Functional Data, Differential Privacy, and Machine Learning

Anthony Della Pella

Abstract. I present a quick introduction to the theory of Differential Privacy first defined in [**7**]. Then, I will give a brief overview of a more general notion of privacy followed by how this notion utilizes kernel techniques. I will also define a notion of differential privacy which is valid for functional data. Lastly, I present several applications of this framework to areas of machine learning and mathematics.

1. Differential Privacy

The desire for privacy of sensitive data is one that has been increasingly valid in the past several decades. In particular, one would like to be reassured that the release of a database containing their information will (given that they have been promised some type of privacy) not affect their lives. In most cases this is not an issue, however there are times when an adversary may wish to obtain information about someone with harmful intentions. Since such adversaries do exist, there are two potential courses of action:

- Give no information (or similarly give useless or untruthful information) when asked to provide details of one's private life.
- Require a type of security of your information which gives a "reasonable" promise of privacy.

The second option which is more desirable from the viewpoint of science and the general utility of acquired data puts a great burden of responsibility on the person who collects the data. In fact, we've seen in the past that such promises didn't hold much weight and peoples information could be recovered from seemingly private releases of data [**13**], [**11**]. Cynthia Dwork has defined a notion of privacy which is both flexible and guarantees a certain level of privacy for ones information contained in a database.

Definition 1.1 (Differential Privacy). A randomized algorithm $\mathcal{M}$ with domain $\mathbb{N}^{|\mathcal{X}|}$ is (ϵ, δ)-differentially private if for all $\mathcal{S} \subseteq \text{Range}(\mathcal{M})$ and for all $x, y \in \mathbb{N}^{|\mathcal{X}|}$ such that $||x - y||_1 \leq 1$:

$$\mathbb{P}(\mathcal{M}(x) \in \mathcal{S}) \leq \exp(\epsilon)\mathbb{P}(\mathcal{M}(y) \in \mathcal{S}) + \delta,$$

where the probability space is over the coin flips of the mechanism $\mathcal{M}$.

1.1. A More General Form.

This definition, as taken directly from [**8**] is particularly useful (due to its explicit nature) when dealing with histograms of databases, however a more robust, though largely equivalent definition involving measure theory is needed for our purposes in this paper.

Definition 1.2 (Alternative Differential Privacy). A set of distributions $\{P_D : D \in \mathfrak{D}\}$ is called $(\epsilon, \delta)-$differentially private whenever for all $D \sim D' \in \mathfrak{D}$ we have

$$P_D(A) \leq \exp(\epsilon)P_{D'}(A) + \delta, \forall A \in \mathfrak{A}.$$

Here, we make several notes: P_D is representative of the output of function $P : \mathfrak{D} \to \mathbb{R}^n$ when the input database D is given. Also, more generally, the algorithms will (on input database from $\mathfrak{D}$) give values in some measure space $(\Omega, \mathfrak{A})$. In the definition above, we assume $\mathfrak{A}$ to be the finest σ-field possible on Ω. Notice that the above definition is very general, and even when the space Ω is multidimensional (such as $\mathbb{R}^d$) we still can define a very clear notion of Differential Privacy.

REMARK 1.3. An even stronger notion of differential privacy (ϵ,0) or $\epsilon-$differential privacy is often used, and in fact this was the first form of privacy featured in [**7**].

1.2. Implications.

PROPOSITION 1.4. Let $X \sim P_D$ where $\{P_D : D \in \mathfrak{D}\}$ achieves (ϵ, δ)-differential privacy. Any level $\hat{p}$ hypothesis test of: $H_0 : D = D_0$ versus $H_a : D \neq D_0$ has power bounded above by $\hat{p}\exp(\epsilon) + \delta$.

A proposition such as this follows almost immediately from our more general framework, and while it is also apparent (with a little work) from the definition in [**8**], our framework here makes analysis of this type much more easier.

Also worth noting is that this proposition is where the privacy guarantee stems from. Namely, notice that is $\epsilon and \delta$ are sufficiently small, the test is not much more likely to correctly reject a false hypothesis than it is to incorrectly reject the true one. This sense that the probability of making a type 1 error being roughly equal to the probability of not making such an error is the true provider of the differential privacy guarantee.

