Screening and Interpreting Multi-item Associations Based on Log-linear Modeling

Xintao Wu
UNC at Charlotte
9201 Univ. City Blvd
Charlotte, NC 28223
xwu@uncc.edu

Daniel Barbará
George Mason University
ISE Dept.
Fairfax, VA 22303
dbarbara@gmu.edu

Yong Ye
UNC at Charlotte
9201 Univ. City Blvd
Charlotte, NC 28223
yye@uncc.edu

ABSTRACT

Association rules have received a lot of attention in the data mining community since their introduction. The classical approach to find rules whose items enjoy high support (appear in a lot of the transactions in the data set) is, however, filled with shortcomings. It has been shown that support can be misleading as an indicator of how interesting the rule is. Alternative measures, such as lift, have been proposed. More recently, a paper by DuMouchel et al. proposed the use of all-two-factor loglinear models to discover sets of items that cannot be explained by pairwise associations between the items involved. This approach, however, has its limitations, since it stops short of considering higher order interactions (other than pairwise) among the items. In this paper, we propose a method that examines the parameters of the fitted loglinear models to find all the significant association patterns among the items. Since fitting loglinear models for large data sets can be computationally prohibitive, we apply graph-theoretical results to divide the original set of items into components (sets of items) that are statistically independent from each other. We then apply loglinear modeling to each of the components and find the interesting associations among items in them. The technique is experimentally evaluated with a real data set (insurance data) and a series of synthetic data sets. The results show that the technique is effective in finding interesting associations among the items involved.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications - data mining, statistical database

Keywords

Association Rule, Log-linear Model, Graphical Model

SIGKDD '03, August 24-27, 2003, Washington, DC, USA

1. INTRODUCTION

Since their introduction in [1], association rules have received a lot of attention in the data mining community, having been used in multiple applications. Association rules are defined by the support of the set of items (itemset) that are involved in the rule (number of transactions in the database that contain the items), and their confidence (number of times that the right hand side appears in records where the left hand side itemset appears). Algorithms to discover association rules usually prune the choices by considering only itemsets whose support exceeds a threshold. A key property, called Apriori, states that for an itemset to exhibit high support, all its subsets must have high support. This has given way to a popular algorithm (Apriori [2]) that searches for high support itemsets incrementally, beginning from itemsets of size 1, and considering candidates for high support whose size is one unit higher than those considered in the previous iteration. Other efficient algorithms have been investigated ([14]).

In spite of the success of association rules, there are inherent problems with the concept of finding rules based on their support and confidence. In [19], Silverstein et. al show the pitfalls of using support as the guide for pruning rules. It shows that "interest" (or lift), the ratio between the actual probability of the itemset divided by the product of the individual probabilities of each item, is a better guide. Obviously, the denominator in the interest is simply the estimated probability using independence. So, this ratio simply compares the actual support with the estimation that results from assuming independence among the items. Contrary to rules based exclusively on support, those that are found by using lift show that there exists some correlation between the itemset on the right hand side of the rule and the one on the left hand side (as long as the lift value is greater than 1). As the authors of [19] show, rules of the type $X -> Y$, that have high support for the itemset XY, may be misleading in the sense that the itemset Y overall support may be higher when considered by itself than when considering only transactions that also contain the itemset X.

In [12], DuMouchel and Pregibon go further in showing the limitations of support-based algorithms. Assume you have a three item set ABC with strong support and lift. You really do not know if these are the consequence of a strong show of the triplet ABC or because a combination of two attributes is the strong one (e.g., AB or AC) – They present a meaningful example using two drugs (AB) and

kidney failure (C): given that you find association between the three is it due to the combined effect of the two drugs (ABC)? Or is it simply the effect of one (AC)? The authors propose to select the multi-item associations that can not be explained by the **p**airwise associations in the item set by using the standard statistical theory of log-linear models.

However, DuMouchel and Pregibon stop short of fully analyzing the **i**nterestingness of multi-item associations. The interpretation of "interesting" large item sets can be confusing since it is often unclear whether the item set is interesting because it contains all the items, or if it is interesting because it consists of interesting subsets of items.

In this paper, we will analyze and interpret the associations among items by using loglinear modeling. Loglinear models describe association patterns among categorical variables. With the loglinear approach, we model cell counts in a contingency table in terms of associations among the variables. There are several problems that need to be addressed in order to apply loglinear models to market basket data. First, loglinear modeling is usually applied to domains with low or medium dimensionality (< 20). In a typical market basket application, the number of dimensions may be much larger than that. The number of transactions may be very large as well [19], as opposed to the typical data sets for which loglinear modeling is applied. Also, with loglinear models, we need to have at least 5 times the number of cases as cells in our data, a requirement that is not commonly met by market basket data (as the contingency table is very sparse). Lastly, the complexity of algorithms for computing the maximum likelihood estimates (MLE) in loglinear models is exponential in the dimension of the table thus computationally expensive for large tables. Hence building loglinear models directly over all items is prohibitive. Fortunately for us, not all combinations of items exhibit associations: some itemsets may be independent from other itemsets. We apply graph-theoretical results to divide the problem into smaller components of items and fit each component using a loglinear model.

Our work is different from DuMouchel's work [12] in the following aspects. First, we aim to get only one optimal loglinear model to describe all the possible associations among the items in the component instead of building many all-two-factor models. For example, in the component composed by five variables (item $ABCDE$), they need to build 15 all-two-factor models, one for each multi-item set (i.e, $ABC, ABD, \cdots, ABCD, \cdots, ABCDE$) and compare with shrinkage estimates. Second, we interpret the associations among the items by using standardized parameters of fitted loglinear model instead of the EXCESS2 measure used in [12] [1]. The large EXCESS2 value indicates complex relationships involving more than pairwise association among the items of the item set. However, from EXCESS2 we can not always infer what causes the support of the itemset to be a large value. For example, if we know that the EXCESS2 measure for $ABCD$ is large, is it due to ABC, ABD or $ABCD$? By analyzing the parameters of the fitted loglinear model, we can interpret the interestingness of associations among items. The γ-term included in the fitted loglinear model ($\gamma^{ABC}, \gamma^{ABCD}$ etc.) precisely describes the interactions of items. Third, by analyzing residuals, we can automatically pick out the multi-item associations that can not be explained by all the (not just pairwise) associations included in our fitted loglinear model in the item set. As our model fits better than the assumed all-two-factor model, the number of residuals generated by our method is far less than that generated by all-two-factor model.

