Università degli Studi di Padova

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE
Corso di Laurea Magistrale in Ingegneria Informatica

Tesi di Laurea

On the use of the Rademacher complexity in mining sequential patterns

Candidato:
Diego Santoro

Relatore:
Prof. Fabio Vandin

Anno Accademico 2018–2019

Abstract

A *sequential pattern* is a sequence of sets of items. Mining sequential patterns from very large datasets is a fundamental problem in data mining. The *Rademacher complexity*, a key concept of statistical learning theory, is a measure of the expressiveness of a set of real-valued functions. This thesis formally proves the first rigorous and efficiently computable bound on the Rademacher complexity of sequential patterns. This result is then applied to two key tasks in mining sequential patterns. First, it is used to develop an efficient progressive sampling algorithm for mining *frequent* sequential patterns, which are sequential patterns that appear in fraction at least θ of the transactions of a dataset, where θ is a parameter provided by the user. Second, the Rademacher complexity is used to design an efficient algorithm for mining *true frequent sequential patterns*, which are sequential patterns that appear with probability at least γ in a transaction from an unknown generative process, by analyzing a sample generated by the process.

This thesis is dedicated to my family

Acknowledgements

Five university years of passion and dedication to study are gone. My immense gratitude goes to my family: Antonella, Marcello, and Fabio. You have been crucial for overcoming many difficulties. I cannot forget my study partner, Gaetano, my cat. You have been keeping me company for several very long days.

Special thanks also go to professor Fabio Vandin, which has become the reference point of my academic life over these years. Thank you.

Contents

Chapter 1

Introduction

The common definition of *data mining* is the *discovery of models for data*, where, thinking in an algorithmic manner, a model of the data can be viewed as *the answer to a complex query about it* [6]. A very common query is the extraction of patterns that frequently appear in a given dataset. Exact algorithms for this issue require multiple scanning of the dataset, which become impractical for massive datasets. Thus, a research direction that has drawn a lot of interest is to find a high-quality approximation of the set of frequent patterns. In this thesis we use the framework provided by the *statistical learning theory* for gaining meaningful information from data, where the observations (i.e., data organized in a dataset) are assumed to be generated independently from the same probability distribution on the the universe of patterns (i.e., observations are i.i.d).

An *itemset* is a set of items and a *sequential pattern* is a sequence of itemsets. A *sequential dataset* T is a bag of *transactions*, which are sequential patterns. In e-commerce scenario, itemsets can be thought as sets of purchased objects, or on-line purchases, and a sequential pattern can be considered as a sequence of on-line purchases of a customer during a time period. In a sequential dataset each transaction is a sequence of on-line purchases associated to a customer.

The task of *frequent sequential pattern mining* from a given dataset T consist in extracting all sequential patterns that appear in at least θ transactions of T, where θ is a given frequency threshold. For the e-commerce scenario this means to find all sequences of itemsets that are frequently purchased by customers in T. Exact algorithms have been developed for solving this issue, however they become impractical for very large dataset. Instead

of searching for the exact set of frequent sequential patterns, we can extract its approximation using a *sampling technique*. This approach consist in using a *sample* S, i.e., a small subset of transactions of T, in order to mine from S an high-quality approximation of the exact set of frequent sequential patterns of T. This can be obtained using a *progressive sampling* approach, which uses a sequence of samples of T of progressively increasing size until a suitable *stopping condition* is verified. A key challenge with this approach is to derive a rigorous bound on the number of samples needed to extract rigorous approximations of the set of frequent sequential patterns.

Now let consider the dataset T as a sample of transactions independently drawn from a probability distribution π on the universe of sequential patterns. The task of *true frequent sequential pattern mining* from an unknown generative process consist in finding all sequential patterns that are frequently generated from π. Again, for the e-commerce scenario, this means to find all sequences of itemsets that are frequently purchased by the entire population of customers and not just by customers of a given dataset. Current approaches for mining frequent sequential patterns completely ignore the fact that the dataset is obtained from a generative process.

In this thesis we propose

- the first algorithm based on a (progressive) sampling approach for mining frequent sequential patterns from a given dataset, and
- the first algorithm for mining true frequent sequential patterns from an unknown generative process

using the *Rademacher complexity*, a key concept of statistical learning theory which represents a measure of the expressiveness of a set of real-valued functions [10, 5]. These two algorithms are based on another key contribution of this thesis: the first rigorous and efficiently computable bound on the Rademacher complexity of sequential patterns.

1.1 Related work

The problem of frequent sequential pattern mining has been introduced in [2]. Several exact algorithms [4, 11] have been designed to extract the set of frequent sequential patterns. ProSecCo [9] represents the first algorithm

for mining high-quality approximations of frequent sequential patterns from a given dataset. It is based on another key concept of statistical learning theory, the *VC-dimension*. However, ProSecCo is not a sampling algorithm, since it progressively processes the dataset in blocks. When a new block is processed, it outputs a more accurate approximation set of frequent sequential patterns. After the last block has been analyzed, ProSecCo returns the exact collection of frequent sequential patterns, since the entire dataset is processed. The VC-dimension is also used in [8] for mining the set of true frequent itemsets from an unknown generative process. The authors formally define the *true frequent itemset mining problem* and develop and analyze an algorithm to solve it. They identify a frequency threshold $\overline{\omega}$ such that, with high probability, all itemsets that appear in at least $\overline{\omega}$ transactions of a given dataset have probability to appear in a transaction sampled from a distribution π at least ω, for a given threshold ω. In [7] the authors proposed an upper bound on the Rademacher complexity of itemsets, which is used for developing a progressive sampling algorithm for extracting an approximation of the set of frequent itemsets from a given dataset. We describe this algorithm more in detail in Chapter 4.

Looking at the state of the art in the sequential pattern mining scenario, there is no upper bound on the Rademacher complexity of sequential patterns. Moreover, there are not a progressive sampling algorithm for solving the frequent sequential pattern mining problem and an algorithm for solving the true frequent sequential pattern mining problem. The objective of this thesis is to close this gap.

1.2 Contributions

The following are the contributions of this thesis to the state of the art:

- a different (and more accurate) analysis of the algorithm for mining frequent itemsets presented in [7];
- an improvement of the computation of an upper bound to the Rademacher complexity of itemsets proposed in [7];
- the first rigorous and efficiently computable upper bound for the Rademacher complexity of sequential patterns;

- the first rigorous progressive sampling-based algorithm for mining frequent sequential patterns from a given dataset;
- the first rigorous algorithm for mining true frequent sequential patterns from an unknown generative process.

1.3 Outline

Chapter 2 introduces some preliminary concepts such as patterns (itemsets and sequential patterns), the frequent pattern mining problem, and a progressive sampling technique to solve it. Chapter 3 presents the Rademacher complexity and some theoretical results about it. In Chapter 4 we describe the progressive sampling approach for mining frequent itemsets presented in [7], which is the basis for our progressive sampling algorithm for sequential pattern mining. Here we propose a different analysis of the algorithm and an improvement of the computation of an upper bound to the Rademacher complexity of itemsets compared to what has been done by the authors. In Chapter 5 we formally prove the first rigorous and efficiently computable upper bound for the Rademacher complexity of sequential patterns. In Chapter 6 we use this result for developing an algorithm for mining frequent sequential patterns using the progressive sampling technique and an algorithm for mining true frequent sequential patterns. At the end, in Chapter 7 there are some final considerations and future works.

Chapter 2

Preliminaries

In this chapter we introduce some preliminary definitions.

2.1 Itemsets

Let $I = \{i_1, i_2, \ldots, i_d\}$ be a set of d items for which there is a total ordering, and let a *transaction* t be a subset of I: $t \subseteq I$. Let $T = \{t_1, t_2, \ldots, t_N\}$ be a bag of N transactions called *transactional dataset*, or simply *dataset*, over I. An *itemset* X is a set of items from I, $X \subseteq I$, and its size $|X|$ is the number of items in it. Note that transactions of T are itemsets. Let consider an arbitrary transaction $t \in T$: we say that an itemset X *appears* in t, and t *contains* X, if $X \subseteq t$. Thanks to the total ordering of items, transactions and itemsets can be represented as sorted vectors. Given an itemset X, we define its *support set* $T_X \subseteq T$ as the subset of transactions of T that contain X, and its *support* $Supp_T(X) = |T_X|/N$ as the fraction of transactions of T that contain X. Given a *support threshold* $\theta \in (0, 1]$, the set $FI(T, \theta)$ represents all itemsets with support at least θ, i.e., the set of all frequent itemsets (and their supports), in T with respect to (w.r.t.) θ:

$$FI(T, \theta) = \{(X, Supp_T(X)) : X \subseteq I \wedge Supp_T(X) \geq \theta\}.$$

A key property of itemset support is the *anti-monotonicity support property*: given two itemsets $X, Y \subseteq I$,

$$X \subseteq Y \implies Supp_T(X) \geq Supp_T(Y).$$

This property implies the following two consequences (w.r.t. a given support threshold):

1. itemset X is frequent $\implies \forall Y \subseteq X$, Y is frequent;

2. itemset X is not frequent $\implies \forall Y \supseteq X$, Y is not frequent.

In words: if X is frequent, each of its subsets is frequent and if X is not frequent, each of its supersets is not frequent.

Note that, given two itemsets $X, Y \subseteq I$, if $X \subset Y$ and X,Y have the same support, then it makes sense to report Y only. In order to define this kind of lossless succint representation of the set of frequent itemsets, it is necessary to introduce the concept of *closed itemset*. An itemset $X \subseteq I$ is *closed* w.r.t. T (i.e., X is a *closed* itemset) if for each superset $Y \supset X$ we have $Supp_T(Y) < Supp_T(X)$, or, equivalently, none of its supersets has support equal to $Supp_T(X)$. Let $CI(T) = \{X \subseteq I : X \text{ is closed w.r.t. } T\}$ be the set of closed itemsets w.r.t. T, and $CFI(T, \theta) = \{X \in CI(T) : Supp_T(X) \geq \theta\}$ be the set of frequent closed itemsets w.r.t. T and θ. $CFI(T, \theta)$ provides a succint representation of $FI(T, \theta)$: the set of all subsets of the frequent closed itemsets coincides with the set of all frequent itemsets. In addition, the representation provided by $CFI(T, \theta)$ is lossless: the support of any frequent itemset X can be extrapolated by taking the maximum support over all frequent closed itemsets that are supersets of X, i.e., $Supp_T(X) = \max\{Supp_T(Y) : Y \supseteq X \wedge Y \in CFI(T, \theta)\}$.

2.2 Sequential patterns

A *sequential pattern* (or *sequence*) $\mathbf{x} = \langle X_1, X_2, \ldots, X_\ell \rangle$ is a finite ordered list of itemsets. Let U be the (infinite) universe of sequences that we can build using I. The number of itemsets of a sequence $\mathbf{x}$ is its *length* $|\mathbf{x}|$, i.e., $|\mathbf{x}| = \ell$. The *item-length* $\|\mathbf{x}\|$ of a sequence $\mathbf{x}$ is the sum of the sizes of the $|\mathbf{x}|$ itemsets that occur in it:

$$\|\mathbf{x}\| = \sum_{i=1}^{|\mathbf{x}|} |X_i|.$$

Let $\mathbf{x} = \langle X_1, X_2, \ldots, X_\ell \rangle$ and $\mathbf{y} = \langle Y_1, Y_2, \ldots, Y_m \rangle$ be two sequences. We say that $\mathbf{x}$ is a *subsequence* of $\mathbf{y}$, or $\mathbf{y}$ is a *super-sequence* of $\mathbf{x}$, denoted by $\mathbf{x} \sqsubseteq \mathbf{y}$, if there exists an increasing sequence of indexes $1 \leq i_1 < i_2 < \cdots < i_\ell \leq m$ such that $X_1 \subseteq Y_{i_1}$, $X_2 \subseteq Y_{i_2}$, $\ldots$, $X_\ell \subseteq Y_{i_\ell}$.

In a sequential pattern scenario, a *(sequential) dataset* $T = \{t_1, t_2, \ldots, t_N\}$ is a bag of N sequences. Thus, transaction t_i is a sequence, for $1 \leq i \leq N$. Given a sequence $\mathbf{x}$, we define its *support set* $T_{\mathbf{x}}$ as the bag of transactions where $\mathbf{x}$ appears, i.e., $T_{\mathbf{x}} = \{t \in T \ : \ \mathbf{x} \sqsubseteq t\}$. The *support* $Supp_T(\mathbf{x})$ of $\mathbf{x}$ in T is the fraction of transactions that contain $\mathbf{x}$, i.e., $Supp_T(\mathbf{x}) = |T_{\mathbf{x}}|/N$. The anti-monotonicity support property for support sequences follows: if two sequences $x, y \in U$ are such that $x \sqsubseteq y$, then $Supp_T(x) \geq Supp_T(y)$. Given a *support threshold* $\theta \in (0, 1]$, the set $FS(T, \theta)$ represents all sequences with support at least θ, i.e., the set of all frequent sequences (and their supports) in T w.r.t. θ:

$$FS(T, \theta) = \{(\mathbf{x}, Supp_T(\mathbf{x})) : \mathbf{x} \in U \wedge Supp_T(\mathbf{x}) \geq \theta\}.$$

A sequence $\mathbf{x}$ is *closed* with respect to (w.r.t.) T if for each of its super-sequences $\mathbf{y} \sqsupset \mathbf{x}$ we have $Supp_T(\mathbf{y}) < Supp_T(\mathbf{x})$, or, equivalently, none of its super-sequence has support equal to $Supp_T(\mathbf{x})$. Let $CS(T)$ be the set of all closed sequences in T. The set $CFS(T, \theta)$ is made of all frequent sequences of $CS(T)$.

