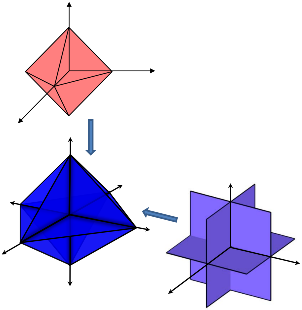
Combinatorial Selection and Least Absolute Shrinkage via the CLASH Algorithm

Anastasios Kyrillidis and Volkan Cevher

{anastasios.kyrillidis,volkan.cevher}@epfl.ch

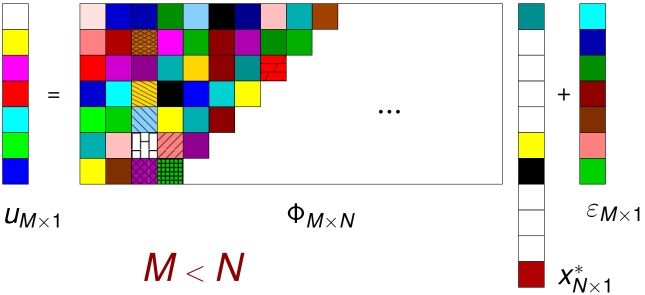


Laboratory for Information and Inference Systems
LIONS/EPFL

http://lions.epfl.ch/

- **Underdetermined linear regression**:

$$u = \Phi x^* + \varepsilon,$$



- **Goal**: find signal x^* that generated the measurements u.
- **Difficulties:** $M < N \rightarrow$ Non-trivial nullspace of $\Phi \rightarrow$ ill-posed problem.
- **Main assumption:** x^* is K-sparse (i.e., $x^* \in \Sigma_K$) for $K < M$.

Algorithmic approaches

- "Natural" criteria:

$$\begin{array}{ll} \underset{x}{\text{minimize}} & \|x\|_0 \\ \text{subject to} & u = \Phi x \end{array}$$

where $\|x\|_0 = \#\{x_i \neq 0,\ i = 1, \ldots, N\}$.

Algorithmic approaches

- "Natural" criteria:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \|x\|_0 \\ \text{subject to} \quad & u = \Phi x \end{aligned}$$

where $\|x\|_0 = \#\{x_i \neq 0,\ i = 1, \ldots, N\}$.

NP-hard [Natarajan'95]

Algorithmic approaches

- "Natural" criteria:

$$\underset{x}{\text{minimize}} \quad \|x\|_0 \qquad\qquad \underset{x}{\text{minimize}} \quad \|x\|_1$$
$$\text{subject to} \quad u = \Phi x \qquad\qquad \text{subject to} \quad u = \Phi x$$

where $\|x\|_0 = \# \{x_i \neq 0, \ i = 1, \dots, N\}$.

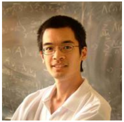
~~NP-hard [Natarajan'95]~~ $\Longleftarrow$ Not with the RIP assumption!

Restricted Isometry Property (RIP) [Candes & Tao.'06]

Φ satisfies the RIP with constant δ_K iff

$$(1 - \delta_K)\|x\|_2^2 \leq \|\Phi x\|_2^2 \leq (1 + \delta_K)\|x\|_2^2,$$

is satisfied for any $x \in \Sigma_K$



Algorithmic approaches

- "Natural" criteria:

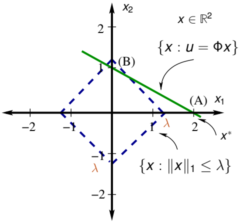
$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \|x\|_0 \\ \text{subject to} \quad & u = \Phi x \end{aligned}$$

where $\|x\|_0 = \#\{x_i \neq 0, \; i = 1, \dots, N\}$.

- What people usually use: Convex criteria...

Lasso:

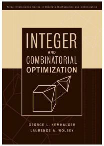
$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \|u - \Phi x\|_2 \\ \text{subject to} \quad & \|x\|_1 \leq \lambda \end{aligned}$$

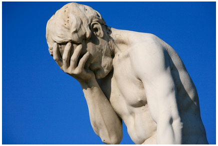


- ...instead of: Combinatorial/non-convex criteria

Greedy, Projected gradient descent:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \|u - \Phi x\|_2 \\ \text{subject to} \quad & \|x\|_0 \leq K \end{aligned}$$

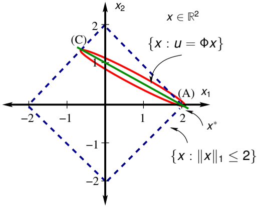




Q&A

- Does Lasso "know" that we are looking for a K-sparse solution?

- Does Lasso "know" that we are looking for a K-sparse solution? **No.**

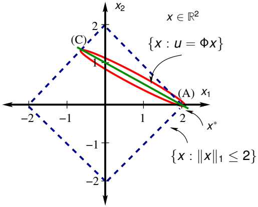


Feasible Solution Candidate Set:

[(**A**), . . . , (**C**)].

Q&A

- Does Lasso “know” that we are looking for a K-sparse solution? **No.**

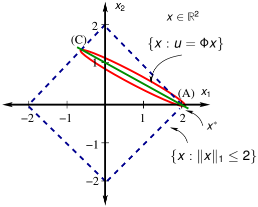


Feasible Solution Candidate Set:

$[(\mathbf{A}), \ldots, (\mathbf{C})]$.

- Can Lasso exploit discrete models?

- Does Lasso "know" that we are looking for a K-sparse solution? **No.**

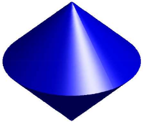


Feasible Solution Candidate Set:

[(**A**), . . . , (**C**)].

