Fast Learning from Distributed Datasets without Entity Matching

Giorgio Patrini,[1,2] **Richard Nock,**[2,1] **Stephen Hardy,**[2] **Tiberio Caetano,**[3,1,4]
Australian National University[1], NICTA[2], Ambiata[3], University of New South Wales[4]
{giorgio.patrini, richard.nock, stephen.hardy}@nicta.com.au, tiberio.caetano@gmail.com

Abstract

Consider the following scenario: two datasets/peers contain the same real-world entities described using partially shared features, *e.g.* banking and insurance company records of the same customer base. Our goal is to learn a classifier in the cross product space of the two domains, in the hard case in which no shared ID is available –*e.g.* due to anonymization. Traditionally, the problem is approached by first addressing entity matching and subsequently learning the classifier in a standard manner. We present an end-to-end solution which bypasses matching entities, based on the recently introduced concept of *Rademacher observations* (rados). Informally, we replace the minimisation of a loss over examples, which requires entity resolution, by the *equivalent* minimisation of a (different) loss over rados. We show that (i) a potentially exponential-size subset of these rados *does not require* entity matching, and (ii) the algorithm that provably minimizes the loss over rados has time and space complexities *smaller* than the algorithm minimizing the equivalent example loss. Last, we relax a key assumption, that the data is vertically partitioned among peers — in this case, we would not even know the *existence* of a solution to entity resolution. In this more general setting, experiments validate the possibility of beating even the *optimal* peer in hindsight.

1 Introduction

Learning from massively distributed data collections and multiple information sources has become a pivotal problem, yet it faces critical challenges, among which is the fact that it relies on reconstructing consistent examples from diverse features distributed between different data handling *peers*. Exhaustive search to solve this problem is simply not scalable, nor communication efficient, and sometimes not even accurate [Estrada *et al.*, 2010; Zhang *et al.*, 2015].
— A key technical message of our paper is:

Entity resolution can be bypassed to carry out supervised learning almost as accurately as if its ***solution*** *were known.*

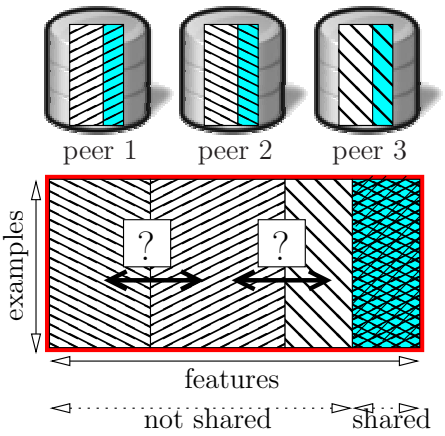

Figure 1: Schematic view of our setting (**VP**), with $p = 3$ peers. Some features (cyan) are described in each peer and one these shared features is a class. Non-shared features are split among peers. A so-called *total* sample $\mathcal{S}$ is represented by the red rectangle. All peers see different views of the same examples, but do not know who is who ("?").

A main motivation of this work comes from the reported experience that combining features from different sources leads to better predictive power. For instance, insurance and banking data together can improve fraud detection; shopping records well complement medical history for estimating risk of disease [Tsui *et al.*, 2003]; joining heterogeneous data helps prediction in genomics [Lanckriet *et al.,* 2004; Yamanishi *et al.,* 2004]; security agencies integrate various sources for terrorism intelligence [Sweeney, 2005; Christen, 2006; Sproull *et al.,* 2015].

A typical data fusion framework relies however on a known map between entities [Bleiholder and Naumann, 2008], *i.e.* peers have partially different views of the *same* examples. Instead, we assume the datasets do not share a common ID, as shown in Figure 1; that is, for example, the case when data collection of was performed independently by each peer, or when sources were deliberately anonimized. Thus, we can think the data as vertically partitioned (**VP**). *Entity resolution* (ER), or entity matching [Christen, 2012], would be the tra-

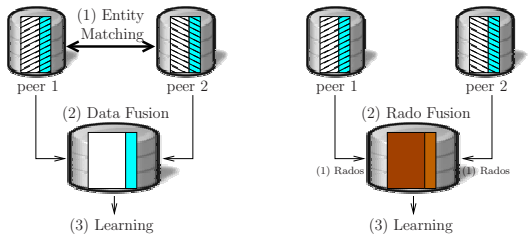

Figure 2: Learning on top of ER (left) or with rados (right).

ditional approach for reconciling entities with no shared ID[1]. It approximates a JOIN operation, assuming that some of the attributes are shared, *e.g.*, *age-band*, *gender*, *postcode* (etc.), and hence can be used as "weak IDs". Most techniques for ER are based on similarity functions and thresholding: candidate entities are selected as matches when their similarity is above a threshold. Both components can be tuned on some ground truth matches and effectively enhanced with learning techniques [Bilenko and Mooney, 2003][Christen, 2012]. The various metrics of ER encompass lots of different parameters, including generality, accuracy, soundness, scalability, parallelizability [Rastogi *et al.*, 2011]. The standard pipeline for learning with ER is depicted in Figure 2 (left): (1) entities are matched based on similarity and heuristics, (2) they are merged in one unique database and (3) a model is learnt on the joint data. Common issues in fusion, such as *conflicts* and *heterogeneity* [Bleiholder and Naumann, 2008], are not considered in this work.

From a high level view, ER integrates data as a pre-process for other tasks. When it comes to learning from ER'ed data, small changes in ER can have large impact on evaluating classifiers, even for simple classifiers as linear models. To see this, suppose we are in the toy example of Table 1. Here, all shared variables have the same values, so entity matching has two potential solutions (notice that one of the shared variables is class c). One, say ER1, is matching e_1 with e'_1 and e_2 with e'_2. We denote the examples obtained by $e_{11} \doteq ((1,-1,1),1)$ and $e_{22} \doteq ((-1,1,1),1)$ (an example is a pair (observation, class)). The other solution, say ER2, is matching e_1 with e'_2 and e_2 with e'_1. We denote the examples obtained by $e_{12} \doteq ((1,1,1),1)$ and $e_{21} \doteq ((-1,-1,1),1)$. Now, consider linear classifier $\boldsymbol{\theta} = (1,1,1) \in \mathbb{R}^3$; the class it gives is the sign of its inner product with an observation, $\boldsymbol{\theta}(\boldsymbol{z}) \doteq \text{sign}(\boldsymbol{\theta}^\top \boldsymbol{z})$. While $\boldsymbol{\theta}$ classifies perfectly on $\{e_{11}, e_{22}\}$ (zero error), it classifies no better than random on $\{e_{12}, e_{21}\}$ (error 50%).

To cope with (**VP**) data, we use a recent trick for learning from private data [Nock *et al.*, 2015]: examples are not necessary to learn an accurate linear classifier. We stress the fact that "accurate" refers to the quality of the class prediction from observations. The input of the algorithm consists of *Rademacher observations*, rados. A rado is just a sum, over a subset of examples, of the observations times their class. Surprisingly, we can learn with data in this form and, moreover,

[1]This is clearly non trivial: if just two rows in each dataset have the same exact values for the shared features across the p peers, this yields 2^p possible matchings for the two examples involved.

	Peer 1				Peer 2		
		shared				shared	
	x_1	x_3	c		x_2	x_3	c
e_1	1	1	1	e'_1	-1	1	1
e_2	-1	1	1	e'_2	1	1	1

Table 1: A simple case of (**VP**), with $p = 2$ peers, with two shared variables x_3 and c (the class to predict). This toy example has binary description features and a binary shared feature, but this restriction does not need to hold in the general case. For example, each shared feature can be any categorical/ordinal feature, like "postcode", "age-bracket", etc.

the output classifier does not need any post-processing since it is the same as if we were learning with examples.