2.3 Frequent pattern mining problem

Let consider a generic *pattern* p be either an itemset or a sequence (sequential pattern) and P be the universe of all patterns. Given two patterns p_1,p_2 (both itemsets or both sequences) we say that p_1 is a sub-pattern of p_2, denoted with $p_1 \Subset p_2$, when $p_1 \subseteq p_2$ if we are in the itemset scenario, or $p_1 \sqsubseteq p_2$ in the sequence scenario. The frequent pattern mining problem follows: given a set of items I, a support threshold θ and a dataset T, we are interested in finding the set of frequent patterns in T w.r.t. θ, i.e., $FP(T, \theta)$. Note that, in itemset scenario, P is the power set of I, transactions of T are itemsets and $FP(T, \theta) = FI(T, \theta)$, instead, in sequence scenario, $P = U$, transactions of T are sequences and $FP(T, \theta) = FS(T, \theta)$. Exact algorithms for this issue require access to the entire dataset T, thus for big data purposes (i.e., T is very large) they become impractical and one can use the following *sampling approach*.

2.3.1 Progressive sampling approach

The idea of the sampling approach is the following: considering only a small sample of T, mine from it a set of frequent patterns, showing that it is a good approximation of the set of frequent patterns (and their supports) $FP(T, \theta)$ w.r.t. T and θ.

We have to define more precisely the concept of *sample* of T and the meaning of a *good approximation* of $FP(T, \theta)$. Let $S \subseteq T$ be a sample drawn at random with uniform probability and with replacement. Note that in a given dataset it is possible to find sets of identical transactions. The draw is made with replacement, so a specific transaction of T can appear multiple times in S. In order to define a good approximation of $FP(T, \theta)$, we need the following definition:

Definition 1. *(Def.1 of [7] for itemsets, Def.1 of [9] for sequences)*
Let $\epsilon \in (0, 1)$ be the accuracy parameter, T be a dataset over a set of items I and $\theta \in (0, 1]$ be a support threshold. A set $C = \{(p, s_p) : p \in P, s_p \in (0, 1]\}$ is an ϵ-approximation *to $FP(T, \theta)$ if the following conditions hold:*

1. *For each $(p, Supp_T(p)) \in FP(T, \theta)$ there exists a pair $(p, s_p) \in C$;*
2. *For each $(p, s_p) \in C$, $Supp_T(p) \geq \theta - \epsilon$;*
3. *For each $(p, s_p) \in C$, $|Supp_T(p) - s_p| \leq \epsilon/2$.*

From the previous definition we can observe some facts. Condition 1 tells us that C contains all frequent patterns (and their supports) of T with eventually some false positives, but there are no false negatives. Condition 2 ensures that any pattern contained in the approximation set C has support on T that could be lower than θ, but in this case it is not too far from θ (in particular, within a gap of ϵ). Finally, from condition 3 we have that for each pattern p such that $(p, s_p) \in C$, s_p is a good estimate of the support $Supp_T(p)$ (in fact $s_p - \epsilon/2 \leq Supp_T(p) \leq s_p + \epsilon/2$).

Let $\delta \in (0, 1)$ be a *confidence parameter*. The following procedure returns an *ϵ-approximation* to $FP(T, \theta)$ with probability (w.p.) at least $1 - \delta$ when Lemma 1 below is satisfied (consider in Lemma 1 $\delta_i = \delta$, $S_i = S$):

- let S, as previously mentioned, be a sample of T drawn at random with uniform probability and with replacement;

- return $C = \{(p, s_p = Supp_S(p)) : p \in FP(S, \theta')\}$, where $\theta' < \theta$ is a properly lower support threshold and $FP(S, \theta')$ is the set of frequent patterns w.r.t. the sample S and θ'.

M. Riondato and E. Upfal [7] used the *progressive sampling approach* for mining an ϵ-approximation to the set of frequent itemsets $FI(T, \theta)$ with high probability. The same approach could be used in sequential pattern scenario. Algorithm 1 represents the pseudocode of this approach applied to a generic a pattern.

Algorithm 1: Progressive sampling approach

Data: : a dataset T built on alphabet I, parameters θ,ϵ,$\delta \in (0, 1)$, a sampling schedule $(|S_i|)_{i \geq 1}$ of increasing sample sizes
Result: an ϵ-*approximation* to $FP(T, \theta)$ w. p. $\geq 1 - \delta$
$i \leftarrow 0$;
do
  $i \leftarrow i + 1$;
  if $|S_i| \geq |T|$ **then return** $FP(T, \theta)$;
  $S_i \leftarrow$ sample of T of some predefined size $|S_i|$;
while *stopping condition* is not satisfied;
$\theta' \leftarrow \theta - \epsilon/2$;
return $FP(S_i, \theta')$;

A progressive sampling approach uses a sequence of samples of T of progressively increasing size. Let $\delta \in (0, 1)$ be the confidence parameter. Our goal is to mine an ϵ-approximation to $FP(T, \theta)$ w. p. at least $1 - \delta$. If we take into account a generic iteration i of the algorithm, let S_i and $\delta_i = \delta/2^i$ be respectively the sample and the confidence parameter for iteration i. At the end of iteration i we check a *stopping condition*, in order to establish if it is possible to extract an ϵ-*approximation* to $FP(T, \theta)$ from S_i w. p. at least $1 - \delta_i$. Is this not the case, i is increased and we repeat this process for S_{i+1}, otherwise we return $FP(S_i, \theta')$, where $\theta' = \theta - \epsilon/2$ as justified by Lemma 1. However, when $|S_i| \geq |T|$ we stop the procedure and return the set $FP(T, \theta)$.

Lemma 1. *(Lemma 1 of [7] for itemsets, Lemma 2 of [9] for sequences)* *Let ϵ,δ,$\theta \in (0, 1)$. For $i \in \mathbb{N}$, $i \geq 1$, let S_i be a sample of a dataset T. Let*

us consider the event

$$E_{S_i} : \text{“}|Supp_T(p) - Supp_{S_i}(p)| \leq \epsilon/2 \ \forall p \in P\text{”}.$$

If

$$P(E_{S_i}) \geq 1 - \delta_i$$

then $FP(S_i, \theta')$ *is an* ϵ*-approximation to* $FP(T, \theta)$ *w. p. at least* $1-\delta_i$*, where* $\theta' = \theta - \epsilon/2$ *and* $\delta_i = \delta/2^i$*.*

Proof. In order to prove this lemma, assuming that the event E_{S_i} is verified with probability at least $1-\delta_i$, we show that the three points of Definition 1, with $C = FP(S_i, \theta')$, are satisfied. First, by $|Supp_T(p) - Supp_{S_i}(p)| \leq \epsilon/2 \ \forall p \in P$, we have that $\forall p \in P$:

- (a) $Supp_{S_i}(p) - \epsilon/2 \leq Supp_T(p) \leq Supp_{S_i}(p) + \epsilon/2$, and
- (b) $Supp_T(p) - \epsilon/2 \leq Supp_{S_i}(p) \leq Supp_T(p) + \epsilon/2$

Now:

1. For each $(p, Supp_T(p)) \in FP(T, \theta)$, we have $Supp_T(p) \geq \theta$. Thus, by (b) we have $Supp_{S_i}(p) \geq Supp_T(p) - \epsilon/2 \geq \theta - \epsilon/2 = \theta'$ and this implies that the pair $(p, Supp_{S_i}(p))$ belongs to $FP(S_i, \theta')$. The point 1 of Definition 1 is verified;

2. For each $(p, Supp_{S_i}(p)) \in FP(S_i, \theta')$, we have $Supp_{S_i}(p) \geq \theta'$. By using (a) we have $Supp_T(p) \geq Supp_{S_i}(p) - \epsilon/2 \geq \theta' - \epsilon/2 = \theta - \epsilon$. Thus, $Supp_T(p) \geq \theta - \epsilon$ and the point 2 of Definition 1 is satisfied;

3. For each $(p, Supp_{S_i}(p)) \in FP(S_i, \theta')$, by hypothesis $|Supp_T(p) - Supp_{S_i}(p)| \leq \epsilon/2$. Thus, the point 3 of Definition 1 holds.

Since every point of Def.1 is satisfied for $C = FP(S_i, \theta')$, $FP(S_i, \theta')$ is an ϵ*-approximation* to $FP(T, \theta)$ w. p. at least $1 - \delta_i$. □

In the next chapters we will define a stopping condition with an efficient procedure to check it. If we consider every iteration of Algorithm 1 and not only a generic one, the theoretical guarantee achieved by this approach is that the output of the algorithm is an ϵ-approximation to $FP(T, \theta)$ w. p. at least $1 - \delta$, as stated by the proof of correctness of Theorem 1.

Theorem 1. *The algorithm returns an* ϵ-approximation *to* $FP(T, \theta)$ *with probability at least* $1 - \delta$.

Proof. Let E_i be the event "at iteration i, for S_i and $\forall$ pattern $p \in P$ it holds that $|Supp_T(p) - Supp_{S_i}(p)| \leq \eta_i$ ", where η_i is a quantity related to the stopping condition that we will define in the following chapters. Thus,

$$P(\textit{output is an } \epsilon\textit{-approximation}) \geq P\left(\bigcap_{i=1}^{+\infty} E_i\right) =$$

$$= 1 - P\left(\bigcup_{i=1}^{+\infty} \bar{E}_i\right) \geq 1 - \delta,$$

since

$$P\left(\bigcup_{i=1}^{+\infty} \bar{E}_i\right) \overset{\text{u.b.}}{\leq} \sum_{i=1}^{+\infty} P(\bar{E}_i) \leq \sum_{i=1}^{+\infty} \frac{\delta}{2^i} = \delta,$$

by using the union bound (u.b.). Note that, by adopting a progressive sampling strategy, we do not know a priori when the stopping condition is verified. This implies that the event E_i must be satisfied with probability at least $1 - \delta_i$ at each iteration i, stopping the procedure the first time that $\eta_i \leq \epsilon/2$. Thus, we obtain an ϵ-*approximation* to $FP(T, \theta)$ w. p. at least $1 - \delta$ by mining $FP(S_i, \theta')$, where $\theta' = \theta - \epsilon/2$. □

Note that if δ_i's were set to δ for every iteration i, then we would have

$$P(\textit{output is an } \epsilon\textit{-approx.}) \geq 0 \tag{2.1}$$

which does not give us any theoretical guarantees.

In this chapter we presented some basic notions. The key point is that if, with high probability, the maximum difference between the support on the dataset T of a pattern p and its support on a sample S_i is lower or equal than $\epsilon/2$, i.e.

$$\sup_{p \in P} |Supp_T(p) - Supp_{S_i}(p)| \leq \epsilon/2 \tag{2.2}$$

then we can extract an ϵ-approximation to $FP(T, \theta)$ from S_i with high probability.

Chapter 3

The Rademacher complexity and its use in pattern mining

Checking the condition $\sup_{p \in P} |Supp_T(p) - Supp_{S_i}(p)| \leq \epsilon/2$ is computational expensive because it requires to mine all patterns that appear in a sample S_i of T. In this chapter we go through some theoretical results related to the Rademacher average, which is a crucial concept of statistical learning theory [10, 5]. These results will be useful for the next chapters, where we define a quantity η_i, easy to compute, which is an upper bound to $\sup_{p \in P} |Supp_T(p) - Supp_{S_i}(p)|$ with high probability. This implies that when $\eta_i \leq \epsilon/2$, we can extract an ϵ-approximation to $FP(T, \theta)$ from S_i with high probability.

Note that the concepts and results presented in this chapter have been introduced by [7] for itemsets, while this thesis is the first time they are introduced for sequential patterns.

We define, for each pattern $p \in P$, the *indicator function* $\phi_p : P \to \{0, 1\}$ as

$$\phi_p(t) = \begin{cases} 1 & if\ p \Subset t \\ 0 & otherwise \end{cases}$$

where t is a transaction. If we consider t as a transaction of a dataset T ($|T| = N$), $\phi_p(t)$ is 1 if p appears in t, otherwise it is 0. The support of p in T can be defined using the indicator function ϕ_p:

$$Supp_T(p) = \frac{1}{N} \sum_{t \in T} \phi_p(t).$$

The same consideration can be done for

$$Supp_S(p) = \frac{1}{n} \sum_{t \in S} \phi_p(t),$$

where S is a sample of T with $|S| = n$. For each transaction $t_i \in S$, $1 \leq i \leq n$, let σ_i be a Rademacher random variable which takes value 1 or -1, each with probability 1/2 and the σ_i's are independent. The *(sample) conditional Rademacher average* R_S is defined as:

$$R_S = \mathbb{E}_\sigma \left[\sup_{p \in P} \frac{1}{n} \sum_{i=1}^{n} \sigma_i \phi_p(t_i) \right],$$

where the expectation is taken w.r.t. the Rademacher random variables σ_i, i.e., conditionally on the sample S. The naïve computation of the exact value of R_S is expensive since it requires to mine all patterns from S and to generate all possible 2^n combination values of the Rademacher variables for the computation of the expectation.

