Consistent Multitask Learning with Nonlinear Output Constraints

Carlo Ciliberto
Department of Computer Science, UCL

joint work w/ Alessandro Rudi, Lorenzo Rosasco and Massi Pontil

Multitask Learning (MTL)

MTL Mantra:
leverage the similarities among multiple learning problems (tasks) to reduce the complexity of the overall learning process.

Prev. Literature:
investigated **linear** tasks relations (more on this in a minute).

This work:
we address the problem of learning multiple tasks that are **nonlinearly** related one to the other

MTL Setting

Given T datasets $S_t = (x_{it}, y_{it})_{i=1}^{n_t}$ learn $\hat{f}_t : \mathcal{X} \to \mathbb{R}$ by solving

$$(\hat{f}_1, \dots, \hat{f}_T) = \underset{f_1,\dots,f_T \in \mathcal{H}}{\operatorname{argmin}} \ \frac{1}{T} \sum_{t=1}^{T} \mathcal{L}(f_t, S_t) + R(f_1, \dots, f_T)$$

- $\mathcal{H}$ space of hypotheses.
- $\mathcal{L}(f_t, S_t) = \frac{1}{n_t} \sum_{i=1}^{n_t} \ell(f_t(x_{it}, y_{it}))$ Data fitting term. Loss $\ell : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ (e.g. leas squares, logistic, hinge, etc.).
- $R(f_1, \dots, f_T)$ a **joint** tasks-structure regularizer

Previous Work: Linear MTL

For example $R(f_1, \ldots, f_T) =$

- Single task learning $\lambda \sum_{t=1}^{T} \|f_t\|_{\mathcal{H}}^2$
- Variance Regularization $\lambda \sum_{t=1}^{T} \|f_t - \bar{f}\|_{\mathcal{H}}^2$ with $\bar{f} = \frac{1}{T} \sum_{t=1}^{T}$
- Clustered tasks $\lambda_1 \sum_{\substack{t \in \mathcal{C}(c) \\ c=1}}^{|\mathcal{C}|} \|f_t - \bar{f}_c\|_{\mathcal{H}}^2 + \lambda_2 \sum_{c=1}^{|\mathcal{C}|} \|\bar{f}_c - \bar{f}\|_{\mathcal{H}}^2$
- Similarity regularizer $\lambda \sum_{t,s}^{T} W_{s,t} \|f_t - f_s\|_{\mathcal{H}}^2 \quad W_{s,t} \geq 0$

Why "Linear"? Because the tasks relations are encoded in a matrix.

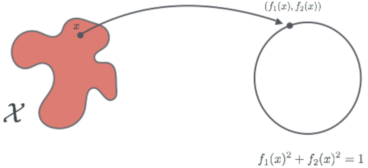
$$R(f_1, \ldots, f_T) = \sum_{t,s=1}^{T} A_{t,s} \langle f_t, f_s \rangle_{\mathcal{H}} \qquad \text{with} \qquad A \in \mathbb{R}^{T \times T}$$

Nonlinear MTL: Setting

What if relations are **nonlinear**? We study the case where tasks satisfy a set of k equations $\gamma(f_1(x), \cdots, f_T(x)) = 0$ identified by $\gamma : \mathbb{R}^T \to \mathbb{R}^k$.

Examples

- Manifold-valued learning
- Physical systems (e.g. robotics)
- Logical constraints (e.g. ranking)



Nonlinear MTL: Setting

NL-MTL Goal: approximate $f^* : \mathcal{X} \to \mathcal{C}$ minimizer the **Expected Risk**

$$\min_{f:\mathcal{X}\to\mathcal{C}} \ \mathcal{E}(f), \qquad\qquad \mathcal{E}(f) = \frac{1}{T}\int \ell(f_t(x), y)\ d\rho_t(x, y)$$

where

- $f : \mathcal{X} \to \mathcal{C}$ is such that $f(x) = (f_1(x), \dots, f_T(x))$ for all $x \in \mathcal{X}$.
- $\mathcal{C} = \{c \in \mathbb{R}^T \mid \gamma(c) = 0\}$ is the **constraints set** induced by γ.
- $\rho_t(x, y) = \rho_t(y|x)\rho_{\mathcal{X}}(x)$ is the *unknown* data distribution.

Nonlinear MTL: Challenges

Why not try **Empirical Risk Minimization**?

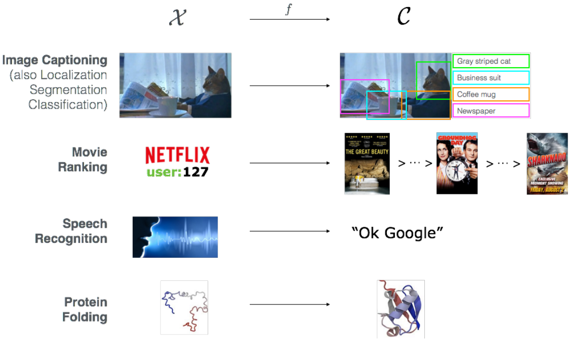
$$\hat{f} = \underset{\substack{\mathcal{H} \subset \{f:\mathcal{X} \to \mathcal{C}\} \\ f \in \mathcal{H}}}{\operatorname{argmin}} \frac{1}{T} \sum_{t=1}^{T} \mathcal{L}(f_t, S_t)$$

Problems:

- **Modeling**: $f_1, f_2 : \mathcal{X} \to \mathcal{C}$ does not guarantee $f_1 + f_2 : \mathcal{X} \to \mathcal{C}$. $\mathcal{H}$ not a linear space. How to choose a "good" $\mathcal{H}$ in practice?
- **Computations**: Hard (non-convex) optimization. How to solve it?
- **Statistics**: How to study the generalization properties of $\hat{f}$?

