In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from mlxtend.frequent_patterns import apriori,association_rules
from mlxtend.preprocessing import TransactionEncoder
```

In [2]:
```
data=pd.read_csv("my_movies.csv")
data.head()
```

Out[2]:

	V1	V2	V3	V4	V5	Sixth Sense	Gladiator	LOTR1	Harry Potter1	Patriot
0	Sixth Sense	LOTR1	Harry Potter1	Green Mile	LOTR2	1	0	1	1	0
1	Gladiator	Patriot	Braveheart	NaN	NaN	0	1	0	0	1
2	LOTR1	LOTR2	NaN	NaN	NaN	0	0	1	0	0
3	Gladiator	Patriot	Sixth Sense	NaN	NaN	1	1	0	0	1
4	Gladiator	Patriot	Sixth Sense	NaN	NaN	1	1	0	0	1

In [3]:
```
data.describe()
```

Out[3]:

	Sixth Sense	Gladiator	LOTR1	Harry Potter1	Patriot	LOTR2	Harry Potter2	L(
count	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.00(
mean	0.600000	0.700000	0.200000	0.200000	0.600000	0.200000	0.100000	0.10(
std	0.516398	0.483046	0.421637	0.421637	0.516398	0.421637	0.316228	0.31(
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00(
25%	0.000000	0.250000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00(
50%	1.000000	1.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.00(
75%	1.000000	1.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.00(
max	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.00(

In [4]:
```
data.shape
```

Out[4]: (10, 15)

In [5]:
```
data.isnull().sum()
```

Out[5]:
```
V1              0
V2              0
V3              3
V4              8
V5              9
Sixth Sense     0
Gladiator       0
LOTR1           0
Harry Potter1   0
Patriot         0
LOTR2           0
Harry Potter2   0
LOTR            0
Braveheart      0
Green Mile      0
dtype: int64
```

In [6]:
```
data.dtypes
```

Out[6]:
```
V1              object
V2              object
V3              object
V4              object
V5              object
Sixth Sense      int64
Gladiator        int64
LOTR1            int64
Harry Potter1    int64
Patriot          int64
LOTR2            int64
Harry Potter2    int64
LOTR             int64
Braveheart       int64
Green Mile       int64
dtype: object
```

In [8]:
```
data1=data.iloc[:,5:]
data1
```

Out[8]:

	Sixth Sense	Gladiator	LOTR1	Harry Potter1	Patriot	LOTR2	Harry Potter2	LOTR	Braveheart	Green Mile
0	1	0	1	1	0	1	0	0	0	1
1	0	1	0	0	1	0	0	0	1	0
2	0	0	1	0	0	1	0	0	0	0
3	1	1	0	0	1	0	0	0	0	0
4	1	1	0	0	1	0	0	0	0	0
5	1	1	0	0	1	0	0	0	0	0
6	0	0	0	1	0	0	1	0	0	0
7	0	1	0	0	1	0	0	0	0	0
8	1	1	0	0	1	0	0	0	0	0
9	1	1	0	0	0	0	0	1	0	1

In [9]:
```
data1[data1.duplicated()]
```

Out[9]:

	Sixth Sense	Gladiator	LOTR1	Harry Potter1	Patriot	LOTR2	Harry Potter2	LOTR	Braveheart	Green Mile
4	1	1	0	0	1	0	0	0	0	0
5	1	1	0	0	1	0	0	0	0	0
8	1	1	0	0	1	0	0	0	0	0

In [10]:
```
data2=data1.drop_duplicates()
data2
```

Out[10]:

	Sixth Sense	Gladiator	LOTR1	Harry Potter1	Patriot	LOTR2	Harry Potter2	LOTR	Braveheart	Green Mile
0	1	0	1	1	0	1	0	0	0	1
1	0	1	0	0	1	0	0	0	1	0
2	0	0	1	0	0	1	0	0	0	0
