In [2]:
```python
import pandas as pd
from mlxtend.frequent_patterns import apriori,association_rules
from mlxtend.preprocessing import TransactionEncoder
import matplotlib.pyplot as plt
import seaborn as sns
```

In [3]:
```python
mov = pd.read_csv('/Users/SAURABH/Saurabh patil/DATA SCIENCE/Association Rules/my_movies.csv',)
```

In [4]:
```python
mov
```

Out[4]:

	V1	V2	V3	V4	V5	Sixth Sense	Gladiator	LOTR1	Harry Potter1	Patriot	LOTR2	Harry Potter2	LOTR	Braveheart	Green Mile
0	Sixth Sense	LOTR1	Harry Potter1	Green Mile	LOTR2	1	0	1	1	0	1	0	0	0	1
1	Gladiator	Patriot	Braveheart	NaN	NaN	0	1	0	0	1	0	0	0	1	0
2	LOTR1	LOTR2	NaN	NaN	NaN	0	0	1	0	0	1	0	0	0	0
3	Gladiator	Patriot	Sixth Sense	NaN	NaN	1	1	0	0	1	0	0	0	0	0
4	Gladiator	Patriot	Sixth Sense	NaN	NaN	1	1	0	0	1	0	0	0	0	0
5	Gladiator	Patriot	Sixth Sense	NaN	NaN	1	1	0	0	1	0	0	0	0	0
6	Harry Potter1	Harry Potter2	NaN	NaN	NaN	0	0	0	1	0	0	1	0	0	0
7	Gladiator	Patriot	NaN	NaN	NaN	0	1	0	0	1	0	0	0	0	0
8	Gladiator	Patriot	Sixth Sense	NaN	NaN	1	1	0	0	1	0	0	0	0	0
9	Sixth Sense	LOTR	Gladiator	Green Mile	NaN	1	1	0	0	0	0	0	1	0	1

In [5]:
```python
mov.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 15 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   V1             10 non-null     object
 1   V2             10 non-null     object
 2   V3             7 non-null      object
 3   V4             2 non-null      object
 4   V5             1 non-null      object
 5   Sixth Sense    10 non-null     int64
 6   Gladiator      10 non-null     int64
 7   LOTR1          10 non-null     int64
 8   Harry Potter1  10 non-null     int64
 9   Patriot        10 non-null     int64
 10  LOTR2          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: 1.0+ KB
```

In [6]:
```python
mov_1 = mov.iloc[:,0:5]
```

In [7]:
```python
mov_1
```

Out[7]:

	V1	V2	V3	V4	V5
0	Sixth Sense	LOTR1	Harry Potter1	Green Mile	LOTR2
1	Gladiator	Patriot	Braveheart	NaN	NaN
2	LOTR1	LOTR2	NaN	NaN	NaN
3	Gladiator	Patriot	Sixth Sense	NaN	NaN
4	Gladiator	Patriot	Sixth Sense	NaN	NaN
5	Gladiator	Patriot	Sixth Sense	NaN	NaN
6	Harry Potter1	Harry Potter2	NaN	NaN	NaN
7	Gladiator	Patriot	NaN	NaN	NaN
8	Gladiator	Patriot	Sixth Sense	NaN	NaN
9	Sixth Sense	LOTR	Gladiator	Green Mile	NaN

In [8]:
```python
df1 = pd.get_dummies(mov_1)
```

In [9]:
```python
df1
```

Out[9]:

	V1_Gladiator	V1_Harry Potter1	V1_LOTR1	V1_Sixth Sense	V2_Harry Potter2	V2_LOTR	V2_LOTR1	V2_LOTR2	V2_Patriot	V3_Braveheart	V3_Gladiator	V3_Harry Potter1	V3_Sixth Sense	V4_Green Mile	V5_LOTR2
0	0	0	0	1	0	0	1	0	0	0	0	1	0	1	1
1	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0
2	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
3	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0
4	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0
5	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0
6	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0
7	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0
8	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0
9	0	0	0	1	0	1	0	0	0	0	1	0	0	1	0

In [10]:
```python
mov_2 = mov.iloc[:,5:15]
```

In [11]:
```python
mov_2
```

Out[11]:

	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

Apriori Algorithm

Keeping Min Support for Apriori as 10%

In [12]:
```python
freq_items1 = apriori(mov_2, min_support=0.10, use_colnames=True)
freq_items1
```

