Efficient Privacy Preservation of Big Data Using Random Number Generators and Geometric Data Transformations

[1]D.Kavitha, [2]Dr.T.Adilaxmi (Supervisor), [3]Dr.M.Chandra Mohan(Co-Supervisor)
[1]PH. D Scholar, JNTUH, Kukatpally,Hyderabad**. kavithadasari.it2005@gmail.com**
[2]Prof. & Head,Dept of CSE,Vasavi College of Engineering,Hyderabad.
[3]Prof. of CSE & Director of Evaluation, JNTUH,Computer Science & Engineering,Hyderabad.

Article Info
Page Number: 268 – 282
Publication Issue:
Vol. 71 No. 3 (2022)

Abstract
Recent trends indicate that the volume of data stored in repositories is exploding due to the development of technology and the pervasive usage of web-based activities. This vast collection of data may contain personal information, which may occasionally pose privacy problems. The purpose of this research work is to establish privacy-preserving data publication mechanisms that are applicable to any numerical attributes and to design a distributed data model to process big data. Four distinct classifiers, namely Decision Tree, Naive Bayes, Adaboost, and KNN, are used to evaluate the classification accuracy of the suggested model. A geometric data perturbation-based method (RSUGP) and random number generators is used to protect sensitive data. For geometric data perturbation, a noise model based on random number generators was utilized instead of random noise or Gaussian noise. A Graphical neural network (GNN) is utilized for training, testing, and classification with an accuracy of 93%. Experiments indicate that the proposed strategy is superior to the other three in terms of attack resistance, classification precision, and runtime.

Keywords: RSUGP, GNN, Decision tree, Naïve Bayes, Adaboost, Data perturbation, Random Number Generators.

Article History
Article Received: *12 January 2022*
Revised: *25 February 2022*
Accepted: *20 April 2022*
Publication: *09 June 2022*

1. Introduction

Information plays a crucial role in organizational decision-making nowadays. Every day, vast amounts of data are collected in our globe. The ability to analyse such data is critical. Data mining is a modern technology for extracting knowledge from massive amounts of data. It can potentially disclose sensitive information about individuals and organizations throughout this procedure. Moreover, data mining techniques might expose typical information about corporate operational data and individuals. Confidentiality is the primary issue that arises in any huge gathering of data. As a reaction to privacy protection in data mining, privacy-preserving data mining is a topic of research in data mining. It is sometimes referred to as privacy-enhanced data mining and privacy-sensitive data mining.

We live in a society based on the processing of information. Data is the most valuable asset for any company or organization. There is a huge amount of sensitive data generated by many companies' operational apps. It is critical to share information from several sources through channels that have been approved. To discover new information, a variety of data can be mined with the help of a technique called Data Mining. Securing data from illegal access is of utmost importance when sharing information over several channels or extracting data from numerous external sources. Databases, data warehouses, and other types of data repositories are all used in the process of "data mining." Data Mining is one of the fastest-growing academic topics in the world of data privacy and security. Centralized and distributed data mining environments have privacy challenges, which are addressed by a variety of privacy-preserving distributed data mining (PPDDM) and privacy-preserving data mining (PPDM) solutions. Data mining is important in real-world commercial applications since it provides various methodologies and algorithms. When data is extracted from different data sources for business decisions or for business processing, it is mandatory to secure the data of individuals or groups and provide privacy of data.

RSUGP is a technique for perturbing inputs in an irreversible manner using a noise model. Introducing noise with random number generators is a technique for anonymization.

In order to speed up the process of training a model in a distributed environment with enormous volumes of data, this project aims to establish privacy-preserving data publication mechanisms that are applicable to any numerical attributes and to design a distributed data model to process big data. Four distinct classifiers, namely DT, NB, ADB, and KNN, are used to evaluate the classification accuracy of the suggested model.

The rest of the paper is structured as follows. Section 2 summarizes previous related work on data privacy and anonymization strategies. Section 3 describes a distributed data model for processing big data Geometric Data Perturbation employing a random rotation matrix (RX). Section 4 discusses the suggested model's performance outcomes. In the concluding section, the contribution of the proposed method was mentioned in Section 5.

2. Literature Review

Le et al. (2018) devised a privacy-preserving methodology based on m-signature and fuzzy processing. This method made use of fuzzy processing to ensure the utility and privacy of data. This method processed update operations using a heuristic algorithm in order to reduce information loss when using released data. Additionally, the approach proved more secure in terms of revocation of anonymity and link ability of selected data. In this case, the -Ching divination, niching evolutionary algorithm, or greedy heuristic algorithm can be utilized to perform the privacy protection model's update activities.

