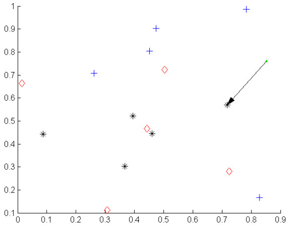
Compressed Sensing Meets Machine Learning
- Classification of Mixture Subspace Models via Sparse Representation

Allen Y. Yang
<yang@eecs.berkeley.edu>

Mini Lectures in Image Processing (Part II), UC Berkeley

Nearest Neighbor Algorithm



1. **Training**: Provide labeled samples for K classes.
2. **Test**: Present a new sample
   - Compute its distances with all training samples.
   - Assign its label as the same label of the nearest neighbor.

Nearest Subspace

Estimation of single subspace models

- Suppose $R = [\mathbf{w}_1, \cdots, \mathbf{w}_d]$ is a basis for a d-dim subspace in $\mathbb{R}^D$.
- For $\mathbf{x}_i \in \mathbb{R}^D$, its coordinate in the new coordinate system: $\mathbf{w}^T \mathbf{x}_i = y_i \in \mathbb{R}$.

Nearest Subspace

Estimation of single subspace models

- Suppose $R = [\mathbf{w}_1, \cdots, \mathbf{w}_d]$ is a basis for a d-dim subspace in $\mathbb{R}^D$.
- For $\mathbf{x}_i \in \mathbb{R}^D$, its coordinate in the new coordinate system: $\mathbf{w}^T \mathbf{x}_i = y_i \in \mathbb{R}$.
- **Principal component analysis**

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \sum_{i=1}^{n} (y_i)^2 = \arg\max \mathbf{w}^T \Sigma \mathbf{w}$$

Nearest Subspace

Estimation of single subspace models

- Suppose $R = [\mathbf{w}_1, \cdots, \mathbf{w}_d]$ is a basis for a d-dim subspace in $\mathbb{R}^D$.
- For $\mathbf{x}_i \in \mathbb{R}^D$, its coordinate in the new coordinate system: $\mathbf{w}^T \mathbf{x}_i = y_i \in \mathbb{R}$.
- **Principal component analysis**

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \sum_{i=1}^{n} (y_i)^2 = \arg\max \mathbf{w}^T \Sigma \mathbf{w}$$

- **Numerical solution**: Singular value decomposition (SVD)

$$\text{svd}(A) = USV^T, \text{ where } U \in \mathbb{R}^{D \times D}, S \in \mathbb{R}^{D \times n}, V \in \mathbb{R}^{n \times n}.$$

Denote $U = [U_1 \in \mathbb{R}^{D \times d}; U_2 \in \mathbb{R}^{D \times (D-d)}]$. Then $R = U_1^T$.

Nearest Subspace

Estimation of single subspace models

- Suppose $R = [\mathbf{w}_1, \cdots, \mathbf{w}_d]$ is a basis for a d-dim subspace in $\mathbb{R}^D$.
- For $\mathbf{x}_i \in \mathbb{R}^D$, its coordinate in the new coordinate system: $\mathbf{w}^T\mathbf{x}_i = y_i \in \mathbb{R}$.
- **Principal component analysis**

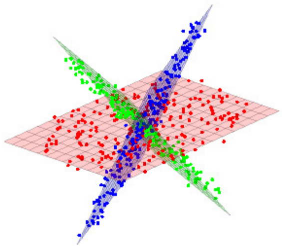
$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \sum_{i=1}^{n} (y_i)^2 = \arg\max \mathbf{w}^T \Sigma \mathbf{w}$$

- **Numerical solution**: Singular value decomposition (SVD)

$$\text{svd}(A) = USV^T, \text{ where } U \in \mathbb{R}^{D\times D}, S \in \mathbb{R}^{D\times n}, V \in \mathbb{R}^{n\times n}.$$

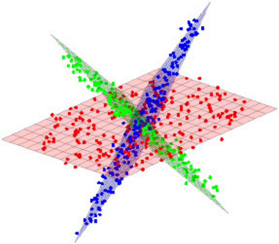
  Denote $U = [U_1 \in \mathbb{R}^{D\times d}; U_2 \in \mathbb{R}^{D\times(D-d)}]$. Then $R = U_1^T$.
- **Eigenfaces** If $\mathbf{x}_i$ are vectors of face images, the principal vectors $\mathbf{w}_i$ are then called Eigenfaces.