The rest of the paper is organized as follows. In Section 2 we review the loglinear model. Section 3 presents our method. Experimental results are discussed in Section 4. In Section 5 we draw conclusions and describe directions for future work.

2. LOGLINEAR MODELS REVISITED

Loglinear modeling is a methodology for approximating discrete multidimensional probability distributions. The multiway table of joint probabilities is approximated by a product of lower-order tables. In the database area, loglinear modeling techniques have been successfully applied to high dimensional data compression [5, 6], histogram synopses [11], query approximation [17], and exploratory data cube analysis [18]. Here we should note that loglinear models use only categorical attributes and continuous attributes must be converted to discrete values first.

For a value $y_{i1i2\cdots in}$ at position i_r of the rth dimension d_r ($1 \le r \le n$), we define the log of anticipated value $\hat{y}_{i1i2\cdots in}$ as a linear additive function of contributions from various higher level group-bys as:

$$\hat{l}_{i1i2\cdots in} = \log \hat{y}_{i1i2\cdots in} = \sum_{G \subseteq \{d_1, d_2, \cdots, d_n\}} \gamma^G_{(i_r | d_r \in G)} \qquad (1)$$

We will refer to the γ terms as the coefficients of the model. The coefficients corresponding to any group-by G are obtained by subtracting from the average l value at group-by G all the coefficients from higher level group-by-s.

For instance, in a 4-dimensional table with dimensions A, B, C, D, we use (i, j, k, l, y_{ijkl}) to denote the cell in a 4-D cube space, where $i = 0, \cdots, I-1, j = 0, \cdots, J-1, k = 0, \cdots, K-1, l = 0, \cdots, L-1$. Equation 2 shows the saturated loglinear model which contains all the possible k-factor effects, all the possible $k-1$-factor effects, and so on up to the 1-factor effects and the mean γ. For example, γ^A_i is one-factor effect, γ^{AB}_{ij} is two-factor effect which shows the dependency within the distributions of the associated attributes A, B. The singly-subscripted terms are analogous to main effects, and the doubly-subscripted terms are analogous to two-factor interactions.

$$\begin{aligned} \log \hat{y}_{ijkl} \quad = \quad & \gamma + \gamma^A_i + \gamma^B_j + \gamma^C_k + \gamma^D_l \\ & + \gamma^{AB}_{ij} + \gamma^{AC}_{ik} + \gamma^{AD}_{il} + \gamma^{BC}_{jk} + \gamma^{BD}_{jl} + \gamma^{CD}_{kl} \\ & + \gamma^{ABC}_{ijk} + \gamma^{ABD}_{ijl} + \gamma^{ACD}_{ikl} + \gamma^{BCD}_{jkl} \\ & + \gamma^{ABCD}_{ijkl} \end{aligned} \qquad (2)$$

Equation 3 shows the linear constraints among coefficients, where a dot "." means that the parameter has been summed over the index (For example, $\gamma^{AB}_{i.} = \sum_{j=0}^{J-1} \gamma^{AB}_{ij}$). In short, the constraints specify that the loglinear parameters sum to 0 over all indices.

[1] $EXCESS2 = \Lambda \times e - e_{All2F}$ denotes an estimate of the number of transactions containing the item set over and above those that can be explained by the pairwise associations of the items in the item set, $\Lambda \times e$ is shrinkage estimates which is a substitute of raw data, e_{All2F} is predicted count of all-two-factor model based on all two-way distribution.

$$\begin{aligned}
\gamma_{.}^{A} = \gamma_{.}^{B} = \gamma_{.}^{C} = \gamma_{.}^{D} = 0 \\
\gamma_{i.}^{AB} = \gamma_{.j}^{AB} = \gamma_{i.}^{AC} = \gamma_{.k}^{AC} = \cdots = \gamma_{.l}^{CD} = 0 \\
\cdots \\
\gamma_{ijk.}^{ABCD} = \gamma_{ij.l}^{ABCD} = \gamma_{i.kl}^{ABCD} = \gamma_{.jkl}^{ABCD} = 0
\end{aligned} \tag{3}$$

Equation 4 shows how to compute the coefficients in a 4-dimensional table.

$$\begin{aligned}
\gamma &= l_{....} \\
\gamma_i^A &= l_{i...} - \gamma \\
&\cdots \\
\gamma_{ij}^{AB} &= l_{ij..} - \gamma_i^A - \gamma_j^B - \gamma \\
\gamma_{ijk}^{ABC} &= l_{ijk.} - \gamma_{ij}^{AB} - \gamma_{ik}^{AC} - \gamma_{jk}^{BC} - \gamma_i^A - \gamma_j^B - \gamma_k^C - \gamma \\
&\cdots
\end{aligned} \tag{4}$$

In [18] a fast computation technique called the UpDown method that makes this approach feasible for large sets is described. In the Up-phase, all the l parameters shown before are computed. For each group-by in Equation 4, the corresponding l value from the parameters in the previous group-by-s is computed. For example, in order to compute $l_{ij..}$, we could use the values of $l_{ijk.}$, aggregating for all k. (In general, there is more than one way of computing the parameters, since there is a lattice of group-by aggregations; A benefit analysis approach like the one in [15] can be used to select the best choice.) We need to start from the most detailed group-by: in general this is the one defined by the raw data.

In the Down-phase, for each group-by starting from the least detailed (for instance, $l....$ in Equation 4), we can compute the corresponding effect (i.e., γ) at G by subtracting from the corresponding l value the parameters from all the group-by-s H where $H \subset G$. (For instance, to compute γ_{ij}^{AB}, we need to subtract from $l_{ij..}$ the values of γ_i^A, γ_j^B and γ.)

It is obvious that a large number of models can be used to fit a given data set. For an k-dimensional loglinear model, there are a total 2^{2^k} possible models (determined by which parameters of the saturated model are set to zero). There are several possible strategies of model selection (see [8] for more discussion). One approach consists of fitting the model having only single-factor terms, then the model having only single-factor and two-factor terms, then the model having only three-factor and lower order terms, and so forth. Fitting such models often reveals a restricted range of good-fitting models. In our earlier work [6], we apply this strategy to compress data cubes where the objective is to achieve a good compression ratio instead of interpreting the associations.