Contributions — Our contribution starts from noticing that many rados are invariant to the selection of different solutions for entity resolution. For example, consider again Table 1. Since all classes are positive, computing a rado is just summing observations. Let $\boldsymbol{\pi}_{ij,kl}$ be the rado that sums those of examples e_{ij} and e_{kl}. Then, surprisingly, regardless of the solution to ER, this rado is the *same*:

$$\begin{array}{rrcl} \text{(E1)} & \boldsymbol{\pi}_{11,22} & = & (1,-1,1)+(-1,1,1) \\ & & = & (0,0,2) \\ & & = & (1,1,1)+(-1,-1,1) = \boldsymbol{\pi}_{12,21} \ \text{(E2)} \ . \end{array}$$

This, as we show, always holds in the (**VP**) setting: there exists a huge, *i.e.*, of potential exponential size, set of rados that match the set that could be built *knowing* the true entity resolution. We give the algorithm that builds these rados. It is easily parallelizable and requires *sublinear* communication, *i.e.* the amount of information that transits is no larger — and may be much smaller — than the size of all peers' data.

These "ideal" rados are not just interesting *per se*: learning from them (Figure 2, right) is both efficient and accurate. We show that using them leads to approximating the classifier that would be optimal *on the set of all (ideally ER'ed) examples*. This involves three technical contributions:

- The first is an elementary proof that the minimisation of the Ridge regularized square loss [Hoerl and Kennard, 1970] *on examples* is equivalent to the minimisation of a regularized loss *on rados*, which we call the M-loss.
- We then give the closed-form solution for the classifier minimizing the M-loss. Surprisingly, it shows that the minimisation of the regularized M-loss, over the complete (possibly exponential-size) set of "ideal" rados can be done not just in polynomial time: it is in general *faster* than the minimization of the Ridge regularized square loss over examples.
- Finally, the optimal M-loss classifier learnt using only the set of "ideal" rados converges — as the number of shared features increases — to the minimizer of the Ridge regularized square loss over *all* ideally ER'ed examples. In other words, as the number of shared features (or their modalities) increases, we are *guaranteed* to converge to the best classifier learned over examples.

Last, but not least, while we focus on the two-classes setting, description features need not be boolean. There is in fact no restriction apart from the fact that shared features are treated as ordinal instead of plain real: if one feature had as many modalities as there are examples, then there would be no need to address ER. The rest of this paper is as follows. Section §2 provides preliminaries. § 3 follows that shows how to learn from distributed data to minimise, indirectly, the Ridge regularized square loss over the ER'ed complete data. § 4 introduces a more realistic learning setting, then used in the experimental analysis of § 5. Finally, § 6 discusses our approach and § 7 concludes the paper.

2 Preliminaries

Learning setting We let $[n] \doteq \{1, 2, \ldots, n\}$ for $n \in \mathbb{N}_*$; boldfaces like $\boldsymbol{x}$ indicate vectors, whose coordinates are denoted as x_i. Notation $1\{.\}$ is the indicator function. We briefly recall the task of binary classification with linear models $\boldsymbol{\theta}$ as learning a predictor for label (or class) $y \in \{\pm 1\}$, from a *total* (learning, training) sample $\mathcal{S} \doteq \{(\boldsymbol{x}_i, y_i), i \in [m]\}$. Each example is an observation-label pair $(\boldsymbol{x}_i, y_i) \in \mathcal{X} \times \{\pm 1\}$, with $\mathcal{X} \subseteq \mathbb{R}^d$ the *feature space*, and it is drawn i.i.d. from an unknown distribution. We also denote $\mathcal{X} \doteq \times_{k=1}^{d} \mathcal{X}_k$, the cartesian product of its component spaces. We reserve the word *entity* for a generic record in a dataset, the object of matching, and *attributes* or *features* to its fields.

Our setting departs from the standard setting in what follows. Instead of one total training sample, we have p (sub)samples, $\mathcal{S}^j$ of size m_j, $j \in [p]$ for some $p > 1$. Each one is defined in its own feature space $\mathcal{X}^j \doteq \times_{k=1}^{d_j} \mathcal{X}_{j_k}$, where $j_k \in [d], \forall k$. To get a simple case of this framework, shown in Figure 1, one may see each $\mathcal{S}^j \doteq \{(\boldsymbol{x}_i^j, y_i^j), i \in [m_j]\}$ handled by a *peer* P^j. Throughout the paper, subscripts i will refer to an example or entity, while superscripts j to a peer. We rely on the following assumption:

(**A**) The class and a subset of features $\mathcal{J} \subseteq \{\mathcal{X}_k\}_{k=1}^{d}$ are shared by all peers. Each other feature is exclusive to one peer.

Hence, there exists $\dim(\mathcal{J}) + 1$ columns that represent the same set of attributes among peers, and one of them is the class. Each of the dimensions of $\mathcal{J}$ is in all $\mathcal{X}^j$s. This is a realistic assumption for the features in $\mathcal{J}$: in the (**VP**) setting, which is a gold standard of database frameworks, the domain is vertically partitioned for the non-shared features, implying $m_j = m, \forall j \in [p]$. In this case, there exists a (unknown) one-to-one mapping between the peers' rows. The shared label might be harder to justify, since it is the attribute we aim to predict. However, as argued in Section 6, if at least one peer has classes than all peers can get their labels as well *without entity resolution*, by the use of algorithms that learn with label proportions [Patrini *et al.*, 2014; Quadrianto *et al.*, 2009].

Rademacher observations In the standard classification model, a Rademacher observation (rado) is a simple linear transformation of the examples in sample $\mathcal{S}$ [Nock, 2015; Nock *et al.*, 2015]. Now, let $\boldsymbol{\sigma} \in \Sigma_m \doteq \{-1, 1\}^m$. Then a rado is $\boldsymbol{\pi}_{\boldsymbol{\sigma}} \doteq \sum_{i=1}^{m} 1\{y_i = \sigma_i\}\, y_i \cdot \boldsymbol{x}_i$, where $y_i \cdot \boldsymbol{x}_i$ is termed an *edge* vector. One of the 2^m rados, $\boldsymbol{\pi}_{\boldsymbol{y}} = \sum_{i=1}^{m} y_i \cdot \boldsymbol{x}_i$, ($\sigma = \boldsymbol{y}$), is a *sufficient statistic w.r.t.* class $\boldsymbol{y}$ for a wide set of losses; see the *mean operator* [Patrini *et al.*, 2014], [Patrini *et al.*, 2016a]. In our distributed setting, we extend the definition as follows. We let $\boldsymbol{s} \in \mathcal{J}$ denote a *signature*, *i.e.* a vector of shared attributes, $y \in \{\pm 1\}$ and let j index peer P^j. A rado is then:

$$\boldsymbol{\pi}_{(\boldsymbol{s},y)}^{j} \doteq \sum_{i=1}^{m} 1\{\mathrm{proj}_{\mathcal{J}}(\boldsymbol{x}_i^j) = \boldsymbol{s} \wedge y_i^j = y\}\, y_i^j \cdot \boldsymbol{x}_i^j \ , \qquad (1)$$

where $\mathrm{proj}_{\mathcal{J}}(\boldsymbol{z})$ denotes the restriction of a vector $\boldsymbol{z}$ to $\mathcal{J}$. In short, $\boldsymbol{\pi}_{(\boldsymbol{s},y)}^{j}$ sums edge vectors local to P^j whose examples match signature $\boldsymbol{s}$ and class y. Intuitively, we can conceptualize those rados and expressing statistics *locally* sufficient for the examples sharing the same signature $\boldsymbol{s}$ in the data of P^j. Let $\mathcal{F}(\boldsymbol{z}) \subseteq \mathcal{X}$ be the set of features of $\boldsymbol{z}$. We also define, for any $\mathcal{F}' \supseteq \mathcal{F}(\boldsymbol{z})$, $\mathrm{lift}_{\mathcal{F}'}(\boldsymbol{z})$ to be the vector $\boldsymbol{z}'$ described using $\mathcal{F}'$ such that $\mathrm{proj}_{\mathcal{F}(\boldsymbol{z})}(\boldsymbol{z}') = \boldsymbol{z}$ and $\mathrm{proj}_{\mathcal{F}' \setminus \mathcal{F}(\boldsymbol{z})}(\boldsymbol{z}') = \boldsymbol{0}$. While $\mathrm{proj}_{\mathcal{F}}(\boldsymbol{z})$ removes coordinates of $\boldsymbol{z}$, $\mathrm{lift}_{\mathcal{F}'}(\boldsymbol{z})$ "completes" the coordinates of $\boldsymbol{z}$ with zeroes.

