Spotify Billboard Classifier
Course Project - DATA1030

SAYAN SAMANTA

School of Engineering | Brown University

Instructor: Dr. Andras Zsom | TA: Natalie Delworth

December 2, 2019

Contents

Abstract

In this report, an attempt has been made to predict whether a particular song will appear in the US top 200 billboard based on the songs given it's acoustic features. Song data that appeared on the top 200 billboard has been received from Kaggle while it was appended with songs that didn't appear on the billboard using the Spotify Python API. The data obtained was highly imbalanced, with the acoustic features higly overlapping. The baseline accuracy is at 88%. A no. of classification algorithms were performed with stratified k-fold cross validation. Among all the models analysed, only the ones with accuracy above baseline are discussed here. These include Random Forest, XGBoost and Adaboost. The entire machine learning pipeline including data description, exploratory data analysis, cross-validation, model inspection and feature importance have been performed and the results have been discussed with measures of improvement for future studies.

Keywords: Machine Learning, Classification, Random Forest, XGBoost, Adaboost, Spotify, Music, Acoustic Features.

1 Introduction

The dataset under consideration is the **The Billboard 200 acoustic data** which encompasses the entire chart from 1963-2019, along with the EchoNest acoustic features of as many songs as available. All the features of the songs were obtained using Spotify's python API.

The dataset mentioned above was curated for a piece on the data science analytics website Components. However that piece dealt with the album length alone, there could be plethora of other questions and analysis that can be answered/performed with this data.

Since this dataset contain only songs that appeared on the billboard, further data (songs not on the billboard) was obtained using the Python Spotify API (aptly called Spotipy).

1.1 Dataset Description:

The dataset has the following features:

Features	**Description**
duration_ms	The duration of the track in milliseconds
key	The estimated overall key of the track. Integers map to pitches using standard Pitch Class notation . E.g. 0 = C, 1 = CâŹŕ/DâŹ■, 2 = D, and so on. If no key was detected, the value is -1.
mode	Mode indicates the modality (major or minor) of a track, the type of scale from which its melodic content is derived. Major is represented by 1 and minor is 0.
time_signature	An estimated overall time signature of a track. The time signature (meter) is a notational convention to specify how many beats are in each bar (or measure).
acousticness	A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic
danceability	describes how suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity. A value of 0.0 is least danceable and 1.0 is most danceable.
energy	is a measure from 0.0 to 1.0 and represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. For example, death metal has high energy, while a Bach prelude scores low on the scale.
instrumentalness	Predicts whether a track contains no vocals. "Ooh" and "aah" sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly "vocal". The closer the instrumentalness value is to 1.0, the greater likelihood the track contains no vocal content. Values above 0.5 are intended to represent instrumental tracks, but confidence is higher as the value approaches 1.0
liveness	Detects the presence of an audience in the recording. Higher liveness values represent an increased probability that the track was performed live. A value above 0.8 provides strong likelihood that the track is live.
loudness	The overall loudness of a track in decibels (dB). Loudness values are averaged across the entire track and are useful for comparing relative loudness of tracks.

speechiness	Speechiness detects the presence of spoken words in a track. The more exclusively speech-like the recording (e.g. talk show, audio book, poetry), the closer to 1.0 the attribute value. Values above 0.66 describe tracks that are probably made entirely of spoken words. Values between 0.33 and 0.66 describe tracks that may contain both music and speech, either in sections or layered, including such cases as rap music. Values below 0.33 most likely represent music and other non-speech-like tracks.
valence	A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track. Tracks with high valence sound more positive (e.g. happy, cheerful, euphoric), while tracks with low valence sound more negative (e.g. sad, depressed, angry).
tempo	The overall estimated tempo of a track in beats per minute (BPM). In musical terminology, tempo is the speed or pace of a given piece and derives directly from the average beat duration

Table 1: Acoustic Features Description

In addition, we have the name of the song, name of the artist, the album it appeared on, the date of release and the no. of tracks on the album.

1.2 Objectives

1.2.1 Target Variable

In this project, we would attempt to predict if a song would appear in the billboard or would it not. Hence it is a **classification** problem.

1.2.2 Motivation behind the problem

This classifier would help a no. of group both in the music industry or otherwise.