Liu et al. (2018) established a complete privacy-preserving architecture for cloud users to adhere to privacy requirements. The framework uses key-value query processing to ensure the privacy of the query, storage, and data. A method was created here that utilized a baseline scheme that utilized commutative encryption to provide privacy. The query processing was accelerated by balancing security and performance in order to minimize the expense of commutative encryption. The research was conducted by weighing the feasibility and effectiveness of two different implementation approaches. Complex queries, such as semi-

join queries, were not supported by this approach. Additionally, the method omitted advanced cryptographic mechanisms for enhancing the framework's security.

Eyupoglii et al. (2018) developed an algorithm for data anonymization that preserves both privacy and utility in huge data. The scalability and viability of this approach were assessed using a variety of datasets of differing sizes. In this case, a chaotic function was used to disturb the data in order to ensure its eligibility for sharing and publication. This algorithm was found to be superior in terms of F-measure, accuracy, and Kullback—Leibler divergence.

Using dispersed and incremental datasets, Aldeen et al. (2016) devised an anonymization technique for achieving adequate privacy protection while maximizing data value. This approach was employed in this case to initiate privacy security with the most usefulness possible by utilizing incremental and dispersed datasets. Additionally, the progressive anonymization technique allows for fine-grained control of the performance overhead. The combined anonymized datasets were used to maintain privacy when stored

Srisungsittisunti and Natwichai (2015) created a strategy for preserving privacy called the polynomial-time algorithm. Here, computational complexity was greatly decreased while ensuring the optimal result. The methodology proved effective and extremely efficient in comparison to other methods of privacy preservation. The approach made no attempt to study the difficulty of querying a subset of the dataset. Additionally, the concurrency control issue remained a significant concern for operations such as adding, deleting or querying the dataset.

Panackal and Pillai (2015) devised an intelligent strategy for commencing privacy preservation called Adaptive Utility-based Anonymization (AUA). This approach utilized a variety of data sets and data providers to tailor the QI attributes to the users' preferences. The model was an anonymized version that gave maximum benefits, including k-anonymity for privacy protection, while also providing the highest amount of information to consumers.

Zhang et al. (2013) devised a quasi-identifier index-based strategy for preserving privacy while achieving high data usefulness in dispersed and incremental data sets stored in a cloud architecture. To maximize efficiency, Quasi-identifiers were employed to represent the anonymized data in this case. The solution utilized computation-intensive programs to process large datasets, but it did not take into account privacy-aware efficient scheduling for anonymizing data sets in the cloud or commencing privacy preservation in the cloud. In the investigation described by Dhiraj et al. (2009), data clustered using the k-mean clustering approach were disturbed by rotating the cluster centres at various angles. In further work, a four-dimensional rotation model for anonymization was presented by Javid et al. (2019). After a thorough investigation, it was found that the data was divided into two groups, and the values of each group were updated by rotating them on the xy and zw planes accordingly.

Geometric data perturbation was employed in a study by Sreekumar et al. (2012) to ensure this level of privacy. The perturbation of geometric data contains method steps. Altering the order of these steps was investigated in a distinct study by Oliveira et al. (2010); the effects of rotation, noise addition, and scaling on the outcome were discussed. In another extensive study of Geometric data, perturbation was studied in depth by Chamikara et al. (2019). Data separation and geometric approaches were used in the same investigation, which gave

excellent results. In a related work by Yeliz (2019) the same authors, an effective and dependable perturbation approach using the Laplace noise method was proposed.

A new way of generating the random noise was proposed by Merve Kanmaz using random generators [15].

3. Problem Definition

Data disruption aids in the preservation of privacy. It assists data owners in disseminating data while protecting sensitive personal information. Before posting material, the owner may edit it to remove sensitive information while maintaining the integrity of the piece. Perfect data disruption reduces the loss of information and privacy. Approaches to data disturbance that maximize privacy guarantee compromise data utility. Designing a data perturbation approach is difficult due to their intrinsic interdependence. The majority of perturbation techniques do not balance these two variables. Geometric data perturbation simplifies the process of balancing data utility and data privacy while maintaining both.

Existing standard data mining methods that protect privacy have the following drawbacks:

- Traditional privacy preservation solutions are unable to handle large volumes of data due to computational complexity.
- Confidential data will be at risk since present privacy preservation measures are less reliable and more vulnerable to attack.

4. Proposed Methodology

According to a survey of recent publications, perturbation approaches are still being employed often in de-identification investigations now adays. Geometric data perturbation with random number generator (Noise) has a high success rate among perturbation methods when tested with smaller datasets.For this reason, in this study, we focused on describing a distributed data model for privacy preserving of big data using geometric data perturbation and random number generators. The following

Figure1 Shows the work flow of proposed method.

