Sparse regularization path by differential inclusion

Wotao Yin (UCLA Math)

joint with: Stanley Osher, Ming Yan (UCLA)
Feng Ruan, Jiechao Xiong, Yuan Yao (Peking U)

ICERM Approximation, Integration, and Optimization Workshop
October 2, 2014

Background

- Assume vector $x^* \in \mathbb{R}^n$ is **sparse, unknown**
- Goal: Recover x^* from

$$b = Ax^* + \epsilon$$

  where $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ and ϵ is **unknown noise**.
- Consider $m \ll n$, the **under-determined** case

Background: Regularization by optimization

Examples:

- convex: LASSO: $\text{minimize}\, \lambda\|x\|_1 + \frac{1}{2m}\|Ax - b\|_2^2$
- nonconvex: SCAD, ℓ_p-seminorm minimization $p \in (0, 1)$

Optimization approach:

- **convex** penalty: avoid overfitting, tractable, lead to **bias**
- **nonconvex** penalty: may works better, but performance **unpredictable**

They have a **tuning parameter**, the best choice of which is often unknown

So, we need **model selection:** vary the parameter values, solve many instances, and then pick the best one

Background: Regularization path by an algorithm

Algorithmic regularization:

- An algorithm generates a regularization path. (Points on the path may not minimize an energy function)
- Model selection is done by deciding when to stop at a time (for continuous dynamic) or stop at an iteration (for discrete update)

Examples:

- LASSO/LARS: solve parameterized LASSO KKT conditions
- xMP family

This talk introduces a **continuous regularization path** by **differential inclusions**, with

- recovery guarantees
- fast implementation
- and generalization to other structured solution

Introduced: two Inverse Scale Space (ISS) Dynamics

- Let $x(t), p(t) \in \mathbb{R}^n$ by primal-dual regularization path. t is time.
- **Bregman ISS dynamic**: $\{x(t), p(t)\}_{t \geq 0}$ is governed by

$$\dot{p}(t) = \frac{1}{m} A^T (b - Ax(t)),$$
$$p(t) \in \partial \|x(t)\|_1.$$

  Initial solution $x(0) = p(0) = 0$.

- **Linearized Bregman ISS dynamic**: $\{x(t), p(t)\}_{t \geq 0}$ is governed by

$$\dot{p}(t) + \frac{1}{\kappa} \dot{x}(t) = \frac{1}{m} A^T (b - Ax(t)),$$
$$p(t) \in \partial \|x(t)\|_1.$$

  Initial solution $x(0) = p(0) = 0$.

- For well-defined path (and uniqueness), make technical assumptions:
  - $p(t)$ is right *continuously differentiable*, and
  - $x(t)$ is right *continuous*

Generalization

Given any **convex optimization** model:

$$\underset{x}{\text{minimize}} \;\; r(x) + t \cdot f(x)$$

one can generate the related **Bregman ISS** model:

$$\dot{p}(t) = -f'(x),$$
$$p(t) \in \partial r(x(t)).$$

where

- r is **convex regularization**: weighted ℓ_1, $\ell_{1,2}$, nuclear norm, and so on; can incorporate nonnegative or box constraints as indicator functions
- f is **convex fitting**: square loss, logistic loss, etc.

Linearized Bregman ISS: add a strongly convex function to r.

Major claims for Bregman ISS applied to ℓ_1

The solution path $\{x(t), p(t)\}_{t \geq 0}$:

- $x(t)$ is **sparse**, if $p \in \partial \|x\|_1 \cap \mathcal{R}(A^T)$ and A is **fat**, w.h.p.
- $x(t)$ is **less biased** than LASSO, better than LASSO+debiasing
- path can be piece-wise **computed very quickly**
- **sign-consistency**: $\operatorname{sign}(x(t)) = \operatorname{sign}(x^*)$ at some t under conditions

In less technical languages, the new method

- recovers sparse nonzero elements like ℓ_1 but avoids its bias
- can generate a regularization path much quicker than ℓ_1
- whereas ℓ_1 extends, it does too

Background: ℓ_1 subgradient

- **Subdifferential** of convex function f

$$\partial f(y) = \{p : f(x) \geq f(y) + \langle p, x - y \rangle,\ \forall y \in \mathrm{dom} f\}.$$

  Each $p \in \partial f(y)$ is a subgradient of f at y.

- Subdifferential of $|\cdot|$:

$$\partial |x_i| = \begin{cases} \{1\}, & x_i > 0; \\ [-1, 1], & x_i = 0; \\ \{-1\}, & x_i < 0. \end{cases}$$

  $\Longrightarrow$ let $p_i \in \partial |x_i|$, then

$$x_i \begin{cases} \geq 0, & \text{if } p_i = 1; \\ = 0, & \text{if } p_i \in (-1, 1); \\ \leq 0, & \text{if } p_i = -1. \end{cases}$$

Sparsity and ℓ_1 subgradient

Although $x \leftrightarrow p$ is not one-one, it is in some cases. When x_i is nonzero, p_i must equal its sign; when $p_i \in (-1, 1)$, x_i has to be zero.

p is like an array of 3-position switches: -1, $(-1, 1)$, $+1$

Toy example 1

Consider:

$$b = ax + \epsilon,$$

where b, a, x are *strictly positive scalars*.

Bregman ISS:

- start: $x(0) = 0$ and $p(0) = 0$
- stage 1: p evolves before reaching 1, meanwhile x stays 0.

$$\dot{p} = a(b - ax) = ab \quad \Rightarrow \quad p(t) = (ab)t$$

- stage 2: p reaches 1 at $t = 1/(ab)$, but cannot exceed 1, so $\dot{p}(t) \leq 0$ and thus $x(t) \neq 0$. Right continuity assumption makes $\dot{p}(t) < 0$ impossible as it will immediately make $x(t+) = 0$. Therefore,

$$\text{at } t \geq 1/(ab), \quad 0 = \dot{p}(t) = a(b - ax(t)) \;\Rightarrow\; x(t) = b/a,\; p(t) = 1.$$

Once $p(t) = 1$ "switch is on", the signal $x(t) = b/a$ immediately pops out!

LASSO:

$$x(t) = \arg\min_x \ |x| + \frac{t}{2}|ax - b|^2$$

$\Rightarrow$ optimality condition:

$$0 = p + ta(ax - b), \quad p \in \partial|x|$$

$\Rightarrow$ solution:

$$p(t) = \begin{cases} (ab)t, \\ 1, \end{cases} \qquad x(t) = \begin{cases} 0, & t \in [0, \frac{1}{ab}), \\ \frac{b}{a} - \frac{1}{ta^2}, & t \in [\frac{1}{ab}, \infty). \end{cases}$$

In this example, **LASSO** has the same $p(t)$ path but a different $x(t)$ path. **LASSO's $x(t)$ path reduced the signal strength**.

