Multi-Task Kernel Learning

J. Saketha Nath

CSE, IIT-Bombay

Multi-Task Learning

Setting:

- Multiple related learning tasks
  - Eg. Object recognition
- Exploit task relatedness for better generalization

Setting:

- Multiple related learning tasks
  - Eg. Object recognition
- Exploit task relatedness for better generalization

The problem:

- Learn shared features across tasks
- If possible, sparse feature representations

A simple case...

Suppose:

- Tasks share a few input features.

A simple case...

Suppose:

- Tasks share a few input features.

Formulation:

$$\min_{\mathbf{w},b,\xi} \quad \sum_{t=1}^{T} \sum_{i=1}^{m_t} \xi_{ti}$$
$$\text{s.t.} \quad y_{ti}(\mathbf{w}_t^\top \mathbf{x}_{ti} - b_t) \geq 1 - \xi_{ti},\ \ \xi_{ti} \geq 0$$

A simple case...

Suppose:

- Tasks share a few input features.

Formulation:

$$\min_{\mathbf{w},b,\xi} \quad \frac{1}{2}\sum_{t=1}^{T}\|\mathbf{w}_t\|_2^2 + C\sum_{t=1}^{T}\sum_{i=1}^{m_t}\xi_{ti}$$

$$\text{s.t.} \quad y_{ti}(\mathbf{w}_t^\top \mathbf{x}_{ti} - b_t) \geq 1 - \xi_{ti},\ \xi_{ti} \geq 0$$

A simple case...

Suppose:

- Tasks share a few input features.

Formulation:

$$\min_{\mathbf{w},b,\xi} \quad \frac{1}{2}\sum_{t=1}^{T}\|\mathbf{w}_t\|_1^2 + C\sum_{t=1}^{T}\sum_{i=1}^{m_t}\xi_{ti}$$

$$\text{s.t.} \quad y_{ti}(\mathbf{w}_t^\top \mathbf{x}_{ti} - b_t) \geq 1 - \xi_{ti},\ \xi_{ti} \geq 0$$

A simple case...

Suppose:

- Tasks share a few input features.

Formulation:

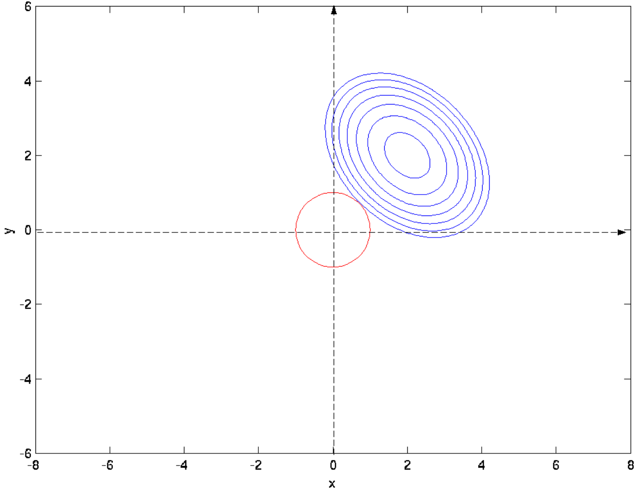
$$\min_{\mathbf{w},b,\xi} \quad \frac{1}{2}\left(\sum_{f=1}^{d} \|\mathbf{w}^f\|_2\right)^2 + C\sum_{t=1}^{T}\sum_{i=1}^{m_t}\xi_{ti}$$

$$\text{s.t.} \quad y_{ti}(\mathbf{w}_t^\top \mathbf{x}_{ti} - b_t) \geq 1 - \xi_{ti},\ \xi_{ti} \geq 0$$

