# Predicting Song Likeability for a Spotify User

Bipin Dhoddamane Ravi
Deron Martin
Nimisha Gulati
Sarthak Jain
Siddhant Dushyant Purohit
Utkarsh Neema

UC RIVERSIDE

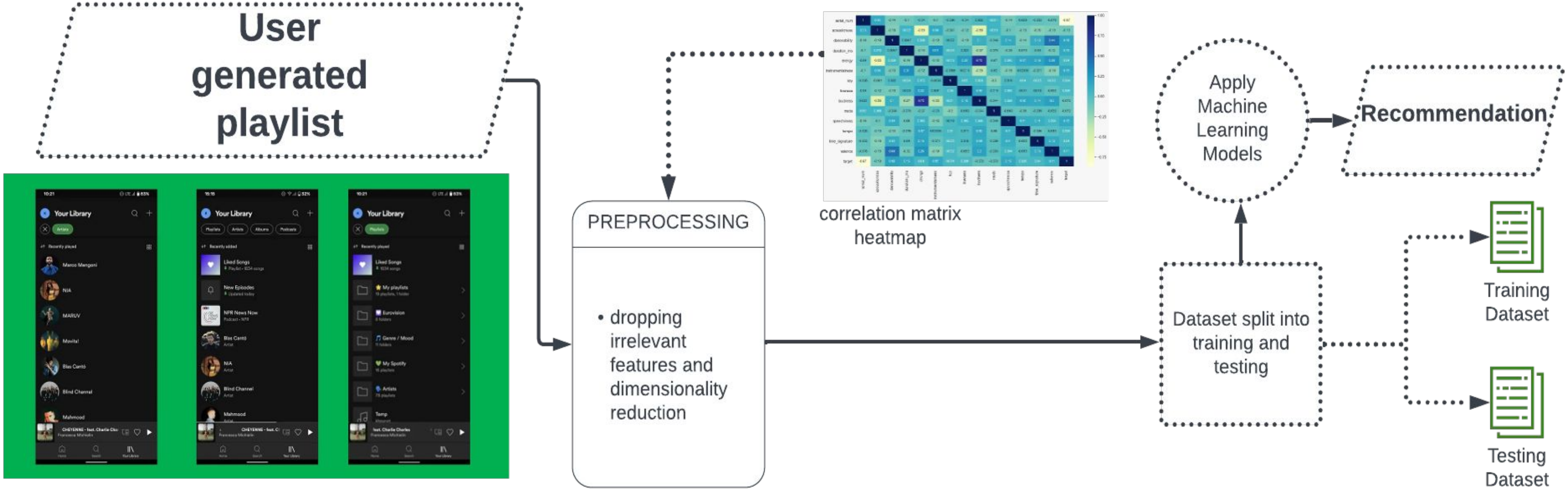
# Introduction

1

## Why spotify?

- Variations over 1300! music genres
- readily available large datasets
- real world problem