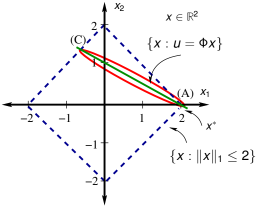
- Can Lasso exploit discrete models? Well, ... **Maybe** $\Longrightarrow$ Convexify discrete structure and regularize $\Omega(x)$ [Jenatton et al.'10]:

$$\underset{x}{\text{minimize}} \quad \|u - \Phi x\|_2 + \tau\Omega(x).$$



Q&A

- Does Lasso "know" that we are looking for a K-sparse solution? **No.**

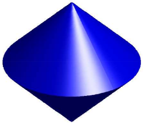


Feasible Solution Candidate Set:

[(**A**), . . . , (**C**)].

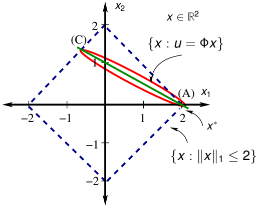
- Can Lasso exploit discrete models? Well, ... **Maybe** $\Longrightarrow$ Convexify discrete structure and regularize $\Omega(x)$ [Jenatton et al.'10]:

$$\underset{x}{\text{minimize}} \quad \|u - \Phi x\|_2 + \tau\Omega(x).$$



- **But**, is it intuitive to construct $\Omega(x)$?

- Does Lasso "know" that we are looking for a K-sparse solution? **No.**

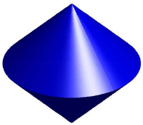


Feasible Solution Candidate Set:

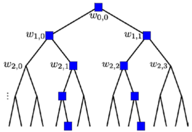
[(**A**), . . . , (**C**)].

- Can Lasso exploit discrete models? Well, ... **Maybe** $\Longrightarrow$ Convexify discrete structure and regularize $\Omega(x)$ [Jenatton et al.'10]:

$$\underset{x}{\text{minimize}} \quad \|u - \Phi x\|_2 + \tau\Omega(x).$$

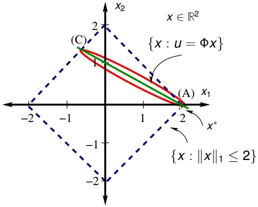


- **But**, is it intuitive to construct $\Omega(x)$? Generally, **No**.



Q&A

- Does Lasso "know" that we are looking for a K-sparse solution? **No.**

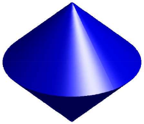


Feasible Solution Candidate Set:

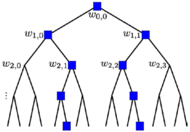
[(A), . . . , (C)].

- Can Lasso exploit discrete models? Well, ... **Maybe** $\Longrightarrow$ Convexify discrete structure and regularize $\Omega(x)$ [Jenatton et al.'10]:

$$\underset{x}{\text{minimize}} \quad \|u - \Phi x\|_2 + \tau\Omega(x).$$



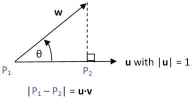
- **But**, is it intuitive to construct $\Omega(x)$? Generally, **No**.



However, due to convexification, Lasso is easier to analyze!

What is this presentation about?

- **New** hybrid optimization framework.



- **New** model-based projection framework.

- **New** algorithm: **Combinatorial selection and Least Absolute SHrinkage (CLASH)**.

The CLASH criteria

- Our proposal:

CLASH:

$$\begin{array}{ll} \underset{x}{\text{minimize}} & \|u - \Phi x\|_2 \\ \text{subject to} & \|x\|_0 \leq K \\ & \|x\|_1 \leq \lambda \end{array}$$

The CLASH criteria

- Our proposal:

$$\textbf{CLASH:} \quad \begin{array}{ll} \underset{x}{\text{minimize}} & \|u - \Phi x\|_2 \\ \text{subject to} & \|x\|_0 \leq K \\ & \|x\|_1 \leq \lambda \end{array}$$

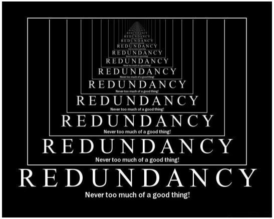
- But, wait a moment...

The CLASH criteria

- Our proposal:

$$\text{CLASH:} \quad \begin{array}{ll} \underset{x}{\text{minimize}} & \|u - \Phi x\|_2 \\ \text{subject to} & \|x\|_0 \leq K \\ & \|x\|_1 \leq \lambda \end{array}$$

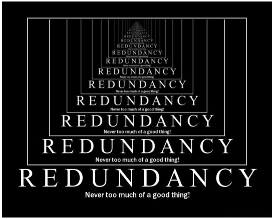
- Our perspective: “redundancy” helps.



The CLASH criteria

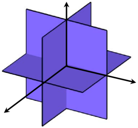
- Our proposal:

$$\textbf{CLASH:} \quad \begin{array}{ll} \underset{x}{\text{minimize}} & \|u - \Phi x\|_2 \\ \text{subject to} & \|x\|_0 \leq K \\ & \|x\|_1 \leq \lambda \end{array}$$

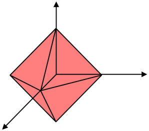


- Our perspective: "redundancy" helps.

