Available online at www.ijournalse.org

Emerging Science Journal

(ISSN: 2610-9182)

Vol. 6, No. 6, December, 2022

Data Mining Applications in Banking Sector While Preserving Customer Privacy

Özge Doğuç [1*]

[1] *Department of Management Information Systems, Istanbul Medipol University, Istanbul, Turkey.*

Abstract

In real-life data mining applications, organizations cooperate by using each other's data on the same data mining task for more accurate results, although they may have different security and privacy concerns. Privacy-preserving data mining (PPDM) practices involve rules and techniques that allow parties to collaborate on data mining applications while keeping their data private. The objective of this paper is to present a number of PPDM protocols and show how PPDM can be used in data mining applications in the banking sector. For this purpose, the paper discusses homomorphic cryptosystems and secure multiparty computing. Supported by experimental analysis, the paper demonstrates that data mining tasks such as clustering and Bayesian networks (association rules) that are commonly used in the banking sector can be efficiently and securely performed. This is the first study that combines PPDM protocols with applications for banking data mining.

Keywords:
Data Management;
Data Security; Data Mining;
Banking Processes.

Article History:

Received:	10	July	2022
Revised:	17	September	2022
Accepted:	23	September	2022
Available online:	28	September	2022

1- Introduction

Data mining is the technique of searching large volumes of data and specific data patterns electronically and developing data models based on them. Significant amounts of data are required to achieve the best results in the models created, and often the data is distributed among different parties with various security and privacy concerns. However, in many data mining applications, it is not possible for parties to share the original datasets with each other due to privacy concerns. To solve this problem, data mining protocols that protect data privacy have been proposed. The privacy-preserving data mining (PPDM) protocols introduce rules and techniques that allow parties to collaborate on data mining applications while keeping their data private. This study discusses the building blocks for PPDM protocols and shows how PPDM can be applied in the banking domain.

In real-life applications, organizations cooperate by sharing data for the same data mining task to improve the accuracy of the results. However, these organizations may have different security and privacy concerns. For example, different credit card companies may need to combine their datasets to build a better credit card fraud detection system; however, they may not be willing to share their datasets directly as this may reveal valuable competitive information and conflict with customers' privacy expectations. In another scenario, different states that cooperate to discover data patterns for terrorist detection may not trust each other completely and as any of the participants could cheat to gain some advantage. In both scenarios, the participants have different concerns about adversarial behavior. To address this problem and protect cooperating parties from adversaries, PPDM is defined to help them fulfill their privacy requirements and securely cooperate on data mining tasks.

* **CONTACT**: oduguc@medipol.edu.tr

DOI: http://dx.doi.org/10.28991/ESJ-2022-06-06-014

Many PPDM protocols have been suggested in the literature [1-3]. In these studies, the PPDM protocols are defined to be distributed and carried out by multiple communicating parties. Studies in the literature follow one of two alternative methods to provide security: 1- In some studies, data for PPDM applications is randomly perturbed while preserving underlying probabilistic properties [4-6]. This approach aims to preserve data privacy by adding random "noise" without losing the patterns the data may contain. 2- Alternatively, cryptographic techniques are used to preserve data privacy [3, 7, 8]. These techniques provide ways of securely executing data mining protocols using encryption.

Depending on the privacy concerns of the parties, PPDM protocols are designed through one of two security models: the semi-honest or the malicious model. The semi-honest model assumes that both parties follow the protocol without any deviations, but at the same time, the parties may try to extract information from the data that is available to them during the operation of the protocol. On the other hand, in the malicious model, one party can deviate from the protocol or change the rules in their favor to gain more information.

This paper discusses how homomorphic cryptosystems and secure multi-party computing can be used to perform common data mining tasks that are regularly performed by banks for customer classification and risk assessment. Bayesian network construction (i.e., association rule mining) and k-means clustering are shown as examples of PPDM applications in the domain. Multiple banks can securely share data while following the instructions given for each application and receive results of the data mining tasks. This paper also presents experimental analysis of performance and accuracy of the secure protocols.

2- Material and Methods

Yao introduced the protocol for making secure transactions between two parties with different security concerns. He has shown that in this scenario, two parties that do not trust each other can establish a result with a joint work by using Boolean circuits. Such scenarios are called as secure multi-party computing (SMC). Nearly two decades after the SMC concept was introduced, Malkhi et al. [9] developed a programming interface to Yao's solution; and they named their system Fairplay. The Fairplay system allows two parties to create and safely operate Boolean circuits for any computable F function. The Fairplay system provides a C-style functional programming language called Secure Function Definition Language (SFDL), so that parties can efficiently create the required Boolean circuits. Using Fairplay, parties build identical Boolean circuits, and then communicate via predefined ports. The next section provides details about the Fairplay system.

2-1- The Fairplay System

According to Yao's setup of SMC, both parties create Boolean circuits that are designed to evaluate the public function F. During circuit evaluation, one of the parties (Bob) converts his copy of the Boolean circuit into a garbled circuit by using his private input. Then the other party (Alice) evaluates this garbled circuit by providing her private input. In the end, both parties acquire the function output.

A secure evaluation of a function F using Fairplay consists of several rounds. In each round, the parties are required to accomplish certain tasks either by going offline or by communicating with the other party. These rounds can be categorized into two routines: circuit creation and circuit evaluation. The Fairplay system provides the parties with a compiler, which creates the Boolean circuit of any declared function by using specialized SFDL. With this system, the burden of creating complex Boolean circuits is reduced and it ensures that the resulting circuits are oblivious to violating the security requirements.

