Frequent Itemset Mining (1)

Course Notes 3

Nadjib Lazaar
University of Paris-Saclay
lazaar@lisn.fr

1 Data Mining (DM) and Knowledge Discovery in Databases (KDD)

Data Mining (DM), often referred to as *Knowledge Discovery in Databases (KDD)*, is a multidisciplinary field that focuses on uncovering meaningful patterns, relationships, and insights from large and complex datasets. The ultimate goal of DM is to transform raw data into actionable and valuable knowledge that can be used for decision-making, prediction, and exploration.

1.1 Definition and Scope

DM and KDD encompass a broad range of activities that revolve around:

- **Investigation of Knowledge:** Analyzing data to identify hidden structures, trends, and relationships that are not immediately apparent.
- **Development of Processes:** Designing workflows and methodologies for systematically extracting useful information from data.
- **Algorithm Design:** Creating computational techniques and models to efficiently process and analyze large datasets.
- **Mechanisms for Knowledge Retrieval:** Implementing tools and systems that enable the discovery of potential knowledge from vast data collections.

1.2 Steps in the KDD Process

The KDD process involves several sequential steps to ensure meaningful and accurate knowledge extraction:

1. **Data Selection:** Identifying the relevant datasets for analysis, often from heterogeneous sources.

2. **Data Preprocessing:** Cleaning, transforming, and normalizing the data to ensure consistency and quality.

3. **Data Transformation:** Converting raw data into formats or features suitable for analysis.

4. **Data Mining:** Applying algorithms to extract patterns, associations, or predictive models.

5. **Evaluation and Interpretation:** Assessing the validity, relevance, and usefulness of the discovered knowledge.

6. **Knowledge Presentation:** Visualizing and communicating results in a form that stakeholders can readily understand and act upon.

1.3 Applications of Data Mining

DM has become an indispensable tool in various domains:

- **Business Intelligence:** Customer segmentation, fraud detection, market basket analysis, and sales forecasting.

- **Healthcare:** Disease prediction, drug discovery, and patient care optimization.

- **Scientific Research:** Analyzing experimental data in fields like genomics, astronomy, and materials science.

- **Social Media Analytics:** Sentiment analysis, trend detection, and user behavior modeling.

- **Cybersecurity:** Intrusion detection and anomaly detection in networks.

1.4 Challenges in DM and KDD

While DM has proven to be a powerful tool, it faces several challenges:

- **Data Complexity:** Handling high-dimensional, noisy, or incomplete data.

- **Scalability:** Processing and analyzing large-scale datasets efficiently.

- **Interpretability:** Ensuring that the patterns and models discovered are understandable to non-experts.

- **Ethical Concerns:** Addressing privacy, bias, and fairness in data-driven decision-making.

- **User Centering and Human-in-the-Loop:** Ensuring that human input is integrated into the data mining process, enabling user feedback to guide and refine algorithms. This challenge revolves around making sure that users' domain knowledge, insights, and experiences are effectively incorporated in the decision-making process, improving the overall accuracy and relevance of the discovered knowledge. Balancing automated processes with human intuition remains a significant hurdle in ensuring that data mining methods are both useful and aligned with real-world needs.

DM and KDD have evolved as critical components of modern data science, bridging the gap between raw data and actionable knowledge. By combining theoretical rigor with practical applications, they empower organizations and researchers to make informed, data-driven decisions.

2 Frequent Pattern Mining

Frequent Pattern Mining (FPM) has gained considerable attention due to its broad applicability and theoretical richness. Frequent Pattern Mining, originally introduced by Agrawal et al. [Agrawal et al., 1993], involves identifying and analyzing recurring patterns within datasets. These patterns reveal inherent structures and relationships in data, making them valuable for various domains such as market analysis, bioinformatics, and network security.

