Probabilistic Metric Spaces for Privacy by Design Machine Learning Algorithms: Modeling Database Changes

Vicenç Torra[1(✉)] and Guillermo Navarro-Arribas[2(✉)]

[1] University of Skövde, Skövde, Sweden
vtorra@ieee.org
[2] Department of Information and Communications Engineering, CYBERCAT-Center for Cybersecurity Research of Catalonia, Universitat Autònoma de Barcelona, Barcelona, Spain
guillermo.navarro@uab.cat

Abstract. Machine learning, data mining and statistics are used to analyze the data and to build models from them. Data privacy for big data needs to find a compromise between data analysis and disclosure risk. Privacy by design machine learning algorithms need to take into account the space of models and the relationship between the data that generates the models and the models themselves. In this paper we propose the use of probabilistic metric spaces for comparing these models.

Keywords: Data privacy · Integral privacy
Probabilistic metric spaces

1 Introduction

Machine learning and statistics are powerful tools to extract knowledge from data. Knowledge is expressed in terms of models or indices from the data. Nevertheless, as it is well known, these models and indices can compromise information and can lead to disclosure [14].

Differential privacy [4] and integral privacy [13,16] are privacy models provided to avoid inferences from models and statistics. Other tools are to evaluate the analysis of disclosure risk from models. For example, membership attacks are about inferring the presence of a record in the database that was used to generate a model.

Machine learning and statistics build models from data, which are analyzed and compared by researchers and users, for example, with respect to their accuracy. Privacy by design machine learning algorithms [15] need to take into account additional aspects. In particular, the space of models, and how these

Partial support from the Vetenskapsrådet project "Disclosure risk and transparency in big data privacy" (VR 2016-03346, 2017-2020), and Spanish project TIN2017-87211-R is gratefully acknowledged.

J. Garcia-Alfaro et al. (Eds.): DPM 2018/CBT 2018, LNCS 11025, pp. 422–430, 2018.
https://doi.org/10.1007/978-3-030-00305-0_30

models are generated. We consider that there are two additional aspects to take into account besides just applying an algorithm and deliver the resulting model.

One is the direct comparison of the models. For example, there are works that study regression coefficients and how the regression coefficients are modified when data is perturbed by a masking method (e.g., microaggregation [3,9] or recoding [10] are applied to achieve k-anonymity [12]).

Another is the comparison of models with respect to the similarity of the databases that have generated them. Up to our knowledge, this aspect has not been studied in the literature until now. This topic is of relevance because databases are dynamic and it is usual that changes are applied to them. Changes can be due to different causes. E.g., the GDPR (e.g., right to rectification or deletion) can require businesses to update their data. When databases change, we may need to revise the models. Therefore, it is useful to know when two models can be generated with similar databases. I.e., how changes in the database are propagated to the models.

In this paper we propose the use of probabilistic metric spaces for modeling the relationships between machine learning models and statistics. This type of spaces define metrics in terms of a distance distribution function, which permits us to represent randomness. We will define the distance between two models in terms of distances between the databases that generate the models. Randomness permits us to represent the fact that the possible modifications that are applied to a database are not know. As we will see, in the context of data privacy, these distances can be applied to measure similarities between models with respect to their training set, or to define disclosure measures on anonymized models.

The structure of the paper is as follows. In Sect. 2 we discuss distances and metrics. In Sect. 3 we introduce a definition of probabilistic metric spaces for machine learning models. The paper finishes with a discussion.

2 Distances and Metrics

Metric spaces are defined in terms of a non-empty set and a distance function or metric. Let (S, d) be a metric space, then $d(a, b)$ for $a, b \in S$ measures the distance between the two elements a and b in S. It is known that d needs to satisfy some properties: positiveness, symmetry, and triangle inequality. Also, that if a and b are different then the distance should be strictly positive. Naturally, triangle inequality is that $d(a, b) \leq d(a, c) + d(c, b)$ for any a, b, c in S. When the distance does not satisfy the symmetry condition, (S, d) is a quasimetric space. If the distance does not satisfy the triangle inequality, (S, d) is a semimetric space.

2.1 Metrics for Sets of Objects

Given a metric space (S, d), its extension to a set of elements of S is not trivial. Several distances have been defined on sets but not all of them satisfy the triangle inequality, thus, do not lead to metrics. For example, with

$dm(x, A) = \min_{y \in A} d(x, y)$ we can define the Hausdorff distance, dH, and the sum of minimum distances, ds, as

$$dH(A, B) = \max\{\max_{y \in A} dm(y, B), \max_{y \in B} dm(y, A)\}$$

$$ds(A, B) = \frac{1}{2}\left(\sum_{y \in A} dm(y, B) + \sum_{y \in B} dm(y, A)\right).$$

However, these distances are not metrics (triangle inequality does not hold).

