International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Investigations and Design of Privacy-Preserving Data Mining Technique for Secure Data Publishing

Mayur Rathi[1], Anand Rajavat[2]

Submitted: 25/04/2023 **Revised**: 28/06/2023 **Accepted**: 07/07/2023

Abstract: Growing digital data motivates us to develop data-intensive applications, for providing diverse areas of solutions. In this context, sometimes we need cross-domain data, where multiple data owners are contributing their data. In this situation, data dimensionality and privacy is the main concern. Thus, to deal with the privacy issues recent techniques are applying cryptographic and noise-based techniques. But, these techniques are resource-consuming and suffer from poor performance in terms of data utility. In this paper, we investigate a Privacy Preserving Data Mining (PPDM) model to deal with complexities of PPDM modeling. First, we have described the dimensionality issue with PPDM and discuss the utilization of dimensionality reduction techniques such as Principle Component Analysis (PCA), Linear Discriminant Analysis (LDA), Kernel Principle Component Analysis (KPCA), and Correlation Coefficient (CC). Second, we have considered the issue of privacy and utility of data. The aim is to balance privacy requirements and data utility. Thus, a client-side random noise inclusion algorithm is proposed. Finally, by picking suitable and effective techniques, we implemented a complete PPDM model. The experiments on publically available UCI datasets are performed. Finally, performance in terms of data quality matrix, privacy matrix, and performance matrix is discussed.

Keywords: *PPDM, Privacy, Noise Inclusion Algorithm, Dimensionality Reduction, Experimental Study.*

1. Introduction

The different business domains are utilizing Business Intelligence (BI) tools. These tools are utilized to analyze consumer data and capture essential insights for business administrators. The business administrators are utilizing these insights for planning and future business growth and sustainability. Here, some of the data is collected from different other data sources. But, data collection has security and privacy risk. In this context, we need a Privacy-Preserving Data Mining (PPDM) environment. In the PPDM environment, data mining is essential, which helps to understand and distinguish data variations, perform decision-making, and predict future trends [1] [4]. The PPDM is essential because the collected data may contain a person's confidential, sensitive and private information [41]. The disclosure of such data can harm the person socially and financially [2] [5]. Principally, PPDM is used when different data owners are associating their data for concluding a common solution. Even though, no one wants to expose their data to others [3].

1.1. *Problem Domain*

The data disclosure under PPDM scenarios in public or in the use of experiments can be harmful to both i.e. a company's reputation and the end client. Therefore, before data publishing, we need to sanitize sensitive information. That prevents privacy loss done either intentionally or by mistake [6]. In addition, there are some other issues, which need attention.

- **Data Dimensionality:** When multiple parties are combining the data, the dimension can be increased. The processing of higher-dimensional data requires a significant amount of processing cost i.e. time and space. Thus, we are required to identify the techniques to minimize computational overheads [7].
- **Impact of data sanitization process:** The sanitization process is required for hiding the sensitive attributes, which influences the application's quality of service. Because, when a Machine Learning (ML) algorithm learns from noisy data, it impacts the learning algorithm's performance. Therefore, it is required to make a balance between noise inclusion and quality of data [8].
- **Use of ML technique:** Supervised learning techniques are popular in data mining [9]. The employment of an appropriate ML algorithm is essential to satisfy the need of application [10].
- **Data collaboration:** Sometimes it is not feasible to take decisions alone. Therefore, it is required to delegate the information from other sources to smoothen the decision-making process. The delegation of data is required to be secure and

1,2Department of Computer Science & Engineering,
Shri Vaishnav Vidyapeeth Vishwavidyalaya
Indore, India
1mayurrathi.31dec@gmail.com, 2anandrajavat@yahoo.co.in

preserve privacy [11][40].

1.2. *Related work*

The PPDM is a promising area of BI. For understanding the role of PPDM, we consider the tour and travel industry, where hotels, restaurants, cabs, and others have dependencies on each other. Thus, they can combine their data for making future strategies [12]. We can understand this using one more example where different medical institutions wish to conduct joint research while preserving the privacy of their patients. Here we need a trusted third party. The trusted party collects the data from all the sources, who want to participate. In this context, we need a Secure Multiparty Computation (SMC) [13]. Additionally, to understand the working of PPDM, we have studied different recent contributions. Chen et al. [14] review techniques and applications. They show when the data is in higher dimensions then classifier performance becomes low, thus needing to employ a Dimensionality reduction technique. These techniques can work both unsupervised [15] and supervised [16]. Sunitha et al. [17] used noisy data and outliers to study the impact on classifier performance. Xiong et al. [18] suggest the technique of noise removal. And Zhang et al. [19] describe the issues of privacy in multiparty data collaboration. These techniques are useful in dealing with multiparty data for secure computation.

1.3. *Components of the PPDM framework*

PPDM framework involves four key components, which are as follows:

- **Connectivity among parties:** To satisfy the requirements of PPDM in multi-party settings. The framework is responsible for, Connectivity with ML server to data owners, and also enables security and privacy.
- **Data synchronization:** Data synchronization required with all the parties. The un-synchronized data can misguide the ML algorithm. Therefore, need to verify the data instances and relevance predictable variable during collaboration.
- **Noise and Dimensionality reduction:** It is required to verify the quality of data and utility. Thus, we need to deal with the impact of noise and the cost of processing.
- **Classification and decision making:** It is used for mining the data to make decisions. Additionally, the consequences can be distributed to all the participating parties. It is also responsible for maintaining privacy to ensure the application's requirements.

1.4. *Contributions*

PPDM is important, because of the increasing need to store user data and utilization of ML algorithms for analysis. It is a multiparty environment to achieve a common objective. Thus, the aim is to preserve the expected level of privacy with minimum information loss or utility. Thus, paper demonstrates the following contributions:

- Analysis of Data Dimensionality Reduction Techniques.
- Measure Impact of sanitization process over Classifier's Performance.
- Development of Optimal Noise Mixture Algorithm.
- Design of an Efficient Common Data Publishing Technique.

2. Impact of Dimensions

In this experiment, the datasets are used from the public UCI database. Additionally, a provision is developed to experiment with the PCA, KPCA, LDA, and CC-based dimensionality reduction techniques. The experimental system is defined in Figure 1. The dimensionality reduction techniques are used for selecting suitable features [27]. The figure 1 shows the technique for demonstrating the comparative performance. The ML algorithms are required to take training. Thus, five datasets are involved in this experiment. Here, a provision is developed to select an algorithm to experiment with the PCA, K-PCA, LDA, and CC techniques.

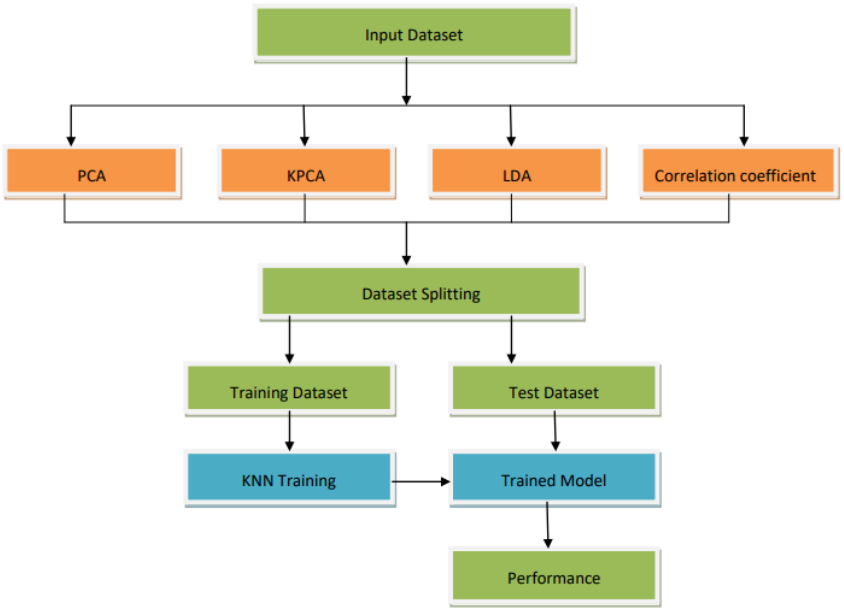


Fig. 1. Model for Feature selection Methods Comparison

PCA: In PCA [22][42], using d dimensions we construct a covariance matrix of size d x d. In this matrix, for two features x_j and x_k we can use the following equation:

$$\sigma_{j,k} = \frac{1}{N}\sum_{i=1}^{N}\left(x_j^{(i)} - \mu_j\right)\left(x_k^{(i)} - \mu_k\right) \qquad (1)$$

Where, μ_k and μ_j is mean of features, when covariance is positive then we get increase or decrease in features, and when it is negative then features found with opposite directions. The principal components are described using eigenvector, and Eigen values. Obtain Eigen-pairs satisfies:

$$\sum v = \lambda v \qquad (2)$$

Here, λ is a scalar Eigen-value. To reduce dimensions new feature subset of eigenvectors with higher variance.

