Realistic Synthetic Data for Testing Association Rule Mining Algorithms for Market Basket Databases

Colin Cooper[1,*] and Michele Zito[2,⋆]

[1] Department of Computer Science, Kings' College, London WC2R 2LS, UK
colin.cooper@kcl.ac.uk
[2] Department of Computer Science, University of Liverpool, Liverpool, L69 3BX, UK
michele@liverpool.ac.uk

Abstract. We investigate the statistical properties of the databases generated by the IBM QUEST program. Motivated by the claim (also supported empirical evidence) that item occurrences in real life market basket databases follow a rather different pattern, we propose an alternative model for generating artificial data.

1 Introduction

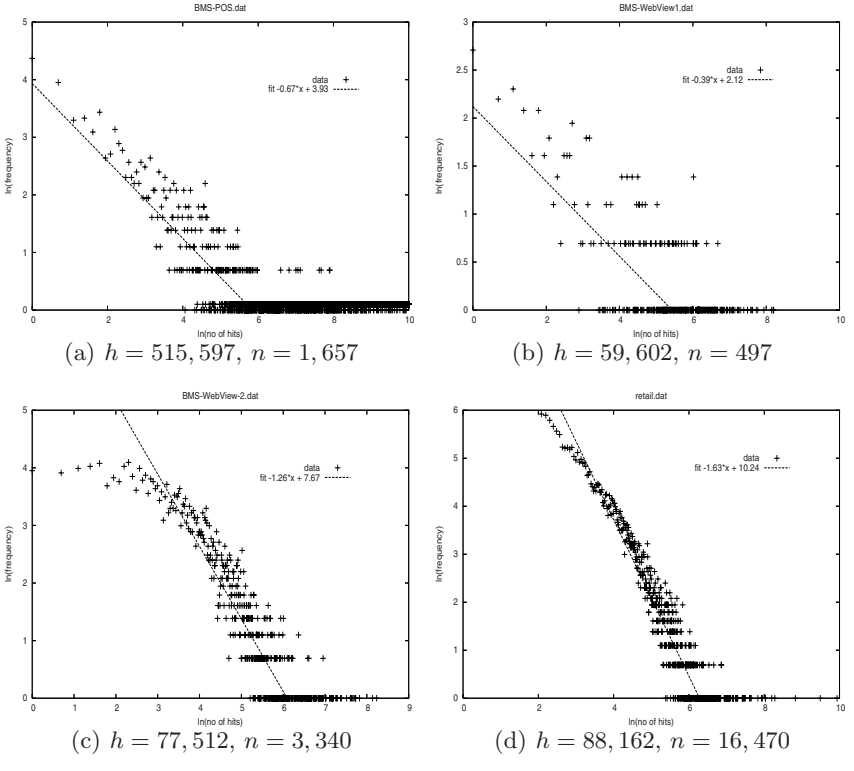
The ARM problem is a well established topic in KDD. Many techniques have been developed to solve this problem (e.g. [1,3,5,9]), however several fundamental issues are still open. The evaluation of ARM algorithms is a difficult task [12], often tackled by resorting to data generated by the well established QUEST program from the IBM Quest Research Group [1]. The intricacy of this program makes it difficult to draw theoretical predictions on the behaviour of the various algorithms on such databases. Empirical analyses are also difficult to generalise because of the wide range of possible variation, both in the characteristics of the data (the structural characteristics of the synthetic data bases generated by QUEST are governed by a several interacting parameters), and in the environment in which the algorithms are being applied. It has also been noted [3] that data produced using QUEST might be inherently not the hardest to deal with. In fact it seems that the performance of some algorithms on real data is much worse than on synthetic data generated using QUEST [11].

In this paper we first claim that *heavy tail* statistical distributions (see [10] for a survey on the topic) arise naturally in characterizing the item occurrence distribution in market basket databases, but are not evident in data generated by QUEST. Statistical differences have been found before [11] between real-life and QUEST generated databases. We contend that our analysis points to possible differences at a much deeper level. Motivated by the outcomes of our empirical investigation, we then study mathematically the distribution of item occurrences in a typical

⋆ The work of both authors was supported by EPSRC grant EP/D059372/1 *Scale-free structures: models and algorithms*.