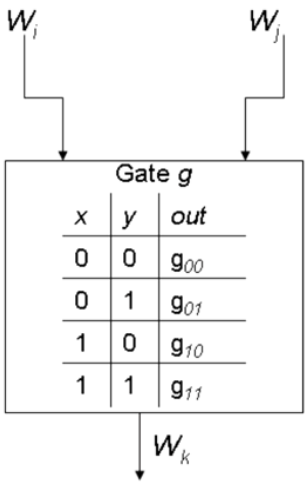

Figure 1. A gate in the garbled circuit

While the Fairplay system provides an easy and simple way for creating SMC applications, it is inefficient for large inputs [9]. Therefore, new methods are needed to avoid secure circuit evaluation and to improve performance of SMC applications. As an alternative to Fairplay, efficient homomorphic cryptosystems can be used to securely perform several algebraic operations. In the rest of this paper, homomorphic cryptosystems, and their applications in PPDM are discussed.

2-2- Secure Two-Party Computation

Secure two-party computation is the simplest case of SMC, where two parties jointly and securely compute a public function F. As shown in Figure 2 two parties jointly define a public function F and provide their private inputs. In the end they receive the function output without revealing their inputs to the other party.

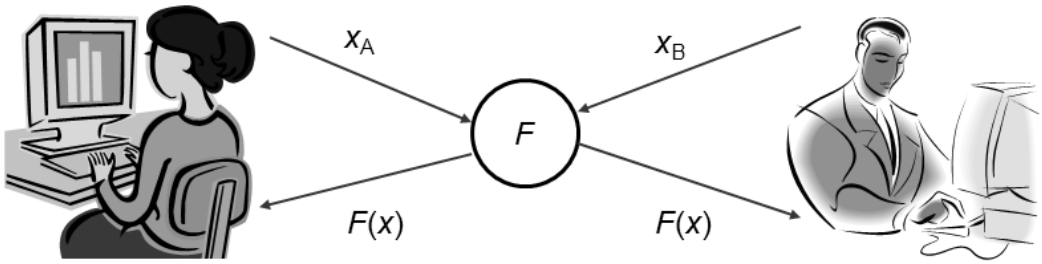

Figure 2. Secure two-party computation

Yao introduced the concept of secure two-party computation with the millionaires' problem. In this problem, Alice and Bob are two millionaires who wish to find out who is richer without revealing the actual amount of their fortunes. In his study Yao also provides a specific solution to the millionaires' problem [10] as defined for Alice and Bob who have i and j millions and wish to learn whether $i < j$. In this protocol Yao limits $1 < i, j < 10$ for the sake of simplicity without sacrificing the correctness of the protocol.

2-3- Paillier's Homomorphic Cryptosystem

Homomorphic cryptosystems find applications in PPDM for their support for basic arithmetic operations such as addition, multiplication and exponentiation using encrypted data. Homomorphic Encryption (HE) is a special encryption mechanism that allows operations to be done on encrypted data, without requiring to decrypt it at any step. Also, the result of a homomorphic operation is simply the encryption of the actual result; so only the key holder can decrypt and access it.

2-4- Homomorphic Addition

As an efficient alternative to circuit evaluation, many studies use additively homomorphic cryptosystems for arithmetic operations [11, 12]. Paillier (1999) introduced an efficient additively homomorphic cryptosystem [13] which is based on the composite residuality assumption. To emphasize, addition of two ciphertexts returns E_{pk} (m_1 + m_2 mod n). Thus, when the result is decrypted, it is ensured that the results will still be in the range of 0 to n. One of the simplest features of homomorphic cryptosystems is to allow computing addition of two numbers, by using their encryptions.

Paillier's homomorphic cryptosystem can compute the addition of two numbers by multiplying their encryption. The result of this operation is the encryption of the sum of the numbers. Let D_{pr} and E_{pk} denote homomorphic decryption and encryption function using private and public keys pr and pk respectively. Paillier's cryptosystem provides additive homomorphism as follows:

$$E_{pk}(m) \cdot E_{pk}(n) = E_{pk}(m + n) \rightarrow D_{pr}(E_{pk}(m + n)) = m + n \quad (1)$$

This requirement provides another useful property:

- Given a constant k and the encryption of m with public key pk, $E_{pk}(m)$, Paillier's homomorphic cryptosystem can compute the public key encryption of km, shown as $E_{pk}(km) := k \times_h E_{pk}(m)$, where $\times_h$ represents homomorphic multiplication.

Similarly, the result of the multiplication is $E_{pk}(k \cdot m_1 \bmod n)$.

2-5- Data Partitioning

In PPDM protocols defined in the literature, data is usually distributed between two or more parties. Two different assumptions have been used about how data is distributed between the parties. In the first assumption data is distributed horizontally, where parties have the same set of data for different entities. For example, different banks store the same attributes of millions of individuals. In this case, the banks have the same types of data about the individuals (name-

surname, age, income, etc.). On the other hand, in a vertically partitioned dataset parties have different types of information about the same entity set. For example, information about the same individual may exist in both university and hospital databases; however, while there is information about the education status of the person at the university, information about the health status of the same person is found at the hospital. As a third alternative, the data may be arbitrarily distributed among the parties. Figure 3 shows the different partitioning types.