Nearest Subspace Algorithm



1. **Training**: For each of K classes, estimate its d-dim subspace model $R_i = [\mathbf{w}_1, \cdots, \mathbf{w}_d]$.
2. **Test**: Present a new sample $\mathbf{y}$, compute its distances to K subspaces.
3. **Assignment**: label of $\mathbf{y}$ as the closest subspace.

Nearest Subspace Algorithm



1. **Training**: For each of K classes, estimate its d-dim subspace model $R_i = [\mathbf{w}_1, \cdots, \mathbf{w}_d]$.
2. **Test**: Present a new sample $\mathbf{y}$, compute its distances to K subspaces.
3. **Assignment**: label of $\mathbf{y}$ as the closest subspace.

Question

- Equation for computing distance from $\mathbf{y}$ to R_i?
- Why NS likely outperforms NN?

Noiseless ℓ^1-Minimization is a Linear Program

Recall last lecture: Compute **sparsest** solution **x** that satisfies

$$\tilde{\mathbf{y}} = \tilde{A}\mathbf{x} \in \mathbb{R}^d$$

Noiseless ℓ^1-Minimization is a Linear Program

Recall last lecture: Compute **sparsest** solution $\mathbf{x}$ that satisfies

$$\tilde{\mathbf{y}} = \tilde{A}\mathbf{x} \in \mathbb{R}^d$$

Formulate as **linear programming**:

1. Problem statement:

$$(P_1): \quad \mathbf{x}^* = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_1 \text{ subject to } \tilde{\mathbf{y}} = \tilde{A}\mathbf{x} \in \mathbb{R}^d$$

Noiseless ℓ^1-Minimization is a Linear Program

Recall last lecture: Compute **sparsest** solution $\mathbf{x}$ that satisfies

$$\tilde{\mathbf{y}} = \tilde{A}\mathbf{x} \in \mathbb{R}^d$$

Formulate as **linear programming**:

1. Problem statement:

$$(P_1): \quad \mathbf{x}^* = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_1 \text{ subject to } \tilde{\mathbf{y}} = \tilde{A}\mathbf{x} \in \mathbb{R}^d$$

2. Denote $\Phi = (\tilde{A}, -\tilde{A}) \in \mathbb{R}^{d\times 2n}$, $\mathbf{c} = (1, 1, \cdots, 1)^T \in \mathbb{R}^{2n}$. We have the following linear program

$$\begin{aligned} \mathbf{w}^* \quad = \quad & \arg\min_{\mathbf{w}} \mathbf{c}^T\mathbf{w} \\ \text{subject to} \quad & \tilde{\mathbf{y}} = \Phi\mathbf{w} \\ & \mathbf{w} \geq 0 \end{aligned}$$

ℓ^1-Minimization Routines

- **Matching pursuit** [Mallat 1993]
  1. Find most correlated vector $\mathbf{v}_i$ in $\tilde{A}$ with $\mathbf{y}$: $i = \arg\max \langle \mathbf{y}, \mathbf{v}_j \rangle$.
  2. $\tilde{A} \leftarrow \tilde{A}^{\hat{i}}$, $x_i \leftarrow \langle \mathbf{y}, \mathbf{v}_i \rangle$, $\mathbf{y} \leftarrow \mathbf{y} - x_i \mathbf{v}_i$.
  3. Repeat until $\|\mathbf{y}\| < \epsilon$.