Among the many tasks encompassed by FPM, the discovery of frequent itemsets [Agrawal et al., 1994, Han et al., 2000] is foundational. Frequent itemsets represent sets of items that co-occur frequently in transactional datasets, serving as the basis for more complex analyses. The concept has been extended and refined over time, leading to notable advancements such as:

- **Closed Itemsets:** These are itemsets that are both frequent and maximal in terms of their frequency-preserving subsets [Pasquier et al., 1999, Zaki and Hsiao, 2002, Pei et al., 2000, Uno et al., 2004]. Closed itemsets minimize redundancy and serve as a compact representation of frequent patterns.

- **Association Rules:** Introduced by Agrawal et al. [Agrawal et al., 1994], association rules aim to uncover relationships between items in the form of "if-then" statements (e.g., "If a customer buys bread, they are likely to buy butter").

- **Rare Itemsets:** While most studies focus on frequently occurring patterns, rare itemsets [Szathmary et al., 2007, Adda et al., 2007] highlight less common but potentially insightful combinations of items, such as in fraud detection or rare disease analysis.

- **Sequence Patterns:** These patterns [Agrawal and Srikant, 1995] capture the order of occurrences in data, making them essential for applications like customer behavior prediction or genomic sequence analysis.

- **Emerging Patterns:** First introduced by Dong and Li [Dong and Li, 1999], emerging patterns identify trends or transitions in datasets, such as changes in purchasing habits or shifts in medical diagnoses over time.

Frequent Pattern Mining has evolved into a cornerstone of data mining research, with numerous algorithms and methodologies addressing its challenges. These include efficiency improvements for handling large-scale datasets, theoretical refinements such as anti-monotonicity properties, and adaptations for specific domains. The enduring relevance of this field lies in its ability to uncover hidden insights and provide actionable knowledge, making it indispensable in both academic and industrial contexts.

3 Preliminaries

3.1 Itemsets

Let $\mathcal{I} = \{p_1, \ldots, p_n\}$ be a set of n distinct objects, referred to as *items*. An *itemset* P is defined as a non-empty subset of $\mathcal{I}$. A *transactional dataset* $\mathcal{D}$ consists of a collection of m itemsets, denoted as $t_1, \ldots, t_m$, which are called *transactions*. The space of itemsets corresponds to the power set $2^{\mathcal{I}}$, where this search space is typically organized under a partial order based on the inclusion relation $\subseteq$.

The *cover* of an itemset P in $\mathcal{D}$, denoted as $cover(P)$, refers to the set of transactions in $\mathcal{D}$ that contain P. We define $\texttt{cover}(t, i)$ as a predicate that indicates whether item i is present in transaction t.

The *frequency* of an itemset P in $\mathcal{D}$, denoted as $\texttt{freq}(P)$, is the cardinality of its cover, i.e., $\texttt{freq}(P) = |\texttt{cover}(P)|$. The frequency of an itemset is an important metric in data mining. Given a frequency threshold α (also referred to as the *minimum frequency*), itemsets are classified as frequent if they appear in at least α transactions. Itemsets that appear in fewer than α transactions are considered infrequent (or rare).

Definition 1 (Frequent/Infrequent Itemset) *Given a dataset $\mathcal{D}$, a minimum frequency α and an itemset P. P is frequent if $freq(P) \geq \alpha$. Otherwise, P is infrequent.*

We denote by $\texttt{FIs}_\alpha$ and $\texttt{RIs}_\alpha$ the set of frequent and infrequent, respectively, w.r.t. a given minimum frequency α. Given a dataset $\mathcal{D}$ and a minimum frequency threshold α, an itemset in the language $2^{\mathcal{I}}$ is either frequent or infrequent (i.e., $\forall \alpha : \texttt{FIs}_\alpha \cup \texttt{RIs}_\alpha = 2^{\mathcal{I}}$).

