$$A_1 = 55 + \left\lfloor \frac{(\lfloor ^{-10}/_{2} \rfloor)+1}{2} \right\rfloor = 55 + \left\lfloor \frac{-5+1}{2} \right\rfloor = 55 + \frac{-4}{2} = 55 - 2 = 53 \quad (14)$$

$$A_2 = 55 - \left\lfloor \frac{(\lfloor ^{-10}/_{2} \rfloor)}{2} \right\rfloor = 55 - \left\lfloor \frac{-5}{2} \right\rfloor = 55 - \lfloor -2.5 \rfloor = 55 + 3 = 58 \quad (15)$$

3. Methodology

The proposed method is divided into two phases: Watermark Insertion and Watermark Extraction. Fig. 1 shows the various steps involved in watermark insertion phase and Fig. 2 depicts watermark extraction phase.

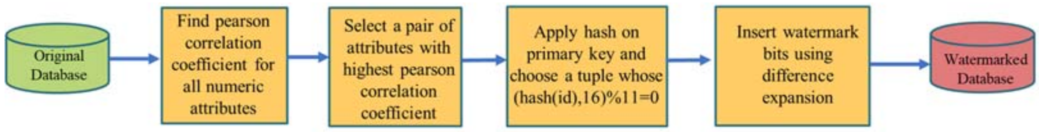

Fig. 1 Watermark Insertion

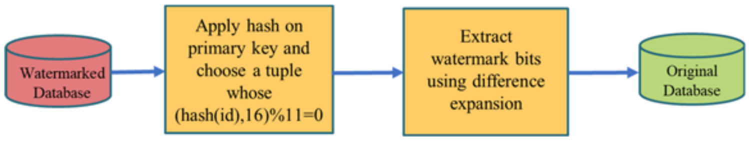

Fig. 2 Watermark Extraction

The proposed method has following features:

- Robustness: Watermark remains unaltered despite the modifications made to the database.
- Blindness: Watermark extraction does not require the knowledge of original database as well as watermark information.
- Reversible: The original database can be recovered after extracting the watermark from the watermarked database.
- Distortion control: The selection of a pair of attributes with highest PCC ensures lower distortion.

3.1. *Watermark Insertion*

During this phase, watermark bits are inserted into specific positions in the database. These bits can later be extracted to prove database ownership. The following steps are performed to insert watermark bits into the original database:

Step 1: Find Pearson Correlation Coefficient (PCC) for all the numeric attributes using Eq. (1)
Step 2: Select a pair of attributes whose Pearson Correlation Coefficient (PCC) is highest among all
Step 3: Apply hash on the primary key and choose a tuple whose (hash(id),16) % 11=0
Step 4: Insert the watermark bits in selected tuples using difference expansion
In this step, we calculate the average and difference between a pair of attributes that are selected in step 2 as follows:

$$avg = \left\lfloor \frac{A_1 + A_2}{2} \right\rfloor \quad (16)$$

$$diff = A_1 - A_2 \quad (17)$$

Determine the watermark bit to be inserted using Eq. (18)

$$b = diff \;\&\; 1 \quad (18)$$

Compute the new difference *diff'* as shown in Eq. (19) by assuming the watermark bit *b* to be inserted.

$$diff' = 2 * diff + b \quad (19)$$

Modified attribute values denoted by A_1' and A_2' are computed as in Eq. (20) and Eq. (21)

$$A_1' = avg + \left\lfloor \frac{diff' + 1}{2} \right\rfloor \quad (20)$$

$$A_2' = avg - \left\lfloor \frac{diff'}{2} \right\rfloor \quad (21)$$

The input of this phase is the original database, and the output is the watermarked database.

