Rademacher Observations, Private Data, and Boosting

Richard Nock[†] RICHARD.NOCK@NICTA.COM.AU
Giorgio Patrini[†] GIORGIO.PATRINI@NICTA.COM.AU
Arik Friedman[‡] ARIK.FRIEDMAN@NICTA.COM.AU
NICTA & {[†]The Australian National University, [‡]The University of New South Wales}, Sydney, Australia

Abstract

The minimization of the logistic loss is a popular approach to batch supervised learning. Our paper starts from the surprising observation that, when fitting linear classifiers, the minimization of the logistic loss is *equivalent* to the minimization of an exponential *rado*-loss computed (i) over transformed data that we call Rademacher observations (rados), and (ii) over the *same* classifier as the one of the logistic loss. Thus, a classifier learnt from rados can be *directly* used to classify *observations*. We provide a learning algorithm over rados with boosting-compliant convergence rates on the *logistic loss* (computed over examples). Experiments on domains with up to millions of examples, backed up by theoretical arguments, display that learning over a small set of random rados can challenge the state of the art that learns over the *complete* set of examples. We show that rados comply with various privacy requirements that make them good candidates for machine learning in a privacy framework. We give several algebraic, geometric and computational hardness results on reconstructing examples from rados. We also show how it is possible to craft, and efficiently learn from, rados in a differential privacy framework. Tests reveal that learning from differentially private rados brings non-trivial privacy vs accuracy tradeoffs.

1. Introduction

This paper deals with the following fundamental question:

> *What information is sufficient for learning, and what guarantees can it bring that regular data cannot* ?

By "regular", we mean the usual inputs provided to a learner. In our context of batch supervised learning, this is a training set of examples, each of which is an observation with a class, and learning means inducing in reduced time an accurate function from observations to classes, a *classifier*. It turns out that we do not need the detail of classes to learn a linear classifier: an aggregate, whose size is the dimension of the observation space, is minimally sufficient, the mean operator (Patrini et al., 2014).

But do we need examples ?

This perhaps surprising and non-trivial question is becoming crucial now that the nature of stored and processed signals intelligence data is heavily debated in the public sphere (Landau, 2015; Sproull et al., 2015). In the context of machine learning (ML), the objective of being accurate is more and more frequently subsumed by more complex goals, sometimes involving challenging tradeoffs in which accuracy does not ultimately appear in the topmost requirements. Privacy is one such crucial goal (Duchi et al., 2014; Enserink & Chin, 2015; Goroff, 2015). There are various models to capture the privacy requirement, such as secure multi-party computation and differential privacy (DP, (Dwork & Roth, 2014)). The former usually relies on cryptographic protocols, which can be heavy even for bare classification and simple algorithms (Bost et al., 2014). The latter usually relies on the power of randomization to ensure that any "local" change cannot be spotted from the output delivered (Dwork et al., 2010; Dwork & Roth, 2014). In a ML setting, randomization can be performed at various stages, from the examples to the output of a classifier. We focus on the upstream stage of the process, *i.e.* the input to the learner, which grants the benefits that *all* subsequent stages also comply with differential privacy. Randomization has its power: it also has its limits in this case, as it may significantly degrade the performance of learners.

The way we address this problem starts from a surprising observation, whose relevance to supervised ML goes beyond learning with private data: learning a linear classifier over examples throughout the minimization of the expected logistic loss is equivalent to learning *the same classifier*

Proceedings of the 32^{nd} International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37. Copyright 2015 by the author(s).

by minimizing an exponential loss over a complete set of transformed data that we call *Rademacher observations*, rados. Each rado is the sum of *edge vectors* over examples (edge = observation × label). We also show that efficient learning from all rados may also be achieved when carried out over *subsets* of all possible rados.

This is our first contribution, and we expect it to be useful in several other areas of supervised learning. In the context of learning with private data, our other contributions can be summarized as showing how rados may yield new privacy guarantees — not limited to differential privacy — while authorizing boosting-compliant rates for learning. More precisely, our second contribution is to propose a rado-based learning algorithm, which has boosting-compliant convergence rates over the *logistic loss computed over the examples*. Thus, we learn an accurate classifier over rados, and the same classifier is accurate over examples as well.

The fact that efficient learning may be achieved through subset of rados is interesting because it opens the problem of designing this particular subset to address domain-specific requirements that add to the ML accuracy requirement. Among our other contributions, we provide one important design example, showing how to build differentially private mechanisms for rado delivery, such as when protecting specific sensitive features in data. Experiments confirm in this case that learning from differentially private rados may still be competitive with learning from examples. We provide another design which pairs to our rado-based boosting algorithm, with the crucial property that when examples have been DP-protected by the popular Gaussian mechanism (Dwork & Roth, 2014), the joint pair (rado delivery design, boosting algorithm) may achieve convergence rates *comparable to the noise-free* setting with high probability, even over strong DP protection regimes. Our last contribution is to show that rados may protect the privacy of the original examples not only in the DP framework, but also from several algebraic, geometric and even computational-complexity theoretic standpoints.

The remainder of this paper is organized as follows. Section §2 presents Rademacher observations, shows the equivalence between learning from examples and learning from rados, and how learning from subsets of rados may be sufficient for efficient learning; §3 presents our rado-based boosting algorithm, and §4 presents experiments with this algorithm; §5 presents our results in DP models, §6 presents related experiments; §7 provides results on the hardness of reconstructing examples from rados from algebraic, geometric and computational standpoints. To keep a readable paper, proofs and additional experiments are given in a companion ArXiv paper (Nock et al., 2015).

2. Rados and supervised learning

Let $[n] = \{1, 2, ..., n\}$. We are given a set of m examples $\mathcal{S} \doteq \{(\boldsymbol{x}_i, y_i), i \in [m]\}$, where $\boldsymbol{x}_i \in \mathcal{X} \subseteq \mathbb{R}^d$ is an observation and $y_i \in \{-1, 1\}$ is a label, or class. $\mathcal{X}$ is the domain. A linear classifier $\boldsymbol{\theta} \in \Theta$ for some fixed $\Theta \subseteq \mathbb{R}^d$ gives a label to $\boldsymbol{x} \in \mathcal{X}$ equal to the sign of $\boldsymbol{\theta}^\top \boldsymbol{x} \in \mathbb{R}$. Our results can be lifted to kernels (at least with finite dimension feature maps) following standard arguments (Quadrianto et al., 2009). We let $\Sigma_m \doteq \{-1, 1\}^m$.