ℓ^1-Minimization Routines

- **Matching pursuit** [Mallat 1993]
  1. Find most correlated vector $\mathbf{v}_i$ in $\tilde{A}$ with $\mathbf{y}$: $i = \arg\max \langle \mathbf{y}, \mathbf{v}_j \rangle$.
  2. $\tilde{A} \leftarrow \tilde{A}^{\hat{i}}$, $x_i \leftarrow \langle \mathbf{y}, \mathbf{v}_i \rangle$, $\mathbf{y} \leftarrow \mathbf{y} - x_i \mathbf{v}_i$.
  3. Repeat until $\|\mathbf{y}\| < \epsilon$.
- **Basis pursuit** [Chen 1998]
  1. Assume $\mathbf{x}_0$ is m-sparse.
  2. Select m linearly independent vectors B_m in $\tilde{A}$ as a basis
  $$\mathbf{x}_m = B_m^\dagger \mathbf{y}.$$
  3. Repeat swapping one basis vector in B_m with another vector in $\tilde{A}$ if improve $\|\mathbf{y} - B_m \mathbf{x}_m\|$.
  4. If $\|\mathbf{y} - B_m \mathbf{x}_m\|_2 < \epsilon$, stop.

ℓ^1-Minimization Routines

- **Matching pursuit** [Mallat 1993]
  1. Find most correlated vector $\mathbf{v}_i$ in $\tilde{A}$ with $\mathbf{y}$: $i = \arg\max \langle \mathbf{y}, \mathbf{v}_j \rangle$.
  2. $\tilde{A} \leftarrow \tilde{A}^{\hat{i}}$, $x_i \leftarrow \langle \mathbf{y}, \mathbf{v}_i \rangle$, $\mathbf{y} \leftarrow \mathbf{y} - x_i \mathbf{v}_i$.
  3. Repeat until $\|\mathbf{y}\| < \epsilon$.
- **Basis pursuit** [Chen 1998]
  1. Assume $\mathbf{x}_0$ is m-sparse.
  2. Select m linearly independent vectors B_m in $\tilde{A}$ as a basis

$$\mathbf{x}_m = B_m^\dagger \mathbf{y}.$$

  3. Repeat swapping one basis vector in B_m with another vector in $\tilde{A}$ if improve $\|\mathbf{y} - B_m \mathbf{x}_m\|$.
  4. If $\|\mathbf{y} - B_m \mathbf{x}_m\|_2 < \epsilon$, stop.

Matlab Toolboxes

- **SparseLab** by Donoho at Stanford.
- **cvx** by Boyd at Stanford.

ℓ^1-Minimization with Bounded ℓ^2-Noise is Quadratic Programming

ℓ^1-Minimization with Bounded ℓ^2-Noise:

$$\tilde{\mathbf{y}} = \tilde{A}\mathbf{x}_0 + \mathbf{z} \in \mathbb{R}^d, \text{ where } \|\mathbf{z}\|_2 < \epsilon$$

ℓ^1-Minimization with Bounded ℓ^2-Noise is Quadratic Programming

ℓ^1-Minimization with Bounded ℓ^2-Noise:

$$\tilde{\mathbf{y}} = \tilde{A}\mathbf{x}_0 + \mathbf{z} \in \mathbb{R}^d, \text{ where } \|\mathbf{z}\|_2 < \epsilon$$

- Problem statement:

$$(P_1') : \quad \mathbf{x}^* = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_1 \text{ subject to } \|\tilde{\mathbf{y}} - \tilde{A}\mathbf{x}\|_2 < \epsilon$$

ℓ^1-Minimization with Bounded ℓ^2-Noise is Quadratic Programming

ℓ^1-Minimization with Bounded ℓ^2-Noise:

$$\tilde{\mathbf{y}} = \tilde{A}\mathbf{x}_0 + \mathbf{z} \in \mathbb{R}^d, \text{ where } \|\mathbf{z}\|_2 < \epsilon$$

- Problem statement:

$$(P_1') : \quad \mathbf{x}^* = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_1 \text{ subject to } \|\tilde{\mathbf{y}} - \tilde{A}\mathbf{x}\|_2 < \epsilon$$

- Quadratic program:

$$\mathbf{x}^* = \arg\min\{\|\mathbf{x}\|_1 + \lambda\|\mathbf{y} - \tilde{A}\mathbf{x}\|_2\}$$

