Feasibility problems: from alternating projections to matrix completions

Dmitriy Drusvyatskiy
C&O, Waterloo
Mathematics, Washington

Joint work with
V. Cheung (Waterloo), A.D. Ioffe (Technion), N. Krislock (NIU),
A.S. Lewis (Cornell), G. Pataki (UNC), and H. Wolkowicz (Waterloo)

March 21, 2014

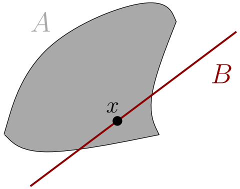
Outline

- **Method of alternating projections**

- **Euclidean distance completions: when alternating projections fail**

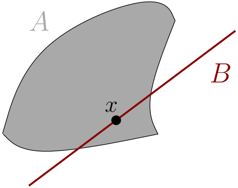
Method of alternating projections

Given sets $A, B \subset \mathbf{R}^n$, find some point $x \in A \cap B$.



Method of alternating projections

Given sets $A, B \subset \mathbf{R}^n$, find some point $x \in A \cap B$.

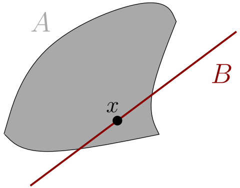


Distance and projection:

$$d_B(x) = \min_{y \in B} |x - y| \quad \text{and} \quad \mathcal{P}_B(x) = \{\text{nearest points of } B \text{ to } x\}.$$

Method of alternating projections

Given sets $A, B \subset \mathbf{R}^n,$ find some point $x \in A \cap B.$



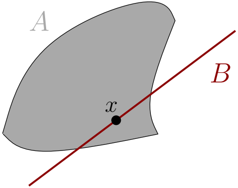
Distance and projection:

$$d_B(x) = \min_{y \in B} |x - y| \quad \text{and} \quad \mathcal{P}_B(x) = \{\text{nearest points of } B \text{ to } x\}.$$

Finding points in $\mathcal{P}_A(x)$ and $\mathcal{P}_B(x)$ is often easy!

Method of alternating projections

Given sets $A, B \subset \mathbf{R}^n$, find some point $x \in A \cap B$.



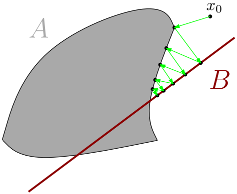
Distance and projection:

$$d_B(x) = \min_{y \in B} |x - y| \quad \text{and} \quad \mathcal{P}_B(x) = \{\text{nearest points of } B \text{ to } x\}.$$

Finding points in $\mathcal{P}_A(x)$ and $\mathcal{P}_B(x)$ is often easy!

Method of alternating projections (von Neumann '33):

$$\begin{array}{lll} x_{k+1} & \in & \mathcal{P}_B(x_k) \\ x_{k+2} & \in & \mathcal{P}_A(x_{k+1}) \end{array}$$



Method of alternating projections

Linear programming:

$$\{x : x \geq 0\} \bigcap \{x : Ax = b\}.$$

Method of alternating projections

Linear programming:

$$\{x : x \geq 0\} \bigcap \{x : Ax = b\}.$$

- $\mathcal{P}_{\mathbf{R}^n_+}(y) = y^+$

Method of alternating projections

Linear programming:

$$\{x : x \geq 0\} \bigcap \{x : Ax = b\}.$$

- $\mathcal{P}_{\mathbf{R}^n_+}(y) = y^+$

Semi-definite programming:

$$\{X : X \succeq 0\} \bigcap \{X : \mathcal{A}(X) = b\}.$$

Method of alternating projections

Linear programming:

$$\{x : x \geq 0\} \bigcap \{x : Ax = b\}.$$

- $\mathcal{P}_{\mathbf{R}^n_+}(y) = y^+$

Semi-definite programming:

$$\{X : X \succeq 0\} \bigcap \{X : \mathcal{A}(X) = b\}.$$

- $\mathcal{P}_{\mathcal{S}^n_+}(Y) = U^T [\mathrm{Diag}\, \lambda(Y)]^+ U$

Method of alternating projections

Linear programming:

$$\{x : x \geq 0\} \bigcap \{x : Ax = b\}.$$

- $\mathcal{P}_{\mathbf{R}^n_+}(y) = y^+$

Semi-definite programming:

$$\{X : X \succeq 0\} \bigcap \{X : \mathcal{A}(X) = b\}.$$

- $\mathcal{P}_{\mathcal{S}^n_+}(Y) = U^T [\operatorname{Diag} \lambda(Y)]^+ U$

Compressed sensing:

$$\{x : \operatorname{rank} x \leq r\} \bigcap \{x : Ax = b\}.$$

Method of alternating projections

Linear programming:

$$\{x : x \geq 0\} \bigcap \{x : Ax = b\}.$$

- $\mathcal{P}_{\mathbf{R}^n_+}(y) = y^+$

Semi-definite programming:

$$\{X : X \succeq 0\} \bigcap \{X : \mathcal{A}(X) = b\}.$$

- $\mathcal{P}_{\mathcal{S}^n_+}(Y) = U^T \left[\operatorname{Diag} \lambda(Y)\right]^+ U$

Compressed sensing:

$$\{x : \operatorname{rank} x \leq r\} \bigcap \{x : Ax = b\}.$$

- $\mathcal{P}_{\{x :\, \operatorname{rank} x \leq r\}}(y) \Longleftrightarrow$ set $n - r$ smallest coordinates in $|\cdot|$ to zero.

