INFORMATICA, 2013, Vol. 24, No. 1, 87–102
© 2013 *Vilnius University*

# Markov Models in the Analysis of Frequent Patterns in Financial Data

Julija PRAGARAUSKAITĖ, Gintautas DZEMYDA
*Institute of Mathematics and Informatics, Vilnius University*
*Akademijos 4, LT-08663 Vilnius*
*e-mail: julija.pragarauskaite@gmail.com, gintautas.dzemyda@mii.vu.lt*

Received: September 2012; accepted: January 2013

**Abstract.** Frequent sequence mining is one of the main challenges in data mining and especially in large databases, which consist of millions of records. There is a number of different applications where frequent sequence mining is very important: medicine, finance, internet behavioural data, marketing data, etc. Exact frequent sequence mining methods make multiple passes over the database and if the database is large, then it is a time consuming and expensive task. Approximate methods for frequent sequence mining are faster than exact methods because instead of doing multiple passes over the original database, they analyze a much shorter sample of the original database formed in a specific way. This paper presents Markov Property Based Method (MPBM) – an approximate method for mining frequent sequences based on $k$th order Markov models, which makes only several passes over the original database. The method has been implemented and evaluated using real-world foreign exchange database and compared to exact and approximate frequent sequent mining algorithms.

**Keywords:** frequent sequence mining, approximate methods, Markov models, financial data.

## 1. Introduction and Motivation

The world contains a vast amount of digital information nowadays which is increasing every day. This brings significant benefits and makes it possible to do many things that previously could not be done: forecast financial trends, spot business trends, prevent and identify possible diseases, suspect crime facts, etc. Moore's law, which the computer industry now takes for granted, says that the processing power and storage capacity of computer chips double or their prices halve roughly every 18 months. Keeping in mind the amount of raw data today's algorithms and powerful computers can reveal new important insights that could previously remained hidden.

Frequent sequence mining is one of the main challenges in data mining and especially in large databases, which consist of millions of records. Exact frequent sequence mining methods deliver accurate results on frequent and rare sequences, however they make multiple passes over the original database and if the database is large, then it is a time consuming and expensive task. Frequent sequence mining with estimated error probability is acceptable in many applications, e.g., marketing, internet user behavior, stock market, biological databases, etc., because such algorithms work under the time constraints and the speed of algorithm is more important than high precision. For example, if an investor can obtain all approximate frequent sequences from the stock data quickly, then such approximate result might be sufficient to supplement the investor to take an optimal and profitable decision on a specific investment. On the other hand, if a stock investor attempted to extract all the accurate frequent sequences from the stock data, it would be very time-consuming and by the time a decision needs to be made, the best time for investment might have passed. Knowledge of frequently occurring sequences and the ability to identify them as quickly as possible is the key to a trader's success and typically this is a challenging task. First, the amount of data is typically so huge that it becomes difficult to store it and process in time when the decision maker needs it. Second, the previously acquired knowledge may age as the time goes buy and lose its importance. Therefore, in financial markets the speed of finding frequent patterns in huge databases is more important than accuracy and the approximate frequent sequence mining algorithms are used with the acceptable error threshold, because quick decisions with higher profitability could be taken on the trading strategy. Frequent sequences have to be identified and presented to the trader as quickly as possible as every second is important to decide whether to open/exit the trading position and the delay could delay the right moment of opening the trading position.

Approximate frequent sequence mining methods are much faster than exact because instead of doing multiple passes over the original database as exact methods do; they analyze a much shorter sample of the original database or are using specific assumptions on the structure of the original database.

In many cases finding an exact result in frequent sequence mining is not compatible with limited availability of resources and real time constraints, but an approximation of the exact result is enough for most purposes. Therefore approximate frequent sequence mining methods are irreplaceable for many real world applications, such as biological data analysis, stock market analysis, behavior analysis in the internet, etc., which require immediate answer which sequences are frequent or rare. In the past decade there was a number of various approximate frequent sequence mining algorithms proposed (ApproxMAP (Kum *et al.*, 2003), ProMFS (Tumasonis and Dzemyda, 2004), RSM (Pragarauskaite and Dzemyda, 2011), etc.), that are very fast compared to exact frequent sequence mining algorithms. The most popular methods used for frequent sequence mining do not use theoretical approximation, only empirical; therefore they require extensive experimentation and observation of algorithm results on different databases.

In this paper, we suggest a new approach for approximately mining frequent sequences in huge databases that uses Markov property based method (MPBM) and makes only several passes over the original database. MPBM is compared to the random sampling method – RSM (Pragarauskaite and Dzemyda, 2011). The models have been implemented and evaluated in real-world foreign exchange database; however the method we propose is not limited to stock exchange setting.

## 2. Related Work

Ever since the introduction of market basket analysis problem in Agrawal *et al.* (1993) the issue of frequently appearing pattern has attracted numerous research efforts. These studies can broadly be divided into two categories. Mining of frequent itemsets focuses on finding the frequency in which items are used together with the order of items not being important (Agrawal and Srikant, 1994; Brin *et al.*, 1997; Cheng *et al.*, 2007; Han *et al.*, 2000; Park *et al.*, 1995a, 1995b; Sarawagi *et al.*, 2000; Savasere *et al.*, 1995; Zaki, 2000). Frequent sequence mining, on the other hand, is concerned with the order in which the items arrive (Agrawal and Srikant, 1995; Ayres *et al.*, 2002; Han *et al.*, 2001; Srikant and Agrawal, 1996; Zaki, 2001). The patterns can be itemsets, sequences, subtrees, subgraphs, etc. Recently new approaches to the problem have looked for frequent sequences of itemsets (Gouda *et al.*, 2007).

### 2.1. *Frequent Itemset Mining*

There are three main basic frequent itemset mining methodologies: Apriori, FP-growth (Frequent Pattern growth) and Eclat (Equivalence Class Transformation; Cheng *et al.*, 2007). Apriori scans the database multiple times and generates candidate itemsets of length k from itemsets of length $k-1$. Then it prunes the candidates which have infrequent sub-items. Apriori algorithm refers to downward closure property, called Apriori, which states that a itemset is frequent only if all of its sub-items are frequent too. This is the essence of the Apriori algorithm (Agrawal and Srikant, 1994). After the Apriori algorithm was proposed, there was a number of improvements and extensions, e.g., hashing technique, partitioning technique, dynamic itemset counting, etc. (Brin *et al.*, 1997; Park *et al.*, 1995a, 1995b; Sarawagi *et al.*, 2000; Savasere *et al.*, 1995). Even though in many cases the Apriori algorithm significantly reduces the size of candidate sets using the Apriori principle, however it can suffer from 2 nontrivial costs: (1) generating a huge number of candidate sets, and (2) repeatedly scanning the database and checking the candidates by pattern matching. However, it requires multiple database scans, as many as the longest frequent itemset and therefore Apriori algorithm could not be used for mining frequent patterns in the data streams.