By analogy with entity resolution [Whang *et al.*, 2009], we define *block rados* as rados, lifted to $\mathcal{X}$, that are sums of edges matching a particular signature and class in all peers.

Definition 1 *For any $\boldsymbol{s} \in \mathcal{J}$, $y \in \{-1, 1\}$, let $m_{(\boldsymbol{s},y)}$ be the number of examples matching signature $(\boldsymbol{s}, y)$. Then a **basic block** (BB) rado for $(\boldsymbol{s}, y)$ is*

$$\boldsymbol{\pi}_{(\boldsymbol{s},y)} \doteq \sum_{j=1}^{p} \mathrm{lift}_{\mathcal{X}}(\boldsymbol{\pi}_{(\boldsymbol{s},y)}^{j}) - m_{(\boldsymbol{s},y)}(p-1) \cdot \mathrm{lift}_{\mathcal{X}}(y \cdot \boldsymbol{s}) \ .$$

We need to subtract the second term to take into account that $\boldsymbol{s}$ has already been summed up $m_{(\boldsymbol{s},y)}$ times by each peer. Let $\mathcal{J}_* \doteq \{(\boldsymbol{s}, y) \in \mathcal{J} \times \{-1, 1\} : \exists j \in [p], \boldsymbol{\pi}_{(\boldsymbol{s},y)}^{j} \neq \boldsymbol{0}\}$. This latter set, which can easily be computed from all peers, has cardinal $m_* \doteq |\mathcal{J}_*| \leq m$, and even $m_* \ll m$ when few features are shared. We let $\mathcal{R}_{\mathrm{B}} \doteq \{\boldsymbol{\pi}_{\boldsymbol{v}_i}, \forall i \in [m_*]\}$ denote the ordered set of each BB rado, each coordinate of $\boldsymbol{v} = (\boldsymbol{s}, y)$ being in one-one correspondence with an element of $\mathcal{J}_*$. A superset of $\mathcal{R}_{\mathrm{B}}$ is interesting, that considers all sums of vectors from $\mathcal{R}_{\mathrm{B}}$:

$$\mathcal{R}_* \ \doteq \ \left\{ \sum_{i \in \mathcal{U}} \boldsymbol{\pi}_{\boldsymbol{v}_i}, \forall \mathcal{U} \subseteq [m_*] \right\} \ . \qquad (2)$$

We call $\mathcal{R}_*$ the set of **block rados**. Notice that we may have $|\mathcal{R}_*| = \Omega(2^{\sum_j |\mathcal{S}^j|})$. It is therefore intractable in general to *explicitly* compute $\mathcal{R}_*$. However, $|\mathcal{R}_{\mathrm{B}}| = O(\sum_j |\mathcal{S}^j|)$ and to compute it, we just need the set of $\boldsymbol{\pi}_{(\boldsymbol{s},y)}^{j}$, hence a communication complexity that can be much smaller than $\sum_j |\mathcal{S}^j|$.

3 Building and learning from BB rados

Why and how can we use rados to learn accurate classifiers? This first subsection does not concern the distributed setting. Instead, it summarizes and comments findings from [Nock *et al.*, 2015], [Nock, 2015].

Example vs rado losses Learning $\boldsymbol{\theta}$ on $\mathcal{S}$ is done by minimizing a loss function. Here, we consider the Ridge regularized square loss [Hoerl and Kennard, 1970] (Γ is sym. positive definite, SPD),

$$\ell_{\rm sql}(\mathcal{S}, \boldsymbol{\theta}; \Gamma) \doteq \frac{1}{m} \cdot \sum_i (1 - y_i \boldsymbol{\theta}^\top \boldsymbol{x}_i)^2 + \boldsymbol{\theta}^\top \Gamma \boldsymbol{\theta} \ . \quad (3)$$

It is crucial to remark that this loss is described over the total sample $\mathcal{S}$ of examples (see the red rectangle in Figure 1). This *is* the loss we want to minimize, exactly or approximately. One reason we choose this loss is that in the standard classification framework, it admits a simple closed form solution:

$$\boldsymbol{\theta}^\star_{\rm ex} \doteq \arg\min_{\boldsymbol{\theta}} \ell_{\rm sql}(\mathcal{S}, \boldsymbol{\theta}; \Gamma) = \left(\mathsf{X}\mathsf{X}^\top + m \cdot \Gamma\right)^{-1} \boldsymbol{\pi}_{\boldsymbol{y}} \ , \quad (4)$$

where $\mathsf{X} \doteq [\boldsymbol{x}_1 | \boldsymbol{x}_2 | \cdots | \boldsymbol{x}_m]$, and so, $\mathsf{X}\mathsf{X}^\top = \sum_i \boldsymbol{x}_i \boldsymbol{x}_i^\top$. Remark that $\boldsymbol{\theta}^\star_{\rm ex}$ involves $\boldsymbol{\pi}_{\boldsymbol{y}}$, one particular rado[2]. For any $\Sigma'_m \subseteq \Sigma_m \doteq \{-1, 1\}^m$, we let $\mathcal{R}_{\mathcal{S}, \Sigma'_m} \doteq \{\boldsymbol{\pi}_{\boldsymbol{\sigma}} : \boldsymbol{\pi}_{\boldsymbol{\sigma}} \in \Sigma'_m\}$ denote the set of rados that can be crafted from Σ'_m using $\mathcal{S}$.

Definition 2 *The* M*-loss over* $\mathcal{R}_{\mathcal{S}, \Sigma'_m}$ *of classifier* $\boldsymbol{\theta}$ *is:*

$$\ell_{\rm M}(\mathcal{R}_{\mathcal{S}, \Sigma'_m}, \boldsymbol{\theta}) \doteq - \left(\mathbb{E}_{\Sigma'_m}[\boldsymbol{\theta}^\top \boldsymbol{\pi}_{\boldsymbol{\sigma}}] - \frac{1}{2} \cdot \mathbb{V}_{\Sigma'_m}[\boldsymbol{\theta}^\top \boldsymbol{\pi}_{\boldsymbol{\sigma}}] \right) , \quad (5)$$

where expectation and variance are computed with respect to the uniform sampling of $\boldsymbol{\sigma}$ *in* Σ'_m*.*

Eq. (5) resembles a Markowitz mean-variance criterion [Markowitz, 1952] —with no coefficient for the risk aversion. What this means is that a good classifier trained on rados should have large "return" and small "risk", where the risk is the variance of its predictions and the return is its inner product with the expected rado.

The next Theorem shows that what was known for the logistic loss in [Nock *et al.*, 2015] also holds for the square loss: $\ell_{\rm sql}(\boldsymbol{\theta})$ (other dependences omitted) is equal to a strictly increasing function of $\ell_{\rm M}(\boldsymbol{\theta})$, described over rados, *for any* $\boldsymbol{\theta}$. Hence, minimizing $\ell_{\rm sql}(\boldsymbol{\theta})$ over examples is *equivalent* to minimizing $\ell_{\rm M}(\boldsymbol{\theta})$ for the *same* classifier. The proof of the Theorem is interesting in itself as it simplifies the long derivation of the more general equivalence in [Nock, 2015].