Definition 1 *For any $\boldsymbol{\sigma} \in \Sigma_m$, the Rademacher observation $\boldsymbol{\pi}_{\boldsymbol{\sigma}}$ with signature $\boldsymbol{\sigma}$ is $\boldsymbol{\pi}_{\boldsymbol{\sigma}} \doteq (1/2) \cdot \sum_i (\sigma_i + y_i)\boldsymbol{x}_i$.*

The simplest way to randomly sample rados is to pick $\boldsymbol{\sigma}$ as i.i.d. Rademacher variables, hence the name. Reference to $\mathcal{S}$ is implicit in the definition of $\boldsymbol{\pi}_{\boldsymbol{\sigma}}$. A Rademacher observation sums *edge vectors* (the terms $y_i\boldsymbol{x}_i$), over the subset of examples for which $y_i = \sigma_i$. When $\boldsymbol{\sigma} = \boldsymbol{y}$ is the vector of classes, $\boldsymbol{\pi}_{\boldsymbol{\sigma}} = m\boldsymbol{\mu}_{\mathcal{S}}$ is m times the mean operator, a minimal sufficient statistics for the class (Quadrianto et al., 2009; Patrini et al., 2014). Thus, up to the normalization by m, any rado is a minimal sufficient statistic for the class in a subset of the training sample. A popular approach to learn $\boldsymbol{\theta}$ over $\mathcal{S}$ is to minimize the surrogate risk $F_{\log}(\mathcal{S}, \boldsymbol{\theta})$ built from the logistic loss (logloss):

$$F_{\log}(\mathcal{S}, \boldsymbol{\theta}) \doteq \frac{1}{m}\sum_i \log\left(1 + \exp\left(-y_i\boldsymbol{\theta}^\top\boldsymbol{x}_i\right)\right) . \quad (1)$$

We define the *exponential rado-risk* $F^r_{\mathrm{exp}}(\mathcal{S}, \boldsymbol{\theta}, \mathcal{U})$, computed on any $\mathcal{U} \subseteq \Sigma_m$ with cardinal $|\mathcal{U}| = n$, as:

$$F^r_{\mathrm{exp}}(\mathcal{S}, \boldsymbol{\theta}, \mathcal{U}) \doteq \frac{1}{n}\sum_{\boldsymbol{\sigma}\in\mathcal{U}} \exp\left(-\boldsymbol{\theta}^\top\boldsymbol{\pi}_{\boldsymbol{\sigma}}\right) . \quad (2)$$

It turns out that $F_{\log} = g(F^r_{\mathrm{exp}})$ for some continuous strictly increasing g and specific choice of $\mathcal{U}$ (in fact, $\mathcal{U} = \Sigma_m$); hence, minimizing one criterion is equivalent to minimizing the other and *vice versa*. This is formalized below.

Lemma 2 *The following holds true, for any $\boldsymbol{\theta}$ and $\mathcal{S}$:*

$$F_{\log}(\mathcal{S}, \boldsymbol{\theta}) = \log(2) + \frac{1}{m}\log F^r_{\mathrm{exp}}(\mathcal{S}, \boldsymbol{\theta}, \Sigma_m) . \quad (3)$$

Lemma 2 shows that learning with examples via the minimization of $F_{\log}(\mathcal{S}, \boldsymbol{\theta})$, and learning with all rados via the minimization of $F^r_{\mathrm{exp}}(\mathcal{S}, \boldsymbol{\theta}, \Sigma_m)$, are essentially equivalent tasks. Since the cardinal $|\Sigma_m| = 2^m$ is exponential, it is unrealistic, even on moderate-size samples, to pick that latter option. This raises however a very interesting question: if we replace Σ_m by subset $\mathcal{U}$ of size $\ll 2^m$,what does the relationship between examples and rados in eq. (3) become? We answer this question under the setting that:

(i) instead of Σ_m, we consider a predefined $\Sigma_r \subseteq \Sigma_m$;

(ii) instead of considering $\mathcal{U} = \Sigma_r$, we sample uniformly i.i.d. $\mathcal{U} \sim \Sigma_r$ for $n \geq 1$ rados.

While (ii) is directly targeted at reducing the number of rados, (i) is an upper-level strategic design to tackle additional constraints, such as differential privacy. We now need following definition of the *logistic rado-risk*:

$$F^r_{\log}(\mathcal{S}, \boldsymbol{\theta}, \mathcal{U}) \quad \doteq \quad \log(2) + \frac{1}{m} \log F^r_{\exp}(\mathcal{S}, \boldsymbol{\theta}, \mathcal{U}) \ , (4)$$

for any $\mathcal{U} \subseteq \Sigma_m$, so that $F_{\log}(\mathcal{S}, \boldsymbol{\theta}) = F^r_{\log}(\mathcal{S}, \boldsymbol{\theta}, \Sigma_m)$. We also define the open ball $\mathcal{B}(\mathbf{0}, r) \doteq \{\boldsymbol{x} \in \mathbb{R}^d : \|\boldsymbol{x}\|_2 < r\}$.

Theorem 3 *Assume $\Theta \subseteq \mathcal{B}(\mathbf{0}, r_\theta)$, for some $r_\theta > 0$. Let:*

$$\varrho \quad \doteq \quad \frac{\sup_{\boldsymbol{\theta}' \in \Theta} \max_{\boldsymbol{\pi_\sigma} \in \Sigma_r} \exp(-\boldsymbol{\theta}'^\top \boldsymbol{\pi_\sigma})}{F^r_{\exp}(\mathcal{S}, \boldsymbol{\theta}, \Sigma_r)} \ ,$$

$$\varrho' \quad \doteq \quad \frac{F^r_{\exp}(\mathcal{S}, \boldsymbol{\theta}, \Sigma_r)}{F^r_{\exp}(\mathcal{S}, \boldsymbol{\theta}, \Sigma_m)} \ ,$$

where Σ_r follows (i) above. Then $\forall \eta > 0$, there is probability $\geq 1 - \eta$ over the sampling of $\mathcal{U}$ in (ii) above that:

$$F_{\log}(\mathcal{S}, \boldsymbol{\theta}) \ \leq \ F^r_{\log}(\mathcal{S}, \boldsymbol{\theta}, \mathcal{U}) + Q - \frac{1}{m} \cdot \log\left(1 - \frac{q}{\sqrt{n}}\right) (5)$$

with

$$q = \Omega\left(\varrho \cdot \sqrt{r_\theta \max_{\Sigma_r} \|\boldsymbol{\pi_\sigma}\|_2 + d \log \frac{2en}{d} + \log \frac{1}{\eta}}\right) \ (6)$$

and $Q \doteq -(1/m) \cdot \log \varrho'$ satisfies $Q = 0$ if $\Sigma_r = \Sigma_m$ and

$$Q \quad \leq \quad r_\theta \left(\|\boldsymbol{\nabla_\theta} F^r_{\log}(\mathcal{S}, \boldsymbol{\theta}, \Sigma_m)\|_2 + \overline{\pi}_r\right) \qquad (7)$$

otherwise, letting $\overline{\pi}_r \doteq \|\mathbb{E}_{\boldsymbol{\sigma} \sim \Sigma_r}(1/m) \cdot \boldsymbol{\pi_\sigma}\|_2$. Furthermore, $\forall 0 \leq \beta < 1/2$, if m is sufficiently large, then letting $\pi^*_r \doteq \max_{\Sigma_r} \|(1/m) \cdot \boldsymbol{\pi_\sigma}\|_2$, ineq. (5) becomes:

$$F_{\log}(\mathcal{S}, \boldsymbol{\theta}) \ \leq \ F^r_{\log}(\mathcal{S}, \boldsymbol{\theta}, \mathcal{U}) + Q + O\left(\frac{\varrho}{m^\beta} \cdot \sqrt{\frac{r_\theta \pi^*_r}{n} + \frac{d}{nm} \log \frac{2en}{d\eta}}\right) \ (8)$$