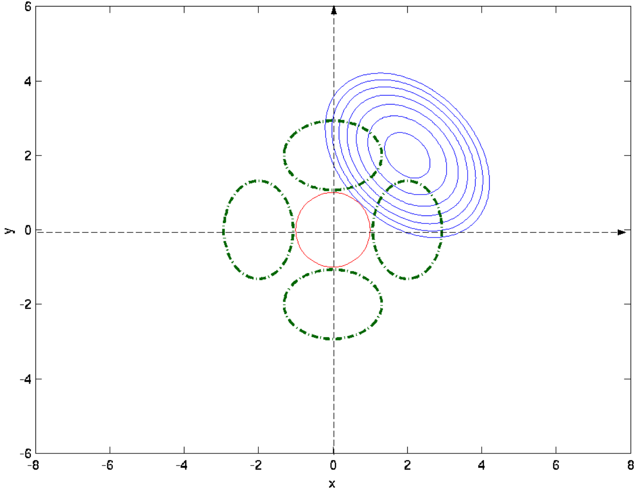
l_1-l_2 Regularizer

$$\sum_{f=1}^{d} \|\mathbf{w}^f\|_2 \underbrace{\Leftarrow}_{l_1} \left\{ \begin{array}{c} \|\mathbf{w}^1\|_2 \\ \vdots \\ \|\mathbf{w}^d\|_2 \end{array} \right. \underbrace{\Leftarrow}_{l_2} \left\{ \begin{array}{cccc} w_{11} & \dots & w_{T1} & \leftarrow \mathbf{w}^1 \\ \vdots & \vdots & \vdots & \vdots \\ w_{1d} & \dots & w_{Td} & \leftarrow \mathbf{w}^d \\ \uparrow & & \uparrow & \\ \mathbf{w}_1 & \dots & \mathbf{w}_T & \end{array} \right.$$

