Appendix 4: Outlier Removal

In [1]:

```
import pandas as pd
import numpy as np
from scipy import stats
%matplotlib inline
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns
import random
import math
import statsmodels.api as sm
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import MinMaxScaler
```

In [2]:

```
df=pd.read_csv('spot_up_2_all.csv')
```

In [3]:

```
df.head()
```

Out[3]:

	Unnamed: 0	track	artist	uri	danceability	energy
0	0	Wild Things	Alessia Cara	spotify:track:2ZyuwVvV6Z3XJaXIFbspeE	0.741	0.626
1	1	Surfboard	Esquivel!	spotify:track:61APOtq25SCMuK0V5w2Kgp	0.447	0.247
2	2	Love Someone	Lukas Graham	spotify:track:2JqnpexlO9dmvjUMCaLCLJ	0.550	0.415
3	3	Music To My Ears (feat. Tory Lanez)	Keys N Krates	spotify:track:0cjfLhk8WJ3etPTCseKXtk	0.502	0.648
4	4	Juju On That Beat (TZ Anthem)	Zay Hilfigerrr & Zayion McCall	spotify:track:1lltf5ZXJc1by9SbPeljFd	0.807	0.887

5 rows × 31 columns

In [4]:

```
df=df.iloc[:,1:]
df.head()
```

Out[4]:

	track	artist	uri	danceability	energy	key	loud
0	Wild Things	Alessia Cara	spotify:track:2ZyuwVvV6Z3XJaXIFbspeE	0.741	0.626	1	-4
1	Surfboard	Esquivel!	spotify:track:61APOtq25SCMuK0V5w2Kgp	0.447	0.247	5	-14
2	Love Someone	Lukas Graham	spotify:track:2JqnpexIO9dmvjUMCaLCLJ	0.550	0.415	9	-6
3	Music To My Ears (feat. Tory Lanez)	Keys N Krates	spotify:track:0cjfLhk8WJ3etPTCseKXtk	0.502	0.648	0	-5
4	Juju On That Beat (TZ Anthem)	Zay Hilfigerrr & Zayion McCall	spotify:track:1lltf5ZXJc1by9SbPeljFd	0.807	0.887	1	-3

5 rows × 30 columns

In [5]:

```
df_maings=df.drop(df[(df['genres']=='Avant-garde')].index)
df_maings=df_maings.drop(df[(df['genres']=='Comedy')].index)
df_maings=df_maings.drop(df[(df['genres']=='Other')].index)
df_maings=df_maings.drop(df[(df['genres']=='Flamenco')].index)
```

In [6]:

```
df_maings=df_maings.reset_index(drop=True)
df.head()
```

Out[6]:

	track	artist	uri	danceability	energy	key	loud
0	Wild Things	Alessia Cara	spotify:track:2ZyuwVvV6Z3XJaXIFbspeE	0.741	0.626	1	-4
1	Surfboard	Esquivel!	spotify:track:61APOtq25SCMuK0V5w2Kgp	0.447	0.247	5	-14
2	Love Someone	Lukas Graham	spotify:track:2JqnpexlO9dmvjUMCaLCLJ	0.550	0.415	9	-6
3	Music To My Ears (feat. Tory Lanez)	Keys N Krates	spotify:track:0cjfLhk8WJ3etPTCseKXtk	0.502	0.648	0	-5
4	Juju On That Beat (TZ Anthem)	Zay Hilfigerrr & Zayion McCall	spotify:track:1lltf5ZXJc1by9SbPeljFd	0.807	0.887	1	-3

