ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 2, pp: (1084-1092), Month: April - June 2015, Available at: www.researchpublish.com

FINDU: A Methodology for Finding and Fine Tuning the Data Utility for Privacy Preserving Data Mining

[1]N.Sridhar, [2]Dr Y. Srinivas

[1]Researcher, [2]Professor, Dept of CSE, GITAM University Vishakhapatnam, INDIA

***Abstract*: The field of Preserving Privacy in Data Mining is gaining momentum in the recent times as the data sets are more open towards mining by organizations and academic institutes. Ensuring privacy in data before publishing it to wider audience is always an open challenge. There have been many techniques evolved to exploit the perturbed data and get some sensitive knowledge. In the process of ensuring more privacy, the data perturbation techniques also became complex and more distortive in nature. The Data Utility is level of usefulness of the distorted data. The study of data utility comes into play as the distortion level increases. In this paper we are going to propose a pre and post perturbation analysis for measuring the data utility and using this as an input to choose the balance the Data Utility and Privacy in the datasets. This paper primarily focuses the privacy and data utility for datasets which are relational in nature.**

***Keywords*: Privacy Preserving, Data Mining, Tuning Data Utility.**

1. INTRODUCTION

Data Mining takes key position in today's data world where it has been extensively used in many institutions. The current strategy of mining activities needs to exchange data for mutual benefit. This leads to concern over privacy issues in the recent times. It has also been pointed out that a possible threat of leaking sensitive data of an individual when the data is published to outside word. Several anonymization methods have been came in to picture to deal with privacy in networks. Some of the methods came to preserve the dataset are anonymization and perturbation. But the natural side effect of privacy preservation is data quality loss. The loss of specific details about certain individuals may affect the data utility and in some cases the data may become completely meaningless. The cryptographic methods also came in to existence which completely anonymize the dataset and which makes the dataset difficult to use. So the quality of the resultant data is completely lost. This drives the need to protect the private data and making the data utility as much as possible. The objective of this paper is to find an optimum balance between privacy of the dataset and utility while publishing the dataset of any institutions.

2. EXISTING PRIVACY METHODS FOR DATASETS

There are numerous methods for privacy came in to existence in today's world of privacy preserving data mining. Some of the methods are discussed here

***K-anonymity*:** Suppose a data publisher like a bank or hospital that wants to publish the data for research outside with some privacy. One simple approach they can take is to remove all the identifier attributes and simply publish this data. Removing just the identifier attributes will not suffice as it is clearly possible to identify the data by having some more non identifier attributes (Quasi Identifiers) about the individual victim. So the solution is to publish the data by masking or generalizing the quasi identifiers to have same values, so that the chance of identifying the individual record will be less. Making sure there are k-1 such records for the quasi-identifier values is known as k-anonymization. This provides

privacy for the individual record in the particular release; so that there are k-1 such records for the same attribute values. [1] [2]. There are some extensions to k-anonymity are proposed. ℓ-diversity [2] and t-closeness [3] are major such improvements to anonymization.

In the recent years there have been other approaches based on matrix factorization; here this paper presents some of them which have been used in privacy preserving data mining. Generally the matrix factorization-based techniques perform the attribute extraction by matrix factorization to analyze data, identify the key pieces of information for data mining and eliminate the unimportant information to modify data for privacy.

Singular Value Decomposition: Singular value decomposition (SVD) is the process of factorizing the matrix into smaller dimension so that mining algorithm can be applied on smaller dimensional data efficiently in terms of space and time. The factored matrix preserves almost all of the original dataset attributes with lesser dimension. SVD can be pictured as a method for translating correlated variables into a set of uncorrelated variables that better visualizes the various relationships among the original attributes. It is also used as a method for finding and arranging the dimensions along which attributes represent the most variation. Once we have found where the most variation is present, then it's possible to identify the approximation that fits best for the original data points using fewer dimensions. Hence, SVD can be used as a process for dimension reduction for larger datasets. The SVD-based method provides perturbed data instead of original data to the researcher, and the researcher finds original data patterns from perturbed data. [2]

Non-negative Matrix Factorization: The Non-negative Matrix factorization is the process of factoring the matrix in to product of two matrices of smaller dimensions. Given a data matrix A with order m*n, NMF breaks the matrix into appropriate matrices X and Y such that A ≈*XY*, where A, X, and Y are non negative matrices. The orders of the matrices X is m*r, Y is r*n respectively. The value of r is usually smaller than that of m and n. The product XY can be considered as compressed or transformed data of the original matrix A. The key idea here is to select best possible selection of non-negative matrices X and Y that reduces the reconstruction error among *A* and *XY*. Different error estimation functions are currently in practice. The most common used are the square Euclidean distance function and K-L Divergence function. [5] So clearly the NMF can be used as an alternative for perturbing the data for preserving the datasets.