Now, let us recall some basic concepts of machine learning theory from [10]. Let $\mathcal{X}$ be a domain set, $\mathcal{Y}$ be a label set, $\mathcal{D}$ be a probability distribution over $\mathcal{X} \times \mathcal{Y}$ (not known to the learner), and h be a prediction rule $h : \mathcal{X} \to \mathcal{Y}$. Let H be a set of prediction rules. The generalization error $L_{\mathcal{D}}(h)$ follows:

$$L_{\mathcal{D}}(h) = P_{(x,y) \backsim \mathcal{D}}[h(x) \neq y].$$

Let $\mathcal{S}$ be a finite sequence of pairs (x, y) from $\mathcal{X} \times \mathcal{Y}$, the so called *training set* (known to the learner). The generalization error $L_{\mathcal{S}}(h)$ follows:

$$L_{\mathcal{S}}(h) = \frac{|\{i, 1 \leq i \leq |\mathcal{S}| : h(x_i) \neq y_i\}|}{|\mathcal{S}|}.$$

The following theorem is a key result from statistical learning theory (Theorem 3.2 from [3]):

Theorem 2. *With probability at least* $1 - \delta$*:*

$$\sup_{h \in H} |L_{\mathcal{D}}(h) - L_{\mathcal{S}}(h)| \leq 2R_S + \sqrt{\frac{2 \ln (2/\delta)}{n}}.$$

The following fundamental theorem connects (2.2) with the Rademacher average:

Theorem 3. *With probability at least* $1-\delta$*:*

$$\sup_{p\in P} |Supp_T(p) - Supp_S(p)| \leq 2R_S + \sqrt{\frac{2\ln(2/\delta)}{n}}. \tag{3.1}$$

Proof. The proof requires just an interpretation of $Supp_T(p)$ and $Supp_S(p)$ as generalization and empirical measure, respectively. Let us associate a very large dataset T (but with finite size) with the probability distribution $\mathcal{D}$ and let t be a transaction drawn uniformly at random from T. If we consider a pattern $p \in P$, the true support $Supp_T(p)$ of p on T can be referred to as the probability that the transaction t contains p

$$Supp_T(p) = \frac{|\{t \in T : t \textit{ contains } p\}|}{|T|} = \mathbb{E}\left[\phi_p(t)\right] = P_{t\backsim T}[\phi_p(t) = 1],$$

that can be considered as a *generalization measure*, where $\phi_p(t)$ represents a Bernoulli random variable. Now, let us associate a sample S of T to the training set $\mathcal{S}$. Taking into account a pattern $p \in P$, the support $Supp_S(p)$ of p on S is just the fraction of transactions of S which contain p:

$$Supp_S(p) = \frac{1}{|S|}\sum_{t\in S} \mathbb{1}[t \textit{ contains } p],$$

which can be seen as an *empirical measure*, where $\mathbb{1}$ denotes the indicator function. Thus, the bound holds from Theorem 2. □

Theorem 3 gives us an intuition to the usefulness of the Rademacher average for our purpose: if R_S is small, then also the r.h.s. of (3.1) is small. Thus, we expect that the sample S has a sufficiently large size to ensure good estimates of the true support $Supp_T(p)$ for every pattern $p \in P$, which implies a small value of $\sup_{p\in P} |Supp_T(p) - Supp_S(p)|$. Note that, in order to satisfy Lemma 1, we want a sufficiently small value for $\sup_{p\in P} |Supp_T(p) - Supp_S(p)|$ which is lower or equal than $\epsilon/2$ with high probability.

In a progressive sampling scenario the right thing to do is to weigh the confidence parameter δ with the iteration index i, differently from [7] where δ is constant for every iteration. Let S_i be the sample considered in iteration $i \geq 1$ of Algorithm 1, and $\delta_i = \dfrac{\delta}{2^i}$. The reason of this claim aims to guarantee the proof of correctness of Theorem 1. Thus, we use the following revised version of Theorem 3:

Theorem 4. *(Revised Theorem 3) With probability at least* $1 - \delta_i$

$$\sup_{p \in P} |Supp_T(p) - Supp_{S_i}(p)| \leq 2R_{S_i} + \sqrt{\frac{2\ln(2/\delta_i)}{|S_i|}},$$

where $\delta_i = \dfrac{\delta}{2^i}$*, for any iteration* $i \geq 1$*.*

In order to define a stopping condition that is computationally reasonable to verify, we need to introduce some other theoretical results. Let S_i be the sample involved in the iteration i of Algorithm 1. Theorem 4 gives us a first bound to $\sup_{p \in P} |Supp_T(p) - Supp_{S_i}(p)|$. It would require to compute R_{S_i} which, if done naïvely, is computationally expensive, since we need to mine all patterns from S_i (i.e., $FP(S_i, 1/|S_i|)$) and compute the expectation over the σ Rademacher random variables. We focus on finding an upper bound to R_{S_i} that is easy and fast to compute. For any pattern $p \in P$, let define the following $|S_i|$-dimensional vector

$$v_{S_i}(p) = (\phi_p(t_1), \ldots, \phi_p(t_{|S_i|}))$$

and let $V_{S_i} = \{v_{S_i}(p), p \in P\}$, where $t_1, t_2, \ldots, t_{|S_i|}$ are the $|S_i|$ transactions of S_i. In itemsets scenario, we have that the number of all possible itemsets is 2^d. In sequential patterns scenario, note that all the infinite sequences of the universe U which does not appear in S_i are associated with the vector $(0, \ldots, 0)$ of $|S_i|$ zeros. The two fact combined imply the finiteness of the size of V_{S_i}: $|V_{S_i}| < \infty$. The following two theorems derive from Thm 3.3 of [3]. Their adaptations for the itemsets scenario can be found in [7], instead they are contibutions of this thesis for the patterns scenario. For their proofs we need to use the Jensen inequality and the Hoeffding's inequality. The Jensen inequality (theorem 2.4 from [5]) states that if f is a convex function and X is a random variable, then

$$\mathbb{E}[f(X)] \geq f(\mathbb{E}[X]).$$

The Hoeffding's inequality (lemma 4.13 from [5]) states that if X represents a bounded random variable with $\mathbb{E}[X] = 0$ and $a \leq X \leq b$ then, for any $s > 0$,

$$\mathbb{E}[\exp(sX)] \leq \exp\left(\frac{s^2(b-a)^2}{8}\right).$$

Theorem 5. *(Massart's Lemma)*

$$R_{S_i} \leq \max_{p \in P} ||v_{S_i}(p)|| \frac{\sqrt{2 \ln |V_{S_i}|}}{|S_i|}$$

where $|| \cdot ||$ *indicates the Euclidean norm.*

Proof. (by Lemma 26.8 of [10]) First of all, note that $\max_{p \in P} ||v_{S_i}(p)|| = \max_{v \in V_{S_i}} ||v||$. Let $\lambda > 0$ and let $A' = \{\lambda v_1, \ldots, \lambda v_{|V_{S_i}|}\}$. Now,

$$|S_i| R_{A'} = \mathbb{E}_\sigma \left[\max_{a \in A'} \langle \sigma, a \rangle \right] = \mathbb{E}_\sigma \left[\log \left(\max_{a \in A'} e^{\langle \sigma, a \rangle} \right) \right] \leq$$

$$\mathbb{E}_\sigma \left[\log \left(\sum_{a \in A'} e^{\langle \sigma, a \rangle} \right) \right] \overset{J.}{\leq} \log \left(\mathbb{E}_\sigma \left[\sum_{a \in A'} e^{\langle \sigma, a \rangle} \right] \right) = \log \left(\sum_{a \in A'} \prod_{i=1}^{|S_i|} \mathbb{E}_{\sigma_i} \left[e^{\sigma_i a_i} \right] \right),$$

where the last inequality holds by using the linearity of the expectation and the independence of σ_i's. Since, by using Lemma A.6 of [10],

$$\mathbb{E}_{\sigma_i} \left[e^{\sigma_i a_i} \right] = \frac{\exp(a_i) + \exp(-a_i)}{2} \leq \exp(a_i^2/2)$$

we have that

$$|S_i| R_{A'} \leq \log \left(\sum_{a \in A'} \prod_{i=1}^{|S_i|} \exp \left(\frac{a_i^2}{2} \right) \right) = \log \left(\sum_{a \in A'} \exp \left(||a||^2/2 \right) \right) \leq$$

$$\log \left(|A'| \max_{a \in A'} \exp(||a||^2/2) \right) = \log(|A'|) + \max_{a \in A'} (||a||^2/2).$$

Now, since $R_{S_i} = \frac{1}{\lambda} R_{A'}$, we obtain

$$R_{S_i} \leq \frac{\log(|A'|) + \max_{a \in A'} (||a||^2/2)}{\lambda |S_i|} = \frac{\log(|V_{S_i}|) + \lambda^2 \max_{v \in V_{S_i}} (||v||^2/2)}{\lambda |S_i|}.$$

and, setting $\lambda = \sqrt{\frac{2 \log(|V_{S_i}|)}{\max_{v \in V_{S_i}} ||v||^2}}$, the thesis is true by rearranging the terms. $\square$

The following theorem is a stronger version of the previous one.

Theorem 6. *Let* $w : \mathbb{R}^+ \to \mathbb{R}^+$ *be the function*

$$w(s) = \frac{1}{s} \ln \sum_{v \in V_{S_i}} \exp\left(\frac{s^2 ||v||^2}{2|S_i|^2}\right)$$

then

$$R_{S_i} \leq \min_{s \in \mathbb{R}^+} w(s).$$

Proof. Let $n = |S_i|$. For any $s > 0$ and for any $p \in P$, by using the independence of σ_i's and the Hoeffding's inequality, we have that

$$\mathbb{E}_\sigma\left[\exp\left(s\frac{1}{n}\sum_{i=1}^{n}\sigma_i\phi_p(t_i)\right)\right] \overset{\text{ind.}}{=} \prod_{i=1}^{n}\mathbb{E}_\sigma\left[\exp\left(s\frac{1}{n}\sigma_i\phi_p(t_i)\right)\right]$$

$$\overset{\text{H.}}{\leq} \prod_{i=1}^{n}\exp\left(\frac{s^2\phi_p(t_i)^2}{2n^2}\right) = \exp\left(\frac{s^2||v_{S_i}(p)||^2}{2n^2}\right)$$

where we have applied the Hoeffding's inequality using $\frac{1}{n}\sigma_i\phi_p(t_i)$ as random variables which take value in $\frac{1}{n}\phi_p(t_i)\,[-1, 1]$. The last equality follows because

$$\sum_{i=1}^{n}\phi_p(t_i)^2 = ||v_{S_i}(p)||^2.$$

Thus,

$$\mathbb{E}_\sigma\left[\exp\left(s\frac{1}{n}\sum_{i=1}^{n}\sigma_i\phi_p(t_i)\right)\right] \leq \exp\left(\frac{s^2||v_{S_i}(p)||^2}{2n^2}\right). \tag{3.2}$$

Now, using the above inequality and the Jensen inequality, we have that

$$e^{sR_{S_i}} = \exp\left(s\mathbb{E}_\sigma\left[\max_{p\in P}\frac{1}{n}\sum_{i=1}^{n}\sigma_i\phi_p(t_i)\right]\right) = \exp\left(s\mathbb{E}_\sigma\left[\max_{v\in V_{S_i}}\frac{1}{n}\sum_{i=1}^{n}\sigma_i v_i\right]\right)$$

$$\overset{\text{J.}}{\leq} \mathbb{E}_\sigma\left[\exp\left(s\max_{v\in V_{S_i}}\frac{1}{n}\sum_{i=1}^{n}\sigma_i v_i\right)\right] \leq \sum_{v\in V_{S_i}}\mathbb{E}_\sigma\left[\exp\left(s\frac{1}{n}\sum_{i=1}^{n}\sigma_i v_i\right)\right]$$

$$\leq \sum_{v \in V_{S_i}} \exp\left(\frac{s^2||v||^2}{2n^2}\right)$$

where the last and the second to last inequality holds respectively using equation 3.2 and by taking into account that:

$$\mathbb{E}\left[e^{\max_v g(v)}\right] \leq \sum_v \mathbb{E}\left[e^{g(v)}\right].$$

Thus,

$$e^{sR_{S_i}} \leq \sum_{v \in V_{S_i}} \exp\left(\frac{s^2||v||^2}{2n^2}\right).$$

Now if we take the logarithm on both sides and divide by s, we obtain

$$R_{S_i} \leq w(s)$$

and, since every inequality described is true for any $s > 0$, we can take the one that minimizes $w(s)$ in order to reach the thesis. □

Note that the function w is defined as a sum over elements in V_{S_i} and not over all patterns $p \in P$. In general, we have potentially $|V_{S_i}| \ll |P|$. Indeed, there may be two or more patterns with the same vector $v_{S_i} \in V_{S_i}$ associated (i.e., these patterns appear exactly in the same transactions). However, the upper bound on R_{S_i} of Theorem 6 is not directly applicable since it requires to determine the entire set V_{S_i}, which is connected to the set of closed patterns on S_i as we present in the next chapters.

In this chapter we presented some basic concepts from data mining and statistical learning theory, which are crucial for understanding the next chapters.

Chapter 4

Mining frequent itemsets using progressive sampling approach

This chapter is dedicated to the progressive sampling approach for mining frequent itemsets presented in [7]. The authors describe an efficiently computable upper bound on R_{S_i} and then to $\sup_{X \subseteq I} |Supp_T(X) - Supp_{S_i}(X)|$. In section 4.3, which is a contribution of this thesis, we improve the computation of an upper bound to the Rademacher complexity of itemsets proposed in [7]. In addition, this thesis proposes a different analysis of the algorithm compared to what has been done in [7]. We set the confidence parameter to be $\delta_i = \delta/2^i$, i.e., dependent on the iterations of the progressive sampling approach. This leads to the proof of correctness of Theorem 1. Instead, in [7] the authors used the same confidence parameter regardless the iterations of the procedure, i.e., $\delta_i = \delta$, which does not give any theoretical guarantees as stated in inequality 2.1.