Out[12]:

	support	itemsets
0	0.6	(Sixth Sense)
1	0.7	(Gladiator)
2	0.2	(LOTR1)
3	0.2	(Harry Potter1)
4	0.6	(Patriot)
5	0.2	(LOTR2)
6	0.1	(Harry Potter2)
7	0.1	(LOTR)
8	0.1	(Braveheart)
9	0.2	(Green Mile)
10	0.5	(Sixth Sense, Gladiator)
11	0.1	(LOTR1, Sixth Sense)
12	0.1	(Harry Potter1, Sixth Sense)
13	0.4	(Patriot, Sixth Sense)
14	0.1	(LOTR2, Sixth Sense)
15	0.1	(Sixth Sense, LOTR)
16	0.2	(Green Mile, Sixth Sense)
17	0.6	(Patriot, Gladiator)
18	0.1	(LOTR, Gladiator)
19	0.1	(Braveheart, Gladiator)
20	0.1	(Green Mile, Gladiator)
21	0.1	(LOTR1, Harry Potter1)
22	0.2	(LOTR1, LOTR2)
23	0.1	(LOTR1, Green Mile)
24	0.1	(Harry Potter1, LOTR2)
25	0.1	(Harry Potter1, Harry Potter2)
26	0.1	(Green Mile, Harry Potter1)
27	0.1	(Braveheart, Patriot)
28	0.1	(Green Mile, LOTR2)
29	0.1	(Green Mile, LOTR)
30	0.4	(Patriot, Sixth Sense, Gladiator)
31	0.1	(LOTR, Sixth Sense, Gladiator)
32	0.1	(Green Mile, Sixth Sense, Gladiator)
33	0.1	(LOTR1, Harry Potter1, Sixth Sense)
34	0.1	(LOTR1, LOTR2, Sixth Sense)
35	0.1	(LOTR1, Green Mile, Sixth Sense)
36	0.1	(LOTR2, Harry Potter1, Sixth Sense)
37	0.1	(Green Mile, Sixth Sense, Harry Potter1)
38	0.1	(LOTR2, Green Mile, Sixth Sense)
39	0.1	(Green Mile, Sixth Sense, LOTR)
40	0.1	(Braveheart, Patriot, Gladiator)
41	0.1	(LOTR, Green Mile, Gladiator)
42	0.1	(LOTR1, Harry Potter1, LOTR2)
43	0.1	(LOTR1, Green Mile, Harry Potter1)
44	0.1	(LOTR1, Green Mile, LOTR2)
45	0.1	(Green Mile, LOTR2, Harry Potter1)
46	0.1	(LOTR, Green Mile, Sixth Sense, Gladiator)
47	0.1	(LOTR1, LOTR2, Harry Potter1, Sixth Sense)
48	0.1	(LOTR1, Green Mile, Sixth Sense, Harry Potter1)
49	0.1	(LOTR1, LOTR2, Green Mile, Sixth Sense)
50	0.1	(LOTR2, Green Mile, Sixth Sense, Harry Potter1)
51	0.1	(LOTR1, Green Mile, LOTR2, Harry Potter1)
52	0.1	(LOTR1, LOTR2, Sixth Sense, Harry Potter1, Gre...

