ISSN: 0974-2115
www.jchps.com Journal of Chemical and Pharmaceutical Sciences

Value and Pattern Anonymization of Time Series Data for Privacy Preserving Data Mining

J.S.Adeline Johnsana[*1], A.Rajesh[2], S.Sangeetha[2] and S.Kishore Verma[3]
[1]Computer Science and Engineering, St.Peters University, Avadi, India.
[2]Computer Science and Engineering, C. Abdul Hakeem College of Engineering and Technology, Melvisharam.
[3]Computer Science and Engineering, SCSVMV University, Kanchipuram, India.
***Corresponding author: E-mail: Email Id:adeline.j.s@gmail.com**

ABSTRACT

Time Series Data Mining (TSDM) as of late pulled in the consideration of researchers in Data Mining because of expansion accessibility of data with temporal dependency. The reputation of these data on the internet has nurtured the most innovative applications extending from financial analysis to social community tracking and partner matching. However, such a monstrous data additionally suggest tremendous measure of privacy, which if not properly ensured, might get to be exploited as a hotspot for misuse and violations. To protect privacy on time series data more number of techniques have been proposed, out of which the conventional k-anonymity picked up the significance, yet it comes up short in giving limited protection to the patterns of the time series data as it might endure extreme pattern loss. In this paper a combination of novel methodologies K-anonymization (SKY), Symbolic polynomial with cross validation for pattern representation is proposed to reveal a promising level information loss and pattern loss for the Privacy Preserved Data Mining field of exploration.

KEY WORDS: Privacy, Time series, pattern, polynomial, anonymity.

1. INTRODUCTION

Time series has for quite some time been viewed as a standout amongst the most critical sorts of data accessible in both nature and human culture. Privacy protection in the distribution of time series is a testing theme for the most part because of the mind boggling nature of the data and the way that they are utilized. Specifically, the spectrum of frequently used "complex" queries on time series covers not just range queries on the attribute values at specified time instants but also pattern similarity queries which treat each sequence more comprehensively.

The traditional answer for anticipate linkage attacks is to authorize k- anonymity, on the published database, with the goal that every record has its QI attributes indistinguishable to at least k -1 other records. Although conventional k-anonymity can be used to resist linkage attacks, it cannot effectively preserve the patterns, which are critical for performing queries on time series. Another issue with anonymization based on pattern similarity is that the ranges on attribute values of an envelope may be vast, in this manner fundamentally harming the query accuracy.

Time series data mining (TSDM) as of late pulled in the consideration of researchers in data mining because of the expansion accessibility of data with temporal dependency. TSDM algorithms such as classification/ clustering of time series, pattern extraction, and similarity search require a distance measure between time series. The computation of these distances is chiefly done utilizing the established Euclidean distance or the Dynamic Time Warping distance. These computations might prompt untraceable expenses for long series and/or tremendous databases. Henceforth, many approximate representations of time series have been created in the course of the most recent decade. Symbolic representation is one strategy to approximate time series. The most utilized symbolization method is called SAX (Symbolic Aggregate approximation). It is an exceptionally basic system to symbolize time series without the need for any a priori information.

The nature of the SAX representation of a time series relies on i) the Number of PAA coefficients, i.e. the number of segments the time- series is divided in, ii)) the number of symbols utilized for quantization (the alphabet size), iii) the gaussianity assumption. A few works have tended to these issues. However, the simplicity of SAX is lost by introducing a pre-processing phase using a clustering method. Other approaches endeavour to enrich the PAA representation and, further, the SAX symbols. Extended SAX (ESAX) associates the symbolic minimum and maximum of the PAA segment to the related SAX symbol and in addition the order of their occurrences.