KPCA: The K-PCA is a non-linear dimensionality reduction method [23]. The PCA is finding components which will maximize the variance by highest Eigen values.

$$cov = \frac{1}{N}\sum_{i=1}^{N} x_i x_i^T \qquad (3)$$

The kernel function is used for mapping of data into higher dimensions:

$$cov = \frac{1}{N}\sum_{i=1}^{N} \phi(x_i)\phi(x_i)^T \qquad (4)$$

Basically, it project covariance matrix in higher dimensional space. Thus compute the kernel matrix [24]

$$k(x_i, x_j) = \exp\left(-\gamma\left\|x_i - x_j\right\|^2\right) \qquad (5)$$

Then, Eigen decomposition of kernel matrix is carried out

$$k' = k - 1_n k - k 1_n + 1_n k 1_n \qquad (6)$$

Where 1_n is a matrix of n by n, and the values is equivalent to $\frac{1}{N}$. Now, we need to obtain eigenvectors of the kernel relevant to higher Eigen values. The projected data is given in eigenvectors.

CC: To established relationship among two variables we can use Correlation Coefficient (CC). The CC is producing the values in range of -1 to +1. The values +1 demonstrate strong relationship, 0 shows no relation and -1 describe opposite nature of features. The calculation of CC is performed using [25]:

$$CC = \frac{n(\sum xy) - \sum x \sum y}{\sqrt{[n\sum x_i^2 - (\sum x_i)^2][n\sum y_i^2 - (\sum y_i)^2]}} \qquad (7)$$

Where sample size is n, individual samples are x_i, y_i, and mean if x and y vectors are $\bar{x}, \bar{y}$. In order to reducing dimensions, we consider x as attributes and class labels as y. After calculating CC all attributes are placed in sorted order and higher CC attributes are selected for further use [26].

LDA: LDA is a popular approach for dimensionality reduction. It is used for finding a vector in vector space that provides better separation. In this method we are projecting the data for better separability. It is used when the classes are overlapped, to better classification the LDA maximize the fisher ratio using:

$$\frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2} \qquad (8)$$

Where, σ_1 and σ_2 is variance of first and second class. (μ1 − μ2) is the difference between means of the classes.

Fisher ratio will increase the distance among classes. SB is making the classes condense by minimizing $\sigma_1^2 + \sigma_2^2$. Thus, the fisher ratio can be written as:

$$\frac{S_B}{S_W} \quad (9)$$

The objective is to maximize ratio by transforming data to lower dimensional space, using a matrix w. So SB and SW can be defined as:

$$S_B = w^T S_B w \quad (10)$$

$$S_W = w^T S_w w \quad (11)$$

Thus,

$$\frac{w^T S_B w}{w^T S_w w} \quad (12)$$

LDA evaluates w matrix using eigenvectors of $S_W^{-1} S_B$. The LDA uses the transformation matrix to transform p dimensional data into k dimensions. LDA improves the strength of ML model by projecting the features into low vector space. Additionally, it minimizes the time complexity of the learning algorithm.

After applying the dimensionality reduction techniques the selected features are divided into training (70%) and testing (30%) dataset is used [28]. After dataset splitting the K-nearest neighbor (k-NN) classifier is used. The k-NN is the usage of Euclidean distance [29], which is represented as:

$$D(M,N) = \sqrt{\sum_{i=1}^{n} (M_i - N_i)^2} \quad (13)$$

Where, $D(M,N)$ is the distance between vectors M and N. And the total number of samples is defined as n. The table 1 shows the process of the implemented dimensionality reduction testing technique:

Table 1 Algorithm to Test Dimensionality Reduction Techniques

Input: Dataset D, Algorithm List $L_2 = \{L_0, L_1, L_2\}$, Number of features U

Output: Class Labels C

Process:

1. $D_n = readDataset(D)$
2. $SA = L_2.SelectAlgorithm(U)$
3. $F_m = SA.ReduceDimension(D_n)$
4. $[Train, Test] = F_m.Split(70,30)$
5. $for(j = 1; j \leq Test.Lenght; j++)$
    a. $C = KNN.Classify(Test_j)$
6. $end\ for$
7. Return C

The dimensions of the data are affecting the performance in terms of processing time and memory usage. The processing time is the time required for extracting patterns [30]. The mean time consumption of the dimensionality reduction algorithms is given in table 2 and fig. 2(c). The results are measured here in milliseconds (MS). According to the experimental results, PCA is expensive as compared to the other implemented approaches, and, KPCA, LDA, and CC show similar performance, but most of the time CC shows superiority.

Table 2 Effect of Dimensionality Reduction Techniques on Classifier's Performance

	Accuracy				Error Rate				Time				Memory			
Dataset	PCA	KPCA	CC	LDA	PCA	KPCA	CC	LDA	PCA	KPCA	CC	LDA	PCA	KPCA	CC	LDA
diabetes	74.4	88.34	97.21	89.28	25.6	11.66	2.78	10.72	370.62	1127.25	287.5	370.87	1208.25	1370.12	1337.12	1688

Iris	73.01	86.39	95.39	87.6	26.99	13.61	4.61	12.4	115.62	205.12	64.25	133.12	1171.87	1177	1264.75	1242.37
heart	74.01	87.85	96.71	88.81	25.99	12.15	3.29	11.19	314.87	627.37	174.75	300.62	1222.37	1448.37	1472.5	1558.87
wine	65.62	71.5	85.75	78.75	34.37	28.5	14.2	21.25	66	99.5	19.37	86	1089.75	1006.25	1085.25	1134.37
vehicle	74.4	82.57	97.21	89.27	25.6	17.43	2.78	10.72	424.25	1678.5	386.25	599.5	940.12	1434.25	1613.12	2044.62

In addition, memory usages of an algorithm show space complexity. That is the part of the main memory utilized during the process execution. The memory usage is reported in bar graph 2(d). The Y-axis denotes the utilized memory in kilobytes (KB). The result shows the CC and PCA are less expensive as compared to the KPCA and LDA. Accuracy is the description of successful classification based on the correctly classified samples. The accuracy of dimensionality reduction techniques is demonstrated in fig. 2(A). The results show the CC provides high accuracy as compared to KPCA, LDA, and PCA. Similarly, the error rate indicates the misclassification rate of algorithm. It is reported in fig. 2(b). The error rate for the PCA-based classification is higher as compared to KPCA, LDA, and CC-based techniques. This section demonstrates how ML performance is influenced by data dimensions. According to the results, the CC-based technique is accurate and efficient.

