Music Classifier – Singer Identifier

G1-MusicDeciphers

**Data Science Capstone Project   
Exploratory Data Analytics Report**

Date:

[11/30/2020]

Team Members:

Likhil Kumar Rachuri - [lkr46@drexel.edu](mailto:lkr46@drexel.edu)

Nandakishor Reddy Pulagam - [np689@drexel.edu](mailto:np689@drexel.edu)

Nirupam Kumar Velagapudi – [nkv27@drexel.edu](mailto:nkv27@drexel.edu)

Soujanya Navuluru - [sn863@drexel.edu](mailto:sn863@drexel.edu)

Vuthej Krishna Reddy Verama Reddy - [vv334@drexel.edu](mailto:vv334@drexel.edu)

[The purpose of this report is to describe the exploratory data analytics. It includes five major sections:

1. Analyzing the basic metrics of variables: data types, size, descriptive statistics
2. Non-graphical and graphical univariate analysis: identifying unique value and counts, histogram, box plots, etc.
3. Missing value analysis and outlier analysis
4. Feature engineering and analysis: correlation analysis, dimensionality reduction, deriving new variables
5. Appendix]

**Analysis the basic metrics of variables**

[In this section, we identify all the variables in the dataset and conduct the basic metrics of the variables. What are the data types (numerical/categorical, discrete or continuous, ordinal or nominal) and size? Provide the descriptive statistics of the variables such as mean, standard deviation, min, max, percentiles, etc.]

In our dataset, we have the audio file in wav format which was preprocessed from mp3 format and then from the wav file, the important features are extracted and collected. These features will be used for modelling and inputs the model.

From the overall collection, we have the below segregation of attributes.

**Numerical attributes**: 'tempo', 'zero\_crossings', 'spectral\_bandwidth\_mean', 'spectral\_bandwidth\_var', 'spectral\_contrast\_mean', 'spectral\_contrast\_var', 'spectral\_centroids\_mean', 'spectral\_centroids\_var', 'spectral\_rolloff\_mean', 'spectral\_rolloff\_var', 'spectral\_flatness\_mean', 'spectral\_flatness\_var', 'chroma\_stft1\_mean', 'chroma\_stft2\_mean', 'chroma\_stft3\_mean', 'chroma\_stft4\_mean', 'chroma\_stft5\_mean', 'chroma\_stft6\_mean', 'chroma\_stft7\_mean', 'chroma\_stft8\_mean', 'chroma\_stft9\_mean', 'chroma\_stft10\_mean', 'chroma\_stft11\_mean', 'chroma\_stft12\_mean', 'chroma\_stft1\_var', 'chroma\_stft2\_var', 'chroma\_stft3\_var', 'chroma\_stft4\_var', 'chroma\_stft5\_var', 'chroma\_stft6\_var', 'chroma\_stft7\_var', 'chroma\_stft8\_var', 'chroma\_stft9\_var', 'chroma\_stft10\_var', 'chroma\_stft11\_var', 'chroma\_stft12\_var', 'chroma\_cqt1\_mean', 'chroma\_cqt2\_mean', 'chroma\_cqt3\_mean', 'chroma\_cqt4\_mean', 'chroma\_cqt5\_mean', 'chroma\_cqt6\_mean', 'chroma\_cqt7\_mean', 'chroma\_cqt8\_mean', 'chroma\_cqt9\_mean', 'chroma\_cqt10\_mean', 'chroma\_cqt11\_mean', 'chroma\_cqt12\_mean', 'chroma\_cqt1\_var', 'chroma\_cqt2\_var', 'chroma\_cqt3\_var', 'chroma\_cqt4\_var', 'chroma\_cqt5\_var', 'chroma\_cqt6\_var', 'chroma\_cqt7\_var', 'chroma\_cqt8\_var', 'chroma\_cqt9\_var', 'chroma\_cqt10\_var', 'chroma\_cqt11\_var', 'chroma\_cqt12\_var', 'chroma\_cens1\_mean', 'chroma\_cens2\_mean', 'chroma\_cens3\_mean', 'chroma\_cens4\_mean', 'chroma\_cens5\_mean', 'chroma\_cens6\_mean', 'chroma\_cens7\_mean', 'chroma\_cens8\_mean', 'chroma\_cens9\_mean', 'chroma\_cens10\_mean', 'chroma\_cens11\_mean', 'chroma\_cens12\_mean', 'chroma\_cens1\_var', 'chroma\_cens2\_var', 'chroma\_cens3\_var', 'chroma\_cens4\_var', 'chroma\_cens5\_var', 'chroma\_cens6\_var', 'chroma\_cens7\_var', 'chroma\_cens8\_var', 'chroma\_cens9\_var', 'chroma\_cens10\_var', 'chroma\_cens11\_var', 'chroma\_cens12\_var', 'mfcc1\_mean', 'mfcc2\_mean', 'mfcc3\_mean', 'mfcc4\_mean', 'mfcc5\_mean', 'mfcc6\_mean', 'mfcc7\_mean', 'mfcc8\_mean', 'mfcc9\_mean', 'mfcc10\_mean', 'mfcc11\_mean', 'mfcc12\_mean', 'mfcc13\_mean', 'mfcc14\_mean', 'mfcc15\_mean', 'mfcc16\_mean', 'mfcc17\_mean', 'mfcc18\_mean', 'mfcc19\_mean', 'mfcc20\_mean', 'mfcc1\_var', 'mfcc2\_var', 'mfcc3\_var', 'mfcc4\_var', 'mfcc5\_var', 'mfcc6\_var', 'mfcc7\_var', 'mfcc8\_var', 'mfcc9\_var', 'mfcc10\_var', 'mfcc11\_var', 'mfcc12\_var', 'mfcc13\_var', 'mfcc14\_var', 'mfcc15\_var', 'mfcc16\_var', 'mfcc17\_var', 'mfcc18\_var', 'mfcc19\_var', 'mfcc20\_var'

**Categorical attributes**: 'song\_id', 'language', 'target'

All the numerical attributes displayed above are continuous in nature and categorical are discrete

The size of the dataset formed and processed is about 22775 rows or instances with 127 columns including the target.

When looked at artist and genre data separately, we have

* 13116 instances for artists data
* 9659 instances for genre data

Descriptive statistics for each attribute are displayed below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **tempo** | **zero\_crossings** | **spectral\_bandwidth\_mean** | **spectral\_bandwidth\_var** |
| **count** | 22775 | 22775 | 22775 | 22775 |
| **mean** | 125.9319293 | 13895.12645 | 0.659301999 | 0.02408021 |
| **std** | 21.74614609 | 7754.808822 | 0.130154107 | 0.025115284 |
| **min** | 25.08722694 | 0 | 0 | 0 |
| **25%** | 112.3471467 | 9044 | 0.59860026 | 0.010638516 |
| **50%** | 123.046875 | 12409 | 0.680306479 | 0.016406462 |
| **75%** | 135.9991776 | 16793 | 0.746904373 | 0.026707816 |
| **max** | 287.109375 | 95105 | 0.96406414 | 0.222211954 |
|  | **spectral\_contrast\_mean** | **spectral\_contrast\_var** | **spectral\_centroids\_mean** | **spectral\_centroids\_var** |
| **count** | 22775 | 22775 | 22775 | 22775 |
| **mean** | 0.489445669 | 0.022954361 | 0.46962589 | 0.028362914 |
| **std** | 0.090788812 | 0.010367791 | 0.129386101 | 0.018004864 |
| **min** | 0 | 0 | 0 | 0 |
| **25%** | 0.433478867 | 0.015362374 | 0.384939502 | 0.017412829 |
| **50%** | 0.493698936 | 0.021121482 | 0.457851289 | 0.024269336 |
| **75%** | 0.54952265 | 0.028482717 | 0.549389694 | 0.033767412 |
| **max** | 0.968582204 | 0.137849941 | 0.922743634 | 0.210145849 |
|  | **spectral\_rolloff\_mean** | **spectral\_rolloff\_var** | **spectral\_flatness\_mean** | **spectral\_flatness\_var** |
| **count** | 22775 | 22775 | 22775 | 22775 |
| **mean** | 0.560466974 | 0.043550527 | 0.132646592 | 0.033433004 |
| **std** | 0.143964201 | 0.028070395 | 0.117009055 | 0.030522629 |
| **min** | 0 | 0 | 0.004903224 | 0 |
| **25%** | 0.471933706 | 0.024391385 | 0.067748982 | 0.018610372 |
| **50%** | 0.566289514 | 0.037645433 | 0.104933225 | 0.027027139 |
| **75%** | 0.659290957 | 0.055303458 | 0.158468053 | 0.037408765 |
| **max** | 0.973483138 | 0.228752151 | 1 | 0.24760884 |
|  | **chroma\_stft1\_mean** | **chroma\_stft2\_mean** | **chroma\_stft1\_var** | **chroma\_stft2\_var** |
| **count** | 22775 | 22775 | 22775 | 22775 |
| **mean** | 0.297242192 | 0.28856694 | 0.078912007 | 0.074784852 |
| **std** | 0.148445281 | 0.137803015 | 0.040605834 | 0.04000289 |
| **min** | 0 | 0 | 0 | 0 |
| **25%** | 0.189780317 | 0.187470816 | 0.049015313 | 0.046098772 |
| **50%** | 0.27794528 | 0.276006192 | 0.073946625 | 0.068253934 |
| **75%** | 0.385055229 | 0.375275701 | 0.104398202 | 0.097606547 |
| **max** | 0.974764764 | 0.999965668 | 0.231286794 | 0.225011408 |
|  | **chroma\_cqt1\_mean** | **chroma\_cqt2\_mean** | **chroma\_cqt1\_var** | **chroma\_cqt2\_var** |
| **count** | 22775 | 22775 | 22775 | 22775 |
| **mean** | 0.376058279 | 0.367729014 | 0.064880383 | 0.061596864 |
| **std** | 0.173313117 | 0.165798278 | 0.039558859 | 0.038187168 |
| **min** | 0 | 0 | 0 | 0 |
| **25%** | 0.246029477 | 0.2411939 | 0.033333343 | 0.031219602 |
| **50%** | 0.356775165 | 0.353944467 | 0.058319971 | 0.054826442 |
| **75%** | 0.482939422 | 0.472615257 | 0.090117737 | 0.084418085 |
| **max** | 1 | 1 | 0.23652926 | 0.21402581 |
|  | **chroma\_cens1\_mean** | **chroma\_cens2\_mean** | **chroma\_cens1\_var** | **chroma\_cens2\_var** |
| **count** | 22775 | 22775 | 22775 | 22775 |
| **mean** | 0.227013737 | 0.222981201 | 0.017187949 | 0.016183745 |
| **std** | 0.111385296 | 0.107176991 | 0.016831902 | 0.016954117 |
| **min** | 0 | 0 | 0 | 0 |
| **25%** | 0.145953303 | 0.14620899 | 0.005066301 | 0.004594518 |
| **50%** | 0.219083907 | 0.216953406 | 0.011664693 | 0.010398795 |
| **75%** | 0.292943856 | 0.28721611 | 0.024024277 | 0.021879818 |
| **max** | 0.934513234 | 0.894088628 | 0.136110302 | 0.162544256 |
|  | **mfcc1\_mean** | **mfcc2\_mean** | **mfcc1\_var** | **mfcc2\_var** |
| **count** | 22775 | 22775 | 22775 | 22775 |
| **mean** | -214.271128 | 61.53446082 | 6389.519639 | 1440.823172 |
| **std** | 201.1379883 | 44.79909646 | 7183.039318 | 1114.582404 |
| **min** | -1131.37109 | -192.5678101 | 0 | 0 |
| **25%** | -273.205902 | 27.75522995 | 2021.661255 | 601.7267151 |
| **50%** | -192.313309 | 65.23937225 | 3811.822998 | 1216.069214 |
| **75%** | -72.6412544 | 94.80488968 | 7535.159668 | 1996.036438 |
| **max** | 65.96318054 | 265.19104 | 68481.42188 | 18241.65039 |

**Non-graphical and graphical univariate analysis**

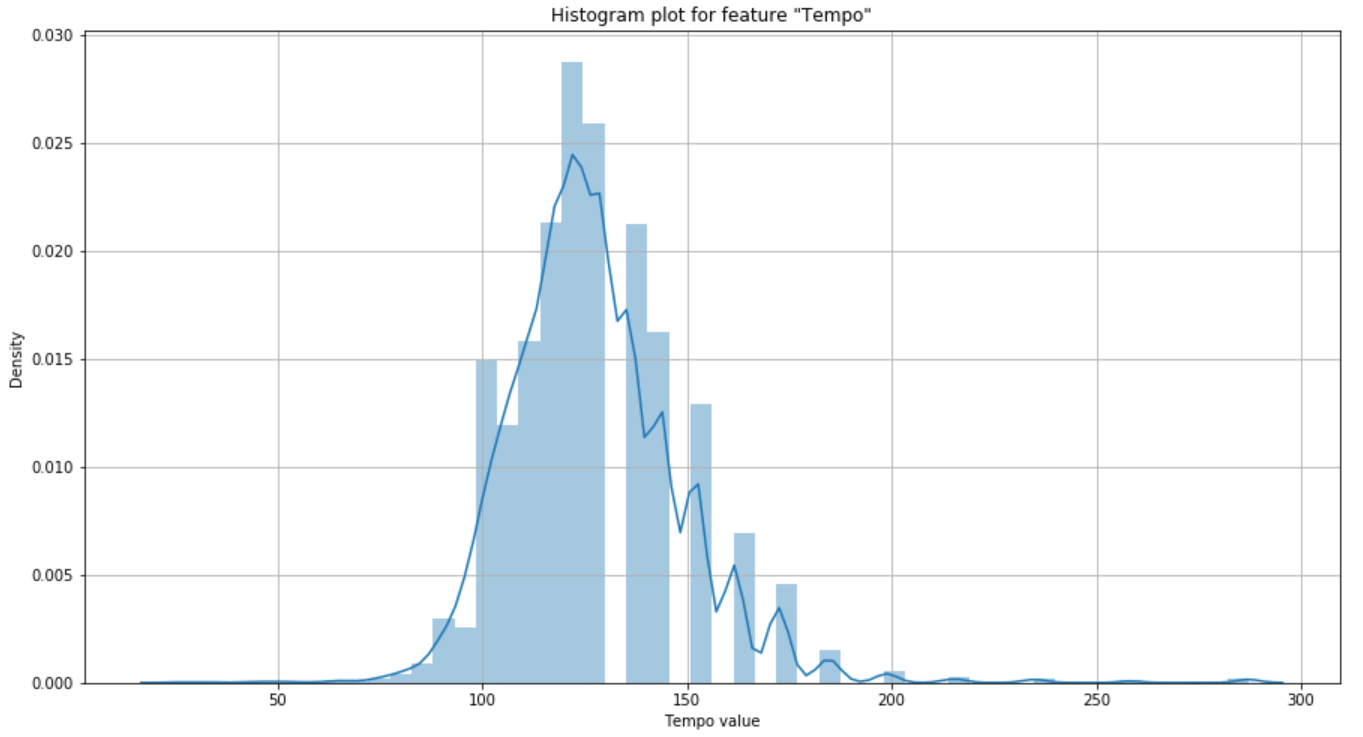
[In this section, we identify the list and number of unique values for each variable and provide the histogram and box plots to understand the distribution of the data.]

Below specified features are from the **Artist Data** which are then followed by **Genre Data**.

**Tempo**

This feature is extracted from the audio data, this speaks about the speed of the music present inside the audio. It is the rate of musical beat present in the audio, this is calculated using the reciprocal of the beat period. It is defined in the units of beats per minute (BPM).

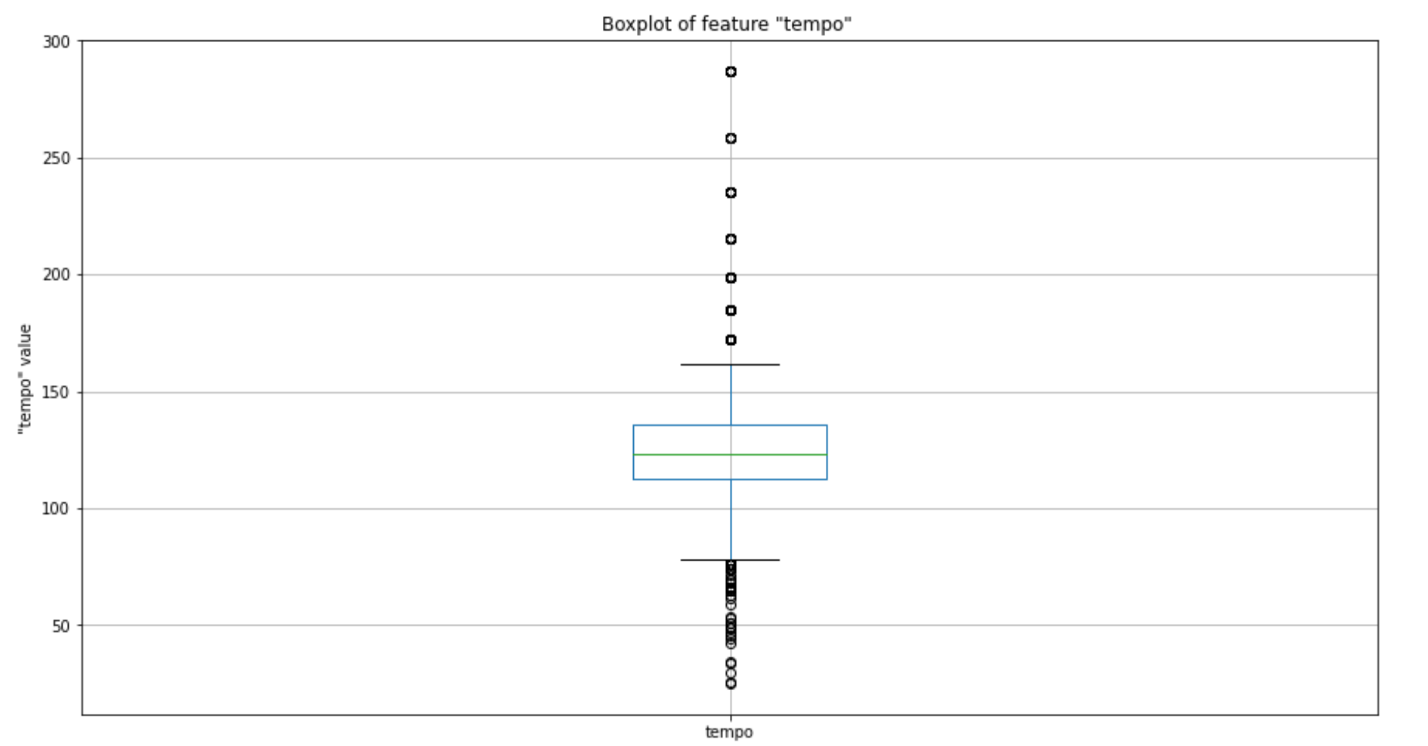
In our dataset, we find 48 unique values for tempo, and no null values are present.



On observing the histogram, we can say on a general it is following a normal distribution, with some breaks on the right side of the curve, whereas it is smooth on the left part of the curve. Overall, the curve is bell shaped and it is following a normal distribution with maximum frequency at approximately with tempo of 125 density of 0.025.

Boxplot distribution:

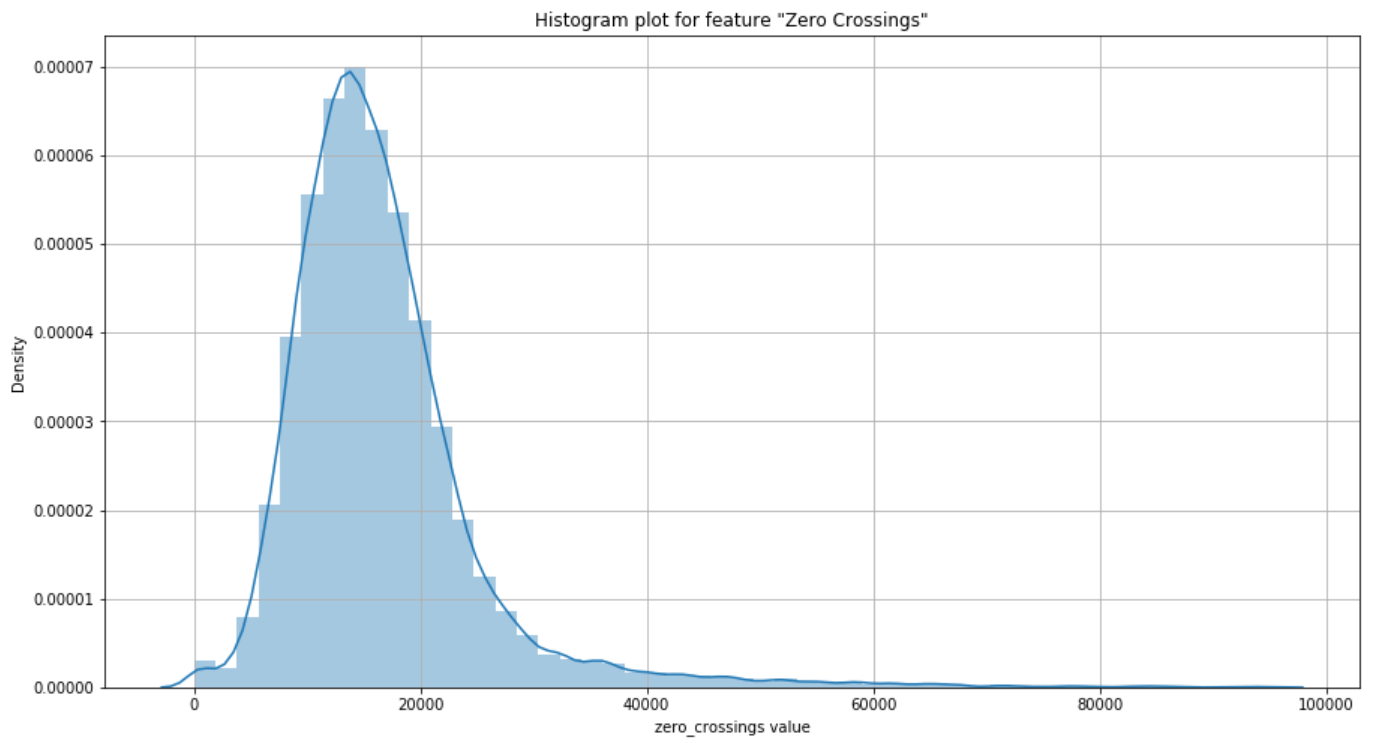
Below is the distribution of tempo variable using boxplot, on observing we can find that the data is more onto the downside so we can observe the less distance between the 1st quartile and minimum value. Most data i.e., from 1st quartile to 3rd quartile is lying in between tempo values of 75 to 160. We can also consider the values from the 3rd quartile to maximum as outliers. The average value or 50th percentile is located at value 125 which was also the same in histogram.



**Zero Crossings**

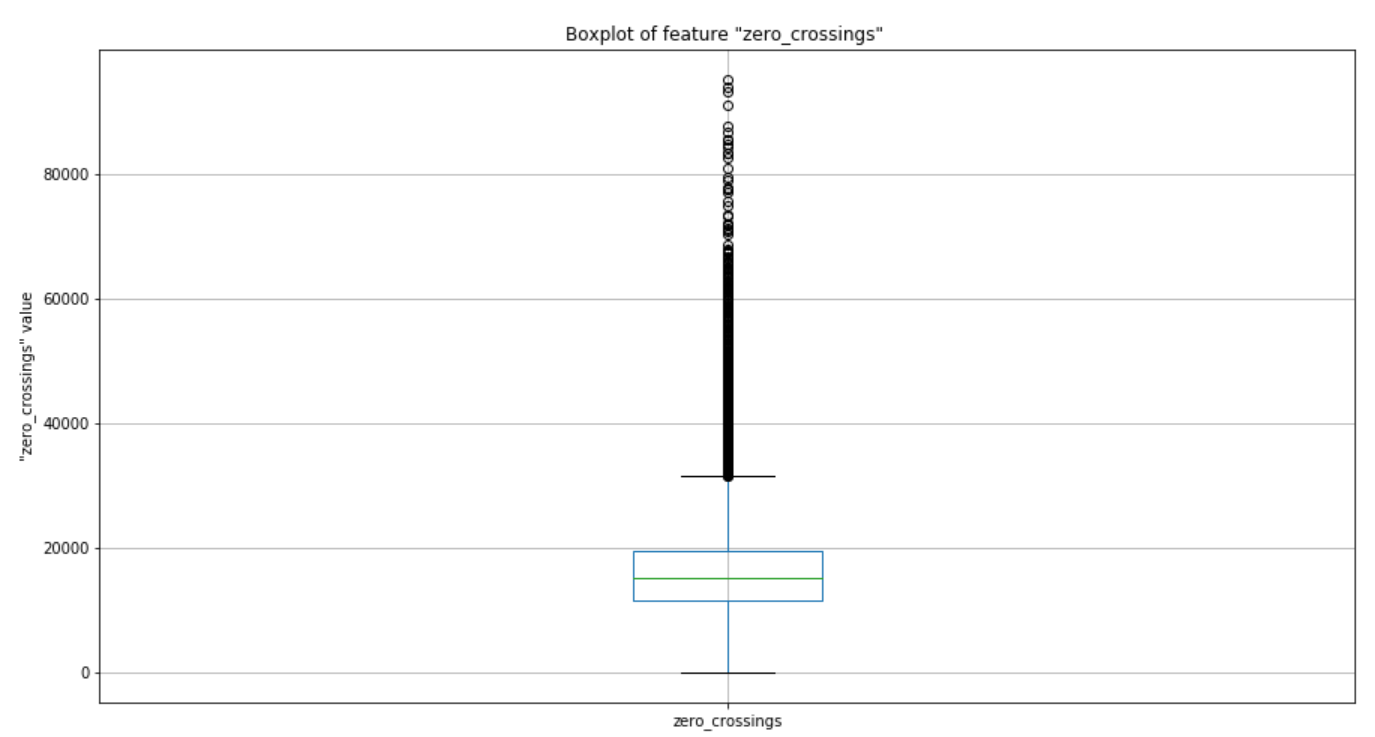
It is defined as the number of times the signal in audio is crossing the zero line, which indirectly speaks about the shift of values from positive to negative or vice versa.

The unique values for this attribute are 9872 and no nulls are present.



On observing the above histogram, we can infer that the histogram is of right skewed, Most of the songs do have the zero crossing rate of approximately 15000, the width of the curve is also very low which also conveys that 90% of the songs are in between 5000 and 25000 zero crossing rate value.

