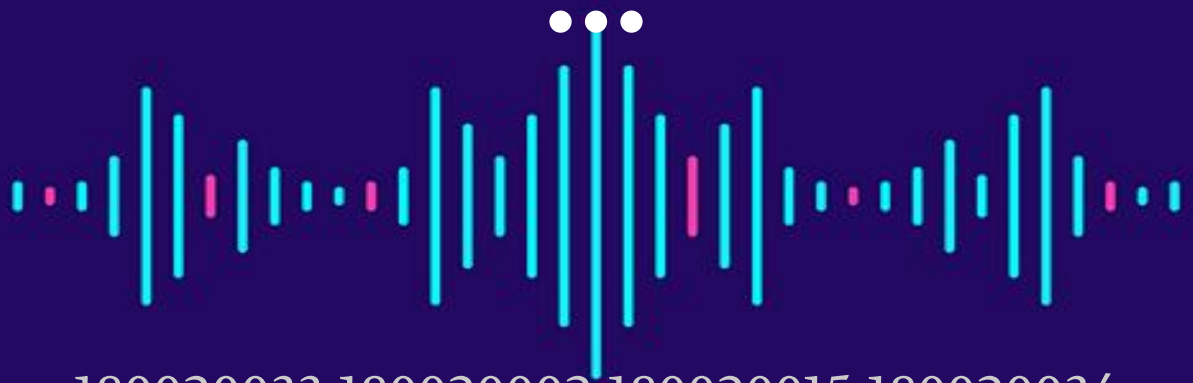
SPOOF DETECTION



180020033 180020002 180020015 180020034

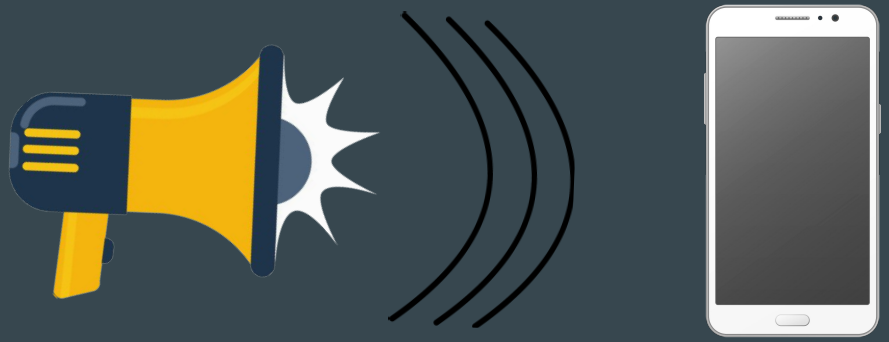
MOTIVATION

Nowadays, there is a wide focus on the research of spoofing and anti-spoofing for Automatic Speaker Verification (ASV) system. The recent technological advancement in the ASV system leads to an increased interest to secure these voice biometric systems for real world applications. The ASV systems are vulnerable to various kinds of spoofing attacks, namely, speech synthesis (SS), voice conversion (VC), replay, twins, and impersonation.

Here, we attempt to build a robust system which should be able to improve upon the working of present ASV systems.

ABSTRACT

Spoofing detection for automatic speaker verification (ASV) aims to discriminate between genuine and spoofed speech. This topic has received increased attention recently due to safety concerns with deploying an ASV system. We aim to develop a system capable of accurately distinguishing between human and synthesized speech in order to thwart spoofing attacks on ASV systems wherein the attacker tries to use artificial speech synthesizers to try and fool the system.



REVIEW OF LITERATURE

1. Spoof Detection Using Source, Instantaneous Frequency and Cepstral Features, Sarfaraz Jelil, Rohan Kumar Das, S. R. M. Prasanna and Rohit Sinha, Department of Electronics and Electrical Engineering, Indian Institute of Technology Guwahati, Guwahati-781039, India{sarfaraz, rohankd, prasanna, rsinha}@iitg.ernet.in:

   This work describes the methods used for detection of spoofed speech signals for the ASVspoof2017 challenge. The features extracted from the speech signals are: Epoch and Epoch Strength, Peak to Side Lobe Ratio(PSR) of the Hilbert Transform(HT) of Linear Prediction(LP) Residual, Instantaneous Frequency Cosine Coefficient and CQCC. The Experimental setup consists of 5 systems which use the following features namely, EF(S1), PSRMS(S2), IFCC(S3), CQCC(S4) and MFCC(S5). Individually, these systems give an equal error rate(EER) of 36.29%, 31.60%, 24.81%, 9.79 % and 18.90 % respectively on the development set. When we combine all of them we get an EER of 5.31 %.

