Association Rules Assignment Data Set: my_movies 1. Import Necessary libraries import pandas as pd import numpy as np import matplotlib.pyplot as plt import seaborn as sns 2. Import Data movies = pd.read_csv('my_movies.csv') movies Out[2]: **V1** V2 V3 V4 V5 Sixth Sense Gladiator LOTR1 Harry Potter1 Patriot LOTR2 Harry Potter2 LOTR Braveheart Green Mile Sixth Sense LOTR1 Harry Potter1 Green Mile LOTR2 0 1 0 Gladiator 1 0 0 0 0 Patriot Braveheart NaN NaN 0 1 LOTR1 LOTR2 0 0 1 0 0 0 0 2 NaN NaN NaN 0 3 Gladiator Patriot Sixth Sense NaN NaN 0 1 0 0 0 0 0 0 0 4 Gladiator Patriot Sixth Sense NaN NaN 1 0 Gladiator Sixth Sense NaN 1 0 0 0 0 Patriot NaN 0 0 0 1 0 0 0 6 Harry Potter1 Harry Potter2 NaN NaN NaN 1 0 0 Gladiator Patriot NaN 1 0 1 0 0 0 0 NaN NaN Gladiator Patriot Sixth Sense NaN NaN 1 1 0 0 1 0 0 0 0 0 Sixth Sense LOTR 0 0 0 1 Gladiator Green Mile NaN 3. Data Understanding movies.head() In [3]: V3 V5 Sixth Sense Gladiator LOTR1 Harry Potter1 Patriot LOTR2 Harry Potter2 LOTR Braveheart Green Mile Out[3]: **V1** V2 V4 0 0 0 **0** Sixth Sense LOTR1 Harry Potter1 Green Mile LOTR2 1 1 0 0 1 Gladiator NaN Patriot Braveheart NaN 0 0 LOTR1 LOTR2 0 0 1 0 1 0 0 0 NaN NaN NaN Sixth Sense 0 Gladiator Patriot NaN NaN 0 0 0 0 1 0 0 Gladiator Patriot Sixth Sense NaN NaN movies.shape (10, 15)Out[4]: In [5]: movies.info() <class 'pandas.core.frame.DataFrame'> RangeIndex: 10 entries, 0 to 9 Data columns (total 15 columns): Column Non-Null Count Dtype 10 non-null 0 V1 object 10 non-null V2 1 object 2 V3 7 non-null object 3 V4 2 non-null object 4 V5 1 non-null object Sixth Sense 10 non-null 5 int64 Gladiator 10 non-null 6 int64 7 LOTR1 10 non-null int64 8 Harry Potter1 10 non-null int64 Patriot 9 10 non-null int64 L0TR2 10 10 non-null int64 11 Harry Potter2 10 non-null int64 12 LOTR 10 non-null int64 13 Braveheart 10 non-null int64 14 Green Mile 10 non-null int64 dtypes: int64(10), object(5) memory usage: 1.3+ KB movies.isna().sum() In [6]: V1 0 Out[6]: V2 0 3 V3 V4 8 V5 9 0 Sixth Sense Gladiator 0 LOTR1 0 Harry Potter1 0 Patriot 0 L0TR2 0 0 Harry Potter2 L0TR Braveheart 0 Green Mile dtype: int64 movies.describe() Sixth Sense Gladiator LOTR1 Harry Potter1 **Patriot** LOTR2 Harry Potter2 LOTR Braveheart Green Mile Out[7]: 10.000000 10.000000 10.000000 10.000000 10.000000 10.000000 10.000000 10.000000 10.000000 10.000000 count 0.600000 0.700000 0.200000 0.200000 0.600000 0.200000 0.100000 0.100000 0.100000 0.200000 mean 0.516398 0.483046 0.516398 0.316228 0.421637 0.421637 0.421637 0.316228 0.421637 std 0.316228 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 min 0.000000 0.250000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 25% 0.000000 **50**% 1.000000 1.000000 0.000000 0.000000 1.