Large-scale classification with sparse matrix regularization

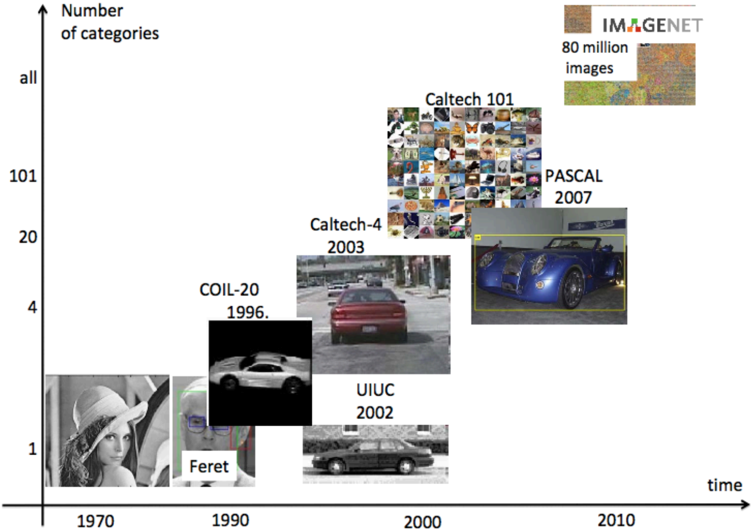
Zaid Harchaoui

LEAR project-team, INRIA

Joint work with Miro Dudik (Yahoo!) and Jerome Malick (CNRS, LJK)

December 6th, 2011

The advent of "big" data



Large-scale supervised learning

Large-scale supervised learning

Let $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n) \in \mathbb{R}^d \times \mathcal{Y}$ be a set of i.i.d. labelled training data

$$\underset{\mathbf{W} \in \mathbb{R}^{d \times k}}{\text{Minimize}} \quad \lambda\,\Omega(\mathbf{W}) + \frac{1}{n}\sum_{i=1}^{n} L\big(y_i, \mathbf{W}^T \mathbf{x}_i\big) \tag{1}$$

- Multi-output regression : $\mathcal{Y} = \mathbb{R}^k$
- Multi-class classification : $\mathcal{Y} = \{0, 1\}^k$

Problem : minimizing such objectives in the large-scale setting

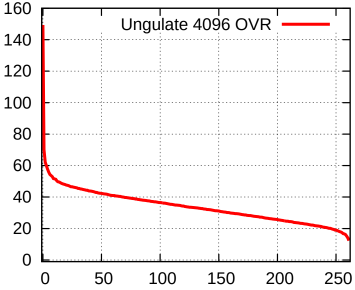
$$\min(d, k) \gg 1 \tag{2}$$

Motivation

Image classification with large number of classes

- Embedding assumption : classes may embedded in a low-dimensional subspace of the feature space.

  Example :



- Computational efficiency : training time and test time efficiency require sparse matrix regularizers

Learning with trace-norm penalty

Supervised learning with trace-norm regularization penalty

Let $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n) \in \mathbb{R}^d \times \mathcal{Y}$ be a set of i.i.d. labelled training data ; e.g. $\mathcal{Y} = \{0,1\}^k$ for multi-class classification

$$\underset{\mathbf{W} \in \mathbb{R}^{d \times k}}{\text{Minimize}} \quad \lambda \Omega(\mathbf{W}) + \frac{1}{n} \sum_{i=1}^{n} L\left(y_i, \mathbf{W}^T \mathbf{x}_i\right) \qquad \text{(P1)}$$

Important case : Trace-norm penalty

$$\Omega_{\mathsf{trace}}(\mathbf{W}) = \|\sigma(\mathbf{W})\|_1 \qquad (3)$$

where $\sigma(\mathbf{W}) = \{\sigma_1(\mathbf{W}), \ldots, \sigma_{\min(d,k)}(\mathbf{W})\}$ singular spectrum

