Algorithm chi-squared support

input: a chi-squared significance level a, support s, support fraction p>0.25,
Basket Data B
output: A set of minimal correlated itemsets from B.

1. for each item i in I, count O(i)

2. initialize CAND <- 0, SIG <- 0, NOTSIG <- 0

3. For each pair ia, ib such that O(ia) > s and O(ib) > s, add {ia, ib} to CAND.

4. NOTSIG <- 0

5. IF CAND is empty, return SIG, end

6. for each itemset in CAND, construct the contingency table, if less than p percent of the cells have count s, then go to 8

7. if the chi-squared value exceeds a threshold, then add itemset to SIG, else add it to NOTSIG.

8. continue to next itemset in CAND. If there are no more items in CAND, set CAND to be the set of all S such that S-1 itemsets are not significant. go to step 4.

Algorithms for correlation rules:

Finding correlations is identical to finding a border in the itemset lattice.

Borders can be very large, and therefore, we need to identify interesting itemsets in the border.

Use support to develop a level-wise algorithm that identifies interesting borders at i+1 itemsets from i itemsets.

The notion of support must be changes wince we also take into account negation.

We redefine support of items S as s at p% level if atleast p% of the cells in the contingency table for S have the value s.

Other pruning techniques are anti-support.

Prune based on chi-squared values (prune off high values under the assumption that they are probably obvious).

Contrast with support-confidence framework

- The chi-squared test is better grounded in statistics theory
- The test takes into account systematically the presence and absence of attributes.
- The interest measure is prefered to confidence.
- Experimental data supports this approach.

Limitations:

Chi-square tests rely on normal approximations of distributions.
This approximation breaks down when the expected values are small.

As a rule of thumb, the following are recommended:

- all cells in the contingency table have expected value > 1
- at least 80% of the cells in the contingency table have expected value > 5

This is rarely satisfied!

However, we can still use this test.

Measures of interest:

The statistical definition of dependence between two sets A and B is

P(A and B) / (P(A) * P(B))

The dependence of the cell is defined to be its interest.

In contingency table notation,

I(r) = O(r) / E(r).

The farther this number is from 1, the higher the dependence, and interest.

Example:

The interest values for the previous example are:

	i2	i2’
i7	1.07	0.44
i7’	0.89	1.91

This indicates the interesting cells!

Example:

i2: never served in military
i7: no more than 40 years old.

	i2	i2'	
i7	17918	911	18829
i7'	9111	2430	11541
	27029	3341	30370

The chi-squared value is 2006.34

(independence hypothesis is discarded).

The biggest value comes from i2'i7'

(i.e. people serve in the military and people over 40 are strongly correlated.)

For comparison, if we run the support-confidence framework on this, we get

i2' => i7' i2 => i7
i7' => i2 i7 => i2

(at confidence of 0.5 and support of 1%) These rules make no sense.

Example:

	i8	i8’	SUM(row)
i9	1	2	3
i9’	4	2	6
SUM	5	4	9

E(i9) = O(i9) = 3, E(i8) = O(i8) = 5

The Chi-squared value is K(i8,i9) + K(i8,i9’) + K(i8’,i9) + K(i8’i9’)

can be computed to be 0.9

Since 0.9 < 3.84 at the 95% confidence interval, we do not reject the independence hypothesis.

$E(ij) = O(ij)$ for a single item.

$E(ij') = n - O(ij)$

$E(r) = n * (E(r1) / n) * (E(r2) / n) * ... * (E(rk)/n))$

$Chi^2 = SUM_{r \in R} (O(r) - E(r))^2 / E(r)$

i.e. it is the normalized deviation from expectation.

We can restrict the range of r if we need to compute the value for a subset.

- calculate the value of the chi-squared statistic
- if all variables are independent, they would all be 0.
- if it is higher than a cutoff, we reject the independence assumption.

It can be shown that chi-squared test is upward closed!

The chi-squared test

Let R = {i1,i1'} X {i2,i2'} x ... x {ik,ik'}

and r = r1...rk in R.

R is the set of all basket values.

r is a single basket value.

Each value of r denotes a cell (in a k-dimensional space)

Let O(r) denote the number of baskets falling in cell r.

To test if a given cell is dependent, we must determine if the actual count in r differs sufficiently from the expectation.

In the chi-squared test, expectation is calculated under the assumption of independence.

The border of correlation:

Instead of a border on top, here we have a border on the bottom.

