A Spectral Regularization Framework for Multi-Task Structure Learning

Massimiliano Pontil

Department of Computer Science
University College London

(Joint work with A. Argyriou, T. Evgeniou, C.A. Micchelli, and Y. Ying)
Learning Multiple Tasks Simultaneously

- By a task we mean a real-valued function (for regression / classification)

- Learning multiple related tasks vs. learning independently

- Few data per task; pooling data across related tasks

  Example 1: predict users’ preferences to products

  Example 2: object detection in computer vision
Learning Paradigm

• Tasks index: \( t = 1, \ldots, T \)

• \( m \) examples per task: \((x_{t1}, y_{t1}), \ldots, (x_{tm}, y_{tm}) \in \mathbb{R}^d \times \mathbb{R}\)

• Predict using functions \( f_t(x) = w_t^\top x \)

• Goal is to learn parameters \( w_t \) and their “structure”

• Use the notation

\[
W = \begin{pmatrix}
\vdots \\
\vdots \\
\vdots
\end{pmatrix}
= \begin{pmatrix}
-w_1 \\
\vdots \\
-w^d
\end{pmatrix}
= \begin{pmatrix}
w_1 \\
\vdots \\
w_T
\end{pmatrix}
\]
Approach

• Learn each task by ridge regression:

\[
\min_{w \in \mathbb{R}^d} \sum_{i=1}^{m} (w^\top x_{ti} - y_{ti})^2 + \gamma w^\top D^{-1}w, \quad \gamma > 0
\]

• Further minimize over structure matrix \( D \):

\[
\min_{D \in \mathcal{D}} \sum_{t=1}^{T} \left( \min_{w \in \mathbb{R}^d} \sum_{i=1}^{m} (w^\top x_{ti} - y_{ti})^2 + \gamma w^\top D^{-1}w \right)
\]

• \( \mathcal{D} \): subset of positive definite matrices with bounded trace
Matrix Regularization Problem

Rewrite above problem as a matrix regularization one:

Minimize \[ \text{Error}(W) + \gamma \text{tr}(W^\top D^{-1}W) \] (*)

where \[ \text{Error}(W) = \sum_{t=1}^{T} \sum_{i=1}^{m} (w_t^\top x_{ti} - y_{ti})^2 \]
Diagonal Case

If $D = \Lambda = \text{diag}(\lambda_1, \ldots, \lambda_d)$, minimizing over $\lambda$ leads to

$$\text{Minimize} \quad \text{Error}(W) + \gamma \|W\|_2^2 \quad (\diamond)$$

where $\|W\|_{2,1} := \sum_{i=1}^{d} \|w^i\|_2$

- Related to methods for learning the kernel and the group lasso / cosso methods used in statistics
- Method selects important variables shared by the tasks
Effect of $(2, 1)$-Norm

- Compare matrices favored by different norms:

$$\|W\|_{1,1}$$  (Vector 1-norm)  
Sparsity

$$\|W\|_{2,2}$$  (Frobenius norm)  
Uniformity

$$\|W\|_{2,1}$$  (Mixed norm)  
Structured sparsity
Degree of regularization

If $\hat{W}$ solves $\diamond$, we say that variable $i$ is relevant if $\|\hat{w}^i\|_2 > 0$

- The number of relevant variables typically decreases with $\gamma$
General Case : Interpretation

If $\mathcal{D} = \{D > 0, \text{tr}(D) \leq 1\}$: write $D = U\Lambda U^\top$, with $U$ orthogonal, and set $A = U^\top W$

Again, minimizing over $\Lambda$ problem $(\ast)$ reduces to

$$\text{Minimize}_{A, U^\top U = I} \quad \text{Error}(UA) + \gamma \| A \|_{2,1}^2$$

- The number of non-zero rows of $A$ typically decreases with $\gamma$
- Interpretation: learn a small set of common features shared by the tasks
- Non-convex problem!
If we minimize over $D$, problem $(\ast)$ reduces to

$$\text{Minimize } W \quad \text{Error}(W) + \gamma \|W\|_1^2$$

where $\|W\|_1$ is the sum of singular values of $W$ (trace norm of $W$)

- Interpretation: tasks $w_t$ all lie in a low dimensional space

Let us explain this connection in a more general context...
Spectral Regularization (I)