5 rows × 30 columns

In [7]:

```
decades=df_maings['Decade'].unique()
decades
```

Out[7]:

```
array(['10s', '00s', '90s', '80s', '70s', '60s'], dtype=object)
```

In [8]:

```
df_total=pd.DataFrame(columns=df_maings.columns)
for k in range(len(decades)):
    totals=[]
    df_1=df_maings[df_maings['Decade']==decades[k]]
    df_1=df_1.reset_index(drop=True)
    genres=df_1['genres'].unique()
    for j in range(len(genres)):
        df_3=df_1[df_1['genres']==genres[j]]
        df_3=df_3.reset_index(drop=True)
        totals.append(len(df_3))
        iters=np.zeros(10)
        frac=totals[j]/100
        complete=False
        checked=False
        while len(df_3)/totals[j]>0.9 and not complete:
            totalh=len(df_3[df_3['target']==1])
            totalf=len(df_3[df_3['target']==0])
            if(totalh>totalf):
                df_2=df_3[df_3['target']==1]
            elif(totalh<totalf):
                df_2=df_3[df_3['target']==0]
            else:
                df_2=df_3
            maxdifs=(np.max(df_2[['danceability','energy','instrumentalness','liveness'
,'speechiness','acousticness','valence','duration_ms','tempo','chorus_hit']])-np.mean(d
f_2[['danceability','energy','instrumentalness','liveness','speechiness','acousticness'
,'valence','duration_ms','tempo','chorus_hit']]))/np.std(df_2[['danceability','energy',
'instrumentalness','liveness','speechiness','acousticness','valence','duration_ms','tem
po','chorus_hit']],ddof=1)
            mindifs=(np.mean(df_2[['danceability','energy','instrumentalness','livenes
s','speechiness','acousticness','valence','duration_ms','tempo','chorus_hit']])-np.min(
df_2[['danceability','energy','instrumentalness','liveness','speechiness','acousticnes
s','valence','duration_ms','tempo','chorus_hit']]))/np.std(df_2[['danceability','energ
y','instrumentalness','liveness','speechiness','acousticness','valence','duration_ms',
'tempo','chorus_hit']],ddof=1)
            redo=True
            while redo:
                if((np.max(maxdifs)>np.max(mindifs)) and np.max(maxdifs)>3):
                    for i in range(len(maxdifs)):
                        if(maxdifs[i]==np.max(maxdifs)):
                            if(i==0):
                                if((iters[i]<frac) or checked==True):
                                    df_3=df_3.drop(df_2[df_2['danceability']==np.max(df
_2['danceability'])].index)
                                    df_3=df_3.reset_index(drop=True)
                                    iters[i]=iters[i]+1

                                    redo=False
                                    checked=False
                                else:
                                    maxdifs[i]=0
                            elif(i==1):
                                if((iters[i]<frac) or checked==True):
                                    df_3=df_3.drop(df_2[df_2['energy']==np.max(df_2['en
ergy'])].index)
                                    df_3=df_3.reset_index(drop=True)
                                    iters[i]=iters[i]+1

                                    redo=False
```

```
                                checked=False
                            else:
                                maxdifs[i]=0
                        elif(i==2):
                            if((iters[i]<frac) or checked==True):
                                df_3=df_3.drop(df_2[df_2['instrumentalness']==np.ma
x(df_2['instrumentalness'])].index)
                                df_3=df_3.reset_index(drop=True)
                                iters[i]=iters[i]+1

                                redo=False
                                checked=False
                            else:
                                maxdifs[i]=0

                        elif(i==3):
                            if((iters[i]<frac) or checked==True):
                                iters[i]=iters[i]+1
                                df_3=df_3.drop(df_2[df_2['liveness']==np.max(df_2[
'liveness'])].index)

                                df_3=df_3.reset_index(drop=True)

                                redo=False
                                checked=False
                            else:
                                maxdifs[i]=0

                        elif(i==4):
                            if((iters[i]<frac) or checked==True):
                                iters[i]=iters[i]+1
                                df_3=df_3.drop(df_2[df_2['speechiness']==np.max(df_
2['speechiness'])].index)

                                df_3=df_3.reset_index(drop=True)

                                redo=False
                                checked=False
                            else:
                                maxdifs[i]=0
                        elif(i==5):
                            if((iters[i]<frac) or checked==True):
                                iters[i]=iters[i]+1
                                df_3=df_3.drop(df_2[df_2['acousticness']==np.max(df
_2['acousticness'])].index)

                                df_3=df_3.reset_index(drop=True)

                                redo=False
                                checked=False
                            else:
                                maxdifs[i]=0
                        elif(i==6):
                            if((iters[i]<frac) or checked==True):
                                iters[i]=iters[i]+1
                                df_3=df_3.drop(df_2[df_2['valence']==np.max(df_2['v
alence'])].index)

                                df_3=df_3.reset_index(drop=True)

                                redo=False
                                checked=False
                            else:
                                maxdifs[i]=0
                        elif(i==7):
```

```
                                if((iters[i]<frac) or checked==True):
                                    iters[i]=iters[i]+1
                                    df_3=df_3.drop(df_2[df_2['duration_ms']==np.max(df_
2['duration_ms'])].index)
                                    df_3=df_3.reset_index(drop=True)

                                    redo=False
                                    checked=False
                                else:
                                    maxdifs[i]=0
                            elif(i==8):
                                if((iters[i]<frac) or checked==True):
                                    iters[i]=iters[i]+1
                                    df_3=df_3.drop(df_2[df_2['tempo']==np.max(df_2['tem
po'])].index)
                                    df_3=df_3.reset_index(drop=True)

                                    redo=False