Box plot distribution:



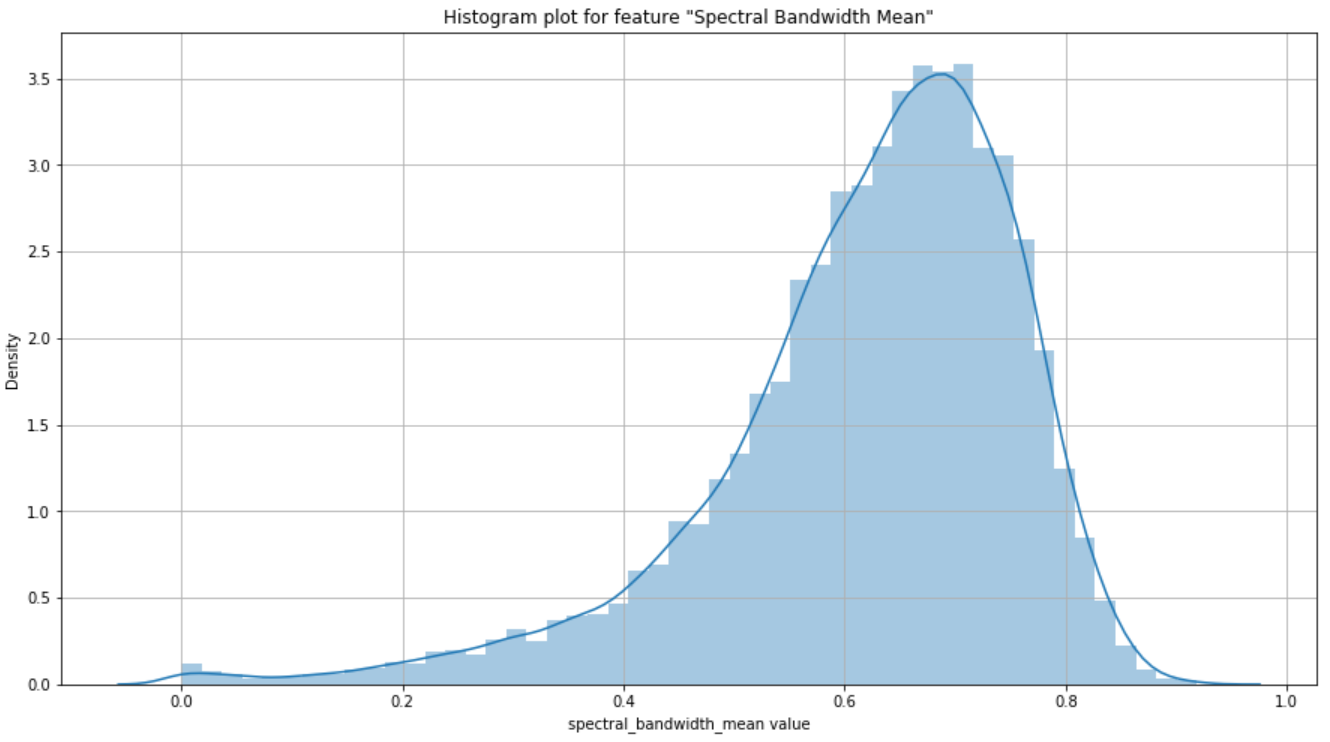
On observation we can infer that 50% of the data is present in the lower half of the plot. After 3rd quartile, we can observe the datapoints, but they are having a huge range from 26000 to 100000. The average value is 15000 which is similar to the value from the most frequent value obtained by histogram. Minimum is on zero and maximum is at 100000.

**Spectral bandwidth mean**

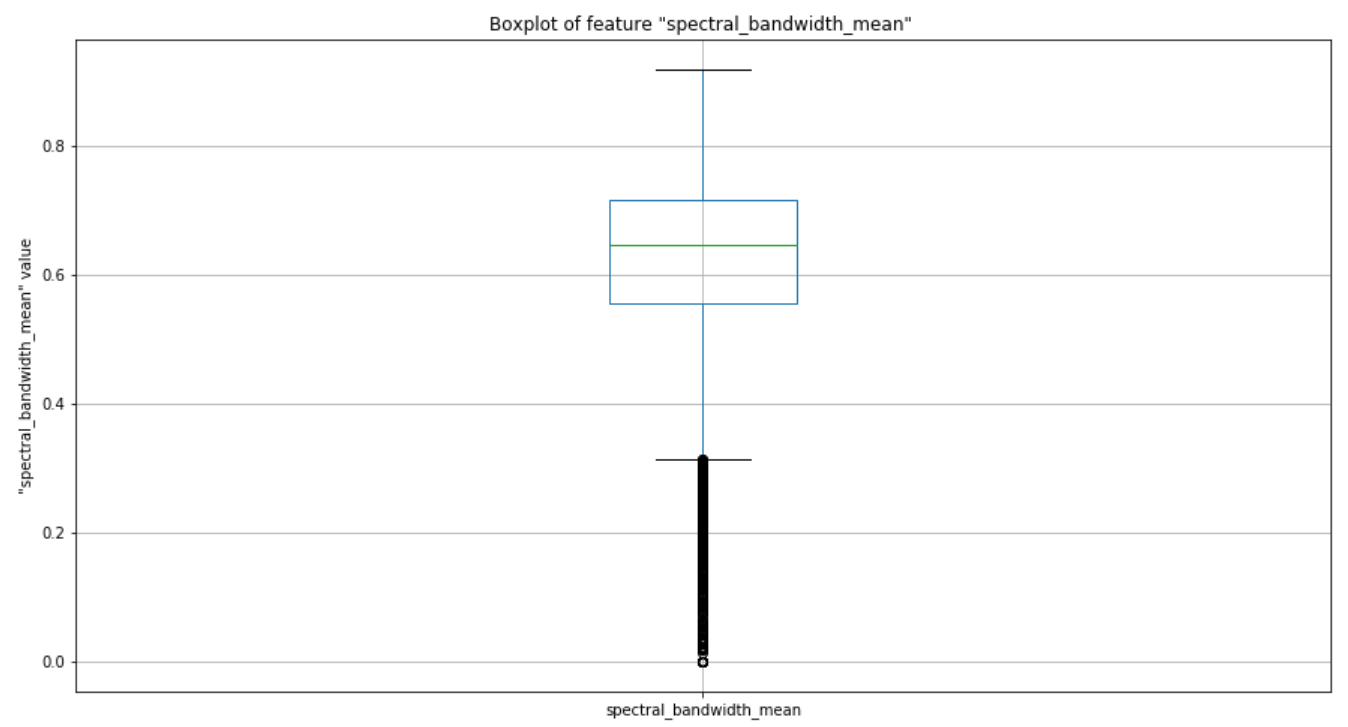
It is the range of values in which the signal is not less than the half of its maximum value. This feature is also important in analyzing the signal. The values are considered into our dataset after normalization.

The unique values for this attribute are 13080 with no null values present in our dataset.

On observing the below displayed histogram, we can observe that the graph is a left skewed curved distribution with a long tail starting from zero till 0.4. The highest occurrences are at 0.7 value with a density of approximately 3.6. The graph is denser at this area.



Boxplot distribution:

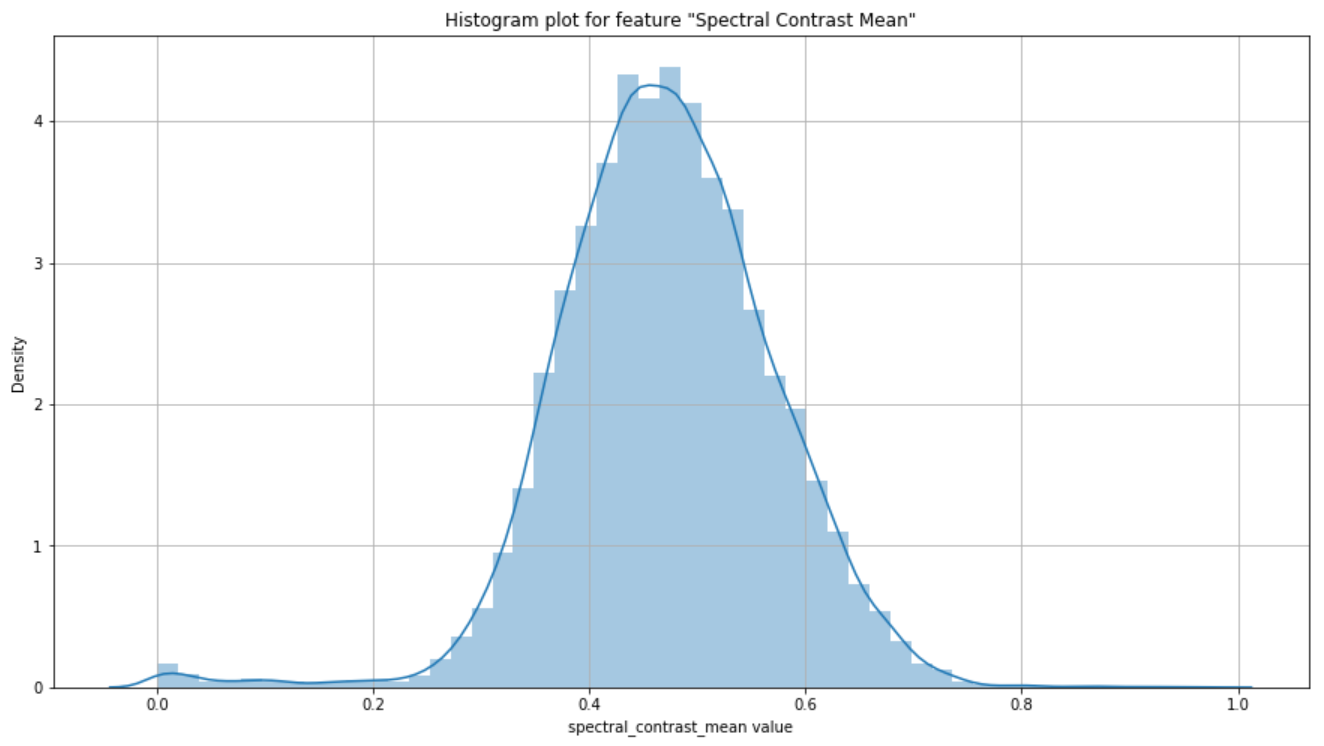


On observing the distribution, we can confirm this is a left skewed as we can see sparsely distributed values below the 1st quartile. The width for the 50% of the data is quite more in comparison with others with range from 0.3 to 0.7. The mean is located at 0.62 value and maximum at 0.87. We can observe that the maximum and 3rd quartile are closer than the distance between 1st quartile and the minimum value.

**Spectral Contrast mean**

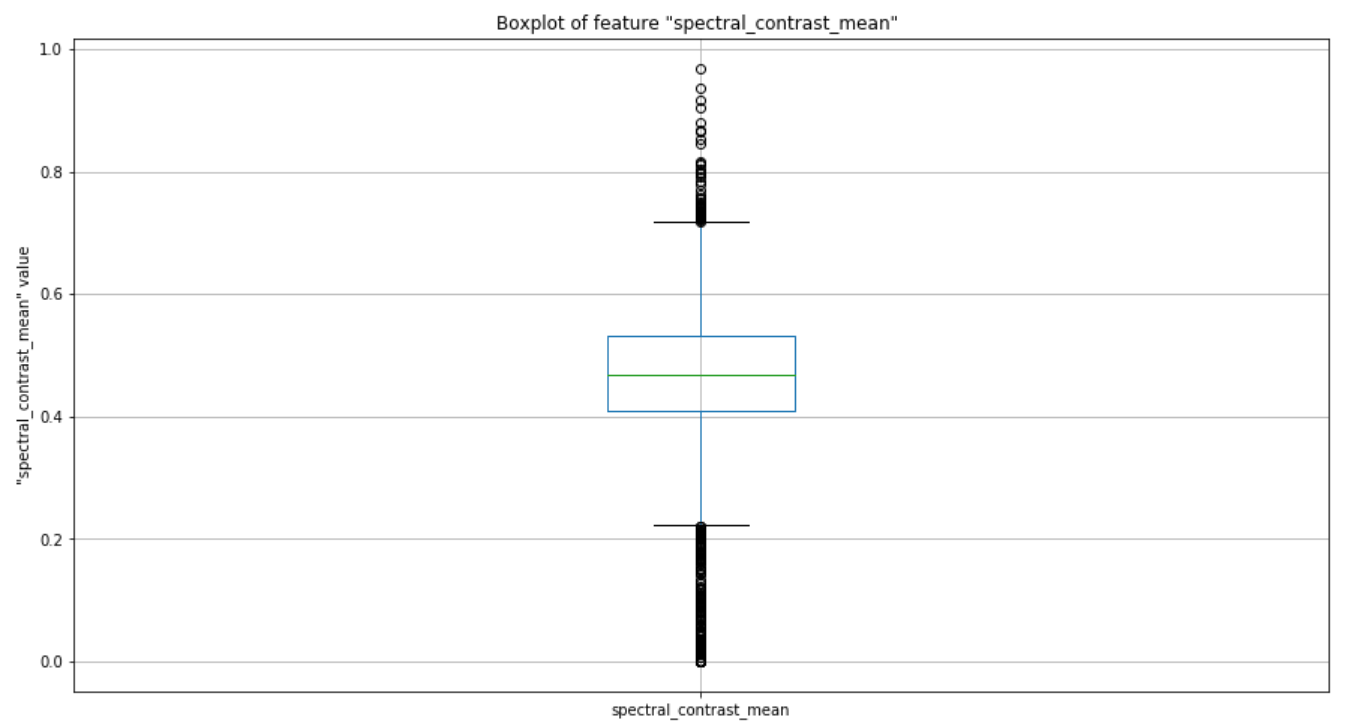
Spectral contrast is defined as the difference of peaks and valleys in the spectrum. It considers the spectral peak, valley and their difference in each sub-band.

The unique values present for this attribute in our dataset are 13080 and no nulls are present for this attribute.



On observing the distribution of histogram, we can state that it is following a clear normal distribution, this looks so close to the actual normal distributed curve. Mean is at the 0.5 with almost the density of 4.5.

Box plot distribution:

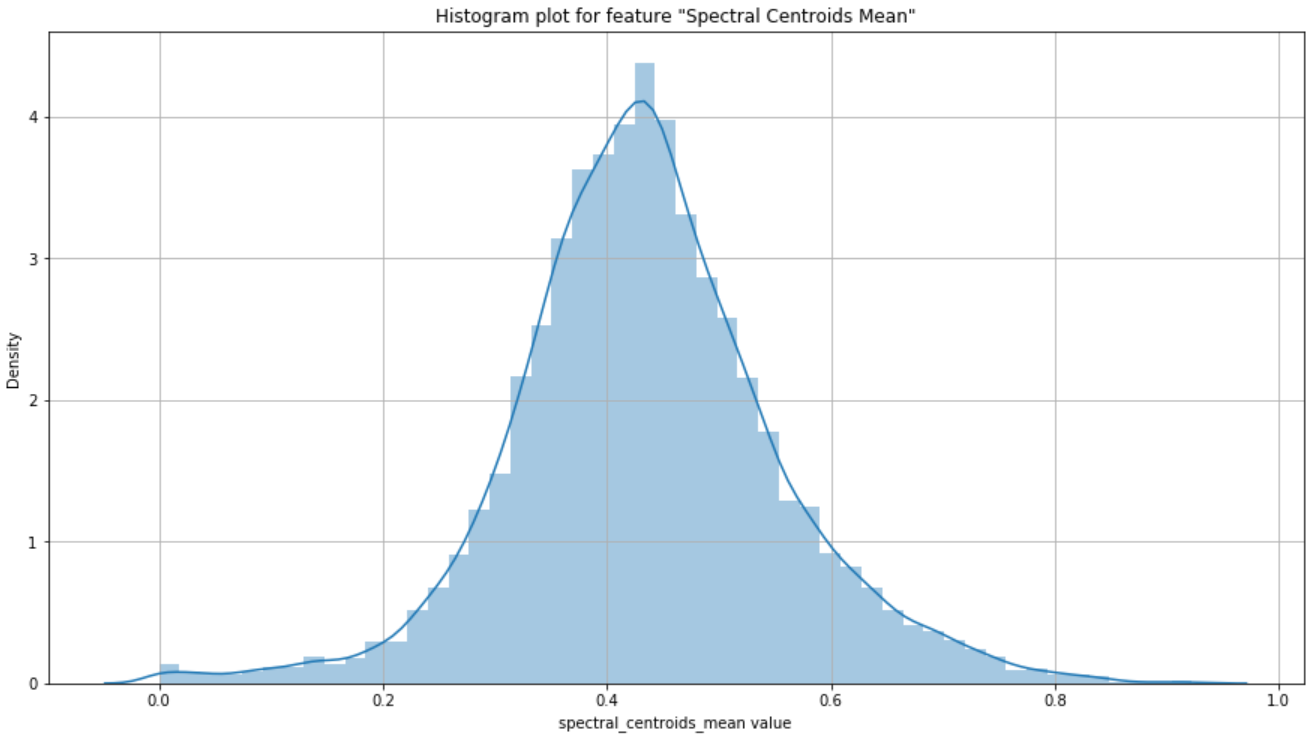


On observing the boxplot above, we can state that the mean is located at exactly in the center at 0.4. with equal distribution on both the sides, still there are less instances on top when compare to bottom but the difference seems very less and can be negligible. The 1st quartile and 3rd quartiles are at 0.41 and 0.52 points respectively.

**Spectral centroids mean**

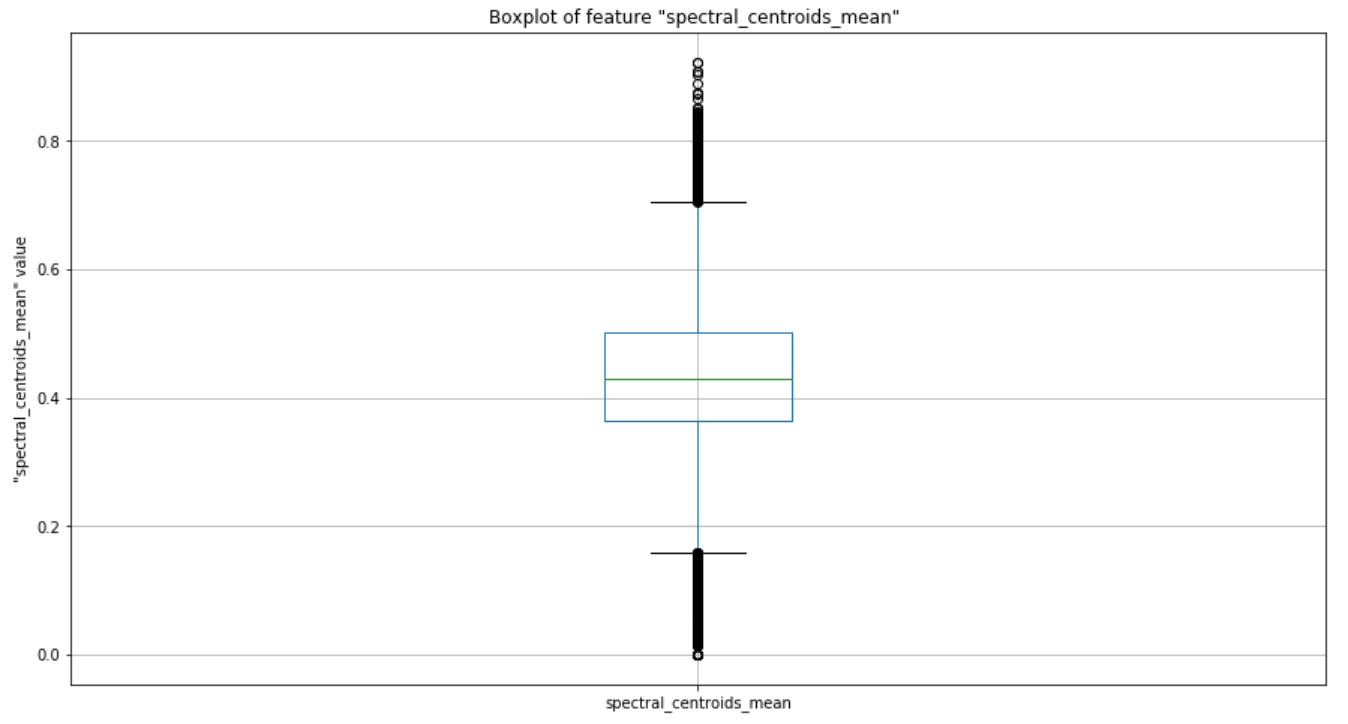
It is a metric in digital signal processing, where spectral centroids are defined as the points where the center of mass for the spectrum is located. This is an important feature for our data as we may depend on the center of mass for each song which differentiates them accordingly.

There are 13080 unique values and no nulls are present for this attribute in the dataset.



On observing the histogram, we can infer that it is almost very near to the normal distribution with slightly more data onto the left and tailed onto the right. Overall, we can state this as the normal distribution as the small tail over the right can be negligible. The mean is at approximately 0.45 with a density of nearly 4.7.

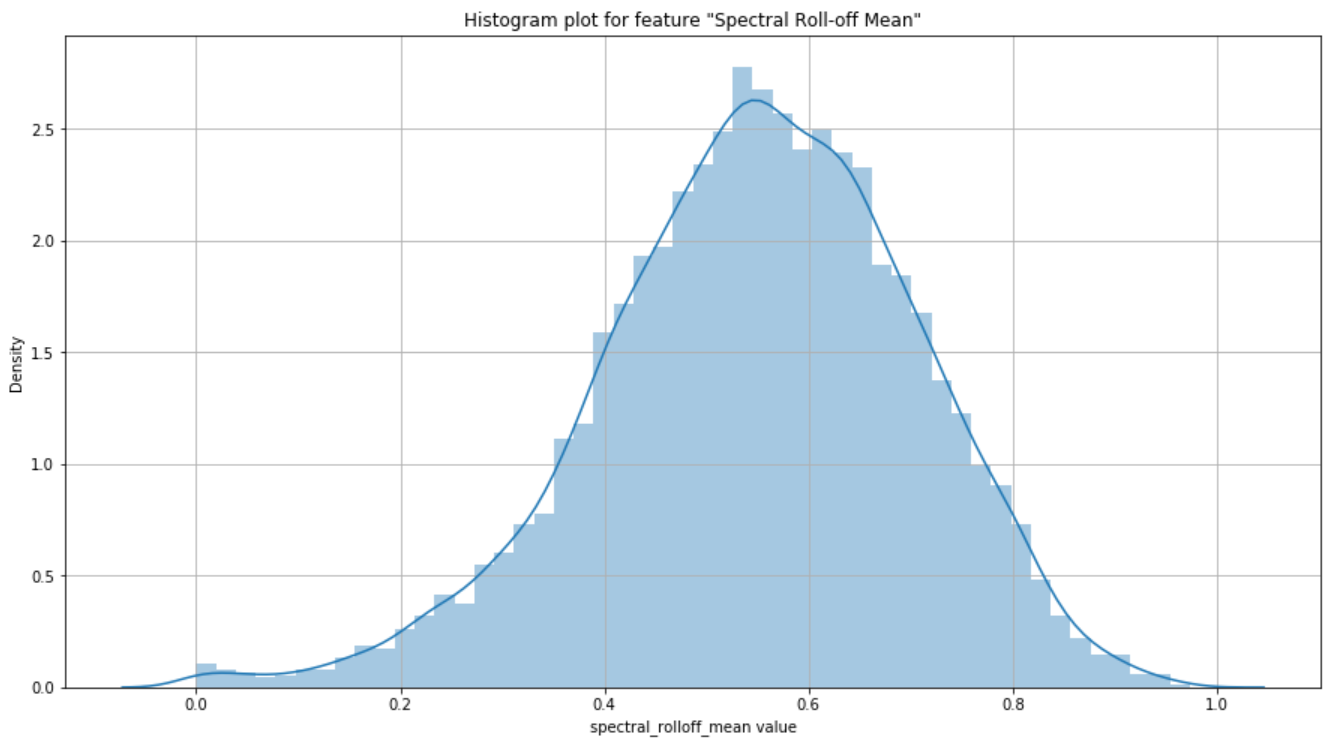
On observing the below specified boxplot distribution, we can again iterate that the distribution is almost near to the normal distribution. The difference between the 1st quartile to 3rd quartile which is the 50% of the data is also uniformly distributed on to both the sides. The 1st quartile is located at approximately 0.38 and 3rd quartile is located at 0.7 respectively.



**Spectral rolloff mean**

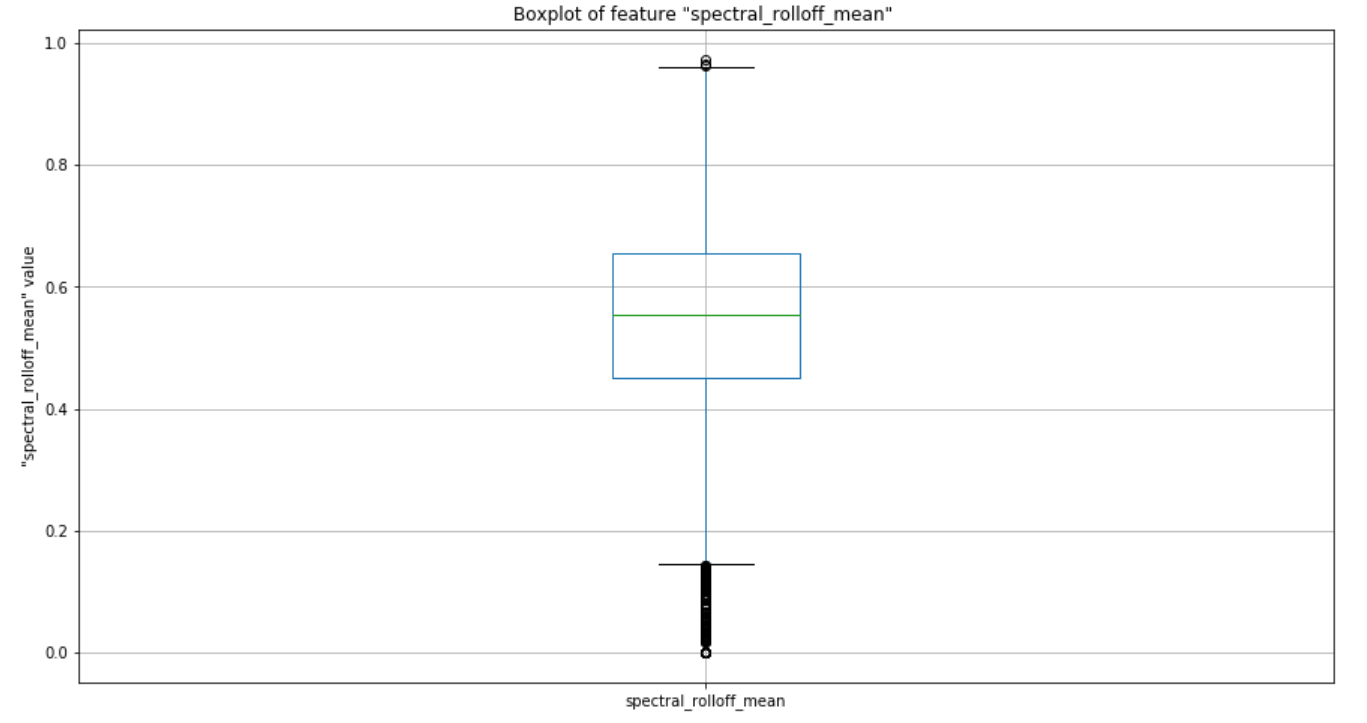
Spectral rolloff is the range where in 85% of the spectrum energy falls into. This gives us the value where all the energy is located which helps us in differentiating the songs and thus helps in our modelling.

There are 13073 unique values and no nulls are present for this attribute in the dataset.



The above histogram tells us that the data distribution is normal and uniform all over the curve and is very close to normal. There is a slight tail to the left by which it might tend to be a slightly left skewed, but on observing the dataset we can neglect that slight difference. The highest occurred value is at 0.52 with a density of 2.8 approximately.

