Learning Good Representations for Multiple Related Tasks

Massimiliano Pontil

Computer Science Dept UCL and ENSAE-CREST

March 25, 2015

Plan

- Problem formulation and examples
- Analysis of multitask learning
- Comparison to independent task learning
- Analysis of learning to learn
- Multilinear MTL (if time)
- Latent subcategory models (if time)

Problem Formulation (cont.)

- Fix probability distributions $\mu_1, \ldots, \mu_T$ on $\mathbb{R}^d \times \mathbb{R}$
- Draw data: $(x_{t1}, y_{t1}), \ldots, (x_{tn}, y_{tn}) \sim \mu_t, \quad t = 1,...,T$
- Two interesting examples (linear models)
  - Regression: $y_{ti} = \langle u_t, x_{ti} \rangle + \epsilon_{ti}$
  - Binary classification: $y_{ti} = \mathrm{sign}\langle u_t, x_{ti} \rangle \epsilon_{ti}$
- Learning method: $\displaystyle \min_{[w_1,...,w_T] \in \mathcal{S}} \frac{1}{T} \sum_{t=1}^{T} \frac{1}{n} \sum_{i=1}^{n} \ell(y_{ti}, \langle w_t, x_{ti} \rangle)$
- Set $\mathcal{S}$ encourages "common structure" among tasks, e.g. the ball of a matrix norm or other regularizer
- Independent task learning (ITL): $\mathcal{S} = \underbrace{\mathcal{B} \times \cdots \times \mathcal{B}}_{\text{T times}}$

Problem Formulation (cont.)

$$\min_{[w_1,\ldots,w_T]\in\mathcal{S}} \ \frac{1}{T}\sum_{t=1}^{T}\frac{1}{n}\sum_{i=1}^{n}\ell(y_{ti},\langle w_t,x_{ti}\rangle)$$

- Want to find weights vectors which have a small average error

$$\frac{1}{T}\sum_{t=1}^{T}\mathop{\mathbb{E}}_{(x,y)\sim\mu_t}\ell(y,\langle w_t,x\rangle)$$

- Typically: **many tasks** but only **few examples per task**

- If $n < d$ we don't have enough data to learn tasks one by one [Maurer & P., 2008]. If tasks are *"related"*, learning them *jointly* should improve over ITL

Applications

- **User modelling:**
  - ◇ each task is to predict a user's ratings of products
  - ◇ the ways different people make decisions about products are related

- **Multiple object detection in scenes:**
  - ◇ detection of each object corresponds to a binary classification task: $y_{ti} \in \{-1, 1\}$
  - ◇ learning common features enhances performance

Many more: affective computing, bioinformatics, neuroimaging, NLP,...

Examples of Regularizers

- Quadratic, e.g. $\sum_{t=1}^{T} \left\| w_t - \bar{w} \right\|_2^2$ or $\sum_{s,t=1}^{T} A_{st} \left\| w_t - w_s \right\|_2^2$

- Learning shared representations

  - Joint sparsity: $\sum_{j=1}^{d} \sqrt{\sum_{t=1}^{T} w_{jt}^2}$

  - Low rank: $\|[w_1, ..., w_T]\|_{\mathrm{tr}}$ (common low dimensional representation / subspace)

- Nonlinear extension using RKHS (not discussed in this talk)

[Argyriou et al. 2006, 2008, 2009; Baldassarre et al. 2012; Ben-David and Schuller, 2003; Caponnetto et al. 2008; Carmeli et al. 2006; Cavallanti et al. 2009; Dinuzzo & Fukumizu, 2012; Evgeniou & P. 2004; Evgeniou et al. 2005; Jacob et al. 2008; Koltchinskii et al. 2011; Kumar & Daumé III, 2012; Lounici et al., 2009, 2011; Maurer, 2006; Micchelli & P., 2005; Obozinski et al. 2009; Romera-Paredes et al. 2012; Salakhutdinov et al, 2011,...]