Theorem 3 does not depend on the algorithm that learns $\boldsymbol{\theta}$. The right-hand side of ineq. (5) shows two penalties. Q arises from the choice of Σ_r and is therefore structural. Regardless of Σ_r, when the classifier is reasonably accurate over all rados and expected examples edges in Σ_r average to a ball of reduced radius, the upperbound on Q in ineq. (7) can be very small. The other penalty, which depends on q, is statistical and comes from the sampling in Σ_r. Theorem 3 shows that when $\Sigma_r = \Sigma_m$, even when $n \ll m$, the minimization of $F^r_{\log}(\mathcal{S}, \boldsymbol{\theta}, \mathcal{U})$ may still bring, with high probability, guarantees on the minimization of $F_{\log}(\mathcal{S}, \boldsymbol{\theta})$. Thus, a lightweight optimization procedure over a small number of rados may bring guarantees on the minimization of the expected logloss over *examples* for the *same* classifier. The following Section exhibits one such algorithm.

Algorithm 1 Rado boosting (RADOBOOST)

```
**Input** set of rados $\mathcal{S}^r \doteq \{\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, ..., \boldsymbol{\pi}_n\}$; $T \in \mathbb{N}_*$;
Step 1 : let $\boldsymbol{\theta}_0 \leftarrow \mathbf{0}$, $\boldsymbol{w}_0 \leftarrow (1/n)\mathbf{1}$ ;
Step 2 : **for** $t = 1, 2, ..., T$
    Step 2.1 : $[d] \ni \iota(t) \leftarrow \textsc{wfi}(\mathcal{S}^r, \boldsymbol{w}_t)$;
    Step 2.2 : let
        $r_t \leftarrow \frac{1}{\pi_{*\iota(t)}} \sum_{j=1}^{n} w_{tj} \pi_{j\iota(t)}$ ;   (9)
        $\alpha_t \leftarrow \frac{1}{2\pi_{*\iota(t)}} \log \frac{1 + r_t}{1 - r_t}$ ;   (10)
    Step 2.3 : **for** $j = 1, 2, ..., n$
        $w_{(t+1)j} \leftarrow w_{tj} \cdot \left(\frac{1 - \frac{r_t \pi_{j\iota(t)}}{\pi_{*\iota(t)}}}{1 - r_t^2}\right)$ ;   (11)
**Return** $\boldsymbol{\theta}_T$ defined by $\theta_{Tk} \doteq \sum_{t:\iota(t)=k} \alpha_t$ , $\forall k \in [d]$;
```

3. Boosting using rados

Algorithm 1 provides a boosting algorithm, RADOBOOST, that learns from a set of Rademacher observations $\mathcal{S}^r \doteq \{\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, ..., \boldsymbol{\pi}_n\}$. Their (unknown) Rademacher assignments are denoted $\mathcal{U} \doteq \{\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2, ..., \boldsymbol{\sigma}_n\} \subseteq \Sigma_m$. These rados have been computed from some sample $\mathcal{S}$, unknown to RADOBOOST. In the statement of the algorithm, π_{jk} denotes coordinate k of $\boldsymbol{\pi}_j$, and $\pi_{*k} \doteq \max_j |\pi_{jk}|$. More generally, the coordinates of some vector $\boldsymbol{z} \in \mathbb{R}^d$ are denoted $z_1, z_2, ..., z_d$. Step 2.1 gets a feature index $\iota(t)$ from a *weak feature index oracle*, WFI. In its general form, WFI returns a feature index maximizing $|r_t|$ in (9). The weight update was preferred to AdaBoost's because rados can have large feature values and the weight update prevents numerical precision errors that could otherwise occur using AdaBoost's exponential weight update. We now prove a key Lemma on RADOBOOST, namely the fast convergence of the exponential rado-risk $F^r_{\exp}(\mathcal{S}, \boldsymbol{\theta}, \mathcal{U})$ under a weak learning assumption (**WLA**). We shall then obtain the convergence of the logistic rado-risk (4), and, via Theorem 3, the convergence with high probability of $F_{\log}(\mathcal{S}, \boldsymbol{\theta})$.

(**WLA**) $\exists \gamma > 0$ such that $\forall t \geq 1$, the feature returned by WFI in Step 2.2 (9) satisfies $|r_t| \geq \gamma$.

				AdaBoost	AdaBoost(n)		RadoBoost			
Domain	m	d	100σ	err$\pm\sigma$	err$\pm\sigma$	$\frac{n}{m}$	err$\pm\sigma$	$\frac{n}{2^m}$	p	p'
Abalone	4 177	8	–	22.96±1.44	23.20±1.44	0.24	25.14±1.83	[3:−[1:3]]	ε	ε
Wine-white	4 898	11	1	30.93±3.42	30.44±3.25	0.20	32.48±3.55	[3:−[1:3]]	ε	ε
Magic	19 020	10	–	21.07±0.98	20.91±0.99	0.05	22.75±1.51	[3:−[5:3]]	ε	0.01
EEG	14 980	14	14	46.04±1.38	44.36±1.99	0.07	44.23±1.73	[4:−[4:3]]	ε	0.86
Hardware	28 179	95	–	16.82±0.72	16.76±0.73	0.04	7.61±3.24	[2:−[8:3]]	ε	ε
Twitter	583 250	77	44	53.75±1.48	53.09±11.23	[1:−3]	6.00±0.77	[1:−[1:5]]	ε	ε
SuSy	5 000 000	17	–	27.76±0.14	27.43±0.19	[2:−4]	27.26±0.55	[1:−[1:6]]	0.02	0.39
Higgs	11 000 000	28	–	42.55±0.19	45.39±0.28	[9:−5]	47.86±0.06	[1:−[1:7]]	ε	ε

Table 1. Comparison of RadoBoost (n random rados), AdaBoost (Schapire & Singer, 1999) (full training fold) and AdaBoost(n) (n random examples in training fold); domains ranked in increasing $d \cdot m$ value. Column "n/m" (resp. "$n/2^m$") for AdaBoost(n) (resp RadoBoost) is proportion of training data wrt fold size (resp. full set of rados). Notation $[a$:$b]$ is shorthand for $a \times 10^b$. Column "100σ" is the number of features with outlier values distant from the mean by more than 100σ in absolute value. Column p (resp. p') is p-value for a two-tailed paired t-test on AdaBoost (resp. AdaBoost(n)) vs RadoBoost. ε means < 0.01.