Han *et al.* (2000) proposed FP-growth algorithm (Frequent Pattern growth), which mines the complete set of frequent itemsets without candidate generation. The FP-growth algorithm uses frequent pattern tree (FP-tree) structure, which is an extended prefix-tree structure for storing compressed, crucial quantitative information about frequent patterns. Only frequent items which consist of 1 element have nodes in the tree, and the tree nodes are arranged in such a way that more frequently occurring nodes will have better chances of sharing nodes than less frequently occurring ones. The further candidate generation is achieved via concatenation of the suffix pattern with the new ones generated from an FP-tree. The search technique employed in mining is a partitioning-based, divide-and-conquer method rather than Apriori-like bottom-up generation of frequent itemsets combinations. Moreover, it transforms the problem of finding long frequent patterns to looking for shorter ones and then concatenating the suffix. The performance study showed that the FP-growth method is efficient and scalable for mining both long and short frequent patterns, and is about an order of magnitude faster than the Apriori algorithm.

Eclat (Equivalence CLASS Transformation) algorithm was proposed by Zaki (2000) and it uses vertical data format, whereas both Apriori and FP-growth algorithms use horizontal data format. Horizontal data format consists of a list of transactions, where each transaction has an identifier followed by a list of items in that transaction, whereas vertical data format associates each itemset with a list of transactions in which it occurs. First of all, the algorithm involves the computation of the frequent itemsets of size 1 and 2. Further frequent itemsets are generated by intersecting the tid-lists of all distinct pairs of atoms and checking the cardinality of the resulting tid-list. A recursive procedure call is made with those itemsets found to be frequent at the current level. This process is repeated until all frequent itemsets have been enumerated. Eclat algorithm scans the database only once to discover frequent itemsets.

### 2.2. *Frequent Sequence Mining*

Frequent sequential pattern mining focuses on finding frequent subsequences, where the order of subsequences is important. While in frequent itemset mining only the combinations of the items has to be generated, in case of sequence mining the variations of the combinations of the items has to be generated. Sequential pattern mining was first introduced by Agrawal and Srikant (1995). In 1996 the same authors proposed GSP (Generalized Sequence Patterns) algorithm (Srikant and Agrawal, 1996), which was a representative of Apriori-based algorithms and implements a candidate generation and testing approach. GSP became a well known and widely used algorithm in the frequent sequential pattern mining.

Zaki (2001) proposed a SPADE algorithm, which uses a vertical format to store the sequence patterns, which reduced the number of database scans. However both algorithms GSP and SPADE use Apriori pruning and both generate a large set of candidates for longer sequences, therefore due to multiple passes over the database, these algorithms could not be used for mining frequent sequences in the data streams.

A different approach – PrefixSpan algorithm was proposed by Pei *et al.* (2001) which works in a divide-conquer way and each sequential pattern is treated as a prefix and the complete set of sequential patterns can be partitioned into different subsets according to the different prefix and the mining of sequential patterns is done recursively. According to the algorithms performance comparison, PrefixSpan shows overall best performance, then SPADE outperforms GSP, which is in the last place.

SPAM algorithm (Sequential PAttern Mining) for mining sequential patterns was proposed by Ayres *et al.* (2002). The algorithm uses the first depth-first search strategy for mining sequential patterns and a vertical bitmap representation of the database allowing for efficient support counting. PrefixSpan outperforms SPAM slightly on very small datasets, but on large datasets SPAM outperforms PrefixSpan and SPADE by over an order of magnitude. However SPAM assumes that the entire database (and all data structures used for the algorithm) completely fit into main memory and therefore the algorithm could not be applied to frequent sequence mining in data streams.

Gouda *et al.* (2007, 2010) proposed PRISM algorithm (PRIme-Encoding Based Sequence Mining), which PRISM utilizes a vertical approach for enumeration and support counting, based on the prime block encoding, which in turn is based on prime factorization theory. According to the performance study on both synthetic and real datasets, Gouda *et al.* (2010) shows that PRISM outperforms SPADE, PrefixSpan and SPAM algorithms at least by an order of magnitude.

All mentioned algorithms for frequent sequence mining are exact algorithms and require reading the entire original database. If the database is huge, these algorithms are very time-consuming and require more time to find frequent sequences and in financial markets by the time a decision could be made by trader, the best time for investment might have passed. To satisfy such time constraints, the efficient probabilistic (approximate) mining methods are being developed. Probabilistic methods are faster than exact methods because instead of doing multiple passes over the original database they analyze much shorter random sample and/or are using specific assumptions on the structure of the original database. Recently, some efficient methods for mining approximate frequent sequences were proposed: ApproxMAP (Kum *et al.*, 2003). The idea of ApproxMAP algorithm is that, instead of finding exact patterns, the algorithms identifies patterns approximately shared by many sequences. Another approach is being proposed by approximate ProMFS algorithm (Tumasonis and Dzemyda, 2004), which is based on the estimated probabilistic-statistical characteristics of the appearance of elements of the sequence and their order. The algorithm builds a new much shorter sequence and analyses it using exact GSP algorithm (Srikant and Agrawal, 1996) and makes decisions on the main sequence in accordance with the results of analysis of the shorter one.

Probabilistic random sampling algorithm proposed in Pragarauskaite and Dzemyda (2011) analyzes a random sample of the original database. The algorithm makes decisions about the original database according to the random sample analysis results and performs much faster than the exact mining algorithms. The probability of errors made by the probabilistic algorithm is estimated using statistical methods.

In this paper a novel probabilistic Markov property based algorithm (MPBM) is proposed for mining frequent sequences in large databases. MPBM method makes only several passes over the original database, whereas the exact methods make multiple passes over the original database until there are no frequent sequences in the original database. The MPBM method is compared with the random sampling method (RSM), which analyzes the original database random sample and makes statistical inference about the frequent sequences in the original database. The errors made by the RSM method are estimated using the central limit theorem. The method has been implemented and evaluated in real-world foreign exchange database.