Theorem 3 *Let* $\Sigma_m \doteq \{-1, 1\}^m$*. Then, for any* $\mathcal{S}$*, any* Γ *and* ***any*** $\boldsymbol{\theta}$*,* $\ell_{\rm sql}(\mathcal{S}, \boldsymbol{\theta}; \Gamma) = 1 + (4/m) \cdot \ell_{\rm M}(\mathcal{R}_{\mathcal{S}, \Sigma_m}, \boldsymbol{\theta}; \Gamma)$ *with*

$$\ell_{\rm M}(\mathcal{R}_{\mathcal{S}, \Sigma_m}, \boldsymbol{\theta}; \Gamma) = \ell_{\rm M}(\mathcal{R}_{\mathcal{S}, \Sigma_m}, \boldsymbol{\theta}) + \frac{m}{4} \boldsymbol{\theta}^\top \Gamma \boldsymbol{\theta} \ . \quad (6)$$

Proof First, we remark that $\mathbb{E}_{\Sigma_m}[\boldsymbol{\theta}^\top \boldsymbol{\pi}_{\boldsymbol{\sigma}}] = \boldsymbol{\theta}^\top \mathbb{E}_{\Sigma_m}[\boldsymbol{\pi}_{\boldsymbol{\sigma}}] = (1/2) \cdot \boldsymbol{\theta}^\top \boldsymbol{\pi}_{\boldsymbol{y}}$, since each example participates to half of the 2^m rados. Letting $\tilde{v} \doteq 2^{m+2} \cdot \mathbb{V}_{\Sigma_m}[\boldsymbol{\theta}^\top \boldsymbol{\pi}_{\boldsymbol{\sigma}}]$, we also have

$$\begin{aligned}
\tilde{v} &= 4 \cdot \sum_{\boldsymbol{\sigma} \in \Sigma_m} \left(\boldsymbol{\theta}^\top \boldsymbol{\pi}_{\boldsymbol{\sigma}} - \frac{1}{2} \cdot \boldsymbol{\theta}^\top \boldsymbol{\pi}_{\boldsymbol{y}} \right)^2 \\
&= \sum_{\boldsymbol{\sigma} \in \Sigma_m} \left(\sum_i \sigma_i \boldsymbol{\theta}^\top \boldsymbol{x}_i \right)^2 \\
&= \sum_{\boldsymbol{\sigma} \in \Sigma_m} \left[\sum_{i=1}^m (\boldsymbol{\theta}^\top \boldsymbol{x}_i)^2 + \sum_{i=1}^e m \sum_{i' \neq i} \sigma_i \sigma_{i'} \boldsymbol{\theta}^\top \boldsymbol{x}_i \boldsymbol{\theta}^\top \boldsymbol{x}_{i'} \right] \\
&= 2^m \cdot \sum_{i=1}^m (\boldsymbol{\theta}^\top \boldsymbol{x}_i)^2 + \sum_{i=1}^m \sum_{i' \neq i} v_{ii'} \cdot \boldsymbol{\theta}^\top \boldsymbol{x}_i \boldsymbol{\theta}^\top \boldsymbol{x}_{i'} \ , \quad (7)
\end{aligned}$$

with $v_{ii'} \doteq \sum_{\boldsymbol{\sigma} \in \Sigma_m} \sigma_i \sigma_{i'}$. Now, for any $i \neq i'$, $\sigma_i \sigma_{i'}$ takes exactly the same number of times value $+1$ and value -1, and so $v_{ii'} = 0, \forall i \neq i'$. We get from eq. (7) $\mathbb{V}_{\Sigma_m}[\boldsymbol{\theta}^\top \boldsymbol{\pi}_{\boldsymbol{\sigma}}] = (1/4) \cdot \sum_{i=1}^m (\boldsymbol{\theta}^\top \boldsymbol{x}_i)^2 = (1/4) \cdot \sum_{i=1}^m (y_i \boldsymbol{\theta}^\top \boldsymbol{x}_i)^2$. Finally,

$$\begin{aligned}
&1 + \frac{4}{m} \cdot \ell_{\rm M}(\mathcal{S}, \Sigma_m, \boldsymbol{\theta}) \\
&= 1 - \frac{2}{m} \cdot \sum_{i=1}^m y_i \boldsymbol{\theta}^\top \boldsymbol{x}_i + \frac{1}{m} \cdot \sum_{i=1}^m (y_i \boldsymbol{\theta}^\top \boldsymbol{x}_i)^2 \\
&= \frac{1}{m} \cdot \sum_i (1 - y_i \boldsymbol{\theta}^\top \boldsymbol{x}_i)_2^2 \ , \quad (8)
\end{aligned}$$

and we get Theorem 3 by integrating Ridge regularization. ■

Hence, minimizing the Ridge regularized square loss over examples is equivalent to minimizing a regularized version of the M-loss, over the complete set of all rados. This set has exponential size. A possibility is to randomly subsample the set, along with proving good uniform convergence bounds for the M-loss — this can be done in the same way as for the logistic loss [Nock *et al.*, 2015]. However, in the case of the square loss, greed pays twice: learning from all rados in $\mathcal{R}_*$ may be both cheap (computationally) and accurate.

Computation and optimality of $\mathcal{R}_*$ In our distributed context, we do not have access to all rados because we do not assume to known the entity matching function. Yet, we are going to show a first result which is, in a sense, *stronger*: $\mathcal{R}_*$ always belongs to $\mathcal{R}_{\mathcal{S}, \Sigma_m}$. Therefore $\mathcal{R}_*$ —potentially exponential-size— gives us a set of rados that would have been built *from* $\mathcal{S}$*, had we known the perfect solution to entity matching*. So, even without carrying out entity matching, we have access to a potentially huge set of "ideal" rados which we can use to learn $\boldsymbol{\theta}$ via the minimization of $\ell_{\rm M}(., \boldsymbol{\theta}; \Gamma)$. Furthermore, there exists a simple algorithm to build $\mathcal{R}_{\rm B}$.

Algorithm 1 summarizes the protocol. Each peer P^j crafts rados upon request of a particular signature and label; "CRAFT$(\boldsymbol{s}, y) \rightsquigarrow$" symbolizes a message sent, expecting $\boldsymbol{\pi}^j_{(\boldsymbol{s}, y)}$ in return. Remark that the computation of each rado for each peer can easily be performed in parallel. We now show one of the main results of this paper: Algorithm 1 always provides the basis for the set $\mathcal{R}_*$ of the "ideal" rados.

Theorem 4 *In setting (**VP**), for any* $p \geq 2$*,* ***any*** $\mathcal{S}$*,* ***any*** $\mathcal{J}$*. Let* $\mathcal{R}_{\rm B}$ *be the output of Algorithm 1 and let* $\mathcal{R}_*$ *its superset by eq. (2). Then,* $\mathcal{R}_* \subseteq \mathcal{R}_{\mathcal{S}, \Sigma_m}$.

[2]Notice here its sufficiency *w.r.t.* $\boldsymbol{y}$, as labels do not appear anywhere else in the formula.

Algorithm 1 RADOCRAFT($\mathsf{P}^1, \mathsf{P}^2, ..., \mathsf{P}^p$)

Input Peers $\mathsf{P}^1, \mathsf{P}^2, ..., \mathsf{P}^p$;
Step 1: Let $\mathcal{R}_{\mathrm{B}} \leftarrow \emptyset$;
Step 2: **for** $\boldsymbol{s} \in \mathcal{J}$, $y \in \{\pm 1\}$:
2.1: Let $\boldsymbol{\pi}_{(\boldsymbol{s},y)} \leftarrow \mathbf{0} \in \mathbb{R}^d$;
2.2: **for** $j \in [p]$:
2.2.1: $\boldsymbol{\pi}_{(\boldsymbol{s},y)} \leftarrow \boldsymbol{\pi}_{(\boldsymbol{s},y)} + \mathrm{lift}_{\mathcal{X}}(\textsc{craft}(\boldsymbol{s}, y) \rightsquigarrow \mathsf{P}^j)$
2.3: $\mathcal{R}_{\mathrm{B}} \leftarrow \mathcal{R}_{\mathrm{B}} \cup \{\boldsymbol{\pi}_{(\boldsymbol{s},y)}\}$;
Return $\mathcal{R}_{\mathrm{B}}$;

Proof (sketch) The Theorem follows once three simple facts are established in the (**VP**) setting: (a) the true entity matching exists, (b) any BB rado for pair $(\boldsymbol{s}, y)$ would be obtained as a rado summing the contributions of all examples in $\mathcal{S}$ matching the corresponding signature $\boldsymbol{s}$ and class y, (c) we obtain $\mathcal{R}_{\mathrm{B}} \subseteq \mathcal{R}_{\mathcal{S},\Sigma_m}$, from which follows the Theorem's statement with eq. (2) and the fact that any sum of a subset of rados in $\mathcal{R}_{\mathrm{B}}$ would also be in $\mathcal{R}_{\mathcal{S},\Sigma_m}$ since an example cannot match two distinct couples (signature, class). ■

Learning from *all* rados of $\mathcal{R}_*$ How do we minimize the regularized M-loss and, more importantly, which subset of rados from $\mathcal{R}_*$ shall we use? As already discussed, we choose "greediness" against randomization: instead of picking a (small) random subset of $\mathcal{R}_*$, we want to use them *all* because we know that all of them are "ideal" or close to being so via Theorems 4. Recall that $|\mathcal{R}_*|$ may be of exponential size (in m, d, $|\mathcal{J}_*|$, etc.). We now show that if we consider all of $\mathcal{R}_*$, the optimal $\boldsymbol{\theta}^\star_{\mathrm{rad}}$ of $\ell_{\mathrm{M}}(\mathcal{R}_*, \boldsymbol{\theta}; \Gamma)$ has an analytic expression which depends *only* on the rados of $\mathcal{R}_{\mathrm{B}}$. In short, it is even *faster* to compute than $\boldsymbol{\theta}^\star_{\mathrm{ex}}$ from $\mathcal{S}$ in eq. (4), and can be directly computed from the output of Algorithm 1.