Box plot distribution



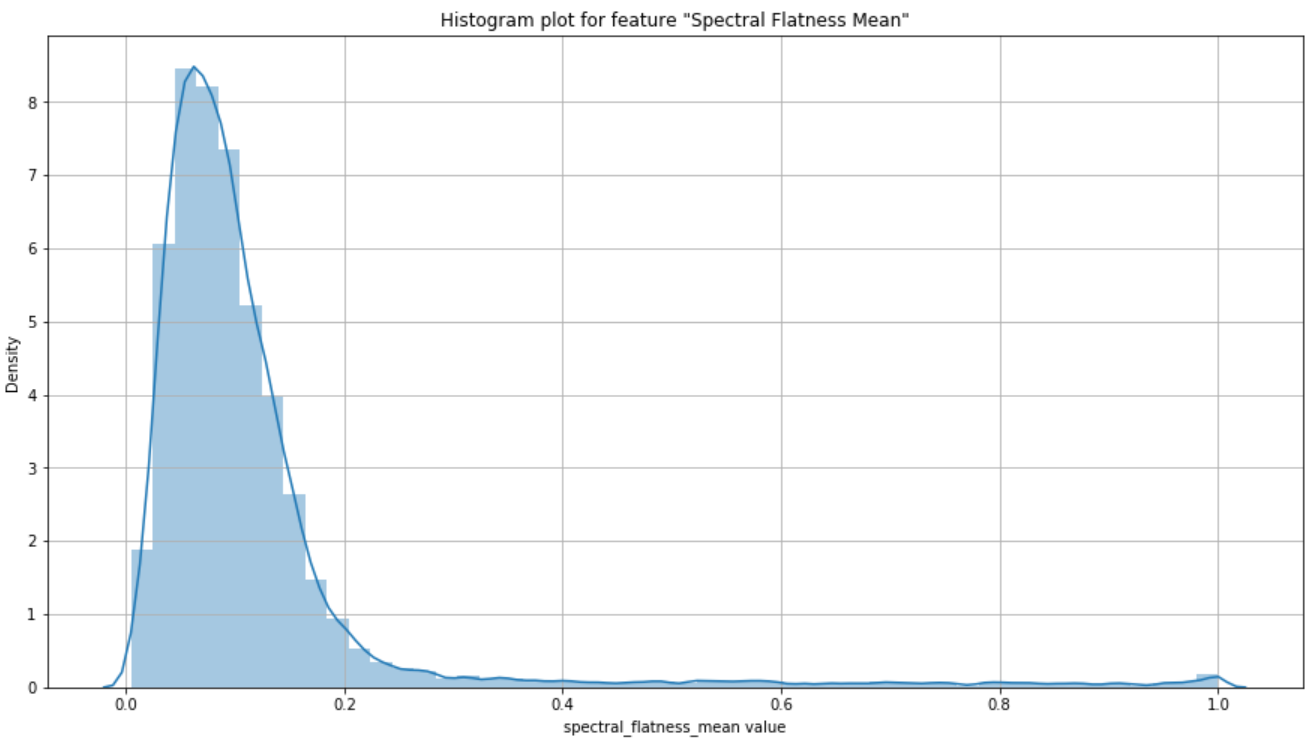
On observing the above boxplot distribution, we can infer that the data is slightly left skewed in distribution as we can observe there are few data points below the 1st quartile, and very few above the 3rd quartile. The data between the 1st and 3rd quartiles is very large, and the mean is located at the 0.57 approximately.

The Q1 and Q2 ends are at 0.43 and 0.63 respectively.

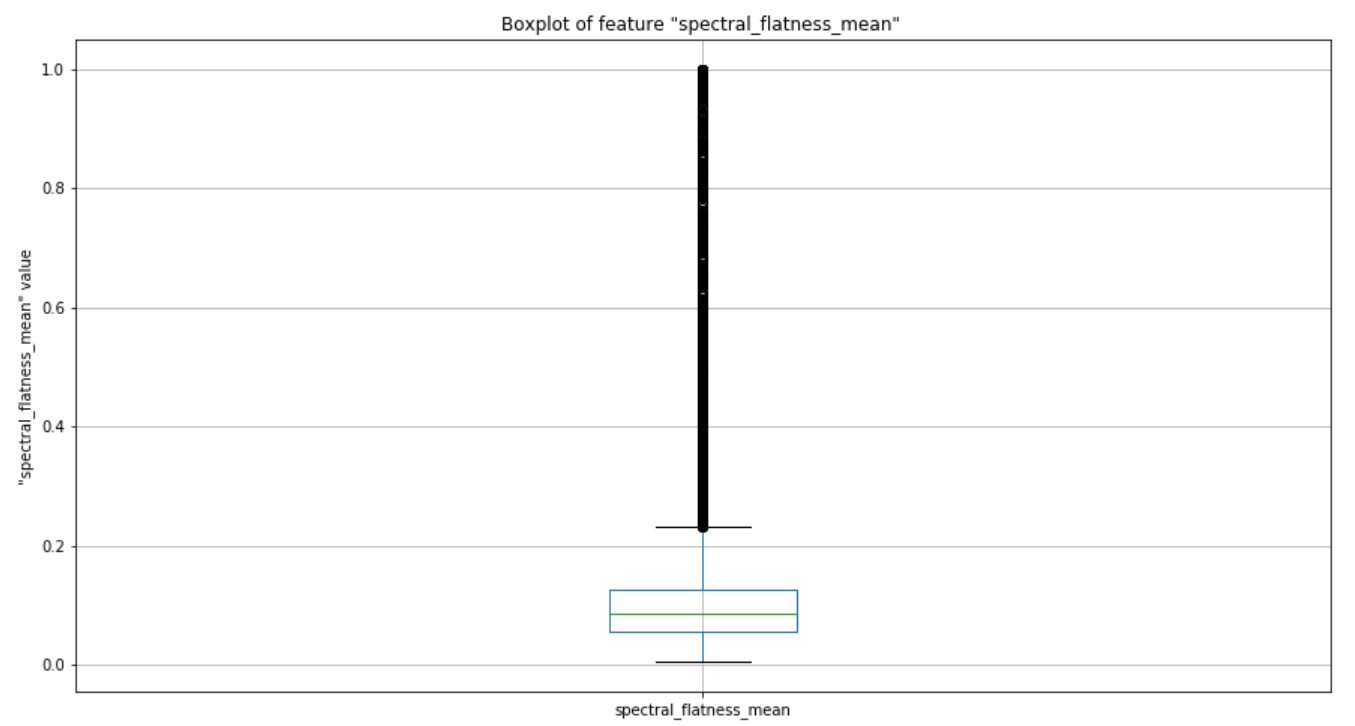
**Spectral Flatness**

In order to capture the presence of noise within the data, ‘Spectral Flatness’ measure can be used. It measures the amount of noise that is present in the input audio rather than the presence of toned data.

This value takes the range of 0 to 1. Higher ‘Spectral Flatness’ value (1) indicates the data being similar to White-Noise whereas a lower value (zero) indicates the absence of White-Noise.



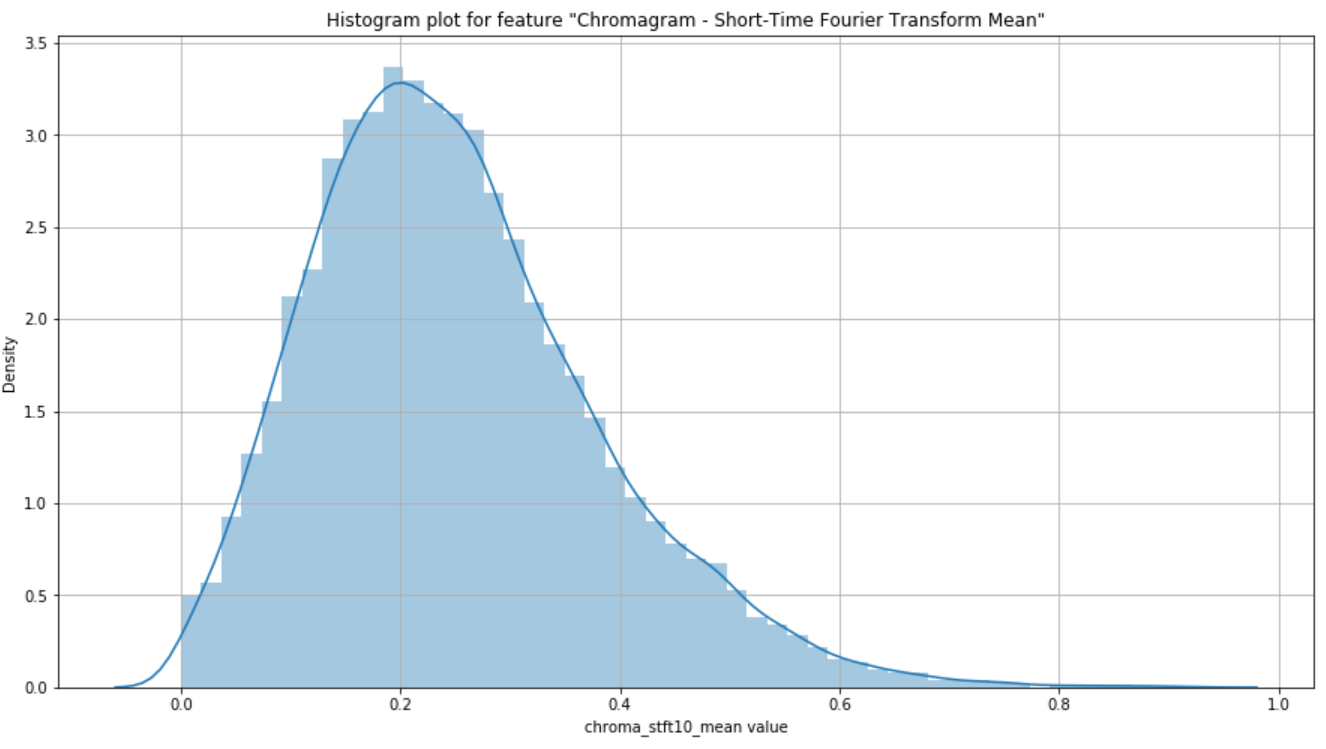
From the above Histogram, we can depict that there is very little White-Noise in the data as the majority of the data is observed to be farther away from 1 and much closer to 0. Amongst 13105 instances, we can observe 13076 unique values and 0 nulls.



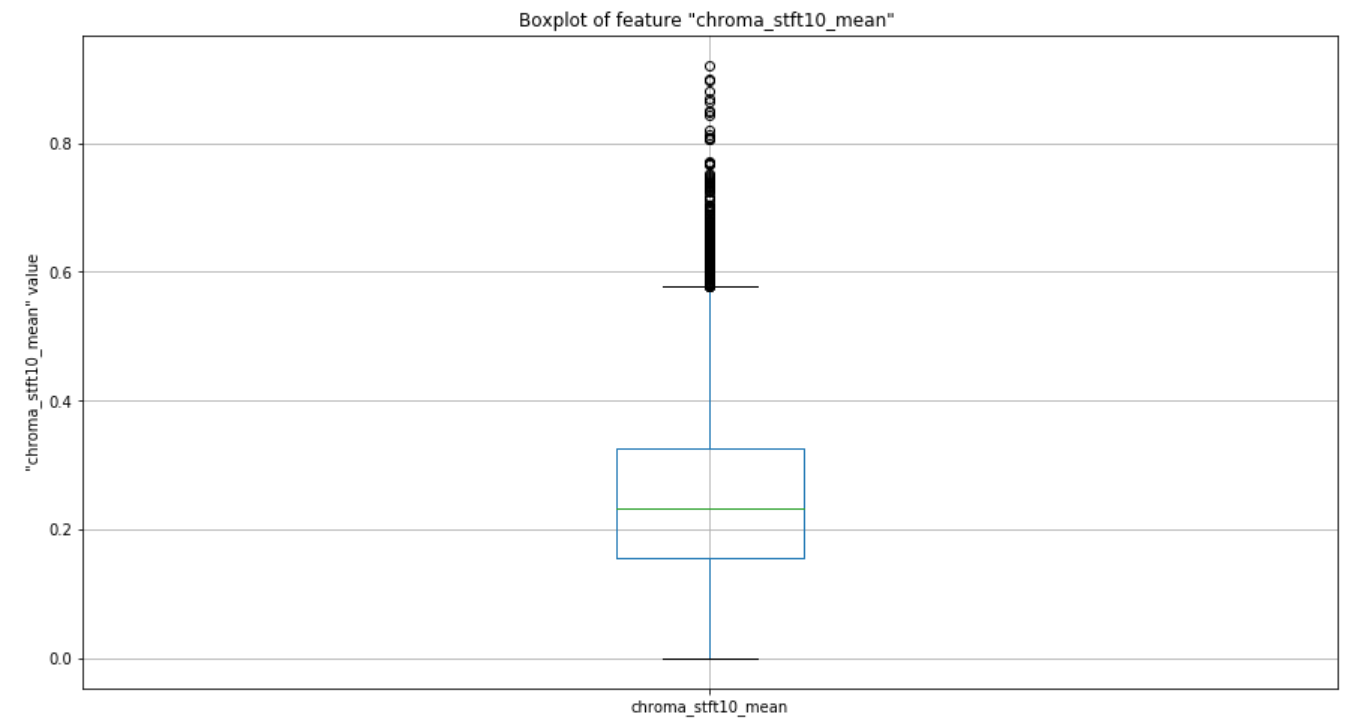
Above Boxplot conveys the presence of outliers in the fourth quartile. Though majority of the data lies between 0 and 0.22, we do see few data points lying in the range of 0.23 to 1 suggesting very minimal or fractional White-Noise in the data.

**Chroma Short-Time Fourier Transform**

Chroma feature describes the presence of tonal content in the audio data by classifying input into twelve different pitch classes. Short-Time Fourier Transformation helps to get the information about the frequency distribution of the input data by provides a list of 20 features as output.



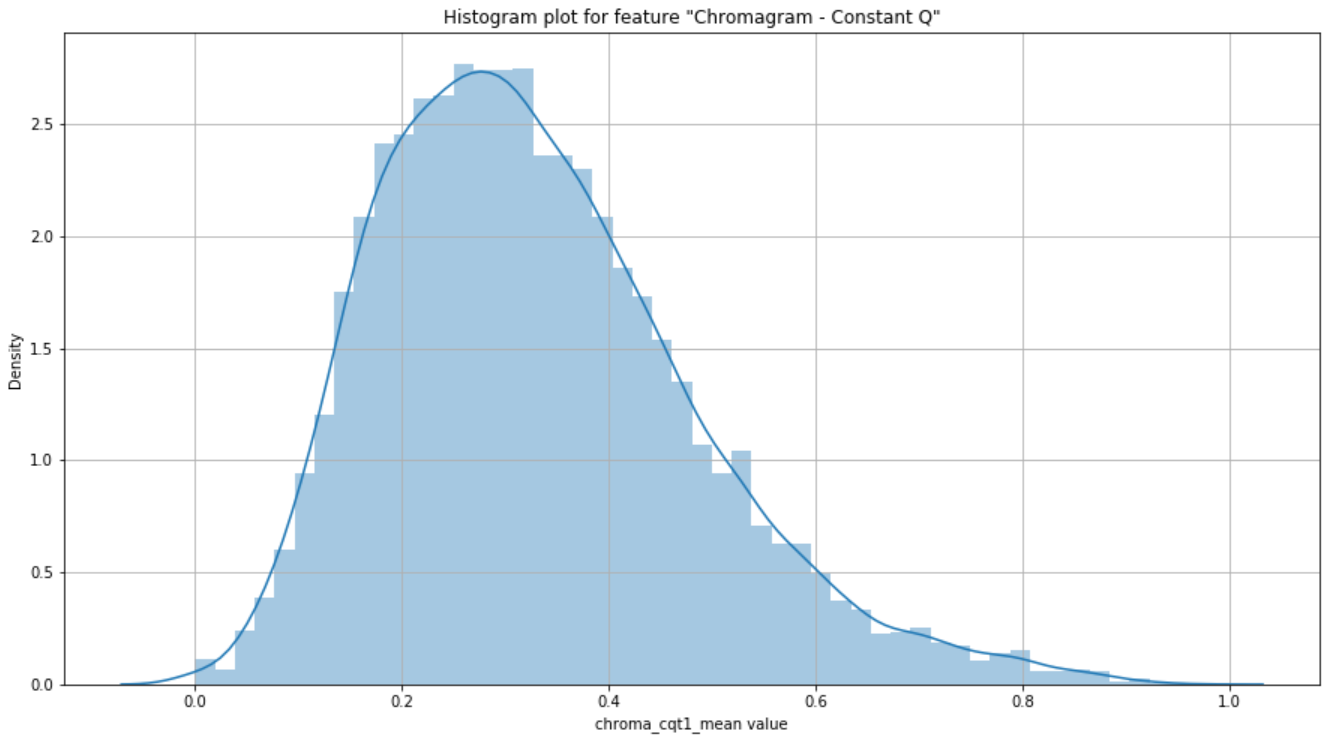
Off the 20 features that Chroma STFT provides, mean and variance values have been considered and used in our data. Majority of the data is present in the range of 0.0 to 0.5 thus suggesting the presence of positive skewness or right-skewed data.



With the help of boxplot, outlier detection can be achieved. For Chroma STFT feature, outliers are ranging around the value of 0.58 through 0.9. Median can be observed around 0.23. Amongst 13105 instances, we can observe 13077 unique values and 0 nulls.

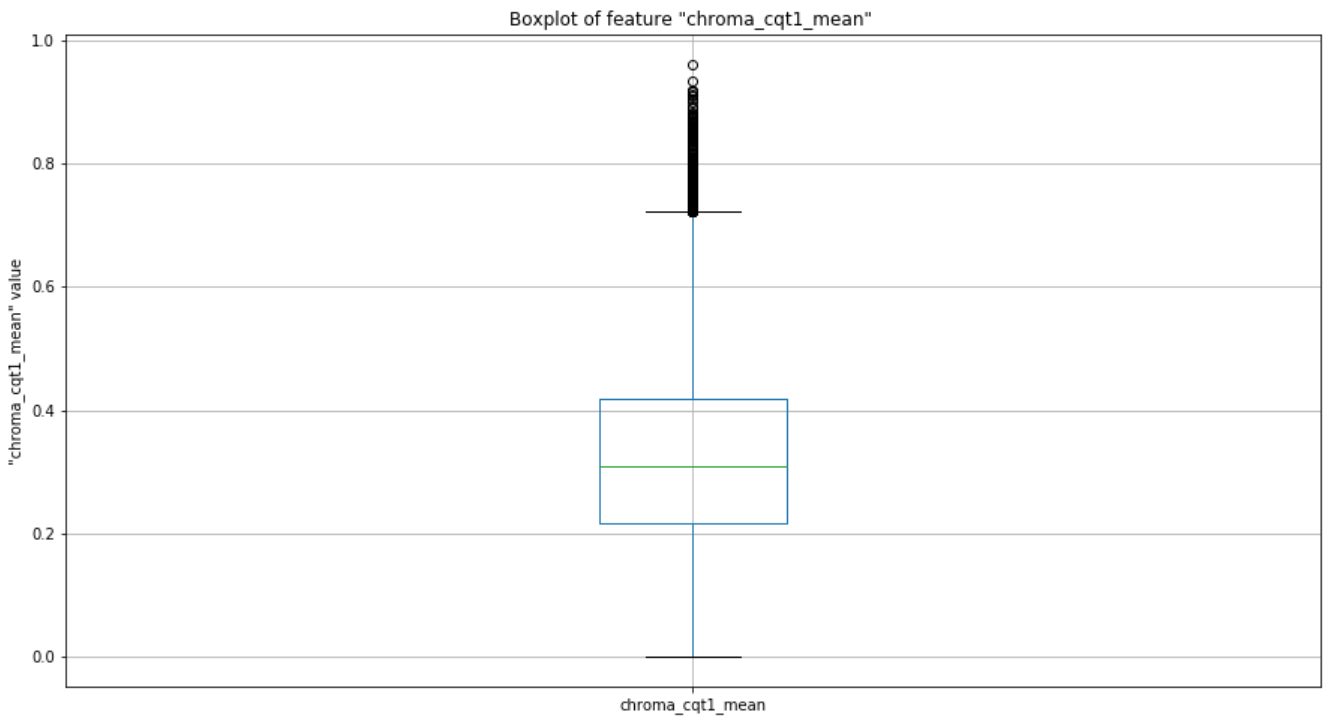
**Chroma Constant-Q Transform**

The frequency obtained by the Constant-Q Transform directly corresponds to the central frequencies of the musical notes. At both very high frequencies and very low frequencies, it is often difficult for humans to capture and process the data. Hence by allocating more processing and more time, better resolution is obtained at these frequencies. But owing to this high processing time and complex time-frequency matrix, DFT is preferred over the CQT.



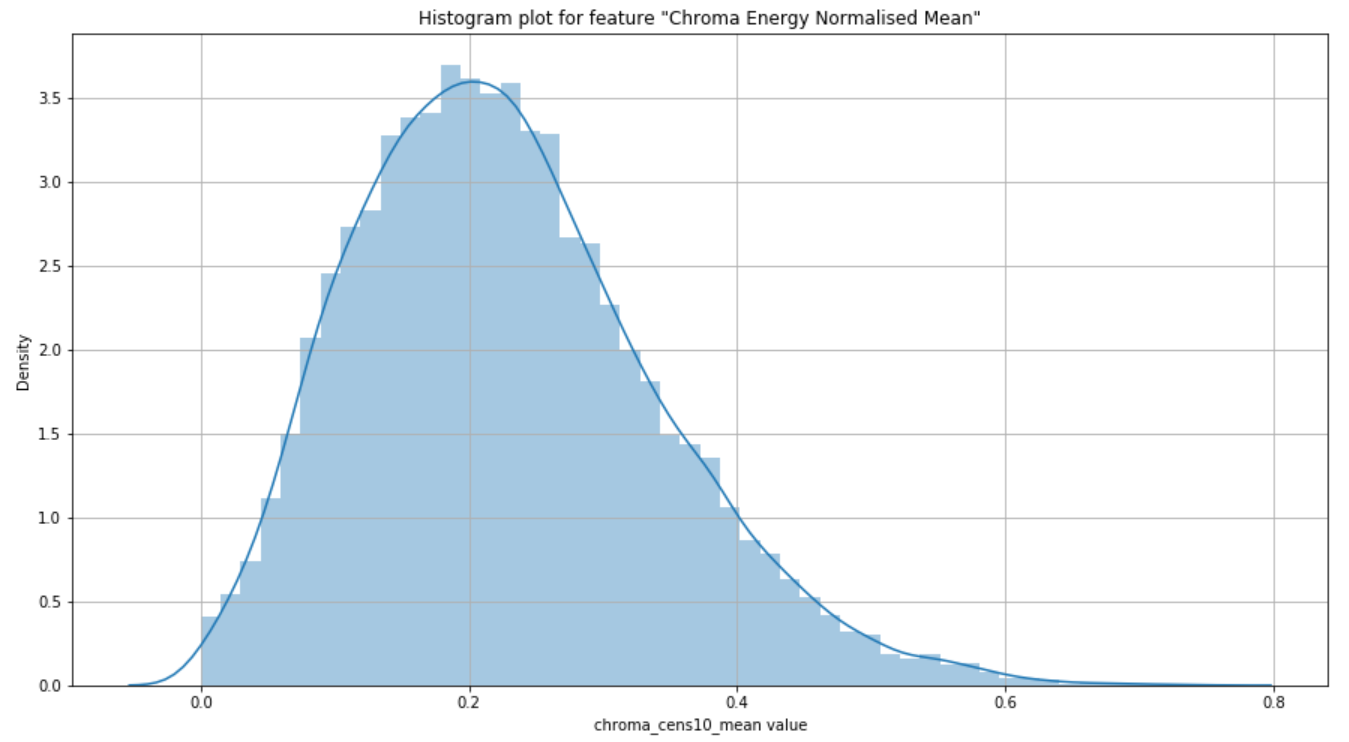
The data obtained from this feature has minimal skewness i.e. the data is slightly right skewed or positively skewed. 0.1 to 0.7 is the range that captures majority of the data under this feature.

Amongst 13105 instances, we can observe 13079 unique values and 0 nulls. Belo displayed boxplot distribution of the Chroma Constant Q Transform feature informs about the statistics such as the minimum value of 0.0 and a maximum of slightly over 0.72, Chroma CQT median lies at 0.3 with very few outliers.

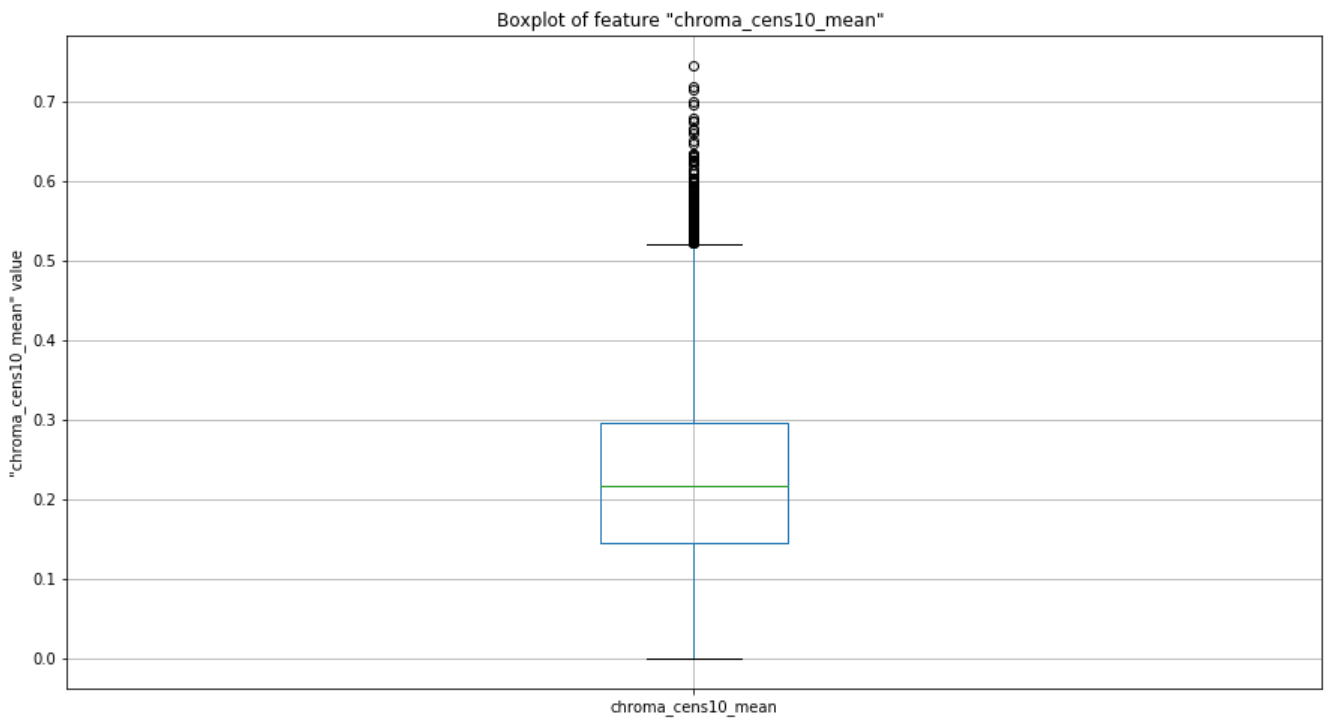


**Chroma Energy Normalized Statistics**

This is another variant of the chromagram feature and is robust to dynamics, timbre and articulation. This effectively helps in identifying and matching the audio data and also in performing the audio classification accurately.