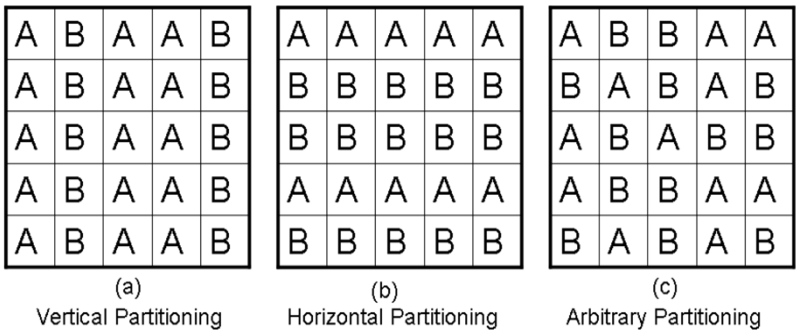

Figure 3. Types of partitioning

2-6- Bayesian Belief Networks

A Bayesian Belief Network – or simply Bayesian Network (BN) – is an important tool for illustrating common probabilities and causal relationships between variables. BN is a directed acyclic graph, where variables are represented as nodes, and arcs represent causal relationships between these variables. The degrees of relationships between variables are represented by associating probability values with arcs between variables. Variables modelled in a BN can be of any type and include any probability distribution. Figure 4 shows an example of an eight-node BN.

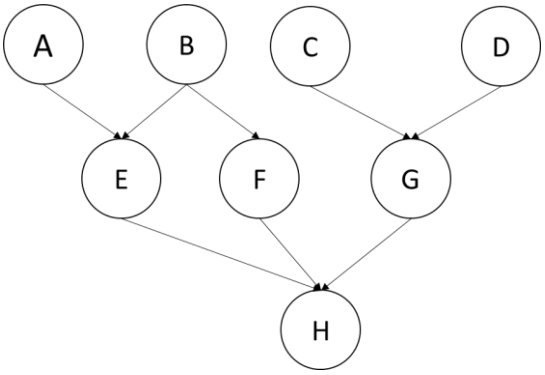

Figure 4. Sample Bayesian network

An arc between two nodes in the BN structure represents a dependency between variables and creates a child and parent relationship where the dependent variable is a child. A child node can have more than one parent, and the probability of a child node depends on its parents. The degrees of relationships between a child and its parents are represented in a conditional probability table (CPT) for the child node. BNs are created by uncovering relationships in the data and obtaining a probabilistic expression of each relationship. Computing common probabilities between nodes (variables) with the help of graphical representation and CPTs reduces the amount of computational work by reducing the amount of storage required to evaluate all variables. However, large BNs may have many parents per node, so the compute resources required to compute and store each node's CPT may be insufficient. Therefore, in many applications there are limitations on the number of parents a node can have [14-16].

2-7- Adversary Models

2-7-1- The Semi-Honest Model

Security properties of a protocol can be better understood when information about the adversarial environment is provided. In Yao's millionaires' problem, Alice and Bob are two millionaires who wish to find out who is richer without revealing the actual amounts of their fortunes. In the protocol for the millionaires' problem, if both parties follow the proposed protocol without any deviation, they are ensured to learn the correct result [10] and the parties cannot derive additional information about the other party. The semi-honest model assumes that the parties would follow the protocol without any deviation to learn the correct function output. On the other hand, the protocol output may change depending on the behavior of the parties.

2-7-2- The Malicious Model

The security definition for SMC does not rule out the case that parties can manipulate the protocol output by cheating. For example, Bob can change the output of the protocol by sharing incorrect information about his data. This type of cheating is out of the scope of the security assumptions of SMC, and thus cannot be prohibited in the security definitions that were discussed earlier. Instead, the focus should be on the other forms of cheating, such as inconsistent behavior or properly choosing random numbers. For example, Alice can successfully cheat if she does not behave consistently with her value i and Bob cannot detect it [10]. Preventing successful cheating by other party requires additional security mechanisms such as zero-knowledge proofs [2], commodity servers [17, 18], etc. These mechanisms are very complicated in their nature, and thus definition of a secure protocol against malicious behavior of the participants is expected to be more complex than the same protocol that is secure only against the semi-honest behavior. In the applications that are discussed in this paper, the focus is on the two-party cases, and only one and the same party is assumed to be malicious during the execution of the protocol.

3- Results and Discussion

Heuristic data mining applications such as clustering, association rule mining, decision trees are commonly used in banking and finance for risk scoring and churn analysis [19, 20] Data mining applications provide more accurate results when more data is available for the same constituents [21, 22]. Organizations often collaboratively use data mining applications while sharing data with each other. There are studies in the literature that show how data mining applications can be performed with vertically or horizontally partitioned data between several parties (organizations) [23–25].

Banking data often contains sensitive information about the customers and the banking industry is heavily regulated in many countries. Privacy-preserving data mining (PPDM) offers collaborative solutions where parties can privately share data while performing the same data mining task. There are two security models in the PPDM domain as discussed earlier: The semi-honest model and the malicious model.

Below are detailed explanations of how two banks can cooperate on several data mining applications in the semi-honest model. Data is assumed to be partitioned vertically between the banks, where different attributes (salary, address, credit use, etc.) of the same customers are split between the banks and are shared during the execution of the application.