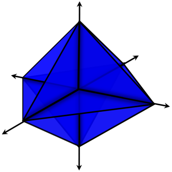
- CLASH geometry:



$+$



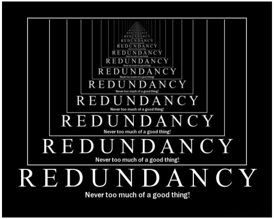
$\approx$



The CLASH criteria

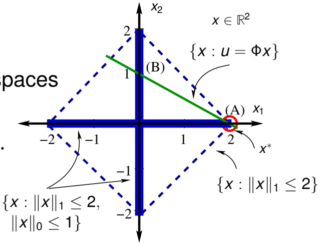
- Our proposal:

$$\textbf{CLASH:} \quad \begin{array}{ll} \underset{x}{\text{minimize}} & \|u - \Phi x\|_2 \\ \text{subject to} & \|x\|_0 \leq K \\ & \|x\|_1 \leq \lambda \end{array}$$



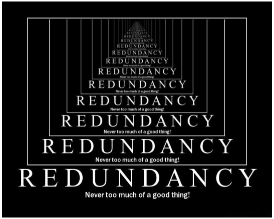
- Our perspective: "redundancy" helps.

- It turns out CLASH...
  1. Fixes the scale of the infinite-extended union of subspaces Σ_K using ℓ_1-norm.
  2. Reduces the cardinality of the candidate solution set.
  3. Exploits geometry — ℓ_1/ℓ_2-norm interplay.
  4. Exploits combinatorics — exact support selections.



- Our proposal:

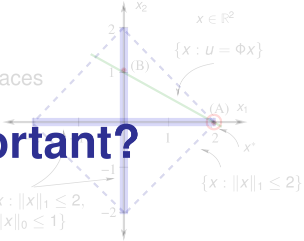
$$\textbf{CLASH:}\quad \begin{array}{ll} \underset{x}{\text{minimize}} & \|u - \Phi x\|_2 \\ \text{subject to} & \|x\|_0 \leq K \\ & \|x\|_1 \leq \lambda \end{array}$$



- Our perspective: "redundancy" helps.

- It turns out CLASH...
  - Fixes the scale of the infinite-extended union of subspaces Σ_K using ℓ_1-norm.
  - Reduces the … solution set.
  - Exploits geometry — ℓ_1/ℓ_2-norm interplay.
  - Exploits combinatorics — exact support selections.

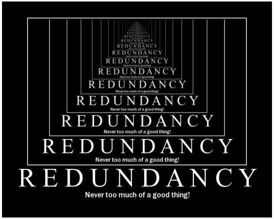
What's most important?



The CLASH criteria

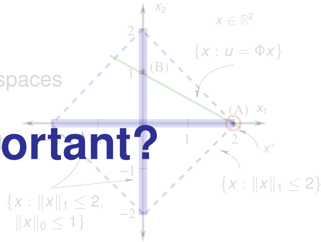
- Our proposal:

$$\textbf{CLASH:}\quad \begin{array}{ll} \underset{x}{\text{minimize}} & \|u - \Phi x\|_2 \\ \text{subject to} & \|x\|_0 \leq K \\ & \|x\|_1 \leq \lambda \end{array}$$



- Our perspective: “redundancy” helps.

- It turns out CLASH...
  - Fixes the scale of the infinite-extended union of subspaces Σ_K using ℓ_1-norm.
  - Reduces the cardinality of the candidate solution set.
  - Exploits geometry — ℓ_1/ℓ_2-norm interplay.
  - Exploits combinatorics — exact support selections.

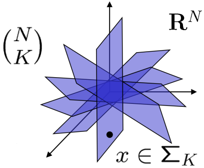


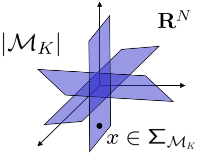
What’s most important?

We can further leverage signal structure!

Structure signal models

- Simple sparsity model Σ_K: only K out of N coordinates nonzero.
- Combinatorial sparsity model $\Sigma_{\mathcal{M}_K}$: reduced set of subspaces.





Combinatorial sparsity model (CSM) [Baraniuk et al'10]

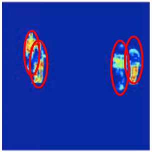
We define a CSM as

$$\mathcal{M}_K = \{\mathcal{S}_m : \forall m, \mathcal{S}_m \subseteq \mathcal{N}, |\mathcal{S}_m| \leq K\}, \; |\mathcal{M}_K| \leq \binom{N}{K}$$

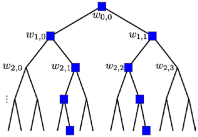
with sparsity K as a collection of index subsets $\mathcal{S}_m$.

Example: clustered signals

General cluster model:



Rooted-connected tree model:

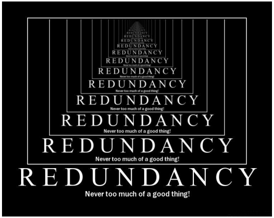


(K, C)-clustered model:

The CLASH criteria

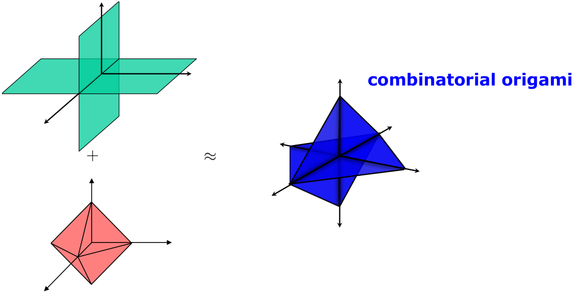
- Our proposal:

$$\textbf{CLASH:}\quad \begin{array}{ll} \underset{x}{\text{minimize}} & \|u - \Phi x\|_2 \\ \text{subject to} & \|x\|_0 \leq K \\ & \|x\|_1 \leq \lambda \end{array}$$



- Our perspective: "redundancy" helps.