J.N. Kok et al. (Eds.): PKDD 2007, LNAI 4702, pp. 398–405, 2007.

large QUEST database. At least in a simplified setting, such study confirms the empirical findings. To the best of our knowledge, this is the first analysis of the structural properties of the databases generated by QUEST. Such properties may well be responsible for the observed [3,11] behaviour of various mining algorithms on such datasets. The final contribution of this paper is the description of an alternative synthetic data generator. Our model is reminiscent of the proposal put forward in the context of author citation networks and the web by Barabási and Albert [2]. The mechanism that leads to the desired properties is the so called *preferential attachment*, whereby successive transactions are filled by selecting items based on some measure of their popularity. We complete our argument by proving mathematically that the resulting databases show an asymptotic heavy tailed item occurrence distribution and giving similar empirical evidence.



(a) $h = 515,597,\ n = 1,657$ (b) $h = 59,602,\ n = 497$

(c) $h = 77,512,\ n = 3,340$ (d) $h = 88,162,\ n = 16,470$

Fig. 1. Log-log plots of the real-file data sets along with the best fitting lines

The rest of the paper has the following structure. Section 2 reports our empirical analysis of a number of real and synthetic databases. Section 3 presents the results of our mathematical investigation of the structural properties of the databases generated by QUEST. Finally in Section 4 we describe our proposal for an alternative synthetic data generator.

2 Analysis of Real Data

From now on a database $\mathcal{D}$ is a collection of h transactions, each containing items out of a set $\mathcal{I}$ of n items. For $r \in \{0, \ldots, h\}$ let N_r be the number of items that occur in r transactions. In this section we substantiate the claim that, at least for market basket data, the sequence $(N_r)_{r\in\{0,\ldots,h\}}$ follows a distribution that has a "fat" tail and, on the contrary, the typical QUEST data shows rather different patterns. To this end we use the real data sets `BMS-POS`, `BMS-WebView-1`, `BMS-WebView-2`, and `retail.data` and the synthetic QUEST data `T10I4D100K.dat` and `T40I10D100K.dat` already used in [11], all available from `http://fimi.cs.helsinki.fi/data/`. The plots in Figure 1 show the sequence $(N_r)_{r\in\{0,\ldots,h\}}$ in each case, along with the least square fitting lines computed using the `fit` command of `gnuplot`, over the whole range of values (slope values are reported in each picture). Figure 2 shows the same statistics obtained using the two synthetic databases.

Although it may be argued that the number of real datasets examined is too small and the test carried out too coarse, our calculations indicate that the sequences $(N_r)_{r\in\{0,\ldots,h\}}$ obtained from real-life databases fit a straight line much better than the sequences obtained from the synthetic QUEST databases. Furthermore this phenomenon leads to the additional conjecture that the studied distributions may be *heavy tailed*, i.e. decay at a sub-exponential rate [10]. In the case of *power law* distributions such decay is proportional to x^{-z} for some fixed $z > 0$. On a doubly logarithmic scale data points having such decay would seem to be clustered around a line, which is exactly what happens in the case of the four market-basket datasets described above.

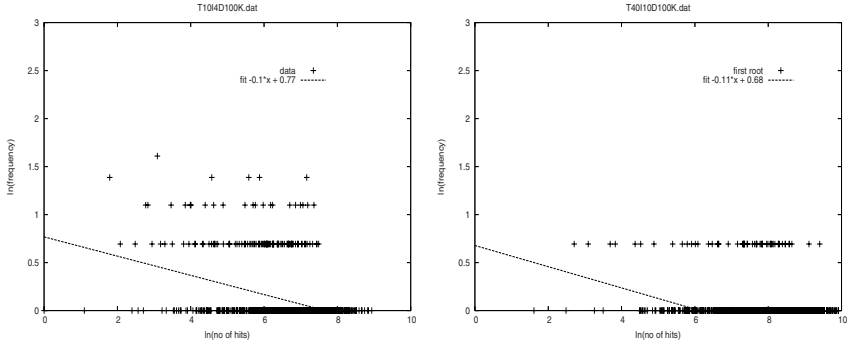


Fig. 2. Log-log plots of the QUEST data sets along with the best fitting lines

3 A Closer Look at QUEST

The QUEST program returns two related structures: the actual database $\mathcal{D}$ and a collection $\mathcal{T}$ of l *potentially large itemsets* or *patterns*, that are used to populate $\mathcal{D}$. The transactions in $\mathcal{D}$ are generated by first determining their size

