# Compression and Privacy Preservation of Data Streams using Moments

Anushree Goutam Ringne, Deeksha Sood, and Durga Toshniwal

***Abstract*— To prevent the misuse of sensitive data, it is essential that the privacy of the data is adequately maintained without compromising on its usability. Privacy preservation thus has become an essential prerequisite to the process of data mining. Various methods including association rule mining, k-anonymizing and data hiding have been suggested for the same. In this paper, a novel technique is suggested that makes use of the concept of moments to preserve the privacy of data streams along with compression of data. The technique uses overlapping fixed size sliding windows to calculate the sequence of moments which would constitute the compressed and privacy preserved data stream. Group of points in each window is mapped to a single point in the two dimensional plane which is the centroid/moment of the graph represented by the group. This approach is promising as all the details of the data are maintained in the moments representing them. Applying this technique on a data stream reduces its size tremendously thereby making the analysis of the resultant data stream faster. Also, this method inherently achieves privacy preservation of the stream as the actual values cannot be retrieved from the moments. This technique has been tested on real world data sets as well as on synthetic data sets.**

***Index Terms*—Privacy preservation, data compression, data mining, data streams, moments, centroids**

## I. Introduction

As the amount of data stored online has reached a critical mass with more companies, governments and individuals choosing to store their information on the digital platform, the need for data compression has become a prerequisite to efficient analysis of data. It not only reduces the cost of storing the data but also speeds up the post processing. Many data streams comprise of huge amount of sensitive data which are collected continuously from various sources. Examples of data streams include computer network traffic, phone conversations, web searches and sensor data. These data sets need to be analyzed for identifying trends and patterns which help us in isolating anomalies and predicting future behavior. However, data owners or publishers may not be willing to exactly reveal the true values of their data due to various reasons, most notably privacy considerations. Hence, some amount of privacy preservation needs to be done on the data before it can be made publicly available [1],[2]. The interpretation of data is important and it is conjoined with the need to maintain privacy using suitable algorithms. Various methods have been proposed for this purpose like data perturbation [9],[10] and [18], encryption and masking, k-anonymity [3], association rule mining [17] etc.

In this paper, we suggest a novel technique for preserving the privacy of data streams along with providing an additional incentive of data compression. We define a fixed size window where size denotes the number of data points in that window. This fixed size window keeps sliding with the arrival of new data points in the stream. Each such window is represented by the moment (centroid) of the curve generated by its data points. As more data comes in via the data stream, it can be represented with a new centroid corresponding to it. The set of centroids can then be used for subsequent analysis of the data stream. The size of the window depends on the extent to which privacy preservation is to be done. For achieving higher levels of privacy, the window size should be large and vice-versa. This technique removes the exact values of data sets but still retains the comparative characteristics of data. For example, similar curves will have centroids close to each other while dissimilar curves will have centroids far apart. Hence, the variations in the curves shall be reflected in the corresponding changes in the position of their centroids. Also, as all the data points in each window are replaced by a single point, this method inherently achieves data compression.

Let us take an example to understand the functioning of this method. Consider a group of sensors which record the nuclear emission levels in a particular nuclear plant. These sensors collect data every minute. Hence, for each day, we have around 1440 values per sensor. This is a lot of data considering the fact that there will be many sensors and data has to be collected and stored for a long duration. Hence, compression of such a data is essential to keep the processing overhead low. Also, the plant employees would not prefer to release this data to the public fearing its misuse. However, for research purposes, to optimize the plant environment, i.e. working temperature, optimal coolant characteristics etc., one would like to study the variations of radiation recorded by the sensor with these parameters. For such analysis, we do not need the exact values of the sensor data, but a comparative graph between the datasets would suffice. We suggest that the data be represented as a series of points which characterize the trends and patterns of radiations recorded by the sensors but do not reveal the actual values. This can be done by mapping the sensor data within a particular time frame, say a day or a week, to a single point by using the concept of moments. Thus, this transformed data can be provided to the research personnel without compromising on privacy or processing power.

Manuscript received September 21, 2011, revised September 23, 2011.