Let $F$ be a spectral matrix function:

$$F(U \Lambda U^T) = U \text{diag}[f(\lambda_1), \ldots, f(\lambda_d)] U^T$$

where $f : [0, \infty) \to [0, \infty)$, and consider the problem

$$\text{Minimize} \quad \text{Error}(W) + \gamma \text{tr}(W^T F(D)W)$$

subject to $W \in \mathbb{R}^{d \times T}, D \in \mathcal{D}$

- Previous case corresponds to $f(\lambda) = \lambda^{-1}$
Spectral Regularization (II)

\[ \Omega_f(W) = \inf_{D \in D} \text{tr}(W^T F(D) W) \]

This is always a spectral function of the covariance matrix \(WW^T\)

In particular, if \(f(\lambda) = \lambda^{1 - \frac{2}{p}}, p \in (0, 2]\), we have

\[ \Omega_f(W) = \|W\|_p^2 \]

where \(\|W\|_p\) is the \(L_p\) pre-norm of the singular values of \(W\)

Moreover, the infimizer is \(D = \frac{(WW^T)^{\frac{p}{2}}}{\text{tr}(WW^T)^{\frac{p}{2}}}\)
In summary, the following problems are equivalent

\[
\begin{align*}
\text{Minimize } & \quad \text{Error}(W) + \gamma \text{tr}(WD^{1-\frac{2}{p}}W) \quad (1) \\
\text{Minimize } & \quad \text{Error}(W) + \gamma \|W\|_p^2 \quad (2) \\
\text{Minimize } & \quad \text{Error}(UA) + \gamma \|A\|_{2,p}^2 \quad (3)
\end{align*}
\]

- (1) is our original proposal and is jointly convex (next slide)
- (2) is also convex but more difficult to solve (later...)
- (3) helps us gain intuition on our proposal but is non convex
Condition for Joint Convexity

**Theorem:** The regularizer $\text{tr}(W^T F(D) W)$ is jointly convex if and only if $\frac{1}{f}$ is matrix concave of order $d$, that is,

$$
\mu \left( \frac{1}{F} \right) (A) + (1 - \mu) \left( \frac{1}{F} \right) (B) \preceq \left( \frac{1}{F} \right) (\mu A + (1 - \mu) B)
$$

for all $A, B \succ 0$ and $\mu \in [0, 1]$

- Condition is verified for $F(D) = D^{1-\frac{2}{p}}$, $p \in [1, 2]$
Alternate Minimization Algorithm

- Alternating minimization over $W$ (supervised learning) and $D$ (unsupervised “correlation” of tasks).

Initialization: set $D = \frac{I_{d \times d}}{d}$

while convergence condition is not true do
  for $t = 1, \ldots, T$, learn $w_t$ independently by minimizing
  $$\sum_{i=1}^{m} (w^\top x_{ti} - y_{ti})^2 + \gamma w_t^\top D^{1-\frac{2}{p}} w_t$$
  end for
  set $D = \frac{(WW^\top)^{\frac{p}{2}}}{\text{tr}(WW^\top)^{\frac{p}{2}}}$
end while
Additional Observations

- Under the above condition on $F$, the alternating algorithm (with some perturbation) converges to the optimal solution.

- The optimal $D$ is a spectral function of the tasks’ covariance $WW^T$.

- The eigenvectors of $D$ are the features $U$ solving problem (3).

- Minimization wrt. $D$ is a vector problem; it may have a closed form solution (e.g. $L_p$ case) or may be done using Lagrange multipliers.

- We may also use a nonlinear feature map in place of the input $x$ (via a prescribed reproducing kernel Hilbert space).
• Compare computational cost of alternating minimization vs. gradient descent (on problem (2)), for $p = 1.5$

• Curves for different learning rates are shown
Computational Cost (contd.)

• Alternating algorithm typically needed less than 30 iterations to converge.