REVIEW OF LITERATURE (continued)

2. Spoofing Speech Detection Using Modified Relative Phase Information Longbiao Wang, Member, IEEE, Seiichi Nakagawa, Zhaofeng Zhang, Yohei Yoshida, and Yuta Kawakami:

In this paper, modified relative phase (MRP) information extracted from a Fourier spectrum is proposed for spoofing speech detection. The features extracted here were Modified group delay cepstral coefficient(MGDCC), Cosine Phase, Relative Phase Shift(RPS), Relative Phase(RP), Pseudo pitch synchronized relative phase(PPSRP) and Proposed Modified Relative Phase(MRP). Here they have compared the Average EERs of each feature with MFCC. Average EER for MFCC alone is 2.436 %. The EERs of other features are as follows: MGDCC: 2.048 %, RPS: 4.473 %, Cosine Phase: 4.487 %, RP: 4.239 %, PPSRP: 3.925 % and MRP: 2.987 %. Various Combinations of these features were tried and the combination that gave the least EER was MFCC + MGDCC + MRP with least EER equal to 0.764 %. The experimental setup was made up of 10 systems.

PROPOSED METHODOLOGY

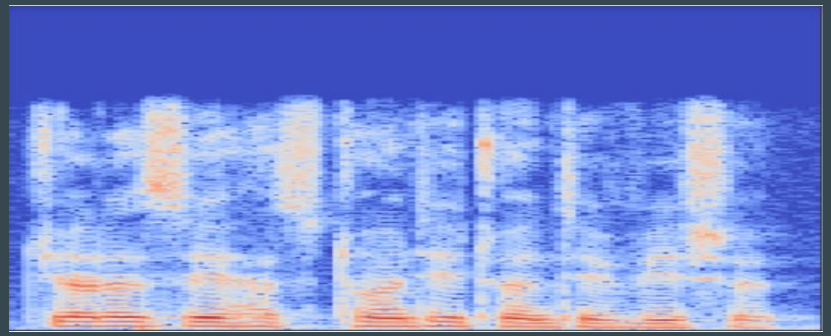
- We intend to make use of the dataset available at The 2nd Automatic Speaker Verification Spoofing and Countermeasures Challenge (ASVspoof 2017) Database, Version 2.
- For the task of feature extraction, we plan to use the Librosa library in python, which is a popular library for audio feature extraction.
- We intend to try out a variety of models starting from SVMs to Gaussian Mixture models and evaluate the accuracy for each type of model used.
- To train our models, we will use the sklearn library, which is a very user-friendly library for machine learning applications.
- We first intend to duplicate the results arrived at in the listed papers for review, then we will try to improve on the techniques mentioned therein if time permits.

PRELIMINARY DATA VISUALIZATION

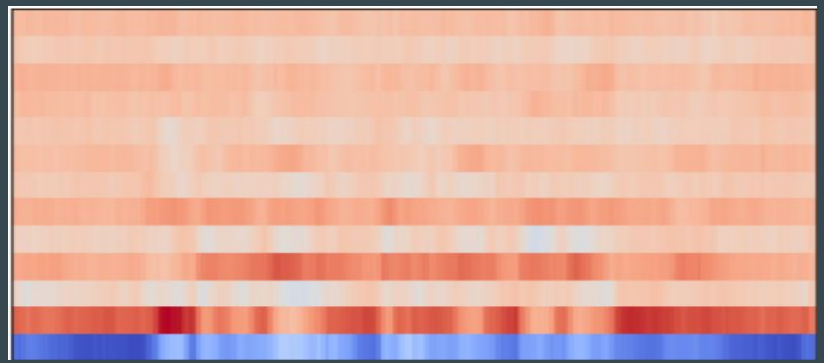
Time domain waveform for a non spoofed speech signal