Method of alternating projections

Linear programming:

$$\{x : x \geq 0\} \bigcap \{x : Ax = b\}.$$

- $\mathcal{P}_{\mathbf{R}^n_+}(y) = y^+$

Semi-definite programming:

$$\{X : X \succeq 0\} \bigcap \{X : \mathcal{A}(X) = b\}.$$

- $\mathcal{P}_{\mathcal{S}^n_+}(Y) = U^T [\operatorname{Diag} \lambda(Y)]^+ U$

Compressed sensing:

$$\{x : \operatorname{rank} x \leq r\} \bigcap \{x : Ax = b\}.$$

- $\mathcal{P}_{\{x:\, \operatorname{rank} x \leq r\}}(y) \Longleftrightarrow$ set $n - r$ smallest coordinates in $|\cdot|$ to zero.

Low-order control:

$$\{X \succeq 0 : \operatorname{rank} X \leq r\} \bigcap \{X : \mathcal{A}(X) = b\}.$$

Method of alternating projections

Linear programming:

$$\{x : x \geq 0\} \bigcap \{x : Ax = b\}.$$

- $\mathcal{P}_{\mathbf{R}^n_+}(y) = y^+$

Semi-definite programming:

$$\{X : X \succeq 0\} \bigcap \{X : \mathcal{A}(X) = b\}.$$

- $\mathcal{P}_{\mathcal{S}^n_+}(Y) = U^T [\mathrm{Diag}\,\lambda(Y)]^+ U$

Compressed sensing:

$$\{x : \mathrm{rank}\, x \leq r\} \bigcap \{x : Ax = b\}.$$

- $\mathcal{P}_{\{x:\, \mathrm{rank}\, x \leq r\}}(y) \iff$ set $n - r$ smallest coordinates in $|\cdot|$ to zero.

Low-order control:

$$\{X \succeq 0 : \mathrm{rank}\, X \leq r\} \bigcap \{X : \mathcal{A}(X) = b\}.$$

- $\mathcal{P}_{\{X \succeq 0:\, \mathrm{rank}\, X \leq r\}}(Y) \iff$ diagonalize, set $n - r$ smallest eigenvalues to zero, set negative eigenvalues to zero.

Successes

Alternating projection heuristics remain popular!

Successes

Alternating projection heuristics remain popular!

Simple:

- require a few lines of code,
- no data structures, not memory or space intensive

Successes

Alternating projection heuristics remain popular!

Simple:

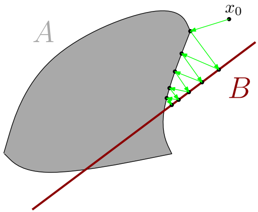
- require a few lines of code,
- no data structures, not memory or space intensive

Versatile:

- Inverse eigenvalue problems (Chen, Chu '96)
- Pole placement (Orsi, Yang '06)
- Information theory (Tropp, Dhillon, Heath, Strohmer '05)
- Low-order control design (Grigoriadis, Skelton '96)
- Image processing (Bauschke, Combettes, Luke '02)
- Hubble telescope (NASA '95)

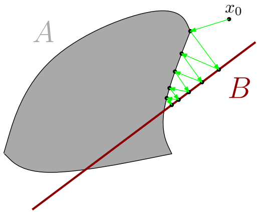
Convergence of alternating projections

$$\begin{array}{lcl} x_{k+1} & \in & \mathcal{P}_B(x_k) \\ x_{k+2} & \in & \mathcal{P}_A(x_{k+1}) \end{array}$$



Convergence of alternating projections

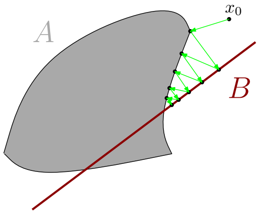
$$\begin{array}{|lcl|}\hline x_{k+1} & \in & \mathcal{P}_B(x_k) \\ x_{k+2} & \in & \mathcal{P}_A(x_{k+1}) \\ \hline\end{array}$$



The "angle" between A and B drives the convergence!

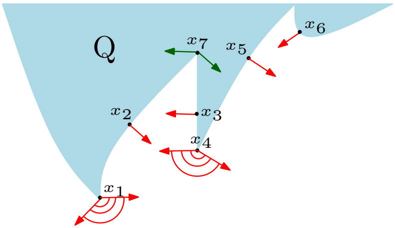
Convergence of alternating projections

$$\boxed{\begin{array}{lcl} x_{k+1} & \in & \mathcal{P}_B(x_k) \\ x_{k+2} & \in & \mathcal{P}_A(x_{k+1}) \end{array}}$$



The "angle" between A and B drives the convergence!

Key tool: Normal cones $N_A(x)$ and $N_B(x)$



Convergence of alternating projections

Transversality: $N_B(\bar{x}) \cap -N_A(\bar{x}) = \{0\}.$

Convergence of alternating projections

Transversality: $N_B(\bar{x}) \cap -N_A(\bar{x}) = \{0\}$.