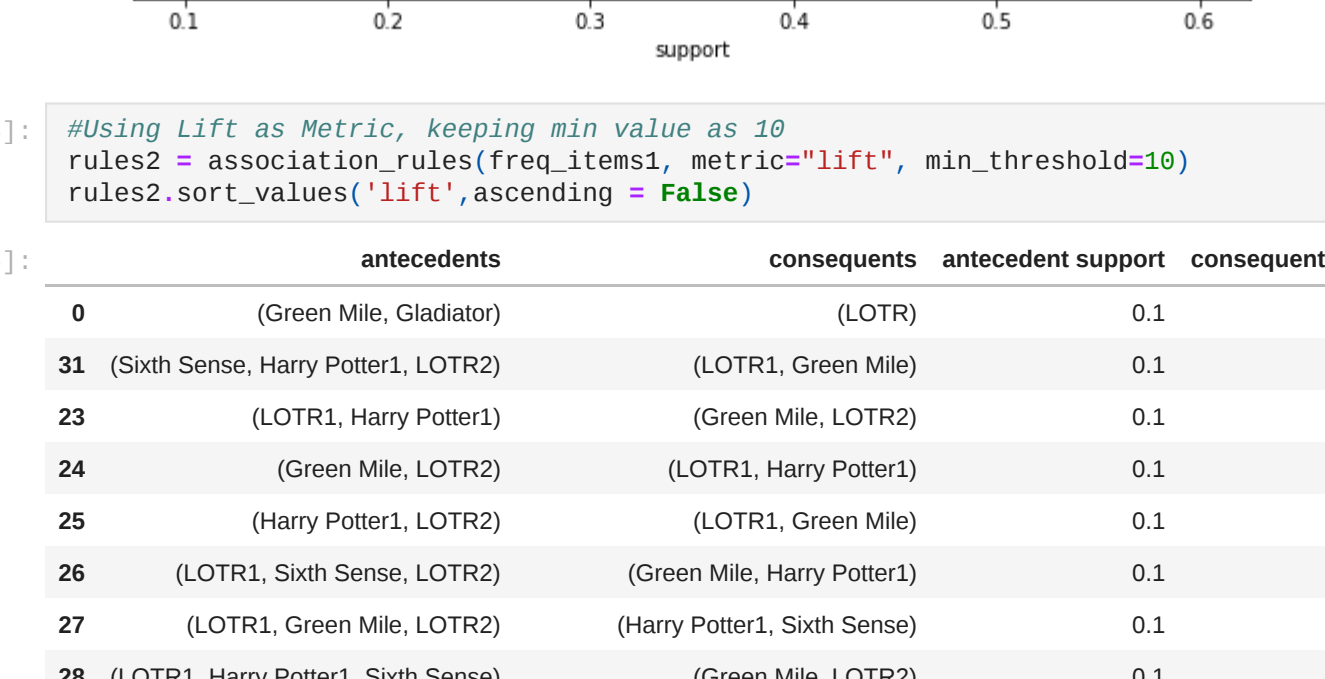
In [13]:
```python
#Using Lift as Metric, keeping min value as 1
rules1 = association_rules(freq_items1, metric="lift", min_threshold=1)
rules1.sort_values('lift',ascending = False)
```

Out[13]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
161	(Harry Potter1, Sixth Sense)	(LOTR1, Green Mile)	0.1	0.1	0.1	1.000000	10.000000	0.09	inf
201	(Green Mile, LOTR2)	(LOTR1, Harry Potter1)	0.1	0.1	0.1	1.000000	10.000000	0.09	inf
184	(Green Mile, LOTR2)	(Harry Potter1, Sixth Sense)	0.1	0.1	0.1	1.000000	10.000000	0.09	inf
173	(Green Mile, LOTR2)	(LOTR1, Sixth Sense)	0.1	0.1	0.1	1.000000	10.000000	0.09	inf
174	(Sixth Sense, LOTR2)	(LOTR1, Green Mile)	0.1	0.1	0.1	1.000000	10.000000	0.09	inf
...	...	...	...	...	...	...	...	...	...
36	(Sixth Sense)	(Patriot, Gladiator)	0.6	0.6	0.4	0.666667	1.111111	0.04	1.2
3	(Sixth Sense)	(Patriot)	0.6	0.6	0.4	0.666667	1.111111	0.04	1.2
2	(Patriot)	(Sixth Sense)	0.6	0.6	0.4	0.666667	1.111111	0.04	1.2
46	(Green Mile)	(Sixth Sense, Gladiator)	0.2	0.5	0.1	0.500000	1.000000	0.00	1.0
45	(Sixth Sense, Gladiator)	(Green Mile)	0.5	0.2	0.1	0.200000	1.000000	0.00	1.0

238 rows × 9 columns

In [14]:
```python
plt.figure(figsize=(10,5))
plt.scatter(rules1['support'], rules1['confidence'], alpha=0.5)
plt.xlabel('support')
plt.ylabel('confidence')
plt.title('Support vs Confidence')
plt.show()
```



In [15]:
```python
#Using Lift as Metric, keeping min value as 10
rules2 = association_rules(freq_items1, metric="lift", min_threshold=10)
rules2.sort_values('lift',ascending = False)
```