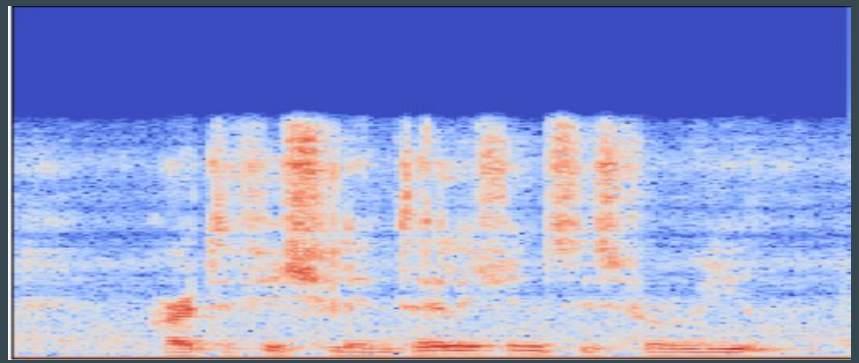
Time domain waveform for a spoofed speech signal

PRELIMINARY DATA VISUALIZATION (continued)



MFCC features and spectrogram for a non spoofed speech signal





MFCC features and spectrogram for a spoofed speech signal

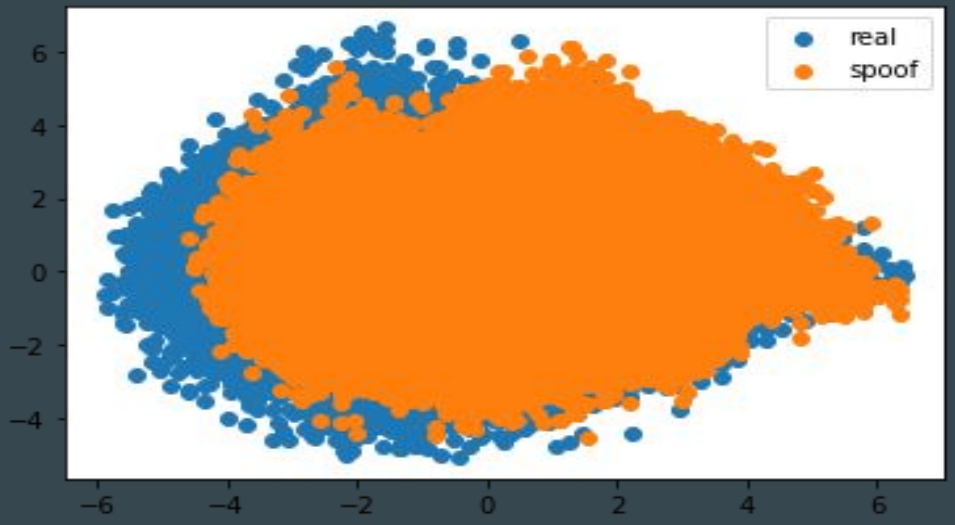
DATA CLEANING

- The dataset contained repeated file names, which were filtered out using string matching
- There were 3 corrupted audio files in the data which the librosa library failed to read. This was handled by using try-except blocks.
- Loading the data using librosa was very time consuming, so the data was loaded just once, and then stored in array format into a binary file
- Due to computational constraints, we had to undersample the dataset for training. Only 40% of the data was used to train our models..
- The model was tested on 200 spoof and 200 non spoof examples respectively.

FEATURE EXTRACTION

- The librosa library was used to extract MFCC features. 20 such features (representing the first 20 terms of the DCT) were considered.
- Each training example was windowed into 20ms intervals and the MFCCs of each window were computed. Each window was considered to be one training example
- After this around 100,000 training examples were obtained.
- During testing phase, each training example was broken down into windows and each window classified as spoof or non spoof.
- A majority voting based system was used to determine if a particular testing example was spoof or not.

PCA PLOT



The above plot shows that there is significant overlap between spoof and non-spoof examples and thus non linear classifiers should work well on the given data. We can also see that the data can be well approximated by a gaussian random variable, so a GMM might work well on this data.