## 3. Markov Property Based Method (MPBM)

Markov property bases method (MPBM) makes several passes over the original database and identifies higher level approximate frequent sequences using exactly identified frequent sequences of the lower levels. For the 1st order Markov model, in order to find all approximate frequent sequences in the original databases, it is sufficient to make 2 passes over the original database (for the 2nd order Markov model it is sufficient to make 3 passes, etc.). The detailed method description is given below.

Let the original database be a set $S = \{x_1,\ x_2, \dots,\ x_N\}$ whose elements $x_i$ can take $M$ different values in a set $V = \{\alpha_1, \alpha_2, \dots,\ \alpha_M\}$. A sequence $(a_{i_1}, \dots,\ a_{i_m})$ is called frequent if (# is a number):

$$\frac{1}{N-m+1}\#\{j \in \{1,\dots,N-m+1\}\colon x_j = \alpha_{i_1},\ x_{j+1} = \alpha_{i_2}, \dots, \\ x_{j+m-1} = \alpha_{i_m}\} \geqslant \varepsilon,$$

where $\varepsilon \in\ (0,1)$ is a given threshold also known as a minimum support.

Assume $X_n,\ -\infty < n < \infty$, is a stationary finite set $V$-valued stochastic process: for each $n, m, t_1, \dots, t_n$ and $\alpha_{i_1}, \dots,\ \alpha_{i_n} \in V$,

$$P(X_{t_1} = \alpha_{i_1}, \dots,\ X_{t_n} = \alpha_{i_n}) = P(X_{t_1+m} = \alpha_{i_1}, \dots,\ X_{t_n+m} = \alpha_{i_n}).$$

According to the Theorem 6.1 in Varadhan (2001) we determine the empirical frequencies of sequences:

$$\overline{p}(\alpha_{i_1}, \dots, \alpha_{i_m}) = \frac{1}{N-m+1}\#\{j \in \{1,\dots,N-m+1\}\colon\ X_j = \alpha_{i_1}, \\ X_{j+1} = \alpha_{i_2}, \dots, X_{j+m-1} = \alpha_{i_m}\}.$$

Has a limit a.s. and in $L_1(P)$. If this limit is constant, then necessarily the limit is $p(\alpha_{i_1}, \dots, \alpha_{i_m}) = P\ \{X_0 = \alpha_{i_1},\ X_1 = \alpha_{i_2}, \dots, X_{m-1} = \alpha_{i_m}\}$:

$$\overline{p}(\alpha_{i_1}, \dots, \alpha_{i_m}) \to p(\alpha_{i_1}, \dots, \alpha_{i_m}) \quad \text{as } N \to \infty. \tag{1}$$

When in Theorem 6.1 the limit is constant for any $f$ (probability function), we say that $X_n$is ergodic. It is reasonable to say from the beginning that $X_n$ is ergodic and in the following two Remarks 1 and 2 we discuss some sufficient conditions for it.

DEFINITION 1. The whole space $\Omega$ and the empty set $\Phi$ are in $D$ (class of sets). For any two sets A and B in $D$, the sets A ∪ B and A ∩ B are again in $D$. If A ∈ B, then the complement $A^C$ is again in $D$. The class of sets $D$ that satisfies these properties is called a field.

DEFINITION 2. The class $D$, in addition to being a field is assumed to be closed under countable union (or equivalently, countable intersection); i.e., if $A_n \in D$ for every $n$, then $A = \cup_n A_n \in D$. Such a class $D$ is called a $\sigma$-field. The 'probability' itself is presumed to be defined on a $\sigma$-field $D$.

REMARK 1. $X_n$ is ergodic if infinitely remote past $F_{-\infty} = \cap_n \sigma(X_k,\ k \leqslant n)$ and infinitely remote future are independent $F_{\infty} = \cap_n \sigma(X_k,\ k \geqslant n)$: for any $A \in F_{-\infty}$, $B \in F_{\infty}$,

$$P(A \cap B) = P(A)P(B).$$

The result of Remark 1 follows from the proof of Theorem 6.1 in Varadhan (2001).

Now, let us assume in addition that $X_n$ is Markov (it is meant 1st order Markov). It is characterized by transition probabilities:

$$p(\alpha;\beta) = P(X_{n+1} = \beta | X_n = \alpha), \quad \alpha, \beta \in V,$$

and $\mu(\alpha) = P(X_n = \alpha)$. Note that $p(\alpha,\beta) = P(X_n = \alpha,\ X_{n+1} = \beta) = p(\beta;\alpha)\mu(\alpha)$. Note the difference: $p(\alpha,\beta)$ is joint probability and $p(\alpha;\beta)$ is transition (conditional) probability. Also,

$$\mu(\beta) = \sum_{\alpha} p(\beta;\alpha)\mu(\alpha), \quad \beta \in V. \tag{2}$$

A probability measure $\tau$ on $V$ is called invariant if (2) holds with $\mu$ replaced by $\tau$.

REMARK 2. According to Chapter 6.3 in Varadhan (2001), $X_n$ is ergodic if it has a unique invariant measure $\mu$. For example, by Theorem 1.10.2 in Norris (1997), invariant measure is unique if $X_n$ is irreducible and positive recurrent:

(i) Irreducible: for all $\alpha, \beta \in V$, there exists $m = m(\alpha, \beta)$such that $X_n$ can reach $\beta$ from $\alpha$ in m steps with positive probability.
(ii) Positive recurrent: For any $\alpha \in V$ the expected return time to $\alpha$ is finite.

Let's assume, that $X_n,\ n = 1, 2, \ldots$ is a stationary 1st order Markov process. We denote:

$$p(\alpha_{i_1}\alpha_{i_2}\ldots\alpha_{i_k}) = P\{X_n = \alpha_{i1},\ X_{n+1} = \alpha_{i_2}, \ldots,\ X_{n+k-1} = \alpha_{i_k}\},$$
$$k = 1, 2, \ldots.$$

Let's assume we know $p(\alpha_i),\ p(\alpha_i\alpha_j), i, j = 1, \ldots, M$.

