International Journal of Pure and Applied Mathematics
Volume 83 No. 2 2013, 279-285
ISSN: 1311-8080 (printed version); ISSN: 1314-3395 (on-line version)
url: http://www.ijpam.eu
doi: http://dx.doi.org/10.12732/ijpam.v83i2.7

ALGORITHM FOR FINDING MAXIMAL FREQUENT SETS

Eungchun Cho

Division of Mathematics and Sciences
Kentucky State University
Frankfort, KY, 40601, USA

Abstract: Given a set X and a set $\mathcal{C}$ of subsets of X, subsets of X covered by k sets in $\mathcal{C}$ are called k-frequent. Frequent sets are of interest in large scale data analysis, pattern recognition and data mining. Characterization of maximal k-frequent sets in terms of equivalence relation and partial order is given. A general algorithm for finding maximal k-frequent sets, efficient for wide range of practical applications, is given.

AMS Subject Classification: 03E04, 06A07
Key Words: frequent set, association rules, similarity, pattern recognition, data mining

1. Introduction

Features shared by the members of a substantial subpopulation are of interest, for example, symptoms shared by various groups of patients are important objects of study. The patterns shared by natural or artificial disasters (flood, earthquake, war, riot, etc) are of great concern. Identifying patterns in large complex data is not easy. Exhaustive search is infeasible even for relatively small data.

Maximal k-frequent sets was introduced in [1], [2], [3], [4] and [5]. An iterative (greedy) algorithms for finding them are available (see [2], [3], and [4]. In this paper, we give a general algorithm for finding maximal k-frequent sets specially efficient in the presence of many equivalent (defined below) elements

Received: September 26, 2012

in X. In applications, $\mathcal{C}$ is a set of records, X the set of items in the records and a k-frequent set is the collection of items that appear in k records. Maximal k-frequent sets (defined below) will be characterized in terms of an equivalence relation on X and a partial ordering on equivalence classes. The algorithm for finding maximal k-frequent sets is given using matrix expression. Matlab code of the algorithm is available (send request to eung.cho@kysu.edu).

2. Notation and Definitions

Following notation and definitions will be used.

$$\begin{aligned} X &= \{x_1, \ldots, x_m\}, && (1)\\ \mathcal{P}(X) &= \text{The power set of } X, && (2)\\ \mathcal{C} &= \{S_1, \ldots, S_n\} \subseteq \mathcal{P}(X), && (3)\\ f(A) &= \{S_i \in \mathcal{C} : A \subseteq S_i\}, && (4)\\ F_k &= \{A \in \mathcal{P}(X) : n(f(A)) = k\}, && (5)\\ n(A) &= \text{ The cardinality of } A\ . && (6) \end{aligned}$$

We define maximal k-frequent sets, introduce equvalence relation on the data set X and a partial ordering on the equivalence classes.

Definition 1. f is a function on $\mathcal{P}(X)$ into $\mathcal{P}(\mathcal{C})$ that assigns $A \subseteq X$ the set of all $S_i \in \mathcal{C}$ that contain A:

$$f(A) = \{S_i \in \mathcal{C} : A \subseteq S_i\} \tag{7}$$

$f(\{x\})$ will be abbreviated as $f(x)$.

Definition 2. $A \in \mathcal{P}(X)$ is said to be k-frequent if $n(f(A)) = k$, i.e. there exist exactly k distinct subsets $S_i \in \mathcal{C}$ that contain A.

Definition 3. F_k is the set of all k-frequent sets.

$$F_k = \{A \in \mathcal{P}(X) : n(f(A)) = k\} \tag{8}$$

$A \in F_k$ is called a maximal k-frequent set if $A \subseteq B \in F_k$ implies $A = B$.

Equivalence relations on $\mathcal{P}(X)$ and X are introduced via f.

Definition 4. Let $A, B \in \mathcal{P}(X)$. Define

$$A \sim B \quad \text{iff} \quad f(A) = f(B) \tag{9}$$

The equivalence relation $\sim$ restricts to X as

$$x \sim y \quad \text{iff} \quad f(x) = f(y) \tag{10}$$

A partial order on $X/\sim$ is induced by $f|_X$ from the partial order $\subseteq$ on $\mathcal{P}(\mathcal{C})$,

Definition 5. A partial order $\leq$ on $X/\sim$ is defined by

$$[x] \leq [y] \quad \text{iff} \quad f(x) \subseteq f(y) \tag{11}$$

Let g be a function on $\mathcal{P}(\mathcal{C})$ into $\mathcal{P}(X)$ that assigns $B \in \mathcal{P}(\mathcal{C})$ the intersection of all $S_i \in B$.

$$g(B) = \bigcap_{S_i \in B} S_i \tag{12}$$

Then the following lemmas, easily proved, lead us to the algorithm.

