Assignment 3

CS422 - Data Mining
Amit Nikam (A20470263)

Recitation Exercises

Exercise 2:

1. Support for {e} = 8/10 = 0.8
   Support for {b, d} = 2/10 = 0.2
   Support for {b, d, e} = 2/10 = 0.2

2. Confidence for {b, d}→{e} = s({b, d, e}) / s({b, d}) = 0.2 / 0.2 = 100%
   Confidence for {e}→{b, d} = s({b, d, e}) / s({e}) = 0.2 / 0.8 = 25%
   No, confidence is not a symmetric measure.

3. Support for {e} = 4/5 = 0.8
   Support for {b, d} = 5/5 = 1
   Support for {b, d, e} = 4/5 = 0.8

4. Confidence for {b, d}→{e} = s({b, d, e}) / s({b, d}) = 0.8 / 1 = 80%
   Confidence for {e}→{b, d} = s({b, d, e}) / s({e}) = 0.8 / 0.8 = 100%

5. There is no relationship between an association rule r when treating each transaction ID as market basket and an association rule r when treating each Customer ID as market basket.

Exercise 6:

1. There are six items in the dataset. Therefore the total number of association rules that can be extracted are = $3^6 - 2^7 + 1 = 602$.

2. As the maximum transaction length is 4, the max size of frequent itemsets will be 4.

3. An expression for the maximum number of size-3 itemsets that can be derived from this dataset is: ${}^6C_3 = 6! / (6-3)! * 3! = 20$.

4. Since Beer and Cookies have low individual support, we do not use combinations with these items. After finding the support for the rest of the itemsets, we get the highest support for {Bread, Butter} with a support of 5.

5. Confidence {a} → {b} = support({a, b}) / support {a} and Confidence {b} → {a} = support({a, b}) / support {b}. Therefore if we want rules with confidence of {a} → {b} = {b} → {a} we need to find items with the same individual support. Therefore the pair of items are, (Beer, Cookies) and (Bread, Butter).

Exercise 8:

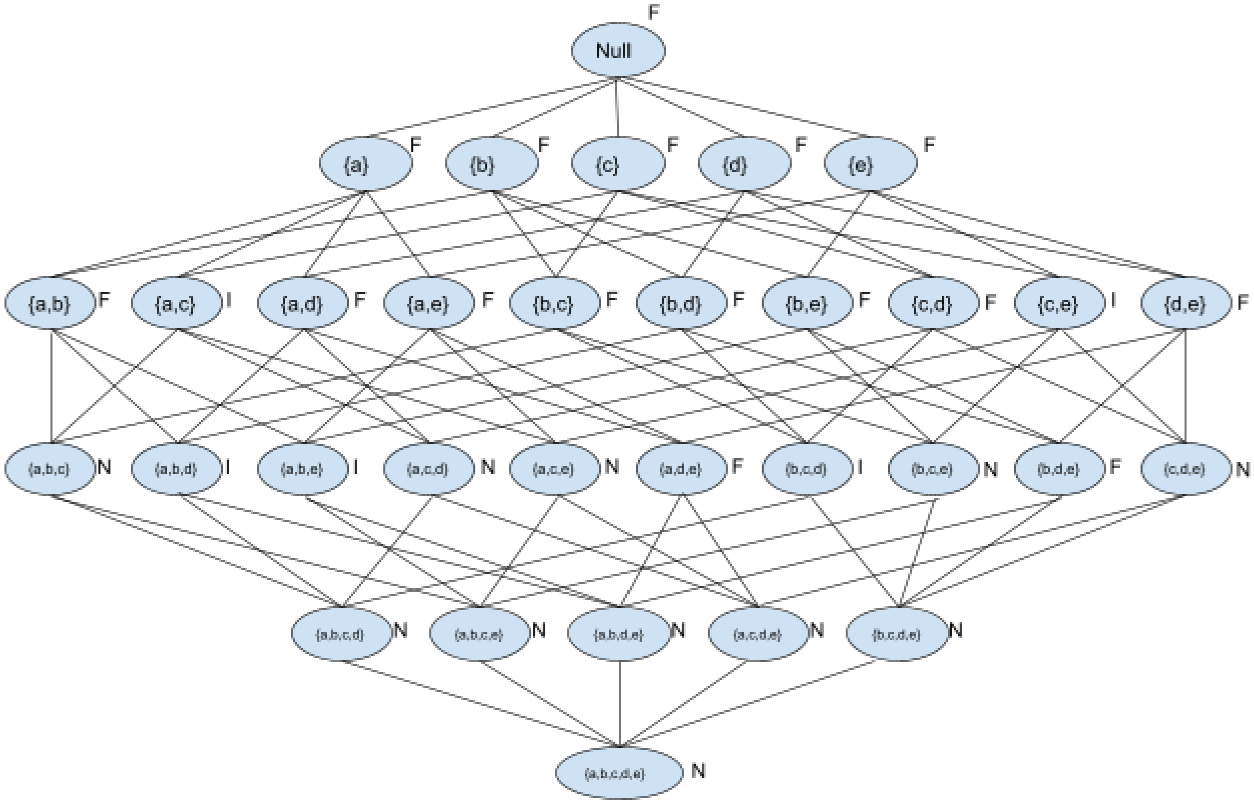
1. {1,2,3} gives {1,2,3,4},{1,2,3,5}
   {1,2,4} gives {1,2,4,5}
   {1,3,4} gives {1,3,4,5}
   {2,3,4} gives {2,3,4,5}
   Rest all combinations are duplicates.

