Slide Template

Neal Parikh and Stephen Boyd

Stanford University

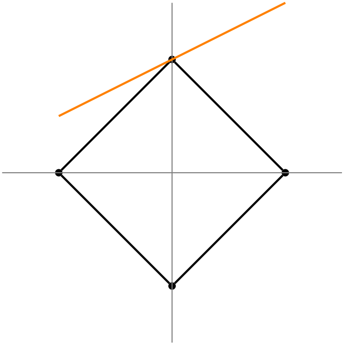
July 1, 2025

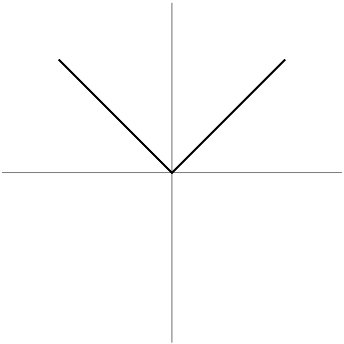
Outline

Bulleted list

- XXX
  - XXX
  - XXX
  - XXX
- XXX
  - XXX
  - XXX
  - XXX
- XXX

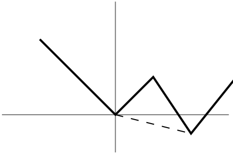
Pictures with tikz





Pictures with tikz

- convex envelope of (nonconvex) f is the largest convex underestimator g
- *i.e.*, the best convex lower bound to a function



- **example**: ℓ_1 is the envelope of $\mathbf{card}$ (on unit ℓ_∞ ball)
- **example**: $\|\cdot\|_*$ is the envelope of $\mathbf{rank}$ (on unit spectral norm ball)
- various characterizations: *e.g.*, f^{**} or convex hull of epigraph

Outline

Group lasso

(*e.g.*, Yuan & Lin; Meier, van de Geer, Bühlmann; Jacob, Obozinski, Vert)

- problem:

$$\text{minimize} \quad f(x) + \lambda \sum_{i=1}^{N} \|x_i\|_2$$

  i.e., like lasso, but require groups of variables to be zero or not

- also called $\ell_{1,2}$ mixed norm regularization

Structured group lasso

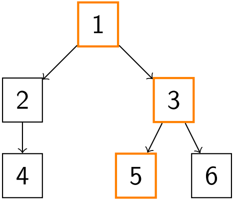
(Jacob, Obozinski, Vert; Bach et al.; Zhao, Rocha, Yu; . . .)

- problem:
$$\text{minimize} \quad f(x) + \textstyle\sum_{i=1}^{N} \lambda_i \|x_{g_i}\|_2$$
where $g_i \subseteq [n]$ and $\mathcal{G} = \{g_1, \ldots, g_N\}$
- like group lasso, but the groups can overlap arbitrarily
- particular choices of groups can impose 'structured' sparsity
- *e.g.*, topic models, selecting interaction terms for (graphical) models, tree structure of gene networks, fMRI data
- generalizes to the **composite absolute penalties family**:
$$r(x) = \|(\|x_{g_1}\|_{p_1}, \ldots, \|x_{g_N}\|_{p_N})\|_{p_0}$$

Structured group lasso

(Jacob, Obozinski, Vert; Bach et al.; Zhao, Rocha, Yu; . . .)

hierarchical selection:

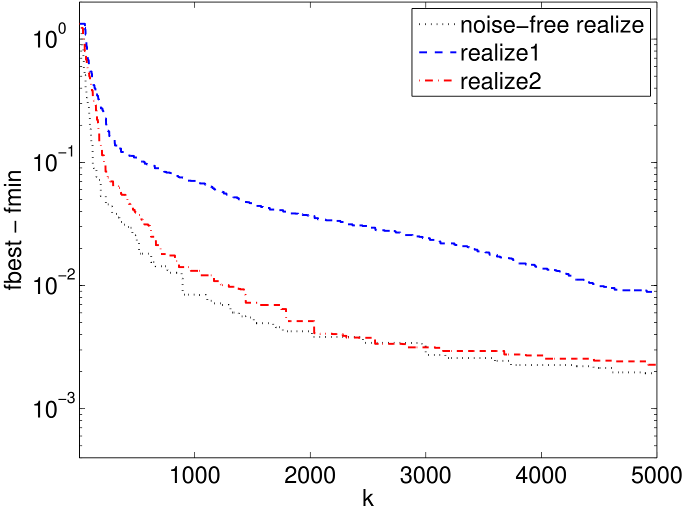


- $\mathcal{G} = \{\{4\}, \{5\}, \{6\}, \{2,4\}, \{3,5,6\}, \{1,2,3,4,5,6\}\}$
- nonzero variables form a rooted and connected subtree
  - if node is selected, so are its ancestors
  - if node is not selected, neither are its descendants

Sample ADMM implementation: lasso

```
prox_f = @(v,rho) (rho/(1 + rho))*(v - b) + b;
prox_g = @(v,rho) (max(0, v - 1/rho) - max(0, -v - 1/rho));

AA = A*A';
L  = chol(eye(m) + AA);

for iter = 1:MAX_ITER
    xx = prox_g(xz - xt, rho);
    yx = prox_f(yz - yt, rho);

    yz = L \ (L' \ (A*(xx + xt) + AA*(yx + yt)));
    xz = xx + xt + A'*(yx + yt - yz);

    xt = xt + xx - xz;
    yt = yt + yx - yz;
end
```

Figure



noise−free realize
realize1
realize2
10^0
10^{-1}
10^{-2}
10^{-3}
fbest − fmin
1000
2000
3000
4000
5000
k

Algorithm

if L is not known (usually the case), can use the following line search:

---

given x^k, λ^{k-1}, and parameter $\beta \in (0, 1)$.

Let $\lambda := \lambda^{k-1}$.

repeat

1. Let $z := \mathbf{prox}_{\lambda g}(x^k - \lambda \nabla f(x^k))$.
2. **break if** $f(z) \leq \hat{f}_\lambda(z, x^k)$.
3. Update $\lambda := \beta\lambda$.

return $\lambda^k := \lambda$, $x^{k+1} := z$.

---

typical value of β is $1/2$, and

$$\hat{f}_\lambda(x, y) = f(y) + \nabla f(y)^T(x - y) + (1/2\lambda)\|x - y\|_2^2$$