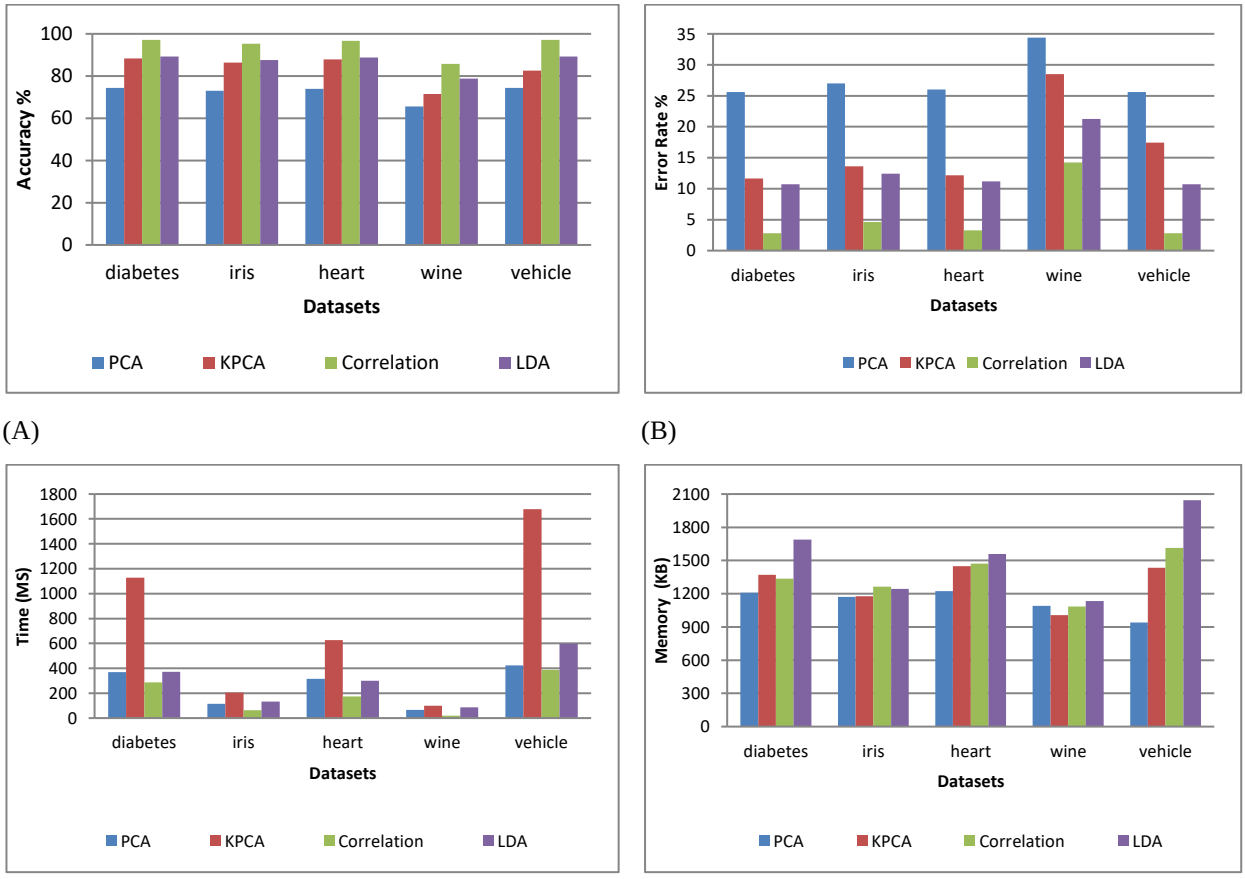


Fig. 2. Performance of Dimensionality Reduction Techniques (A) Accuracy (B) Error Rate (C) Time consumed (D) Memory Usages

3. Effect of Sanitization Process

The preprocessing techniques are adopted to reduce noise and less relevant information. These steps optimize the data quality and improve the classification accuracy [31]. Unlike the normal preprocessing, in PPDM we utilize the noise with data to sanitize. However, unregulated amount of noise can impact the application's performance negatively [32]. Thus, how and when the noise affects the data utility is needed to know [33]. The different techniques of noise inclusion with their advantages and disadvantages are given in table 3 [34] [39][43][44][45].

Table 3 Data Sanitization Techniques

Technique	Advantages	Disadvantages
Perturbation	Simple, Efficient, Accurate, preserve statistical information, Treat attributes individually	Information loss, Modified Data mining technique required
Condensation or Aggregation	Preserve statistical information, suitable for synthetic data, Provides security	High information loss due to larger size data in a single group, influence data mining outcomes
Anonymization	Preserves identity of individuals	High information loss, No strength on linking attacks
Cryptography	Multi-party support, Supported by number of cryptographic algorithms	Limited number of parties are supported, less secure output, Low security of sensitive data
Swapping	Re-identification of data is complex, dataset originality is maximized	Patterns analysis is time taking, less secure for diversity attack
Randomization	Simple, Efficient, prior knowledge of data distribution not required	All data treated similarly also with outliers, Data reconstruction is not feasible

In this table, the data sanitization methods are described, which are applicable for centralized and distributed databases. Using these techniques the original dataset values are replaced with synthetic data. Among them, random noise is an effective method to introduce the noise. But the random noise is suitable for the numerical data. Therefore, we proposed an extended version of random noise for supporting the text attributes also. Additionally, a model is also demonstrated to perform the comparative study, for finding the impact and influence of noise on the ML algorithm's performance. The model for noisy data classification is demonstrated in fig. 3.

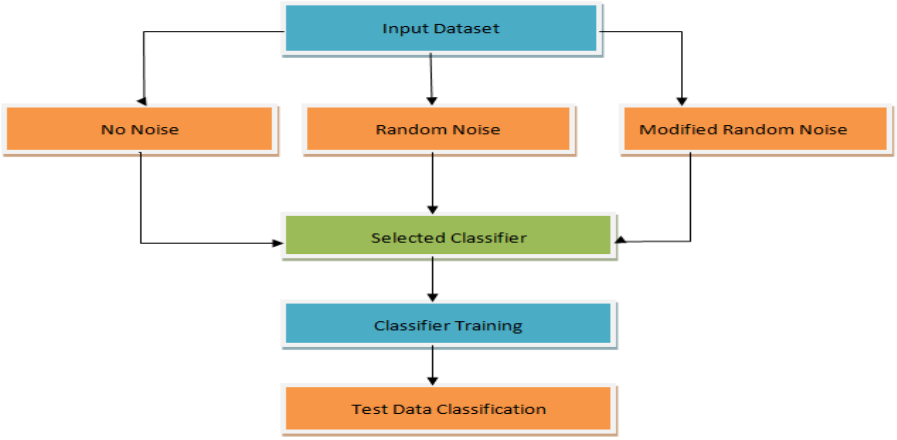


Fig. 3. Noisy data Classifier

The UCI machine learning repository [35] based datasets are used and discussed in table 4. Both types of datasets numerical and categorical are used. Next, we need to prepare the noisy dataset.

Table 4 Datasets Used

Dataset Name	Attribute Type
Diabetes	Numeric
Heart	Numeric
Iris	Numeric
Vehicle	Numeric
Forestfires	Numeric & categorical (Mix)

There are three components are described, first is no noise or original dataset. Second is random noise, which implements the classical additive random noise. In this context, a range of values is provided which generates a random value to replace original dataset values. But, it is not able to work with categorical data. Thus, we implement a technique that can improve random noise to enable the process of the categorical data also. The process of noise inclusion is given in Tables 5, 6, and table 7. Table 5 demonstrates the computation of the amount of noise that needs to be added to data, and table 6 provides the process to include the noise in data. The algorithm described in table 5 computes the fraction for adding noise into the data. According to the algorithm, first, we have computed the number of rows and columns. Then, for all the columns we check the data types to know attribute is numeric or categorical. If the attributes are identified as numerical, then we calculate a mean value, which is denoted as μ. Using μ we calculate the variance σ^2. Similarly for categorical attributes we are utilizing the length of attribute values to compute μ and σ^2. Next, the noise factor is calculated using:

$$\xi = \frac{var}{col} \tag{14}$$

Where, var is the sum of variance of all attributes, col is the number of attributes, and ξ is the noise to be added. To manipulate the dataset values this ξ is used.

Table 5 Noise Factor Computation

Input: Dataset D

Output: noisy to be add ξ

Process:

1. $[row, col] = readDataset(D)$
2. $Var = 0$
3. $for(i = 1; i \leq col; i++)$
    a. $if\ (D_i.isNeumeric == true)$
        i. $\mu = \frac{1}{row}\sum_{j=1}^{row} D(j,i)$
        ii. $\sigma^2 = \frac{1}{row}\sum_{j=1}^{row}(D(j,i) - \mu)^2$
    b. $else$
        i. $\mu = \frac{1}{row}\sum_{j=1}^{row} D(j,i).length$
        ii. $\sigma^2 = \frac{1}{row}\sum_{j=1}^{row}(D(j,i).Length - \mu)^2$
    c. End if
    d. $var = var + \sigma^2$
4. $End\ for$
5. $\xi = \frac{var}{col}$
6. $return\ \xi$

Table 6 shows the process to manipulate all the dataset values. Thus we normalize the dataset using the min-max normalization technique [36] using:

$$Norm = \frac{Actual\ Value - min}{max - min} \tag{15}$$

The normalized values and noise factor ξ is used to introduce multiplicative noise. The noisy values are used to replace the original values.