The number of frequent/infrequent itemsets can be huge, making it hard even to print the result. Hence, we often reduce the problem of mining frequent or infrequent itemsets to the problem of mining the *borders* [Borgelt, 2012]. The *positive* border is the set of frequent itemsets with only infrequent supersets (Maximal Itemsets $\texttt{MaxIs}_\alpha$). The *negative* border is the set of infrequent itemsets with only frequent subsets (Minimal Itemsets $\texttt{MinIs}_\alpha$).

Figure 1: Transaction dataset example with five items and five transactions ($\mathcal{D}_1$).

trans.	Items				
t_1	A	B		D	E
t_2	A		C		
t_3	A	B	C		E
t_4		B	C		E
t_5	A	B	C		E

Definition 2 (Maximal/Minimal Itemset) *Given a dataset $\mathcal{D}$, a minimum frequency α and an itemset P:*

- *P is* maximal *iff P is frequent and there does not exist any* superset $Q \supseteq P$ *such that*
  $freq(Q) \geq \alpha$.
- *P is* minimal *iff P is infrequent and there does not exist any* subset $Q \subseteq P$ *such that*
  $freq(Q) < \alpha$.

The subsets of maximals and the supersets of minimals represent the frequent and infrequent itemsets, respectively. However, such borders do not preserve the knowledge on the support values. For instance, if an itemset P is present in half of the dataset, it can only be said that an itemset $Q \subset P$ is at least present in that half. To preserve the cover knowledge by identifying the exact cover to each itemset, other condensed representations have been proposed, namely, closed itemsets ($\mathtt{CIs}_\alpha$ set) for the positive side [Pasquier et al., 1999] and generator itemsets ($\mathtt{GIs}_\alpha$ set) for the negative side [Bastide et al., 2000b].

Definition 3 (Closed/Generator Itemset) *Given a dataset $\mathcal{D}$, a minimum frequency α and an itemset P:*

- *P is* closed *iff there does not exist any* superset $Q \supseteq P$ *such that*
  $freq(Q) =$
  $freq(P)$.
- *P is* generator *iff there does not exist any* subset $Q \subseteq P$ *such that*
  $freq(Q) =$
  $freq(P)$.

Note that a maximal (resp. a minimal) is a closed itemset (resp. a generator), but the reverse is not true. Every frequent itemset has a closed superset with the same cover:

$$\forall \alpha, \forall P \in \mathtt{FIs}_\alpha : \mathtt{cover}(P) = \max_{Q \in \mathtt{CIs}_\alpha, Q \subseteq P} \mathtt{cover}(Q)$$

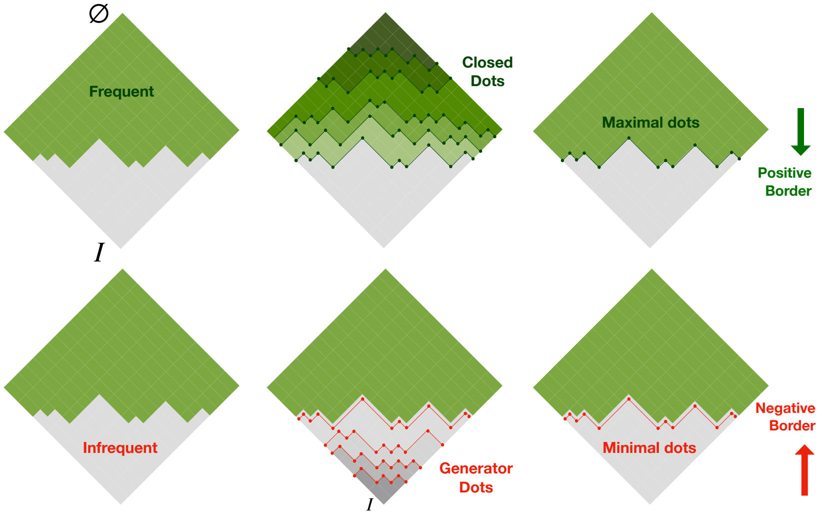

Figure 2: Frequent/Infrequent, Closed/Generator and Maximal/Minimal Itemsets.

