Deciphering and Optimizing Multi-Task and Transfer Learning: a Random Matrix Approach

Anonymous Authors[1]

Abstract

This article provides theoretical insights into the inner workings of multi-task and transfer learning methods. To this end, we study the particularly tractable (least-square) support vector machine (LS-SVM) extension to multi-task learning (MTL) in the limit of large (p) and numerous data (n). This is achieved by means of a random matrix analysis applied to a Gaussian mixture model, and demonstrates that, as $n, p \to \infty$, the performance of MTL LS-SVM converges to a fully deterministic limit involving basic (small-dimensional) statistics of the data model.

Our major conclusions are that (i) the standard MTL LS-SVM algorithm is in general strongly biased and may dramatically fail (to the point that individual single-task LS-SVMs may outperform the MTL approach, even for quite resembling tasks): our analysis provides a simple method to correct these biases, and (ii) the sufficient statistics at play in the method are revealed and can be efficiently estimated, even for quite small datasets. The latter aspect is exploited to automatically optimize the hyperparameters without resorting to any cross-validation procedure.

Experiments on popular datasets are provided to further justify the applicability of our proposed approach to actual data. These experiments notably demonstrate that, when hyperparametrized using our theoretical findings, the simple and computationally-efficient as MTL LS-SVM algorithm is largely competitive, and even outperforms, much more elaborate state-of-the-art multi-task and transfer learning methods.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

1. Introduction

The advent of elaborate learning machines capable to surpass human performances on dedicated tasks has reopened past challenges in machine learning. Transfer learning and multitask learning (MTL) in general, by which learned tasks are used to help a machine learn other related tasks, is one of them. The particularly interesting aspects of multi-task learning lie here in the possibility (i) to exploit the resemblance (or dissemblance) between the datasets associated to each task so the tasks "help each other" and (ii) to train a machine on a specific target dataset comprised of few labelled data by exploiting much larger labelled datasets, however composed of different data.

Practical applications are numerous: the prediction of student test results for a collection of schools (Aitkin & Longford, 1986), of survival of patients in different clinics, of the value of many possibly related financial indicators (Allenby & Rossi, 1998), the modelling of the preferences of many individuals in a marketing context, etc.

Since MTL seeks to improve the performance of a task with the help of other related tasks, a central issue (i) to understand the functioning of MTL, (ii) to adequately adapt the hyperparameters and eventually (iii) to improve its performances consists in characterizing how MTL relates tasks to one another and in identifying which features are being "transferred". This article aims to decipher these fundamental aspects of MTL for sufficiently general data models.

Several data models may indeed be accounted for to enforce relatedness between tasks. A common assumption is that the data lie close to each other in a geometrical sense (Evgeniou & Pontil, 2004), live in a low dimensional manifold (Agarwal et al., 2010), or share a common prior (Daumé III, 2009). In the present article, we assume that, for all tasks, the data x arise from a 2-class mixture of generative models, which we will particularize for readability of our results to a Gaussian mixture with different statistical means in each class for each task.[1]

Methodologically, in its simplest approach, MTL algorithms

[1]The supplementary material extends this setting to a much broader and more realistic scope.

can be obtained from a mere extension of support vector machines (SVM), which accounts for more than one task. That is, instead of finding the hyperplane (through a normal vector ω) best separating the two classes of a unique dataset, (Evgeniou & Pontil, 2004) proposes to produce best separating hyperplanes (or normal vectors) $\omega_1, \ldots, \omega_k$ for each pair of data classes of k tasks, with the additional constraint that the normal vectors take the form $\omega_i = \omega_0 + v_i$ for some common vector w_0 and dedicated vectors v_i. The amplitude of the vectors v_i is controlled (through an additional hyperparameter) to enforce or relax task relatedness. We study this approach in the present article. Yet, in order to obtain explicit and thus more insightful results, we specifically resort to a least-square SVM (as proposed e.g., in (Xu et al., 2013)) rather than a margin-based SVM. This only marginally alters the overall behavior of the MTL algorithm and has no impact on the main insights drawn in the article.

Technically, the article exploits the powerful random matrix theory to study the performance of the MTL least-square (LS) SVM algorithm for data arising from a Gaussian mixture model, when the total number n and dimension p of the data are both large, i.e., as $n, p \to \infty$ with $p/n \to c \in (0, \infty)$. As such, our work follows after the recent wave of interest into the asymptotics of machine learning algorithms, such as studied lately in e.g., (Liao & Couillet, 2019; Deng et al., 2019; Mai & Couillet, 2018; El Karoui et al., 2010). Our main findings are as follows:

- we exhibit the sufficient statistics, which concretely enable task comparison in the MTL LS-SVM algorithm; we show that, even when data are of large dimensions ($p \to \infty$), these statistics remain small dimensional (they only scale with the number k of tasks);
- while it is conventional to manually set labels associated to each dataset within $\{-1, 1\}$, we prove that this choice is largely suboptimal and may even cause MTL to severely fail (even for tasks associated to "visually" extremely resembling datasets); we instead provide the optimal values for the labels of each dataset, which depend on the sought-for objective: these optimal values are furthermore easily estimated from very few training data (i.e., no cross-validation is needed);
- all the same, once the hyperplanes defined, for each new data $\mathbf{x}$ not part of the training set, the (LS-)SVM algorithm decides on a class allocation based on the comparison of a score $g(\mathbf{x})$ to a threshold ζ, usually set to zero. We demonstrate that, depending on the statistics and number of elements composing each dataset within each task, a bias is naturally induced that makes $\zeta = 0$ a largely suboptimal choice in general. We provide a correction for this bias, which again can be estimated from the training data alone;
- we establish the asymptotic performance of MTL LS-SVM which, here also, can be empirically anticipated from (very few) training data. This importantly allows one to set the decision threshold ζ to a value optimizing a given objective (e.g., minimizing a false alarm rate);
- we demonstrate on popular real datasets that our proposed optimized MTL LS-SVM is both resilient to real data and also manages, despite its not being a best-in-class MTL algorithm, to rival and sometimes largely outperform competing state-of-the-art algorithms.

Our many conclusions thus allow for an optimal use of MTL LS-SVM with performance-maximizing hyperparameters and strong performance guarantees. As such, the present article offers through MTL LS-SVM a viable fully-controlled (even better performing) alternative to state-of-the-art MTL.

The remainder of the article is organized as follows: Section 2 introduces the setting and assumptions of MTL LS-SVM, Section 3 then provides our main technical results and their interpretations and new insights, and Section 4 confirms through numerical experiments both on synthetic and real datasets that our theoretical results are valid and robust to genuine data.

Reproducibility. Matlab codes for reproducing the results of the paper are available in the supplementary materials.

Notation. $e_m^{[n]} \in \mathbb{R}^n$ is the canonical vector of $\mathbb{R}^n$ with $[e_m^{[n]}]_i = \delta_{mi}$. Moreover, $e_{ij}^{[2k]} = e_{2(i-1)+j}^{[2k]}$. Similarly, $E_{ij}^{[n]} \in \mathbb{R}^{n \times n}$ is the canonical matrix with $[E_{ij}^{[n]}]_{ab} = \delta_{ia}\delta_{jb}$. The notation $A \otimes B$ for matrices or vectors A, B is the Kronecker product. The notation $A \odot B$ for matrices or vectors A, B is the Hadamard product. $\mathcal{D}_x$ stands for a diagonal matrix containing on its diagonal the elements of the vector x and $A_{i\cdot}$ is the i-th line of matrix A.