2. From the given list of frequent 3-itemsets we get the following: {1,2,3,4}, {1,2,3,5}, {1,2,4,5}, {1,3,4,5}, {2,3,4,5}.

3. 4-itemset will have ${}^4C_3$ = 4 subset (of 3-itemset). So the sets with all four 3-itemsets in the frequent list will survive.
   {1,2,3,4} survives as it has subsets {1,2,3}, {1,2,4}, {1,3,4} and {2,3,4} which are frequent.
   {1,2,3,5} also survives as it has subsets {1,2,3}, {1,2,5}, {1,3,5} and {2,3,5} which are frequent.

Exercise 9:

1.

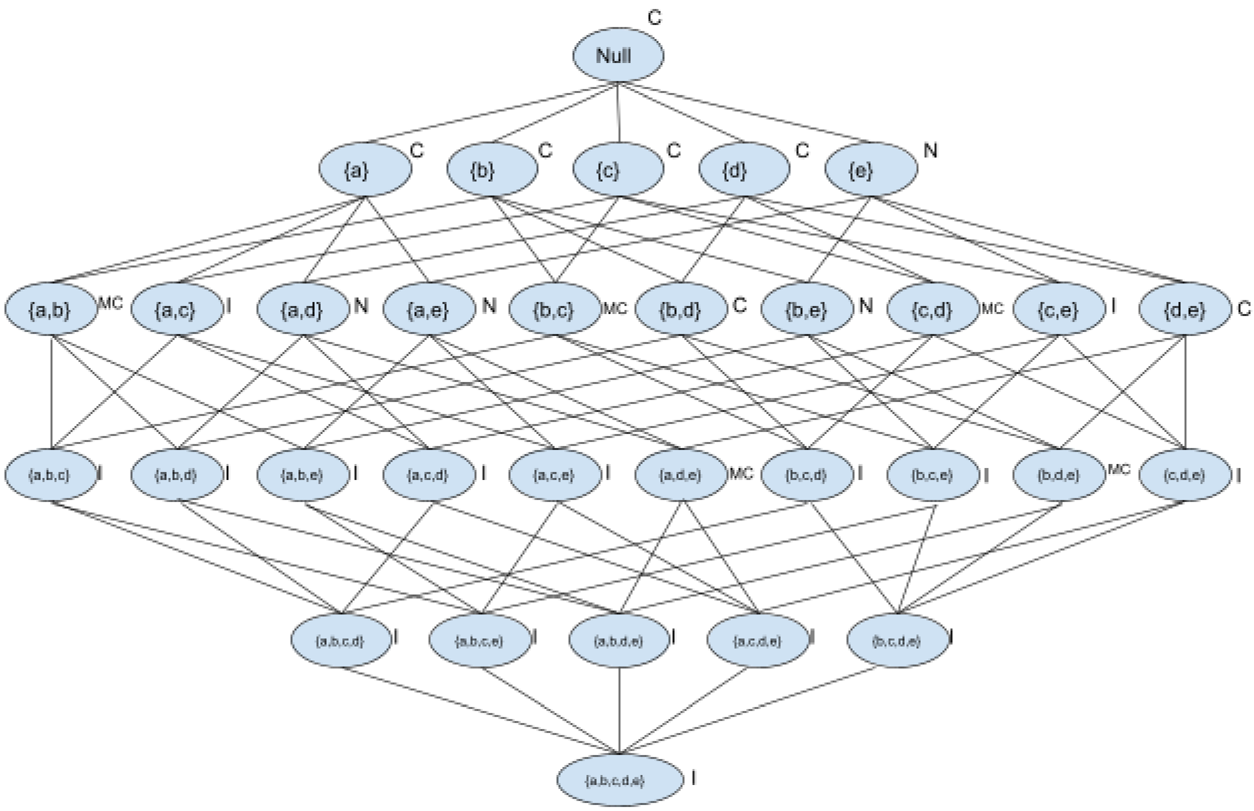


2. Percent of frequent itemsets = itemsets with F / total itemsets = 16/32 = 50%

3. Pruning ration = itemsets with N / total itemsets = 11/32 = 34.4%

4. False alarm ratio = itemsets with I / total itemsets = 5/32 = 15.6%

Exercise 12:



Exercise 13:

1.