Table 6 Noise Add-on Algorithm

Input: noise to be Add ξ, Dataset D
Output: noisy dataset N

Process:

1. $[row, col] = readDataset(D)$
2. $for(i = 1; i \leq col; i++)$
    a. $if\ (D_i.isNeumeric == true)$
        i. $min = getMin(D_i)$
        ii. $max = getMax(D_i)$
        iii. $for(j = 1; j \leq row; j++)$
            1. $Norm = \frac{D(j,i)-min}{max-min}$
            2. $NewD(j,i) = Norm * \xi$
        iv. $end\ for$
    b. Else
        i. $for(j = 1; j \leq row; j++)$
            1. $NewD(j.i) = Randomize(D(j,i), \xi)$
        ii. $end\ for$
    c. End if
    d. $N.Add(NewD_i)$
3. End for
4. Return N

In addition, for dealing with the categorical data the supporting function is defined in table 7.

Table 7 Randomization Function for Categorical Data

Input: String S, Noise Value ξ, Character Array $C = \{a, \dots z, \&, 0, \dots, 9\}$

Output: Randomize String R

Process:

1. $SC_n = String2CharArray(S)$
2. $for(i = 0; i \leq n; i++)$
    a. $NewIndex = i + \xi$
    b. $|Diff| = (NewIndex) mod (36)$
    c. $R.Add(SC_i.replace(C_{Diff}))$
3. End for
4. Return R

Further, we have implemented four popular ML methods namely Support Vector Machine (SVM), CART [37], C4.5 [38], and Bays. The classifier is trained using randomized datasets. Then classifier's performance is measured. The aim is to know the effect of sanitization process. The accuracy of the classifiers for identifying the utility of sanitized data is given in fig. 4 (A) and fig 4(B) shows the error rate. The accuracy measures the correctness of classifiers and the error rate used for measuring misclassification. To show the performance a data without noise has also been used. Then the experiments are performed with noisy dataset based on modified random noise and simple random noise.

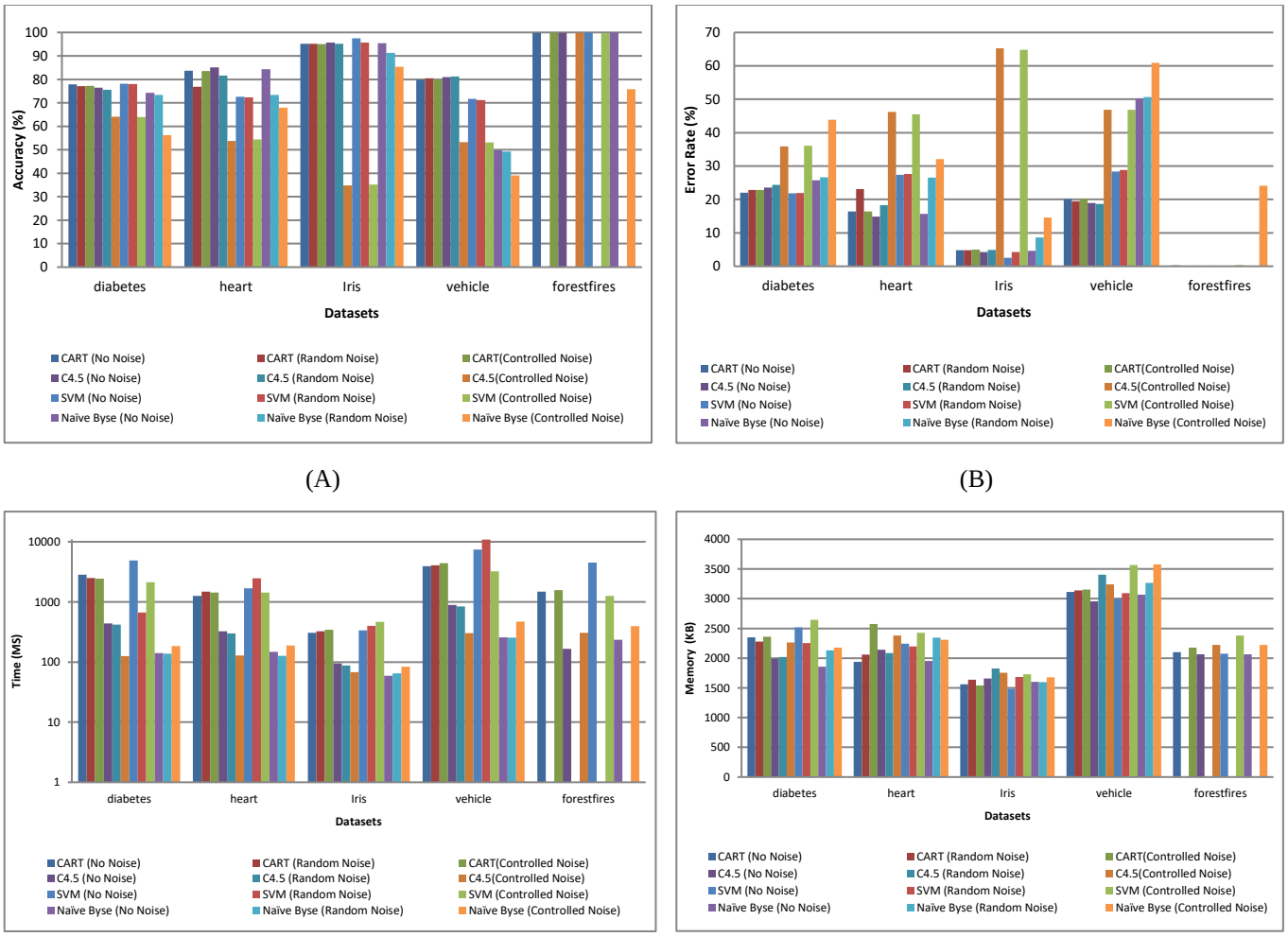


Fig. 4. Impact of Noise on Classifier's Performance (A) Accuracy (B) Error Rate (C) Time Consumed (D) Memory Usage

The results show the low variation in controlled noise data as compared to the traditional random noise data. Additionally, the random noise technique is highly fluctuating in terms of classification accuracy. Therefore, traditional random noise has significant influence on the classifier's performance. The time and memory requirements are demonstrated in Fig. 4(C) and 4(D). The results show the memory and time is not affected by type of noise. But when we use a mixed dataset (combined numerical and categorical) then the time and memory requirement is increasing. The observed values of experiments are given in table 8.

Table 8 Performance Of Classifiers For Noise Mixed Datasets

CART-N = CART (No Noise), CART-R = CART (Random Noise), CART-C = CART (Controlled Noise), C4.5-N = C4.5 (No Noise), C4.5-R = C4.5 (Random Noise), C4.5-C = C4.5(Controlled Noise), SVM-N = SVM (No Noise), SVR-R = SVM (Random Noise), SVM-C = SVM (Controlled Noise), Bays-N = Naïve Bays (No Noise), Bays – R= Naïve Bays (Random Noise), Bays-C = Naïve Bays (Controlled Noise)

Datasets	**CART-N**	**CART-R**	**CART-C**	**C4.5 - N**	**C4.5-R**	**C4.5-C**	**SVM-N**	**SVM-R**	**SVM – C**	**Bays-N**	**Bays - R**	**Bays - C**
Error rate												
diabetes	22.071	22.875	22.813	23.524	24.421	35.858	21.855	21.977	36.105	25.781	26.655	43.833
heart	16.369	23.12	16.42	14.834	18.312	46.241	27.417	27.622	45.524	15.652	26.598	32.072
Iris	4.792	4.792	5	4.271	4.896	65.208	2.5	4.271	64.792	4.584	8.646	14.587
vehicle	20.155	19.505	20.217	18.978	18.7	46.811	28.359	28.855	46.873	50.19	50.683	60.898
forest	0.239	0.0	0.179	0.209	0.0	0.209	0.06	0.0	0.329	0.12	0.0	24.12