3	1	1	0	0	1	0	0	0	0	0
6	0	0	0	1	0	0	1	0	0	0
7	0	1	0	0	1	0	0	0	0	0
9	1	1	0	0	0	0	0	1	0	1

In [11]:
```
data2.isnull().sum()
```

Out[11]:
```
Sixth Sense      0
Gladiator        0
LOTR1            0
Harry Potter1    0
Patriot          0
LOTR2            0
Harry Potter2    0
LOTR             0
Braveheart       0
Green Mile       0
dtype: int64
```

In [12]:
```
frequent_itemsets = apriori(data2,min_support = 0.1,use_colnames=Tr
frequent_itemsets
```

```
/opt/anaconda3/lib/python3.9/site-packages/mlxtend/frequent_patter
ns/fpcommon.py:111: DeprecationWarning: DataFrames with non-bool t
ypes result in worse computationalperformance and their support mi
ght be discontinued in the future.Please use a DataFrame with bool
type
  warnings.warn(
```

Out[12]:

	support	itemsets
0	0.428571	(Sixth Sense)
1	0.571429	(Gladiator)
2	0.285714	(LOTR1)
3	0.285714	(Harry Potter1)
4	0.428571	(Patriot)
5	0.285714	(LOTR2)
6	0.142857	(Harry Potter2)
7	0.142857	(LOTR)
8	0.142857	(Braveheart)
9	0.285714	(Green Mile)
10	0.285714	(Gladiator, Sixth Sense)
11	0.142857	(LOTR1, Sixth Sense)
12	0.142857	(Harry Potter1, Sixth Sense)
13	0.142857	(Patriot, Sixth Sense)
14	0.142857	(Sixth Sense, LOTR2)
15	0.142857	(Sixth Sense, LOTR)
16	0.285714	(Sixth Sense, Green Mile)

17	0.428571	(Patriot, Gladiator)
18	0.142857	(Gladiator, LOTR)
19	0.142857	(Gladiator, Braveheart)
20	0.142857	(Gladiator, Green Mile)
21	0.142857	(LOTR1, Harry Potter1)
22	0.285714	(LOTR1, LOTR2)
23	0.142857	(LOTR1, Green Mile)
24	0.142857	(Harry Potter1, LOTR2)
25	0.142857	(Harry Potter1, Harry Potter2)
26	0.142857	(Harry Potter1, Green Mile)
27	0.142857	(Patriot, Braveheart)
28	0.142857	(LOTR2, Green Mile)
29	0.142857	(LOTR, Green Mile)
30	0.142857	(Patriot, Gladiator, Sixth Sense)
31	0.142857	(Gladiator, Sixth Sense, LOTR)
32	0.142857	(Gladiator, Sixth Sense, Green Mile)
33	0.142857	(LOTR1, Harry Potter1, Sixth Sense)
34	0.142857	(LOTR1, Sixth Sense, LOTR2)
35	0.142857	(LOTR1, Sixth Sense, Green Mile)
36	0.142857	(Harry Potter1, Sixth Sense, LOTR2)
37	0.142857	(Harry Potter1, Sixth Sense, Green Mile)
38	0.142857	(LOTR2, Sixth Sense, Green Mile)
39	0.142857	(Sixth Sense, LOTR, Green Mile)
40	0.142857	(Gladiator, Braveheart, Patriot)
41	0.142857	(Gladiator, LOTR, Green Mile)
42	0.142857	(LOTR1, Harry Potter1, LOTR2)
43	0.142857	(LOTR1, Harry Potter1, Green Mile)
44	0.142857	(LOTR1, LOTR2, Green Mile)
45	0.142857	(LOTR2, Harry Potter1, Green Mile)
46	0.142857	(Gladiator, Sixth Sense, LOTR, Green Mile)
47	0.142857	(LOTR1, Harry Potter1, Sixth Sense, LOTR2)
48	0.142857	(LOTR1, Harry Potter1, Sixth Sense, Green Mile)
49	0.142857	(LOTR1, LOTR2, Sixth Sense, Green Mile)
50	0.142857	(LOTR2, Harry Potter1, Sixth Sense, Green Mile)

51	0.142857	(LOTR1, LOTR2, Harry Potter1, Green Mile)
52	0.142857	(LOTR1, LOTR2, Harry Potter1, Sixth Sense, Gre...