Brown et. al., in [9, 7], suggested model fitting by using two tests to screen the importance of each possible term. In one test the term is the most complex parameter in a simple model, whereas in the other test all parameters of its level of complexity are included. This strategy works well when our main objective is to test whether a particular interaction is present or significant. However, this strategy involves a large computational cost to build loglinear model as it needs to evaluate the importance of all possible terms.

3. OUR METHOD

In this section we describe in detail how we screen and interpret associations by means of building loglinear models and examining their parameters and residuals using market basket data. For market basket data, we define each transaction, such as list of items purchased, as a subset of all possible items.

DEFINITION 1. *Let $I_1, \cdots, I_k$ be a set of k boolean variables called attributes. Then a set of baskets $B = \{b_1, \cdots, b_n\}$ is a collection of n k-tuples from $\{TRUE, FALSE\}^k$ which represent a collection of value assignments to the k attributes.*

Our method involves decomposing the initial set of items into groups that are mutually independent, building loglinear models for these components, interpreting associations and examining residuals. The method can be sketched as follows:

- Step 1. Decompose k items into m groups $S = \{S_1, \cdots, S_m\}$, where $\|S_i\| = k_i$. (Section 3.3.)
- Step 2. Transform market basket data into m contingency tables with dimension size k_i respectively.
- Step 3. For each contingency table,
  - Step 3.1. Apply the UpDown method to compute the parameters of saturated model over each derived 2^{k_i} contingency table.
  - Step 3.2. Order and partition the parameters into bins according to their magnitude. Fit and compare two models iteratively by including in the first model those parameters from the first j bins and in the second model those parameters from the first $j+1$ bins. If the second model fits well while the first one does not, go to Step 3.3. Otherwise, increase j by one and repeat Step 3.2.
  - Step 3.3. Examine iteratively each interaction from the $j+1$-th bin by comparing the current model with the new model including one new parameter. The likelihood estimation is used to test the significance of each interaction.
  - Step 3.4. Examine the parameters of the fitted model to derive the interestingness patterns of associations.
  - Step 3.5. Examine residuals computed from the fitted model.

As we stated in the introduction, to effectively process a data set with large number of items, we need to decompose the items into components and build a loglinear model for each component separately. All the significant interactions of a loglinear model built over the original data set must remain unchanged in the loglinear models built over components. In other words, the MLEs for each parameter of the original model should equal to the MLEs of models for components. We apply graph-theoretical results to decompose the items into components while keeping the MLEs of parameters unchanged. We leave the discussion of this part in Section 3.3 and assume the number of dimensions is low or medium (i.e., $k < 20$) in Sections 3.1 and 3.2. In Section 3.1

Table 1: COIL 2000 data set with four dimensions denoted by A, B, C, D respectively

		B True		B False	
		A True	A False	A True	A False
D True	C True	457	1162	175	526
	C False	156	48	12	0
D False	C True	944	851	89	307
	C False	901	187	6	1

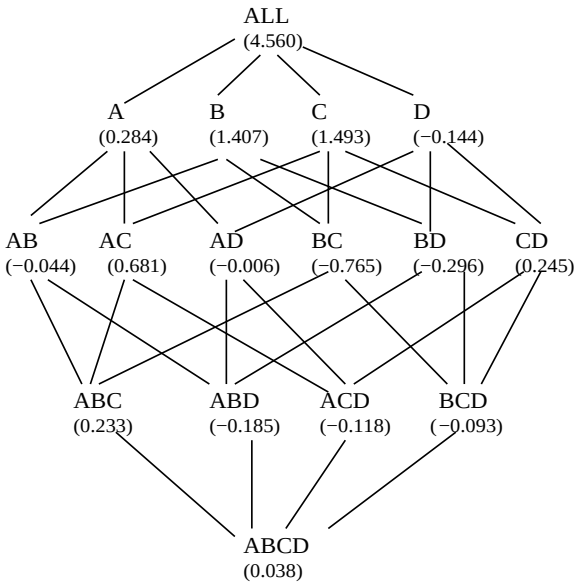


Figure 1: Lattice for the data set with four dimensions denoted by A, B, C, D respectively. The value in () denotes the value of γ-term of saturated loglinear model

we focus on how to fit loglinear model for each component. In Section 3.2 we present how to interpret the interesting patterns of associations and how to screen interesting itemsets by examining the parameters and residuals of the fitted loglinear model.

3.1 Loglinear Model Fitting

The first step of loglinear model fitting is to compute the parameters of the saturated model by applying the UpDown method(Step 3.1).

We present our strategy by using one example. Table 1 shows a contingency table with four attributes. This table is derived from COIL real data set [10]. The original data set contains 86 attributes and we present our experiment with full 86 attributes in Section 4.1. Table 2 shows the meaning of the four attributes (A,B,C,D) and the other six attributes (E-J). These ten attributes are the most significant attributes after applying univariate analysis [22]. We use these ten attributes to illustrate our method (including decomposition and loglinear model fitting).

Figure 1 shows the parameter values from the saturated model computed by using the UpDown method. Each of the γ-term in the saturated loglinear model describes the interaction of item variables. For example, γ^{AB} represents the interaction between item A and B. Notice that in market basket data, each item variable can only have two categories: presence, absence. Hence, each of the γ-term has only one absolute value due to linear constraints of coefficients (See Equation 3) and the positive (negative) value implies positive (negative) associations. For example, $\gamma^{AB} = -0.044$ in Figure 1 implies $\gamma_{00}^{AB} = -0.044$, $\gamma_{01}^{AB} = 0.044$, $\gamma_{10}^{AB} = 0.044$, and $\gamma_{11}^{AB} = -0.044$. It can be interpreted that the presence (absence) of A implies the absence (presence) of B with interaction effect 0.044. Note in general contingency table cases, the γ-term for a particular interaction (i.e., γ^{AB}) has more than one absolute value due to variables with more than two categories.