ℓ_1 subgradient and sparsity

Faces of $\partial\|x\|$

- ℓ_1 subdifferential: $\partial\|x\|_1 = \partial|x_1| \times \cdots \times \partial|x_n|$.
- The image of $\partial\|x\|_1$ is $[-1,1]^n$
- Let $p \in \partial\|x\|_1$. For $x_i \neq$, p_i must equal ± 1 and is thus *exposed*.
- More p_i exposed $\Leftrightarrow$ p lies on a low-dim face of $[-1,1]^n$

Observation:

vector x is sparse $\Leftrightarrow$ few $p_i = \pm 1$ $\Leftrightarrow$ $p \notin$ a low-dim face of $[-1,1]^n$

- If matrix A is **fat** (or A^T is **thin**), then $\mathcal{R}(A^T)$ is a **small subspace**
- A is random and $p \in \partial\|x\|_1 \cap \mathcal{R}(A^T)$

  $\Rightarrow$ p is *unlikely* on a low-dim face of $[-1,1]^n$

  $\Rightarrow$ very few $p_i = \pm 1$

  $\Rightarrow$ sparse x

Bregman ISS update:

$$\dot{p}(t) = \frac{1}{m} A^T(b - Ax(t)), \qquad p(t) \in \partial\|x(t)\|_1,$$

$\Rightarrow$ $p \in \partial\|x\|_1 \cap \mathcal{R}(A^T)$

Conclusion: if A is fat, then $x(t)$ is typically sparse.

Toy example 2

- $x \in \mathbb{R}^n$, measurement b is a scalar:

$$b = \mathbf{a}^T x + \epsilon \in \mathbb{R}$$

  Suppose $a_1 = 1 > a_2 \geq \ldots \geq a_n > 0$ and $b > 0$. w.o.l.g.

- **Bregman ISS** solution:

$$x_1(t) = \begin{cases} 0, & t < 1/b, \\ b, & t \geq 1/b. \end{cases}$$

$$x_2(t) = \cdots = x_n(t) = 0, \quad t \geq 0.$$

- **LASSO** solution:

$$x_1(t) = \begin{cases} 0, & t < 1/b, \\ b - \frac{1}{t}, & t \geq 1/b. \end{cases}$$

$$x_2(t) = \cdots = x_n(t) = 0, \quad t \geq 0.$$

- Both solutions are sparse. Like before, LASSO solution is a *strength–reduced* signal, which is not good.

Oracle estimator

- Unknown $S := \mathrm{supp}(x^*)$ is disclosed by an *oracle*
- *Oracle estimator* is the least-squares solution restricted to S

$$\tilde{x}_S^* = \arg\min\{\frac{1}{2m}\|Ax - b\|_2^2 : \mathrm{supp}(x) = S\}$$

- Define: submatrix A_S of A and $\Sigma_m := \frac{1}{m} A_S^T A_S$. The *oracle estimate*

$$\tilde{x}_S^* = \Sigma_m^{-1}(\frac{1}{m} A_S^T b) = x_S^* + \frac{1}{m}\Sigma_m^{-1} A_S^T \epsilon$$

has **oracle properties:**

  - **consistency**: $\mathrm{supp}(\tilde{x}_S^*) = S$
  - **normality:** $\tilde{x}_S^* \sim \mathcal{N}(x_S^*, \frac{\sigma^2}{m}\Sigma_m^{-1})$. In particular, $\mathbb{E}[\tilde{x}_S^*] = x_S^*$ **unbiased**.

LASSO fails to have oracle properties

Tibshirani'96 (LASSO) and Chen-Donoho-Saunders'96 (BPDN):

$$\text{minimize } \|x\|_1 + \frac{t}{2m}\|Ax - b\|_2^2$$

Optimality conditions:

$$p = \frac{t}{m}A^T(b - Ax), \qquad p \in \partial\|x\|_1.$$

Pros:

- $p \in \partial\|u\|_1 \cap \mathcal{R}(A^T)$, so x is sparse
- efficient solvers for fixed t
- sign–consistency under conditions

Cons:

- $x(t)$ is **always biased!**
- computing for **many values of t is slow or inaccurate**

The LASSO bias

At some t, **suppose** $\mathrm{supp}(\tilde{x}^{\mathrm{LASSO}}) = \mathrm{supp}(x^*) =: S$.

Then,

$$\tilde{x}_S^{\mathrm{LASSO}} = \underbrace{x_S^* + \frac{1}{m}\Sigma_m^{-1} A_S^T \epsilon}_{\text{oracle estimate}} \underbrace{-\frac{1}{t}\Sigma_m^{-1}\mathrm{sign}(\tilde{x}_S^{\mathrm{LASSO}})}_{\text{bias}}.$$

The bias is caused by the part ℓ_1-norm applied to x_S.

LASSO's ℓ_1 minimization enforces $x_{S^c} = 0$ but hurts the signals in x_S!

Debias LASSO

Two approaches:

- Exact debias: Add $\frac{1}{t}\Sigma_m^{-1}\mathrm{sign}(\tilde{x}_S^{\mathrm{LASSO}})$ to $\tilde{x}_S^{\mathrm{LASSO}}$
- Pseudo debias:

$$\underset{x}{\text{minimize}} \|Ax - b\|^2 \quad \text{subject to } \mathrm{supp}(x) = \mathrm{supp}(\tilde{x}^{\mathrm{LASSO}})$$

  It's "psuedo" since the debiased solution may have changed signs.

Issues:

- **extra computation**
- bias has negative effect on the signs of $\tilde{x}^{\mathrm{LASSO}}$, which is not removed by debiasing, therefore:
  $\tilde{x}_S^{\mathrm{LASSO}}$ often **misses small signals**, which are not recovered by debiasing.
- **not work for problems with "continuous support"** (e.g., in low-rank matrix recovery)

Bregman ISS: a "debiasing" interpretation

- LASSO optimality condition:

$$p = \frac{t}{m} A^T (b - Ax)$$

- Differentiate w.r.t. $t \Rightarrow$

$$\dot{p} = \frac{1}{m} A^T (b - A(t\dot{x} + x))$$

- **Important:** recognize that $t\dot{x} + x$ is the *debiased* LASSO solution!
- **Idea:** replace $t\dot{x} + x$ by $x \;\Rightarrow$ Bregman ISS:

$$\dot{p} = \frac{1}{m} A^T (b - Ax)$$

- No bias is ever introduced!
- Note: Bregman ISS$\neq$LASSO+debiasing. Bregman ISS is better and faster.