Eg:

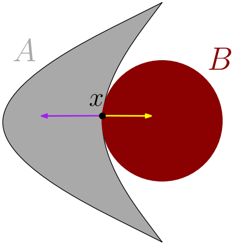


Figure : Not transverse

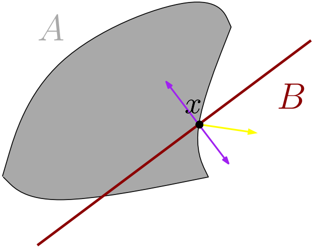


Figure : Transverse

Convergence of alternating projections

Transversality: $\qquad N_B(\bar{x}) \cap -N_A(\bar{x}) = \{0\}.$

Eg:

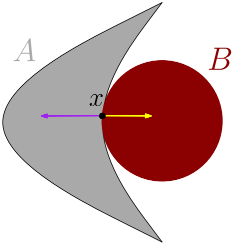


Figure : Not transverse

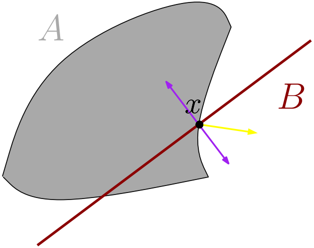


Figure : Transverse

Convergence of alternating projections (D-Ioffe-Lewis '13):

A and B transverse at $\bar{x}$ $\quad \Longrightarrow \quad$ local **R**-linear convergence.

Convergence of alternating projections

Transversality: $$N_B(\bar{x}) \cap -N_A(\bar{x}) = \{0\}.$$

Eg:

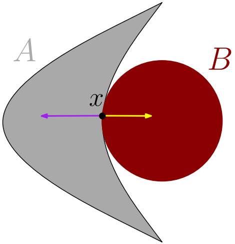


Figure : Not transverse

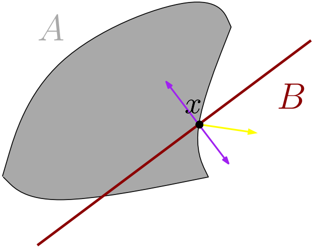


Figure : Transverse

Convergence of alternating projections (D-Ioffe-Lewis '13):

$$A \text{ and } B \text{ transverse at } \bar{x} \quad \Longrightarrow \quad \text{local } \mathbf{R}\text{-linear convergence.}$$

All earlier results imposed technical properties on the sets; eg (Lewis-Malick '07, Bauschke et al ' 11).

Slope and error bounds

Common problem: Estimate

$$\operatorname{dist}(x, [f \leq r]) \quad \text{(difficult)}$$

Slope and error bounds

Common problem: Estimate

$$\mathrm{dist}\,(x, [f \leq r]) \quad \text{(difficult)}$$

"The residual":

$$f(x) - r \qquad \text{(easy)}$$

Slope and error bounds

Common problem: Estimate

$$\mathrm{dist}\,(x, [f \leq r]) \quad \text{(difficult)}$$

"The residual":

$$f(x) - r \qquad \text{(easy)}$$

Desirable quality: Exists κ with

$$\mathrm{dist}\,(x, [f \leq r]) \leq \kappa(f(x) - r)$$

Slope and error bounds

Common problem: Estimate

$$\operatorname{dist}(x, [f \le r]) \quad \text{(difficult)}$$

"The residual":

$$f(x) - r \qquad \text{(easy)}$$

Desirable quality: Exists κ with

$$\operatorname{dist}(x, [f \le r]) \le \kappa(f(x) - r)$$

Eg: Hoffman's error bound for linear programming.

Slope and error bounds

Common problem: Estimate

$$\operatorname{dist}(x, [f \leq r]) \quad \text{(difficult)}$$

"The residual":

$$f(x) - r \qquad \text{(easy)}$$

Desirable quality: Exists κ with

$$\operatorname{dist}(x, [f \leq r]) \leq \kappa(f(x) - r)$$

Eg: Hoffman's error bound for linear programming.

Slope: "Fastest instantaneous rate of decrease"

$$\boxed{|\nabla f|(\bar{x}) := \limsup_{x \to \bar{x}} \frac{f(\bar{x}) - f(x)}{\|\bar{x} - x\|}}$$

Slope and error bounds

Common problem: Estimate

$$\operatorname{dist}(x, [f \le r]) \quad \text{(difficult)}$$

"The residual":

$$f(x) - r \qquad \text{(easy)}$$

Desirable quality: Exists κ with

$$\operatorname{dist}(x, [f \le r]) \le \kappa(f(x) - r)$$

Eg: Hoffman's error bound for linear programming.

Slope: "Fastest instantaneous rate of decrease"

$$\boxed{|\nabla f|(\bar{x}) := \limsup_{x \to \bar{x}} \frac{f(\bar{x}) - f(x)}{\|\bar{x} - x\|}}$$

Restrict $f \colon \mathbf{R}^n \to \overline{\mathbf{R}}$ to a "slice" $f^{-1}(a, b)$.

Lemma (Error bound)

The following are equivalent.

Non-criticality:

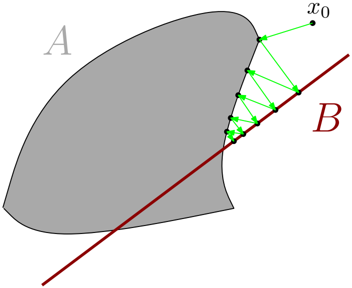
$$|\nabla f| \ge \frac{1}{\kappa}$$

Error-bound:

$$\operatorname{dist}(x, [f \le r]) \le \kappa(f(x) - r), \quad \text{when } r \in (a, f(x))$$

Coupling function:

$$\boxed{\psi(x,y) = \delta_A(x) + |x-y| + \delta_B(y).}$$



Transversality & error bounds

Coupling function:

$$\psi(x,y) = \delta_A(x) + |x - y| + \delta_B(y).$$

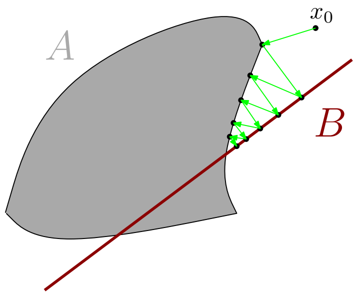
Error bound:

$$N_B(\bar{x}) \cap -N_A(\bar{x}) = \{0\}$$

$$\Downarrow$$

$$\max\{|\nabla\psi_x|(y),\ |\nabla\psi_y|(x)\} \geq \kappa$$

for $x \in A$ and $y \in B$, not in $A \cap B$.



