**Introduction**

Make an R Shiny application.

**Dataset used**

Dataset used from [*https://www.kaggle.com/leonardopena/top50spotify2019*](https://www.kaggle.com/leonardopena/top50spotify2019)

**Algorithm used for this dataset**

Clustering

**Mathematical/statistical details of the algorithm**

Clustering – K Means Clustering is common a data mining technique to create clusters of data that can be identified as “*data with some characteristics*”. Since we do not have outlier from the data so we do need to remove outlier step and can proceed to the next step

Principal component analysis (PCA) is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables (entities of which takes on various numerical values) into a set of values of linearly uncorrelated variables called principal components. This transformation is defined in such a way that the first principal component has the largest possible variance (that is, accounts for as much of the variability in the data as possible), and each succeeding component in turn has the highest variance possible under the constraint that it is orthogonal to the preceding components. The resulting vectors (each being a linear combination of the variables and containing n observations) are an uncorrelated orthogonal basis set. PCA is sensitive to the relative scaling of the original variables.

**Shiny application giving end user options to change parameters/results and deploy**

<https://sreebitbos.shinyapps.io/SreeSpotify/>

**Check in your code into GitHub**

<https://github.com/sreebitbos/SpotifyR>

**Summary**

**What data you collected?**

Data from Music platform Spotify's Top 50 Songs in 2019 to predict the popularity of songs in the future based on a wide variety of characteristics.

**Why this topic is interesting or important to you? (Motivation)**

Music of all types of genres is always interesting and motivating. This was exciting assignment since it helps Learn how to make an intuitive Spotify + R Shiny app that tells users things like their music personality, their most popular artists, and album comparison

**How did you analyze the data?**

**Table

Description automatically generated**

**Chart, funnel chart

Description automatically generated**

**What did you find in the data? (please include figures or tables in the report)**

Chart, diagram, engineering drawing

Description automatically generated

Chart

Description automatically generated

From seeing the graph, we know that the data has been clustered into 6 categories with their own distinct characteristics.There are 6 big groups of popular songs that people hear via spotify in 2019. \* **Cluster 1** has high *beats per minute* and *danceability*. So we can say that cluster 1 containing the largest data of our grouping, is consisted of songs that upbeat and danceable with average length of music 167 seconds (under three minutes) which the lowest length compared to other clusters.

The Multiple Linear Regression tab displays the Top 50 Songs in 2019 dataset. This data was taken from Spotify. The tab then computes parameters of the independent variables and plots the regression model of Valence against Popularity. The tab also interprets results.

The Hit or Miss? tab uses the regression model computed in the Multiple Linear Regression tab to predict whether a song may be a hit or a miss. This prediction is based on many independent variables that characterize a song from Beats per Minute to the Energy of the song. The tab allows the user to use the sliding scale to input an appropriate value for the corresponding variables and see how the popularity level goes up or down. A song above a certain popularity level is considered a Hit song whereas a song below this level is considered a Miss.

I decided to choose this specific Spotify dataset, because it was small and had many independent variables. This topic and dataset was interesting to me, because I was very curious about patterns within musical characteristics and whether popular songs tend to have similar characteristics. However, the analysis showed that this is not always the case. I analyzed the data using a linear regression model and decision tree. I used the decision tree initially to explore the data and understand relationships that drive the popularity of a song. I decided to create a Hit or Miss predictor to allow functionality for a user to directly explore what variables strongly affect popularity and what variables do not affect popularity as much. I also decided to use a regression model due to the number of variables available. After conducting the analysis, I found that these independent variables are not very highly correlated with popularity. The variable that most strongly affected popularity was Valence, the mood of a song. In fact, it was inversely correlated so that the lower the Valence level, the more popular the song would be.