Out[15]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(Green Mile, Gladiator)	(LOTR)	0.1	0.1	0.1	1.0	10.0	0.09	inf
31	(Sixth Sense, Harry Potter1, LOTR2)	(LOTR1, Green Mile)	0.1	0.1	0.1	1.0	10.0	0.09	inf
23	(LOTR1, Harry Potter1)	(Green Mile, LOTR2)	0.1	0.1	0.1	1.0	10.0	0.09	inf
24	(Green Mile, LOTR2)	(LOTR1, Harry Potter1)	0.1	0.1	0.1	1.0	10.0	0.09	inf
25	(Harry Potter1, LOTR2)	(LOTR1, Green Mile)	0.1	0.1	0.1	1.0	10.0	0.09	inf
26	(LOTR1, Sixth Sense, LOTR2)	(Green Mile, Harry Potter1)	0.1	0.1	0.1	1.0	10.0	0.09	inf
27	(LOTR1, Green Mile, LOTR2)	(Harry Potter1, Sixth Sense)	0.1	0.1	0.1	1.0	10.0	0.09	inf
28	(LOTR1, Harry Potter1, Sixth Sense)	(Green Mile, LOTR2)	0.1	0.1	0.1	1.0	10.0	0.09	inf
29	(LOTR1, Green Mile, Sixth Sense)	(Harry Potter1, LOTR2)	0.1	0.1	0.1	1.0	10.0	0.09	inf
30	(LOTR1, Harry Potter1, Green Mile)	(Sixth Sense, LOTR2)	0.1	0.1	0.1	1.0	10.0	0.09	inf
32	(Sixth Sense, Green Mile, LOTR2)	(LOTR1, Harry Potter1)	0.1	0.1	0.1	1.0	10.0	0.09	inf
1	(LOTR)	(Green Mile, Gladiator)	0.1	0.1	0.1	1.0	10.0	0.09	inf
33	(Green Mile, Harry Potter1, LOTR2)	(LOTR1, Sixth Sense)	0.1	0.1	0.1	1.0	10.0	0.09	inf
34	(LOTR1, Sixth Sense)	(Green Mile, Harry Potter1, LOTR2)	0.1	0.1	0.1	1.0	10.0	0.09	inf
35	(LOTR1, Harry Potter1)	(Sixth Sense, Green Mile, LOTR2)	0.1	0.1	0.1	1.0	10.0	0.09	inf
36	(LOTR1, Green Mile)	(Sixth Sense, Harry Potter1, LOTR2)	0.1	0.1	0.1	1.0	10.0	0.09	inf
37	(Sixth Sense, LOTR2)	(LOTR1, Harry Potter1, Green Mile)	0.1	0.1	0.1	1.0	10.0	0.09	inf
38	(Harry Potter1, LOTR2)	(LOTR1, Green Mile, Sixth Sense)	0.1	0.1	0.1	1.0	10.0	0.09	inf
39	(Green Mile, LOTR2)	(LOTR1, Harry Potter1, Sixth Sense)	0.1	0.1	0.1	1.0	10.0	0.09	inf
40	(Harry Potter1, Sixth Sense)	(LOTR1, Green Mile, LOTR2)	0.1	0.1	0.1	1.0	10.0	0.09	inf
22	(LOTR1, Green Mile)	(Harry Potter1, LOTR2)	0.1	0.1	0.1	1.0	10.0	0.09	inf
21	(Harry Potter1, Sixth Sense)	(Green Mile, LOTR2)	0.1	0.1	0.1	1.0	10.0	0.09	inf
20	(Green Mile, Harry Potter1)	(Sixth Sense, LOTR2)	0.1	0.1	0.1	1.0	10.0	0.09	inf
19	(Sixth Sense, LOTR2)	(Green Mile, Harry Potter1)	0.1	0.1	0.1	1.0	10.0	0.09	inf
2	(Green Mile, Sixth Sense, Gladiator)	(LOTR)	0.1	0.1	0.1	1.0	10.0	0.09	inf
3	(Sixth Sense, LOTR)	(Green Mile, Gladiator)	0.1	0.1	0.1	1.0	10.0	0.09	inf
4	(Green Mile, Gladiator)	(Sixth Sense, LOTR)	0.1	0.1	0.1	1.0	10.0	0.09	inf
5	(LOTR)	(Green Mile, Sixth Sense, Gladiator)	0.1	0.1	0.1	1.0	10.0	0.09	inf
6	(LOTR1, Harry Potter1)	(Sixth Sense, LOTR2)	0.1	0.1	0.1	1.0	10.0	0.09	inf
7	(LOTR1, Sixth Sense)	(Harry Potter1, LOTR2)	0.1	0.1	0.1	1.0	10.0	0.09	inf
8	(Harry Potter1, LOTR2)	(LOTR1, Sixth Sense)	0.1	0.1	0.1	1.0	10.0	0.09	inf
9	(Sixth Sense, LOTR2)	(LOTR1, Harry Potter1)	0.1	0.1	0.1	1.0	10.0	0.09	inf
10	(LOTR1, Green Mile)	(Harry Potter1, Sixth Sense)	0.1	0.1	0.1	1.0	10.0	0.09	inf
11	(LOTR1, Sixth Sense)	(Green Mile, Harry Potter1)	0.1	0.1	0.1	1.0	10.0	0.09	inf
12	(Green Mile, Harry Potter1)	(LOTR1, Sixth Sense)	0.1	0.1	0.1	1.0	10.0	0.09	inf
13	(Harry Potter1, Sixth Sense)	(LOTR1, Green Mile)	0.1	0.1	0.1	1.0	10.0	0.09	inf
14	(LOTR1, Green Mile)	(Sixth Sense, LOTR2)	0.1	0.1	0.1	1.0	10.0	0.09	inf
15	(LOTR1, Sixth Sense)	(Green Mile, LOTR2)	0.1	0.1	0.1	1.0	10.0	0.09	inf
16	(Green Mile, LOTR2)	(LOTR1, Sixth Sense)	0.1	0.1	0.1	1.0	10.0	0.09	inf
17	(Sixth Sense, LOTR2)	(LOTR1, Green Mile)	0.1	0.1	0.1	1.0	10.0	0.09	inf
18	(Green Mile, LOTR2)	(Harry Potter1, Sixth Sense)	0.1	0.1	0.1	1.0	10.0	0.09	inf
41	(Green Mile, LOTR2)	(LOTR1, Harry Potter1, Sixth Sense)	0.1	0.1	0.1	1.0	10.0	0.09	inf