3-1- Privacy-Preserving Bayesian Network Construction

Yang and Wright present an efficient privacy-preserving BN construction protocol [26, 27]. In this protocol, Yang and Wright adapt the heuristic K2 algorithm for BN construction to a privacy- preserving version where data is vertically partitioned between two parties. They first transform the f scoring function in the K2 algorithm and express the transformed function as a group of primitives that can be computed securely. Their protocol benefits from recent improvements in the area, such as Paillier's homomorphic cryptosystem [13], and Lindell and Pinkas' secure natural logarithm protocol [28]. Secure Boolean circuit evaluation is shown to be inefficient [11]; therefore, Yang and Wright aim to limit the amount of secure Boolean circuit evaluation and to benefit from the efficiency of Paillier's homomorphic scheme as much as possible. However, Yang and Wright's BN construction protocol does not provide complete privacy as per the definitions of SMC given earlier. That is, the order in which the parents are added is revealed to the parties during protocol execution. Figure 5 shows the pseudocode for the K2 algorithm.

```
Input: An order set of n nodes
Output: For each node, the parent assignments in the Bayesian network.
for each node i
{
  πi = Ø;
  Calculate scoring function g(i, πi);

  While there are nodes to consider for node i
  {
    Find node z that precedes i maximizes the score
    If 'score with z' is greater than the old score
        Save the new score
        Add z to the predecessor list for node i
    Else break;
  }
}
```

Figure 5. Pseudocode for the K2 algorithm

Unlike standard arithmetic functions, there is no simple solution for the factorial calculations, which have an important role in the scoring function of the K2 algorithm, to be performed safely. Whether the safe Boolean circuits used are calculating multiplications or searching for factorial values in very large tables, these solutions will not be practical. Very large Boolean circuits will be required for multiple multiplication operations, while large amounts of disk storage and processing power will be required to store factorial results in a table. In response to these problems, the Yang-Wright protocol [29] solves this problem by replacing each factorial in the K2 scoring function with a Stirling approach. The g function resulting from this approach is expressed as:

$$g(i,\pi i) = \sum_{j=1}^{q_i}\left(\sum_{k=1}^{d_i}(\frac{1}{2}\ln\beta_{ijk} + \beta_{ijk}\ln\beta_{ijk}) - (\frac{1}{2}\ln\ell_{ij} + \ell_{ij}\ln\ell_{ij})\right) + pub(d_i, q_i) \quad (2)$$

During the execution of the secure protocol, the parties do not know the β-parameters used in the g scoring function. The β-parameters in the function are values that the parties must calculate, not only to hide their input from each other and hide the credentials in the records used, but also to hide the calculation results and the number of matching records. Wright and Yang suggested using the safe scalar multiplication protocol described by Goethals et al. [30] for β-parameter calculations. However, in order to perform secure scalar multiplication, Yang and Wright's proposed protocol [29] also requires an encryption algorithm with additional homomorphic feature. For this purpose, Paillier's homomorphic encryption algorithm is used [13]. In the next step, the natural logarithm of the entire Stirling approximation is safely taken since the result of the scoring function is used only for ranking. The transformed function in this way can be safely calculated using the additive numerators of the β-parameters using the protocol of Lindell and Pinkas [28].

3-2- Privacy-Preserving K-Means Clustering

K-means is one of the simplest unsupervised clustering algorithms. It starts with prior knowledge of the target number of clusters (k) and randomly selected k cluster centers. The algorithm iteratively improves the cluster centers by calculating the means of the data points that fall in each cluster and, then, reassigning data points to the closest current centers. The algorithm stops when the cluster centers' change is below a predefined threshold from one iteration to the next, or when they do not change at all. While it can be proven that k-means algorithm always terminates [31], there is no guarantee that it will find an optimal solution, and the result is highly dependent on the selection of the initial cluster centers. Jagannathan & Wright described an efficient privacy-preserving k-means clustering protocol that works for randomly fragmented data [32]. K-means is a simple and widely used clustering algorithm in data mining. The algorithm starts with a non-clustered database of n items and l attributes and assigns each item in the database to the cluster that best fits that item. As a result it outputs the set assignments for each item. However, for the k-means clustering algorithm the number of clusters should be known beforehand.

There are many recent studies on the k-means clustering algorithm that can operate while maintaining confidentiality in the semi-honest model in which the database used as the input is divided between two parties [33, 34]. The protocol described by Jagannathan & Wright [32] can also work in the scenario where any data item and/or attribute is randomly split between parties. Figure 6 shows the pseudocode for the generic k-means algorithm.

```
Input: Database D of size n, k the number of cluster centers
Output: Cluster Assignments for n elements

Randomly select k initial cluster centers(μ1`,μ2`,..,μk`)
Do
  Set μ1, μ2, .., μk = μ1`, μ2`, .., μk`
  Calculate the distances between the cluster centers and the n data elements
  Assign each data element to the cluster center within the shortest distance.
  Recalculate the cluster centers(μ1`,μ2`,..,μk`)
Until (μ1, μ2, .., μk) and (μ1`, μ2`, .., μk`) are not different than the
threshold.
```

Figure 6. The k-means clustering algorithm

In Jagannathan & Wright's two-party privacy-preserving k-means clustering protocol [32], the partitioned database contains *l* dimensions (or attributes). In this protocol, the database is assumed to be arbitrarily partitioned between the parties. The parties want to learn the cluster assignments of the records in their shared database. The parties learn the cluster assignments of records that they have at least a share of. Besides the cluster assignments, the parties also learn the shares of the cluster centers. Moreover, after each iteration the parties learn intermediate cluster assignments of the protocol. At the first step of Jagannathan & Wright's protocol, *k* cluster centers are chosen randomly, and the cluster centers are shared between the parties. In the next step, the k-means algorithm requires calculation of Euclidean distances between each data point and the cluster centers. For this purpose, the parties cooperate to calculate the distances between data points and each cluster center. For a data point xi and a cluster center c_j; the distance is calculated as follows:

$$dist(x_i, c_j)^2 = (x_{i1}, - c_{j1})^2 + (x_{i2} - c_{j2})^2 + \ldots + (x_{in}, - c_{jn})^2 \quad (3)$$

Jagannathan & Wright convert this distance equation into a combination of local computations and a series of scalar products, which can be calculated by using the secure scalar protocol as defined by Goethals et al. earlier [30]. After securely calculating the distances for every data point, the parties decide the cluster center assignments. In this case, they select the cluster centers closest to each data point (i.e., that has the smallest distance). This function is performed by secure Boolean circuit evaluation [35]. There are 2k inputs for the circuit. As output, each party securely learns the cluster assignments. Figure 7 shows the workflow for the secure 2-party application of the k-means clustering protocol.