2. Functional Data

Many data sets contain information in the form of n-tuples of data. For instance, on can look at viewers ratings of various movies (such as in [**11**]) and place each rating in a tuple with the i^{th} entry being that viewers rating for the i^{th} movie in the list. If we keep the movie positions constant in the list, that is the rating for movie i recorded for person p in the same entry (i) as it is for person q, we can put the records of many people in a matrix just by joining the tuples together as rows of the matrix where each row represents a different person, and the entries are movie ratings for a given movie (column). A technique such as this is very common in data collection especially (but definitely not limited to) the collection of subjective "ranking" or "rating" data.

There is however the need to take readings of data on a continuous (or near continuous scale). For instance, say one wanted to look at a handwriting example. You could then take the (x, y)- coordinate of a pen at each instant of a continuum (often with time denoted by t). Then, instead of having a tuple of data we may have a continuous function $h : \mathbb{R} \to \mathbb{R}^2$ whose input is a time t and where $h(t)$ represents the location of our tracked pen at this time. We could surely approximate our function with a tuple (say sample h every 60 milliseconds and record the x, y pair in a vector) however this would be a bad approach in all but a few exceptional cases. The downfalls of this method are:

- As the time we record goes on, eventually the tuples (and the corresponding matrix if we consider multiple recordings) will become

unwieldy unless our scale is defined in a way which bounds the number of entries our tuple will contain.
- If we dial back our scaling (say bump up to 600 milliseconds in our example) then we may lose sight of a bigger picture. That is, our approximated function $h(t)$ may have finer details which will be missed by considering time on a larger scale.

In order to combat these two downfalls of the "tuple" method, it should be clear that there are occasions in which we NEED functional data. For a very refined treatment of this notion, as well as some of the techniques used in this field one can find a great presentation of functional data analysis in [**12**].

3. Back to Privacy

We have already shown that privacy is something desirable anytime a set of data are collected which contain sensitive information. Can we find a definition of privacy (preferably differential privacy) which works for functional data? That is can we find a guarantee similar to that of Proposition 1.4 This is the subject of [**9**].

We need two more major theoretic results from [**9**] before we can gain utility in our definition of privacy for functional data.

PROPOSITION 3.1. Suppose that for all $D \sim D'$ there exists a set $A^*_{D,D'} \in \mathfrak{A}$ such that for all $S \in \mathfrak{A}$,

$$S \subseteq A^*_{D,D'} \implies P_D(S) \leq \exp(\epsilon) P_{D'}(S) \tag{3.1}$$

and

$$P_D(A^*_{D,D'}) \geq 1 - \delta. \tag{3.2}$$

Then the family of distributions $\{P_D\}$ is (ϵ, δ)-differentially private.

PROPOSITION 3.2. Suppose that for a symmetric p.s.d. matrix M of dimension d the set of vectors $\{v_D | D \in \mathfrak{D}\} \subset \mathbb{R}^d$ satisfies

$$\sup_{D \sim D'} ||M^{-\frac{1}{2}}(v_D - v_{D'})||_2 \leq \Delta.$$

Then the randomized algorithm which for input database D outputs

$$\tilde{v_D} = v_D + \frac{c(\delta)\Delta}{\epsilon} Z, \qquad Z \sim \mathcal{N}_d(0, M)$$

achieves (ϵ, δ) differential privacy whenever

$$c(\delta) \geq \sqrt{2 \log\left(\frac{2}{\delta}\right)}.$$

4. Privacy for Functions

Now we can begin our definition of differential privacy for functions/functional data. First, we begin by considering the family of functions f_D indexed by databases D and taking values in $T = \mathbb{R}^d$. That is, the set:

$$\{f_D : D \in \mathfrak{D}\} \subset \mathbb{R}^T.$$

As we've done with the "vector" data, we will look to randomized algorithms which on input D output some $\tilde{f_D} \sim P_D$ where P_D is a measure on $\mathbb{R}^T$ corresponding to the database D. In order to provide meaning to the measure above we will define a σ-field below.

4.1. The Field of Cylinders.

Definition 4.1. Cylinder sets of functions (see [**5**]) are defined for all finite tuples $S = (x_1, \ldots, x_n)$ where $x_i \in T$ and Borel sets B of $\mathbb{R}^n$:

$$C_{S,B} = \{f \in \mathbb{R}^T | (f(x_1), \ldots, f(x_n)) \in B\}.$$

Put simply, these "cylinder sets" are just spaces of functions which take on values (prescribed by the Borel sets B) for given inputs S.