fires												
Accuracy												
diabetes	77.929	77.125	77.187	76.476	75.579	64.142	78.145	78.023	63.895	74.219	73.345	56.167
heart	83.631	76.88	83.58	85.166	81.688	53.759	72.583	72.378	54.476	84.348	73.402	67.928
Iris	95.208	95.208	95	95.729	95.104	34.792	97.5	95.729	35.208	95.416	91.354	85.413
vehicle	79.845	80.495	79.783	81.022	81.3	53.189	71.641	71.145	53.127	49.81	49.317	39.102
forest fires	99.761	0.0	99.821	99.791	0.0	99.791	99.94	0.0	99.671	99.88	0.0	75.88
Time Consumed												
diabetes	2845.2	2485.4	2453.4	436.8	420.8	125	4913	666.8	2116.2	140.6	137.8	184.2
heart	1265.2	1475.4	1437.2	326.4	300.6	128.6	1682	2466.8	1438.4	146.8	126.8	188.8
Iris	305.8	325.6	342.6	95	87.2	68.2	337.6	400.6	467.8	59	64.6	83.8
vehicle	3921.6	4063.4	4408	887.8	839.4	303.6	7459.2	10867.2	3222.6	256.6	255.2	468.6
forest fires	1479.2	0.0	1572.6	165.2	0.0	305.8	4511.2	0.0	1257.4	234	0.0	395
Memory Usage												
diabetes	2351.8	2281.4	2359.6	1989.8	2017.4	2262	2517.6	2253.2	2643.4	1858	2131.8	2175.4
heart	1939	2062	2575.8	2142.6	2084.6	2382.8	2245.2	2198.4	2428.8	1951.4	2342.8	2308.4
Iris	1561.4	1637.2	1539	1657.2	1824.4	1752.2	1482	1682.6	1726.6	1601.8	1595.6	1678.2
vehicle	3114.4	3139.6	3152.6	2956.8	3406.2	3243.2	2996.8	3092.6	3569.6	3070.2	3268	3575.6
forest fires	2102.8	0.0	2176.6	2066	0.0	2221.6	2075.6	0.0	2379.4	2065.6	0.0	2222.8

4. Experimental Conclusion

In the last two experimental scenarios,

1. The data collaborating parties are combining data in vertical manner, and due to increasing number of parties the data extent are also rising. Here, we conclude that the data dimensions do not affect learning algorithm's classification accuracy. But, highly influence the computational complexity (i.e. time and memory).
2. The experiments are performed on noisy data. In this context, we have implemented four ML algorithms and the performance influence for original data and sanitized data is measured. According to the results, noise does not have influence on the memory and time but can damage the data utility. Additionally, Noise more than a specific limit can change the data utility completely. Thus, the proposed noise inclusion technique includes a limited noise on data to preserve the utility and classification performance.

5. An Enhanced PPDM Model

The PPDM models are utilizing three main components first parties who agreed to combine data. Second is the data aggregation technique and third is the data processing and knowledge discovery. In order to combine these components, we proposed functional model to demonstrate the entire functional aspects of PPDM in fig. 5.

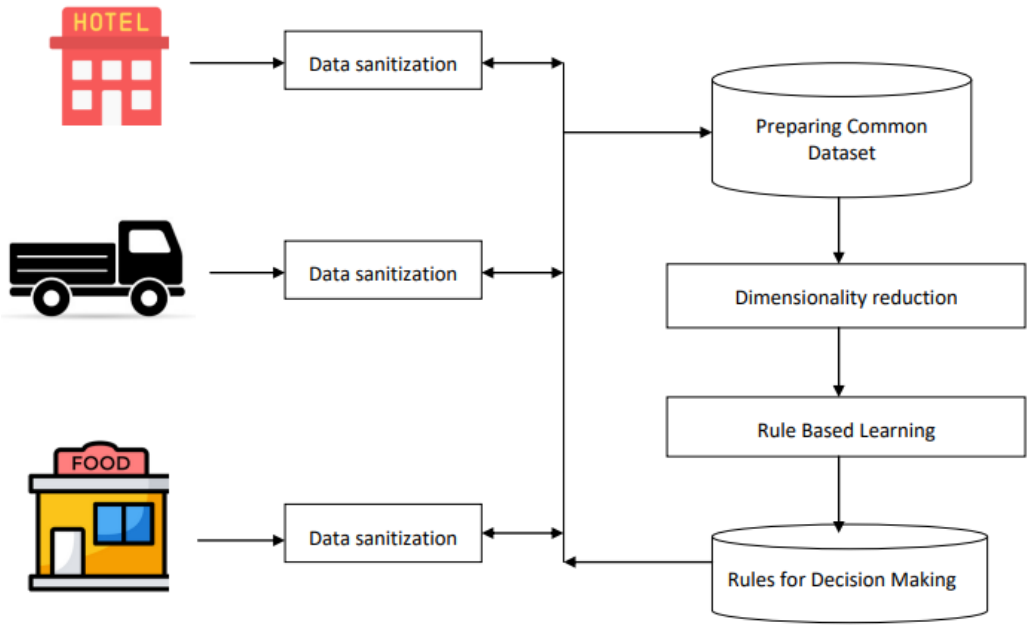


Fig. 5 Proposed Functional Model for PPDM

We initiate the process with different parties who are agreed to collaborate on the data. Therefore, we have demonstrated different industries, who want to combine their data for mining. But, due to privacy requirements, we have implemented the data sensitization process for each individual client's application. Here for sanitization of the data, we have applied the modified noise mixing algorithm. In addition, we have also used the classical random noise for comparative study. The noise inclusion algorithm implements a diverse level of noise according to the contribution of data. After sensitization, the different parties are communicating the data to a centralized server (Trusted Authority). Therefore, a combined database is prepared by using the collected data from the different data sources. This dataset consists of all the communicated data by the contributors. Now we apply the ML algorithm. Here, the performance of the ML algorithm can be fluctuating due to higher dimensions and noisy data. Thus, at the server end, we have implemented the feature selection technique based on the CC. The CC-based feature selection technique finds the rank of attributes that has higher significance with respect to the class labels. In this context, we have computed a threshold value. The threshold value will be used for selecting suitable features. The selected set of attributes is utilizable with the data mining algorithm. This data can be directly publishable for another research purpose. In this presented work, we demonstrated both kinds of data usage. Thus, next we have applied two decision tree algorithms to data. The aim of applying these algorithms is to measure the data utility after the data sanitization process. The consequences of mining can be useful in different application tasks. In future application designs, we implement data models, which are based on cross platforms and data sources. That may help to understand the demand and supply process and prepare future business plans. In addition, this model will also be used for other kinds of survey and marketing companies, research agencies, public service, banking and finance, and many more. Therefore, given architecture is useful for two kinds of applications, first where the combined data need to be used for common decision making, and second the use for research and academic study.

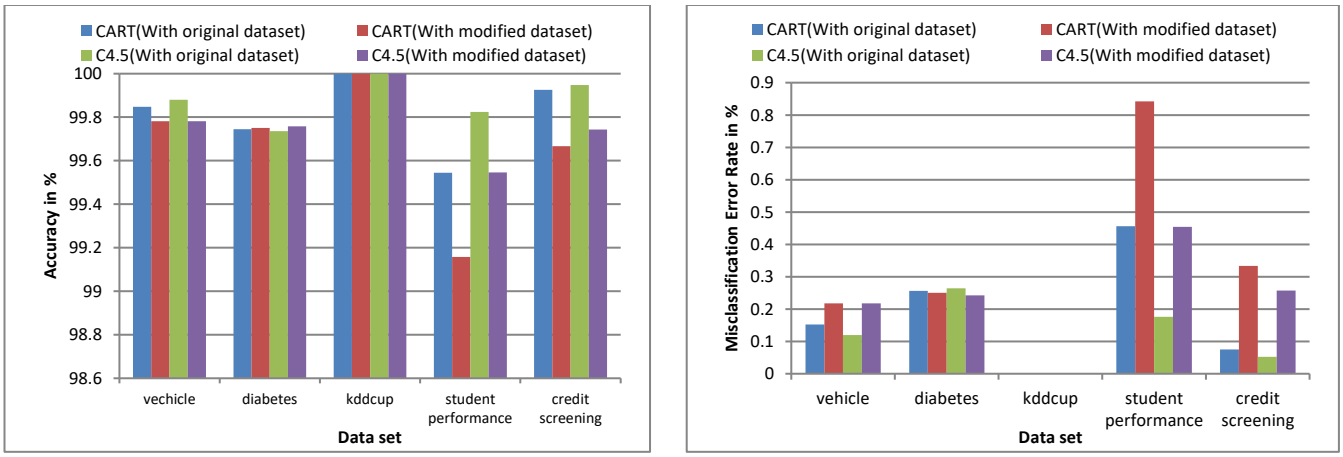
Advantages: the proposed work is providing an end-to-end secure and accurate PPDM framework. The proposed model includes the diversity in including noise in the contributed data. That makes it robust against attacks. Additionally, for dealing with dimensionality issue we also include the feature selection processes.

