UNECE Project on Input Privacy Preservation: Final Report

Version 28 March 2023

About this document

This document describes the work done under and findings from the UNECE Input Privacy Preservation Project that was done under the High-Level Group for the Modernisation of Official statistics in 2021-2022.

This work is licensed under the Creative Commons Attribution 4.0 International License. To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/. If you re-use all or part of this work, please attribute it to the United Nations Economic Commission for Europe (UNECE), on behalf of the international statistical community.

Acknowledgements

The UNECE Input Privacy Preservation Project was part of the work programme of the High-Level Group for the Modernisation of Official Statistics. It started in January 2021 and finished in December 2022. The following persons participated in the project:

Mauro Bruno (Istat, Italy), Nita Boushey (Statistics Canada), Massimo De Cubellis (Istat, Italy), Abel Dasylva (Statistics Canada), Joeri van Etten (Statistics Netherlands), Fabrizio De Fausti (Istat, Italy), Loe Franssen (Statistics Netherlands), Matjaz Jug (Statistics Netherlands), Chris Maloney (Statistics Canada), Saeid Molladavoudi (Statistics Canada), Alexandre Noyvirt (Office for National Statistics, UK),Owen Daniel (Office for National Statistics, UK), Dennis Ramondt (Statistics Netherlands), Fabio Ricciato (Eurostat), Ralph Schreijen (Statistics Netherlands), Benjamin Santos (Statistics Canada), Monica Scannapieco (Istat, Italy), Julian Templeton (Statistics Canada), Mat Weldon (Office for National Statistics, UK), Zachary Zanussi (Statistics Canada).

Acronyms

CA	Central Authority
CETA	Comprehensive Economic and Trade Agreement
DP	Differential Privacy
DP-SGD	Differentially Private Stochastic Gradient Descent
FL	Federated Learning
FPR	False Positive Rate
HE	Homomorphic Encryption
IoT	Internet of Things
IPP	Input Privacy Preservation
MIA	Membership Inference Attack
ML	Machine Learning
MLP	Multi-Layer Perceptron
MPC	Multi-Party Computation
MPSPC	Multi-Party Secure Private Computing
MPSPCaaS	MPSPC as a Service
NN	Neural Network
NSO	National Statistical Organization
PET	Privacy Enhancing Technology
PML	Private ML
PPML	Privacy Preserving ML
PSI	Private Set Intersection
ReLU	Rectified Linear Unit
SGD	Stochastic Gradient Descent
SSI	Sensitive Statistical Information
TEE	Trusted Execution Environment
TTP	Trusted Third Party
WP	Work Package

Table of Contents

Acknowledgements 2

Acronyms 3

Table of Contents 4

Introduction 5

Chapter 2 - Track 1: Private set intersection 7

2.1 Motivation 7

2.2 Private set intersection 8

2.3 Implemented protocol 9

2.4 Private set intersection with a quasi-identifier 10

2.5 Evaluation 12

2.6 Lessons learnt 16

2.7 Potential next steps 17

Chapter 3 - Track 2: Private machine learning 18

3.1 Privacy-Preserving Federated Machine Learning 18

3.1.1 Methodology 18

3.1.2 Experiments 19

3.1.3 Results 20

3.1.4 Conclusions 21

3.2 Defending Against Membership Inference Attacks with Differential Privacy on a Linked Dataset 22

3.2.1 Scenario 1 - Differentially Private Data and Differentially Private Training 23

3.2.2 Scenario 2 - Investigation of Membership Inference Attacks 27

3.2.3 Scenario 3 - In-Depth Investigation of Membership Inference Attacks and Differential Privacy 30

3.2.4 Conclusions 33

Chapter 4 - Track 3: Open consultation 34

4.1 Overview 34

4.2 Multi-Party Secure Private Computing-as-a-Service concept 35

4.3 Summary of received responses 40

Chapter 5 - General takeaways 44

Annex - Private set intersection with analytics: use case Istat-Bank of Italy 46

Motivations 46

Use case 47

Conclusions 49

References 50

Introduction

1. In this document, we describe the work done within the UNECE Project on Input Privacy Preservation (IPP). The project started in January 2021 with the aim of investigating statistical use cases requiring input side protection, evaluating and determining the applicability of selected classes of techniques for selected scenarios, identifying opportunities for sharing knowledge within the statistical community and creating a community of practice between statistical organizations and external partners (academia, private sector).

2. Modern statistical organizations need to become part of a data ecosystem and integrate data from multiple sources with the aim of providing richer statistical products. The potential of IPP techniques can help National Statistical Organizations (NSOs) to (re)use private and administrative data for statistical purposes while preserving data confidentiality and individual privacy rights. The main IPP techniques investigated in the project were Secure Multi-Party Computation (SMPC), Homomorphic Encryption, Differential Privacy, and Federated Learning.

3. The project lasted for two years. In the first year (2021) main organizations involved in the project (UNECE, Eurostat, Istat (Italy), Statistics Canada (StatCan) and Statistics Netherlands) shared and explored five use cases that required the application of IPP techniques. We worked on the definition of a logical framework to describe the use cases in a standard way, and we generalized the use cases with the aim of making them applicable in other similar contexts. Moreover, we made two mini work packages (WPs): one on Private Set Intersection (WP1) and the other on Federated Learning (WP2). Furthermore, the initial ideas of the Multi-Party Secure Private Computing-as-a-service (MPSPCaaS) concept were sewed that led in the second year to organizing an open consultation (WP3) with the aim of collecting feedback from external experts and stakeholder.

4. In the second year (2022) of the project we followed a more vertical approach. We considered the concept of privacy preservation in a broader sense, such as Privacy Enhancing Technologies (PETs), also touching on the possible integration between IPP and Differential Privacy approaches. The activities have been divided into the following three tracks: WP1 Private Set Intersection, WP2 Private Machine Learning and WP3 Open consultation (on the MPSPCaaS concept). The organizations most involved in the 2022 tracks were UNECE, Eurostat, Istat (Italy), Statistics Canada, Statistics Netherlands and ONS (UK).

5. We give below an idea of the work done in the three tracks; further details are provided in their respective chapters in this document.

6. Track 1: Private Set Intersection (PSI). This deals with an international trade use case where two NSIs want to match their international trade transactions, but for confidentiality reasons the microdata cannot be shared between the two agencies. Different types of PSI solutions are illustrated. A comparison of methods with and without typos is made by applying and comparing techniques such as bloom filters.

7. Track 2: Private Machine Learning (PML). The focus was mainly to evaluate the robustness in terms of privacy preservation of the machine learning models trained in privacy preserving ways. We investigated the following three scenarios :(i) the application of Differential Privacy (DP) to the data compared with the application of DP to the model training; (ii) investigation of membership inference attacks; (iii) a custom implementation of a membership inference attack.

8. Track 3: Open consultation. We defined the concept of Multi-Party Secure Private Computing-as-a-Service (MPSPCaaS) and we launched an open consultation aimed at stakeholders (prospective users, technology experts, and privacy experts) to gather their views on various technical and non-technical aspects.

9. In addition to this, in the appendix, we described the use case "Private set intersection with analytics: use case Istat-Bank of Italy" analyzed and shared during the first year of the project. It was part of the input to track 1 developed in 2022 and described in chapter 2.

Chapter 2 - Track 1: Private set intersection

2.1 Motivation

10. National Statistical Organizations (NSOs) typically have much more information about their import of goods than of their exports. This is partly due to efficiency reasons and partly due to an increased interest in controlling what enters the national borders than what exits it. While import values from one country should equal the export values from its partner country, in practice this is not the case. As a result, there is wide interest from both statistical and academic organizations to get a better understanding where these bilateral trade asymmetries come from. In addition, research into drivers and effects of exports can often run into statistical limitations, even though the required information would be available with the trading partner. To alleviate these problems, NSOs would benefit from linking their international trade data.

11. Unfortunately, there are obstacles to linking such data at a micro (i.e., transactional) level in practice. One of the main issues concerns privacy. There does not typically exist a common transaction identifier between two NSOs that would protect the anonymity of the firms involved. National laws prevent sharing any information that might jeopardize this anonymity. While exceptions may exist to share such data amongst NSOs within the European Union, there is virtually no possibility to share data between NSOs where at least one is not an EU member.

12. This case study examines the use of Privacy Preserving Techniques (PPT) to share micro data on international trade between the Netherlands and Canada. This serves several interests. First, it would be a way to investigate bilateral trade asymmetries at a highly detailed level. Previous studies have typically looked at country or product characteristics to explain asymmetries. By looking at a transaction level, we can also investigate firm determinants for example. Second, it would allow the enrichment of each NSOs' trade statistics. Specifically, we are interested in the use by both importers and exporters of the Comprehensive Economic and Trade Agreement (CETA) between the EU and Canada that has been provisionally in use since September 2017. Currently, NSOs only record whether an imported good made use of this agreement (and as such obtained preferential tariffs), but not whether exports made use of it. While this information could also be provided at an aggregate level by the importing agency to the exporting agency, it would not allow disaggregated statistics on, e.g., the use of the agreement by exporters of different firm sizes. In addition, academic research that try to explain why the utilization rate of trade agreements typically stagnates around 60-70% need to be able to examine these barriers at the most detailed perspective, ideally at the importer*exporter*product level (see also Nilsson, 2022). In fact, there are various applications from international trade where research arguably stagnates due to the lack of matched importer-exporter data. This case study would be a first attempt to match these data using PPT. If successful, it might serve as a template for other NSOs.

13. Figure 2.1 shows a simplified view of the micro-data in possession of both Statistics Netherlands and Statistics Canada. The white variables are common to both data sets, the green variables are only present on the import (in this case Canadian) side and the blue variable is only available on the export (i.e., Dutch) side. The different variables may be classified as identifiers (e.g., the exporter id) or analytical variables. An identifier is found on both data sets and may be used as a linkage variable. However, it typically has no analytical value. An identifier is further classified as a unique identifier or a quasi-identifier. As the name suggests, a unique identifier uniquely identifies each unit in the data set. However, a quasi-identifier may

assign the same value of the identifier to many units in the data set. Additionally, a quasi-identifier may have typos. For example, the exporter id is a quasi-identifier for a transaction because many transactions may be associated with the same exporter id. By contrast, an analytical variable is a variable of interest for the analysis, such as the date, the value, the preferential code, the exporter size or the importer size. It need not be on both data sets but may be also used as linkage variable if this is the case, such as the transaction date or the transaction value, which are quasi-identifiers.

14. By linking the transactions of the two datasets, it is possible to compile the total value of Dutch exports into Canada by tariff type and exporter size and analyze how different kinds of Dutch exporters take advantage of CETA. However, the micro-data cannot be freely exchanged between the two NSOs because it is Sensitive Statistical Information (SSI). Thus, the two NSOs must find a way to combine their micro-data without revealing the size and identity of any Dutch exporter to StatCan, and without revealing the size and identity of any Canadian importer or the tariff applied to any transaction to CBS. A solution is to perform a private set intersection of the two data sets.

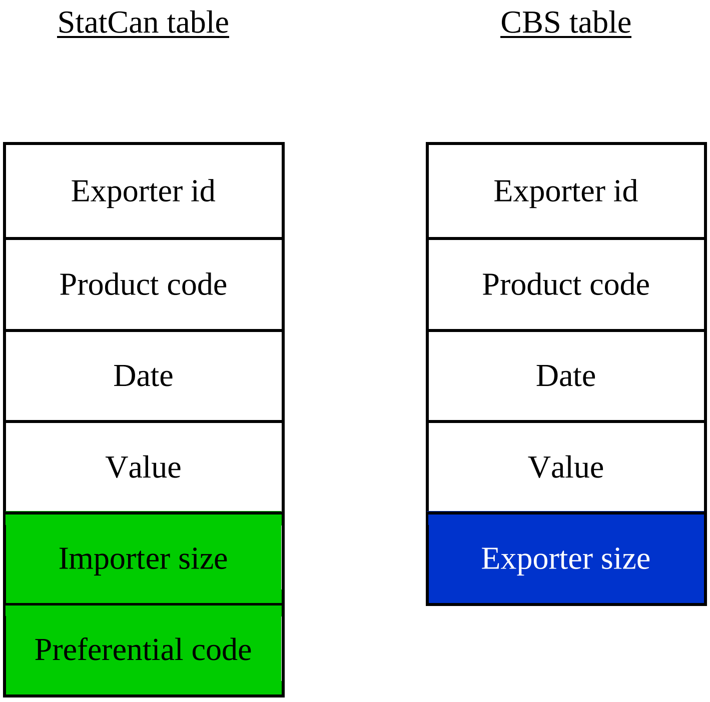


Figure 2.1: The transaction micro-data.

2.2 Private set intersection

15. Private set intersection methods have been developed for situations where two parties wish to privately determine the intersection of their data sets, without revealing any information about the units that are outside the intersection. Based on the intersection, some total may be computed, such as the coverage of one of the data sets (Dasylva and Zanussi, 2021), or some micro-data may be transferred to one party. When the two data sets have a common unique identifier, a simple idea is to hash this identifier and determine the intersection by comparing the two sets of hashed values. In general, the parties are assumed to be honest but curious. This means that they are expected to follow the protocol and seek any available information about

the other data set or party. Quite a few solutions have been proposed for implementing a private set intersection of which many have been described by Kamara et al. (2014). These solutions involve two or three parties. A three-party solution includes a trusted third party, which does the linkage, beyond the two data-holding parties. Additionally, the linker may transfer some micro-data on the intersection units to one of the other parties. However, this requires a perfect linkage of the two data sets.

2.3 Implemented protocol

16. The implemented PSI protocol is described by Bruno et al. (2021). It is based on the protocol by De Cristofaro and Tsudik (2010) and comprises three steps. In the first step, the two data-holding parties exchange the protocol and encryption parameters. In this step, they also determine the intersection of the two data sets by exchanging the hashed and encrypted values of the identifier for their respective data sets. In the second step, each party sends the micro-data from the intersection to the linker, including the analytical variables. This micro-data is encrypted and comprises the micro-data of each unit that is in the intersection. It is used to compute the totals of interest. In the third step, one data-holding party sends a query to have a total computed by the linker. The linker computes the requested total and replies to the party. The solution is based on the availability of a unique identifier and exact comparisons of the hashed values of this identifier. A quasi-identifier may be used but this has important implications that are discussed in the next section. As mentioned before, the linker computes an aggregate on the intersection based on the analytical variables. The linker may perturb these outputs, e.g., with differential privacy techniques, to make them safe.

17. The code is available on Github at https://github.com/UNECE/Input-Privacy-Preserving-Project.

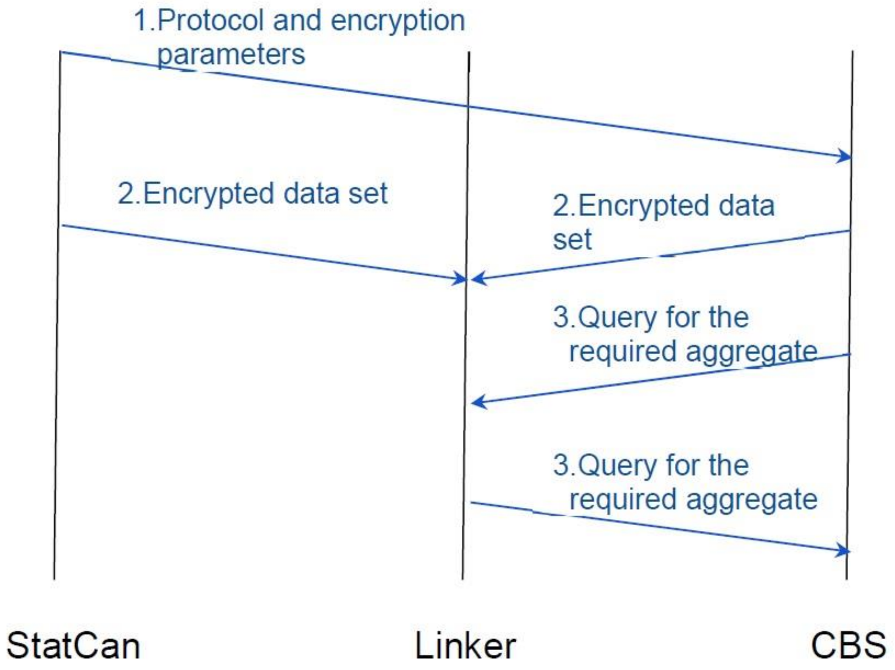

1.Protocol and encryption parameters
2.Encrypted data set
2.Encrypted data set
3.Query for the required aggregate
3.Query for the required aggregate
StatCan
Linker
CBS

Figure 2.2: The PSI protocol.

2.4 Private set intersection with a quasi-identifier

18. When there is no unique identifier, it is possible to use a quasi-identifier that is based on a combination of attributes such as the exporter id, product code, transaction date and transaction value. However, this may lead to linkage errors because a quasi-identifier is not unique, unlike a business number that is expressly designed to uniquely identify businesses and has no analytical value. Also, a quasi-identifier is possibly recorded with typos or spelling variations. For example, the same international trade transaction may be recorded with a different date or value by two countries. The linkage errors include false negatives and false positives, where a false negative is not linking records from the same unit and a false positive is linking records from different units. From the private set intersection perspective, these errors lead to a trade-off between omitting to include a unit that is indeed in the intersection, or erroneously including a unit in this intersection. They are also a potential source of bias, when computing the totals of interest. The issue may be addressed by performing approximate record comparisons, estimating the linkage accuracy and adjusting the computed totals according to the reported accuracy.