- CLASH geometry:



Projections onto discrete sets

- Non-convex projection onto $\mathcal{M}_K$:

$$\mathcal{P}_{\mathcal{M}_K}(x) = \underset{w \in \mathbb{R}^N}{\operatorname{argmin}} \{ \|w - x\|_2^2 : \operatorname{supp}(w) \in \Sigma_{\mathcal{M}_K} \}$$

- **Main difficulty:** find the support pattern.

Projections onto discrete sets

- Non-convex projection onto $\mathcal{M}_K$:

$$\mathcal{P}_{\mathcal{M}_K}(x) = \underset{w \in \mathbb{R}^N}{\operatorname{argmin}} \{ \|w - x\|_2^2 : \operatorname{supp}(w) \in \Sigma_{\mathcal{M}_K} \}$$

- **Main difficulty:** find the support pattern.
- **Desired:** project in polynomial or pseudo-polynomial time.

Projections onto discrete sets

- Non-convex projection onto $\mathcal{M}_K$:

$$\mathcal{P}_{\mathcal{M}_K}(x) = \underset{w \in \mathbb{R}^N}{\operatorname{argmin}} \{\|w - x\|_2^2 : \operatorname{supp}(w) \in \Sigma_{\mathcal{M}_K}\}$$

- **Main difficulty:** find the support pattern.
- **Desired:** project in polynomial or pseudo-polynomial time.
- **Key observation #1:** Modularity of Euclidean projections onto CSMs

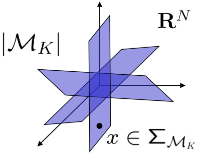
$$\begin{aligned}
\operatorname{supp}\left(\mathcal{P}_{\mathcal{M}_K}(x)\right) &= \operatorname{supp}\left(\underset{w \in \mathbb{R}^N : \operatorname{supp}(w) \in \Sigma_{\mathcal{M}_K}}{\operatorname{argmin}} \{\|w - x\|_2^2\}\right) \\
&= \underset{\mathcal{S} : \mathcal{S} \in \Sigma_{\mathcal{M}_K}}{\operatorname{argmin}} \{\|(x)_{\mathcal{S}} - x\|_2^2\} \\
&= \underset{\mathcal{S} : \mathcal{S} \in \Sigma_{\mathcal{M}_K}}{\operatorname{argmax}} \{\|x\|_2^2 - \|(x)_{\mathcal{S}} - x\|_2^2\} \\
&= \underset{\mathcal{S} : \mathcal{S} \in \Sigma_{\mathcal{M}_K}}{\operatorname{argmax}} \sum_{i \in \mathcal{S}} |x_i|^2 \\
&\triangleq \underset{\mathcal{S} : \mathcal{S} \in \Sigma_{\mathcal{M}_K}}{\operatorname{argmax}} F(\mathcal{S}, x)
\end{aligned}$$

Projections onto discrete sets

Example: Matroid structured sparse models:

$$\mathcal{M} := \left(\mathcal{N}, \mathcal{I} \subseteq 2^{\mathcal{N}}\right), \ \mathcal{N} = \{1, \dots, N\}$$

where: *i*. $\mathcal{N}$: ground set,
ii. $\mathcal{I}$: base set.



- Given a matroid $\mathcal{M}$ and $F(\mathcal{S}, x) = \sum_{i \in \mathcal{S}} |x_i|^2$, greedy basis algorithm efficiently solves matroid constrained problems.
- Highlight: Uniform Matroid $\mathcal{M}_K^U \longrightarrow \mathcal{I} = \{\mathcal{S} : \ \mathcal{S} \subseteq \mathcal{N}, \ |\mathcal{S}| \leq K\}$.

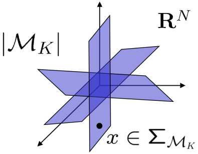
$$\text{supp}\left(\mathcal{P}_{\mathcal{M}_K^U}(x)\right) = \operatorname*{argmax}_{\mathcal{S}:\mathcal{S} \in \mathcal{M}_K^U} F(\mathcal{S}, x) = \text{supp}\Bigg(\underbrace{\operatorname*{argmin}_{y: y \in \Sigma_K} \|x - y\|_2^2}_{\text{Hard thresholding}}\Bigg)$$

Projections onto discrete sets

Example: Matroid structured sparse models:

$$\mathcal{M} := \left(\mathcal{N}, \mathcal{I} \subseteq 2^{\mathcal{N}}\right), \; \mathcal{N} = \{1, \dots, N\}$$

where: *i*. $\mathcal{N}$: ground set,
ii. $\mathcal{I}$: base set.



- Given a matroid $\mathcal{M}$ and $F(\mathcal{S}, x) = \sum_{i \in \mathcal{S}} |x_i|^2$, greedy basis algorithm efficiently solves matroid constrained problems.

Intersections of matroids

The intersection of the uniform matroid with any other matroid defines a new matroid.

- Partition matroid $\longrightarrow$ Distributed sparsity.
- Graphic matroid $\longrightarrow$ Spanning tree sparsity.
- Matching matroid $\longrightarrow$ Graph matching sparsity.