The proposed work called Symbolic Polynomials (SymPol) has fast technique to process sliding window content which has a linear run-time complexity. Our guideline depends on detecting local polynomial patterns which are extracted with a sliding window approach, subsequently fitting one polynomial to each sliding window segment. Once the polynomial coefficients of each sliding window content are computed, we convert those coefficients into symbolic forms (i.e. alphabet words). The inspiration for calling the strategy Symbolic Polynomial emerges from that procedure. Such a discretization of polynomial coefficients, as words, permits the discovery of similar patterns by converting similar coefficient values into the same word.

However, in the vicinity of noise high degree polynomials can over-fit the content of the data. Over-fitting prompts different polynomial coefficients for similar patterns. Such a problem arises because the complexity of the approximative model is higher than the bias (informative signal) of the data. Though, as a rule models are sufficiently

prepared to keep away from over-fitting, but in general there is a manual intervention required to ensure the model does not demolish more than enough attributes. There are at least two approaches to handle over-fitting issues, by (i) diminishing the complexity of the approximative model, or (ii) strongly regularizing/penalizing the high coefficients of the polynomial. The optimal degree of polynomial or regularization parameter is chosen in light of the method called Cross Validation utilizing only training instances.

The method utilizes an equi-area discretization of the distributions of the polynomial coefficients to compute the symbolic words. Threshold values separate the distribution into equal volumes and each volume is assigned one alphabet letter. Consequently, each polynomial coefficient is assigned to the area its value belongs to, and is replaced by the area's character. Ultimately, the word of a polynomial is the concatenation of the characters of each polynomial coefficient merged together. The words of each time series are then stored in a separate 'bag'. Such a representation offers a powerful mean to reflect which patterns (i.e. symbolic polynomial words) and how often they occur in a series (i.e. the frequency value in each histogram cell). The technical novelty of our method, compared to state-of-art approaches (SAX, Id-SAX) which utilize constant functions to express local patterns of series, relies on offering an expressive technique to represent patterns as polynomials of arbitrary degrees. Furthermore, we present a fitting algorithm which can compute the polynomial coefficients in linear time; therefore our method offers superior expressiveness without compromising run-time complexity.

2. RELATED WORK

In this section, we will outline existing works of representation and anonymization of time series data. The large amount of information easily accessible today and the increased computational power available to the attackers make such linking attacks a serious problem. Formal foundation for the anonymity problem against linking and for the application of generalization and suppression techniques towards its solution provided in former works. Data publishing is essential for providing resources for research and for the transparency of government institutions and companies. However, data publishing is also risky since published data may contain sensitive information. This paper addresses the privacy issues regarding the identification of individuals in static trajectory datasets.

Previous work for privacy preserving data mining uses a perturbation approach which reconstructs data distributions in order to perform the mining. Since it does not use the multi-dimensional records, but uses aggregate distributions of the data as input. This leads to a fundamental re-design of data mining algorithms. To overcome this, propose a new framework for privacy preserving data mining of multi-dimensional data. it maps the original data set into a new anonymized data set, give a detailed analysis of following two attacks that a k-anonymized dataset has some subtle, but severe privacy problems. First, an attacker can discover the values of sensitive attributes when there is little diversity in those sensitive attributes. Second, attackers often have background knowledge, and we show that k-anonymity does not guarantee privacy against attackers using background knowledge. This paper introduced ℓ-diversity, a framework that gives stronger privacy guarantees.

The ultimate target of representation methods is to encapsulate the regularities of time-series patterns by omitting the intrinsic noise. Discrete Fourier transforms have attempted to represent repeating series structures as a sum of sinusoidal signals. Similarly, wavelet transformations approximate a time-series via orthonormal representations in the form of wavelets. In addition to those approaches, researchers have been also focused on preserving the original form of the time series without transforming them to different representations. Nevertheless, the large number of measurement points negatively influences the run-time of algorithms. Attempts to shorten time series by preserving their structure started by linearly averaging chunks of series points. Those chunks are converted to a single mean value and the concatenation of means create a short form known as a Piecewise Constant Approximation. Another technique operates by converting the mean values into a symbolic form and is called Symbolic Aggregate Approximation, denoted shortly as SAX.