Theorem 5 *Let* $\boldsymbol{\theta}^\star_{\mathrm{rad}} \doteq \arg\min_{\boldsymbol{\theta}} \ell_{\mathrm{M}}(\mathcal{R}_*, \boldsymbol{\theta}; \Gamma)$ *(eq. (6)). Then*

$$\boldsymbol{\theta}^\star_{\mathrm{rad}} = \left(\mathsf{B}\mathsf{B}^\top + \dim_{\mathrm{c}}(\mathsf{B}) \cdot \Gamma\right)^{-1} \mathsf{B}\mathbf{1} \ , \qquad (9)$$

where B *stacks in columns the rados of* $\mathcal{R}_{\mathrm{B}}$*, and* $\dim_{\mathrm{c}}(\mathsf{B})$ *is the number of columns of* B*.*

Proof The proof uses the following trick: consider any sample $\mathcal{S}'$ such that its edge vectors match the basic block rados. Remark that $\mathsf{X}\mathsf{X}^\top = \sum_i (y_i \boldsymbol{x}_i)(y_i \boldsymbol{x}_i)^\top$ in eq. (4) depends only on edge vectors, and so, since $\boldsymbol{\pi}_{\boldsymbol{y}} = \mathsf{B}\mathbf{1}$, the optimal square loss classifier on $\mathcal{S}'$ is $\boldsymbol{\theta}^\star_{\mathrm{rad}}$ in eq. (9), which, through Theorem 3, is also the optimal classifier on $\ell_{\mathrm{M}}(\mathcal{R}_*, \boldsymbol{\theta}; \Gamma)$. ■

When $m_* = m$, each element of $\mathcal{R}_{\mathrm{B}}$ is in fact an example, and we retrieve eq. (4). One consequence of Theorem 5 is the following convergence property which we sketch: in the (**VP**) setting, for any $\varepsilon \geq 0$, there exists a minimal size for $\mathcal{J}_*$ such that $\boldsymbol{\theta}^\star_{\mathrm{rad}}$ will be ε-close to $\boldsymbol{\theta}^\star_{\mathrm{ex}}$, where the closeness can be measured by $\|\boldsymbol{\theta}^\star_{\mathrm{rad}} - \boldsymbol{\theta}^\star_{\mathrm{ex}}\|_2$ or $|\cos(\boldsymbol{\theta}^\star_{\mathrm{rad}}, \boldsymbol{\theta}^\star_{\mathrm{ex}})|$. The statement of DRL (Distributed Rado-Learn) is given in Algorithm 2. In Step 1, "$\mathrm{column}(.)$" takes a set of vectors and put them in column in a matrix.

Algorithm 2 DRL($\mathsf{P}^1, \mathsf{P}^2, ..., \mathsf{P}^p; \Gamma$)

Input Peers $\mathsf{P}^1, \mathsf{P}^2, ..., \mathsf{P}^p$, SPD matrix Γ, $\gamma > 0$;
Step 1: $\mathsf{B} \leftarrow \mathrm{Column}(\textsc{RadoCraft}(\mathsf{P}^1, \mathsf{P}^2, ..., \mathsf{P}^p))$;
Step 2: $\boldsymbol{\theta} \leftarrow \left(\mathsf{B}\mathsf{B}^\top + \gamma \cdot \Gamma\right)^{-1} \mathsf{B}\mathbf{1}$;
Return $\boldsymbol{\theta}$;

4 A more realistic setting

What happens if we drop the assumption of data being vertically partitioned (**VP**)? Or equivalently, what if examples are not shared by *all* peers ? This is a much more realistic scenario. Since there is no shared ID — and the data may have been anonymized — we are not even in a situation where we can guarantee that a specific client of the bank *is*, or *is not*, a client of the insurance company. Thus, there may be significant unknown data "to reconstruct" the total sample $\mathcal{S}$, but we do not know which specific examples have missing features. In this most general setting (**G**), it is possible to show that a very simple transformation of the rados, involving only the shared features, has in expectation the same properties so far described and for which Theorem 4 holds *in expectation*. Due to lack of space, details are left to a longer version of the paper [Patrini *et al.*, 2016b]. However, in the next Section, we provide an experimental validation of our approach in both the settings.

5 Experiments

Algorithms We have evaluated the leverage that DRL provides compared to the peers, that would learn using only their local dataset. Each peer P^j estimates learns through a ten-folds stratified cross-validation (CV) minimization of $\ell_{\mathrm{sql}}(\mathcal{S}^j, \boldsymbol{\theta}; \gamma \cdot \mathrm{Id}_{d_j})$ (see eq. (4)), where γ is also locally optimized through a ten-folds CV in set $\mathcal{G} \doteq \{.01, 1.0, 100.0\}$. DRL minimizes $\ell_{\mathrm{M}}(\mathcal{R}_*, \boldsymbol{\theta}; \Gamma)$ (solution in eq. (9)) where $\mathcal{R}_{\mathrm{B}}$ is built using RADOCRAFT, with the set of all peers as input.

We have carried out a very simple optimisation of the regularisation matrix of DRL as a diagonal matrix which weights differently the shared features, $\Gamma \doteq \mathrm{Diag}(\mathrm{lift}_{\mathcal{X}}(\mathrm{proj}_{\mathcal{J}}(\mathbf{1}))) + \gamma \cdot \mathrm{Diag}(\mathrm{lift}_{\mathcal{X}}(\mathrm{proj}_{\mathcal{X}\setminus\mathcal{J}}(\mathbf{1})))$, for $\gamma \in \mathcal{G}$. γ is optimized by a 10-folds CV on $\mathcal{I}_*$. CV is performed on *rados* as follows: first, $\mathcal{R}_{\mathrm{B}}$ is split in 10 folds, $\mathcal{R}_{\mathrm{B},\ell}$, for $\ell = 1, 2, ..., 10$. Then, we repeat for $\ell = 1, 2, ..., 10$ (and then average) the following CV routine:

1. DRL is trained using $\mathcal{R}_{\mathrm{B}} \backslash \mathcal{R}_{\mathrm{B},\ell}$;
2. DRL's solution, $\boldsymbol{\theta}^\star_{\mathrm{rad}}$, is evaluated on "test rados" by computing $\ell_{\mathrm{M}}(\mathcal{R}_{\mathrm{B},\ell}, \boldsymbol{\theta}^\star_{\mathrm{rad}}; \Gamma)$.

The expression of Γ for rados exploits the idea that the estimations related to a shared feature can be much more accurate than for another, non-shared feature.