19. In general, it is challenging to perform an approximate comparison of encrypted records. One solution is based on using Bloom filters, which encode strings into a long sequence of bits (e.g., 1,000 bits) such that it is computationally infeasible to map the bit sequence back to the input strings. In a basic Bloom filter, all the bits are initially set to 0. To encode a string, it is broken into n-grams, typically bigrams (i.e., n=2), which are visited in sequence from the beginning to the end of the string. Each n-gram is mapped to many bit positions by a set of hash functions (e.g., 20 such functions) and each selected bit position is flipped to 1 if it is not already flipped, otherwise, it is left unchanged, i.e., a bit is never flipped back to 0. The similarity of two Bloom filters may be measured with the Dice similarity that is based on the number of bit positions that are set to 1 in both filters. It is possible to use a separate Bloom filter for each linkage variable. However, for greater security, a record-level Bloom filter is preferred, which is built by mapping each linkage variable to a separate filter and integrating the resulting filters into a single bit string. Fully homomorphic encryption offers an alternative to Bloom filters that is more flexible but also more complex. However, both methods require significant changes to the PSI protocol. To avoid such changes, one can still perform exact comparisons at the expense of having a greater number of false negatives.

	Linked	Not linked
Matched	TP	FN
Unmatched	FP	TN

Table 2.1: Confusion matrix

20. Measuring the linkage accuracy is key because linkage errors may occur at the linker. To further discuss this accuracy, call a record pair matched if the records are from the same unit. Otherwise call it unmatched. Then a false negative is a matched pair that is not linked,

and a false positive is an unmatched pair that is linked. Also define a true positive as a matched pair that is linked and a true negative as an unmatched pair that is not linked. The four pair types are usually represented in a 2x2 table called confusion matrix, which is shown on Table 2.1.

21. The linkage accuracy is assessed with many measures including the recall, precision and false positive rate (FPR), where the recall is the proportion of matched pairs that are linked, the precision is the proportion of linked pairs that are matched, and the false positive rate is the proportion of unmatched pairs that are linked. Estimating these measures is a challenge with encrypted or hashed records because clerical reviews are impossible. Instead, one must use one of many proposed statistical models by Fellegi and Sunter (1969), Thibaudeau (1993), Armstrong and Mayda (1993), Winkler (1993), Belin and Rubin (1995), Blakely and Salmond (2002), Daggy et al. (2013), Dasylva and Goussanou (2022) or others. The models from Blakely and Salmond (2002) and Dasylva and Goussou (2022) are quite convenient because they make no assumption about the dependence of the linkage variables. They apply when linking two duplicate-free data sets where one data set is contained in the other and the decision to link two records involves no other record. The solution by Dasylva and Goussanou (2022) has the additional advantage of accounting for the records heterogeneity.

22. At the linker, a total may be estimated by simply taking the corresponding total over the linked pairs. However, this yields a naïve estimator that is biased, because it ignores the linkage errors. When the probability of a true positive and that of a false positive do not depend on the analytical variables, the bias may be removed by using the following estimator:

$$\frac{(total\ over\ the\ linked\ pairs\ ) - FPR \times (total\ over\ the\ Cartesian\ product\ )}{recall - FPR},$$

23. This coincides with the naïve estimator if the linkage is perfect, i.e., if $recall = 1.0$ and $FPR = 0.0$. This estimator also corresponds to reweighting the links when there are no false positives, i.e., $FPR = 0.0$. Of course, with large files, it is challenging to compute the total over the Cartesian product. Instead, one may replace this total by a total over the pairs that agree on blocking keys. Such keys are commonly used with the probabilistic method of record linkage, where they serve to select a small subset of the Cartesian product that includes most if not all matched pairs. A pair is deemed to have a non-negligible probability of being matched if the records agree on at least one blocking key. Blocking keys are usually based on quasi-identifiers that have few typos and stable values over time. Call a pair blocked if it is selected by at least one blocking key and let the blocking FPR be the proportion of unmatched pairs that are blocked. Now suppose that each matched pair is blocked, and that the same assumption holds regarding the relation between the analytical variables and the true and false positive probabilities. Then one may use the following bias-corrected estimator.

$$\frac{(total\ over\ the\ linked\ pairs\ ) - \frac{FPR}{(Blocking\ FPR)} \times (total\ over\ the\ blocked\ pairs\ )}{recall - \frac{FPR}{(Blocking\ FPR)}}.$$

24. It is easy to see that it coincides with the first bias-corrected estimator if no blocking keys are used, i.e., every pair is blocked and $Blocking\ FPR = 1.0$. Lahiri and Larsen (2005),

Chambers and Kim (2018), Dasylva (2018) have described other error adjustment methods, with a focus on fitting a parametric statistical model.

25. When each transaction is recorded in both data sets with typos, the private set intersection may be based on modifying the PSI protocol to incorporate the model-based estimation of the linkage accuracy and the adjustment, as described above. In the following section the performance of this solution is evaluated through simulations in R.

2.5 Evaluation

26. The implemented PSI protocol is evaluated with synthetic international trade micro-data. Additionally, simulations are conducted to evaluate the described solution components when there are typos, including the use of approximate comparisons with Bloom filters, the model-based estimation of the linkage accuracy, and the adjustment to remove the bias caused by the linkage errors.

27. Synthetic micro-data: It is based on 100K transactions, where each transaction is associated with an (Dutch) exporter id, an exporter firm size indicator (small or large sized firm), an importer size indicator, a product code (according to the 6 digits harmonized system), a date, a value and a preferential code indicating whether the good was imported preferentially (code 300) or not (code 100). Each transaction is recorded in StatCan import micro-data and CBS export micro-data. StatCan import micro-data contains the vendor id, product code, date, value and the preferential code. CBS export micro-data contains the exporter id, exporter size, product code, date and value. The exporter id, product code, date and value may be used as linkage variables as they are available in both data sets. Initially, the micro-data is generated without typos such that each transaction is recorded with the same value for each linkage variable in each data set. In this case, the combination of the exporter id, product code, date and value, is unique in each data set. Table 2.2 shows the number of transactions by exporter size and tariff type in the micro-data with no typos. Then, the date and value of each transaction are perturbed in the CBS data set as follows. With probability 0.8, the date and value are left unchanged. Otherwise, the date is moved by a random number of days according to a normal distribution with mean 0 and standard deviation 20, while the transaction value is deflated at random with a mean of 15% and a standard deviation of 5%.

Exporter size	Tariff regime	Number of transactions
Small	Non preferential	35,433
	Preferential	11,923
Large	Non preferential	39,567
	Preferential	13,077

Table 2.2: Number of transactions by exporter size and tariff regime in the synthetic data.

28. *Analytical need*: The goal is to privately link the two data sets to study how small Dutch exporters take advantage of CETA preferential tariffs into Canada. To this end, one may estimate the number of export transactions by tariff type for such exporters, with the described PSI protocols, without any modification of these protocols. One may also estimate the total

value of export transactions by tariff regime. In practice, computing this value requires homomorphic additions at the linker. However, this feature is mimicked in the simulations.

29. *Evaluating the PSI protocol*: The implemented protocol is run on a single Windows 10 virtual machine on the Microsoft Azure cloud. The same machine plays each role, including the two data holding parties (StatCan and CBS) and the linker. The synthetic data with no typos is used where the combination of the exporter id, product code, date and value, is unique and used as a linkage key. In the third protocol step, the CBS client sends a query to the linker to compute the number of transactions by exporter size and tariff type. The whole operation takes about 10 min.

30. In the third protocol step, the CBS client sends a query to the linker to compute the same frequencies. The obtained output (at the CBS client) is shown on Table 2.3, where the estimated frequencies agree with the actual frequencies on Table 2.2.

```
…
QUERY MODE
exp_size ; pref_kode ; IDKEY
small ; 100 ; b'35433'
small ; 300 ; b'11923'
large ; 100 ; b'39567'
large ; 300 ; b'13077'
```

Table 2.3: Query response at the CBS client.

31. The above results show that the protocol performs as expected. The execution time shows that the code can be optimized to handle a larger volume of transactions.

32. Monte Carlo simulations: The simulations are conducted to further evaluate the methodology to account for the linkage errors.

33. The Monte Carlo simulations comprise 100 repetitions. They are conducted by taking a subset of 10K transactions with typos from the 100K synthetic transactions. These transactions are then broken into 100 independent chunks, where each chunk comprises 100 transactions and corresponds to a repetition. In a repetition, each related transaction is recorded on each data set (i.e., on the import micro-data and the export micro-data), the data sets are linked, the linkage errors are estimated, and the totals of interest are estimated. These totals include the number of transactions by small Dutch exporters, which had a value equal to or greater than 500K and benefited from the preferential tariff, and the total value of transactions by small Dutch exporters, which benefited from the preferential tariff.

34. In the first scenario, the data sets are linked using the blocking key based on the concatenation of the exporter id and the product code and the linkage key based on the concatenation of the exporter id, product code, date and value, with an exact comparison of the linkage key. In the second scenario, the data sets are linked using the same blocking key and a record-level Bloom filter based on the exporter id, product code, date and value. Two records are linked if they have the same value of the blocking key and the Dice similarity of the Bloom filters is equal to or greater than a threshold, which is set to 0.8 or 0.9. For each scenario, the actual recall and false positive rate are compared to the model-based estimates. Also, for each total of interest (number of export transactions or total value exported), the actual value is

compared to the naïve estimate based on the links and the adjusted estimate, which accounts for the linkage errors.

35. For the first scenario, the results appear on Tables 2.4-2.6, with three decimals of accuracy. They show that the linkage accuracy is accurately estimated with the model. For each total of interest, they also show that the adjusted estimator has a small bias while the naïve estimator has a large bias.

36. For the second scenario, the results appear on Tables 2.7-2.9, again with three decimals of accuracy. They show that the linkage accuracy is accurately estimated with the model and that the adjusted estimator has a small bias for each total, as before. However, they also show that the bias of the naïve estimator is small when the recall is high.

37. The comparison of the results for the two simulation scenarios shows that the adjusted estimator tends to have a smaller bias and a smaller variance when using the Bloom filters. This is expected because the Bloom filters enable finer approximate comparisons of the records.

38. Overall, the obtained results demonstrate that the implemented protocol allows the private computation of a total over the intersection when the analytical variables are categorical, and the linkage variables have no typos and can be combined into a key that is unique. The simulation results show that it is still possible to accurately estimate the total by performing exact comparisons when there are typos, so long as one estimates the linkage accuracy and adjusts the naïve estimator to remove the bias. They also show that the naïve estimator may be used when the recall is high. Using Bloom filters does provide an advantage over exact comparisons, when it comes to the bias and variance of the adjusted estimator. However, it requires substantial changes to the protocol. Important changes are also required if one is to perform more sophisticated approximate comparisons through homomorphic encryption, to estimate totals based on quantitative analytical variables or to fit a statistical model.

Measure	Estimator	Mean	Standard error
Recall	Actual	0.8	0.038
	Estimate	0.8	0.038
FPR	Actual	0.0	0.0
	Estimate	0.0	0.0

Table 2.4: Rates of linkage error in the first simulation scenario.

Estimator	Mean	Standard error
Actual	6.370	2.116
Naïve	5.320	2.000
Adjusted	6.680	2.530

Table 2.5: Number of export transactions with a value equal to or greater than 500K and a preferential tariff, by small Dutch exporters, in the first simulation scenario.

Estimator	Mean (M)	Standard error (M)
Actual	6.260	1.697
Naïve	5.214	1.585
Adjusted	6.514	1.952

Table 2.6: Total value of export transactions with a preferential tariff by small Dutch exporters, in the first simulation scenario.

Dice similarity	Measure	Estimator	Mean	Standard error
0.9	Recall	Actual	0.796	0.042
		Estimate	0.796	0.042
	FPR	Actual	0.0	0.0
		Estimate	0.0	0.0
0.8	Recall	Actual	0.949	0.020
		Estimate	0.949	0.020
	FPR	Actual	0.0	0.0
		Estimate	0.0	0.0

Table 2.7: Rates of linkage error in the second simulation scenario.

Dice similarity	Estimator	Mean	Standard error
0.9	Actual	6.310	2.187
	Naïve	5.190	2.083
	Adjusted	6.481	2.634
0.8	Naïve	5.960	2.183
	Adjusted	6.266	2.303

Table 2.8: Number of export transactions with a value equal to or greater than 500K and a preferential tariff, by small Dutch exporters, in the second simulation scenario.

Dice similarity	Estimator	Mean (M)	Standard error (M)
0.9	Actual	6.231	1.705
	Naïve	5.023	1.665
	Adjusted	6.310	2.066
0.8	Naïve	5.977	1.722
	Adjusted	6.293	1.785

Table 2.9: Total value of export transactions with a preferential tariff by small Dutch exporters, in the second simulation scenario.

2.6 Lessons learnt

39. The first lesson is that private set intersection technologies are promising but require more work. For example, the described protocol may be used to study certain bilateral trade asymmetries, when each transaction is recorded by each trading partner and the analytical variables are categorical. However more work is required if the data sets overlap partially, the analytical variables are quantitative or if fitting a statistical model.

40. The second lesson is that there are many technological components, but no complete solution integrates all the required features, which include the verification of inputs (to only accept legitimate inputs), the verification of outputs (to only permit legitimate computations), private inputs (e.g., through encryption), safe outputs (to protect against inferential disclosure) and flow governance, where the latter refers to policies and procedures to control who can use the Privacy Enhancing Technology infrastructure. In general, this last component is lacking, which makes it difficult to provide evidence that the environment is safe, and the privacy guaranteed.

41. The third lesson is that the main obstacles are not technological. Indeed, cybersecurity concerns are currently preventing a joint test of the protocol over the Internet. There are also legal concerns, which prevent the transmission of encrypted micro-data from a statistical agency to an external party, since encryption is not considered sufficient for de-identification

from a legal standpoint. Finally, there is a need for a dedicated IT infrastructure to further test PETs.

42. The last lesson is that collaboration and teamwork are key. Both objectives are best met by building multidisciplinary teams with subject matter specialists, methodologists, computer scientists and legal experts. It is also important to pool the resources, e.g., with a solution for multi-party computation as a service.

2.7 Potential next steps

43. In the future, the protocol must be evaluated further by performing a joint test over the Internet, with StatCan and CBS as the data holding parties and Istat (or another organization, e.g., the UN PET Lab) as the linker. It must also be modified to incorporate the proposed features to deal with typos, including the estimation of the linkage accuracy and the adjustment of the estimated totals. For a first step in this direction, see Dasylva (2022). Beyond these solutions, one must evaluate other approaches that only involve two parties without trust, such as those based on multi-party computation or garbled circuits. This goal can be met by leveraging the UN PET lab infrastructure, where many state-of-art solutions are implemented.

44. Another avenue for further exploration is how to do more advanced analyses on the matched dataset. Instead of descriptive statistics, there is a demand for academic econometric analysis on such linked datasets, which comes with further requirements to the protocol.

Chapter 3 - Track 2: Private machine learning

3.1 Privacy-Preserving Federated Machine Learning

45. In the following, we describe the works and results for the previous phase of the Private Machine Learning (PML) track. The goal was to test different approaches of Federated Learning (FL) using open and public accelerometer data. The scope of this phase is to Investigate best practices and open source tools for distributed and collaborative Machine Learning (ML) training among multiple organizations in a low trust environment whilst mutually benefitting from the outcomes (the final model) or allowing safe 3rd party access.

46. In the experiments, multiple National Statistical Organizations (NSOs) from Canada (StatCan), Netherlands (CBS), Italy (Istat) and UK (ONS) took part in building a simulated environment to validate the concept of multi-party privacy preserving ML (PPML) for both training and inference. A distributed and containerized PPML architecture was built, utilizing FL in combination with other privacy enhancing techniques, such as additive homomorphic encryption (HE), to train a neural network model on isolated lifestyle data collected by smart and wearable devices.

3.1.1 Methodology

47. A simulated environment was created by using open source tools and libraries to recognize and classify human activities into multiple categories based on publicly available accelerometer data collected from smart and wearable devices (Reyes-Ortiz, et al.). To this end, we split the data into 4 different chunks each one to be privately held by each participating NSO (StatCan, ONS, Istat, and CBS). Each chunk is split on train and validation datasets. The assumption is that neither NSO has access to others' data but each one wants to collaboratively train a classification model for activity recognition. Each NSO can train a classification model, e.g. a multilayer perceptron (MLP) locally with its local data. An additional assumption is that a Central Authority (CA)1 exists and contributes by averaging the locally trained models (weights) by each NSO (see fig. 3.1).

48. In detail, we used:

- Flower library to set up the clients (NSOs) and the CA for the FL setting.
- PyTorch to train and validate the MLP model.
- Numpy library for serialization/deserialization of the weights and tensor manipulations.
- Python-Paillier library for additive HE.
- Matplotlib for plotting the results.
- Pandas library for data wrangling.
- Python-Click library for command line interface.

[1] NSO/client and CA/server are indistinctly used.