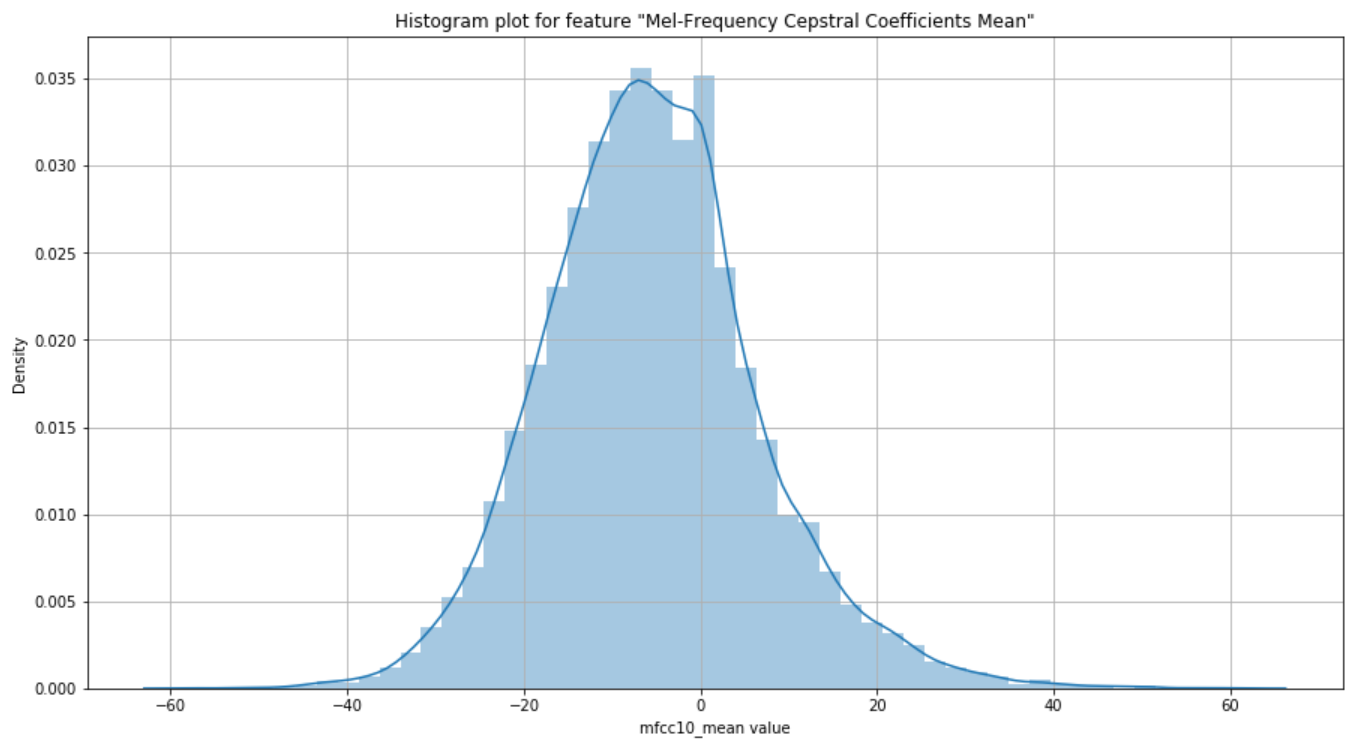
Data under this feature approximately resembles the bell-curve data with the presence of some outliers towards the end i.e. though extremely negligible, a slight positive skewness can be observed.



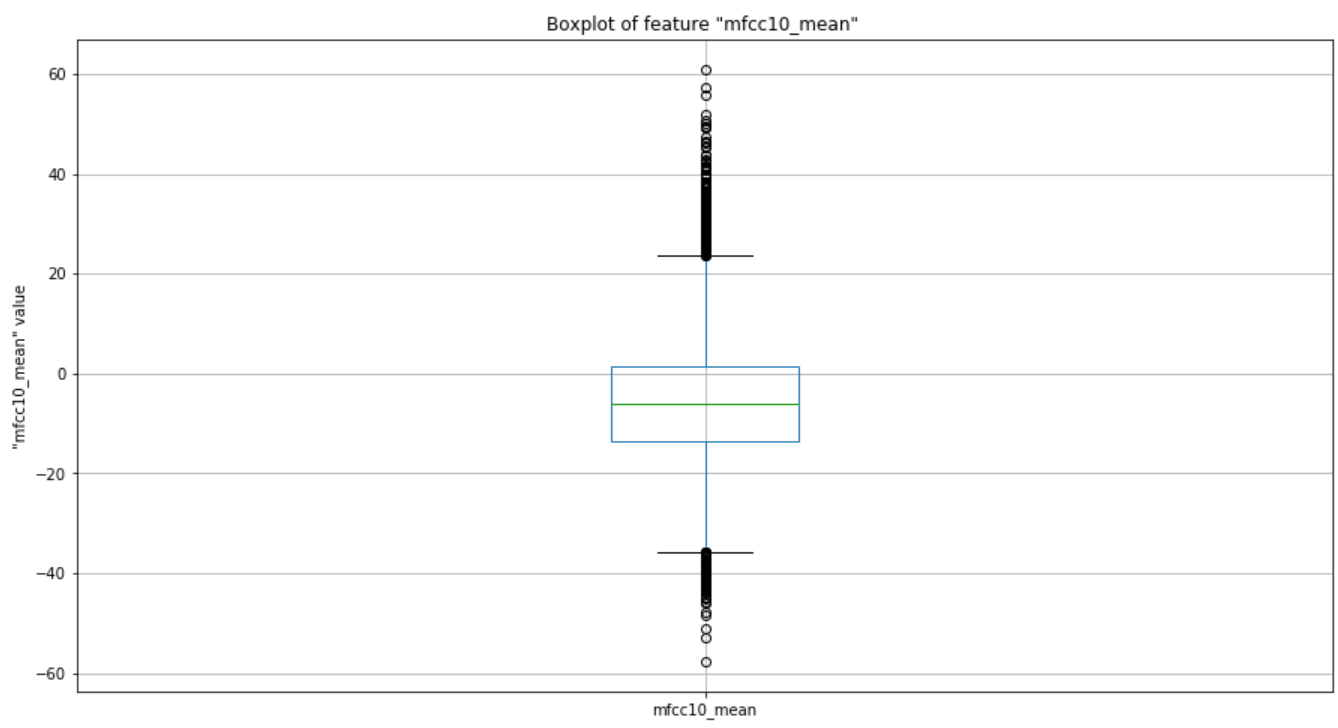
With a maximum value of 0.55, Chroma CENS for our data possess a minimum value of 0 with the median value lying around 0.52. Amongst 13105 instances, we can observe 13075 unique values and 0 nulls.

**Mel-Frequency Cepstral Coefficients**

Mel-Frequency Cepstrum is a combination of many Mel-Frequency Cepstral Coefficients (in short MFCC’s) derived from the non-linear spectrum of spectrums. This feature plays a vital role in speech recognition and also in identifying the singers from the audio input in our case.



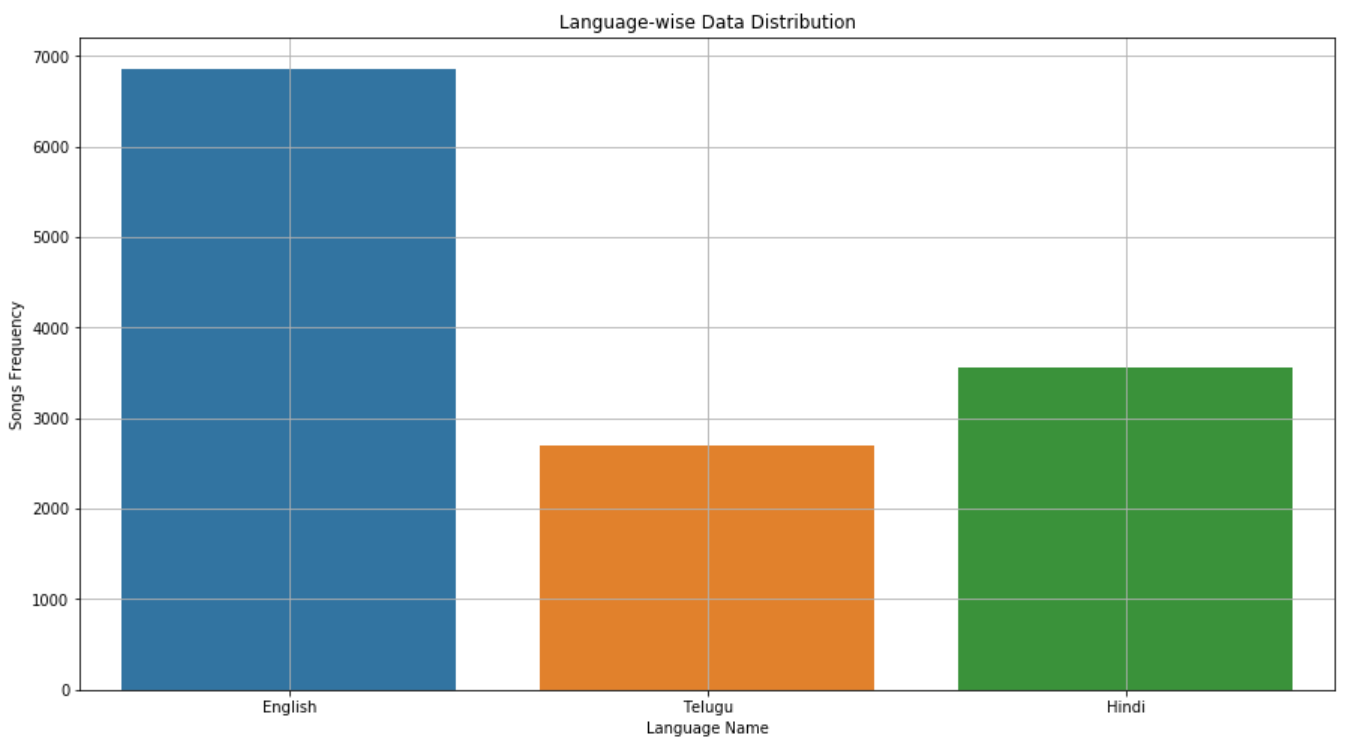
Most of the data under this feature is in the range of -30 to +30. Though the features obtained from MFCC are not robust to additive noise, it works well and suits the need in obtaining our goal as there is minimum or zero noise in our audio sample dataset.



The median is perfectly placed close to -6 suggesting the importance of this feature in speech recognition process. For our data, outliers of this feature are the data instances lying outside the range of -30 to +30 and the same information can be inferred from both histogram and boxplots. Amongst 13105 instances, we can observe 13073 unique values and 0 nulls.

**Language:**

Our dataset contains attribute ‘language’ which indicates to which language the song belongs to. This attribute would be helpful to identify the song language and shows how the songs are distributed among each language. The languages that we have chosen for the project are ‘English’, ‘Hindi’ and ‘Telugu’. Our data consists of more than ten thousand instances for English language, near to six thousand instances for Telugu language and more than six thousand instances for Hindi language. Below plot is the visualization for Artists Data spread across the selected three languages.



**Target:**

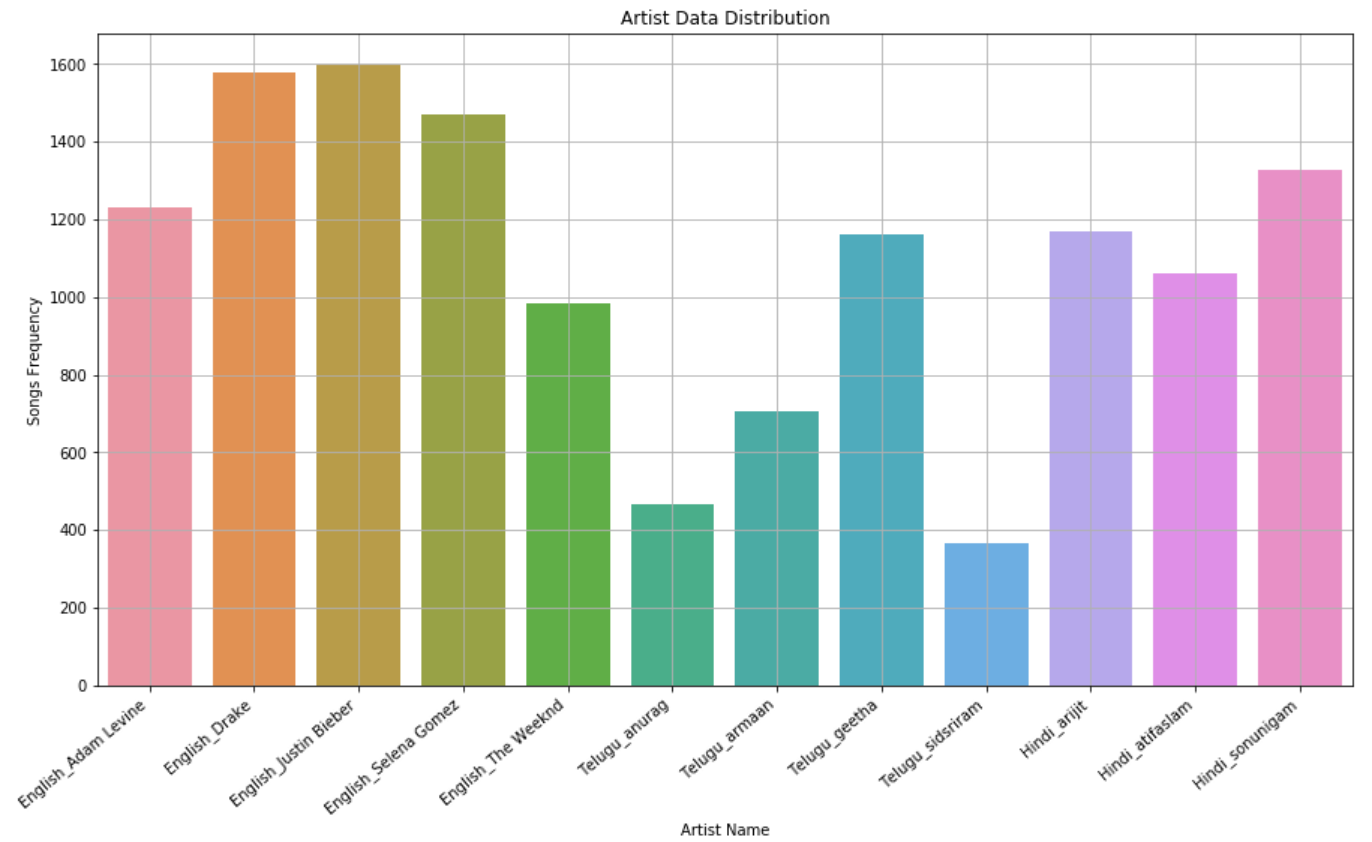
Our target for this project cosists of Genre classification and Artist voice detection. Though we have combined both the artists and Genre data into one dataframe to perform the data analysis, we have also analysed our data by focusing separately on artists and genres without clubbing them. Our target consists of 12 classes for artists data and 10 classes for genre data. The taget is distributed as follows:

* 5 English artists data
* 4 Telugu artists data
* 3 Hindi artists data
* 4 Engilish genre data
* 3 Telugu genre data
* 3 Hindi genre data

Majority of the data is concentrated for English language. We observed that there is data imbalance for the target varuable. When performing modelling if the model is overfitted or underfitted we would separate each language datasets and perform modelling individually. If the model is still overfitted or underfitted we would go ahead and perform oversampling or downsampling techniques where required.

Below plot visualizes the data distribution of songs amongst the 12 identified artists. It is to note that this visualization only depicts the artists data without including any genre specific data.

Artist ‘Justin Bieber’ has most number of songs followed by ‘Drake’ while ‘SidSriRam’ has the lowest number of songs followed by ‘Anurag’ in our data.

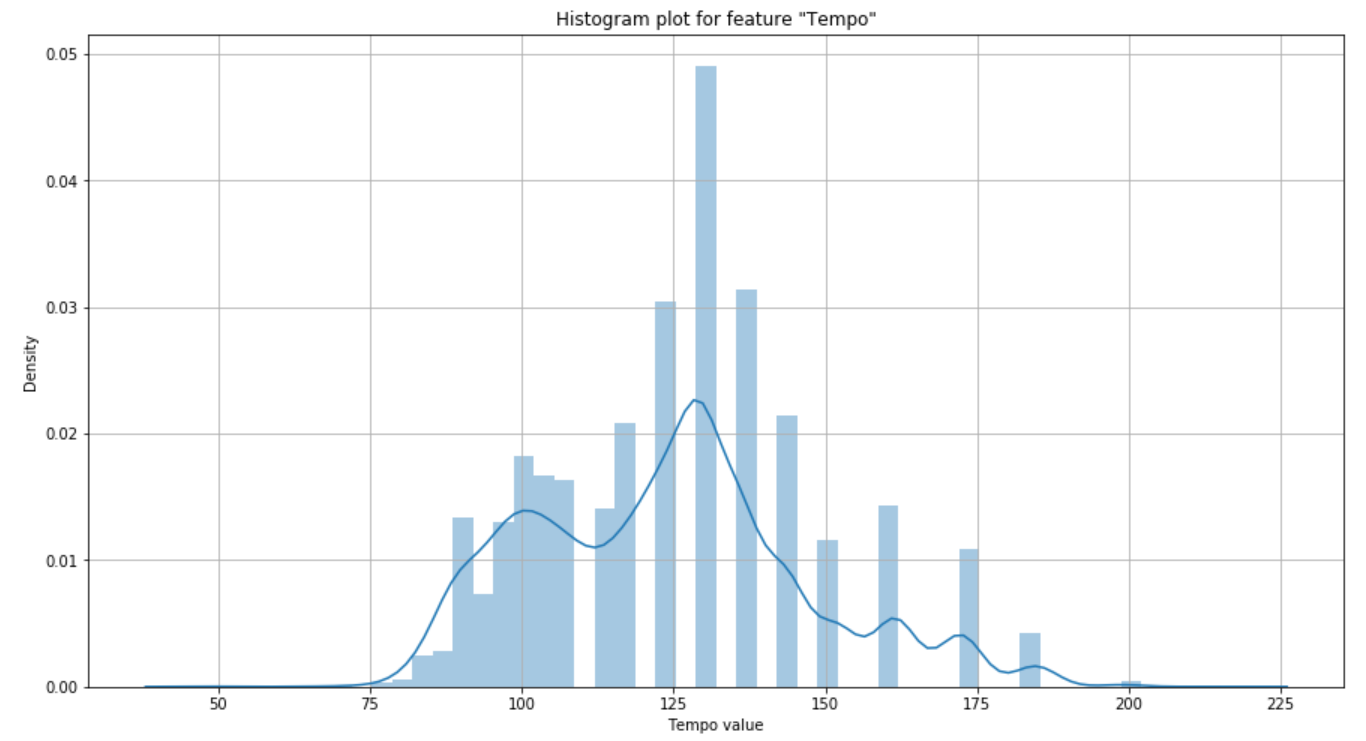


Below are the features obtained from the Genre Data.

**Tempo**

This feature is extracted from the audio data, this speaks about the speed of the music present inside the audio. It is the rate of musical beat present in the audio, this is calculated using the reciprocal of the beat period. It is defined in the units of beats per minute (BPM).

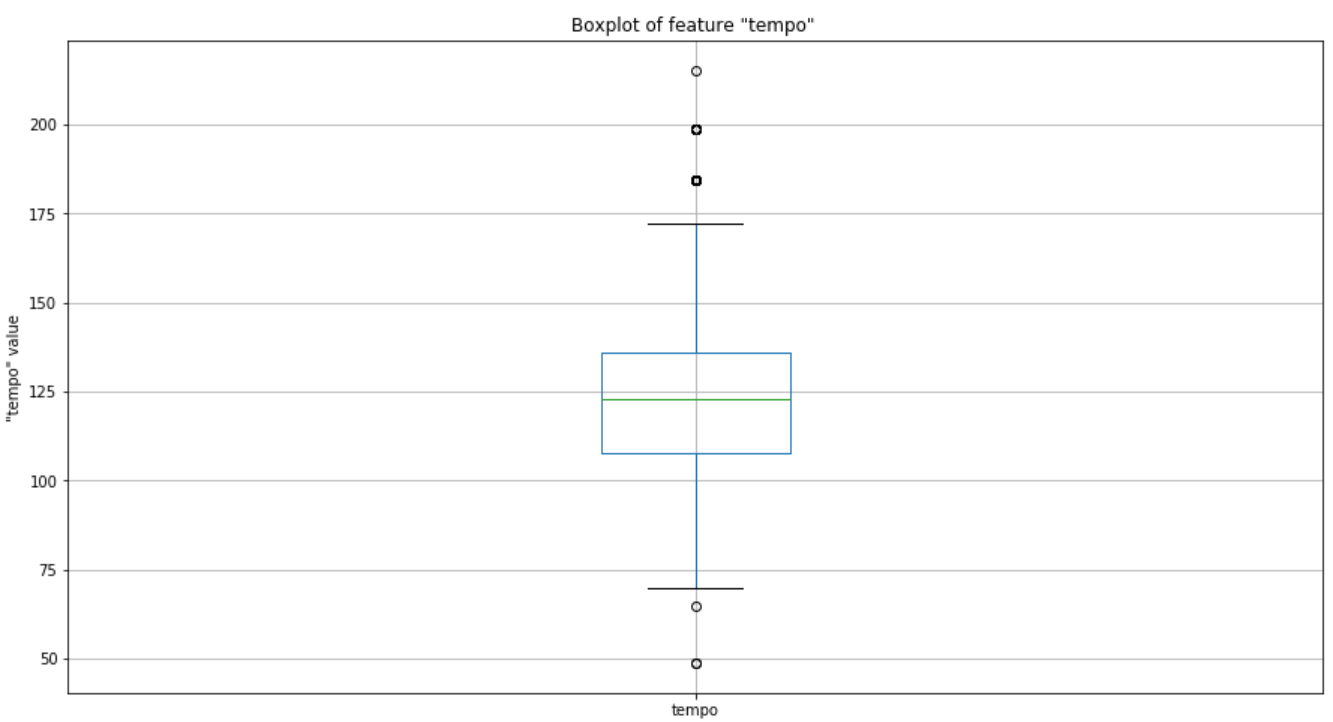
In our dataset, we find 27 unique values for tempo, and no null values are present.



On observing the histogram, we can understand that the data distribution is not a normal distribution, as we can observe some irregularities in the curve obtained by the histogram. Overall, there are couple of peaks in the data and the highest point or the maximum frequency at approximately with tempo of 132 density closed to 0.05.

Boxplot distribution:

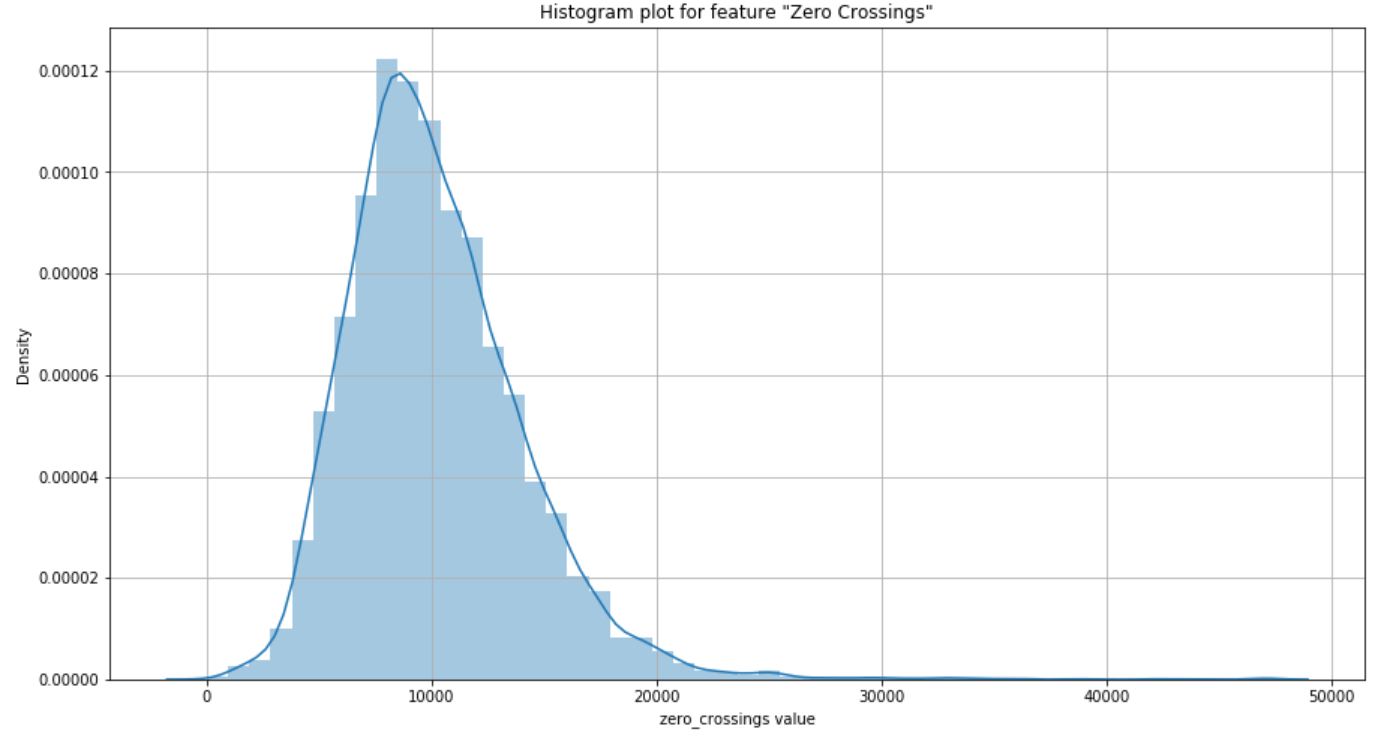
Below is the distribution of tempo variable using boxplot, on observing we can find that the data is more onto the downside so we can observe the less distance between the 1st quartile and minimum value. Most data i.e., from 1st quartile to 3rd quartile is lying in between tempo values of 25 to 50. We can also consider the values from the 3rd quartile to maximum as outliers. The average value or 50th percentile is located at value 125 which is almost similar to that observed from the histogram.