{b}→{c}	c	!c
b	3	4
b!	2	1

{a}→{d}	d	!d
a	4	1
!a	5	0

{b}→{d}	d	!d
b	6	1
!b	3	0

{e}→{c}	c	!c
e	2	4
!e	3	1

{c}→{a}	a	!a
c	2	3
!c	3	2

2. Measures:

2.1. Support:

Rule	Support	Rank
{b}→{c}	3/10 = 0.3	3
{a}→{d}	4/10 = 0.4	2
{b}→{d}	6/10 = 0.6	1
{e}→{c}	2/10 = 0.2	4
{c}→{a}	2/10 = 0.2	4

2.2. Confidence:

Rule	Confidence	Rank
{b}→{c}	3/7 = 0.4285	3
{a}→{d}	⅘ = 0.8	2
{b}→{d}	6/7 = 0.86	1
{e}→{c}	2/6 = 0.333	5
{c}→{a}	⅖ = 0.4	4

2.3. Interest (X→Y) = (P(X, Y) / P(X)) * P(Y):

Rule	Interest	Rank
{b}→{c}	(0.3/0.7)*0.5 = 0.214	3
{a}→{d}	(0.4/0.5)*0.9 = 0.72	2
{b}→{d}	(0.6/0.7)*0.9 = 0.771	1
{e}→{c}	(0.2/0.6)*0.5 = 0.167	5
{c}→{a}	(0.2/0.5)*0.5 = 0.2	4

2.4. $IS(X \rightarrow Y) = P(X, Y) / \sqrt{P(X)*P(Y)}$

Rule	IS	Rank
{b}→{c}	$0.3/\sqrt{0.7*0.5} = 0.507$	3
{a}→{d}	$0.4/\sqrt{0.5*0.9} = 0.596$	2
{b}→{d}	$0.6/\sqrt{0.7*0.9} = 0.756$	1
{e}→{c}	$0.2/\sqrt{0.6*0.5} = 0.365$	5
{c}→{a}	$0.2/\sqrt{0.5*0.5} = 0.4$	4

2.5. $Klosgen(X \rightarrow Y) = \sqrt{P(X,Y)} * \max(P(Y|X))-P(Y), P(X|Y)-P(X))$, where $P(Y|X) = P(X, Y)/P(X)$.

Rule	P(Y|X)−P(Y)	P(X|Y)−P(X)	Klosgen	Rank
{b}→{c}	-0.0715	-0.1	-0.039	2
{a}→{d}	-0.1	-0.056	-0.063	4
{b}→{d}	-0.04	-0.034	-0.033	1
{e}→{c}	-0.167	-0.2	-0.075	5
{c}→{a}	-0.1	-0.1	-0.045	3

2.6. $\text{Odds ratio}(X \rightarrow Y)= P(X, Y)*P(\bar{X}, \bar{Y}) / P(X, \bar{Y})*P(\bar{X}, Y)$

Rule	Odds ratio	Rank

{b}→{c}	0.3*0.1 / 0.4*0.2 = 0.375	2
{a}→{d}	0.4*0 / 0.1*0.5 = 0	4
{b}→{d}	0.6*0 / 0.1*0.3 = 0	4
{e}→{c}	0.2*0.1 / 0.4*0.3 = 0.167	3
{c}→{a}	0.2*0.2 / 0.3*0.3 = 0.444	1

Exercise 20:

1. s(A) = 10/100 = 0.1
   s(B) = 10/100 = 0.1
   s(A,B) = 9/100 = 0.09

   I(A,B) = (9/10) * 10 = 9

   φ(A, B) = (9*89 - 1*1) / √(10*10*90*90) = 800/900 = 0.89

   c(A → B) = 9 / 10 = 0.9
   c(B → A) = 9 / 10 = 0.9

2. s(A) = 90/100 = 0.9
   s(B) = 90/100 = 0.9
   s(A,B) = 89/100 = 0.89

   I(A,B) = (89 / 90) * 90 = 89

   φ(A, B) = (9*89 - 1*1) / √(10*10*90*90) = 800/900 = 0.89

   c(A → B) = 89 / 90 = 0.98
   c(B → A) = 89 / 90 = 0.98

3. It is observed that correlation coefficient is invariant under the inversion operation of the sets, while the Interest, support and confidence vary. This is since correlation coefficient takes into account both absence and presence of an item in a transaction.