2. The multi-task learning setting

Let $X \in \mathbb{R}^{p \times n}$ be a collection of n independent data vectors of dimension p. The data are divided into k subsets attached to individual "tasks". Specifically, letting $X = [X_1, \ldots, X_k]$, Task i is a binary classification problem from the training samples $X_i = [X_i^{(1)}, X_i^{(2)}] \in \mathbb{R}^{p \times n_i}$ with $X_i^{(j)} = [x_{i1}, \ldots, x_{in_{ij}}] \in \mathbb{R}^{p \times n_{ij}}$ the n_{ij} vectors of class $j \in \{1, 2\}$ for Task i. In particular, $n = \sum_{i=1}^k n_i$ and $n_i = n_{i1} + n_{i2}$ for each $i \in \{1, \ldots, k\}$.

To each datum $x_{il} \in \mathbb{R}^p$ of the training set is attached a corresponding "label" (or score) $y_{il} \in \mathbb{R}$. We will denote $y_i = [y_{i1}, \ldots, y_{in_i}]^\mathsf{T} \in \mathbb{R}^{n_i}$ the vector of all labels for Task i, and $y = [y_1^\mathsf{T}, \ldots, y_k^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^n$ the vector of all labels altogether. These labels are generally chosen to be ± 1 but,

for reasons that will become clear in the course of the article, we voluntarily do not enforce binary labels here.

Before detailing the multitask classification scheme, a preliminary task-wise centering operation is performed on the data, i.e., we will consider in the following the datasets

$$\mathring{X}_i = X_i \left(I_{n_i} - \frac{1}{n_i} \mathbb{1}_{n_i} \mathbb{1}_{n_i}^\mathsf{T} \right), \quad \forall i \in \{1, \ldots, k\}.$$

As such, we will systematically work on the labeled datasets $(\mathring{X}_1, y_1), \ldots, (\mathring{X}_k, y_k)$. Remark 1 motivates this important choice, which avoids extra biases produced by the algorithm.

2.1. The optimization framework

The multitask learning least square support vector machine (MTL LS-SVM) aims to predict, for input vectors $\mathbf{x} \in \mathbb{R}^p$ not belonging to the training set, their associated score y upon which a decision on the class allocation of $\mathbf{x}$ is taken, *for a given target task*. To this end, based on the labeled sets $(\mathring{X}_1, y_1), \ldots, (\mathring{X}_k, y_k)$, MTL LS-SVM determines the normal vectors $W = [\omega_1, \omega_2, \ldots, \omega_k] \in \mathbb{R}^{p \times k}$ and intercepts $b = [b_1, b_2, \ldots, b_k]^\mathsf{T} \in \mathbb{R}^k$ defining the k separating hyperplanes of the binary classification tasks. In order to account for the relatedness between tasks, each vector ω_i is expressed under the form $\omega_i = \omega_0 + v_i$ for some common vector $\omega_0 \in \mathbb{R}^p$ and task-dedicated vectors $v_i \in \mathbb{R}^p$, the norm of which will be constrained to enforce task-relatedness.

Formally, writing $V = [v_1, \ldots, v_k] \in \mathbb{R}^{p \times k}$ (so that $W = \omega_0 \mathbb{1}_k^\top + V$) and following the work of (Evgeniou & Pontil, 2004; Xu et al., 2013), the optimization function is given by

$$\min_{(\omega_0, V, b) \in \mathbb{R}^p \times \mathbb{R}^{p \times k} \times \mathbb{R}^k} \mathcal{J}(\omega_0, V, b) \tag{1}$$

where

$$\mathcal{J}(\omega_0, V, b) \equiv \frac{1}{2\lambda} \|\omega_0\|^2 + \frac{1}{2} \sum_{i=1}^k \frac{\|v_i\|^2}{\gamma_i} + \frac{1}{2} \sum_{i=1}^k \|\xi_i\|^2$$
$$\xi_i = y_i - (\mathring{X}_i^\mathsf{T} \omega_i + b_i \mathbb{1}_{n_i}), \quad \forall i \in \{1, \ldots, k\}$$

In this expression, the parameter λ enforces the relatedness of the tasks, while the parameters $\gamma_1, \ldots, \gamma_k$ enforce the correct classification of data in their respective classes.[2]

Being a quadratic optimization problem under linear equality constraints, ω_0, V, b can be obtained explicitly (see details in Section 1 of the Supplementary Material). The solution is best described through the expression of the hyperplanes $\omega_1, \ldots, \omega_k \in \mathbb{R}^p$ which take the form:

$$\omega_i = \left(e_i^{[k]\mathsf{T}} \otimes I_p \right) AZ\alpha,$$

with $b = (P^\mathsf{T} Q P)^{-1} P^\mathsf{T} Q y$, where

$$\begin{aligned} \alpha &= Q(y - Pb) \\ &= Q^{\frac{1}{2}} \left(I_n - Q^{\frac{1}{2}} P (P^\mathsf{T} Q P)^{-1} P^\mathsf{T} Q^{\frac{1}{2}} \right) Q^{\frac{1}{2}} y \in \mathbb{R}^n \end{aligned}$$

is the usual LS-SVM dual parameter in which

$$Q = \left(\frac{1}{kp} Z^\mathsf{T} A Z + I_n \right)^{-1} \in \mathbb{R}^{n \times n}$$
$$Z = \sum_{i=1}^k E_{ii}^{[k]} \otimes \mathring{X}_i \in \mathbb{R}^{pk \times n}$$
$$A = \left(\mathcal{D}_\gamma + \lambda \mathbb{1}\mathbb{1}^\mathsf{T} \right) \otimes I_p \in \mathbb{R}^{kp \times kp}$$
$$P = \sum_{i=1}^k E_{ii}^{[k]} \otimes \mathbb{1}_{n_i} \in \mathbb{R}^{n \times k}$$

with $\gamma = [\gamma_1, \ldots, \gamma_k]^\mathsf{T}$ and we recall that $\mathcal{D}_\gamma = \mathrm{diag}(\gamma)$.

The major difference between the single-task LS-SVM formulation and the present MTL version lies in the fact that the data $\mathring{X}_1, \ldots, \mathring{X}_k$ are not treated here simultaneously but through k distinct filters: this explains why $Z \in \mathbb{R}^{kp \times n}$ is not the mere concatenation $[\mathring{X}_1, \ldots, \mathring{X}_k]$ but a block-diagonal structure isolating each $\mathring{X}_i$. As such, the matrix A, which contains and balances all the hyperparameters $\lambda, \gamma_1, \ldots, \gamma_k$ of the problem, is the most important object in the MTL performance analysis.