In [16]:
```python
#Using Confidence as Metric, keeping min value as 75%
rules3 = association_rules(freq_items1, metric="confidence", min_threshold=0.75)
rules3.sort_values('lift',ascending = False)
```

Out[16]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
127	(Green Mile, Harry Potter1)	(LOTR1, Sixth Sense, LOTR2)	0.1	0.1	0.1	1.000000	10.000000	0.09	inf
102	(Green Mile, LOTR2)	(LOTR1, Harry Potter1)	0.1	0.1	0.1	1.000000	10.000000	0.09	inf
100	(LOTR1, Green Mile)	(Harry Potter1, LOTR2)	0.1	0.1	0.1	1.000000	10.000000	0.09	inf
65	(LOTR1, Harry Potter1)	(Sixth Sense, LOTR2)	0.1	0.1	0.1	1.000000	10.000000	0.09	inf
94	(Green Mile, Harry Potter1)	(Sixth Sense, LOTR2)	0.1	0.1	0.1	1.000000	10.000000	0.09	inf
...	...	...	...	...	...	...	...	...	...
6	(Braveheart)	(Gladiator)	0.1	0.7	0.1	1.000000	1.428571	0.03	inf
5	(LOTR)	(Gladiator)	0.1	0.7	0.1	1.000000	1.428571	0.03	inf
3	(Patriot)	(Gladiator)	0.6	0.7	0.6	1.000000	1.428571	0.18	inf
13	(Sixth Sense, Gladiator)	(Patriot)	0.5	0.6	0.4	0.800000	1.333333	0.10	2.0
0	(Sixth Sense)	(Gladiator)	0.6	0.7	0.5	0.833333	1.190476	0.08	1.8

128 rows × 9 columns

In [17]:
```python
#Using Confidence as Metric, keeping min value as 100%
rules4 = association_rules(freq_items1, metric="confidence", min_threshold=1)
rules4.sort_values('lift',ascending = False)
```

Out[17]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
62	(LOTR1, Harry Potter1)	(Sixth Sense, LOTR2)	0.1	0.1	0.1	1.0	10.000000	0.09	inf
75	(Harry Potter1, Sixth Sense)	(LOTR1, Green Mile)	0.1	0.1	0.1	1.0	10.000000	0.09	inf
91	(Green Mile, Harry Potter1)	(Sixth Sense, LOTR2)	0.1	0.1	0.1	1.0	10.000000	0.09	inf
89	(Sixth Sense, LOTR2)	(Green Mile, Harry Potter1)	0.1	0.1	0.1	1.0	10.000000	0.09	inf
88	(Green Mile, LOTR2)	(Harry Potter1, Sixth Sense)	0.1	0.1	0.1	1.0	10.000000	0.09	inf
...	...	...	...	...	...	...	...	...	...
35	(Green Mile, LOTR)	(Gladiator)	0.1	0.7	0.1	1.0	1.428571	0.03	inf
32	(Braveheart, Patriot)	(Gladiator)	0.1	0.7	0.1	1.0	1.428571	0.03	inf
4	(Braveheart)	(Gladiator)	0.1	0.7	0.1	1.0	1.428571	0.03	inf
3	(LOTR)	(Gladiator)	0.1	0.7	0.1	1.0	1.428571	0.03	inf
2	(Patriot)	(Gladiator)	0.6	0.7	0.6	1.0	1.428571	0.18	inf