1. Would help artists gauge their music from an unbiased (?) and single metric.
2. Would help producers asses their clients music with a single metric.
3. Would help music listeners with their choices of what to listen to.

2 Exploratory Data Analysis

2.1 Feature Correlation and Dataset Balance

The features are highly correlated from each other as observed in Fig. 1 and the dataset is imbalanced ($\approx$ 88%) (refer to Fig. 2). However the data is distributed to match the overall dataset balance when observed on a yearly basis (Fig 3 . To study the evolution and behaviour, we checked how the acoustic features evolve with time and with respect to songs on or not on the billboard.

Surprisingly, the range of values for most features do not vary with time, except acousticness, valence and energy. With the advent of electronics, it makes sense that songs have

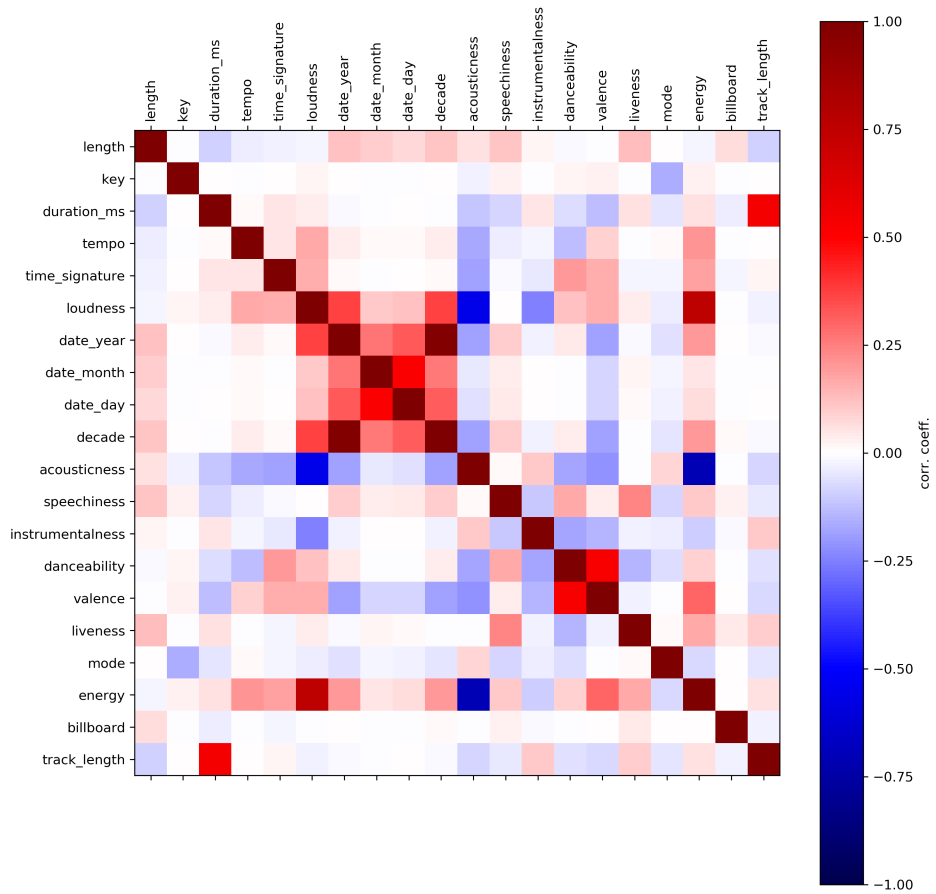

Figure 1: Feature correlation

departing from pure acoustic nature (Fig. 4) . Likewise, rock, metal, and more recently hip hop and electronic dance music have increased the energy levels of songs (Fig. 5 . Suprisingly, the valence (an indicator of the positiveness of a song has slightly decreased over the years (Fig. 6).

Due the low variablity of acoustic features and imbalance of the dataset, before we move to model evaluation, we expect the classifier to have a tough time (as can be guessed from the plot of the 1st two principal component (Fig. 7 where the points are highly overlapping)

3 Machine Learning Pipeline

3.1 Preprocessing

The following are the broad categories of pre-processing applied to the datasets:

1. The date values (date released and dates on the chart) were each parsed to python datetime object and split to 3 columns - day,month, year. No scaling is done as yet on these features.
2. The song_id and album_id are kept in the spotify unique id format. To be scaled upon further discussion.
3. Acousticness, danceability, energy, instrumentalness, liveness, mode, speechiness, valence features are already scaled between $[0, 1]$. No further pre-processing was required.