Learning with trace-norm penalty

Supervised learning with trace-norm regularization penalty

Let $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n) \in \mathbb{R}^d \times \mathcal{Y}$ be a set of i.i.d. labelled training data ; e.g. $\mathcal{Y} = \{0, 1\}^k$ for multi-class classification

$$\underset{\mathbf{W} \in \mathbb{R}^{d \times k}}{\text{Minimize}} \quad \underbrace{\lambda \Omega(\mathbf{W})}_{non-smooth} + \underbrace{\frac{1}{n} \sum_{i=1}^{n} L\left(y_i, \mathbf{W}^T \mathbf{x}_i\right)}_{smooth} \qquad \text{(P1)}$$

Important case : Trace-norm penalty

$$\Omega_{\mathsf{trace}}(\mathbf{W}) = \|\sigma(\mathbf{W})\|_1 \qquad (4)$$

where $\sigma(\mathbf{W}) = \{\sigma_1(\mathbf{W}), \ldots, \sigma_{\min(d,k)}(\mathbf{W})\}$ singular spectrum

Trace-norm penalty

Properties of trace-norm penalty

- Non-differentiable penalty, just as the vector ℓ_1-norm
- Convex relaxation of the rank$(\mathbf{W})$ penalty
- Enforces a low-rank structure on $\mathbf{W}$

Possible approaches

- "Blind" approach : subgradient, ε-subgradient, bundle method $\rightarrow$ slow convergence rate
- Alternating minimization $\rightarrow$ not-convex
- Composite minimization : (accelerated) proximal gradient $\rightarrow$ good convergence rate in $O(1/t)$

Composite minimization algorithms

Strengths of composite minimization algorithms

- Attractive algorithms when proximal operator is cheap, as e.g. for vector ℓ_1-norm
- Highly accurate with finite-time accuracy guarantees

Weaknesses of composite minimization algorithms

- Inappropriate when proximal operator is expensive to compute
- Heavily sensitive to design matrix conditioning

Situation with trace-norm

- proximal operator corresponds to singular value thresholding, requiring an SVD running in $O(kd^2)$ in time $\rightarrow$ impractical for large-scale problems

Proposed approach : coordinate descent

We want an algorithm with no SVD...
Let's get inspiration from ℓ_1 case...

Coordinate descent algorithms

- efficient and scalable algorithms
- competitive with composite minimization algorithms
- more robust to ill-conditioned design matrices

Open problem for trace-norm

- unclear how to devise one in the matrix case : what are the "coordinates"?
- good coordinates are the ones along the (unknown) singular vectors basis of the minimizer...life is unfair

Our solution : Lifting to an infinite-dimensional space

Reformulation of trace-norm

The trace-norm is the smallest ℓ_1-norm of the weight vector associated with an atomic decomposition onto rank-one subspaces

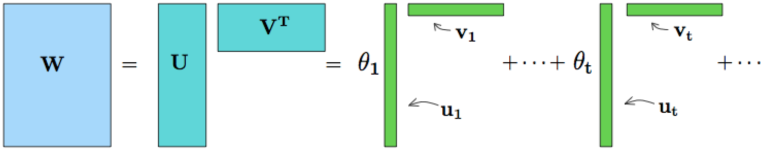
$$\|\sigma(\mathbf{W})\|_1 = \inf_{\theta} \left\{ \|\theta\|_1 \;\mid \exists N, \theta_i > 0, \mathbf{M}_i \in \mathcal{M} \text{ with } \mathbf{W} = \sum_{i=1}^{N} \theta_i \mathbf{M}_i \right\}$$

where the generating family is

$$\mathcal{M} = \{\mathbf{u}\mathbf{v}^T \mid \mathbf{u} \in \mathbb{R}^d, \, \mathbf{v} \in \mathbb{R}^{\mathcal{Y}}, \, \|\mathbf{u}\|_2 = \|\mathbf{v}\|_2 = 1\}$$