Definitions and Notations: In this section, we first introduce preliminaries of conventional k-anonymity and the definitions of patterns. At that point, we define the issue that we attempt to explain.

Conventional K-Anonymity of Time Series: The conventional k-anonymity of time series presumes that every original time series (record) r in a database T contains the subsequent three parts of data:

- an identifier *id*;
- a set of quasi-identifier (QI) attributes at n different but typically successive time instants, indicate by QI $=\{I_1, I_2; \ldots ; I_n\}$;
- a set of sensitive attributes which are indicated by I_s as an entirety.

During the anonymization, all records are initially de identified. The conventional k-anonymity ensures that all the QI attribute values of each record in the published data, namely $R_1 \ldots\ldots R_n$, are identical to at least k -1 other record.

Table.1. K-Anonymized Table (Unit: thousand)

Id	2011	2012	2013	2014	2015	2016
1	157-181	147-188	134-197	125-213	112-221	200
2	157-181	147-188	134-197	125-213	112-221	180
3	157-181	147-188	134-197	125-213	112-221	160
4	54-120	59-125	67-132	51-151	56-161	110
5	54-120	59-125	67-132	51-151	56-161	85
6	54-120	59-125	67-132	51-151	56-161	90
7	63-600	47-300	38-100	20-500	20-1000	55
8	63-600	47-300	38-100	20-500	20-1000	46
9	63-600	47-300	38-100	20-500	20-1000	600
10	800-800	600-600	500-500	200-200	600-600	400

δ- constraint K-anonymity Problem: A data stream S is an infinite time sequence with the incremental order S = {<s_1,p_1>, ……,<s_n,p_n>,…..}; here s_i is a tuple with sequence number p_i, and $p_i < p_j$ denotes that p_i arrives before p_j. Each tuple s_i comprises a vector of m values (a1, a_2, …., a_m), where a_i is drawn from a finite domain D_i.

Given an input stream S_1, an output stream S_2 that it satisfies the K-anonymity property with respect to quasi-identifier QI is produced. Moreover, the output stream order deviates from the input stream order by at most δ.

Specialization Tree: For each quasi-identifier attribute qi ϵ { q1, ……, qm } , We assume that it has a pre-defined domain generalization hierarchy DGHi. A specialization tree is defined as a directed tree, where each node is a vector < v1, ……,vm>, of which vi is drawn from DGHi. The root of specialization tree is the most general node; that is, the value of each of its attributes is the root value of the corresponding domain generalization hierarchy. If there is a directed edge from node u to node v in the specialization tree, it must satisfy the following conditions:

ɔj ϵ 1,…., m such that domain generalization hierarchy DGH_j contains the edge v (q_j) → u (q_j).

For all other i ϵ 1,……, m and i ≠ j, u (q_i) = v (q_i).

In the specialization tree, all the child nodes of a node u must split the same attribute‘s domain of u.

Pattern Representation: Given the m correlation functions defined for patterns, a pattern representation (SymPol) of a time series is an entity which, a) Can be obtained by a transformation M (.) from the time series itself; and b) Can be transformed to a pattern determinedly. Given a time series r, they denote by PR[r] = M(r) its pattern representation. The second item in the above ensures that a pattern can be reconstructed from its PR. Therefore, the pattern matching range/similarity queries defined previously can be per-formed on the reconstructed patterns.

When generating PRs, transformation M (.) would certainly incur information loss, which is called the pattern loss. As a result, the reconstructed *pattern* might be distorted, leading to inaccuracy in the subsequent pattern matching queries.