3.2. *Watermark Extraction*

During watermark extraction phase, the same pair of attributes that were selected in step 2 of watermark insertion phase are considered. The following steps are performed to extract watermark bits from the watermarked database:
Step 1: Apply hash on the primary key and choose a tuple whose (hash(id),16) % 11=0
Step 2: Extract the watermark bits from selected tuples using difference expansion
In this step, we calculate the average and difference as shown in Eq. (22) and Eq. (23)

$$avg' = \left\lfloor \frac{A_1' + A_2'}{2} \right\rfloor \quad (22)$$

$$diff' = A_1' - A_2' \quad (23)$$

The watermark bit *b* can be extracted using in Eq. (24)

$$b = diff' - 2 * \left\lfloor {diff'}/{2} \right\rfloor \quad (24)$$

Original attribute values denoted by A_1 and A_2 are recovered with the help of Eq. (25) and Eq. (26)

$$A_1 = avg' + \left\lfloor \frac{\left(\left\lfloor {diff'}/{2} \right\rfloor\right) + 1}{2} \right\rfloor \quad (25)$$

$$A_2 = avg' - \left\lfloor \frac{\left\lfloor {diff'}/{2} \right\rfloor}{2} \right\rfloor \quad (26)$$

The input of this phase is the watermarked database, and the original database is recovered as a result.

4. Results and Discussion

Experiments were conducted on the system with Intel Core i3 CPU of 2.00GHz and 4GB RAM. Indian Liver Patient Dataset with 583 instances is the test database. The dataset contains 10 variables that are Age, Gender, Total Bilirubin, Direct Bilirubin, Alkaline Phosphatase, Alamine Aminotransferase, Aspartate Aminotransferase, Total Proteins, Albumin, Albumin and Globulin Ratio. An attribute ID has been added as primary key with values ranging from 1 to 583. Gender is a categorical attribute, so it is not considered for calculating Pearson Correlation Coefficient (PCC).

For simplicity, the attributes ID, Age, Total Bilirubin, Direct Bilirubin, Alkaline Phosphatase, Alamine Aminotransferase, Aspartate Aminotransferase, Total Proteins, Albumin, Albumin and Globulin Ratio are renamed as A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9, and A_{10}. The Pearson Correlation Coefficient (PCC) for all the numeric values are as shown in Table 1. The attributes A_6 and A_7 i.e., Alamine Aminotransferase and Aspartate Aminotransferase have the highest PCC value 0.791966 as highlighted Table 1. Higher the PCC, lower the distortion. 10% of the total tuples i.e., 59 out of 583, are watermarked.

	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}
A_1	1									
A_2	-0.052385	1								
A_3	0.1015259	0.011763	1							
A_4	0.1362854	0.007529	0.874618	1						
A_5	-0.078805	0.080425	0.206669	0.234939	1					
A_6	-0.12486	-0.08688	0.214065	0.233894	0.125680	1				
A_7	-0.094106	-0.01991	0.237831	0.257544	0.167196	**0.791966**	1			
A_8	0.189634	-0.18746	-0.0081	-0.000139	-0.028514	-0.042518	-0.025645	1		
A_9	0.0656466	-0.26592	-0.22225	-0.228531	-0.165453	-0.029742	-0.085290	0.784053	1	
A_{10}	-0.024123	-0.21853	-0.20595	-0.199745	-0.235087	-0.002821	-0.070033	0.239690	0.691239	1

Table 1. Pearson Correlation Coefficient (PCC) for all numeric attributes

4.1. *Robustness Analysis*

Robustness can be demonstrated with the help of different types of attacks on the database. We consider three types of attacks: insertion, deletion, and modification. Suppose an attacker attempts to insert, delete, or modify the tuples in the database. Experiments are conducted to simulate all these attacks with attack rate of 10%, 20%, up to 90%. Fig. 3 represents attack ratio on X-axis and watermark detection rate on Y-axis.

The first experiment is performed to demonstrate insertion attack. We compute the hash function of the primary key and based on its value the tuples are selected for watermarking. Whenever any new tuple is added into the database, it will not tamper the existing watermark. As the watermark is determined by the primary key, it will remain intact even though the attacker tries to insert any number of new tuples. Thus, the proposed method is robust to insertion attack as shown in Fig. 3

The second experiment is conducted to determine the robustness of the proposed method under deletion attack. In this type of attack, the attacker attempts to randomly delete tuples from the database. As we increase the attack rate from 10% to 90% by deleting 59 tuples to 525 tuples respectively, the watermark detection rate decreases from 94.91% to 10.16%. It means that the chances of watermark detection go on decreasing as we increase the deletion attack percentage. It can be observed in Fig. 3. Moreover, it is not possible to recover the complete watermark if many tuples that contain watermark information are deleted.

