Estimating Joint Probabilities from Marginal Ones*

Tao Li, Shenghuo Zhu, Mitsunori Ogihara, and Yinhe Cheng

Computer Science Department
University of Rochester
Rochester, New York 14627-0226
{taoli,zsh,ogihara,cheng}@cs.rochester.edu

Abstract. Estimating joint probabilities plays an important role in many data mining and machine learning tasks. In this paper we introduce two methods, *minAB* and *prodAB*, to estimate joint probabilities. Both methods are based on a light-weight structure, *partition support*. The core idea is to maintain the partition support of itemsets over logically disjoint partitions and then use it to estimate joint probabilities of itemsets of higher cardinalitiess. We present extensive mathematical analyses on both methods and compare their performances on synthetic datasets. We also demonstrate a case study of using the estimation methods in *Apriori* algorithm for fast association mining. Moreover, we explore the usefulness of the estimation methods in other mining/learning tasks [9]. Experimental results show the effectiveness of the estimation methods.

Keywords: *Joint Probability, Estimation, Association Mining*

1 Introduction

Estimating the joint probabilities in a collection of N observations on M events is the problem of estimating the joint probabilities of events, given the probabilities of single events. Generally the collection of N observations on M events is represented by a $N \times M$ binary table D where $D_{ij} = 1$ denotes that the i-th event occurs in the j-th observation and $D_{ij} = 0$ otherwise. Let $\{\mathcal{I}_j\}$, $j = 1, \cdots, M$, represent the events. $P(\mathcal{I}_j)$ can be estimated by computing its occurrence frequency in the table. Thus given $P(\mathcal{I}_j)$, $j = 1, \cdots, M$, the goal is to estimate the joint probability $P(\mathcal{I}_{j_1}, \cdots, \mathcal{I}_{j_l})$, $l \geq 2, 1 \leq j_1, \cdots, j_l \leq M$.

A simple way to estimate joint probabilities is, like we approximate the probabilities of a single event, to just count the co-occurrences of the events in the table D (i.e., via combinatory counting). Although in many cases this simple method does provide satisfactory solutions, there are cases in which the time

* The project is supported in part by NIH Grants 5-P41-RR09283, RO1-AG18231, and P30-AG18254 and by NSF Grants EIA-0080124, NSF CCR-9701911, and DUE-9980943. We would also like to thank Dr. Meng Xiang Tang and Xianghui Liu for their helpful discussions.

Y. Kambayashi, W. Winiwarter, M. Arikawa (Eds.): DaWaK 2002, LNCS 2454, pp. 31–41, 2002.

or space complexity of combinatory counting is very large even unacceptable. For example, in a large dataset which cannot fit in the main memory, combinatory counting would incur considerable overheads. Even in cases in which the complexity of combinatory counting is manageable, there can also be reasons to consider the estimation methods. First, the given dataset can be viewed as a sample of some source distribution. So, even the exact counting just provides an approximation to the source distribution. On the other hand, in many application domains, the goals are finding the interesting patterns which satisfies some given thresholds to support the decision processes. Most of those thresholds are given by estimations or specified manually. Hence, the combinatory counting may not be worth the computation cost in these cases.

Since $P(\mathcal{A}|\mathcal{B}) = \frac{P(\mathcal{A},\mathcal{B})}{P(\mathcal{B})}$, knowing the joint probabilities is useful to infer the intrinsic relations between events such as associations [1], correlations [3], causalities [10] and multidimensional patterns [7]. Hence, it plays an important role in many data mining tasks. For example, frequent itemset mining in the discovery of association rules [1,4] can be thought as finding the set of items whose joint probabilities satisfy the given parameters. In the rest of the paper, we use D to denote the given $N \times M$ dataset, and $I = \mathcal{I}_1, \mathcal{I}_2, ..., \mathcal{I}_M$ be a set of events[1]. Each row, R_i, in the dataset, D, is referred as an observation (a record). An itemset with k items is called k-itemset.

In this paper we present two methods, *minAB* and *prodAB*, to estimate the joint probabilities without combinatory counting and explores their applications in data mining. Both methods are based on a structure called *partition support*. The main idea is to maintain the partition support information of items (or itemsets) over each logically disjoint partition and then use it to carry out the estimation. The rest of the paper is organized as follows: Section 2 introduces the basic concepts of *partition support*. Section 3 describes the *minAB* estimation methods and analyzes its properties. Section 4 presents *prodAB* estimation methods. Section 5 shows the performances of the two estimation methods on synthetic datasets. Section 6 gives a case study on using the estimation methods for fast association mining. Section 7 concludes and proposes our future work.

2 Partition Support

Definition 1. *The* **support count** $\lambda(S, D)$ *of itemset* S *in dataset* D *is the number of records in* D *containing* S*. If we logically divided the datasets* D *into disjoint partition* $D_1, ..., D_n, n \geq 1$*, the* **partition support** $PS(S, D, n)$ *of an itemset* S *over* D *is a* n*-tuple* $(\lambda(S, D_1), \lambda(S, D_2), ..., \lambda(S, D_n))$*.*

The support count of itemset S in D, $|\lambda(S, D)|$, is the sum of all the elements in $PS(S, D, n)$. The partition support $PS(S, D, n)$ is a structure consisting of the support counts of S in each partition. Figure 2 shows an example of the dataset. It has five items (A, B, C, E, F) and six records $(1, 2, 3, 4, 5, 6)$. Clearly,

[1] items or attributes.