cite 2



User generated playlist

PREPROCESSING

- dropping irrelevant features and dimensionality reduction

correlation matrix heatmap

Apply Machine Learning Models

Recommendation

Dataset split into training and testing

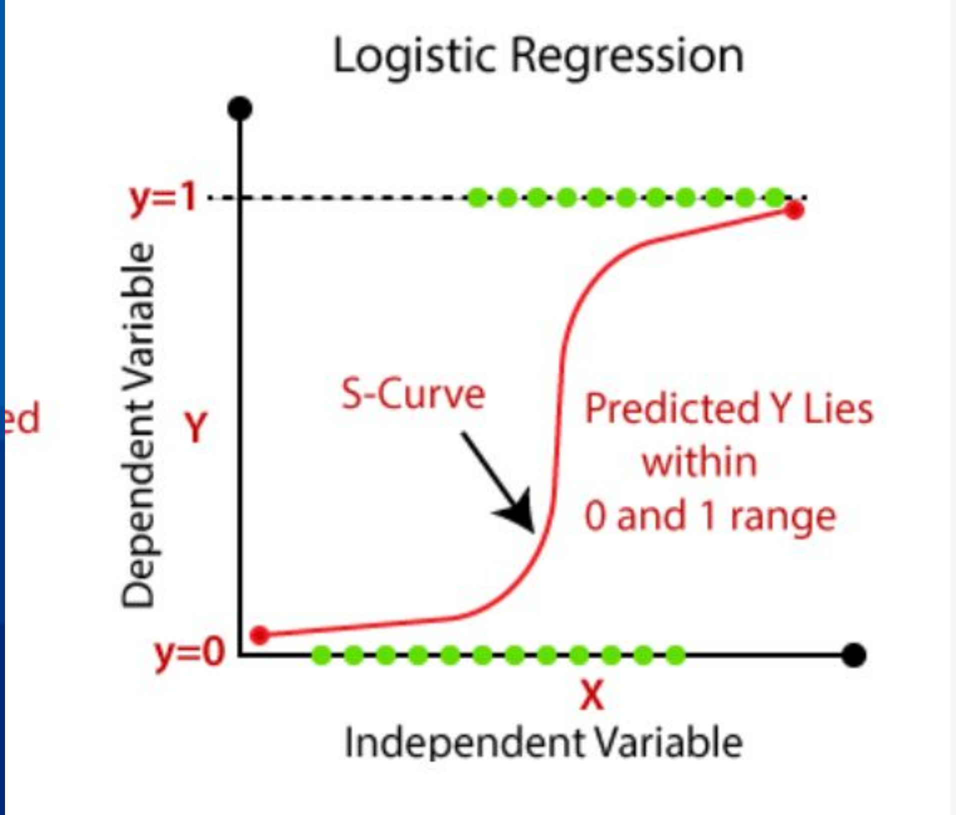
Training Dataset

Testing Dataset

# Logistic Regression

By Siddhant Dushyant Purohit

# Logistic Regression

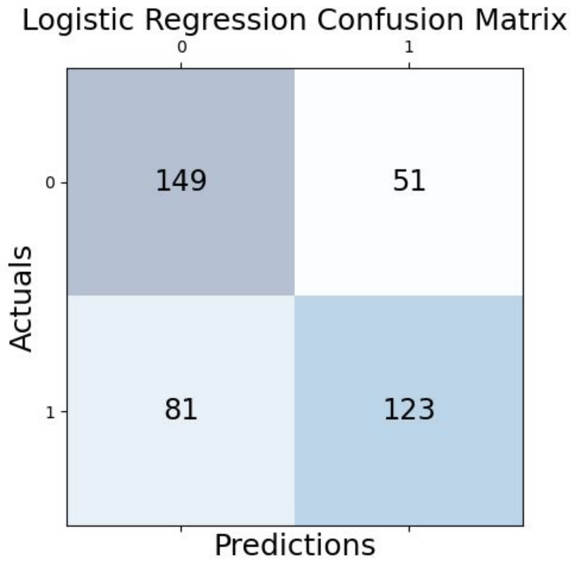


- Supervised learning algorithm used for linear classification.
- Ideal for categorical binary classification
- Weights features to obtain probabilistic outcomes.

# Logistic Regression

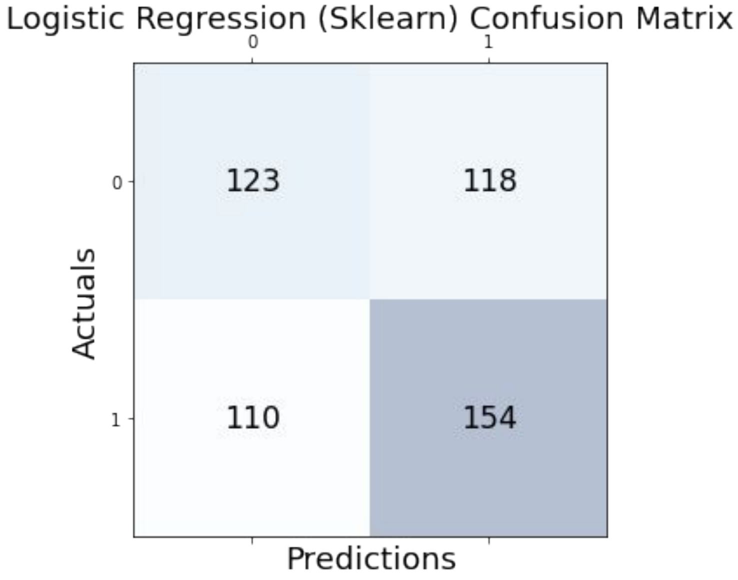
**Results:**

- Accuracy with sklearn: 0.54
- Accuracy with our implementation(standard weight initialization): 0.52
- Accuracy with our implementation(custom weight initialization): 0.67



Logistic Regression Confusion Matrix

| Actuals \ Predictions | 0 | 1 |
|---|---|---|
| 0 | 149 | 51 |
| 1 | 81 | 123 |



Logistic Regression (Sklearn) Confusion Matrix

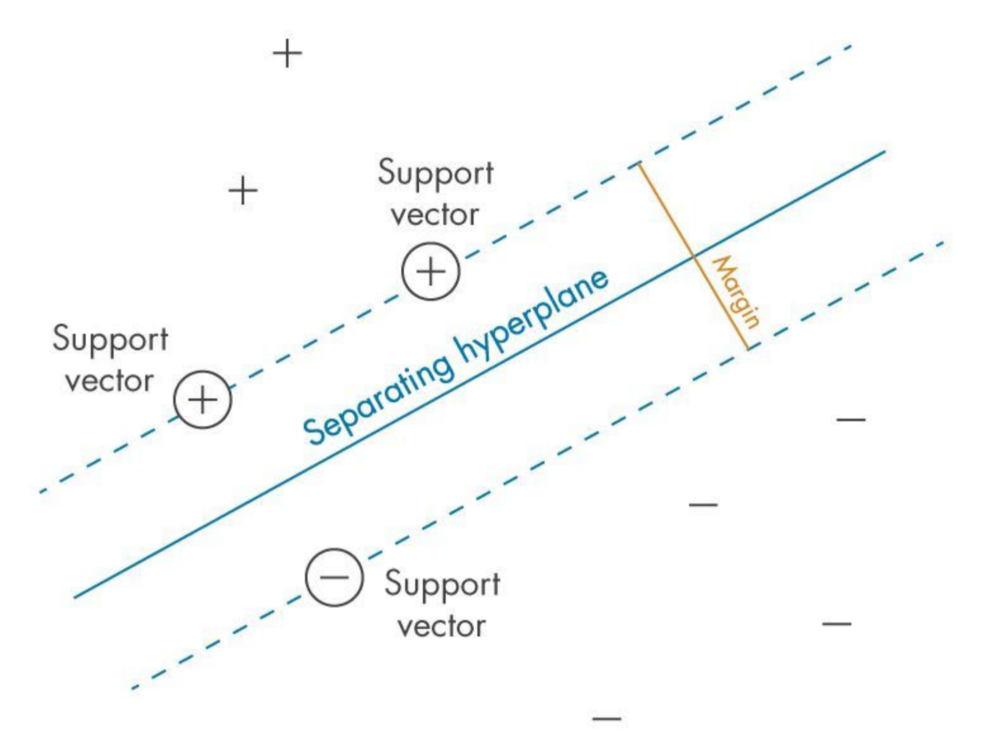
| Actuals \ Predictions | 0 | 1 |
|---|---|---|
| 0 | 123 | 118 |
| 1 | 110 | 154 |