Table.2. Anonymized pattern using SYMPOL approach (Unit: Thousand)

Id	2012	2013	2014	2015	2016	Original PR	Anonymized PR
1	147-188	134-197	125-213	112-221	200	cabbcc	aaaaab
2	147-188	134-197	125-213	112-221	180	ccbcba	aaaaab
3	147-188	134-197	125-213	112-221	160	ccbcab	aaaaab
4	59-125	67-132	51-151	56-161	110	ccacbc	ccacbc
5	59-125	67-132	51-151	56-161	85	cabbcc	ccacbc
6	59-125	67-132	51-151	56-161	90	cbcacb	ccacbc
7	47-300	38-100	20-500	20-1000	55	ccaccb	ddcadd
8	47-300	38-100	20-500	20-1000	46	abbccc	ddcadd
9	47-300	38-100	20-500	20-1000	600	bacbcc	ddcadd
10	600-600	500-500	200-200	600-600	400	ccbcba	aaaaaa

Utility Measures: To produce the utility measures of anonymity model, including the infringe probability, which represents the privacy preservation ability and the information loss which represents the utility of available data. There are two variety of information loss, immediate value loss (VL) and pattern loss (PL).

Instant value Loss Metric: For a categorical attribute, given a value v in its DGH, the information loss of the information loss of the value v is defined as follows,

$$\text{Instant value loss (IVL)} = \frac{|S_v|-1}{|S|-1}$$

Here S_v is a set of leaf nodes of the subtree rooted at v in the DGH, and S is the set of leaf nodes in the same DGH. Intuitively, the information loss of a leaf node in DGH is 0 and the information loss of root is 1 according to the definition.

For continuous attributes, given a value interval I = [l,u] from domain [L, U], its information loss is defined as follows,

$$\text{Instant value loss (IVL)} = \frac{u-l}{U-L}$$

Using the above formulas, we can calculate the amount of information loss of each generalization node in a specialization tree. The information loss of child node must be less than that of its parent node in the specialization tree.

Pattern Loss Metric: The pattern loss can be considered by the distance p(Q) and p^{*} (Q) which characterize the pattern information sealed in PR[Q]

$$\text{PLM}(Q) = \text{distance}(p(Q), p*(Q))$$

Where distance (.) is a distance measure (using cosine distance metric) defined in the feature vector space of patterns.

Shortcoming of SAX and 1D-SAX: The SAX representation explicate above only on the average value of the time series on each division. Consequently, two subdivision having dissimilar properties but with close averages will be quantized into the identical symbol.

The 1D-SAX words are built from locally constant approximation which are generally less expressive than the Polynomials representations. For example Figure demonstrates the deficiencies of the locally constant approximation in detecting the curvature of a sliding window sub-series. We have fit both a constant model and our polynomial model to the series data. Assume we want to have a four character 1D-SAX word for each of the sliding windows segment. As we can see our method can accurately distinguish between those patterns, while the 1D-SAX method averages the content and loses information about their curvatures. Polynomial method is more expressive even though it has exactly the same complexity. In this case both methods use four characters. In terms of run-time, 1D-SAX needs only one pass through the data, so from the algorithmic complexity point of view both methods have the same algorithmic complexity.

So to defeat this draw backs the planned methods explains about both the average and trend values taken into account and to prevent loss of information occurred in the curvature of time series that novel technique is called Symbolic Polynomials (SymPol).

Algorithms and Implementation issues: This section presents our algorithms which transform an original data set and produce output conforming to (k,P)-anonymity. We start by introducing a specific PR extraction technique based on the well-known Symbolic Polynomial (SymPol) with Cross validation method of time series. Then, we propose a naïve SKY algorithm as an extension to a conventional k-anonymity algorithm.

The SKY Algorithm: This section proposes the SKY algorithm to address the δ-constraint k-anonymity problem.

Algorithm SKY

Input: Stream S, specialization tree Sptree, parameter k of k-anonymity, parameter δ of δ- constraint

Output: Anonymized tuples.