We have the same property in the negative border with the infrequent itemsets, where every infrequent itemset has a generator subset with the same cover:

$$\forall \alpha, \forall P \in \texttt{RIs}_\alpha : \texttt{cover}(P) = \min_{Q \in \texttt{GIs}_\alpha, Q \supseteq P} \texttt{cover}(Q)$$

Inspired by the schematic Hasse diagram of Borgelt [Borgelt, 2012], figure 3.1 shows the set of frequent/closed/maximal itemsets (the green upper part of the lattice), as well as the infrequent/generator/minimal itemsets (the grey lower part of the lattice).

Example 1 *Consider the transaction dataset presented in Figure 1 and a minimum frequency $\alpha = 50\%$:*

- *AB is covered by three transactions where $\texttt{cover}(AB) = \{t_1, t_3, t_5\}$ and a (relative) frequency*
  $freq(AB) = 60\%$ and thus, AB is a frequent itemset
  $freq(AB) \geq \alpha$.
- *ABD is an infrequent itemset where*
  $freq(ABD) = 20\%$.
- *BCE is a maximal (any extension of BCE is an infrequent), but BE is not as $\texttt{freq}(ABE) \geq \alpha$.*
- *ACE is a minimal (subsets of ACE are frequent), but BE is not as $\texttt{freq}(BE) \geq \alpha$.*

- *ABE is closed (any extension of ABE has a lower frequency), but AB is not as $\texttt{freq}(AB) = \texttt{freq}(ABE)$.*

- *$ABCE$ is a generator (any subset of $ABCE$ has a larger frequency), but $ABDE$ is not as $\texttt{freq}(ABDE) = \texttt{freq}(D)$.*

♢

3.2 Association Rules

An association rule captures information of the kind *"if we have A and B, the chances to have C are high"*. An association rule is an implication of form $X \to Y$, where X and Y are itemsets such that $X \cap Y = \emptyset$ and $Y \neq \emptyset$. X represents the *body* of the rule and Y represents the *head*. The frequency of rule $X \to Y$ is the frequency of the itemset $X \cup Y$, that is,
$freq(X \to Y) =$
$freq(X \cup Y)$. The *confidence* of a rule captures how often Y occurs in transactions containing X, that is, $conf(X \to Y) = \frac{freq(X \to Y)}{freq(X)}$. A rule is known as a valid rule if its frequency and confidence are greater than or equal to user-specified thresholds (minimum frequency α and minimum confidence β). Given a dataset $\mathcal{D}$, a minimum frequency α and a minimum confidence β, the problem of mining association rules consists in generating all valid rules.

The user generally expects some *representative rules* according to a condensed representation [Kryszkiewicz, 1998]. The most common condensed representation of rules is the *minimal non-redundant association rules* (MNR) [Bastide et al., 2000a]. A rule is known as an MNR if there does not exist any other rule with the same frequency and the same confidence that is obtained by reducing the number of premises (removing items from the body) or by adding conclusions (adding items to the head).

Definition 4 (Minimal Non-redundant Rule (MNR)) *Given a dataset $\mathcal{D}$, a minimum frequency α and a minimum confidence β, a* minimal non-redundant association rule *$r : X \to Y$ is a valid rule such that there does not exist any rule $r' : X' \to Y'$ with $X' \subseteq X$, $Y \subseteq Y'$,*
$freq(r) =$
$freq(r')$, $conf(r) = conf(r')$, and $r \neq r'$.

Example 2 *Consider again the transaction dataset presented in Figure 1. With $\alpha = 60\%$ and $\beta = 70\%$, $C \to A$ is a valid association rule because*
$freq(C \to A) = 60\%$ and $conf(C \to A) = \frac{freq(C \to A)}{freq(C)} = 75\%$. $AB \to E$ is an MNR where a rule with a shorter body and/or a longer head will affect the original frequency/confidence. ♢

4 Anti-monotonicity and Apriori Property

In the context of Itemset mining, the concepts of anti-monotonicity and the Apriori property play a central role in efficiently mining frequent itemsets. These properties help to reduce the search space by pruning unpromising candidate itemsets.