ℓ^1-Minimization with Bounded ℓ^2-Noise is Quadratic Programming

ℓ^1-Minimization with Bounded ℓ^2-Noise:

$$\tilde{\mathbf{y}} = \tilde{A}\mathbf{x}_0 + \mathbf{z} \in \mathbb{R}^d, \text{ where } \|\mathbf{z}\|_2 < \epsilon$$

- Problem statement:

$$(P_1') : \quad \mathbf{x}^* = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_1 \text{ subject to } \|\tilde{\mathbf{y}} - \tilde{A}\mathbf{x}\|_2 < \epsilon$$

- Quadratic program:

$$\mathbf{x}^* \quad = \quad \arg\min\{\|\mathbf{x}\|_1 + \lambda\|\mathbf{y} - \tilde{A}\mathbf{x}\|_2\}$$

- Matlab toolboxes:
  ℓ^1-**Magic** by Candès at Caltech.
  cvx by Boyd at Stanford.

Recall last lecture...

1. ℓ^0-Minimization

$$\mathbf{x}_0 = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_0 \text{ s.t. } \tilde{\mathbf{y}} = \tilde{A}\mathbf{x}.$$

$\|\cdot\|_0$ simply counts the number of nonzero terms.

Recall last lecture...

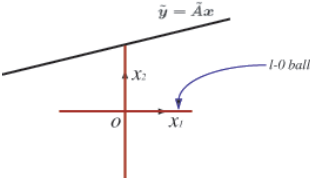
1. ℓ^0-Minimization

$$\mathbf{x}_0 = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_0 \text{ s.t. } \tilde{\mathbf{y}} = \tilde{A}\mathbf{x}.$$

   $\|\cdot\|_0$ simply counts the number of nonzero terms.

2. ℓ^0-Ball

- ℓ^0-ball is not convex.
- ℓ^0-minimization is NP-hard.



ℓ^1/ℓ^0 Equivalence

1. **Compressed sensing**: If $\mathbf{x}_0$ is *sparse enough*, ℓ^0-minimization is equivalent to

$$(P_1) \quad \min \|\mathbf{x}\|_1 \text{ s.t. } \tilde{\mathbf{y}} = \tilde{A}\mathbf{x}.$$

$\|\mathbf{x}\|_1 = |x_1| + |x_2| + \cdots + |x_n|$.

ℓ^1/ℓ^0 Equivalence

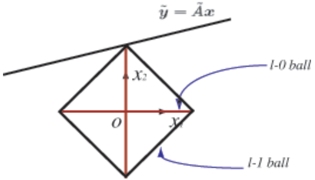
1. **Compressed sensing**: If $\mathbf{x}_0$ is *sparse enough*, ℓ^0-minimization is equivalent to

$$(P_1) \quad \min \|\mathbf{x}\|_1 \text{ s.t. } \tilde{\mathbf{y}} = \tilde{A}\mathbf{x}.$$

$\|\mathbf{x}\|_1 = |x_1| + |x_2| + \cdots + |x_n|$.

2. ℓ^1-Ball

- ℓ^1-Minimization is convex.
- Solution equal to ℓ^0-minimization.



ℓ^1/ℓ^0 Equivalence

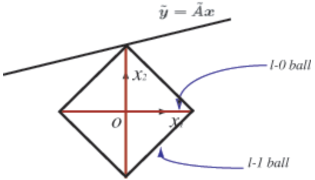
1. **Compressed sensing**: If $\mathbf{x}_0$ is *sparse enough*, ℓ^0-minimization is equivalent to

$$(P_1) \quad \min \|\mathbf{x}\|_1 \text{ s.t. } \tilde{\mathbf{y}} = \tilde{A}\mathbf{x}.$$

$\|\mathbf{x}\|_1 = |x_1| + |x_2| + \cdots + |x_n|$.

2. ℓ^1-Ball

- ℓ^1-Minimization is convex.
- Solution equal to ℓ^0-minimization.