WHILE (*True*)
Read a new tuple *t* from *S*
Search *Sptree* from the root until reach a node *g* which is the most specific node that contains *t,*
IF (*g* is a *work node*)
Aonoymize tuple *t* with *g* and
output it ;
ELSE
Insert *t* into *g's* buffer *FS(g)*;
IF (*FS(g)* satisfies k-anonymity)
Anonymize and output *FS(g)* with *g;*
Label *g* as a *work node;*
Drop *FS(g);*
END IF
END IF
Process Delta (δ);
END WHILE

In SKY, the nodes of the specialization tree are grouped into two classes, candidate nodes and work nodes. The candidate nodes are those nodes whose current set of matching tuples does not satisfy k-anonymity. The work nodes, on the other hand, are those nodes whose current set of matching tuples satisfies k-anonymity property. Each candidate node g is assigned to a buffer FS (g) as holding area.

When a tuple t is read from the input stream, SKY searches the specialization tree to find the most specific generalization node g that contains t. That is, g is a generalization of t and no child node of g is generalization of t. If g is a work node, then tuple t is anonymized with g and output immediately. Otherwise g is a candidate node, which means that tuples anonymized by g still do not satisfy the k-anonymity property, then t is stored into FS (g)

satisfies k-anonymity property or δ-constraint is satisfied. The function Process Delta is devised to check whether a tuple in the buffer satisfying δ-constraint exists, if yes, then anonymize and output it.

Local Polynomial Fitting:

Definition 1. (Polynomial): A polynomial of degree d with coefficients $\beta \in R^{d+1}$ defined as a sum of terms known as monomials. Each monomial is a product of a coefficient with a particular power of the predictor value $X \in R^N$, as shown in below equation.

$$\hat{Y} = \sum_{j=0}^{d} \beta_j X^j = Z\beta,$$

Where Z: = (X^0, X^1, X^2, …,X^d)

Definition 2. (Sliding Window Segment): A sliding window segment is a continuous subsequence of S having length n denoted by $S_{t,n} \in R^n$. The time index t represents the starting point of the series, while the index n the length of the sliding window, i.e. $S_{t,n} = [S_t^{(i)}, S_{t+1}^{(i)}, \ldots\ldots\ldots S^{(i)}_{t+n-1}]$. The total number of sliding window segments of size n for a series of length N is N-n+1, in case we slide the window by incrementing the start index t by one index at a time.

Our method operates by sliding a window throughout a time-series and computing the polynomial coefficients in each sliding window segment. The segment of time series inside the sliding window is normalized before being approximated to a mean 0 and deviation of 1. The incremental step for sliding a window is one, such that every segment is considered. Computing the coefficients of a polynomial regression is conducted by minimizing the least squares error between the polynomial estimates and the true values of the segment. The objective function is denoted by L and is shown in Equation 1. The task is to fit a polynomial that approximates the real values Y of the time-series window of length n, previously denoted as $S_{t,n}$.

$$L(Y,\hat{Y}) = \|Y-Z\beta\|^2 \quad (1)$$

The predictors represent the time indexes X= [0, 1, ……n-1] and are converted to the linear regression form by introducing a Vandermonde matrix $Z \in R^{MxN}$ as shown in below

$$\begin{pmatrix} 0^0 & 0^1 \ldots\ldots\ldots\ldots & 0^d \\ 1^0 & 1^1 \ldots\ldots\ldots & 1^d. \\ \vdots & \vdots \quad \ldots\ldots\ldots\ldots & \vdots \\ (n-2)^0 & (n-2)^1 \ldots\ldots & (n-2)^d \\ (n-1)^0 & (n-1)^1 \ldots\ldots & (n-1)^d \end{pmatrix}$$

The solution of the least squares system is conducted by solving the first derivative with respect the polynomial coefficients β are presented in Equation (2)

$$\frac{\partial L(Y,\hat{Y})}{\partial \beta} = 0$$

$$\beta = (Z^T Z)^{-1} Z^T \quad (2)$$

Since the relative time inside each sliding window is between 0 and n-1, the predictors Z are the same for all the sliding windows of all-time series. Consequently, we can pre-compute the term $P = (Z^T Z)^{-1} Z^T$ in the beginning of the program and use the projection matrix P to compute the polynomial coefficients β of any local segment Y as β = PY. The optimal degree of polynomial or regularization parameter is chosen in light of the method called Cross Validation utilizing only training instances. Because we provide optimal degree as a input to find the polynomial coefficients in order to avoid over fitting of training set. Below algorithm describes the Cross validation method to find the regularization parameter d'.