From a technical standpoint, the fact that the data Z are expressed in a block structure will make the large dimensional random matrix analysis more challenging. Indeed, even in the simplest setting where the x_{ij} would be vectors of i.i.d. $\mathcal{N}(0,1)$ entries, the matrix Z is *not* a matrix of i.i.d. entries (due to blocks of zeros): the singular values of Z therefore do not asymptotically follow the popular Marčenko-Pastur distribution and more elaborate considerations are needed.

The matrix Q in the expression of α is often called in random matrix theory the *resolvent* of $\frac{1}{kp} Z^\mathsf{T} A Z$ and will play a central role in the analysis.

With this formulation for the solution (W, b), the prediction of the class of any new data point $\mathbf{x} \in \mathbb{R}^p$ for the target Task i is then obtained from the classification score

$$g_i(\mathbf{x}) = \frac{1}{kp} \left(e_i^{[k]} \otimes \mathring{\mathbf{x}}_i \right)^\mathsf{T} AZ\alpha + b_i \tag{2}$$

where, similar to the training data, $\mathring{\mathbf{x}} = (\mathbf{x} - \frac{1}{n_i} X_i \mathbb{1}_{n_i})$ is a centered version of $\mathbf{x}$ with respect to the training dataset for Task i.

[2] It is fundamental here, for the upcoming analysis, to place the hyperparameters λ and γ_i in front of $\|\omega_0\|^2$ and $\|v_i\|^2$, respectively (and not in front of $\|\zeta_i\|^2$). This is different from the proposed normalization in (Evgeniou & Pontil, 2004; Xu et al., 2013) but, as we will see, more consistent for the problem at hand and leading to a large simplification of our theoretical results.

Remark 1 (Shift invariance of the scores)**.** *If the score vectors $y_i \in \mathbb{R}^{n_i}$ are constant vectors, i.e., all data of the same task are affected the same score (or label), then $y = P\bar{y}$ for some vector $\bar{y} \in \mathbb{R}^k$ and we find that*

$$\alpha = Q\left(I_n - P(P^\mathsf{T}QP)^{-1}P^\mathsf{T}Q\right)P\bar{y} = 0.$$

As such, the normal vectors $\omega_i = (e_i^{[k]\mathsf{T}} \otimes I_p)AZ\alpha$, and as a consequence the performance of MTL LS-SVM, are insensitive to a constant shift in all the scores of each class.

2.2. The large dimensional statistical model

The first objective of the article is to quantify the performance of multitask learning, and thus of the (a priori intricate) statistics of $g_i(\mathbf{x})$, under a sufficiently simple and telling statistical model for training and test data. To this end, in the core of the article, we merely assume that the data arise from a Gaussian mixture model.

Assumption 1 (Distribution of X and $\mathbf{x}$)**.** *The columns of $[X, \mathbf{x}]$ are independent Gaussian random variables. Specifically, the n_{ij} samples $x_{i1}, \ldots, x_{in_{ij}}$ of class j for Task i are independent $\mathcal{N}(\mu_{ij}, I_p)$ random variables. As for $\mathbf{x}$, it follows an independent $\mathcal{N}(\mu_{\mathbf{x}}, I_p)$ random variable.*

In the Supplementary Material, Assumption 1 is relaxed to $[X, \mathbf{x}]$ arising from a generative model of the type $x_{ij} = h_{ij}(z_{ij})$ for $z_{ij} \sim \mathcal{N}(0, I_p)$ and $h_{ij} : \mathbb{R}^p \to \mathbb{R}^p$ a 1-Lipschitz function. This relaxed assumption has the strong advantage to cover extremely realistic data models, such as data arising from generative networks (e.g., GANs), and is shown in the Supplementary Material to be universal in the sense that the large n, p asymptotic performances of MTL LS-SVM only depend on the statistical means and covariances of the x_{ij} (in particular, the performances coincide with those under a mere Gaussian mixture model).

Our central technical method for the performance evaluation of the MTL LS-SVM algorithm consists in placing ourselves under the large p, n regime of random matrix theory.

Assumption 2 (Growth Rate)**.** *As $n \to \infty$, $n/p \to c_0 > 0$ and, for $1 \le i \le k$, $1 \le j \le m$, $\frac{n_{ij}}{n} \to c_{ij} > 0$. We further denote $c_i = c_{i1} + c_{i2}$ and $c = [c_1, \ldots, c_k]^\mathsf{T} \in \mathbb{R}^k$. Besides, for each i, letting $\Delta\mu_i \equiv \mu_{i1} - \mu_{i2}$, $\|\Delta\mu_i\| = \mathcal{O}(1)$.*

The quantity $\Delta\mu_i = \mu_{i1} - \mu_{i2}$ naturally arises as it corresponds to the statistical mean of $\mathring{x}_{ij}$ (the j-th column of $\mathring{X}_i$), up to scaling by $\frac{n_{i2}}{n}$ if x_{ij} is in Class 1 or $-\frac{n_{i1}}{n}$ if in Class 2. The fact that $\|\mu_{i1} - \mu_{i2}\| = \mathcal{O}(1)$ with respect to p, n is fundamental to ensure that, in the large p, n limit, the asymptotic classification performance of MTL LS-SVM remains non-trivial (i.e., is neither 0 nor 1 in the limit). This will become obvious from the statement of our main results.

With these notations and assumptions at hand, we are in position to present our main theoretical results.

3. The multi-task Learning analysis

3.1. Asymptotic classification error of MTL LS-SVM

3.1.1. Technical strategy and notations

To compute the statistics of $g_i(\mathbf{x})$ defined in (2), we shall resort to determining so-called *deterministic equivalents* for the matrices Q, AZQ, etc., which appear at the core of the formulation of $g_i(\mathbf{x})$. A deterministic equivalent, say $\bar{F} \in \mathbb{R}^{n\times p}$, of a given random matrix $F \in \mathbb{R}^{n\times p}$, denoted $F \leftrightarrow \bar{F}$, is defined by the fact that, for any deterministic linear functional $f : \mathbb{R}^{n\times p} \to \mathbb{R}$, $f(F - \bar{F}) \to 0$ almost surely (for instance, for u, v of unit norm, $u^\mathsf{T}(F - \bar{F})v \xrightarrow{\text{a.s.}} 0$ and, for $A \in \mathbb{R}^{p\times n}$ deterministic of bounded operator norm, $\frac{1}{n}\mathrm{tr}A(F - \bar{F}) \xrightarrow{\text{a.s.}} 0$). Deterministic equivalents are thus particularly suitable to handle bilinear forms involving the random matrix F.