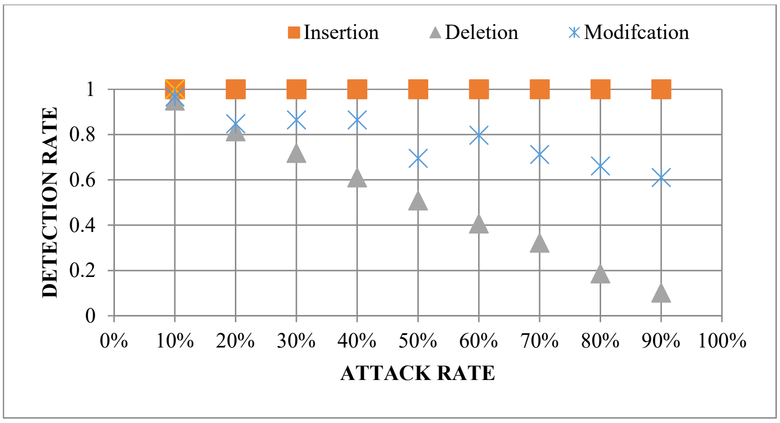

Fig. 3 Watermark Detection Rate

The third experiment is performed to demonstrate modification attack. In modification attack, the attacker attempts to randomly modify tuples in the database. Fig. 3 shows that the increase in attack rate from 10% to 90% leads to detection rate of 96.61% to 61.01% respectively. It means that there are high chances of watermark detection even with greater attack rates. From Fig. 3, it can be observed that around 60% of the watermark can be recovered even in case of 90% modification attack.

4.2. *Statistical Distortion*

The data quality of the proposed algorithm is evaluated with statistical distortion. We measure the statistical distortion through mean absolute error and variations of mean and standard deviation between the attributes before and after watermark insertion.

4.2.1 Mean Absolute Error (MAE)

Mean Absolute Error (MAE) can be calculated as:

$$MAE = \frac{\sum_{i=1}^{n} |A_i - A_i^w|}{n} \qquad (27)$$

where, n is the total number of tuples in the database, A_i is the attribute of original database and A_i^w is the attribute of watermarked database.

Attribute name	MAE
Alamine Aminotransferase (A_6)	1.63
Aspartate Aminotransferase (A_7)	1.63

Table 2. Mean Absolute Error (MAE)

Table 2 shows the mean absolute error for the selected attributes is 1.63. It will be same for the pair of selected attributes because of difference expansion.

4.2.2 Mean and Standard Deviation

Table 3 provides the mean and standard deviation obtained for the selected attributes Alamine Aminotransferase and Aspartate Aminotransferase. These measures are computed for the original as well as the watermarked database as seen in Table 3.

Attribute name	**Original Database**		**Watermarked Database**	
	Mean	**Std**	**Mean**	**Std**
Alamine Aminotransferase (A_6)	80.71	182.62	82.28	144.35
Aspartate Aminotransferase (A_7)	109.91	288.91	114.43	247.78

Table 3. Mean and Standard Deviation

To see the change of mean and standard deviation, the difference in mean and the difference in standard deviation for the watermarked attributes are calculated as in Eq. (28) and Eq. (29):

$$Difference\ in\ mean = |Mean_{Db} - Mean_{WDb}| \tag{28}$$

$$Difference\ in\ standard\ deviation = |Std_{Db} - Std_{Wdb}| \tag{29}$$

where, MeanDb and StdDb represent the mean and standard deviation of the original database. MeanWDb and StdWDb represent mean and standard deviation of the watermarked database.

Attribute name	**Proposed method**	
	Difference in mean	**Difference in std**
Alamine Aminotransferase (A_6)	1.57	38.27
Aspartate Aminotransferase (A_7)	4.52	41.13

Table 4. Difference in mean and difference in standard deviation

As shown in Table 4, the proposed method introduces minor change in mean of the selected attributes in original and watermarked database.

