**Introduction:**

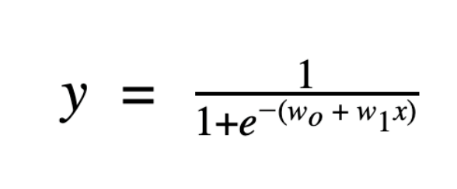
In this project, we are interested in making a model that can predict songs which are to the taste of a user using the ‘Training Data Set’ of the user. Such prediction models are used by all big production and media services providers like Spotify, Netflix and even Facebook.

Training Data set is the data of the all the songs listened to, by the user and are liked or disliked depending on different features of the songs. Different Machine learning algorithms are applied to train the model and these algorithms uses different features like songs duration, loudness, liveness, speechess, time signature, instrumentalness and some other features to make a reliable predictive model.

To get best model, we tried using Logistic Regression, Random Forest, Boosting and KNN method which are discussed below in detail.

**Methods:**

1 - Logistic regression is used for classification problems as an alternative for the linear regression method. The method practices sigmoid function on linear regression to calculate the output to be restricted to [0,1] range in our case it’s like/dislike which is estimated probability. The mathematical form for the model is



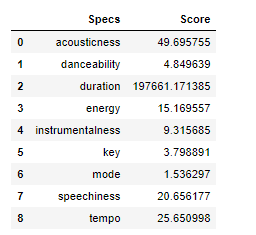
There are many types of logistic regression and the one we used was the binary logistic regression which has only 2 possible outputs/classification categories.

We used sklearn.linear\_model for applying logistic regression and adjusted the class\_weight to 'balanced' which uses y to adjust the value of the weights. Then we use fit() method to train the model with our training data by giving the model the n\_samples and the n\_classes(0/1). Our model is lying on regularized logistic regression with 2 classes only as a result we choose the ‘liblinear’ library for the solver. In addition, we raised the iterations to 1000

We almost tried most of the method and the linear regression gave us the least accurate results among other methods with 0.7933333333333333

2 - Another method we explored is Random Forest Classifier, also known as Bootstrap Aggregation.

How it worked? In simple words it worked on combining multiple models to produce the result. This technique is called Ensembles Technique.

Random Forest use multiple decision trees to get the output. From our training data, multiple decision trees were created randomly. As this method works on randomness it reduces the variance and produce more accurate model as compared to some other models. Each decision tree then produces an output and by taking the mean of the output of each decision tree, we get the final output of the Random Forest; this process is called aggregation. Please note that here we are using Random Forest for classification.

We can also control the number of decision trees by passing the argument **n estimators** tothe algorithm used. In our case, we created 100 decisions trees, since it was giving us the most accurate model for the training set data as compared to 99 or 101 decisions trees.

We use **SelectKBest** method to choose all those features which contribute more to our model using Random Forest. This function uses two important arguments: chi2 and K. K is the number columns and chi2 (chi-squared) which uses some statistics and then informs us about the variables which is essential lot to the model. The higher statistical value means better variable for model.

Using this method, we get the highest accuracy of 0.844 without features selection, with features selection it reduces the accuracy to 0.832.

The visual representation of the first tree has been shared in appendix for reference.

3 - Bagging Classifier:

The method creates bags of data (random subsets of the training data) and use the bags with different data sets and train different models, then we use the same inputs for all models and we get the mean output.

The method gave the highest accuracy score by choosing the DecisionTreeClassifier()  as base\_estimator, by trying different number on estimators [figure 4] and comparing the accuracy using cross validation in each value we get that the best value is setting n\_estimators = 500 and random\_state = 8

4 - The k-nearest neighbors (KNN) postulates that similar things should be near to each other (close proximity) and predicts the outcome based on distance and k-value and it classifies the object by means of the majority voting of its neighbors \cite{altman1992american}. The significant part of this method is the selecting of the most suitable k for the problem, as well as the increasing accuracy and prevent from overfitting. For this reason, for the application of the KNN, a series steps were driven such that feature selection, feature scaling, normalization, dimension reduction (in this case PCA) and their combinations were utilized. Firstly, the dataset was divided into training and test subset as 80 and 20 percent respectively. In order to increase the accuracy, feature selection was done in a way that as shown in the Figure 1, the top 9 effective features (from ‘speechiness’ to ‘tempo’) were selected. In order to improve classification, as stated in \cite{nigsch2006bb}, feature scaling, a method for the normalization of features, can be additionally utilized for this purpose. Data standardization and subsequent data normalization were operated data scaled into the range in between 0 and 1. In this case, the test accuracy was obtained as 0.84 when K = 8.

After that, K-Fold cross validation was utilized for evaluation of the model in terms of whether there is an overfitting issue or not. In K-fold cross validation, data is divided into k different subgroups. K-1 subsets are used to train the model and the last one is, however, for test. The obtained average error value indicates the validity of the model. As shown in the below Figure 2, the suitable k of the preprocessed data (the dataset after feature selection, standardization and normalization) is at k = 25 with the accuracy of 0.808. This model then was used for estimation of the test data set and it is resulting in 0.775 accuracy

We used the bagging classifier method with the selected features, the bagging method outperformed the random forest method by 0.01.as it reduces variance, in other words it limits the data overfitting which gives accurate results. Also, It considers all the features in each node of the tree.

The bagging method with selected features got 0.81 accuracy on leader board.

**Refection Tasks:**

Machine Learning has an impact on the people in terms of both legal and ethical aspects when it is utilized for making the decision about such field like insurance. Machine learning engineers have to have legal responsibilities and be accountable for their products. Firstly, the product should provide two vital things, which are fairness and explainability, in order to supply the trustworthiness. However, this trustworthiness can be damaged by bias because bias means reflecting the social inequalities or something like it due to some of personal information such as age gender, race. Explainable system is that, for example, a system should be able to explain the decision of why people are not selected for credit or loan. In this case, for this reason, the dataset should be purified or minimized from bias in order to give better result. To do this, the machine learning engineer should explain the situation to the customers and educate them as well. The idea behind this is that minimization bias could be done by many methods, but the most effective is getting feedback from the customers. Besides, this case also rubs shoulders with tort law because this law compensates the any victims loses and provides product liability. In this case, the company might not make an insurance policy to someone who deserves (true negative). For this reason, engineers should provide bias information to the customers

Sometimes we don’t need to inform our customers about every small details or problems:

1 - We know models are working on the given data. So, it is possible that the model did not work well on the future unpredicted data. This can be kept as a secret from customer as we don’t know about their future data and the model that machine learning engineer design worked fine on the provided data and showed no biasness. Considering GDPR rules and the company policies, we may not inform the customers to avoid violations of the secret terms of the company.

2 - As we know no model is perfect, but some are useful. There will always be some small technical problems which may cause biasness, and these are not the issues we faced while testing but the ones we know it may occur in future. ML algorithms works differently on different data. One algorithm might work fine on the given data set, but the other doesn’t. As, if the complexity of the data increased the model may show some biasness.

CONCLUSION:

In this project we looked at the data set of the songs liked or disliked by Andreas Lindholm. We use different methods to make a model which can predict his future preferences of the songs. First, we tired Logistic Regression and train our model, but the accuracy of the model was not that good. Then we used KNN method to train the model using the same data, the accuracy improved from 0.602 to 0.808 but it’s not what we expected. We also used Random forest with selected features, and it gives a more accurate model. This time the accuracy was 0.832 better than the previous ones. But the method which gives us the highest accuracy is Bagging. Please check the below table for values comparison between different models.

A picture containing text, black

Description automatically generated