We will be using Bayes theorem (3) that gives the relationship between the probabilities of $A$ and $B$, $P(A)$ and $P(B)$, and the conditional probabilities of $A$ given $B$ and $B$ given $A$, $P(A|B)$ and $P(B|A)$:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}. \tag{3}$$

**Lemma 1.** *Let the conditions of Markov property and Bayes theorem* (3) *are satisfied. Then approximating the original database by a stationary first order Markov process*

*($m = 1$) for each $\alpha_{i_1}, \ldots, \alpha_{i_m} \in V$, we derive*

$$p(\alpha_{i_1}, \ldots, \alpha_{i_m}) = \frac{p(\alpha_{i_1}, \alpha_{i_2})p(\alpha_{i_2}, \alpha_{i_3}) \ldots p(\alpha_{i_{m-1}}, \alpha_{i_m})}{p(\alpha_{i_2}) \ldots p(\alpha_{i_{m-1}})}, \quad m \geqslant 3.$$

*Proof.* We will be using Markov property for proving this lemma. According to Markov property for each $n \in \{1, \ldots, N\}$, $k$ is the order of Markov process and $\alpha_{i1}, \ldots, \alpha_{i_n} \in V$

$$\begin{aligned} &P\left\{X_n = \alpha_{i_n} | X_{n-1} = \alpha_{i_{n-1}}, \ldots, X_1 = \alpha_{i_1}\right\} \\ &\quad = P\left\{X_n = \alpha_{i_n} | X_{n-1} = \alpha_{i_{n-1}}, \ldots,\ X_{n-k} = \alpha_{i_{n-k}}\right\}. \end{aligned}$$

According to Markov property and Bayes theorem (3):

$$\begin{aligned} p(\alpha_{i_1}, \alpha_{i_2}, \alpha_{i_3}) &= P\left\{X_1 = \alpha_{i_1}, X_2 = \alpha_{i_2},\ X_3 = \alpha_{i_3}\right\} \\ &= P\left\{X_3 = \alpha_{i_3} | X_2 = \alpha_{i_2},\ X_1 = \alpha_{i_1}\right\} P\left\{X_2 = \alpha_{i_2},\ X_1 = \alpha_{i_1}\right\} \\ &= P\left\{X_3 = \alpha_{i_3} | X_2 = \alpha_{i_2}\right\} P\left\{X_2 = \alpha_{i_2}, X_1 = \alpha_{i_1}\right\} \\ &= \frac{P\{X_3 = \alpha_{i_3}, X_2 = \alpha_{i_2}\}}{P\{X_2 = \alpha_{i_2}\}} P\{X_2 = \alpha_{i_2},\ X_1 = \alpha_{i_1}\} \\ &= \frac{p(\alpha_{i_1}, \alpha_{i_2})p(\alpha_{i_2}, \alpha_{i_3})}{p(\alpha_{i_2})}. \end{aligned}$$

For simplicity the case of $m = 3$ is analyzed, and when $m > 3$ the proof is analogous. So, approximating the database by a stationary 1st order process, we have to know only $p(\alpha_{i_1})$ and $p(\alpha_{i_1}, \alpha_{i_2})$ for all $\alpha_{i_1},\ \alpha_{i_2} \in V$. The lemma is proved. □

On the other hand, due to (2),

$$\overline{p}(\alpha_{i_1}, \ldots, \alpha_{i_m}) \approx \frac{\overline{p}(\alpha_{i_1}, \alpha_{i_2})\overline{p}(\alpha_{i_2}, \alpha_{i_3}) \ldots \overline{p}(\alpha_{i_{m-1}}, \alpha_{i_m})}{\overline{p}(\alpha_{i_2})\overline{p}(\alpha_{i_3}) \ldots \overline{p}(\alpha_{i_{m-1}})} \tag{4}$$

for $m \geqslant 3$ and sufficiently large $N$. Therefore, computation of $\overline{p}(\alpha_{i1})$ and $\overline{p}(\alpha_{i_1,}\alpha_{i_2})$ for all $\alpha_{i_1},\ \alpha_{i_2} \in V$ enables us to compute approximately all the frequencies $\overline{p}(\alpha_{i_1}, \ldots, \alpha_{i_m}),\ m \geqslant 3$. For mining frequent sequences the formula (4) should be used together with the following property of frequent sequences: if $\overline{p}(\alpha_{i_1}, \ldots, \alpha_{i_m}) \geqslant \varepsilon$ and $(\alpha_{j1}, \ldots, \alpha_{jl})$ is a subset of $(\alpha_{i_1}, \ldots, \alpha_{i_m})$, then $\overline{p}(\alpha_{j1}, \ldots, \alpha_{jl}) \geqslant \overline{p}(\alpha_{i1}, \ldots, \alpha_{i_m}) \geqslant \varepsilon$; in particular, all the multipliers in the right hand side of (4) should be not less than $\varepsilon$.

**Lemma 2.** *Let the conditions of Markov property and Bayes theorem* (3) *are satisfied. Then approximating the original database by a stationary second order Markov process (m=* 2*) we get:*

$$p(\alpha_{i_1}, \ldots, \alpha_{i_m}) = \frac{p(\alpha_{i_1}, \alpha_{i2}, \alpha_{i_3})p(\alpha_{i_2}, \alpha_{i_3}, \alpha_{i_4}) \ldots p(\alpha_{i_{m-2}}, \alpha_{i_{m-1}}, \alpha_{i_m})}{p(\alpha_{i_2}, \alpha_{i3})p(\alpha_{i_3}, \alpha_{i_4}) \ldots p(\alpha_{i_{m-2}}, \alpha_{i_{m-1}})},$$

$$m \geqslant 4.$$

*Proof.* As in Lemma 1 proof, we will use Markov property and Bayes theorem (3).

$$\begin{aligned}
&p(\alpha_{i_1}, \alpha_{i_2}, \alpha_{i_3}, \alpha_{i_4}) \\
&\quad = P\bigl\{X_{n+3} = \alpha_{i_4} | X_n = \alpha_{i_1},\ X_{n+1} = \alpha_{i_2},\ X_{n+2} = \alpha_{i_3}\bigr\} \\
&\qquad \times P\left\{X_n = \alpha_{i_1},\ X_{n+1} = \alpha_{i_2},\ X_{n+2} = \alpha_{i_3}\right\} \\
&\quad = P\left\{X_{n+1} = \alpha_{i_2} | X_{n+2} = \alpha_{i_3},\ X_{n+3} = \alpha_{i4}\right\} \frac{p(\alpha_{i_1},\alpha_{i_2},\alpha_{i_3})}{p(\alpha_{i2},\alpha_{i_3})} \\
&\quad = \frac{p(\alpha_{i_1},\alpha_{i_2},\alpha_{i_3}) p(\alpha_{i_2},\alpha_{i_3},\alpha_{i_4})}{p(\alpha_{i_2},\alpha_{i_3})}.
\end{aligned}$$