Domain generation We ran experiments on a dozen UCI domains. Only two are fully detailed here, due to reason of space: they are *ionosphere* $(m \times d) = (351 \times 33)$ and *musk* $= (6598 \times 166)$ — the others appears in [Patrini *et al.*, 2016b]. For each domain, we have varied (i) the number of peers p, (ii) the number of shared features $\dim(\mathcal{J})$, and (iii)

the number b of numeric modalities ("bins") each shared feature was reduced to (it controls the size of $\mathcal{I}_*$). The training sample is split among peers, each keeping record of $\mathcal{I}$ and its own features (non-shared features are evenly partitioned among peers). Finally, for some $p_s \in [0,1]$, each peer P^j selects a proportion p_s of its examples index and for each of them, another peer $\mathsf{P}^{j'}$, chosen at random, gets the example as well (on its own set of features $\mathcal{X}^{j'}$). This policy implements (**G**). When $p_s = 0$, this is setting (**VP**). We then run *all* algorithms for *each* value $p, \dim(\mathcal{J}), b, p_s$. As we shall see, b appears to have a relatively small influence compared to the other factors, so we mainly report results combining various values for $p, \dim(\mathcal{J})$ and p_s, for the range of values of $p, \dim(\mathcal{J})$ specified in the corresponding Tables (3, 4), and for $p_s \in \{0.0, 0.2\}$. We have chosen $b = 4$ for all domains, except when it is not possible (if for example all features are boolean), in which case we pick $b = 2$.

Metric We used two metrics. The first,

$$\Delta \doteq \hat{p}_{\text{err}}(\text{DRL}) - \min_j \hat{p}_{\text{err}}(\mathsf{P}^j) \ (\in [-1,1]) \ ,$$

is the test error for DRL minus that of the *optimal* peer *in hindsight* (since we consider the peer's test error). when $\Delta < 0$, DRL beats *all* peers. For example, Table 3 (left) provides the results obtained on UCI domain ionisphere. We see that for almost all combinations of p and $\dim(\mathcal{J})$, DRL beats all peers. We give for each domain the smallest test error obtained for a peer among all runs for each domain: this is an indication of the room of improvement for DRL, and it also shows that in general, at least some (and in fact most) peers were always very significantly better than random guessing, a safe-check that DRL is not just beating unbiased coins.

To evaluate the statistical significance, we compute

$$\mathrm{q} \doteq \text{proportion of peers } \textit{statistically} \text{ beaten by DRL} \ .$$

To compute the test, we use the powerful Benjamini-Hochberg procedure on top of paired t-tests with $q^* = p\text{-val} = 0.05$, [Benjamini and Hochberg, 1995]; $q = 0.8$ surface helps see when DRL *statistically beats all peers*. For example, Table 3 (right) displays that DRL does not always *statistically* beat all peers when $\Delta < 0$, yet it manages to stastically beat all of them in a wide range of $p, \dim(\mathcal{J})$ values. Table 4 display that DRL tends to systematically beat all peers when p is sufficiently large.

Results To summarize our results, all domains display that there exists regimes $(p, \dim(\mathcal{J}))$ for which DRL improves on all peers, in some cases significantly. Sometimes, the improvement is sparse, but sometimes it is quite spectacular and in fact (almost) systematic. Drilling down into the results displays two patterns that seem to be quite general: the first is when the so-called *Oracle*, *i.e.* the learner that learns from the complete training fold *before* it is split among peers — and therefore knows the solution to entity matching —, has almost optimal error, but local peers are in fact very far from this optimum. This indicates that many features, properly combined, are necessary to attain the best performances. In such cases, DRL can manage to have performances close to the Oracle, and yields to a gap in classification compared to peers which can properly be huge — sometimes, DRL's test error is smaller than that of the *best* peer by more than 20% —. The second pattern is that for many domains, there is a threshold value for p beyond which DRL progressively improves on all peers (statistically). These two patterns seems to advocate that DRL may represent a very significant leverage of peer's data for moderately to massively distributed learning problems, when entity resolution is not available.

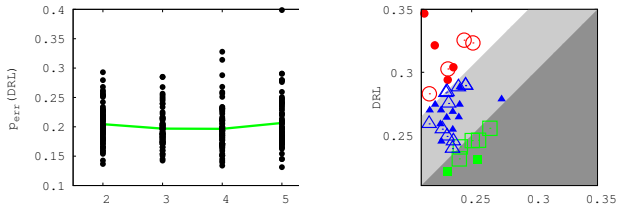

Table 2: *Left*: test error of DRL on domain ionosphere, as a function of the number of bins, aggregating all results varying p and $\dim(\mathcal{J})$; the green line denotes the average values. *Right*: scatterplot of the test error of DRL (y) *vs.* that of the Oracle (learning using the complete entity-resolved domain). Green in the dark grey area denote better performances of DRL; blue in the light grey area denote better performances of the Oracle (but not statistically better). Red in the white area denote *statistically* better performances of the Oracle (filled points: $p_s = 0.2$; empty points: $p_s = 0.8$).

To analyze further our results, Table 2 (left) displays that binning indeed does not affect significantly DRL on average, which is also good news, since it means that there is no restriction on the shared features for DRL to perform well: shared features can be binary, or categorical with any number of modalities. Additionally, Table 2 (right) compares the performances of DRL with respect to those of the Oracle on a domain for which DRL obtains somehow "median" performances among all domains, *sonar*, $(m \times d) = (208 \times 60)$. The Oracle (10-folds CV from the *total* ER'ed 8) is *idealistic* since in general we do not know the solution to ER, yet it gives clues on how close DRL may be from the "graal". Interestingly, DRL comes frequently under the statistical significance radar ($\alpha = 0.05$). In notable cases (more frequent as p_s increases), DRL beats Oracle — but not significantly. Aside from theory, these are good news as DRL does not assume ER'ed data, and uses an amount of data which can be $\sim p^2$ times *smaller* than Oracle.

6 Discussion and related work

We remark that our framework is not formally comparable with ER, since the two address different problems. On one hand, ER has a much broader applicability than the problem object of this paper; learning on distributed datasets is less general than ER: in fact, we show a solution that bypasses ER. On the other hand, *learning-based* ER [Bilenko and Mooney, 2003] as well as manifold alignment techniques [Lafon *et al.*, 2006] are viable only knowing some ground truth matches — which are not required for working with ra-

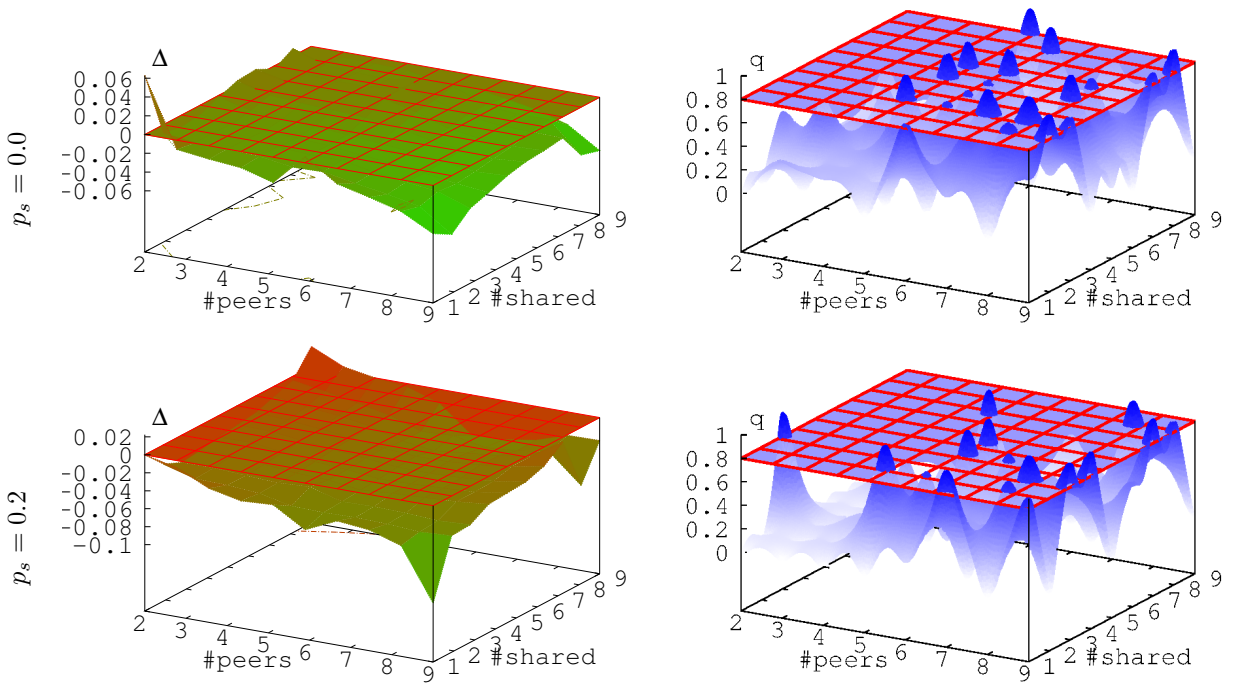

Table 3: Results on domain ionosphere: plots of $\Delta \doteq \hat{p}_{\text{err}}(\text{DRL}) - \min_j \hat{p}_{\text{err}}(\text{P}^j)$ (left) and $q =$ prop. peers *simultaneously* beaten by DRL (right) as a function of the number of peers p and the number of shared features $\dim(\mathcal{J})$. On ionosphere, $\min_j \hat{p}_{\text{err}}(\text{P}^j) = 0.20$. Top: proportion of shared examples $p_s = 0.0$ (**VP**); bottom: proportion of shared examples $p_s = 0.2$ (**G**). The isoline on the left plots is $\Delta = 0$.