The deterministic equivalents required here are provided explicitly in Lemma 1 of the Supplementary Material. Our strategy then consists in "decoupling" the effect of the data statistics from those of the MTL hyperparameters $\lambda, \gamma_1, \ldots, \gamma_k$. Specifically, we extract two fundamental quantities for our analysis: the data-related matrix $\mathcal{M} \in \mathbb{R}^{2k\times 2k}$ and the hyperparameter matrix $\mathcal{A} \in \mathbb{R}^{k\times k}$:

$$\mathcal{M} = \sum_{i,j=1}^{k} \Delta\mu_i^\mathsf{T}\Delta\mu_j \left(E_{ij}^{[k]} \otimes \mathbb{c}_i\mathbb{c}_j^\mathsf{T}\right),\ \mathbb{c}_i = \begin{bmatrix} \frac{c_{i2}}{c_i}\sqrt{\frac{c_{i1}}{c_i}} \\ -\frac{c_{i1}}{c_i}\sqrt{\frac{c_{i2}}{c_i}} \end{bmatrix}$$

$$\mathcal{A} = \left(I_k + \mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}}\left(\mathcal{D}_\gamma + \lambda\mathbb{1}_k\mathbb{1}_k^\mathsf{T}\right)^{-1}\mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}}\right)^{-1}$$

where $\tilde{\Delta} = [\tilde{\Delta}_1, \ldots, \tilde{\Delta}_k]^\mathsf{T}$ are the unique positive solutions to the implicit system

$$\tilde{\Delta}_i = \frac{c_i}{c_0} - \mathcal{A}_{ii}$$

(this is implicit because $\mathcal{A}$ is a function of the $\tilde{\Delta}_i$'s). In passing, it will appear convenient to use the shortcut notation $\bar{\Delta} = [\bar{\Delta}_{11}, \ldots, \bar{\Delta}_{k2}]^\mathsf{T} \in \mathbb{R}^{2k}$ where

$$\bar{\Delta}_{ij} = \frac{c_{ij}}{c_i}c_0\tilde{\Delta}_i. \tag{3}$$

We will see that $\mathcal{M}$ plays the role, in the limit of large p, n, of a sufficient statistic for the performance of the MTL LS-SVM algorithm with respect to (i) the data statistics $\mu_{11}, \ldots, \mu_{k2}$ and (ii) the (limiting) relative number $c_{11}/c_1, \ldots, c_{k2}/c_k$ of elements per class in each task.

As for $\mathcal{A}$, it captures the information about the impact of the hyperparameters $\lambda, \gamma_1, \ldots, \gamma_k$ and the dimension ratios $c_1, \ldots, c_k$ and c_0.

These two matrices will be combined in the core matrix $\Gamma \in \mathbb{R}^{2k\times 2k}$ of the upcoming MTL LS-SVM performance

analysis, defined as

$$\Gamma = \left(I_{2k} + \left(\mathcal{A} \otimes \mathbb{1}_2\mathbb{1}_2^\mathsf{T}\right) \odot \mathcal{M}\right)^{-1}$$

where we recall that '$\odot$' is the Hadamard (element-wise) matrix product.

We raised in the introduction of Section 2 that we purposely relax the binary "labels" y_{ij} associated to each datum x_{ij} in each task to become "scores" $y_{ij} \in \mathbb{R}$. This will be shown to have fundamental consequences to the MTL performances. Yet, since $x_{i1}, \ldots, x_{in_i}$ are i.i.d. data vectors, we impose the natural constraint of equal scores $y_{i1} = \ldots = y_{in_i}$ within every class. As such, we may reduce the complete score vector $y \in \mathbb{R}^n$ under the form

$$y = \left[\tilde{y}_{11}\mathbb{1}_{n_{11}}^\mathsf{T}, \ldots, \tilde{y}_{k2}\mathbb{1}_{n_{k2}}^\mathsf{T}\right]^\mathsf{T}$$

for $\tilde{y} = [\tilde{y}_{11}, \ldots, \tilde{y}_{k2}]^\mathsf{T} \in \mathbb{R}^{2k}$. From Remark 1, it is also clear that, the performances of MTL being insensitive to a constant shift in the scores y_{i1} and y_{i2} of every given task i, the recentered version $\mathring{\tilde{y}} = [\mathring{\tilde{y}}_{11}, \ldots, \mathring{\tilde{y}}_{k2}]^\mathsf{T}$ of $\tilde{y}$, where

$$\mathring{\tilde{y}}_{ij} = \tilde{y}_{ij} - \left(\frac{n_{i1}}{n_i}\tilde{y}_{i1} + \frac{n_{i2}}{n_i}\tilde{y}_{i2}\right)$$

will be central in the upcoming results.

With these notations at hand, the asymptotic behavior of the output of the MTL LS-SVM algorithm can be described.

3.1.2. Main result

Theorem 1 (Asymptotic of $g_i(\mathbf{x})$)**.** *Under Assumptions 1–2, for $\mathbf{x} \sim \mathcal{N}(\mu_\mathbf{x}, I_p)$ with $\mu_\mathbf{x} = \mu_{ij}$,*

$$g_i(\mathbf{x}) - G_{ij} \to 0, \quad G_{ij} \sim \mathcal{N}(m_{ij}, \sigma_i^2)$$

in distribution where, letting $m = [m_{11}, \ldots, m_{k2}]^\mathsf{T}$,

$$m = \tilde{y} - \mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}}\Gamma\mathcal{D}_{\tilde{\Delta}}^{\frac{1}{2}}\mathring{\tilde{y}}$$
$$\sigma_i^2 = \frac{1}{\tilde{\Delta}_i}\mathring{\tilde{y}}^\mathsf{T}\mathcal{D}_{\tilde{\Delta}}^{\frac{1}{2}}\Gamma\left(\mathcal{D}_{\frac{\mathcal{K}_{i\cdot}}{c}\otimes\mathbb{1}_2} + \mathcal{V}_i\right)\Gamma\mathcal{D}_{\tilde{\Delta}}^{\frac{1}{2}}\mathring{\tilde{y}}$$

with

$$\mathcal{V}_i = \frac{1}{c_0}\left(\mathcal{A}\mathcal{D}_{\frac{c_0\mathcal{K}_{i\cdot}}{c} + e_i^{[k]}\mathcal{A}} \otimes \mathbb{1}_2\mathbb{1}_2^\mathsf{T}\right) \odot M$$
$$\mathcal{K} = [\mathcal{A} \odot \mathcal{A}]\left(I_k - \mathcal{D}_{\frac{c_0}{kc}[\mathcal{A}\odot\mathcal{A}]}\right)^{-1}.$$

As anticipated, the statistics of the classification scores $g_i(\mathbf{x})$ mainly depend on the data statistics $\mu_{i'j'}$ and on the hyper-parameters λ and $\gamma_1, \ldots, \gamma_k$ through the matrix Γ (and more marginally through $\mathcal{V}_i$ and $\mathcal{K}$ for the variances). Comparing the expression of $g_i(\mathbf{x})$ in (2) and that of m_{ij} (i.e., the large dimensional approximation of $\mathbb{E}[g_i(\mathbf{x})]$), we may see $\Gamma \in \mathbb{R}^{2k\times 2k}$ as a "condensed" form of $Q \in \mathbb{R}^{n\times n}$ (both are resolvents, of the form $(I + B)^{-1}$ for different matrices B) gathering the sufficient statistics of the MTL problem.