(picked from a Poisson distribution) and then filling the transaction with the items contained in a number of itemsets selected independently and uniformly at random (u.a.r.) from $\mathcal{T}$. Each itemset in $\mathcal{T}$ is generated by first picking its size from a Poisson distribution with mean equal to I. Items in the first itemset are then chosen randomly. To model the phenomenon that large itemsets often have common items, some fraction (chosen from an exponentially distributed random variable with mean equal to the expected correlation level ρ) of items in subsequent itemsets are chosen from the previous itemset generated. The remaining items are picked at random. The use of QUEST is further complicated by its dependence on a number of additional parameters. Agrawal and Srikant claim that the resulting database $\mathcal{D}$ mimics a set of transactions in the retailing environment. Furthermore they justify the use of the structure $\mathcal{T}$ based on the fact that people tend to buy sets of items together, some people may buy only some of the items from a large itemset, others items from many large itemsets. Finally they observe that transaction sizes and the sizes of the large itemsets (although variable) are typically clustered around a mean and a few collections have many items.

To maximise the clarity of exposition we simplify the definition of the QUEST process. A part from $n = |\mathcal{I}|$, h and l, we assume that system parameters will be the number of patterns per transaction, k, the number of items in each pattern, s, and the number of items shared between two consecutive patterns, ρ, with $\rho \in \{0, \ldots, s\}$. Typically $h >> n$ (e.g. $h = n^{O(1)}$) while $l = O(n)$. Parameters k and s are small constants independent of n. Note that the assumption that the size of the transactions is variable as stated in Agrawal and Srikant's work is "simulated" by the fact that different patterns used to form a transaction may share some items. Following Agrawal and Srikant we use a correlation level ρ between subsequent patterns, but assume it takes the fixed constant value $s/2$. Hence $s/2$ items in each pattern (except the first one) belong its predecessor in the generation sequence and the remaining $s - \rho$ are chosen u.a.r. in $\mathcal{I}$. We assume that $\mathcal{D}$ and $\mathcal{T}$ are populated as follows:

1. Generate $\mathcal{T}$ by selecting s random elements in $\mathcal{I}$ and any subsequent pattern by choosing (with replacement) ρ elements u.a.r. from the last generated pattern and $s - \rho$ elements u.a.r. (with replacement) in $\mathcal{I}$.
2. Generate $\mathcal{D}$ by filling each transaction independently with the elements of k patterns in $\mathcal{T}$ chosen independently u.a.r. with replacement.

Let $\mathrm{d}_{\mathcal{D}}(v)$ (resp. $\mathrm{d}_{\mathcal{T}}(v)$) denote the number of transactions in $\mathcal{D}$ (resp. patterns in $\mathcal{T}$) containing item v. Also, $b(x; n, p) = \binom{n}{x} p^x (1-p)^{n-x}$. Obviously items occurring in many patterns of $\mathcal{T}$ have a higher chance of occurring in many database transactions. The following result quantify the influence of $\mathcal{T}$ on the item occurrence distribution in $\mathcal{D}$.

Theorem 1. *Let $(\mathcal{D}, \mathcal{T})$ with parameters n, h, l, k, s, and ρ as described above. Then for each $v \in \mathcal{I}$, $\mathrm{d}_{\mathcal{D}}(v)$ has binomial distribution with parameters h and $p_{k,l} = \sum_{i=1}^{k} \binom{k}{i} (-1)^{i+1} \frac{\mathrm{E}(\mathrm{d}_{\mathcal{T}}(v))^i}{l^i}$, where $\mathrm{E}(\mathrm{d}_{\mathcal{T}}(v))^i$ is the i-th moment of $\mathrm{d}_{\mathcal{T}}(v)$.*