4.1 Stopping condition

The following two results show that the upper bound to R_{S_i} of Theorem 6 is not sufficient to define an efficiently computable stopping condition.

Lemma 2. *Let $H \subseteq S_i$. There is at most one closed itemset X in S_i whose support set in S_i is $S_{iX} = H$.*

Proof. Suppose that the statement is not true, i.e., there could be two closed itemsets C and D with the same support set H. If we consider the itemset

$C \cup D$, its support set in S_i would be exactly H. So, there exists a superset of both the itemsets with the same support set and, consequently, the same support. This implies that C and D cannot be closed, which lead to a contradiction. Thus, the thesis is true. □

Lemma 3. *The set V_{S_i} contains all and only the vectors $v_{S_i}(X)$ for all $X \in CI(S_i)$, i.e.,*

$$V_{S_i} = \{v_{S_i}(X), X \in CI(S_i)\}, \text{ and } |V_{S_i}| = |CI(S_i)|.$$

Proof. Let $X \in CI(S_i)$, and let H_X be the set of subsets of X with the same support $Supp_{S_i}(X)$:

$$H_X = \{B \subseteq X : \ Supp_{S_i}(B) = Supp_{S_i}(X)\}.$$

We observe the following two facts:

- in H_X there are the itemsets, subsets of X, which appear in all and only the transactions of the support set S_{iX}. This implies that, for all $B \in H_X$, $v_{S_i}(B) = v_{S_i}(X)$. Thus, each set H_X where $X \in CI(S_i)$ is represented by $v_{S_i}(X)$ in V_{S_i};
- note that Lemma 2 tells us that for each pair of closed itemsets C and D in S_i there must be $v_{S_i}(C) \neq v_{S_i}(D)$.

Thus, each element of V_{S_i} is associated with a different closed itemset in S_i. □

The previous lemma shows that the computation of the function w defined in Theorem 6 is not advisable because it requires to know the set V_{S_i}, i.e., to extract all the closed itemsets of S_i.

Now we introduce some definitions and results which allow us to define a function $\tilde{w}$ that is an upper bound to w. Let I_{S_i} be the set of items that appear in the sample S_i and $<_o$ be its increasing ordering by their support in S_i (ties broken arbitrarily). Given an item a, let $S_{i\{a\}}$ be its support set on S_i. Let $<_a$ denote the increasing ordering of the transactions $S_{i\{a\}}$ by the number of items contained that come after a w.r.t. the ordering $<_o$. Let $CI_1 = CI(S_i) \cap I_{S_i}$ and CI_{2+} be the set of closed itemsets of size one and at least two, respectively. Let us focus on partitioning CI_{2+}. Let $A \in CI_{2+}$ and let $a \in A$ be the item in A which comes before any other item in A w.r.t.

the order $<_o$. Let τ be the transaction containing A which comes before any other transaction containing A w.r.t. the order $<_a$ (clearly, $a \in \tau$). We assign A to the set $CI_{a,\tau}$. Now, let us consider a transaction $\tau \in S_{i\{a\}}$ assuming that it contains exactly $k_{a,\tau}$ items that come after a in $<_o$. In the ordering $<_a$, τ comes

- before every transaction containing more than $k_{a,\tau}$ items that come after a in $<_o$, and
- before zero or more of the transactions with exactly $k_{a,\tau}$ items that come after a in $<_o$ (the exact number depends on the tie-breaking criteria).

For each $r \geq 1$, let $g_{a,r}$ be the number of transactions in $S_{i\{a\}}$ that contain exactly r items located after a in the ordering $<_o$. Let $\chi_a = max\{r : g_{a,r} > 0\}$, i.e., the maximum r for which there exists at least one transaction in $S_{i\{a\}}$ containing exactly r items that come after a in $<_o$. Let

$$h_{a,r} = \sum_{j \geq r}^{\chi_a} g_{a,j}$$

be the number of transactions in $S_{i\{a\}}$ that contain at least r items that come after a in $<_o$. A graphical representation of the quantities just described is depicted in figure 4.1.

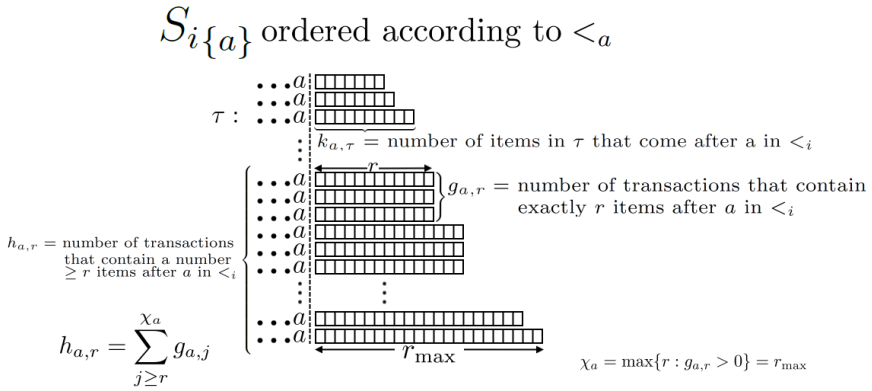

Figure 4.1: Graphical representation of $k_{a,\tau}$, $g_{a,r}$, $h_{a,r}$ and χ_a

Now, let τ be the $\ell_{a,\tau}$-th transaction of all such transactions in $<_a$ that contain exactly $k_{a,\tau}$ items that come after a in $<_o$. The following lemma gives us an upper bound to the size of $CI_{a,\tau}$.

Lemma 4. *We have*

$$|CI_{a,\tau}| \leq 2^{min\{k_{a,\tau}, h_{a,k_{a,\tau}} - \ell_{a,\tau}\}}$$

.

Proof. The quantity $2^{k_{a,\tau}}$ corresponds to the number of subsets B of τ such that $B = \{a\} \cup C$, where C is any subset of τ which contains only items that come after a in $<_o$. Note that $CI_{a,\tau}$ contains only those itemsets that appear in τ and are in the form of B. Thus, $|CI_{a,\tau}| \leq 2^{k_{a,\tau}}$.

Now, consider an itemset $A \in CI_{a,\tau}$ ($A = \{a\} \cup C$, for C as above). Let Θ be the set of the transactions in which A could appear (a part from τ), i.e., every transaction $\tau' \in S_{i\{a\}}$ such that $\tau <_a \tau'$, then $|\Theta| = h_{a,k_{a,\tau}} - \ell_{a,\tau}$. Lemma 2 tells us that there is at most one closed itemset for each set $D = \{\tau\} \cup F$ of transactions, where $F \subseteq \Theta$. Thus, there are at most $2^{h_{a,k_{a,\tau}} - \ell_{a,\tau}}$ closed itemsets in $CI_{a,\tau}$ (i.e., the number of all possible subsets of Θ). Since $|CI_{a,\tau}| \leq 2^{k_{a,\tau}}$ and $|CI_{a,\tau}| \leq 2^{h_{a,k_{a,\tau}} - \ell_{a,\tau}}$, then the thesis is true. □

Now, we can represent the set of closed itemsets as

$$CI(S_i) = CI_1 \cup CI_{2+} = CI_1 \cup \left(\bigcup_{a \in I_{S_i}} \bigcup_{\tau \in S_{i\{a\}}} CI_{a,\tau} \right) \tag{4.1}$$

and by taking into account the previous lemma we have

$$|CI(S_i)| \leq |I_{S_i}| + \sum_{a \in I_{S_i}} \sum_{\tau \in S_{i\{a\}}} 2^{min\{k_{a,\tau}, h_{a,k_{a,\tau}} - \ell_{a,\tau}\}}.$$

In the following lemma we define the function $\tilde{w}$ and an upper bound to R_{S_i} which will be used in the stopping condition.

Lemma 5. *Let $\tilde{w} : \mathbb{R}^+ \to \mathbb{R}^+$ be the function*

$$\tilde{w}(s) = \frac{1}{s} \ln \sum_{a \in I_{S_i}} \left(\left(1 + \sum_{r=1}^{\chi_a} \sum_{j=1}^{g_{a,r}} 2^{min\{r, h_{a,r} - j\}} \right) e^{\frac{s^2 Supp_{S_i}(\{a\})}{2|S_i|}} \right).$$

Then

$$R_{S_i} \leq \min_{s \in \mathbb{R}^+} \tilde{w}(s).$$

Proof. Let us consider the function w from Theorem 6:

$$w(s) = \frac{1}{s} \ln \sum_{v \in V_{S_i}} \exp\left(\frac{s^2||v||^2}{2n^2}\right),$$

where $n = |S_i|$. By using the definition of Euclidean norm, we have that, for any itemset $X \subseteq I$,

$$||v_{S_i}(X)|| = \sqrt{\sum_{i=1}^{n} \phi_X(t_i)^2} = \sqrt{\sum_{i=1}^{n} \mathbb{1}[t_i \ contains \ X]} = \sqrt{nSupp_{S_i}(X)}.$$

From Lemma 3 we can write the sum over V_{S_i} as the sum over $CI(S_i)$ which can be broken by using the equation (4.1) . Thus, we can rewrite w as

$$w(s) = \frac{1}{s} \ln \sum_{v \in V_{S_i}} \exp\left(\frac{s^2||v||^2}{2n^2}\right) = \frac{1}{s} \ln \sum_{X \in CI(S_i)} \exp\left(\frac{s^2 Supp_{S_i}(X)}{2n}\right)$$

$$= \frac{1}{s} \ln \left(\sum_{a \in CI_1} \exp\left(\frac{s^2 Supp_{S_i}(\{a\})}{2n}\right) + \sum_{a \in I_{S_i}} \sum_{\tau \in S_{i\{a\}}} \sum_{A \in CI_{a,\tau}} \exp\left(\frac{s^2 Supp_{S_i}(A)}{2n}\right) \right).$$

Now, since $CI_1 \subseteq I_{S_i}$, we have

$$\sum_{a \in CI_1} \exp\left(\frac{s^2 Supp_{S_i}(\{a\})}{2n}\right) \leq \sum_{a \in I_{S_i}} \exp\left(\frac{s^2 Supp_{S_i}(\{a\})}{2n}\right).$$

By using lemma 4 which gives us an upper bound to the size of $CI_{a,\tau}$ and the fact that, for any $X \subseteq CI_{a,\tau}$, $Supp_{S_i}(X) \leq Supp_{S_i}(\{a\})$ by the anti-monotonicity support property, we have

$$\sum_{\tau \in S_{i\{a\}}} \sum_{A \in CI_{a,\tau}} \exp\left(\frac{s^2 Supp_{S_i}(A)}{2n}\right) \leq$$

$$\sum_{\tau \in S_{i\{a\}}} 2^{min\{k_{a,\tau}, h_{a,k_{a,\tau}} - \ell_{a,\tau}\}} \exp\left(\frac{s^2 Supp_{S_i}(\{a\})}{2n}\right).$$

Finally, the right-hand side of the last inequality can be rewritten as

$$\sum_{r=1}^{\chi_a} \sum_{j=1}^{g_{a,r}} 2^{min\{r, h_{a,r} - j\}} \exp\left(\frac{s^2 Supp_{S_i}(\{a\})}{2n}\right).$$

Thus, we define

$$\tilde{w}(s) = \frac{1}{s} \ln \sum_{a \in I_{S_i}} \left(\left(1 + \sum_{r=1}^{\chi_a} \sum_{j=1}^{g_{a,r}} 2^{min\{r, h_{a,r}-j\}} \right) e^{\frac{s^2 Supp_{S_i}(\{a\})}{2|S_i|}} \right),$$

and by using the above arguments we have that $w(s) \leq \tilde{w}(s)$ for any $s \in \mathbb{R}^+$. Since $R_{S_i} \leq \min_{s \in \mathbb{R}^+} w(s)$ (by Theorem 6) and $w(s) \leq \tilde{w}(s)$, we conclude that

$$R_{S_i} \leq \min_{s \in \mathbb{R}^+} \tilde{w}(s).$$

□

Note that the function $\tilde{w}$ is not expensive to compute. Indeed, it requires to know just the support of each item in I_{S_i} and some additional information achievable with a single scan of S_i ($g_{a,r}$ and $h_{a,r}$ for each $a \in I_{S_i}$ and for each r, $1 \leq r \leq \chi_a$). In addition, since $\tilde{w}$ is convex and has first and second derivatives w.r.t. s everywhere in $\mathbb{R}^+$, its global minimum can be computed using a non-linear optimization solver (NLopt in [7]).
Thus, by combining Theorem 4 and Lemma 5, at each iteration $i \geq 1$ we have that

$$\sup_{X \subseteq I} |Supp_T(X) - Supp_{S_i}(X)| \leq 2 \min_{s \in \mathbb{R}^+} \tilde{w}(s) + \sqrt{\frac{2 \ln (2/\delta_i)}{|S_i|}} = \eta_i$$

with probability at least $1 - \delta_i$, where $\delta_i = \frac{\delta}{2^i}$. We stop the procedure as soon as the *stopping condition*

$$\eta_i \leq \epsilon/2$$

is verified, in order to output an ϵ-*approximation* to $FI(T, \theta)$ with probability at least $1 - \delta$, as stated by the Theorem 1.