Anushree Goutam Ringne and Deeksha Sood are students of Indian Institute of Technology Roorkee, Roorkee, India. (e-mail: anu03uec@ iitr.ernet.in; deeksuec@ iitr.ernet.in).

Durga Toshniwal is an Assistant Professor at Indian Institute of Technology Roorkee, Roorkee, India (e-mail: durgafec@iitr.ernet.in).

## II. Related Work

With the advances in technology the amount of data is constantly growing. A lot of data, such as, sensor data and network data are in the form of streams. These stream data increase exponentially making the analysis of data streams tedious. An attempt to achieve data compression in time series data is outlined in [13].

Along with it, new techniques for breaching privacy of this data are increasing, which makes privacy preservation of this data even more important. An efficient algorithm for processing the data streams would be one that can efficiently reduce the size of data and preserve the privacy of the data stream at the same time. Also, it should also be fast enough so that it can process the frequently arriving data points in the stream.

Various techniques have been proposed for privacy preservation of sensitive data, viz. data perturbation, k-anonymization, and secular multiparty computation. Data perturbation refers to modification of sensitive data so that it retains its statistical properties but loses its sensitive meaning. Normally one is interested in mining the aggregated, statistically significant properties, while the owners want to preserve the privacy of their data. In such cases privacy can be preserved by releasing the transformed versions of the data. The randomization approach is particularly well suited to privacy-preserving data mining of streams, since the noise added to a given record is independent of the rest of the data. In case of additive perturbation, randomized noise is added to the data records. The overall data distributions can be recovered from the randomized records. Whereas in multiplicative perturbation, the random projection or random rotation techniques are used in order to perturb the records [6].

An issue that has risen lately in database research is the publication of micro-data (e.g., hospital records, criminal records) that contain one or more sensitive attributes. Organizations want to release such data (e.g., for research) without compromising the privacy. Simply hiding the explicit identity of persons (i.e., name, ID) before publication does not suffice. In particular, a set of non-sensitive attributes of a person (e.g., gender, age) may act as a quasi-identifier to reveal the association of him/her with a published record. To address this problem, methods like k-anonymity have been used. The idea in k-anonymity is to reduce the granularity of representation of the data in such a way that a given record cannot be distinguished from at least $(k-1)$ other records.

Secure multi-party computation is a computation performed by multiple parties where each party has in its possession a part of the input needed to perform the computation. The concept of secular multiparty computation was introduced in [15]. The basic idea behind it is that a computation is secure if at the end of the computation each party knows nothing except its own input and results. Details can be found in [16].

A different approach towards privacy enhancement is association rule mining where the goal is to find specific patterns that represent knowledge in generalized form without referring to particular data items.

Another technique for privacy preservation in time series data is based on the concept of moments[7]. In this technique, a set of points is mapped to a point in the two dimensional plane. This point, which is the moment of the curve represented by the data points, maintains the comparative characteristics of the data points. But it is not possible to retrieve the actual values of data from the moment, hence preserving the privacy of the time series data. Let us take a look at the method for calculating the moments $(M_x,M_y)$ for a fixed size data set [12][5][4]. The moment of an area A about the y-axis is expressed by

$$\int_A x\, dA \tag{1}$$

in which the integration is carried out over the entire area $A$ beneath the curve. The centroid axis perpendicular to the y-axis is obtained as:

$$\frac{1}{A}\int_A x\, dA \tag{2}$$

Similarly, the moment of area $A$ about the x-axis is given by

$$\int_A y\, dA \tag{3}$$

and the corresponding centroid axis perpendicular to the x-axis is given by:

$$\frac{1}{A}\int_A y\, dA \tag{4}$$

The point of intersection of the centroid axes is called the centroid/moment of the area.

Given an area, $A$, of any shape. Divide the area into infinite number of very small, horizontal area strips $dA_x$ and vertical area strips $dA_y$. Let x and y be the distances (coordinates) to each elemental area strip measured from the x and y axis respectively. Now, the moment of area in the x and y directions are respectively given by:

$$\mathrm{M_x} = \frac{\int_A x dA_y}{\int_A dA_y} \tag{5}$$

$$\mathrm{M_y} = \frac{\int_A y dA_x}{\int_A dA_x} \tag{6}$$

Here, the moment $(M_x,M_y)$ represents the privacy preserved value for the curve. In this paper, we use the above concept on data streams to provide compression as well as privacy preservation.