# Support Vector Machines

By Deron Martin

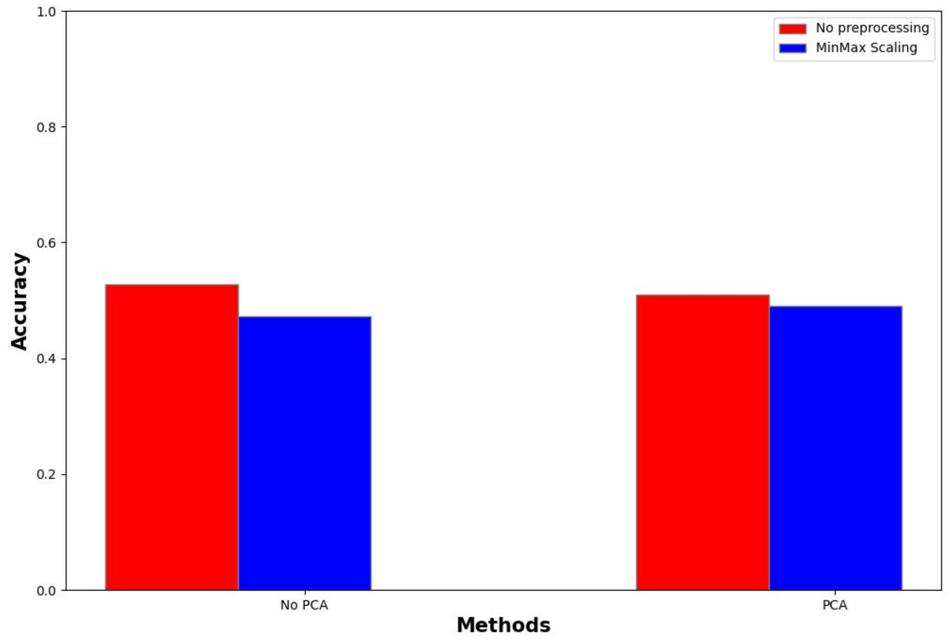
UC RIVERSIDE

# Support Vector Machines



- Used to create decision boundaries for linearly separable data
- Supervised learning method which creates decision boundaries in high dimension space
- Once trained, only requires the support vectors
- Can be used for non-linearly separable data using kernel tricks