Practicum Problems

In [1]:
```
import pandas as pd
from mlxtend.frequent_patterns import apriori, association_rules
```

Problem 1

In [2]:
```
# import excel
retail_df = pd.read_excel('./data/Online Retail.xlsx')

retail_df
```

Out[2]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
...	...	...	...	...	...	...	...	...
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	2011-12-09 12:50:00	0.85	12680.0	France
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	2011-12-09 12:50:00	2.10	12680.0	France
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	2011-12-09 12:50:00	4.15	12680.0	France
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	2011-12-09 12:50:00	4.15	12680.0	France
541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	2011-12-09 12:50:00	4.95	12680.0	France

541909 rows × 8 columns

In [3]:
```
# prepare data
retail_df['Description'] = retail_df['Description'].str.strip()
retail_df.dropna(axis = 0, subset=['InvoiceNo'], inplace = True)
retail_df['InvoiceNo'] = retail_df['InvoiceNo'].astype('str')
retail_df = retail_df[~retail_df['InvoiceNo'].str.contains('C')]

retail_df
```

Out[3]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
...	...	...	...	...	...	...	...	...
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	2011-12-09 12:50:00	0.85	12680.0	France
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	2011-12-09 12:50:00	2.10	12680.0	France
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	2011-12-09 12:50:00	4.15	12680.0	France
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	2011-12-09 12:50:00	4.15	12680.0	France
					2011-12-09			

541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	12:50:00	4.95	12680.0	France

532621 rows × 8 columns

In [4]:

```python
# group data and one-hot encode
Basket = (retail_df[retail_df['Country']=="France"]
          .groupby(['InvoiceNo', 'Description'])['Quantity']
          .sum().unstack().reset_index().fillna(0)
          .set_index('InvoiceNo'))

def sum_to_boolean(x):
    if x<=0:
        return 0
    else:
        return 1

Basket = Basket.applymap(sum_to_boolean)

Basket.drop('POSTAGE', inplace=True, axis=1)

Basket
```

Out[4]:

Description	10 COLOUR SPACEBOY PEN	12 COLOURED PARTY BALLOONS	12 EGG HOUSE PAINTED WOOD	12 MESSAGE CARDS WITH ENVELOPES	12 PENCIL SMALL TUBE WOODLAND	12 PENCILS SMALL TUBE RED RETROSPOT	12 PENCILS SMALL TUBE SKULL	12 PENCILS TALL TUBE POSY	12 PENCILS TALL TUBE RED RETROSPOT	12 PENCILS TALL TUBE WOODLAND	...	V
InvoiceNo												
536370	0	0	0	0	0	0	0	0	0	0	...	
536852	0	0	0	0	0	0	0	0	0	0	...	
536974	0	0	0	0	0	0	0	0	0	0	...	
537065	0	0	0	0	0	0	0	0	0	0	...	
537463	0	0	0	0	0	0	0	0	0	0	...	
...	...	...	...	...	...	...	...	...	...	...	...	
580986	0	0	0	0	0	0	0	0	0	0	...	
581001	0	0	0	0	0	0	0	0	0	0	...	
581171	0	0	0	0	0	0	0	0	0	0	...	
581279	0	0	0	0	0	0	0	0	0	0	...	
581587	0	0	0	0	0	0	0	0	0	0	...	

392 rows × 1562 columns

In [5]:

```python
# extract frequent itemsets & find itemset having the largest support
Frequent_itemsets = apriori(Basket, min_support = 0.05, use_colnames = True)
Frequent_itemsets.sort_values('support', ascending = False).head(1)
```

Out[5]:

	support	itemsets
46	0.188776	(RABBIT NIGHT LIGHT)

In [6]:

```python
# extract the association rules with the highest 'confidence'
Asso_Rules = association_rules(Frequent_itemsets, metric = "confidence")
Asso_Rules.sort_values('confidence',ascending = False).head(1)
```

Out[6]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
12	(SET/6 RED SPOTTY PAPER PLATES, SET/20 RED RET...	(SET/6 RED SPOTTY PAPER CUPS)	0.102041	0.137755	0.09949	0.975	7.077778	0.085433	34.489796