6. Experimental Results

The PPDM models can be evaluated using three kinds of parameters namely data quality, data processing, and privacy.

6.1. *Data Quality*

In this section, we are providing the results for the data quality matrix. Thus, we have measured accuracy, f-score, and misclassification error. The performance is demonstrated in fig. 6. In fig. 6(A) the accuracy of the model is explained, additionally, in fig. 6(B) we provide the error rate or misclassification rate.



(A) (B)

Fig. 6 Performance of the Proposed PPDM Model (A) Accuracy (B) Misclassification Rate

The performance in terms of accuracy and error rate demonstrates the correctness of the data pattern and reveals the utility of the data. According to the experimental performance, the proposed model demonstrates accurate results after the sanitization process. One more parameter which is equivalent to accuracy and error rate is known as the f-score or f-measures. That is also known as the harmonic mean of precision and recall. The f-score is also used to find the correctness of the algorithm. The obtained f-score of the model is given in figure 7. The Y axis demonstrates the obtained f-score between 0-1. Additionally, the observed values of accuracy, error rate and f1-score is given in table 9.

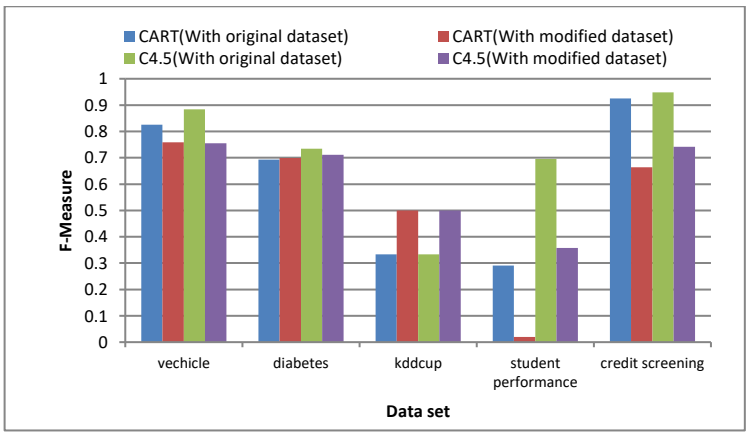


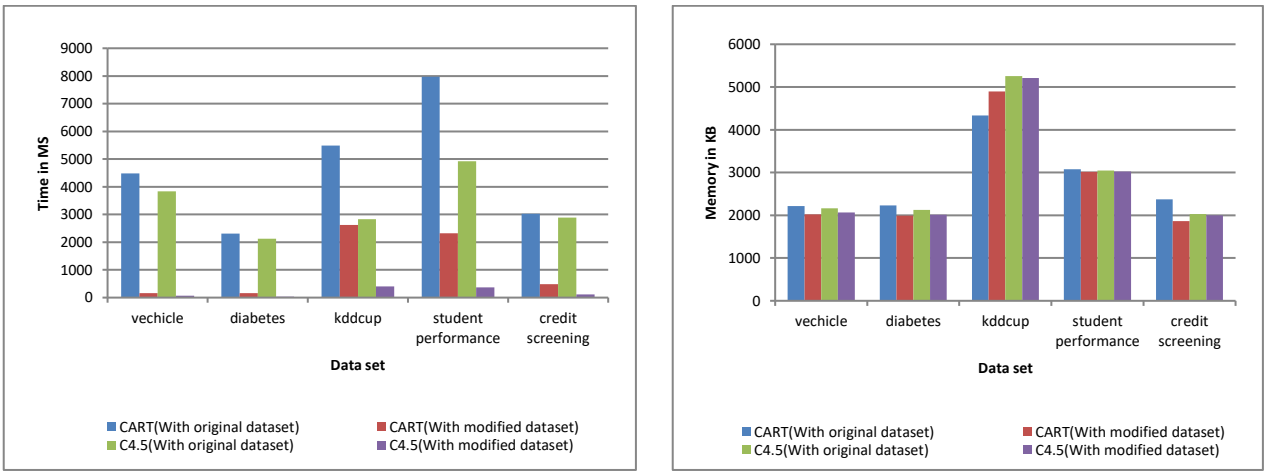
Fig. 7 Shows the F1-score obtained from sanitized data

Table 9 Performance of the Proposed PPDM Model

OD = With original dataset, MOD = With modified dataset												
Dataset	Accuracy				Misclassification Rate				F-Measures			
	CART		C4.5		CART		C4.5		CART		C4.5	
	OD	MOD	OD	MOD	OD	MOD	OD	MOD	OD	MOD	OD	MOD
D1	99.84	99.78	99.88	99.78	0.15	0.218	0.12	0.218	0.826	0.759	0.884	0.755
D2	99.74	99.75	99.73	99.75	0.25	0.25	0.26	0.242	0.693	0.699	0.735	0.711
D3	100	100	100	100	0	0	0	0	0.333	0.5	0.333	0.5
D4	99.54	99.15	99.82	99.54	0.45	0.842	0.17	0.454	0.291	0.02	0.696	0.358
D5	99.92	99.66	99.94	99.74	0.07	0.333	0.05	0.257	0.925	0.664	0.948	0.742
D1 = vehicle, D2 = diabetes, D3= kddcup, D4 = student performance, D5= credit screening												

6.2. *Processing Efficiency*

Data processing ability is measured in terms of memory and time. Table 10 and figure 8 shows the results of time in MS and memory usage in KB.



(A) (B)

Fig. 8. Shows (A) Time Consumption (B) Memory Usage

Table 10 Shows Performance in Time and Memory

Dataset	Time consumption				Memory Usages			
	CART		C4.5		CART		C4.5	
	OD	MOD	OD	MOD	OD	MOD	OD	MOD
D1	4477.33	158.667	3838.33	72.667	2215	2022.833	2165.167	2066.333
D2	2308.83	155.833	2121.5	33.833	2232.66	1992.833	2127.167	2011.333
D3	5491.83	2618.5	2831.33	400.5	4336.16	4896.5	5255.5	5208.833
D4	7968.33	2326.83	4919.33	374.333	3079	3015.83	3049.333	3023.833
D5	3024.66	483.667	2882.83	116.833	2370.16	1867.167	2029.167	1996.333

D1 = vehicle, D2 = diabetes, D3= kddcup, D4 = student performance, D5= credit screening

6.3. *Privacy and security*

In order to evaluate the PPDM models in terms of security prospective, there are three main metrics are used namely, Misses Cost (MC), Hidding failure and Artifactual Patterns (AP). The MC is used for computing the number of patterns, which are hidden incorrectly. In other words, the patterns are lost during privacy preservation. Let D is the database and D' is sanitized form of D. Thus the MC is given by:

$$MC = \frac{\#\sim R_P(D) - \#\sim R_P(D')}{!\sim R_P(D)} \quad (16)$$

Where $\#\sim R_P(X)$ is the number of patterns generated by data X. MC = 0% is needed for ideal cases, when MC=0 all patterns are non-sensitive in sanitized dataset.