SVM (linear kernel)

Accuracy: 95.91 %

Execution time: 9.475 seconds

true negatives: 15528

false positives: 727

false negatives: 653

true positives: 16874

Precision:0.959

Recall: 0.963

SVM (rbf kernel)

Accuracy: 94.36%

Execution time: 62.824 seconds

true negatives: 15182

false positives: 846

false negatives: 1059

true positives: 16705

Precision:0.952

Recall: 0.940

SVM (polynomial kernel)

Accuracy: 93.79%

Execution time: 43.799 seconds

true negatives: 15354

false positives: 674

false negatives: 1425

true positives: 16339

Precision:0.960

Recall: 0.920

KNN

Accuracy : 96.67 %

Execution time: 20.123 seconds

true negatives: 15471

false positives: 557

false negatives: 567

true positives: 17197

Precision: 0.969

Recall: 0.968

GMM

Accuracy : 96.36 %

Execution time: 1.163 seconds

true negatives: 15222

false positives: 473

false negatives: 732

true positives: 16706

Precision: 0.972

Recall: 0.958

RANDOM FOREST

Accuracy : 88.73%

Execution time: 11.275 seconds

true negatives: 13613

false positives: 2415

false negatives: 1392

true positives: 16372

Precision: 0.871

Recall: 0.922

LOGISTIC REGRESSION

Accuracy: 54.62 %

Execution time: 1.65 seconds

true negatives: 15986

false positives: 42

false negatives: 15292

true positives: 2472

Precision:0.983

Recall: 0.139

All Models(Majority Voting)

Accuracy: 96.18 %

Execution time: 1.607 seconds

true negatives: 15381

false positives: 647

false negatives: 642

true positives: 17122

Precision:0.964

Recall: 0.964

Gaussian Naive Bayes

Accuracy : 96.36 %

Execution time: 1.292 seconds

true negatives: 15222

false positives: 473

false negatives: 732

true positives: 16706

Precision: 0.972

Recall: 0.958

Stacking(Gaussian NB with KNN)

Accuracy: 96.94 %

Execution time: 1.096 seconds

true negatives: 15303

false positives: 392

false negatives: 622

true positives: 16816

Precision: 0.977

Recall: 0.964

Stacking(KNN with KNN) logistic regression (endclassifier)

Accuracy: 96.90 %

Execution time: 1.306 seconds

true negatives: 15220

false positives: 475

false negatives: 552

true positives: 16886

Precision:0.973

Recall: 0.968

Stacking Logistic Regression and KNN with Gaussian Naive Bayes (end class)