Lemma 1. *fg and gf are idempotent, i.e.*

$$(gf)^2 = gf \quad \textit{and} \quad (fg)^2 = fg.$$

Proof. f reverses the inclusion, i.e. $A \subseteq B$ implies $f(B) \subseteq f(A)$. Let $A \in \mathcal{P}(X)$. Since $A \subseteq gf(A)$ $fgf(A) \subseteq f(A)$. To prove $f(A) \subseteq fgf(A)$, let $S \in f(A)$. Then $gf(A) \subseteq S$ and $S \in fgf(A)$. This proves $fgf = f$, from which $gfgf = gf$ follows. q.e.d.

Lemma 2. *$gf(A) = \bigcup_{B \sim A} B$ for every $A \in \mathcal{P}(X)$. It follows $gf(A)$ is the largest set in $[A]$.*

Proof. Since $fgf(A) = f(A)$, we have $gf(A) \sim A$ and $gf(A) \subseteq \bigcup_{B \sim A} B$. If $B \in [A]$ then $f(B) = f(A)$ and $B \subseteq gf(A)$. q.e.d.

Lemma 3. *$gf([x]) = \bigcup_{[x] \leq [y]} [y]$*

Proof. N$[x] \leq [y]$ implies $gf([y]) \subseteq gf([x])$. Since $[y] \subseteq gf([y])$ and $gf([y]) \subseteq gf([x])$, we have $[y] \subseteq gf([x])$. Since $fgf([x]) = f([x])$, we have $gf([x]) \sim [x]$ and $gf([x]) \subseteq \bigcup_{[x] \leq [y]} [y]$. q.e.d.

Lemma 4. *Let $A \in \mathcal{P}(X)$, $I = \{i : [x_i] \cap A \neq \emptyset\}$, $J = \{j : [x_i] \leq [x_j]$ for some $i \in I\}$, $B = \bigcup_{i \in I} [x_i]$ and $C = \bigcup_{j \in J} [x_j]$. Then $gf(A) = C$.*

Proof. By the definitions, $A \subseteq B \subseteq C$. We show $gf(A) = gf(B) = gf(C) = C$. Since $f(A) = f(B)$, $gf(A) = gf(B)$. If $j \in J$ then $f([x_j]) \subseteq f([x_i])$ for some $i \in I$, and $[x_j] \subseteq gf([x_i])$. This implies $C \subseteq gf(B)$. The other direction $gf(B) \subseteq C$ is obvious. q.e.d.

Now we make an observatioon: If $A \in \mathcal{P}(X)$ is maximal with respect to $\leq$ then $A = [x]$ for any $x \in A$. It is a maximal $n(f(x))$-frequent set and every nonempty subset of it is also $n(f(x))$-frequent. It is not true for other maximal k-frequent sets that are not maximal with respect to $'le$.

3. Matrix Expression

We give matrix expressions of the constructions used for the implementation of the algorithm.

Definition 6. M is the $m \times n$ matrix representing the incidence relation on X and $\mathcal{C}$:

$$M_{ij} = \begin{cases} 1 & \text{if } x_i \in S_j \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

Definition 7. A is the $m \times m$ matrix representing the equivalence relation on X:

$$A_{ij} = \begin{cases} 1 & \text{if } x_i \sim x_j \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

The equivalence relation on X corresponds to the equality of the rows of M, i.e.

$$A_{ij} = 1 \quad \text{iff} \quad M_{ik} = M_{jk}\ \forall k. \tag{15}$$

We note $n([x_i]) = \sum_j A_{ij}$.

Definition 8. Let $X/\sim = \{[x_1], \ldots, [x_r]\}$. N is the $r \times n$ matrix representing the incidence relation between elements of $X/\sim$ and the elements of $\mathcal{C}$:

$$N_{ij} = \begin{cases} 1 & \text{if } [x_i] \subseteq S_j \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

Definition 9. P is the $r \times r$ matrix representing the partial order on $X/\sim$:

$$P_{ij} = \begin{cases} 1 & \text{if } [x_i] \leq [x_j] \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

The partial order on $X/\sim$ corresponds to the inequality of the rows of N,

$$[x_i] \leq [x_j] \quad \text{iff} \quad N_{it} \leq N_{jt},\ \forall t. \tag{18}$$

We note $[x_i]$ is maximal iff $P_{ij} = \delta_{ij}$.