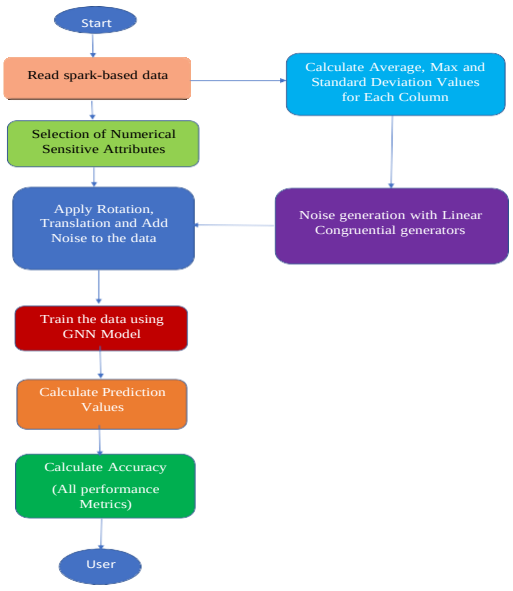


Fig: 1 Flow chart of the proposed work

a. Stepwise explanation of workflow

1. Select BIG data
2. Feature Selection and Extraction: Identify the Sensitive attributes from the database

a) Variance threshold (delete the unnecessary value whose outcome of variance calculation is very low)

b) Co-relation coefficient (delete the value based on how a particular attribute is co-related with another attribute)

3. Split the dataset into train and test (Xtrain, Ytrain, Xtest, Ytest)
4. Apply Rotation on Training data.
5. Apply Translation on Training data.
6. Add Noise to the Training data.

a) Add **GAUSSIAN** Noise to the copy of training data.

b) Add **RSUGP** noise to the main training data.

c) Add **RANDOM NORMAL** noise to the copy of training data.

7. Calculation of Edges and Nodes based on the perturbed data.
8. Graphical representation of edge and node.
9. Train the data using GNN (Graphical neural network).
10. Normalize the node data to calculate Friedman Mean Score.
11. Visualize the performance matrices of GNN.
12. Calculating the performance matrices through Decision Tree, Naive Bayes, Adaboost, and KNN, using the same training testing perturbed data that is provided as an input to GNN.
13. Create some signals and wavelength using training and testing perturbed data for ICA (Independent Component Analysis) attack resistance.
14. Calculate the s-value and signal/wavelength flow graph through ICA algorithm.
15. Execution time comparison between all the noise generation algorithms.

b. Geometric Data Perturbation

Geometric data perturbation (GDP) consists of a sequence of random geometric transformation, including distance perturbation (noise additive component), translational transformation (T), and rotation matrix (R):

$$G(X) = RX + T + \Delta \quad (1)$$

The matrix element R is a rotation matrix, component T is a translation transformation, and delta is an additive noise component.

Table 4. 1 transformation before perturbation

	VhcPerMin	StopagePerHour
0	78.0	47.0
1	76.0	48.0
2	76.0	48.0
3	60.0	61.0
4	60.0	61.0

Table 4. 2transformation after data perturbation

	VhcPerMin	StopagePerHour
0	-0.021629	-0.027293
1	-12.429631	-15.219956
2	0.561459	0.715860
3	-0.068247	0.715860
4	5.906125	7.490695

c. Translational transformation

The additive noise-based disruption of sensitive data on the Z data set is referred to as data translation. During the translation process, a constant value is applied to an attribute's total value.

Table 4. 3 transformation before translation

	VhcPerMin	StopagePerHour
0	42.0	53.0
1	49.0	60.0
2	40.0	51.0
3	41.0	52.0
4	41.0	52.0

Table 4. 4transformation after translation

	VhcPerMin	StopagePerHour
0	9.01134	5.42991
1	8.78028	5.54544
2	8.78028	5.54544
3	6.3180	7.04733
4	6.3180	7.04733

d. Multiplicative transformation

The multiplicative transformation approach is used to rotate sensitive data on the Z data set. In order to convert the data, a matrix is multiplied by a series of highly sensitive numerical values. A random projection matrix or random rotation matrix can be generated using this matrix. The projection matrix, on the other hand, does not preserve the distance exactly after processing the data. As a result, the random rotation matrix is the method of choice. A matrix for rotating a set of N-dimensional coordinate system points around a predetermined axis is called a rotation matrix (4.1).Rotation angle, $\alpha = 90$ degree

Table 4. 5 transformation before data perturbation

	VhcPerMin	StopagePerHour
0	78.0	47.0
1	76.0	48.0
2	76.0	48.0
3	60.0	61.0
4	60.0	61.0