Learning Sparse Representations

[Maurer, P., Romera-Paredes. ICML 2013]

- Represent w_t's as **sparse combinations** of some vectors:

$$w_t = D\gamma_t = \sum_{k=1}^{K} D_k \gamma_{kt} \;\; : \|\gamma_t\|_1 \leq \alpha$$

- Set of **dictionaries** $\mathcal{D}_K := \left\{ D \in \mathbb{R}^{d \times K} : \max_{k=1}^{K} \|D_k\|_2 \leq 1 \right\}$

- Learning method: $\displaystyle \min_{D \in \mathcal{D}_K} \frac{1}{T} \sum_{t=1}^{T} \min_{\|\gamma\|_1 \leq \alpha} \frac{1}{n} \sum_{i=1}^{n} \ell\big(\langle D\gamma, x_{ti} \rangle, y_{ti}\big)$

- Two regularization parameters: K and α

- For fixed D this is Lasso with **feature map** $\phi(x) = D^\top x$

See also: [Kumar & Daumé III 2012; Mehta & Gray 2013; Ruvolo and Eaton 2013]

Connection to Sparse Coding

If $\ell(z,y) = (z-y)^2$, $y_{ti} = \langle w_t, x_{ti} \rangle$, $x_{ti} \sim \mathcal{N}(0, I)$ and $n \to \infty$, we recover **sparse coding** [Olshausen and Field 1996]:

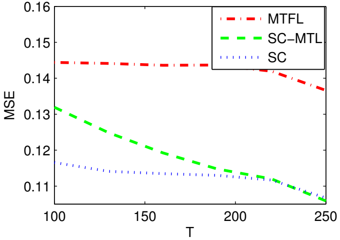
$$\min_{D \in \mathcal{D}_K} \frac{1}{T} \sum_{t=1}^{T} \min_{\|\gamma\|_1 \leq \alpha} \|w_t - D\gamma\|_2^2$$

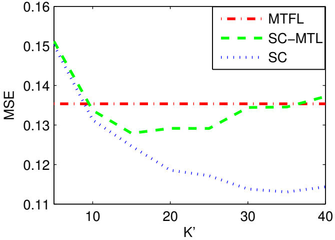
May extend to a general set of **codevectors** $\mathcal{C}$ [Maurer and P. 2010], such as:

- K-means clustering: $\mathcal{C} = \{e_1, \ldots, e_K\}$
- Subspace learning: $\mathcal{C} = \{\|\gamma\|_2 \leq \alpha\}$
- Union of subspaces: $\mathcal{C} = \left\{\gamma = (\gamma^{(1)}, \ldots, \gamma^{(L)}) \in (\mathbb{R}^q)^L : \sum_{\ell=1}^{L} \|\gamma^{(\ell)}\|_2 \leq \alpha\right\}$

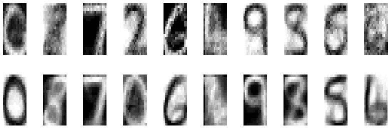
Experiment

Learn a dictionary for image reconstruction from few pixel values (input space is the set of possible pixels indices, output space represents the gray level)





Found dictionary (top) vs. dictionary by standard SC (bottom):



MTL Analysis

Goal is to bound the *excess error*

$$\frac{1}{T}\sum_{t=1}^{T} \mathop{\mathbb{E}}_{(x,y)\sim\mu_t} \ell(\langle \widehat{D}\hat{\gamma}_t, x\rangle, y) - \min_{D\in\mathcal{D}_K} \frac{1}{T}\sum_{t=1}^{T} \min_{\|\gamma_t\|_1\le\alpha} \mathop{\mathbb{E}}_{(x,y)\sim\mu_t} \ell(\langle D\gamma_t, x\rangle, y)$$

Assumptions: $\ell(y,\cdot)$ is L-Lipschitz and $\|x_{ti}\| \le 1$ a.s.

Bound for MTL

Theorem 1. Let $\widehat{S}_p := \frac{1}{T}\sum_{t=1}^{T} \|\widehat{\Sigma}_t\|_p$, $p \geq 1$. With prob. $\geq 1-\delta$ the excess error is upper bounded by

$$L\alpha\sqrt{\frac{8\widehat{S}_\infty \log(2K)}{n}} + L\alpha\sqrt{\frac{2\widehat{S}_1(K+12)}{nT}} + \sqrt{\frac{8\log\frac{4}{\delta}}{nT}}$$

- If T grows, bound is **comparable to Lasso** with best a-priori known dictionary! [Kakade et al. 2012]

- If input distribution is uniform on the unit sphere then $\widehat{S}_1 = 1$ and $\widehat{S}_\infty \approx \frac{1}{n}$ (assuming $n < d$)