Combining correlation with support, we can prune both tests simultaneously.

To show that support does not form a border, consider the following:

Example:

d	c	c’	sum
t	8	1	9
t’	40	2	42
sum	48	3	51

d’	c	c’	sum
t	10	2	12
t’	35	2	37
sum	45	4	49

P(c AND d) = 48, P(c) = 93; therefore c => d has a confidence of 0.52

on the other hand P(c AND d and t) = 8 and P(t and c) = 18 therefore c,t => d has a confidence of 0.44.

For a confidence cutoff of 0.5, c => d has the confidence, but c,t => d does not.

Border properties are extremely useful for developing fast algorithms.

The closure property:

Support is downward closed (i.e. if an itemset has support, so do all of its subsets).

(random walk and level-wise algorithms).
Downward closure is a pruning property. To use it, we assume that all (i+1) itemsets are supported. As we examing i-itemsets, we cross out certain (i+1) itemsets since they are guaranteed to not have support. Then we check (i+1) itemsets to make sure they have the support.

Upward closure is a constructuve procedure.

We start with the assumption that no (i+1) itemset is correlated.
Finding an i itemset that is correlated, we can determine all i+1 itemsets that are correlated.

Unlike in the former where we generate false positives, here we generate false negatives (ignored i+1 itemsets)

Because of this, upward closure is most useful when we are looking for an unwanted property.

Then we are looking to prune some (i+1) itemset; and all that happens if we miss some correlated itemsets is that our pruning is less efficient.

Thus we look for minimal correlated itemsets.

Correlation Rules:

Let P(A) be the probability of event A and P'(A) = 1 - P(A)
(the probability that A does not occur)

P(AB) = probability that both A and B occur
P(A'B) = probability that A does not occur but B occurs.

Events A and B are independent if P(AB) = P(A)*P(B)

If A and B are not independent, then they are said to be dependent.

A and B are said to be correlated if any of P(AB), P(A'B), P(AB') or P(A'B') are dependent.

If we have n trials, we denote the number of occurances as O(A)

P(A) = O(A) / n

In the context of itemsets, {ia1, ia2, ..., iam} is a correlation rule if the occurances are correlated.

An important property of correlations is that it is upward closed.

If a set of items S is correlated, so is every superset of S.
(i.e. by adding items to a set, you cannot get rid of the correlation!)

In the coffee and tea example, we calculated the correlation but could not tell if it was statistically significant.

The preferred test for correlation involves the chi-squared test which is easy to calculate as well as generally reliable.

The test is also useful because in addition to computing correlation, it can also be used to calculate negative implication.

let us apply the support-confidence framework to this data for the rule t => c

The support is 20%.

Confidence = 20 / 25 = 0.8

We are tempted to believe that t => c is a good rule.

However, consider now the fact that 90% of the people buy coffee.
A customer who is known to buy tea is 10% less likely to buy coffee than another for who we have no data!!

Infact, there is a negative correlation between tea and coffee.

One way of measuring correlation is to compute

P(t and c) / P(t) * P(c) i.e. 0.2 / (0.25 * 0.9) = 0.89.

The fact that this number is 1 indicates a poor correlation between the two!

The support-confidence framework works well for market-basket problems.

Negations are more difficult to build into this framework:

for instance, assertions of the kind

“People who had fatal accidents did not usually wear seat belts.”

are difficult.

The support-confidence framework does not work well when correlation is the appropriate measure.

Example

t = tea
c = coffee

	c	NOT(c)	Sum (rows)
t	20	5	25
NOT(t)	70	5	75
Sum	90	10	100

Let I = {i1, i2, ..., ik} be the set of k items
B = {b1, b2, ..., bn} be a set of n subsets of I.

Each bi is called a basket of items.(a grocery basket)

Examples:

Grocery lists

Web documents

Note that in this framework, the ordering of documents is lost.

An association rule of the form i1 => i2 if

1. i1 and i2 occur together in atleast s% of the baskets (support)
and
2. of all baskets containing i1, at least c% also contain i2 (confidence)

The definition extends easily to I => J where I and J are disjoint sets.

Generalizing associations to correlations

Associations are used for rules of the type

A customer purchasing item A also purchases item B

How do we mine generalized baskets?
(Generalized baskets are subsets from an item space)

eg. Corpus of documents
Census data

Develop notion of mining rules that identify correlations
as well as take into consideration the absence of an item.

The significance of the rule is measured using the chi-squared test.