# Results

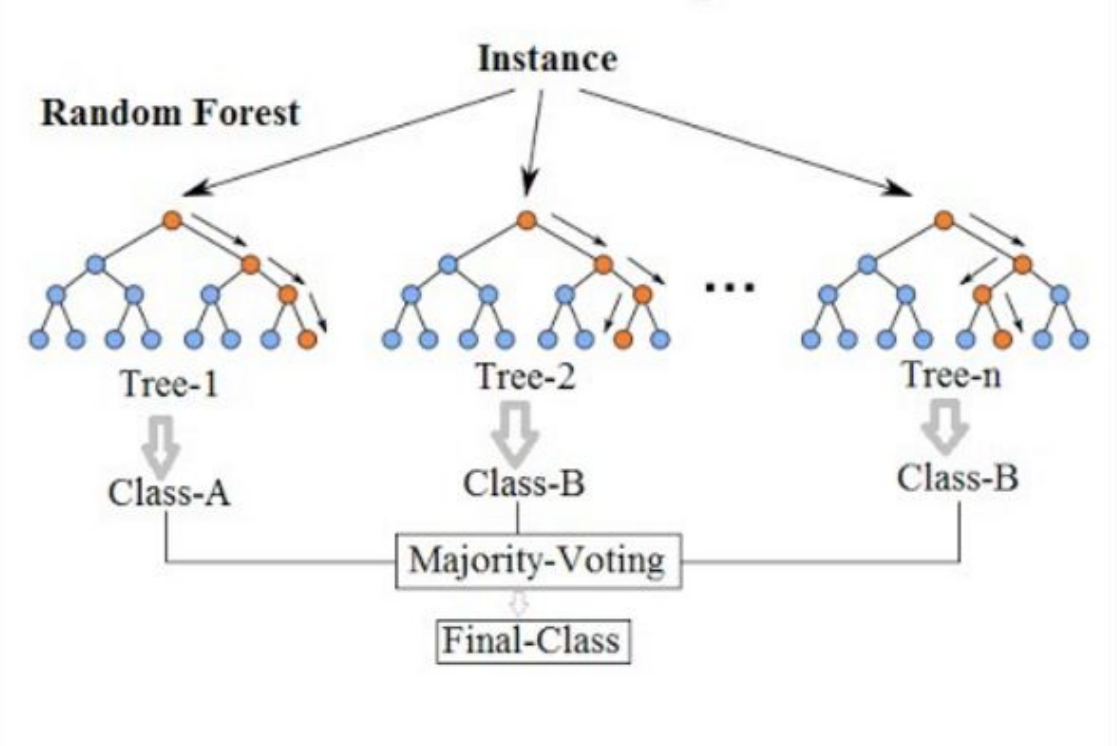


- Accuracy: 0.5272
- Only slightly above randomly guessing
- Like and disliked songs are not separable in linear space
- Comparable to Scikit-learn implementation which has an accuracy of 0.5594

# Random Forest

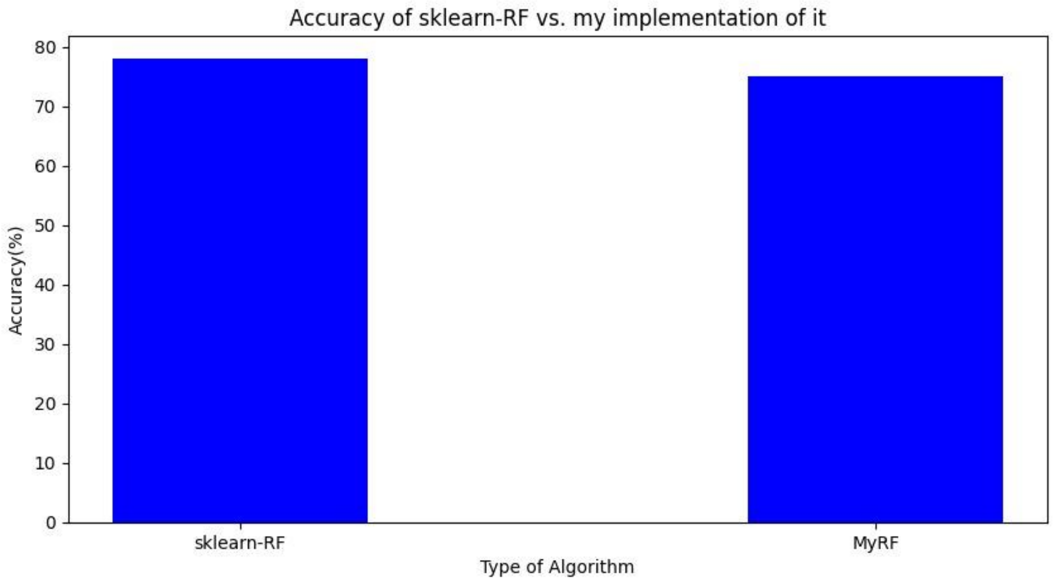
By Nimisha Gulati

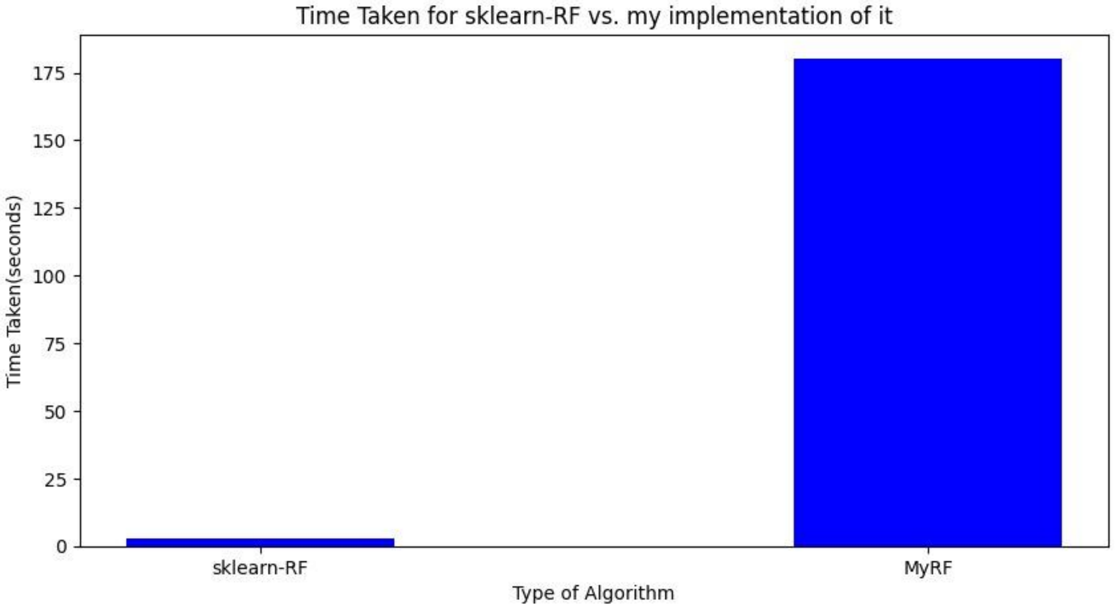
# Random Forest



- Random Forest algorithm utilizes the power of decision trees to make classifications
- It uses random subspacing and bootstrapping while making decision trees
- It then considers the prediction made by majority of the decision trees as final output
- Each decision tree created is independent of the other
- Splits in decision tree are created based on ‘information gain’