Before entering further considerations about Γ, a few comments and immediate corollaries relating Theorem 1 to the performance of MTL LS-SVM are in order. Since $g_i(\mathbf{x})$ has a Gaussian limit centered about m_{ij}, the (asymptotic) standard decision for $\mathbf{x}$ to be allocated to Class 1 ($\mathbf{x} \to \mathcal{C}_1$) or Class 2 ($\mathbf{x} \to \mathcal{C}_2$) for Task i is obtained by the "averaged-mean" test

$$g_i(\mathbf{x}) \underset{\mathcal{C}_2}{\overset{\mathcal{C}_1}{\gtrless}} \frac{1}{2}(m_{i1} + m_{i2}) \tag{4}$$

the probability of classification error of which is then

$$\begin{aligned}\epsilon_{i1} &\equiv P\left(g_i(\mathbf{x}) \geq \frac{m_{i1} + m_{i2}}{2}\middle|\mathbf{x} \in \mathcal{C}_1\right)\\ &= \mathcal{Q}\left(\frac{m_{i1} - m_{i2}}{2\sigma_i}\right) + o(1)\end{aligned} \tag{5}$$

with m_{ij}, σ_i as in Theorem 1 and $\mathcal{Q}(t) = \int_t^\infty e^{-\frac{u^2}{2}}du$.

In particular, since $\tilde{y}_{i1} - \tilde{y}_{i2} = \mathring{\tilde{y}}_{i1} - \mathring{\tilde{y}}_{i2}$, this result confirms Remark 1 according to which the classification performance only depends on $\mathring{\tilde{y}}$. We may then freely impose that $\tilde{y} = \mathring{\tilde{y}}$ and obtain in particular that the mean limiting scores become

$$m = (I_{2k} - \mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}}\Gamma\mathcal{D}_{\tilde{\Delta}}^{\frac{1}{2}})\mathring{\tilde{y}} \tag{6}$$

and the classification error for Task i is directly related to

$$\begin{aligned}m_{i1} - m_{i2} &= (e_{i1}^{[2k]} - e_{i2}^{[2k]})^\mathsf{T}m\\ &= \mathring{\tilde{y}}_{i1} - \mathring{\tilde{y}}_{i2} - (e_{i1}^{[2k]} - e_{i2}^{[2k]})^\mathsf{T}\mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}}\Gamma\mathcal{D}_{\tilde{\Delta}}^{\frac{1}{2}}\mathring{\tilde{y}}.\end{aligned}$$

We therefore importantly conclude that, in general, while it is intuitively expected that the difference in the scores $g_i(\mathbf{x})$ for $\mathbf{x} \in \mathcal{C}_1$ or $\mathcal{C}_2$ be well approximated by $\mathring{\tilde{y}}_{i1} - \mathring{\tilde{y}}_{i2}$ (the difference of data scores in the same task), the second term $(e_{i1}^{[2k]} - e_{i2}^{[2k]})^\mathsf{T}\mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}}\Gamma\mathcal{D}_{\tilde{\Delta}}^{\frac{1}{2}}\mathring{\tilde{y}}$ of the equality may dramatically alter this expected behavior. The impact of matrix Γ on the training scores $\mathring{\tilde{y}}$, as well as an appropriate preliminary choice of $\mathring{\tilde{y}}$ (possibly quite counterproductive as we will see), are thus central to the performance of MTL LS-SVM.

Further comment on Γ is due before moving to practical consequences of Theorem 1. From the expression $(\mathcal{A} \otimes \mathbb{1}_2\mathbb{1}_2^\mathsf{T})^{-1} \odot \mathcal{M}$, we observe that: (i) if $\lambda \ll 1$, then $\mathcal{A}$ is diagonal dominant and thus "filters out" in the Hadamard product all off-diagonal entries of $\mathcal{M}$, i.e., all cross-terms $\Delta\mu_i^\mathsf{T}\Delta\mu_j$ for $i \neq j$, therefore refusing to exploit the correlation between tasks; (ii) if instead λ is not small, $\mathcal{A}$ may be developed (using the Sherman-Morrison matrix inverse

formulas) as the sum of a diagonal matrix, which again filters out the $\Delta\mu_i^\mathsf{T}\Delta\mu_j$ for $i \neq j$, and of a rank-one matrix which instead performs a weighted sum (through the γ_i and the $\tilde{\Delta}_i$ of the entries of $\mathcal{M}$. Specifically, letting $\gamma^{-1} = (\gamma_1^{-1}, \ldots, \gamma_k^{-1})^\mathsf{T}$, we have

$$\left(D_\gamma + \lambda \mathbb{1}_k \mathbb{1}_k^\mathsf{T}\right)^{-1} = D_\gamma^{-1} - \frac{\lambda \gamma^{-1}\gamma^{-1\mathsf{T}}}{1 + \lambda \frac{1}{k}\sum_{i=1}^k \gamma_i^{-1}}.$$

As such, disregarding the regularization effect of the $\tilde{\Delta}_i$'s, the off-diagonal $\Delta\mu_i^\mathsf{T}\Delta\mu_j$ entry of $\mathcal{M}$ is weighted with coefficient $(\gamma_i\gamma_j)^{-1}$: the impact of the γ_i's is thus strongly associated to the relevance of the correlation between tasks.

3.1.3. Practical consequences

A fundamental aspect of Theorem 1 is that it concludes that the performances of the *large dimensional* (large n, large p) classification problem at hand merely boils down to $2k$ dimensional statistics, as all objects defined in the theorem statement are at most of size $2k$. More importantly from a practical perspective, these $2k$-dimensional "sufficient statistics" are easily amenable to fast and efficient estimation: it indeed only requires a few training data samples to estimate all quantities involved in the theorem (which, as a corollary, lets one envision the possibility of efficient transfer learning methods based on very scarce data samples).

Remark 2 (On the estimation of m_{ij} and σ_i)**.** *All quantities defined in Theorem 1 are a priori known, apart from the k^2 inner products $\Delta\mu_i^\mathsf{T}\Delta\mu_j$ for $1 \leq i, j \leq k$. For these, define, for $l = 1, 2$, $\mathcal{S}_{il} \subset \{1, \ldots, n_{il}\}$ and the corresponding indicator vector $\mathbb{j}_{il} \in \mathbb{R}^{n_i}$ with $[\mathbb{j}_{ij}]_a = \delta_{a \in \mathcal{S}_{ij}}$. For $i = j$, further let $\mathcal{S}'_{il} \subset \{1, \ldots, n_{il}\}$ with $\mathcal{S}'_{il} \cap \mathcal{S}_{il} = \emptyset$ and the corresponding indicator vector $\mathbb{j}'_{il} \in \mathbb{R}^{n_i}$. Then, the following estimates hold:*

$$\Delta\mu_i^\mathsf{T}\Delta\mu_j - \left(\frac{\mathbb{j}_{i1}}{|\mathcal{S}_{i1}|} - \frac{\mathbb{j}_{i2}}{|\mathcal{S}_{i2}|}\right)^\mathsf{T} X_i^\mathsf{T} X_j \left(\frac{\mathbb{j}_{j1}}{|\mathcal{S}_{j1}|} - \frac{\mathbb{j}_{j2}}{|\mathcal{S}_{j2}|}\right)$$
$$= O\left((p \min_{l\in\{1,2\}} \{|\mathcal{S}_{il}|, |\mathcal{S}_{jl}|\})^{-\frac{1}{2}}\right)$$
$$\Delta\mu_i^\mathsf{T}\Delta\mu_i - \left(\frac{\mathbb{j}_{i1}}{|\mathcal{S}_{i1}|} - \frac{\mathbb{j}_{i2}}{|\mathcal{S}_{i2}|}\right)^\mathsf{T} X_i^\mathsf{T} X_i \left(\frac{\mathbb{j}'_{i1}}{|\mathcal{S}'_{i1}|} - \frac{\mathbb{j}'_{i2}}{|\mathcal{S}'_{i2}|}\right)$$
$$= O\left((p \min_{l\in\{1,2\}} \{|\mathcal{S}_{il}|, |\mathcal{S}'_{il}|\})^{-\frac{1}{2}}\right).$$