Accuracy: 68.06 %

Execution time: 1.048 seconds

true negatives: 5731

false positives: 9964

false negatives: 619

true positives: 16819

Precision:0.627

Recall: 0.964

Boosted Logistic Regression

Accuracy: 85.30 %

Execution time: 1.107 seconds

true negatives: 14725

false positives: 970

false negatives: 3899

true positives: 13539

Precision:0.933

Recall: 0.776

Boosted Decision Trees

Accuracy: 80.34 %

Execution time: 1.118 seconds

true negatives: 11874

false positives: 3821

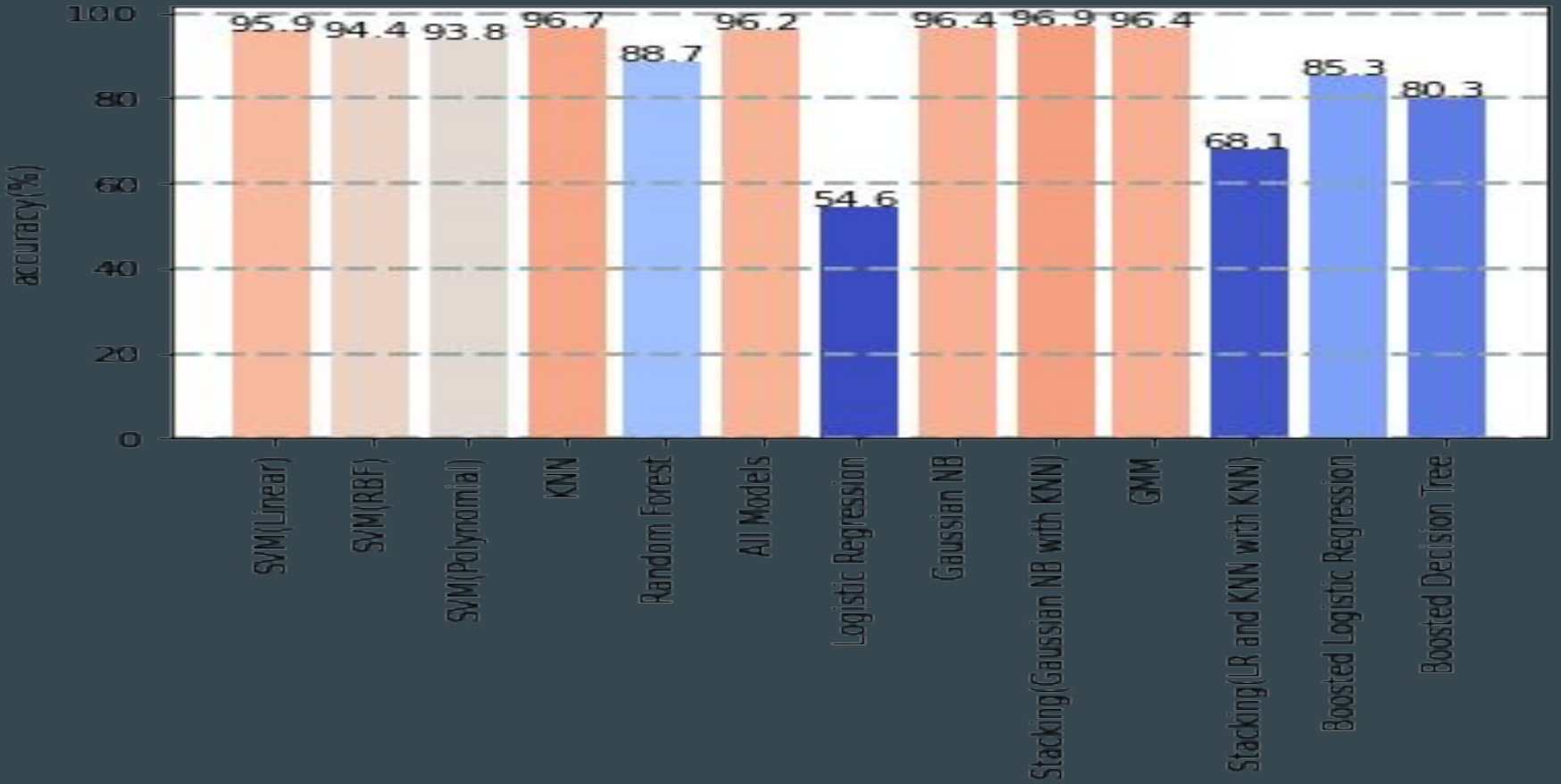
false negatives: 2692