One notable work on privacy preservation of data streams is in [14]. Many of the conventional privacy preserving techniques [11] are not particularly suitable for privacy preservation of data streams due to the continuous arrival of large amount of data in streams. Hence compression of data streams is required so that the processing time and complexity reduces. But, none of the compression algorithms provide the additional feature of privacy preservation. In our approach, we maintain the privacy of data by representing a subset of the actual data by the centroid of the graph generated by it. It can be classified under randomization and compression algorithms, where the random value generated has all the features of the actual data and yet, the original data cannot be regenerated. Privacy preservation is inherently achieved by this compression mechanism as all the data points lying in the sliding window are replaced by a single

point i.e. the moment. The sequence of moments thus generated is used in further analysis and processing of the data.

## III. Proposed Work

In this paper, we suggest a simple technique which provides solutions to two problems in data streams at a time-data privacy and data compression. This technique is based on the method of data compression as suggested in [19] and privacy preservation using the concept of moments suggested in [7]. A fixed size window is defined, say containing $N$ data points. For each such window, the moment of the curve generated by the data points of that window is calculated. These $N$ data points are then replaced by this single point in the two-dimensional plane. Hence, a compression of the order of $N$ is done using this technique. Also, it is not possible to retrieve the original data points from the moment, hence providing privacy. The $y$-coordinate of the point represents the privacy preserved value of the parameter and the $x$-coordinate represents the approximate time at which the value is observed. Another point to be noted is that the moment represents the combined characteristics of the data points in a window, hence maintaining the usability of data.

### A. Data Compression

#### 1) Sliding Window Generation

A fixed size window of size, say $N$ is defined. The value of $n$ depends on the amount of privacy preservation and compression required. Higher the degree of privacy required, higher is the value of $N$. For the $N$ points in a window, a moment is calculated. The $N$ points are then replaced by this new value which has all the comparative attributes of the data set. The fixed size window keeps sliding after say, $K$ data points, i.e. as soon as new $(N\text{-}K)$ data points are available, the window slides by $K$ points, and a new moment is calculated. The value of $K$ is again dependent on the type of data stream. In this manner, a sensitive data stream is transformed into a privacy preserved compressed data stream.

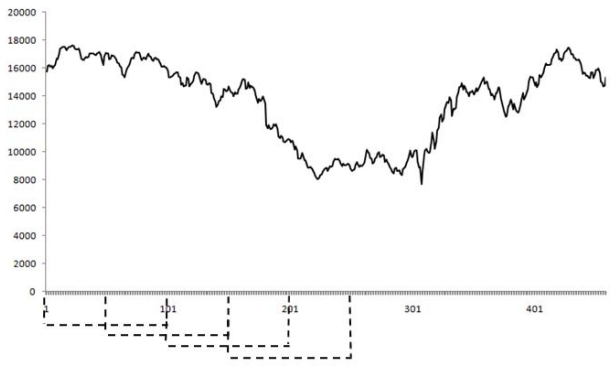


Fig. 1. Sliding window

#### 2) Window summarization

After the incoming data stream has been divided into fixed size windows, the moment has to be calculated for each window. We term this as window summarization. The comparative characteristics of all the data points in a window are represented by a single point, which is the moment of the curve represented by them. As our dataset consists of discrete values at different points of time, we divide the area under the curve represented by them into $M$ number of very small, horizontal area strips $dA_{xi}$ and $N$ vertical area strips $dA_{yi}$ where $N$ denotes the size of window. Let $x_i$ and $y_i$ be the distances (coordinates) to each elemental area strip measured from the x-y axis. Now, the moment of area in the x and y directions are respectively given by:

$$M_x = \frac{\sum_{i=1}^{N} x_i dA_{yi}}{\sum_{i=1}^{N} dA_{yi}} \quad (7)$$

$$M_y = \frac{\sum_{i=1}^{M} y_i dA_{xi}}{\sum_{i=1}^{M} dA_{xi}} \quad (8)$$

Depending upon the accuracy required, we modify the value of $M$, the number of horizontal divisions. Higher the value of $M$, higher is the accuracy of $M_y$ obtained.