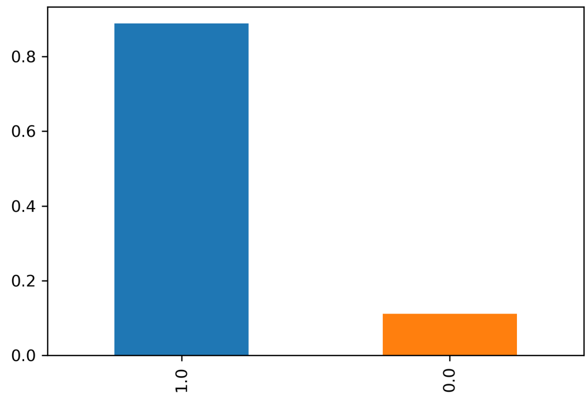

Figure 2: Dataset Class Balance

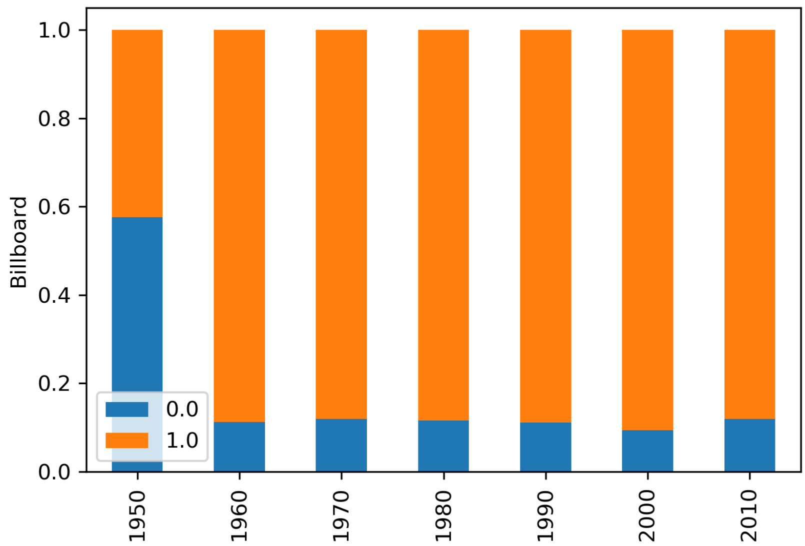

Figure 3: Track distribution over ages

4. Rank, track duration, key, tempo, time signature, loudness, album duration, album length, were all scaled using the StandardScaler.
5. The billboard label is added as 1 (if present) and 0 otherwise.

Note for the final calculation, I have ignored the month and day (since that is arbitrary and does not play any role when it comes to billboard consideration, and thus work only with the year. Since artist name, album name and song name was only used to curate songs, they are dropped from the final dataset before pre-processing.

3.2 Cross Validation Pipeline

In this study, we are trying to predict whether future songs would appear on the billboard, hence time of the song is not a factor in the requirements. We could have dropped the date feature but would anyway expect it have least importance.

However the data is highly imbalanced. Therefore the choice of cross-validation pipeline was chosen to be a stratified KFold Cross Validation, where the global distribution of classes