Compute the Bregman ISS path

Theorem

The solution path to

$$\dot{p}_+(t) = \frac{1}{m} A^T(b - Ax(t)), \quad p(t) \in \partial \|x(t)\|_1$$

with initial conditions $t_0 = 0$, $p(0) = 0$, $x(0) = 0$, *is given* ***piece-wise*** *by:*

- *for* $k = 1, 2, \ldots, K$
  - $p(t)$ *is* ***piece-wise linear***

$$p(t) = p(t_{k-1}) + \frac{t - t_{k-1}}{m} A^T(b - Ax(t_{k-1})), \quad t \in [t_{k-1}, t_k],$$

  where $t_k := \sup\{t > t_{k-1} : p(t) \in \|x(t_{k-1})\|_1\}$.

  - $x(t) = x(t_{k-1})$ *is* ***piece-wise constant*** *for* $t \in [t_{k-1}, t_k)$*; if* $t_k \neq \infty$,

$$x(t_k) = \arg\min_u \|Au - b\|_2^2 \quad \text{subject to } u_i \begin{cases} \geq 0, & p_i(t_k) = 1, \\ = 0, & p_i(t_k) \in (-1, 1), \\ \leq 0, & p_i(t_k) = -1. \end{cases}$$

Faster alternative: Linearized Bregman ISS

$$\dot{p}(t) + \frac{1}{\kappa}\dot{x}(t) = \frac{1}{m}A^T(b - Ax(t)),$$
$$p(t) \in \partial\|x(t)\|_1.$$

- Solution is **piece-wise smooth, closed form**.
- It approximates Bregman ISS. Converges to the Bregman ISS solution exponentially fast in κ
- Reduce to one nonlinear ODE:

$$\dot{z}(t) = \frac{1}{m}A^T(b - \kappa A\,\mathrm{shrink}(z(t))).$$

Insight: The mapping $z(t) = p(t) + \frac{1}{\kappa}x(t)$ is one-one. Given $z(t)$, recover

$$x(t) = \kappa\,\mathrm{shrink}(z(t)), \quad p(t) = z(t) - \frac{1}{\kappa}x(t),$$

where

$$\mathrm{shrink}(u) = \mathbf{prox}_{\|\cdot\|_1}(u) = \arg\min_y \|y\|_1 + \frac{1}{2}\|y - u\|_2^2.$$

Discrete Linearized Bregman Iteration

- Nonlinear ODE (from last slide):

$$\dot{z} = \frac{1}{m} A^T (b - \kappa A \operatorname{shrink}(z(t))).$$

- Forward Euler:

$$z^{k+1} = z^k + \frac{\alpha_k}{m} A^T (b - A \underbrace{(\kappa \operatorname{shrink}(z^k))}_{x^k})$$

- Easy to parallelize for **very large dataset**. For example:

$$A = [A_1 \ A_2 \ \cdots \ A_L], \quad \text{where } A_\ell \text{ is } \textbf{distributed}$$

Distributed implementation:

$$\text{for } \ell = 1, \ldots, L \text{ in parallel: } \begin{cases} z_\ell^{k+1} = z_\ell^k + \frac{\alpha_k}{m} A_\ell^T (b - w^k) \\ w_\ell^{k+1} = \kappa A_\ell \operatorname{shrink}(z_\ell^{k+1}) \end{cases}$$

$$\text{all-reduce sum: } w^{k+1} = \sum_{\ell=1}^{L} w_\ell^{k+1}.$$

Comparison to ISTA iteration for LASSO

- Linearized Bregman (LB) iteration:

$$z^{k+1} = z^k - \frac{\alpha_k}{m} A^T (A(\kappa \operatorname{shrink}(z^k)) - b)$$

- ISTA (forward-backward splitting, FPC, SpaSRA, ...) iteration:

$$x^{k+1} = \operatorname{shrink}(x^k - \frac{\alpha_k}{m} A^T (Ax^k - b), \lambda)$$

Comparison:

- ISTA: intermediate x^k is dense, solves LASSO for fixed λ as $k \to \infty$
- LBreg: intermediate x^k is sparse (useful as a regularization path) as $k \to \infty$, solves:

$$\text{minimize } \|x\|_1 + \frac{1}{2\kappa}\|x\|^2 \quad \text{subject to } Ax = b,$$

with **exact penalty property:** sufficiently large κ gives $\|x\|_1$ minimizer

Comparison to orthogonal matching pursuit (OMP)[1]

OMP: start with index set $\mathcal{S} = \emptyset$ and vector $x = 0$;

iterate

1. compute residual vector $A^*(b - Ax)$, add its largest entry to $\mathcal{S}$
2. set $x \leftarrow \arg\min \|b - Ax\|_2^2 \quad \text{subject to } x_i = 0 \ \ \forall i \notin \mathcal{S}$.

Differences:

- OMP: increase index set $\mathcal{S}$ (OMP variants evolve $\mathcal{S}$ in other ways)
- ISS: evolves $p \in \|x\|_1$, which encodes more information

[1] Mallat-Zhang'93, Tropp-Gilbert'07

Generalization (once again)

Bregman ISS model:

$$\dot{p}(t) = -f'(x),$$
$$p(t) \in \partial r(x(t)).$$

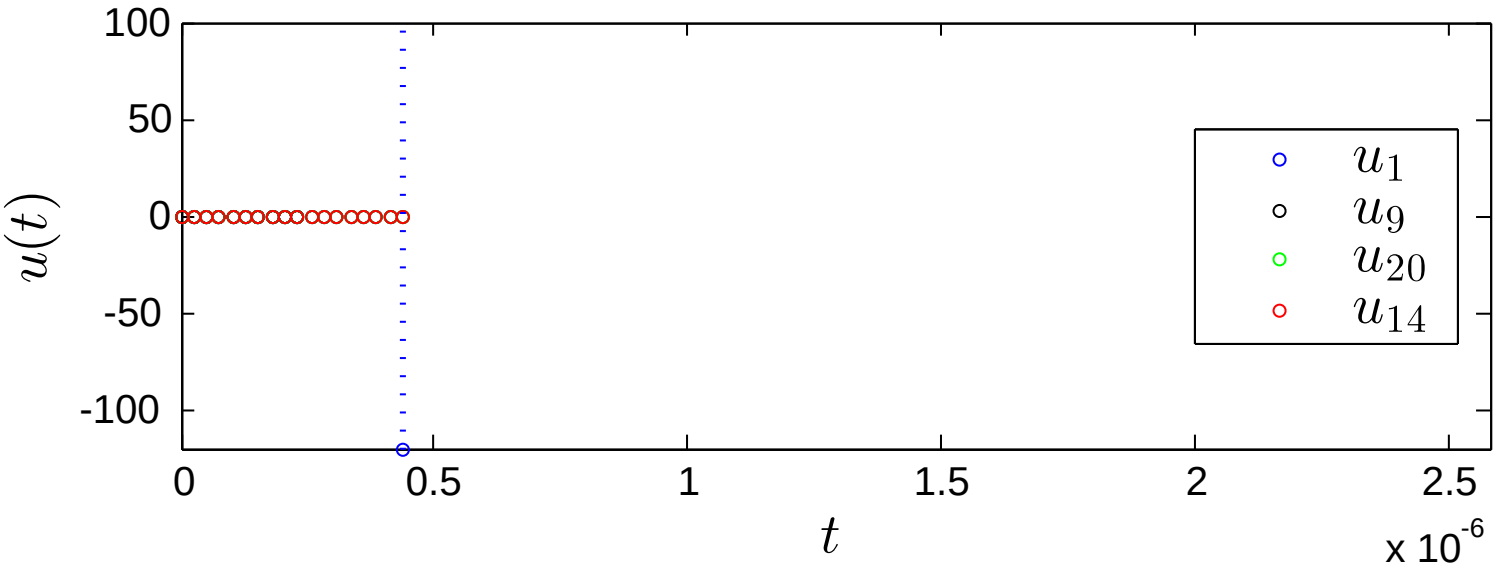
where

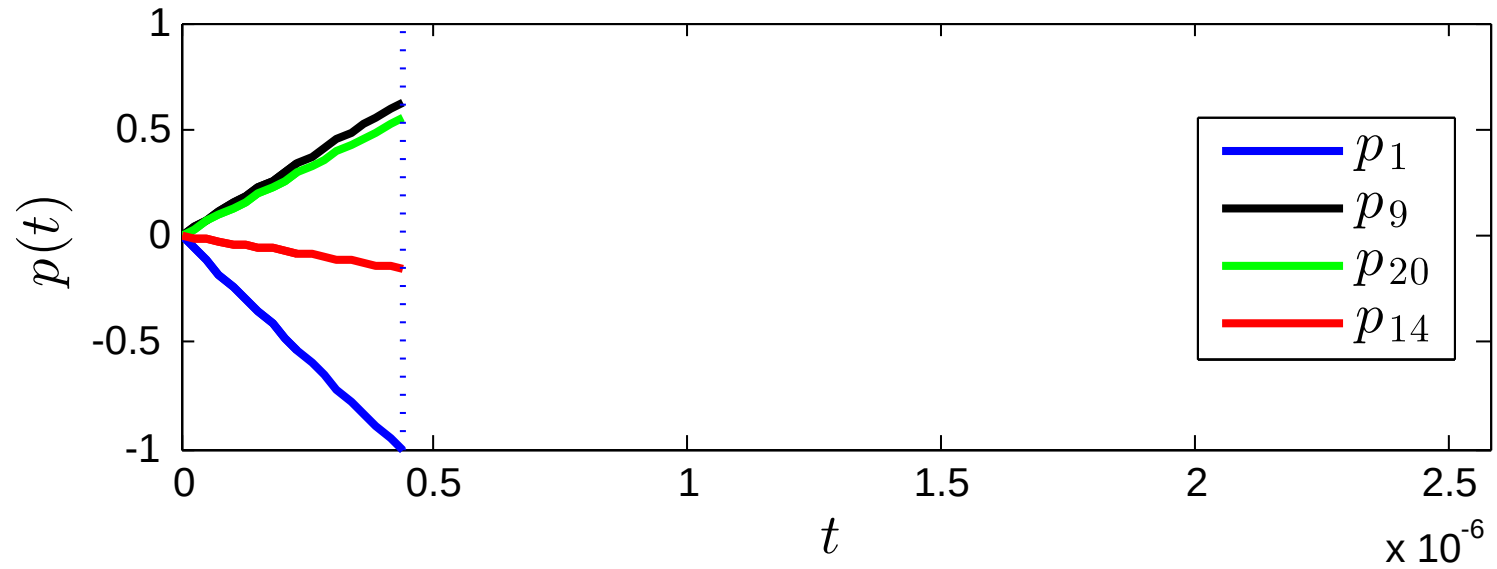
- r is **convex regularization**: weighted ℓ_1, $\ell_{1,2}$, nuclear norm, etc.
- f is **convex fitting**: square loss, logistic loss, etc.

Linearized Bregman ISS model: add a strongly convex function to r.