Lemma 4 *Suppose the* (**WLA**) *holds. Then after T rounds of boosting in* RadoBoost, *the following upperbound holds on the exponential rado-loss of $\boldsymbol{\theta}_T$:*

$$F^r_{\mathrm{exp}}(\mathcal{S}, \boldsymbol{\theta}_T, \mathcal{U}) \quad \leq \quad \exp\left(-T\gamma^2/2\right) \quad . \tag{12}$$

We now consider Theorem 3 with $\Sigma_r = \Sigma_m$, and therefore $Q = 0$. Blending Lemma 4 and Theorem 3 using (4) yields that, under the (**WLA**), we may observe with high probability (again, fixing $\Sigma_r = \Sigma_m$, so $Q = 0$ in Theorem 3):

$$F_{\log}\left(\mathcal{S}, \boldsymbol{\theta}_T\right) \quad \leq \quad \log(2) - \frac{T\gamma^2}{2m} + Q' \ , \tag{13}$$

where Q' is the rightmost term in ineq. (5) or ineq. (8). So provided $n \ll 2^m$ is sufficiently large, minimizing the exponential rado-risk over a *subset of rados* brings a classifier whose average logloss on the *whole set of examples* may decrease at rate $\Omega(\gamma^2/m)$ under a weak learning assumption made over *rados* only. This rate competes with those for direct approaches to boosting the logloss (Nock & Nielsen, 2008), and we now show that our weak learning assumption is also essentially equivalent to the one done in boosting over examples (Schapire & Singer, 1999). Let us rewrite $r_t(\boldsymbol{w})$ as the normalized edge in (9), making explicit the dependence in the current rado weights. Let

$$r^{ex}_t(\tilde{\boldsymbol{w}}) \quad \doteq \quad \frac{1}{x_{*\iota(t)}} \sum_{i=1}^{m} w_i x_{i\iota(t)} \tag{14}$$

be the normalized edges for the same feature $\iota(t)$ as the one picked in step 2.1 of RadoBoost, but computed over examples using some weight vector $\tilde{\boldsymbol{w}} \in \mathbb{P}^m$; here, $\mathbb{P}^m$ is the m-dim probability simplex and $x_{*\iota(t)} \doteq \max_i |x_{ik}|$.

Lemma 5 $\forall \boldsymbol{w}_t \in \mathbb{P}^n$, $\forall \gamma > 0$, *there exists* $\tilde{\boldsymbol{w}} \in \mathbb{P}^m$ *and* $\gamma^{ex} > 0$ *such that* $|r_t(\boldsymbol{w}_t)| \geq \gamma$ *iff* $|r^{ex}_t(\tilde{\boldsymbol{w}})| \geq \gamma^{ex}$.

The proof of the Lemma gives clues to explain why the presence of outlier feature values may favor RadoBoost.

4. Basic experiments with RadoBoost

We have compared RadoBoost to its main contender, AdaBoost (Schapire & Singer, 1999), using the same weak learner; in AdaBoost, it returns a feature maximizing $|r_t|$ as in eq. (14). In these basic experiments, we have deliberately not optimized the set of rados in which we sample $\mathcal{U}$ for RadoBoost; hence, we have $\Sigma_r = \Sigma_m$. We have performed comparisons with 10 folds stratified cross-validation (CV) on 16 domains of the UCI repository (Bache & Lichman, 2013) of varying size. For space considerations, Table 1 presents the results on the 8 largest domains. SI presents the complete experiments. Each algorithm was ran for a total number of $T = 1000$ iterations; furthermore, the classifier kept for testing is the one minimizing the empirical risk throughout the T iterations; in doing so, we also assessed the early convergence of algorithms. We fixed $n = \min\{1000, \text{train fold size}/2\}$. Table 1 displays that RadoBoost compares favourably to AdaBoost, and furthermore it tends to be all the better as m and d increase. On some domains like Hardware and Twitter, the difference is impressive and clearly in favor of RadoBoost. Experimentally, we interpret it by the fact that random rados may have large norms on big domains, which may yield large boosting leveraging coefficients. On domains like Twitter, this boosts convergence. Also, outlier features (see column 100σ in Table 1) can trick AdaBoost in picking the wrong sign for α_t for a large number of iterations. This drawback can be easily corrected (SI (Nock et al., 2015)) by enforcing minimal $|r_t|$ values. This improves AdaBoost on Hardware and Twitter. Improvements observed on RadoBoost are even more favorable.

5. Rados and differential privacy

We discuss the delivery of rados to comply with DP constraints and their eventual impact on boosting. We thus adress both levels (i+ii) of rado delivery in §2. Our model is the standard DP model (Dwork & Roth, 2014). Intuitively,

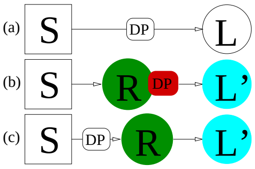


Figure 1. Summary of the DP-related contributions of Section 5 (in color). (a) : usual DP mechanism that protects examples (S) prior to delivery to learner (L); (b) : mechanism that crafts differentially private rados (R) from unprotected examples (§5.1); (c) : mechanism crafting rados from DP examples with objective to improve performances of rado-based learner L' (§5.2).

Algorithm 2 Feature-wise DP rados (DP-FEAT)

```
Input set of examples $\mathcal{S}$, sensitive feature $j_* \in [d]$, number of rados $n$, differential privacy parameter $\epsilon > 0$;
Step 1 : let $\beta \leftarrow 1/(1+\exp(\epsilon/2)) \in [0, 1/2)$;
Step 2 : sample $\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2, ..., \boldsymbol{\sigma}_n$ i.i.d. (uniform) in $\Sigma_m^{\beta, j_*}$;
Return set of rados $\{\boldsymbol{\pi}_{\boldsymbol{\sigma}} : \boldsymbol{\sigma}$ sampled in Step 2$\}$;
```

an algorithm is DP if for any two neighboring datasets, it assigns similar probability to any possible output O. In other words, any particular record has only limited influence on the probability of any given output of the algorithm, and therefore the output discloses very little information about any particular record in the input. Formally, a randomized algorithm $\mathcal{A}$ is (ϵ, δ)-differentially-private (Dwork et al., 2006) for some $\epsilon, \delta > 0$ iff:

$$\mathbb{P}_{\mathcal{A}}[O|\mathcal{S}] \quad \leq \quad \exp(\epsilon) \cdot \mathbb{P}_{\mathcal{A}}[O|\mathcal{S}'] + \delta, \forall \mathcal{S} \approx \mathcal{S}', O \tag{15}$$

where the probability is over the coin tosses of $\mathcal{A}$. This model is very strong, especially when $\delta = 0$, and in the context of ML, maintaining high accuracy in strong DP regimes is generally a tricky tradeoff (Duchi et al., 2014). Because rados are an intermediate step between training sample $\mathcal{S}$ and a rado-based learner, there are two ways to design rados with respect to the DP framework: crafting DP rados from unprotected examples, or crafting rados from DP examples with the aim to improve the performance of the rado-based learner (Figure 5.2). These scenarii can be reduced to the design of Σ_r.