4.2 The algorithm

In [7] the sampling schedule adopted is *automatic*, i.e., the size of the next sample depends only on some information about the current sample. In such way we avoid to define additional parameters for this purpose. First of all, we have to select the initial sample size $|S_1|$, i.e., the minimum sample size for which it is possible to satisfy the stopping condition.

Lemma 6. *(Initial sample size). Let*

$$|S_1| = \frac{8 \ln (2/\delta_1)}{\epsilon^2}.$$

The stopping condition cannot be satisfied on sample with size smaller than $|S_1|$.

Proof. Assume that, by contradiction, there exists a sample S with $|S| < |S_1|$ for which the stopping condition can be satisfied. We have

$$|S| < \frac{8 \ln (2/\delta_1)}{\epsilon^2}$$

which implies

$$\sqrt{\frac{2 \ln (2/\delta_1)}{|S|}} > \epsilon/2.$$

Clearly, the left-hand side of the previous inequality plus $2 \min_{s \in \mathbb{R}^+} \tilde{w}(s)$ represents a quantity greater than $\epsilon/2$, since $2 \min_{s \in \mathbb{R}^+} \tilde{w}(s) \geq 0$. Thus, the stopping condition is not satisfied and we can conclude that the thesis is true since a contradiction is reached. □

Assume to be at the end of the iteration $i \geq 1$ of the progressive sampling procedure. Let $|S_i|$ be the current sample size and

$$\eta_i = 2 \min_{s \in \mathbb{R}^+} \tilde{w}(s) + \sqrt{\frac{2 \ln (2/\delta_i)}{|S_i|}},$$

i.e., the l.h.s. of the stopping condition. If the stopping condition is not satisfied, then we compute the next sample size as follows:

$$|S_{i+1}| = \left(\frac{2\eta_i}{\epsilon} \right)^2 |S_i|.$$

Note that $|S_{i+1}|$ depends only on the current sample size $|S_i|$ and η_i, i.e., information about the quality of S_i. However, there are not theoretical guarantees about the optimality of this kind of schedule.

Algorithm 2 represents the pseudocode of the progressive sampling algorithm for mining frequent itemsets by using Rademacher Average presented in [7], which follows the general progressive sampling approach illustrated in

Algorithm 1. The revised minimum frequency threshold θ' is $\theta - \epsilon/2$, the stopping condition is $\eta_i \leq \epsilon/2$, and $|S_1| = \dfrac{8 \ln (2/\delta_1)}{\epsilon^2}$, $|S_{i+1}| = \left(\dfrac{2\eta_i}{\epsilon}\right)^2 |S_i|$ as sampling schedule. As previously mentioned, the computation of the function $\tilde{w}$ requires to know $g_{a,r}$ and $h_{a,r}$ for each $a \in I_{S_i}$ and for each r, $1 \leq r \leq \chi_a$, which can be done with a single scan of the sample. The support of each item and consequently the ordering $<_o$ are obtained during the sample creation. Thus, we look at each transaction τ, sort its items according to $<_o$, and for each item a in τ, increase by one $g_{a,k_{a,\tau}}$ and all counters $h_{a,r}$ for $1 \leq r \leq k_{a,\tau}$. In the following algorithm the function $random\_sample(T, m)$ returns m transactions drawn uniformly at random with replacement from T.

Algorithm 2: Progressive sampling algorithm for mining frequent itemsets

Data: : a dataset T built on alphabet I, parameters θ,ϵ,$\delta \in (0,1)$, a sampling schedule $(|S_i|)_{i\geq 1}$ of sample sizes

Result: an ϵ-*approximation* to $FI(T,\theta)$ w. p. at least $1-\delta$

$i \leftarrow 0$;

$S_0 \leftarrow \emptyset$, $|S_0| \leftarrow 0$;

do

$i \leftarrow i+1$;

if $|S_i| \geq |T|$ **then return** $FI(T,\theta)$;

$S^* \leftarrow random\_sample(T, |S_i| - |S_{i-1}|)$;

$S_i \leftarrow S_{i-1} \cup S^*$;

`/* the supports of the items are computed during the sample creation */`

$g_{a,r} \leftarrow 0$, $\forall a \in I_{S_i},\ r \in \mathbb{N}$;

$h_{a,r} \leftarrow 0$, $\forall a \in I_{S_i},\ r \in \mathbb{N}$;

for $\tau \in S_i$ **do**

for $a \in \tau$ **do**

$k_{a,\tau} \leftarrow$ number of items in τ that come after a in the order $<_o$;

$g_{a,k_{a,\tau}} \leftarrow g_{a,k_{a,\tau}} + 1$;

for $j \leftarrow 1, \ldots, k_{a,\tau}$ **do**

$h_{a,j} \leftarrow h_{a,j} + 1$;

end

$\chi_a \leftarrow max\{r :\ g_{a,r} > 0\}$;

end

end

$$\tilde{w}(s) \leftarrow \frac{1}{s} \ln \sum_{a \in I_{S_i}} \left(\left(1 + \sum_{r=1}^{\chi_a} \sum_{j=1}^{g_{a,r}} 2^{min\{r, h_{a,r}-j\}} \right) e^{\frac{s^2 Supp_{S_i}(\{a\})}{2|S_i|}} \right)$$

$s^* \leftarrow arg\, min_{s \in \mathbb{R}^+} \tilde{w}(s)$;

$\eta_i \leftarrow 2\tilde{w}(s^*) + \sqrt{\frac{2\ln(2/\delta_i)}{|S_i|}}$;

while $\eta_i > \epsilon/2$;

$\theta' \leftarrow \theta - \epsilon/2$;

return $FI(S_i, \theta')$;

4.3 An improvement to the algorithm

In each iteration of Algorithm 2 $k_{a,\tau}$, $g_{a,r}$, and $h_{a,r}$ are computed from scratch, without taking into account information that is already known. This section presents how these parameters can be computed using their values of the previous iteration.

Let consider a generic iteration i of the algorithm, for which the sample S_i is considered. Given a transaction $\tau \in S_i$ and an item $a \in \tau$, let consider:

- $<_o^i$ be the increasing ordering of the items I_{S_i} w.r.t. their support in S_i (ties broken arbitrarily) at iteration i;
- $k_{a,\tau}^i$ is the number of items of τ that come after a in the ordering $<_o^i$;
- $g_{a,r}^i$ is the number of transactions in $S_{i\{a\}}$ that contain exactly r items located after a in the ordering $<_o^i$;
- $h_{a,r}^i$ is the number of transactions in $S_{i\{a\}}$ that contain at least r items that come after a in $<_o^i$.

The sample S_{i+1} at iteration $i+1$ is composed by the previous sample S_i and S^*, i.e., $S_{i+1} = S_i \cup S^*$, where S^* is a set of transactions drawn uniformly at random with replacement from T. At iteration $i+1$ the support set size of each item increases of a natural number in the range $[0, \ldots, |S^*|]$, since additional $|S^*|$ transactions are considered. This leads to the updated ordering $<_o^{i+1}$. Now, for a generic item a, let A_i and A_{i+1} be the set of items that come before a in $<_o^i$ and $<_o^{i+1}$, respectively. Considering the transition from $<_o^i$ to $<_o^{i+1}$, let $\overrightarrow{A} = A_{i+1} \setminus A_i$ be the set of items surpassed by a, and $\overleftarrow{A} = A_i \setminus A_{i+1}$ be the set of items that have surpassed a. Note that: (a) if $|\overrightarrow{A}| > |\overleftarrow{A}|$ then the position of a in $<_o^{i+1}$ is greater than its position in $<_o^i$; (b) if $|\overrightarrow{A}| < |\overleftarrow{A}|$ then the position of a in $<_o^{i+1}$ is lower than its position in $<_o^i$; (c) $|\overrightarrow{A}| = |\overleftarrow{A}|$ the positions of a in $<_o^i$ and $<_o^{i+1}$ are the same.

The following procedure avoids to compute the parameters $k_{a,\tau}$, $g_{a,r}$, and $h_{a,r}$ from scratch at iteration $i+1$:

- for each additional transaction of S^* compute $k_{a,\tau}$, $g_{a,r}$, and $h_{a,r}$ as in Algorithm 2;

- for each transaction τ of the sample S_i of the previous iteration i and for each item a of τ compute

$$x_{a,\tau} = |\overrightarrow{A} \cap \tau| - |\overleftarrow{A} \cap \tau|,$$

i.e., the difference between the number of items of τ surpassed by a and the number of items of τ that have surpassed a. There are three different cases: $x_{a,\tau} = 0$,$x_{a,\tau} > 0$, and $x_{a,\tau} < 0$.

If $x_{a,\tau} = 0$, then $k^{i+1}_{a,\tau} = k^i_{a,\tau}$ since the number of items of τ that come after a in the ordering $<^{i+1}_o$ is not changed w.r.t the ordering $<^i_o$. Consequently, there is no need to update $g_{a,r}$ and $h_{a,r}$.

If $x_{a,\tau} > 0$, then $k^{i+1}_{a,\tau} = k^i_{a,\tau} - x_{a,\tau}$ since there are $x_{a,\tau}$ additional items of τ that come before a in the ordering $<^{i+1}_o$ w.r.t $<^i_o$. This implies that there is one less transaction with exactly $k^i_{a,\tau}$ items located after a in the ordering $<^{i+1}_o$, i.e., $g^{i+1}_{a,k^i_{a,\tau}} = g^i_{a,k^i_{a,\tau}} - 1$, and one more transaction with exactly $k^{i+1}_{a,\tau}$ items located after a in the ordering $<^{i+1}_o$, i.e., $g^{i+1}_{a,k^{i+1}_{a,\tau}} = g^i_{a,k^{i+1}_{a,\tau}} + 1$. Consequently, $h^{i+1}_{a,r} = h^i_{a,r} - 1\ \forall r = k^{i+1}_{a,\tau} + 1, \ldots, k^i_{a,\tau}$.

If $x_{a,\tau} < 0$, then $k^{i+1}_{a,\tau} = k^i_{a,\tau} + |x_{a,\tau}|$ since there are $|x_{a,\tau}|$ additional items of τ that come after a in the ordering $<^{i+1}_o$ w.r.t $<^i_o$. Thus, $g^{i+1}_{a,k^i_{a,\tau}} = g^i_{a,k^i_{a,\tau}} - 1$, $g^{i+1}_{a,k^{i+1}_{a,\tau}} = g^i_{a,k^{i+1}_{a,\tau}} + 1$, and $h^{i+1}_{a,r} = h^i_{a,r} + 1\ \forall r = k^i_{a,\tau} + 1, \ldots, k^{i+1}_{a,\tau}$.

The computation of $x_{a,\tau}$ requires to sort the items of each transaction according to the order $<_o$, as in [7]. Thus, this procedure does not improve the computational complexity of the portion of the while iteration which computes the parameters mentioned above. Instead, we expect a little improvement for the average running time since the case $x_{a,\tau} = 0$ does not lead to any update of the parameters. However, this procedure represents an intelligent way to avoid the computation of $k_{a,\tau}$, $g_{a,r}$, and $h_{a,r}$ from scratch.

In this chapter we presented the progressive sampling algorithm for mining frequent itemsets by using the Rademacher average proposed in [7] with a slightly different analysis of its guarantees and an improvement in computing some parameters.

Chapter 5

A bound for the Rademacher complexity of sequential patterns

In this chapter we formally prove the first rigorous and efficiently computable bound for the Rademacher complexity of sequential patterns.

Let S be a sample of the sequential dataset T. The following two results give us an upper bound to the size of V_S which depends on the number of closed sequential patterns of S.

Lemma 7. *Consider a subset W of the sample S, $W \subseteq S$. Let $CS_W(S)$ be the set of closed sequential patterns in S whose support set in S is W, i.e., $CS_W(S) = \{\boldsymbol{x} \in CS(S) : S_{\boldsymbol{x}} = W\}$, with $C = |CS_W(S)|$. Then the number of closed sequential patterns in S with W as support set satisfies: $0 \leq C \leq |CS(S)|$.*

Proof. The proof is organized in such a way: first, we show that the basic cases $C = 0$ and $C = 1$ hold, second, we prove that the cases for which $2 \leq C \leq |CS(S)|$ could happen, providing a toy example for $C = 2$.

Let us consider the case where W is a particular subset of S for which no sequence has W as support set in S. Thus, $CS_W(S)$ is an empty set and $C = 0$. The case $C = 1$ is trivial, since it could happen that only one closed sequential pattern has W as support set in S.