4.2. Privacy via the Field of Cylinders.

Remark 4.2. We can consider the family of sets $\mathcal{L}_S = \{C_{S,B} | B \in \mathcal{B}(\mathbb{R}^n)\}$ which forms a $\sigma-$field for fixed sets S. Taking the union of these $\mathcal{L}_S$ over all finite tuples S (of length $n = 1, 2, \ldots$ provides us with the family $\mathcal{F}_0$. This is not quite a $\sigma-$field (it is not closed under countable intersections). We thus will consider a notion of privacy holding over this field of "cylinder sets" which considers algorithms such that for all $D \sim D' \in \mathfrak{D}$:

$$P(\tilde{f}_D \in A) \leq \exp(\epsilon) P(\tilde{f}_{D'} \in A) + \delta, \qquad \forall A \in \mathcal{F}_0. \tag{4.1}$$

Theorem 4.3. *Let $x_1, \ldots, x_n$ be any finite set of points in T chosen a-priori. Then whenever Equation 4.1 holds, the release of the vector:*

$$\left(\tilde{f}_D(x_1), \ldots, \tilde{f}_D(x_n)\right)$$

satisfies (ϵ, δ)-differential privacy.

Proof. We have

$$P_D\Big(\left(\tilde{f}_D(x_1), \ldots, \tilde{f}_D(x_n)\right) \in A\Big) = P_D(\tilde{f} \in C_{\{x_1,\ldots,x_n\},A}.$$

Finally, the privacy guarantee can be seen to hold from Equation 4.1. □

In [**5**] it was shown that we can extend the result Equation 4.1 to the generated $\sigma-$field, namely:

$$\mathcal{F} \stackrel{\text{def}}{=} \sigma(\mathcal{F}_0) = \bigcup_S \mathcal{L}_S.$$

4.3. A General Notion of Privacy for Functions. Finally, we come to our most general definition of differential privacy for functions:

THEOREM 4.4. *Let Equation 4.1 hold. Then the family* $\{P_D|D \in \mathfrak{D}\}$ *on* $(\mathbb{R}^T, \mathcal{F})$ *satisfies for all* $D \sim D' \in \mathfrak{D}$*:*

$$P_D(A) \leq \exp(\epsilon)P_{D'}(A) + \delta, \qquad \forall A \in \mathcal{F}.$$

One proof of this theorem is found in [**9**]. Due to the nature of computers being unable to provide a "complete" description of the function $\tilde{f}_D$, this result is mainly for theoretical demonstration.

REMARK 4.5. We can also restrict our output functions to a smaller subclass of functions (say $C[0,1]$) and then similarly restrict our measures P_D to these functions. This process provides us with the characterization that differential privacy over the sub field $\mathcal{F}_0 \implies$ differential privacy over $\mathcal{F}$, the σ-field generated by $\mathcal{F}_0$.

5. Achieving Differential Privacy for Functions

The previous sections have been very theoretical, and all looked at privacy through a very mathematical lens. Now, we consider some examples of mechanisms which output functional data in a way which achieves differential privacy.

5.1. The Exponential Mechanism. The first way that one may think to provide this guarantee of privacy is via the "exponential mechanism". This technique considered in [**10**]. Essentially, we will construct a (finite) set of real valued functions $E = \{e_1, \ldots, e_n\}$ where every f_D under consideration has a "reasonable" representation as one of these e_i functions. Note that here reasonable is taken to mean a good approximation with respect to a distance function d.

When we are given D as input then, we choose a function to output by sampling E with probabilities:

$$P_D(e_i) \propto \exp\left\{\frac{\epsilon}{2s}d(e_i, f_D)\right\}, \qquad s \stackrel{\text{def}}{=} \sup_{D \sim D'} d(f_D, f_{D'}).$$

In [**10**] it is shown that in fact this mechanism achieves the even stronger notion of $\epsilon-$differential privacy. The similarities between this technique and discretizing the function space $\mathbb{R}^T$ are abundant, and in [**9**] it is claimed that this is essentially the same process.

5.2. Gaussian Process Noise. The main contributions of [**9**] are:

- Giving an explicit definition of differential privacy for functions, and a treatment of this definition in a single source.
- Providing the following technique where adding Gaussian noise in a specific way to our functions allows us to achieve the differential privacy guarantee up release of our data.