Projections onto discrete sets

- Non-convex projection onto $\mathcal{M}_K$:

$$\mathcal{P}_{\mathcal{M}_K}(x) = \underset{w \in \mathbb{R}^N}{\operatorname{argmin}} \{ \|w - x\|_2^2 : \operatorname{supp}(w) \in \Sigma_{\mathcal{M}_K} \}$$

- **Main difficulty:** find the support pattern.
- **Desired:** project in polynomial or pseudo-polynomial time.
- **Key observation #2:**

Integer LP nature of $\mathcal{P}_{\mathcal{M}_K}$

The support index set of $\mathcal{P}_{\mathcal{M}_K}$ can be equivalently found using following integer linear program (ILP):

$$\operatorname{supp}\Big(\underset{\substack{z : z_i \in \{0,1\}, \\ \operatorname{supp}(z) \in \Sigma_{\mathcal{M}_K}}}{\operatorname{argmin}} \; \{ w^T z : w_i = -|x_i|^2 \} \Big),$$

where z_i, $(i = 1, \dots, n)$, are support indicator variables.

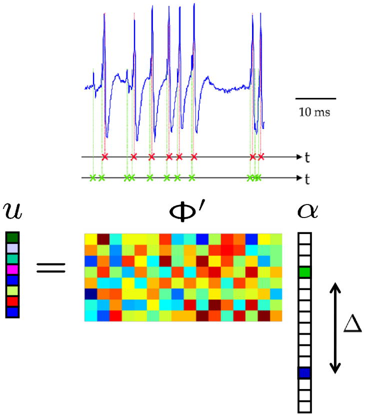
Projections onto discrete sets

Example: Linear support constraints:

Example: neuronal spike model

$z \in \{0,1\}^N$: binary support variables

$$
\begin{aligned}
z_1 + z_2 + \ldots + z_N &\leq K \\
z_1 + z_2 + \ldots + z_\Delta &\leq 1 \\
z_2 + z_3 + \ldots + z_{\Delta+1} &\leq 1 \\
&\vdots \\
z_{N-\Delta+1} + z_{N-\Delta+2} + \ldots + z_N &\leq 1
\end{aligned}
$$



[Hedge et al.'09]

Projections onto discrete sets

Example: Linear support constraints:

Definition

$$\Sigma_{\mathcal{M}_K} = \cup_{\forall z \in \mathfrak{Z}} \text{supp}(z), \text{ where } \mathfrak{Z} := \{z_i \in \{0,1\} : Az \leq b\},$$

where $[A; b]$ is an integral matrix, and the first row of A is all 1's and $b_1 = K$.

- We are interested in solving:

$$\begin{aligned} \text{supp}\left(\mathcal{P}_{\Sigma_{\mathcal{M}_K}}(x)\right) &= \text{supp}\left(\underset{y : y \in \Sigma_{\mathcal{M}_K}}{\text{argmin}} \|x - y\|_2^2\right) \\ &= \text{supp}\left(\underset{z}{\text{argmin}} \left\{\sum_i -|x_i|^2 z_i : \; z_i \in \{0,1\}, \; Az \leq b\right\}\right) \end{aligned}$$

Lemma [Nemhauser & Wosley'99]

Linear programming can exactly solve the linear-support constrained integer linear programming when A is totally unimodular, i.e., the determinant of each square submatrix of A is $\{0, \pm 1\}$.

Projections onto discrete sets

- Non-convex projection onto $\mathcal{M}_K$:

$$\mathcal{P}_{\mathcal{M}_K}(x) = \underset{w \in \mathbb{R}^N}{\operatorname{argmin}} \{ \|w - x\|_2^2 : \operatorname{supp}(w) \in \Sigma_{\mathcal{M}_K} \}$$

- **Main difficulty:** find the support pattern.
- **Desired:** project in polynomial or pseudo-polynomial time.
- **Key observation #3:**

Polynomial time modular ϵ-approximation property

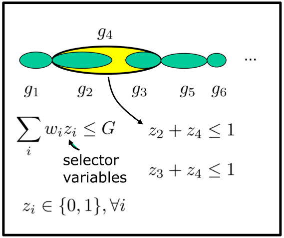
A CSM has the PMAP$_\epsilon$ with constant ϵ, if the modular subset selection problem or the ILP admit an ϵ-approximation scheme with polynomial or pseudo-polynomial time complexity as a function of N, $\forall x \in \mathbb{R}^N$, i.e.,

$$F(\widehat{\mathcal{S}}_\epsilon; x) \geq (1 - \epsilon) \max_{\mathcal{S} \in \Sigma_{\mathcal{M}_K}} F(\mathcal{S}; x).$$

Projections onto discrete sets

Example: Multi-knapsack instances:

- Knapsack
  1. Multi-knapsack constraints
  2. Weighted multi-knapsack constraints
- Pairwise overlapping groups
  1. Quadratic binary program. with cardinality constraints.



Pairwise overlapping groups

$$\max_{\mathcal{S}:\ \mathcal{S}\in\Sigma_{\mathcal{M}_K}} F(\mathcal{S}, x) = -\min\left\{\sum_{i>j} \|(x)_{g_i\cap g_j}\|_2^2 z_i z_j - \sum_i \|(x)_{g_i}\|_2^2 z_i : \sum_i z_i \leq G\right\}$$

Combinatorial selection

+

convex geometry

$\Downarrow$

$\Rightarrow$

$\Uparrow$

Structure sparsity + PMAP$_\epsilon$

CLASH pseudocode and approximation guarantees

- Algorithm code @ http://lions.epfl.ch/CLASH

1. Active set expansion via selection
2. Greedy descend
3. Combinatorial selection
4. De-bias with convex (ℓ_1-norm) constraint

Theorem

Let $x^* \in \mathbb{R}^N$ and $u = \Phi x^* + \varepsilon$. Define the signal-to-noise ratio of x^* as $\text{SNR} = \frac{\|x^*\|_2}{\sqrt{f(x^*)}}$. Then, the i-th iterate x_i of CLASH satisfies the following recursion

$$\frac{\|x_{i+1} - x^*\|_2}{\|x^*\|_2} \leq \rho \frac{\|x_i - x^*\|_2}{\|x^*\|_2} + \text{SNR terms.}$$

For $\epsilon = 0$, when $\delta_{3K} < 0.3658$, the iterations are contractive (i.e., $\rho < 1$).