**Zero Crossings**

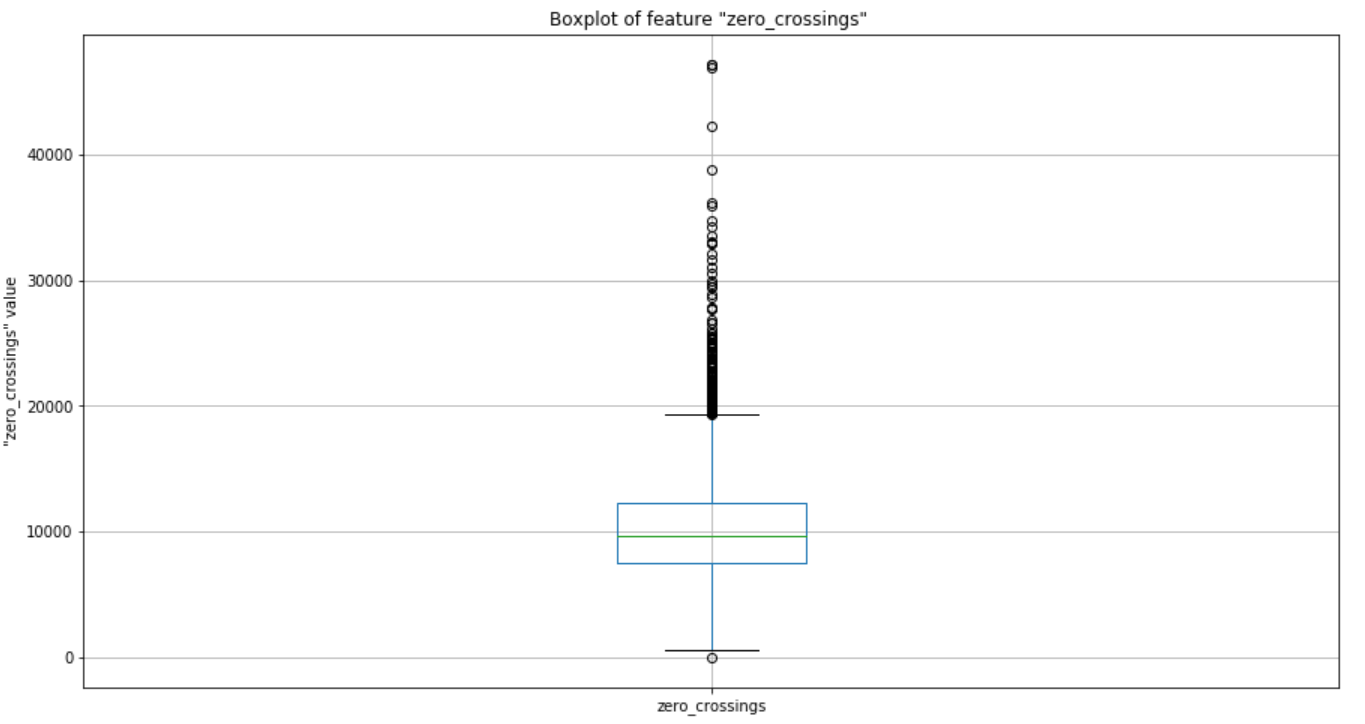
It is defined as the number of times the signal in audio is crossing the zero line, which indirectly speaks about the shift of values from positive to negative or vice versa.

The unique values for this attribute are 6744 and no nulls are present.



On observing the above histogram, we can infer that the histogram is right skewed, Most of the songs do have the zero crossing rate of approximately 8000, the width of the curve is also very low which also conveys that 90% of the songs are in between 5000 and 15000 zero crossing rate value.

Box plot distribution:



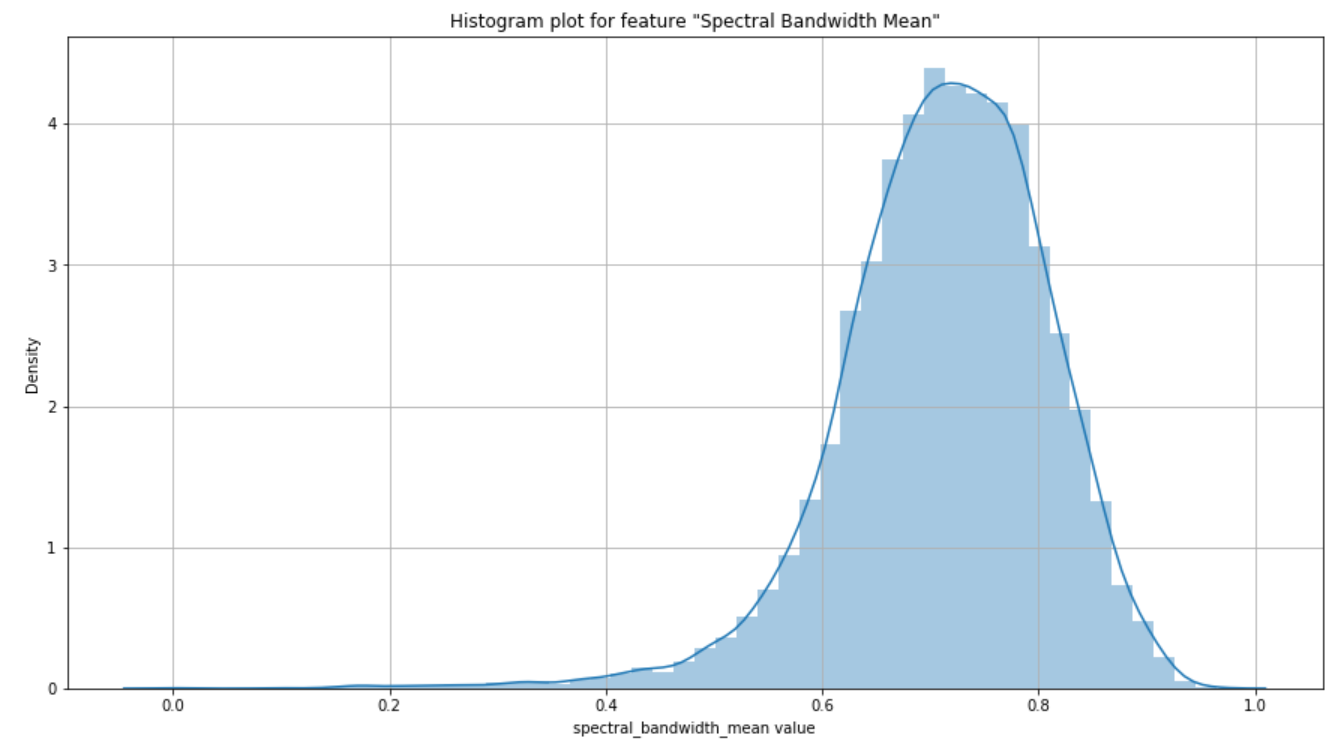
On observation we can infer that 50% of the data is present in the lower half of the plot. After 3rd quartile, we can observe the datapoints, but they are having a huge range from 20000 to 50000. The average value is 9000 which is similar to the value from the most frequent value obtained by histogram. Minimum is around zero and maximum is at 50000.

**Spectral bandwidth mean**

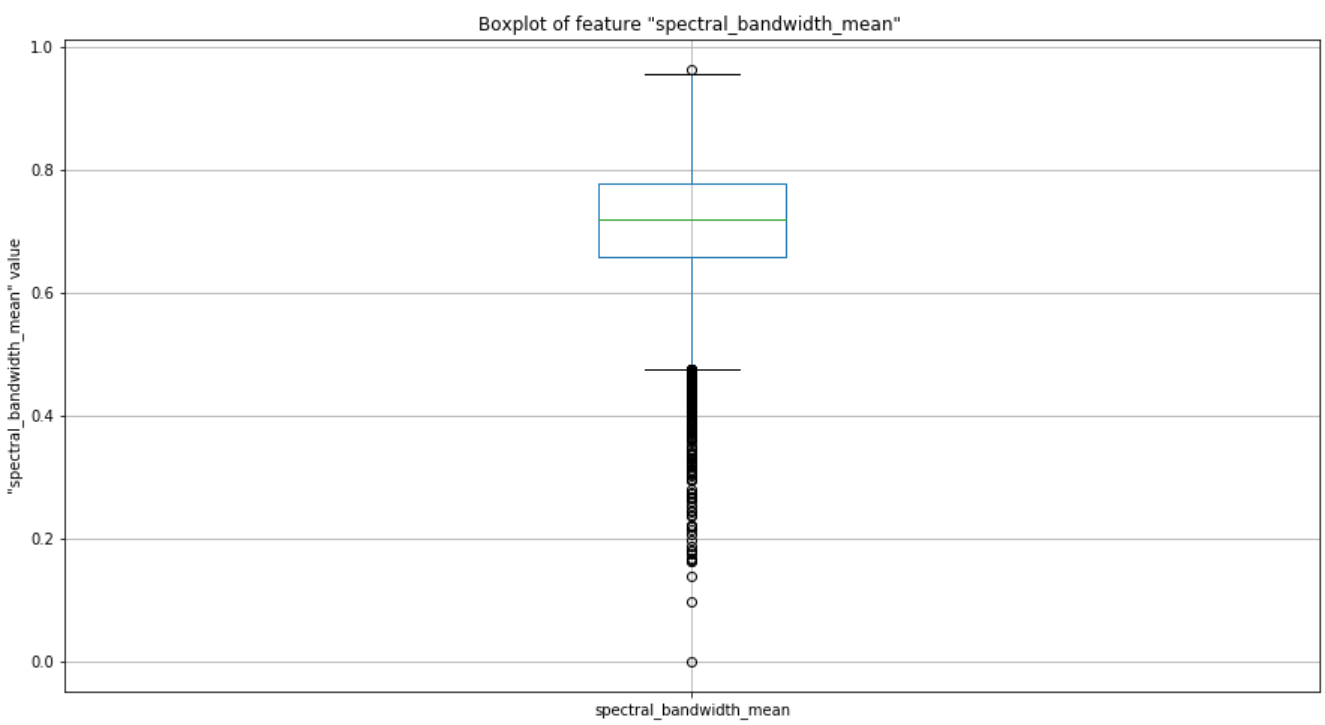
It is the range of values in which the signal is not less than the half of its maximum value. This feature is also important in analyzing the signal. The values are considered into our dataset after normalization.

The unique values for this attribute are 9523 with no null values present in our dataset.

On observing the below displayed histogram, we can observe that the graph is a left skewed curved distribution with a long tail starting from zero till 0.4. The highest occurrences are at 0.7 value with a density of approximately 4.5. The graph is denser at this area.



Boxplot distribution:

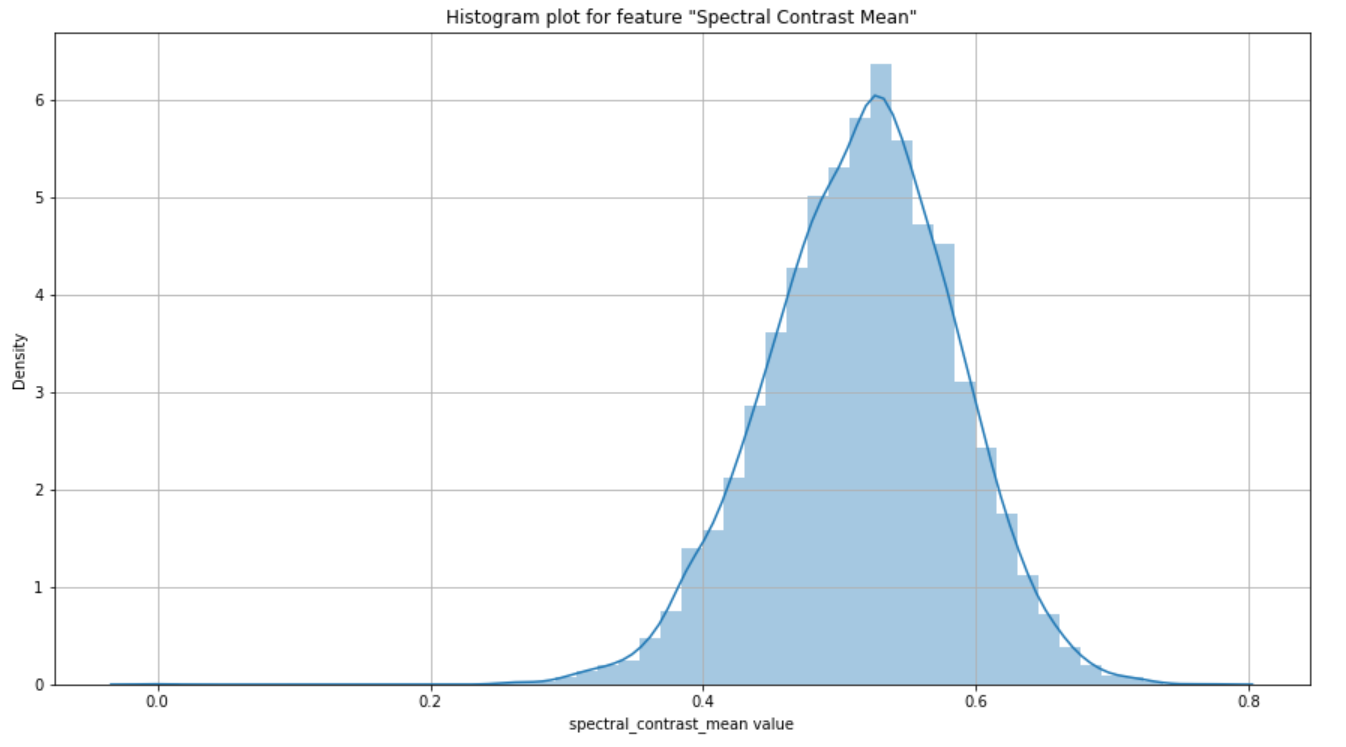


On observing the distribution, we can confirm this is a left skewed as we can see sparsely distributed values below the 1st quartile. The width for the 50% of the data is quite more in comparison with others with range from 0.48 to 0.97. The mean is located at 0.72 value and maximum at 0.97. We can observe that the maximum and 3rd quartile are closer than the distance between 1st quartile and the minimum value.

**Spectral Contrast mean**

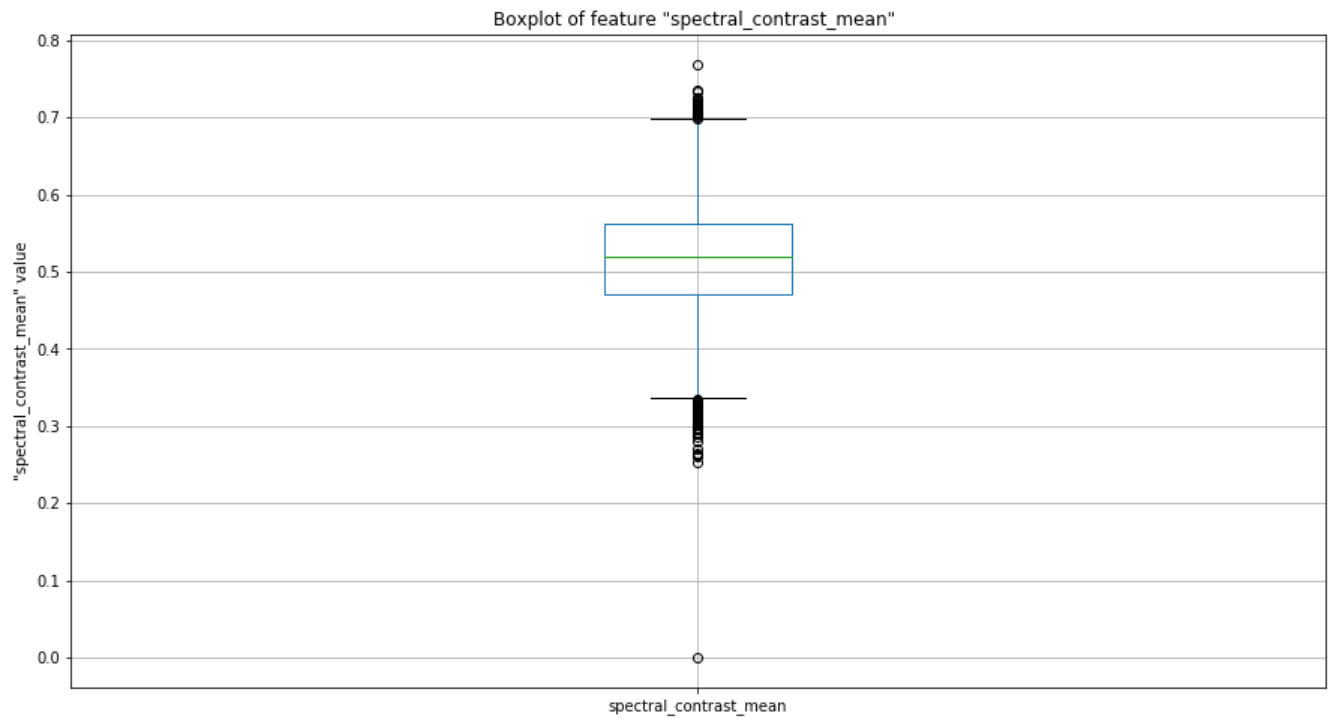
Spectral contrast is defined as the difference of peaks and valleys in the spectrum. It considers the spectral peak, valley and their difference in each sub-band.

The unique values present for this attribute in our dataset are 9523 and no nulls are present for this attribute.



On observing the distribution of histogram, we can state that it is following a clear normal distribution, this looks so close to the actual normal distributed curve. Mean is at the 0.5 with almost the density of 6.5.

Box plot distribution:

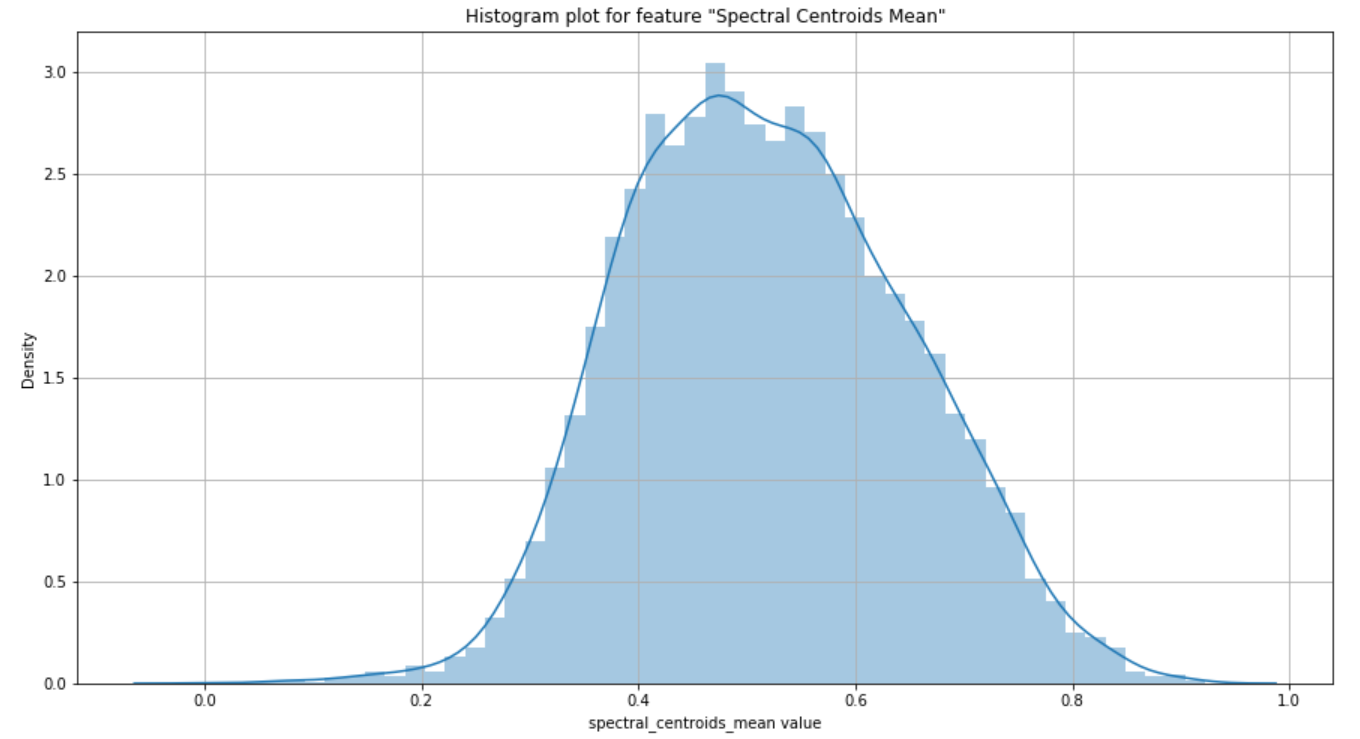


On observing the boxplot above, we can state that the mean is located at exactly in the center at 0.52. with equal distribution on both the sides, still there are less instances on top when compare to bottom but the difference seems very less and can be negligible. The 1st quartile and 3rd quartiles are at 0.48 and 0.57 points respectively.

**Spectral centroids mean**

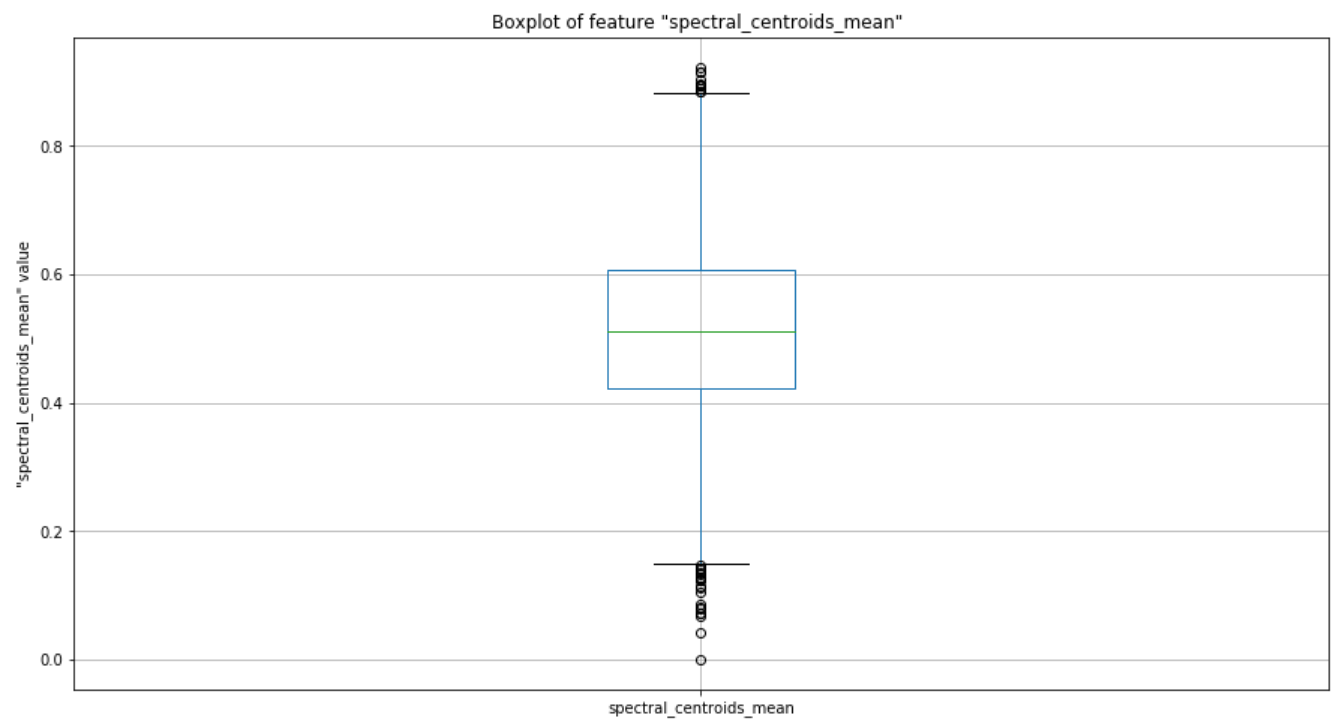
It is a metric in digital signal processing, where spectral centroids are defined as the points where the center of mass for the spectrum is located. This is an important feature for our data as we may depend on the center of mass for each song which differentiates them accordingly.

There are 9523 unique values and no nulls are present for this attribute in the dataset.



On observing the histogram, we can infer that it is almost very near to the normal distribution with slightly more data onto the left and tailed onto the right. Overall, we can state this as the normal distribution as the small tail over the right can be negligible. The mean is at approximately 0.5 with a density of nearly 3.05.

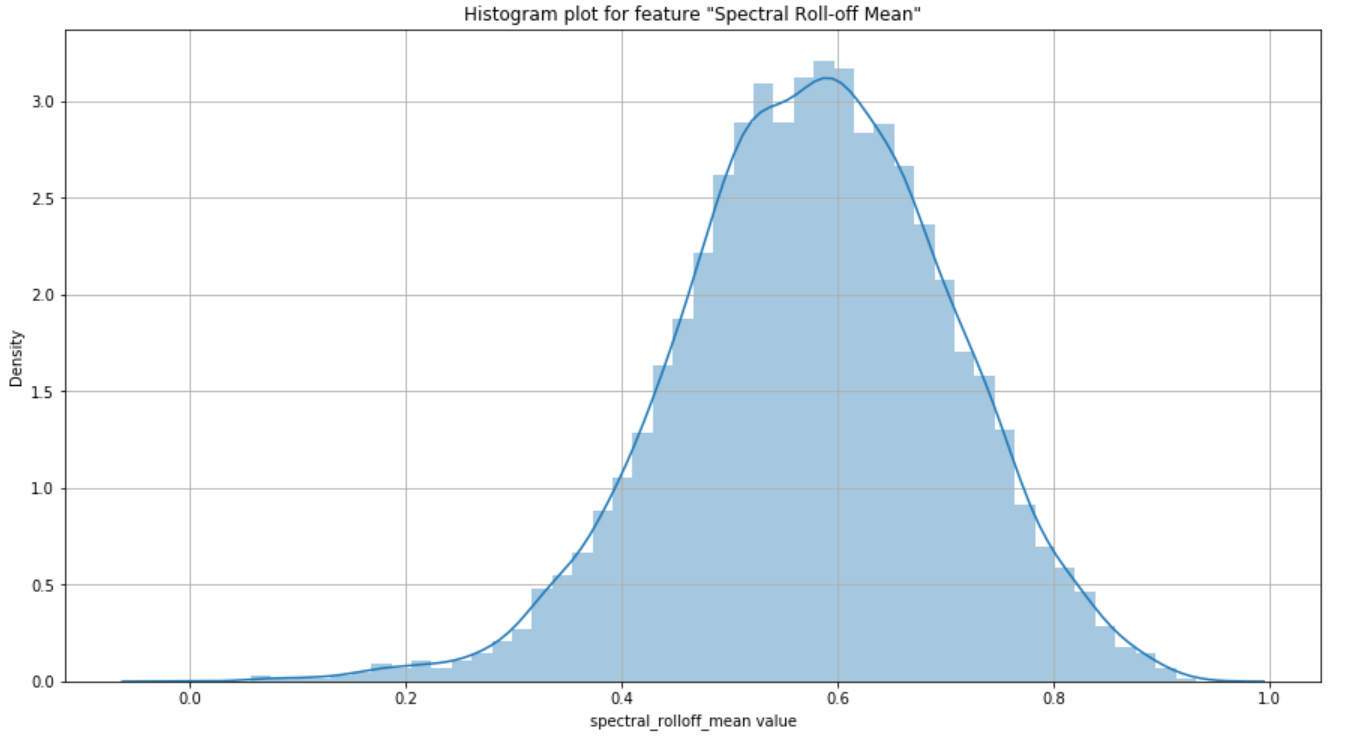
On observing the below specified boxplot distribution, we can again iterate that the distribution is almost near to the normal distribution. The difference between the 1st quartile to 3rd quartile which is the 50% of the data is also uniformly distributed on to both the sides. The 1st quartile is located at approximately 0.42 and 3rd quartile is located at 0.61 respectively.