                                    checked=False
                                else:
                                    maxdifs[i]=0
                            elif(i==9):
                                if((iters[i]<frac) or checked==True):
                                    iters[i]=iters[i]+1
                                    df_3=df_3.drop(df_2[df_2['chorus_hit']==np.max(df_2
['chorus_hit'])].index)
                                    df_3=df_3.reset_index(drop=True)

                                    redo=False
                                    checked=False
                                else:
                                    maxdifs[i]=0

                            break
                    elif((np.max(maxdifs)<np.max(mindifs)) and np.max(mindifs)>3):

                        for i in range(len(mindifs)):
                            if(mindifs[i]==np.max(mindifs)):
                                if(i==0):
                                    if((iters[i]<frac) or checked==True):
                                        iters[i]=iters[i]+1
                                        df_3=df_3.drop(df_2[df_2['danceability']==np.min(df
_2['danceability'])].index)
                                        df_3=df_3.reset_index(drop=True)

                                        redo=False
                                        checked=False
                                    else:
                                        mindifs[i]=0
                                elif(i==1):
                                    if((iters[i]<frac) or checked==True):
                                        iters[i]=iters[i]+1
                                        df_3=df_3.drop(df_2[df_2['energy']==np.min(df_2['en
ergy'])].index)
                                        df_3=df_3.reset_index(drop=True)

                                        redo=False
                                        checked=False
                                    else:
                                        mindifs[i]=0
                                elif(i==2):
```

```
                                if((iters[i]<frac) or checked==True):
                                    iters[i]=iters[i]+1
                                    df_3=df_3.drop(df_2[df_2['instrumentalness']==np.mi
n(df_2['instrumentalness'])].index)
                                    df_3=df_3.reset_index(drop=True)

                                    redo=False
                                    checked=False
                                else:
                                    mindifs[i]=0
                            elif(i==3):
                                if((iters[i]<frac) or checked==True):
                                    iters[i]=iters[i]+1
                                    df_3=df_3.drop(df_2[df_2['liveness']==np.min(df_2[
'liveness'])].index)
                                    df_3=df_3.reset_index(drop=True)

                                    redo=False
                                    checked=False
                                else:
                                    mindifs[i]=0
                            elif(i==4):
                                if((iters[i]<frac) or checked==True):
                                    iters[i]=iters[i]+1
                                    df_3=df_3.drop(df_2[df_2['speechiness']==np.min(df_
2['speechiness'])].index)
                                    df_3=df_3.reset_index(drop=True)

                                    redo=False
                                    checked=False
                                else:
                                    mindifs[i]=0
                            elif(i==5):
                                if((iters[i]<frac) or checked==True):
                                    iters[i]=iters[i]+1
                                    df_3=df_3.drop(df_2[df_2['acousticness']==np.min(df
_2['acousticness'])].index)
                                    df_3=df_3.reset_index(drop=True)

                                    redo=False
                                    checked=False
                                else:
                                    mindifs[i]=0
                            elif(i==6):
                                if((iters[i]<frac) or checked==True):
                                    iters[i]=iters[i]+1
                                    df_3=df_3.drop(df_2[df_2['valence']==np.min(df_2['v
alence'])].index)
                                    df_3=df_3.reset_index(drop=True)

                                    redo=False
                                    checked=False
                                else:
                                    mindifs[i]=0
                            elif(i==7):
                                if((iters[i]<frac) or checked==True):
                                    iters[i]=iters[i]+1
                                    df_3=df_3.drop(df_2[df_2['duration_ms']==np.min(df_
2['duration_ms'])].index)
                                    df_3=df_3.reset_index(drop=True)
```

```
                                    redo=False
                                    checked=False
                                else:
                                    mindifs[i]=0
                            elif(i==8):
                                if((iters[i]<frac) or checked==True):
                                    iters[i]=iters[i]+1
                                    df_3=df_3.drop(df_2[df_2['tempo']==np.min(df_2['tem
po'])].index)
                                    df_3=df_3.reset_index(drop=True)

                                    redo=False
                                    checked=False
                                else:
                                    mindifs[i]=0
                            elif(i==9):
                                if((iters[i]<frac) or checked==True):
                                    iters[i]=iters[i]+1
                                    df_3=df_3.drop(df_2[df_2['chorus_hit']==np.min(df_2
['chorus_hit'])].index)
                                    df_3=df_3.reset_index(drop=True)

                                    redo=False
                                    checked=False
                                else:
                                    mindifs[i]=0
                            break
                    else:
                        if(checked==False):
                            checked=True
                            break
                        else:
                            complete=True
                            break
            df_total=pd.concat([df_total,df_3],axis=0)
```