Now, in order to prove the cases for a generic value of C in $[2, \ldots, |CS(S)|]$, we start with an example for $C = 2$. Let $\mathbf{x}_1$,$\mathbf{x}_2$ be two sequences with W as support set. Assume that each super-sequence of $\mathbf{x}_1$ but not of $\mathbf{x}_2$ has support lower than the support of $\mathbf{x}_1$, and each super-sequence of $\mathbf{x}_2$ but not of $\mathbf{x}_1$ has support lower than the support of $\mathbf{x}_2$. Now, let us focus on

super-sequences of both $\mathbf{x}_1$ and $\mathbf{x}_2$. Let $\tau \in W$ be a transaction of W. We define $\mathbf{y}_\tau = \tau_{\mathbf{x}_1,\mathbf{x}_2}$ as the subsequence of τ restricted to only the sequences $\mathbf{x}_1$ and $\mathbf{x}_2$, preserving the relative order of the their itemsets. For instance, let $\mathbf{x}_1 = \langle A, B\rangle$, $\mathbf{x}_2 = \langle C, D\rangle$ and $\tau = \langle A, C, F, D, B\rangle$, where A, B, C, D, F are itemsets: thus, $\mathbf{y}_\tau = \langle A, C, B, D\rangle$. Now, if the support set of $\mathbf{y}_\tau$ in W does not coincide with W, i.e., $W_{\mathbf{y}_\tau} \subset W$, then for each transaction $t \in W$ we have $|W_{\mathbf{y}_\tau}| < |W_{\mathbf{x}_1}| = |W_{\mathbf{x}_2}| = |W|$. Note that this could happen because the set of itemsets of $\mathbf{x}_1$ and $\mathbf{x}_2$ may not appear in the same order in all transactions. Hence each super-sequence of both $\mathbf{x}_1$ and $\mathbf{x}_2$ has support lower than the support of $\mathbf{x}_1$ (that is equal to the support of $\mathbf{x}_2$). Thus, each super-sequence of $\mathbf{x}_i$ has a lower support compared to the support of $\mathbf{x}_i$, for $i = 1, 2$. This implies that $\mathbf{x}_1$ and $\mathbf{x}_2$ are closed sequences in S and since their support set is W, they belongs to $CS_W(S)$. Thus, the case $C = 2$ could happen. A simple example is depicted in Figure 5.1. Note first of all that $\mathbf{x}_1$ and $\mathbf{x}_2$ are closed sequences in S. Then we can see that $y_{\tau_1} = y_{\tau_3} \neq y_{\tau_2}$ which implies $|W_{y_{\tau_1}}|$, $|W_{y_{\tau_2}}|$, and $|W_{y_{\tau_3}}|$ be lower than $|W_{\mathbf{x}_1}| = |W_{\mathbf{x}_2}|$.

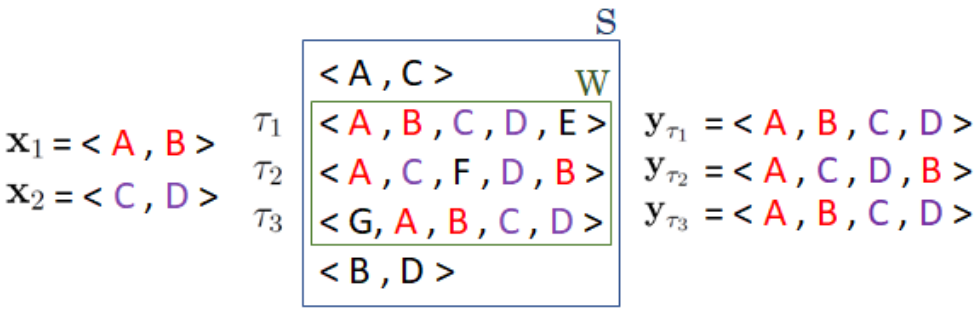

Figure 5.1: Graphical representation of the case $CS_W(S) = 2$

Now we generalize this concept for a generic number C of closed sequential patterns, where $2 \leq C \leq |CS(S)|$. Let $H = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_C\}$ be a set of C sequential patterns with W as support set. Assume that each super-sequence of $\mathbf{x}_i$ but not of $\mathbf{x}_k$ has support lower than the support of $\mathbf{x}_i$, for each $i, k \in [1, \ldots, C]$ with $k \neq i$. Let H_p be the power set of H without the empty set and the sets made of only one sequence, i.e., $H_p = P(H) \setminus \{\{\emptyset\}, \{\mathbf{x}_1\}, \{\mathbf{x}_2\}, \ldots, \{\mathbf{x}_C\}\}$. So, in H_p there are every possible subset of H of size greater than one. For a transaction $\tau \in W$ and $h_p \in H_p$, we define $\mathbf{y}_\tau(h_p) = \tau_{h_p}$ as the subsequence of τ restricted to h_p, i.e., to only the sequences $\mathbf{x} \in h_p$, preserving the relative order of the their itemsets.

If $\forall h_p \in H_p$ there exits a transaction $\tau \in W$ such that the support set of $\mathbf{y}_\tau(h_p)$ in W does not coincide with W, i.e., $W_{\mathbf{y}_\tau(h_p)} \subset W$, then for each transaction $t \in W$ we have $|W_{\mathbf{y}_\tau(h_p)}| < |W_{\mathbf{x}_1}| = \cdots = |W_{\mathbf{x}_C}| = |W|$. Hence each super-sequence made of only sequences of h_p has support lower than the support of $\mathbf{x}_i$, for $i = 1, \ldots, C$. Thus, each super-sequence of $\mathbf{x}_i$ has a lower support compared to the support of $\mathbf{x}_i$, for $i = 1, \ldots, C$. This implies that all sequences of H are closed sequence in S and since their support set is W, they belongs to $CS_W(S)$. Thus, the generic case $2 \leq C \leq |CS(S)|$ happens and the thesis holds. □

Note that the previous lemma represents a sequential patterns version of Lemma 3 for itemsets, where the upper bound to the number of closed itemsets in S with W as support set is one (this holds by the nature of the itemsets where the notion of "ordering" is not defined).

Lemma 8. *$V_S = \{v_S(\boldsymbol{x}) : \boldsymbol{x} \in CS(S)\} \cup \{(0, \ldots, 0)\}$ and $|V_S| \leq |CS(S)|+1$, i.e., each vector of V_S different from $(0, \ldots, 0)$ is associated to at least one closed sequential pattern in S.*

Proof. Let $V_S = \overline{V}_S \cup \{(0, \ldots, 0)\}$, where $\overline{V}_S = \{v \in V_S : v \neq (0, \ldots, 0)\}$. Let $\mathbf{x} \in U$ be a sequence of non-empty support set in S, i.e., $v_S(\mathbf{x}) \neq (0, \ldots, 0)$. There are two possibilities: $\mathbf{x}$ is or is not a closed sequence in S. If $\mathbf{x}$ is not a closed sequence, then there exists a closed super-sequence $\mathbf{y} \sqsupset \mathbf{x}$ with support equal to the support of $\mathbf{x}$, so with $v_S(\mathbf{x}) = v_S(\mathbf{y})$. Thus, $v_S(\mathbf{x})$ is associated with at least one closed sequence. Combining this with the fact that each vector $v \in \overline{V}_S$ is associated to at least one sequence $\mathbf{x} \in U$ and Lemma 7, then each vector of V_S different from $(0, \ldots, 0)$ is associated to at least one closed sequential pattern of S. To conclude our proof is sufficient to show that there are no closed sequences associated to the vector $(0, \ldots, 0)$. Let $SP_\infty = \{\mathbf{x} \in U : v_S(\mathbf{x}) = (0, \ldots, 0)\}$. Note that $|SP_\infty| = \infty$. For each $\mathbf{x} \in SP_\infty$, there always exists a super-sequence $\mathbf{y} \sqsupset \mathbf{x}$ such that $Supp_S(\mathbf{x}) = Supp_S(\mathbf{y}) = 0$. This implies that each sequence of SP_∞ is not closed. Thus, $\overline{V}_S = \{v_S(\mathbf{x}) : \mathbf{x} \in CS(S)\}$ and $|V_S| = |\overline{V}_S| + 1 \leq |CS(S)| + 1$. □

Combining a partitioning of $CS(S)$ with the previous lemma we can define a function $\tilde{w}^*$, an upper bound to the function w of Theorem 6, which is efficient to compute with a single scan of S.

Let I_S be the set of items that appear in the sample S and $<_o$ be its increasing ordering by their support in S (ties broken arbitrarily). Given

an item a, let $S_{\langle\{a\}\rangle}$ be its support set on S. Let $<_a$ denote the increasing ordering of the transactions $S_{\langle\{a\}\rangle}$ by the number of items contained that come after a w.r.t. the ordering $<_o$ (ties broken arbitrarily). Let $CS(S) = C_1 \cup C_{2+}$, where $C_1 = \{\mathbf{x} \in CS(S) : ||\mathbf{x}|| = 1\}$ and $C_{2+} = \{\mathbf{x} \in CS(S) : ||\mathbf{x}|| \geq 2\}$. Let us focus on partitioning C_{2+}. Let $\mathbf{x} \in C_{2+}$ and let a be the item in $\mathbf{x}$ which comes before any other item in $\mathbf{x}$ w.r.t. the order $<_o$. Let τ be the transaction containing $\mathbf{x}$ which comes before any other transaction containing $\mathbf{x}$ w.r.t. the order $<_a$. We assign $\mathbf{x}$ to the set $C_{a,\tau}$. Remember that an item can appear multiple times in a sequence. Given a transaction $\tau \in S_{\langle\{a\}\rangle}$, $k_{a,\tau}$ is the number of items in τ (counted with their multiplicity) equal to a or that come after a in $<_o$. Let $m_{a,\tau}$ be the multiplicity of a in τ. For each $r, m \geq 1$, $m \leq r$, let $g_{a,r,m}$ be the number of transactions in $S_{\langle\{a\}\rangle}$ that contain exactly r items (counted with their multiplicity) equal to a or located after a in the ordering $<_o$, with exactly m repetition of a. Let $\chi_a = max\{r : g_{a,r,m} > 0\}$. The following lemma gives us an upper bound to the size of $C_{a,\tau}$.

Lemma 9. *We have*

$$|C_{a,\tau}| \leq 2^{k_{a,\tau}-m_{a,\tau}}(2^{m_{a,\tau}} - 1).$$

Proof. $C_{a,\tau}$ represents a subset of the set Φ of all those subsequences of τ that are made of only items equal to a or that come after a in $<_o$, with item-length at least two and with at least one occurrence of a. Let us focus on finding an upper bound to $|\Phi|$. In order to build such a generic subsequence of τ, it is sufficient to select i occurrences of a among the $m_{a,\tau}$ available, with $1 \leq i \leq m_{a,\tau}$, and choose j items among the remaining $k_{a,\tau} - m_{a,\tau}$ items different from a. Note that if $i = 1$, then $j > 0$. Thus, using the fact that the sum of $\binom{n}{k}$ for $k = 0, \ldots, n$ is equal to 2^n, we have

$$|\Phi| \leq \binom{m_{a,\tau}}{1} \sum_{j=1}^{k_{a,\tau}-m_{a,\tau}} \binom{k_{a,\tau} - m_{a,\tau}}{j} +$$

$$+ \sum_{i=2}^{m_{a,\tau}} \left[\binom{m_{a,\tau}}{i} \sum_{j=0}^{k_{a,\tau}-m_{a,\tau}} \binom{k_{a,\tau} - m_{a,\tau}}{j} \right]$$

$$\leq 2^{k_{a,\tau}-m_{a,\tau}} \sum_{i=1}^{m_{a,\tau}} \binom{m_{a,\tau}}{i} =$$

$$= 2^{k_{a,\tau}-m_{a,\tau}}(2^{m_{a,\tau}}-1),$$

where the first inequality holds because some sequences of Φ are counted more times. Since $|C_{a,\tau}| \leq |\Phi|$, the thesis holds. □

Combining the partitioning of $CS(S)$

$$CS(S) = C_1 \cup C_{2+} = C_1 \cup \left(\bigcup_{a \in I_S} \bigcup_{\tau \in S_{\{\langle a \rangle\}}} C_{a,\tau} \right) \tag{5.1}$$

with the previous lemma, we have

$$|CS(S)| \leq |I_S| + \sum_{a \in I_S} \sum_{\tau \in S_{\{\langle a \rangle\}}} 2^{k_{a,\tau}-m_{a,\tau}}(2^{m_{a,\tau}}-1).$$

Now we are ready to define the function $\tilde{w}^*$, an efficiently computable upper bound to R_S. The following lemma represents the analogous of Lemma 5, adjusted for sequential patterns.

Lemma 10. *Let $\tilde{w}^* : \mathbb{R}^+ \to \mathbb{R}^+$ be the function*

$$\tilde{w}^*(s) = \frac{1}{s} \ln \sum_{a \in I_S} \left(\left(1 + \sum_{r=1}^{\chi_a} \sum_{m=1}^{r} \sum_{j=1}^{g_{a,r,m}} 2^{r-m}(2^m - 1) \right) e^{\frac{s^2 Supp_S(\langle\{a\}\rangle)}{2|S|}} + 1 \right).$$

Then

$$R_S \leq \min_{s \in \mathbb{R}^+} \tilde{w}^*(s).$$

Proof. Let us consider the function w from Theorem 6:

$$w(s) = \frac{1}{s} \ln \sum_{v \in V_S} \exp\left(\frac{s^2 ||v||^2}{2n^2} \right),$$

where $n = |S|$. By using the definition of Euclidean norm, we have that, for any sequence $\mathbf{x} \in U$,

$$||v_S(\mathbf{x})|| = \sqrt{\sum_{i=1}^{n} \phi_{\mathbf{x}}(t_i)^2} = \sqrt{n Supp_S(\mathbf{x})}.$$

From Lemma 8 we can use the the sum over $CS(S)$ as an upper bound on the sum over V_S. Thus,

$$w(s) \leq \frac{1}{s} \ln \left(\sum_{\mathbf{x} \in CS(S)} \exp \left(\frac{s^2 Supp_S(\mathbf{x})}{2n} \right) + 1 \right).$$

Note that the vector $(0, \ldots, 0)$ of V_S provides a $+1$ in the logarithm.