Coupling function:

$$\boxed{\psi(x,y) = \delta_A(x) + |x-y| + \delta_B(y).}$$

Error bound:

$$N_B(\bar{x}) \cap -N_A(\bar{x}) = \{0\}$$

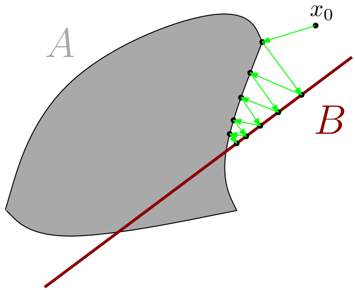
$$\Downarrow$$

$$\boxed{\max\{|\nabla\psi_x|(y),\ |\nabla\psi_y|(x)\} \geq \kappa}$$

for $x \in A$ and $y \in B$, not in $A \cap B$.

$$\Downarrow$$

Local linear convergence

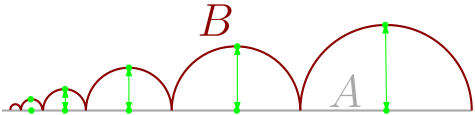


Convergence

- Transversality is necessary but not verifiable.

- Transversality is necessary but not verifiable.

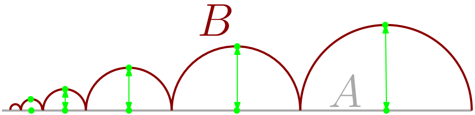
Eg:



Convergence

- Transversality is necessary but not verifiable.

Eg:

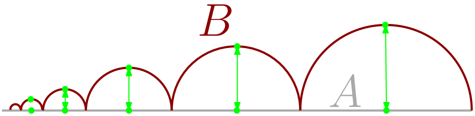


What about sublinear convergence?

Convergence

- Transversality is necessary but not verifiable.

Eg:



What about sublinear convergence?

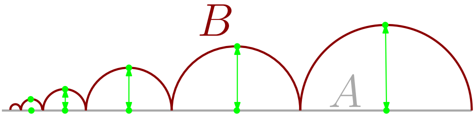
$Q \subset \mathbf{R}^n$ is semi-algebraic if is a finite union of

solution sets to finitely many polynomial inequalities.

Convergence

- Transversality is necessary but not verifiable.

Eg:



What about sublinear convergence?

$Q \subset \mathbf{R}^n$ is semi-algebraic if is a finite union of

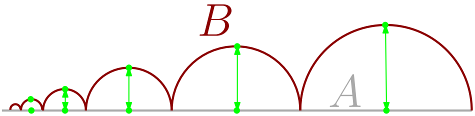
solution sets to finitely many polynomial inequalities.

Eg: semi-definite representable sets (Nesterov-Nemirovskii)

Convergence

- Transversality is necessary but not verifiable.

Eg:



What about sublinear convergence?

$Q \subset \mathbf{R}^n$ is semi-algebraic if is a finite union of

solution sets to finitely many polynomial inequalities.

Eg: semi-definite representable sets (Nesterov-Nemirovskii)

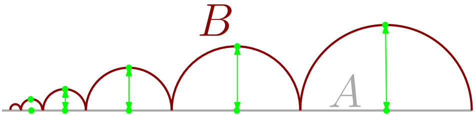
Theorem (D-Ioffe-Lewis)

A and B semi-algebraic $\implies$ alternating projections converge locally.

Convergence

- Transversality is necessary but not verifiable.

Eg:



What about sublinear convergence?

$Q \subset \mathbf{R}^n$ is semi-algebraic if is a finite union of

solution sets to finitely many polynomial inequalities.

Eg: semi-definite representable sets (Nesterov-Nemirovskii)

Theorem (D-Ioffe-Lewis)

A and B semi-algebraic $\Longrightarrow$ alternating projections converge locally.

Generic transversality (D-Ioffe-Lewis):

If A and B are semi-algebraic, then

$A + a$ and B are transverse for a.e. a

- **When alternating projections fail...**

SDP feasibility

Common problem: find a point in

$$\{X \in \mathcal{S}_+^n : \mathcal{A}(X) = d\}$$

where $\mathcal{A}$ is surjective.

SDP feasibility

Common problem: find a point in

$$\{X \in \mathcal{S}_+^n : \mathcal{A}(X) = d\}$$

where $\mathcal{A}$ is surjective.

$$\boxed{\text{Transversality} \quad \Longleftrightarrow \quad \text{Slater} \quad \Longleftrightarrow \quad d \in \operatorname{int} \mathcal{A}(\mathcal{S}_+^n)}$$

SDP feasibility

Common problem: find a point in

$$\{X \in \mathcal{S}^n_+ : \mathcal{A}(X) = d\}$$

where $\mathcal{A}$ is surjective.

$$\boxed{\text{Transversality} \quad \Longleftrightarrow \quad \text{Slater} \quad \Longleftrightarrow \quad d \in \operatorname{int} \mathcal{A}(\mathcal{S}^n_+)}$$

Slater condition often fails in practice.