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 75% 1.000000 1.000000 0.000000 0.000000 1.000000 0.000000 0.000000 0.000000 0.000000 1.000000 1.000000 1.000000 1.000000 1.000000 1.000000 1.000000 1.000000 1.000000 1.000000 movies.dtypes object Out[8]: V2 object V3 object V4 object object V5 int64 Sixth Sense Gladiator int64 LOTR1 int64 Harry Potter1 int64 Patriot int64 L0TR2 int64 Harry Potter2 int64 L0TR int64 int64 Braveheart Green Mile int64 dtype: object holly_movies = movies.iloc[:,5:] holly_movies Sixth Sense Gladiator LOTR1 Harry Potter1 Patriot LOTR2 Harry Potter2 LOTR Braveheart Green Mile Out[9]: 0 1 0 2 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 4. Apriori Algorithm In [10]: pip install mlxtend Requirement already satisfied: mlxtend in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (0.21.0) Requirement already satisfied: numpy>=1.16.2 in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from mlxtend) (1.21.5) Requirement already satisfied: joblib>=0.13.2 in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from mlxtend) (1.1.0) Requirement already satisfied: scipy>=1.2.1 in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from mlxtend) (1.7.3) Requirement already satisfied: pandas>=0.24.2 in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from mlxtend) (1.4.2) Requirement already satisfied: matplotlib>=3.0.0 in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from mlxtend) (3.5.1) Requirement already satisfied: scikit-learn>=1.0.2 in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from mlxtend) (1.0.2) Requirement already satisfied: setuptools in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from mlxtend) (61.2.0) Requirement already satisfied: fonttools>=4.22.0 in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (4.25.0) Requirement already satisfied: pillow>=6.2.0 in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (9.0.1) Requirement already satisfied: packaging>=20.0 in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (21.3) Requirement already satisfied: cycler>=0.10 in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (0.11.0) Requirement already satisfied: pyparsing>=2.2.1 in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (3.0.4) Requirement already satisfied: python-dateutil>=2.7 in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (2.8.2) Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (1.3.2) Requirement already satisfied: pytz>=2020.1 in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from pandas>=0.24.2->mlxtend) (2021.3) Requirement already satisfied: six>=1.5 in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from