Proof Idea

Bound the Rademacher average

$$R = \frac{1}{nT}\mathbb{E}_\sigma \sup_{D,\gamma} \sum_{t=1}^{T}\sum_{i=1}^{n}\sigma_{ti}\langle D\gamma_t, x_{ti}\rangle$$

Lemma. If $F_\gamma(\sigma) = \sup_{D} \sum_{t=1}^{T}\sum_{i=1}^{n}\sigma_{ti}\langle D\gamma_t, x_{ti}\rangle$ then

$$\Pr\left(F_\gamma \geq \mathbb{E}F_\gamma + \epsilon\right) \leq \exp\left(\frac{-\epsilon^2}{8nT\hat{S}_\infty}\right)$$

Follows from a generalization of McDiarmid's inequality [Boucheron et al., 2013]

Proof Idea (cont)

Step 1: a direct computation gives $\mathbb{E}F_\gamma \leq \sqrt{nTK\hat{S}_1} := c$

Step 2: observe that

$$\begin{aligned} nTR &= \mathbb{E}\max_{\gamma\in\mathcal{C}^T} F_\gamma = \mathbb{E}\max_{\gamma\in\mathrm{ext}(\mathcal{C})^{\mathrm{T}}} F_\gamma \\ &= \int_0^\infty \Pr\left(\max_{\gamma\in\mathrm{ext}(\mathcal{C})^{\mathrm{T}}} F_\gamma > s\right) ds \\ &\leq c + \delta + \sum_{\gamma\in\mathrm{ext}(\mathcal{C})^{\mathrm{T}}} \int_{\delta+c}^\infty \Pr\left(F_\gamma > s\right) ds \\ &\leq c + \delta + (2K)^T \int_\delta^\infty \Pr\left(F_\gamma > \mathbb{E}F_\gamma + s\right) ds \end{aligned}$$

Step 3: use above lemma and optimize over δ, then use standard bound on uniform deviation between empirical and true error [Koltchinskii & Panchenko, 2002; Bartlett & Mendelson, 2002]

Subspace Learning

Same as before but now use L2 norm on the code vectors

$$\min_{D \in \mathcal{D}_K} \frac{1}{T} \sum_{t=1}^{T} \min_{\|\gamma\|_2 \leq \alpha} \frac{1}{n} \sum_{i=1}^{n} \ell\big(\langle D\gamma, x_{ti} \rangle, y_{ti}\big)$$

Excess error:

$$\frac{1}{T} \sum_{t=1}^{T} \mathop{\mathbb{E}}_{(x,y)\sim\mu_t} \ell(\langle \widehat{D}\hat{\gamma}_t, x \rangle, y) - \min_{D \in \mathcal{D}_K} \frac{1}{T} \sum_{t=1}^{T} \min_{\|\gamma_t\|_2 \leq \alpha} \mathop{\mathbb{E}}_{(x,y)\sim\mu_t} \ell(\langle D\gamma_t, x \rangle, y)$$

Subspace Learning

Theorem Let $\hat{C} = \frac{1}{nT} \sum_{t,i} x_{ti} \otimes x_{ti}$. With probability $\geq 1 - \delta$ the excess error is upper bounded by

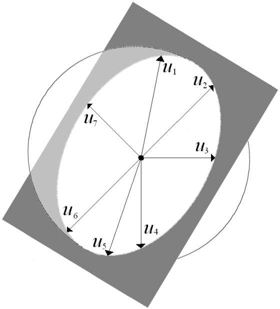
$$2L\left(\sqrt{\frac{K\|\hat{C}\|_\infty}{n}} + \sqrt{\frac{2K\left(\ln\left(nT\right)+1\right)}{nT}}\right) + \sqrt{\frac{8\ln\left(4/\delta\right)}{nT}}$$

- Leading term $O\left(\sqrt{\frac{K}{n}}\right)$ vs. $O\left(\sqrt{\frac{\log K}{n}}\right)$ for sparse coding, but possibly smaller minimum
- Based on bound for trace norm regularization [Maurer & P, 2013]
- Larger constant involving the total covariance

Binary Classification (Halfspace Learning)