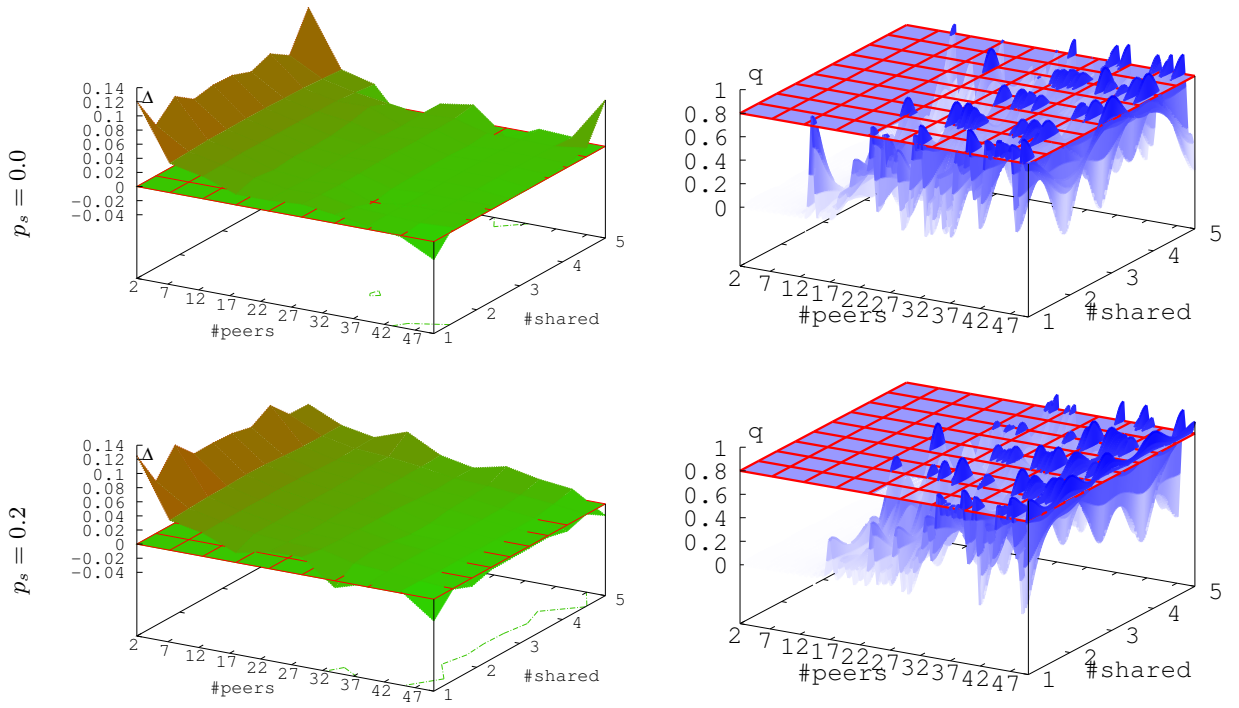

Table 4: Results on domain musk, using the same convention as Table 3. On musk, $\min_j \hat{p}_{\text{err}}(\text{P}^j) = 0.25$

Metric	ER + *Learning*	Algorithms 1+2
Hp: shared IDs	no	no
Hp: shared variables	necessary	necessary
Hp: shared labels	no	may be relaxed
Fusion / RADOCRAFT	$O(m^2/m^\star\ T_{sim})$	$O(m)$
Communication	$m \times d$	$m^\star \times d$
Learning problem	$m \times d$	$m^\star \times d$
Privacy	complex	many guarantees

Table 5: Multiple metrics of comparison between learning on top of ER and our approach. Time complexity are estimated for 2 peers in the (**VP**) scenario, assuming all blocks of equal size. "Hp" is short for hypothesis. See Section 6 for details.

dos. From another perspective, in concert with the *open issues* in [Getoor and Machanavajjhala, 2012], we study ER as component of a pipeline for classification, and highlight how matching is not necessary for the purpose of learning.

In spite of those considerations, we can still draw comparisons with methods that learn on top of data merged through ER (Table 5). In both settings, no ID is shared between datasets but some attributes must be so, in order to allow entities comparison for matching or for building rados. Obviously, entity matching does not require the labels to be one of those shared attributes, while this is a fundamental hypothesis of our approach. Although, it is not as restrictive as it might seem at first: if just one peer has labels, then *all* can obtain labels on their own data, via *learning from label proportions* [Quadrianto *et al.*, 2009], [Patrini *et al.*, 2014]: the label handling peer computes the label proportions per each block; the "bags" are defined by examples matching a particular signature. Proportions are then shared among all other peers, which can train a classifier with them so as to estimate a label for each observation.

To discuss time complexity, let us consider a simplified problem with only 2 peers in the (**VP**) scenario. In terms of complexity of fusion, if we assume that examples are uniformly distributed in the blocks, each block has size $m/m^\star$. DRL builds each block rado in time $O(m/m^\star)$, with total cost linear in m. ER takes $O((m/m^\star)^2 \cdot T_{sim})$ to match entities *in each* of the $m^\star$ blocks, where T_{sim} is the cost of evaluation any similarity function; learning-based methods spend additional time for training; advanced blocking strategies can reduce the average complexity [Bilenko *et al.*, 2006],[Whang *et al.*, 2009], [Whang and Garcia-Molina, 2012].

Most literature on distributed learning is concerned with limiting communication and designing optimal strategies for merging models [Balcan *et al.*, 2012], [Liu and Ihler, 2014]; beside that, previous works focus on horizontal split by observations, with few exceptions [Liu and Ihler, 2012]. In contrast, we exploit what is sufficient to merge *about the data*. The communication protocol is extremely simple. Once rados are crafted locally, they are sent to a central learner in one shot. By Theorem 5, only d-dimensional $m^\star$ basic block rados are needed. *Data is not accessed anymore* and learning takes place centrally. Moreover, rados help with data compression, being $m^\star \times d$, $m^\star \ll m$ the problem size. With ER we learn from all entities, with a total size of $m \times d$.

Learning on data described by different feature sets is the topic of multiple view learning and co-training [Blum and Mitchell, 1998], [Sindhwani *et al.*, 2005]. To the best of our knowledge, co-training with unknown matches has not been addressed before. [Brefeld *et al.*, 2006] presents a multi-view distributed algorithm with co-regularization; although it requires matches for all unlabelled examples.

In settings with multiple data providers, privacy can be crucial. The peers have to trade off model enhancements and information leaks. A learner receives rados to train the model; this can be done by one of the peers, or by a third party — paralleling multi-party ER scenarios [Christen, 2006]. The only information sent through the channel consists of rados, while examples, with their individual sensitive features, are never shared. Hardness results on reconstruct-ability of examples have been proven, along with NP-HARD characterizations, and protection in the sense of differential privacy [Nock *et al.*, 2015]. Regarding ER, since matching has the potential of de-anonimizing the entities, privacy is usually a very relevant issue to address [Christen, 2006]. However, solutions are not straightforward, as proven by the vast amount of research on the topic [Vatsalan *et al.*, 2013].