The sum over $CS(S)$ can be broken using the equation (4.1) in the sum over C_1

$$\sum_{\mathbf{x} \in C_1} \exp \left(\frac{s^2 Supp_S(\mathbf{x})}{2n} \right)$$

plus the sum over C_{2^+}

$$\sum_{a \in I_S} \sum_{\tau \in S_{\langle \{a\} \rangle}} \sum_{\mathbf{x} \in C_{a,\tau}} \exp \left(\frac{s^2 Supp_S(\mathbf{x})}{2n} \right).$$

Since the set of items of the sequences in C_1 is a subset of I_S, we have

$$\sum_{\mathbf{x} \in C_1} \exp \left(\frac{s^2 Supp_S(\mathbf{x})}{2n} \right) \leq \sum_{a \in I_S} \exp \left(\frac{s^2 Supp_S(\langle \{a\} \rangle)}{2n} \right).$$

By using Lemma 9 which gives us an upper bound to the size of $C_{a,\tau}$ and the fact that, for any $\mathbf{x} \in C_{a,\tau}$, $Supp_S(\mathbf{x}) \leq Supp_S(\langle \{a\} \rangle))$ by the anti-monotonicity support property for sequential patterns, we have

$$\sum_{\tau \in S_{\langle \{a\} \rangle}} \sum_{\mathbf{x} \in C_{a,\tau}} \exp \left(\frac{s^2 Supp_S(\mathbf{x})}{2n} \right) \leq$$

$$\sum_{\tau \in S_{\langle \{a\} \rangle}} 2^{k_{a,\tau} - m_{a,\tau}} (2^{m_{a,\tau}} - 1) \exp \left(\frac{s^2 Supp_S(\langle \{a\} \rangle)}{2n} \right).$$

Finally, the right-hand side of the last inequality can be rewritten as

$$\sum_{r=1}^{\chi_a} \sum_{m=1}^{r} \sum_{j=1}^{g_{a,r,m}} 2^{r-m} (2^m - 1) \exp \left(\frac{s^2 Supp_S(\langle \{a\} \rangle)}{2n} \right).$$

Thus, rearranging all the terms we reach the definition of $\tilde{w}^*$. Using the above arguments we have that $w(s) \leq \tilde{w}^*(s)$ for any $s \in \mathbb{R}^+$. Since $R_S \leq \min_{s \in \mathbb{R}^+} w(s)$ (by Theorem 6), we conclude that $R_S \leq \min_{s \in \mathbb{R}^+} \tilde{w}^*(s)$. $\square$

The function $\tilde{w}^*$ can be compute with a single scan of the sample, since it requires to know $g_{a,r,m}$ for each $a \in I_S$ and for each r, m, $1 \leq r \leq \chi_a$, $1 \leq m \leq r$. The support of each item and consequently the ordering $<_o$ are obtained during the sample creation. Thus, it is sufficient to look at each transaction τ, sorting I_τ according to $<_o$, and, for each item of I_τ, keep track of its multiplicity $m_{a,\tau}$, compute $k_{a,\tau}$ and increase by one $g_{a,k_{a,\tau},m_{a,\tau}}$. Finally, since $\tilde{w}^*$ is convex and has first and second derivatives w.r.t. s everywhere in $\mathbb{R}^+$, its global minimum can be computed using a non-linear optimization solver (e.g., NLopt).

Chapter 6

Mining sequential patterns using the Rademacher complexity

The bound on the Rademacher complexity we presented in the previous chapter can be used for the following two scenarios:

- extract a good approximation of the frequent sequences from a given sequential dataset, using the progressive sampling technique;
- mine a good approximation of the true frequent sequences of a generative process.

We now present such applications.

6.1 Mining frequent sequences using progressive sampling approach

In this section we present the first progressive sampling algorithm for mining frequent sequential patterns using the Rademacher average. Algorithm 3 represents the analogous of Algorithm 2, adjusted for sequential pattern mining scenario.

Given a sequential dataset T, we use the progressive sampling technique described in Algorithm 1 in order to extract a good approximation of the set of frequent sequences. Let S_i be the sample of iteration i in Algorithm 1. As for the itemset scenario of Chapter 4, the focus is to find an efficiently

computable bound to $\sup_{\mathbf{x}\in U} |Supp_T(\mathbf{x}) - Supp_{S_i}(\mathbf{x})|$. Combining Theorem 4 and Lemma 10, at each iteration $i \geq 1$ we have that

$$\sup_{\mathbf{x}\in U} |Supp_T(\mathbf{x}) - Supp_{S_i}(\mathbf{x})| \leq 2 \min_{s\in\mathbb{R}^+} \tilde{w}^*(s) + \sqrt{\frac{2\ln(2/\delta_i)}{|S_i|}} = \eta_i$$

with probability at least $1 - \delta_i$, where $\delta = \delta/2^i$. We stop the procedure as soon as the *stopping condition*

$$\eta_i \leq \epsilon/2$$

is verified, in order to output an ϵ-*approximation* to $FS(T, \theta)$ with probability at least $1 - \delta$, as stated by the proof of correctness of Theorem 1.

We have already mentioned that the bound of Lemma 10 is efficiently computable with a single scan of the sample. Now we present how the parameters $k_{a,\tau}$ and $g_{a,r,m}$ can be computed using their values of the previous iteration, avoiding to recomputing them from scratch, similarly to what we have done for the itemset scenario.

Let consider a generic iteration i of the algorithm. Given a transaction $\tau \in S_i$ and an item $a \in \tau$, let consider:

- $<_o^i$ be the increasing ordering of the items I_{S_i} w.r.t. their support in S_i (ties broken arbitrarily) at iteration i;
- $k_{a,\tau}^i$ is the number of items in τ (counted with their multiplicity) equal to a or that come after a in $<_o^i$;
- $m_{a,\tau}$ is the multiplicity of a in τ;
- $g_{a,r,m}^i$ is the number of transactions in $S_{\langle\{a\}\rangle}$ that contain exactly r items (counted with their multiplicity) equal to a or located after a in the ordering $<_o^i$, with exactly m repetition of a.

The sample S_{i+1} at iteration $i+1$ is composed by the previous sample S_i and S^*, i.e., $S_{i+1} = S_i \cup S^*$, where S^* is a set of transactions drawn at random with uniform probability and with replacement from T. At iteration $i+1$ the support set size of each item increases of a natural number in the range $[0, \ldots, |S^*|]$, since additional $|S^*|$ transactions are considered. This leads to the updated ordering $<_o^{i+1}$. Now, for a generic item a, let A_i and A_{i+1} be the set of items that come before a in $<_o^i$ and $<_o^{i+1}$, respectively. Considering the

transition from $<_o^i$ to $<_o^{i+1}$, let $\overrightarrow{A} = A_{i+1} \setminus A_i$ be the set of items surpassed by a, and $\overleftarrow{A} = A_i \setminus A_{i+1}$ be the set of items that have surpassed a. Considering a transaction τ, let b_τ be the bag of items that are in τ.

The following procedure avoids to compute the parameters $k_{a,\tau}$ and $g_{a,r,m}$ from scratch at iteration $i+1$:

- for each additional transaction τ of S^* compute $k_{a,\tau}$ and $m_{a,\tau}$, increasing by one $g_{a,k_{a,\tau},m_{a,\tau}}$;

- let $\overrightarrow{X}_{a,\tau}$ and $\overleftarrow{X}_{a,\tau}$ be two bag of items as follows: an item a of b_τ is added to $\overrightarrow{X}_{a,\tau}$ or $\overleftarrow{X}_{a,\tau}$ respectively if $a \in \overrightarrow{A}$ or $a \in \overleftarrow{A}$. Then, for each transaction τ of the sample S_i of the previous iteration i and for each item a of τ compute

$$x_{a,\tau} = |\overrightarrow{X}_{a,\tau}| - |\overleftarrow{X}_{a,\tau}|,$$

  i.e., the difference between the number of items of b_τ surpassed by a and the number of items of τ that have surpassed a. There are three different cases: $x_{a,\tau} = 0$, $x_{a,\tau} > 0$, and $x_{a,\tau} < 0$.

  If $x_{a,\tau} = 0$, then $k_{a,\tau}^{i+1} = k_{a,\tau}^i$ since the number of items of b_τ that come after a in the ordering $<_o^{i+1}$ is not changed w.r.t the ordering $<_o^i$. Consequently, there is no need to update $g_{a,r,m}$.

  If $x_{a,\tau} > 0$, then $k_{a,\tau}^{i+1} = k_{a,\tau}^i - x_{a,\tau}$ since there are $x_{a,\tau}$ additional items of b_τ that come before a in the ordering $<_o^{i+1}$ w.r.t $<_o^i$. This implies that: (a) there is one less transaction with exactly $k_{a,\tau}^i$ items (counted with their multiplicity) equal to a or located after a in the ordering $<_o^{i+1}$ with exactly $m_{a,\tau}$ repetitions of a, i.e., $g_{a,k_{a,\tau}^i,m_{a,\tau}}^{i+1} = g_{a,k_{a,\tau}^i,m_{a,\tau}}^i - 1$; (b) there is one more transaction with exactly $k_{a,\tau}^{i+1}$ items (counted with their multiplicity) equal to a or located after a in the ordering $<_o^{i+1}$ with exactly $m_{a,\tau}$ repetitions of a, i.e., $g_{a,k_{a,\tau}^{i+1},m_{a,\tau}}^{i+1} = g_{a,k_{a,\tau}^{i+1},m_{a,\tau}}^i + 1$.

  If $x_{a,\tau} < 0$, then $k_{a,\tau}^{i+1} = k_{a,\tau}^i + |x_{a,\tau}|$ since there are $|x_{a,\tau}|$ additional items of b_τ that come after a in the ordering $<_o^{i+1}$ w.r.t $<_o^i$. Thus, $g_{a,k_{a,\tau}^i,m_{a,\tau}}^{i+1} = g_{a,k_{a,\tau}^i,m_{a,\tau}}^i - 1$ and $g_{a,k_{a,\tau}^{i+1},m_{a,\tau}}^{i+1} = g_{a,k_{a,\tau}^{i+1},m_{a,\tau}}^i + 1$.

Thus, the algorithm follows.

Algorithm 3: Progressive sampling algorithm for mining frequent sequences

Data: : a sequential dataset T built on alphabet I, parameters θ,ϵ,$\delta \in (0,1)$, a sampling schedule $(|S_i|)_{i\geq 1}$ of sample sizes

Result: an ϵ-*approximation* to $FS(T,\theta)$ w. p. at least $1-\delta$

$i \leftarrow 0$;

$S_0 \leftarrow \emptyset$, $|S_0| \leftarrow 0$;

$g_{a,r,m} \leftarrow 0, \forall a \in I,\ r,m \in \mathbb{N},\ m \leq r$;

do

  $i \leftarrow i+1$;

  if $|S_i| \geq |T|$ **then return** $FS(T,\theta)$;

  $S^* \leftarrow random\_sample(T, |S_i| - |S_{i-1}|)$;

  $S_i \leftarrow S_{i-1} \cup S^*$;

  `/* the support of the items are computed during the sample creation */`

  for $\tau \in S^*$ **do**

    for $a \in \tau$ **do**

      $k_{a,\tau} \leftarrow$ number of items in τ (counted with their multiplicity) equal to a or that come after a in $<_o$;

      $m_{a,\tau} \leftarrow$ number of repetitions of a in τ;

      $g_{a,k_{a,\tau},m_{a,\tau}} += 1$;

    end

  end

  for $\tau \in S_i$ **do**

    for $a \in \tau$ **do**

      $\overrightarrow{A} \leftarrow$ set of items surpassed by a;

      $\overleftarrow{A} \leftarrow$ set of items that have surpassed a;

      $b_\tau \leftarrow$ the bag of items that are in τ;

      compute $\overrightarrow{X}_{a,\tau}, \overleftarrow{X}_{a,\tau}$;

      $x_{a,\tau} \leftarrow |\overrightarrow{X}_{a,\tau}| - |\overleftarrow{X}_{a,\tau}|$;

      if $x_{a,\tau} == 0$ **then** continue;

      $g_{a,k_{a,\tau},m_{a,\tau}} -= 1$;

      if $x_{a,\tau} > 0$ **then** $k_{a,\tau} -= x_{a,\tau}$;

      else $k_{a,\tau} += |x_{a,\tau}|$;

      $g_{a,k_{a,\tau},m_{a,\tau}} += 1$;

    end

  end

  /*

$$sum(a) \leftarrow \sum_{r=1}^{X_a} \sum_{m=1}^{r} \sum_{j=1}^{g_{a,r,m}} 2^{r-m}(2^m - 1),\ \ a \in I_{S_i}$$

$$exp(a) \leftarrow e^{\frac{s^2 Supp_{S_i}(\langle\{a\}\rangle)}{2|S_i|}},\ \ a \in I_{S_i}$$

  */

  $$\tilde{w}^*(s) \leftarrow \frac{1}{s} \ln \sum_{a \in I_{S_i}} [(1 + sum(a))exp(a) + 1]$$

  $s^* \leftarrow arg\, min_{s \in \mathbb{R}^+} \tilde{w}^*(s)$;

  $\eta_i \leftarrow 2\tilde{w}^*(s^*) + \sqrt{\frac{2\ln(2/\delta_i)}{|S_i|}}$;

while $\eta_i > \epsilon/2$;

$\theta' \leftarrow \theta - \epsilon/2$;

return $FS(S_i, \theta')$;

6.2 Mining true frequent sequences

In the previous section, using the progressive sampling technique, we extract a high-quality approximation of the frequent sequences of a sequential dataset T. Let consider the latter as a sample of transactions independently drawn from a probability distribution on the universe of sequences. Now we want to use the dataset T to learn frequent sequences of the unknown process that generates them. This section is the dedicated to present the first algorithm for mining *true* frequent sequential patterns of their underlying generative process.