Consider the following simple experiment (binary classification on the sphere with no noise): a unit weight vector w in $\mathbb{R}^d$ is chosen from a low dimensional subspace (or union of few subspaces) of dimension $K \ll d$. A random set on input vectors $x_i \sim \sigma$, are labeled by u: $y_i = \text{sign}(\langle u, x_i \rangle)$. Let $\mathbf{z} = \big((x_1, y_1), ..., (x_n, y_n)\big)$

The experiment is repeated T times, so we have weight vectors $u_1, ..., u_T$ and corresponding datasets $\mathbf{z}_1, ..., \mathbf{z}_T$

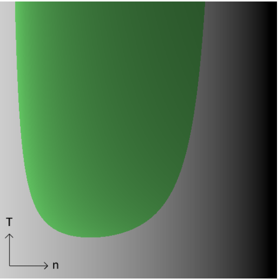


Halfspace Learning (cont.)

Compare MTL to ITL with orthogonal equivariant algorithm (OEA)

Lower bound for ITL [Maurer & P. 2008] For any OEA and $n < d$

$$\Pr\left\{\mathrm{err} \geq \frac{1}{\pi}\sqrt{\frac{d-n}{d}} - \eta\right\} \geq 1 - e^{-d(\pi\eta)^2}$$



green area: MTL is better, gray area: ITL may be better

Proof of the Lower Bound

Assume inputs are uniformly sampled on the unit sphere in $\mathbb{R}^d$

Step 1: the classification error of weight w relative to true vector u is

$$\mathrm{error}(w) = \sigma\{x : \mathrm{sign}(\langle w, x\rangle) \neq \mathrm{sign}(\langle u, x\rangle)\} = \frac{\mathrm{angle}(u, w)}{\pi} \geq \frac{d(u, w)}{\pi}$$

which implies

$$\mathrm{Pr}_{\mathbf{x}\sim\sigma^n}\{\mathrm{error}(u, w) < t\} \leq \mathrm{Pr}_{\mathbf{x}\sim\sigma^n}\{d(u, w) < \frac{t}{\pi}\}$$

Step 2: symmetry of the algorithm $\Rightarrow$ $w \in [\mathbf{x}] := \mathrm{span}(x_1, ..., x_n)$. This gives a further bound

$$\mathrm{Pr}_{\mathbf{x}\sim\sigma^n}\{d(u, [\mathbf{x}]) < \frac{t}{\pi}\}$$

Step 3: use the symmetry of σ to bound the above by

$$\sup_{\dim(M)\leq n} \mathrm{Pr}_{w\sim\sigma}\{d(w, M) < \frac{t}{\pi}\}$$

Step 4: use the fact that $\mathrm{Pr}_{w\sim\sigma}\{d(w, M) < \sqrt{\frac{d-n}{d}} - t\} \leq e^{-dt^2}$, which in turn follows from a result by [Dasgupta and Gupta, 2003]

Analysis of Transfer Learning

General setting involving distinct sets of *training* and *target tasks* sampled i.i.d. from a meta-distribution $\mathcal{E}$ (aka *learning to learn* [Baxter, 2000])

- sample $\mu_1, \ldots, \mu_T \sim \mathcal{E}$
- sample $\mathbf{z}_t \sim (\mu_t)^n$, $t = 1, \ldots, T$
- apply the method to learn a representation (dictionary) on the training task
- goal is to bound the *transfer error*

$$\mathcal{R}(D) = \mathbb{E}_{\mu\sim\mathcal{E}}\mathbb{E}_{\mathbf{z}\sim(\mu)^n} \mathop{\mathbb{E}}_{(x,y)\sim\mu} \ell(\langle D\gamma(D,\mathbf{z}), x\rangle, y)$$

where $\gamma(D,\mathbf{z}) = \mathop{\mathrm{argmin}}_{\|\gamma\|_1\le\alpha} \sum_{i=1}^n \ell(\langle D\gamma, x_i\rangle, y_i)$

- relative to $\mathcal{R}_{\rm opt} := \min\limits_{D\in\mathcal{D}_K} \mathop{\mathbb{E}}\limits_{\mu\sim\mathcal{E}} \min\limits_{\|\gamma\|_1\le\alpha} \mathop{\mathbb{E}}\limits_{(x,y)\sim\mu} \ell(\langle D\gamma, x\rangle, y)$

Analysis of Transfer Learning (cont.)