Observe in particular that a single sample (two when $i = j$) per task and per class ($|\mathcal{S}_{il}| = 1$) is sufficient to obtain a consistent estimate for all quantities so long that p is large. In a transfer learning setting where some tasks may contain few labeled data, it is thus still possible to infer (and, as seen next, optimize) the MTL algorithm. Of course, when more data are available, under our assumption that $p \sim n$, taking all samples in the averaging, the convergence speed is of order $O(1/\sqrt{np}) = O(1/n)$, which is twice the speed of the usual central-limit theorem.

Estimating m_{ij} and σ_i not only allows one to anticipate theoretical performances but also enables the actual estimation of the decision threshold $\frac{1}{2}(m_{i1} + m_{i2})$ of the test (4) and, as shown next, opens the possibility to largely optimize MTL LS-SVM through an (asymptotically) optimal choice of the training scores $\tilde{y}$. This is explored in Section 3.2.

Remark 3 (From binary to multiclass MTL)**.** *Accessing the vector m extends the MTL framework to a multiclass-per-task MTL by discarding the well known inherent biases of multiclass SVM. In the context of L_i classes for Task i, using a one-versus-all approach (for each $\ell \in \{1, \ldots, L_i\}$, one MTL LS-SVM algorithms with Class "1" being the target class ℓ and Class "2" all other classes at once), one needs to access L_i pairs of values $(m_{i1}(\ell), m_{i2}(\ell))$ and, for a new $\mathbf{x}$, decide on the genuine class of $\mathbf{x}$ based on the largest value among $g_i(\mathbf{x}; 1) - m_{i1}(1), \ldots, g_i(\mathbf{x}; L_i) - m_{i1}(L_i)$ with $g_i(\mathbf{x}; \ell)$ the output score for "Class ℓ versus all".*

For simplicity, from Remark 1, one may choose smart shift vectors $\bar{y}(\ell) \in \mathbb{R}^k$ for the scores $\tilde{y}(\ell) \in \mathbb{R}^{2k}$ such that, $\tilde{y}(\ell) + \bar{y}(\ell) \otimes \mathbb{1}_2$ gives $m_{i1}(\ell) = 0$ for each ℓ. Under this shift, the elected class is that for which $g_i(\mathbf{x}; \ell)$ is maximum.

The simulation results on multiclass MTL reported in Section 4 fundamentally exploit this remark.

As a last important remark for the MTL optimization performed in the coming section, the variance σ_i^2 in Theorem 1 does not depend on the genuine class j of $\mathbf{x}$ (which follows from $\mathcal{V}_i$ and $\mathcal{K}$ being only functions of i). This remark no longer holds if the data covariance matrices $\mathbb{E}[x_{ij}x_{ij}^\mathsf{T}] - \mu_{ij}\mu_{ij}^\mathsf{T}$ differ from the identity matrix (see details in Supplementary Material).

3.2. Optimization of $\tilde{y}$

The asymptotics in Theorem 1 depend in an elegant manner on the training data labels (scores) $\tilde{y}$. Since the variance σ_i^2 is independent of the classes, we easily determine the vector $\tilde{y}$ minimizing the misclassification probability for Task i:

$$\tilde{y}^\star = \arg\max_{\tilde{y}\in\mathbb{R}^{2k}} \frac{(m_{i1} - m_{i2})^2}{\sigma_i^2}$$
$$= \arg\max_{\tilde{y}\in\mathbb{R}^{2k}} \frac{\|\tilde{y}^\mathsf{T}(I_{2k} - \mathcal{D}_{\tilde{\Delta}}^{\frac{1}{2}}\Gamma\mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}})(e_{i1}^{[2k]} - e_{i2}^{[2k]})\|^2}{\tilde{y}^\mathsf{T}\mathcal{D}_{\tilde{\Delta}}^{\frac{1}{2}}\Gamma(\mathcal{D}_{\frac{\mathcal{K}_i}{c}\otimes\mathbb{1}_2} + \mathcal{V}_i)\Gamma\mathcal{D}_{\tilde{\Delta}}^{\frac{1}{2}}\tilde{y}}$$

for which the solution is explicitly defined by:

$$\tilde{y}^\star = \mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}}\Gamma^{-1}\mathcal{H}[(\mathcal{A}\otimes\mathbb{1}_2\mathbb{1}_2^\mathsf{T}) \odot \mathcal{M}]\mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}}(e_{i1}^{[2k]} - e_{i2}^{[2k]}) \quad (7)$$

in which $\mathcal{H} \equiv (\mathcal{D}_{\frac{\mathcal{K}_i}{c}\otimes\mathbb{1}_2} + \mathcal{V}_i)^{-1}$.

For this choice of $\tilde{y}^\star$, the corresponding (asymptotically) optimal classification error ϵ_{i1} defined in (5) is then

$$\epsilon_{i1}^\star = \mathcal{Q}\left(\frac{1}{2}\sqrt{(e_{i1}^{[2k]} - e_{i2}^{[2k]})^\mathsf{T}\mathcal{G}(e_{i1}^{[2k]} - e_{i2}^{[2k]})}\right)$$

for $\mathcal{G} = \mathcal{D}_\Delta^{\frac{1}{2}}[(\mathcal{A}\otimes \mathbb{1}_2\mathbb{1}_2^\mathsf{T}) \odot M]\mathcal{H}[(\mathcal{A}\otimes \mathbb{1}_2\mathbb{1}_2^\mathsf{T}) \odot M]\mathcal{D}_\Delta^{\frac{1}{2}}$.

The only non-diagonal matrices in (7) are Γ and $\mathcal{V}_i$ in which $\mathcal{M}$ plays the role of a "variance profile" matrix. In particular, assume $\Delta\mu_i^\mathsf{T}\Delta\mu_\ell = 0$ for all $\ell \neq i$, i.e., the statistical means of all tasks are orthogonal to those of Task i. Then the two rows and columns of $\mathcal{M}$ associated to Task i are all zero but on the 2×2 diagonal block. Therefore, $\tilde{y}^\star$ will have all zero entries but on its Task i two elements. All other choices for the null entries of $\tilde{y}^\star$ (such as the usual $\tilde{y} = [1, -1, \ldots, 1, -1]^\mathsf{T}$) would be suboptimal and (possibly severely) detrimental to the classification performance of Task i (not by altering the means m_{i1}, m_{i2} but by increasing the variance σ_i^2). This extreme example strongly suggests that, in order to maximize the MTL performance on Task i, one must impose low scores $\tilde{y}_{jl}$ to all Tasks j strongly different from Task i.