Wavelet Based Data Distortion*:* Wavelets are applied in image processing and compression areas effectively. A wavelet is a series function which represents the time-frequency variation of the original data. Wavelets are based on thresholding concept where it removes the excessively present small features. In recent times wavelets are also used for data distortion or data reduction in privacy preserving data mining. [6]

3. EXISTING METHODS FOR MEASURING DATA UTILITY

There has been lot of attention on data modification techniques for privacy preserving has been well adopted, but one key aspect about the data perturbation is the data utility. Once the data is perturbed the data is published for mining. The data utility is the usefulness of the mining results that are about to come. Generally the data publisher does not know how the published data will be mined by the researcher. There has been very less attention for quantifying the data utility aspect for the published data.

Hiding Failure: as the percentage of restrictive patterns that are discovered from the sanitized database. It is measured as follows:

$$HF = \frac{\#R_P(D')}{\#R_P(D)}$$

#RP(D) and #RP(D′) denote the number of restrictive patterns discovered from the original data base D and the sanitized database D′ respectively. [8]

ILoss : is one such metric to evaluate the information loss on generalizing a specific value to a generic value, proposed by Xiao and Tao. [7]

V_g: $ILoss(v_g) = (|v_g| - 1) / (|DA|)$

here $|v_g|$ is the total count of possible domain values that are children of v_g,

|DA| is the count of domain values for the attribute A of vg.

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 2, pp: (1084-1092), Month: April - June 2015, Available at: www.researchpublish.com

ILoss for a specific row in a table is denoted as

$$ILoss(r) = \sum_{v_g \in r} (w_i \times ILoss(v_g)),$$

where w_i is a non negative constant which denotes the weight of attribute A_i of v_g. The total loss of a generalized table T is given by

$$ILoss(T) = \sum_{r \in T} ILoss(r).$$

These only talk about the noise which is added before analysis. But none of the methodologies which are providing any insight for the data and mining results obtained after mining the modified data. This paper is an attempt to contribute something in this area.

Dataset Dissimilarity: the information loss can be measured in terms of the dissimilarity between the original dataset D and the sanitized one D ′.

$$Diss(D, D') = \frac{\sum_{i=1}^{n} |f_D(i) - f_{D'}(i)|}{\sum_{i=1}^{n} f_D(i)}$$

where i is a data item in the original database D and fD(i) is its frequency within the database, whereas i' is the given data item after the application of a privacy preservation and fD′(i) is its new frequency within the transformed database D′. [9]

4. PROBLEM STATEMENT

Identifying the right utility with privacy is challenging. There are so many measurements to find the data utility, but most of them suffer from two problems. First, data utility measurements with respective to mining results are missing. Second, there are no proper preview techniques for data utility. The preview means, finding how many rounds of perturbation needed to achieve good amount of data utility? When to stop further perturbation so that the data utility gets balanced? How do we validate new perturbation methods for Data Utility objectives? Here in this paper we are proposing measurements for data utility and proposing algorithms to preview the data utility.

5. SOLUTION

FINDU - Metric for Finding Data Utility in Relational Datasets: True Positive Count is the total number of correctly classified instances in the mining process. For example if the original dataset (**D**) has 80 classifications as True Positives according to the mining classifier model. Now we apply the classifier on the distorted dataset (**D'**). If the True Positives count drops to 60 then, the change % in the distorted data is around 25%. Hence the Data Utility of the distorted dataset (**D'**) is approximately 75% with respective to mining results.