Lifting to an infinite-dimensional space

The trace-norm is the smallest ℓ_1-norm of the weight vector associated with an atomic decomposition onto rank-one subspaces



$$\|\sigma(\mathbf{W})\|_1 \;=\; \inf_{\theta}\left\{\|\theta\|_1 \;\mid \exists N, \theta_i > 0, \mathbf{M}_i \in \mathcal{M} \text{ with } \mathbf{W} = \sum_{i=1}^{N} \theta_i \mathbf{M}_i\right\}$$

$$\mathcal{M} \;=\; \{\mathbf{u}\mathbf{v}^T \mid \mathbf{u} \in \mathbb{R}^d,\, \mathbf{v} \in \mathbb{R}^{\mathcal{Y}},\, \|\mathbf{u}\|_2 = \|\mathbf{v}\|_2 = 1\}$$

Landing back

Assumptions

- $\mathcal{M}$ is a compact subset of $\mathbb{R}^{d\times k}$, 0 lies in the interior of $\mathcal{B} = \operatorname{conv}\mathcal{M}$.
- For any $y \in \mathcal{Y}$, the loss function $L(y,\cdot)$ is convex, bounded below, and has Lipchitz-continuous derivative

Notations

- Denote $\mathcal{I}$ the index set spanning the set of rank-one matrices in $\mathcal{M}$, $\Theta := \{\boldsymbol{\theta} \in \mathbb{R}^{\mathcal{I}} \mid \operatorname{supp}\boldsymbol{\theta}$ is finite$\}$
- Denote
$$\phi_\lambda(\mathbf{W}) := \lambda\Omega(\mathbf{W}) + \frac{1}{n}\sum_{i=1}^{n} L\left(y_i, \mathbf{W}^T\mathbf{x}_i\right)$$

Equivalence

We prove the equivalence of the infinite-dimensional formulation.

Landing back

Theorem

1. the function $\psi_\lambda(\cdot)$ is convex and differentiable, where

$$\psi_\lambda(\boldsymbol{\theta}) := \lambda \sum_{j \in \operatorname{supp} \boldsymbol{\theta}} \theta_j + \frac{1}{n} \sum_{i=1}^{n} L\left(y_i, \mathbf{W}_{\boldsymbol{\theta}}^T \mathbf{x}_i\right) \ .$$

2. for all $\boldsymbol{\theta} \in \Theta^+$, $\phi_\lambda(\mathbf{W}_{\boldsymbol{\theta}}) \leq \psi_\lambda(\boldsymbol{\theta})$
3. the two problems are equivalent, i.e.

$$\hat{\boldsymbol{\theta}} \in \operatorname*{Arg\,min}_{\boldsymbol{\theta} \in \Theta^+} \psi_\lambda(\boldsymbol{\theta}) \quad \text{if and only if} \quad \mathbf{W}_{\hat{\boldsymbol{\theta}}} \in \operatorname*{Arg\,min}_{\mathbf{W} \in \mathbb{R}^{d \times k}} \phi_\lambda(\mathbf{W}).$$

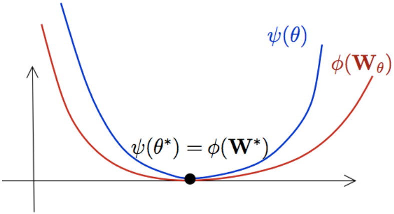
Landing back

Theorem

1. the function $\psi_\lambda(\cdot)$ is convex and differentiable, where

$$\psi_\lambda(\boldsymbol{\theta}) := \lambda \sum_{j \in \operatorname{supp} \boldsymbol{\theta}} \theta_j + \frac{1}{n} \sum_{i=1}^{n} L\left(y_i, \mathbf{W}_{\boldsymbol{\theta}}^T \mathbf{x}_i\right) .$$

2. for all $\boldsymbol{\theta} \in \Theta^+$, $\phi_\lambda(\mathbf{W}_{\boldsymbol{\theta}}) \leq \psi_\lambda(\boldsymbol{\theta})$
3. the two problems are equivalent, i.e.