Therefore, computation of $\overline{p}(\alpha_{i_1})$, $\overline{p}(\alpha_{i_1},\alpha_{i_2})$ and $\overline{p}(\alpha_{i_1},\alpha_{i_2},\alpha_{i_3})$ for all $\alpha_{i_1}$, $\alpha_{i_2}$, $\alpha_{i_3} \in V$ enables us to compute approximately all the frequencies $\overline{p}(\alpha_{i_1},\dots,\alpha_{i_m})$, $m \geqslant 4$. The lemma is proved. □

Again according to theorem (1),

$$\overline{p}(\alpha_{i_1},\dots,\alpha_{i_m}) \approx \frac{\overline{p}(\alpha_{i_1},\alpha_{i_2},\alpha_{i_3})\overline{p}(\alpha_{i_2},\alpha_{i3},\alpha_{i_4})\dots\overline{p}(\alpha_{i_{m-2}},\alpha_{i_{m-1}},\alpha_{i_m})}{\overline{p}(\alpha_{i_2},\alpha_{i_3})\overline{p}(\alpha_{i3},\alpha_{i_4})\dots\overline{p}(\alpha_{i_{m-2}},\alpha_{i_{m-1}})},$$

for $m \geqslant 4$ and sufficiently large $N$. Similar procedure can be derived for the approximation of the database by stationary $k$th order Markov process.

## 4. Randon Sampling Method (RSM)

The random sampling method (RSM) analyzes the random sample of the original database and makes statistical inferences about the original database with estimated error probability. Detailed description of RSM method is available in Pragarauskaite and Dzemyda (2011). Brief description of the creation of the random sample of the original database and the RSM method is explained in the following chapters.

### 4.1. *Creating a Random Sample from the Original Database*

The original database random sample $\overline{S}_n$ is chosen so that the central limit theorem could be applied to evaluate error probabilities made by probabilistic algorithm. The original database random sample $\overline{S}_n$ is formed as follows:

- A random sample $\eta_1$, $\eta_2,\dots,$ $\eta_n$ of a random variable $\eta$ taking values $1,\ 2,\dots,\ N$ each with probability $\frac{1}{N}$, is generated.
- When searching for the first level (one-element) frequent sequences, the random sample $\overline{S}_n$ for elements $a_i$ is $x_{\eta 1}, x_{\eta 2},\dots,x_{\eta n}$. The second level random sample for element pairs $a_i a_j$ is $(x_{\eta 1}, x_{\eta 1+1})$, $(x_{\eta 2}, x_{\eta 2+1}),\dots,(x_{\eta n}, x_{\eta n+1})$. The $k$th level random sample for elements $a_i \dots a_k$ is $(x_{\eta 1},\dots,x_{\eta 1+k-1})$, $(x_{\eta 2},\dots,x_{\eta 2+k-1}),\dots,(x_{\eta n},\dots,x_{\eta n+k-1})$, etc. If the sample is formed with

replacement then some integers $\eta_i$ can be the same. The random sample formed without replacement consists of non-repeatable numbers $\eta_i$, by eliminating all repeatable numbers and additionally generating new numbers until we get a sequence $\eta_1, \eta_2, \ldots, \eta_n$ without repeatable integers.

### 4.2. *Analysing the Sample Using Random Sampling Method (RSM)*

Using the exact GSP algorithm (Srikant and Agrawal, 1996; any other exact algorithm could be used in RSM method too), we determine the empirical frequencies of sequences $a_{i_1}, a_{i_2}, \ldots, a_{i_k}$ in the previously formed random sample $\overline{S}_n$

$$\overline{p}_n(a_{i_1}, \ldots, a_{i_k}) = \frac{\#\{j\colon S_{\eta_j} = a_{i_1}, S_{\eta_{j+1}} = a_{i_2}, \ldots, S_{\eta_{j+k-1}} = a_{i_k}\}}{n},$$
$$k = 1, 2, \ldots.$$

Then we choose a number $\delta > 0$ such that $(0 < \varepsilon - \delta < \varepsilon + \delta < 1)$. Sequences $a_{i_1}, a_{i_2}, \ldots, a_{i_k}$ are classified into the following 3 classes:

(1) if $\overline{p}_n(a_{i_1}, \ldots, a_{i_k}) \geqslant \varepsilon + \delta$, then the sequence $a_{i_1}, \ldots, a_{i_k}$ is assigned to the class of frequent sequences;

(2) if $\overline{p}_n(a_{i_1}, \ldots, a_{i_k}) \leqslant \varepsilon - \delta$, then the sequence $a_{i_1}, \ldots, a_{i_k}$ is assigned to the class of rare sequences;

(3) if $\overline{p}_n(a_{i_1}, \ldots, a_{i_k}) \in (\varepsilon - \delta, \varepsilon + \delta)$, then the sequence $a_{i_1}, \ldots, a_{i_k}$ is assigned to the class of intermediate sequences.

The error probabilities can be evaluated as follows. Let's take any sequence $a_{i_1}, \ldots, a_{i_k}$. There are two types of errors:

(1) a sequence is assigned to the class of frequent sequences, but the sequence is rare;

(2) a sequence is assigned to the class of rare sequences, but the sequence is frequent.

Let $p = p(a_{i_1}, \ldots, a_{i_k})$ be the true but unknown frequency of the sequence $(a_{i_1}, \ldots, a_{i_k})$ and the empirical frequency $\overline{p}_n = \overline{p}_n(a_{i_1}, \ldots, a_{i_k})$. It is obvious, that the first type error probability estimate is less than

$$\max_{p<\varepsilon} P\big(\overline{p_n} - p > \delta\big), \tag{5}$$

and the second type error pobability estimate is less than

$$\max_{p\geqslant\varepsilon} P\big(\overline{p_n} - p < -\delta\big). \tag{6}$$

When estimating these probabilities it is convenient to use the following scheme. Let‘s define random variables

$$Z_i = \begin{cases} 1, & \text{if } x_{\eta_i} = a_{i_1},\ x_{\eta_{i+1}} = a_{i_2}, \ldots, x_{\eta_{i+k-1}} = a_{i_k} \\ 0, & \text{otherwise} \end{cases}, \quad i = 1, \ldots, n.$$

By the construction of the sequence $\eta_1, \eta_2, \ldots, \eta_n$, the random variables $Z_1, Z_2, \ldots, Z_n$ are independent and identically distributed with the mean $EZ_i = p$, and the variance $DZ_i = p(1-p)$.