Table 4. 6 transformation after data perturbation

	VhcPerMin	StopagePerHour
0	42.0	53.0
1	49.0	60.0
2	40.0	51.0
3	41.0	52.0
4	41.0	52.0

e. Noise Addition

In the geometric data perturbation, similar translation and rotation procedures have been applied as a result of the researched studies.In various investigations, random noise has been used as a noise model. However, in the majority of studies, Gaussian noise was used, which produces more effective findings due to its use of data, namely the mean and standard deviation. The data is used by Gauss to generate a random noise value.[15]

Table 4.7 transformation before noise addition

	VhcPerMin	StopagePerHour
0	9.01134	5.42991
1	8.78028	5.54544
2	8.78028	5.54544
3	6.3180	7.04733
4	6.3180	7.04733

Table 4. 8 transformation after noise addition

	VhcPerMin	StopagePerHour
0	-0.021629	-0.027293
1	-12.429631	-15.219956
2	0.561459	0.715860
3	-0.068247	-0.086558
4	5.906125	7.490695

f. Graphical Neural Network (GNN)

Graph Neural Networks (GNNs) are a category of deep learning techniques meant to do inference on graph-described data. GNNs are neural networks that may be directly applied to graphs and facilitate prediction tasks at the node, edge, and graph levels. GNNs are able to accomplish what Convolutional Neural Networks (CNNs) could not.

More practical applications of GNN include traffic control,human behaviour detection, molecular structure study, program verification,recommender system, social influence prediction,logical reasoning, and adversarial attack prevention.

The main idea behind this framework is that each node feature is updated with the features of its neighbours. The neighbour features are passed to the target node as messages through the

edges. As a consequence, the new representation of the node encodes and represents the local structure of the graph. It consists of the following layers.

The dataset we are using (Traffic based IOT Sensor dataset) is not suitable for feed-forward network/or any other common distributed neural network since those NN doesn't have multi-layer node layer handling capacity and doesn't work on tree-basedhierarchical model while GNN can make the traversing of data easier in a distributed network system, sothe used dataset will have best chances to give better results in GNN.After training of our GNN model obtained approx...93% of test accuracy which means around 7% of our testing data is inaccurately predicted or lostsx so data loss is there but has very little effect because no NN with big data can't provide 100% accuracy, so the loss of data is always there.

Classification accuracy of the proposed model (GNN with RSUGP) is investigated using four different classifiers which are namely DT, NB, ADB, and KNN. The training and testing data is split in the ratio of 75:25.

g. System architecture

- **Base system configuration**

Computer with 16 GB RAM, an Intel with i5processor, and Windows 10, PySpark Library for Spark Connection for Big Data, and TensorFlow (Anaconda) is used as a framework. Jupyter Notebook is used as editor.Scikit-Learn library for neural network, and classifier support,Matplotlib, Seaborn are used for data visualization, Network, os-sys in Python was used to test for attack resistance, and creation of ICA model.

5. Performance Evaluation

Dataset description

Traffic Based IoT Sensor Big Data is used for this research work. A collection of vehicle traffic data observed between two places for a predetermined period of time over the course of six months (449 observation points in total). Using the city pulse information model, the data is accessible in both raw (CSV) and semantically annotated formats (semanticannotation)

Dataset link

http://iot.ee.surrey.ac.uk:8080/datasets.html

Table 5. 1 Description of Traffic based IOT Sensor dataset

Datasets	Description
Traffic based IOT Sensor big data	Total Instances: 3795475 Attributes: 9

Table 5. 2 Description of Attributes in Traffic based IOT Sensor dataset

Attribute	datatype
Cell	String
VhcPerMin	Float
StopagePerHour	Float
GenVal	Float
VhcCycMin	Float
Date	DateTime
DetecPos	Integer
TrafficRatio	LongFloat
MaxPoint	Float

Base Dataset Shape-(3795475, 9)

