Multi-Task Learning and Matrix Regularization

Andreas Argyriou

TTI Chicago

Outline

- Multi-task learning and related problems
- Multi-task feature learning (trace norm, Schatten L_p norms, non-convex regularizers)
- Representer theorems; “kernelization”

Multi-Task Learning

- Tasks $t = 1, \ldots, n$
- m examples per task are given: $(x_{t1}, y_{t1}), \ldots, (x_{tm}, y_{tm}) \in \mathcal{X} \times \mathcal{Y}$

  (simplification: sample sizes need not be equal; subsumes case of common input data)
- Predict using functions $f_t : \mathcal{X} \to \mathcal{Y}\,,\ t = 1, \ldots, n$
- When the tasks are related, learning the tasks *jointly* should perform better than learning each task *independently*
- Especially important when *few data points* are available *per task* (small m); in such cases, independent learning is not successful

Transfer

- Want good generalization on the n given tasks but also on new tasks (*transfer learning*)

- Given *a few examples* from a new task t', $\{(x_{t'1}, y_{t'1}), \ldots, (x_{t'\ell}, y_{t'\ell})\}$, want to learn $f_{t'}$

- Do this by "transferring" the common task structure / features learned from the n tasks

- Transfer is an important feature of human intelligence

Multi-Task Applications

- Marketing databases, collaborative filtering, recommendation systems (e.g. Netflix); task = product preferences for each person

Description				
Closure	Type of winery	Type of wine	Price	Your rating
Metacork	International	Blush red	$25	
Metacork	Mid-sized regional	Dry white	$20	
Traditional cork	Small boutique	Dry red	$20	
Screwcap	International	Dry red	$30	
Metacork	Small boutique	Aromatic white	$30	
Traditional cork	International	Dry white	$15	
Screwcap	Large national	Blush red	$20	
Synthetic cork	International	Aromatic white	$20	

Matrix Completion

- Matrix completion

$$\begin{aligned} &\underset{W\in\mathbb{R}^{d\times n}}{\text{minimize}} \quad \text{rank}(W) \\ &\text{s.t. } w_{ij} = y_{ij}, \ \forall\,(i,j)\in E \end{aligned}$$

- Special case of multi-task learning (input vectors are elements of the canonical basis)

- Each column of the matrix corresponds to the regression vector for a task; emphasis is on recovery of the matrix; in learning we are also interested in generalization

Related Problems

- Domain adaptation / transfer
- Multi-view learning
- Multi-label learning
- Multi-task learning is a *broad problem*; no single method can solve everything;

Learning Multiple Tasks with a Common Kernel

- Learn a common kernel $K(x,x') = \langle x, Dx' \rangle$ from a *convex* set of kernels:

$$\inf_{\substack{w_1,\dots,w_n \in \mathbb{R}^d \\ D \succ 0,\ \operatorname{tr}(D) \le 1}} \sum_{t=1}^{n} \sum_{i=1}^{m} E\left(\langle w_t, x_{ti} \rangle, y_{ti}\right) \quad + \quad \gamma \quad \operatorname{tr}(W^\top D^{-1} W) \qquad (\mathcal{MTL})$$

$$\uparrow$$

$$\sum_{t=1}^{n} \langle w_t, D^{-1} w_t \rangle$$

where $W = \begin{pmatrix} | & & | \\ w_1 & \dots & w_n \\ | & & | \end{pmatrix}$

Learning Multiple Tasks with a Common Kernel

- *Jointly convex* problem in (W, D)

- The choice of constraint $\mathrm{tr}(D) \leq 1$ is important; intuitively, penalizes the number of common features (eigenvectors of D)

- Once we have learned $\hat{D}$, we can *transfer* it to learning of a new task t'

$$\min_{w \in \mathbb{R}^d} \sum_{i=1}^{m} E\left(\langle w, x_{t'i} \rangle, y_{t'i}\right) + \gamma \left\langle w, \hat{D}^{-1} w \right\rangle$$

Alternating Minimization Algorithm

- Alternating minimization over W and D

Initialization: given initial D, e.g. $D = \frac{I_d}{d}$

while convergence condition is not true **do**

for $t = 1, \ldots, n$ learn w_t *independently* by minimizing

$$\sum_{i=1}^{m} E(\langle w, x_{ti} \rangle, y_{ti}) + \gamma \langle w, D^{-1} w \rangle$$