Consider $\min_{\mathbf{x}:\|\mathbf{x}\|_2\leq 1} f(\mathbf{x})$

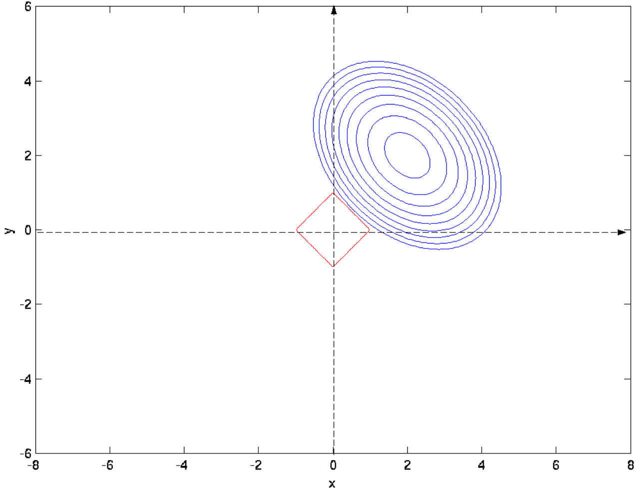


Consider $\min_{\mathbf{x}:\|\mathbf{x}\|_2\leq 1} f(\mathbf{x})$

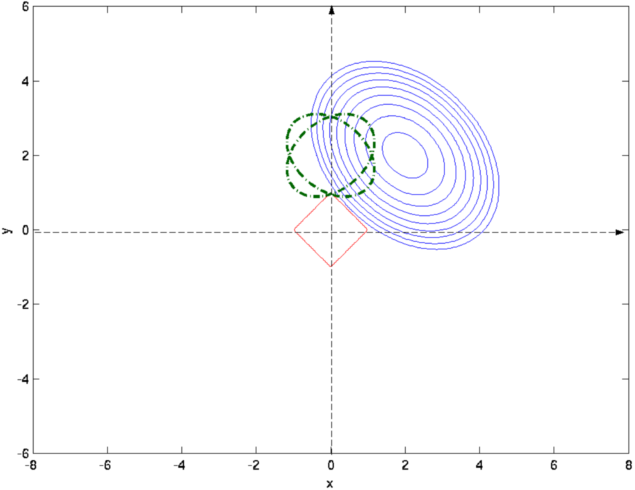


Interpretation of l_p regularization

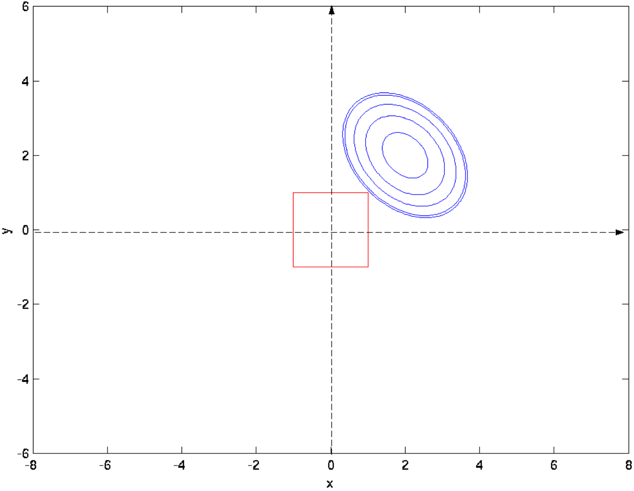
Consider $\min_{\mathbf{x}:\|\mathbf{x}\|_1 \leq 1} f(\mathbf{x})$



Consider $\min_{\mathbf{x}:\|\mathbf{x}\|_1 \leq 1} f(\mathbf{x})$



Consider $\min_{\mathbf{x}:\|\mathbf{x}\|_\infty \leq 1} f(\mathbf{x})$



Summary:

- $1 \leq p < 2$ promote sparsity
- $p = 2$ induces robustness, rotation-invariant
- $2 < p < \infty$ promote non-sparse combinations
- $p = \infty$ promotes equal weightages

A Bit More Realistic Case...

Suppose: [Argyriou et.al., 08]

- Tasks share a few (may be learnt) features.

A Bit More Realistic Case...

Suppose: [Argyriou et.al., 08]

- Tasks share a few (may be learnt) features.
  - Rotationally transformed features

A Bit More Realistic Case...

Suppose: [Argyriou et.al., 08]

- Tasks share a few (may be learnt) features.
  - Rotationally transformed features

Formulation:

$$\min_{\mathbf{w},b,\xi,\mathbf{L}} \left(\sum_{f=1}^{d} \|\mathbf{w}^f\|_2\right)^2 + C\sum_{t=1}^{T}\sum_{i=1}^{m_t}\xi_{ti}$$

$$\text{s.t.} \quad y_{ti}(\mathbf{w}_t^\top \mathbf{L}^\top \mathbf{x}_{ti} - b_t) \geq 1 - \xi_{ti},\ \xi_{ti} \geq 0,\ \mathbf{L} \in O^d$$

Multi-task Sparse Feature Learning (MTSFL) Formulation

Summary:

- Though non-convex global optimum can be obtained
- Can be kernelized
- Efficient alternate minimization algorithm (EVD per iteration)
- Achieves state-of-the-art performance on benchmarks

Multi-task Sparse Feature Learning (MTSFL) Formulation

Summary:

- Though non-convex global optimum can be obtained
- Can be kernelized
- Efficient alternate minimization algorithm (EVD per iteration)
- Achieves state-of-the-art performance on benchmarks

Discussion:

- Rotationally transformed features — too restrictive
  - Essential for convexity

Multi-task Sparse Feature Learning (MTSFL) Formulation

Summary:

- Though non-convex global optimum can be obtained
- Can be kernelized
- Efficient alternate minimization algorithm (EVD per iteration)
- Achieves state-of-the-art performance on benchmarks

Discussion:

- Rotationally transformed features — too restrictive
  - Essential for convexity
- Idea: Enrich the input space itself
  - Multiple Kernel Learning (MKL) ??

Central Idea

Pose the problem as that of learning a shared kernel

Central Idea

Pose the problem as that of learning a shared kernel

Outline:

- Two formulations:
  - learn kernel shared across tasks (MK-MTFL)
    - Extension of standard MKL to multi-task case
  - learn sparse representation from shared kernel (MK-MTSFL)
    - Extension of MTSFL to multiple base kernels

Notational stuff...