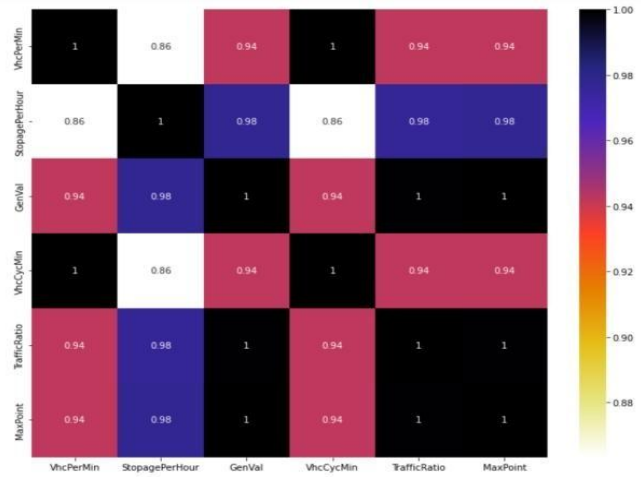
- **Feature Selection and Extraction**

  Variance Threshold

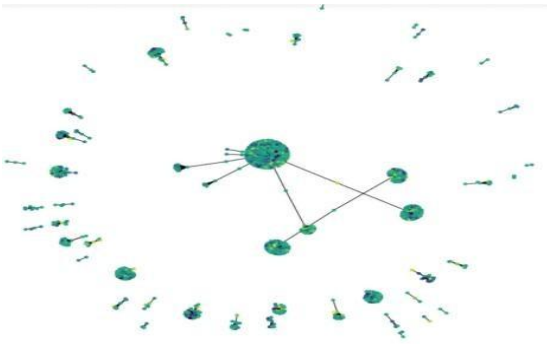
  A selector for eliminating all low-variance traits. This feature selection approach considers only the features (X_train) and not the desired outputs (y_train), and is therefore applicable for unsupervised learning.

- **Co-relation Coefficient Matrix (Heat map View)**

  This heat map represents a co-relation between two attributes, the co-relation score is written when the row and column of specific attributes collide with another's of row/column. There is a numeric score written on that area where two cell collides that is the co-relation value of two attributes. −1‖ is representing-highly co-related, lower than −0.5‖ represents non-co-related.



- **Creating Nodes and Edge's Connections in GNN Model**



When we convert our tabular data into node base graphical, data is converted into nodes and edges format. Node represents the attribute;edges represent the relationship between the attributes.

- **Node and Edge Shape that is going to be passed to GNN**

```
Edges data shape: (4622, 2)

Nodes data shape: (4679, 2)
```

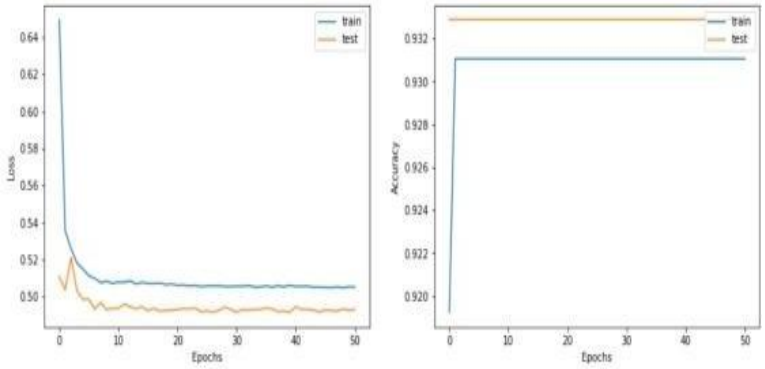
- **GNN Model Summery With all Layers**

Model: "GNN Model"

Layer (type)	Output Shape	Param #	Connected to
input_features (InputLayer)	[(None, 2)]	0	[]
ffn_block1 (Sequential)	(None, 32)	1288	['input_features[0][0]']
ffn_block2 (Sequential)	(None, 32)	2368	['ffn_block1[0][0]']
skip_connection2 (Add)	(None, 32)	0	['ffn_block1[0][0]', 'ffn_block2[0][0]']
ffn_block3 (Sequential)	(None, 32)	2368	['skip_connection2[0][0]']
skip_connection3 (Add)	(None, 32)	0	['skip_connection2[0][0]', 'ffn_block3[0][0]']
ffn_block4 (Sequential)	(None, 32)	2368	['skip_connection3[0][0]']
skip_connection4 (Add)	(None, 32)	0	['skip_connection3[0][0]', 'ffn_block4[0][0]']
ffn_block5 (Sequential)	(None, 32)	2368	['skip_connection4[0][0]']
skip_connection5 (Add)	(None, 32)	0	['skip_connection4[0][0]', 'ffn_block5[0][0]']
logits (Dense)	(None, 79)	2607	['skip_connection5[0][0]']

Total params: 13,367
Trainable params: 12,787
Non-trainable params: 580

- **Accuracy and Loess Graph for Training process in GNN**



Trajectory of accuracy to loss raetio during the training process.

Classification Accuracy

Classification accuracy refers to the proportion of data sets that were correctly classified following classification. TP is the number of positively labelled groups, TN is the number of positively labelled negative groups, T is all true, and F is all false. FP is the number of positive groups that have been labelled as negative, while FN is the number of negative groups that have been labelled as negative.

$$\text{Classification Accuracy} = (T\,P + T\,N)/(P + N) \quad (3)$$

Using four distinct classifiers, namely Naive Bayes, KNN, Decision tree, and Adaboost, the classification accuracy of the proposed model is evaluated.