Eiter and Mannila [5] introduced a way to define a metric. It is based on considering a finite sequence $P = (P_1, \ldots, P_m)$ with $m \geq 2$ and $P_i \subseteq S$ for all $i \in \{1, \ldots, m\}$. The cost of such P is $c_d(P) = \sum_{i=1}^{m-1} d(P_i, P_{i+1})$. The distance $d^w : \wp_\emptyset(S) \times \wp_\emptyset(S) \to \mathbb{R}^+$ is defined as follows where $\wp_\emptyset(S)$ is the power set of S without the emptyset, and $P(A, B)$ denotes all paths between A and B.

$$d^w(A, B) = \min\{c_d(P) : P \in P(A, B)\}.$$

The authors prove in [5] that this definition is a metric when d is a distance.

2.2 Probabilistic Metric Spaces

Probabilistic metric spaces generalize the concept of a metric. Informally, they are based on distribution functions. So, the distance is not a number but a distribution on these numbers.

Definition 1. [11] *A nondecreasing function F defined on $\mathbb{R}^+$ that satisfies (i) $F(0) = 0$; (ii) $F(\infty) = 1$, and (iii) that is left continuous on $(0, \infty)$ is a distance distribution function. Δ^+ denotes the set of all distance distribution functions.*

We can interpret $F(x)$ as the probability that the distance is less than or equal to x. In this way, this definition is a generalization of a distance.

We will use ϵ_a to denote the distance distribution function that can be said to represent the classical distance a. This ϵ function is just a step function at a.

Definition 2. [11] *For any a in $\mathbb{R}$, we define ϵ_a as the function given by*

$$\epsilon_a(x) = \begin{cases} 0, & -\infty \leq x \leq a \\ 1, & a < x \leq \infty \end{cases}$$

Probabilistic metric spaces are defined by means of distance distribution functions. In order to define a counterpart of the triangle equality we introduce triangle functions. They are defined as follows.

Definition 3. [11] *Let Δ^+ be defined as above, then a binary operation on Δ^+ is a triangle function if it is commutative, associative, and nondecreasing in each place, and has ϵ_0 as the identity.*

Triangle functions has close links with t-norms [2]. If T is a t-norm, then $\tau_T(F,G)(x) = T(F(x),G(x))$ is a triangle function. See Def. 7.1.3 and Sect. 7.1 in [11]. The maximal triangle function is $\tau_{\min}$.

We are now in conditions to define probabilistic metric spaces.

Definition 4. [11] *Let $(S,\mathcal{F},\tau)$ be a triple where S is a nonempty set, $\mathcal{F}$ is a function from $S \times S$ into Δ^+, τ is a triangle function; then $(S,\mathcal{F},\tau)$ is a probabilistic metric space if the following conditions are satisfied for all p, q, and r in S:*

- *(i)* $\mathcal{F}(p,p) = \epsilon_0$
- *(ii)* $\mathcal{F}(p,q) \neq \epsilon_0$ *if* $p \neq q$
- *(iii)* $\mathcal{F}(p,q) = \mathcal{F}(q,p)$
- *(iv)* $\mathcal{F}(p,r) \geq \tau(\mathcal{F}(p,q),\mathcal{F}(q,r))$.

We will use F_{pq} instead of $\mathcal{F}(p,q)$ and, then, the value of the latter at x by the expression: $F_{pq}(x)$.

3 Probabilistic Metric Spaces for Machine Learning Models

In this section we define a probabilistic metric space for machine learning models based on the databases that permit to build these models. So, we are considering two spaces. On the one hand we have the space of databases. In this space we can consider transitions from one database to another. These transitions correspond to changes in the database. Naturally, they correspond to record deletion, record addition, and record modification. On the other hand we have the space of models. Each model can be generated by one or more databases in the space of databases. Figure 1 represent these two spaces and some relationships between them.

Formally, the space of databases is a graph. Note that each possible database can be considered the vertex or node in the graph; and that any type of database transformation is represented in terms of an edge (transforms a database into another one). In the figure, we only include directed edges that represent deletions.