### B. Privacy Preservation

This technique of compression inherently preserves the privacy of the data stream. $M_y$ represents the privacy protected value of the parameter under consideration within a particular window. We say that privacy is preserved as through this point $(M_x,M_y)$, it is not possible to trace back the various points which were used to calculate it in the first place.

Also, another feature to improve the privacy of the stream which is suggested is that the final output data stream be normalized. This means that all the y attributes of the output stream must be mapped between 0 and say, 10. This will ensure that even the slightest possibility of retrieving the actual values of the data stream is lost. Also, common patterns between two different streams will be easily visible, hence increasing the usability while improving the privacy at the same time.

Each window can thus, be represented by a single privacy preserved point with coordinates $(M_x,M_y)$ where $M_y$ is a normalized value. The window keeps sliding as new data points arrive. In a similar manner, we get a point corresponding to each window. These set of points can now be used for comparative analysis. Note that in this paper, we have considered overlapping fixed size sliding windows.

In addition, we can also implement this technique on non-overlapping windows. Let us say that the window size is W and the interval after which a new window is created is T. The value of T can be changed with the requirements of compression. Higher the value of T, higher is the compression. Also, if T is very small, we get a new modified dataset which can maintain the privacy as well as accuracy of the data but data compression is very less. In case T=1, we get a privacy preserved data stream but with no compression. Till now, we have discussed the case where T < W. If T = W, we will have a situation where windows will be non-overlapping. Hence this technique can be used to produce different types of datasets, all possessing the ability to differentiate the data on the basis of their features, yet maintaining the privacy of the data.

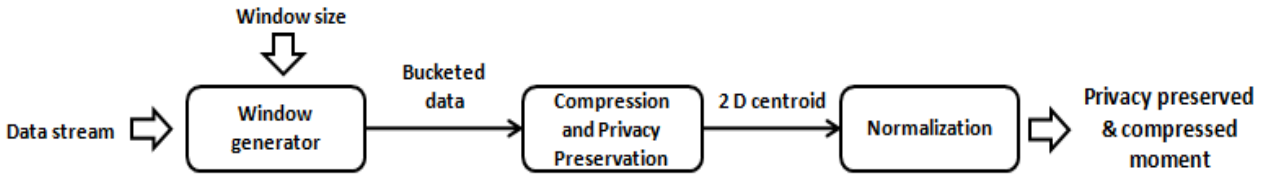


Fig. 2. Flowchart representation of the algorithm steps

## IV. Experiments

To evaluate the effectiveness of this method, we would be applying this method to various graphical patterns. The main aim of these experiments is to prove that the modified data stream can still be used for comparative analysis even though it is compressed and privacy preserved. The experiments which would be following show that dissimilar patterns lead to a difference in the position of their centroid. Also, the distance between the centroids is dependent on the difference in pattern of the graphs. Hence, the graphical patterns can easily be noted from the position of centroids. Also, by comparing the position of various centroids, we can get to know about the trend that a particular data stream follows. Hence, this technique is highly beneficial for preserving the privacy of data where only comparative analysis is required.