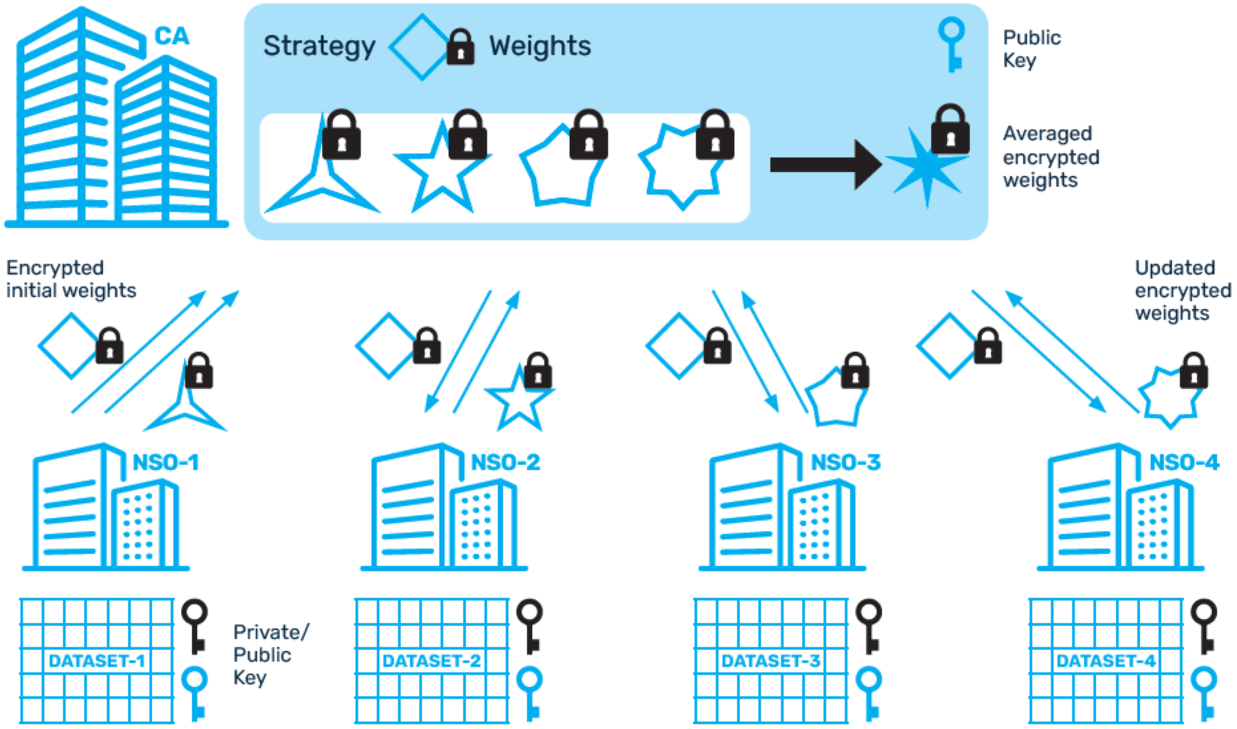


Figure 3.1: An overview of a distributed privacy preserving machine learning involving four National Statistical Organizations *(NSOs) and a Central Authority (CA). A combination of Federated Learning and additive Homomorphic Encryption is used to train a neural network to classify human activities based on publicly available accelerometer data collected from smart devices.*

3.1.2 Experiments

49. The data is about Human activity recognition using smart devices' accelerometer and gyroscope data, after pre-processing. The goal is to classify the data into 6 classes: WALKING, WALKING_UPSTAIRS, WALKING_DOWNSTAIRS, SITTING, STANDING, LAYING. A local neural network (MLP with linear layers and ReLU activations) is used for the purpose of classification.

50. Four scenarios with incremental difficulty to implement were studied. The main characteristics are summarized in table 3.1.

51. By default the Flower library allows one client to send the initial weights (thus the architecture of the model) to the server (CA). This was the case in scenarios 1, 2, and 4. Then we performed custom implementations to allow the CA to set up the architecture and initial weights. In this way, the learning process is controlled by the CA (scenario 3).

Scenario	Clients	Data	FL Aggregation Strategy	Notes
1	2 NSOs	Same data	FedAvg (MacMahan et al.)	Initial model and hyperparameter are sent by one NSO to the CA.
2	4 NSOs	Split (in different chunks)	FedAvg	Saved aggregated model and validation.
3	4 NSOs	Split	FedAvg	Initial model and hyperparameters are sent by the CA.
4	4 NSOs	Split	Encrypted FedAvg	Initial model and hyperparameters are sent by one NSO.

Table 3.1: Implemented Federated Learning Scenarios

52. The vanilla implementation for the aggregation of models is called FedAvg (MacMahan et al., 2017). Once the CA receives all the trained models, it simply averages the weights and sends back the updates to the clients for additional rounds of training.

53. A more interesting case is scenario 4, where each NSOs holds a common set of private and public keys[2] set using the Paillier cryptosystem (Paillier, 1999). This public key cryptosystem is homomorphically additive, that means that given ciphertexts $\mathbf{ct_1}$, $\mathbf{ct_2}$, …, $\mathbf{ct_n}$, we can compute $\mathbf{ct} = \mathbf{ct_1} + \mathbf{ct_2} + \ldots \mathbf{ct_n}$ and if we decrypt $\mathbf{ct}$ using the private key we will get the result of the sum of the plaintexts corresponding to each ciphertext. Then using Paillier HE, the CA which only holds the public key can easily compute the average of the weights without knowing anything about the plaintext weights, the motivation is to add an additional layer of privacy. Beyond Paillier HE, we implemented the serialization of the ciphertexts to be ready to be sent over the internet without losing its meaning. In that way, clients and the CA can deserialize, use and manipulate the ciphertexts. It is important to note that each NSO can decrypt the model and train locally on the clear, so NSOs perform encryption/decryption and the CA performs the addition homomorphically on ciphertexts. Another remark is that the final encrypted model was sent to each NSO to perform the validation.

3.1.3 Results

54. Each NSO holds its own train and test datasets. Local training is performed with the train dataset in the clear and after attaining a certain number of epochs the local encrypted model weights are sent to the CA in serialized form. Then, when all the encrypted weights are received, the CA deserializes them and performs the average to obtain the averaged encrypted weights (see fig. 3.1). Finally, these weights are sent to each NSO that decrypts them and

[2] Key exchange is possible on a cryptographic key management system (CKMS). We assume that the keys were already shared, thus considering a CKMS is out of scope for this project.

computes the accuracy and the loss using the test dataset. All this process is called a *round of training*.

55. In figure 3.2, we show plots for test accuracy and loss by round for each NSO. For this case, after 25 rounds the training is stopped. Plots indicate that the model accuracy is increasing (and loss is decreasing), thus we achieved our goal of training a model using FL.

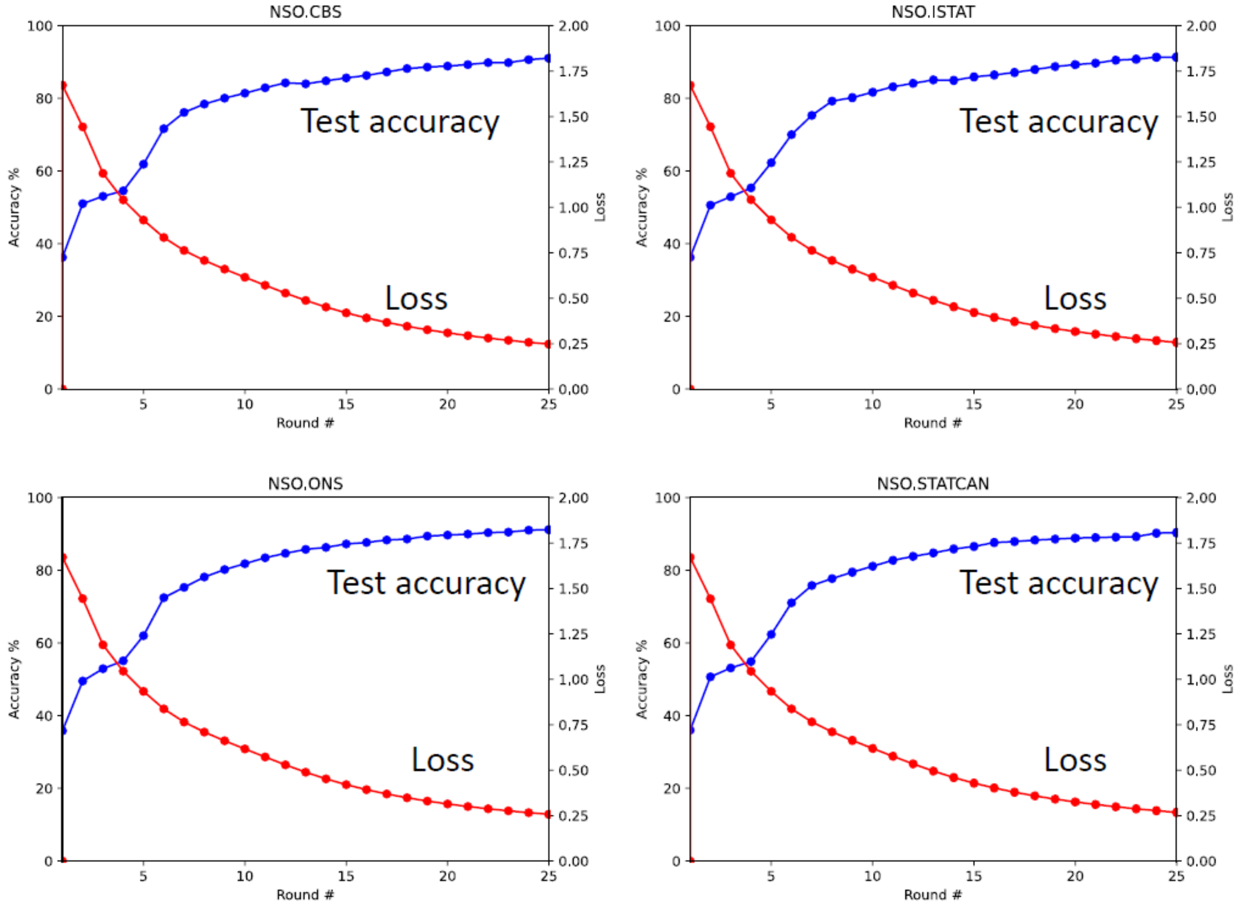


Figure 3.2: Test accuracy and loss for each NSO vs number of rounds of training

56. Two kinds of experiments were performed, a local test where all the clients and the CA reside on the same computer. A network test where all the clients ran on computers located in each country and the CA ran in the Amazon Web Services cloud.

57. More results and details of this project can be found at the following URL: https://unece.org/sites/default/files/2021-11/Saeid%20Molladavoudi_Private%20machine%20learning.pdf

3.1.4 Conclusions

58. This experiment had a simplified scope and was performed in a simulated environment. We were able to train a model for activity recognition under a FL setting. To this end, we considered different scenarios with increasing levels of customization. For the last experiment, we implemented serialization and deserialization of Paillier ciphertexts, and the encrypted averaged weights computation. It is worth noting that the results from scenarios 2 and 4 are

practically equivalent with the difference on the runtime overhead given by the encryption/decryption and serialization/deserialization processes[3].

59. This work can be extended by applying different aggregation strategies, models, and by using other frameworks for FL. Differential privacy can be added to help with the output privacy. Finally, we did not perform hyperparameter optimization for the initial model and architecture.

3.2 Defending Against Membership Inference Attacks with Differential Privacy on a Linked Dataset

60. We assume a scenario in which a NSO wants to offer a Privacy Enhancing Technique (PET)-based remote analytics service, e.g. a predictive model trained on its own internal data. The service could be used by another party (public, private or another NSO) that needs to use NSO's private data while maintaining the appropriate privacy requirements. The consuming party may also use their own confidential data alongside the NSO's data.

61. More specifically, in our use case, a university wants to infer whether a student will pass or fail using demographic data owned by the NSO alongside their private data. This allows the university to predict how to prepare for the coming years (number of classrooms to prepare, number of professors, etc.). The university hasn't access to the sensitive socio-demographic data. Only by using PETs can the university have access to this analytical output. The goal is to build a private ML model using the linked data, or a synthetic representation, to analyze the impact of PETs when used to protect the ML model from various attacks and the training data from being identified or reconstructed.

62. Note that this use case does not follow responsible ML principles and it is meant to analyze the impact of differential privacy (DP) and various ML attacks. The results can be translated to similar use cases in which responsible ML principles are followed. Each of these results can also be further optimized to improve the overall results.

63. In this use case we have two datasets: a Portuguese university's student performance dataset for Mathematics; and a NSO's demographic one; they have a common anonymized key that allows training a predictive model with the use of PETs. The linked dataset contained a total of 395 samples. We focused our work on model attacks and not on the datasets so we assumed that such linkage between the datasets already exists. The university wants to train a private ML model using data owned by the other party (NSO); the goal is to train a predictive model in a privacy preserving manner with the aim to provide an inferential model as output. We assume there is a partial trust relationship between the two parties (Honest but Curious scenario).

64. We have worked on a specific attack model named membership inference and on its mitigation methods during ML training and/or to inputs. Finally, we made the following assumptions:

- the dataset linked for training is secure and cannot be accessed

[3] Since the weights are bundled in an array of arrays (or tensors) each one corresponding to a layer of the MLP, we experimented encrypting all these layers or just the output layer to speed up the computation.

- the ML model is partially accessible with unrestricted access to queries
- relevant attacks are membership inference, model reconstruction and data linkage.

3.2.1 Scenario 1 - Differentially Private Data and Differentially Private Training

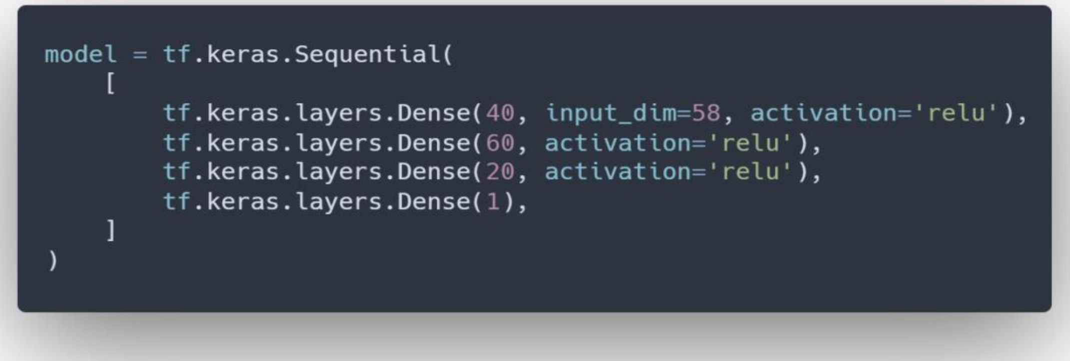
65. In this section we address the question of the relative efficiency of different methods for applying DP, by comparing the predictive performance of two methods with the same privacy budget. We consider two approaches to achieving (ε,δ)-Differential Privacy in the following outputs:

- Using non-privatized data, but applying an (ε,δ)-DP optimization method to the model
- Creating (ε,δ)-DP synthetic data and then applying a classical (non-private) model

66. The post-processing Theorem of Differential Privacy ensures that the same privacy guarantee will hold, but we anticipate accuracy will vary.

3.2.1.1 Framework

67. Using the Portuguese Mathematics dataset with 395 samples, the task was to predict a binarized test score from all other variables in the dataset (58 features including binary dummies derived from categorical variables). The binary target variable was derived by splitting an integer test score at a threshold value. The base rate and mean of the target variable was approximately 0.67. We use the same feed-forward neural network (NN) for each method. The network has three hidden layers with 40, 60 and 20 nodes, respectively and Rectified Linear Unit (ReLU) activations for each layer (see figure 3.3).



```
model = tf.keras.Sequential(
    [
        tf.keras.layers.Dense(40, input_dim=58, activation='relu'),
        tf.keras.layers.Dense(60, activation='relu'),
        tf.keras.layers.Dense(20, activation='relu'),
        tf.keras.layers.Dense(1),
    ]
)
```

Figure 3.3: Neural network design used for testing

68. The first experiment uses the original dataset, but applies noise to the optimisation of the model, ensuring that the fitted model cannot have memorized too much information from any single row of the data. This is achieved using the Differentially Private Stochastic Gradient

Descent (DP-SGD) algorithm (Abadi et al., 2016) implemented as part of the `tensorflow-privacy` package[4] (Abadi et al., 2015).

69. The second experiment replaces the original data with a synthetic dataset that has been synthesized by a differentially-private algorithm. In this experiment we use MST (McKenna et al., 2021), which won the 2018 NIST DP Challenge. The MST synthesis method discretises each variable of the original data into a number of bins (set by the researcher), and then estimates multivariate histograms of the most important data margins with appropriate noise added to ensure the mechanism satisfies DP, before reconstructing a synthesized dataset from the noisy histograms.

70. The main evaluation metric for the two methods is the validation-set predictive accuracy. One fifth of the data is set aside for validation, while the other four fifths are used for training the model in each experiment.

3.2.1.2 Results

Experiment 1: Original data with DP-SGD training

71. In this experiment we used the original data. The DP was applied to the optimisation of the NN model. We varied $\varepsilon \in \{1,10,1000,10^6\}$. Table 3.2 shows that the validation set accuracy achieved by the 'vanilla', non-DP algorithm was 90%. When ε was set to a very high value, the DP-SGD algorithm achieved a validation-set accuracy of 84%. However, for smaller ε values the accuracy achieved was worse than randomly guessing the value of the binary target variable.

72. Table 3.2 also shows the training-set accuracy for $\varepsilon \leq 1000$ was worse than random guessing; in practical settings users are normally advised to use values of ε on the scale 1 - 10. This suggests that, rather than fitting the model in a way that failed to generalize, the algorithm failed to train the model at all. The trace plots in Figure 3.4 confirm that the predictive accuracy of the model did not improve with each epoch, on the training set or the validation set.