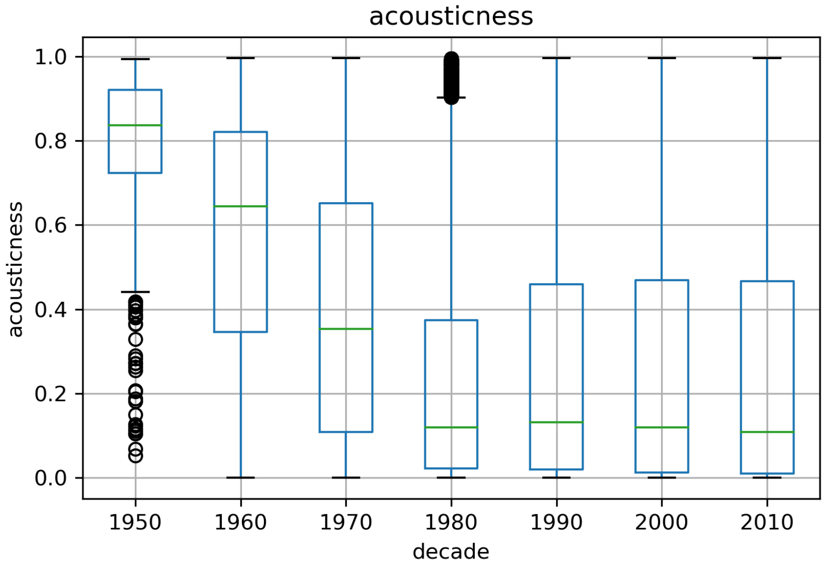

Figure 4: Acousticness evolution with time

is maintained in the K fold train,CV splits. The result of the splits are shown in Fig. 8.

3.3 Model Selection

Due to the large number of datapoints, certain popular techniques such as K-nearest Neighbours, and Support Vector Machine Classification failed to converge after 12 hours. Other methods such as Linear and Quadratic discriminant analysis failed to improve model performance over the baseline. For each of the model, we performed an exhaustive search for the parameter value in each estimator. Details of which is also described under each method.

Evaluation Metric: For all the models, we used the accuracy score since it is a binary classification problem.

In this report, we shall discuss 3 methods which showed improvement above the baseline. The 3 methods are:

1. Random Forest Classification
2. XGBoost Classification
3. AdaBoost Classification

3.3.1 Random Forest Classification

Test Baseline: 0.8807
Test Score: 0.8865
Optimal Parameters:
n_estimators: 100, criterion: gini, max_depth=10, min_samples_split: 3

The normalised confusion matrix is shown in Fig. 9
The feature importance under random forest is shown in Fig. 10

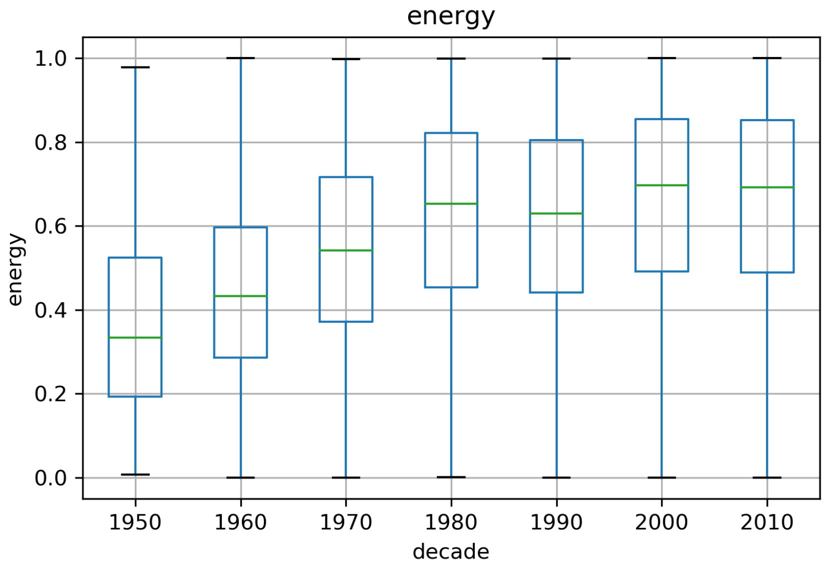

Figure 5: Energy evolution with time

3.3.2 XGBoost Classification

Test Baseline: 0.8807
Test Score: 0.8865
Optimal Parameters:
max_depth=15, gamma: 0.4, min_child_weight: 1, learning_rate: 0.3 . . .

The normalised confusion matrix is shown in Fig. 11

The feature importance by based on F score (as per SelectFromModel) under XGBoost is shown in Fig. 12

3.3.3 AdaBoost Classification

Test Baseline: 0.8807
Test Score: 0.8865
Optimal Parameters:
algorithm: SAMME.R, gamma: 0.4, n_estimator: 100, learning_rate: 0.1

The normalised confusion matrix is shown in Fig. 13

The feature importance under AdaBoost is shown in Fig. 15

The tree estimator with least error of the adaboost classifier is show in Fig. **??**