3. ℓ^1/ℓ^0 Equivalence: [Donoho 2002, 2004; Candes et al. 2004; Baraniuk 2006]
Given $\tilde{\mathbf{y}} = \tilde{A}\mathbf{x}_0$, there exists **equivalence breakdown point** (EBP) $\rho(\tilde{A})$, if $\|\mathbf{x}_0\|_0 < \rho$:
   - ℓ^1-solution is unique
   - $\mathbf{x}_1 = \mathbf{x}_0$

ℓ^1/ℓ^0 Equivalence in Noisy Case

Reconsider ℓ^2-bounded linear system

$$\tilde{\mathbf{y}} = \tilde{A}\mathbf{x}_0 + \mathbf{z} \in \mathbb{R}^d, \text{ where } \|\mathbf{z}\|_2 < \epsilon$$

Is corresponding ℓ^1 solution stable?

ℓ^1/ℓ^0 Equivalence in Noisy Case

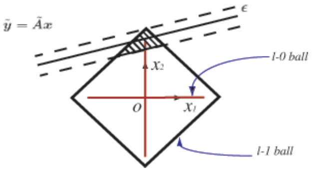
Reconsider ℓ^2-bounded linear system

$$\tilde{\mathbf{y}} = \tilde{A}\mathbf{x}_0 + \mathbf{z} \in \mathbb{R}^d, \text{ where } \|\mathbf{z}\|_2 < \epsilon$$

Is corresponding ℓ^1 solution stable?

1. ℓ^1-Ball

- No exact solution possible.
- Bounded measurement error causes bounded estimation error.
- Yes, ℓ^1 solution is stable!



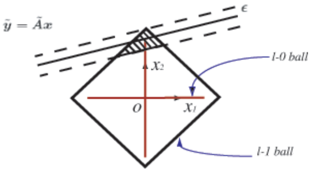
ℓ^1/ℓ^0 Equivalence in Noisy Case

Reconsider ℓ^2-bounded linear system

$$\tilde{\mathbf{y}} = \tilde{A}\mathbf{x}_0 + \mathbf{z} \in \mathbb{R}^d, \text{ where } \|\mathbf{z}\|_2 < \epsilon$$

Is corresponding ℓ^1 solution stable?

1. ℓ^1-Ball
   - No exact solution possible.
   - Bounded measurement error causes bounded estimation error.
   - Yes, ℓ^1 solution is stable!



2. ℓ^1/ℓ^0 Equivalence [Donoho 2004]
   Suppose $\tilde{\mathbf{y}} = \tilde{A}\mathbf{x}_0 + \mathbf{z}$ where $\|\mathbf{z}\|_2 < \epsilon$. There exists **equivalence breakdown point** (EBP) $\rho(\tilde{A})$, if $\|\mathbf{x}_0\|_0 < \rho$:

$$\|\mathbf{x}_1 - \mathbf{x}_0\|_2 \leq C \cdot \epsilon$$

Compressed Sensing in the View of Convex Polytopes

For the rest of the lecture, investigate the estimation of EBP ρ.

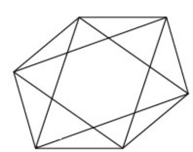
To simplify notations, assume underdetermined system $\mathbf{y} = A\mathbf{x} \in \mathbb{R}^d$, where $A = \mathbb{R}^{d \times n}$.

Compressed Sensing in the View of Convex Polytopes

For the rest of the lecture, investigate the estimation of EBP ρ.
To simplify notations, assume underdetermined system $\mathbf{y} = A\mathbf{x} \in \mathbb{R}^d$, where $A = \mathbb{R}^{d\times n}$.

Definition (Quotient Polytopes)

Consider the convex hull P of the $2n$ vectors $(A, -A)$. P is called the **quotient polytope** associated to A.

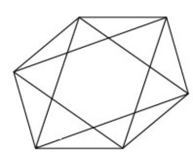


Compressed Sensing in the View of Convex Polytopes

For the rest of the lecture, investigate the estimation of EBP ρ.
To simplify notations, assume underdetermined system $\mathbf{y} = A\mathbf{x} \in \mathbb{R}^d$, where $A = \mathbb{R}^{d\times n}$.

Definition (Quotient Polytopes)

Consider the convex hull P of the $2n$ vectors $(A, -A)$. P is called the **quotient polytope** associated to A.