The choice $\tilde{y} = [1, -1, \ldots, 1, -1]^\mathsf{T}$ can also be very detrimental when $\Delta\mu_i^\mathsf{T}\Delta\mu_j < 0$ for some i, j: that is, when the mapping of the two classes within each task is reversed (e.g., if Class 1 in Task 1 is closer to Class 2 than Class 1 in Task 2). In this setting, it is easily seen that $\tilde{y} = [1, -1, \ldots, 1, -1]^\mathsf{T}$ works against the classification and performs much worse than a single-task LS-SVM.

Another interesting conclusion arises from the simplified setting of equal number of samples per task and per class, i.e., $n_{11} = \ldots = n_{k2}$. In this case, $\tilde{y}^\star = \Gamma^{-1}\mathcal{H}\left((\mathcal{A}\otimes \mathbb{1}_2\mathbb{1}_2^\mathsf{T}) \odot \mathcal{M}\right)(e_{i1}^{[2k]} - e_{i2}^{[2k]})$ in which all matrices are organized in 2×2 blocks of equal entries. This immediately implies that $\tilde{y}_{j1}^\star = -\tilde{y}_{j2}^\star$ for all j. So in particular, the detection threshold $\frac{1}{2}(m_{i1} + m_{i2})$ of the averaged-mean test (4) is zero (as conventionally assumed). In all other settings for the n_{jl}'s, it is very unlikely that $\tilde{y}_{i1}^\star = -\tilde{y}_{i2}^\star$ and the optimal decision threshold *must* also be estimated.

4. Experiments

Our theoretical results (the data-driven optimal tuning of MTL LS-SVM, as well as the anticipation of classification performance) find various practical applications and consequences. We here exploit them consecutively in the context of transfer learning, first on a binary (hypothesis test-like) decision on both synthetic and real data, and then on a multiclass classification on real data.[3]

[3]Real data are all centered and normalized *per task* in such a way that the averaged squared norm after centering equals p (coinciding with the trace of the identity covariance matrix).

4.1. Binary decision

We here apply the results of MTL LS-SVM to a hypothesis test on a *target* task t based on training samples from both a source task s and the target task t. That is, instead of relying on the "averaged-mean" decision procedure from (4), we instead consider the test

$$g_t(\mathbf{x}) \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \zeta$$

where $\mathcal{H}_0$ is the null hypothesis (say, Class 2) and $\mathcal{H}_1$ the alternative (say, Class 1) and $\zeta = \zeta(\eta)$ is a decision threshold selected in such a way to have the false alarm constraint rate $P(g_t(\mathbf{x}) \geq \zeta \mid \mathbf{x} \in \mathcal{H}_0) \leq \eta$, for some given η. The objective is then here to maximize the correct detection rate $P(g_t(\mathbf{x}) \geq \zeta \mid \mathbf{x} \in \mathcal{H}_1)$, which induces a different value for the optimal scores $\tilde{y}^\star$ than presented in (7) but can be constructed according to the same procedure.

Figure 1 depicts the algorithm performance through a receiver-operating curve (ROC) for false alarm rates η on both synthetic and real-world data. Both theoretical (Th) asymptotics (used to set the decision threshold ζ) and actual performances (Sim) are displayed for optimal (Opt) choices of $\tilde{y}$ (Opt) and for $\tilde{y} = [-1, 1, -1, 1]$ (Non-Opt).

The synthetic data is a two-task ($k = 2$) setting in which $x_{1j} \sim \mathcal{N}(\pm\mu_{11}, I_p)$ (i.e., $\mu_{12} = -\mu_{11}$) and $x_{2j} \sim \mathcal{N}(\pm\mu_{21}, I_p)$, where $\mu_{21} = \beta\mu_{11} + \sqrt{1-\beta^2}\mu_{11}^\perp$, μ_{11} is a unit-norm vector and $\mu_{11}^\perp$ any unit-norm vector orthogonal to μ_{11}. We take here $\beta = 0.5$, so that both tasks are "slightly" correlated.

The real-world experiments exploit the MIT-BIH Arrhythmia dataset (Moody & Mark, 2001). The dataset consists of 109 446 samples from 5 medical heart condition categories: "Normal (N)": 0, "Atrial premature (S)": 1, "Ventricular (V)": 2, "Ventricular-Norma (F)": 3, "Unclassifiable (Q)": 4. For illustration, we considered the binary classification with source Classes $\{1, 2\}$ and target Classes $\{3, 4\}$. The false alarm rates consists in misclassifying (target) Class 3 into Class 4 and the performance objective is on maximizing the correct classification of target Class 4.

Both synthetic and real data graphs confirm, here in the hypothesis testing problem, the large superiority of our proposed optimized MTL LS-SVM over the standard non optimized alternative. Besides, the theoretical classification error prediction is an accurate fit to the actual empirical performance, even for not so large values of p, n_{ij}.

4.2. Multiclass transfer learning

We next turn to the classical Office+Caltech256 (Saenko et al., 2010; Griffin et al., 2007) real data benchmark (images) for transfer learning, consisting of the 10 categories shared by both datasets. As in previous works, we compare

Table 1. Classification accuracy over Office+Caltech256 database. c(Caltech), w(Webcam), a(Amazon), d(dslr)

S/T	c→w	w→c	c→a	a→c	w→a	a→d	d→a	w→d	c→d	d→c	a→w	d→w
LSSVM	79.47	47.70	71.04	49.65	68.13	57.50	70.00	73.75	67.50	46.45	74.83	84.11
MMDT	69.47	**69.47**	66.53	39.70	65.24	59.50	62.16	**86.06**	56.94	27.92	68.54	**87.88**
ILS	24.5	20.92	25.21	21.10	22.92	26.25	27.08	43.75	30.00	26.95	15.23	57.62
CDLS	*82.28*	*54.21*	*71.58*	**54.49**	**71.52**	*68.56*	*70.54*	69.44	*69.44*	**53.86**	**81.59**	82.78
Proposed	**86.09**	49.65	**72.29**	*50.35*	*68.83*	**73.75**	**71.25**	*72.50*	**77.50**	*48.05*	*80.13*	*85.43*

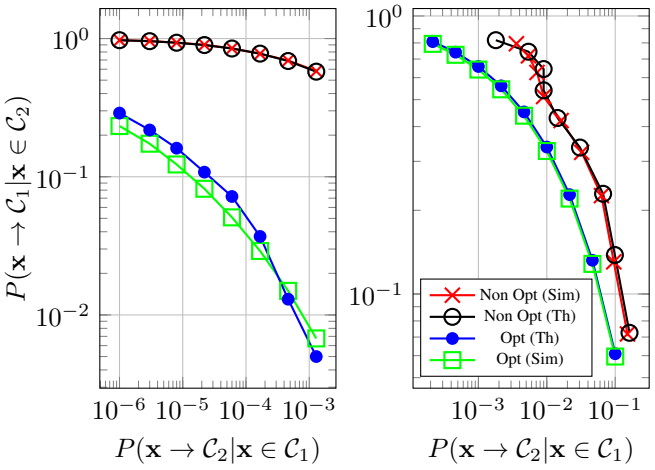

Figure 1. ROC curve for proposed versus non-optimized MTL LS-SVM. **(Left)** Synthetic data with $p = 128$, $n_{11} = 384$, $n_{12} = 256$, $n_{21} = 64$, $n_{22} = 40$, $\mu_{11} = -\mu_{12} = [1, 0, \ldots, 0]^\mathsf{T}$, $\mu_{21} = -\mu_{22} = [.87, .5, 0, \ldots, 0]^\mathsf{T}$. **(Right)** MIT-BIH arrhythmia database, with $p = 550$, $n_{ij} = 500$.