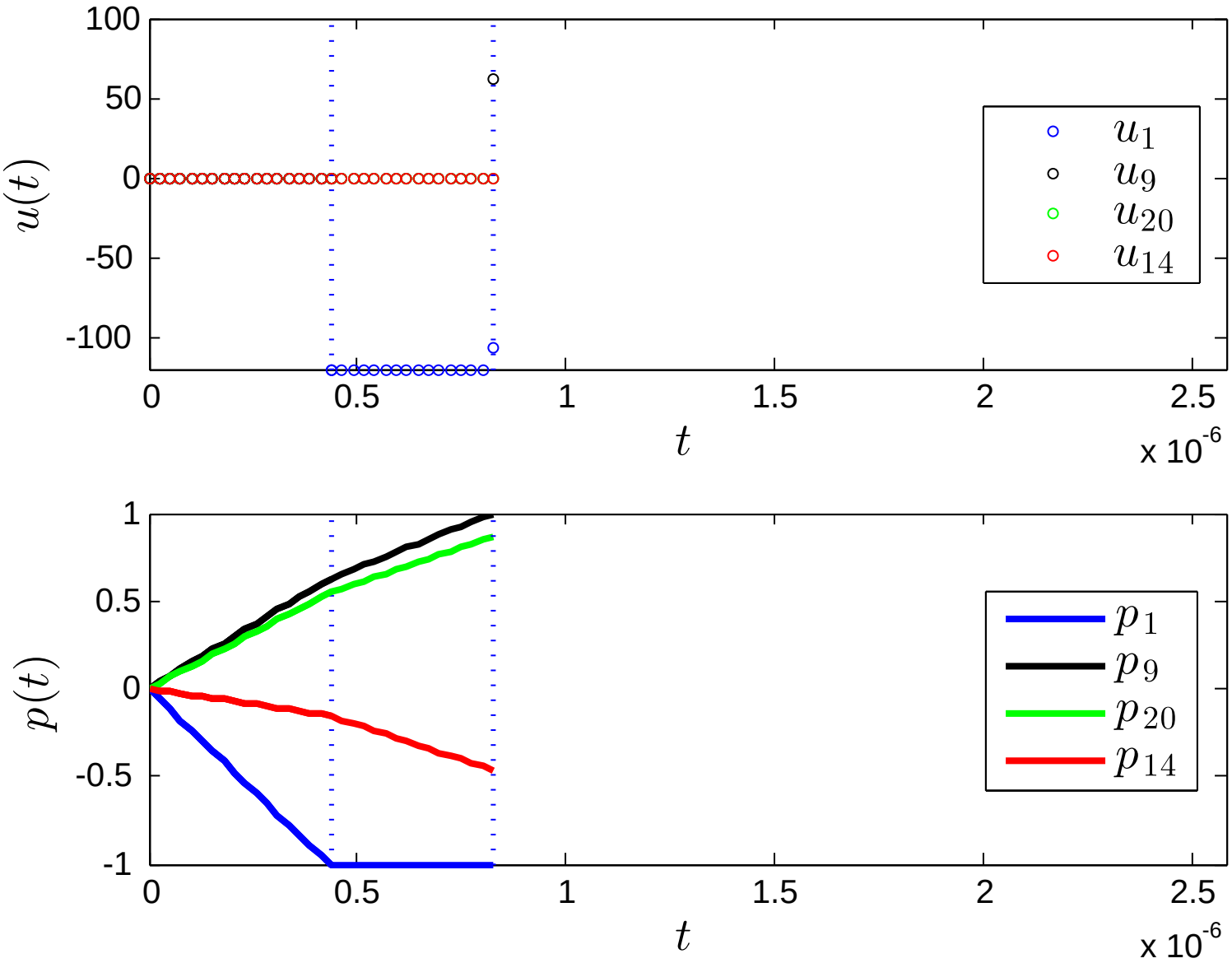
Next: Numerical examples

20-Dimensional Example

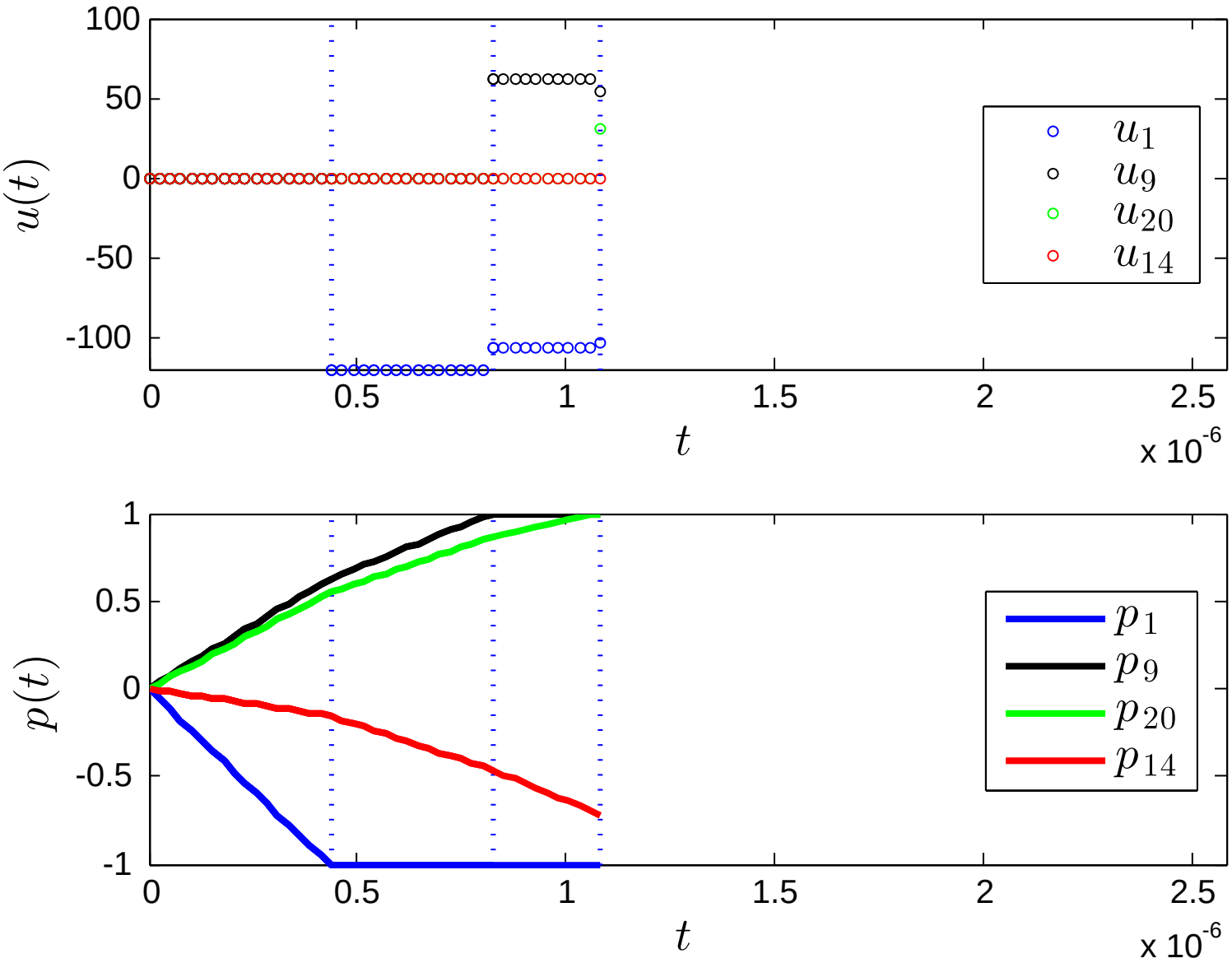




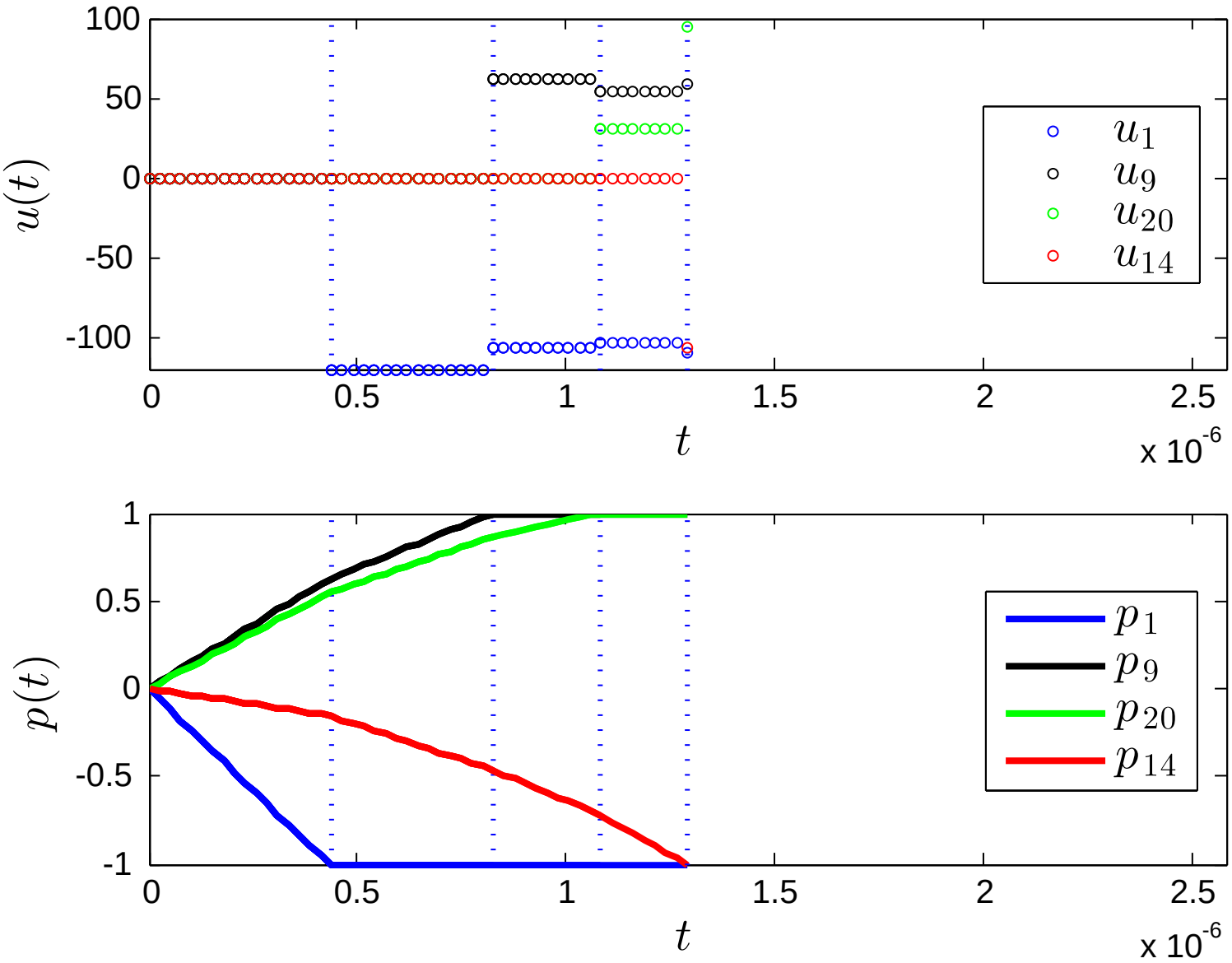
Example


100
50
0
-50
-100
$u(t)$
0
0.5
1
1.5
2
2.5
t
x 10^-6
u_1
u_9
u_{20}
u_{14}
1
0.5
0
-0.5
-1
$p(t)$
0
0.5
1
1.5
2
2.5
t
x 10^-6
p_1
p_9
p_{20}
p_{14}