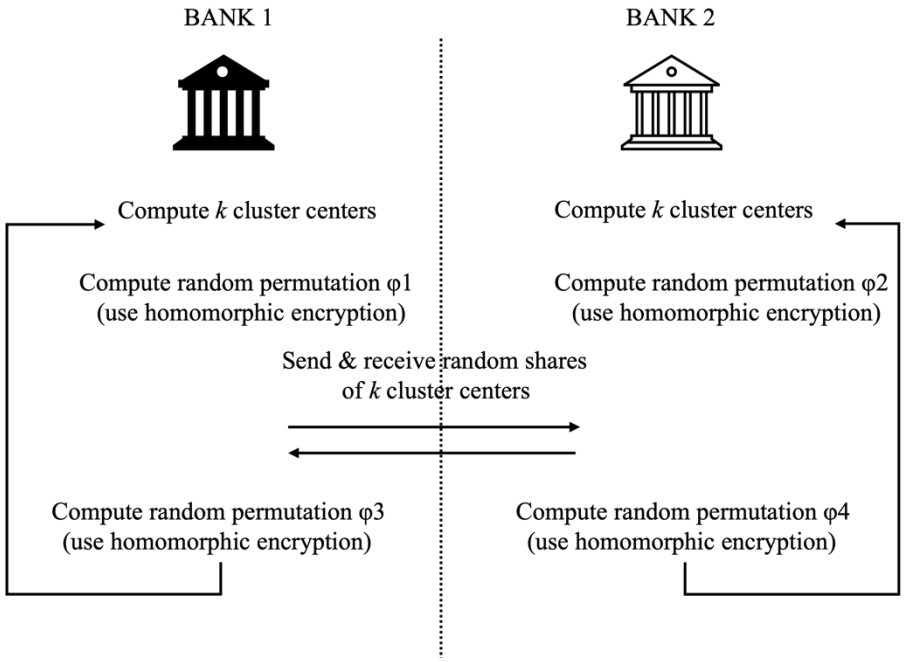

Figure 7. Workflow for the secure 2-party k-means clustering protocol

3-3- Experimental Analysis

This section provides analysis of the secure 2-party applications of Bayesian Network construction and k-means clustering protocols. Analysis focuses on the performance of the protocols and accuracy of the results. In order to simulate a 2-party environment, two computers with same specifications are used and they are connected in the same local area network. The network delays are measured to be negligible. Figure 8 shows the impact of the key size on the Bayesian network construction protocol, and distribution of the run time over the arithmetic operations. It can be observed that the as the key size increases, performance of the logarithmic function (ln *x*) becomes more significant.

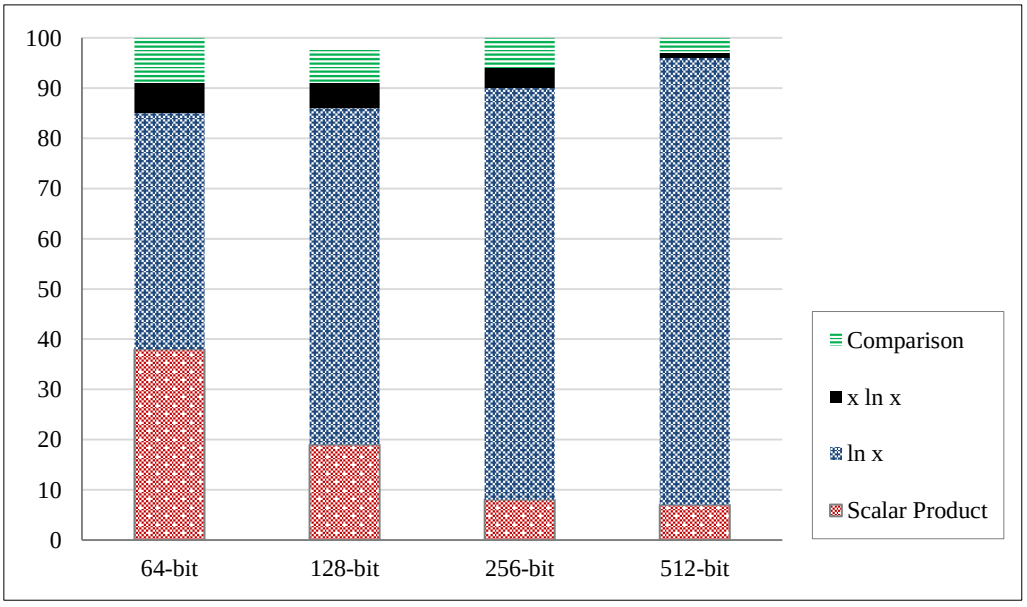

Figure 8. Run time distribution of the Bayesian network construction protocol between arithmetic operations

Focusing on the secure natural logarithm operation, Figure 9 shows that the performance is inversely related to the key size. In other words, as the banks double the size of the encryption keys for more security, run time of the Bayesian network protocol also doubles.