Proof. By definition the transactions of $\mathcal{D}$ are generated independently of each other. An item v has degree r in $\mathcal{D}$ if it belongs to r fixed transactions. If we assume that each transaction is formed by the union of k patterns chosen independently u.a.r. from $\mathcal{T}$ then $\mathrm{d}_{\mathcal{D}}(v)$ has binomial distribution. The result then follows from the binomial theorem after noticing that the probability that v belongs to a given transaction is: $1 - \frac{\mathrm{E}(l-\mathrm{d}_{\mathcal{T}}(v))^k}{l^k}$. □

The study of the item occurrence distribution in $\mathcal{D}$ is thus reduced to finding the first k moments of $\mathrm{d}_{\mathcal{T}}(v)$. Solving the latter is not easy in general. In the remainder of this Section we sketch our analysis under more restricted assumptions.

Item occurrences in $\mathcal{T}$ when $s = 2$. If $s = 2$, $\mathcal{T}$ is a graph and its structure depends on the value of ρ. W.l.o.g. we focus on the case $\rho = 1$. In such case, the resulting graph can be seen as directed, with edges chosen one after the other according to the following process:

1. The first directed edge e_1 is a random pair from $\mathcal{I}$.
2. If the edge chosen as step i is $e_i = (w, z)$, (for $i \geq 1$), then e_{i+1} is chosen by selecting an item u.a.r. in $\mathcal{I}$, and then selecting the second element of the pair at random as either w or z with probability $\frac{1}{2}$.

Define the degree of v in $\mathcal{T}$ as the sum of its *in-degree* $\mathrm{d}^-_{\mathcal{T}}(v)$ (number of edges having v as second component) and out-degree $\mathrm{d}^+_{\mathcal{T}}(v)$ (number of edges having v as first component).

Theorem 2. *Let $\mathcal{T}$ be given with $s = 2$, $\rho = 1$ and all other parameters specified arbitrarily. Then for each $v \in \mathcal{I}$, $\mathrm{d}^+_{\mathcal{T}}(v)$ has binomial distribution with parameters l and $\frac{1}{n}$. Furthermore, the distribution of $\mathrm{d}^-_{\mathcal{T}}(v)$ can also be computed exactly. In particular $\lim_{n\to\infty} \frac{\mathrm{Ed}^-_{\mathcal{T}}(v)}{\mathrm{Ed}^+_{\mathcal{T}}(v)} = 1$.*

Proof. (Sketch) Under the given assumptions, the first result follows from classical work random allocation of l identical balls in n distinct urns (see for instance [6]).

The in-degrees can also be estimated through a slightly more elaborate argument. Essentially v can occur as second end-point of an edge only if it occurred as first end-point in some previous step. Therefore assuming that $\mathrm{d}^+_{\mathcal{T}}(v) = d$, $\mathrm{d}^-_{\mathcal{T}}(v)$ can be defined as a sum of d non-negative and independent contributions. The asymptotic result on $\mathrm{Ed}^-_{\mathcal{T}}(v)$ is a consequence of the fact that for n large $\mathrm{d}^+_{\mathcal{T}}(v)$ stays very close to its expected value. □

The occurrence distribution in $\mathcal{D}$ in a very simple case. In this Section we further simplify our model, assuming that $k = 1$, i.e. each transaction in $\mathcal{D}$ is formed by a single random edge of $\mathcal{T}$. The following result shows that, when n becomes large, in such simple setting the item occurrence distribution decays super-polynomially (and therefore it cannot have, asymptotically, a heavy tail).

Theorem 3. *If r is such that $n \cdot b(r; h, \frac{2}{n}) \to \infty$ then $\frac{N_r}{n} \to b(r; h, \frac{2}{n})$ with probability tending to one.*

Proof. (Sketch) If $k = 1$ by Theorem 1 $\mathrm{d}_{\mathcal{D}}(v)$ has binomial distribution. By linearity of expectation and Theorem 2 $\mathrm{Ed}_{\mathcal{T}}(v)$ is approximately $2l/n$ as n tends to infinity. Thus, for large n, $p_{1,l}$ is approximately $\frac{2}{n}$. Hence $\mathrm{E}N_r$ tends to $n \cdot b(r; h, \frac{2}{n})$ and the stated result follows from Chebyshev inequality. □

We close this section by noticing another peculiar feature of QUEST. Since l is much smaller than h, there is a constant probability that a given item will never occur in a transaction of $\mathcal{D}$. Equivalently a constant fraction of the n available items will never occur in the resulting database. This phenomenon was observed in the two synthetic databases analysed in Section 2: `T40I10D100K.dat` only uses 941 of the 1,000 available items, `T10I4D100K.dat`, only 861. Of course this irrelevant from the practical point of view, but it's a strange artifact of the choice of having a two-component structure in the QUEST generator.