True Positives Change Rate: **ΔTPCR** (**D**, **D'**) = ((**tp**(**D**) – **tp**(**D'**))/ **tp**(**D**)) *100

(if **tp**(**D**) > **tp**(**D'**))

= ((**tp**(**D'**) – **tp**(**D**))/ **tp**(**D'**)) *100

(if **tp**(**D'**) > **tp**(**D**))

Here **tp**: True Positives Count from the respective Classification/Clustering algorithm result

D : Original Dataset

D': Perturbed Dataset

Cumulative True Positives Change % across Classifiers: Is the average of the true positives change rate across the chosen classifiers for mining.

CTPCR = (1/|c|) * ∑ (**Δ**Ci**TPCR**(**D**, **D'**))

|c| = Number of Classifier Algorithms Considered

Ci = Classifier Algorithm

Δci TPCR(D, D') : True Positives Change rate for Classifier Ci

The algorithm outlined here computes the data utility using the proposed metric above. Initially all the chosen classifiers are applied and the true positive counts will be initialized from the classification results. The classification model that is built here is saved, so that the same model will be feed with perturbed dataset in subsequent iterations. The perturbation is applied on the dataset. The perturbed dataset is evaluated against the classifier, the metric is computed again. This is repeated until the used specified number of iterations reached.

Algorithm 1 : Data Utility Measurement based on Confusion Matrix.

Input:

ds_0 : Dataset Initial

mAlg : Mining algorithm

pAlg : Perturbation Algorithm

numIter : Number of iterations

selClassifiers : Selected Mining Classifiers

Output:

ds_{iter} : Dataset Perturbed after balancing the Data Utility at each stage.

ctp : Cumulative True Positives Rate

```
ctp := 0;
selClassfiers := {c1, c2, c3, ...};
for ci := 1 to |selClassfiers| do
begin
  ConfusionMatrix0 := ApplyMiningAlgorithm(ci, ds0, mAlg);
   for iter := 1 to numIter do
   begin
       dsiter := ApplyPerturbarion(dsiter-1, pAlg);
       ConfusionMatrixiter := ApplyMiningClassifier(ci , dsiter-1, mAlg);
        /* Calculate True Positives change rate */
       If (dsiter> dsiter-1 )
          Δtp ci (dsiter, dsiter-1) := ( (tp(dsiter) – tp(dsiter-1))/ tp(dsiter)) *100;
       else
           Δtp ci(dsiter, dsiter-1) := ( (tp(dsiter-1) – tp(dsiter))/ tp(dsiter-1)) *100;
        ctp := cptr + Δtp ci (dsiter, dsiter-1);
        Output: Δtp ci (dsiter, dsiter-1), dsiter;
   end;
  end;
  /* Cumulative True Positives change rate */
 ctp := ctp / |selClassfiers|;
 Output: ctp;
end;
```

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 2, pp: (1084-1092), Month: April - June 2015, Available at: www.researchpublish.com

FINDU - *Algorithm to Fine Tune Data Utility* in Privacy Preserving Relational Datasets: The algorithm outlined here takes user supplied threshold and uses it pause the distortion process once the desired data utility score is reached.

Algorithm 2: Threshold driven Data Utility

Input:

ds_0 : *Dataset Initial*
mAlg : *Mining algorithm*
pAlg : *Perturbation Algorithm*
limitDu : *Threshold to stop*
numIter : *Number of iterations*
selClassifiers : *Selected Mining Classifiers*

Output:

ds_{iter} : *Dataset Perturbed after balancing the Data Utility at each stage.*

```
ctp := 0;
selClassfiers := {c1, c2, c3, ...};
for ci := 1 to |selClassfiers|  do
begin
  ConfusionMatrix0 := ApplyMiningAlgorithm(ci, ds0, mAlg);
  for iter := 1 to numIter do
  begin
      dsiter  := ApplyPerturbarion(dsiter-1, pAlg);
      ConfusionMatrixiter :=  ApplyMiningClassifier(ci , dsiter-1, mAlg);
       /* Calculate True Positives change rate */
      If  (dsiter> dsiter-1 )
        Δtp ci (dsiter, dsiter-1) := ( (tp(dsiter) – tp(dsiter-1))/ tp(dsiter)) *100;
      else
         Δtp ci(dsiter, dsiter-1) := ( (tp(dsiter-1) – tp(dsiter))/ tp(dsiter-1)) *100;
      ctp := ctr + Δtp ci (dsiter, dsiter-1);
      if ( (ctp / ci ) > limitDu )
       begin
          Output : (ctp / ci );
              Output : dsiter;
            Exit;
        end;
  end;
end;
```

It clearly tells that if $|\Delta tp|$ is close to zero means the data is classified properly after the distortion. This is the desired result. This may only happen for relatively small detests. In reality the dataset under consideration is too large. So achieving closeness between ***ds*** and ***ds*** close to zero with real-time dataset is almost impossible as the distortion level is at large scale.