Probabilities (5) and (6) can be estimated using standard statistical methods: using the properties of binomial distribution for a sample with replacement and hypergeometric distribution for a sample without replacement.

Let‘s define the random variable

$$\Sigma_n = Z_1 + Z_2 + \cdots + Z_n.$$

For the sample without replacement, the distribution function of the statistics $\Sigma_n$ is given by

$$F(l, M) = \sum_{i=0}^{l} \frac{C_M^i C_{N-M}^{n-i}}{C_N^n}, \quad l = 0, 1, \ldots, M,$$

where $M = \#\{j \colon x_j = a_{i_1},\ x_{j+1} = a_{i_2}, \ldots, x_{j+k-1} = a_{i_k}\}$. Let us fix $\alpha \in (0,1)$ and define the numbers $\underline{M}$ and $\overline{M}$ as follows:

- $\underline{M}$ is the minimum integer such that

$$F\bigl(\Sigma_n - 1, \underline{M}\bigr) \geqslant 1 - \alpha,$$

- $\overline{M}$ is the maximum integer such that

$$F\bigl(\Sigma_n - 1, \overline{M}\bigr) \leqslant \alpha.$$

The integers $\underline{M}$ and $\overline{M}$ are the parameter‘s $M$ lower and upper $(1-\alpha)$ confidence bounds. Hence,

$$P\{M \geqslant \underline{M}\} = P\left\{p(a_{i_1}, \ldots, a_{i_k}) \geqslant \frac{\underline{M}}{N}\right\} \geqslant 1 - \alpha,$$
$$P\{M \leqslant \overline{M}\} = P\left\{p(a_{i_1}, \ldots, a_{i_k}) \leqslant \frac{\overline{M}}{N}\right\} \geqslant 1 - \alpha.$$

**Lemma 3.** *If the sample size $n$ is sufficiently large, then the asymptotic estimates of error probabilities are effective. By the central limit theorem, for all $a \leqslant b$ we get:*

$$P\left(a\frac{\sqrt{p(1-p)}}{\sqrt{n}} \leqslant \overline{p_n} - p \leqslant b\frac{\sqrt{p(1-p)}}{\sqrt{n}}\right) \to \Phi(b) - \Phi(a), \quad n \to \infty,$$

*where $\Phi$ is the standard normal distribution function.*

*Proof.* Central limit theorem claims that for all $a \leqslant b$:

$$P\left(a \leqslant \frac{Z_n - EZ_n}{\sqrt{DZ_n}} \leqslant b\right) \to \Phi(b) - \Phi(a), \quad n \to \infty.$$

In our case $Z_n = n\overline{p}_n$ therefore we get

$$\frac{Z_n - EZ_n}{\sqrt{DZ_n}} = \frac{n\overline{p}_n - np}{\sqrt{np(1-p)}} = \sqrt{\frac{n}{p(1-p)}}\big(\overline{p}_n - p\big).$$

We rewrite Central limit theorem as follows

$$P\biggl(a\frac{\sqrt{p(1-p)}}{\sqrt{n}} \leqslant \overline{p_n} - p \leqslant b\frac{\sqrt{p(1-p)}}{\sqrt{n}}\biggr) \to \Phi(b) - \Phi(a), \quad n \to \infty.$$

Lemma is proved. □

If $a = -\infty$, then for each $b$

$$P(\overline{p_n} - p \leqslant b\frac{\sqrt{p(1-p)}}{\sqrt{n}}) \to \Phi(b), \quad n \to \infty. \tag{7}$$

If $b = +\infty$, then for each $a$

$$P\biggl(a\frac{\sqrt{p(1-p)}}{\sqrt{n}} \leqslant \overline{p_n} - p\biggr) \to 1 - \Phi(a), \quad n \to \infty. \tag{8}$$

If $n$ is sufficiently large, then using (7) and (8), we have

$$\begin{aligned}\max_{p<\varepsilon} P\big(\overline{p_n} - p > \delta\big) &\approx \max_{p<\varepsilon}\biggl(1 - \Phi\biggl(\delta\frac{\sqrt{n}}{\sqrt{p(1-p)}}\biggr)\biggr)\\ &\leqslant 1 - \Phi\biggl(\delta\frac{\sqrt{n}}{\sqrt{\varepsilon_0(1-\varepsilon_0)}}\biggr),\end{aligned} \tag{9}$$

where $\varepsilon_0 = \min(\varepsilon, \frac{1}{2})$, and

$$\max_{p\geqslant\varepsilon} P\big(\overline{p_n} - p < -\delta\big) \approx \max_{p\geqslant\varepsilon}\biggl(\Phi\biggl(-\delta\frac{\sqrt{n}}{\sqrt{p(1-p)}}\biggr)\biggr) \tag{10}$$

$$\leqslant \Phi\biggl(-\delta\frac{\sqrt{n}}{\sqrt{\varepsilon_1(1-\varepsilon_1)}}\biggr), \tag{11}$$

where $\varepsilon_1 = \max(\frac{1}{2}, \varepsilon)$. If $\overline{p}_n \in (\varepsilon - \delta, \varepsilon + \delta)$, then the classification is undefined, because the classification error probability can be large. The classification error probability depends on how much the true frequency $p$ differs from $\varepsilon$. Assume that $p = \varepsilon$. According to the central limit theorem $P(\overline{p}_n \geqslant \varepsilon) \to \frac{1}{2}$, $n \to \infty$ and $P(\overline{p}_n < \varepsilon) \to \frac{1}{2}$, $n \to \infty$.