4 An Alternative Proposal

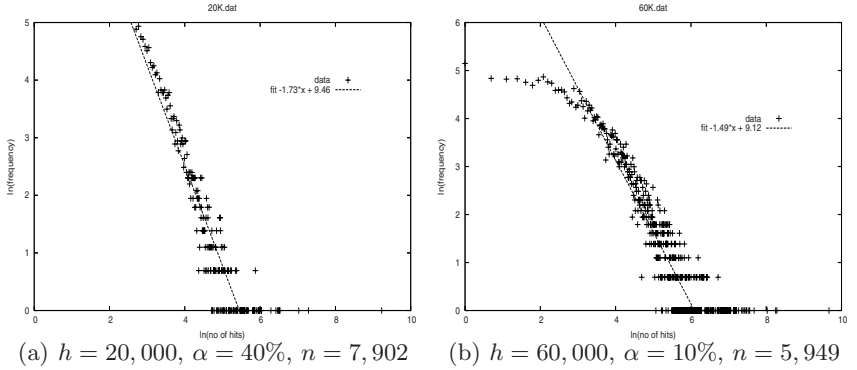
In this Section we describe an alternative way of generating synthetic databases. Our model is in line with the proposal of Barabási and Albert [2], introduced to model structures like the scientific author citation network or the world-wide web. A mechanism, called *preferential attachment*, that allows the process that generates one after the other the transactions in $\mathcal{D}$ to choose their components based on the frequency of such items in previously generated transactions, leads to databases with the desired properties. Instead of assuming an underlying set of patterns $\mathcal{T}$ from which the transactions are built up, the elements of $\mathcal{D}$ are generated sequentially. At the start there is an initial set of e_0 transactions on n_0 existing items. The model can generate transactions based entirely on the n_0 initial items, but in general we assume that new items can also be added to newly defined transactions, so that at the end of the simulation the total number of items is $n > n_0$. The simulation proceeds for a number of steps generating groups of transactions at each step. For each group in the sequence there are four choices made by the simulation at step t:

1. The type of transaction. An OLD transaction (chosen with probability $1-\alpha$) consists of items occurring in previous transactions. A NEW transaction (chosen with probability α) consists of a mix of new items and items occurring in previous transactions.
2. The number of transactions in a group, $m_O(t)$ (resp. $m_N(t)$) for OLD (resp. NEW) transactions. This can be a fixed value, or given any discrete distribution with mean $\overline{m_O}$ (resp. $\overline{m_N}$). Grouping corresponds to e.g. the persistence of a particular item in a group of transactions in the QUEST model.
3. The transaction size. This can again be a constant, or given by a probability distribution with mean $\overline{\pi}$.
4. The method of choosing the items in the transaction. If transactions of type OLD (resp. NEW) are chosen in a step we assume that each of them is selected using preferential attachment with probability P_O (resp. P_N) and randomly otherwise.

Our main result (its proof, along the lines of similar results given by Cooper in [4], is skipped due to space limitations) is that, provided that the number of transactions is large, with probability approaching one, the distribution of item occurrence in $\mathcal{D}$ follows a power law distribution with parameter $z = 1 + \frac{1}{\eta}$, where $\eta = \frac{\alpha\overline{m_N}(\overline{\pi}-1)P_N+(1-\alpha)\overline{m_O}\ \overline{\pi}P_O}{(\alpha\overline{m_N}+(1-\alpha)\overline{m_O})\overline{\pi}}$. In other words, the number of items occurring r times after t steps of the generation process is approximately Ctr^{-z} for large r and some constant C. Furthermore, for fixed values of t, the expected number of items and transactions after t steps are, respectively, $n_0 + \alpha t$ and $e_0 + t(\alpha\overline{m_N} + (1-\alpha)\overline{m_O})$.