SDP feasibility

Common problem: find a point in

$$\{X \in \mathcal{S}^n_+ : \mathcal{A}(X) = d\}$$

where $\mathcal{A}$ is surjective.

$$\boxed{\text{Transversality} \quad \Longleftrightarrow \quad \text{Slater} \quad \Longleftrightarrow \quad d \in \operatorname{int} \mathcal{A}(\mathcal{S}^n_+)}$$

Slater condition often fails in practice.
Eg: Structured data

$$\begin{bmatrix} 1 & 1 & ? \\ 1 & 1 & 1 \\ ? & 1 & 1 \end{bmatrix} \succeq 0$$

SDP feasibility

Common problem: find a point in

$$\{X \in \mathcal{S}_+^n : \mathcal{A}(X) = d\}$$

where $\mathcal{A}$ is surjective.

Transversality $\iff$ Slater $\iff$ $d \in \operatorname{int} \mathcal{A}(\mathcal{S}_+^n)$

Slater condition often fails in practice.
Eg: Structured data

$$\begin{bmatrix} 1 & 1 & ? \\ 1 & 1 & 1 \\ ? & 1 & 1 \end{bmatrix} \succeq 0$$

Eg: Uniqueness
The intersection $\{x \in \mathcal{S}_+^n : \mathcal{A}(x) = d\}$ is a singleton.

Euclidean distance matrices

D is a Euclidean distance matrix ($D \in \mathcal{E}^n$) if

$$\exists\ x_1, \ldots, x_n \in \mathbf{R}^r \quad \text{with} \quad D_{ij} = \|x_i - x_j\|^2$$

Euclidean distance matrices

D is a Euclidean distance matrix ($D \in \mathcal{E}^n$) if

$$\exists\ x_1, \dots, x_n \in \mathbf{R}^r \quad \text{with} \quad D_{ij} = \|x_i - x_j\|^2$$

and then

$$\mathbf{embdim}\, D = \text{ minimal } r.$$

Euclidean distance matrices

D is a Euclidean distance matrix ($D \in \mathcal{E}^n$) if

$$\exists\ x_1, \ldots, x_n \in \mathbf{R}^r \quad \text{with} \quad D_{ij} = \|x_i - x_j\|^2$$

and then

$$\operatorname{embdim} D = \text{ minimal } r.$$

Centered and hollow matrices

$$\mathcal{S}_c = \{X : Xe = 0\} \qquad \text{and} \qquad \mathcal{S}_H = \{D : \operatorname{Diag}(D) = 0\}.$$

Euclidean distance matrices

D is a Euclidean distance matrix ($D \in \mathcal{E}^n$) if

$$\exists\ x_1, \ldots, x_n \in \mathbf{R}^r \quad \text{with} \quad D_{ij} = \|x_i - x_j\|^2$$

and then

$$\operatorname{embdim} D = \text{ minimal } r.$$

Centered and hollow matrices

$$\mathcal{S}_c = \{X : Xe = 0\} \qquad \text{and} \qquad \mathcal{S}_H = \{D : \operatorname{Diag}(D) = 0\}.$$

The map

$$\mathcal{K}(X) = [X_{ii} + X_{jj} - 2X_{ij}]_{ij}$$

Euclidean distance matrices

D is a Euclidean distance matrix ($D \in \mathcal{E}^n$) if

$$\exists\ x_1, \ldots, x_n \in \mathbf{R}^r \quad \text{with} \quad D_{ij} = \|x_i - x_j\|^2$$

and then

$$\operatorname{embdim} D = \text{ minimal } r.$$

Centered and hollow matrices

$$\mathcal{S}_c = \{X : Xe = 0\} \qquad \text{and} \qquad \mathcal{S}_H = \{D : \operatorname{Diag}(D) = 0\}.$$

The map

$$\mathcal{K}(X) = [X_{ii} + X_{jj} - 2X_{ij}]_{ij}$$

restricts to an isomorphism

$$\boxed{\mathcal{K} \colon \mathcal{S}_c \to \mathcal{S}_H \quad \text{carrying} \quad \mathcal{S}_+^n \cap \mathcal{S}_c \text{ onto } \mathcal{E}^n}$$

Euclidean distance matrices

D is a Euclidean distance matrix ($D \in \mathcal{E}^n$) if

$$\exists\ x_1, \ldots, x_n \in \mathbf{R}^r \quad \text{with} \quad D_{ij} = \|x_i - x_j\|^2$$

and then

$$\text{embdim}\, D = \text{ minimal } r.$$

Centered and hollow matrices

$$\mathcal{S}_c = \{X : Xe = 0\} \qquad \text{and} \qquad \mathcal{S}_H = \{D : \text{Diag}(D) = 0\}.$$

The map

$$\mathcal{K}(X) = [X_{ii} + X_{jj} - 2X_{ij}]_{ij}$$

restricts to an isomorphism

$$\boxed{\mathcal{K} \colon \mathcal{S}_c \to \mathcal{S}_H \quad \text{carrying} \quad \mathcal{S}^n_+ \cap \mathcal{S}_c \text{ onto } \mathcal{E}^n}$$