Furthermore, we can compare the interactions according to their magnitude of γ-terms derived from the saturated models in market basket case. For example, the comparison of γ^{AC} (0.681) and γ^{CD} (0.245) implies the interaction of AC is more significant than that of CD. It is important to point out that, in general, we cannot compare the magnitude of γ-terms directly. This is due to several reasons. First, the degree of freedom (d.f.) for each particular interaction varies (however, in market basket data, the d.f. for each particular interaction is always 1). Secondly, the variance for each interaction varies (however, in market basket data, the variances for all γ-terms equal to the same value -see Appendix A for proof details). The values $\gamma_{00}^{AC} = 0.681$ and $\gamma_{00}^{CD} = 0.245$ do not necessarily imply that the interaction of AC is greater than that of CD, since the variances of γ_{00}^{AC}, γ_{00}^{CD} can be different. So in the general case, we have to compute the standardized parameter value $(\gamma/\sigma(\gamma))$ for each γ-term in order to compare the significance of each interaction. The cost to standardize each γ-term is very high. Thirdly, in general, there can be more than one absolute value for each γ-term and we have to combine the estimates in some way to form an overall test statistic (This is usually hard and subjective [13]). However, for market basket data, we do not need to do this since each γ term exactly has one absolute value.

Our modeling strategy consists in ordering the γ-terms based on their magnitude and including those γ-terms exceeding some threshold (we can do this since the γ-terms are comparable). The idea of fitting the saturated model and noting which estimates of association and interaction parameters are large compared to their estimated standard errors was first proposed by Goodman, in [13]. However, it is not widely used because the high computational cost of standardizing parameters. For market basket data, this idea is very attractive as we can drastically decrease the cost of modeling without computing the variance of each γ-term.

To determine which interactions should be included in the fitted model, we need a threshold. However, there is no good way to determine the threshold for a given data set. It is unknown what the distribution of all γ-terms estimates is, although each γ-term estimate follows an approximate normal distribution with mean γ and variance $\sigma(\gamma)$ [3]. We apply a heuristic strategy here. We order γ-terms according to their magnitude and divide them into bins (equi-width). We first include in the starting model those terms in the first bin. When that model fits well, it may be possible to simplify it and remove some terms with small absolute values. When it does not fit well, we need to include additional parameters in the second bin. In other words, we

Table 2: COIL significant attributes used in example. The column "Mapping" shows how to map each original variable to binary variable.

attribute	$i-$th attribute	Name	Description	Mapping
A	18	MOPLLAAG	Lower level education	$> 4 \rightarrow 1$
B	37	MINKM30	Income < 30K	$> 4 \rightarrow 1$
C	42	MINKGEM	Average income	$> 4 \rightarrow 1$
D	43	MKOOPKLA	Purchasing power class	$> 3 \rightarrow 1$
E	44	PWAPART	Contribution private third party insurance	$> 0 \rightarrow 1$
F	47	PPERSAUT	Contribution car policies	$> 0 \rightarrow 1$
G	59	PBRAND	Contribution fire policies	$> 0 \rightarrow 1$
H	65	AWAPART	Number of private third party insurance	$> 0 \rightarrow 1$
I	68	APERSAUT	Number of car policies	$> 0 \rightarrow 1$
J	86	CARAVAN	Number of mobile home policies	$> 0 \rightarrow 1$

keep comparing models built with parameters up to the j-th and $j+1$-th bin until the latter fits well. During step 3.3, we apply the likelihood ratio L^2 (see equation 5) to assess the importance of terms in $j+1$-th bin. The likelihood ratio is minimized and follows a chi-square distribution with the d.f. equal to the number of γ-terms set equal to zero. For a given d.f., larger L^2 values give smaller right-tail probabilities (P-values), and represent poor fits. Equation 6 shows how the L^2 statistics is used for comparison of two models. The d.f. is calculated by subtracting the d.f. of model2 from the d.f. of model1. In this step, the difference of d.f. is always 1 as the two models we compared are same except the tested γ-term.

$$L^2 = 2\sum y_i Log(y_i/\hat{y}_i) \tag{5}$$

$$\begin{aligned} L^2_{comparison} &= L^2_{model1} - L^2_{model2} \\ &= 2\sum y_i Log(\hat{y}_i^{model2}/\hat{y}_i^{model1}) \end{aligned} \tag{6}$$

3.2 Interpreting and Screening Associations

As we stated in the introduction, we are departing from the majority of published approaches to the market basket problem by going beyond the examination of frequent itemsets. The idea of using measures other than itemset frequency has been explored a few times. For example, in [19], they propose measuring significance of dependence via the chi-squared test for independence from classical statistics. In [12], they only distinguish between multi-item associations that can be explained by all pairwise associations, and item sets that are significantly more frequent than their pairwise associations would suggest. In our framework, we interpret associations by examining the γ-terms of fitted loglinear models instead of by examining the differences between observed frequencies of itemsets and expected frequencies computed from assumed models.

$$\log \hat{y}_{lift} = \gamma + \gamma^A + \gamma^B + \gamma^C + \gamma^D \tag{7}$$

$$\begin{aligned} \log \hat{y}_{pairwise} &= \gamma + \gamma^A + \gamma^B + \gamma^C + \gamma^D + \gamma^{AB} \\ &\quad + \gamma^{AC} + \gamma^{AD} + \gamma^{BC} + \gamma^{BD} + \gamma^{CD} \end{aligned} \tag{8}$$

$$\begin{aligned} \log \hat{y}_{fitted} &= \gamma + \gamma^A + \gamma^B + \gamma^C + \gamma^D + \gamma^{AC} + \gamma^{BC} \\ &\quad + \gamma^{BD} + \gamma^{CD} + \gamma^{ABC} + \gamma^{ABD} \end{aligned} \tag{9}$$

Now we illustrate the difference of our work with previous approaches using an example. Equation 7 assumes the independence model and includes all-one-factor (main) effects and grand mean. Equation 8 includes all-two-factor effects apart from all-one-factor effects and grand mean. The comparison between the observed value y with either $\hat{y}_{lift}$ or $\hat{y}_{pairwise}$ is used to screen interesting itemsets in [19] or [12] respectively. The assumed independence model (shown in Equation 7) or pairwise model (shown as Equation 8) may be inaccurate. By comparing with an inaccurate model, false interpretations may be introduced when we examine itemsets. In our framework, we fit the market basket data to derive the fitted loglinear model (as shown in Equation 9) instead of just assuming some specific model (independence or pairwise model).