DEFINITION 5.1. A *Gaussian Process* indexed by T is a collection of random variables $\{X_t | t \in T\}$, for which each finite subset is distributed as a multivariate Gaussian for reference see [**1**] or [**2**]. A sample from a Gaussian process may be considered as a function from $T \to \mathbb{R}$ by examining the sample path $t \to X_t$. Notice that this Gaussian process is determined entirely by:

$$m(t) = \mathbb{E}(X_t), \qquad K(t_1, t_2) = \mathrm{Cov}(X_{t_1}, X_{t_2})$$

where $m(t) : T \to \mathbb{R}$ and $K(\cdot, \cdot) : T^2 \to \mathbb{R}$.

For any finite subset $S \subseteq T$ the random vector $\{X_t | t \in S\}$ has a normal distribution with the means and variances/covariances given by the above functions.

THEOREM 5.2. *Let G be the sample path of a Gaussian process having mean zero and covariance given by $K(\cdot, \cdot)$. Let m denote the Gram matrix:*

$$M(x_1, \ldots, x_n) = \begin{pmatrix} K(x_1, x_1) & \cdots & K(x_1, x_n) \\ \vdots & \ddots & \vdots \\ K(x_n, x_1) & \cdots & K(x_n, x_n) \end{pmatrix}$$

Let $\{f_D | D \in \mathfrak{D}\}$ be a family of functions indexed by databases. Then the release of

$$\tilde{f}_D = f_D + \frac{\Delta c(\delta)}{\epsilon} G$$

is an (ϵ, δ)-differentially private mechanism whenever:

$$\sup_{D \sim D'} \sup_{n < \infty} \sup_{\{x_1, \ldots, x_n\}} \left|\left| M^{-1/2}(x_1, \ldots, x_n) \begin{pmatrix} f_D(x_1) - f_{D'}(x_1) \\ \vdots \\ f_D(x_n) - f_{D'}(x_n) \end{pmatrix} \right|\right|_2 \leq \Delta.$$

PROOF. See [**9**] for a complete proof. □

6. Functions from a Reproducing Kernel Hilbert Space

[**4**] gives a great treatment of the theory of Reproducing Kernel Hilbert Spaces (RKHS's). In light of that, we make the following few notes.

A RKHS also known as a Proper Hilbert Space is a Hilbert Space $\mathcal{H}$ generated by considering the closure of those functions in the underlying space $\mathcal{H}_0$ which can be represented as a linear combination of the kernel K. It is noted in [**9**] that these spaces give us an easy avenue for which to acquire bounds of the form required in Theorem 5.2.

This is a result of the following theorem which allows us to bound the normed quantity in Theorem 5.2 by the $|| \cdot ||_{\mathcal{H}}$ where $\mathcal{H}$ is our RKHS. The importance of this bound will become clear with the corollary following Theorem 6.1.

THEOREM 6.1. *For $f \in \mathcal{H}$ where $\mathcal{H}$ is the RKHS corresponding to the kernel function K, and for any finite sequence $S = (x_1, \ldots, x_n)$ of distinct points in T, we have:*

$$\left\| \begin{pmatrix} K(x_1,x_1) & \cdots & K(x_1,x_n) \\ \vdots & \ddots & \vdots \\ K(x_n,x_1) & \cdots & K(x_n,x_n) \end{pmatrix}^{-1/2} \begin{pmatrix} f(x_1) \\ \vdots \\ f(x_n) \end{pmatrix} \right\|_2 \leq ||f||_{\mathcal{H}}.$$

COROLLARY 6.2. For $\{f_D | D \in \mathfrak{D}\} \subseteq \mathcal{H}$, the release of

$$\tilde{f}_D \stackrel{\text{def}}{=} f_D + \frac{\Delta c(\delta)}{\epsilon} G$$

is (ϵ, δ)-differentially private (again, with respect to the cylinder σ-field) whenever we have

$$\sup_{D \sim D'} ||f_D - f_{D'}||_{\mathcal{H}},$$

and G is the sample path of a Gaussian process having mean zero and covariance function K, given by the reproducing kernel of $\mathcal{H}$.

7. Applications

Below we discuss some potential applications of the machinery developed in [**9**].