Coordinate descent

Coordinate descent algorithm

Fix $\varepsilon > 0$ and set $\boldsymbol{\theta}_0 = 0$
Loop on t

1) Use oracle to get $j_t = \operatorname{Arg\,min}_{j \in I} \langle \nabla \psi_\lambda(\boldsymbol{\theta}_t), \mathbf{M}_j \rangle$
2) Set $e_t = e_{j_t}$ and $g_t = \partial_{j_t} \psi_\lambda(\boldsymbol{\theta}_t)$
3) [case 1] If $g_t \leq -\varepsilon$, $\quad \boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \delta e_t$ with suitable δ
4) [case 2] Else $g_t > -\varepsilon$, $\quad \boldsymbol{\theta}_{t+1} = \min_{\boldsymbol{\theta} \in \mathbb{R}^{\operatorname{supp} \theta_t}} \psi_\lambda(\boldsymbol{\theta}_t)$
5) Terminate if $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t$

End

Coordinate descent

Coordinate descent algorithm

Fix $\varepsilon > 0$ and set $\boldsymbol{\theta}_0 = 0$

Loop on t

1) Use oracle to get $j_t = \operatorname{Arg\,min}_{j \in I} \langle \nabla \psi_\lambda(\boldsymbol{\theta}_t), \mathbf{M}_j \rangle$
2) Set $e_t = e_{j_t}$ and $g_t = \partial_{j_t} \psi_\lambda(\boldsymbol{\theta}_t)$
3) [case 1] If $g_t \leq -\varepsilon$, $\quad \boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \delta e_t$ with suitable δ
4) [case 2] Else $g_t > -\varepsilon$, $\quad \boldsymbol{\theta}_{t+1} = \min_{\boldsymbol{\theta} \in \mathbb{R}^{\operatorname{supp} \theta_t}} \psi_\lambda(\boldsymbol{\theta}_t)$
5) Terminate if $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t$

End

Oracle for coordinate descent

The notion of oracle

- Exact oracle : "machine" that ouputs the steepest descent rank-one matrix "direction" $\mathbf{M}_i = \mathbf{u}_i \mathbf{v}_i^T$

$$\begin{aligned} \operatorname*{Arg\,min}_{i \in \mathcal{I}} \partial_i \psi_\lambda(\boldsymbol{\theta}) &= \operatorname*{Arg\,max}_{i \in \mathcal{I}} \langle \mathbf{M}_i, -\nabla \phi(\boldsymbol{\theta}) \rangle \\ &= \operatorname*{Arg\,max}_{i \in \mathcal{I}} \ \mathbf{u}_i^T \big(-\nabla \phi(\mathbf{W}) \big) \mathbf{v}_i \end{aligned}$$

where

$$\phi(\mathbf{W}_{\boldsymbol{\theta}}) := \frac{1}{n} \sum_{i=1}^{n} L \left(y_i, \mathbf{W}_{\boldsymbol{\theta}}^T \mathbf{x}_i \right) \tag{5}$$

- ε-approximate oracle : "machine" that ouputs a descent rank-one matrix "direction" $\mathbf{M}_i = \mathbf{u}_i \mathbf{v}_i^T$

$$\langle \mathbf{M}_i, -\nabla \phi(\boldsymbol{\theta}) \rangle \leq \max_{i \in \mathcal{I}} \langle \mathbf{M}_i, -\nabla \phi(\boldsymbol{\theta}) \rangle + \varepsilon \tag{6}$$

Oracle for coordinate descent

Oracle for the trace-norm

- Exact oracle : top singular vectors $\mathbf{u}_1$ and $\mathbf{v}_1$ of $-\nabla\phi(\boldsymbol{\theta})$
- ε-approximate oracle :

  approximate singular vectors $\mathbf{u}_1$ and $\mathbf{v}_1$ of $-\nabla\phi(\boldsymbol{\theta})$