As our model really fits the underlying data and includes significant interactions at all possible levels, we can derive the association patterns by examining the γ-terms of our fitted model directly. For example, from $\gamma^{AC} = 0.681$, $\gamma^{BC} = -0.765$, and $\gamma^{ABC} = 0.223$, we can see the positive two-factor interaction (i.e., the presence of one item implies the presence of the other one) between item A and C, the negative two-factor interaction between item B and C, no significant two-factor interaction between item A and B, and positive three-factor interaction among ABC. From $\gamma^{BD} = -0.296, \gamma^{ABD} = -0.185$, we can see the negative two-factor interaction between item B and D, the three-factor negative interaction among ABD, however no significant two-factor interaction for item sets AB or AD.

We would like to point out that we apply a non-hierarchical modeling strategy (step 3.2 and 3.3). Hierarchical models are nested models in which when an interaction of d factors is present, all the interactions of lower order between the variables of that interaction are also present. For example, if a three-way interaction (γ^{ABC}) is present, the model must also include all two-way effects ($\gamma^{AB}, \gamma^{AC}, \gamma^{BC}$) as well as the single variable effects ($\gamma^A, \gamma^B, \gamma^C$) and the grand mean ($\gamma$). Non-hierarchical modeling can better interpret the associations for market basket data. Consider one example for the concept of synergism[2] where each item from A, B, C is sold independently and the third item is free if cus-

[2]A response occurs when two factors are present together but not when either occurs alone. This is the perfect illustration for the example that two drugs together cause kidney failure.

Table 3: Comparison of three models. Residuals is the number of cells by comparing standardized residuals to standard normal percentage points 3.29

Model	Likelihood ratio L^2	d.f.	Residuals
independence	2597.4	11	10
all-two-factor	226.7	5	6
our model	64.8	5	2

tomer buys any other two items. Clearly there exists a three-way interaction effect (γ^{ABC}) but no two-way interaction between any pairs among A, B, C ($\gamma^{AB}, \gamma^{AC}, \gamma^{BC}$). Clearly only non-hierarchical model (i.e., $\log \hat{y}_{ijk} = \gamma + \gamma_i^A + \gamma_j^B + \gamma_k^C + \gamma_{ijk}^{ABC}$) can explain this case correctly. Notice in the non-hierarchical model, the two-way effects are not included in the model therefore violating the hierarchical requirement.

The parameters of the loglinear model provide the interactions between item variables. Further analysis of residuals may reveal in cell-by-cell comparisons of observed and fitted frequencies. Note here our loglinear model is built at the finest level (containing all variables) and it is easy to compute expected frequencies of itemsets at any upper level by simply summing those cells from the finest level. Equation 10 shows the standardized residual form used in our framework.

$$e_i = \frac{y_i - \hat{y}_i}{\hat{y}_i^{1/2}} \tag{10}$$

When the model holds, e_i is asymptotically normal with mean 0. In comparing standardized residuals to standard normal percentage points, we obtain conservative indications of cells having lack of fit. Table 3 shows the comparison of independence model, pairwise model, and our fitted model for the COIL data set. The likelihood ratio and the size of residuals from Table 3 clearly show that our fitted model is better than the independence and pairwise models as it includes significant high-factor effects and excludes those non-significant 2-factor effects (even the main effect).

3.3 Graphical Decompositions for Large Contingency Tables

As we stated in the introduction, we cannot build loglinear models over the very sparse and large contingency table that results from market basket data. Besides, even if the data set is dense, the complexity of algorithms for computing the MLEs in loglinear models is generally exponential in the dimension of the item variables and thus computationally expensive for large tables. In this section, we discuss how to decompose the problem into subsets and build loglinear models for each subset without losing any significant interaction. We do this by using graph-theoretical results. The procedure involves two steps: 1) we build one *i*ndependence graph for all item variables; 2) we apply graph-theoretical results to decompose the graph into non-decomposable irreducible components.

The *i*ndependence graph is defined by making every vertex of the graph correspond to a discrete random variable, and the edges denoting the dependency of the two variables linked. A missing edge in the graph represents the conditional independence of the two variables associated with the two vertices. Models with the maximal permissible higher-order interactions corresponding to a given independence graph are called *g*raphical models. (See [16, 21] for comprehensive treatment of graphical models.) Figure 2(a) shows the independence graph (two disconnected subgraphs FI, ABCDJGEH) for our COIL data set. From this graph, we can infer for instance that variables I and F are independent with respect to the remaining variables. We can also derive that variables E and H are conditionally independent with respect to the set ACDEJ given the variable G. Intuitively, there is no interaction between any variable from set EH and any variable from the set ACDEJ given variable G.

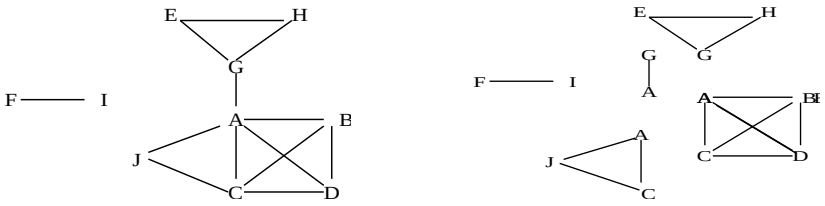
The second step is to decompose the graph into basic, irreducible components. Graph-theoretical results show that if a graph corresponding to a graphical model for a contingency table is decomposable into subgraphs by a clique separator [3], the MLEs for the parameters of the model can easily be derived by combining the estimates of the models on the lower dimensional tables represented by the simpler subgraphs. Hence, applying a divide-and-conquer approach based on the decompositions will make the procedure applicable to much larger tables.

The theory may be interpreted by the following way: if two disjoint subsets of vertices S_a and S_b are separated by a subset S_c in the sense that all paths from S_a to S_b go through S_c, then the variables in S_a are conditionally independent of those in S_b given the variables in S_c. The subgraphs may be further decomposed into subgraphs. The requirement that the subgraph on S_c is complete implies that there is no further independence constraints on the elements of S_c, so that this factorization contains all the information about the joint distribution.

Figure 2(b) shows the components (ABCD, ACJ, EHG, AG, IF). We can see the interactions among ABCD are independent with respect to other variables. The interactions among ABCD (i.e., γ^{AB}, γ^{AC}, γ^{ABC} etc.) can be derived directly from the condensed 4-dimensional contingency table (i.e., ABCD) instead from the original 10-dimensional contingency table (i.e., ABCDEFGHIJ). The MLEs of the interactions for each component are the same as those for the original graphs. In our experiments, we apply CoCo [4] within XLISP-STAT with a complexity of $O(nm^3)$, with m the number of generators and n the number of variables, to perform decomposition for large contingency tables. To find the clique separators of a graph or to find the vertex-sets of the irreducible components of the graphs, an algorithm with a complexity of $O(ne + n^2)$ can be used [20], where n is the number of vertices and e is the number of edges.