**Spectral rolloff mean**

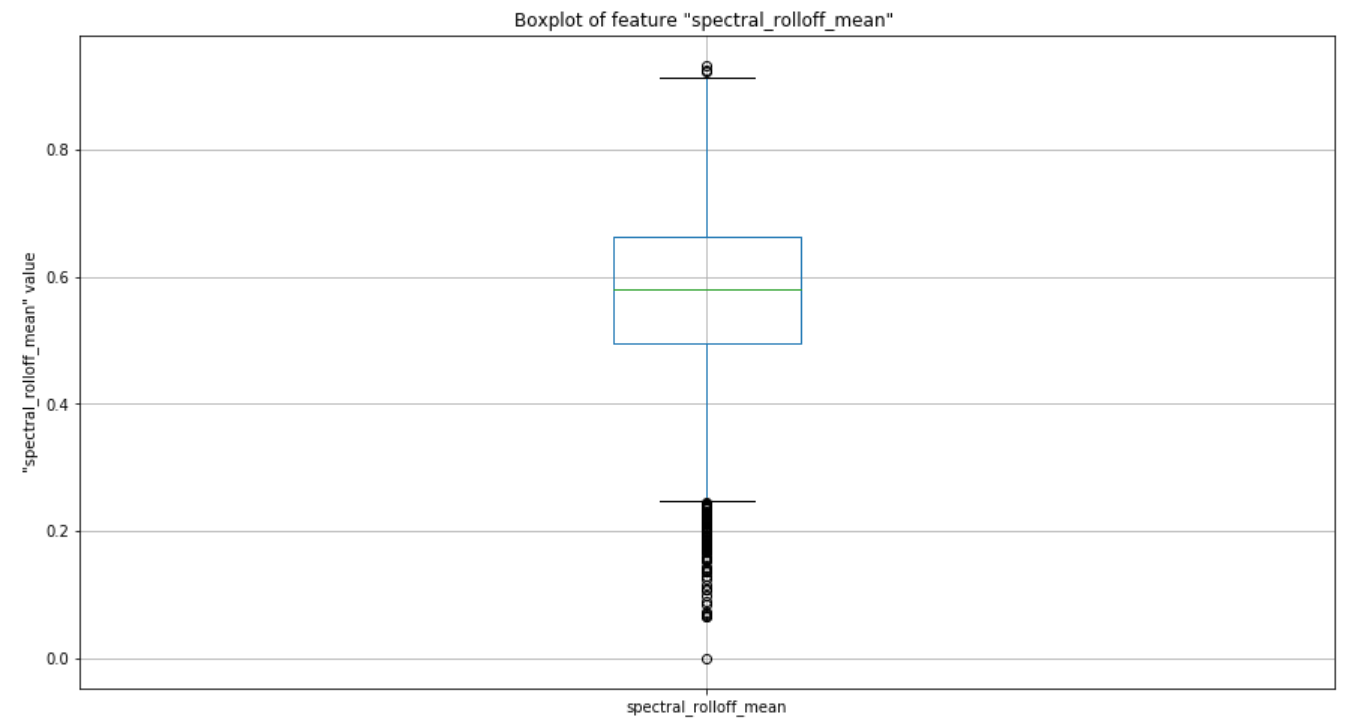
Spectral rolloff is the range where in 85% of the spectrum energy falls into. This gives us the value where all the energy is located which helps us in differentiating the songs and thus helps in our modelling.

There are 9520 unique values and no nulls are present for this attribute in the dataset.



The above histogram tells us that the data distribution is normal and uniform all over the curve and is very close to normal. There is a slight tail to the left by which it might tend to be a slightly left skewed, but on observing the dataset we can neglect that slight difference. The highest occurred value is at 0.6 with a density of 3.2 approximately.

Box plot distribution



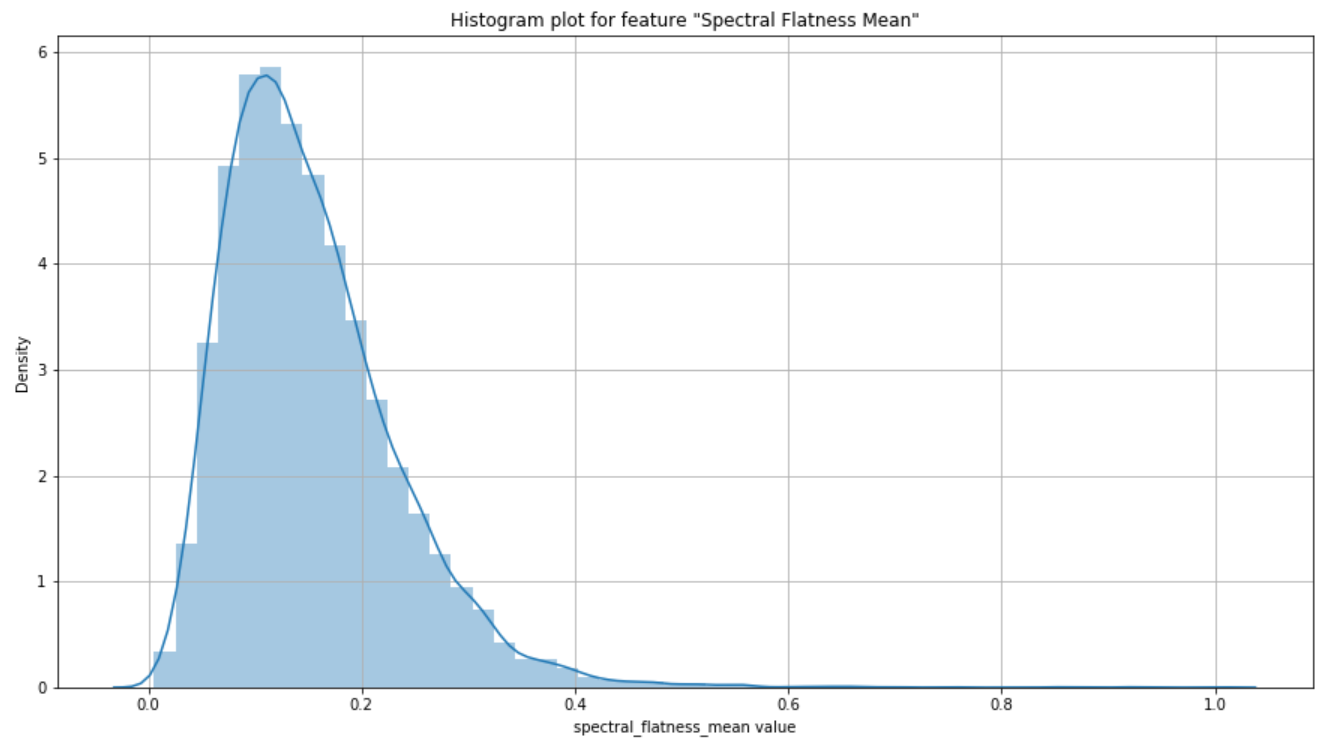
On observing the above boxplot distribution, we can infer that the data is slightly left skewed in distribution as we can observe there are few data points below the 1st quartile, and very few above the 3rd quartile. The data between the 1st and 3rd quartiles is very large, and the mean is located at the 0.58 approximately.

The Q1 and Q2 ends are at 0.23 and 0.67 respectively.

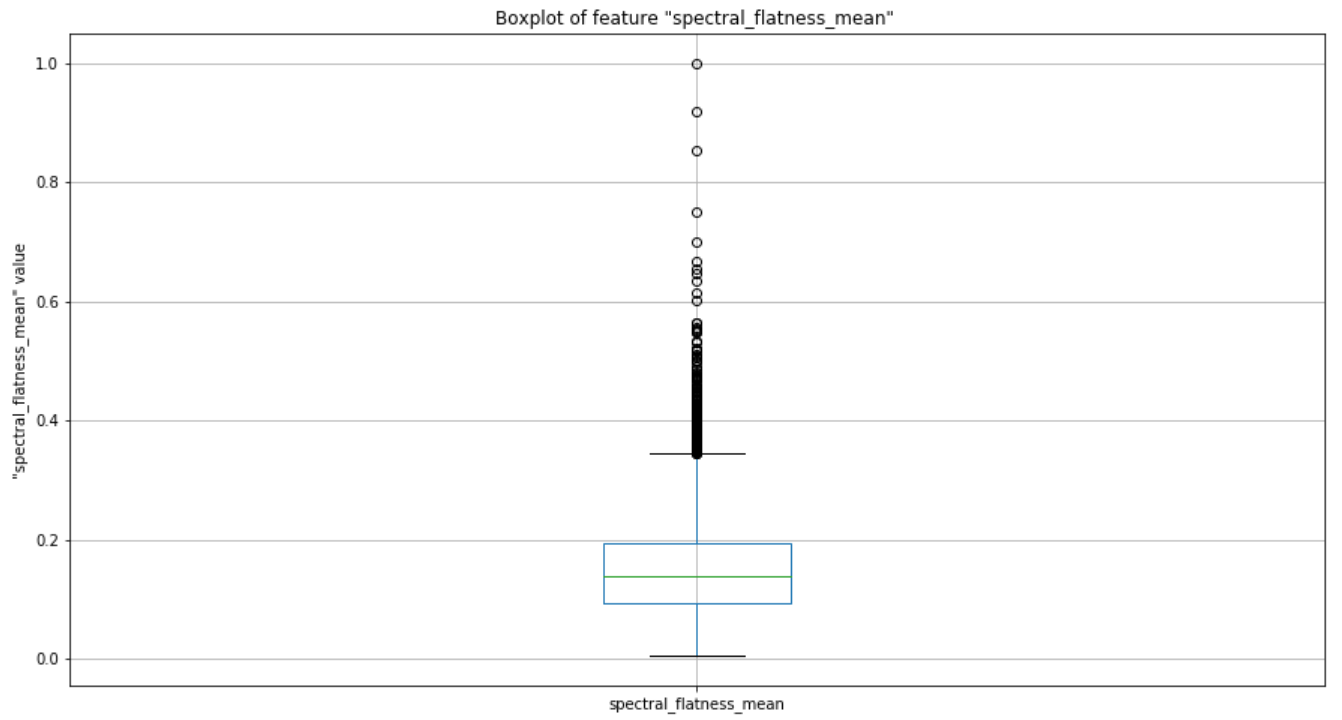
**Spectral Flatness**

In order to capture the presence of noise within the data, ‘Spectral Flatness’ measure can be used. It measures the amount of noise that is present in the input audio rather than the presence of toned data.

This value takes the range of 0 to 1. Higher ‘Spectral Flatness’ value (1) indicates the data being similar to White-Noise whereas a lower value (zero) indicates the absence of White-Noise.



From the above Histogram, we can depict that there is very little White-Noise in the data as the majority of the data is observed to be farther away from 1 and much closer to 0. Amongst 9636 instances, we can observe 9519 unique values and 0 nulls.



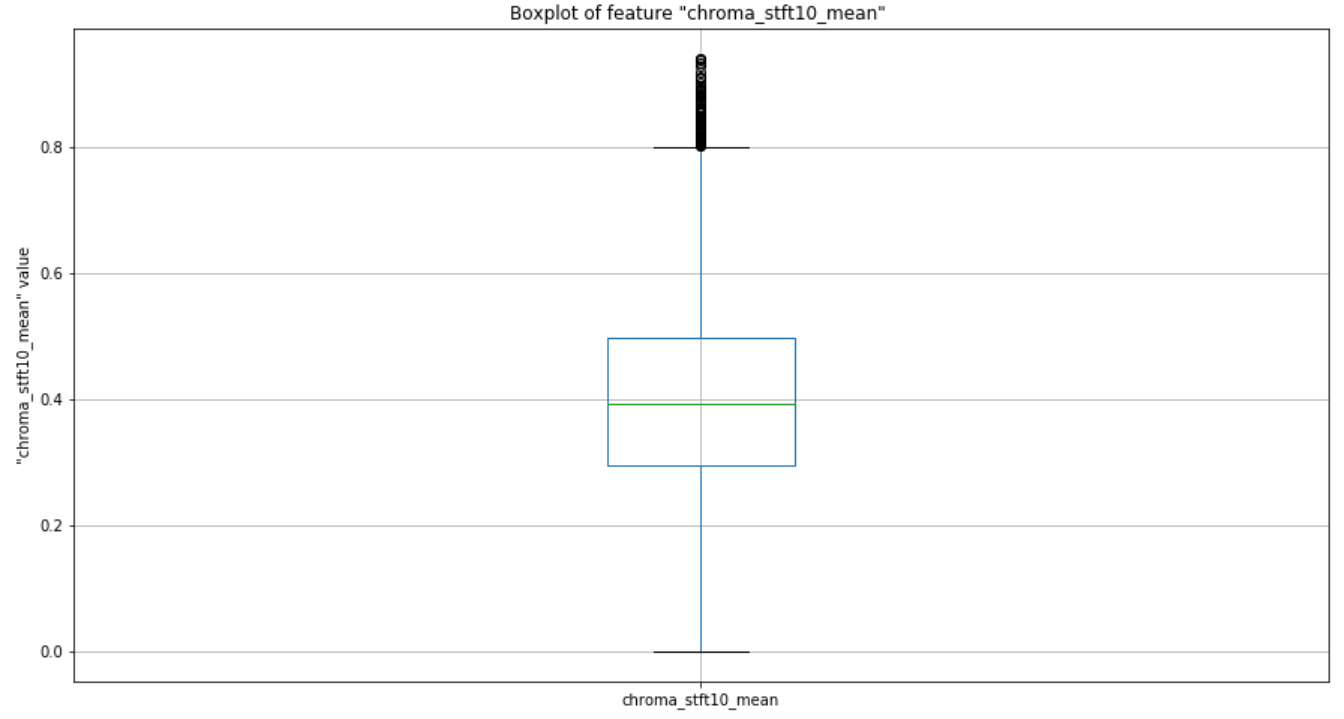
Above Boxplot conveys the presence of outliers in the fourth quartile. Though majority of the data lies between 0 and 0.37, we do see few data points lying in the range of 0.38 to 1 suggesting very minimal or fractional White-Noise in the data.

**Chroma Short-Time Fourier Transform**

Chroma feature describes the presence of tonal content in the audio data by classifying input into twelve different pitch classes. Short-Time Fourier Transformation helps to get the information about the frequency distribution of the input data by provides a list of 20 features as output.



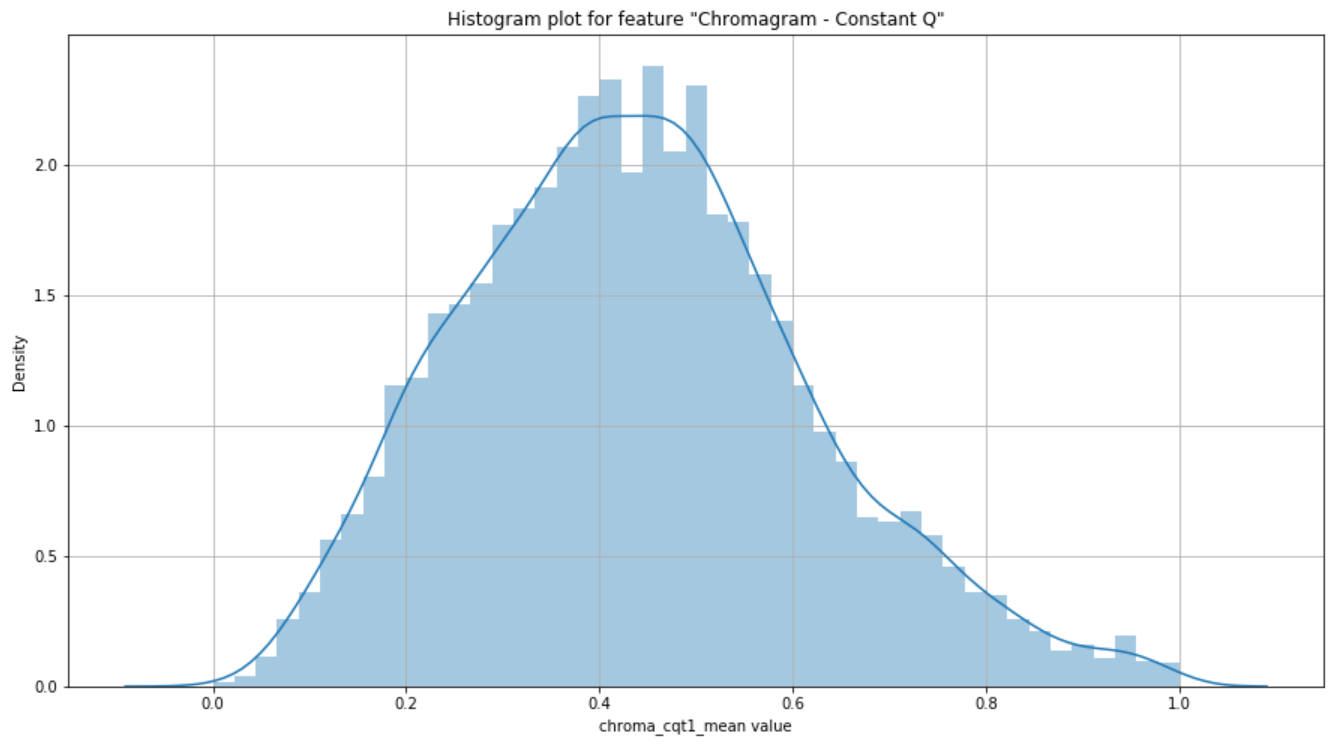
Off the 20 features that Chroma STFT provides, mean and variance values have been considered and used in our data. Majority of the data is present in the range of 0.1 to 0.8. The data distribution resembles a normal distribution with mean of 0.4 and maximum density of 3. Though there is a minute tail extending towards the right, it is negligible as the majority of the data can be seen within the bell-shaped curve.



With the help of boxplot, outlier detection can be achieved. For Chroma STFT feature, outliers are ranging around the value of 0.8 through 0.97. Median can be observed around 0.4. Amongst 9636 instances, we can observe 9519 unique values and 0 nulls.

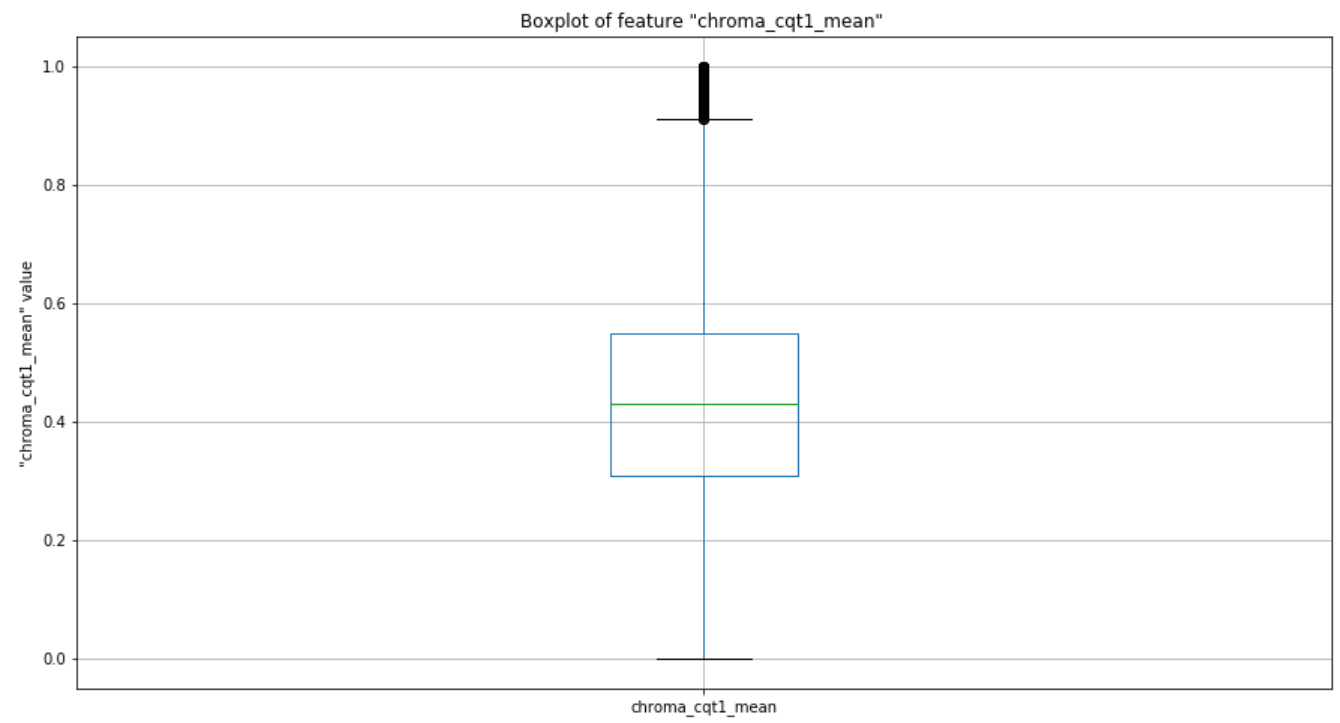
**Chroma Constant-Q Transform**

The frequency obtained by the Constant-Q Transform directly corresponds to the central frequencies of the musical notes. At both very high frequencies and very low frequencies, it is often difficult for humans to capture and process the data. Hence by allocating more processing and more time, better resolution is obtained at these frequencies. But owing to this high processing time and complex time-frequency matrix, DFT is preferred over the CQT.



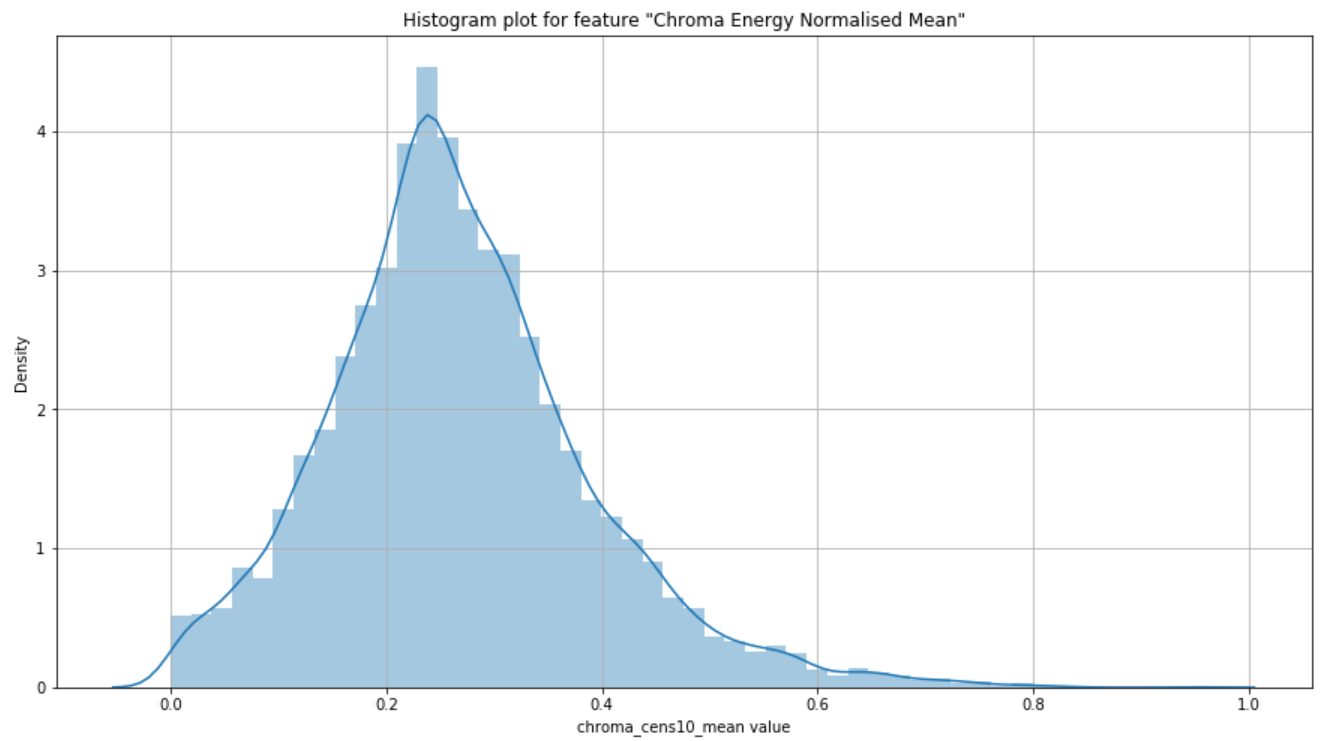
The data obtained from this feature has minimal skewness i.e. the data is slightly right skewed or positively skewed. 0.1 to 0.8 is the range that captures majority of the data under this feature.

Amongst 9636 instances, we can observe 9516 unique values and 0 nulls. Belo displayed boxplot distribution of the Chroma Constant Q Transform feature informs about the statistics such as the minimum value of 0.0 and a maximum of slightly over 0.9, Chroma CQT median lies at 0.42 with very few outliers.

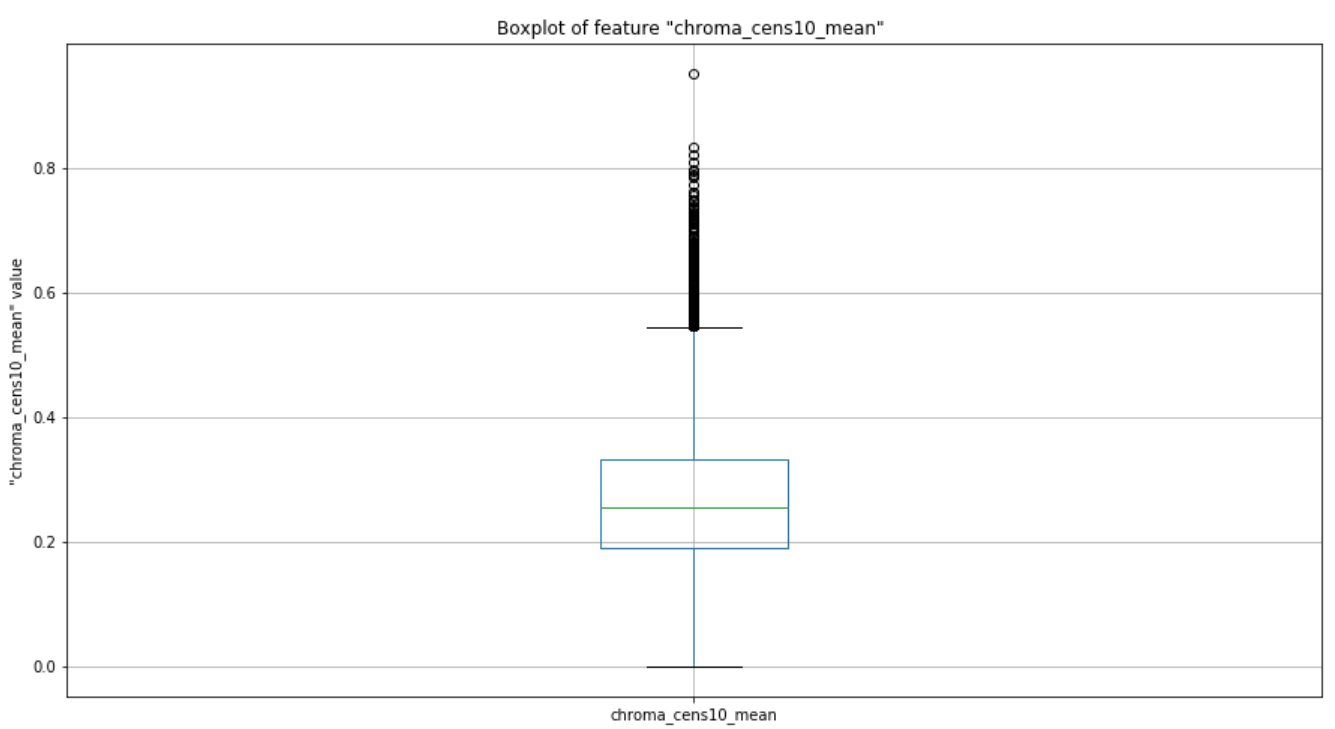


**Chroma Energy Normalized Statistics**

This is another variant of the chromagram feature and is robust to dynamics, timbre and articulation. This effectively helps in identifying and matching the audio data and also in performing the audio classification accurately.