In [13]:
```
rules = association_rules(frequent_itemsets,metric='lift',min_thres
rules
```

Out[13]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	l
0	(Gladiator)	(Sixth Sense)	0.571429	0.428571	0.285714	0.500000	1.166667	(
1	(Sixth Sense)	(Gladiator)	0.428571	0.571429	0.285714	0.666667	1.166667	(
2	(LOTR1)	(Sixth Sense)	0.285714	0.428571	0.142857	0.500000	1.166667	(
3	(Sixth Sense)	(LOTR1)	0.428571	0.285714	0.142857	0.333333	1.166667	(
4	(Harry Potter1)	(Sixth Sense)	0.285714	0.428571	0.142857	0.500000	1.166667	(
...	...	...	...	...	...	...	...	
245	(LOTR1)	(Green Mile, Harry Potter1, Sixth Sense, LOTR2)	0.285714	0.142857	0.142857	0.500000	3.500000	(
246	(LOTR2)	(LOTR1, Harry Potter1, Sixth Sense, Green Mile)	0.285714	0.142857	0.142857	0.500000	3.500000	(
247	(Harry Potter1)	(LOTR1, Green Mile, Sixth Sense, LOTR2)	0.285714	0.142857	0.142857	0.500000	3.500000	(
248	(Sixth Sense)	(LOTR1, Green Mile, Harry Potter1, LOTR2)	0.428571	0.142857	0.142857	0.333333	2.333333	(
249	(Green Mile)	(LOTR1, Harry Potter1, Sixth Sense, LOTR2)	0.285714	0.142857	0.142857	0.500000	3.500000	(

250 rows × 9 columns

In [14]:
```
rules1=rules.sort_values('lift',ascending = False)[0:20]
```

In [15]:
```
rules[rules.lift>1]
```

Out[15]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	l
0	(Gladiator)	(Sixth Sense)	0.571429	0.428571	0.285714	0.500000	1.166667	(
1	(Sixth Sense)	(Gladiator)	0.428571	0.571429	0.285714	0.666667	1.166667	(
2	(LOTR1)	(Sixth Sense)	0.285714	0.428571	0.142857	0.500000	1.166667	(
3	(Sixth Sense)	(LOTR1)	0.428571	0.285714	0.142857	0.333333	1.166667	(
4	(Harry Potter1)	(Sixth Sense)	0.285714	0.428571	0.142857	0.500000	1.166667	(
...	...	...	...	...	...	...	...	
245	(LOTR1)	(Green Mile, Harry Potter1, Sixth Sense, LOTR2)	0.285714	0.142857	0.142857	0.500000	3.500000	(
246	(LOTR2)	(LOTR1, Harry Potter1, Sixth Sense, Green Mile)	0.285714	0.142857	0.142857	0.500000	3.500000	(
247	(Harry Potter1)	(LOTR1, Green Mile, Sixth Sense, LOTR2)	0.285714	0.142857	0.142857	0.500000	3.500000	(
248	(Sixth Sense)	(LOTR1, Green Mile, Harry Potter1, LOTR2)	0.428571	0.142857	0.142857	0.333333	2.333333	(
249	(Green Mile)	(LOTR1, Harry Potter1, Sixth Sense, LOTR2)	0.285714	0.142857	0.142857	0.500000	3.500000	(

242 rows × 9 columns

In [16]:
```
rules3 =rules[(rules['lift']>1)&(rules['confidence']>0.7)]
rules3
```

Out[16]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	l
11	(LOTR)	(Sixth Sense)	0.142857	0.428571	0.142857	1.00	2.333333	(
13	(Green Mile)	(Sixth Sense)	0.285714	0.428571	0.285714	1.00	2.333333	(
14	(Patriot)	(Gladiator)	0.428571	0.571429	0.428571	1.00	1.750000	(
15	(Gladiator)	(Patriot)	0.571429	0.428571	0.428571	0.75	1.750000	(
17	(LOTR)	(Gladiator)	0.142857	0.571429	0.142857	1.00	1.750000	(
...	...	...	...	...	...	...	...	