Theorem 2. Let $S_\infty(\mathcal{E}) := \mathop{\mathbb{E}}_{\mu\sim\mathcal{E}} \mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})\sim\mu^n} \|\Sigma(\mathbf{x})\|_\infty$. With pr. $\geq 1-\delta$

$$\mathcal{R}(\hat{D}) - \mathcal{R}_{\mathrm{opt}} \leq 4L\alpha\sqrt{\frac{S_\infty(\mathcal{E})\,(2+\ln K)}{n}} + L\alpha K\sqrt{\frac{2\pi\widehat{S}_1}{T}} + \sqrt{\frac{8\ln\frac{4}{\delta}}{T}}$$

Choosing $\mu_t(x,y) = p(x)\delta(\langle w_t, x\rangle - y)$ and taking $n \to \infty$, we recover a previous bound for sparse coding [Maurer and P. 2010]

$$\mathop{\mathbb{E}}_{w\sim\rho}\big[g(w;\widehat{D})\big] - \min_{D\in\mathcal{D}_K}\mathop{\mathbb{E}}_{w\sim\rho}\big[g(w;D)\big] \leq 2\alpha(1+\alpha)K\sqrt{\frac{2\pi}{T}} + \sqrt{\frac{8\ln\frac{4}{\delta}}{T}}$$

where $g(w;D) := \min_{\|\gamma\|_1\leq\alpha}\|w - D\gamma\|_2^2$

Nonlinear Extension

- Let x, D_k be elements of a Hilbert space $\mathbb{H}$

  $$f(x) = \langle D\gamma, x \rangle = \sum_{k=1}^{K} \gamma_k \langle d_k, x \rangle = \sum_{k=1}^{K} \gamma_k f_k(x)$$

  look for f_k to be in some RKHS

- Multilinear neural networks with share internal weights

  $$f_t(x) = \langle \gamma_t, h(D^q h(D^{q-1}) \cdots h(D^1 x) \cdots)) \rangle$$

  where h is an "activation" function, e.g. sigmoid

Conclusions

- Presented a method to learn a dictionary which allows for sparse representations of a set of linear predictors
- Learning bounds in both the context of MTL and LTL and quantified advantage over ITL
- Learning method matches performance of Lasso with best a priori-known dictionary when $T \to \infty$ and standard sparse coding when $n \to \infty$
- Future directions: faster rates under stronger conditions? nonlinear extensions? efficient algorithms?

Bonus 1: Multilinear MTL

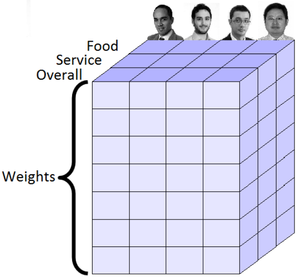
- Problem formulation
- Modelling low rank tensors
- Convex relaxation

[B. Romera-Paredes and M. Pontil. A new convex relaxation for tensor completion. NIPS 2013]

[B. Romera-Paredes, H. Aung, N. Bianchi-Berthouze, M. Pontil. Multilinear multitask learning. ICML 2013]

Multilinear Models

- Example: predict rating given to different aspects of a restaurant by different critics



- Tensor completion from few entries

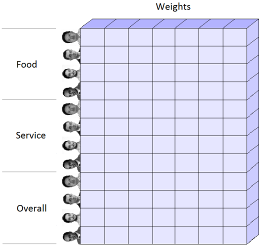
- MTL: tasks' regression vectors are "vertical" fibers of the tensor

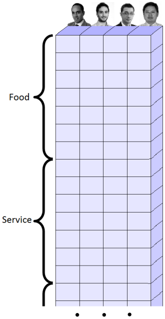
Low Rank Tensors

- Tensor $\mathcal{W} \in \mathbb{R}^{p_1 \times \cdots \times p_N}$, with entries $\mathcal{W}_{i,j,k,\ldots}$
- Penalize average rank of the *matricizations* of $\mathcal{W}$:

$$R(\mathcal{W}) := \sum_n Rank(W_{(n)})$$

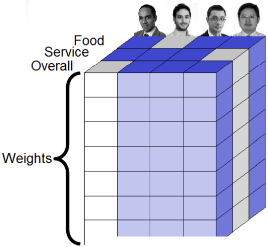
$W_{(n)}$: n-th matricization of the tensor, for example:





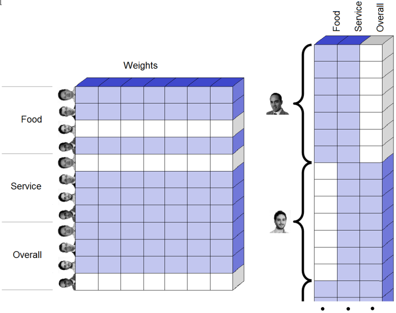
0-Shot Transfer Learning

Learning tasks for which no training instances are provided



🟦 Training instances available

⬜ No training instances available



Convex Relaxation

- Standard convex proxy for average rank:

$$\|\boldsymbol{\mathcal{W}}\|_{\mathrm{tr}} = \sum_n \|W_{(n)}\|_{\mathrm{tr}}$$

- Convex lower bound on the set $\mathcal{G}_\infty = \left\{ \max_n \|W_{(n)}\|_\infty \leq 1 \right\}$

- Relaxation is not tight! Can we do better?

- Yes, relax on unit ball $\mathcal{G}_2 = \{\|\mathcal{W}\|_2 \leq 1\}$

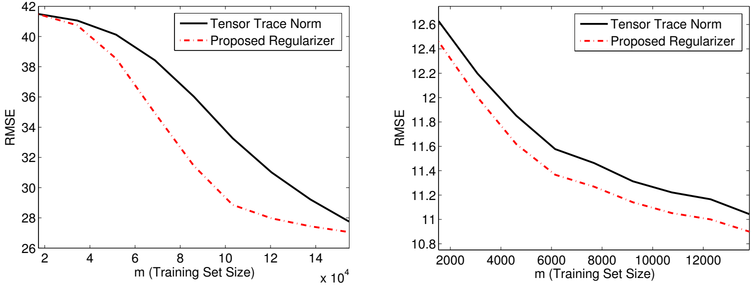
$$\Omega_\alpha(\boldsymbol{\mathcal{W}}) = \sum_n \omega_\alpha(W_{(n)})$$

  ω_α : convex envelope of *matrix rank* on L2 unit ball of radius α

Theorem. There exists $\boldsymbol{\mathcal{W}} \in \mathcal{G}_\infty$ such that $\Omega_\alpha(\boldsymbol{\mathcal{W}}) > \|\boldsymbol{\mathcal{W}}\|_{\mathrm{tr}}$ for $\alpha = \sqrt{\min_n p_n}$. In addition, if $\|\boldsymbol{\mathcal{W}}\|_2 \leq 1$ then $\Omega_1(\boldsymbol{\mathcal{W}}) \geq \|\boldsymbol{\mathcal{W}}\|_{\mathrm{tr}}$

Tensor Completion Experiments

Video compression (Left) and RC Dataset (Right):



Bonus 2: Sharing across Latent Subcategories

- Problem formulation
- Learning subcategories
- Multitask learning formulation
- Experiments

[D. Stamos, S. Martelli. M. Nabi, A. McDonald, V. Murino, M. Pontil. Learning with dataset bias in latent subcategory models. CVPR 2015]

Latent subcategory models

- In computer vision, an object class (e.g. pedestrian) often is a mixture of subcategories which provide "fine granularity" (e.g. "frontal", "side", "thin", "fat", etc.)
- Each latent subcategory k is associated with a weight vector, $w_k \in \mathbb{R}^d$, inducing a subclassifier. We separate the object class from the background class by the classification rule

$$\max_{k=1}^{K} \langle w_k, x \rangle$$

- The positive class is a union of half-spaces: if at least one subclassifier give a positive classification we output a positive classification (note classifiers are not mutually exclusive, e.g. "frontal" and "thin" class)

Latent subcategory models (cont.)

We find the weight vectors $\{w_k\}$ by minimizing

$$\sum_{i=1}^{n} \ell(y_i \max_{k=1}^{K} \langle w_k, x_i \rangle) + \lambda \sum_{k=1}^{K} \|w_k\|^2$$

Nonconvex problem, attempt to solve it by alternate minimization

Initialization heuristic: cluster positive points with K-means. Let P_k be set of positive point in cluster k. Then solve convex problem

$$\sum_{i=1}^{n} \sum_{i \in P_k} \ell(\langle w_k, x_i \rangle) + \sum_{i \in N} \ell(- \max_{k} \langle w_k, x_i \rangle) + \lambda \sum_{k=1}^{K} \|w_k\|^2$$

If the clusters have a small variance and are well separated the heuristic gives a good suboptimal solution (see paper for details)

Sharing across Latent Subcategories

- Dataset bias problem in vision [Ponce et al. 2006; Torralba and Efros 2011]: it may be harmful to train on the concatenation of all datasets!