To build the independence graph, we need to test conditional independence for every pair of variables, controlling for the other variables. There are several approaches to test conditional independence (See [3]). In our paper, we build the independence graph by applying the Cochran-Mantel-Hasenzel test. For any pair of two items I_i, I_j from item set $I = \{I_1, \cdots, I_k\}$, we derive one partial 2×2 contingency table (stratum) for each possible value from set $I \setminus \{I_i, I_j\}$. Hence we can have L ($L = 2^{k-2}$) strata. For each stratum l,

[3] A clique is a subset of vertices which induce a complete subgraph for which the addition of any further vertex renders the induced subgraph incomplete. A graph is complete if all vertices are joined with undirected edges. In other words, the clique is maximally complete.



(a) independence graph (b) decomposed component

Figure 2: Composition of COIL data set with 10 variables

we need to compute the marginal totals $\{n_{.0}^{(l)}, n_{.1}^{(l)}, n_{0.}^{(l)}, n_{0.}^{(l)}\}$. Table 5(a) shows the stratum form for item variable A and B while Table 5(b) shows one stratum ($C = 1, D = 1$) derived from Table 1. Equation 11 shows the summary statistics where $m_{11}^{(l)}$ and $V(n_{11})$ is mean and variance respectively.

$$m_{11}^{(l)} = E(n_{11}^{(l)}) = \frac{n_{1.}^{(l)} n_{.1}^{(l)}}{n_{..}^{(l)}}$$

$$V(n_{11}^{(l)}) = \frac{n_{1.}^{(l)} n_{0.}^{(l)} n_{.1}^{(l)} n_{.0}^{(l)}}{n_{..}^{(l)} n_{..}^{(l)} (n_{..}^{(l)} - 1)}$$

$$M^2 = \frac{(\| \sum n_{11}^{(l)} - \sum m_{11}^{(l)} \| - 0.5)^2}{\sum V(n_{11}^{(l)})} \quad (11)$$

The summary statistics M^2 has approximately a chi-squared distribution with d.f. $= 1$ under the null hypothesis of conditional independence. Hence, if $M^2 > P_\alpha$, we can reject the null hypothesis of conditional independence and include the edge of I_i and I_j in the interaction graph. In our experiments, we choose $\alpha = 0.05$ and $P_\alpha = 3.84146$. However, the Cochran-Mantel-Hasenzel test does not work well for very sparse data sets because the marginal totals for a given partial table usually equal zero. To deal with very sparse market basket data sets, we may test marginal independence for each pair of variables by applying log odds ratio over one *m*arginal 2×2 table (shown in Table 5(c)) which contains summary frequencies and ignores the other controlling variables.

Table 4: A 2×2 contingency table for variable A and B

	B	$\widetilde{B}$	
A	n_{11}	n_{10}	$n_{1.}$
$\widetilde{A}$	n_{01}	n_{00}	$n_{0.}$
	$n_{.1}$	$n_{.0}$	$n_{..}$

(a) stratum form

	B	$\widetilde{B}$	
A	457	175	632
$\widetilde{A}$	1162	526	1688
	1619	737	2320

(b) stratum for $C = 1, D = 1$

	B	$\widetilde{B}$	
A	2458	282	2740
$\widetilde{A}$	2248	834	3082
	4706	1116	5822

(c) marginal table

4. EXPERIMENTAL RESULTS

In this section we show the results of experimenting with one real data set and some synthetic data sets. The experiments were conducted in a DELL Dimension 8100, with one 1.7G processor, and 640 Mbytes of RAM.

4.1 COIL Data

The COIL Challenge 2000 [10] provides data from a real insurance business. The competition consisted of two tasks: 1) Predict which customers are potentially interested in a Caravan insurance policy; 2) Describe the actual or potential customers; and possibly explain why these customers buy a Caravan policy. Information about customers consists of 86 attributes and includes product usage data and socio-demographic data derived from zip area codes. The training set consists 5822 descriptions of customers, including the information of whether or not they have a Caravan insurance policy. A test data set contains 4000 tuples which only the organizers know if they have a Caravan insurance policy. Here our aim is to identify interaction patterns among 86 attributes varying from product usage to socio-demographic. Our data is formed by collapsing non-binary categorical attributes into binary form (the data can be found at www.cs.uncc.edu/ xwu/classify/b86.dat), with $n = 5822$ baskets and $k = 86$ binary items.

We successfully decomposed the data set with 86 variables into components with much less variables (the largest one with 20 variables and most components with less than 5 variables). After decomposition, we got 22 components (Figure 5 shows 10 components which contain 3 or more variables) and we then fit each component by using loglinear model.

We also did the experiment over this data set by using the Apriori algorithm. The algorithm generated 6050 large item sets and 13131 rules under support 0.1 and confidence 0.8. We found it was much harder to draw interesting conclusions about data from support-confidence results. We compared the significant interactions discovered by our algorithm with the large item sets discovered by Apriori algorithm and found the percentage of overlap is very low. Table 6 shows several significant interactions discovered by our loglinear fitting algorithm and the actual support val-

Table 5: Component after decomposition for COIL data set, we omit 12 components which contain less than 3 variables.

component	variables
1	44, 45,46,48,50, 51,52,53,56,58, 59,65,67,68,69,71,72,73,74,79
2	16,17,18,19,22,23,24,25,26,29,41
3	61, 68, 81,82,84
4	1,5,21,43
5	37, 38,39
6	19, 25,36
7	19,25,35
8	16,20,25
9	10,12,13
10	3, 10,13

Table 6: Several significant interactions

term	interaction	actual support(%)
$\gamma^{16,17,18}$	-1.21	0
$\gamma^{1,5,21}$	1.01	14.2
$\gamma^{10,12,13}$	-0.41	0.02
$\gamma^{61,68,81,82}$	0.302	0
$\gamma^{48,74,79}$	0.28	0

ues for those subsets. The lower support value for all subsets (except for $\gamma^{1,5,21}$) definitely prevent them to be discovered by traditional support-confidence framework. For instance, the association $\gamma^{48,74,79}$ reveals that people are inclined to buy delivery van policies (48), agricultural machines policies (74) and disability insurance policies (79) together.