Hence it is possible to determine if the sequence $a_{i_1}, \ldots, a_{i_k}$ is frequent or rare only passing over the whole original database. On the other hand, $p$ is close to the empirical frequency $\overline{p}_n$ when $n$ is sufficiently large, because again according to the central limit theorem for any $\mu > 0$

Table 1

Results of RSM method where the sample size $n = 1000$ and $n = 2000$; MPBM 1st, 2nd and 3rd order Markov method and exact GSP method.

| Metrics | GSP method | Markov property based method (MPBM) | | | Random sampling method (RSM) | |
|---|---|---|---|---|---|---|
| | | 1st order | 2nd order | 3rd order | $n = 1000$ | $n = 2000$ |
| Frequent sequences identified | 265 | 268 | 266 | 266 | 193 (163 intermediate) | 212 (86 intermediate) |
| 1st type error (number of rare sequences identified as frequent) | 0 | 6 | 4 | 2 | 25 | 17 |
| 2nd type error (number of frequent sequences identified as rare) | 0 | 3 | 3 | 1 | 15 | 5 |

$$P\big(|\overline{p}_n - p| > \mu\big) \to 0, \quad n \to \infty.$$

The probability of the event $\overline{p}_n \in (\varepsilon - \delta, \varepsilon + \delta)$ can be reduced when reducing $\delta$, but then the first and the second type classification error probabilities are increasing. They can be reduced increasing $n$. So the compatibility between $\delta$ and $n$ is mandatory, and their relation can be expressed as $\delta\sqrt{n} = \mathrm{const}$.

## 5. Experimental Results and Comparison of MPBM and RSM Methods

Markov property based method (MPBM) and random sampling method (RSM) were implemented and tested on the foreign exchange database. The foreign exchange database consists of the currency pair EUR-USD hourly data from 03/01/2000 till 25/05/2012 (data is taken from *Online Trading Platform MetaTrader 4 History Center*). The financial database consists of $N = 4{,}168{,}143$ elements with possible values {A, B, C}, which indicate if the currency exchange rate is growing, falling or being the same as previous hour.

For the original database $S$ analysis we used GSP, MPBM and RSM methods to identify frequent sequences in the original database. Assume that sequence is frequent, if it‘s true frequency is not less than $0.05$ (i.e., minimum support $\varepsilon = 0.05$).

First of all, we analyze the original database using exact GSP method and determine frequent sequences without any errors. Then using the RSM method we determine the frequent sequences in the random sample (for sizes $n = 1000$ and $n = 2000$; $\delta = 0.02$) and MPBM 1st, 2nd and 3rd order Markov method to identify frequent sequences in the original database. The results of both methods are displayed in Table 1.

The results displayed in Table 1 show that MPBM method does fewer mistakes classifying the frequent and rare sequences, whereas for RSM method does more mistakes

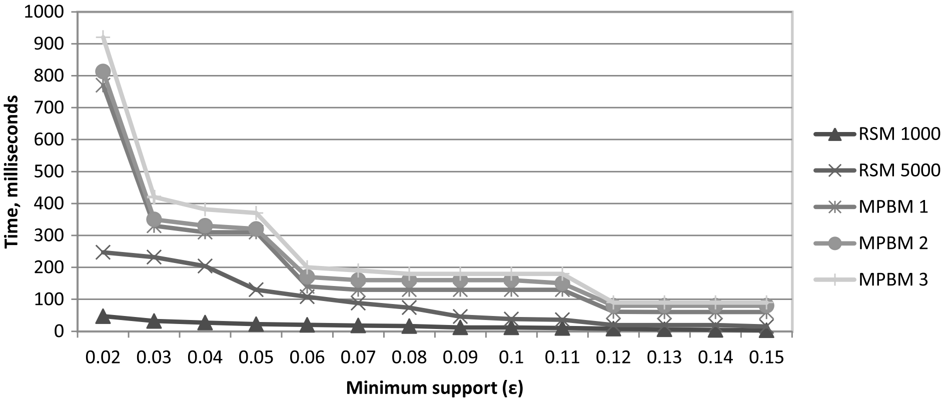

Fig. 1. Time comparison for both approximate methods: RSM and MPBM.

and requires additional analysis on intermediate sequences as their frequency is very near $\varepsilon = 0.05$ and the probability to make a mistake classifying the sequence is high. RSM method is much less time consuming compared to MPBM method as displayed in Fig. 1.

The experiment results show that the 3rd order MPBM method showed the best results on the foreign exchange database; however, it requires reading the original database 4 times and requires more time than RSM method as displayed in Fig. 1. RSM method is less time consuming as it is not reading the original database, but only the random sample on the original database; however it has a quite large number of intermediate sequences. In order to make the classification if such sequence is rare or frequent, it would be beneficial to find their exact frequencies in the original database. It can be done using any exact method, e.g., GSP. The classification error probabilities made by RSM method are estimated below. According to the (9) formula the first type classification error probability, when the frequent sequence is assigned to the class of rare sequences is given by:

$$n = 1000: \quad 1 - \Phi\left(\delta\frac{\sqrt{n}}{\sqrt{\varepsilon(1-\varepsilon)}}\right) = 1 - \Phi(2.9019) \approx 0.0019;$$

$$n = 2000: \quad 1 - \Phi\left(\delta\frac{\sqrt{n}}{\sqrt{\varepsilon(1-\varepsilon)}}\right) = 1 - \Phi(4.1039) \approx 0.0003.$$

According to (10), the second type classification error probability (the rare sequence is assigned to the class of frequent sequences) is given by:

$$n = 1000: \quad \Phi\left(-\delta\frac{\sqrt{n}}{\sqrt{\varepsilon_1(1-\varepsilon_1)}}\right) = \Phi(-2\delta\sqrt{n}) = \Phi(-1.2649) \approx 0.103;$$

$$n = 2000: \quad \Phi\left(-\delta\frac{\sqrt{n}}{\sqrt{\varepsilon_1(1-\varepsilon_1)}}\right) = \Phi(-2\delta\sqrt{n}) = \Phi(-1.7889) \approx 0.0368.$$

## 6. Conclusions

In this paper the approximate Markov method (MPBM) for mining frequent sequences is presented. The method is based on the Markov property and requires only several passes over the original database, whereas the exact methods (e.g., GSP) make multiple passes over the original database. The MPBM method is compared to the random sampling method (RSM), which analyzes in a specific way generated random sample of original database and makes statistical inference about frequent sequences in the original database. Both methods were implemented and tested on the foreign exchange database. Approximate methods are much faster than exact algorithms and suitable quick statistical analysis in the areas where the precision is less important than the decision time.