125 rows × 9 columns

In [18]:
```python
#Using Supprort as Metric, keeping min value as 50%
rules5 = association_rules(freq_items1, metric="support", min_threshold=0.50)
rules5.sort_values('lift',ascending = False)
```

Out[18]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
3	(Gladiator)	(Patriot)	0.7	0.6	0.6	0.857143	1.428571	0.18	2.8
2	(Patriot)	(Gladiator)	0.6	0.7	0.6	1.000000	1.428571	0.18	inf
0	(Sixth Sense)	(Gladiator)	0.6	0.7	0.5	0.833333	1.190476	0.08	1.8
1	(Gladiator)	(Sixth Sense)	0.7	0.6	0.5	0.714286	1.190476	0.08	1.4

Keeping Min Support for Apriori as 20%

In [19]:
```python
freq_items2 = apriori(mov_2, min_support=0.20, use_colnames=True)
freq_items2
```

Out[19]:

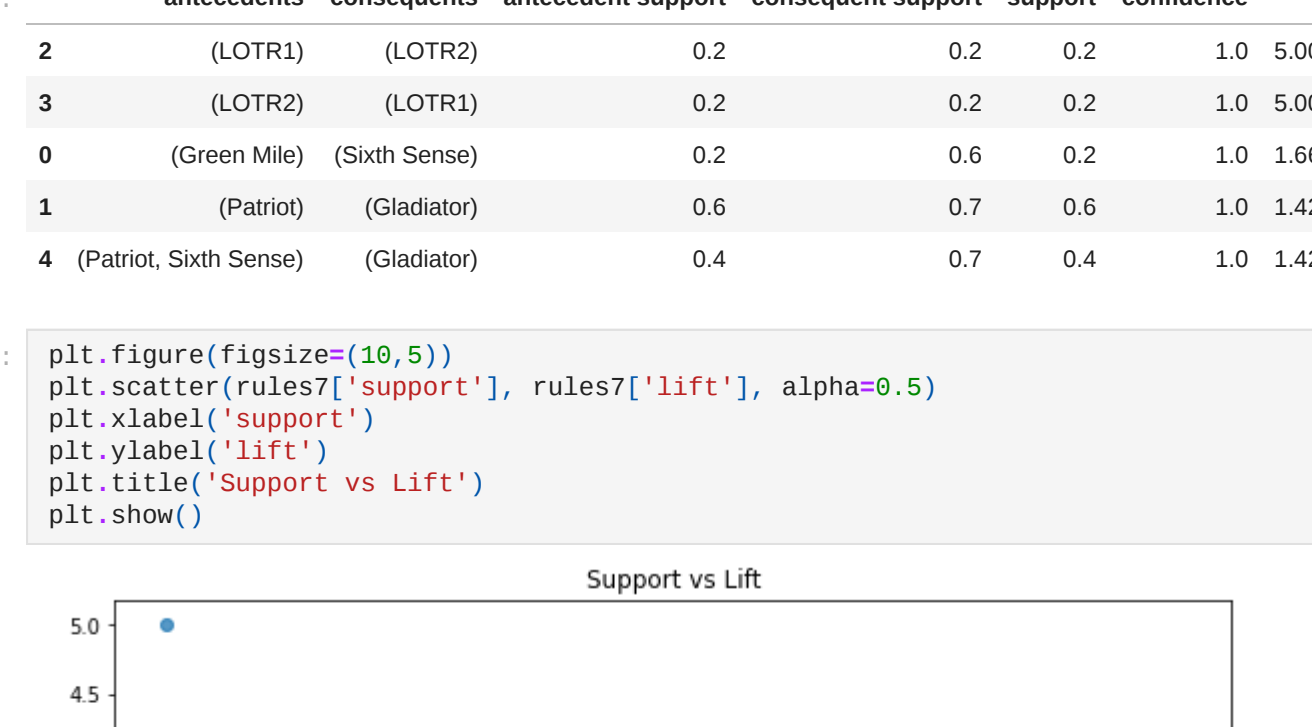
	support	itemsets
0	0.6	(Sixth Sense)
1	0.7	(Gladiator)
2	0.2	(LOTR1)
3	0.2	(Harry Potter1)
4	0.6	(Patriot)
5	0.2	(LOTR2)
6	0.2	(Green Mile)
7	0.5	(Sixth Sense, Gladiator)
8	0.4	(Patriot, Sixth Sense)
9	0.2	(Green Mile, Sixth Sense)
10	0.6	(Patriot, Gladiator)
11	0.2	(LOTR1, LOTR2)
12	0.4	(Patriot, Sixth Sense, Gladiator)