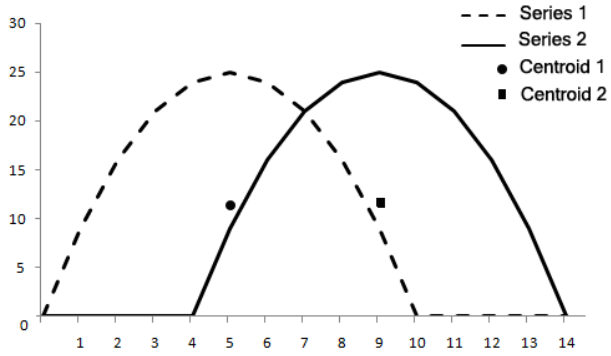


Fig. 3. Graph representing the centroids of two dissimilar curves

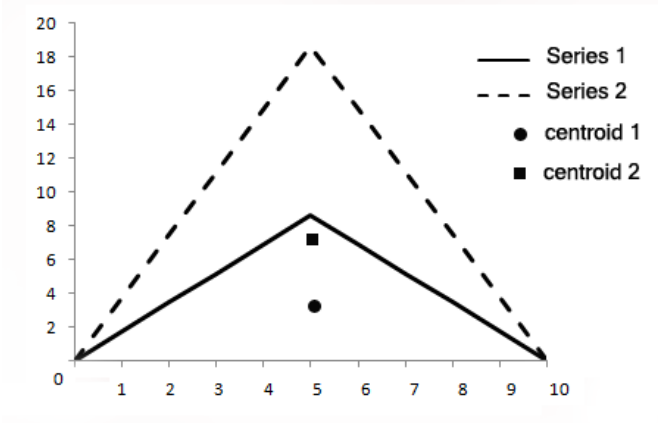


Fig. 4. Graph representing the centroids of two similar curves with different magnitudes

Let us consider two graphs: a downward parabola that has a peak at $x = 5$ and another downward parabola with a peak at $x = 9$. Both the parabolas are exactly similar, except that one of them is shifted to the right by *4* units. Here the x axis corresponds to time and the y axis corresponds to the value being measured.

Consider the Figure 3. We take the time slot for calculation of the centroid as $t = 0$ to $t = 14$. Applying the algorithm, the centroid comes out to be *(5, 10.15)*. Consider the second graph which has a peak at $x = 9$. On calculating the centroid of this graph for the same time slot, we get the value as *(9, 10.15)*.

It can be seen that as the second graph is actually the right-shifted version of the first graph, the centroid of the second graph is also shifted to the right of the centroid of the first graph by an equal distance. This can be correlated to two peaks which occur at different instances and hence, both peaks would have different $M_x$ but similar or close $M_y$. The example shows that similar patterns appearing at different points of time are denoted by a difference in the position of their centroids. Hence, the comparative study of data is made possible.

Let us consider another example. The figure 4 shows two isosceles triangles, each with a different slope. In a time varying graph, they would be representing two peaks, each of different tip value. On calculating their centroids, we get the values as *(5, 2.88)* and *(5, 6.25)* respectively. Thus, a higher peak in the graph is denoted by a higher value of the y component of the centroid. Hence, apart from distinguishing between different types of graphs, this technique can also differentiate between the magnitudes of similar graphs.

## V. Case Study

To validate the effectiveness of our compression and privacy preservation technique, we apply the algorithm to a real life data set. The data chosen for testing is a gold price data in yen per ounce from January 1979 to January 2011[8]. This data consists of the price of gold on a daily basis, excluding the value on weekends. It comprises of approximately 8500 data points. This data is actually a time series data but for testing purposes, we have assumed it to represent a data stream. The reason for choosing this data is to prove the efficiency of this technique on read world data sets. Figure 5 represents the data in a graphical format.

In order to maintain the privacy of this data, we introduce the concept of moments. We use a fixed size sliding window that contains the gold prices for one year. For each year, we have approximately 260 data points, as the gold price is not recorded on weekends. Hence, each window, which has 260 points, is now represented by the moment of the curve represented by these points. The window slides by half a year and the algorithm is run on the new set of data points. The normalization constant is chosen as 70,000. This means that the new data points will lie between 0 and 70,000. Thus, the original graph is now modified into a sequence of moments. Note that the size of the data has reduced tremendously making it convenient to process the incoming data stream.

These values represent the data in a compact yet useful form. The variations in these points represent the variations in the actual data. Hence, the new modified data can be easily used for comparative analysis without compromising on privacy.

To prove the efficiency of this technique, let us analyze the data in detail. It can be seen that the gold price peaks in 1980, followed by a decline till 1990, then stagnates for a few years and in the end shows an increasing trend after 2004 (Figure 5).Figure 6 shows the graph obtained after applying the algorithm with window size of one year which slides by half a year. The same trend is evident in this graph as well. The two outlier points around 1980 depict the peak, followed by numerous points that show the decline till 1990. Also, the increasing trend from 2004 is clearly visible from the moments depicted in the graph.