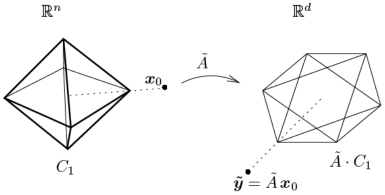


Definition (k-Neighborliness)

A quotient polytope P is called k-**neighborly** if whenever we take k vertices not including an antipodal pair, the resulting vertices span a face of P.
(Above example is 1-neighborly.)

ℓ^1-Minimization and Quotient Polytopes

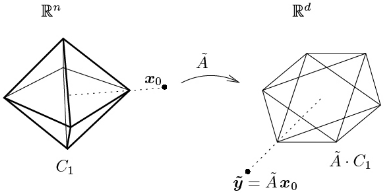
Why ℓ^1-minimization is related to quotient polytopes?



- Consider **x** represent an ℓ^1-ball C in $\mathbb{R}^n$.

ℓ^1-Minimization and Quotient Polytopes

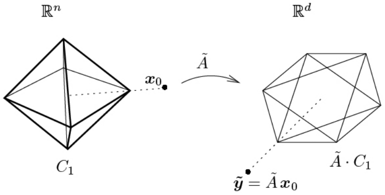
Why ℓ^1-minimization is related to quotient polytopes?



- Consider $\mathbf{x}$ represent an ℓ^1-ball C in $\mathbb{R}^n$.
- If $\mathbf{x}_0$ is k-sparse, $\mathbf{x}_0$ will intersect the ℓ^1-ball on one of its $(k-1)$-D faces.

ℓ^1-Minimization and Quotient Polytopes

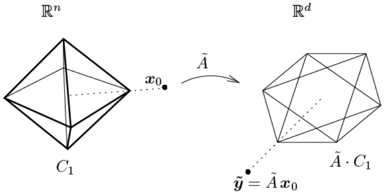
Why ℓ^1-minimization is related to quotient polytopes?



- Consider $\mathbf{x}$ represent an ℓ^1-ball C in $\mathbb{R}^n$.
- If $\mathbf{x}_0$ is k-sparse, $\mathbf{x}_0$ will intersect the ℓ^1-ball on one of its $(k-1)$-D faces.
- Matrix A maps ℓ^1-ball in $\mathbb{R}^n$ to the quotient polytope P in $\mathbb{R}^d$, $d \ll n$.

ℓ^1-Minimization and Quotient Polytopes

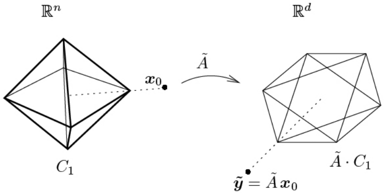
Why ℓ^1-minimization is related to quotient polytopes?



- Consider $\mathbf{x}$ represent an ℓ^1-ball C in $\mathbb{R}^n$.
- If $\mathbf{x}_0$ is k-sparse, $\mathbf{x}_0$ will intersect the ℓ^1-ball on one of its $(k-1)$-D faces.
- Matrix A maps ℓ^1-ball in $\mathbb{R}^n$ to the quotient polytope P in $\mathbb{R}^d$, $d \ll n$.
- Such mapping is linear!

ℓ^1-Minimization and Quotient Polytopes

Why ℓ^1-minimization is related to quotient polytopes?

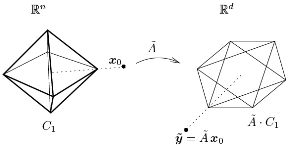


- Consider $\mathbf{x}$ represent an ℓ^1-ball C in $\mathbb{R}^n$.
- If $\mathbf{x}_0$ is k-sparse, $\mathbf{x}_0$ will intersect the ℓ^1-ball on one of its $(k-1)$-D faces.
- Matrix A maps ℓ^1-ball in $\mathbb{R}^n$ to the quotient polytope P in $\mathbb{R}^d$, $d \ll n$.
- Such mapping is linear!

Theorem (ℓ^1/ℓ^0 equivalence condition)

If the quotient polytope P associated with A is k-neighborly, for $\mathbf{y} = A\mathbf{x}_0$ with $\mathbf{x}_0$ to be k-sparse, then $\mathbf{x}_0$ is the unique optimal solution of the ℓ^1-minimization.