In [20]:
```python
#Using Lift as Metric, keeping min value as 1
rules6 = association_rules(freq_items2, metric="lift", min_threshold=1)
rules6.sort_values('lift',ascending = False)
```

Out[20]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
8	(LOTR1)	(LOTR2)	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
4	(Green Mile)	(Sixth Sense)	0.2	0.6	0.2	1.000000	1.666667	0.08	inf
5	(Sixth Sense)	(Green Mile)	0.6	0.2	0.2	0.333333	1.666667	0.08	1.2
7	(Gladiator)	(Patriot)	0.7	0.6	0.6	0.857143	1.428571	0.18	2.8
6	(Patriot)	(Gladiator)	0.6	0.7	0.6	1.000000	1.428571	0.18	inf
10	(Patriot, Sixth Sense)	(Gladiator)	0.4	0.7	0.4	1.000000	1.428571	0.12	inf
15	(Gladiator)	(Patriot, Sixth Sense)	0.7	0.4	0.4	0.571429	1.428571	0.12	1.4
12	(Sixth Sense, Gladiator)	(Patriot)	0.5	0.6	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	0.5	0.833333	1.190476	0.08	1.8
1	(Gladiator)	(Sixth Sense)	0.7	0.6	0.5	0.714286	1.190476	0.08	1.4
2	(Patriot)	(Sixth Sense)	0.6	0.6	0.4	0.666667	1.111111	0.04	1.2
3	(Sixth Sense)	(Patriot)	0.6	0.6	0.4	0.666667	1.111111	0.04	1.2
11	(Patriot, Gladiator)	(Sixth Sense)	0.6	0.6	0.4	0.666667	1.111111	0.04	1.2
14	(Sixth Sense)	(Patriot, Gladiator)	0.6	0.6	0.4	0.666667	1.111111	0.04	1.2

In [21]:
```python
plt.figure(figsize=(10,5))
plt.scatter(rules6['support'], rules6['confidence'], alpha=0.5)
plt.xlabel('support')
plt.ylabel('confidence')
plt.title('Support vs Confidence')
plt.show()
```