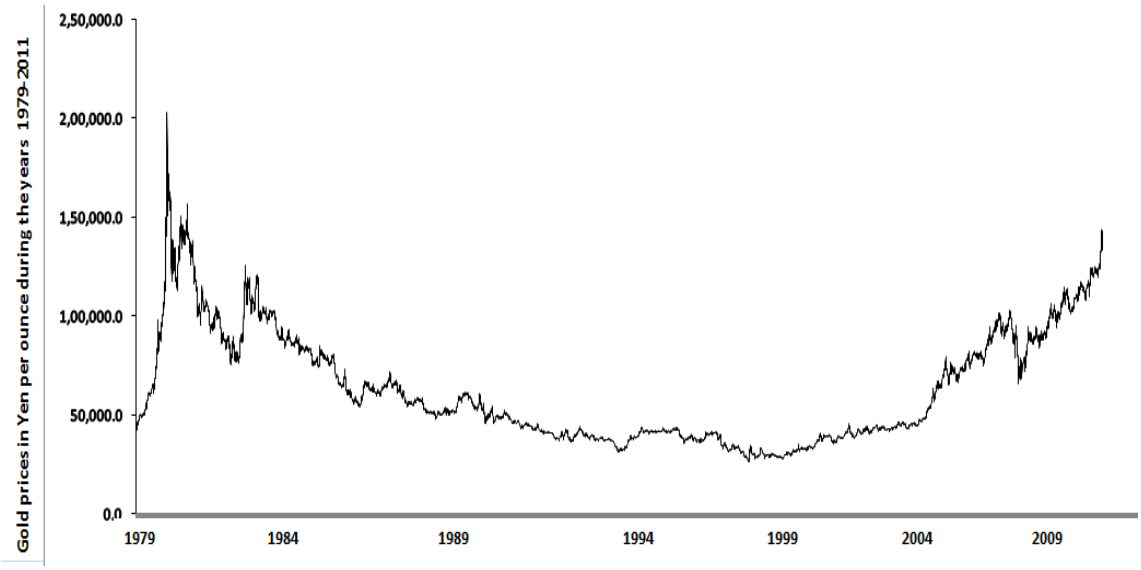


Fig. 5. Graph representing the gold price in Yen per ounce from 1979-2011

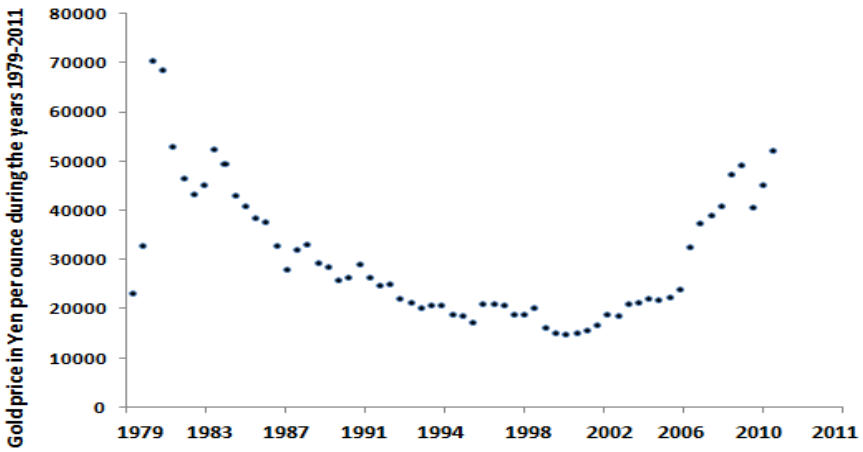


Fig. 6. Points obtained by applying the algorithm for window size equivalent to one year

Note that the values of the calculated moments are very different from the input data. Hence, it is not possible to trace back the original data values from the moments, thereby preserving the privacy of data. The difference between the y components of two consecutive centroids tells us about the rate of increase or decrease in the value of gold. Hence, it can be seen that the comparative characteristics of the data are maintained even after the implementation of our technique.

