Project 6

Part A

1) Eta - Roman Formicola, Andrew Peters, Paul Rayment

2) Rule: $(I1, I2, I3, I4, I6 \rightarrow I8)$

b)

$supp(I1 \rightarrow I6) = 2$

$conf(I1 \rightarrow I6) = supp(I1 \rightarrow I6)/supp(I1) = 2/3$

$lift(I1 \rightarrow I6) = P(I1 \cap I6)/(P(I1) * P(I6)) = 10/9 \approx 1.11$

$conviction(I1 \rightarrow I6) = (P(I1) * P(\neg I6))/P(I1\neg I6) = (3/5) * (2/5)/(1/5) = 6/5$

3)

In [10]:
```
import pandas as pd

C1= [['{M}', 3], ['{O}', 3], ['{N}', 2], ['{K}', 5], ['{E}', 4], ['{Y}', 3], ['{D}', 1], ['{I}', 1]]
df = pd.DataFrame(C1, columns=["Item Sets", "Supp"])
df.style.set_caption("C1")
```

Out[10]:

C1

	Item Sets	Supp
0	{M}	3
1	{O}	3
2	{N}	2
3	{K}	5
4	{E}	4
5	{Y}	3
6	{D}	1
7	{I}	1

In [13]:
```
L1= [['{M}', 3], ['{O}', 3],['{K}', 5], ['{E}', 4], ['{Y}', 3]]
df = pd.DataFrame(L1, columns=["Item Sets", "Supp"])
df.style.set_caption("L1")
```

Out[13]:

L1

	Item Sets	Supp
0	{M}	3
1	{O}	3
2	{K}	5
3	{E}	4
4	{Y}	3

In [14]:
```
C2= [['{MO}', 1], ['{M,K}', 3], ['{M, E}', 2], ['{M, Y}', 2], ['{O, K}', 3], ['{O, E}', 3],
     ['{O, Y}', 2], ['{K, E}', 4], ['{K, Y}', 3], ['E, Y', 2]]
df = pd.DataFrame(C2, columns=["Item Sets", "Supp"])
df.style.set_caption("C2")
```

Out[14]:

C2

	Item Sets	Supp
0	{MO}	1
1	{M,K}	3
2	{M, E}	2
3	{M, Y}	2
4	{O, K}	3
5	{O, E}	3
6	{O, Y}	2
7	{K, E}	4
8	{K, Y}	3
9	E, Y	2

In [15]:
```
L2= [['{M,K}', 3], ['{O, K}', 3], ['{O, E}', 3],
     ['{K, E}', 4], ['{K, Y}', 3]]
df = pd.DataFrame(L2, columns=["Item Sets", "Supp"])
df.style.set_caption("L2")
```

Out[15]:

L2

	Item Sets	Supp
0	{M,K}	3
1	{O, K}	3
2	{O, E}	3
3	{K, E}	4
4	{K, Y}	3

In [16]:
```
C3= [['{O, K, E}', 3]]
df = pd.DataFrame(C3, columns=["Item Sets", "Supp"])
df.style.set_caption("C3")
```

Out[16]:

C3

	Item Sets	Supp
0	{O, K, E}	3

In [17]:
```
L3= [['{O, K, E}', 3]]
df = pd.DataFrame(L3, columns=["Item Sets", "Supp"])
df.style.set_caption("L3")
```

Out[17]:

L3

	Item Sets	Supp
0	{O, K, E}	3

Frequent Itemsets = {M}, {O}, {K}, {E}, {Y}, {M, K}, {O, K}, {O, E}, {K, E}, {K, Y}, {O, K, E}

Part B

In [23]:
```
from mlxtend.preprocessing import TransactionEncoder
from mlxtend.frequent_patterns import apriori

dataset = [['M', 'O', 'N', 'K', 'E', 'Y'],
           ['D', 'O', 'N', 'K', 'E', 'Y'],
           ['M', 'A', 'K', 'E'],
           ['M', 'U', 'C', 'K', 'Y'],
           ['C', 'O', 'O', 'K', 'I', 'E']]
te = TransactionEncoder()
te_ary = te.fit(dataset).transform(dataset)
df = pd.DataFrame(te_ary, columns=te.columns_)
apriori(df, min_support=0.6, use_colnames=True)
```

Out[23]:

	support	itemsets
0	0.8	(E)
1	1.0	(K)
2	0.6	(M)
3	0.6	(O)
4	0.6	(Y)
5	0.8	(E, K)
6	0.6	(E, O)
7	0.6	(M, K)
8	0.6	(O, K)
9	0.6	(K, Y)
10	0.6	(E, O, K)