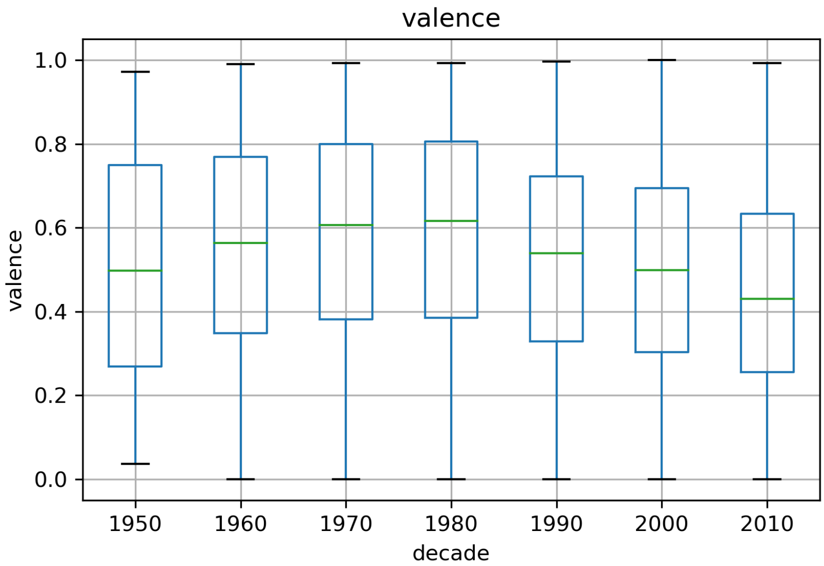

Figure 6: Valence evolution with time

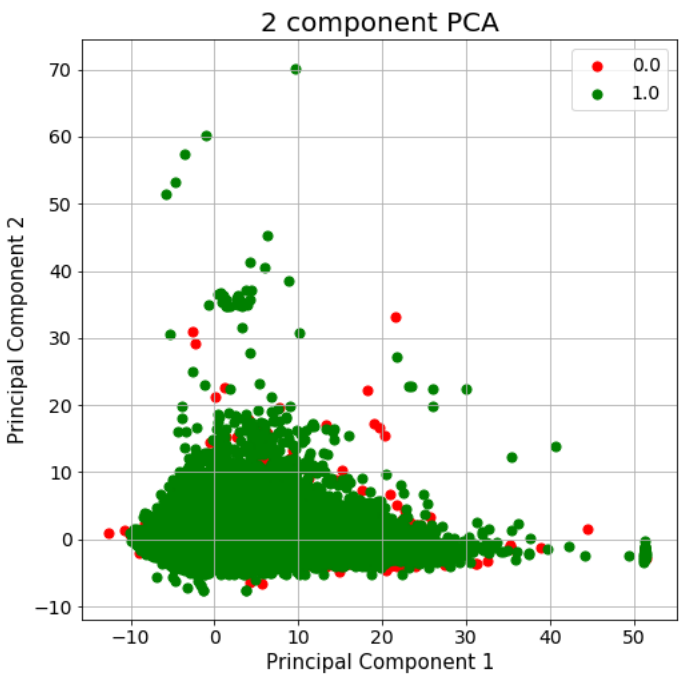

Figure 7: Plot of 1st two principal component

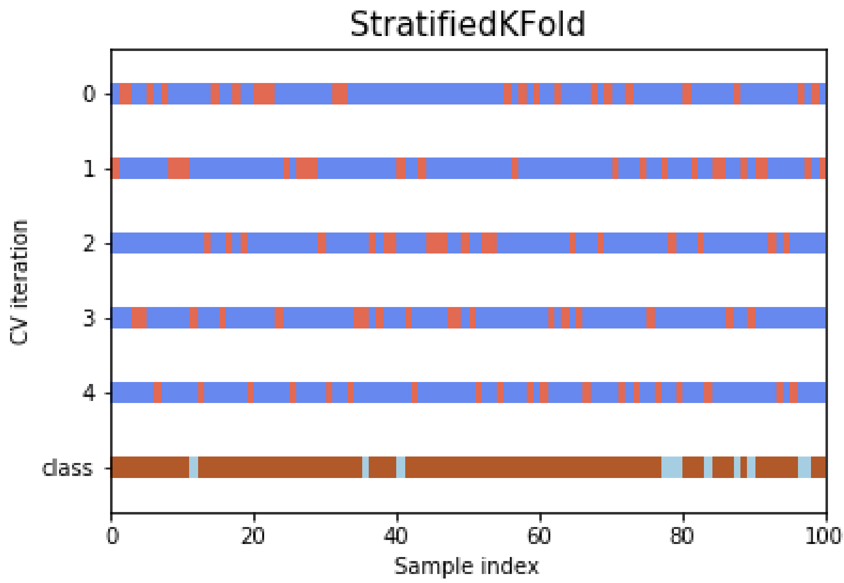

Figure 8: Cross Validation Folds. Training set is shown in blue, CV set in red.

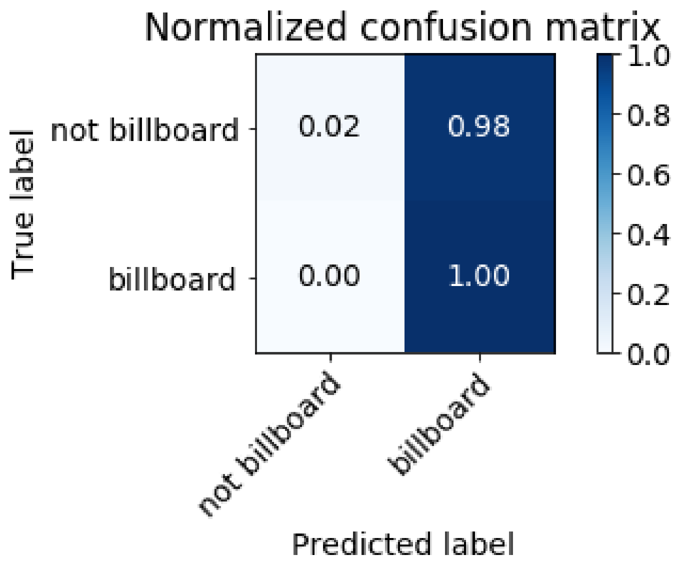

Figure 9: Normalised confusion matrix for Random Forest Classification

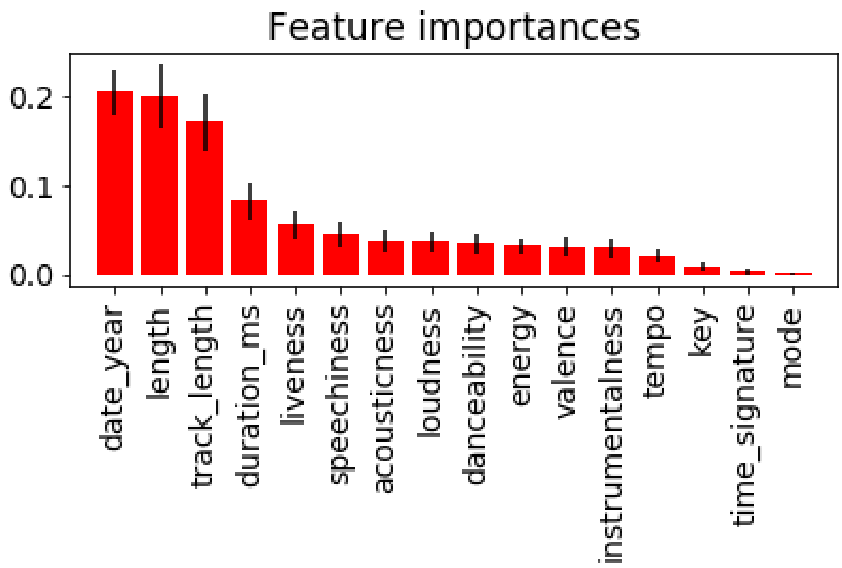

Figure 10: Feature importance for Random Forest Classification

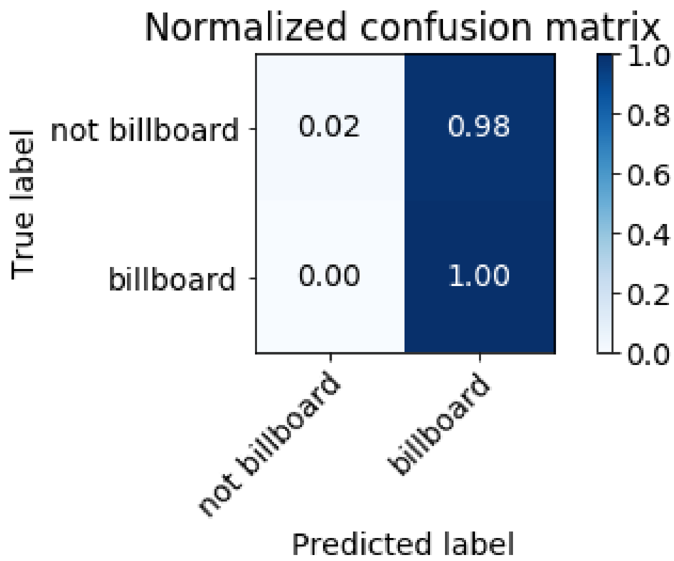