The degree of privacy preservation and compression achieved depends on the window size. Changing the window size affects the graph obtained by this technique. If we keep the window very small, we get a graph which is very similar to the original graph, hence privacy is compromised in order to achieve a higher degree of accuracy. Also, very minimal data compression is provided by taking a small window. A window size of one would generate the same graph without anonymizing any of the data. On the other hand, a large window improves data compression as well as the anonymity of data but the comparative character.

Figure 7 shows the set of points obtained by running the algorithm on the same data but with a larger window of size two years. The window in this case slides by one year. As it can be seen, the number of data points now obtained is half of the data points in the previous case. The graph is still able to depict the major trends but the minor peaks and troughs have been smoothened out. For example, there is a small dip in 1989 which is depicted in Figure 6 but is smoothened out in Figure 7. Hence, we improve the privacy and compression by sacrificing on the accuracy.

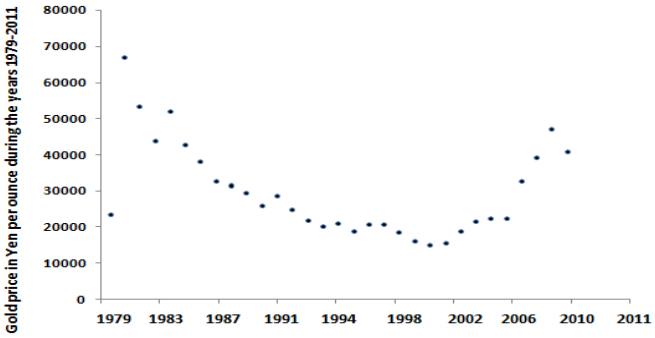


Fig. 7. Points obtained by applying the algorithm for window size equivalent to two years

## VI. Conclusion

In this paper, a technique has been proposed which provides data compression as well as privacy preservation in data streams. In this approach, a fixed size window is replaced by a point in the two-dimensional plane, thereby reducing the size of data stream while preserving the privacy of the data at the same time. Reducing the size of the data leads to a reduction in processing power required to analyze the data. Also, ensuring the privacy of data allows the owner to release the transformed stream to the public for further usage. As the sequence of moments also retains the comparative characteristics of the original data, not much usability is compromised. This technique is promising as not many algorithms exist for privacy preservation of data streams due to the high volume of data involved.

Till now, we have implemented this technique for uni-variate data stream. The data set used in the ase study had just one attribute, i.e. the gold price. This technique can also be extended to multi-variate data streams. By multi-variate streams, we mean multiple attributes varying with time.

Also, this technique can be extended to Boolean data as well. In addition to that, the values obtained by this technique can be further encrypted by adding noise

## References


[1] R. Agrawal, R. Srikant, "Privacy-Preserving Data Mining",in Proc. 2000 ACM SIGMOD Conference on Management of Data, Dallas, Texas. May 2000 ,pp. 439-450.

[2] D. Agrawal, C. C. Aggarwal, "On the Design and Quantification of Privacy Preserving Data Mining Algorithms", in *Proc. 20th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*. Santa Barbara, California, USA. May 2001. pp.247-255

[3] P. Samarati, L. Sweeney, "Protecting Privacy when Disclosing Information: k-Anonymity and its Enforcement Through Generalization and Suppression", *IEEE Symp. on Security and Privacy,* 1998.

[4] D. Toshniwal , R. C. Joshi, "Finding similarity in time series data by method of time weighted moments",in *Proc. of the 16th Australasian database conference,* 2005, Newcastle, Australia, p.155-164.

[5] D. Toshniwal, R. C. Joshi, "Using Cumulative Weighted Slopes for Clustering Time Series Data", *International Transactions on Computer Science and Engineering*, Oct 2005.

[6] K. Liu, H. Kargupta, J. Ryan, "Random Projection Based Multiplicative Data Perturbation for Privacy Preserving Distributed Data Mining", *IEEE Transactions on Knowledge and Data Engineering,* 18(1), 2006.