  $\hookrightarrow$ obtained by early-stopped power or Lanczos iterations

Coordinate descent

Coordinate descent algorithm

Fix $\varepsilon > 0$ and set $\boldsymbol{\theta}_0 = 0$

Loop on t

1) Use oracle to get $j_t = \operatorname{Arg\,min}_{j \in I} \langle \nabla \psi_\lambda(\boldsymbol{\theta}_t), \mathbf{M}_j \rangle$
2) Set $e_t = e_{j_t}$ and $g_t = \partial_{j_t} \psi_\lambda(\boldsymbol{\theta}_t)$
3) [case 1] If $g_t \leq -\varepsilon$, $\quad \boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \delta e_t$ with suitable δ
4) [case 2] Else $g_t > -\varepsilon$, $\quad \boldsymbol{\theta}_{t+1} = \min_{\boldsymbol{\theta} \in \mathbb{R}^{\operatorname{supp} \theta_t}} \psi_\lambda(\boldsymbol{\theta}_t)$
5) Terminate if $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t$

End

Fast coordinate descent

Acceleration with second-order subspace optimization

- Smooth minimization with box constraints (Step 4)
  $\hookrightarrow$ "Projected" Newton/Quasi-Newton

Running time

- Time-complexity of the oracle : $O(dk)$ up to log-factors

Related ideas

Column-matrix generation and boosting

- Oracle call is similar to a "matrix generation" step
- Similarities with LP-view and subsequent coordinate descent algorithms of boosting

Franke-Wolfe and friends

- Greedy updates are similar to algorithms for solving SDPs with low-rank constraints ; see also (Jaggi & Sulovsky, 2010)

Experimental results

Benchmark

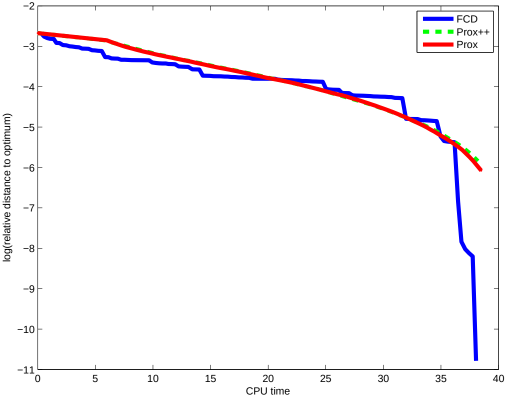
- Inspired by the benchmark of optimization algorithms for sparsity-inducing vector penalties of (Bach et al., 2011)
- Varying scales $n = 100, 500$, varying strength of penalty λ, varying conditioning of design matrix (low-correlation and high-correlation of features)

Optimization accuracy comparison

- Relative accuracy $|(f - f^\star)/f^\star|$ against CPU running time
- Competitors : our algorithn (FCD) and accelerated proximal gradient algorithm (Prox++, FISTA-like implementation)

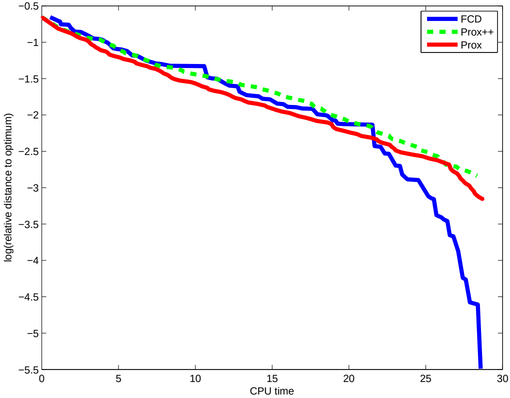
Experimental results

For small-scale, light regularization, and ill-conditioned design.



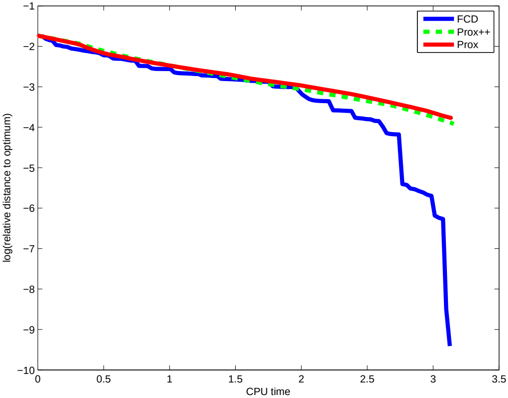
Experimental results

For large-scale, light regularization, and ill-conditioned design.