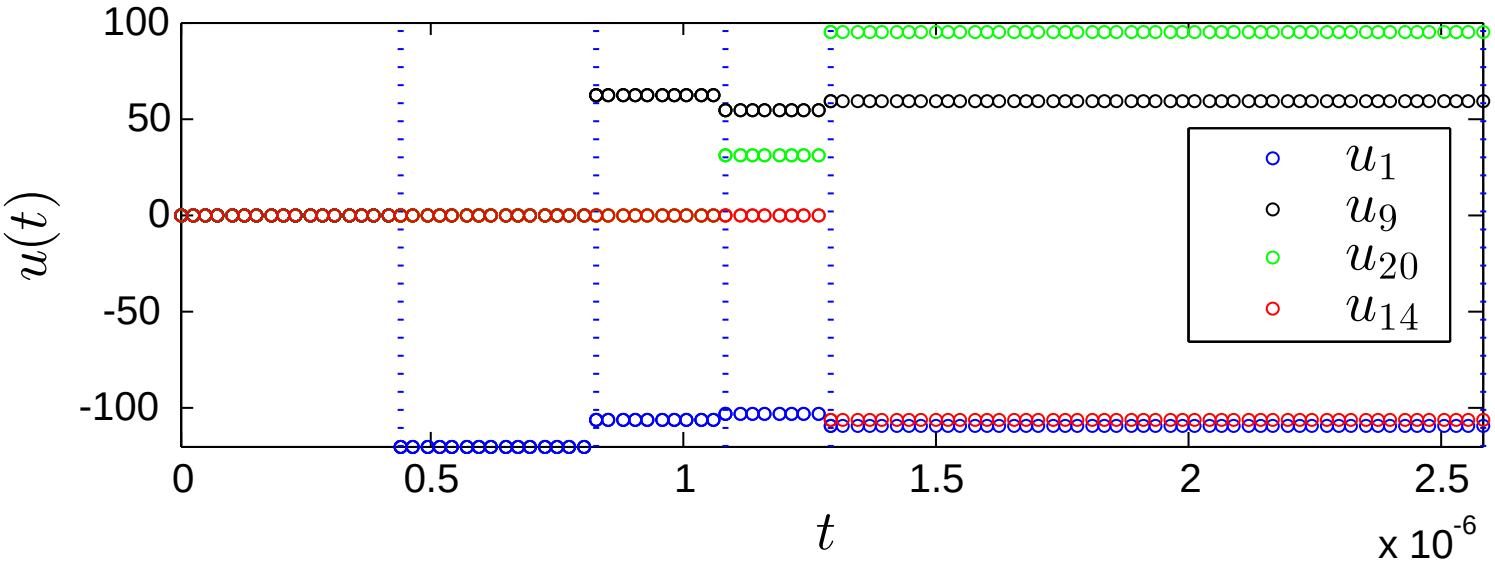
Example

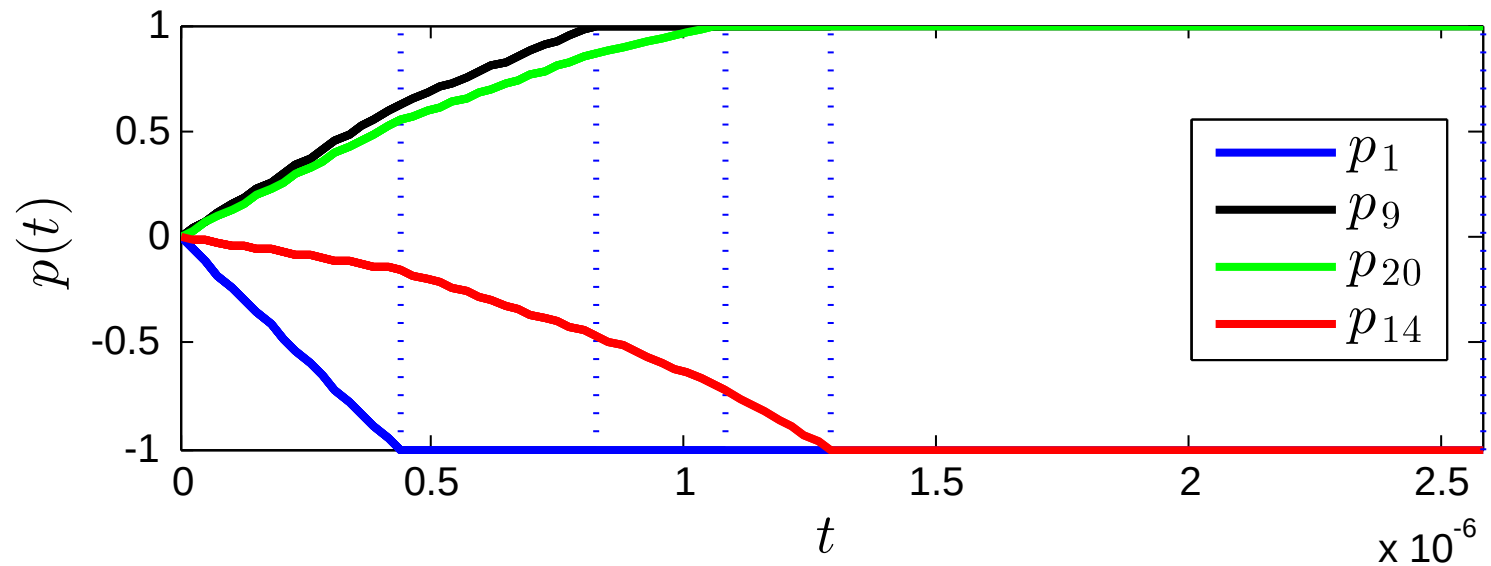

100
50
0
-50
-100
$u(t)$
0
0.5
1
1.5
2
2.5
t
x 10^-6
u_1
u_9
u_{20}
u_{14}
1
0.5
0
-0.5
-1
$p(t)$
0
0.5
1
1.5
2
2.5
t
x 10^-6
p_1
p_9
p_{20}
p_{14}

Example


100
50
0
-50
-100
$u(t)$
0
0.5
1
1.5
2
2.5
t
x 10^-6
u_1
u_9
u_{20}
u_{14}
1
0.5
0
-0.5
-1
$p(t)$
0
0.5
1
1.5
2
2.5
t
x 10^-6
p_1
p_9
p_{20}
p_{14}

Example


100
50
0
-50
-100
$u(t)$
0
0.5
1
1.5
2
2.5
t
x 10^-6
u_1
u_9
u_{20}
u_{14}


1
0.5
0
-0.5
-1
$p(t)$
0
0.5
1
1.5
2
2.5
t
x 10^-6
p_1
p_9
p_{20}
p_{14}

Predict prostate tumor size

- given 8 clinical features, select predictors for prostate tumor size
  - data: 67 training cases + 30 testing cases

Predictor	**LS**	**Subset**	**LASSO**	**ISS**
Intercept	2.452	2.466	2.481	2.476
lcavol	0.716	0.667	0.622	0.554
lweight	0.293	0.366	0.289	0.279
age	-0.143	0	-0.096	0
lbph	0.212	0	0.188	0.198
svi	0.310	0.268	0.262	0.238
lcp	-0.289	-0.291	-0.164	0
gleason	-0.021	0	0	0
pgg45	0.277	0.227	0.187	0.122
#Features	8	**5**	7	**5**
Test Error	0.586	0.587	0.543	**0.541**

LS = least squares, Subset = best subset regression, LASSO solved by glmnet

Bregman ISS achieves least test error with fewest features!

Relation to discrete Bregman iteration

- Forward Euler of $\dot{p} = \frac{1}{m} A^T (b - Ax)$:

$$p^{k+1} = p^k + \frac{\delta}{m} A^T (b - Ax^k),$$

which is the first-order optimality condition to

$$x^{k+1} \leftarrow \arg\min_x D_{\|\cdot\|_1}(x; x^k) + \frac{\delta}{2m} \|Ax - b\|^2,$$

$$\text{where} \quad D_{\|\cdot\|_1}(x; x^k) := \|x\|_1 - \|x^k\|_1 - \langle p^k, x - x^k \rangle.$$

- By change of variable, "**add-back-the-residual**" iteration

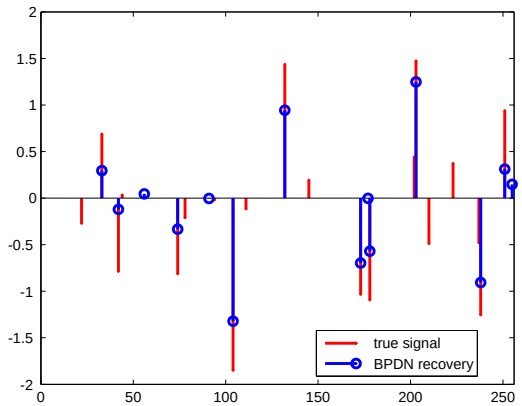
$$x^{k+1} \leftarrow \arg\min_x \|x\|_1 + \frac{\delta}{2m} \|Ax - b^k\|^2,$$

$$b^{k+1} \leftarrow b^k + (b - Ax^k).$$

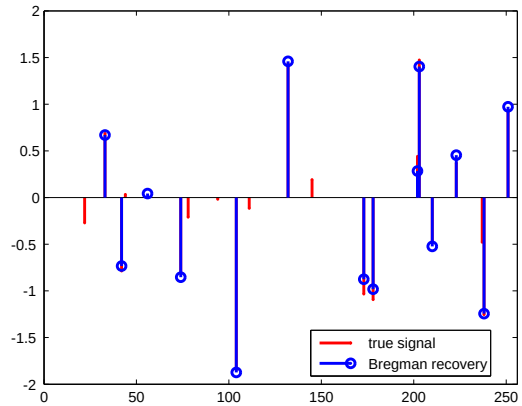
Still true if $\|\cdot\|_1$ is replace by any convex regularizer

- **Message:** keep existing solver, use a small δ, "add-back-the-residual"

Test with noisy measurements and tiny signals



LASSO (hand tuned)