# Results





- Accuracy of my implementation : 73%
- Accuracy of scikit-learn : 75%

- Time taken by my implementation : 3 mins
- Time taken by scikit-learn : 3 seconds

# K-Means Clustering

By Bipin Dhoddmane Ravi

UC RIVERSIDE

# K-Means Clustering

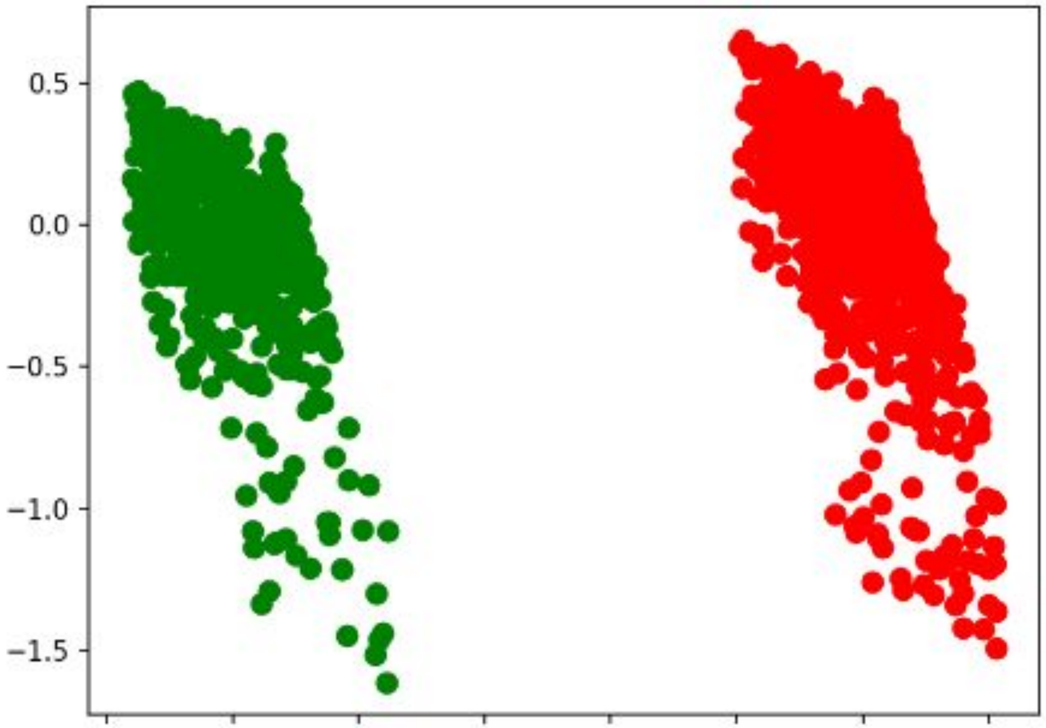


*Figure 12: After clustering, with 2 clusters.*

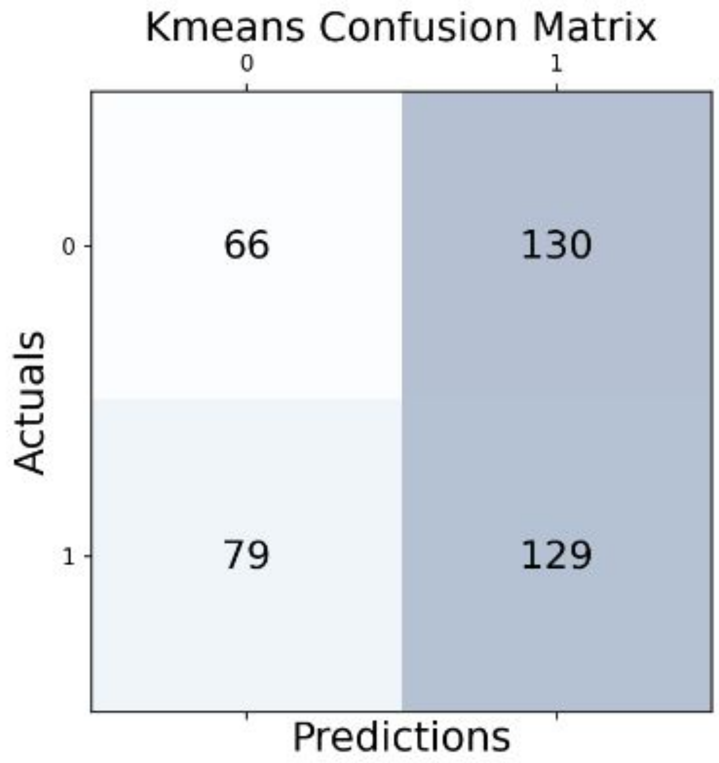
Kmeans is an unsupervised clustering algorithm.

It is used to find structure in data when we have no labels.

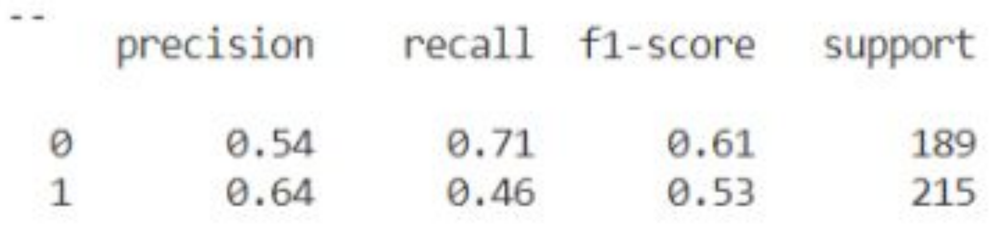
We use Kmeans with k=2 for creating 2 cluster used for binary classification of liked and disliked songs.