Definition 5. *Let $\mathcal{D}$ represent the space of possible databases. I.e., $db \in \mathcal{D}$ are the possible databases we may encounter. Let $\mathcal{O}$ represent the possible minimal set of modifications. More particularly, $\mathcal{O}$ will typically include erasure of a single record, addition of a single record, and rectification of a value of a variable in a record. Then, given $db \in \mathcal{D}$, we have that o_{db} are the operations in $\mathcal{O}$ that are valid for db. For each $o \in o_{db}$, we have that $o(db) \in \mathcal{D}$ and $o(db) \neq db$.*

With these definitions, we can define the graph associated to a space of databases as follows. We assume that the construction leads to a proper graph. That is, there are no multiedges. Formally, $o_1(db) \neq o_2(db)$ for any $o_1, o_2 \in \mathcal{O}$ with $o_1 \neq o_2$.

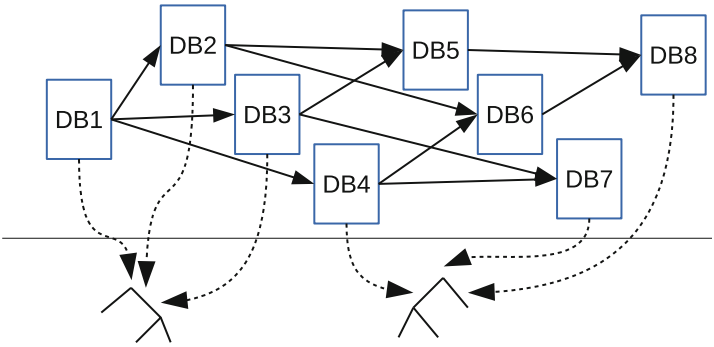

Fig. 1. Space of databases (top) and space of models (bottom) generated from the databases (dotted lines). Some transitions between databases (DB) are represented in the figure (arrows). For the sake of simplicity, we only consider directed transitions (e.g., as the only allowed transition is deletion of a single record) between databases.

Definition 6. *Let $\mathcal{D}$ be a space of databases, and $\mathcal{O}$ be the minimal set of considered modifications. Then, we define the graph for the space $\mathcal{D}$ inferred from $\mathcal{O}$ as the graph $G_{\mathcal{D},\mathcal{O}} = (V, E)$ with the set of vertices defined by $V = \mathcal{D}$ and the set of edges defined by*

$$E = \cup_{db \in \mathcal{D}} \cup_{o \in o_{db}} \{(db, o(db))\}.$$

We say that the set $\mathcal{O}$ is reversible if for any $o \in \mathcal{O}$ such that $db' = o(db)$ with $db \in \mathcal{D}$, we have an $o' \in \mathcal{O}$ such that $db = o'(db')$. If $\mathcal{O}$ is reversible, the graph $G_{\mathcal{D},\mathcal{O}} = (V, E)$ can be seen as an undirected graph. When $\mathcal{O}$ contains only deletions, it is not reversible; while with deletions and additions, it is reversible.

Given a space of databases and an algorithm that generates a model for each database, we can build a space of models. The definition of the space of models is based on a deterministic algorithm A. That is, the algorithm always returns the same model when the same database is used.

Definition 7. *Let $\mathcal{D}$ be a space of databases, and let A be a deterministic algorithm that applied to any $db \in \mathcal{D}$ builds a model m. Then, $\mathcal{M}_{\mathcal{D},A}$ is the space of models that can be inferred from $\mathcal{D}$ using A. Naturally,*

$$\mathcal{M} = \cup_{db \in \mathcal{D}} \{A(db)\}.$$

Now, let us consider a pair of models. As stated above, our goal is to define a distance between pairs of models in terms of the similarities between the databases that have generated them. Then, it is relevant to us how these models are constructed. In our context, this means finding pairs of databases that can generate our pair of models. We formalize this below.

Given two models m_1 and m_2, we define $t(m_1, m_2)$ as the pairs of databases that permit us to *transit* from m_1 to m_2. That is, pairs of databases (db_1, db_2)

such that m_1 is the model generated from db_1 and m_2 is the model generated by db_2:

$$t(m_1, m_2) = \{(db_1, db_2) | A(db_1) = m_1, A(db_2) = m_2\}$$

Then, for each pair (db_1, db_2), we consider all paths from db_1 to db_2 and the corresponding lengths. We define $l(m_1, m_2)$ as the multiset of these lengths. Let $paths(db_1, db_2)$ represent all paths from db_1 to db_2. Then, $l(m_1, m_2)$ is the following multiset:

$$l(m_1, m_2) = \{length(path) | path \in paths(db_1, db_2) \text{ for } (db_1, db_2) \in t(m_1, m_2)\}.$$

Note that this is a multiset as when there are several paths for a pair of databases, it is possible that several of these paths have the same length. For example, there are two paths of length two between $DB1$ and $DB5$ in Fig. 1. When edges represent record deletion, we can find several paths between two databases as records can be removed in different order.