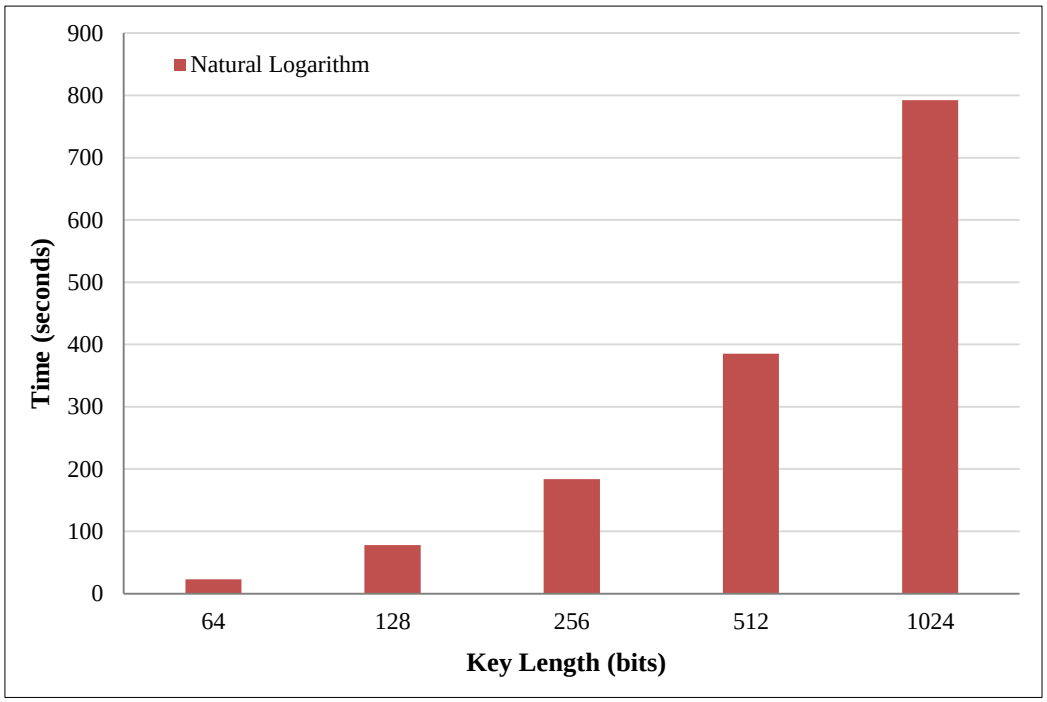

Figure 9. Performance of the secure natural logarithm function over key size

Figure 10 shows the performance of the secure k-means algorithm based on the database size and number of clusters. The impact of the database size on the performance reduces as the size increases. So, for very large databases (common in banking applications), the secure k-means algorithm still performs well.

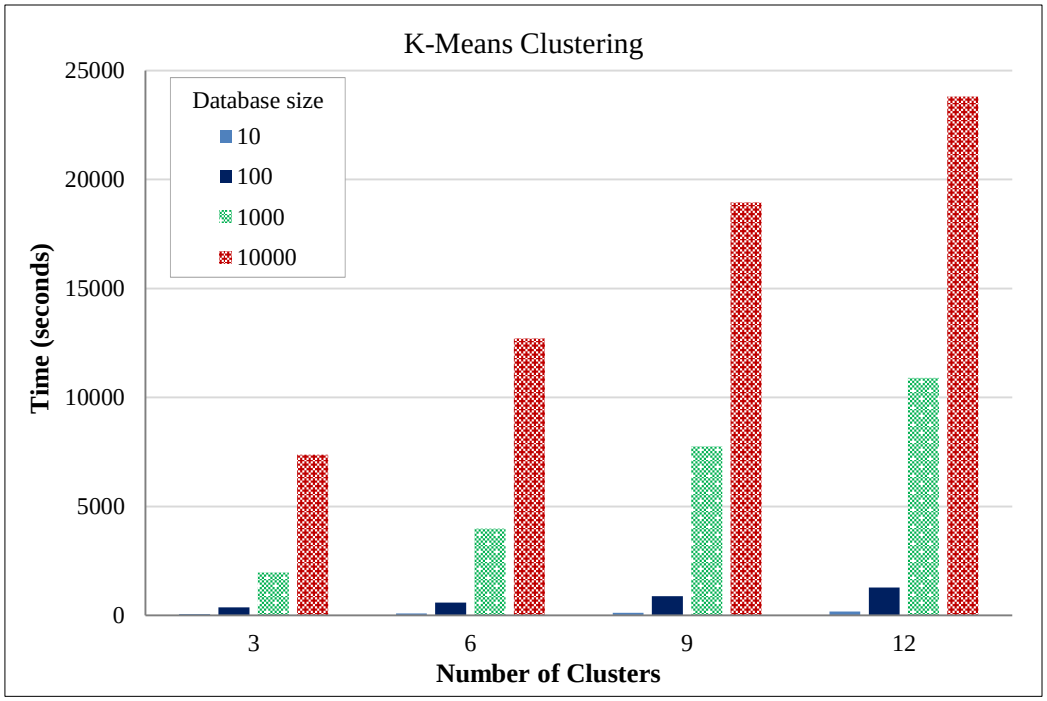

Figure 10. Performance of the secure k-means protocol based on the database size and number of clusters

Finally, Table 1 shows the accuracy of the results from the secure k-means protocol compared to the original algorithm. As the accuracy of the results drops as the database size increases, the secure k-means protocol results are within 5% of the results from the original algorithm.