Accuracy	Training	Validation
DP $\epsilon = 1$	0.54	0.47
DP $\epsilon = 10$	0.59	0.65
DP $\epsilon = 1{,}000$	0.56	0.52
DP $\epsilon = 10^6$	1.00	0.84
Non DP	1.00	0.90
Base rate	0.67	0.67

Table 3.2: Accuracy comparison when training with DP-SGD and various ε values

[4] TensorFlow Privacy | Responsible AI Toolkit

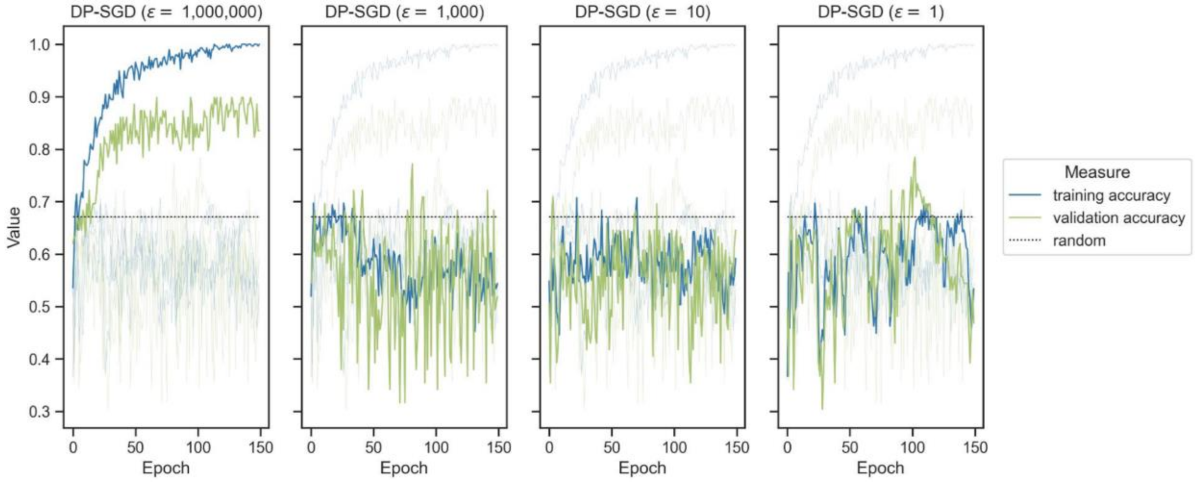


Figure 3.4: Accuracy plots when training with DP-SGD and various ε values

Experiment 2: DP Synthesised Data with non-DP training

73. In this experiment we created (ε,δ)-DP synthetic data, and trained the NN without DP. We varied ε ∈ {1, 10, 1000}.

74. The dataset is synthesized three times: with 1, 10, and 1000 ε budget, respectively. The quality of the synthetic data on univariate margins (ε=1) was reasonably high (figure 3.5). However the synthesizer struggled to accurately estimate correlations with lower ε (figure 3.6).

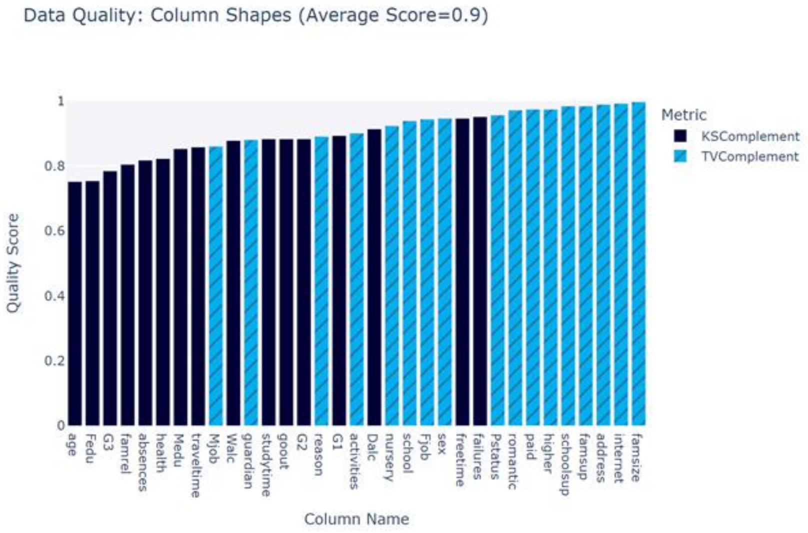
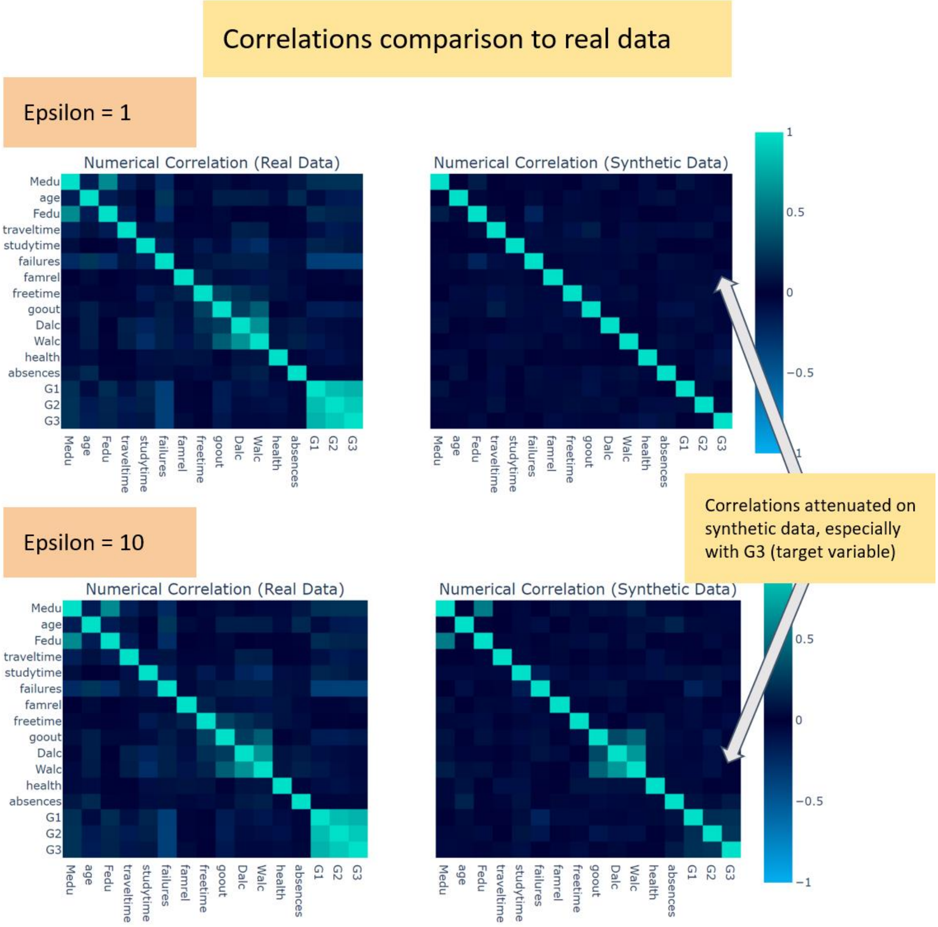


Figure 3.5: Synthetic data quality on univariate margins



Correlations comparison to real data

Epsilon = 1

Epsilon = 10

Correlations attenuated on synthetic data, especially with G3 (target variable)

Figure 3.6: Synthetic data correlations with lower ε values

75. Table 3.3 exhibits that the validation-set accuracy on both the original data, and the dataset with ε = 1000 were close. The accuracy achieved by the 'vanilla' model was 87%, whereas the accuracy achieved by the synthetic data was 84%. For ε ∈ {10,1}, however, the validation-set accuracy was lower than random guessing on the original data. This is shown in Figure 3.7, where the horizontal black line is the accuracy achieved by random guessing on the original data.

Accuracy	Training	Validation	Original data
DP $\epsilon = 1$	1.00	0.52	0.60
DP $\epsilon = 10$	1.00	0.68	0.67
DP $\epsilon = 1{,}000$	1.00	0.84	0.77
Original data	1.00	0.87	NA
Base rate	0.67	0.67	0.67

Table 3.3: Accuracy comparison when training with differentially private input data

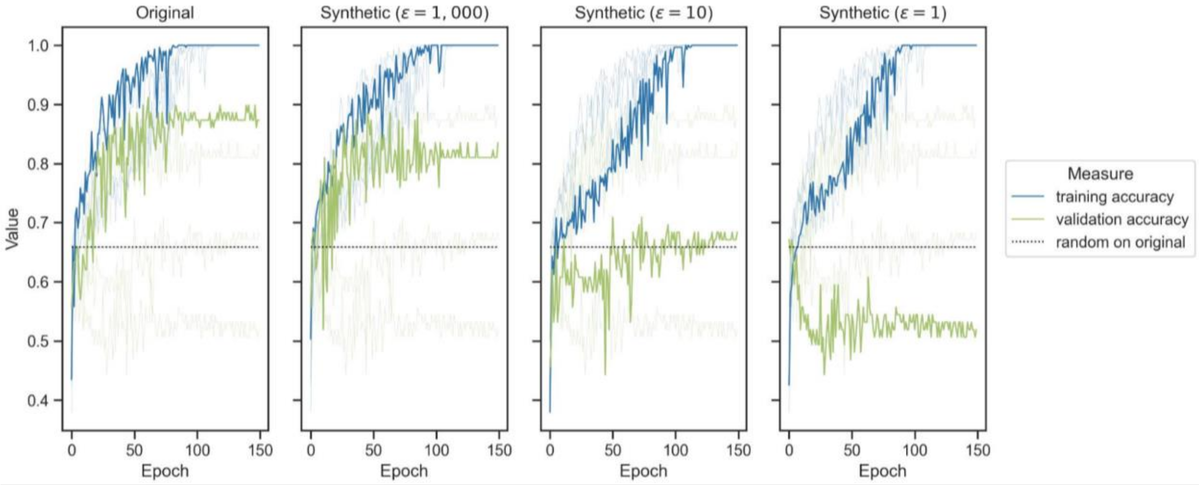


Figure 3.7: Accuracy plots when training with differentially private input data

76. The TensorFlow model was fitted to each of the three synthetic datasets and to the original data. The model fitted to the synthetic data seems to perform reasonably on real data when ε is high.

3.2.1.3 Conclusions

77. For this particular task, neither approach (DP-model, or DP-data) produced performant models at values of ε that would be viable for real-world applications.

78. This limitation is likely at least in part due to the small size of the data set - on which we might not expect deep learning to be effective. Future experiments could seek to replicate this experiment on data sets at larger scales.

79. For DP-synthetic data, correlation plots indicate the algorithm failed to retain relationships between key independent variables and the target variable. This supports findings of the ONS/Alan Turing Institute (Houssieau et al., 2022)[5] that unsupervised synthetic data generators can fail to retain key relationships. This can be mitigated by creating synthetic data sets that are supervised to retain particular relationships. Future work could look at how much performance gain can be achieved by curating synthetic data specifically for the prediction task at hand. Other investigations into DP have not been considered within the scope of this work, such as residual privacy risks with synthetic data generated with DP, but these can be explored in future works.

3.2.2 Scenario 2 - Investigation of Membership Inference Attacks

80. Within the HLG-UNECE PML Track 2 project, we addressed the output privacy issues that arise when a trained ML model is distributed in some way or is deployed.

81. Research on these topics is important; the increased use of ML applications distributed on cloud services and the tendency to share trained models have made it easier for attackers to compromise the privacy of the models. Furthermore, with the increasingly pervasive use of

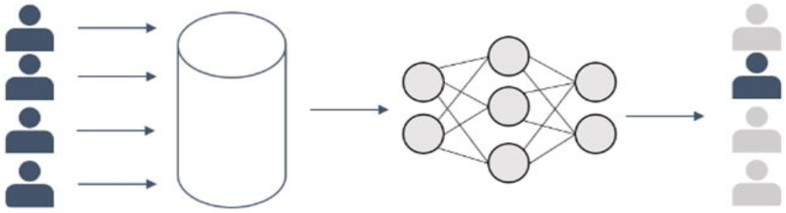
[5] [2211.06550] TAPAS: a Toolbox for Adversarial Privacy Auditing of Synthetic Data (arxiv.org)

Internet of Things (IoT) devices where ML algorithms are often embedded in these devices, it is more important than ever to protect the data used to train these models. This has led to growing concerns about the possibility of these attacks compromising the privacy of people whose data is used in the training process.

82. Among the different privacy attacks which target the outputs of a trained ML model, we can mention:

- Model inversion attacks use a ML model to infer sensitive information about its training data. The basic idea is to use model predictions to reverse engineer the original training data. In particular, the attack aims to estimate some sensitive characteristics of the inputs.
- Membership inference attacks (MIAs) are a type of security attack that aims to determine whether data from a specific individual has been used to train a ML model or not. In this use case we addressed the MIA type.

83. A MIA allows an adversary to query a trained ML model to predict whether or not a particular sample was in the model's training dataset. In practice, given a trained ML model and some data point, an MIA decides whether this sample was part of the model’s training set.

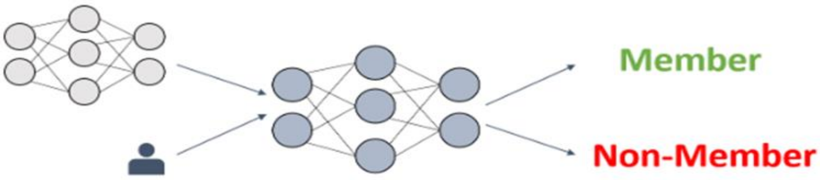


https://fraboeni.github.io/files/2021-01-24-membership-inference/pic1-membership-inference.png

Figure 3.8: Identifying individual samples from the training set

84. MIAs can be used to violate individual privacy; for example, imagine you are in a clinical context; you may have a ML model to predict an adequate medical treatment for cancer patients. This model, naturally, needs to be trained on the data of cancer patients. Hence, given a datapoint, if you are able to determine that it was indeed part of the model’s training data, you will know that the corresponding patient must have cancer.

85. Most MIAs work by building a binary meta-classifier F_{attack} which, given a model F and a data point x_i decides whether or not x_i was part of model training sample X.



https://fraboeni.github.io/files/2021-01-24-membership-inference/pic2-meta-classifier.png

Figure 3.9: An example of what a MIA attempts to learn

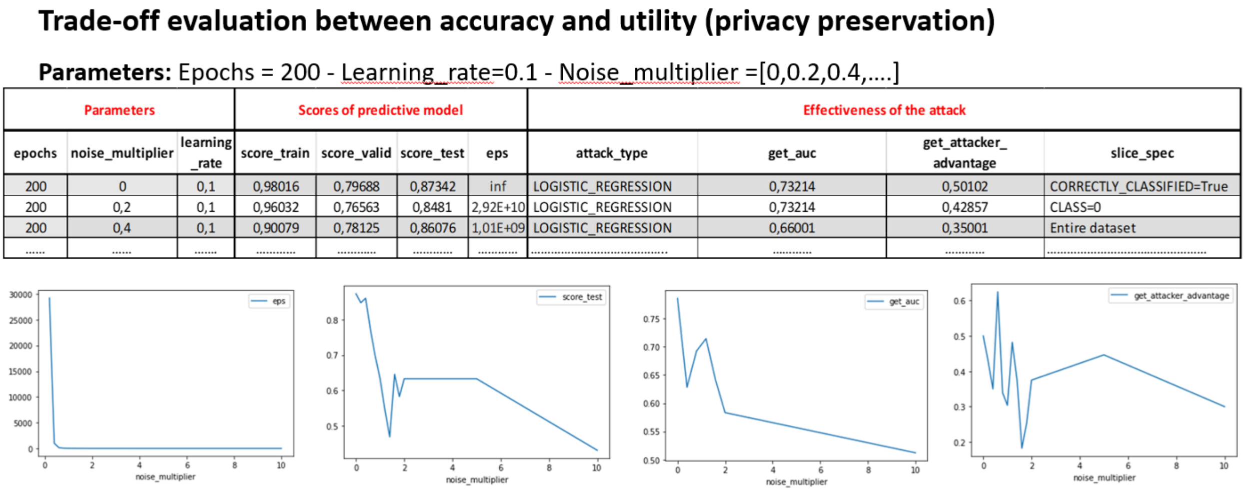
86. To train the binary meta-classifier, shadow models are built that imitate the behavior of the original ML model. Shadow models use training datasets known to the attacker or that are generated by the attacker. By exploiting the knowledge of the input and output data of the shadow models, the binary meta-classifier is trained.

87. There are several tools to implement MIAs such as IBM-ART framework[6] and TensorFlow Privacy's Membership Inference[7], moreover implementations can also be custom-built following the designs from research papers. In this specific use case we applied TensorFlow Privacy to our dataset to simulate MIAs to our predictive model.

88. Within this scenario we test the efficiency of MIAs and how DP can protect against the attacks. Furthermore, we explore how overfitting affects the performance of MIAs.

3.2.2.1 Results

89. The results of the scenario above are presented in the following figure that shows the trade-off evaluation between accuracy and utility (privacy preservation) for some dimensions. The overall performance of the MIA is also compared to when various privacy budgets are utilized.



Trade-off evaluation between accuracy and utility (privacy preservation)

Parameters: Epochs = 200 - Learning_rate=0.1 - Noise_multiplier =[0,0.2,0.4,….]