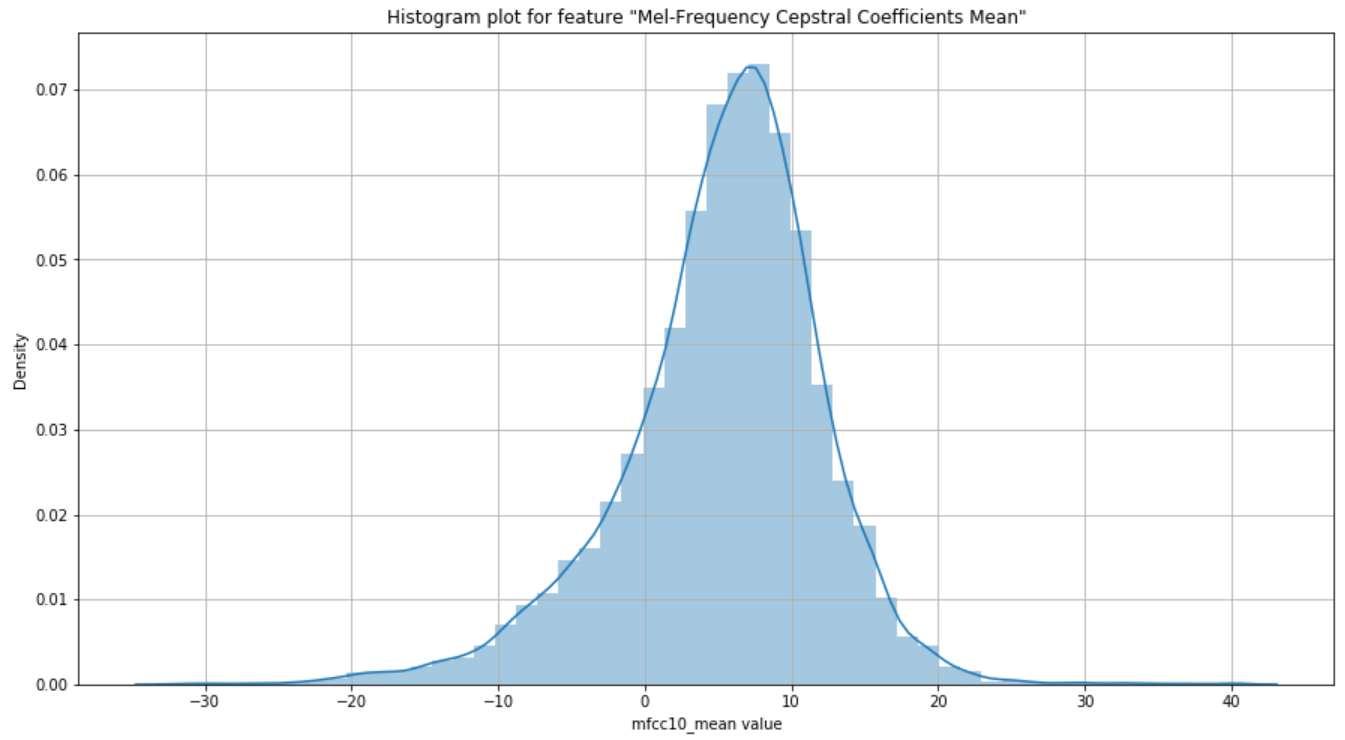
Data under this feature approximately resembles the bell-curve data with the presence of some outliers towards the end i.e. though extremely negligible, a slight positive skewness can be observed.



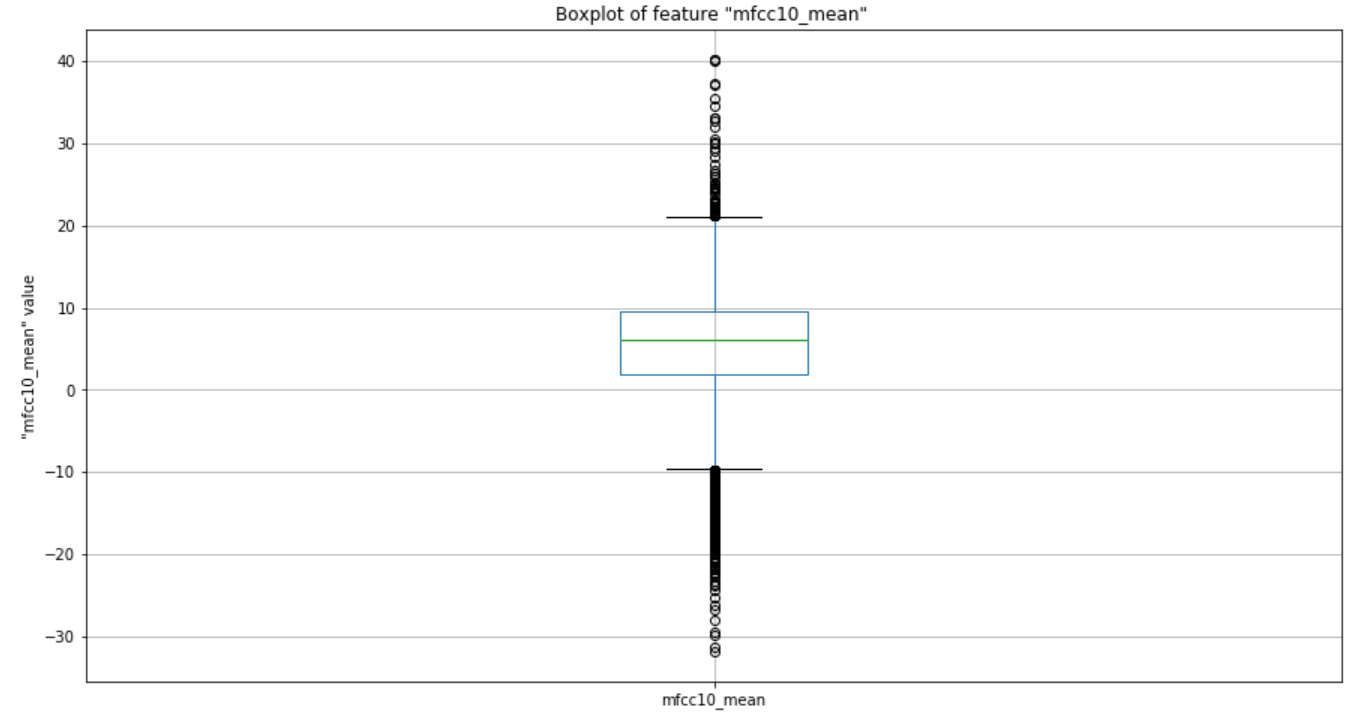
With a maximum value of 0.56, Chroma CENS for our data possess a minimum value of 0 with the median value lying around 0.25. Amongst 9636 instances, we can observe 9511 unique values and 0 nulls.

**Mel-Frequency Cepstral Coefficients**

Mel-Frequency Cepstrum is a combination of many Mel-Frequency Cepstral Coefficients (in short MFCC’s) derived from the non-linear spectrum of spectrums. This feature plays a vital role in speech recognition and also in identifying the singers from the audio input in our case.



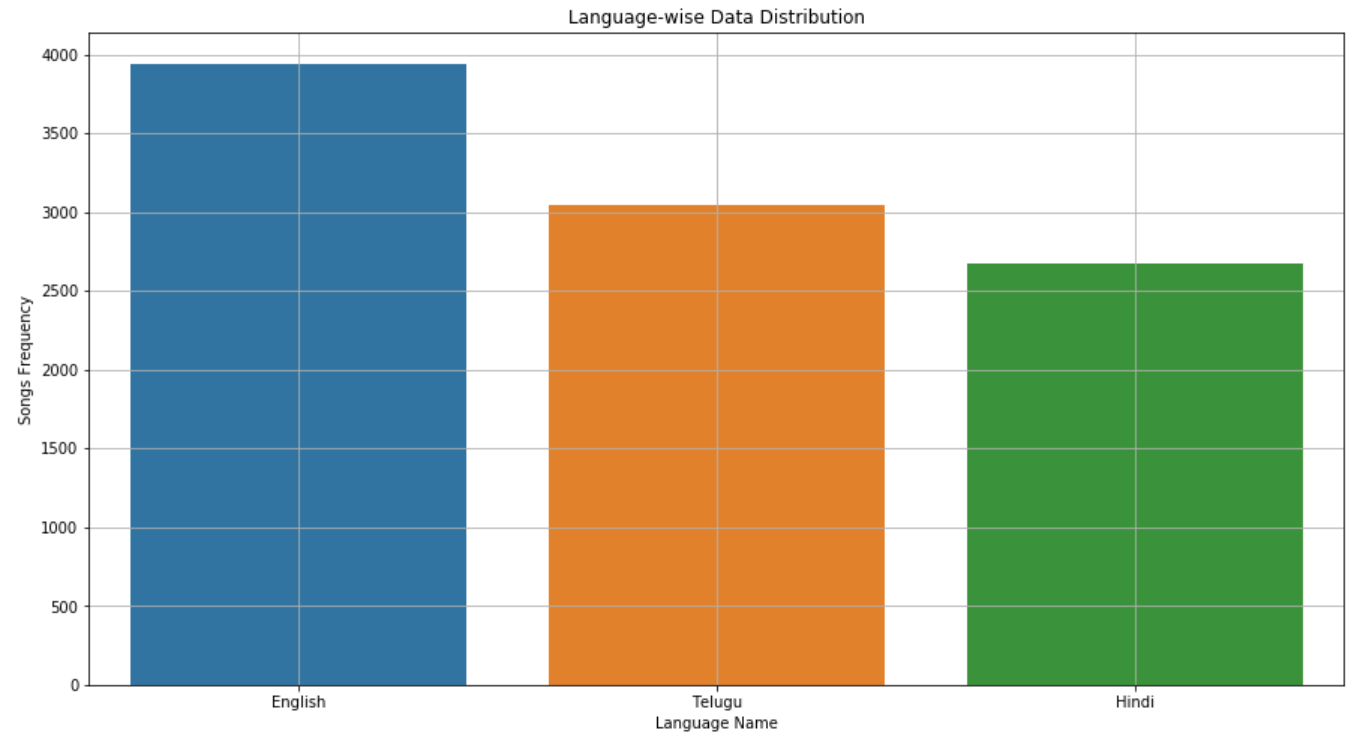
Most of the data under this feature is in the range of -10 to +20. Though the features obtained from MFCC are not robust to additive noise, it works well and suits the need in obtaining our goal as there is minimum or zero noise in our audio sample dataset. The data distribution is not normal and skewness can be observed in the data as we can see the tail protruding in the beginning.



The median is perfectly placed close to +6 suggesting the importance of this feature in speech recognition process. For our data, outliers of this feature are the data instances lying outside the range of -10 to +20 and the same information can be inferred from both histogram and boxplots. Amongst 9636 instances, we can observe 9523 unique values and 0 nulls.

**Language:**

Our dataset contains attribute ‘language’ which indicates to which language the song belongs to. This attribute would be helpful to identify the song language and shows how the songs are distributed among each language. The languages that we have chosen for the project are ‘English’, ‘Hindi’ and ‘Telugu’. Our data consists of more than ten thousand instances for English language, near to six thousand instances for Telugu language and more than six thousand instances for Hindi language. Below plot is the visualization for Genre Data spread across the selected three languages.



**Target:**

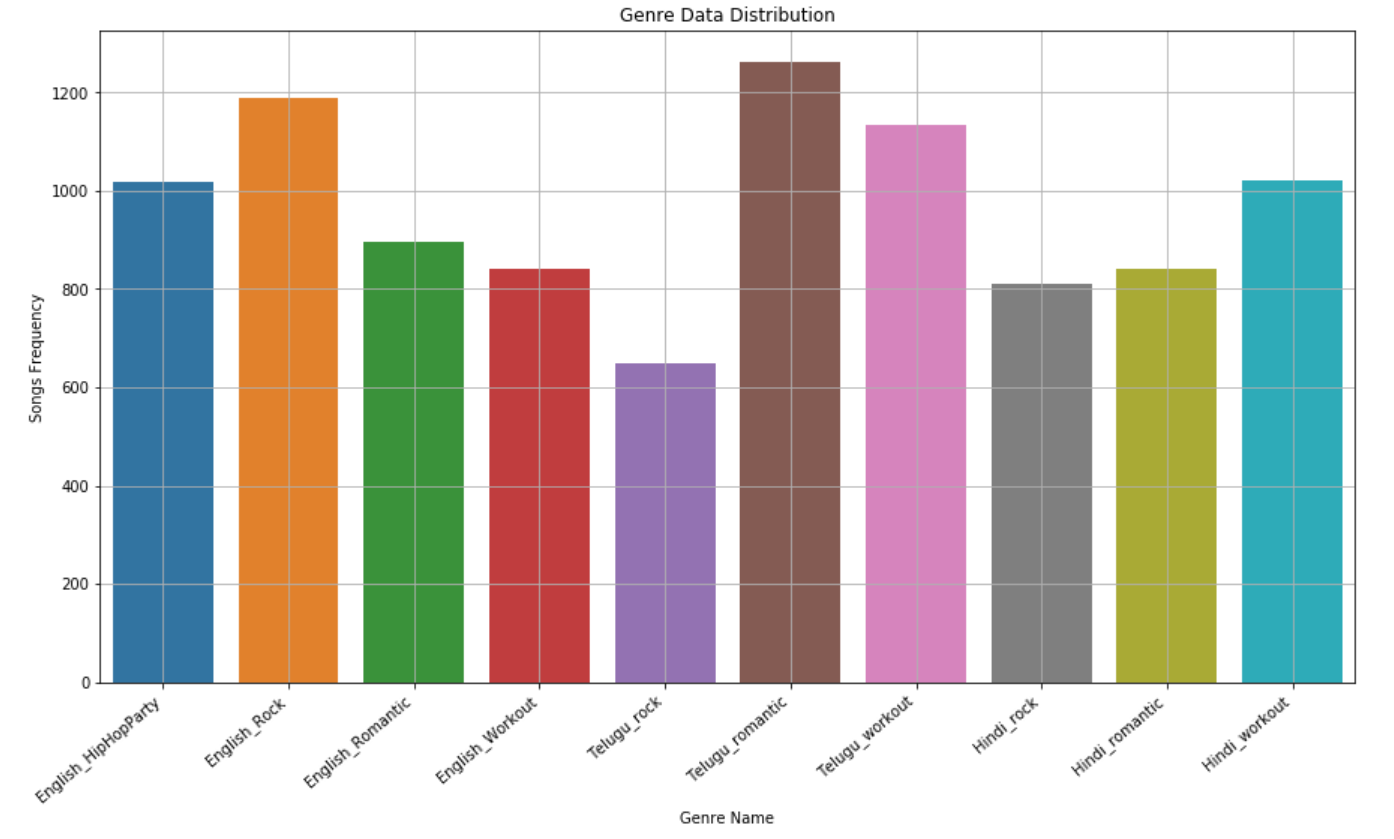
Our target for this project cosists of Genre classification and Artist voice detection. Though we have combined both the artists and Genre data into one dataframe to perform the data analysis, we have also analysed our data by focusing separately on artists and genres without clubbing them. Our target consists of 12 classes for artists data and 10 classes for genre data. The taget is distributed as follows:

* 5 English artists data
* 4 Telugu artists data
* 3 Hindi artists data
* 4 Engilish genre data
* 3 Telugu genre data
* 3 Hindi genre data

Majority of the data is concentrated for English language. We observed that there is data imbalance for the target varuable. When performing modelling if the model is overfitted or underfitted we would separate each language datasets and perform modelling individually. If the model is still overfitted or underfitted we would go ahead and perform oversampling or downsampling techniques where required.

Below plot visualizes the data distribution of songs amongst the 10 identified artists. It is to note that this visualization only depicts the genre data without including any artist specific data.

Artist ‘Telugu Romantic’ has most number of songs followed by ‘English Rock’ while ‘Telugu Rock’ has the lowest number of songs followed by ‘Hindi Rock’ in our data.



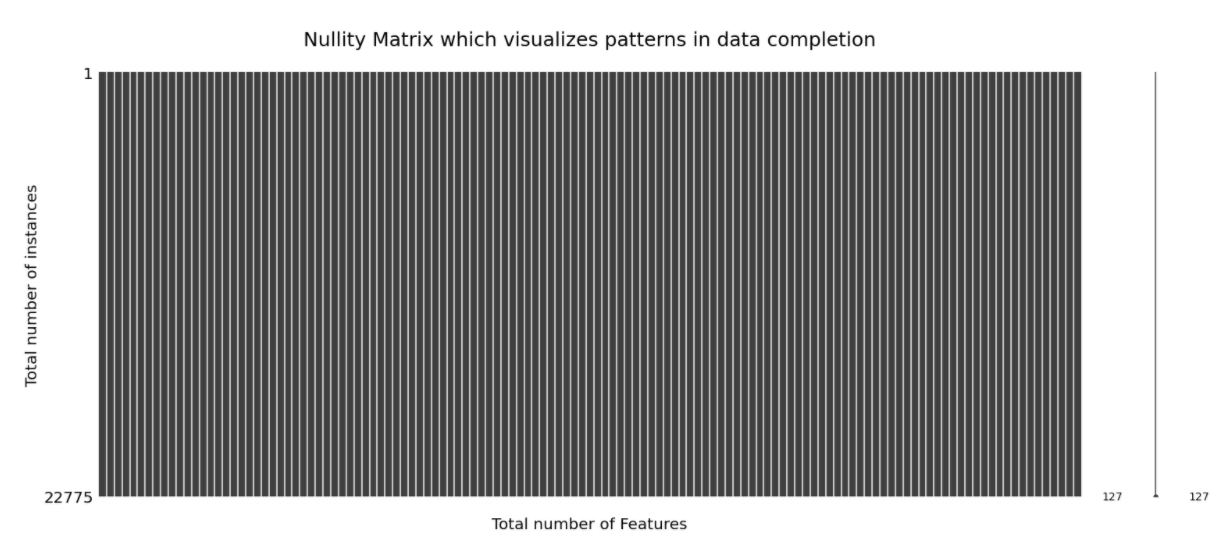
**Missing value analysis and outlier analysis**

[In this section, we identify the missing values and outliers and determine how we handle these values before analysis.]

We have been very cautious in building the dataset without any missing data and this proved to be a boon while performing pre-processing and exploratory data analysis as we were able to maintain a bare minimum value for the count of missing values.

For some instances, feature values can be observed to be zero and these are perfectly acceptable since there are gaps or silence within the audio files contributing towards these zero values. Noise too is eliminated from our data as the audio samples that have been collected (in mp3/mp4 formats) are directly from JioSavn and Deezer which provide good quality audio files without any distortion.

We have Explored the data to check if there are any missing values using the ‘missingno’ module in python. The output shows that the dataset is complete without any missing values. The missingno nullity matrix is a data-dense display which lets you quickly visually pick out patterns in data completion.



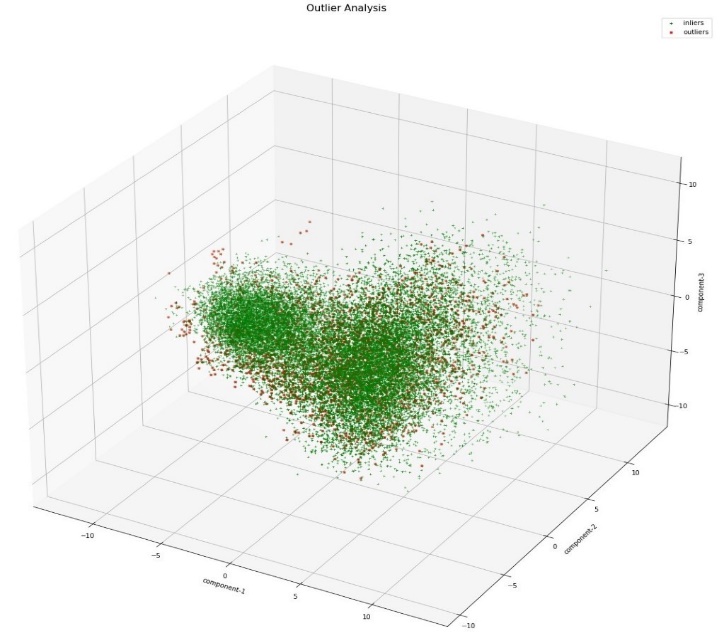
**Outlier Analysis:**

In descriptive statistics, a box plot is a method for graphically depicting groups of numerical data through their quartiles. Box plots above has lines extending vertically from the boxes (whiskers) indicating variability outside the upper and lower quartiles. We believe that there will be outliers in our data since we are extracting only the vocals from the wav file and not using the musical information to detect the voice of the Artist. This extraction would cause vacant spaces in the extracted spectrum which might be the primary reason for the outliers. For some songs, there might be unwanted noise in the wave files which might also be a reason to be an outlier. To detect outliers, we have used 2 methods, one is isolation forest analysis and found 2733 outliers in 22775(20042 are normal points) instances and Z-score analysis which gave us almost the same outliers for the dataset. To visualize the outliers from the isolation forest analysis we have applied the dimensional reduction technique to reduce the data features to 3-components and visualized it in 3-d graph. In the graph, red points are outliers predicted by isolation forest analysis.

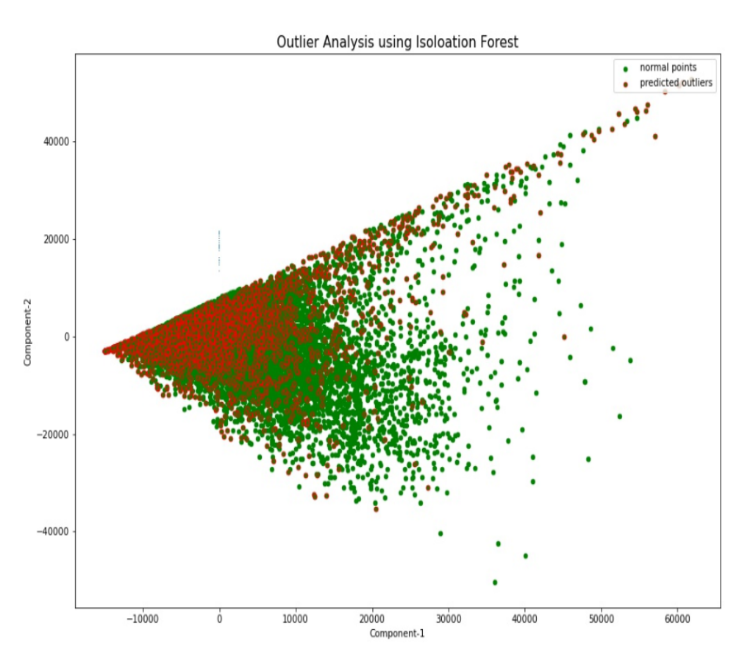
Outlier analysis was also done separately on the Artist Data and Genre Data which resulted in the below results. Using isolation forest analysis, we’ve found 1574 outliers in 13116 instances while the Z-score revealed the outliers to be 1396 with 11720 in-range points.

Similarly, isolation forest analysis resulted in 1159 outliers in 9636 instances while the Z-score provided the outliers count of 1154 which is very similar to the isolation forest outliers count.

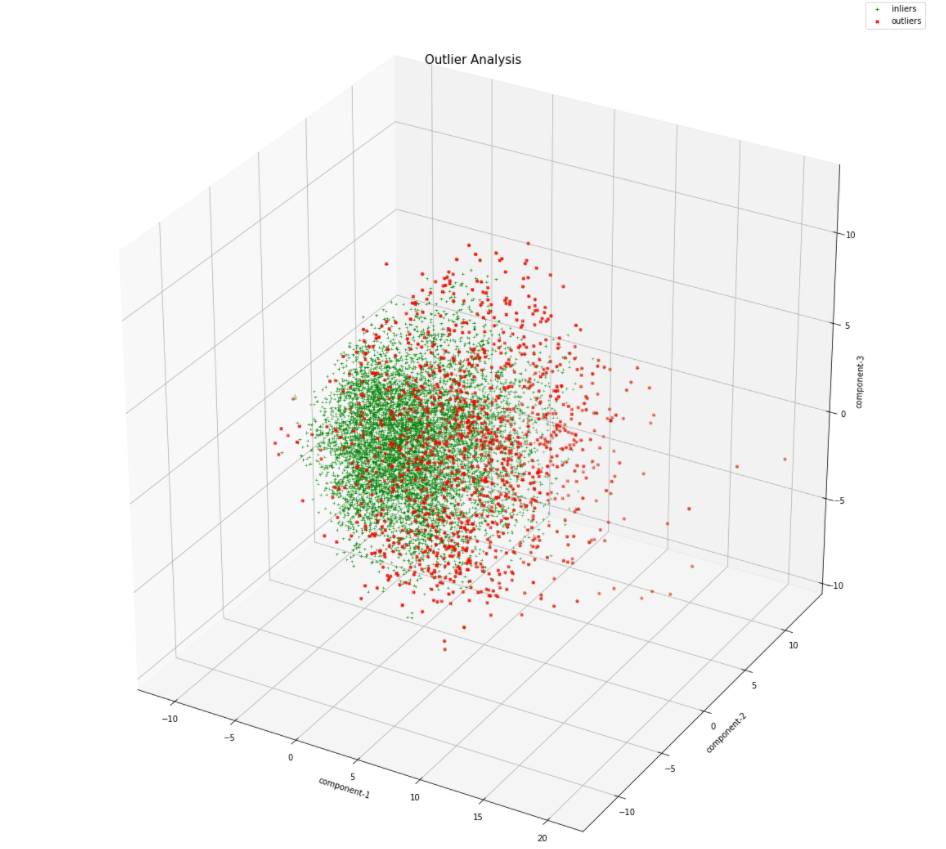
* Artist data:
  + Using z-score, out of 13116 instances 1396 are detected to be outliers for the artist data.
  + Using isolationforest, 1574 outliers from 13116 instances are detected as outliers.
  + By observing the outliers, we concluded that isolationforest is better approach for our data

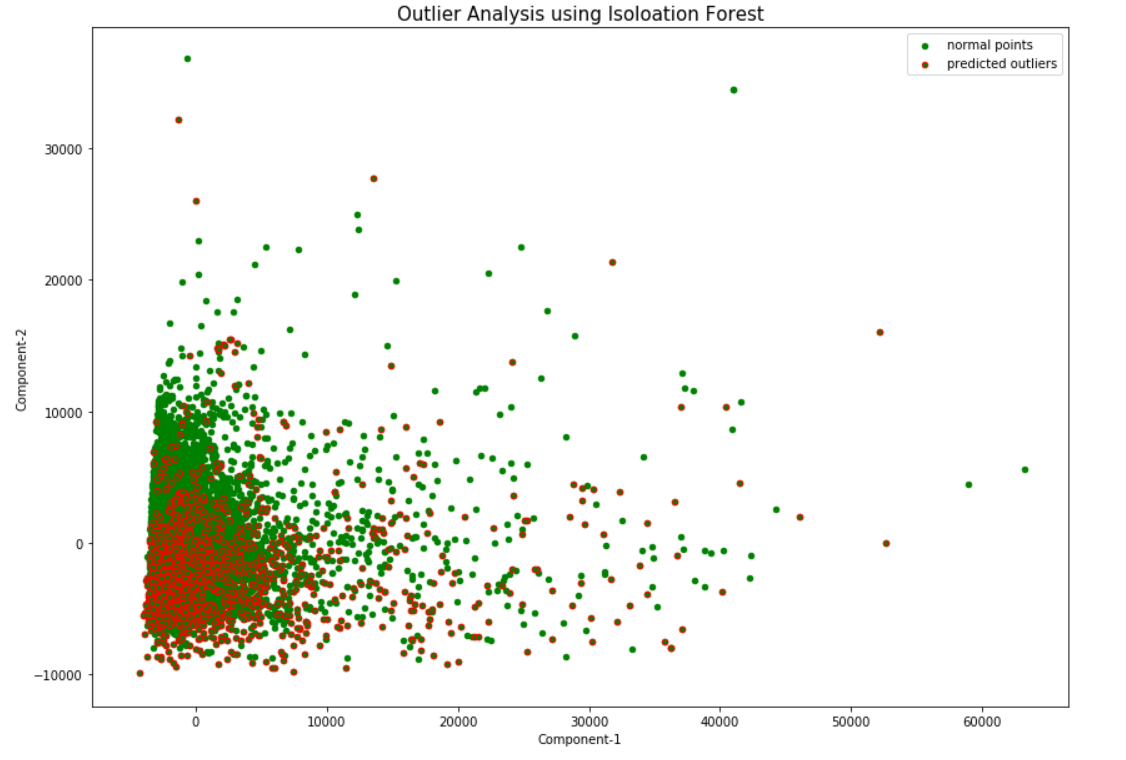


To visualize the outliers in 2-d graph using the isolation forest analysis we have applied the dimensional reduction technique(PCA) and reduce the data features to 2-components. In the graph, red points are outliers predicted by isolation forest analysis.