5.1. A feature-wise DP mechanism for rados

In this Subsection, we consider a relaxation of differential-privacy, namely *feature-wise* differential privacy, where the differential privacy requirement applies to j_**-neighboring datasets*: we say that two samples $\mathcal{S}, \mathcal{S}'$ are j_**-neighbors*, noted $\mathcal{S} \approx_{j_*} \mathcal{S}'$, if they are the same except for the value of the $j_*^{th} \in [d]$ observation feature of some example. We further assume that the feature is boolean. For example, we may have a medical database containing a column representing the HIV status of a doctor's patients (1 row = a patient), and we do not wish that changing a single patient HIV status significantly changes the density of that feature's values in rados. This setting would also be very useful in genetic applications to hide in rados gene disorders that affect one or few genes. Feature-wise DP is analogous to the concept of α-label privacy (Chaudhuri & Hsu, 2011), where differential privacy is guaranteed with respect to the label. Algorithm $\mathcal{A}$ in ineq. (15) is given in Algorithm 2. It relies on the following subset $\Sigma_r \doteq \Sigma_m^{\beta, j_*} \subseteq \Sigma_m$ ($m_+ \doteq |\{i : y_i x_{ij_*} = +1\}| - (m/2)$):

$$\Sigma_m^{\beta, j_*} \doteq \{\boldsymbol{\sigma} \in \Sigma_m : \pi_{\boldsymbol{\sigma} j_*} \in [m_+ - \Delta_\beta, m_+ + \Delta_\beta]\} \quad , \tag{16}$$

with $\Delta_\beta \doteq (m/2) - \beta(m+1)$. The key feature of this mechanism is that it does not alter the examples in the sense that DP rados belong to the set of cardinal 2^m that can be generated from $\mathcal{S}$. Usual data-centered DP mechanisms would rather alter data, *e.g.* via noise injection (Goroff, 2015). Algorithm 2 exploits the fact that it is the tails of feature j_* that leak sensitive information about the feature in rados (see Figure 2).

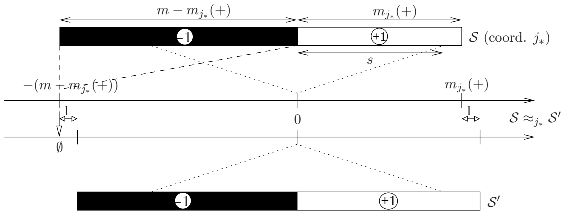


Figure 2. How DP-FEAT works: neighbor samples $\mathcal{S}$ and $\mathcal{S}'$ differ by one value for feature j_* (*i.e.* one edge coordinate, represented); the rado whose support relies only on the "-1" in $\mathcal{S}$ (dashed lines) yields infinite ratio $\mathbb{P}_{\mathcal{A}}[O|I]/\mathbb{P}_{\mathcal{A}}[O|I']$ in (15). This rado would never be sampled by DP-FEAT. On the other hand, a rado that sums an equal number s of "+1" and "-1" (dotted lines) may yield ratio very close to 1 (such a rado can be sampled by DP-FEAT).

Theorem 6 *If* $\epsilon = \Omega(1/m)$ *and* $\epsilon = o(1)$, DP-FEAT *maintains* $(n \cdot \epsilon, n \cdot \delta)$*-differential privacy on feature* j_* *for some* $\delta = o(1/m)$.

We have implemented Step 2 in Algorithm DP-FEAT in the simplest way, using Rademacher rejection sampling where each $\boldsymbol{\sigma}_j$ is picked i.i.d. as $\boldsymbol{\sigma}_j \sim \Sigma_m$ until $\boldsymbol{\sigma}_j \in \Sigma_m^{\beta, j_*}$. The following Theorem shows its algorithmic efficiency.

Theorem 7 *For any* $\eta > 0$, *let* $n_\eta^* \doteq \eta(1 - \exp(2\beta - 1))/(4\beta)$, *and let* n_R *denote the total number of rados sampled in* Σ_m *until* n *rados are found in* Σ_m^{β, j_*}. *Then for any* $\eta > 0$, *there is probability* $\geq 1 - \eta$ *that*

$$n_R \quad \leq \quad n \cdot \left\{ \begin{array}{ccc} 1 & & \textit{if} \quad n \leq n_\eta^* \\ \left\lceil \frac{1}{m D_{BE}(1-\beta \| 1/2)} \log \frac{n}{n_\eta^*} \right\rceil & & \textit{otherwise} \end{array} \right. ,$$

where D_{BE} *is the bit-entropy divergence:* $D_{BE}(p\|q) = p\log(p/q) + (1-p)\log((1-p)/(1-q))$, *for* $p, q \in (0, 1)$.

Remark that replacing Σ_m by $\Sigma_r = \Sigma_m^{\beta, j_*}$ would not necessarily impair the boosting convergence of RadoBoost trained from rados samples from DP-Feat (Lemma 4). The only systematic change would be in ineq. (13) where we would have to integrate the structural penalty Q from Theorem 3 to further upperbound $F_{\log}(\mathcal{S}, \boldsymbol{\theta}_T)$. In this case, the upperbound in (7) reveals that at least when the mean operator in Σ_m^{β, j_*} has small norm — which may be the case even when some examples in $\mathcal{S}$ have large norm — and the gradient penalty is small, then Q may be small as well.

Finally, the tail truncation design exploited in DP-Feat can be fairly simply generalized in two directions, to handle (a) real-valued features, and/or (b) several sensitive features instead of one.

5.2. Boosting from DP examples via rados

We now show how to craft rados from DP protected examples so as to approximately keep the convergence rates of RadoBoost. More precisely, since edge vectors are sufficient to learn (eq. 1), we assume that edge vectors are DP (neighbor samples, $\mathcal{S} \approx \mathcal{S}'$, would differ on one edge vector). A gold standard to protect data in the DP framework is to convolute data with noise. One popular mechanism is the Gaussian mechanism (Dwork & Roth, 2014; Hardt & Price, 2014), which convolutes data with independent Gaussian random variables $\mathcal{N}(\mathbf{0}, \varsigma^2 \mathrm{I})$, whose standard deviation ς depends on the DP requirement (ϵ, δ). Strong DP regimes are tricky to handle for learning algorithms. For example, the approximation factor ρ of the singular vectors under DP noise of the noisy power method roughly behaves as $\rho = \Omega(\varsigma/\Delta)$ (Hardt & Price, 2014) (Corollary 1.1) where $\Delta = O(d)$ is a difference between two singular values. When ς is small, this is a very good bound. When the DP requirement blows up, the bound remains relevant *if* d increases, which may be hard to achieve in practice — it is easier in general to increase m than d, which requires to compute new features for past examples.