6. IMPLEMENTATION AND DATASET

This algorithm has been implemented using WEKA, especially the API. This dataset is the credit history of the account holders who applied for credit card. Only the subset of the dataset has been used for the simulations.

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 2, pp: (1084-1092), Month: April - June 2015, Available at: www.researchpublish.com

7. SIMULATION RESULTS

The cumulative loss of percentage on true positives is calculated at each round of the perturbation. The growth rate of the distortion has been plotted. It has been clearly observed that the accuracy classification results of the mining algorithm have been reduced drastically at each iteration during the process.

***Finding the Data Utility Metric*:** The user uploads the original, perturbed datasets, and selects the mining classifiers. Currently the implementation had only three classifier methods (k-means clustering, Decision tree, Bayes). Once the input has been specified the process will calculate the Cumulative Data Utility Score according to the true positives based methodology outlined in this paper. This process is applied for two different datasets and the resulting simulation screens and tabular data has been plotted. The *Fig. 1* and *Table 1* reflect this simulation for Financial Dataset.

Previewing the Data Utility by Number of Iterations: The user uploads the original datasets, chooses the perturbation mechanism to consider for privacy, the mining classifiers. Currently the implementation had only three classifier methods are provided (K-means clustering, Decision tree, Bayes classifiers). The user also supplies the number of iteration that the [perturbation needs to be performed. Once the input has been specified the process will calculate the Cumulative Data Utility Score at each iteration of the perturbation according to the first algorithm that is outlined. This process is applied for two different datasets and the resulting simulation screens and tabular data has been plotted. The *Fig. 2* and *Table 2* reflects this simulation for Financial Dataset.

Previewing the Data Utility by Threshold: The user uploads the original datasets, chooses the perturbation mechanism to consider for privacy, the mining classifiers. Currently the implementation had only three classifier methods are provided (K-means clustering, Decision tree, Bayes classifiers). The user also supplies the Threshold value. This value is the terminating point of the second algorithm outlined. Once the input has been specified the process will calculate the Cumulative Data Utility Score at each iteration of the perturbation according to the first algorithm that is outlined. The perturbation is performed until the cumulative data utility score reaches this value. This process is applied for two different datasets and the resulting simulation screens and tabular data has been plotted. The *Fig. 3* and *Table 3* reflects this simulation for Financial Dataset.