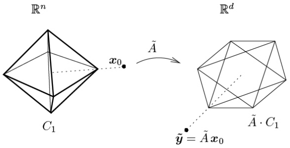
Let's prove the theorem together



Definitions:

- **vertices v** $\in$ vert(P).

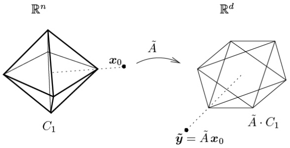
Let's prove the theorem together



Definitions:

- **vertices v** $\in \text{vert}(P)$.
- k-D **faces** $F \in \mathcal{F}_k(P)$. Also define $f_k(P) = \#\mathcal{F}_k(P)$.

Let's prove the theorem together

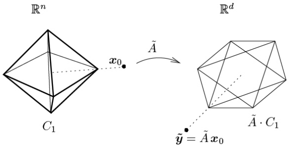


Definitions:

- **vertices v** $\in$ vert(P).
- k-D **faces** $F \in \mathcal{F}_k(P)$. Also define $f_k(P) = \#\mathcal{F}_k(P)$.
- **convex hull** operation conv$(\cdot)$.

(1) vert$(P) = \mathcal{F}_0(P)$. (2) $P =$ conv(vert(P))

Let's prove the theorem together



Definitions:

- **vertices v** $\in$ vert(P).
- k-D **faces** $F \in \mathcal{F}_k(P)$. Also define $f_k(P) = \#\mathcal{F}_k(P)$.
- **convex hull** operation conv$(\cdot)$.

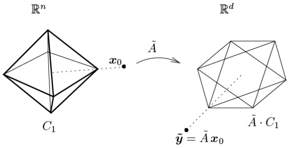
(1) vert$(P) = \mathcal{F}_0(P)$. (2) $P = \text{conv}(\text{vert}(P))$

- $F \in \mathcal{F}_k(P)$ is a **simplex** if $\#\text{vert}(F) = k + 1$.

Properties

$$\text{vert}(AC) \subset A\text{vert}(C); \quad \mathcal{F}_l(AC) \subset A\mathcal{F}_l(C).$$

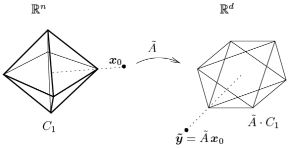
Two Fundamental Lemmas



Lemma (Alternative Definition of k-neighborliness)

*Suppose a centrosymmetric polytope $P = AC$ has $2n$ vertices. Then P is k-**neighborly** iff for any $l = 0, \cdots, k-1$ and $F \in \mathcal{F}_l(C)$, $AF \in \mathcal{F}_l(AC)$.*

Two Fundamental Lemmas



Lemma (Alternative Definition of k-neighborliness)

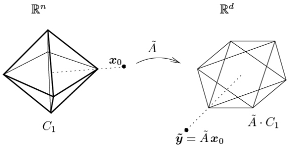
*Suppose a centrosymmetric polytope $P = AC$ has $2n$ vertices. Then P is k-***neighborly** *iff for any $l = 0, \cdots, k-1$ and $F \in \mathcal{F}_l(C)$, $AF \in \mathcal{F}_l(AC)$.*

Lemma (Unique Representation on Simplices)

Consider an l-simplex $F \in \mathcal{F}_l(P)$. Let $\mathbf{x} \in F$. Then

1. $\mathbf{x}$ *has a* **unique representation** *as a linear combination of the vertices of P.*
2. *This representation places* **only nonzero weight** *on vertices of F.*

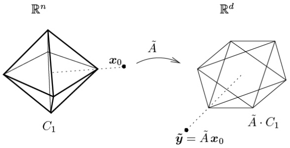
Proof of the Theorem



Suppose P is k-neighborly, and $\mathbf{x}_0$ is k-sparse. WLOG, scale and assume $\|\mathbf{x}_0\|_1 = 1$.