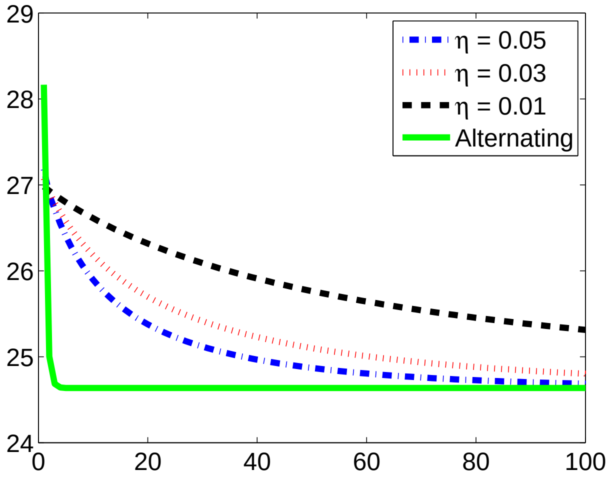
end for

set $D = \frac{(WW^\top)^{\frac{1}{2}}}{\mathrm{tr}(WW^\top)^{\frac{1}{2}}}$

end while

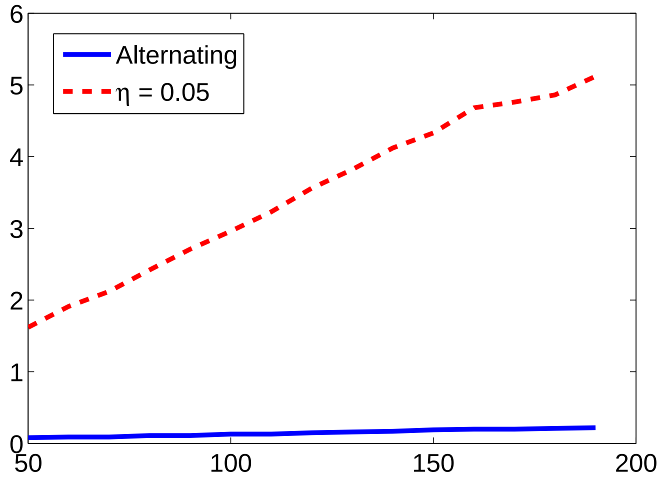
Alternating Minimization (contd.)

objective function



#iterations
(green = alternating)

seconds



#tasks
(blue = alternating)

- Compare computational cost with a gradient descent on W only ($\eta :=$ learning rate)

Alternating Minimization (contd.)

- Small number of iterations (typically fewer than 50 in experiments)

- Alternative algorithms: singular value thresholding [*Cai et al. 2008*], Bregman-type gradient descent [*Ma et al. 2009*] etc.

- Non-SVD alternatives like [*Rennie & Srebro 2005*, *Maurer 2007*] or SOCP methods [*Srebro et al. 2005, Liu and Vandenberghe 2008*]

Trace Norm Regularization

Problem $(\mathcal{MTL})$ is equivalent to

$$\min_{W \in \mathbb{R}^{d \times n}} \sum_{t=1}^{n} \sum_{i=1}^{m} E(\langle w_t, x_{ti} \rangle, y_{ti}) + \gamma \, \|W\|_{tr}^2 \qquad (\mathcal{TR})$$

The *trace norm* (or nuclear norm) $\|W\|_{tr}$ is the sum of the singular values of W

$$W = U \Sigma V^\top$$

$$\|W\|_{tr} = \sum_i \sigma_i(W)$$

Trace Norm vs. Rank

- Problem $(\mathcal{TR})$ is a convex relaxation of the problem

$$\min_{W \in \mathbb{R}^{d \times n}} \sum_{t=1}^{n} \sum_{i=1}^{m} E(\langle w_t, x_{ti} \rangle, y_{ti}) + \gamma \operatorname{rank}(W)$$

- NP-hard problem
- Rank and trace norm correspond to L_0, L_1 on the vector of singular values
- Hence one (qualified) interpretation: we want the task parameter vectors w_t to lie on a *low dimensional* subspace

Machine Learning Interpretations

- Learning a common *linear kernel* for all tasks (discussed already)

- Maximum likelihood (learning a Gaussian covariance with fixed trace)

- Matrix factorization

$$\|W\|_{tr} = \frac{1}{2} \min_{F^\top G = W} (\|F\|_{Fr}^2 + \|G\|_{Fr}^2)$$

- MAP in a graphical model (as above)

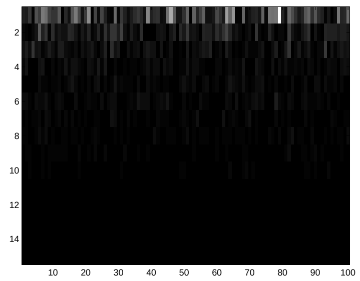
- Gaussian process interpretation

“Rotation invariant” Group Lasso

- Problem $(\mathcal{MTL})$ is equivalent to

$$\min_{\substack{A \in \mathbb{R}^{d \times n} \\ U \in \mathbb{R}^{d \times d},\, U^\top U = I}} \sum_{t=1}^{n} \sum_{i=1}^{m} E(\langle a_t, U^\top x_{ti} \rangle, y_{ti}) + \gamma \, \|A\|_{2,1}^2$$