In [9]:

```
df_total.head()
```

Out[9]:

	track	artist	uri	danceability	energy	key	louc
0	Wild Things	Alessia Cara	spotify:track:2ZyuwVvV6Z3XJaXIFbspeE	0.741	0.626	1	-
1	Love Someone	Lukas Graham	spotify:track:2JqnpexIO9dmvjUMCaLCLJ	0.550	0.415	9	-
2	Here's To Never Growing Up	Avril Lavigne	spotify:track:0qwcGscxUHGZTgq0zcaqk1	0.482	0.873	0	-
3	Crawling Back To You	Daughtry	spotify:track:6BDtTzjbJ5kKKSWcJT8MIX	0.438	0.919	0	-
4	Faster	Matt Nathanson	spotify:track:6pIKFdrBnKF0y3CRuceTDh	0.742	0.853	9	-

5 rows × 30 columns

In [10]:

```
df_total.describe()
```

Out[10]:

	danceability	energy	loudness	speechiness	acousticness	instrumentalnes
count	33354.000000	33354.000000	33354.000000	33354.000000	33354.000000	33354.00000
mean	0.541620	0.587900	-9.989044	0.066563	0.350320	0.14171
std	0.176678	0.251042	5.262882	0.066292	0.335579	0.29473
min	0.058800	0.000251	-49.253000	0.022000	0.000000	0.00000
25%	0.424000	0.407000	-12.545500	0.033400	0.034700	0.00000
50%	0.554000	0.613000	-8.992000	0.042800	0.236000	0.00007
75%	0.670000	0.794000	-6.183000	0.067000	0.648000	0.03010
max	0.988000	1.000000	3.744000	0.957000	0.996000	1.00000

In [11]:

```
df_total=df_total.reset_index(drop=True)
```

In [12]:

```
df_total.head()
```

Out[12]:

	track	artist	uri	danceability	energy	key	louc
0	Wild Things	Alessia Cara	spotify:track:2ZyuwVvV6Z3XJaXIFbspeE	0.741	0.626	1	-
1	Love Someone	Lukas Graham	spotify:track:2JqnpexIO9dmvjUMCaLCLJ	0.550	0.415	9	-
2	Here's To Never Growing Up	Avril Lavigne	spotify:track:0qwcGscxUHGZTgq0zcaqk1	0.482	0.873	0	-
3	Crawling Back To You	Daughtry	spotify:track:6BDtTzjbJ5kKKSWcJT8MIX	0.438	0.919	0	-
4	Faster	Matt Nathanson	spotify:track:6pIKFdrBnKF0y3CRuceTDh	0.742	0.853	9	-

5 rows × 30 columns

In [13]:

```
df_total.to_csv('data_final.csv')
```

In [ ]:

```

```