python-dateutil>=2.7->matplotlib>=3.0.0->mlxtend) (1.16.0) Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\mohammed faisal khan\anaconda3\lib\site-packages (from scikit-learn>=1.0.2->mlxtend) (2.2.0) Note: you may need to restart the kernel to use updated packages. a) Association Rule for support = 0.1from mlxtend.frequent_patterns import apriori from mlxtend.frequent_patterns import association_rules import warnings warnings.filterwarnings('ignore') movie_items_1 = apriori(holly_movies, min_support = 0.1, use_colnames = True) movie_items_1 Out[12]: support itemsets (Sixth Sense) 0.7 1 (Gladiator) 2 0.2 (LOTR1) 3 0.2 (Harry Potter1) 4 0.6 (Patriot) (LOTR2) 0.2 (Harry Potter2) (LOTR) 0.1 8 0.1 (Braveheart) 0.2 9 (Green Mile) 10 0.5 (Sixth Sense, Gladiator) (LOTR1, Sixth Sense) 11 0.1 12 0.1 (Harry Potter1, Sixth Sense) 13 0.4 (Patriot, Sixth Sense) 14 0.1 (LOTR2, Sixth Sense) (LOTR, Sixth Sense) 15 0.1 16 0.2 (Green Mile, Sixth Sense) 17 0.6 (Patriot, Gladiator) 18 0.1 (LOTR, Gladiator) 19 0.1 (Gladiator, Braveheart) 20 0.1 (Green Mile, Gladiator) 21 0.1 (LOTR1, Harry Potter1) 22 0.2 (LOTR1, LOTR2) 23 0.1 (Green Mile, LOTR1) 24 0.1 (Harry Potter1, LOTR2) 25 0.1 (Harry Potter2, Harry Potter1) (Green Mile, Harry Potter1) 26 0.1 27 0.1 (Patriot, Braveheart) 28 0.1 (Green Mile, LOTR2) 29 (Green Mile, LOTR) 0.1 (Patriot, Sixth Sense, Gladiator) 30 0.4 (LOTR, Sixth Sense, Gladiator) 31 0.1 (Green Mile, Sixth Sense, Gladiator) 32 0.1 (LOTR1, Harry Potter1, Sixth Sense) 33 0.1 34 0.1 (LOTR1, LOTR2, Sixth Sense) 35 0.1 (Green Mile, LOTR1, Sixth Sense) (LOTR2, Harry Potter1, Sixth Sense) 36 0.1 37 0.1 (Green Mile, Harry Potter1, Sixth Sense) 38 0.1 (Green Mile, LOTR2, Sixth Sense) (Green Mile, LOTR, Sixth Sense) 39 0.1 40 0.1 (Patriot, Gladiator, Braveheart) 0.1 41 (Green Mile, LOTR, Gladiator) 42 0.1 (LOTR1, Harry Potter1, LOTR2) 43 0.1 (Green Mile, LOTR1, Harry Potter1) 44 0.1 (Green Mile, LOTR1, LOTR2) (Green Mile, Harry Potter1, LOTR2) 45 0.1 (Green Mile, LOTR, Sixth Sense, Gladiator) 46 0.1 (LOTR2, LOTR1, Harry Potter1, Sixth Sense) 47 0.1 (Green Mile, LOTR1, Harry Potter1, Sixth Sense) 48 0.1 49 0.1 (Green Mile, LOTR1, LOTR2, Sixth Sense) 50 (Green Mile, Harry Potter1, Sixth Sense, LOTR2) (Green Mile, LOTR1, Harry Potter1, LOTR2) 51 0.1 (LOTR1, Sixth Sense, Green Mile, LOTR2, Harry ... 