4.1 Anti-monotonicity

An important property used in the mining of frequent itemsets is the anti-monotonicity property, which is defined as follows:

Definition 5 (Anti-monotonicity) *An itemset P is* anti-monotone *with respect to frequency if, for any itemset Q such that $Q \supseteq P$, the frequency of Q is always less than or equal to the frequency of P. That is, if P is frequent, all of its supersets must also be frequent. Formally, if $\texttt{freq}(P) \geq \alpha$, then for all $Q \supseteq P$, $\texttt{freq}(Q) \geq \alpha$.*

The anti-monotonicity property implies that if a set of items is infrequent, then any larger set that contains this set must also be infrequent. This helps prune the search space by eliminating unpromising supersets early in the process.

4.2 Apriori Property

The Apriori property is a direct consequence of anti-monotonicity. It states that if an itemset is frequent, then all of its subsets must also be frequent. In other words, a frequent itemset cannot contain an infrequent subset. This property significantly reduces the number of candidate itemsets to consider, which leads to more efficient mining.

Definition 6 (Apriori Property) *Let P be an itemset. If P is frequent, then all subsets of P must also be frequent. That is, for any subset S of P, if $\texttt{freq}(P) \geq \alpha$, then $\texttt{freq}(S) \geq \alpha$ for all $S \subseteq P$.*

The Apriori property allows for a level-wise search in the space of itemsets: starting with the smallest itemsets and gradually building larger ones by adding items. This ensures that only candidate itemsets that have frequent subsets are considered.

4.3 The Apriori Algorithm

The Apriori algorithm is one of the most well-known algorithms used to mine frequent itemsets [Agrawal et al., 1994]. It uses the Apriori property to efficiently generate candidate itemsets and prune those that are infrequent. The basic idea behind the Apriori algorithm is a breadth-first search where candidate itemsets are generated by joining frequent itemsets from the previous level, and infrequent itemsets are pruned using the anti-monotonicity property.

The algorithm works as follows:

1. **Initialization:** Start with the set of all frequent 1-itemsets, which are the items that appear in at least the minimum threshold of transactions.

2. **Iterative Process:**

   - Generate candidate itemsets of size k by joining frequent itemsets of size $k-1$.
   - For each candidate itemset, scan the dataset and calculate its frequency.
   - Prune candidates that are infrequent (i.e., their frequency is less than the minimum threshold α).

3. **Repeat the Process:** Repeat the process until no more frequent itemsets can be found.

4. **Generate Rules:** Once all frequent itemsets are found, generate association rules using the frequent itemsets.

Algorithm 1: Apriori Algorithm

```
Input: Transaction database $\mathcal{D}$, minimum support threshold $\alpha$
Output: Frequent itemsets
$k \leftarrow 1$ ;
$L_k \leftarrow \{p_i \mid p_i \in \mathcal{I} \wedge \texttt{freq}(p_i) \geq \alpha\}$ ;
while $L_k \neq \emptyset$ do
    $C \leftarrow \texttt{aprioriGen}(L_k)$ ;
    $k \leftarrow k + 1$ ;
    $L_k \leftarrow \{c \mid c \in C \wedge \texttt{freq}(c) \geq \alpha\}$ ;
return $\bigcup_i L_i$ ;

Function aprioriGen($L_k$):
    $E \leftarrow \emptyset$ ;
    for each pair of itemsets $P', P'' \in L_k$ such that
     $P' = \{p_{i_1}, \dots, p_{i_{k-1}}, p_{i_k}\}$ and $P'' = \{p_{i_1}, \dots, p_{i_{k-1}}, p_{i'_k}\}$ do
        if $p_{i_k} \neq p_{i'_k}$ then
            $P \leftarrow P' \cup P''$ ;
            if $\forall p_i \in P$, $P \setminus \{p_i\} \in L_k$ then
                $E \leftarrow E \cup \{P\}$ ;
    return $E$ ;
```

The key steps in the Apriori algorithm involve iteratively generating candidate itemsets, scanning the dataset to determine their frequencies, and pruning infrequent itemsets based on the Apriori property. This process is repeated until no more frequent itemsets can be found, after which association rules can be generated from the frequent itemsets.