4.2 Synthetic Data set

The COIL data set is too sparse to study the performance (running time) of our algorithm. In order evaluate the performance our algorithm properly, we turn to synthetic data (the same market basket data generator used in [1]) from IBM's Quest Group.

We generated two data sets (one with 50 items and the other with 100 items). We have not done the experiments over data sets with more than 100 variables as we have used CoCo [4] (an environment for graphical models) which can not deal with more than 128 variables. We are currently implementing the decomposition algorithm proposed by [20] to be able to handle data sets with larger number of variables. We set the average basket size to be 10, the average of large itemsets to be 4, the correlation between large itemsets to be 0.25, the confidence in a rule to be 0.75, the number of transactions varying from 10k to 1M. We ran some experiments with the tlen set to 6 or the correlation level set to 0.75 but did not find significant difference in the nature of our performance results.

Table 7: Parameter description

parameter	value	meaning
ntrans	10k-1M	number of transactions
nitems	50,100	number of different items
tlen	10	average items per transaction
npats	10000	number of patterns (large item sets)
patlen	4	average length of maximal pattern
corr	0.25	correlation between patterns
conf	0.75	average confidence in a rule

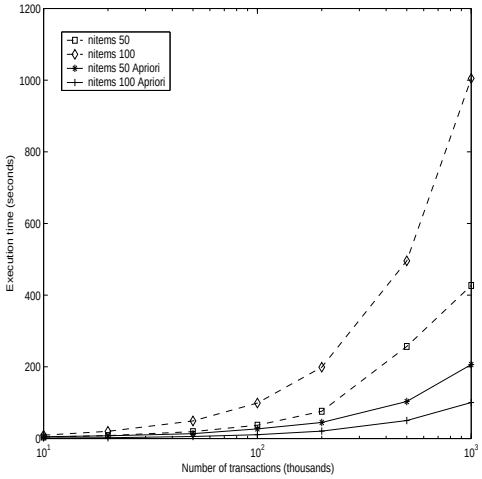


Figure 3: Execution time by varying ntrans from (10K, 20K, 50K, 100K, 200K, 500K, 1M)

Figure 3 shows our execution time. Note that decomposition step is determined by the size of independence graph (i.e., the number of variables k, the number of edges e or the number of generators m). We observe the decomposition time is small compared with the preprocessing time because the size of independence graph in our experiment is usually small (with 100 nodes and several hundreds edges). As the number of items contained in each component is comparably small (most less than 10), the time of loglinear model fitting for each component is trivial. In Figure 3, we also include the execution time of Apriori algorithm (with mininum support 0.1% and minimum confidence 80%). We can see the execution time of our algorithm is comparable to that of Apriori algorithm for medium dimension size (50, 100). Figure 4 shows the number of components generated in our experiment. When we fix the other parameters of market basket generater and increase the number of transactions, the number of components decreases because the number of edges in independence graph increases.

5. CONCLUSIONS AND FUTURE WORK

In this paper we presented how to interpret associations among items by fitting loglinear models and examining those magnitude of parameters for market basket data. Our work departed from earlier work that just aims to find large or interesting itemsets and leaves those itemsets to domain expert directly. On the contrary, we build loglinear model and apply the values of γ-terms as measures of associations among item variables directly. We believe those values provided by our loglinear model are very helpful for domain expert to make judgments about cause and effect relations among items. To deal with large number of variables, we

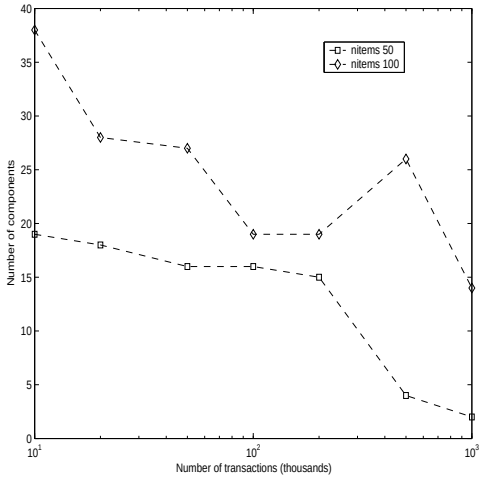


Figure 4: Number of components by varying ntrans from (10K, 20K, 50K, 100K, 200K, 500K, 1M)

applied graph-theoretical results to decompose items into subsets without losing any significant interaction.

There are some aspects of this work that merit further research. Among them, we are trying to automatically derive rules from the γ-terms included in fitted loglinear model. For components with more than 10 variables, it is hard for user to grasp all the association patterns. We will be exploring how to combine visualization techniques and association graph for this issue.

Another aspect that merits further research is that of interactive analysis of associations among items. For example, the user may want to examine a given subset (say ABC). Clearly collapsing into contingency table of ABC directly will lose information as item A, B, or C may have interactions with other items. To find the smallest set containing a given set (i.e., ABC) and onto which the model is collapsible was studied in [4]. We will investigate this problem for online association analysis.

Finally, we will study how to better deal with sparse data when either structural zero cells present or it contains many small cell values. It is known that loglinear model can still work for small incomplete table with structural or sampling zeros [8]. We will investigate other techniques such as shrinkage estimates [12] for large incomplete market basket data.

6. ACKNOWLEDGMENTS

The authors would like to thank Christian Borgelt for providing his implementation of the Apriori algorithm. We would like to thank Jens Henrik Badsberg for his CoCo program which makes our experiments possible. We would also like to thank IBM Quest group for providing the market basket data generator.

7. REFERENCES

[1] R. Agrawal, T. Imilienski, and A. Swami. Mining association rules between sets of items in large databases. In *Proceedings of the ACM SIGMOD International Conference on Management of Database*, pages 207–216, 1993.

[2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proceedings of the International Conference on Very Large Data Bases*, pages 487–499, September 1994.

[3] A. Agresti. *Categorical data anlysis*. Wiley, 1990.

[4] J. Badsberg. An environment for graphical models. Ph.D. Thesis, Aalborg University, Demark, 1995.

[5] D. Barbará, W. DuMouchel, C. Faloutsos, P. Hass, J. M. Hellerstein, Y. Ioannidis, H. Jagadish, T. Johnson, R. Ng, V. Poosala, K. Ross, and K. Sevcik. The new jersey data reduction report. *Bulletin of the Technical Committee on Data Engineering*, 20(4):3–45, December 1997.