Parameters			Scores of predictive model				Effectiveness of the attack			
epochs	noise_multiplier	learning_rate	score_train	score_valid	score_test	eps	attack_type	get_auc	get_attacker_advantage	slice_spec
200	0	0,1	0,98016	0,79688	0,87342	inf	LOGISTIC_REGRESSION	0,73214	0,50102	CORRECTLY_CLASSIFIED=True
200	0,2	0,1	0,96032	0,76563	0,8481	2,92E+10	LOGISTIC_REGRESSION	0,73214	0,42857	CLASS=0
200	0,4	0,1	0,90079	0,78125	0,86076	1,01E+09	LOGISTIC_REGRESSION	0,66001	0,35001	Entire dataset
……	……	…….	…………	…………	…………	…………	…………………………………….	…………	…………	………………………………………………

Figure 3.10: Trade-off between accuracy and utility with DP and MIAs

90. In the table within figure 3.10, the following three groups of values are presented as they are utilized by the TensorFlow Privacy library: parameters, scores of predictive model and effectiveness of the attack. The first group, in the first three columns, includes the number of epochs used for training, the noise multiplier and the learning rate values used to train the model. These will be discussed below. In the second group (columns 4, 5, 6, and 7) the scores of the training set, validation set and test set are presented alongside the ε values used. In the last group (columns 8, 9, 10, and 11) there are some metrics related with the effectiveness of

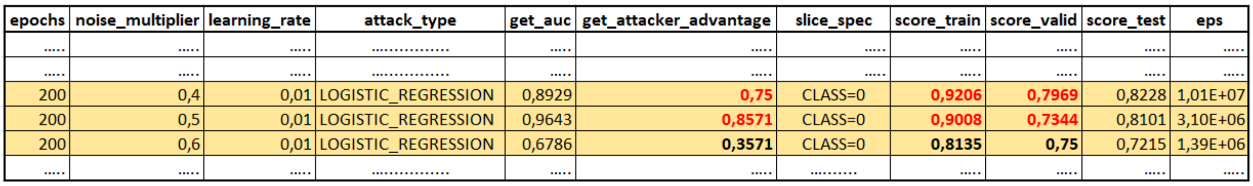
[6] https://github.com/Trusted-AI/adversarial-robustness-toolbox

[7] https://github.com/tensorflow/privacy/tree/master/tensorflow_privacy/privacy/privacy_tests/membership_inference_attack

the attack: the type of attack utilized, the area under the curve (AUC) values, the attack advantage value and the portion of the dataset on which the attack has been performed.

91. The bottom of the figure 3.10 displays four different plots to highlight the results. The first, on the left, shows the relationship between the noise_mulitplier and the ε (eps). The noise_multiplier is the amount of noise fed into the original model (the one that predicts if a student passes or fails) during the training phase, to preserve the privacy of the model input data. Eps is the privacy budget: the lower ε is, the more noise is injected during the training. The second plot shows the performance on the test set, i.e. the measure of accuracy of the trained model on unseen data. It can be observed that the greater the noise entered into the training of the model, the lower the testing score. In the third and fourth plots we measure the performance of MIAs. We consider two evaluation metrics: get_auc (area under the curve) and get_attacker advantage; the greater the "auc" the greater the effectiveness of the attack. The get_attacker_advantage measures how advantageous the attack is; the smaller the advantage of the attacker, the less effective the attacks will be.

92. The vulnerability of a model to MIAs depends on a set of factors, including overfitting, classification problem complexity, in-class standard deviation and the type of the ML model targeted. We will look at an example of overfitting below to illustrate how models that do not fit the training data too tightly are less vulnerable. Overfitting occurs when the model cannot generalize well to unseen data and fits too closely to the training dataset; in this case we can observe a high difference between the validation score and the training score.



epochs	noise_multiplier	learning_rate	attack_type	get_auc	get_attacker_advantage	slice_spec	score_train	score_valid	score_test	eps
.....	.....	.....	....	.....	.....	.....	.....	.....	.....	.....
.....	.....	.....	....	.....	.....	.....	.....	.....	.....	.....
200	0,4	0,01	LOGISTIC_REGRESSION	0,8929	**0,75**	CLASS=0	**0,9206**	**0,7969**	0,8228	1,01E+07
200	0,5	0,01	LOGISTIC_REGRESSION	0,9643	**0,8571**	CLASS=0	**0,9008**	**0,7344**	0,8101	3,10E+06
200	0,6	0,01	LOGISTIC_REGRESSION	0,6786	**0,3571**	CLASS=0	**0,8135**	**0,75**	0,7215	1,39E+06
.....	.....	.....	....	.....	.....	....	.....	.....	.....	.....

Table 3.4: Example of how overfitting impacts the performance of MIAs

93. When the original model, i.e. the model that classifies whether students pass or fail, is overfitted (see the first two rows of the table 3.4), there is an advantage for the attacker because the model fitted too closely to the training dataset. This makes it easier for the MIA to identify which samples have been used for training (see the get_attacker_advantage values). Instead, in the third row, where we have not detected overfitting, we observe that the *get_advantage_attacker* (which represents the effectiveness of the attack) decreases.

94. In conclusion, to help protect against MIAs, it is important to utilize DP and to ensure that the model is not overfit during the training process. This can result in a decrease in performance, but reduces the effectiveness of MIAs to better protect the privacy of the training data. As with the majority of privacy work, a balance between the privacy needed and the target performance must be established based on how the model will be used.

3.2.3 Scenario 3 - In-Depth Investigation of Membership Inference Attacks and Differential Privacy

95. Following the investigations of DP applied during training, of DP applied to the input data, and of differentially private training against established MIAs, this final exploration will highlight the specific impacts DP can provide to defend against MIAs. Specifically, we have

programmed the specified MIA from a classic paper on the attack to observe the defenses provided by DP in more detail than what is output from the libraries used above (Shokri et al, 2017). This helps identify key points in how DP can protect against MIAs, such as how well it protects training samples from being correctly identified for both class labels.

3.2.3.1 Membership Inference Attack Implementation

96. Following the implementation from its paper (Shokri et al, 2017), we have implemented the MIA as follows. First, we utilize the proposed model-based synthesis algorithm to generate synthetic datasets to use when training shadow models. This generates complete samples, with pass or fail class labels, by utilizing the probability outputs from the original model to determine when a synthetic sample is considered to likely be a member of the target class. The likelihood is determined by a set threshold chosen by the programmer. Shadow models are imitations of the original model which are trained with the synthesized datasets, where each shadow model receives a unique training and testing set.

97. There will be k shadow models utilized, where the i^{th} shadow model is designated by k_i and a dataset attached to a shadow model k_i is designated as dataset d_i. These k shadow models are initialized, trained with their corresponding training set, and tested with their corresponding test set (to ensure adequate predictive performance). For each of the k datasets used to train and test the k shadow models, each sample within the datasets is assigned a class label. A sample of dataset d_i is assigned a class label of 1 if that sample has been utilized to train the shadow model k_i. Otherwise, a sample is assigned a class label of 0 (which denotes that it has been used for testing, but not for training). Then each labeled sample within the dataset d_i is input through shadow model k_i to receive the appropriate prediction vector from the trained shadow model.

98. With each synthetic sample now containing a class label of whether it has been used for training and with the pass/fail prediction vectors for each sample, the datasets are split into new unique datasets with only the prediction vectors as the features and the train/test class label as the class to predict (the original pass/fail class label is tracked but will be discarded afterwards). There will be one of these new datasets for each original class label (i.e. pass or fail). Hence, within the context of the work performed, each sample of each dataset d_i will be placed within either d_{pass} if the sample is predicted to pass or d_{fail} if the sample is predicted to fail. These two datasets will be used to train the final attack models, where there will be one attack model for each original class label. Attack model Φ_0 represents the model which determines if samples of class 0 (i.e. a failing grade) have been used for training the target model and Φ_1 represents the same, except for samples of class 1 (i.e. a passing grade).

99. When trained, each attack model will learn how to distinguish the differences in prediction vectors such that it can determine whether a sample has been used for training. These attack models can then be applied on more synthetic data to determine the likelihood that the generated samples have been used to train the original model being attacked. This can result in the training set being reconstructed which is a clear privacy violation.

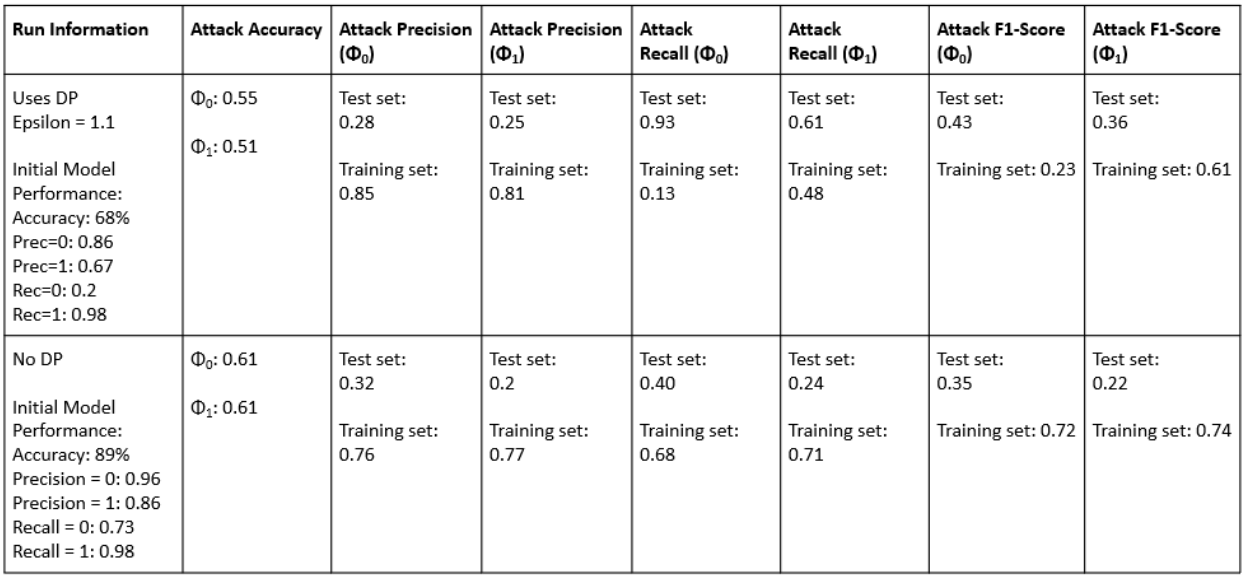
3.2.3.2 Simulation Configuration and Results

100. Through a Python program, the above attack is simulated and utilized against two ML models, one trained normally and the other trained with DP. The attack itself is not fully optimized within this simulation, however the selected attack parameters still provide sufficient

outputs to understand how differentially private training helps protect against the attack. To understand how effective the attack is against both models, the attack accuracies, attack precisions, attack recalls, and attack f1-scores are tracked for both attack models Φ0 and Φ1. Furthermore, the precision, recall, and f1-score values are output for both when the sample is actually part of the training set and when the sample is not a member of the original training set.

101. These results are exhibited in table 3.5 for when the attack has been applied to a differentially private model Φ_{dp} with a privacy budget of 1.1 and when the attack has been applied to a non-differentially private model Φ with the same architecture as Φ_{dp}. Within the table, the run information of the model being attacked is first displayed. This highlights the accuracy, precision, and recall values of training the original models Φ_{dp} and Φ with the actual dataset. With this information, the effects of applying DP in the training can be observed. Note that sufficiently high performance is important since both models will be queried to generate the synthetic datasets for the attack. With a poor initial model, the synthetic data's quality will suffer.

102. The following columns highlight the actual performance of the attack, for both attack models Φ_0 and Φ_1. The attack accuracy represents the overall performance of how well each attack model can correctly determine if a sample is a member of the training set. A high attack accuracy is preferred when accompanied with higher precision, recall, and f1-score values. The attack precision columns denote how precise the predictions are for a given attack model. A higher value is best when the recall values are also sufficiently high. The attack recall columns display the percentage of correct predictions for whether data is or is not part of the training dataset. A higher recall value is good when matched with a sufficient precision value. Finally, the attack f1-score represents how well the attack balances its recall and precision values. The higher this value is, the better the attack is performing in general. Note that the original dataset used to train the initial models are run through the attack models and the results are compiled within table 3.5.



Run Information	Attack Accuracy	Attack Precision (Φ_0)	Attack Precision (Φ_1)	Attack Recall (Φ_0)	Attack Recall (Φ_1)	Attack F1-Score (Φ_0)	Attack F1-Score (Φ_1)
Uses DP Epsilon = 1.1 Initial Model Performance: Accuracy: 68% Prec=0: 0.86 Prec=1: 0.67 Rec=0: 0.2 Rec=1: 0.98	Φ_0: 0.55 Φ_1: 0.51	Test set: 0.28 Training set: 0.85	Test set: 0.25 Training set: 0.81	Test set: 0.93 Training set: 0.13	Test set: 0.61 Training set: 0.48	Test set: 0.43 Training set: 0.23	Test set: 0.36 Training set: 0.61
No DP Initial Model Performance: Accuracy: 89% Precision = 0: 0.96 Precision = 1: 0.86 Recall = 0: 0.73 Recall = 1: 0.98	Φ_0: 0.61 Φ_1: 0.61	Test set: 0.32 Training set: 0.76	Test set: 0.2 Training set: 0.77	Test set: 0.40 Training set: 0.68	Test set: 0.24 Training set: 0.71	Test set: 0.35 Training set: 0.72	Test set: 0.22 Training set: 0.74

Table 3.5: Membership Inference Attack Performance Comparison

103. Table 3.5 illustrates that the MIA performs worse on the target model when DP is used in the training process of the target model compared to when the target model is trained without DP. This is clear due to the overall metrics of the attack performing worse when DP is used.

104. The model being attacked has its performance presented in the run information column. Here, the use of DP results in an overall improvement for each evaluation metric. Of note, the accuracy and precision values of Φ_{dp} have a moderate decrease when compared to the traditionally trained model Φ. By having worse performance, Φ_{dp} inherently provides some protection against the selected attack since the model synthesis approach utilized to obtain synthetic data will not generate as accurate samples compared to when Φ is attacked.

105. When analyzing the attack accuracies of both attack models Φ_0 and Φ_1, DP reduces the attack's accuracy in identifying training and non-training samples to a probability close to that of a coin flip. This is worse than not using DP, but what is most important to analyze is whether this means that the attack is unable to identify training samples, unable to identify non-training samples, or if it performs poorly at identifying both. The precision and recall values indicate that the use of DP in Φ_{dp} results in more samples being classified as not in the training set when compared to Φ. This is clear due to the large increase to the recall values for both pass and fail samples not in the training set and the corresponding low precision values.

106. Without DP used, the recall values are more balanced, but the training set samples are more frequently and precisely identified. The f1-scores exhibit the same patterns, where the f1-scores of identifying the training samples are much higher for the traditionally trained model Φ, especially for samples of class 0 (i.e. a failing grade). The non-training sample f1-scores are only moderately higher when DP is used. Overall, this indicates that more training samples are correctly identified without DP with a similar amount of non-training samples being precisely identified as when DP is used.

107. One note is that the differentially private model Φ_{dp} is still being exploited by the attack despite the attack being less effective. In particular, the small amount of training data identified is being very accurately identified since the precision remains high for a small subset of samples. Although this is better than having more of the training set being identifiable, this still leaves room for concern depending on the sensitivity of the data. In practice this will require an analysis of the attack performance and optimizations to the privacy budget used to provide an appropriate defense against the attacks. However, these results clearly highlight how DP can be used to help defend against MIAs.

3.2.4 Conclusions

108. Preliminary results of our experiments in a simulated environment proves the feasibility of distributed and federated analytics among organizations while protecting the privacy of isolated data sources. We have built a community of Statistical Offices in the area of privacy-enhancing technologies with links to open source community, industry and academia. Moreover, there is a direct link to sustainability, when it comes to collaboration among NSOs, namely novel ways of collaboration, driven by privacy requirements and technological constraints. However, in real scenarios, prior agreements among participating agencies on a standard data format and preprocessing steps on a case-by-case basis seem to be necessary before deployment of distributed ML on sensitive data.