[7] D. Sood, A. G. Ringne, D. Toshniwal, " Privacy preservation of time series data using moments, in *Proc. fourth IEEE conference on Computer Science and Technology,* Chengdu, China, 2011, pp. 144-148.

[8] Data: http://www.research.gold.org/prices/daily/

[9] C. W. Wu, "Privacy preserving data mining: a signal processing perspective and a simple data perturbation protocol," in IEEE Workshop on Privacy Preserving Data Mining, International Conference on Data Mining, 2003. Available electronically at http://www.cis.syr.edu/-wedu/ppdm2003/papers/2.pdf

[10] H. Polat, W. Du, "Privacy-Preservation collaborative filtering using Randomized Perturbation Techniques", in *Proc. of the Third IEEE International Conference on Data Mining* , 2003, pp. 625.

[11] C. Aggarwal, P. S. Yu, *Privacy-Preserving Data Mining models and algorithms,* Springer , 2008

[12] http://en.wikipedia.org/wiki/First_moment_of_area

[13] S. Papadimitriou, F. Li, G. Kollios, P. S. Yu, "Time Series Compressibility and Privacy", in *Proc. VLDB '07 Proceedings of the 33rd international conference on Very large data bases,* Vienna, Austria, pp. 459-470.

[14] F. Li, J. Sun, S. Papadimitriou, G. Mihala and I. Stanoi, "Hiding in the Crowd: Privacy Preservation on Evolving Streams through Correlation Tracking", in *Proc. 23rd IEEE International Conference on Data Engineering*, Los Alamitos, 2007, pp. 686-695.

[15] A. C. Yao, "How to generate and exchange secrets", in *Proc. 27th IEEE symposium on Foundations of Computer Science,* Los Alamitos, CA, 1986, pp. 162-167.

[16] C. Clifton, M. Kantarcioglu, J Vaidya, X. Lin, M. Y. Zhu, "Tools for privacy preserving distributed data mining", *SIGKDD Explorations Newsletter* 4(2), 2002, pp. 28-34

[17] J. Vaidya, C. Clifton, "Privacy preserving association rule mining in vertically partitioned data",in *Proc eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, Edmonton, Alberta, Canada. July 2002, pp. 639-644.

[18] W. Ouyang, H. Xin, Q. Huang, " Privacy preserving sequential pattern mining based on data perturbation",in *Proc. Sixth International Conference on Machine Learning and Cybernetics,* Hong Kong, August 2007, pp. 19-22.

[19] D. Toshniwal, R. C. Joshi, "Similarity search in time series databases using moments", in *Proc. 2004 International Conference on Machine Learning and Applications, ICMLA '04,* USA, 2004, pp. 164-171.



**Anushree Goutam Ringne** is presently an Undergraduate student of Computer Science and Engineering at the Indian Institute of Technology Roorkee, India.

Some of her areas of interest include parallel processing, data mining and network security. She has published a research paper in the conference ICCSIT 2011 (4th IEEE conference on Computer Science and Information Technology 2011).



**Deeksha Sood** is presently an Undergraduate student of Computer Science and Engineering at the Indian Institute of Technology Roorkee, India

Her key areas of interest include text mining, data mining and parallel processing. She has published a research paper in the conference ICCSIT 2011 (4th IEEE conference on Computer Science and Information Technology 2011).



**Dr. Durga Toshniwal** is presently working as an Assistant Professor at the Department of Electronics and Computer Engineering, Indian Institute of Technology Roorkee, India. She completed her Bachelor in Engineering and subsequently earned her Master of Technology from National Institute of Technology, Kurukshetra, India and Doctor of Philosophy from Indian Institute of Technology Roorkee, India.

Previously she worked as a Software Consultant in USA for some years and then pursued her research in data mining. Some of her areas of research interests include – time series data mining, web mining, privacy preserving data mining, data stream mining, applying soft computing techniques in data mining and text mining. Dr. Durga has published her research work in various international journals and conferences. She has attended, chaired sessions and presented her work in several reputed international conferences in USA, Australia, and Europe.is presently an Undergraduate student of Computer Science and Engineering at the Indian Institute of Technology Roorkee, India.