In turn

$$\mathcal{S}^n_+ \cap \mathcal{S}_c \cong \mathcal{S}^{n-1}_+$$

Euclidean distance matrices

D is a Euclidean distance matrix ($D \in \mathcal{E}^n$) if

$$\exists\ x_1, \ldots, x_n \in \mathbf{R}^r \quad \text{with} \quad D_{ij} = \|x_i - x_j\|^2$$

and then

$$\operatorname{embdim} D = \text{ minimal } r.$$

Centered and hollow matrices

$$\mathcal{S}_c = \{X : Xe = 0\} \qquad \text{and} \qquad \mathcal{S}_H = \{D : \operatorname{Diag}(D) = 0\}.$$

The map

$$\mathcal{K}(X) = [X_{ii} + X_{jj} - 2X_{ij}]_{ij}$$

restricts to an isomorphism

$$\boxed{\mathcal{K} \colon \mathcal{S}_c \to \mathcal{S}_H \quad \text{carrying} \quad \mathcal{S}^n_+ \cap \mathcal{S}_c \text{ onto } \mathcal{E}^n}$$

In turn

$$\mathcal{S}^n_{c,+} := \mathcal{S}^n_+ \cap \mathcal{S}_c \cong \mathcal{S}^{n-1}_+$$

Euclidean distance matrices

D is a Euclidean distance matrix ($D \in \mathcal{E}^n$) if

$$\exists\; x_1, \ldots, x_n \in \mathbf{R}^r \quad \text{with} \quad D_{ij} = \|x_i - x_j\|^2$$

and then

$$\operatorname{embdim} D = \text{ minimal } r.$$

Centered and hollow matrices

$$\mathcal{S}_c = \{X : Xe = 0\} \qquad \text{and} \qquad \mathcal{S}_H = \{D : \operatorname{Diag}(D) = 0\}.$$

The map

$$\mathcal{K}(X) = [X_{ii} + X_{jj} - 2X_{ij}]_{ij}$$

restricts to an isomorphism

$$\boxed{\mathcal{K} \colon \mathcal{S}_c \to \mathcal{S}_H \quad \text{carrying} \quad \mathcal{S}_+^n \cap \mathcal{S}_c \text{ onto } \mathcal{E}^n}$$

In turn

$$\mathcal{S}_{c,+}^n := \mathcal{S}_+^n \cap \mathcal{S}_c \cong \mathcal{S}_+^{n-1}$$

Moreover

$$\boxed{\operatorname{embdim} D = \operatorname{rank} \mathcal{K}^\dagger(D)}$$

Euclidean distance completions

Fix an undirected graph

$$G = (V, E).$$

Euclidean distance completions

Fix an undirected graph

$$G = (V, E).$$

Euclidean distance completion problem:

$$\text{Complete } d \in \mathbf{R}^E \text{ to a matrix } D \in \mathcal{E}^n.$$

Euclidean distance completions

Fix an undirected graph

$$G = (V, E).$$

Euclidean distance completion problem:

$$\text{Complete } d \in \mathbf{R}^E \text{ to a matrix } D \in \mathcal{E}^n.$$

- many guises: sensor network localization, molecular conformation,...
- NP-hard on general graphs

Euclidean distance completions

Fix an undirected graph

$$G = (V, E).$$

Euclidean distance completion problem:

$$\text{Complete } d \in \mathbf{R}^E \text{ to a matrix } D \in \mathcal{E}^n.$$

- many guises: sensor network localization, molecular conformation,...
- NP-hard on general graphs

Model this as

$$\{D \in \mathcal{E} : \mathcal{P}_E(D) = d\}$$

or as an SDP

$$\{X \in \mathcal{S}^n_{c,+} : (\mathcal{P}_E \circ \mathcal{K})(X) = d\}.$$

Euclidean distance completions

Fix an undirected graph

$$G = (V, E).$$

Euclidean distance completion problem:

$$\text{Complete } d \in \mathbf{R}^E \text{ to a matrix } D \in \mathcal{E}^n.$$

- many guises: sensor network localization, molecular conformation,...
- NP-hard on general graphs

Model this as

$$\{D \in \mathcal{E} : \mathcal{P}_E(D) = d\}$$

or as an SDP

$$\{X \in \mathcal{S}^n_{c,+} : (\mathcal{P}_E \circ \mathcal{K})(X) = d\}.$$

Striking observation (Krislock-Wolkowicz '10):

For any k-clique χ

$$\text{embdim } d_\chi < k - 1 \qquad \Longrightarrow \qquad \text{Slater fails!}$$

This simple observation leads to an algorithm (KW '10)!

Facial cuts

KW-algorithm is based on intersecting faces:
Any matrix $V \in \mathcal{S}_+^n$ defines an exposed face $\mathcal{S}_+^n \cap V^\perp$ of $\mathcal{S}_+^n$.

Facial cuts

KW-algorithm is based on intersecting faces:
Any matrix $V \in \mathcal{S}_+^n$ defines an exposed face $\mathcal{S}_+^n \cap V^\perp$ of $\mathcal{S}_+^n$.

Given a set

$$\Omega = \{X \in \mathcal{S}_+^n : \mathcal{A}(X) = d\}$$

recall

$$\boxed{\text{Slater} \qquad \iff \qquad d \in \operatorname{int} \mathcal{A}(\mathcal{S}_+^n)}$$

KW-algorithm is based on intersecting faces:
Any matrix $V \in \mathcal{S}_+^n$ defines an exposed face $\mathcal{S}_+^n \cap V^\perp$ of $\mathcal{S}_+^n$.