Chapter 4 - Track 3: Open consultation

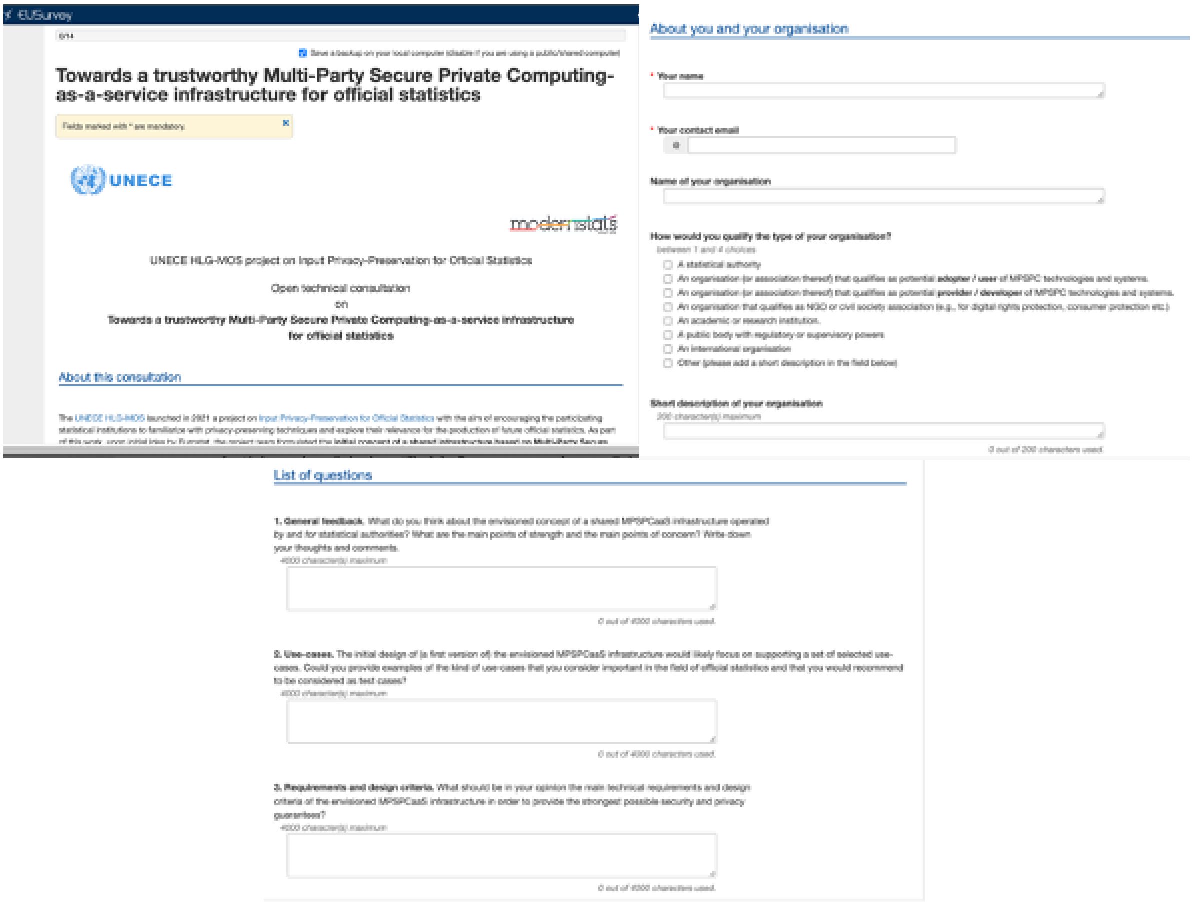
4.1 Overview

109. Through the series of technical discussions held during the project, the team reflected on the barriers and critical factors *on the side of potential adopters* that impede a wider adoption of Input Privacy Preservation (IPP) in statistical production as alternative to plain sharing of clear data. This triggered a wider reflection of possible strategies to promote faster adoption, leading the team to identify the pooling of scarce resources (human and financial) for the development of **shared infrastructures for privacy-preserving computation**, serving multiple organizations and supporting multiple use-cases, as a natural strategy to overcome many of the existing barriers. Based on such elaboration and building upon the initial impulse by Eurostat, the project team formulated the concept of a **Multi-Party Secure Private Computing-as-a-Service (MPSPCaaS)** infrastructure. A high-level outline of the MPSCPaaS concept is given in the following subsection.

110. The team recognized that the detailed design and implementation of the MPSPCaaS concept would need to face a wide range of technical challenges and non-technical issues that may go beyond the skills and expertise of the project team. Furthermore, it was not obvious that such an innovative concept would be well received by external stakeholders, and especially by prospective adopters and developers outside the project team. Therefore, In order to ensure that nothing is missed, the project team decided to launch an **informal and open consultation among experts and stakeholders**. The consultation was mainly targeted at:

- Privacy and security experts from both the technical and legal sides.
- Potential users of the envisioned MPSPC infrastructure, including but not limited to statistical authorities, public bodies and private companies.
- Digital activists and representatives of civil society (e.g., citizen associations).
- Researchers and developers in relevant fields.

111. The consultation was prepared by Eurostat (as Chair of Track 3) in the form of an online questionnaire on the EUSurvey platform made freely available by the European Commission. A short document (5 pages) was prepared and released publicly on the project website of the project to (i) explain the motivations and goals of the consultation; (ii) present the basic elements of the MPSPCaaS concept; and (iii) formulate a set of eight specific questions (plus one additional field open for free comments). Email invitations were sent to individual experts and organizations with the plea to forward the invitation further in their professional networks. The consultation was open on 15th October 2022. The response deadline was set initially to 30th November and later extended to 15th January 2023. Access to the response link required an EU Login account. The response field for each question allowed up to 4000 characters of free text. During this period a total of 20 responses were received. The analysis of the responses was analyzed and the key insight and points made by the respondents are summarized below in a dedicated subsection.



EUSurvey

6/14

☑ Save a backup on your local computer (disable if you are using a public/shared computer)

Towards a trustworthy Multi-Party Secure Private Computing-as-a-service infrastructure for official statistics

Fields marked with * are mandatory.

UNECE

modernstats

UNECE HLG-MOS project on Input Privacy-Preservation for Official Statistics

Open technical consultation
on
Towards a trustworthy Multi-Party Secure Private Computing-as-a-service infrastructure for official statistics

About this consultation

The UNECE HLG-MOS launched in 2021 a project on Input Privacy-Preservation for Official Statistics with the aim of encouraging the participating statistical institutions to familiarize with privacy-preserving techniques and explore their relevance for the production of future official statistics. As part of this work, upon initial idea by Eurostat, the project team formulated the initial concept of a shared infrastructure based on Multi-Party Secure

About you and your organisation

* **Your name**

* **Your contact email**

@

Name of your organisation

How would you qualify the type of your organisation?
between 1 and 4 choices

- □ A statistical authority
- □ An organisation (or association thereof) that qualifies as potential **adopter / user** of MPSPC technologies and systems.
- □ An organisation (or association thereof) that qualifies as potential **provider / developer** of MPSPC technologies and systems.
- □ An organisation that qualifies as NGO or civil society association (e.g., for digital rights protection, consumer protection etc.)
- □ An academic or research institution.
- □ A public body with regulatory or supervisory powers
- □ An international organisation
- □ Other (please add a short description in the field below)

Short description of your organisation
200 character(s) maximum

0 out of 200 characters used.

List of questions

1. General feedback. What do you think about the envisioned concept of a shared MPSPCaaS infrastructure operated by and for statistical authorities? What are the main points of strength and the main points of concern? Write down your thoughts and comments.
4000 character(s) maximum

0 out of 4000 characters used.

2. Use-cases. The initial design of (a first version of) the envisioned MPSPCaaS infrastructure would likely focus on supporting a set of selected use-cases. Could you provide examples of the kind of use-cases that you consider important in the field of official statistics and that you would recommend to be considered as test cases?
4000 character(s) maximum

0 out of 4000 characters used.

3. Requirements and design criteria. What should be in your opinion the main technical requirements and design criteria of the envisioned MPSPCaaS infrastructure in order to provide the strongest possible security and privacy guarantees?
4000 character(s) maximum

0 out of 4000 characters used.

Figure 4.1 - Screenshots of the online questionnaire

4.2 Multi-Party Secure Private Computing-as-a-Service concept

112. The traditional model of statistical production assumes that a single organization, namely the statistical authority, collects the whole input data and from there computes the desired output information, i.e., the final statistics, according to some data analysis methodology. Whenever the desired output information requires the integration/combination of different input data sets held by different organizations, the traditional solution is to arrange for an exchange of input data, either directly between the concerned institutions or with a Trusted Third Party (TTP). In so doing, the receiving party commits to certain terms of use (e.g., to use the data to extract solely the agreed information for the agreed purpose, to delete the data immediately afterwards, to secure the data against intrusions, etc.). The transmitting party and any other involved stakeholder, if any, must trust the receiving entity that it will abide by the agreed terms of data use because they have no technical means to enforce and verify the actual respect of these terms. This approach requires a strong trust relationship between the transmitting and receiving entities. It also amplifies the risks, since it increases the number of copies of the data and the number of actors that have access to the data. But exchanging the input data is a means towards the goal of computing the desired output, not an end. And it is not the only means available today: alternative solutions based on Privacy Enhancing Technologies (PET), and specifically technologies for Multi-Party Secure Private Computing

(MPSPC), allow today to compute the output statistics without necessarily disclosing the input data to any entities other than their respective data holders.

113. The appeal for MPSPC technologies in official statistics stems from the fact that several innovation trends in this domain point towards the need to combine/integrate data sets held by different organizations. For example, the prospective extension of official statistics towards “non-traditional” data sources relies on the possibility to (re)use new types of data generated for non-statistical purposes by other organizations, including public administrations and private companies[8]. In another direction, improving the quality of statistics referring to intrinsically cross-border phenomena (e.g., migration, international trade) requires the integration of data from different countries. These trends concur to increase the appetite for integrating/combining data held by multiple actors. Responding to such increasing demand with the traditional paradigm of data exchange may not be the most effective option in all cases, as any new copy of the data that is passed to another organization creates additional risks and calls for additional protection costs. This motivates the search for alternative models to execute inter-organization computation that do not require direct data exchange.

114. Setting up a robust MPSPC solution requires investments, capacity and also specialized skills on the side of potential adopters. Not all statistical institutions may have the internal resources and/or the necessary knowledge to develop, deploy and maintain their own solutions, and anyway the costs might be disproportionate compared to the expected benefit. The cost factor may discourage adoption wholly or drive towards adoption of sub-optimal solutions with less-than-maximum levels of security and robustness. Furthermore, interoperability may not be guaranteed among solutions developed independently by different institutions.

115. The concerns about costs, robustness and interoperability led the project team[9] to elaborate the vision of a shared MPSPC infrastructure, developed and operated by a network (or consortium) of statistical institutions and then made available on demand to execute computation based on the MPSPC paradigm. As in many other areas of Information Technologies, the basic idea is to decouple the development (and maintenance) from the utilization of the prospective MPSPC infrastructure. This allows to pool together resources and expert knowledge during the development phase, increasing cost-effectiveness and ultimately enabling the achievement of very high levels of robustness and security guarantees, based on state-of-the-art technologies and design criteria.

116. The shared MPSPC infrastructure developed in this way could then be used on demand by statistical institutions and by their partners (e.g. external data providers). This model was named **MPSPC-as-a-service** (MPSPCaaS for short) in order to highlight that what is provisioned to potential users is a (multiparty, secure, private) computation service rather than a computation infrastructure. The *‘servitisation’* of MPSPC is instrumental in providing a cost-effective ready-to-use alternative to direct data exchange, thus accelerating the widespread

[8] See e.g. the Final Report of the Expert Group on Facilitating the use of new data sources for official statistics, June 2022. Available from https://ec.europa.eu/eurostat/documents/7870049/14803739/KS-FT-22-004-EN-N.pdf

[9] Preliminary versions of this concept were presented by Eurostat at international conferences and workshops, see e.g. https://ec.europa.eu/eurostat/cros/content/privacy-enhancing-technologies-official-statistics-pet4os_en

adoption of the MPSPC paradigm in the field of official statistics, i.e., statistical authorities and their partners.

117. In an exemplary usage scenario, two organizations Px and Py have agreed to execute a particular operation f(Dx,Dy) on their respective input data sets Dx and Dy and let organization Pz learn the result Dz= f(Dx,Dy). In this simple example, Px and Py play the role of input parties while Pz is the output party. In practical cases, the same organization may play the role of input party and output party at the same time, i.e. Pz might coincide either with Px or with Py (but not with both, as otherwise the whole set of users would reduce to a single entity in control of all the input data and output result, with no necessity to consider MPSPC solutions). In the field of official statistics, the input data sets Dx and Dy often take the form of confidential micro-data and the domain of the function f lies in the union or intersection between the two input data sets Dx and Dy. Notably, as far as applications in official statistics are concerned, the function f is defined in advance, as part of the adopted statistical methodology, and does not constitute a business secret – an aspect that simplifies the operation compared to other business sectors where the function (model, algorithm) f is itself a confidential component. Also, we assume that the output party (typically a statistical authority) is entitled to receive the computation result Dz, regardless of whether or not it still contains privacy-sensitive or business-sensitive information. MPSPC allows performing such computation without requiring the input parties to share their data sets with any other single entity, be it the other input party, the output party or any other individual third party. What we have described here is a particular MPSPC task with parameters [Px,Py,Pz,Dx,Dy,f] to be configured and executed by the MPSPCaaS infrastructure along with – and independently from – other parallel tasks.

118. In the envisioned scenario, the institutions playing the roles of input parties Px, Py and output party Pz represent the group of users for this particular MPSPC task. In the envisioned MPSPCaaS, they would rely on the MPSPC functionalities made available by the shared infrastructure in order to let the computation result Dz=f(Dx,Dy) flow towards the output party Pz, with no other information disclosed to any other party. In practice, the group of users would connect to the MPSPCaaS infrastructure and configure a new MPSPC task taking advantage of the functionalities offered by the infrastructure. In this way, the marginal cost of configuring a new MPSCP task would be much smaller than the cost of setting up an ad-hoc MPSPC infrastructure dedicated to this specific task.

119. At an abstract level, the MPSPC infrastructure intermediating between the input and output parties may be seen as replacing a centralized Trusted Third Party (TTP), as shown in Figure 4.2. Indeed, if operation of the infrastructure would be such that a single entity would be technically able to control the whole computation process, the central controller would represent the single point of trust corresponding *de facto* to a TTP. In other words, a Secure Private Computing solution with centralized control would not be fundamentally different from the traditional model of data sharing with a TTP. In order to avoid that, at the heart of the MPSPC paradigm lies the requirement that no single entity should ever be technically capable to take over control of the process. Therefore, MPSPC operation must be designed so as to avoid any single point of trust. That means process control must be split (or divided) among a multiplicity of K>1 parties, which will be referred hereafter as processing parties[10]. In

[10] The abstract notion of *processing party* introduced here may possibly, but not necessarily correspond to the role of *computing parties* in secret sharing schemes. In fact, secret sharing

principle, K=2 processing parties would suffice to meet this formal requirement, but for increased robustness we will assume hereafter a minimum number of processing parties equal to K=3 or higher. Furthermore, in addition to the K processing parties with active control over the processing operation, additional entities may be foreseen to act as passive controllers, in order to increase the overall level of security and trust.

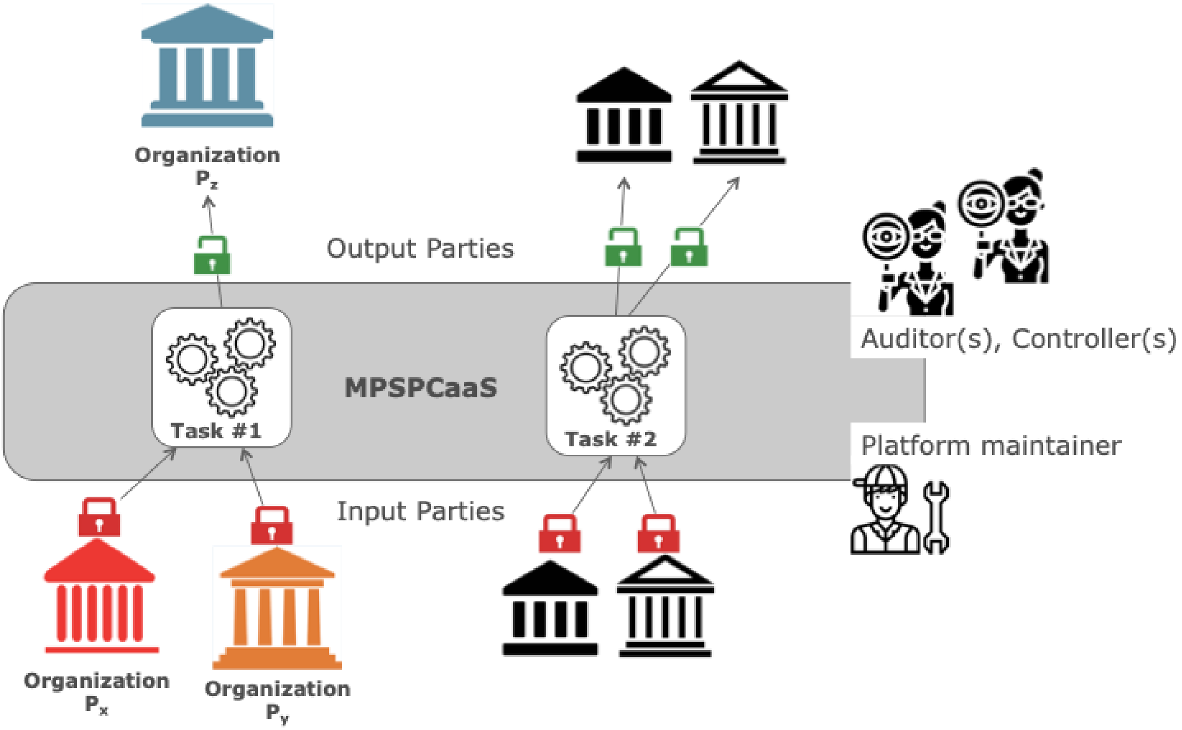


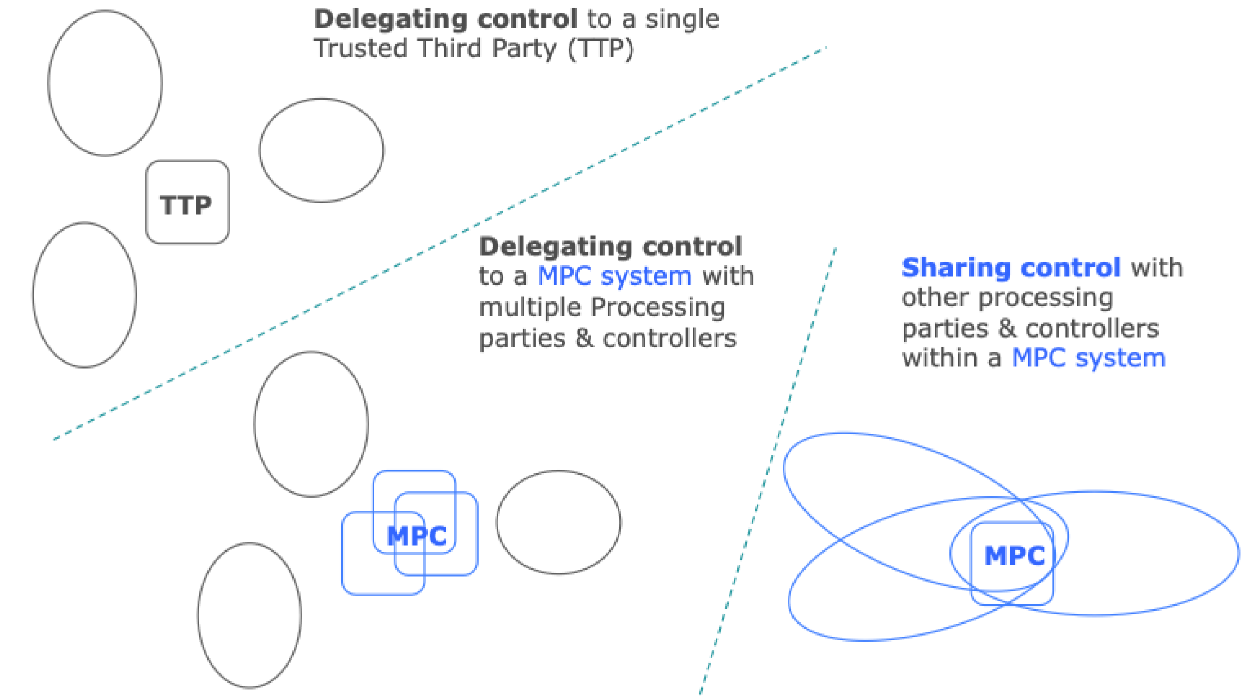
Figure 4.2 - Shared MPSPC-as-a-service infrastructure

120. By definition, the K processing parties are in charge of jointly controlling the computation process, and therefore they are to be trusted collectively, not individually. The MPSPC infrastructure shall operate according to a set of policies centered around the principle that no computation task (thereby including simple queries) may be executed on the data without preliminary explicit approval of all K processing parties. The MPSPC infrastructure shall be engineered based on state-of-the-art technologies that are able to strictly enforce these policies. The robustness of the overall design shall therefore depend jointly (1) on the choice of the processing parties; (2) on the strength of the policies that define the operation of the processing parties; and (3) on the strength of the technologies that enforce these policies at hardware and/or software levels.