7.1. Kernel Density Estimation. [**9**] has a good example of using (under some smoothness assumptions) this technique to release an estimate for a density function (in the example having two peaks) in an (ϵ, δ)-differentially private manner. The key features of the underlying distribution are preserved.

Additional analysis could be done to find probabilistic bounds on the risk of such approximations. In [**9**] an $O(h^4 + \frac{c_2}{nh^d})$ rate is given, however the risk rate for general kernel density estimation (released with no privacy mechanism) is known up to constants, so determining these constants in this case could be interesting.

Additionally, a seperate but similar technique for private data release is sorted out in the article. When the kernel density estimation is done with this technique, a much "uglier" (less smooth) curve is obtained for release. Determining the "error" of this curve as opposed to that of the curve obtained with the first method could also be worthwhile.

7.2. Mathematical Application. In certain fields of mathematics (primarily several complex variables) the underlying kernel function can be used to define a new metric on our spaces (see [**6**]). This Bergman metric can then be used to define local coordinates for multivariate spaces of complex variables. One question of interest (known as Lu-Qi-Keng's conjecture) asks whether or not (and if so where) certain Kernel functions obtain zeroes.

Two potential directions arise here:

- Can we release these sets of local coordinates (which take the form of functions) in a differentially private manner? These types of questions have practical applications in many cryptographic settings.
- Can we use the density estimation techniques to determine (in a computationally friendly way) when certain distributions obtain zeroes?

While the second question is less worked on, I believe that the first bullet could seriously benefit from work in this direction.

7.3. Machine Learning. In the last week, a Master's thesis [**3**] hit the Arxiv which uses Deep Learning techniques to analyze uses of the smart watch technology. The primary direction of the paper is to determine whether smart watches are applicable to "spying". While this isn't of as much interest to me, the thesis does make me aware that these devices are constantly gathering information about the wearer. Some of this information may be useful in the sciences and so having a way to release it differentially privately is important.

Thinking back to the handwriting example, I feel that it would be curious to see if the data which one of these watches collects (in the form of $x(t)$– the horizontal component collected while writing, and $y(t)$ – the vertical component collected during a writing period) could be used to first determine with some accuracy what a person was writing, and then whether this data could be released in a differentially private way which preserved some of the features of the author's penmanship.

It seems that looking at the $x(t)$ and $y(t)$ data separately is a trick used in [**12**], as these individual curves tend to be much more well behaved (resembling those present in [**9**]). Then, another question arises where, since differential privacy in the traditional sense enjoys certain composition properties, are these results also valid in the functional data setting?

References

[1] R.J. Adler. *An Introduction to Continuity, Extrema, and Related topics for General Gaussian Processes*. Institute for Mathematical Statistics, 1990.

[2] R.J. Adler and J.E. Taylor. *Random Fields and Geometry*. Springer Monographs in Mathematics, 1 edition, June 2007.

[3] T. Beltramelli and S. Risi. Deep-Spying: Spying using Smartwatch and Deep Learning. *ArXiv e-prints*, December 2015.

[4] A. Bertinet and Thomas C. Agnan. *Reproducing Kernel Hilbert spaces in Probability and Statistics*. Kluwer Academy Publishers, 2004.

[5] P. Billingsley. *Probability and Measure*. Wiley-Interscience, 3 edition, 1995.

[6] Harold P. Boas. Lu qi-keng's problem. *J. Korean Math. Soc. 37*, 2000.

[7] Cynthia Dwork. Differential privacy. *Proceedings of the International Coloquium on Autotmata, Languages, and Programming (ICALP)*, 2006.

[8] Cynthia Dwork and Aaron Roth. *The Algorithmic Foundations of Differential Privacy*. 2014.

[9] Rob Hall, Alessandro Rinaldo, and Larry Wasserman. Differential privacy for functions and functional data. 2013.

[10] F. McSherry and K. Talwar. Mechanism design via differential privacy. *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, 2007.

[11] A. Narayanan and V. Shmatikov. How To Break Anonymity of the Netflix Prize Dataset. *eprint arXiv:cs/0610105*, October 2006.

[12] James Ramsay and B. W. Silverman. *Functional Data Analysis*. Springer-Verlag New York, 2005.

[13] L. Sweeney. Simple demographics often identify people uniquely. *Carnegie Mellon University, Data Privacy*, 2000.