52 In [13]: movie_rules_1 = association_rules(movie_items_1, metric = 'lift') movie_rules_1 consequents antecedent support consequent support support confidence Out[13]: antecedents lift leverage conviction 0.833333 1.190476 (Sixth Sense) (Gladiator) 0.6 0.7 0.08 1.80 0.5 (Gladiator) (Sixth Sense) 0.7 0.6 0.5 0.714286 1.190476 0.08 1.40 0.6 2 (LOTR1) (Sixth Sense) 0.2 0.500000 0.833333 -0.02 0.80 0.1 (LOTR1) 0.6 0.2 (Sixth Sense) 0.1 -0.02 0.96 0.500000 0.833333 4 (Harry Potter1) (Sixth Sense) 0.2 0.6 0.1 -0.02 0.80 (LOTR1) (Green Mile, LOTR2, Sixth Sense, Harry Potter1) 0.500000 5.000000 241 0.2 0.1 0.1 0.08 1.80 0.6 0.1 242 (Sixth Sense) (Green Mile, LOTR1, LOTR2, Harry Potter1) 0.1 0.166667 1.666667 0.04 1.08 243 (Green Mile) (LOTR1, LOTR2, Sixth Sense, Harry Potter1) 0.2 0.1 0.1 0.500000 5.000000 0.08 1.80 (LOTR2) (Green Mile, LOTR1, Harry Potter1, Sixth Sense) 0.2 0.1 244 0.1 0.500000 5.000000 0.08 1.80 **245** (Harry Potter1) (Green Mile, LOTR1, LOTR2, Sixth Sense) 0.2 0.1 0.500000 5.000000 0.08 1.80 0.1 246 rows × 9 columns In [14]: a_1 = movie_rules_1[movie_rules_1.lift > 1] a_1 consequents antecedent support consequent support support confidence lift leverage conviction Out[14]: antecedents 0 (Sixth Sense) (Gladiator) 0.6 0.7 0.5 0.833333 1.190476 0.08 1.80 (Gladiator) (Sixth Sense) 0.7 0.6 0.714286 1.190476 0.08 1.40 0.5 0.6 0.6 0.666667 1.111111 6 (Patriot) (Sixth Sense) 0.4 0.04 1.20 (Sixth Sense) 0.6 0.6 0.666667 1.111111 0.04 1.20 (Patriot) 0.4 10 (LOTR) 0.1 0.6 0.1 1.000000 1.666667 0.04 (Sixth Sense) inf (LOTR1) (Green Mile, LOTR2, Sixth Sense, Harry Potter1) 0.2 0.500000 5.000000 0.08 1.80 241 0.1 0.1 (Sixth Sense) (Green Mile, LOTR1, LOTR2, Harry Potter1) 0.6 0.1 0.166667 1.666667 0.04 1.08 242 0.1 (LOTR1, LOTR2, Sixth Sense, Harry Potter1) 243 (Green Mile) 0.2 0.1 0.500000 5.000000 0.08 1.80 0.1 (LOTR2) (Green Mile, LOTR1, Harry Potter1, Sixth Sense) 244 0.2 0.1 0.500000 5.000000 0.08 1.80 0.1 0.2 **245** (Harry Potter1) (Green Mile, LOTR1, LOTR2, Sixth Sense) 0.1 0.1 0.500000 5.000000 0.08 1.80 236 rows × 9 columns b_1 = a_1.sort_values("lift", ascending = False) b_1 consequents antecedent support consequent support support confidence antecedents Out[15]: lift leverage conviction (Green Mile, Harry Potter1) 168 (LOTR1, Sixth Sense) 0.1 0.1 0.1 1.000000 10.000000 0.09 inf (LOTR1, LOTR2, Sixth Sense) 1.000000 10.000000 222 (Green Mile, Harry Potter1) 0.1 0.1 0.1 0.09 inf 192 (Green Mile, Harry Potter1) (LOTR2, Sixth Sense) 0.1 0.1 0.1 1.000000 10.000000 0.09 inf (Green Mile, Sixth Sense, Gladiator) 134 (LOTR) 0.1 0.1 1.000000 10.000000 0.09 inf 0.1 183 (LOTR2, Sixth Sense) (Green Mile, LOTR1) 0.1 0.1 0.1 1.000000 10.000000 0.09 inf 0.714286 1.190476 1 (Gladiator) (Sixth Sense) 0.7 0.6 0.5 0.08 1.4 42 (Patriot, Gladiator) 0.6 0.666667 1.111111 0.04 1.2 (Sixth Sense) 0.6 0.4 39 (Sixth Sense) 0.6 1.111111 (Patriot, Gladiator) 0.6 0.4 0.666667 0.04 1.2 0.666667 1.111111 1.2 (Sixth Sense) (Patriot) 0.6 0.6 0.4 0.04 (Sixth Sense) 1.2 (Patriot) 0.666667 1.111111 236 rows × 9 columns