Let U be the universe of sequences and π be a probability distribution on U. Note that we do not make any assumption about the unknown process π that generates sequences. Thus, the approach we present in this section is *distribution-free*. The (observed) sequential dataset T is a bag of $|T|$ independent identically distributed (i.i.d.) transactions drawn from π. We define $\pi_{\mathbf{x}}$ as the true support of the sequence $\mathbf{x}$ w.r.t. π, i.e., the probability that $\mathbf{x}$ appears in a transaction sampled from π.

Lemma 11. *The support $Supp_T(\boldsymbol{x})$ of $\boldsymbol{x}$ in T is an unbiased estimator for $\pi_{\boldsymbol{x}}$ ($Supp_T(\boldsymbol{x})$ is the empirical average of $\pi_{\boldsymbol{x}}$):*

$$\mathbb{E}\left[Supp_T(\boldsymbol{x})\right] = \pi_{\boldsymbol{x}}.$$

Proof. Since $Supp_T(\mathbf{x}) = |T_{\mathbf{x}}|/|T|$, then

$$\mathbb{E}\left[Supp_T(\mathbf{x})\right] = \frac{1}{|T|}\mathbb{E}\left[|T_{\mathbf{x}}|\right] = \frac{1}{|T|}\sum_{\tau \in T}\mathbb{E}\left[\mathbb{1}[\mathbf{x} \in \tau]\right] = \pi_{\mathbf{x}},$$

where $\mathbb{1}$ denotes the indicator function and the second equality holds by the linearity of the expectation. Note that $\mathbb{E}\left[\mathbb{1}[\mathbf{x} \in \tau]\right]$ represents the probability that $\mathbf{x}$ appears in a transaction of T drawn from π, i.e., $\pi_{\mathbf{x}}$. □

Given a support threshold $\theta \in (0, 1]$, the set $TFS(\pi, \theta)$ represents all sequences with true support at least θ, i.e., the set of all true frequent sequences (and their supports) of π w.r.t. θ:

$$TFS(\pi, \theta) = \{(\mathbf{x}, \pi_{\mathbf{x}}) : \mathbf{x} \in U \wedge \pi_{\mathbf{x}} \geq \theta\}.$$

The following theorem represents Theorem 3 for the true frequent sequence mining scenario.

Theorem 7. *With probability at least* $1-\delta$*:*

$$\sup_{\boldsymbol{x}\in U} |\pi_{\boldsymbol{x}} - Supp_T(\boldsymbol{x})| \leq 2R_T + \sqrt{\frac{2\ln(2/\delta)}{|T|}}. \tag{6.1}$$

Proof. The theorem directly derives from Theorem 2 just considering $\pi_{\mathbf{x}}$ and $Supp_T(\mathbf{x})$ as generalization and empirical measure, respectively. Let associate π with the probability distribution $\mathcal{D}$ and let τ be a transaction drawn uniformly at random from T. The true support $\pi_{\mathbf{x}}$ of a sequence $\mathbf{x}$ is the probability that τ contains $\mathbf{x}$, which can be considered as a generalization measure. Now, let us associate the sequential dataset T to the training set $\mathcal{S}$. Taking into account a sequence $\mathbf{x} \in U$, the support $Supp_T(\mathbf{x})$ of $\mathbf{x}$ in T is just the fraction of transactions of T which contain $\mathbf{x}$, which can be seen as an empirical measure. □

This theorem tell us that if the r.h.s. of equation 6.1 is small then we can approximate the true supports of the sequences with their observed support in the dataset T. Lemma 10 gives us an efficiently computable upper bound to R_T:

$$R_T \leq \min_{s\in\mathbb{R}^+} \tilde{w}^*(s),$$

where

$$\tilde{w}^*(s) = \frac{1}{s}\ln \sum_{a\in I} \left(\left(1 + \sum_{r=1}^{\chi_a}\sum_{m=1}^{r}\sum_{j=1}^{g_{a,r,m}} 2^{r-m}(2^m-1)\right) e^{\frac{s^2 Supp_T(\langle\{a\}\rangle)}{2|T|}} + 1 \right).$$

Thus, combining Theorem 7 and Lemma 10, and defining

$$\mu = 2\min_{s\in\mathbb{R}^+} \tilde{w}^*(s) + \sqrt{\frac{2\ln(2/\delta)}{|T|}},$$

we have

$$\sup_{\mathbf{x}\in U} |\pi_{\mathbf{x}} - Supp_T(\mathbf{x})| \leq \mu \tag{6.2}$$

i.e.,

$$Supp_T(\mathbf{x}) - \mu \leq \pi_{\mathbf{x}} \leq Supp_T(\mathbf{x}) + \mu,\ \forall \mathbf{x} \in U$$

with probability at least $1-\delta$.

Let $lb(\mathbf{x})$ and $ub(\mathbf{x})$ be respectively $Supp_T(\mathbf{x}) - \mu$ and $Supp_T(\mathbf{x}) + \mu$, i.e., the lower and upper bound to $\pi_{\mathbf{x}}$ for a generic sequence $\mathbf{x}$. Given a support threshold θ, the support $Supp_T(\mathbf{x})$ and the lower bound $lb(\mathbf{x})$, the sequence $\mathbf{x}$ is considered to be a true frequent sequential pattern if $lb(\mathbf{x}) \geq \theta$, i.e., $\pi_{\mathbf{x}} \geq \theta$. Thus, Algorithm 4 returns, with high probability, the set $\overline{TFS} = \{\mathbf{x} \in FS(T,\theta) : lb(\mathbf{x}) \geq \theta\}$ which is an approximation to the set of true frequent sequences, as stated by Theorem 8. This theorem tells us that with probability at least $1-\delta$ the set $\overline{TFS}$ does not contain *false positives*, which are sequential patterns $\mathbf{x}$ of $FS(T,\theta)$ with $lb(\mathbf{x}) \geq \theta$ but with true support $\pi_{\mathbf{x}} < \theta$.

Algorithm 4: Algorithm for mining true frequent sequences

Data: : a sequential dataset T built on alphabet I, parameters θ,$\delta \in (0,1)$
Result: an approximation to $TFS(T,\theta)$ w. p. at least $1-\delta$
$g_{a,r,m} \leftarrow 0$, $\forall a \in I,\ r,m \in \mathbb{N},\ m \leq r$;
/* the support of the items are computed during the scan of T */
for $\tau \in T$ **do**
  for $a \in \tau$ **do**
    $k_{a,\tau} \leftarrow$ number of items in τ (counted with their multiplicity) equal to a or that come after a in $<_o$;
    $m_{a,\tau} \leftarrow$ number of repetitions of a in τ;
    $g_{a,k_{a,\tau},m_{a,\tau}} += 1$;
  end
end
/*

$$sum(a) \leftarrow \sum_{r=1}^{\chi_a} \sum_{m=1}^{r} \sum_{j=1}^{g_{a,r,m}} 2^{r-m}(2^m - 1),\ \ a \in I$$

$$exp(a) \leftarrow e^{\frac{s^2 Supp_T(\langle\{a\}\rangle)}{2|T|}},\ \ a \in I$$

*/

$$\tilde{w}^*(s) \leftarrow \frac{1}{s} \ln \sum_{a \in I} [(1 + sum(a))exp(a) + 1]$$

$s^* \leftarrow arg\,min_{s \in \mathbb{R}^+} \tilde{w}^*(s)$;
$\mu \leftarrow 2\tilde{w}^*(s^*) + \sqrt{\frac{2\ln(2/\delta)}{|T|}}$;
compute $FS(T,\theta)$;
$lb(\mathbf{x}) \leftarrow Supp_T(\mathbf{x}) - \mu$ for each $\mathbf{x} \in FS(T,\theta)$;
compute $\overline{TFS} = \{\mathbf{x} \in FS(T,\theta) : lb(\mathbf{x}) \geq \theta\}$;
return $\overline{TFS}$;

Theorem 8. *With probability at least $1-\delta$, the set $\overline{TFS}$ provided by Algorithm 4 contains only sequences $\boldsymbol{x}$ such that $\pi_{\boldsymbol{x}} \geq \theta$, i.e., $\overline{TFS}$ contains no false positives.*

Proof. As stated for the inequality 6.2, combining Theorem 7 and Lemma 10 we have

$$\sup_{\mathbf{x} \in U} |\pi_{\mathbf{x}} - Supp_T(\mathbf{x})| \leq \mu$$

with probability at least $1-\delta$, where

$$\mu = 2 \min_{s \in \mathbb{R}^+} \tilde{w}^*(s) + \sqrt{\frac{2 \ln (2/\delta)}{|T|}}$$

and

$$\tilde{w}^*(s) = \frac{1}{s} \ln \sum_{a \in I} \left(\left(1 + \sum_{r=1}^{\chi_a} \sum_{m=1}^{r} \sum_{j=1}^{g_{a,r,m}} 2^{r-m}(2^m - 1) \right) e^{\frac{s^2 Supp_T(\langle \{a\} \rangle)}{2|T|}} + 1 \right).$$

This leads to define for each sequence $\mathbf{x}$ the lower bound $lb(\mathbf{x})$ to $\pi_{\mathbf{x}}$

$$\pi_{\mathbf{x}} \geq Supp_T(\mathbf{x}) - \mu = lb(\mathbf{x}),$$

which hold with probability at least $1-\delta$. Since the set $\overline{TFS}$ is made of frequent sequences $\mathbf{x}$ of T such that $lb(\mathbf{x}) \geq \theta$, then we have $\pi_{\mathbf{x}} \geq \theta$ with probability at least $1-\delta$ and the thesis holds. □

Chapter 7

Conclusions

In this thesis we present the first rigorous and efficiently computable upper bound for the Rademacher complexity of sequential patterns. Then, we propose the first algorithm based on a progressive sampling approach for mining frequent sequential patterns from a given dataset and the first algorithm for mining true frequent sequential patterns from an unknown generative process.

Now, some considerations can be done about the upper bound on the Rademacher complexity of sequential patterns. The first future work will be computing such bound in practice on real sequential datasets, in order to verify how tight it is. In fact, the tighter the upper bound is, the more accurate the algorithms we propose are in identifying frequent sequential patterns. Thus, a possible future work consists in improving the bound presented in this thesis, which represents, to the best of our knowledge, the first bound on the Rademacher complexity of sequential patterns.

Another future work will be the adaptation of the bound on the Rademacher complexity of sequential patterns to the statistically significant pattern mining scenario, where the criterion for which a pattern is flagged as meaningful is the statistical significance (as measured by some statistical test) and not just the frequency. Assume to label each transaction with a binary value (i.e., class) and define a null model as the independence among patterns and binary labels. We will design efficient and rigorous algorithms to identify associations between patterns and class labels. Let consider a pattern p and a class label c. We consider them as associated if the result we found on a given dataset is far from the expected result under the null model. Thus, we reject the null model, i.e. the independence between p and c.

Other future directions consist in finding a way for adjusting the bound on the Rademacher complexity of sequential patterns to design efficient algorithms to rigorously mine sequential patterns in other types of sequential data, such as biological data. In DNA sequencing a very large number of DNA subsequences (a.k.a. reads) are produced. A substring of length k of a read is called k-mer. We will adapt the techniques developed in this thesis for mining meaningful k-mers. In this way we will extract a high-quality approximation of the set of frequent k-mers from the dataset of reads, which are crucial to identify infrequent k-mers that are usually sequencing errors.

Bibliography

[1] R. Agrawal and R. Srikant. "Fast Algorithms for Mining Association Rules in Large Databases". In: VLDB '94 (1994), pp. 487–499.

[2] Rakesh Agrawal and Ramakrishnan Srikant. "Mining Sequential Patterns". In: *Proceedings of the Eleventh International Conference on Data Engineering*. ICDE '95. Washington, DC, USA: IEEE Computer Society, 1995, pp. 3–14.

[3] S. Boucheron, O. Bousquet, and G. Lugosi. "Theory of Classification: A Survey of Some Recent Advances". In: *ESAIM: Probability and Statistics* 9 (2005), pp. 323–375.

[4] Jian Pei et al. "Mining sequential patterns by pattern-growth: the PrefixSpan approach". In: *IEEE Transactions on Knowledge and Data Engineering* 16.11 (2004), pp. 1424–1440.

[5] Michael Mitzenmacher and Eli Upfal. *Probability and Computing: Randomization and Probabilistic Techniques in Algorithms and Data Analysis*. 2nd. New York, NY, USA: Cambridge University Press, 2017.

[6] Anand Rajaraman and Jeffrey David Ullman. *Mining of Massive Datasets*. New York, NY, USA: Cambridge University Press, 2011.

[7] Matteo Riondato and Eli Upfal. "Mining Frequent Itemsets Through Progressive Sampling with Rademacher Averages". In: KDD '15 (2015), pp. 1005–1014.

[8] Matteo Riondato and Fabio Vandin. "Finding the True Frequent Itemsets". In: *Proceedings of the 2014 SIAM International Conference on Data Mining*, pp. 497–505.

[9] Sacha Servan-Schreiber, Matteo Riondato, and Emanuel Zgraggen. "ProSecCo: Progressive Sequence Mining with Convergence Guarantees". In: *Proceedings of the 18th IEEE International Conference on Data Mining*. 2018, pp. 417–426.

[10] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014. Chap. 2,3,26.

[11] Ramakrishnan Srikant and Rakesh Agrawal. "Mining Sequential Patterns: Generalizations and Performance Improvements". In: *EDBT*. 1996.

[12] P.N. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining*. Addison Wesley, 2006. Chap. 6.