Table 1. Accuracy of the secure k-means protocol compared to the original algorithm

Database size	Accuracy of the original k-means algorithm	Accuracy of the privacy-preserving k-means algorithm
37×2	0.94	0.94
41×2	0.90	0.90
44×2	0.86	0.84
73×2	0.86	0.83
55×2	0.87	0.8
50×2	0.9	0.84
52×2	0.88	0.80
80×2	0.86	0.83
11×11	0.81	0.81
96×5	0.81	0.80

4- Conclusion

As the need for data mining applications increases, companies are required to access more data sources for improved results. Distributed data mining techniques allow multiple companies to collaborate on the same data mining applications through sharing data between themselves. This scheme raises privacy concerns, as the shared data often contains PII data that is against regulations such as GDPR and HIPAA. Secure multi-party methods were developed to overcome this challenge. PPDM offers secure solutions to distributed data mining applications through secure multi-party computation (SMC). Homomorphic encryption is a key component of SMC, which provides techniques for performing arithmetic operations using encrypted data. This study discusses several privacy-preserving data mining applications that use homomorphic encryption and provide security even when some of the parties are malicious.

Data split horizontally or vertically between two parties can be combined to provide more accurate recommendations. Parties can use systems to offer recommendations based on split data while maintaining their confidentiality. Thanks to the protocols used in these systems, it is difficult for parties to obtain confidential data in an attack to act as an active user. As a result of randomly filling/subtracting some valuations from the valuation vector of the active user, different subtotals are obtained each time, so no inference can be made from this. Another important point is that when the parties can only access the sum of the valuations given to the products. There are studies in the literature that discuss generic applications of PPDM [26, 29, 36]. This study focuses on data mining applications that are commonly used in banking processes and aims to provide guidance to financial institutions for secure collaboration on data mining tasks.

5- Declarations

5-1- Data Availability Statement

Data sharing is not applicable to this article.

5-2- Funding

The author received no financial support for the research, authorship, and/or publication of this article.

5-3- Institutional Review Board Statement

Not applicable.

5-4- Informed Consent Statement

Not applicable.

5-5- Conflicts of Interest

The author declares that there is no conflict of interest regarding the publication of this manuscript. In addition, the ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancies have been completely observed by the authors.

6- References

[1] Agrawal, R., & Srikant, R. (2000). Privacy-preserving data mining. Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data - SIGMOD '00. doi:10.1145/342009.335438.

[2] Cramer, R., Damgård, I., Nielsen, J.B. (2001). Multiparty Computation from Threshold Homomorphic Encryption. Advances in Cryptology — EUROCRYPT 2001, Lecture Notes in Computer Science, 2045. Springer, Berlin, Germany. doi:10.1007/3-540-44987-6_18.

[3] Kantarcioglu, M., & Clifton, C. (2004). Privacy-preserving distributed mining of association rules on horizontally partitioned data. IEEE Transactions on Knowledge and Data Engineering, 16(9), 1026–1037. doi:10.1109/TKDE.2004.45.

[4] Du, W., & Zhan, Z. (2002). Building decision tree classifier on private data. Proceedings of the IEEE International Conference on Privacy, Security and Data Mining-Volume 14, 1–8. 1 December, Maebashi City, Japan.

[5] Evfimievski, A., Srikant, R., Agrawal, R., & Gehrke, J. (2002). Privacy preserving mining of association rules. Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining-KDD '02. doi:10.1145/775047.775080.

[6] Kantarcıoğlu, M., Clifton, C. (2004). Privately Computing a Distributed K-NN Classifier. Knowledge Discovery in Databases: PKDD 2004. PKDD 2004. Lecture Notes in Computer Science, 3202. Springer, Berlin, Germany. doi:10.1007/978-3-540-30116-5_27.

[7] Jagannathan, G., & Wright, R. N. (2005). Privacy-preserving distributed k-means clustering over arbitrarily partitioned data. Proceeding of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining-KDD '05. doi:10.1145/1081870.1081942.

[8] Wright, R., & Yang, Z. (2004). Privacy-preserving Bayesian network structure computation on distributed heterogeneous data. Proceedings of the 2004 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining- KDD '04. doi:10.1145/1014052.1014145.

[9] Gilburd, B., Schuster, A., & Wolff, R. (2004). Privacy-preserving data mining on data grids in the presence of malicious participants. Proceedings. 13th IEEE International Symposium on High Performance Distributed Computing, 24 August 2004 Honolulu, HI, USA. doi:10.1109/hpdc.2004.1323540.

[10] Yao, A. C. (1982). Protocols for secure computations. 23rd Annual Symposium on Foundations of Computer Science (SFCS 1982). doi:10.1109/sfcs.1982.38.

[11] Atallah, M.J., Du, W. (2001). Secure Multi-party Computational Geometry. Algorithms and Data Structures, WADS 2001, Lecture Notes in Computer Science, 2125. Springer, Berlin, Germany. doi:10.1007/3-540-44634-6_16.

[12] Boudot, F., Schoenmakers, B., & Traoré, J. (2001). A fair and efficient solution to the socialist millionaires' problem. Discrete Applied Mathematics, 111(1–2), 23–36. doi:10.1016/S0166-218X(00)00342-5.

[13] Paillier, P. (1999). Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. Advances in Cryptology—EUROCRYPT '99, EUROCRYPT 1999, Lecture Notes in Computer Science, 1592, Springer, Berlin, Germany. doi:10.1007/3-540-48910-X_16.