Given a set

$$\Omega = \{X \in \mathcal{S}_+^n : \mathcal{A}(X) = d\}$$

recall

$$\boxed{\text{Slater} \qquad \iff \qquad d \in \operatorname{int} \mathcal{A}(\mathcal{S}_+^n)}$$

Simple observation:

$$v \ \text{ exposes } \ \text{face}(d, \mathcal{A}(\mathcal{S}_+^n)) \quad \Longrightarrow \quad \mathcal{A}^* v \ \text{ exposes } \ \text{face}(\Omega, \mathcal{S}_+^n).$$

Faces of Euclidean distance matrices

Cliques and faces (KW '10, D-Pataki-Wolkowicz '14):
For any k-clique χ define

$$\Omega_\chi = \{X \in \mathcal{S}^n_{c,+} : (\mathcal{P}_\chi \circ \mathcal{K})(X) = d_\chi\}.$$

Faces of Euclidean distance matrices

Cliques and faces (KW '10, D-Pataki-Wolkowicz '14):
For any k-clique χ define

$$\Omega_\chi = \{X \in \mathcal{S}^n_{c,+} : (\mathcal{P}_\chi \circ \mathcal{K})(X) = d_\chi\}.$$

Then assuming $\chi = \{1, \ldots, k\}$, if

$$V \qquad \text{exposes} \qquad \text{face}(\mathcal{K}^\dagger(d_\chi), \mathcal{S}^k_{c,+})$$

then

$$\begin{bmatrix} V & 0 \\ 0 & 0 \end{bmatrix} \qquad \text{exposes} \qquad \text{face}(\Omega_\chi, \mathcal{S}^n_{c,+}).$$

Faces of Euclidean distance matrices

Cliques and faces (KW '10, D-Pataki-Wolkowicz '14):
For any k-clique χ define

$$\Omega_\chi = \{X \in \mathcal{S}^n_{c,+} : (\mathcal{P}_\chi \circ \mathcal{K})(X) = d_\chi\}.$$

Then assuming $\chi = \{1, \ldots, k\}$, if

$$V \qquad \text{exposes} \qquad \text{face}(\mathcal{K}^\dagger(d_\chi), \mathcal{S}^k_{c,+})$$

then

$$\begin{bmatrix} V & 0 \\ 0 & 0 \end{bmatrix} \qquad \text{exposes} \qquad \text{face}(\Omega_\chi, \mathcal{S}^n_{c,+}).$$

KW algorithm:

- Explore cliques and intersect faces one-by-one.
- Usually, can grow cliques.
- Purely combinatorial $\Longrightarrow$ can handle $10^6 \times 10^6$-matrices.
- If graph is chordal, then finds the true minimal face (D-Pataki-Wolkowicz '14).

Shortcoming of KW algorithm

Factors not considered

- Noise: $d \in \mathbf{R}^E$ is "noisy".
- Embedding dimension: search for $X \in \mathcal{S}^n_{c,+}$ of a priori known rank r.

Shortcoming of KW algorithm

Factors not considered

- Noise: $d \in \mathbf{R}^E$ is "noisy".
- Embedding dimension: search for $X \in \mathcal{S}^n_{c,+}$ of a priori known rank r.

Least-squares formulation:

$$\begin{aligned} \min\ & \|(\mathcal{P}_E \circ \mathcal{K})(X) - d\|^2 \\ \text{s.t.}\ & \operatorname{rank} X = r \\ & X \in \mathcal{S}^n_{c,+} \end{aligned}$$

This is a large-scale, noisy, nonconvex problem.

Shortcoming of KW algorithm

Factors not considered

- Noise: $d \in \mathbf{R}^E$ is "noisy".
- Embedding dimension: search for $X \in \mathcal{S}^n_{c,+}$ of a priori known rank r.

Least-squares formulation:

$$\begin{aligned} \min \ & \|(\mathcal{P}_E \circ \mathcal{K})(X) - d\|^2 \\ \text{s.t.} \ & \operatorname{rank} X = r \\ & X \in \mathcal{S}^n_{c,+} \end{aligned}$$

This is a large-scale, noisy, nonconvex problem.

- "Relax rank, solve, project" works poorly.
- Too big for SDP.
- Local methods output irrelevant local minimizers.

Robust low-rank facial cuts

Goal:

> Use both convexity & low-rank to solve the problem.

$$\begin{aligned} \min \quad & \|(\mathcal{P}_E \circ \mathcal{K})(X) - d\|^2 \\ \text{s.t.} \quad & \operatorname{rank} X = r \\ & X \in \mathcal{S}^n_{c,+} \end{aligned}$$

Robust low-rank facial cuts

Goal:

Use both convexity & low-rank to solve the problem.

$$\begin{aligned} \min \quad & \|(\mathcal{P}_E \circ \mathcal{K})(X) - d\|^2 \\ \text{s.t.} \quad & \operatorname{rank} X = r \\ & X \in \mathcal{S}^n_{c,+} \end{aligned}$$

First idea: adapt the KW algorithm

- Use projections

$$\mathcal{P}_{\{Z \succeq 0:\ \operatorname{rank} Z \leq r\}}\Big(\mathcal{K}^\dagger\left(d_{\chi^i}\right)\Big).$$

- Compounds error while intersecting faces.

Robust low-rank facial cuts

Goal:

Use both convexity & low-rank to solve the problem.

$$\begin{aligned} \min \; & \|(\mathcal{P}_E \circ \mathcal{K})(X) - d\|^2 \\ \text{s.t.} \; & \operatorname{rank} X = r \\ & X \in \mathcal{S}^n_{c,+} \end{aligned}$$

First idea: adapt the KW algorithm

- Use projections

$$\mathcal{P}_{\{Z \succeq 0:\ \operatorname{rank} Z \leq r\}}\Big(\mathcal{K}^\dagger\left(d_{\chi^i}\right)\Big).$$

- Compounds error while intersecting faces.