121. Conceptually, we may think of the MPSPC infrastructure as a multi-party safe environment, i.e., a locked safe where the key is split into K shares held by K different processing parties. In order to unlock a new computation task, all key-shares must be inserted

is one among several possible schemes of choice for MPSPC operation. In multi-key encryption schemes, where the equivalent of a single decryption key is split among multiple key holders, the notion of processing party may correspond to key holders.

into the lock[11], therefore all K processing parties have to agree to it. The implemented policies and technologies determine the strength of the safe, but the overall level of trust depends also on the choice of the K key-share holders, i.e., on their collective level of trustworthiness.


Delegating control to a single Trusted Third Party (TTP)
TTP
Delegating control to a MPC system with multiple Processing parties & controllers
MPC
Sharing control with other processing parties & controllers within a MPC system
MPC

Explanation: ovals represent Input Parties and Output Parties.
Rectangles represent processing parties & controllers
MPC : Multi-Party Computation

Figure 4.3 – Abstract representation of processing control distribution in the different paradigms. In the Trusted Third Party model a single entity is delegated full processing control (left). Multi-Party Secure Private Computation technologies enable multiple processing parties to share processing control. Therefore the processing parties must be trusted collectively, not individually. The input/output parties may delegate processing control to an external set of multiple processing parties (middle) or share processing control directly with the other parties (right).

122. Moving from the traditional 'single key' paradigm to 'multiple key-shares' is the first innovation of MPC over TTP, as depicted in Figure 4.3. This allows replacing a single external TTP with multiple Processing Parties (PP). The next step is to let the entities serving as Input and/or Output Parties play the role of Processing Parties themselves, as depicted in Figure 4.2. When cast into the MPSPCaaS model, where the set of Input/Output Parties varies from one computation task to another, the distinction as to whether the entities in charge of acting as

[11] The term *Multi-Party* is used here in the general sense, with no intention at this stage to focus on any particular scheme. Different mechanisms and technologies may be adopted (and composed) to build a *multi-party safe environment* like the one outlined here, including but not limited to secret sharing, multi-key homomorphic encryption, trusted execution environment with multi-party authorization and possibly others.

Processing Parties (PP) correspond or not to the Input/Output Parties leads to two different flavors of MPSPCaaS operations:

A) Fixed-PP model, where the set of Processing Parties (PP) is fixed and does not change from one MPSPC task to another;
B) Mixed-PP model, where the PP set varies from one MPSPC task to another in order to let some of the Input/Output Parties play the PP role for that specific task.

123. Both models are in principle applicable to official statistics use-cases, and each of them may be preferred in different contexts. It is clear that the Mixed-PP model is more demanding than the Fixed-PP model for the MPSPCaaS users that are willing to take on the PP role (e.g. in terms of computational and organizational resources).

4.3 Summary of received responses

124. The affiliations of the c.ca 20 respondents were equally distributed between potential users/adopters (mostly statistical authorities) and developers/providers of PETs.

125. All respondents expressed appreciation for the general concept and acknowledged the prospective high value and benefit of setting up a shared infrastructure of that kind, provided that an adequate solution is given to all technical and non-technical open issues. Some of them indicated this model as being valuable beyond official statistics, for other public sectors and even for the private sector.

126. Besides the expressions of appreciation, most other comments provided by the respondents were aimed at pointing to important problem dimensions, key open issues, and critical design decisions. These critical aspects can discriminate between success and failure of the proposed concept at different levels – technology, governance, legal. Some of them were stated to be challenging or even very challenging to solve, but none of them was considered impossible or unfeasible. In other words, the concerns expressed by the experts highlight the importance of various dimensions, as perceived by different stakeholders, and the intrinsic complexity of providing acceptable solutions within the current scenario. However, no clear "showstopper" or "proof of infeasibility" were identified. Furthermore, for a few of such aspects the experts went one step beyond merely posing the issue (which is anyway a very useful and valuable insight for follow-up actions) and helped also to identify a possible solution direction.

127. On the technological side, there was consensus among the respondents that a single PET technology will not be sufficient, and a combination of different PET technologies is necessary to meet all the requirements. They suggest considering a combination of hardware-based (hence, TEE will be part of the game) and software-based solutions. Furthermore, containerization and orchestration of system components are seen as likely important functions.

128. The following problem dimensions were highlighted as being particularly critical for success but also challenging:

- Defining the exact roles of the various stakeholders and the overall governance model; and mapping these roles to the existing legal notions (e.g., "controller" and "processor" in the European legislation). This includes the assignment of verification and auditing powers, among others.

- Clarifying liabilities and defining a coherent "liability model" that is acceptable for the involved entities.
- Defining who provides (builds, operates, maintains, oversees) the physical infrastructure. The respondents that touched this point seem to agree that it should be up to public bodies at least at national level to play this role, also considering the financial costs and the unclear returns on investment. Engagement by international bodies is seen as a particularly positive aspect.
- Defining the "business model" and pricing scheme for using the system, i.e., who charges whom for what – and how much.

129. Warnings and concerns were expressed about the following risks:

- The system design and implementation should be such to avoid vendor lock-in by a particular technology provider (hardware or software).
- Avoiding that the software or hardware stale and get outdated, i.e., falls behind state-of-the-art.
- The (lack of) scalability of certain MPSPC technologies may limit the application of more complex statistical methods.
- The unclear relation between the MPSPCaaS concept (and PET in general) and the current legislation that was formulated in the pre-PET world. One respondent noticed that "outdated legislation" may hamper adoption of PET-based solutions and, if so, legislation would need to be adapted.
- The need to "educate" potential adopters, and especially managers, to the new paradigm enabled by IPP technologies where data are used without being held or even seen.
- The location of the processing nodes participating in a distributed/shared computation task must comply with the applicable legislation, and this may be difficult in case the data providers are subject to different jurisdictions.
- The current PETs are not only difficult to develop but also difficult to operate, use and parametrize. Correct usage of an MPSPCaaS infrastructure might, thus, require expert knowledge that might not be easily available both at the statistical institutions envisaged to operate the infrastructure as well as the input parties.

130. The following system features were indicated to be crucial for success:

- Transparency, to ensure independent scrutiny, hence increased security, and public acceptance. The open-source approach was seen as an important desirable aspect towards increased transparency.
- Usability by people without deep expertise in PET technologies.
- Scalability to effectively support the intended use-cases, with the possibility to easily scale up or scale out in case of increased demands.
- Independent auditing, code reviews, testing and certifications by qualified IT security bodies. Some respondents suggested the launch of bounty programs to encourage external testing by independent security experts and researchers (e.g., taking inspiration

by US-UK PET Prize[12]) based on synthetic data. Such actions should be performed ahead of starting operations but also during operation.

- Ultra-High levels of security, i.e., protection of data confidentiality and data integrity.
- Identity and access management needs to be put in place. All parties involved in the computation need to be identified and authenticated. Strict access control policies need to be in place and technically enforced to prevent illegitimate parties from accessing computation results. This needs to include the implementation of technical and organizational measures on the sides of processors, controllers, input parties and output parties.
- The system needs to support functionality for identifying misbehaving parties and for permanently excluding them from the system. In addition to protection from passive attacks, some protection against active attacks should be provided, at least the capability to detect (if not prevent) them.
- The possibility that input parties act as processors is seen as a necessary condition in some (but not all) scenarios. This is because input parties are inherently incentivized to at least secure their own data against unauthorized disclosure and modification. Especially in case of some MPC protocols, one honest processor suffices to prevent misuse of input data. Consequently, this feature is seen as critical for adoption in those scenarios.
- Capability to evolve the system and keep the software and hardware up to date with respect to advancing state-of-art.

131. The additional system features were seen as desirable or highly desirable:

- Relying on open-source components and standardized interfaces as far as possible.
- Interoperability with other future similar systems in the private sector.
- Some degree of differentiation in the offered levels of security. Use-cases may require different levels of guarantees regarding security and privacy but also different levels of performance. In addition, different privacy and security mechanisms are differently suited depending on the types of computations to be performed. Hence, the system would benefit greatly from a modular approach that allows the selection and/or combination of various privacy and security mechanisms such as to allow solutions that are tailored to the use case at hand.
- Enabling some degree of competition between processors and potential processors may drive down cost and increase quality of the envisaged infrastructure.
- Some respondents indicated that output checking procedures (output privacy) may be offered by the infrastructures alongside IPP functions.

132. Concerning use-cases, deterministic and probabilistic record linkage (and “join” operations), coupled with elementary operations (e.g., computation of cardinality, mean, variance), are seen as very important use-cases by multiple respondents, followed by standard

[12] https://petsprizechallenges.com

low-order statistical regressions. Some respondents indicated Machine Learning models as a generic class of possible use-cases of interest also for statistics. Finally, a some respondents identified additional use-cases falling outside the statistical domain (e.g., confidential benchmarking of private companies or public bodies' performances, confidential aggregation of pollution data13, confidential support to international law enforcement). These suggestions reinforce the idea that similar systems would be potentially relevant and valuable also outside the field of official statistics.

133. Finally, it was highlighted that the adoption of PET does not avoid the need to rely on a clear legal basis for data processing and, when applicable, international data transfers, especially in the case of personal data.

[13] In many cases, data on pollution levels (e.g. in specific cities or parts thereof or from specific industries, factories or machinery) are not easily available as cities and/or private entities are reluctant to share them in case they are not obliged to do so. The envisaged MPSPCaaS infrastructure would enable confidential benchmarking, thereby providing communities and enterprises with previously non-available insights.

Chapter 5 - General takeaways

134. In summary, the main takeaways from the Input Privacy-Preservation Project are the following:

- The project has confirmed that IPP computation solutions have an enormous potential for official statistics. However, there is still some work to be done (also by NSOs) and challenges to be solved before their actual adoption in regular statistical production can take place, beyond experimental projects.
- There is a need to clarify the legal aspects around the use of IPP technologies, particularly when (i) personal data are involved on the input side and/or (ii) the input data fall under different jurisdictions.
- Processing data with IPP still qualifies as … processing data(!). Consequently, the adoption of IPP solutions does not exclude the need for having a clear legal basis for processing the data in the first place. However, the additional protection measures offered by IPP will be instrumental (i) to strengthen and consolidate legal compliance to existing legislation; (ii) improve public acceptance for extending the scope of data processing for statistical purposes; and (iii) lower the barrier towards extending statistical legislation to enable the (re)use of new data sources[14].
- Further work is needed by NSOs to understand the organizational implications of adopting PET solutions, and particularly IPP technologies. Shifting towards a paradigm where data (re)use is decoupled from data gathering and data ownership has several branching implications on different processes and organizational units within the NSO (e.g., methodological development, IT, statistical production) and between the NSO and other organizations (e.g., data providers or NSOs in other countries).
- Liabilities, governance, and process ownership aspects (e.g., cost distribution) need to be carefully re-thought and possibly re-defined as the traditional practices may not be applicable *as is* to the new paradigm.
- On the technological side, it is unlikely that a single PET component (e.g., a single software protocol) can provide a sufficient "solution" for a particular use-case. Most use cases will require the adoption of more articulated "system" solutions based on the combination/integration of different PET components.
- Certain technology trade-offs dimensions need to be better investigated and understood by potential adopters, e.g., complexity vs. scalability; security vs. performance; general-purpose vs. specialized ad-hoc protocols. As the best balance point along each trade-off dimension depends on the use-case scenario, potential technology adopters need to work together with technology providers to design effective and efficient solutions case by case.

[14] Relevant to this point, it is worth highlighting that in January 2023 the European Commission has adopted the proposal for a new Regulation on European statistics on population and housing that explicitly states in the legal text that data sharing should be "based preferably on privacy enhancing technologies that are specifically designed to implement the principles of Regulations (EU) 2016/679 and (EU) 2018/1725, with particular regard to purpose limitation, data minimisation, storage limitation, integrity and confidentiality."

- Given the complexity of the task and the many intermingled technological and non-technological aspects it is necessary for technology adopters (e.g., NSO) to build a sufficient set of internal technical skills and expert knowledge. Such knowledge is required to interact fruitfully with technology providers, not to replace them. Experimental "lab" activities and IT sandboxes are instrumental to build such knowledge on the side of NSOs.
- Fully open-source solutions may have not yet achieved the level of technological maturity of proprietary solutions.
- The robustness of PET solutions depends not only on the "protocol" but also on the details of their actual implementation. If a (nominally) strong protocol is implemented in a simplistic way, the overall system may not be sufficiently robust and could be exposed to attacks (e.g., side-channel attacks).

135. These considerations should be seen as an encouragement for the management of statistical organizations to continue investing in skill building and infrastructure in the field of Input Privacy Preservation, both internally to statistical systems, at national and international level, and in partnership with other private and public organizations.

Annex - Private set intersection with analytics: use case Istat-Bank of Italy

Motivations

136. Producing statistics with the use of multiple data sources allows us to produce quality statistical output. The datafication process of the modern society has favored the proliferation of many new sources of data, and their integration can give benefit to the production of statistics. While the use and integration of data provides benefits, it also carries privacy risks that statistic producers need to consider. In the context of official statistics, NSIs are studying and experimenting privacy preserving techniques that can help to overcome the privacy risks associated with new statistical production scenarios that require the use and integration of multiple data sources. One of the most used techniques is Secure Multi Party Computation (SMPC). It is a subfield of cryptography with the goal of creating methods for parties to jointly compute a function over their inputs while keeping those inputs private.

137. In this appendix we will illustrate a joint work in which we apply the Private Set Intersection (PSI), a problem within the field of Secure multi-party computation. The PSI concept can be explained by this trivial example: there are two friends Alice and Bob such that Alice has a set of items A= (a1,…, an) and Bob has the set B=(b1,…, bn). The goal is to design a protocol by which Alice and Bob obtain the intersection A∩B, under the restriction that the protocol must not reveal anything about items that are not in the intersection. In the context of Private Set Intersection there are several possible scenarios; we list and explain the following: (i) Private Set Intersection (PSI), Private Set Intersection with Enrichment (PSI-E), Private Set Intersection with Analytics (PSI-A), Private data mining (PDM).

138. Exact PSI definition: Let P1 and P2 be parties owning (large) private databases D1 and D2. The parties wish to apply an exact join to D1 and D2 without revealing any unnecessary information about their individual databases. That is, ideally, the only information learned by P1 about D2 and by P1 about D1 is D1 ∩ D2.

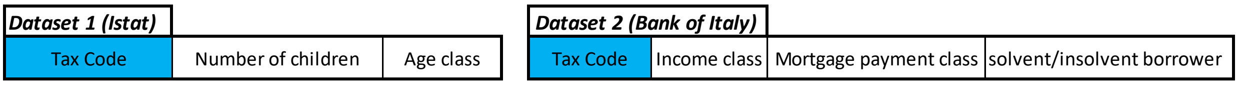
139. PSI-E definition Let P1 and P2 be parties owning (large) private databases D1 and D2. The parties wish to apply an exact or approximate join to D1 and D2 without revealing any unnecessary information about their individual databases. After that, they wish to enrich joined records with variables by both parties. At the end of the process, P1 will learn additional P2 variables on D1∩ D2 and P2 will learn additional P1 variables on the same intersection PSI-A definition. The parties wish to perform a statistical analysis on the intersection of their databases in a private fashion. To identify the records belonging to the intersection, they agree to apply an Exact PSI. At the end of the process, the only information learned by the parties (beyond the keys of the records belonging to the intersection) is the result of the statistical analysis.