The performance accuracy of the model was compared with the confusion matrix of different classifiers: the original classification accuracy (GNN) with a classification accuracy of DT, NB, KNN, and ADB. Friedman's rank test is calculated for GNN

Compared to the existing Naive Bayes and Decision tree classification methods, the performance of the proposed approach is either superior or equivalent in all circumstances. In the Decision tree, classifier outcomes from comparative methods are almost the same the majority of the time. The DT and NB classifiers are superior to the other two classifiers, but GNN outperforms all other classification methods in terms of classification accuracy.

- **Accuracy Comparison of GNN, Decision Tree, Naïve Bayes, Adaboost, KNN**

Table 5. 3 Accuracy Comparison of GNN, Decision Tree, Naïve Bayes, Adaboost, KNN

Model	GNN	Decision Tree	Naïve Bayes	Adaboost	KNN
Test accuracy Values	**0.9325**	0.9073	0.9073	0.7482	0.6843

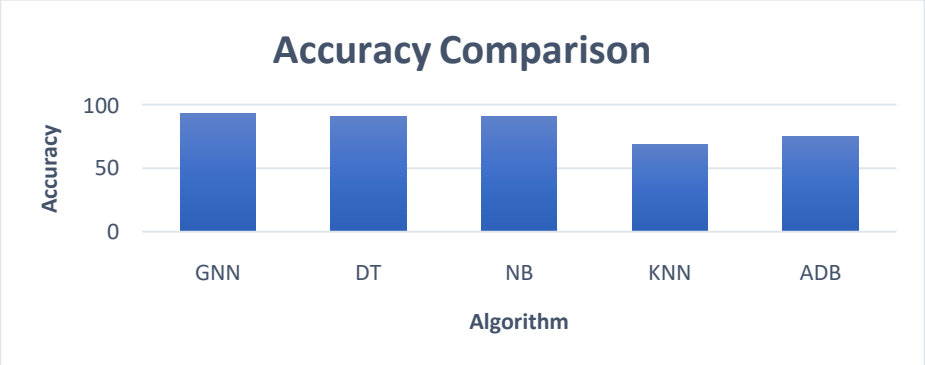


Fig 3. Accuracy comparison graph of different methods

Friedman Mean Scores

According to Friedman's test, repeated tests with diverse conditions are compared to one another and a value is determined. The greater value implies that the approach differs significantly from other methods. It is evident that the proposed strategy differs significantly from existing methods and produces superior outcomes.

$$F = \left[\frac{12}{Nk(k+1)} + \int_{i=1}^{k} R_i^2\right] - 3N(k+1)$$

Friedman Test statistic Score: 9244.0
Friedman Test PValue Score: 0.85

RSUGP's FMR Scores after ICA

Friedman Test statistic Score 153.14 Friedman Test PValue Score 0.76

Table:Accuracy comparison of different methods

Data Set		GNN	Naïve Bayes	Decision Tree	AdaBoost	KNN
Traffic Control	Gaussian Noise	89.68%	84.63%	84.63%	78.61%	59.89%
	RSUGP	93.25%	90.73%	90.73%	78.82%	68.43%
FMR Values	Gaussian Noise	statistic Score **8452.0**, PValue Score 0.63				
	RSUGP	statistic Score :**9244.0**, PValue Score :**0.85**				

ICA Attack Resistance in accordance with GNN Training/Testing Data

From the matrix multiplication-based de-identification algorithms [14], the original data can be retrieved. ICA (Independent Component Analysis) is the most commonly utilized. Measured values are referred to as "standard deviation" because of how much they deviate from "original" data and "perturbed" data. A closer look at these numbers reveals the recommended strategy is more secure.

Packet Distribution = 2000
Signals = 3
Wavelength Shape = (2000, 3)

Table:Results of Resistance to Attack

Data Set	Algorithm	ICA
Traffic Control	Gaussian Noise	0.93
	RSUGP	0.99
FMR Values	Gaussian Noise	statistic Score 34.9, PValue Score 0.83
	RSUGP	statistic Score 153.14, PValue Score 0.77