(BPDN):

$$\begin{array}{ll} \underset{x}{\text{minimize}} & \|x\|_1 \\ \text{subject to} & \|u - \Phi x\|_2 \leq \sigma \end{array}$$

(SP):

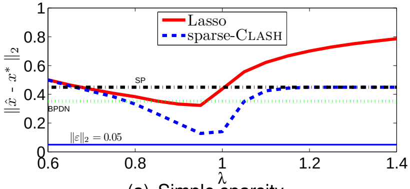
$$\begin{array}{ll} \underset{x}{\text{minimize}} & \|u - \Phi x\|_2^2 \\ \text{subject to} & \|x\|_0 \leq K \end{array}$$

(Lasso):

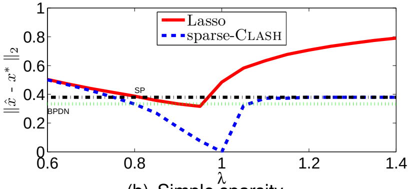
$$\begin{array}{ll} \underset{x}{\text{minimize}} & \|u - \Phi x\|_2 \\ \text{subject to} & \|x\|_1 \leq \lambda \end{array}$$

(sparse-CLASH):

$$\begin{array}{ll} \underset{x}{\text{minimize}} & \|u - \Phi x\|_2^2 \\ \text{subject to} & x \in \Sigma_K \\ & \|x\|_1 \leq \lambda \end{array}$$



(a) Simple sparsity



(b) Simple sparsity

Figure: *Median values of signal error $\|\hat{x} - x^*\|_2$. 500 Monte Carlo iterations. $N = 800,\ M = 240,\ K = 89$ and $\|\varepsilon\|_2 = 0.05$ (left column) and $N = 800,\ M = 250,\ K = 93$ in noiseless $\|\varepsilon\|_2 = 0$ (right column) setting.*

Examples - structured models

(BPDN):

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \|x\|_1 \\ \text{subject to} \quad & \|u - \Phi x\|_2 \leq \sigma \end{aligned}$$

(SP):

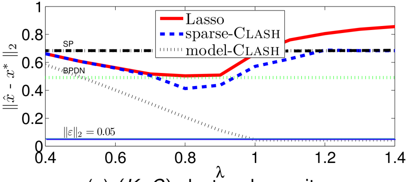
$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \|u - \Phi x\|_2^2 \\ \text{subject to} \quad & \|x\|_0 \leq K \end{aligned}$$

(Lasso):

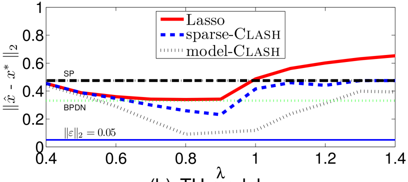
$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \|u - \Phi x\|_2 \\ \text{subject to} \quad & \|x\|_1 \leq \lambda \end{aligned}$$

(model-CLASH):

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \|u - \Phi x\|_2^2 \\ \text{subject to} \quad & x \in \Sigma_{\mathcal{M}_K} \\ & \|x\|_1 \leq \lambda \end{aligned}$$



(a) (K, C)-clustered sparsity



(b) TU model

Figure: *Median values of signal error* $\|\hat{x} - x^*\|_2$. *Bottom row:* 100 *Monte Carlo iterations.* $N = 500,\ M = 125,\ K = 50$. *The* (K, C)*-clustered sparsity model (left column) has* $C = 5$.

Conclusions

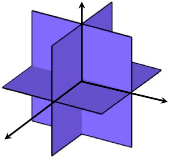
- The CLASH algorithm:
  1. Regression framework where combinatorial $+$ convex geometry interface for interpretable solutions.
  2. Special case: $\lambda \to \infty \Rightarrow$ model-CS.
- PMAP$_\epsilon$ property:
  1. Inherent difficulty in combinatorial selection.
  2. Beyond simple selection models: algorithmic definition of sparsity for various models.
  3. Provable solution quality and runtime bounds.
- Future work: other norms/constraints...

Geometry of CLASH: convex scale + simple sparsity

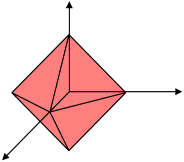
Simple sparsity selection
+
Least Absoulute Shrinkage

$\Longrightarrow$

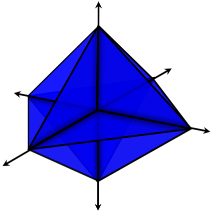
$$\mathcal{P}_{\Sigma_K}(x) = \text{argmin}_{\|y\|_0 \leq K} \|x - y\|_2^2$$
+
$$\mathcal{P}_{\lambda}(x) = \text{argmin}_{\|y\|_1 \leq \lambda} \|x - y\|_2^2$$



$+$



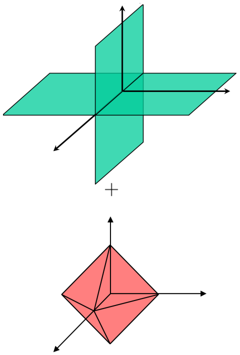
$\approx$



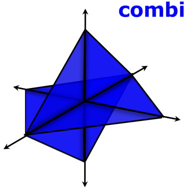
Structure sparsity selection
+
Least Absoulute Shrinkage