- $k_1, \ldots, k_n$ base kernels
- $\phi_j(\cdot)$ implicit mapping with k_j
- w_{tjf} — t^{th} task, j^{th} kernel, f^{th} feature loading
- $\mathbf{w}_{\cdot jf}, \mathbf{w}_{t \cdot f}, \mathbf{w}_{tj \cdot}$
- Linear model: $f_t(\mathbf{x}) = \sum_{j=1}^{n} \mathbf{w}_{tj\cdot}^{\top} \phi_j(\mathbf{x}) - b_t$

MK-MTFL Formulation

Primal:

$$\min_{\mathbf{w},b,\xi} \quad \frac{1}{2}\overbrace{\left(\sum_{j=1}^{n}\left(\sum_{t=1}^{T}(\|\mathbf{w}_{tj\cdot}\|_2)^2\right)^{\frac{1}{2}}\right)^2}^{l_1\text{-}l_2\text{-}l_2} + C\sum_{t=1}^{T}\sum_{i=1}^{m_t}\xi_{ti}$$

$$\text{s.t.} \quad y_{ti}\left(\sum_{j=1}^{n}\mathbf{w}_{tj\cdot}^{\top}\phi_j(\mathbf{x}_{ti}) - b_t\right) \geq 1 - \xi_{ti},\ \ \xi_{ti} \geq 0$$

MK-MTFL Formulation

Primal:

$$\min_{\mathbf{w},b,\xi} \quad \frac{1}{2}\overbrace{\left(\sum_{j=1}^{n}\left(\sum_{t=1}^{T}(\|\mathbf{w}_{tj\cdot}\|_2)^2\right)^{\frac{1}{2}}\right)^2}^{l_1\text{-}l_2\text{-}l_2} + C\sum_{t=1}^{T}\sum_{i=1}^{m_t}\xi_{ti}$$

$$\text{s.t.} \quad y_{ti}\left(\sum_{j=1}^{n}\mathbf{w}_{tj\cdot}^{\top}\phi_j(\mathbf{x}_{ti}) - b_t\right) \geq 1 - \xi_{ti}, \;\; \xi_{ti} \geq 0$$

Partial Dual:

$$\min_{\gamma\in\Delta_n}\max_{\alpha_t\in S_{m_t}(C)}\sum_{t=1}^{T}\left\{\mathbf{1}^{\top}\alpha_t - \frac{1}{2}\alpha_t^{\top}\mathbf{Y}_t\left[\sum_{j=1}^{n}\gamma_j\mathbf{K}_{tj}\right]\mathbf{Y}_t\alpha_t\right\}$$

MK-MTFL Formulation

Primal $(2 \leq p \leq \infty)$:

$$\min_{\mathbf{w},b,\xi} \quad \frac{1}{2} \overbrace{\left(\sum_{j=1}^{n} \left(\sum_{t=1}^{T} (\|\mathbf{w}_{tj\cdot}\|_2)^p \right)^{\frac{1}{p}} \right)^2}^{l_1\text{-}l_p\text{-}l_2,\ p \geq 2} + C \sum_{t=1}^{T} \sum_{i=1}^{m_t} \xi_{ti}$$

$$\text{s.t.} \quad y_{ti}\left(\sum_{j=1}^{n} \mathbf{w}_{tj\cdot}^{\top} \phi_j(\mathbf{x}_{ti}) - b_t\right) \geq 1 - \xi_{ti},\ \ \xi_{ti} \geq 0$$

MK-MTFL Formulation

Primal $(2 \leq p \leq \infty)$:

$$\min_{\mathbf{w},b,\xi} \quad \frac{1}{2}\overbrace{\left(\sum_{j=1}^{n}\left(\sum_{t=1}^{T}(\|\mathbf{w}_{tj\cdot}\|_2)^{p}\right)^{\frac{1}{p}}\right)}^{l_1\text{-}l_p\text{-}l_2,\ p\geq 2}{}^{2} + C\sum_{t=1}^{T}\sum_{i=1}^{m_t}\xi_{ti}$$

$$\text{s.t.} \quad y_{ti}\left(\sum_{j=1}^{n}\mathbf{w}_{tj\cdot}^{\top}\phi_j(\mathbf{x}_{ti}) - b_t\right) \geq 1 - \xi_{ti},\ \ \xi_{ti} \geq 0$$