[6] D. Barbará and X. Wu. Loglinear based quasi cubes. *Journal of Information and Intelligent Systems*, 16(3):255–276, 2001.

[7] J. Benedetti and M. Brown. Stratigies for the selection of loglinear models. *Biometrics*, 34:680–686, 1978.

[8] Y. M. Bishop, S. E. Fienberg, and P. W. Holland. *Discrete Multivariate Analysis: Theory and Practice*. The MIT Press, Cambride, Massachusetts, and London, England, 1975.

[9] M. Brown. Screening effects in multidimensional contigency tables. *Applied Statistics*, 25:37–46, 1976.

[10] COIL challenge 2000. The insurance company (tic) benchmark. http://kdd.ics.uci.edu/databases/tic/tic.html.

[11] A. Deshpande, M. Garofalakis, and R. Rastogi. Independence is good: Dependency-based histogram synopses for high-dimensional data. In *Proceedings of the 2001 ACM SIGMOD International Conference on Management of Data*, pages 199–210. Santa Barbara, California, May 2001.

[12] W. DuMouchel and D. Pregibon. Empirical bayes screening for multi-item association. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data*. San Francisco, CA, August 2001.

[13] L. Goodman. The analysis of multidimensional contigency tables: Stepwise procedures and direct estimation methods for building models for multiple classifications. *Technometrics*, 13:33–61, 1971.

[14] J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. In *Proceedings of the ACM SIGMOD International Conference on Management of Database*, pages 1–12. Dallas, TX, May 2000.

[15] V. Harinarayan, A. Rajaraman, and J. Ullman. Implementing data cubes efficiently. In *Proceedings of the ACM SIGMOD International Conference on Management of Database*, pages 205–216. Montreal, Quebec, Canada, 1996.

[16] S. Lauritzen. *Graphical Models*. Oxford University Press, 1996.

[17] D. Pavlov, H. Mannila, and P. Symth. Probabilistic models for query approximation with large sparse binary data sets. In *Proceedings of Uncertainty in Artificial Intelligence*, pages 199–210. Stanford, California, June 2000.

[18] S. Sarawagi, R. Agrawal, and N. Meggido. Discovery-driven exploration of olap data cubes. In *Proceedings of the International Conference on*

Extending Data Base Technolgy, pages 168–182. Valencia, Spain, 1998.

[19] C. Silverstein, S. Brin, and R. Motwani. Beyond market baskets: Generalizing association rules to dependence rules. *Data Mining and Knowledge Discovery*, 2:39–68, 1998.

[20] R. Tarjan. Decomposition by clique separators. *Discrete Mathematics*, 55:221–232, 1985.

[21] J. Whittaker. *Graphical Models in Applied Mathematical Multivariate Statistics*. Wiley, 1990.

[22] X. Wu and D. Barbará. Modeling and imputation of large incomplete multidimensional datasets. In *Proceedings of the International Conference on Data Warehousing and Knowledge Discovery*. Aix-en-Provence, France, Septemeber 2002.

APPENDIX

A. VARIANCE OF PARAMETERS OF LOG-LINEAR MODELS

Each parameter (γ) in loglinear model (shown in Equation 12) can be rewritten in the form of a linear contrast of the logarithms of the expected cell counts. For example, Equation 13 shows the linear contrast form for each parameter of 2-d loglinear model.

$$\hat{l}_{i1i2\cdots in} = \log \hat{y}_{i1i2\cdots in} = \sum_{G\subseteq\{d_1,d_2,\cdots,d_n\}} \gamma^G_{(i_r|d_r\in G)} \quad (12)$$

$$\begin{aligned}
\gamma^A_i &= \sum_{l\neq i,m} \frac{-1}{IJ} logy_{lm} + \sum_m \frac{1}{J}(1-\frac{1}{I}) logy_{im} \\
\gamma^B_j &= \sum_{l,m\neq j} \frac{-1}{IJ} logy_{lm} + \sum_l \frac{1}{I}(1-\frac{1}{J}) logy_{lj} \\
\gamma^{AB}_{ij} &= (1-\frac{1}{I})(1-\frac{1}{J}) logy_{ij} + \sum_{l=i,m\neq j} \frac{-1}{J}(1-\frac{1}{I}) logy_{lm} \\
&\quad + \sum_{l\neq i,m=j} \frac{-1}{I}(1-\frac{1}{J}) logy_{lm} + \sum_{l\neq i,m\neq j} \frac{1}{IJ} logy_{lm}
\end{aligned} \quad (13)$$

THEORM 1. *The asymptotic variance of* $f_i(\hat{p}) = \sum_{k=1}^{T} c_{ik} log(\hat{p}_k)$ *is* $N^{-1}(\sum_{k=1}^{T} c_{ik}^2 p_{k^{-1}} - c_{i+}^2)$ *where* $c_{i+} = \sum_{k=1} T c_{ik}$, N *is the size of samples and* T *is the size of cells.*

Theorem 1 ([8], page 495) shows the variance form for parameters which can be expressed as linear contrasts. For example, Equation 14 shows how to compute variance for each parameter of 2-d loglinear model.

$$\begin{aligned}
Var(\gamma^A_i) &= (\frac{1}{IJ})^2 \sum_{l,m} (Ny_{lm})^{-1} + (\frac{I-2}{IJ^2}) \sum_m (Ny_{im})^{-1} \\
Var(\gamma^B_j) &= (\frac{1}{IJ})^2 \sum_{l,m} (Ny_{lm})^{-1} + (\frac{J-2}{IJ^2}) \sum_l (Ny_{lj})^{-1} \\
Var(\gamma^{AB}_{ij}) &= (\frac{1}{IJ})^2 \sum_{l,m} (Ny_{lm})^{-1} + (\frac{I-2}{IJ^2}) \sum_m (Ny_{im})^{-1} \\
&\quad + (\frac{J-2}{IJ^2}) \sum_l (Ny_{lj})^{-1} \\
&\quad + \frac{(I-2)(J-2)}{IJ} (Ny_{ij})^{-1}
\end{aligned} \quad (14)$$

In general case, the variances of parameters are different from each other. However, the variance of each parameter is the same for market basket data as the domain size for each variable (I,J etc.) is always 2.