Table 10 Performance In terms of Privacy and security

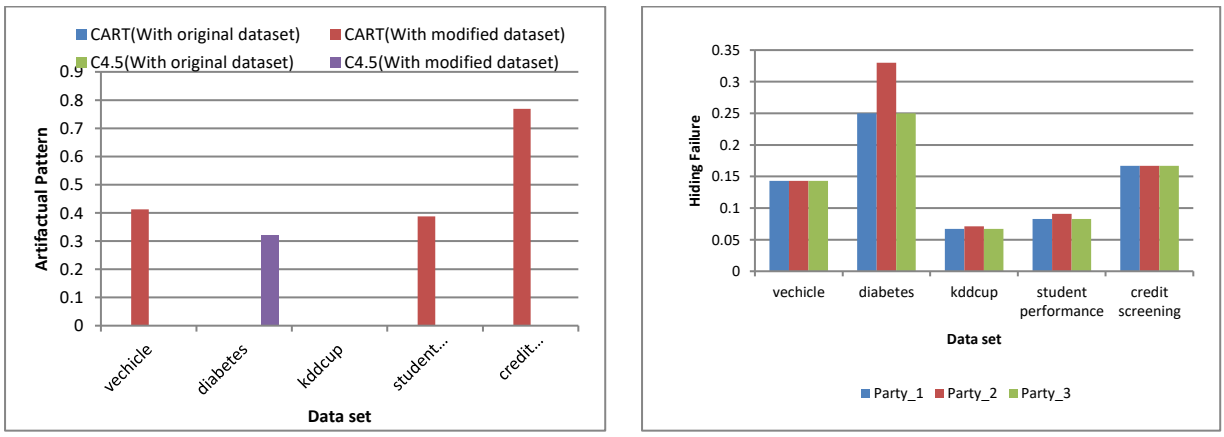
Dataset	Hiding Failure			Artifactual Pattern			
	Party_1	Party_2	Party_3	CART		C4.5	
				OD	MOD	OD	MOD
D1	0.143	0.143	0.143	0	0.413	0	0
D2	0.25	0.33	0.25	0	0	0	0.32
D3	0.067	0.071	0.067	0	0	0	0
D4	0.083	0.091	0.083	0	0.387	0	0
D5	0.167	0.167	0.167	0	0.769	0	0

D1 = vehicle, D2 = diabetes, D3= kddcup, D4 = student performance, D5= credit screening

The AP measures artifacts, or number of patterns which are not available in D, but created in the processed data D'. The AP can be defined by the following equation.

$$AP = \frac{|P'| - |P \cap P'|}{|P'|} \quad (17)$$

Where P is set of all patterns in D and P' is for D', and $|\cdot|$ shows the cardinality. The AP is becomes 0 for the best case, which demonstrate there are no new patterns are appeared during sanitization of data.



(A) (B)

Fig. 9 Describe Performance of the PPDM Model in Terms of (A) Artifactual Pattern (B) Hiding Failure

In our scenario, we have measured the performance using both the matrix. Here, we have found the misses cost of the proposed method is 0 for all the datasets and the numbers of parties. The reason is that the model is altering all the values of the input dataset thus all values are transformed. The result of AP is demonstrated in fig. 9(A). According to the results, the proposed model delivers a 0 AP score, but in some experiments, it increases slightly. Finally, we have measured the model's privacy level, in terms of hiding failure, which is demonstrated in fig. 9(B). The Hiding failure is a measurement of privacy, used for computing the balance between privacy and information. The hidden failure can be measured by the ratio of sensitive patterns that are hidden, and the sensitive information in original data:

$$HF = \frac{\#R_P(D')}{\#R_P(D)} \quad (18)$$

Where, HF is used to denote hidden failure, D' is sanitized data, and D is original data, and $\#R_P(\cdot)$ is amount of sensitive patterns.

In this context, If all sensitive patterns are hidden then HF = 0. In proposed concept we considered all the patterns are sensitive. Therefore by evaluation of the model with the different aspects of PPDM we have found the proposed model is acceptable and provides better balance between the data security and utility of data after sanitization process.

7. Conclusion and Future Work

The PPDM is a technique to privately process the data without disclosing sensitive information. However,

maintaining the balance between the PPDM consequences and privacy methods is the major issue. In addition, the collaboration of data increases the workload on the server. In this context, the proposed work provides a way to design an efficient and accurate PPDM framework for providing maximum data utility of sanitized data with minimum data and privacy loss. In this context, we have proposed a PPDM model which considers all the PPDM requirements. The entire efforts in designing the required PPDM model include the following fruitful observations.

1. **Data sanitization:** The sanitization of data includes the additional noise in datasets which will negatively impact the performance of the data analysis algorithm or data utility. Therefore we have introduced and implemented a controlled noise mixing algorithm that provides a balanced amount of noise for inclusion.
2. **Security threats:** The PPDM models are adopted to securely and privately analyze the multiple contributors' data. A similar type of noise inclusion may be a weak method against various attacks. Thus in order to reduce the risk of security, we sanitize the data at the contributor's end. That process manages the trust of the client in the system as well as the diverse data owner will introduce the diverse level of noise to prevent privacy loss.
3. **Data dimensionality:** The PPDM techniques will include multi-party data, which may increase the dimension of the dataset. The classical machine learning techniques are not much efficient to deal with the bulk amount of data. Additionally, that will negatively impact on the performance of the ML algorithm. Therefore we implement a lightweight feature selection algorithm, which reduces the dimensionality of the data.

By considering these three facts and the relevant solutions we have implemented a PPDM model. This model provides a better and more acceptable solution for all types of PPDM modeling. Additionally, that phenomenon has been proved using different performance parameters and experimental analysis. The domain of privacy-preserving data mining (PPDM) is very promising in new generation applications where machine learning needs privacy and security in their application in industrial as well as domestic applications. Therefore, the proposed can be extendable for the following directions.

1. The PPDM techniques can be lossy and lossless. The noise inclusion techniques are a lossy kind of process which may influence the performance of the application positively or negatively. Therefore we need to explore loss-less approaches too, for implementing the PPDM models
2. The PPDM models are mostly developed for demonstrating the rule-based models, thus in near future, we will also try to investigate the feasibility of PPDM models with the other possible kinds of learning algorithms.

References

[1] Mukhopadhyay, A.; Maulik, U.; Bandyopadhyay, S.; Coello, C. A. (2014) .*A Survey of Multi-objective Evolutionary Algorithms for Data Mining: Part I*. IEEE Transactions on Evolutionary Computation, Volume 18, No. 1, pp. 20-35.

[2] Aldeen, Y. A. A. S.; Salleh M.; Razzaque, M. A. (2015) .*A Comprehensive Review on Privacy Preserving Data Mining*. SpringerPlus 4:694, pp. 1-36.

[3] Danasana, J.; Kumar R.; Dey, D. (2012) .*Mining Association Rules For Horizontally Partitioned Databases Using CK Secure Sum Technique*. International Journal of Distributed and Parallel Systems, Volume 3, No.6, pp. 149-157.

[4] Ponce, J.; Karahoca, A. (2009). *Data Mining and Knowledge Discovery in Real Life Applications*. Published by In-Teh, First published, Printed in Croatia (book)

[5] Mendes, R.; Vilela, J. P. (2014). *Privacy-Preserving Data Mining: Methods, Metrics, and Applications*. IEEE, Vol. 5, 2169-3536.

[6] Xu, L.; Jiang, C.; Wang, J.; Yuan, J.; Ren, Y. (2014) .*Information Security in Big Data: Privacy and Data Mining*. IEEE, Volume 2, 2169-3536.

[7] Vasan, K. K.; Surendiran, B. (2016). *Dimensionality reduction using Principal Component Analysis for network intrusion detection*. Perspectives in Science, Elsevier, 8, 510—512.

[8] Nettleton, D. F.; Puig, A. O.; Fornells, A. (2010) .*A study of the effect of different types of noise on the precision of supervised learning techniques*. Springer Science+Business Media B.V., Artif Intell Rev, 33, 275–306.

[9] Anitha, P.; Krithka, G.; Choudhry, M. D. (2014). *Machine Learning Techniques for learning features of any kind of data: A Case Study*. International Journal of Advanced Research in Computer Engineering & Technology, Vol 3, Issue 12.

[10] Chitradevi, B.; Thinaharan, N. (2015). *Role of Decision Making in Data Mining Systems*. International Journal of Trend in Research and Development, Volume 2, 5.

[11] Swamy, S. K.; Manjula, S. H.; Venugopal, K. R.; Iyengar, S. S.; Patnaik, L. M. (2014). *Association Rule Sharing Model for Privacy Preservation and Collaborative Data Mining Efficiency*. IEEE Proceedings of RAECS UIET Panjab University Chandigarh.

[12] Basiri, A.; Amirian, P.; Winstanley, A.; Moore, T. (2017). *Making Tourist Guidance Systems more Intelligent, Adaptive and Personalised using Crowd Sourced Movement Data*. Journal of Ambient Intell Human Comput, Springer, Volume 9, pp. 413-427.

[13] Kou, G.; Peng, Y.; Shi, Y.; Chen, Z. (2007). *Privacy-Preserving Data Mining of Medical Data Using Data Separation–Based Techniques*. Data Science Journal, Volume 6, Supplement, pp. 429-434.