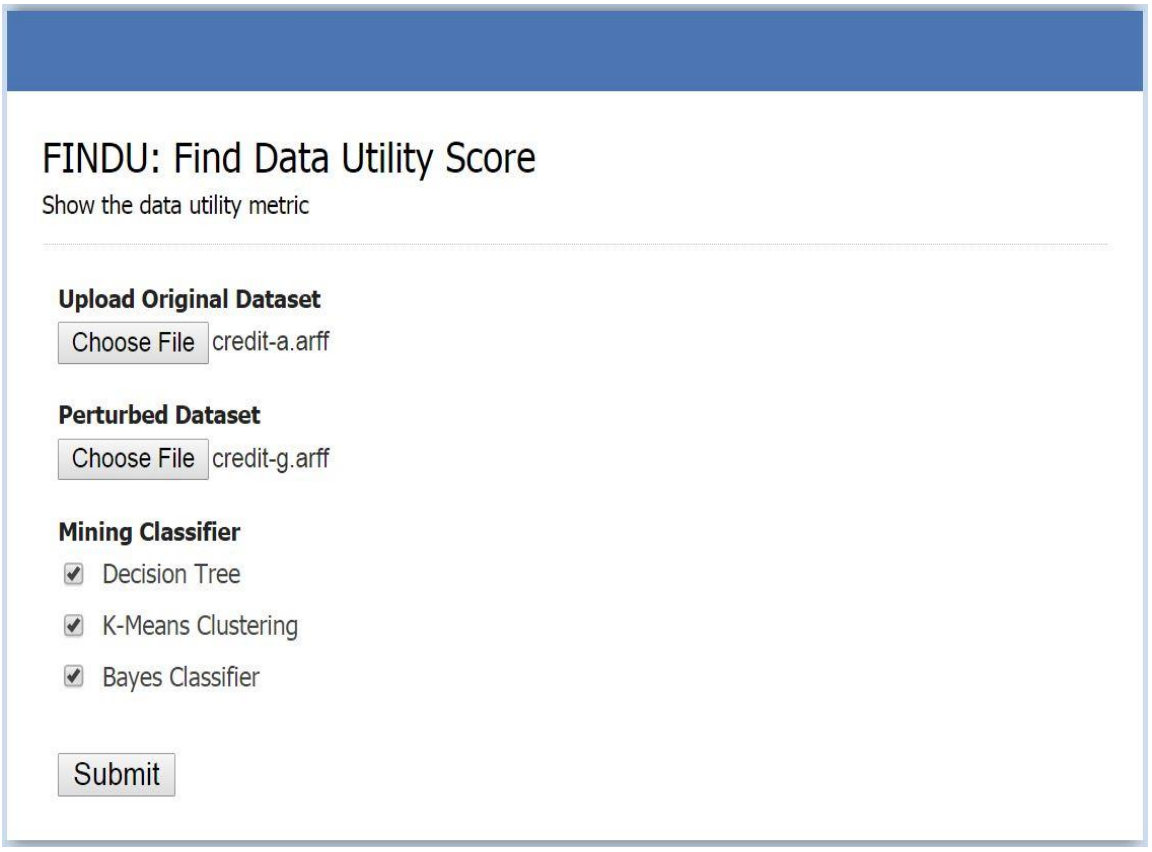

Fig. 1: Finding Data Utility of the Perturbed Financial Dataset

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 2, pp: (1084-1092), Month: April - June 2015, Available at: www.researchpublish.com

Table 1: Data Utility of the Perturbed Dataset

True Positives	K-Means Clustering	Bayes Classifier	Decision Tree
Original	2476	2126	2245
Perturbed	2234	2148	1945
Change %	90.2261712	101.035	86.637