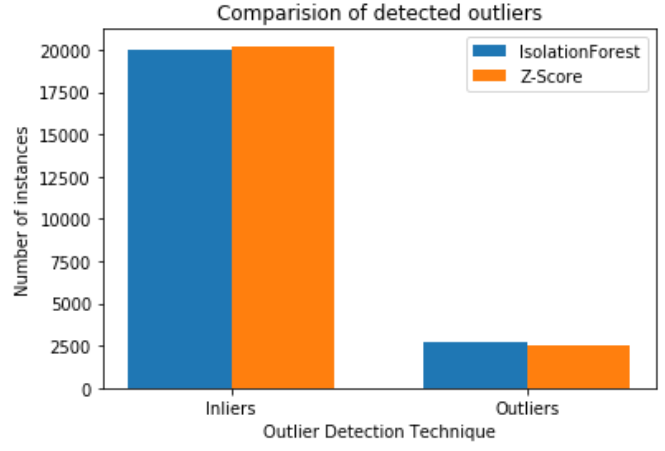
* Genre Data
  + Using Z-score, out of 9659 instances 1154 are detected to be outliers for the artist data.
  + Using IsolationForest, 1159 outliers from 9659 instances are detected as outliers.
  + By observing the outliers, we concluded that IsolationForest is better approach for our data



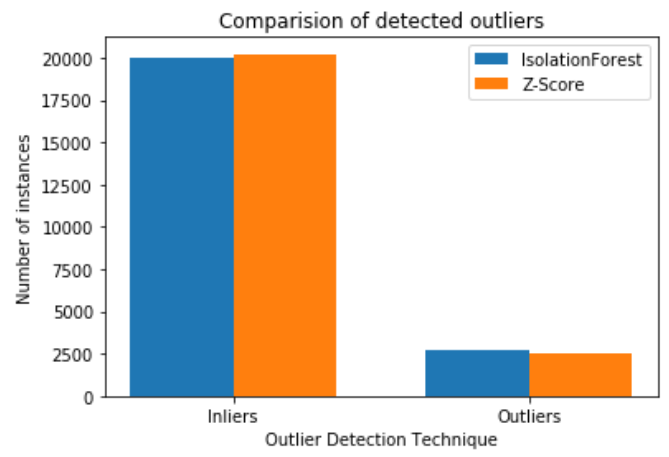


After analysis, we have observed that both the methods are detected approximately similar points as outliers. Hence, we have decided to go ahead with Isolation forest method for our future work. After removal of outliers, the size of our dataset is reduced to 20042 instances.

Below is the comparison of Outliers detected using both the methods Isolation Forest and Z-score for artists data alone.



Below is the comparison of Outliers detected using both the methods Isolation Forest and Z-score for genre data alone.



**Feature engineering and analysis**

[In this section, we identify the variables that are useful for predictive modeling and machine learning through correlation analysis. You may also reduce the dimension or derive new variables so that the predictive modeling can be more efficient and effective.]

In order to understand the complex relationships amongst the variables, it is necessary for us to study the degree to which one variable depends on the other. Correlation takes in three values, namely, positively correlation, negative correlation and neutral correlation.

Following an increase in the value of one variable makes the other variable to increase and this scenario is called positive correlation. Opposite to this, if the value of the other variable is decreased or increases in the opposite direction, this impact is inverse and thus negatively correlated. If we observe no change in the other variables value even after a huge change in one variable, there exists no relationship within these two variables and are neutral.

Pearson Correlation and Spearman Correlation provides a way to calculate the relationships between variables statistically. While the Spearman correlation is best suitable for the ordinal values, Pearson correlation helps better with the interval scaled values.

Pearson Correlation suits best when we are in need of finding the linear relationship between the variables while Spearman Correlation is apt when we look for the monotonic relationship between the variables. Both these methods provide the magnitude and direction along which the variables are related.

**Heat Map:**

With as many as 124 features, heatmap representation for the correlation amongst the variables is a bit congested to display and will not convey important details unless studied intrinsically.

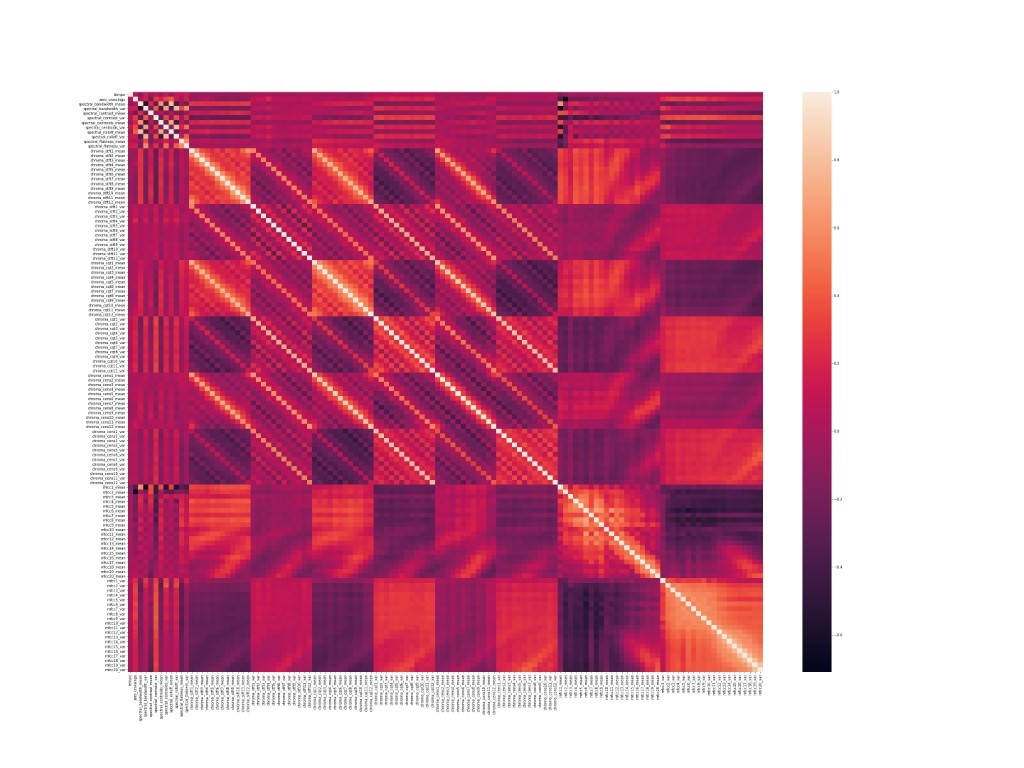
For this purpose, along with the below displayed heat map, we have also individually plotted the correlation between the top 5 positively correlated features, top 5 negatively correlated features.

We have also built a dataframe which conveys the Pearson Correlation values along with the features within which the correlation is obtained. This activity has been performed on all possible combinations of features.

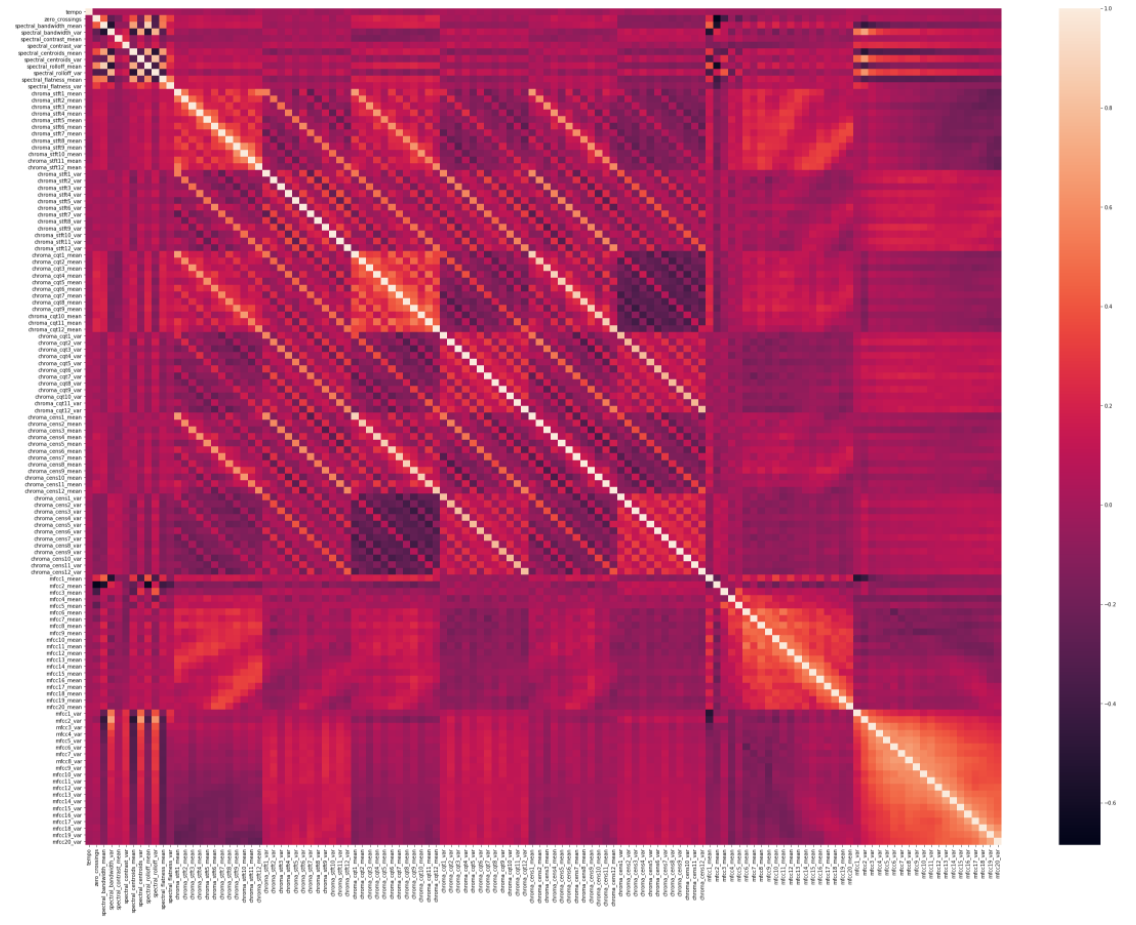
**Note**: With as many as 124 features, heatmap representation for the correlation amongst the variables is a bit congested to display and will not convey important details unless studied intrinsically.

For this purpose, along with the below displayed heat map, we have also individually plotted the correlation between the top 5 positively correlated features, top 5 negatively correlated features.

Below is the correlations between all the variables present in our artist data.



Below is the correlations between all the variables present in our genre data.



Below mentioned is the sample of the dataframe consisting of Pearson correlation values.



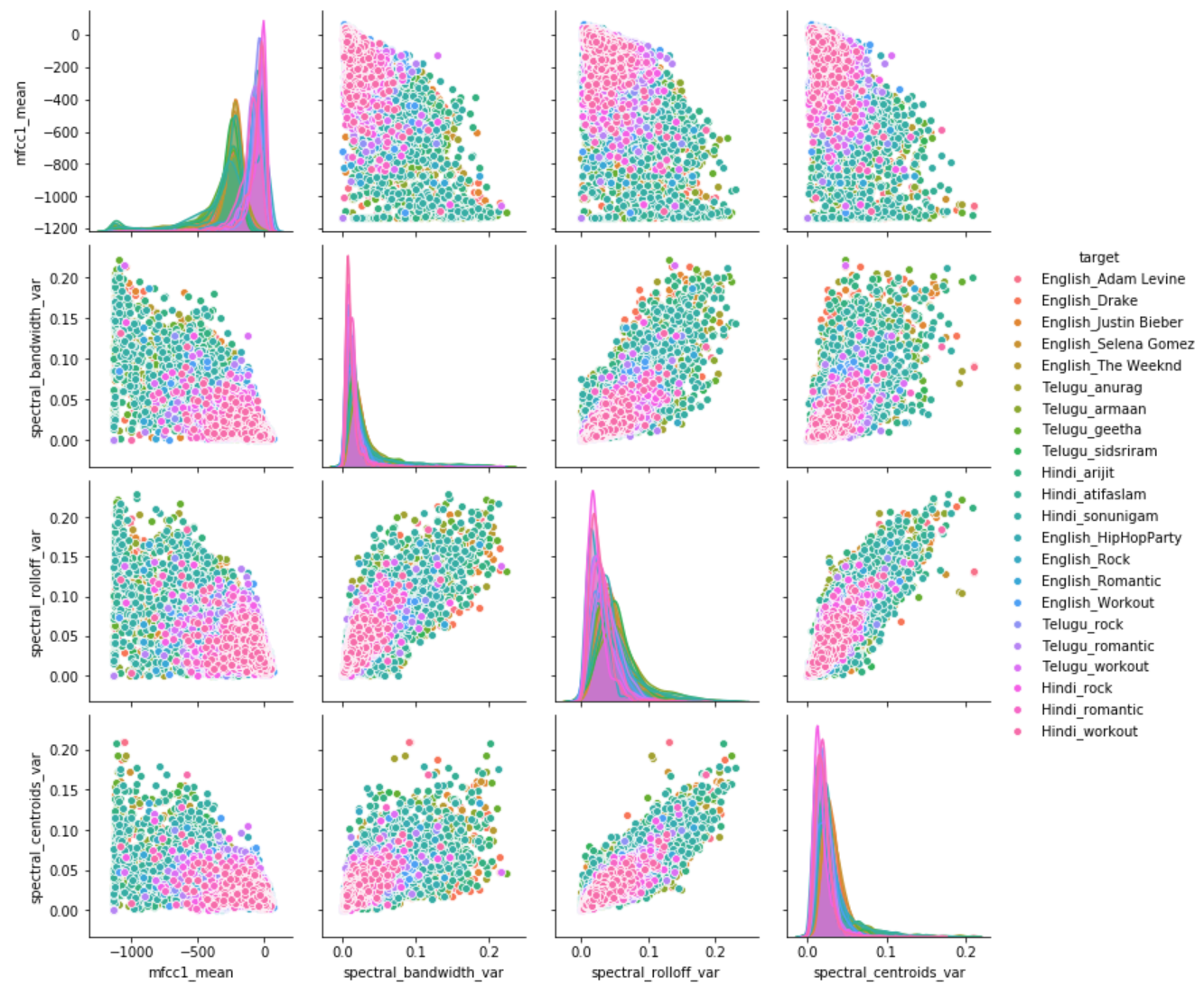
Below visualization is the Pairwise correlation of positively correlated variables on the artist data.

Shape, polygon

Description automatically generated

* Chroma\_cqt10\_mean is highly correlated with Chroma\_cens10\_mean with Pearson Coefficient of 0.8715.
* Spectral\_bandwidth\_mean is highly correlated with Spectral\_rolloff\_mean with Pearson Coefficient of 0.7987.

Below visualization is the Pairwise correlation of negatively correlated variables on the artist data.



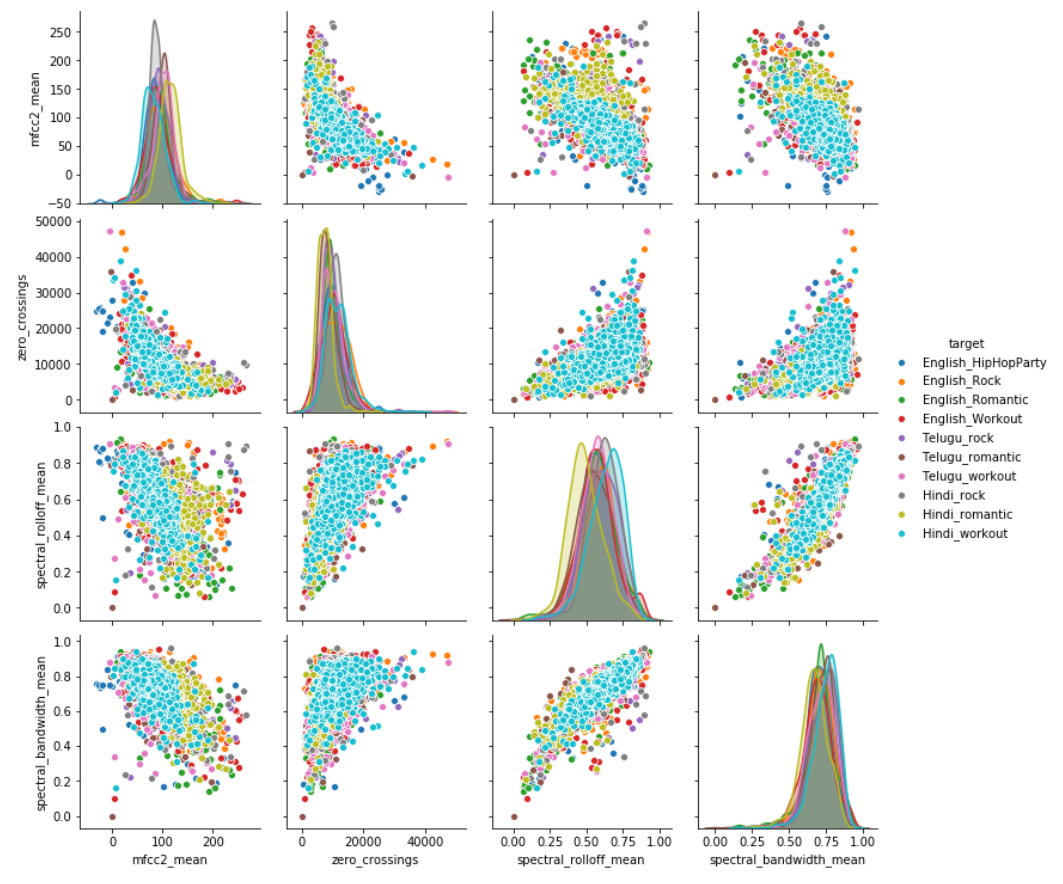
* Mfcc1\_mean is highly correlated with Spectral\_bandwidth\_var with Pearson Coefficient of negative 0.7092.
* Mfcc1\_mean is highly correlated with Spectral\_centroids\_var with Pearson Coefficient of negative 0.5518.

Below visualization is the Pairwise correlation of positively correlated variables on the genre data.



* Chroma\_cqt1\_mean is highly correlated with Chroma\_cens1\_mean with Pearson Coefficient of 0.8927.
* Spectral\_bandwidth\_mean is highly correlated with Spectral\_rolloff\_mean with Pearson Coefficient of 0.8737.

Below visualization is the Pairwise correlation of positively correlated variables on the genre data.



* Mfcc2\_mean is highly correlated with Zero\_crossings with Pearson Coefficient of negative 0.6842.
* Mfcc2\_mean is highly correlated with Spectral\_rolloff\_mean with Pearson Coefficient of negative 0.6203.

After performing the correlation analysis using heatmap, Pearson correlation and by plotting them with seaborn package, we were able to identify few important variables which needs to be included in our feature list while training the model in order to get effective results.

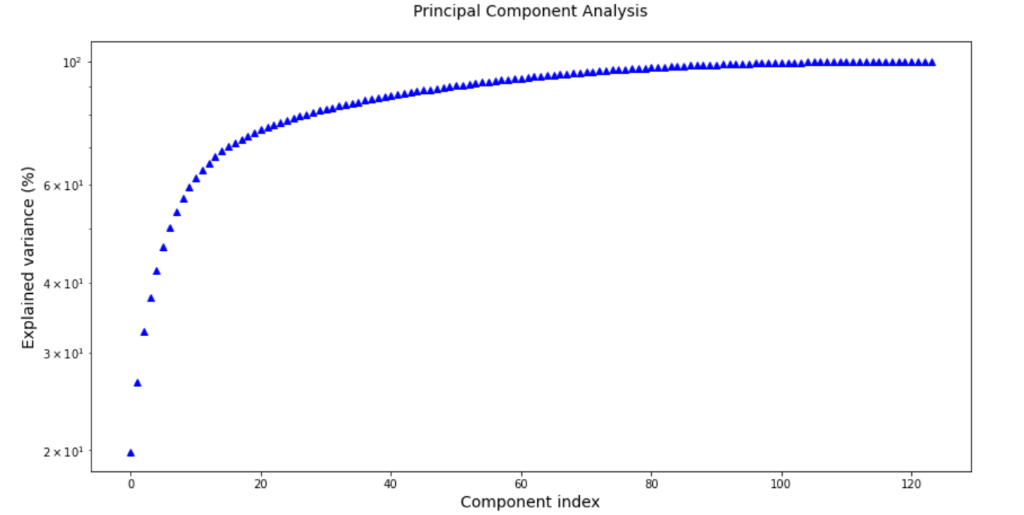
All 20 features from Chroma\_Constant-Q\_Transform along with the features from Chroma\_Energy\_Normalized\_Statistics, Spectral\_rolloff, Spectral\_Centroids and MFCC features.

Upon performing Principal Component Analysis, we were able to identify 96% of the variance being attained with just 80 features and this helped us in reducing the dimensions of our features from 124 to 80.

**Dimension Reduction:**

Our dataset has 127 attributes extracted from a wav file. We believe dimension reduction would help us in reducing the complexity of Machine Learning algorithm. To perform the dimensionality reduction, we have applied principal component analysis(PCA) technique. In 127 attributes of our dataset, we have 3 attributes which must be excluded for applying PCA. The excluded attributes are target variable, language attribute and Song ID(being a cardinal variable). We have conducted the explained variance analysis to know the minimal number of components required to preserve the maximum variance in the data. Below in the graph displaying the explained variance using PCA.

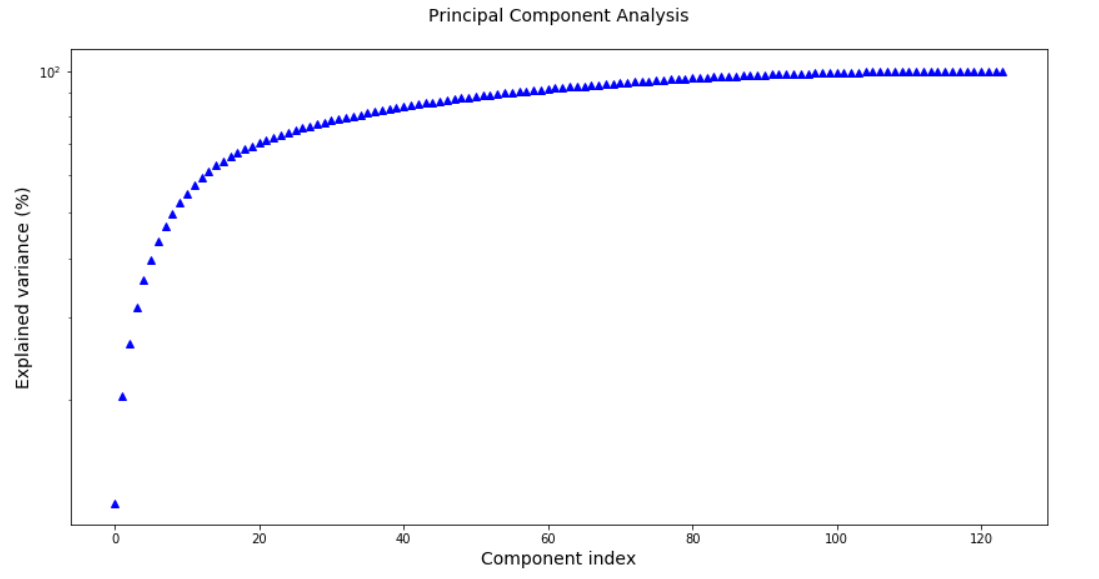
Below plot is generated on the instances from the artist data.



From the above graph we observed that by considering 80 components we could preserve more than 96% of variance in the data. By applying PCA we have reduced number of features from 124 to 80 without losing much variance in the data.

Below plot is generated on the instances from the genre data.

From the below graph we observed that by considering 90 components we could preserve more than 96% of variance in the data. By applying PCA we have reduced number of features from 124 to 90 without losing much variance in the data.

****

**Appendix**

[Provide the code or pseudo code, and any other information in the appendix here.]

Owing to the large size of our notebooks, instead of embedding them within this document, we will be uploading our code separately to the blackboard in two different notebooks.

Analyzing basic metrics, descriptive statistics, Non-graphical and graphical univariate analysis, Visualizations along with outlier detection, correlation analysis and PCA are all present in the notebook named ‘EDA\_Part1\_Artist\_V1.ipynb’ for the artists data.

Analyzing basic metrics, descriptive statistics, Non-graphical and graphical univariate analysis, Visualizations along with outlier detection, correlation analysis and PCA are all present in the notebook named ‘EDA\_Part1\_Genre\_V1.ipynb’ for the genre data.

Table of Contributions

The table below identifies contributors to various sections of this document.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Section** | **Writing** | **Editing** |
| **1** | **Analysis the basic metrics of variables** | **Kishor, Nirupam & Soujanya** | **Likhil & Vuthej** |
| **2** | **Non-graphical and graphical univariate analysis** | **Likhil & Vuthej** | **Kishor, Nirupam & Soujanya** |
| **3** | **Feature engineering and analysis** | **Kishor, Nirupam & Soujanya** | **Likhil & Vuthej** |
| **4** | **Appendix** | **EDA\_Part1\_V1.ipynb**  **Kishor, Nirupam & Soujanya**  **EDA\_Part2\_V1.ipynb**  **Vuthej & Likhil** | **EDA\_Part1\_V1.ipynb**  **Vuthej & Likhil**  **EDA\_Part2\_V1.ipynb**  **Kishor, Nirupam & Soujanya** |

**Grading**

The grade is given on the basis of quality, clarity, presentation, completeness, and writing of each section in the report. This is the grade of the group. Individual grades will be assigned at the end of the term when peer reviews are collected.