Experiment results showed, that the proposed Markov method (MPBM) works efficiently on the foreign exchange database, especially with the 3rd order Markov model. The random sampling method (RSM) is even less time consuming than Markov method, as it does not require reading the full database only the random sample of the original database, but requires further analysis on intermediate sequences.

**Acknowledgments.** The authors would like to thank prof. Remigijus Mikulevicius for valuable comments related to Markov models and its approximation.

## References

Agrawal, R., Imielinski, T., Swami, A. (1993). Mining association rules between sets of items in large databases. *ACM SIGMOD Record*, 22, 207–216.

Agrawal, R., Srikant, R. (1994). Fast algorithms for mining association rules. In: *Proceedings of the 20th International Conference on Very Large Data Bases (VLDB)*, Vol. 1215, pp. 487–499.

Agrawal, R., Srikant, R. (1995). Mining sequential patterns. In: *Proceedings of the Eleventh International Conference on Data Engineering*, IEEE, Comput. Soc., Los Alamitos, pp. 3–14.

Ayres, J., Flannick, J., Gehrke, J., Yiu, T. (2002). Sequential pattern mining using a bitmap representation. In: *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, New York, pp. 429–435.

Brin, S., Motwani, R., Ullman, J.D., Tsur, S. (1997). Dynamic itemset counting and implication rules for market basket data. *ACM SIGMOD Record*, 26, 255–264.

Gouda, K., Hassaan, M., Zaki, M.J. (2007). Prism: a primal-encoding approach for frequent sequence mining. In: *Data Mining (ICDM 2007)*, IEEE, Comput. Soc., Los Alamitos, pp. 487–492.

Gouda, K., Hassaan, M., Zaki, M.J. (2010). Prism: an effective approach for frequent sequence mining via prime-block encoding. *Journal of Computer and System Sciences*, 76(1), 88–102.

Han, J., Cheng, H., Xin, D., Yan, X. (2007). Frequent pattern mining: current status and future directions. *Data Mining and Knowledge Discovery*, 15(1), 55–86.

Han, J., Pei, J., Yin, Y. (2000). Mining frequent patterns without candidate generation. *ACM SIGMOD Record*, 29, 1–12.

Han, J., Pei, J., Mortazavi-Asl, B., Pinto, H., Chen, Q., Dayal, U., Hsu, M.C. (2001). Prefixspan: mining sequential patterns efficiently by prefix-projected pattern growth. In: *Proceedings of the 17th International Conference on Data Engineering*, pp. 215–224.

Kum, H.C., Pei, J., Wang, W., Duncan, D. (2003). ApproxMAP: approximate mining of consensus sequential patterns. In: *Proceedings of the 2003 SIAM International Conference on Data Mining*, pp. 311–315.

Norris, J.R. (1997). *Markov Chains*. Cambridge University Press, Cambridge.

*Online Trading Platform MetaTrader 4 History Center*.
URL: `http://www.metaquotes.net/data_center`.

Park, J.S., Chen, M.S., Yu, P.S. (1995a). An effective hash-based algorithm for mining association rules. In: *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data*, Vol. 24, ACM, New York, pp. 175–186.

Park, J.S., Chen, M.S., Yu, P.S. (1995b). Efficient parallel data mining for association rules. In: *Proceedings of the Fourth International Conference on Information and Knowledge Management*, ACM, New York, pp. 31–36.

Pragarauskaite, J., Dzemyda, G. (2011). Probabilistic algorithm for mining frequent sequences. In: *Proceedings ASMDA 2011*, Sapienza University of Rome, Edizioni ETS, pp. 1454–1460.

Sarawagi, S., Thomas, S., Agrawal, R. (2000). Integrating association rule mining with relational database systems: alternatives and implications. *Data Mining and Knowledge Discovery*, 4(2), 89–125.

Savasere, A., Omiecinski, E.R., Navathe, S.B. (1995).*An Efficient Algorithm for Mining Association Rules in Large Databases*. Georgia Institute of Technology.

Srikant, R., Agrawal, R. (1996). Mining sequential patterns: generalizations and performance improvements. In: *Advances in Database Technology EDBT'96*, pp. 1–17.

Tumasonis, R., Dzemyda, G. (2004). The probabilistic algorithm for mining frequent sequences. In: *Proceedings of Eight East-European Conference on Advances in Databases and Information Systems (ADBIS'04)*, pp. 89–98.

Varadhan, S.R.S. (2001). *Probability Theory* (Courrant Lecture Notes), AMS, New York.

Zaki, M.J. (2000). Scalable algorithms for association mining. *IEEE Transactions on Knowledge and Data Engineering*, 12(3), 372–390.

Zaki, M.J. (2001). Spade: an efficient algorithm for mining frequent sequences. *Machine Learning*, 42(1), 31–60.

**J. Pragarauskaitė** is a PhD student at the System Analysis Department of the Institute of Mathematics and Informatics, Vilnius University, Lithuania. She received her MSc in computer science from the Vilnius University, the Faculty of Mathematics and Informatics, Lithuania in 2008. Her research areas are data mining algorithms and frequent sequence mining algorithms for finding frequent patterns in large databases.

**G. Dzemyda**, habil. dr., is a member of Lithuanian Academy of Sciences, professor, principal researcher and director of the Institute of Informatics and Mathematics of Vilnius University. His main research interests include optimization and visualization, data mining, and medical informatics.

## Markovo modelių taikymas dažnų sekų paieškai finansiniuose duomenyse

Julija PRAGARAUSKAITĖ, Gintautas DZEMYDA

Dažnų sekų paieška didelėse duomenų bazėse yra svarbi biologinių, klimato, finansinių, internetinių ir daugelio kitų duomenų bazių analizei. Tikslieji metodai, skirti dažnų sekų paieškai, daug kartų perrenka visą duomenų bazę. Jeigu duomenų bazė didelė, tai dažnų sekų paieška yra lėta arba labai brangi užduotis. Apytiksliai dažnų sekų paieškos algoritmai yra žnymiai greitesni, nes užuot perrinkę visą duomenų bazę, jie analizuoja tam tikru būdu sudarytą pradinės duomenų bazės imtį. Šiame straipsnyje pasiūlytas Markovo savybe pagrįstas apytikslis metodas (MPBM), kuris tik kelis kartus perrenka duomenų bazę. Šis metodas yra realizuotas ir ištestuotas naudojant realius valiutų kursų duomenis bei palygintas su kitais tiksliaisiais ir apytiksliais dažnų sekų paieškos metodais.