Finally, we define $l^*(m_1, m_2)(x)$ as the function that counts how many elements in $l(m_1, m_2)$ are less or equal to x. That is,

$$l^*(m_1, m_2)(x) = \sum_{d \in l(m_1, m_2) \& d \leq x} count(d). \tag{1}$$

Here $count(d)$ is the function that gives the number of occurrences of d in the multiset. This function is also known as multiplicity.

We now introduce a distance distribution function.

Definition 8. *Let $\mathcal{D}$ be the space of databases, and let $\mathcal{O}$ be the set of minimal modifications. Let $G_{\mathcal{D},\mathcal{O}} = (V, E)$ be the graph on $\mathcal{D}$ inferred from $\mathcal{O}$. Let l^* be defined as in Eq. 1 above. Let K be a constant such that $K > 0$, then, we define F as follows:*

$$F(m_1, m_2)(x) = \begin{cases} \epsilon_0 & \text{if } m_1 = m_2 \\ \min\left(1, \frac{l^*(m_1, m_2)(x)}{K}\right) & \text{if } m_1 \neq m_2 \end{cases} \tag{2}$$

We can prove the following result.

Proposition 1. *Let $\mathcal{D}$ be the space of databases, $\mathcal{O}$ be the set of minimal modifications, A be a deterministic algorithm, $\mathcal{M}_{\mathcal{D},A}$ be the space of models inferred from $\mathcal{D}$ and A, $G_{\mathcal{D},\mathcal{O}} = (V, E)$ be the graph on $\mathcal{D}$ inferred from $\mathcal{O}$, and let l^* and F defined as in Definition 8. Then, the following holds:*

- *$F(m, m) = \epsilon_0$ for all $m \in \mathcal{M}$,*
- *$F(m_1, m_2) \neq \epsilon_0$ for all $m_1, m_2 \in \mathcal{M}$ such that $m_1 \neq m_2$,*
- *$F(m_1, m_2) = F(m_2, m_1)$ when $\mathcal{O}$ is reversible.*

Proof. The proof that $F(m, m) = \epsilon_0$ is by construction.

Let us now consider the proof of $F(m_1, m_2) \neq \epsilon_0$ for $m_1 \neq m_2$. In this case, if $m_1 \neq m_2$, we will have that there are at least two different databases db_1 and

db_2 that generate m_1 and m_2, respectively, and $db_1 \neq db_2$. Therefore, there will be at least a path with a distance at least one between db_1 and db_2. Therefore, if $K > 0$, $F(m_1, m_2)(0) \neq 1$, which proves the equation.

Let us now consider the proof of the third condition. In this case, we have that for each path $path$ in $paths(db_1, db_2)$ we will have a path in $paths(db_2, db_1)$. This naturally follows from the fact that if $path = (db_1 = db^1, db^2, \ldots, db_2 = db^r)$ with $o^i = (db^i, db^{i+1}) \in \mathcal{O}$ for $i = 1, \ldots, r-1$, then by the reversibility condition there are $o'^i = (db^{i+1}, db^i) \in \mathcal{O}$ so that $path' = (db_2 = db^r, \ldots, db^2, db^1 = db_1)$, and $path' \in paths(db_2, db_1)$. □

As a corollary of this proposition, we have that the distance in Definition 8 leads to a probabilistic semimetric space when $\mathcal{O}$ is reversible.

Corollary 1. *Given $\mathcal{D}, \mathcal{O}, A, F$ as in Definition 8, then $(\mathcal{M}_{\mathcal{D},A}, F)$ is a probabilistic semimetric space.*

In general, $(\mathcal{M}_{\mathcal{D},A}, F)$ is not a probabilistic metric space because condition (iv) in Definition 4 does not follow. A counterexample of this condition for three models m_1, m_2 and m_3 is as follows: Some databases generating m_1 are connected to databases generating to m_3, and some generating m_3 are connected to databases generating m_2. This implies that $F(m_1, m_3)(u) + F(m_3, m_2)(v)$ is finite. When there is no connection between databases generating m_1 and those generating m_2, the direct distance will be ∞.

4 Discussion and Conclusions

Machine learning is about building models from data. Given a data set, the goal is to find a model that represents the data in an appropriate way. This problem is usually formulated as finding a model that has a good accuracy.

Nevertheless, this is not the only aspect taken into account in machine learning. As the bias-variance trade-off explains, one may have a high accuracy at the expenses of over-fitting. To avoid this over-fitting, we may select a model with less accuracy but with a good bias-variance trade-off.