Algorithm to find the optimal degree of Polynomial

Input: Data set of size n, Polynomial degree d and polynomial coefficients Φ

Output: Optimal polynomial degree d'

Split the data set of size n into three parts training set m(60%), cross validation set mcv (20%), test set mtest (20%)

{Compute Training Error TRE (Φ)}

for j ϵ {1, …….,d} do

$h_\Phi (x^d) = \Phi_0 + \sum_{j=1}^{d} \Phi_j x^j$

for i ϵ {1,…,m} do

$TRE(\Phi)=1/2m\sum_{i=1}^{m}((h_\Phi(x^{(i)})-y^{(i)}))^2$

end for

{Compute Cross validation Error CVE (Φ)}

for k ϵ {1, ……., mcv} do

$CVE(\Phi)= 1/2mcv \sum_{k=1}^{mcv}((h_\Phi(x^{(k)}))- y^{(k)}))^2$

end for

{Compute Test Error TE (Φ)}

for l ϵ {1,..,mtest} do

TE(Φ)=1/2mtest $\sum_{l=1}^{mtest}((h_{\Phi}(x^{(l)})) - y^{(l)}))^2$
end for
end for
Plot (X,Y) = (Degree of Polynomial d, {Cross validation error CVE(Φ), Training Error TRE(Φ)})
d'→ Choose corresponding degree which has minimum CVE(Φ)
return d' {optimal value of polynomial degree}

Below mentioned algorithm describes the steps needed to compute the polynomial coefficients of all sliding windows (starting at t) of every time series (indexed by i) in the dataset. For every time series we collect all the polynomial coefficients in a bag, denoted as $\Phi^{(i)}$. The outcome of the fitting process is the bags of all-time series Φ.

Algorithm: Polynomial Fitting of a Time-Series Dataset
Input: Dataset T ∈ $R^{MxN,}$ Sliding Window size n, Polynomial degree d'
Output: Polynomial coefficients Φ ∈ $R^{M\ x\ (N-n)\ x\ (d+1)}$
Compute the term P
$P = (Z^T Z)^{-1} Z^T$
for$i \in \{1 \ldots . M\}$do
$\Phi^{(i)} \leftarrow \emptyset$ fort ∈
$\{1 \ldots . . N - n + 1\}$do
$Y \leftarrow [S_t^{(i)}, S_{t+1}^{(i)}, \ldots\ldots\ldots S^{(i)}_{t+n-1}]$
$\beta \leftarrow PY$
$\Phi(i) \leftarrow \Phi(\mathbf{i}) \cup \{\beta\}$
end { for statements end }
return $(\Phi^{(i)})i = 1 \ldots . M$

Converting Coefficients to Symbolic Words: The next step is to convert the computed polynomial coefficients Φ from Algorithm1 into words. The aim of the conversion is to transform each of the d+1 coefficients of every β of Φ to one symbol. Therefore, the extracted words have lengths ofd+ 1 symbol. The first phase of algorithm computes the threshold values $\mu^j_k$to discretize the distribution of each coefficient in an equi-area fashion. The second phase processes all the coefficients β of time-series sliding windows and converts each individual coefficient to a character c, depending on the position of the β values with respect to the threshold values. The concatenation operator is denoted by the symbol o. The characters are concatenated into words ω and stored in bags of words W. While the size of the sliding window n is a significant value, still it is a constant with respect to the data size N and M.