We consider ineq. (15) with neighbors I and I' being two sets of m edge vectors differing by one edge vector, and O is a noisified set of m edge vectors generated through the Gaussian mechanism (Dwork & Roth, 2014) (Appendix A). We show the following non-trivial result: provided we design another particular Σ_r, the convergence rate of RadoBoost, *as measured over non-noisy rados*, essentially survives noise injection in the edge vectors through the Gaussian mechanism, even under strong noise regimes, as long as m is large enough. The intuition is straightforward: we build rados summing a large number of edge vectors only (this is the design of Σ_r), so that the i.i.d. noise component gets sufficiently concentrated for the algorithm to be able to learn almost as fast as in the noise-free setting. We emphasize the non-trivial fact that convergence rate is measured over the non-noisy rados, which of course RadoBoost does *not* see. The result is of independent interest in the boosting framework, since it makes use of a particular weak learner (wfi), which we call *prudential*, which picks features with $|r_t|$ (9) upperbounded.

We start by renormalizing coefficients α_t (eq. (10)) in RadoBoost by a parameter $\kappa \geq 1$ given as input, so that we now have $\alpha_t \leftarrow (1/(\kappa \pi_{*\iota(t)})) \log((1+r_t)/(1-r_t))$ in Step 2.2. It is not hard to check that the convergence rate of RadoBoost now becomes, prior to applying the (**WLA**)

$$F^r_{\log}(\mathcal{S}, \boldsymbol{\theta}_T, \mathcal{U}) \quad \leq \quad \log(2) - \frac{1}{2\kappa m} \sum_t r_t^2 \ . \qquad (17)$$

We say that wfi is λ_p-*prudential* for $\lambda_p > 0$ iff it selects at each iteration a feature such that $|r_t| \leq \lambda_p$. Edges vectors have been DP-protected as $y_i(\boldsymbol{x}_i + \boldsymbol{x}_i^r)$, with $\boldsymbol{x}_i^r \sim \mathcal{N}(\mathbf{0}, \varsigma^2 \mathrm{I})$ (for $i \in [m]$). Let $m_{\boldsymbol{\sigma}} \doteq |\{i : \sigma_i = y_i\}|$ denote the *support* of a rado, and ($m_* > 0$ fixed):

$$\Sigma_r = \Sigma_m^{m_*} \quad \doteq \quad \{\boldsymbol{\sigma} \in \Sigma_m : m_{\boldsymbol{\sigma}} = m_*\} \ . \qquad (18)$$

Theorem 8 *$\forall \mathcal{U} \subseteq \Sigma_r, \forall \tau > 0$, if $\sqrt{m_*} = \Omega(\varsigma \ln(1/\tau))$, then $\exists \lambda_p > 0$ such that* RadoBoost *having access to a λ_p-prudential weak learner returns after T iteration a classifier $\boldsymbol{\theta}_T$ which meets with probability $\geq 1 - \tau$:*

$$F^r_{\log}(\mathcal{S}, \boldsymbol{\theta}_T, \mathcal{U}) \quad \leq \quad \log(2) - \frac{1}{4\kappa m} \sum_t r_t^2 \ . \qquad (19)$$

The proof details parameters and dependencies hidden in the statement. The use of a prudential weak learner is rather intuitive in a noisy setting since α_t blows up when $|r_t|$ is close to 1. Theorem 8 essentially yield that a sufficiently large support for rados is enough to keep with high probability the convergence rate of RadoBoost within noise-free regime. Of course, the weak learner is prudential, which implies bounded $|r_t| < 1$, and furthermore the leveraging coefficients α_t are normalized, which implies smaller margins. Still, Theorem 8 is a good theoretical argument to rely on rados when learning from DP edge vectors.

6. Experiments on differential privacy

Table 2 presents a subset of the experiments carried out with RadoBoost and AdaBoost in the contexts of Subsections 5.1 and 5.2. Due to size constraints, the full Table (and more extensive experiments) can be found in SI (Nock et al., 2015). Unless otherwise stated, experimental settings (cross validation, number of rados for learning, etc.) are the same as in Section 4.

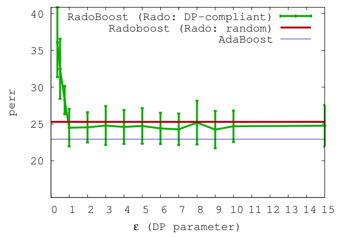
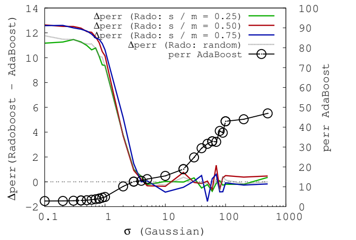
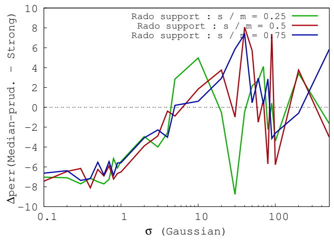
Section 5.1	Section 5.2	
	RadoBoost vs AdaBoost	RadoBoost: §5.2 vs §5.1
		
Abalone	Banknote	Transfusion

Table 2. Left table: RadoBoost on feature-wise DP rados (Subsection 5.1, showing standard deviations) vs RadoBoost on plain random rados baseline and AdaBoost baseline (trained with complete fold). Center: test error of RadoBoost *minus* AdaBoost's (also showing AdaBoost error on right axis, dotted line), for rados with fixed support s $(= m_*$, in green, red, blue) and plain random rados (dotted grey). Right: test error of RadoBoost using fixed support s rados and a prudential learner, *minus* RadoBoost using plain random rados and "strong" learner of Section 4. See text and SI (Nock et al., 2015)).

In a first set of experiments, we have assessed the impact on learning of the feature-wise DP mechanism: on each tested domain, we have selected at random a binary feature, and then used Algorithm DP-Feat to protect the feature for different values of DP parameter ϵ, in a range that covers usual DP experiments (Hsu et al., 2014) (Table 1). The main conclusion that can be drawn from the experiments is that learning from DP rados can compete with learning from random rados, and even learning from examples (AdaBoost), even for rather small ϵ.

We then have assessed the impact on learning of examples that have been protected using the Gaussian mechanism (Dwork & Roth, 2014), with or without rados, with or without a prudential weak learner for boosting, and with or without using a fixed support for rado computation. SI (Nock et al., 2015) provides extensive results for all domains but the largest ones (Twitter, SuSy, Higgs). In the central column (see also SI (Nock et al., 2015)), computing the differences between RadoBoost's error and AdaBoost's reveals that, on domains where it is beaten by AdaBoost when there is no noise, RadoBoost almost always rapidly becomes competitive with AdaBoost as noise increases. Hence, RadoBoost is a good contender from the boosting family to learn from differentially private (or noisy) data. Second, using a prudential weak learner which picks the median feature (instead of the more efficient weak learner that picks the best as in Section 4) can have RadoBoost with fixed support rados compete or beat RadoBoost with plain random rados, at least for small noise levels (see Transfusion and Magic in the right column of Table 2 and SI (Nock et al., 2015)). Replacing the median-prudential weak learner by a strong learner can actually degrade RadoBoost's results (see SI (Nock et al., 2015)). These two observations advocate in favor of the theory developed in Subsection 5.2. Finally, using rados with fixed support instead of plain random rados (Section 4) can significantly improve the performances of RadoBoost (see SI (Nock et al., 2015)).