Even assuming labelled examples, no (observation, label) pair is actually available for training, and thus the task can be seen as weakly supervised [Garcıa-Garcıa and Williamson, 2011], [Patrini *et al.*, 2016a]. Although, a set of aggregate quantities turns out to be enough for the task. Theorem 3 expresses a form of *sufficiency* of the whole set of rados with regard to the square loss; the analogue property is known for logistic loss in [Nock *et al.*, 2015]. One particular rado, $(1/m) \cdot \boldsymbol{\pi}$, *mean operator*, is formally proven a *sufficient statistics* for the class for a wide set of losses [Patrini *et al.*, 2014], [Patrini *et al.*, 2016a]. This work, along with predecessors, shows how the interplay between aggregate statistics and losses can lead to effective solutions to hard learning problems.

7 Conclusion

Entity matching addresses a very general *but* difficult problem, and in the comparatively restricted context of supervised learning from distributed datasets, it is possible to evade the pitfalls of entity matching with Rademacher observations. Rados have other advantages: they provide a cheap, easily parallelizable material which somehow "compresses" examples while allowing accurate learning. Moreover, they also offer readily available solution for guarantees private exchange of data in a distributed setting.

Acknowledgments

The authors are grateful for the contribution of Hugh Durrant-Whyte in many stimulating discussions. 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

[Balcan *et al.*, 2012] M. F. Balcan, A. Blum, S. Fine, and Y. Mansour. Distributed learning, communication complexity and privacy. *arXiv preprint arXiv:1204.3514*, 2012.

[Benjamini and Hochberg, 1995] Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *J. of the Royal Stat. Society. Series B*, 57(1):289–300, 1995.

[Bilenko and Mooney, 2003] M. Bilenko and R. J. Mooney. Adaptive duplicate detection using learnable string similarity measures. In *Proc. of the 9^{th} ACM KDD*, pages 39–48, 2003.

[Bilenko *et al.*, 2006] M. Bilenko, B. Kamath, and R. J. Mooney. Adaptive blocking: Learning to scale up record linkage. In *6^{th} ICDM*, pages 87–96. IEEE, 2006.

[Bleiholder and Naumann, 2008] J. Bleiholder and F. Naumann. Data fusion. *ACM Computing Surveys (CSUR)*, 41(1):1, 2008.

[Blum and Mitchell, 1998] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *9^{th} COLT*, pages 92–100, 1998.

[Brefeld *et al.*, 2006] U. Brefeld, T. Gärtner, T. Scheffer, and S. Wrobel. Efficient co-regularised least squares regression. In *23^{th} ICML*, pages 137–144, 2006.

[Christen, 2006] P Christen. Privacy-preserving data linkage and geocoding: Current approaches and research directions. In *ICDMW06*, pages 497–501. IEEE, 2006.

[Christen, 2012] P. Christen. *Data Matching Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection*. Springer Data-Centric Systems and Applications, 2012.

[Estrada *et al.*, 2010] T. Estrada, R. Armen, and M. Taufer. Automatic selection of near-native protein-ligand conformations using a hierarchical clustering and volunteer computing. In *ACM BCB*, pages 204–213, 2010.

[Garcıa-Garcıa and Williamson, 2011] D. Garcıa-Garcıa and R. C. Williamson. Degrees of supervision. In *NIPS*24 Workshops*, 2011.

[Getoor and Machanavajjhala, 2012] L. Getoor and A. Machanavajjhala. Entity resolution: theory, practice & open challenges. *Proceedings of the VLDB Endowment*, 5(12):2018–2019, 2012.

[Hoerl and Kennard, 1970] A.-E. Hoerl and R.-W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[Lafon *et al.*, 2006] S. Lafon, Y. Keller, and R. R. Coifman. Data fusion and multicue data matching by diffusion maps. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(11):1784–1797, 2006.

[Lanckriet *et al.*, 2004] G. R. G. Lanckriet, T. De Bie, N. Cristianini, M. I. Jordan, and W. S. Noble. A statistical framework for genomic data fusion. *Bioinformatics*, 20(16):2626–2635, 2004.

[Liu and Ihler, 2012] Q. Liu and A.T. Ihler. Distributed parameter estimation via pseudo-likelihood. In *29^{th} ICML*, pages 1487–1494, 2012.

[Liu and Ihler, 2014] Q. Liu and A. T. Ihler. Distributed estimation, information loss and exponential families. In *NIPS*27*, pages 1098–1106, 2014.

[Markowitz, 1952] H. Markowitz. Portfolio selection. *J. of Finance*, 6:77–91, 1952.

[Nock *et al.*, 2015] R. Nock, G. Patrini, and A. Friedman. Rademacher observations, private data, and boosting. *32^{th} ICML*, pages 948–956, 2015.

[Nock, 2015] R. Nock. Learning games and Rademacher observations losses. *CoRR*, abs/1512.05244, 2015.

[Patrini *et al.*, 2014] G. Patrini, R. Nock, P. Rivera, and T. Caetano. (Almost) no label no cry. In *NIPS*27*, pages 190–198, 2014.

[Patrini *et al.*, 2016a] G. Patrini, F. Nielsen, R. Nock, and M. Carioni. Loss factorization, weakly supervised learning and label noise robustness. *CoRR*, abs/1602.02450, 2016.

[Patrini *et al.*, 2016b] G. Patrini, R. Nock, S. Hardy, and T. Caetano. Fast learning from distributed datasets without entity matching. *CoRR*, abs/1603.04002, 2016.

[Quadrianto *et al.*, 2009] N. Quadrianto, A. Smola, T. Caetano, and Q. Le. Estimating labels from label proportions. *JMLR*, 10:2349–2374, 2009.

[Rastogi *et al.*, 2011] V. Rastogi, N.-N. Dalvi, and M.-N. Garofalakis. Large-scale collective entity matching. *Proc. VLDB Endowment*, 4(4):208–218, 2011.

[Sindhwani *et al.*, 2005] V. Sindhwani, P. Niyogi, and M. Belkin. A co-regularized approach to semi-supervised learning with multiple views. In *Proceedings of the ICML Workshop on Learning with Multiple Views*, 2005.

[Sproull *et al.*, 2015] R. F. Sproull, W. H. DuMouchel, M. Kearns, B. W. Lampson, S. Landau, M. E. Leiter, E. R Parker, and P. J. Weinberger. Bulk collection of signal intelligence: technical options. In *Committee on Responding to Section 5(d) of Presidential Policy Directive 28: The Feasibility of Software to Provide Alternatives to Bulk Signals Intelligence Collection*. National Academy Press, 2015.

[Sweeney, 2005] L. Sweeney. Privacy-enhanced linking. *ACM SIGKDD Explorations Newsletter*, 7(2):72–75, 2005.

[Tsui *et al.*, 2003] F.C. Tsui, J. U. Espino, V. M. Dato, P. H. Gesteland, J. Hutman, and M. M. Wagner. Technical description of rods: a real-time public health surveillance system. *Journal of the American Medical Informatics Association*, 10(5):399–408, 2003.

[Vatsalan *et al.*, 2013] D. Vatsalan, P. Christen, and V. S. Verykios. A taxonomy of privacy-preserving record linkage techniques. *Information Systems*, 38(6):946–969, 2013.

[Whang and Garcia-Molina, 2012] S. E. Whang and H. Garcia-Molina. Joint entity resolution. In *ICDE, 2012 IEEE 28th International Conference on Data Engineering*, pages 294–305. IEEE, 2012.

[Whang *et al.*, 2009] S.-E. Whang, D. Menestrina, G. Koutrika, M. Theobald, and H. Garcia-Molina. Entity resolution with iterative blocking. In *Proc. ACM SIGMOD*, pages 219–232, 2009.

[Yamanishi *et al.*, 2004] Y. Yamanishi, J.-P. Vert, and K. Kanehisa. Protein network inference from multiple genomic data: a supervised approach. *Bioinformatics*, 20(suppl 1):i363–i370, 2004.

[Zhang *et al.*, 2015] B. Zhang, T. Estrada, P. Cicotti, P. Balaji, and M. Taufer. Accurate scoring of drug conformations at the extreme scale. In *15^{th} IEEE/ACM CCGrid*, pages 817–822, 2015.