Algorithm: Convert Polynomial Coefficients to Words
Input: Polynomial Coefficients Φ, Alphabet Size α
Output: W∈ $R^{Mx(N-n)\ x\ (d+1)}$
{Compute the thresholds}
for j ∈{0 ….d}do
$B^j \leftarrow sort(\{ \beta j \mid \beta \in \Phi^{(i)},$ i=1…….M})
$s^j \leftarrow |B^j|$
$\mu^j_k \leftarrow \infty$
for k∈ $\{1, \ldots \ldots \alpha - 1\}$do
$\mu^j_k \leftarrow B^j\lfloor s^j k/\alpha \rfloor$
end { for statements end }
{Convert the coefficients to words}
$\Sigma \leftarrow \{A, B, C, \ldots . . , Y, Z\}$
for i ∈ $\{1 \ldots . . M\}$do
$W^{(i)} \leftarrow \emptyset$
for β ϵ Φ(i) do
ω← ∅
for j ∈ {0 ….d} do
k← $argmax_k \in \{1, \ldots . . , \alpha\}$ β <μ_k^j
$\omega \leftarrow \omega$ o Σ_k
end for
$W^{(i)} \leftarrow W^{(i)} \cup \{\boldsymbol{\omega}\}$
end {for statements end}
return ($W^{(i)}$)**i=1……M**

Tuning k and d: We will show how IVL (.) and PLM (.) changes by varying parameter k and d in this experiment. By using constraint d = d'- (k-1) we evaluate the d value based on the given d' and k value. The results confirm that

larger d will cause tremendous information loss. Therefore, on the premise of enough privacy protection stability optimal degree d' always greater than k value i.e.d'>k.

3. EXPERIMENTAL RESULTS

To evaluate the effectiveness and efficiency of our algorithms, we conduct an extensive experimental study on both real and synthetic data sets. All our experiments are conducted on a PC with a Pentium Dual-Core CPU and 3 GB main memory running in the Microsoft Windows 7 Ultimate.

In this experiment, due to the limited feasibility in getting benchmark dataset. We had created my own dataset based on time series data that well stood against the benchmark database. In this dataset, it consists of three kinds of time series about 121,146 tuples in which attributes named as Identifiers, Quasi Identifiers and different sensitive attributes.

Initially 1000 time series of data are generated as Micro data. From that series based upon the given numbers the sensitive attributes are selected. Next, value based K-anonymity is performed and IVL metric is used to measure the quality of the output anonymized tuples. From fig 1, we can see that the SKY outperforms compared to traditional anonymization strategies. Third step is representation of pattern based on the implementation of Symbolic Polynomials (SymPol) is performed to minimize the pattern loss by evaluating PLM metric (fig 2) which consequently improves the efficiency of pattern based anonymization.

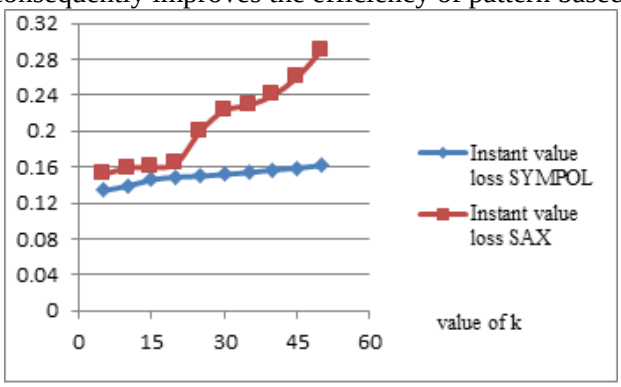

Fig.1.The anonymization quality of SKY (IVL metric)

