**Step-by-step Instructions for Graph Analysis**

import pandas as pd

import yellowbrick

#Step 1: Load data into a dataframe

data = pd.read\_csv("C:\\Users\\Gabe\\Documents\\Bellevue University\\Data Mining\\Project\\Data\\top-spotify-songs-from-20102019-by-year\\top10s.csv", encoding = 'ISO-8859-1')

# Step 2: check the dimension of the table

print("The dimension of the table is: ", data.shape)

#Step 3: Look at the data

print(data.head(5))

#Step 4: what type of variables are in the table

print("Describe Data")

print(data.describe())

print("")

print("Summarized Data")

print(data.describe(include=['O']))

#Step 4: what type of variables are in the table

import matplotlib.pyplot as plt

# set up the figure size

plt.rcParams['figure.figsize'] = (20, 10)

# make subplots

fig, axes = plt.subplots(nrows = 3, ncols = 3)

# Specify the features of interest

num\_features = ['pop', 'spch', 'bpm', 'nrgy', 'dnce', 'live', 'val', 'dur', 'acous']

xaxes = num\_features

yaxes = ['Counts', 'Counts', 'Counts', 'Counts', 'Counts', 'Counts', 'Counts', 'Counts', 'Counts']

# draw histograms

axes = axes.ravel()

for idx, ax in enumerate(axes):

ax.hist(data[num\_features[idx]].dropna(), bins=40)

ax.set\_xlabel(xaxes[idx], fontsize=20)

ax.set\_ylabel(yaxes[idx], fontsize=20)

ax.tick\_params(axis='both', labelsize=15)

plt.show()

#Step 6: Pearson Ranking

#set up the figure size

#%matplotlib inline

plt.rcParams['figure.figsize'] = (15, 7)

# import the package for visulization of the correlation

from yellowbrick.features import Rank2D

# extract the numpy arrays from the data frame

import numpy as np

X = np.asmatrix(data[num\_features])

# instantiate the visualizer with the Covariance ranking algorithm

visualizer = Rank2D(features=num\_features, algorithm='pearson')

visualizer.fit(X) # Fit the data to the visualizer

visualizer.transform(X) # Transform the data

visualizer.poof() # Draw/show/poof the data

plt.show()