Cumulative Data Utility is 92.63265

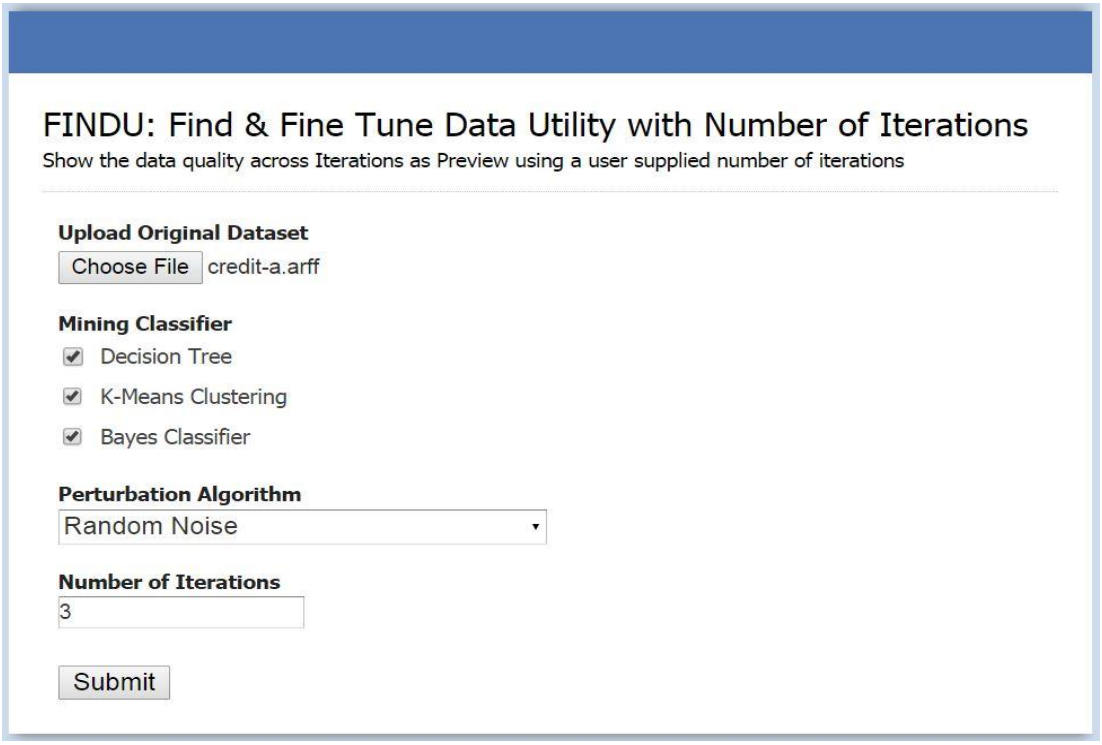

Fig. 2: Previewing Data Utility by user supplied number of Iterations

Table 2: Previewing Data Utility by user supplied number of Iteration

Iteration 1

True Positives	K-Means Clustering	Bayes Classifier	Decision Tree
Original Dataset	2476	2126	2245
Perturbed Dataset	2147	2136	2178
Change %	86.71243942	100.4703669	97.0155902

Data Utility is 94.73279884

Iteration 2

True Positives	K-Means Clustering	Bayes Classifier	Decision Tree
Original Dataset	2147	2136	2178
Perturbed Dataset	2023	1987	2054
Change %	94.2244993	93.02434457	94.3067034

Data Utility is 93.85184909

Iteration 3

True Positives	K-Means Clustering	Bayes Classifier	Decision Tree
Original Dataset	2023	1987	2054
Perturbed Dataset	1876	1729	1678
Change %	92.73356401	87.01560141	81.69425511

Data Utility is 87.14780684

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 2, pp: (1084-1092), Month: April - June 2015, Available at: www.researchpublish.com