In [7]:

```python
# extract the association rules with the highest 'Lift'
Asso_Rules = association_rules(Frequent_itemsets, metric = "lift")
Asso_Rules.sort_values('lift',ascending = False).head(1)
```

Out[7]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
38	(PACK OF 6 SKULL PAPER CUPS)	(PACK OF 6 SKULL PAPER PLATES)	0.063776	0.056122	0.05102	0.8	14.254545	0.047441	4.719388

Conclusion:

Itemset 'RABBIT NIGHT LIGHT' has the highest support with 0.1887

Highest Confidence of 0.975 is found for the following association (antecedents -> consequents):

- SET/20 RED RETROSPOT PAPER NAPKINS, SET/6 RED SPOTTY PAPER PLATES -> SET/6 RED SPOTTY PAPER CUPS

Highest Lift of 14.25 is found for the following association (antecedents -> consequents):

- PACK OF 6 SKULL PAPER CUPS -> PACK OF 6 SKULL PAPER PLATES

The rule with highest confidence is not the same as rule with highest lift. This is because confidence only takes into account the support of antecedents to find the likeliness of occurrence of consequent in the basket. While lift is the rise in probability of having consequent in the basket with the knowledge of antecedent being present.

Problem 2

In [8]:

```
# import csv
df = pd.read_csv('./data/75000-out2-binary.csv')
df
```

Out[8]:

	Transaction Number	Chocolate Cake	Lemon Cake	Casino Cake	Opera Cake	Strawberry Cake	Truffle Cake	Chocolate Eclair	Coffee Eclair	Vanilla Eclair	...	Lemon Lemonade	Raspberry Lemonade	Orange Juice	Gree Te
0	1	0	0	0	0	0	0	0	0	0	...	0	0	0	
1	2	0	0	0	0	0	0	0	1	0	...	0	0	0	
2	3	0	0	0	1	0	0	0	0	0	...	0	0	1	
3	4	0	0	0	0	0	1	0	0	0	...	0	0	0	
4	5	0	0	0	0	0	0	1	0	0	...	0	0	1	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
74995	74996	0	0	0	0	1	0	0	0	0	...	1	0	0	
74996	74997	0	0	0	0	0	0	0	0	0	...	0	0	0	
74997	74998	0	0	0	0	0	0	0	1	0	...	0	0	0	
74998	74999	0	0	0	0	0	0	0	0	0	...	0	0	1	
74999	75000	0	0	0	1	0	0	0	0	0	...	0	0	0	

75000 rows × 51 columns

In [9]:

```
item1_name = 'Chocolate Coffee'
item2_name = 'Chocolate Cake'

selection = df[[item1_name,
                  item2_name]]

item1_count = selection[item1_name] == 1
item2_count = selection[item2_name] == 1

f = selection.groupby([item1_count,
                         item2_count]).count()
f
```

Out[9]:

		Chocolate Coffee	Chocolate Cake
Chocolate Coffee	**Chocolate Cake**		
False	**False**	65802	65802
	True	2962	2962
True	**False**	2933	2933
	True	3303	3303

In [10]:

```
print(f'Chocolate Coffee => Chocolate Cake : {selection["Chocolate Coffee"].corr(selection["Chocolate Cake"])}')
print(f'Chocolate Cake => Chocolate Coffee : {selection["Chocolate Cake"].corr(selection["Chocolate Coffee"])}')
```

```
Chocolate Coffee => Chocolate Cake : 0.48556649252787826
Chocolate Cake => Chocolate Coffee : 0.48556649252787837
```

Conclusion:

Chocolate Coffee and Chocolate Cake items are symmetric binary variables.

Mathemathecally correlation coefficient Φ =
$FREQ_{11}$ *$FREQ_{00}$ - $FREQ_{10}$* $FREQ_{01}$ / sqrt($FREQ_{1x}$ *$FREQ_{x1}$* $FREQ_{0x}$ * $FREQ_{x0}$)

where,
$FREQ_{11}$ = Both Occur
$FREQ_{00}$ = None Occur
$FREQ_{10}$ = Only First one occurs
$FREQ_{01}$ = Only Second one occurs
$FREQ_{1x}$ = All where 1st is occuring irrespective of what second item is
$FREQ_{x1}$ = All where 2nd is occuring irrespective of what first item is
$FREQ_{0x}$ = All where 1st is not occuring irrespective of what second item is
$FREQ_{x0}$ = All where 2nd is not occuring irrespective of what first item is

Therefore changing the order of {Chocolate Coffee} => {Chocolate Cake} to {Chocolate Cake} => {Chocolate Coffee} makes no difference on the outcome. Correlation coefficient Φ simply measures the behavior of both feeatures and compares them.