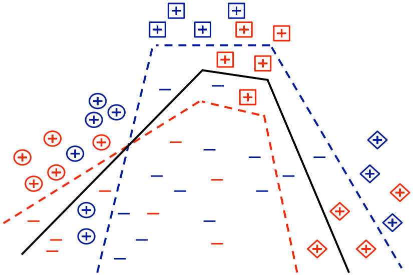
- Multitask learning formulation, based on extension of [Evgeniou and P. 2004; Khosla et al. ECCV 2012]

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\ell(y_{ti}\max_{k=1}^{K}\langle w_0^k + v_t^k, x_{ti}\rangle) + \lambda_1\sum_{k=1}^{K}\|w_0^k\|^2 + \lambda_2\sum_{t=1}^{T}\sum_{k=1}^{K}\|v_t^k\|^2$$

- In practice it is useful add a term controlling the error of "compound model" (no theoretical explanation)

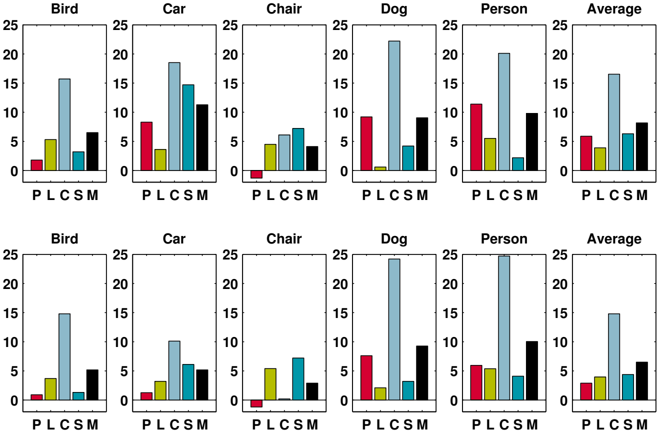
$$\sum_{t=1}^{T}\sum_{i=1}^{n}\ell(y_i\max_{k=1}^{K}\langle w_0^k, x_{ti}\rangle)$$

Sharing across Latent Subcategories (cont.)



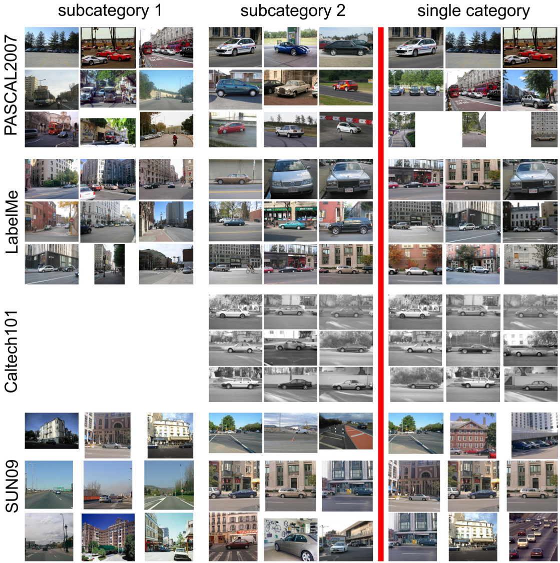
Parameter sharing across datasets can help to train a better subcategory model of the visual world. Here we have two datasets of a class, each of which is divided into three subcategories. The red and blue classifiers are trained on their respective datasets. Our method, in black, both learns the subcategories and undoes the bias inherent in each dataset.

Experiment



Relative improvement of undoing dataset bias LSM vs. the baseline LSM trained on all datasets at once (top) and vs. undoing bias SVM [Khosla2012 et al. 2012] (bottom) on all datasets at once (P: PASCAL, L: LabelMe, C: Caltech101, S: SUN: M: mean)

Experiment



Left and center: top scoring images for subclassifiers w_0^1 and w_0^2 using our method.
Right: top scoring image for single category classifier w_0 from [Khosla et al. 2012]