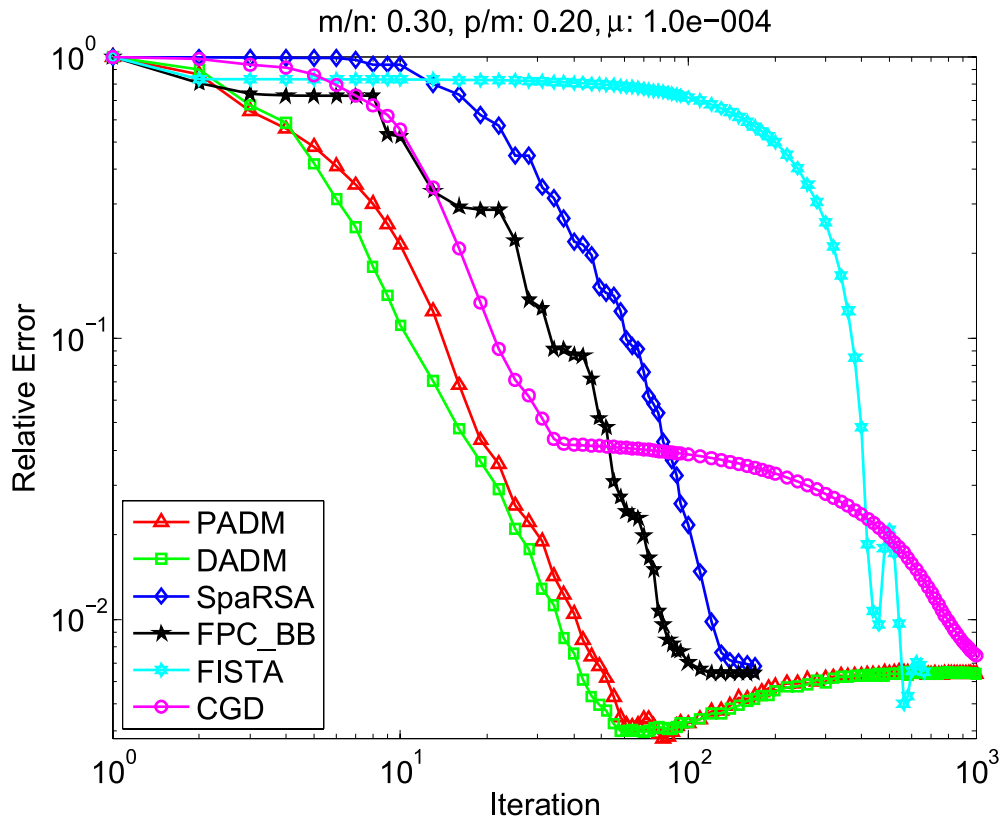
Bregman 5th itr.

Related observation

YALL1 paper (Yang-Zhang'08): tested different algorithms for LASSO

$$\min \|u\|_1 + \frac{t}{2n}\|Au - b\|_2^2.$$

Strange observation: ADM algorithms do better than the model itself!



Theory: path consistency

Question: does there $\exists t$ so that solution $x(t)$ has the following properties?

- **no false positive**: if $u_i = 0$, then $x_i(t) = 0$
- **no false negative**: if $u_i \neq 0$, then $x_i(t) \neq 0$
- **sign consistency**: furthermore, $\text{sign}(x) = \text{sign}(x(t))$.

Theorem

*Under the **assumptions***

- *Gaussian noise:* $\omega \sim N(0, \sigma^2 I)$,
- *normalized column:* $\frac{1}{n} \max_j \|A_j\|^2 \leq 1$,

*and under **irrepresentable** and **strong-signal** conditions, Bregman ISS reaches sign consistency and gives an unbias estimate to* x^*.

Proof is based on the next two lemmas.

No false positive

Define true support $S := \mathrm{supp}(x^*)$, and let $T := S^c$.

Lemma

*Under **assumptions**, if A_S has full column rank and*

$$\max_{j \in T} \|A_j^T A_S (A_S^T A_S)^{-1}\|_1 \leq 1 - \eta$$

for some $\eta \in (0, 1)$, then with high probability

$$\mathrm{supp}(x(s)) \subseteq S, \quad \forall s \leq \bar{t} := O\left(\frac{\eta}{\sigma}\sqrt{\frac{m}{\log n}}\right).$$

Proof uses: (i) concentration inequality and (ii) if $\mathrm{supp}(x(s)) \subseteq S,\ s \leq t$, then

$$p(s)_T = A_T^T A_S (A_S^T A_S)^{-1} p(s)_S + t A_T^* P_{A_S^\perp} w, \quad s \leq t.$$

No false negative / sign consistency

Lemma

*Under **assumptions**, if* $A_S^* A_S \succeq \gamma I$ *and*

$$u_{\min} \geq \max\left\{ O\left(\frac{\sigma}{\sqrt{\gamma}}\sqrt{\frac{\log|S|}{m}}\right), O\left(\frac{\sigma \log|S|}{\eta\gamma}\sqrt{\frac{\log n}{m}}\right)\right\},$$

then there exist t^* *(which can be given explicitly) so that with high probability*

$$\operatorname{sign}(x(t)) = \operatorname{sign}(x^*)$$

and $x(t) = x_S^* - (A_S^* A_S)^{-1} A_S^* \omega$ *obeys*

$$\|x(t) - x^*\|_\infty \leq u_{\min}/2.$$

- first term in $\max$ ensures $\|(A_S^* A_S)^{-1} A_S^* \omega\|_\infty \leq u_{\min}/2$
- second term ensures: $\inf\{t : \operatorname{sign}(x_S(t)) = \operatorname{sign}(x_S)\} \leq \bar{t}$.

Related work

Discrete:

- **Bregman iteration** for imaging (TV) and compressed sensing ℓ_1: Osher-Burger-Goldfarb-Xu-Y'06, Y-Osher-Goldfarb-Darbon'08
- **Linearized Bregman** on ℓ_1: Y-Osher-Goldfarb-Darbon'08, Y'10, Lai-Y'13
- **Matrix completion SVT** on $\|X\|_*$: Cai-Candès-Shen'10
- **Extension and analysis**: Zhang'13, Zhang'14

Continuous:

- **Inverse scale space** (ISS) on TV: Burger-Gilboa-Osher-Xu'06
- **Adaptive ISS** on ℓ_1: Burger-Möller-Benning-Osher'11
- **Greedy ISS** on ℓ_1: Möller-Zhang'13

Summary

Instead of $\text{minimize } r(x) + t \cdot f(x)$, just try

$$\dot{p}(t) = -f'(x), \quad p \in \partial r(x).$$

It will

- keep solution structure
- remove bias
- give a solution path efficiently

Even simpler for you: keep your existing solver, apply "add back the residual"