Visualization for 0.1 support: plt.figure(figsize = (8,6)) plt.scatter(a_1["support"], a_1["confidence"]) plt.title("Association Rules Plot", size = 20, color = "black") plt.xlabel("Support", size = 12) plt.ylabel("Confidence", size = 12) plt.show() Association Rules Plot 1.0 0.8 Confidence 0.4 0.2 0.5 0.1 Support b) Association Rule for support = 0.2 movie_items_2 = apriori(holly_movies, min_support = 0.2, use_colnames = True) movie_items_2 Out[17]: support itemsets 0.6 (Sixth Sense) 0.7 (Gladiator) 1 (LOTR1) 2 0.2 3 0.2 (Harry Potter1) 4 0.6 (Patriot) 0.2 (LOTR2) 6 0.2 (Green Mile) 0.5 (Sixth Sense, Gladiator) (Patriot, Sixth Sense) 8 0.4 0.2 (Green Mile, Sixth Sense) 9 10 0.6 (Patriot, Gladiator) 11 0.2 (LOTR1, LOTR2) 0.4 (Patriot, Sixth Sense, Gladiator) 12 movie_rules_2 = association_rules(movie_items_2, metric = 'lift') movie_rules_2 consequents antecedent support consequent support support confidence lift leverage conviction Out[18]: antecedents 0 (Sixth Sense) (Gladiator) 0.7 0.833333 1.190476 0.08 1.8 0.5 0.714286 1.190476 1 (Gladiator) (Sixth Sense) 0.7 0.6 0.5 0.08 1.4 2 (Sixth Sense) 0.6 0.666667 1.111111 0.04 1.2 (Patriot) 0.6 0.4 3 0.6 0.666667 1.111111 0.04 1.2 (Sixth Sense) (Patriot) 0.6 0.4 4 (Green Mile) (Sixth Sense) 0.2 0.6 0.2 1.000000 1.666667 0.08 inf 0.333333 1.666667 (Sixth Sense) (Green Mile) 0.6 0.2 0.2 0.08 1.2 6 (Gladiator) 0.6 0.7 1.000000 1.428571 0.18 inf (Patriot) 0.6 7 0.857143 1.428571 2.8 (Gladiator) (Patriot) 0.7 0.6 0.6 0.18 8 (LOTR1) (LOTR2) 0.2 0.2 1.000000 5.000000 0.16 inf 0.2 9 (LOTR2) (LOTR1) 0.2 0.2 0.2 1.000000 5.000000 0.16 inf 10 (Patriot, Sixth Sense) 0.4 0.7 1.000000 1.428571 0.12 inf (Gladiator) 0.4 11 0.6 0.666667 1.111111 0.04 (Patriot, Gladiator) (Sixth Sense) 0.6 0.4 1.2 (Sixth Sense, Gladiator) 0.5 0.6 0.800000 1.333333 0.10 2.0 (Patriot) 0.4 0.666667 1.3333333 13 (Patriot) (Sixth Sense, Gladiator) 0.6 0.5 0.4 0.10 1.5 (Sixth Sense) 14 (Patriot, Gladiator) 0.6 0.6 0.4 0.666667 1.111111 0.04 1.2 15 (Patriot, Sixth Sense) 0.4 0.571429 1.428571 0.12 (Gladiator) 0.7 0.4 1.4 In [19]: a_2 = movie_rules_2[movie_rules_2.lift > 1] a_2 Out[19]: antecedents consequents antecedent support consequent support support confidence lift leverage conviction 0.833333 1.190476 0 (Sixth Sense) (Gladiator) 0.6 0.7 0.5 80.0 1.8 1 (Gladiator) 0.7 0.6 0.5 0.714286 1.190476 0.08 1.4 (Sixth Sense) 2 (Patriot) (Sixth Sense) 0.6 0.6 0.666667 1.111111 0.04 1.2 0.4 3 0.6 0.6 0.666667 1.111111 0.04 1.2 (Sixth Sense) (Patriot) 4 (Green Mile) (Sixth Sense) 0.2 0.6 0.2 1.000000 1.666667 0.08 inf 5 0.2 0.2 0.333333 1.666667 0.08 1.2 (Sixth Sense) (Green Mile) 0.6 0.18 6 (Patriot) (Gladiator) 0.6 0.7 1.000000 1.428571 inf 7 (Gladiator) (Patriot) 0.7 0.6 0.857143 1.428571 0.18 2.8 0.6 8 (LOTR2) (LOTR1) 0.2 0.2 0.2 1.000000 5.000000 0.16 inf 9 (LOTR2) (LOTR1) 0.2 0.2 0.2 1.000000 5.000000 0.16 inf (Patriot, Sixth Sense) 1.000000 1.428571 10 (Gladiator) 0.4 0.7 0.12 inf 0.4 11 0.6 0.6 0.666667 1.111111 0.04 1.2 (Patriot, Gladiator) (Sixth Sense) 12 (Sixth Sense, Gladiator) (Patriot) 0.5 0.6 0.800000 1.333333 0.10 2.0 13 0.6 0.5 0.666667 1.3333333 0.10 1.5 (Patriot) (Sixth Sense, Gladiator) 0.4 14 (Sixth Sense) (Patriot, Gladiator) 0.6 0.6 0.666667 1.111111 0.04 1.2 15 (Gladiator) 0.7 0.4 0.4 0.571429 1.428571 0.12 1.4 (Patriot, Sixth Sense) b_2 = a_2.sort_values("lift", ascending = False) b_2 consequents antecedent support consequent support support confidence Out[20]: antecedents lift leverage conviction 8 (LOTR1) (LOTR2) 0.2 0.2 0.2 1.000000 5.000000 0.16 inf (LOTR2) (LOTR1) 0.2 0.2 1.000000 5.000000 0.2 0.16 inf 0.2 1.000000 1.666667 (Green Mile) (Sixth Sense) 0.6 0.2 80.0 inf 0.333333 1.666667 (Sixth Sense) 0.6 0.2 0.2 0.08 1.2 (Green Mile) 7 (Gladiator) 0.7 0.6 0.857143 1.428571 0.18 2.8 (Patriot) (Gladiator) inf (Patriot) 0.6 0.7 0.6 1.000000 1.428571 0.18 (Patriot, Sixth Sense) 1.000000 1.428571 10 (Gladiator) 0.4 0.7 0.12 inf 0.4 0.4 0.571429 1.428571 0.12 1.4 15 (Gladiator) (Patriot, Sixth Sense) 0.7 0.4 (Sixth Sense, Gladiator) 0.6 12 (Patriot) 0.5 0.4 0.800000 1.333333 0.10 2.0 13 (Patriot) (Sixth Sense, Gladiator) 0.6 0.5 0.4 0.666667 1.333333 0.10 1.5 0 (Sixth Sense) (Gladiator) 0.6 0.7 0.833333 1.190476 0.08 1.8 0.5 1 1.4 (Gladiator) (Sixth Sense) 0.7 0.6 0.5 0.714286 1.190476 0.08 2 0.6 0.6 0.666667 1.111111 0.04 1.2 (Patriot) (Sixth Sense) 0.4 3 1.2 (Sixth Sense) (Patriot) 0.6 0.6 0.4 0.666667 1.111111 0.04 11 (Patriot, Gladiator) (Sixth Sense) 0.6 0.6 0.666667 1.111111 0.04 1.2 0.4 0.6 0.6 0.666667 1.111111 0.04 1.2 14 (Sixth Sense) (Patriot, Gladiator) 0.4 Visualization for 0.2 support : plt.figure(figsize = (8,6)) plt.scatter(a_2["support"], a_2["confidence"]) plt.title("Association Rules Plot", size = 20, color = "black") plt.xlabel("Support", size = 12) plt.ylabel("Confidence", size = 12) plt.show() **Association Rules Plot** 1.0 0.9 0.8 Confidence 0.7 0.5 0.4 0.3 0.25 0.30 0.35 0.40 0.45 0.50 0.55 0.60 Support c) Association Rule for support = 0.3 movie_items_3 = apriori(holly_movies, min_support = 0.3, use_colnames = True) movie_items_3 Out[22]: itemsets support (Sixth Sense) 0.6 0.7 (Gladiator) 0.6 (Patriot) 0.5 (Sixth Sense, Gladiator) (Patriot, Sixth Sense) 0.6 (Patriot, Gladiator) 0.4 (Patriot, Sixth Sense, Gladiator) movie_rules_3 = association_rules(movie_items_3, metric = 'lift') In [23]: lift leverage conviction antecedents consequents