5. Conclusion

In this paper, a reversible and blind watermarking technique has been proposed for numeric relational databases. We suggested the use of Pearson Correlation Coefficient to select highly correlated attributes for applying difference expansion. Due to this, the distortion introduced in the database has a negligible effect on data quality. This is supported statistically by the difference in mean of 1.57 and 4.52 for attributes Alamine Aminotransferase (A6) and Aspartate Aminotransferase (A7) respectively. Difference in standard deviation is comparatively higher, that is, 38.47 and 41.13 for attributes A6 and A7 respectively. Experimental results show that the proposed method is robust against insertion attacks. Irrespective of the number of tuples added into the database, the watermark can still be completely recovered. Results show that the proposed method can recover around 60% of the watermark with 90% of modification attack rate. In case of deletion attack, the watermark detection rate decreases from 94.91% to 10.16% as the attack rate increases from 10% to 90%. Our future work is to develop a reversible technique that will reduce distortion and increase watermark detection rate even with higher attack rates.

References

[1] Agarwal, R., & Kiernan, J. (2002). Watermarking Relational Databases. *28th VLDB Conference*, (pp. 155-166).

[2] Chang, C.-C., Nguyen, T.-S., & Lin, C.-C. (2013). A Blind Reversible Robust Watermarking Scheme for Relational Databases. *The Scientific World Journal*, 1-12.

[3] Farfoura, M., Horng, S.-J., Lai, J.-L., Run, R.-S., Chen, R.-J., & Khan, M. (2012). A blind reversible method for watermarking relational databases based on a time-stamping protocol. *Expert Systems with Applications*, 3185-3196.

[4] Franco-Contreras, J., Coatrieux, G., Cuppens, F., Cuppens-Boulahia, N., & Roux, C. (2014). Robust Lossless Watermarking of Relational Databases Based on Circular Histogram Modulation. *IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY*, 397-410.

[5] Ge, C., Sun , J., Sun, Y., Di, Y., Zhu, Y., Xie, L., & Zhang, Y. (2020). Reversible Database Watermarking Based on Random Forest and Genetic Algorithm. *2020 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC).* Chongqing, China: IEEE.

[6] Gupta, G., & Pieprzyk, J. (2009). Reversible and Blind Database Watermarking Using Difference Expansion. *International Journal of Digital Crime and Forensics, 12*, 1-13.

[7] Hu, D., Zhao, D., & Zheng, S. (2019). A New Robust Approach for Reversible Database Watermarking with Distortion Control. *IEEE Transactions on Knowledge and Data Engineering*, 1024 - 1037.

[8] Iftikhar, S., Kamran, M., & Anwar, Z. (2015). RRW—A Robust and Reversible Watermarking Technique for Relational Data. *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*, 1132-1145.

[9] Imamoglu, M., Ulutas, M., & Ulutas, G. (2017). A New Reversible Database Watermarking Approach with Firefly Optimization Algorithm. *Mathematical PRoblems in Engineering*, 1-14.

[10] Jawad, K., & Khan, A. (2013). Genetic algorithm and difference expansion based reversible watermarking for relational databases. *Journal of Systems and Software* , 2742-2753.

[11] Khanduja, V., Verma, O., & Chakraverty, S. (2015). Watermarking relational databases using bacterial foraging algorithm. *Multimedia Tools and Applications*, 813-839.

[12] Li, Y., Wang, J., & Luo, X. (2020). A reversible database watermarking method non-redundancy shifting-based histogram gaps. *International Journal of Distributed Sensor Networks*, 1-11.

[13] Unnikrishnan, K., & Pramod, K. (2021). Prediction-based robust blind reversible watermarking for relational databases. *International Journal of Information and Computer Security*.
[14] Zhang, Y., Yang , B., & Niu, X.-M. (2006). Reversible Watermarking for Relational Database Authentication . *Journal of Computers*, 59-66.
[15] Zhao, M., Jiang, C., & Duan, J. (2019). Reversible Database Watermarking Based on Differential Evolution Algorithm. *2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM).* Dublin, Ireland: IEEE.

Authors Profile

Seema Siledar received M.E. degree in Computer Science and Engineering from Government College of Engineering, Aurangabad. Currently, she is pursuing her Ph.D. from Dr. Babasaheb Marathwada University. She is an Assistant Professor in Computer Science and Engineering department at Marathwada Institute of Technology. Her research interests include privacy and security in database.

Dr. Sharvari Tamane received Ph.D. degree in Computer Science and Engineering from Dr. Babasaheb Marathwada University. Currently, she is working as Professor and Head in University Department of Information and Communication Technology at M.G.M. University. Her research interests include Big Data, Cloud Computing, Network Security.