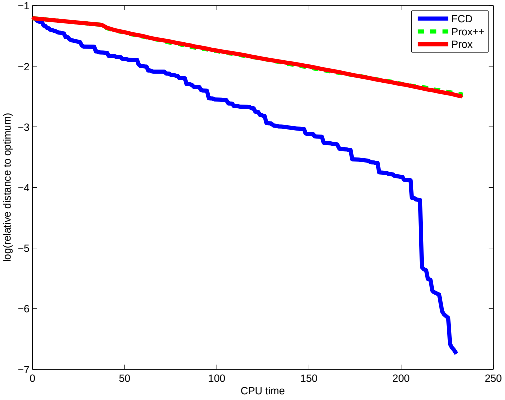
Experimental results

For large-scale, heavy regularization, and ill-conditioned design.



Experimental results

Results for a subset of classes from ImageNet



Experimental results

Benchmark

- Real-world dataset : subset of classes from ImageNet "Vehicles", "Fungus", and "Ungulate"

Some orders of magnitude

- Number of images : $n = 250,000$
- Feature size : $d = 65,000$ (Fisher vectors
- Number of classes : $k = 200$

Experimental results

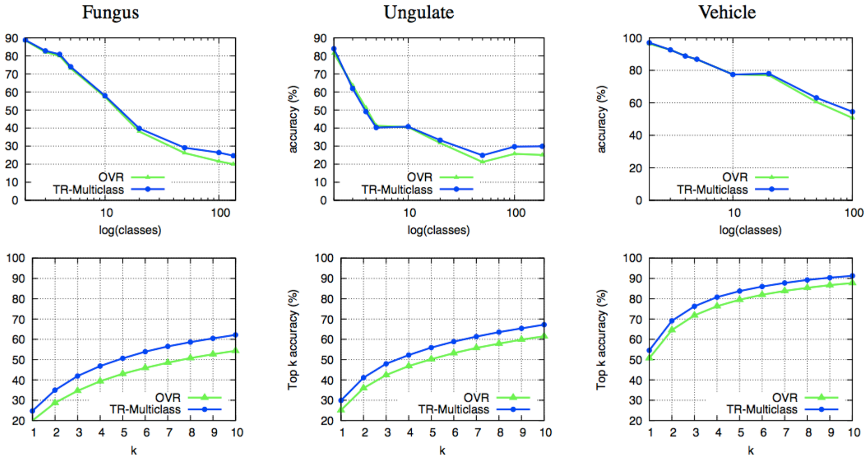
Classification accuracy comparison

- Classification accuracy : top-k accuracy, i.e.

$$\text{Accuracy}_{\text{top}-k} = \frac{\text{\# images whose correct label lies in top-k scores}}{\text{Total number of images}}$$

- Competitors : our approach (TR-Multiclass) and k independently trained one-vs-rest classifiers (OVR)

Experimental results



A posteriori low-dimensional embedding

Conclusion and perspectives

Take-home messages

- the trace-norm is an ℓ_1-norm in some higher-dimensional space
- this fact can be leveraged to design new algorithms

Extensions

- extension to other sparse matrix regularizers : *gauge* regularizers
- risk bounds for learning algorithms with gauge regularization penalties

Conclusion

- efficient alternative of proximal techniques suitable for large-scale problems
- yes, we can build coordinate descent algorithms even for sparse matrix regularizers

The rise of statistical machine learning as an academic discipline

Roots and interactions of statistical machine learning

- Roots : artificial intelligence, statistics, optimization, theoretical computer science, signal processing
- Interactions : computer vision, audio, text, bioinformatics, and many others

Statistical machine learning

- statistical machine learning is a (growing) academic discipline, emancipated from its roots, with its own theory, methodology, and applications.

Interesting open issues

Open scientific issues

- Towards "vegan learning" : close the gap to "raw" data for learning algorithms
- Towards true COLT : more theoretical *computational learning* and more computational *learning theory*