true positives: 14746

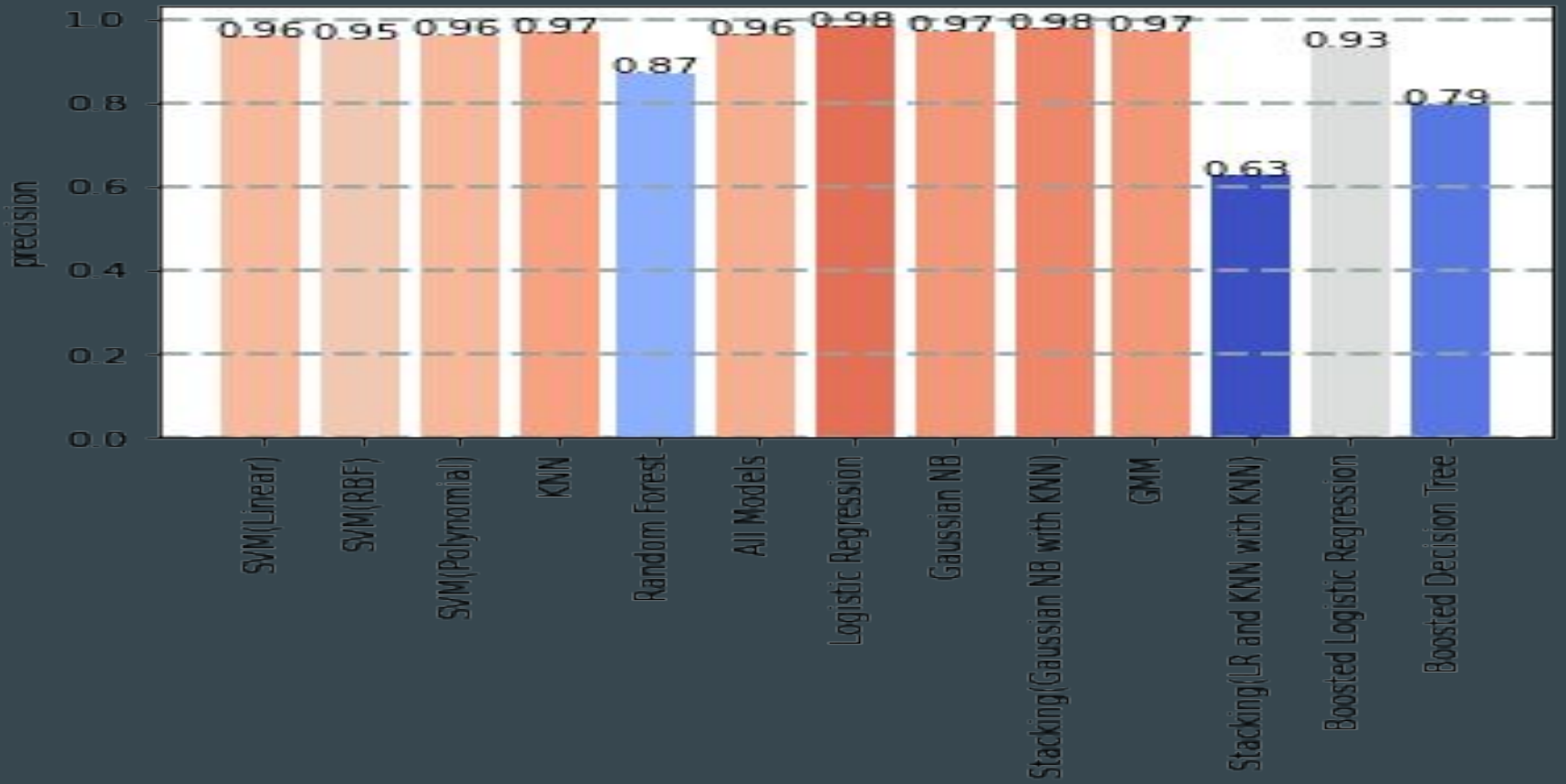
Precision: 0.794

Recall: 0.845

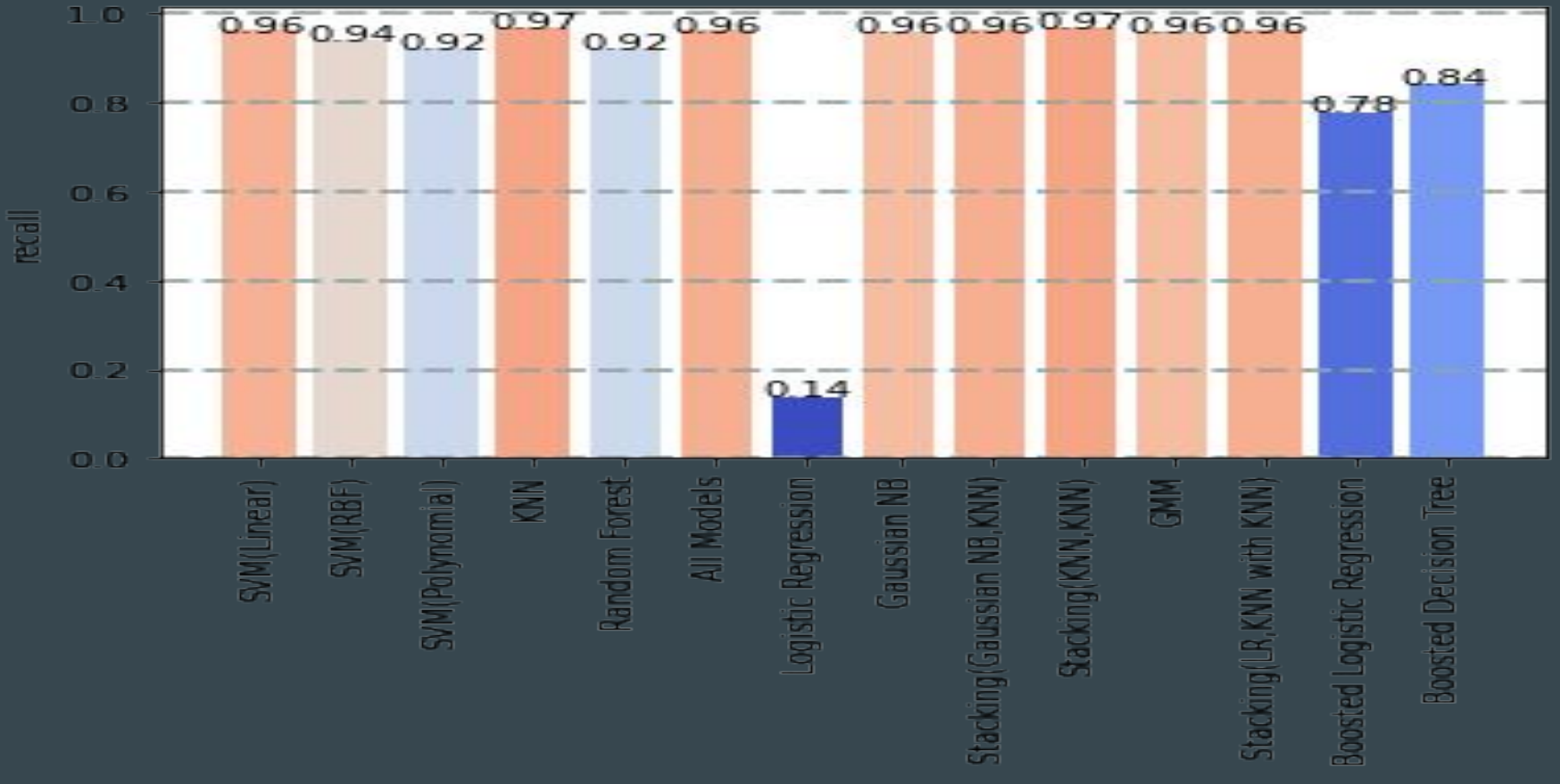
Accuracies


accuracy(%)
0
20
40
60
80
100
95.9
94.4
93.8
96.7
88.7
96.2
54.6
96.4
96.9
96.4
68.1
85.3
80.3
SVM(Linear)
SVM(RBF)
SVM(Polynomial)
KNN
Random Forest
All Models
Logistic Regression
Gaussian NB
Stacking(Gaussian NB with KNN)
GMM
Stacking(LR and KNN with KNN)
Boosted Logistic Regression
Boosted Decision Tree

Precisions


precision
1.0
0.8
0.6
0.4
0.2
0.0
0.96
0.95
0.96
0.97
0.87
0.96
0.98
0.97
0.98
0.97
0.63
0.93
0.79
SVM(Linear)
SVM(RBF)
SVM(Polynomial)
KNN
Random Forest
All Models
Logistic Regression
Gaussian NB
Stacking(Gaussian NB with KNN)
GMM
Stacking(LR and KNN with KNN)
Boosted Logistic Regression
Boosted Decision Tree

Recalls


recall
1.0
0.8
0.6
0.4
0.2
0.0
0.96
0.94
0.92
0.97
0.92
0.96
0.14
0.96
0.96
0.97
0.96
0.96
0.78
0.84
SVM(Linear)
SVM(RBF)
SVM(Polynomial)
KNN
Random Forest
All Models
Logistic Regression
Gaussian NB
Stacking(Gaussian NB,KNN)
Stacking(KNN,KNN)
GMM
Stacking(LR,KNN with KNN)
Boosted Logistic Regression
Boosted Decision Tree