7. From rados to examples: hardness results

The problem we address here is how we can recover examples from rados, and when we *cannot* recover examples from rados. This last setting is particularly useful from the privacy standpoint, as this may save us costly obfuscation techniques that impede ML tasks (Bost et al., 2014).

7.1. Algebraic and geometric hardness

For any $m \in \mathbb{N}_*$, we define matrix $\mathrm{G}_m \in \{0,1\}^{m\times 2^m}$ as:

$$\mathrm{G}_m \doteq \left[\begin{array}{cc} \mathbf{0}_{2^{m-1}}^\top & \mathbf{1}_{2^{m-1}}^\top \\ \mathrm{G}_{m-1} & \mathrm{G}_{m-1} \end{array}\right] \qquad (20)$$

if $m > 1$, and $\mathrm{G}_1 \doteq [0\ 1]$ otherwise ($\boldsymbol{z}_d$ denotes a vector in $\mathbb{R}^d$). Each column of G_m is the binary indicator vector for the edge vectors considered in a rado. Hereafter, we let $\mathrm{E} \in \mathbb{R}^{d\times m}$ the matrix of columnwise edge vectors from $\mathcal{S}$, $\Pi \in \mathbb{R}^{d\times n}$ the columnwise rado matrix and $\mathrm{U} \in \{0,1\}^{2^m \times n}$ in which each column gives the index of a rado computed in $\mathcal{S}^r$. By construction, we have:

$$\Pi = \mathrm{E}\mathrm{G}_m\mathrm{U} \ , \qquad (21)$$

and so we have the following elementary results for the (non) reconstruction of E (proof omitted).

Lemma 9 *(a) when recoverable, edge-vectors satisfy:* $\mathrm{E} = \Pi\mathrm{U}^\top\mathrm{G}_m^\top(\mathrm{G}_m\mathrm{U}\mathrm{U}^\top\mathrm{G}_m^\top)^{-1}$*; (b) when* U, Π, m *are known but* $n < m$*, there is not a single solution to eq. (21) in general.*

Lemma 9 states that even when U, Π and m are known, elementary constraints on rados can make the recovery of edge vectors hard — notice that such constraints are met in our experiments with RadoBoost in Sections 4 and 6.

But this represents a lot of *unnecessary* knowledge to learn from rados: RadoBoost just needs Π to learn. We now explore the guarantees that providing this sole information brings in terms of (not) reconstructing E. $\forall \mathrm{M} \in \mathbb{R}^{a\times b}$, we let $\mathcal{C}(\mathrm{M})$ denote the set of column vectors, and for any $\mathcal{C} \subseteq \mathbb{R}^d$, we let $\mathcal{C} \oplus \epsilon \doteq \cup_{\boldsymbol{z}\in\mathcal{C}} \mathcal{B}(\boldsymbol{z}, \epsilon)$. We define the Hausdorff distance, $D_{\mathrm{H}}(\mathrm{E}, \mathrm{E}')$, between E and E':

$$\begin{aligned} &D_{\mathrm{H}}(\mathrm{E}, \mathrm{E}') \\ &\quad \doteq \quad \inf\{\epsilon : \mathcal{C}(\mathrm{E}) \subseteq \mathcal{C}(\mathrm{E}') \oplus \epsilon \wedge \mathcal{C}(\mathrm{E}') \subseteq \mathcal{C}(\mathrm{E}) \oplus \epsilon\} \ . \end{aligned}$$

The following Lemma shows that if the only information known is Π, then there exist samples that bring the same set of rados $\mathcal{C}(\Pi)$ as the unknown E *but* who are at distance proportional to the "width" of the domain at hand.

Lemma 10 *For any $\Pi \in \mathbb{R}^{d\times n}$, suppose eq. (21) holds, for some unknowns $m > 0$, $\mathrm{E} \in \mathbb{R}^{d\times m}$, $\mathrm{U} \in \{0, 1\}^{2^m\times n}$. Suppose $\mathcal{C}(\mathrm{E}) \subset \mathcal{B}(\mathbf{0}, R)$ for some $R > 0$. Then there exists $\mathrm{E}' \in \mathbb{R}^{d\times(m+1)}$, $\mathrm{U}' \in \{0, 1\}^{2^{m+1}\times n}$ such that*

$$\mathcal{C}(\mathrm{E}') \subset \mathcal{B}(\mathbf{0}, R) \quad \textit{and} \quad \Pi = \mathrm{E}'\mathrm{G}_{m+1}\mathrm{U}' \ , \tag{22}$$

but

$$D_{\mathrm{H}}(\mathrm{E}, \mathrm{E}') \quad = \quad \Omega\left(\frac{R\log d}{\sqrt{d}\log m}\right) \tag{23}$$

if $m \geq 2^d$, and $D_{\mathrm{H}}(\mathrm{E}, \mathrm{E}') = \Omega(R/\sqrt{d})$ otherwise.

Hence, without any more knowledge, leaks, approximations or assumptions on the domain at hand, the recovery of E pays in the worst case a price proportional to the radius of the smallest enclosing $\mathcal{B}(\mathbf{0}, .)$ ball for the unknown set of examples. We emphasize that this inapproximability result does not rely on the computational power at hand.

7.2. Computational hardness

In this Subsection, we investigate two important problems in the recovery of examples. The first problem addresses whether we can *approximately* recover *sparse* examples from a given set of rados, that is, roughly, solve (21) with a sparsity constraint on examples. The first Lemma we give is related to the hardness of solving underdetermined linear systems for sparse solutions (Donoho & Tanner, 2005). The sparsity constraint can be embedded in the compressed sensing framework (Donoho, 2006) to yield finer hardness *and* approximability results, which is beyond the scope of our paper. We define problem "Sparse-Approximation" as:

(**Instance**) : set of rados $\mathcal{S}^r = \{\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, ..., \boldsymbol{\pi}_n\}$, $m \in \mathbb{N}_*$, $r, \ell \in \mathbb{R}_+$, $\|.\|_p$, L_p-norm for $p \in \mathbb{R}_+$;