• At least an order of magnitude faster than gradient descent

• Cost per iteration is smaller for gradient descent but the number of iterations is at least an order of magnitude larger

• Scales better than gradient descent with the number of tasks

• Both methods require SVD (costly if $d$ is large)
Experiment 1 (Computer Survey)

- Consumers’ ratings of products [Lenk et al. 1996]
- 180 persons (tasks)
- 8 PC models (training examples); 4 PC models (test 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 1 (Computer Survey)

<table>
<thead>
<tr>
<th>Method</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p = 2$</td>
<td>3.88</td>
</tr>
<tr>
<td>$p = 1$</td>
<td>1.93</td>
</tr>
<tr>
<td>$p = 0.7$</td>
<td>1.86</td>
</tr>
<tr>
<td>Hierarchical Bayes [Lenk et al.]</td>
<td>1.90</td>
</tr>
</tbody>
</table>

- Performance using $L_p$ regularizers
- Trace norm ($p = 1$) is best among the norms
- A non-convex regularizer ($p < 1$) does even better
- method improves on hierarchical Bayes (which also learns a matrix $D$ using Bayesian inference but with more elaborate priors)
Experiment 1 (Computer Survey)

- Performance improves with more tasks (for independent tasks, error = 16.53)
- A single most important feature shared by all persons
The most important feature weighs \textit{technical characteristics} (RAM, CPU, CD-ROM) vs. \textit{price}. 

\begin{itemize}
\item \textit{Experiment 1 (Computer Survey)}
\item \textbf{$u_1$}
\item \textbf{Diagram:}
\item \text{The most important feature weighs technical characteristics (RAM, CPU, CD-ROM) vs. price}
\end{itemize}
Experiment 2 (School Data)

- Examination scores of 15362 students from 139 schools in London

- 139 tasks (student performance in each school); varying numbers of examples per task (students per school)

- Variables: year of the examination, 4 school-specific and 3 student-specific + bias term

- Generated 10 random splits of the data; 75% training examples per school and 25% test examples

- As in [Bakker & Heskes], the performance measure is explained variance \( 1 - \frac{\text{mean squared test error}}{\text{variance of test data}} \) averaged over tasks

- The square loss was used
Experiment 2 (School Data)

- Explained variance

<table>
<thead>
<tr>
<th>Method</th>
<th>Expl. variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p = 2$</td>
<td>$23.5 \pm 2.0%$</td>
</tr>
<tr>
<td>$p = 1$</td>
<td>$26.7 \pm 2.0%$</td>
</tr>
<tr>
<td>Hierarchical Bayes</td>
<td>$29.5 \pm 0.4%$</td>
</tr>
</tbody>
</table>

- Trace norm ($p = 1$) almost performs the best

- Alternating algorithm is competitive, though not easy to compare (regularizer is simpler than HB, the data splits are different)
Transfer Learning Experiments

<table>
<thead>
<tr>
<th>Computer Survey</th>
<th>School Data</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Tasks’ subset</strong></td>
<td><strong>Expl. variance</strong></td>
</tr>
<tr>
<td>150</td>
<td>110</td>
</tr>
<tr>
<td>30 (with transferred $D$)</td>
<td>24.8 %</td>
</tr>
<tr>
<td>30 (with raw $D$)</td>
<td>19.2 %</td>
</tr>
<tr>
<td>29 (with transferred $D$)</td>
<td>13.9 %</td>
</tr>
<tr>
<td>29 (with raw $D$)</td>
<td></td>
</tr>
</tbody>
</table>

- Trained our method with $p = 1$ on 150 tasks; then used the learned $D$ for training 30 ridge regressions on the remaining tasks.

- Alternatively, trained 30 ridge regressions using raw $D = \frac{I}{d}$

- Transferring the learned $D$ improves on the raw data representation and is not much worse than MTL on many tasks.
Summary

• Spectral regularization framework for learning the structure shared by many supervised tasks

• Structure is summarized by positive definite matrix $D$ which is a spectral function of the tasks’ covariance matrix

• Derived a necessary and sufficient condition for convexity of the problem

• $W$ and $D$ can be efficiently computed using an alternating minimization algorithm

• Good statistical performance on two real data sets; learned structure can be transferred to new tasks
References


[B. Bakker and T. Heskes. Task clustering and gating for Bayesian multi–task learning. JMLR 2003]


References


[T. Evgeniou, C.A. Micchelli and M. Pontil. Learning multiple tasks with kernel methods. JMLR 2005]


[A. Maurer. Bounds for linear multi-task learning. JMLR 2006]

[C.A. Micchelli and M. Pontil. Learning the kernel function via regularization. JMLR 2005]

[R. Raina, A. Y. Ng and D. Koller. Constructing informative priors using transfer learning. ICML 2006]

[N. Srebro, J.D.M. Rennie and T.S. Jaakkola. Maximum-margin matrix factorization. NIPS 2004]