Nonliner MTL: a Structured Prediction Perspective

Idea: formulate NL-MTL as a **structured prediction** problem.



Structured Prediction: originally designed for discrete outputs, but recently genearlized to any set $\mathcal{C}$ within the **SELF** framework [Ciliberto et al. 2016].

Nonlinear MTL Estimator

We propose to approximate f^* via the estimator $\hat{f} : \mathcal{X} \to \mathcal{C}$ such that

$$\hat{f}(x) = \operatorname*{argmin}_{c \in \mathcal{C}} \frac{1}{T} \sum_{t=1}^{T} \sum_{i=1}^{n_t} \alpha_{it}(x) \ell(c_t, y_{it})$$

where the weights are obtained in closed form as

$$(\alpha_{i1}(x), \cdots, \alpha_{in_t}(x)) = (K_t + \lambda I)^{-1} v_t(x)$$

with K_t the kernel matrix $(K_t)_{ij} = k(x_{it}, x_{jt})$ of t-th dataset and $v_t(x) \in \mathbb{R}^n$ with $v_t(x)_i = k(x_{it}, x)$. $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ a **kernel**.

Note. evaluating $\hat{f}(x)$ requires solving an optimization over $\mathcal{C}$ (e.g. for ℓ least squares it $\hat{f}$ reduces to a projection onto $\mathcal{C}$).

Theoretical Results

Thm. 1 (Universal Consistency)

$$\mathcal{E}(\hat{f}) - \mathcal{E}(f^*) \to 0 \qquad \text{with probability } \ 1.$$

Thm. 2 (Rates). Let $n = n_t$ and $g_t^* \in \mathcal{G}$ for all $t = 1, \ldots, T$. Then

$$\mathcal{E}(\hat{f}) - \mathcal{E}(f^*) \leq O(n^{-1/4}) \qquad \text{with high probability}$$

Thm. 3 (Benefits of MTL). Let $\mathcal{C} \subset \mathbb{R}^T$ radius 1 sphere. Let $N = nT$.

$$\text{Then} \qquad \mathcal{E}(\hat{f}) - \mathcal{E}(f^*) \leq O(N^{-1/2}) \qquad \text{with high probability}$$

Intuition

Ok... but how did we get there?

Structure Encoding Loss Function (SELF)

Ciliberto et al. 2016

Def. $\ell : \mathcal{C} \times \mathcal{Y} \to \mathbb{R}$ is a *structure encoding loss function (SELF)* if there exist $\mathcal{H}$ Hilbert space and $\psi : \mathcal{C} \to \mathcal{H}$, $\varphi : \mathcal{Y} \to \mathcal{H}$ such that

$$\ell(c, y) = \langle \psi(c), \varphi(y) \rangle_{\mathcal{H}} \qquad \forall c \in \mathcal{C}, \;\; \forall y \in \mathcal{Y}.$$

Abstract definition... BUT "most" loss functions used in MTL settings are SELF! More precisely any Lipschitz continuous function differentiable almost everywhere (e.g. least squares, logistic, hinge).

Nonlinear MTL + SELF

Minimizer of the expected risk

$$f^*(x) = \operatorname*{argmin}_{c \in \mathcal{C}} \frac{1}{T} \sum_{t=1}^{T} \int \ell(c_t, y) \rho_t(y|x)$$

Nonlinear MTL + SELF

Minimizer of the expected risk

$$f^*(x) = \underset{c \in \mathcal{C}}{\operatorname{argmin}} \frac{1}{T} \sum_{t=1}^{T} \int \langle \psi(c_t), \varphi(y) \rangle_{\mathcal{H}} \rho_t(y|x)$$

Nonlinear MTL + SELF

Minimizer of the expected risk

$$f^*(x) = \operatorname*{argmin}_{c \in \mathcal{C}} \frac{1}{T} \sum_{t=1}^{T} \left\langle \psi(c_t), \int \varphi(y) \rho_t(y|x) \right\rangle_{\mathcal{H}}$$

Nonlinear MTL + SELF

Minimizer of the expected risk

$$f^*(x) = \operatorname*{argmin}_{c \in \mathcal{C}} \frac{1}{T} \sum_{t=1}^{T} \langle \psi(c_t), \boldsymbol{g}_t^*(x) \rangle_{\mathcal{H}}$$

where $g_t^* : \mathcal{X} \to \mathcal{H}$ is such that $g_t^*(x) = \int \varphi(y) \rho_t(y|x)$.