[14] Du, W., Han, Y. S., & Chen, S. (2004). Privacy-Preserving Multivariate Statistical Analysis: Linear Regression and Classification. Proceedings of the 2004 SIAM International Conference on Data Mining. doi:10.1137/1.9781611972740.21.

[15] Li, X., Yi, S., Cundy, A. B., & Chen, W. (2022). Sustainable decision-making for contaminated site risk management: A decision tree model using machine learning algorithms. Journal of Cleaner Production, 371, 133612.doi:10.1016/j.jclepro.2022.133612.

[16] Du, W., & Zhan, Z. (2003). Using randomized response techniques for privacy-preserving data mining. Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining-KDD '03. doi:10.1145/956750.956810.

[17] Beaver, D. (1997). Commodity-based cryptography (extended abstract). Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing-STOC '97. doi:10.1145/258533.258637.

[18] Zhan, J., Matwin, S., Chang, L. (2005). Privacy-Preserving Collaborative Association Rule Mining. Data and Applications Security XIX. DBSec 2005, Lecture Notes in Computer Science, 3654. Springer, Berlin, Germany. doi:10.1007/11535706_12.

[19] Hasheminejad, S. M. H., & Khorrami, M. (2018). Data mining techniques for analyzing bank customers: A survey. Intelligent Decision Technologies, 12(3), 303–321. doi:10.3233/IDT-180335.

[20] Özmen, M., Aydoğan, E. K., Delice, Y., & Toksarı, M. D. (2020). Churn prediction in Turkey's telecommunications sector: A proposed multiobjective–cost-sensitive ant colony optimization. WIREs Data Mining and Knowledge Discovery, 10(1). doi:10.1002/widm.1338.

[21] Matsunaga, F. T., Brancher, J. D., & Busto, R. M. (2014). Data mining applications and techniques: A systematic review. Rev. Eletrônica Argentina-Brasil Tecnologias da Informação e da Comunicação, 1(2).

[22] Olufemi Ogunleye, J. (2022). The Concept of Data Mining. Intechopen, London, United Kingdom. doi:10.5772/intechopen.99417.

[23] Li, Y., Jiang, X., Wang, S., Xiong, H., & Ohno-Machado, L. (2016). VERTIcal Grid lOgistic regression (VERTIGO). Journal of the American Medical Informatics Association, 23(3), 570–579. doi:10.1093/jamia/ocv146.

[24] Das, A., Bhattacharyya, D. K., & Kalita, J. K. (2003). Horizontal vs. vertical partitioning in association rule mining: a comparison. Proceedings of the 6th International Conference on Computational Intelligence and Natural Computation (CINC), 1617-1620, 26-30 September, 2003, Embassy Suites Hotel and Conference Center, Cary, North Carolina, United States.

[25] Hemlata, & Gulia, P. (2017). Novel algorithm for PPDM of vertically partitioned data. International Journal of Applied Engineering Research, 12(12), 3090–3096.

[26] Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, 226–231, 2-4 August, 1996, Portland Oregon, United States.

[27] Evfimievski, A., Gehrke, J., & Srikant, R. (2003). Limiting privacy breaches in privacy preserving data mining. Proceedings of the Twenty-Second ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems-PODS '03. doi:10.1145/773153.773174.

[28] Lindell, Y., & Pinkas, B. (2012). Secure two-party computation via cut-and-choose oblivious transfer. Journal of Cryptology, 25(4), 680–722. doi:10.1007/s00145-011-9107-0.

[29] Yang, Z., & Wright, R. N. (2006). Privacy-preserving computation of bayesian networks on vertically partitioned data. IEEE Transactions on Knowledge and Data Engineering, 18(9), 1253–1264. doi:10.1109/TKDE.2006.147.

[30] Goethals, B., Laur, S., Lipmaa, H., Mielikäinen, T. (2005). On Private Scalar Product Computation for Privacy-Preserving Data Mining. Information Security and Cryptology – ICISC 2004. ICISC 2004, Lecture Notes in Computer Science, 3506. Springer, Berlin, Germany. doi:10.1007/11496618_9.

[31] Har-Peled, S., & Sadri, B. (2005). How fast is the k-means method? Algorithmica, 41(3), 185–202. doi:10.1007/s00453-004-1127-9.

[32] Jagannathan, G., & Wright, R. N. (2005). Privacy-preserving distributed k-means clustering over arbitrarily partitioned data. Proceeding of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining-KDD '05. doi:10.1145/1081870.1081942.

[33] Freedman, M.J., Nissim, K., Pinkas, B. (2004). Efficient Private Matching and Set Intersection. Advances in Cryptology-EUROCRYPT 2004. EUROCRYPT 2004, Lecture Notes in Computer Science, 3027. Springer, Berlin, Germany. doi:10.1007/978-3-540-24676-3_1.

[34] Bunn, P., & Ostrovsky, R. (2007). Secure two-party k-means clustering. Proceedings of the 14th ACM Conference on Computer and Communications Security- CCS2007. doi:10.1145/1315245.1315306.

[35] Malkhi, D., Nisan, N., Pinkas, B., & Sella, Y. (2004). Fairplay-Secure Two-Party Computation System. USENIX Security Symposium, 9-13August, 2004, San Diego, United States.

[36] Kissner, L., Song, D. (2005). Privacy-Preserving Set Operations. Advances in Cryptology – CRYPTO 2005, CRYPTO 2005, Lecture Notes in Computer Science, 3621. Springer, Berlin, Germany. doi:10.1007/11535218_15.