239	(Harry Potter1, LOTR2)	(LOTR1, Sixth Sense, Green Mile)	0.142857	0.142857	0.142857	1.00	7.000000	(
240	(Sixth Sense, LOTR2)	(LOTR1, Harry Potter1, Green Mile)	0.142857	0.142857	0.142857	1.00	7.000000	(
241	(Green Mile, LOTR2)	(LOTR1, Harry Potter1, Sixth Sense)	0.142857	0.142857	0.142857	1.00	7.000000	(
242	(Harry Potter1, Sixth Sense)	(LOTR1, Green Mile, LOTR2)	0.142857	0.142857	0.142857	1.00	7.000000	(
243	(Harry Potter1, Green Mile)	(LOTR1, Sixth Sense, LOTR2)	0.142857	0.142857	0.142857	1.00	7.000000	(

126 rows × 9 columns

In [17]:
```
freq2_mov=apriori(data2,min_support=0.2,use_colnames=True)
freq2_mov
```

```
/opt/anaconda3/lib/python3.9/site-packages/mlxtend/frequent_patter
ns/fpcommon.py:111: DeprecationWarning: DataFrames with non-bool t
ypes result in worse computationalperformance and their support mi
ght be discontinued in the future.Please use a DataFrame with bool
type
  warnings.warn(
```

Out[17]:

	support	itemsets
0	0.428571	(Sixth Sense)
1	0.571429	(Gladiator)
2	0.285714	(LOTR1)
3	0.285714	(Harry Potter1)
4	0.428571	(Patriot)
5	0.285714	(LOTR2)
6	0.285714	(Green Mile)
7	0.285714	(Gladiator, Sixth Sense)
8	0.285714	(Sixth Sense, Green Mile)
9	0.428571	(Patriot, Gladiator)
10	0.285714	(LOTR1, LOTR2)

In [18]:
```
rules4=association_rules(freq2_mov,metric='confidence',min_threshol
rules4
```

Out[18]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	lev
0	(Sixth Sense)	(Gladiator)	0.428571	0.571429	0.285714	0.666667	1.166667	0.0
1	(Sixth Sense)	(Green Mile)	0.428571	0.285714	0.285714	0.666667	2.333333	0.1
2	(Green Mile)	(Sixth Sense)	0.285714	0.428571	0.285714	1.000000	2.333333	0.1
3	(Patriot)	(Gladiator)	0.428571	0.571429	0.428571	1.000000	1.750000	0.1
4	(Gladiator)	(Patriot)	0.571429	0.428571	0.428571	0.750000	1.750000	0.1
5	(LOTR1)	(LOTR2)	0.285714	0.285714	0.285714	1.000000	3.500000	0.2
6	(LOTR2)	(LOTR1)	0.285714	0.285714	0.285714	1.000000	3.500000	0.2

In [19]:
```
rules4[rules4.lift>1]
```

Out[19]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	lev
0	(Sixth Sense)	(Gladiator)	0.428571	0.571429	0.285714	0.666667	1.166667	0.0
1	(Sixth Sense)	(Green Mile)	0.428571	0.285714	0.285714	0.666667	2.333333	0.1
2	(Green Mile)	(Sixth Sense)	0.285714	0.428571	0.285714	1.000000	2.333333	0.1
3	(Patriot)	(Gladiator)	0.428571	0.571429	0.428571	1.000000	1.750000	0.1
4	(Gladiator)	(Patriot)	0.571429	0.428571	0.428571	0.750000	1.750000	0.1
5	(LOTR1)	(LOTR2)	0.285714	0.285714	0.285714	1.000000	3.500000	0.2
6	(LOTR2)	(LOTR1)	0.285714	0.285714	0.285714	1.000000	3.500000	0.2

Conclusion

Lower the Confidence level Higher the no. of rules. Higher the Support, lower the no. of rules.

In [ ]:
```

```