Partial Dual $\left(\bar{p} = \frac{p}{p-2}\right)$:

$$\min_{\gamma\in\Delta_n}\ \max_{\lambda_j\in\Delta_{T,\bar{p}}}\ \max_{\alpha_t\in S_{m_t}(C)} \sum_{t=1}^{T}\left\{\mathbf{1}^{\top}\alpha_t - \frac{1}{2}\alpha_t^{\top}\mathbf{Y}_t\left[\sum_{j=1}^{n}\frac{\gamma_j\mathbf{K}_{tj}}{\lambda_{jt}}\right]\mathbf{Y}_t\alpha_t\right\}$$

MK-MTFL Formulation

Summary:

- Novel formulation for learning shared kernel
- Extension of MKL to multi-task case
- Tasks can be unequally reliable
- Efficient mirror-descent based alg.
  - Each step solves T regular SVMs $O(\sum_{t=1}^{T} m_t^2 dn)$

MK-MTSFL Formulation

Primal ($1 \leq q \leq 2$):

$$\min_{\mathbf{w}, b, \xi, \mathbf{L}} \quad \frac{1}{2} \overbrace{\left(\sum_{j=1}^{n} \left(\sum_{f=1}^{d_j} \|\mathbf{w}_{\cdot jf}\|_2 \right)^q \right)^{\frac{2}{q}}}^{l_q\text{-}l_1\text{-}l_2} + C \sum_{t=1}^{T} \sum_{i=1}^{m_t} \xi_{ti}$$

$$\text{s.t.} \quad y_{ti}\left(\sum_{j=1}^{n} \mathbf{w}_{tj\cdot}^\top \mathbf{L}_j^\top \phi_j(\mathbf{x}_{ti}) - b_t\right) \geq 1 - \xi_{ti}, \ \xi_{ti} \geq 0, \mathbf{L}_j \in O^{d_j}$$

MK-MTSFL Formulation

Primal $(1 \leq q \leq 2)$:

$$\min_{\mathbf{w},b,\xi,\mathbf{L}} \quad \frac{1}{2}\overbrace{\left(\sum_{j=1}^{n}\left(\sum_{f=1}^{d_j}\|\mathbf{w}_{\cdot jf}\|_2\right)^{q}\right)^{\frac{2}{q}}}^{l_q\text{-}l_1\text{-}l_2} + C\sum_{t=1}^{T}\sum_{i=1}^{m_t}\xi_{ti}$$

$$\text{s.t.} \quad y_{ti}\left(\sum_{j=1}^{n}\mathbf{w}_{tj\cdot}^{\top}\mathbf{L}_j^{\top}\phi_j(\mathbf{x}_{ti}) - b_t\right) \geq 1 - \xi_{ti}, \ \ \xi_{ti} \geq 0, \mathbf{L}_j \in O^{d_j}$$

Partial Dual $\left(\bar{q} = \frac{q}{2-q}\right)$:

$$\min_{\mathbf{Q}} \sum_{t=1}^{T} \max_{\alpha_t \in S_{m_t}(C)} \quad \mathbf{1}^{\top}\alpha_t - \frac{1}{2}\alpha_t^{\top}\mathbf{Y}_t\left(\sum_{j=1}^{n}\mathbf{M}_{tj}^{\top}\mathbf{Q}_j\mathbf{M}_{tj}\right)\mathbf{Y}_t\alpha_t$$

$$\text{s.t.} \quad \mathbf{Q}_j \succeq 0, \sum_{j=1}^{n}(trace(\mathbf{Q}_j))^{\bar{q}} \leq 1$$

MK-MTSFL Formulation

Summary:

- Novel formulation for learning shared sparse feature representations
  - Trace-norm constraints lead to low rank matrices
- Extension of MTSFL [Argyriou et.al., 08] to multiple base kernels
- Though non-convex, global optimal can be efficiently obtained
- Efficient mirror-descent based algorithm
  - Each step solves T regular SVMs, n EVDs of full matrices
- Faster convergence in practice than alternate minimization