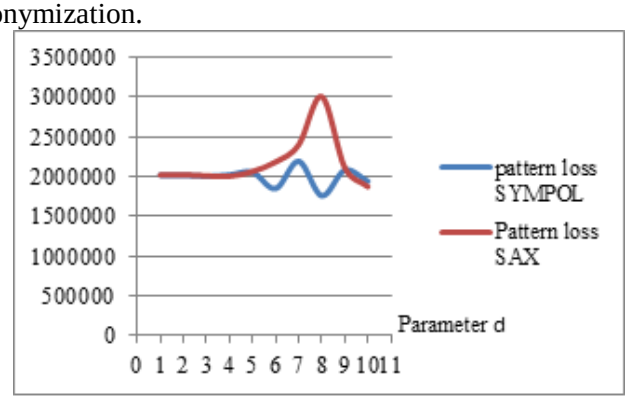

Fig.2.Pattern loss (PLM metric)

4. CONCLUSION

Our works attempts to integrate two fields: value based and pattern based privacy protection. Since the K-anonymity (SKY) and Symbolic Polynomials was proposed for privacy protection, many works have been proposed to generate anonymization data sets effectively and efficiently. However, we have shown that our SymPol (Symbolic Polynomials) method provide a better fitting and prevent the loss of curvature information compared to existing approaches like SAX method. As well as the main advantage of using SKY anonymization method is the existing anonymized tuples will not be modified when inserting new tuples. Conclusion of this work provides both the techniques may greatly reduce the information loss.

REFERENCES

Chen Y, Dong G, Han J, Wah B.W, and Wang J, Multidimensional regression analysis of time-series data streams, In Proc. 28th Int. Conf. Very Large Data Bases, ser. VLDB '02. VLDB Endowment, 2002.

Dewri R, Ray I, and Whitley D, On the Optimal Selection of k in the k-Anonymity Problem, Proc. IEEE 24th Int'l Conf. Data Eng. (ICDE), 2008, 1364-1366.

Gunopulos D, and Das G, Time Series Similarity Measures, Proc. Tutorial Notes of the Sixth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (Tutorial PM-2), 2000, 243-307.

Josif Grabocka, Martin Wistuba, and Lars Schmidt- Thieme, Scalable Classification of Repetitive Time Series Through Frequencies of Local Polynomials, 2014.

Keogh E.J, and Pazzani M.J, An Enhanced Representation of Time Series which Allows Fast and Accurate Classification, Clustering and Relevance Feedback, Proc. Fourth Int'l Conf. Knowledge Discovery and Data Mining (KDD), 1998, 239-243.

Li N, Li T, and Venkatasubramanian S, t-Closeness: Privacy Beyond k-Anonymity and l-Diversity, Proc. IEEE 23rd Int'l Conf. Data Eng. (ICDE), 2007, 106-115.

Lin J, Keogh E, Wei L, and Lonardi S, Experiencing sax: A novel symbolic representation of time series, Data Min. Knowl.Discov., 15 (2), 2007, 107–144.

Lin J, Keogh EJ, Lonardi S, and Chiu BY, A Symbolic Representation of Time Series, with Implications for Streaming Algorithms, Proc. Eighth ACM SIGMOD Workshop Research Issues in Data Mining and Knowledge Discovery (DMKD), 2003, 2-11.

Machanavajjhala A, Gehrke J, Kifer D, and Venkitasubramaniam M, l-Diversity: Privacy Beyond k-Anonymity, Proc. 22nd Int'l Conf. Data Eng. (ICDE), 2006, 24.

Nergiz ME, Atzori M, and Saygin Y, Perturbation-Driven Anonymization of Trajectories, Technical Report 2007-TR-017, ISTI-CNR, 2007.

Pensa R.G, Monreale A, Pinelli F, and Pedreschi D, Pattern-Preserving k-Anonymization of Sequences and its Application to Mobility Data Mining, Proc. Int'l Workshop Privacy in Location-Based Applications (PiLBA), 2008.

Simon Malinowski, Thomas Guyet, Rene Quiniou and RomainTavenard, 1d-SAX:a Novel Symbolic Representation for Time Series, 2013.

Sweeney L, k-Anonymity: Privacy Protection Using Generalization and Suppression, Int'l J. Uncertainty Fuzziness and Knowledge-Based Systems, 10 (5), 2002, 571-588.