antecedent support consequent support support confidence Out[23]: 0 (Sixth Sense) (Gladiator) 0.7 0.833333 1.190476 0.08 1.8 0.6 1.4 1 (Gladiator) (Sixth Sense) 0.7 0.6 0.5 0.714286 1.190476 0.08 2 (Patriot) (Sixth Sense) 0.6 0.6 0.4 0.666667 1.111111 0.04 1.2 3 0.666667 1.111111 (Sixth Sense) (Patriot) 0.6 0.6 0.4 0.04 1.2 4 0.6 0.7 1.000000 1.428571 0.18 inf (Patriot) (Gladiator) 0.6 0.7 0.6 0.857143 1.428571 0.18 5 (Gladiator) (Patriot) 2.8 0.6 6 (Patriot, Sixth Sense) 0.4 0.7 1.000000 1.428571 0.12 inf (Gladiator) 0.4 (Patriot, Gladiator) (Sixth Sense) 0.6 0.6 0.666667 1.111111 0.04 1.2 0.4 0.800000 1.333333 0.10 2.0 8 (Sixth Sense, Gladiator) (Patriot) 0.5 0.6 0.4 (Patriot) (Sixth Sense, Gladiator) 0.6 0.5 0.666667 1.333333 0.10 1.5 0.4 10 (Sixth Sense) (Patriot, Gladiator) 0.6 0.6 0.666667 1.111111 0.04 1.2 0.4 11 (Gladiator) (Patriot, Sixth Sense) 0.7 0.4 0.4 0.571429 1.428571 0.12 1.4 In [24]: a_3 = movie_rules_3[movie_rules_3.lift > 1] a_3 Out[24]: antecedents consequents antecedent support consequent support support confidence lift leverage conviction 0 (Sixth Sense) (Gladiator) 0.6 0.7 0.833333 1.190476 80.0 1.8 (Gladiator) (Sixth Sense) 0.7 0.6 0.5 0.714286 1.190476 0.08 1.4 1 2 (Sixth Sense) (Patriot) 0.6 0.6 0.666667 1.111111 0.04 1.2 0.4 3 0.666667 1.111111 1.2 (Sixth Sense) (Patriot) 0.6 0.6 0.04 4 (Patriot) (Gladiator) 0.6 0.7 1.000000 1.428571 0.18 inf (Patriot) 0.7 0.6 0.857143 1.428571 0.18 2.8 (Gladiator) 0.6 (Patriot, Sixth Sense) (Gladiator) 0.4 0.7 1.000000 1.428571 0.12 inf (Patriot, Gladiator) (Sixth Sense) 0.6 0.666667 1.111111 0.04 1.2 0.6 0.4 0.800000 1.333333 (Sixth Sense, Gladiator) (Patriot) 0.5 0.6 0.10 2.0 0.6 0.666667 1.333333 (Patriot) (Sixth Sense, Gladiator) 0.5 0.4 0.10 1.5 10 (Sixth Sense) (Patriot, Gladiator) 0.6 0.6 0.666667 1.111111 0.04 1.2 11 (Gladiator) (Patriot, Sixth Sense) 0.7 0.4 0.4 0.571429 1.428571 0.12 1.4 b_3 = a_3.sort_values("lift", ascending = False) In [25]: b_3 consequents antecedent support consequent support support confidence lift leverage conviction Out[25]: antecedents 5 (Gladiator) 0.7 0.6 0.6 0.857143 1.428571 0.18 2.8 (Patriot) 0.7 1.000000 1.428571 0.18 (Patriot) (Gladiator) 0.6 0.6 inf (Patriot, Sixth Sense) 1.000000 1.428571 6 (Gladiator) 0.4 0.7 0.12 inf 0.4 11 0.4 0.571429 1.428571 (Gladiator) (Patriot, Sixth Sense) 0.7 0.4 0.12 1.4 (Sixth Sense, Gladiator) 0.800000 1.333333 8 (Patriot) 0.5 0.6 0.4 0.10 2.0 0.666667 1.333333 (Patriot) (Sixth Sense, Gladiator) 0.6 0.5 0.4 0.10 1.5 0.833333 1.190476 0 (Gladiator) 0.7 80.0 1.8 (Sixth Sense) 0.6 0.5 (Gladiator) (Sixth Sense) 0.6 0.714286 1.190476 0.08 0.7 0.5 3 (Sixth Sense) (Patriot) 0.6 0.6 0.4 0.666667 1.111111 0.04 1.2 7 (Patriot, Gladiator) 0.666667 1.111111 0.04 1.2 (Sixth Sense) 0.6 0.6 0.666667 1.111111 10 (Sixth Sense) (Patriot, Gladiator) 0.6 0.6 0.04 1.2 0.4 Visualization for 0.3 support: plt.figure(figsize = (8,6)) plt.scatter(a_3["support"], a_3["confidence"]) plt.title("Association Rules Plot", size = 20, color = "black") plt.xlabel("Support", size = 12) plt.ylabel("Confidence", size = 12) plt.show() **Association Rules Plot** 1.0 0.9 Confidence 0.7 0.6 0.450 0.475 0.500 0.550 0.575 0.400 0.425 0.525 Support d) Association Rule for support = 0.5 movie_items_4 = apriori(holly_movies, min_support = 0.5, use_colnames = True) movie_items_4 support itemsets Out[27]: (Sixth Sense) 0.6 0.7 (Gladiator) 0.5 (Sixth Sense, Gladiator) (Patriot, Gladiator) movie_rules_4 = association_rules(movie_items_4, metric = 'lift') movie_rules_4 antecedents consequents antecedent support consequent support support confidence lift leverage conviction Out[28]: 0 (Sixth Sense) 0.6 0.833333 1.190476 0.08 1.8 (Gladiator) 0.7 0.5 (Gladiator) (Sixth Sense) 0.7 0.6 0.714286 1.190476 0.08 1.4 1.000000 1.428571 0.6 0.7 0.18 (Patriot) (Gladiator) 0.6 inf 0.857143 1.428571 (Gladiator) (Patriot) 0.7 0.6 0.6 0.18 In [29]: a_4 = movie_rules_4[movie_rules_4.lift > 1] a_4 lift leverage conviction antecedents consequents antecedent support consequent support support confidence Out[29]: **0** (Sixth Sense) (Gladiator) 0.6 0.7 0.833333 1.190476 0.08 1.8 (Sixth Sense) 0.7 0.6 0.714286 1.190476 0.08 1.4 (Gladiator) 0.5 0.6 0.7 1.000000 1.428571 0.18 (Patriot) (Gladiator) 0.7 (Gladiator) 0.6 0.6 0.857143 1.428571 0.18 2.8 (Patriot) b_4 = a_4.sort_values("lift", ascending = False) In [30]: b_4 lift leverage conviction Out[30]: antecedents consequents antecedent support consequent support support confidence (Gladiator) (Patriot) 0.7 0.6 0.857143 1.428571 0.18 0.6 0.7 1.000000 1.428571 0.18 inf (Patriot) (Gladiator) (Sixth Sense) (Gladiator) 0.6 0.7 0.833333 1.190476 0.08 1.8 0.7 0.6 0.714286 1.190476 0.08 1.4 (Gladiator) (Sixth Sense) 0.5 Visualization for 0.5 support : plt.figure(figsize = (8,6)) plt.scatter(a_4["support"], a_4["confidence"]) plt.title("Association Rules Plot", size = 20, color = "black") plt.xlabel("Support", size = 12) plt.ylabel("Confidence", size = 12) plt.show() **Association Rules Plot** 1.00 0.95 0.90 Confidence 0.85 0.80 0.75 0.52 0.54 0.58 0.50 0.56 0.60 Support Observation: a) For 0.1 value of support, 246 rules generated out of which 236 are actionable rules. b) For 0.2 value of support we got 16 actionable rules. c) For 0.3 value of support we got 12 actionable rules. d) For 0.5 value of support we got 4 actionable rules. Higher the Support value, lower the number of rules.