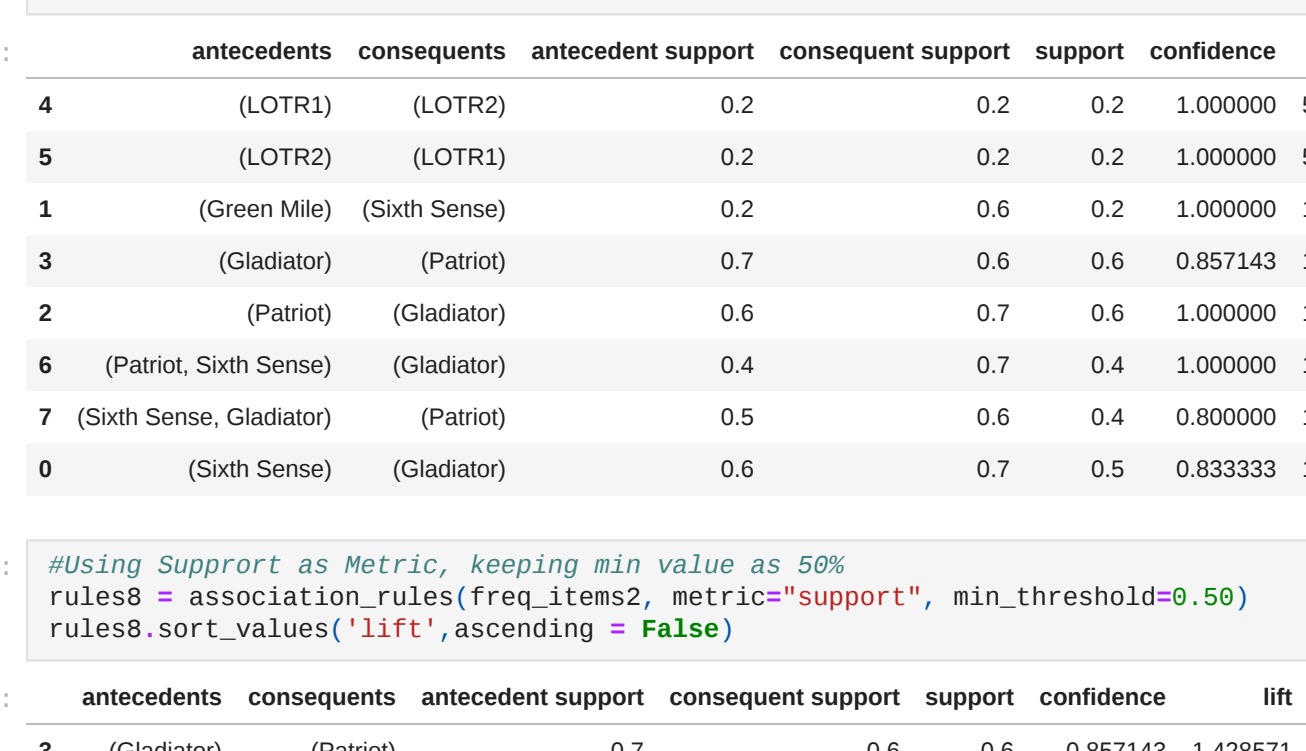


In [22]:
```python
#Using Confidence as Metric, keeping min value as 100%
rules7 = association_rules(freq_items2, metric="confidence", min_threshold=1)
rules7.sort_values('lift',ascending = False)
```

Out[22]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
2	(LOTR1)	(LOTR2)	0.2	0.2	0.2	1.0	5.000000	0.16	inf
3	(LOTR2)	(LOTR1)	0.2	0.2	0.2	1.0	5.000000	0.16	inf
0	(Green Mile)	(Sixth Sense)	0.2	0.6	0.2	1.0	1.666667	0.08	inf
1	(Patriot)	(Gladiator)	0.6	0.7	0.6	1.0	1.428571	0.18	inf
4	(Patriot, Sixth Sense)	(Gladiator)	0.4	0.7	0.4	1.0	1.428571	0.12	inf

In [23]:
```python
plt.figure(figsize=(10,5))
plt.scatter(rules7['support'], rules7['lift'], alpha=0.5)
plt.xlabel('support')
plt.ylabel('lift')
plt.title('Support vs Lift')
plt.show()
```



In [24]:
```python
#Using Confidence as Metric, keeping min value as 75%
rules7 = association_rules(freq_items2, metric="confidence", min_threshold=0.75)
rules7.sort_values('lift',ascending = False)
```

Out[24]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
4	(LOTR1)	(LOTR2)	0.2	0.2	0.2	1.000000	5.000000	0.16	inf
5	(LOTR2)	(LOTR1)	0.2	0.2	0.2	1.000000	5.000000	0.16	inf
1	(Green Mile)	(Sixth Sense)	0.2	0.6	0.2	1.000000	1.666667	0.08	inf
3	(Gladiator)	(Patriot)	0.7	0.6	0.6	0.857143	1.428571	0.18	2.8
2	(Patriot)	(Gladiator)	0.6	0.7	0.6	1.000000	1.428571	0.18	inf
6	(Patriot, Sixth Sense)	(Gladiator)	0.4	0.7	0.4	1.000000	1.428571	0.12	inf
7	(Sixth Sense, Gladiator)	(Patriot)	0.5	0.6	0.4	0.800000	1.333333	0.10	2.0
0	(Sixth Sense)	(Gladiator)	0.6	0.7	0.5	0.833333	1.190476	0.08	1.8

In [25]:
```python
#Using Supprort as Metric, keeping min value as 50%
rules8 = association_rules(freq_items2, metric="support", min_threshold=0.50)
rules8.sort_values('lift',ascending = False)
```

Out[25]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
3	(Gladiator)	(Patriot)	0.7	0.6	0.6	0.857143	1.428571	0.18	2.8
2	(Patriot)	(Gladiator)	0.6	0.7	0.6	1.000000	1.428571	0.18	inf
0	(Sixth Sense)	(Gladiator)	0.6	0.7	0.5	0.833333	1.190476	0.08	1.8
1	(Gladiator)	(Sixth Sense)	0.7	0.6	0.5	0.714286	1.190476	0.08	1.4

In [ ]: