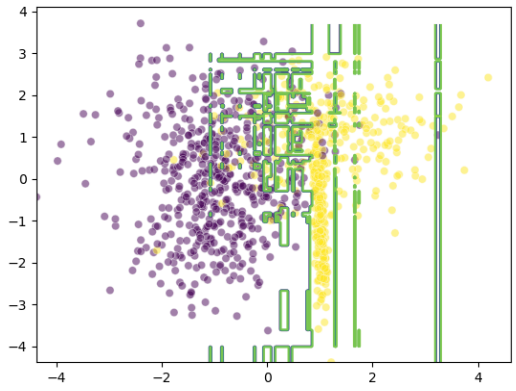
Stages=2000



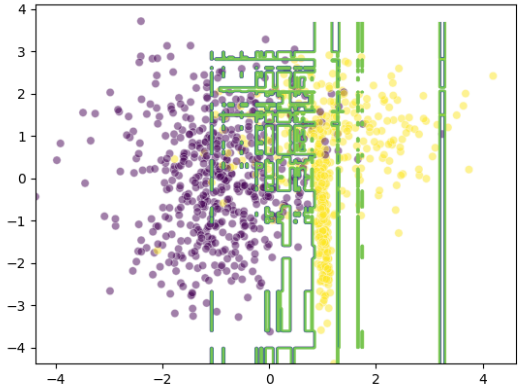
Stages=5000



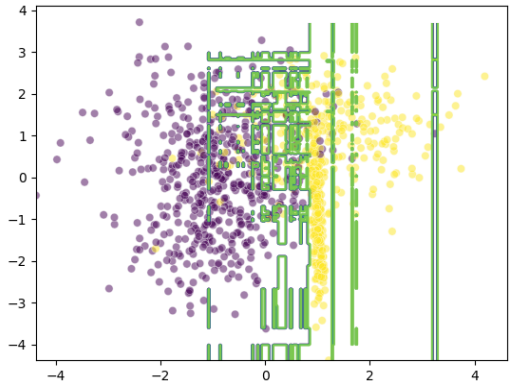
Stages=7000



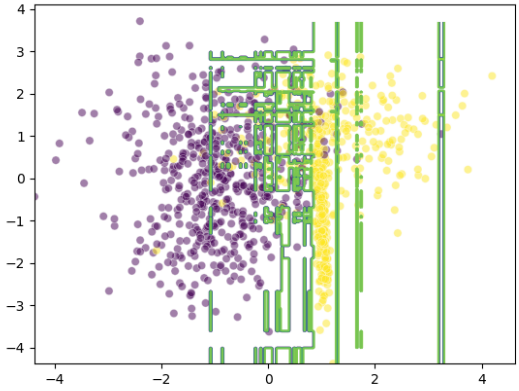
Stages=10000



Stages=15000



Stages=20000



Stages=22000

With every increase in the number of stages, the number of weak classifiers increases. With this, there is an increase in different decision boundaries made by those weak classifiers contributing to the boosting algorithm. Thus, we get a better classification as we increase the number of stages.

The following table shows the accuracy increase with increase in number of stages -

Stages	**Accuracy (in %)**
100	90.4
200	90.4
300	90.5

500	90.60000000000001
700	91.0
1000	91.2
2000	92.30000000000001
5000	94.19999999999999
7000	95.19999999999999
10000	96.1
15000	98.0
20000	99.5
22000	100.0

Increasing the number of weak classifiers can increase the performance of the boosting classifier, rather significantly in most cases. This makes boosting algorithms one of the best and most versatile classifiers.