140. Private data mining definition: Let P1 and P2 be parties owning (large) private databases D1 and D2. The parties wish to apply an analytics function to the joint database D1∪D2 without revealing any unnecessary information about their individual databases. At the end of the process, the only information learned by P1 about D2 is that which can be learned from the output of the analytics function, and vice versa.

Use case

141. The use case refers to the Private Set Intersection with Analytics (PSI-A) scenario. The parties involved, Istat and Bank of Italy, own dataset D1 and D2 respectively. D1 and D2 have a common key, which can be exploited to perform an Exact PSI. The parties wish to enrich their information assets by learning the results of a statistical analysis applied to the intersection of their datasets.

142. The following figure shows the fields of the two datasets D1 and D2:



Dataset 1 (Istat)		
Tax Code	Number of children	Age class

Dataset 2 (Bank of Italy)			
Tax Code	Income class	Mortgage payment class	solvent/insolvent borrower

Figure A1 - Istat and Bank of Italy datasets structure

The role of the parties involved is equal and all parties can make the same kind of analysis. We are in a G2G scenario, in which two or more public entities want to make analytics on the union of their datasets, in a privacy preserving way. This problem is often solved with legal agreements between the entities involved. Without legal ex ante agreements, a technological solution would make it easier to achieve the goal both from a time and an organizational point of view. This solution must provide a trusted infrastructure, which guarantees the levels of input privacy preserving required by the Data Protection Authorities.

143. To implement the use case we assume to be in an honest-but-curious (HbC) context in which the involved parties will respect all the rules defined in the protocol. We assume that, in concrete application scenarios, the typical size of involved dataset will be in the order of dozens of attributes and millions of records. The use case has to provide the following features:

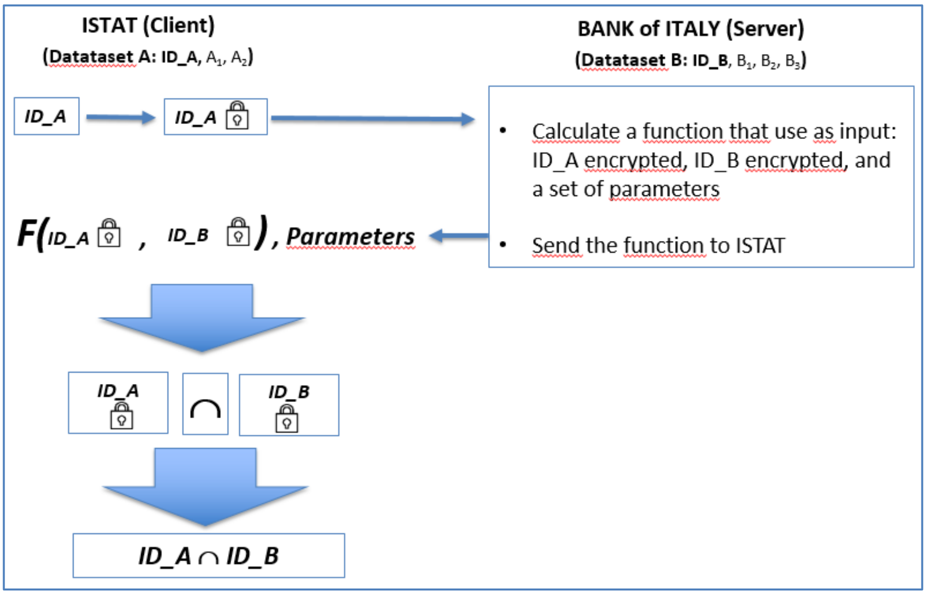
- A step to perform the join on the keys of the datasets owned by Istat and Bank of Italy.
- A step to compute a set of simple statistical functions on an encrypted enriched dataset built with the strictly necessary variables.

144. In the use case, there are two distinct phases:

- An off line phase in which the two parties: (i) share datasets of individuals that have a common key (i.e. Tax Code); (ii) agree to share some variables; (iii) share a symmetric key through RSA protocol; (iv) share IP addresses to use.
- An on line phase in which: (i) is performed a Private Set Intersection following the De Cristoforo protocol[15]; (ii) the data in encrypted format are transmitted to a neutral third party (i.e. the Linker); (iii) Istat and bank of Italy can submit queries to the Linker and obtain the results desired.

145. The De Cristoforo process is briefly described in the following figure:

[15] E. De Cristofaro and G Tsudik Practical Private Set Intersection Protocols with linear Computational and Bandwidth Complexity.

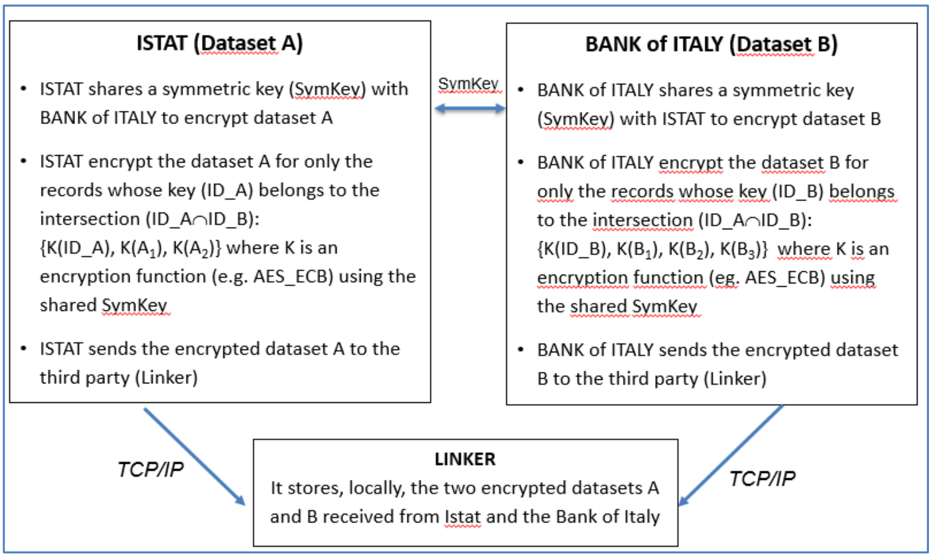

ISTAT (Client)
(Datataset A: ID_A, A_1, A_2)
BANK of ITALY (Server)
(Datataset B: ID_B, B_1, B_2, B_3)
ID_A
ID_A
F(ID_A , ID_B), Parameters
ID_A
∩
ID_B
ID_A ∩ ID_B

- Calculate a function that use as input: ID_A encrypted, ID_B encrypted, and a set of parameters
- Send the function to ISTAT

Figure A2 - Private Set Intersection phase

146. At the end of this step, the two parties, Istat and bank of Italy, own the intersection of their two dataset as result.

147. In the second step (i.e. loading phase), shown in the figure 3, the encrypted datasets of the two parts are uploaded on the Linker server; in this way the linker receive the datasets, of the two parties, encrypted with the same shared key. The use of the same cryptographic key will allow the linker to count equal fields, even if encrypted.



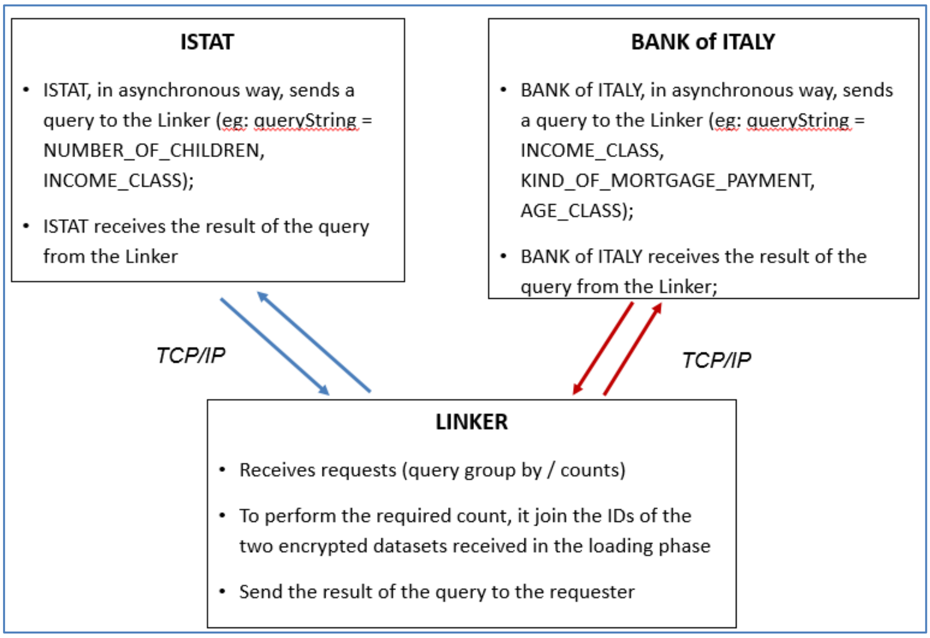
- ISTAT shares a symmetric key (SymKey) with BANK of ITALY to encrypt dataset A
- ISTAT encrypt the dataset A for only the records whose key (ID_A) belongs to the intersection (ID_A∩ID_B): {K(ID_A), K(A_1), K(A_2)} where K is an encryption function (e.g. AES_ECB) using the shared SymKey
- ISTAT sends the encrypted dataset A to the third party (Linker)

- BANK of ITALY shares a symmetric key (SymKey) with ISTAT to encrypt dataset B
- BANK of ITALY encrypt the dataset B for only the records whose key (ID_B) belongs to the intersection (ID_A∩ID_B): {K(ID_B), K(B_1), K(B_2), K(B_3)} where K is an encryption function (eg. AES_ECB) using the shared SymKey
- BANK of ITALY sends the encrypted dataset B to the third party (Linker)

It stores, locally, the two encrypted datasets A and B received from Istat and the Bank of Italy

Figure A3 - Loading phase

148. In the third step (i.e. query phase) as shown in figure X the two parties can send a query (i.e. group by, counts, etc.) to the linker and receive the results.



ISTAT

- ISTAT, in asynchronous way, sends a query to the Linker (eg: queryString = NUMBER_OF_CHILDREN, INCOME_CLASS);
- ISTAT receives the result of the query from the Linker

BANK of ITALY

- BANK of ITALY, in asynchronous way, sends a query to the Linker (eg: queryString = INCOME_CLASS, KIND_OF_MORTGAGE_PAYMENT, AGE_CLASS);
- BANK of ITALY receives the result of the query from the Linker;

LINKER

- Receives requests (query group by / counts)
- To perform the required count, it join the IDs of the two encrypted datasets received in the loading phase
- Send the result of the query to the requester

Figure A4 - Query phase

Conclusions

149. The use case described above has pros and cons. As pros, we mention:

- The possibility of applying it to similar situations in which two or more parties want to enrich their information assets by performing an intersection of their datasets in privacy preserving manner.
- The transmission to the third party is only for the encrypted and strictly necessary data.
- The two parties do not exchange data directly, except those necessary to calculate the intersection on the A∩B keys.
- Third party can carry out checks on the counts returned and ensure that the result cannot be traced back to the individual elements of the population.

150. As cons, we mention:

- Information enrichment takes place only in terms of aggregated data (counts).
- Privacy preservation is not guaranteed in the event that one of the two parties agrees dishonestly with the third party.
- Two parties become aware of which elements of their dataset also belong to that of the other party.

References

Abadi, M. et al. (2015) “TensorFlow: Large-scale machine learning on heterogeneous systems”, Software available from: https://www.tensorflow.org

Abadi, M., Chu, A., Goodfellow, I., Brendan McMahan, H., Mironov, I., Talwar, K. and Zhang, L. (2016). “Deep Learning with Differential Privacy”. ArXiv. Available at: https://arxiv.org/abs/1607.00133

Acar A., Aksu H., Luagac A.S.and Conti M. “A Survey on Homomorphic Encryption”, Schemes: Theory and Implementation, 2017 - https://arxiv.org/abs/1704.03578

Beutel, D. J. et al. (2020). “Flower: A Friendly Federated Learning Research Framework”. arXiv preprint arXiv:2007.14390. https://github.com/adap/flower

Boenisch, F. ”Attacks against Machine Learning Privacy (Part 2): Membership Inference Attacks with TensorFlow Privacy” - Published: January 24, 2021 - https://franziska-boenisch.de/posts/2021/01/membership-inference/

Bruno, M., De Cubellis, M., De Fausti, F., Scannapieco, M. and Vaccari, C. (2021). “Privacy set intersection with analytics - an experimental protocol (PSI De Cristofaro)”, UNECE-IPP presentation.

Chambers, R. and Kim, G. (2016), “Secondary analysis of linked data” in Methodological Developments in Data Linkage, Wiley, pp. 83-108.

Dasylva, A. (2018). “Pairwise estimating equations with linked data”, International Methodology Symposium.

Dasylva, A. (2022). “A three-party private set intersection protocol for data sets with typographical errors”, notes.

Dasylva, A. and Goussanou, A. (2022). “On the consistent estimation of linkage errors without training data”, Japanese Journal of Statistics and Data Science.

Dasylva, A. and Zanussi, Z. (2021). “A private set intersection use case”, presentation at the UNECE-IPP workshop.

De Cristofaro, E. and Tsudik, G. (2010). “Practical private set intersection protocols with linear computational and bandwidth complexity”, in Proceedings of Financial Cryptography and Data Security, 2010.

Evans D., Kolesnikov V. and Rosulek M. “A Pragmatic Introduction to Secure Multi-Party Computation.” - NOW Publishers, 2018.

Harris, C.R., Millman, K.J., van der Walt, S.J. et al. “Array programming with NumPy”. Nature 585, 357–362, 2020. DOI: 10.1038/s41586-020-2649-2. https://numpy.org/

Houssieau, F., Jordan, J., Cohen, S., Daniel, O., Elliott, A., Geddes, J., Mole, C., Rangel-Smith, C. and Szpruch, L. (2022). “TAPAS: a Toolbox for Adversarial Privacy Auditing of Synthetic Data”, ArXiv. Available at: https://arxiv.org/abs/2211.06550

Hunter, J.D. "Matplotlib: A 2D Graphics Environment", Computing in Science & Engineering, vol. 9, no. 3, pp. 90-95, 2007. https://matplotlib.org/stable/index.html

Kamara, S., Mohassel, P., Raykova, M. and Sadeghian, S. (2014). “Scaling private set intersection to billion-element sets”, in Proceedings of the International Conference on Financial Cryptography and Data Security.

Lahiri, P. and Larsen, M. (2005). “Regression analysis with linked data”, Journal of the American Statistical Association, vol. 100, pp. 222-230.

McKenna, R., Miklau, G. and Sheldon, D. (2021). “Winning the NIST Contest: A scalable and general approach to differentially private synthetic data”, ArXiv. Available at: https://arxiv.org/abs/2108.04978

McMahan B, Moore E, Ramage D, Hampson S and y Arcas BA. Communication-efficient learning of deep networks from decentralized data. InArtificial intelligence and statistics 2017 Apr 10 (pp. 1273-1282). PMLR. https://arxiv.org/abs/1602.05629

Nilsson, L. (2022). “Time to Preference”. Journal of Economic Integration, 37(4), 589-648.

Paillier, P. "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes". Advances in Cryptology — EUROCRYPT ’99. EUROCRYPT, 1999. Springer. doi:10.1007/3-540-48910-X_16

Paszke, A. et al., “PyTorch: An Imperative Style, High-Performance Deep Learning Library. In: Advances in Neural Information Processing Systems 32” [Internet]. Curran Associates, Inc.; 2019. p. 8024–35. http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

Reyes-Ortiz, J. L. et al. (2016). “Transition-Aware Human Activity Recognition Using Smartphones”. Neurocomputing. Springer 2015. http://archive.ics.uci.edu/ml/datasets/Smartphone-Based+Recognition+of+Human+Activities+and+Postural+Transitions

Schnell, R., Bachteler, T., Reiher, J. (2009): “Privacy-preserving record linkage using Bloom filters”, BMC Medical Informatics and Decision Making, vol. 9.

Shokri, R., Stronati, M., Song, C. and Shmatikov, V. “Membership inference attacks against machine learning models.” In 2017 IEEE Symposium on Security and Privacy (SP), pp. 3-18. IEEE, 2017.

The pandas development team, pandas-dev/pandas: Pandas, Zenodo, 2020. https://doi.org/10.5281/zenodo.3509134

United Nations Committee of Experts on Big Data and Data Science for Official Statistics, “United Nations Guide on Privacy-Enhancing Technologies for Official Statistics”, United Nations, New York, 2023. Website: https://unstats.un.org/bigdata

Yakoubov S., Gadepally V., Schear N., Shen E. and Yerukhimovich A. “A survey of cryptographic approaches to securing big-data analytics in the cloud IEEE” - High Performance Extreme Computing Conference (HPEC), 2014

Additional Resources

CSIRO's Data61, Python Paillier Library, GitHub Repository, 2013.
https://github.com/data61/python-paillier

The Pallets Project, $click_: python-click, GitHub Repository, 2023.
https://github.com/pallets/click

[2211.06550] TAPAS: a Toolbox for Adversarial Privacy Auditing of Synthetic Data (arxiv.org)

https://github.com/tensorflow/privacy/tree/master/tensorflow_privacy/privacy/privacy_tests/membership_inference_attack

https://github.com/VectorInstitute/PETs-Bootcamp/tree/main/Membership_Inference_Attacks