Figure 11: Normalised confusion matrix for XGBoost Classification

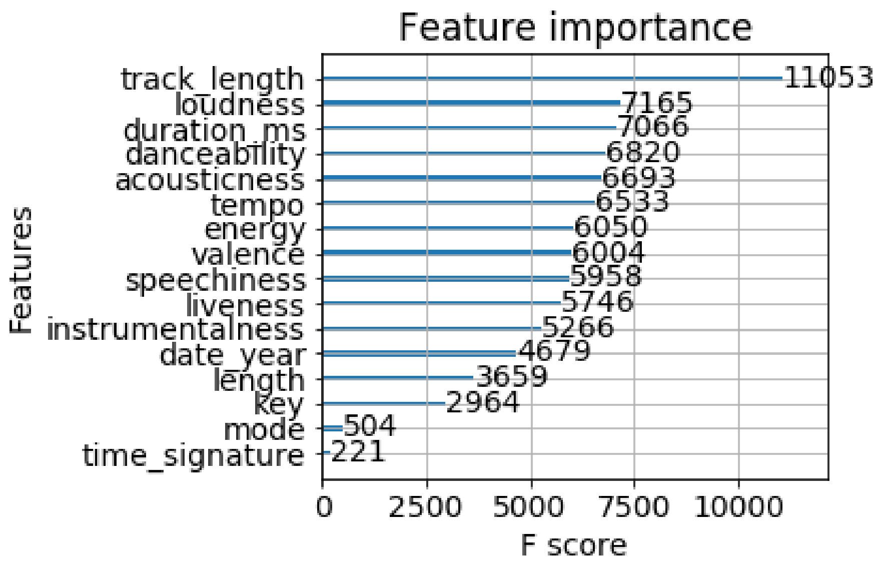

Figure 12: Feature importance by SelectFromModel for XGBoost Classification

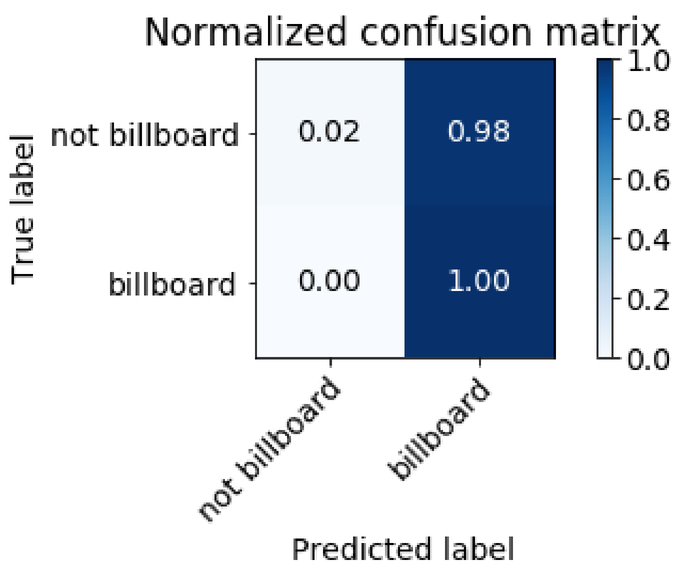