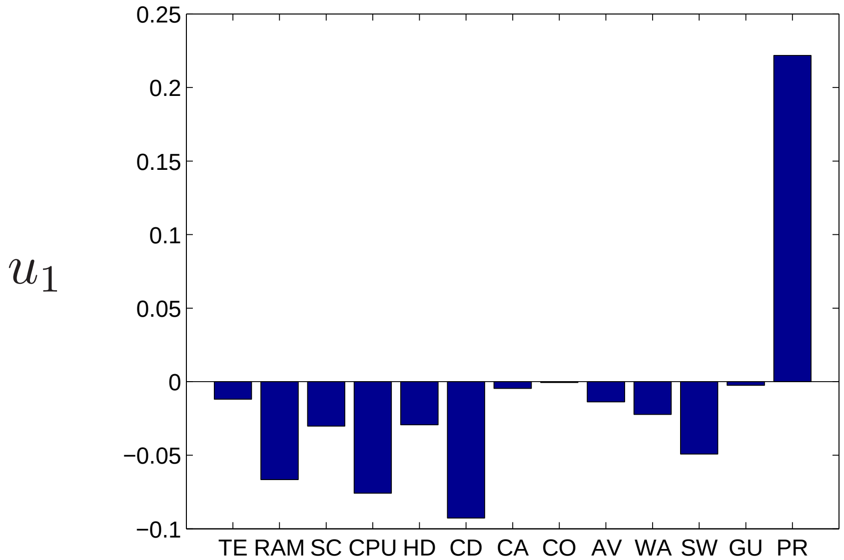
where $\|A\|_{2,1} := \sum_{i=1}^{d} \sqrt{\sum_{t=1}^{n} a_{it}^2}$



Experiment (Computer Survey)

- Consumers' ratings of products [Lenk et al. 1996]
- 180 persons (tasks)
- 8 PC models (training examples)
- 13 binary input variables (RAM, CPU, price etc.) $+$ bias term
- Integer output in $\{0, \ldots, 10\}$ (likelihood of purchase)
- The square loss was used

Experiment (Computer Survey)



Method	RMSE
Alternating Alg.	1.93
Hierarchical Bayes [Lenk et al.]	1.90
Independent	3.88
Aggregate	2.35
Group Lasso	2.01

- The most important feature (eigenvector of D) weighs *technical characteristics* (RAM, CPU, CD-ROM) vs. *price*

Generalizations: Spectral Regularization

- Generalize $(\mathcal{MTL})$:

$$\inf_{W \in \mathbb{R}^{d \times n}} \sum_{t=1}^{n} \sum_{i=1}^{m} E\left(\langle w_t, x_{ti} \rangle, y_{ti}\right) + \gamma \, \|W\|_p^2$$

where $\|W\|_p$ is the Schatten L_p norm of the singular values of W

- $L_1 - L_2$ trade-off

- Can be generalized to a family of spectral functions

- A similar alternating algorithm can be used

Generalizations: Learning Groups of Tasks

- Assume *heterogeneous* environment, i.e. K low dimensional subspaces
- Learn a partition of tasks in K groups

$$\inf_{\substack{D_1,\ldots,D_K \succ 0 \\ \operatorname{tr}(D_k)\leq 1}} \sum_{t=1}^{n} \min_{k=1}^{K} \min_{w_t \in \mathbb{R}^d} \left\{ \sum_{i=1}^{m} E\left(\langle w_t, x_{ti}\rangle, y_{ti}\right) + \gamma \langle w_t, D_k^{-1} w_t \rangle \right\}$$

- The representation learned is $(\hat{D}_1, \ldots, \hat{D}_K)$; we can transfer this representation to easily learn a new task
- *Non-convex* problem; we use stochastic gradient descent

Nonlinear Kernels

- An important note: all methods presented satisfy a *multi-task representer theorem* (a type of necessary optimality condition)
- This fact enables “kernelization”, i.e. we may use a given kernel (e.g. polynomial, RBF) via its Gram matrix
- We now expand on this observation

Representer Theorems

- Consider any learning problem of the form

$$\min_{w \in \mathbb{R}^d} \sum_{i=1}^{m} E\left(\langle w, x_i \rangle, y_i\right) + \Omega(w)$$

- This problem can be "kernelized" if Ω satisfies the "classical" rep. thm.

$$\hat{w} = \sum_{i=1}^{m} c_i x_i$$

(a necessary but not sufficient optimality condition)

Representer Theorems (contd.)