References

[Adda et al., 2007] Adda, M., Wu, L., and Feng, Y. (2007). Rare itemset mining. In *Proceedings of ICMLA'07, Cincinnati, Ohio, USA*, pages 73–80. IEEE.

[Agrawal et al., 1993] Agrawal, R., Imielinski, T., and Swami, A. N. (1993). Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, Washington, DC, USA, May 26-28, 1993.*, pages 207–216.

[Agrawal and Srikant, 1995] Agrawal, R. and Srikant, R. (1995). Mining sequential patterns. In *Proceedings of ICDE'95, Taipei, Taiwan*, pages 3–14. IEEE.

[Agrawal et al., 1994] Agrawal, R., Srikant, R., et al. (1994). Fast algorithms for mining association rules. In *Proceedings of VLDB'94, Santigo de Chile, Chile*, volume 1215, pages 487–499.

[Bastide et al., 2000a] Bastide, Y., Pasquier, N., Taouil, R., Stumme, G., and Lakhal, L. (2000a). Mining minimal non-redundant association rules using frequent closed itemsets. In [Bastide et al., 2000a], pages 972–986.

[Bastide et al., 2000b] Bastide, Y., Taouil, R., Pasquier, N., Stumme, G., and Lakhal, L. (2000b). Mining frequent patterns with counting inference. *ACM SIGKDD Explorations Newsletter*, 2(2):66–75.

[Borgelt, 2012] Borgelt, C. (2012). Frequent item set mining. *WIREs Data Mining Knowl. Discov.*, 2(6):437–456.

[Dong and Li, 1999] Dong, G. and Li, J. (1999). Efficient mining of emerging patterns: Discovering trends and differences. In *Proceedings of KDD'99, San Diego, California, USA*, pages 43–52. Citeseer.

[Han et al., 2000] Han, J., Pei, J., and Yin, Y. (2000). Mining frequent patterns without candidate generation. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, May 16-18, 2000, Dallas, Texas, USA.*, pages 1–12.

[Kryszkiewicz, 1998] Kryszkiewicz, M. (1998). Representative association rules. In [Kryszkiewicz, 1998], pages 198–209.

[Pasquier et al., 1999] Pasquier, N., Bastide, Y., Taouil, R., and Lakhal, L. (1999). Efficient mining of association rules using closed itemset lattices. *Inf. Syst.*, 24(1):25–46.

[Pei et al., 2000] Pei, J., Han, J., and Mao, R. (2000). CLOSET: an efficient algorithm for mining frequent closed itemsets. In *SIGMOD Workshop on Data Mining and Knowledge Discovery*, pages 21–30.

[Szathmary et al., 2007] Szathmary, L., Napoli, A., and Valtchev, P. (2007). Towards rare itemset mining. In *Proceedings of ICTAI'07, Patras, Greece*, volume 1, pages 305–312. IEEE.

[Uno et al., 2004] Uno, T., Kiyomi, M., and Arimura, H. (2004). LCM ver. 2: Efficient mining algorithms for frequent/closed/maximal itemsets. In *FIMI '04, Proceedings of the IEEE ICDM Workshop on Frequent Itemset Mining Implementations, Brighton, UK, November 1, 2004*.

[Zaki and Hsiao, 2002] Zaki, M. J. and Hsiao, C. (2002). CHARM: an efficient algorithm for closed itemset mining. In *Proceedings of the Second SIAM International Conference on Data Mining, Arlington, VA, USA, April 11-13, 2002*, pages 457–473. SIAM.