Figure 13: Normalised confusion matrix for AdaBoost Classification

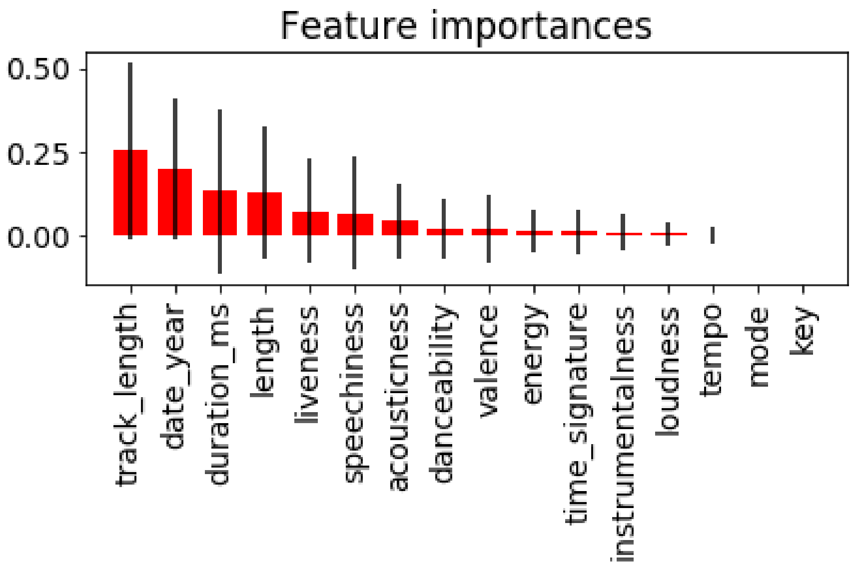

Figure 14: Feature importance by SelectFromModel for AdaBoost Classification

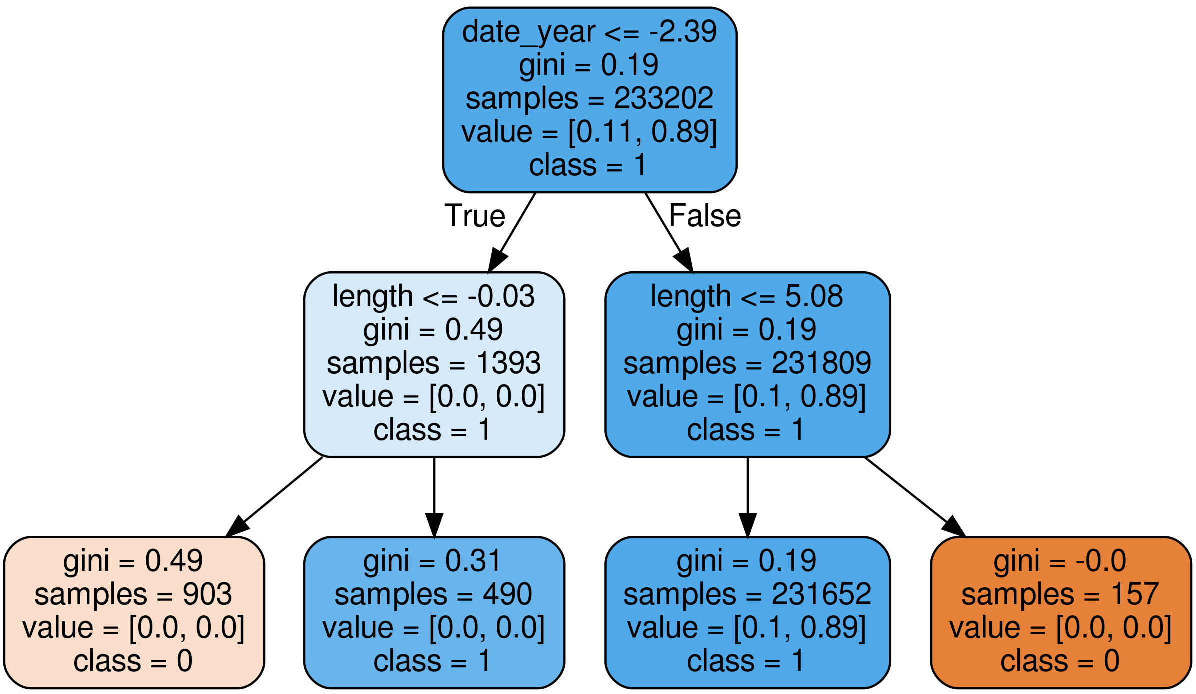

Figure 15: Estimator with least error