# K-Means Clustering

## Results



| | 0 | 1 |
|---|---|---|
| 0 | 66 | 130 |
| 1 | 79 | 129 |



```
--
              precision    recall  f1-score   support

           0       0.54      0.71      0.61       189
           1       0.64      0.46      0.53       215
```

*Figure 10: Evaluation metrics for KMeans*

Kmeans is not an ideal solution to binary classification problems

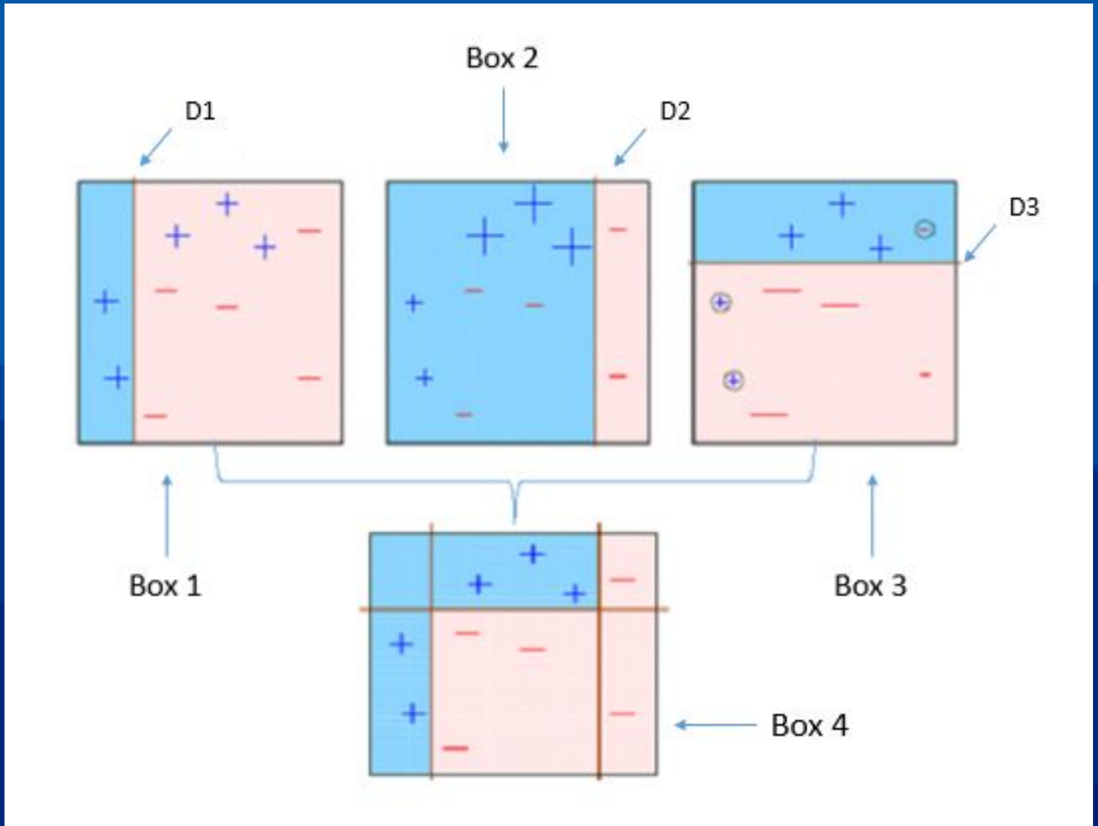
The clusters formed do not have labels and can be any 2 clusters.

# AdaBoost

By Utkarsh Neema

UC RIVERSIDE

# AdaBoost



Ensemble Learning

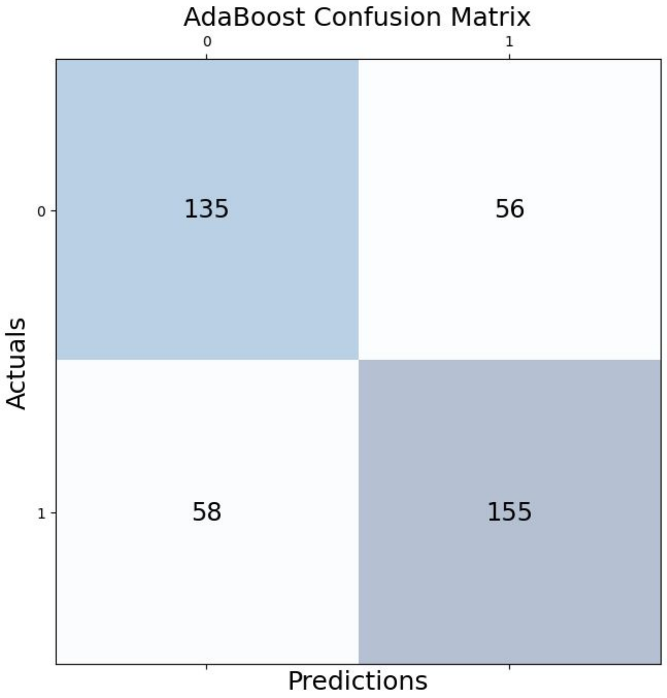
- Weak Classifier
- Misclassified samples get more weight
- Weighted Average for result