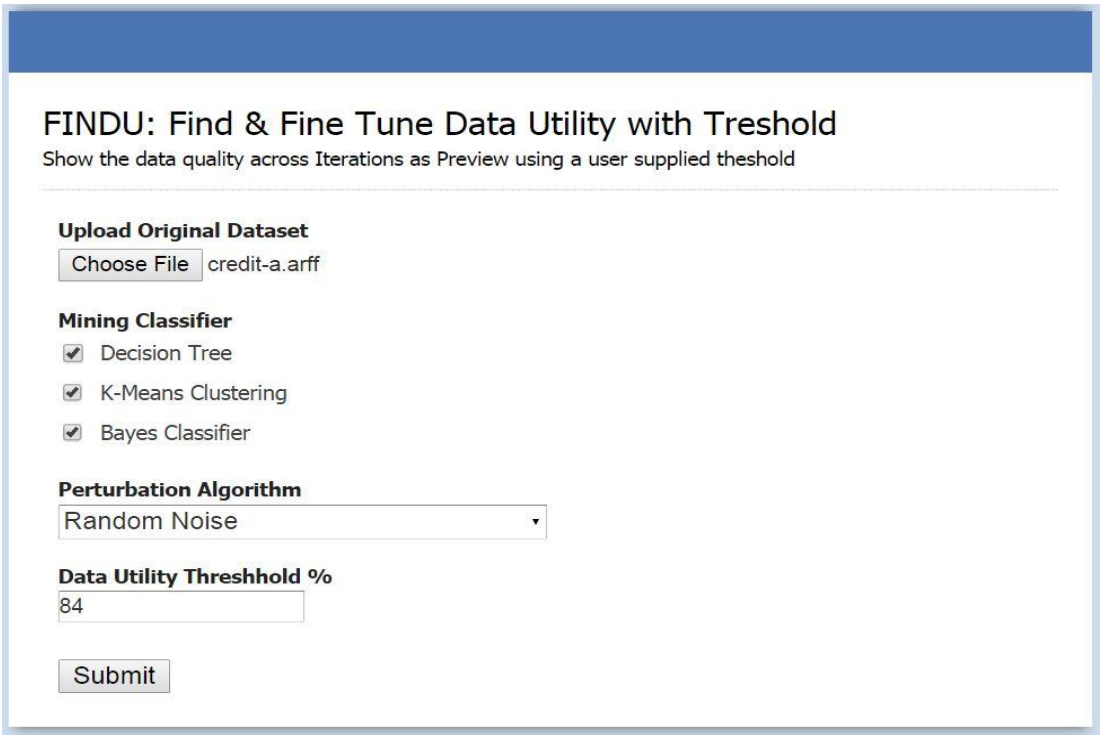

Fig. 3: Previewing Data Utility by user supplied Threshold

8. CONCLUSION

In this paper an attempt has been made to find the data utility considering the mining classifiers. The data utility has been calculated at the end of every iteration. The resulting data utility has been plotted with respective to mining algorithm. It has been shown that the level of distortion clearly impacts the Data Utility, irrespective of the distortion methodology. The pre analysis on the distorted data gives us the data utility aspect and at also allows us to control the level of distortion that need to be added or to decide on when to stop the further distortion. This new metric and preview algorithms can be used to measure the performance of any newly introduced methodology for dataset perturbation.

Table 3: Previewing Data Utility by user supplied Threshold (Financial Dataset)

Iteration 1

True Positives	K-Means Clustering	Bayes Classifier	Decision Tree
Original Dataset	2476	2126	2245
Perturbed Dataset	2146	2568	1925
Change %	86.6720517	120.7902164	85.74610245

Data Utility is 97.7361235

Iteration 2

True Positives	K-Means Clustering	Bayes Classifier	Decision Tree
Original Dataset	2146	2568	1925
Perturbed Dataset	1821	1747	1674
Change %	84.8555452	68.02959502	86.96103896

Data Utility is 79.94872639

Iteration 3

True Positives	K-Means Clustering	Bayes Classifier	Decision Tree
Original Dataset	1821	1747	1674
Perturbed Dataset	1421	1381	1219
Change %	78.03404723	79.04979966	72.81959379

Data Utility is 76.63448022

9. FUTURE SCOPE AND LIMITATIONS

The algorithms proposed works only with relational datasets in nature. This can't be applied for time-series datasets. Current implementation has been carried out only with random noise addition methods. The same implementation can be extended further to apply data perturbation methods such as k-Anonymity, NMF, SVD, PCA and Wavelet based methods.

REFERENCES

[1] Sweeney L. k-anonymity: a model for protecting privacy International Journal on Uncertainty, Fuzziness and Knowledge-based Systems 2002; 10(5):557-570.

[2] A. Machanavajjhala, J. Gehrke, and D. Kifer. l-diversity: Privacy beyond kanonymity.Proceedings of the 22nd International Conference on Data Engineering (ICDE 2006), 3-8 April 2006, Atlanta, GA, USA.

[3] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian (2007). "t-Closeness: Privacy beyond k-anonymity and l-diversity".

[4] A Privacy-Preserving Classification Method Based on Singular Value Decomposition. Guang Li; Yadong Wang. International Arab Journal of Information Technology (IAJIT); Nov2012, Vol. 9 Issue 6, p529.

[5] D. D. Lee and H. S. Seung. Algorithms for nonnegative matrix factorization. In Advances in Neural Information Processing 13(Proc. NIPS 2000). MIT Press, 2001.

[6] Wavelet-Based Data Distortion for Privacy-Preserving Collaborative Analysis Lian Liu, Jie Wang, Zhenmin Lin, and Jun Zhang Laboratory for High Performance Scientific Computing and Computer Simulation, Department of Computer Science, University of Kentucky, Lexington, KY 40506–0046, USA June 8, 2007.

[7] Xiao X. and Tao Y., Personalized Privacy Preservation, Proceedings of the 2006 ACM SIGMOD International conference on Management of data, Chicago, IL, USA, June 27-29, 2006, pp.229-240.

[8] Oliveira, S.R.M., Zaiane, O.R.: Privacy preserving frequent item set mining. In: IEEE icdm Workshop on Privacy, Security and Data Mining, vol. 14, pp. 43–54 (2002).

[9] Bertino, E., Fovino, I.N., Provenza, L.P.: A framework for evaluating privacy preserving data mining algorithms. Data Mining and Knowledge Discovery 11(2), 121–154 (2005).