(**Question**) : Does there exist set $\mathcal{S} \doteq \{(\boldsymbol{x}_i, y_i), i \in [m]\}$ and set $\mathcal{U} \doteq \{\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2, ..., \boldsymbol{\sigma}_n\} \in \{-1, 1\}^m$ such that:

$$\begin{aligned} \|\boldsymbol{x}_i\|_p &\leq \ell \ , \forall i \in [m] \ , \ \text{(Sparse examples)} \\ \|\boldsymbol{\pi}_j - \boldsymbol{\pi}_{\boldsymbol{\sigma}_j}\|_p &\leq r \ , \forall j \in [n] \ . \ \text{(Rado approx.)} \end{aligned}$$

Lemma 11 *Sparse-Approximation is NP-Hard.*

In the context of rados, the second problem we address has very large privacy applications. Suppose entity Ⓐ has a huge database of people (*e.g.* clients), and obtains a set of rados emitted by another entity Ⓑ. An important question that Ⓐ may ask is whether the rados observed *can* be *approximately* constructed by its database, for example to figure out which of its clients are also its competitors'. We define this as problem "Probe-Sample-Subsumption":

(**Instance**) : set of examples $\mathcal{S}$, set of rados $\mathcal{S}^r = \{\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, ..., \boldsymbol{\pi}_n\}$, $m \in \mathbb{N}_*$, $p, r \in \mathbb{R}_+$.

(**Question**) : Does there exist $\mathcal{S}' \doteq \{(\boldsymbol{x}_i, y_i), i \in [m]\} \subseteq \mathcal{S}$ and set $\mathcal{U} \doteq \{\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2, ..., \boldsymbol{\sigma}_n\} \in \{-1, 1\}^m$ such that:

$$\|\boldsymbol{\pi}_j - \boldsymbol{\pi}_{\boldsymbol{\sigma}_j}\|_p \quad \leq \quad r \ , \forall j \in [n] \ . \ \text{(Rado approx.)}$$

Lemma 12 *Probe-Sample-Subsumption is NP-Hard.*

This worst-case result calls for interesting domain-specific qualifications, such as in genetics where the privacy of individual genomes can be compromised by population-wise statistics (Homer et al., 2008; Nietfeld et al., 2011).

8. Conclusion

We have introduced novel quantities that are sufficient for efficient learning, Rademacher observations. The fact that a subset of these can replace traditional examples for efficient learning opens interesting problems on how to craft these subsets to cope with additional constraints. We have illustrated these constraints in the field of efficient learning from privacy-preserving data, from various standpoints that include differential privacy as well as algebraic, geometric and computational considerations. In that last case, results rely on NP-Hardness, and thus go beyond the "hardness" of factoring integers on which rely some popular cryptographic techniques (Bost et al., 2014). Rados are also cryptography-compliant: homomorphic encryption schemes can be used to compute rados in the encrypted domain from encrypted edge vectors or examples — rado computation can thus be easily distributed in secure multiparty computation applications. Finally, rados may allow significant memory savings for learning, and could be of use in areas where speed matters, like on-line learning.

Acknowledgments

Thanks are due to the reviewers for many useful comments about the paper. The authors are indebted to Tiberio Caetano for early discussions that brought the idea of Rademacher observations and their use in privacy related applications. Thanks are also due to Stephen Hardy and Hugh Durrant-Whyte for many stimulating discussions and feedback on the subject. NICTA is funded by the Australian Government through the Department of Communications and the Australian Research Council through the ICT Center of Excellence Program.

References

Bache, K. and Lichman, M. UCI machine learning repository, 2013.

Bost, R., Popa, R.-A., Tu, S., and Goldwasser, S. Machine learning classification over encrypted data. Cryptology ePrint Archive, Report 2014/331, 2014.

Chaudhuri, K. and Hsu, D. Sample complexity bounds for differentially private learning. In *Proc. of the* 24^{th} *COLT*, pp. 155–186, 2011.

Donoho, D.-L. Compressed sensing. *IEEE T. IT*, 52(4): 1289–1306, 2006.

Donoho, D.-L. and Tanner, J. Sparse non-negative solution of underdetermined linear equations by linear programming. *PNAS*, 102:9446–9451, 2005.

Duchi, J.-C., Jordan, M.-I., and Wainwright, M. Privacy-aware learning. *JACM*, 2014.

Dwork, C. and Roth, A. The algorithmic foudations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9:211–407, 2014.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. Calibrating noise to sensitivity in private data analysis. In *Proc. of the* 3^{rd} *TCC*, pp. 265–284, 2006.

Dwork, C., Rothblum, G.-N., and Vadhan, S.-P. Boosting and differential privacy. In *Proc. of the* 51^{st} *FOCS*, pp. 51–60, 2010.

Enserink, M. and Chin, G. The end of privacy. *Science*, 347:490–491, 2015.

Goroff, D.-L. Balancing privacy versus accuracy in research protocols. *Science*, 347:479–480, 2015.

Hardt, M. and Price, E. The noisy power method: a meta algorithm with applications. In *NIPS*27*, pp. 2861–2869, 2014.

Homer, N., Szelinger, S., Redman, M., Duggan, D., Tembe, W., Muehling, J., Pearson, J.-V., Stephan, D.-A., Nelson, S.-F., and Craig, D.-W. Resolving individuals contributing trace amounts of dna to highly complex mixtures using high-density snp genotyping microarrays. *PLoS Genetics*, 4:e100167, 2008.

Hsu, J., Gaboardi, M., Haeberlen, A., Khanna, S., Narayan, A., Pierce, B.-C., and Roth, A. Differential privacy: An economic method for choosing epsilon. In *Proc. of the* 27^{th} *IEEE CSFS*, pp. 398–410, 2014.

Landau, S. Control use of data to protect privacy. *Science*, 347:504–506, 2015.

Nietfeld, J.-J., Sugarman, J., and Litton, J.-E. The bio-pin, a concept to improve biobanking. *Nature Reviews Cancer*, 11:303–308, 2011.

Nock, R. and Nielsen, F. On the efficient minimization of classification-calibrated surrogates. In *NIPS*21*, pp. 1201–1208, 2008.

Nock, R., Patrini, G., and Friedman, A. Rademacher observations, private data, and boosting. *CoRR*, abs/1502.02322, 2015.

Patrini, G., Nock, R., Rivera, P., and Caetano, T.-S. (Almost) no label no cry. In *NIPS*27*, 2014.

Quadrianto, N., Smola, A.-J., Caetano, T.-S., and Le, Q.-V. Estimating labels from label proportions. *JMLR*, 10: 2349–2374, 2009.

Schapire, R. E. and Singer, Y. Improved boosting algorithms using confidence-rated predictions. *MLJ*, 37:297–336, 1999.

Sproull, R.-F., DuMouchel, W.-H., Kearns, M., Lampson, B.-W., Landau, S., Leiter, M.-E., Parker, E. Rindskopf, and Weinberger, P.-J. Bulk collection of signal intelligence: technical options. National Academies Press, 2015. — Committee on responding to section 5(D) of Presidential Policy Directive 28: The Feasibility of Software to Provide Alternatives to Bulk Signals Intelligence Collection.