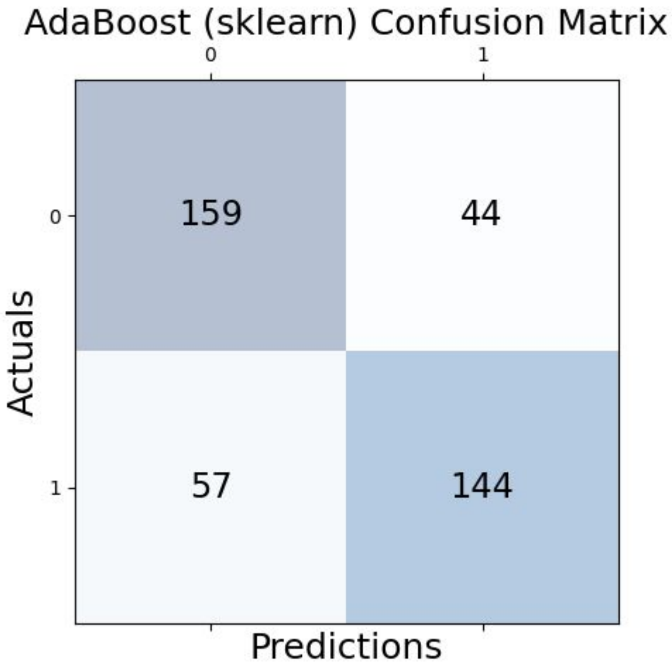
https://towardsdatascience.com/understanding-adaboost-2f94f22d5bfe

# AdaBoost

**Results:**

- Accuracy with sklearn: 0.75
- Accuracy with our implementation: 0.72

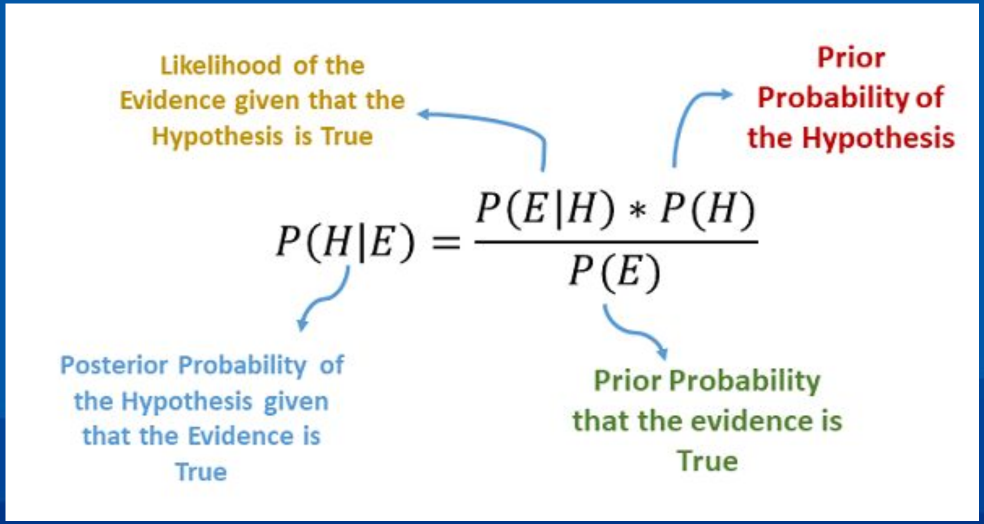




# Naive Bayes Classifier

By Sarthak Jain

# Naive Bayes Classifier



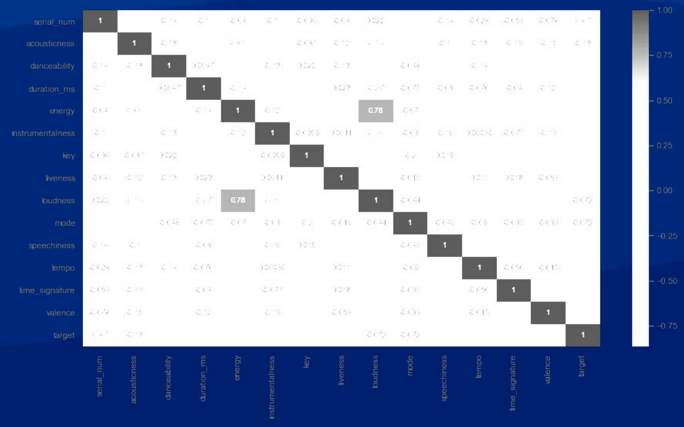
Likelihood of the Evidence given that the Hypothesis is True