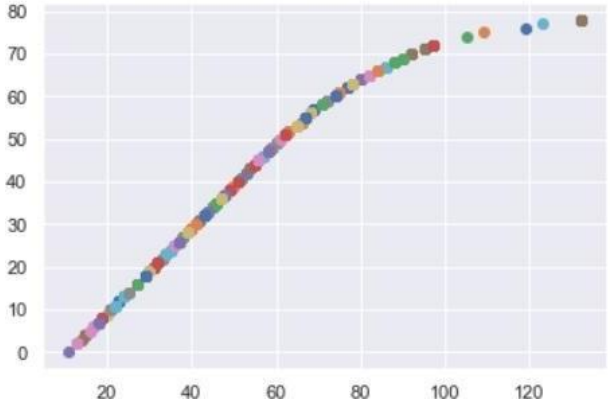
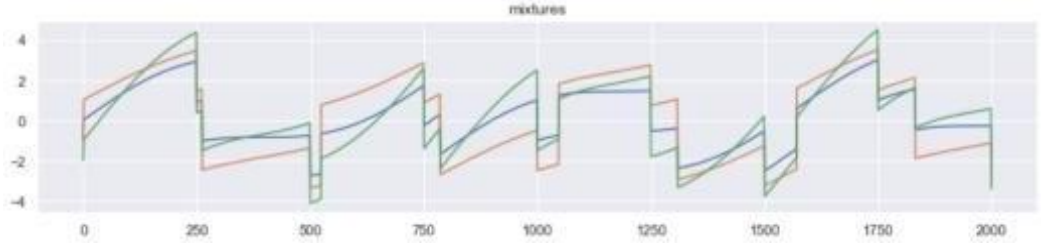
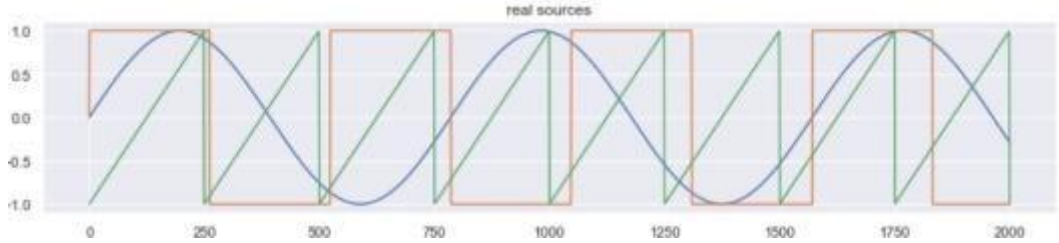


figure 2 x-axis represents number of transmission block, and y-axis represents signals for transmission.

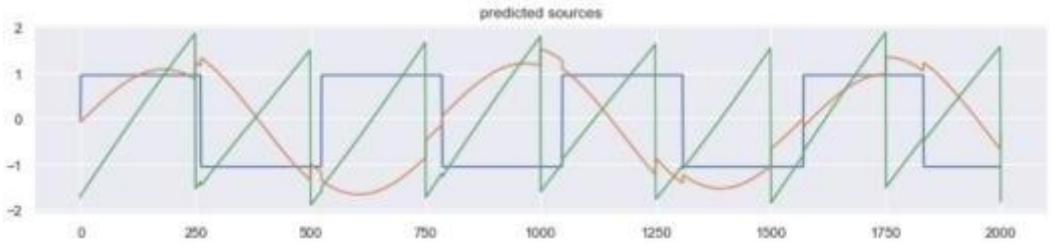
- **Packet Transmission Ratio Based on Training X-train data**



- **Packet Transmission Ratio Based on Training y-train data**



- **Packet Transmission Ratio Based on Training X-test data**



- **ICA S Score**

  It represents the delivery ratio score per wavelength

  S Score: 0.9948

Performance Analysis Time

Three different methods are used to analyse the traffic data in this study. To put it another way, if the data collection and the number of sensitive attributes selected grows, so does the execution time of the procedures. A random variable's statistical behaviour is defined by the probability density function, which is called Gaussian noise. In the graph, the Gauss model,

which generates noise based on the complete dataset, has a longer runtime than the other models. The proposed model RSUGP generates noise solely by selecting sensitive features and iteration values that are of interest to the model's creators. Because it doesn't process the entire data set, it generates noise from the data and does it in a much shorter period of time. Each of these approaches has a graph showing the average execution time of the two in relation to one another.