Partial Dual:

$$\min_{\mathbf{Q}} \sum_{t=1}^{T} \max_{\alpha_t \in S_{m_t}(C)} \mathbf{1}^\top \alpha_t - \frac{1}{2} \alpha_t^\top \mathbf{Y}_t \left(\sum_{j=1}^{n} \mathbf{M}_{tj}^\top \mathbf{Q}_j \mathbf{M}_{tj} \right) \mathbf{Y}_t \alpha_t$$

$$\text{s.t.} \qquad \mathbf{Q}_j \succeq 0, \sum_{j=1}^{n} (trace(\mathbf{Q}_j))^{\bar{q}} \leq 1$$

Partial Dual:

$$\min_{\mathbf{Q}} \overbrace{\sum_{t=1}^{T} \max_{\alpha_t \in S_{m_t}(C)} \mathbf{1}^\top \alpha_t - \frac{1}{2} \alpha_t^\top \mathbf{Y}_t \left(\sum_{j=1}^{n} \mathbf{M}_{tj}^\top \mathbf{Q}_j \mathbf{M}_{tj} \right) \mathbf{Y}_t \alpha_t}^{g(\mathbf{Q})}$$

$$\text{s.t.} \quad \mathbf{Q}_j \succeq 0, \sum_{j=1}^{n} (trace(\mathbf{Q}_j))^{\bar{q}} \leq 1$$

Solving MK-MTSFL

Partial Dual:

$$\min_{\mathbf{Q}} \overbrace{\sum_{t=1}^{T} \max_{\alpha_t \in S_{m_t}(C)} \mathbf{1}^\top \alpha_t - \frac{1}{2}\alpha_t^\top \mathbf{Y}_t \left(\sum_{j=1}^{n} \mathbf{M}_{tj}^\top \mathbf{Q}_j \mathbf{M}_{tj} \right) \mathbf{Y}_t \alpha_t}^{g(\mathbf{Q})}$$

$$\text{s.t.} \qquad \mathbf{Q}_j \succeq 0, \sum_{j=1}^{n} (trace(\mathbf{Q}_j))^{\bar{q}} \leq 1$$

- $g(\mathbf{Q})$ cannot be analytically computed
- Danskin's theorem provides $\nabla g(\mathbf{Q})$
  - Involves solving T regular SVMs

Projected (Sub-)Gradient Descent

- $\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})$ (f is convex, Lipschitz, $\mathcal{X}$ is compact)
- At iteration k:

$$\mathbf{x}_{k+1}$$

$$= \Pi_{\mathcal{X}}(\mathbf{x}_k - s_k \nabla f(\mathbf{x}_k))$$

Projected (Sub-)Gradient Descent

- $\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})$ (f is convex, Lipschitz, $\mathcal{X}$ is compact)
- At iteration k:
  - f is approx. by linear func. $f(\mathbf{x}) = f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^\top(\mathbf{x} - \mathbf{x}_k)$
  - valid only when $\|\mathbf{x} - \mathbf{x}_k\|_2$ is small

$$
\begin{aligned}
\mathbf{x}_{k+1} &= \arg\min_{\mathbf{x}\in\mathcal{X}} \quad s_k \nabla f(\mathbf{x}_k)^\top(\mathbf{x} - \mathbf{x}_k) + \frac{1}{2}\|\mathbf{x} - \mathbf{x}_k\|_2^2 \\
&= \arg\min_{\mathbf{x}\in\mathcal{X}} \quad \frac{1}{2}\|\mathbf{x} - (\mathbf{x}_k - s_k \nabla f(\mathbf{x}_k))\|_2^2 \\
&= \Pi_{\mathcal{X}}(\mathbf{x}_k - s_k \nabla f(\mathbf{x}_k))
\end{aligned}
$$