[14] Chen, C. L. P.; Zhang, C-Y. (2014). *Data-Intensive Applications, Challenges, Techniques and Technologies: A Survey on Big Data*. Information Sciences 275, Elsevier, pp. 314–347.

[15] Xu, C.; Tao, D.; Xu, C.; Rui, Y. (2014). *Large-Margin Weakly Supervised Dimensionality Reduction*. Proceedings of the 31 st International Conference on Machine Learning, Beijing, China, JMLR: W&CP, Volume 32, No 2, pp. 865-873.

[16] Bouzas, D.; Arvanitopoulos, N.; Tefas, A. (2015). *Graph Embedded Nonparametric Mutual Information For Supervised Dimensionality Reduction*. IEEE Transactions on Neural Networks and Learning Systems, Volume 26, No. 5, pp. 957-963.

[17] Sunitha, L.; Raju, M. B.; Srinivasa, B. S. (2013). *A Comparative Study between Noisy Data and Outlier Data in Data Mining*. International Journal of Current Engineering and Technology, Volume 3, No. 2, pp. 575-577.

[18] Xiong, H.; Pandey, G.; Steinbach, M.; Kumar, V. (2006). *Enhancing Data Analysis with Noise Removal*. IEEE Transactions on Knowledge and Data Engineering, Volume 18, Issue 3, pp. 304-319.

[19] Zhang, W.; Lin, Y.; Xiao, S.; Wu, J.; Zhou, S. (2015). *Privacy Preserving Ranked Multi-Keyword Search for Multiple Data Owners in Cloud Computing*. IEEE Transactions on Computers Journal of Latex Class Files, Vol. 6, No. 1, pp. 1-14.

[20] Dong, Y.; Du, B.; Zhang, L.; Zhang, L. (2017). *Dimensionality Reduction and Classification of Hyperspectral Images Using Ensemble Discriminative Local Metric Learning*. IEEE Transactions on Geoscience and Remote Sensing, 0196-2892.

[21] Lin, J. C. W.; Wu, J. M. T.; Viger, P. F.; Djenouri, Y.; Chen, C. H.; Zhang, Y. (2019). *A Sanitization Approach to Secure Shared Data in an IoT Environment*. IEEE, Vol 7, 2169-3536.

[22] Artoni, F.; Delorme, A.; Makeig, S. (2018). *Applying dimension reduction to EEG data by Principal Component Analysis reduces the quality of its subsequent Independent Component decomposition*. Neuroimage; 175: 176–187.

[23] Binol, H. (2018). *Ensemble Learning Based Multiple Kernel Principal Component Analysis for Dimensionality Reduction and Classification of Hyperspectral Imagery*. Hindawi Mathematical Problems in Engineering, Article ID 9632569, 14 pages.

[24] Abrahamsen, T. J.; Hansen, L. K. (2011). *A Cure for Variance Inflation in High Dimensional Kernel Principal Component Analysis*. Journal of Machine Learning Research, 12, 2027-2044.

[25] Ly, A.; Marsman, M.; Wagenmakers, E. J. (2018). *Analytic posteriors for Pearson's correlation coefficient*. Statistica Neerlandica, Vol. 72, nr. 1, pp. 4–13.

[26] Kumar, S.; Chong, I. (2018). *Correlation Analysis to Identify the Effective Data in Machine Learning: Prediction of Depressive Disorder and Emotion States*. Int. J. Environ. Res. Public Health, 15, 2907.

[27] Hira, Z. M.; Gillies, D. F. (2015). *A Review of Feature Selection and Feature Extraction Methods Applied on Microarray Data*. Advances in Bioinformatics, Article ID 198363.

[28] Sameen, M. I.; Pradhan, B.; Lee, S. (2018). *Self-Learning Random Forests Model for Mapping Groundwater Yield in Data-Scarce Areas*. Natural Resources Research.

[29] Ali, N.; Neagu, D.; Trundle, P. (2019). *Evaluation of k nearest neighbour classifer performance for heterogeneous data sets*. SN Applied Sciences, 1, 1559.

[30] Cao, Y.; Li, P.; Zhang, Y. (2018). *Parallel processing algorithm for railway signal fault diagnosis data based on cloud computing*. Future Generation Computer Systems, 88, 279–283.

[31] Gallego, S. R.; Krawczyk, B.; García, S.; Wozniak, M.; Herrera, F. (2017). *A survey on data preprocessing for data stream mining: Current status and future directions*. Neurocomputing, 239, 39–57.

[32] Pyo, J. S.; Seong, L. J.; Juyeon, L. (2020). *Method of improving the performance of public-private*

innovation networks by linking heterogeneous DBs: Prediction using ensemble and PPDM models. Technological Forecasting & Social Change, 161, 120258.

[33] Zhang, J.; Cormode, G.; Procopiuc, C. M.; Srivastava, D.; Xiao, X. (2017). *PrivBayes: Private Data Release via Bayesian Networks*. ACM Trans. Database Syst. 42, 4, Article 25, 41 pages.

[34] Vaghashia, H.; Ganatra, A. (2015). *A Survey: Privacy Preservation Techniques in Data Mining*. International Journal of Computer Applications, 0975 – 8887, Volume 119, No.4.

[35] https://archive.ics.uci.edu/ml/index.php

[36] Munkhdalai, L.; Munkhdalai, T.; Park, K. H.; Lee, H. G.; Li, M.; Ryu, K. H. (2019). *Mixture of Activation Functions With Extended Min-Max Normalization for Forex Market Prediction*. IEEE, VOLUME 7.

[37] Hasanipanah, M.; Faradonbeh, R. S.; Amnieh, H. B.; Armaghani, D. J.; Monjezi, M. (2016). *Forecasting blast induced ground vibration developing a CART model*. Engineering with Computers, Springer-Verlag London.

[38] Lohani, A.; Singh, J.; Lohani, A. (2016). *Comparative Analysis Of Classification Methods Using Privacy Preserving Data Mining*. International Journal of Recent Trends in Engineering & Research, Volume 02, Issue 04.

[39] Nasiri, N.;Keyvanpour, M. R. (2020).*Classification and Evaluation of Privacy Preserving Data Mining Methods.* 11th International Conference on Information and Knowledge Technology (IKT), 17-22.

[40] Cuzzocrea, A.;Leung, C. K.; Olawoyin, A. M.; Fadda, E. (2022).*Supporting Privacy-Preserving Big Data Analytics on Temporal Open Big Data.* Procedia Computer Science, vol. 198, 112-121.

[41] Sei, Y.;Ohsuga, A. (2022).Private True Data Mining: Differential Privacy Featuring Errors to Manage Internet-of-Things Data. in IEEE Access, doi: 10.1109/ACCESS.2022.3143813, vol. 10, 8738-8757.

[42] Mahmoudi, M. R.;Heydari, M. H.; Qasem, S. N.; Mosavi, A. (2021).*Principal Component Analysis To Study The Relations Between The Spread Rates Of COVID-19 In High Risks Countries*. Alexandria Engineering Journal, vol. 60, 457-464.

[43] Ahmed, U.;Srivastava, G.; Lin, J. C. W.; (2021).*A Machine Learning Model for Data Sanitization.* Computer Networks, vol. 189 107914, 107-914.

[44] Hewage, U. H. W. A.; Sinha, R; Naeem, M.A.; (2023).Privacy-preserving data (stream) mining techniques and their impact on data mining accuracy: a systematic literature review. Artif Intell Rev (2023). https://doi.org/10.1007/s10462-023-10425-3.

[45] Naresh, V. S.;Thamarai, M.; (2023).*Privacy-preserving data mining and machine learning in healthcare: Applications, challenges, and solutions.* WIREs Data Mining and Knowledge Discovery, 13(2), e1490. https://doi.org/10.1002/widm.1490.

[46] Yadav, R. ., & Singh, R. . (2023). A Hyper-parameter Tuning based Novel Model for Prediction of Software Maintainability. International Journal on Recent and Innovation Trends in Computing and Communication, 11(2), 106–113. https://doi.org/10.17762/ijritcc.v11i2.6134

[47] Anna, G., Jansen, M., Anna, J., Wagner, A., & Fischer, A. Machine Learning Applications for Quality Assurance in Engineering Education. Kuwait Journal of Machine Learning, 1(1). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/109