Practical considerations. Turning to examples, in the simplest case, the group sizes are fixed (say $m_N(t), m_O(t) = 1$ always) and the preferential attachment behaviour of the transaction types is the same $P_N = P_O = P$. Thus $\eta = 1 - \frac{\alpha P}{\overline{\pi}}$, and $z = 1 + \frac{\overline{\pi}}{\overline{\pi}-\alpha P}$. The following pseudo-code (whose translation in Java can be found at `http://www.csc.liv.ac.uk/~michele/soft.html`) describes a specialization of the proposed procedure under the additional assumption that the transaction sizes are given by the absolute value of a normal distribution with parameters μ and σ (this was done only because Java offers a pseudo-random generator of normally distributed real numbers). Initially $\mathcal{D}$ contains one item and one transaction.

Input: μ, σ, α, P, h
Output: A database $\mathcal{D}$ with h transactions
for $t = 1$ to h
  select the size x as the absolute value of a normally distributed number
    with mean μ and deviation σ
  if $(x > 0)$
    with probability α add a NEW transaction to $\mathcal{D}$
    otherwise add an OLD transaction to $\mathcal{D}$.



(a) $h = 20,000$, $\alpha = 40\%$, $n = 7,902$ (b) $h = 60,000$, $\alpha = 10\%$, $n = 5,949$

Fig. 3. Log-log plots of the item distributions in databases generated from our model

Figure 3 displays item distribution plots obtained from running the program with parameters $\mu = 7$, $\sigma = 3$, $P = 50\%$ for different values of h and α. While

there are many alternative models for generating heavy tailed data (surveys such as [7] or [10] present a rich catalogue) and so different communities may prefer to use alternative processes, we contend that synthetic data generators of this type should be a natural choice for the testing of ARM algorithms.

References

1. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: Proc. of the 20th Int. Conf. on Very Large Data Bases, pp. 487–499. Morgan Kaufmann Publishers Inc, San Francisco (1994)
2. Barabási, A., Albert, R.: Emergence of scaling in random networks. Science 286, 509–512 (1999)
3. Brin, S., Motwani, R., Ullman, J.D., Tsur, S.: Dynamic itemset counting and implication rules for market basket data. In: Proc. of the ACM SIGMOD Int. Conf. on Management of Data, pp. 255–264. ACM Press, New York (1997)
4. Cooper, C.: The age specific degree distribution of web-graphs. Combinatorics. Probability and Computing 15(5), 637–661 (2006)
5. Han, J., Pei, J., Yin, Y.: Mining frequent patterns without candidate generation. In: Proc. of the ACM SIGMOD Int. Conf. on Management of Data, pp. 1–12. ACM Press, New York (2000)
6. Kolchin, V.F., Sevast'yanov, B.A., Chistyakov, V.P.: Random Allocations. Winston & Sons (1978)
7. Mitzenmacher, M.: A brief history of generative models for power law and lognormal distributions. Internet Mathematics 1(2), 226–251 (2004)
8. Redner, S.: How popular is your paper? an empirical study of the citation distribution. European Physical Journal, B 4, 401–404 (1998)
9. Savasere, A., Omiecinski, E., Navathe, S.B.: An efficient algorithm for mining association rules in large databases. In: Proc. of the 21th Int. Conf. on Very Large Data Bases, pp. 432–444. Morgan Kaufmann Publishers Inc, San Francisco (1995)
10. Watts, D.J.: The "new" science of networks. Annual Review of Sociology 30, 243–270 (2004)
11. Zheng, Z., Kohavi, R., Mason, L.: Real world performance of association rule algorithms. In: Proc. of the 7th ACM SIGKDD Int. Conf. on Knowledge Discovery nd Data mining, pp. 401–406. ACM Press, New York (2001)
12. Zaïane, O., El-Hajj, M., Li, Y., Luk, S.: Scrutinizing frequent pattern discovery performance. In: Proc. of the 21st Int. Conf. on Data Engineering (ICDE'05), pp. 1109–1110. IEEE Computer Society, Los Alamitos (2005)