Projected (Sub-)Gradient Descent

- $\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})$ (f is convex, Lipschitz, $\mathcal{X}$ is compact)
- At iteration k:
  - f is approx. by linear func. $f(\mathbf{x}) = f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^\top(\mathbf{x} - \mathbf{x}_k)$
  - valid only when $\|\mathbf{x} - \mathbf{x}_k\|_2$ is small

$$
\begin{aligned}
\mathbf{x}_{k+1} &= \arg\min_{\mathbf{x}\in\mathcal{X}} \quad s_k \nabla f(\mathbf{x}_k)^\top(\mathbf{x} - \mathbf{x}_k) + \frac{1}{2}\|\mathbf{x} - \mathbf{x}_k\|_2^2 \\
&= \arg\min_{\mathbf{x}\in\mathcal{X}} \quad \frac{1}{2}\|\mathbf{x} - (\mathbf{x}_k - s_k \nabla f(\mathbf{x}_k))\|_2^2 \\
&= \Pi_{\mathcal{X}}(\mathbf{x}_k - s_k \nabla f(\mathbf{x}_k))
\end{aligned}
$$

- Convergence guarantees with some choices of step-sizes (s_k)
- "Optimal" for Euclidean geometry

Projected (Sub-)Gradient Descent

- $\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})$ (f is convex, Lipschitz, $\mathcal{X}$ is compact)
- At iteration k:
  - f is approx. by linear func. $f(\mathbf{x}) = f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^\top(\mathbf{x}-\mathbf{x}_k)$
  - valid only when $\|\mathbf{x}-\mathbf{x}_k\|_2$ is small

$$\begin{aligned}\mathbf{x}_{k+1} &= \arg\min_{\mathbf{x}\in\mathcal{X}} \quad s_k\nabla f(\mathbf{x}_k)^\top(\mathbf{x}-\mathbf{x}_k) + \frac{1}{2}\|\mathbf{x}-\mathbf{x}_k\|_2^2 \\ &= \arg\min_{\mathbf{x}\in\mathcal{X}} \quad \frac{1}{2}\|\mathbf{x}-(\mathbf{x}_k - s_k\nabla f(\mathbf{x}_k))\|_2^2 \\ &= \Pi_{\mathcal{X}}(\mathbf{x}_k - s_k\nabla f(\mathbf{x}_k))\end{aligned}$$

- Convergence guarantees with some choices of step-sizes (s_k)
- "Optimal" for Euclidean geometry

Key Idea:

- Bregmann divergence based regularizer so that per-step problem is easy

$$\mathbf{x}_{k+1} = \arg\min_{\mathbf{x}\in\mathcal{X}} \quad s_k \nabla f(\mathbf{x}_k)^\top (\mathbf{x} - \mathbf{x}_k) + \frac{1}{2}\|\mathbf{x} - \mathbf{x}_k\|_2^2$$

Key Idea:

- Bregmann divergence based regularizer so that per-step problem is easy

$$\mathbf{x}_{k+1} = \arg\min_{\mathbf{x}\in\mathcal{X}} \quad s_k \nabla f(\mathbf{x}_k)^\top (\mathbf{x} - \mathbf{x}_k) + D_{\mathbf{x}_k}(\mathbf{x})$$

Mirror Descent

Key Idea:

- Bregmann divergence based regularizer so that per-step problem is easy

$$\mathbf{x}_{k+1} = \arg\min_{\mathbf{x}\in\mathcal{X}} \quad s_k \nabla f(\mathbf{x}_k)^\top (\mathbf{x} - \mathbf{x}_k) + D_{\mathbf{x}_k}(\mathbf{x})$$

Bregmann Divergence:

- Strongly convex $\omega(\cdot)$: $D_x(y) = \omega(y) - \omega(x) - \nabla\omega(x)^\top (y - x)$
- Common choices:
  - $\mathcal{X}$ Sphere: $\omega(x) = \frac{1}{2}\|x\|_2^2$
  - $\mathcal{X}$ Simplex: $\omega(x) = \sum_i x_i \log(x_i)$
  - $\mathcal{X}$ Spectrahedron: $\omega(x) = trace(x \log(x))$

Solving MK-MTSFL

Our Finding:

Entropy function $trace(x\log(x))$ good enough for our problem

Solving MK-MTSFL

Our Finding:

Entropy function $trace(x \log(x))$ good enough for our problem

Per-step problem:

$$\min_{\mathbf{Q}} \quad \sum_{j=1}^{n} \{trace(\zeta_j \mathbf{Q}_j) + trace(\mathbf{Q}_j \log(\mathbf{Q}_j))\}$$

$$\text{s.t.} \quad \mathbf{Q}_j \succeq 0, \sum_{j=1}^{k} (trace(\mathbf{Q}_j))^{\bar{q}} \leq 1$$