Nonlinear MTL Estimator

Idea, learn a $\hat{g}_t : \mathcal{X} \to \mathcal{H}$ for each g_t^*. Then approximate

$$f^*(x) = \operatorname*{argmin}_{c \in \mathcal{C}} \frac{1}{T} \sum_{t=1}^{T} \langle \psi(c_t), \boldsymbol{g}_t^*(x) \rangle_{\mathcal{H}}$$

with $\hat{f} : \mathcal{X} \to \mathcal{C}$

$$\hat{f}(x) = \operatorname*{argmin}_{c \in \mathcal{C}} \frac{1}{T} \sum_{t=1}^{T} \langle \psi(c_t), \boldsymbol{\hat{g}_t(x)} \rangle_{\mathcal{H}}$$

Nonlinear MTL Estimator

This work: learn $\hat{g}_t$ via kernel ridge regression. Let $\mathcal{G}$[1] be a reproducing kernel Hilbert space with kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.

$$\hat{g}_t = \operatorname*{argmin}_{g \in \mathcal{G}} \; \frac{1}{n_t} \sum_{i=1}^{n_t} \| g(x_{it}) - \varphi(y_{it}) \|_{\mathcal{H}}^2 + \lambda \| g \|_{\mathcal{G}}^2$$

Then

$$\hat{g}_t(x) = \sum_{i=1}^{n_t} \alpha_{it}(x) \varphi(y_{it}) \qquad (\alpha_{i1}(x), \cdots, \alpha_{in_t}(x)) = (K_t + \lambda I)^{-1} v_t(x)$$

where K_t kernel matrix of t-th dataset, $v_t(x) \in \mathbb{R}^n$ evaluation vector $v_t(x)_i = k(x_{it}, x)$.

[1] actually $\mathcal{G} \otimes \mathcal{H}$

Nonlinear MTL Estimator

Plugging into

$$\hat{f}(x) = \operatorname*{argmin}_{c \in \mathcal{C}} \frac{1}{T} \sum_{t=1}^{T} \langle \psi(c_t), \hat{g}_t(x) \rangle_{\mathcal{H}}$$

by the SELF property we have

$$\hat{f}(x) = \operatorname*{argmin}_{c \in \mathcal{C}} \frac{1}{T} \sum_{t=1}^{T} \sum_{i=1}^{n_t} \alpha_{it}(x) \left\langle \psi(c_t), \sum_{i=1}^{n_t} \alpha_{it}(x) \varphi(y_{it}) \right\rangle_{\mathcal{H}}$$

Nonlinear MTL Estimator

Plugging into

$$\hat{f}(x) = \operatorname*{argmin}_{c \in \mathcal{C}} \frac{1}{T} \sum_{t=1}^{T} \langle \psi(c_t), \hat{g}_t(x) \rangle_{\mathcal{H}}$$

by the SELF property we have

$$\hat{f}(x) = \operatorname*{argmin}_{c \in \mathcal{C}} \frac{1}{T} \sum_{t=1}^{T} \sum_{i=1}^{n_t} \alpha_{it}(x) \langle \psi(c_t), \varphi(y_{it}) \rangle_{\mathcal{H}}$$

Nonlinear MTL Estimator

Plugging into

$$\hat{f}(x) = \operatorname*{argmin}_{c \in \mathcal{C}} \frac{1}{T} \sum_{t=1}^{T} \langle \psi(c_t), \hat{g}_t(x) \rangle_{\mathcal{H}}$$

by the SELF property we have

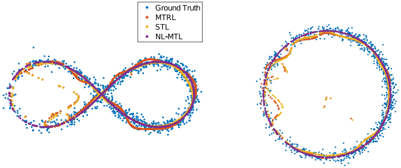
$$\hat{f}(x) = \operatorname*{argmin}_{c \in \mathcal{C}} \frac{1}{T} \sum_{t=1}^{T} \sum_{i=1}^{n_t} \alpha_{it}(x) \ell(c_t, y_{it})$$

as desired.

Note that evaluating $\hat{f}(x)$ Does not require knowledge of $\mathcal{H}$, φ or ψ!

Empirical Results

Synthetic data



Inverse dynamics (Sarcos)

	STL	MTL[36]	CMTL[10]	MTRL[11]	MTFL[13]	FMTL[16]	NL-MTL[R]	NL-MTL[P]
Expl. Var. (%)	40.5 ±7.6	34.5 ±10.2	33.0 ±13.4	41.6 ±7.1	49.9 ±6.3	50.3 ±5.8	**55.4 ±6.5**	54.6 ±5.1

Logic constraints (Ranking Movielens100k)

	NL-MTL	SELF[21]	Linear [37]	Hinge [38]	Logistic [39]	SVMStruct [20]	STL	MTRL[11]
Rank Loss	**0.271 ±0.004**	0.396 ±0.003	0.430 ±0.004	0.432 ±0.008	0.432 ±0.012	0.451 ±0.008	0.581 0.003	0.613 ±0.005