Algorithm 1 Proposed Multi Task Learning algorithm.

Input: Training samples $X = [X_1, \ldots, X_k]$ with $X_i = [X_i^{(1)}, \ldots, X_i^{(L_i)}]$, $X_i^\ell \in \mathbb{R}^{p \times n_{i\ell}}$ and test data $\mathbf{x}$.
Output: Estimated class $\hat{\ell} \in \{1, \ldots, L_t\}$ of $\mathbf{x}$ for target Task t.
for $j = 1$ **to** L_t **do**
  Estimate: Matrix $\mathcal{M}$ from Remark 2 and $\tilde{\Delta}$ by solving (3), using $X_1^{(j)}, \ldots, X_k^{(j)}$ as data of class 1 and $X \setminus \{X_1^{(j)}, \ldots, X_k^{(j)}\}$ as data of class 2.
  Create scores

$$\tilde{y}^\star(j) = \mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}} \Gamma^{-1} \mathcal{H}[(\mathcal{A} \otimes \mathbb{1}_2 \mathbb{1}_2^\mathsf{T}) \odot \mathcal{M}] \mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}} (e_{t1}^{[2k]} - e_{t2}^{[2k]}).$$

  Estimate and center $m = m(j)$ from Theorem 1 as per Remark 3.
  (Optional) Estimate the theoretical classification error $\epsilon_t(\lambda, \gamma)$ from (5) and minimize over (λ, γ).[4]
  Compute classification scores $g_t(\mathbf{x}; j)$ according to (2).
end for
Output: $\hat{\ell} = \arg\max_{\ell \in \{1, \ldots, L_t\}} g_t(\mathbf{x}; \ell)$.

images using $p = 800$ SURF-BoW features. Half of the samples of the target is randomly selected for the test data and the accuracy is evaluated over 20 trials. Based on our discussions and notably on Remark 3, we propose and use Algorithm 1. Table 1 reports the accuracy obtained by the algorithm (Proposed) versus non optimized LS-SVM (Xu et al., 2013) and state-of-the-art transfer learning algorithms: MMDT, CDLS and ILS.[5]

Table 1 demonstrates that our proposed improved MTL LS-SVM, despite its simplicity and unlike the competing methods used for comparison, has stable performances and is quite competitive. It even either outperforms all other methods or is second-to-best (except for a single scenario). But, most importantly, the method comes along with performance predictions and guarantees, which none of the competing works are able to provide.

5. Conclusion: beyond MTL

Through the recently revived example of transfer learning (and more generally multitask learning), we have demonstrated the capability of random matrix theory to (i) predict and improve the performance of machine learning algorithms and, most importantly, to (ii) turn simplistic (and in theory largely suboptimal) methods, such as here LS-SVM, into competitive state-of-the-art algorithms. In the present isotropic setting of Gaussian mixtures with identity covariance which is quite "universal" (as shown in Supplementary Material) and thus already appropriate to handle real data, one may in fact surmise the optimality of the least square methods, thereby opening the possibility to prove that MTL LS-SVM is likely close-to-optimal even in real data settings.

This is merely a first step into a generalized use of random matrix theory and large dimensional statistics to devise much-needed *low computational cost* and *explainable*, yet highly competitive, machine learning methods from elementary optimization schemes.

[5]MMDT: Max Margin Domain Transform, proposed in (Hoffman et al., 2013), applies a weighted SVM on a learnt transformation of the source and target. CDLS: Cross-Domain Landmark Selection, proposed in (Hubert Tsai et al., 2016), derives a domain invariant feature space. ILS: Invariant Latent Space, proposed in (Herath et al., 2017), learns an invariant latent space to reduce the discrepancy between source and target.

References

Agarwal, A., Gerber, S., and Daume, H. Learning multiple tasks using manifold regularization. In *Advances in neural information processing systems*, pp. 46–54, 2010.

Aitkin, M. and Longford, N. Statistical modelling issues in school effectiveness studies. *Journal of the Royal Statistical Society: Series A (General)*, 149(1):1–26, 1986.

Allenby, G. M. and Rossi, P. E. Marketing models of consumer heterogeneity. *Journal of econometrics*, 89(1-2): 57–78, 1998.

Daumé III, H. Bayesian multitask learning with latent hierarchies. *arXiv preprint arXiv:0907.0783*, 2009.

Deng, Z., Kammoun, A., and Thrampoulidis, C. A model of double descent for high-dimensional binary linear classification. *arXiv preprint arXiv:1911.05822*, 2019.

El Karoui, N. et al. On information plus noise kernel random matrices. *The Annals of Statistics*, 38(5):3191–3216, 2010.

Evgeniou, T. and Pontil, M. Regularized multi–task learning. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 109–117. ACM, 2004.

Griffin, G., Holub, A., and Perona, P. Caltech-256 object category dataset. 2007.

Herath, S., Harandi, M., and Porikli, F. Learning an invariant hilbert space for domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3845–3854, 2017.

Hoffman, J., Rodner, E., Donahue, J., Darrell, T., and Saenko, K. Efficient learning of domain-invariant image representations. *arXiv preprint arXiv:1301.3224*, 2013.

Hubert Tsai, Y.-H., Yeh, Y.-R., and Frank Wang, Y.-C. Learning cross-domain landmarks for heterogeneous domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5081–5090, 2016.

Liao, Z. and Couillet, R. A large dimensional analysis of least squares support vector machines. *IEEE Transactions on Signal Processing*, 67(4):1065–1074, 2019.

Mai, X. and Couillet, R. A random matrix analysis and improvement of semi-supervised learning for large dimensional data. *The Journal of Machine Learning Research*, 19(1):3074–3100, 2018.

Moody, G. B. and Mark, R. G. The impact of the mit-bih arrhythmia database. *IEEE Engineering in Medicine and Biology Magazine*, 20(3):45–50, 2001.

Saenko, K., Kulis, B., Fritz, M., and Darrell, T. Adapting visual category models to new domains. In *European conference on computer vision*, pp. 213–226. Springer, 2010.

Xu, S., An, X., Qiao, X., Zhu, L., and Li, L. Multi-output least-squares support vector regression machines. *Pattern Recognition Letters*, 34:1078–1084, 07 2013. doi: 10.1016/j.patrec.2013.01.015.