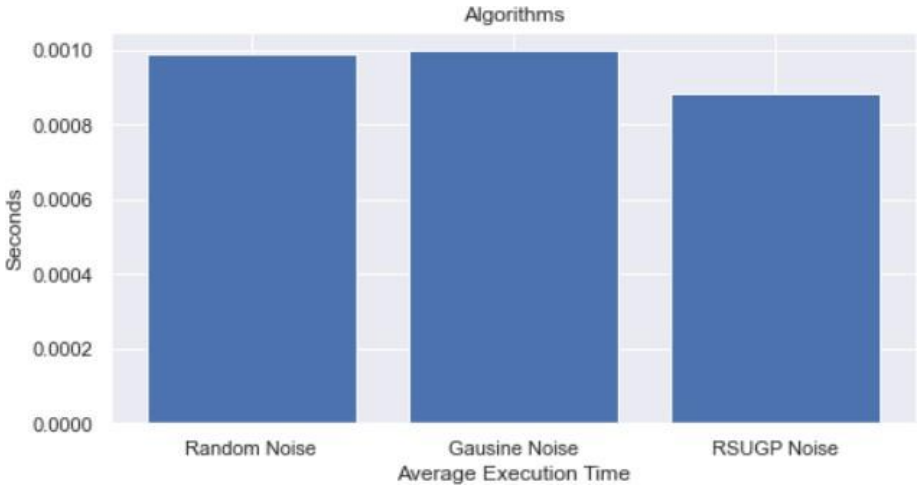


Fig 4. 1 Execution time comparison of noise generation Algorithms

6. Conclusion

AnRSUGP-basedGNN model for perturbing geometric data was proposed in this research. The suggested method's performance is evaluated using traffic data and compared to two alternative ways. In the non-parametric Friedman test, the obtained values were compared to the attack resistance, classification precision, and execution time of the system. According to our findings, RSUGP with random number generators has improved classification accuracy and other criteria, as well as a viable approach for protecting sensitive numerical data.

References

[1] Le, J., Zhang, D., Mu, N., Liao, X., and Yang, F., "Anonymous Privacy Preservation Based on m-Signature and Fuzzy Processing for Real-Time Data Release," IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2018.

[2] Liu, C., Zhou, S., Hu, H., Tang, Y., Guan, J., and Ma, Y., "CPP: Towards comprehensive privacy preserving for query processing in information networks," Information Sciences., 467, pp. 296-311, 2018.

[3] Eyupoglii, C., Aydin, M. A., Zahn., A. H., and Sertbas, A., "An Efficient Big Data Anonymization Algorithm Based on Chaos and Perturbation Techniques, "Entropy., 20(373), 2018.

[4] Dhiraj, S. S., Khan, A. M. A., Khan, W., & Challagalla, A. (2009, January). Privacy preservation in k-means clustering by cluster rotation. In *TENCON 2009-2009 IEEE Region 10 Conference* (pp. 1–7). IEEE.

[5] Javid, T., & Gupta, M. K. (2019, November). Privacy Preserving Classification using 4-Dimensional Rotation Transformation. In *2019 8th International Conference System Modeling and Advancement in Research Trends (SMART)* (pp. 279–284). IEEE.

[6] Sreekumar, K., & Baburaj, E. (2012). Privacy preservation using geometric data perturbation and fragmentation approach in wireless sensor networks.

[7] Oliveira, S. R., & Zaiane, O. R. (2010). Privacy preserving clustering by data transformation. *Journal of Information and Data Management*, *1*(1), 37–37.

[8] Chamikara, M. A. P., Bertok, P., Liu, D., Camtepe, S., & Khalil, I. (2019). An effificient and scalable privacy preserving algorithm for big data and data streams. *Computers & Security*, *87*, 101570.

[9] Yeliz GE. True Random Number Generation Based on Human Movements. *BEU Journal of Science.* 2019;8(1):261–9.

[10] Aldeen, Y. A. A. S., Salleh, M., and Aljeroudi, Y., "An innovative privacy preserving technique for incremental datasets on cloud computing," Journal of biomedical informatics., 62, pp.107-116, 2016.

[11] Srisungsittisunti, B., and Natwichai, J., "An incremental privacy-preservation algorithm for the (k, e)-Anonymous model," Computers & Electrical Engineering., 41, pp_126-141, 2015.

[12] Panackal, J. J., and Pillai, A. S., "Adaptive utility-based anonymization model: Performance evaluation on big data sets," Procedia Computer Science., 50, pp.347-352, 2015.

[13] Zhang, X., Liu, C., Nepal, S., and Chen, J., "An efficient quasi-identifier index based approach for privacy preservation over incremental data sets on cloud," Journal of Computer and System Sciences., 79(5), 542-555, 2013.

[14] Kupeli C., Bulut F. Performance Analysis of Filters over Salt-Pepper and Gauss Noises in Images.*Halic¸ Universitesi Fen Bilimleri Dergisi.*3(2):211–39. DOI: 10.46373/hafebid.768240

[15] Merve Kanmaz1,∗ , Muhammed Ali Aydin2 and Ahmet –A New Geometric Data Perturbation Method for Data Anonymization Based on Random Number Generators‖ https://journals.riverpublishers.com/index.php/JWE/article/view/7983