Prior Probability of the Hypothesis

$$P(H|E) = \frac{P(E|H) * P(H)}{P(E)}$$

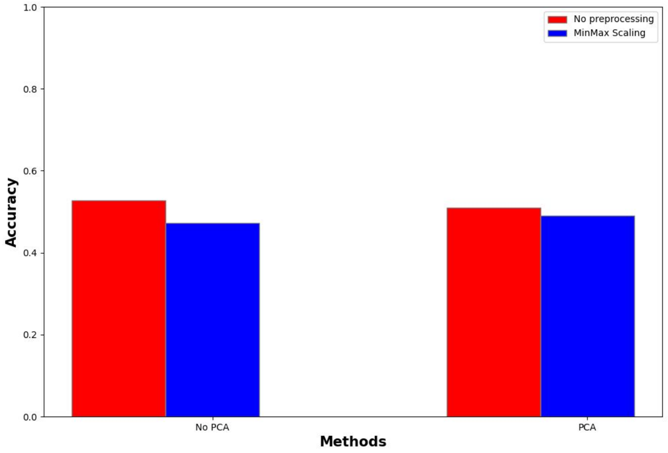
Posterior Probability of the Hypothesis given that the Evidence is True

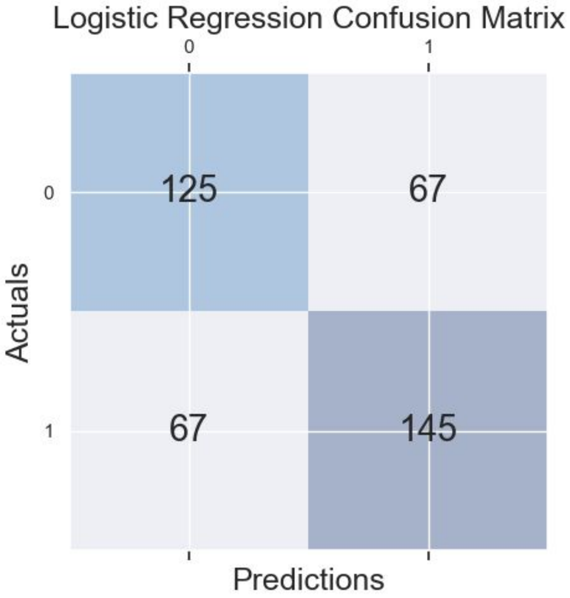
Prior Probability that the evidence is True



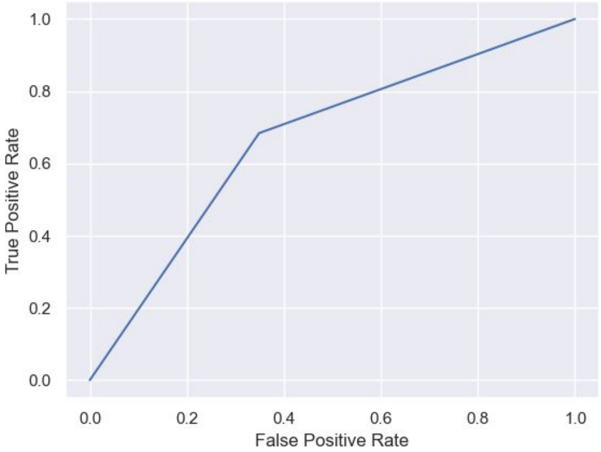
- Naive Bayes classifier is a supervised machine learning algorithm that is used to predict/classify unseen data based on prior probabilities.
- We have used min-max scaling and gaussian distribution functions to normalize our data
- Naive Bayes assumes independence of features which helps our case

# Results



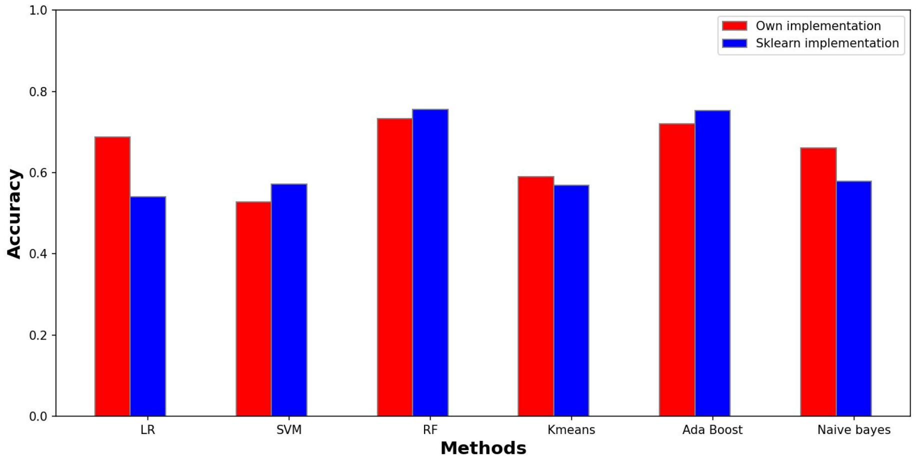

Logistic Regression Confusion Matrix

| Actuals \ Predictions | 0 | 1 |
|---|---|---|
| 0 | 125 | 67 |
| 1 | 67 | 145 |



- Preprocessing improves our accuracy
- accuracy is 66%
- compared to other algorithms, we get a better AUC

# Conclusion



- supervised learning algorithms like logistic regression seen above is comparable but not the most efficient modeling technique when it comes to binary categorical classifications.
- Ensemble learning algorithms like random forest and ADA boost provide higher accuracy metrics close to 75% accuracy while in contrast the supervised method of Logistic regression only reaches upto 68% accuracy with modifications.
- Other clustering algorithms which are also unsupervised in nature provide slightly lower accuracy scores due to inability to distinguish between weighted features.