4. Algorithm

Maximal k-frquent sets with highest possible k's are the equivalence classes $[x_i]$ that are maximal with respect to $\leq$, namely those corresponding to the rows of P with row sum 1. It follows from the previous lemmas that maximal k-frequent sets are of the form $gf(A)$ for some $A \in F_k$. Thus, we find maximal frequent sets by taking

$$gf(x) = \bigcup_{[x] \leq [y]} [y]$$

for each equivalence class $[x] \in F_k$.

Algorithm.

1. Find all $[x_i] \in F_k$, i.e. find the indexing set I_k of F_k:
$$I_k = \{i : \sum_j N_{ij} = k\}.$$
2. For each $[x_i] \in F_k$ find all $[x_j] \geq [x_i]$, i.e. find the indexing set J_i of upper bounds of $[x_i]$:
$$J_i = \{j : P_{ij} = 1\}.$$
3. For each $[x_i] \in F_k$, take the union
$$\bigcup_{j \in J_i} [x_j] = \{x_p : A_{jp} = 1, \text{ for some } j \in J_i\},$$
which is the maximal k-frequent set containing x_i.

We note $\bigcup_{j \in J_i} [x_j] = gf(x_i)$ is $n([x_i])$-frequent and in general, the cardinality of the k-frequent sets get smaller as k increases. Obviously, maximal k-frequent sets of large cardinality for large k are of interest. The cardinality of a maximal k-frequent set is given in the following

Lemma 5. *Let $w = [w_1, \ldots, w_r]^t$ where $w_i = n([x_i]) = \sum_j A_{ij}$, the cradinality of $[x_i]$. Then the cardinality of the maximal k-frequent set $gf([x_i])$ containg $[x_i] \in F_k$ is $(Pw)(i)$:*

$$n(gf([x_i])) = \sum_{[x_i] \leq [x_j]} n([x_j]) = \sum_j P_{ij}\, w(j) \tag{19}$$

Proof. Since the maximal k-requent set containg $[x_i]$ is a disjoint union of equivalence classes greater than or equal to $[x_i]$, the cardinality of $gf([x_i])$ is the sum of w_j for which $[x_j] \geq [x_i]$. q.e.d.

5. Summary

A general algorithm for finding all maximal frequent sets is given. The algorithm decomposes the data set into equivalence classes and partially orders them. The algorithm can be extended (future work) to cases when the data set is more general, for example, weighted sets or lists with multiple entries of identical elements (a large number of clones in the data and records). Future work will also include non-deterministic sub-optimal but efficient algorithms combined with the ideas from topological data analysis that may be useful for big data.

Acknowledgments

Eungchun Cho's work at Seoul National University was supported by The Korea Research Foundation and The Korean Federation of Science and Technology Societies Grant funded by Korea Government (MOEHRD, Basic Research Promotion Fund).

References

[1] R. Agrawal, T. Imielinski, A. Swami, Mining association rules between sets of items in large databases, In: *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data* (Ed-s: P. Buneman, S. Jajodia), **22**, no. 2 of *SIGMOD Record*, ACM Press (1993), 207-216.

[2] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, A. Verkamo, Fast discovery of association rules, In: *Advances in Knowledge Discovery and Data Mining* (Ed-s: U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy), MIT Press (1996), 307-328.

[3] R. Agrawal, R. Srikant, Fast algorithms for mining association rules, In: *Proceedings 20-th International Conference on Very Large Data Bases* (Ed-s: J. Bocca, M. Jarke, C. Zaniolo), Morgan Kaufmann (1994), 487-499.

[4] S.E. Brossette, A.P. Sprague, Frequent Sets in the presence of clones: an example from medical serveillence, *Discrete Mathematical Problems with Medical Applications Series in Discrete Mathematics and Theoretical Computer Science*, **55**, AMS (1999), 165-170.

[5] E. Cho, Maximal frequent sets, *International Journal of Pure and Applied Mathematics*, **78**, No. 2 (2012), 245-251.