Solving MK-MTSFL

Our Finding:

Entropy function $trace(x \log(x))$ good enough for our problem

Per-step problem:

$$\min_{\mathbf{Q}} \quad \sum_{j=1}^{n} \{trace(\zeta_j \mathbf{Q}_j) + trace(\mathbf{Q}_j \log(\mathbf{Q}_j))\}$$
$$\text{s.t.} \quad \mathbf{Q}_j \succeq 0, \sum_{j=1}^{k} (trace(\mathbf{Q}_j))^{\bar{q}} \leq 1$$

After EVDs of $\mathbf{Q}_j$:

$$\min_{\rho} \quad \sum_{j=1}^{n} (\rho_j \log(\rho_j) + \rho_j \pi_j)$$
$$\text{s.t.} \quad \rho_j \geq 0, \sum_{j=1}^{n} {\rho_j}^{\bar{q}} \leq 1$$

Simulations

Datasets:

School: Multi-task benchmark. Prediction of student performance in various schools.

- 139 regression tasks
- 28 input features
- 15 training examples per task

Letters: OCR dataset. Each letter considered as a task.

- 9 binary classification tasks
- 128 input features
- 10 training examples per task

Dermatology: Bio-informatics dataset. Predicting one of six skin-diseases.

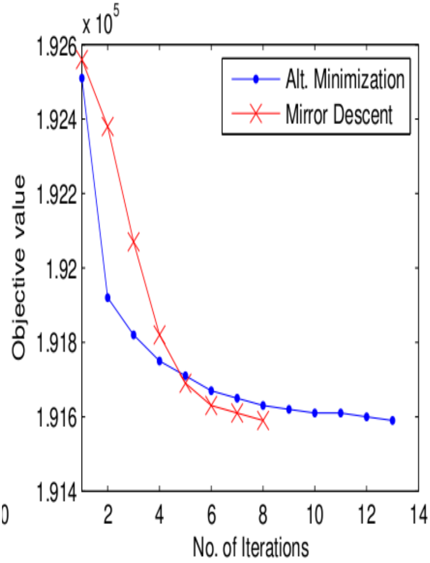
- 15 binary classification tasks
- 33 input features
- 10 training examples per task

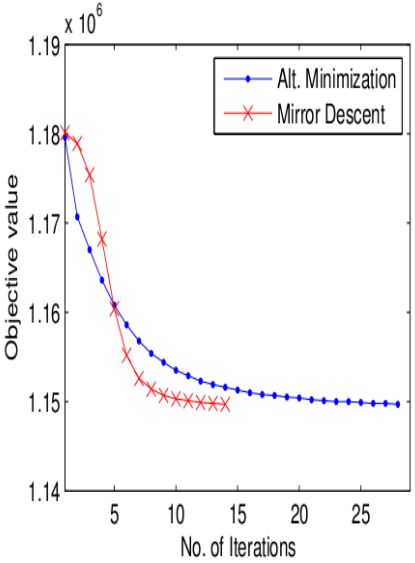
TABLE: Comparison of generalization performance

	SVM	MTSFL	MK-MTFL			MK-MTSFL		
			p =2	7	Inf	q =1	1.5	1.99
S	-45.88	13.94	10.76	13.80	10.52	**14.07**	13.80	13.94
L	74.89	75.54	78.28	78.30	**78.31**	76.38	76.93	74.57
D	8	6	**0**	**0**	**0**	8	7	5.33

MTFSTL – 179sec, **MK-MTFL** – 192sec and **MK-MTSFL** – 15445sec.

Simulations





Conclusions

- Two novel formulations for multi-task feature learning:
  - Extension of MKL to multi-task case (non-sparse)
    - Simple, good generalization, scalable
  - Extension of MTSFL to multiple base kernels (sparse)
    - better generalization than state-of-the-art
- Efficient mirror-descent based algorithm
  - Faster convergence
- Sparse representations may not always be desirable

Questions ?

Thank You