$\Longrightarrow$

$$\mathcal{P}_{\Sigma_{\mathcal{M}_K}}(x) = \text{argmin}_{\|y\|_0 \leq K} \|x - y\|_2^2$$
+
$$\mathcal{P}_{\lambda}(x) = \text{argmin}_{\|y\|_1 \leq \lambda} \|x - y\|_2^2$$



$\approx$



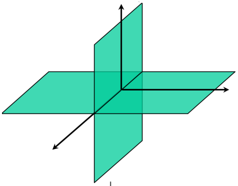
combinatorial origami

Geometry of CLASH: convex scale + structure sparsity

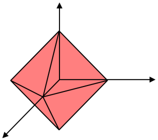
Structure sparsity selection
+
Least Absoulute Shrinkage

$\Longrightarrow$

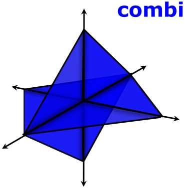
$$\mathcal{P}_{\Sigma_{\mathcal{M}_K}}(x) = \text{argmin}_{\|y\|_0 \leq K} \|x - y\|_2^2$$
$$+$$
$$\mathcal{P}_{\lambda}(x) = \text{argmin}_{\|y\|_1 \leq \lambda} \|x - y\|_2^2$$



$+$



$\approx$

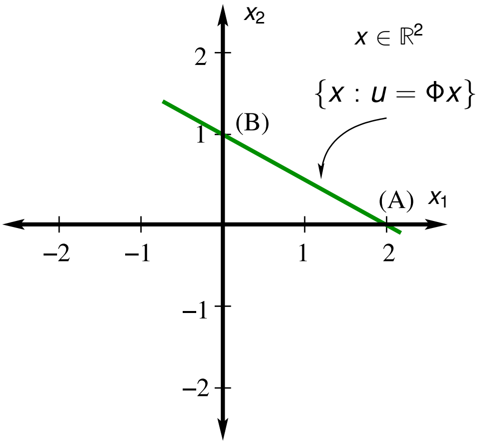


combinatorial origami

Recently: Sparse (structured) projections onto the simplex, ℓ_1-, ℓ_2-, and ℓ_∞-norm balls [Kyrillidis et al.'12].

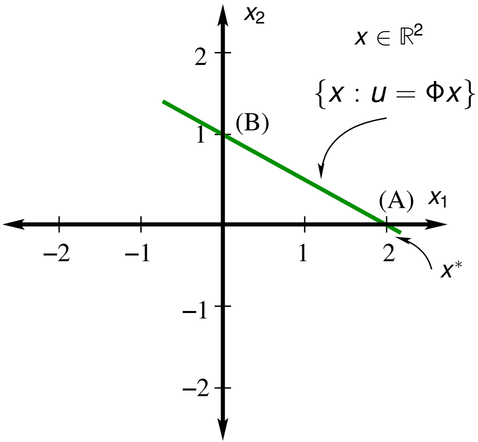
Some intuition - prior work

- Toy example setting:



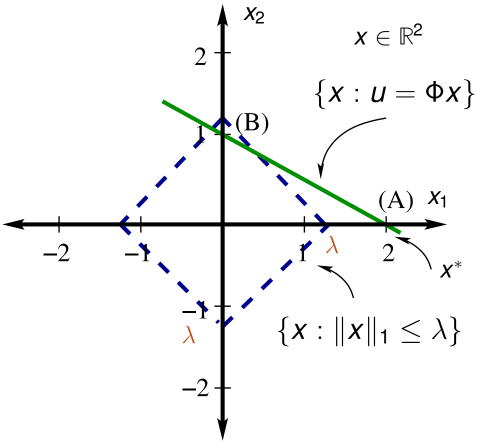
- Facts:
  1. For clarity, $\varepsilon = 0$.

- Toy example setting:



- Facts:
  1. For clarity, $\varepsilon = 0$.
  2. $\|x^*\|_1 = 2, \ \|x^*\|_0 = 1$.

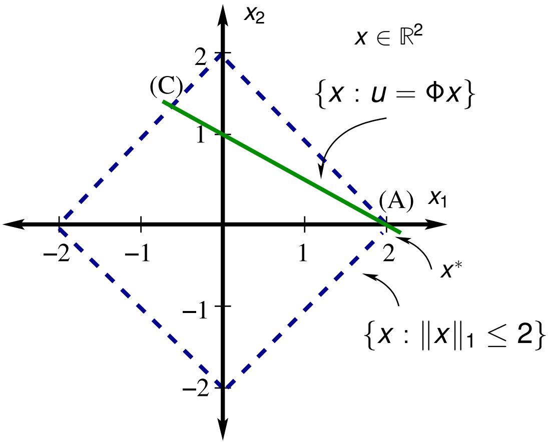
- Toy example setting [Tibshirani'96]:



- Facts:
  1. For clarity, $\varepsilon = 0$.
  2. $\|x^*\|_1 = 2, \ \|x^*\|_0 = 1$.
- **Lasso formulation**:

$$\underset{x}{\text{minimize}} \quad \|u - \Phi x\|_2 \qquad \text{subject to} \quad \|x\|_1 \leq \lambda$$

- Toy example setting [Tibshirani'96]:



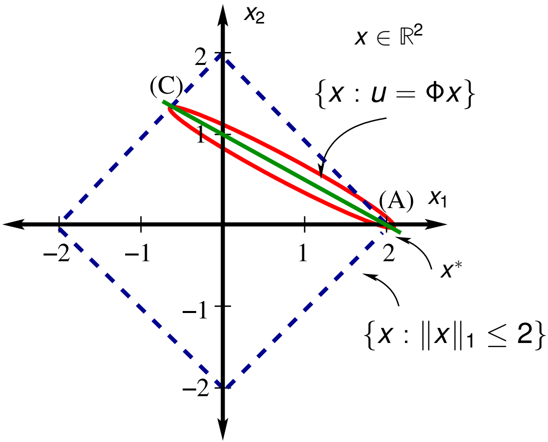
- Facts:
  1. For clarity, $\varepsilon = 0$.
  2. $\|x^*\|_1 = 2, \ \|x^*\|_0 = 1$.
- **Lasso formulation**:

$$\underset{x}{\text{minimize}} \quad \|u - \Phi x\|_2 \qquad \text{subject to} \quad \|x\|_1 \leq \mathbf{2}$$

- Assume we know *apriori:*

$$\lambda = \|x^*\|_1.$$

- Toy example setting [Tibshirani'96]:



- Facts:
  1. For clarity, $\varepsilon = 0$.
  2. $\|x^*\|_1 = 2, \ \|x^*\|_0 = 1$.
- **Lasso formulation**:

$$\begin{array}{ll} \underset{x}{\text{minimize}} & \|u - \Phi x\|_2 \\ \text{subject to} & \|x\|_1 \leq \mathbf{2} \end{array}$$

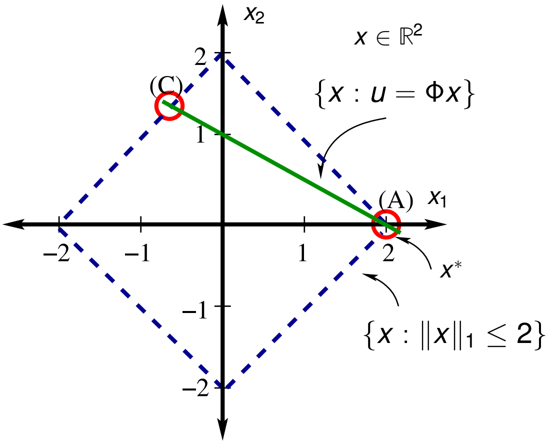
- Assume we know *apriori:*

$$\lambda = \|x^*\|_1.$$

Feasible Solution Candidate Set: $[(\mathrm{A}), \ldots, (\mathrm{C})]$.

Some intuition - prior work

- Toy example setting [Tibshirani'96]:



- Facts:
  1. For clarity, $\varepsilon = 0$.
  2. $\|x^*\|_1 = 2,\ \|x^*\|_0 = 1$.
- **Lasso formulation**:

$$\underset{x}{\text{minimize}} \quad \|u - \Phi x\|_2 \qquad \text{subject to} \quad \|x\|_1 = \mathbf{2}$$

- Assume we know *apriori:*

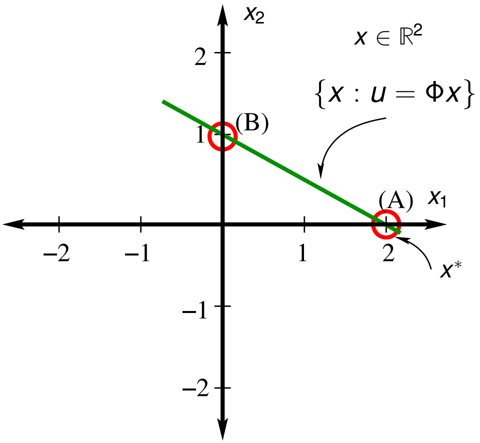
$$\lambda = \|x^*\|_1.$$

- For basic solutions: $\|x\|_1 = 2$.

Feasible Solution Candidate Set: $\{(A), (C)\}$.

Some intuition - prior work

- Toy example setting [Needell et al.'08, Dai et al.'09]:



- Facts:
  1. For clarity, $\varepsilon = 0$.
  2. $\|x^*\|_1 = 2, \ \|x^*\|_0 = 1$.
- **Greedy formulation**:

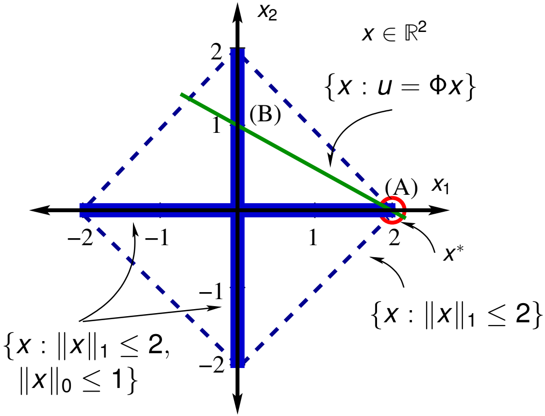
$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \|u - \Phi x\|_2 \\ \text{subject to} \quad & \|x\|_0 \leq K := 1 \end{aligned}$$

- Assume we know *apriori:*

$$K = \|x^*\|_0.$$

Feasible Solution Candidate Set: $\{(A), (B)\}$.

- Toy example setting:



- Facts:
  1. For clarity, $\varepsilon = 0$.
  2. $\|x^*\|_1 = 2, \ \|x^*\|_0 = 1$.
- **CLASH formulation**:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \|u - \Phi x\|_2 \\ \text{subject to} \quad & \|x\|_1 = 2 \\ & \|x\|_0 \leq 1 \end{aligned}$$

Feasible Solution Candidate Set: $\{(\text{A})\}$.