Theorem. *The "classical" rep. thm. for* single-task learning, *holds if and only if there exists a nondecreasing function* $h : \mathbb{R}_+ \to \mathbb{R}$ *such that*

$$\Omega(w) = h(\langle w, w \rangle) \qquad \forall\, w \in \mathbb{R}^d$$

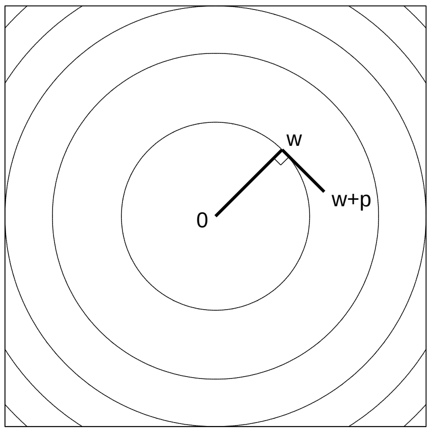
(under differentiability assumptions)

- Sufficiency of the condition was known [*Kimeldorf & Wahba, 1970, Schölkopf et al., 2001* etc.]

Representer Theorems (contd.)

- Sketch of the proof: equivalent condition is

$$\Omega(w+p) \geq \Omega(w) \qquad \text{for all } w, p \text{ such that } \langle w, p \rangle = 0.$$



Multi-Task Representer Theorems

- Our multi-task formulations satisfy a *multi-task representer theorem*

$$\hat{w}_t = \sum_{s=1}^{n} \sum_{i=1}^{m} c_{si}^{(t)} x_{si} \qquad \forall\, t \in \{1, \dots, n\} \qquad (\mathcal{R}.\mathcal{T}.)$$

- *All tasks* are involved in this expression (unlike the single-task representer theorem $\Leftrightarrow$ Frobenius norm regularization)

- Generally, consider any matrix optimization problem of the form

$$\min_{w_1,\dots,w_n \in \mathbb{R}^d} \sum_{t=1}^{n} \sum_{i=1}^{m} E\left(\langle w_t, x_{ti}\rangle, y_{ti}\right) + \Omega(W)$$

Multi-Task Representer Theorems (contd.)

- **Definitions:**

  $\mathbf{S}^n_+$ = the positive semidefinite cone
  The function $h : \mathbf{S}^n_+ \to \mathbb{R}$ is matrix nondecreasing, if

$$h(A) \le h(B) \qquad \forall\ A, B \in \mathbf{S}^n_+ \quad \text{s.t. } A \preceq B$$

Theorem. *Rep. thm. ($\mathcal{R}.\mathcal{T}.$) holds if and only if there exists a matrix nondecreasing function $h : \mathbf{S}^n_+ \to \mathbb{R}$ such that*

$$\Omega(W) = h(W^\top W) \qquad \forall\ W \in \mathbb{R}^{d\times n}$$

(under differentiability assumptions)

Implications

- The theorem tells us when a matrix learning problem can be "kernelized"

- In single-task learning, the choice of h does not matter essentially

- However, in multi-task learning, the choice of h is important (since $\preceq$ is a partial ordering)

- Many valid regularizers: Schatten L_p norms $\|\cdot\|_p$, rank, orthogonally invariant norms, norms of type $W \mapsto \|WM\|_p$ etc.

Refinements of the MTL Representer Theorem

- Write $(\mathcal{R}.\mathcal{T}.)$ in matrix notation

$$\hat{W} = XC$$

where

$$X = \left(\cdots \quad \begin{matrix} | \\ x_{si} \\ | \end{matrix} \quad \cdots \right)_{s=1}^{n} {}_{i=1}^{m}$$

includes all the input data (for all the tasks)

- $\{$Total sample size$\} \times n$ variables to learn

- How does it relate to "per task" representations of the form

$$\left(\ldots \quad X_s \alpha_s \quad \ldots \right)_{s=1}^{n}$$

Refinements of the MTL Representer Theorem (contd.)

Theorem.

$$\hat{W} = \begin{pmatrix} \ldots & X_s \alpha_s & \ldots \end{pmatrix}_{s=1}^{n} \; R$$

for some positive semidefinite matrix R *and some* α_s

- The input sample for task s appears with *the same coefficients* α_s across all tasks, up to normalization

- Intuitively, the dependences among tasks may vary; but the input sample for each task is like a "module"

- Equivalently, C consists of blocks of rank one matrices

Refinements of the MTL Representer Theorem (contd.)

- Only $\{\text{total sample size}\} + \frac{1}{2}(n^2 + n)$ variables are needed
- This holds for all Schatten L_p norms except the spectral norm (for which one may choose one such solution from an infinite set)
- It also holds for a more general family of orthogonally invariant norms

Conclusion

- Multi-task learning is ubiquitous; exploiting task relatedness can enhance learning performance significantly

- Multi-task learning by learning a common linear kernel

- Gives rise to regularization with the *trace norm*, *spectral norms* and *non-convex* regularizers

- Necessary and sufficient conditions for *representer theorems* (in both the multi-task and single-task setting); implies kernelization of many multi-task methods