1. $\mathbf{x}_0$ is k-sparse $\Rightarrow \exists F \in \mathcal{F}_{k-1}(C)$, $\mathbf{x}_0 \in F$ and $\mathbf{y} \doteq A\mathbf{x}_0 \in AF$.
2. $P = AC$ is k-neighborly $\Rightarrow AF \in \mathcal{F}_{k-1}(AC)$ is a simplex.
3. By (1) and (2), $\mathbf{y} \in AF$ has a unique representation with at most k nonzero weights on the vertices of AF.
4. Hence, $\mathbf{x}_1$ given by ℓ^1-minimization is unique, and $\mathbf{x}_1 = \mathbf{x}_0$.

Proof of the Theorem



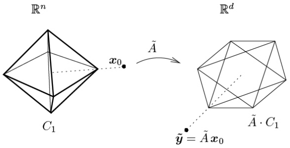
Suppose P is k-neighborly, and $\mathbf{x}_0$ is k-sparse. WLOG, scale and assume $\|\mathbf{x}_0\|_1 = 1$.

1. $\mathbf{x}_0$ is k-sparse $\Rightarrow \exists F \in \mathcal{F}_{k-1}(C)$, $\mathbf{x}_0 \in F$ and $\mathbf{y} \doteq A\mathbf{x}_0 \in AF$.
2. $P = AC$ is k-neighborly $\Rightarrow AF \in \mathcal{F}_{k-1}(AC)$ is a simplex.
3. By (1) and (2), $\mathbf{y} \in AF$ has a unique representation with at most k nonzero weights on the vertices of AF.
4. Hence, $\mathbf{x}_1$ given by ℓ^1-minimization is unique, and $\mathbf{x}_1 = \mathbf{x}_0$.

Corollary [Gribonval & Nielsen 2003]

Assume for all columns of matrix A, $\|\mathbf{v}_i\|_2 = 1$, and for all $i \neq j$, $\langle \mathbf{v}_i, \mathbf{v}_j \rangle \leq \frac{1}{2k-1}$, then $P = AC$ is k-neighborly.

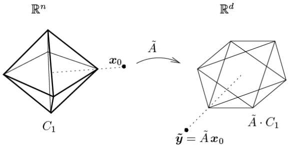
Last question: Why random projection works well in ℓ^1-minimization?



Revisit the above corollary

Define **coherence** $M \doteq \max_{i \neq j} |\langle \mathbf{v}_i, \mathbf{v}_j \rangle|$, then $\text{EBP}(A) > \frac{M^{-1}+1}{2}$.

Last question: Why random projection works well in ℓ^1-minimization?

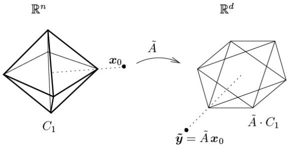


Revisit the above corollary

Define **coherence** $M \doteq \max_{i \neq j} |\langle \mathbf{v}_i, \mathbf{v}_j \rangle|$, then $\mathrm{EBP}(A) > \frac{M^{-1}+1}{2}$.

1. in HD space $\mathbb{R}^d$, two randomly generated unit vectors have small coherence M.

Last question: Why random projection works well in ℓ^1-minimization?



Revisit the above corollary

Define **coherence** $M \doteq \max_{i \neq j} |\langle \mathbf{v}_i, \mathbf{v}_j \rangle|$, then $\mathrm{EBP}(A) > \frac{M^{-1}+1}{2}$.

1. in HD space $\mathbb{R}^d$, two randomly generated unit vectors have small coherence M.
2. Further define **coherence** of two dictionaries $M(A, B) = \max_{\mathbf{u} \in A, \mathbf{v} \in B} |\langle \mathbf{u}, \mathbf{v} \rangle|$.
    - $\frac{1}{\sqrt{d}} \leq M(A, B) \leq 1$.
    - Let T be the spike basis in time domain, F be the Fourier basis, then $M(T, F) = \frac{1}{\sqrt{d}}$. **Max incoherence!**
    - Random projection R in general is not coherent with most traditional bases.

Conclusion

1. Classical classifiers: NN & NS.
2. Linear and quadratic ℓ^1 solvers.
3. Stability of ℓ^0/ℓ^1 equivalence with bounded error.
4. Computation of equivalence breakdown point (EBP) via quotient polytopes.