In addition to that, other aspects are often taken into account. E.g., explainability [8]. We may be interested in a model with less accuracy if decisions are better explained. The same applies to fairness [7] and no-discrimination [6].

Within the privacy context, models need to avoid disclosure, and this requirement can be formally defined into different ways. Differential privacy [4] is one way, that is that the model does not differ much whether a record is present or not. Integral privacy [13,16] is another way, that is that the model can be generated by a sufficiently large number of possible data sets. Resistant to membership attacks is another way. This means that we cannot infer that a particular record was present in the training set.

Under this perspective, it is relevant to compare the models and their similarities with respect to the training data sets. To do so, we need to define a distance for models based on a distance on the training data sets. In this paper

we have proposed the use of probabilistic metric spaces for this purpose. We have proposed a first definition in this direction.

More broadly, in the privacy context, these distances can also be used to define disclosure or information loss metrics (see e.g. [1]). By measuring the differences between a privacy preserving model and the original model, one can establish the information that has been lost in the anonymization process.

Further work is needed in this direction. Actual computation of distance distribution functions can only be done easily for small data sets. So, we need to develop solutions for larger data sets. Secondly, we have assumed in this work that A is an algorithm that builds a model deterministically. This assumption does not always apply. On the one hand there are machine learning algorithms that include some randomness. This is the case, precisely, of some algorithms for big data based on sampling. On the other hand, there are randomized algorithms as the ones used in differential privacy. Appropriate models need to be developed to deal with this situation.

We have shown that our distance does not satisfy Equation (iv) in Definition 4. Definition of d^w in Sect. 2.1 satisfies triangle inequality for a distance d, so d^w could lead to a probabilistic metric space (see Definition 8.4.1 in [2]), but we need to explore if this distance is actually computable with actual data. Its cost, based on the set of all paths $P(A, B)$, seems too costly in our context.

References

1. Abril, D., Navarro-Arribas, G., Torra, V.: Supervised learning using a symmetric bilinear form for record linkage. Inf. Fusion **26**, 144–153 (2016)
2. Alsina, C., Frank, M.J., Schweizer, B.: Associative Functions: Triangular Norms and Copulas. World Scientific, Singapore (2006)
3. Domingo-Ferrer, J., Torra, V.: Ordinal, continuous and heterogeneous k-anonymity through microaggregation. Data Min. Knowl. Disc. **11**(2), 195–212 (2005)
4. Dwork, C.: Differential privacy. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4052, pp. 1–12. Springer, Heidelberg (2006). https://doi.org/10.1007/11787006_1
5. Eiter, T., Mannila, H.: Distance measures for point sets and their computation. Acta Informatica **34**, 109–133 (1997)
6. Hajian, S.: Simultaneous discrimination prevention and privacy protection in data publishing and mining, Ph.D. Dissertation, Universitat Rovira i Virgili (2013)
7. Hardt, M., Price, E., Srebro, N.: Equality of opportunity in supervised learning. In: Advances in Neural Information Processing Systems, pp. 3315–3323 (2016)
8. Knight, W.: The Dark Secret at the Heart of AI. MIT Technology Review, April 11 2017
9. Laszlo, M., Mukherjee, S.: Iterated local search for microaggregation. J. Syst. Softw. **100**, 15–26 (2015)
10. LeFevre, K., DeWitt, D.J., Ramakrishnan, R.: Multidimensional k-anonymity, Technical report 1521, University of Wisconsin (2005)
11. Schweizer, B., Sklar, A.: Probabilistic Metric Spaces. Elsevier-North-Holland, New York (1983)
12. Samarati, P.: Protecting respondents' identities in microdata release. IEEE Trans. Knowl. Data Eng. **13**(6), 1010–1027 (2001)

13. Senavirathne, N., Torra, V.: Approximating robust linear regression with an integral privacy guarantee. In: Proceedings of PST 2018 (2018, to appear)
14. Torra, V.: Data Privacy: Foundations, New Developments and the Big Data Challenge. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57358-8
15. Torra, V., Navarro-Arribas, G.: Big data privacy and anonymization. In: Lehmann, A., Whitehouse, D., Fischer-Hübner, S., Fritsch, L., Raab, C. (eds.) Privacy and Identity 2016. IAICT, vol. 498, pp. 15–26. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-55783-0_2
16. Torra, V., Navarro-Arribas, G.: Integral privacy. In: Foresti, S., Persiano, G. (eds.) CANS 2016. LNCS, vol. 10052, pp. 661–669. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-48965-0_44