Key observation: for a convex cone C

$$\left.\begin{array}{l} v_1 \text{ exposes a face } F_1 \text{ of } C \\ v_2 \text{ exposes a face } F_2 \text{ of } C \end{array}\right\} \quad \Longrightarrow \quad v_1 + v_2 \text{ exposes } F_1 \cap F_2$$

Robust low-rank facial cuts

$$\begin{aligned} \min \quad & \|(\mathcal{P}_{\mathcal{E}} \circ \mathcal{K})(X) - d\|^2 \\ \text{s.t.} \quad & \operatorname{rank} X = r \\ & X \in \mathcal{S}^n_{c,+} \end{aligned}$$

Algorithmic framework (Cheung-D-Krislock-Wolkowicz'14):

1. Fix a set of cliques χ^i and weights θ_i.
2. Form
$$Z_i \in \mathcal{P}_{\{Z \succeq 0:\ \operatorname{rank} Z \leq r\}}\Big(\mathcal{K}^\dagger(d_{\chi^i})\Big).$$
3. Choose W_i exposing face$(Z_i, \mathcal{S}^{|\chi^i|}_{c,+})$ and form the "aggregate"
$$\mathcal{M} = \sum_i \theta_i \cdot \mathcal{P}^*_{\chi^i}(W_i)$$
4. Set $\mathcal{N} = \mathcal{P}_{\{Z \succeq 0:\ \operatorname{rank} Z = n-r\}}(\mathcal{M})$ and solve Least Squares on $\mathcal{S}^n_{c,+} \cap \mathcal{N}^\perp$.

Robust low-rank facial cuts

$$\begin{aligned} \min \quad & \|(\mathcal{P}_{\mathcal{E}} \circ \mathcal{K})(X) - d\|^2 \\ \text{s.t.} \quad & \operatorname{rank} X = r \\ & X \in \mathcal{S}^n_{c,+} \end{aligned}$$

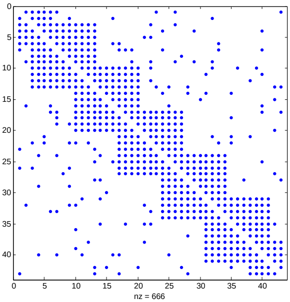
Algorithmic framework (Cheung-D-Krislock-Wolkowicz'14):

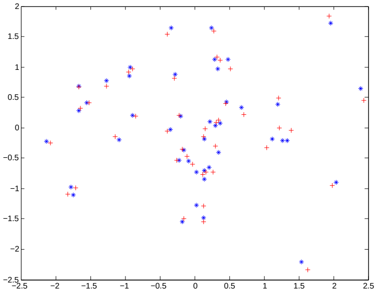
1. Fix a set of cliques χ^i and weights θ_i.
2. Form
$$Z_i \in \mathcal{P}_{\{Z \succeq 0:\ \operatorname{rank} Z \le r\}}\Big(\mathcal{K}^\dagger(d_{\chi^i})\Big).$$
3. Choose W_i exposing face$(Z_i, \mathcal{S}^{|\chi^i|}_{c,+})$ and form the "aggregate"
$$\mathcal{M} = \sum_i \theta_i \cdot \mathcal{P}^*_{\chi^i}(W_i)$$
4. Set $\mathcal{N} = \mathcal{P}_{\{Z \succeq 0:\ \operatorname{rank} Z = n-r\}}(\mathcal{M})$ and solve Least Squares on $\mathcal{S}^n_{c,+} \cap \mathcal{N}^\perp$.

Eg:
$$\theta_i = \Big(\sum_j \|Z_j - \mathcal{K}^\dagger(d_{\chi^j})\|^2\Big) - \|Z_i - \mathcal{K}^\dagger(d_{\chi^i})\|^2$$

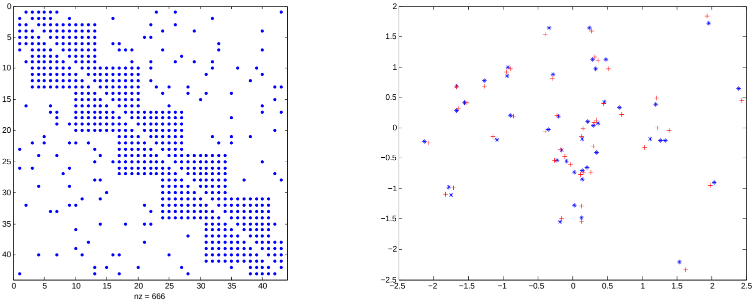
Robust low-rank facial cuts

5% noise ($n = 45$, $r = 2$):





5% noise ($n = 45$, $r = 2$):

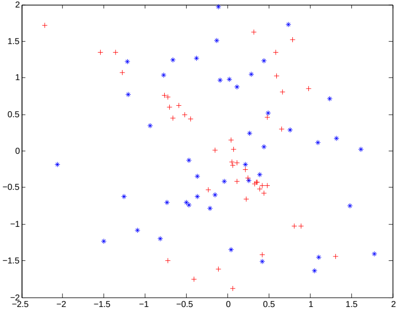


Eigenvalues of $\mathcal{M}$ (multiples of 10^{-6}):

$$0 \quad 1 \quad 200 \quad 300 \quad 350 \quad \ldots$$

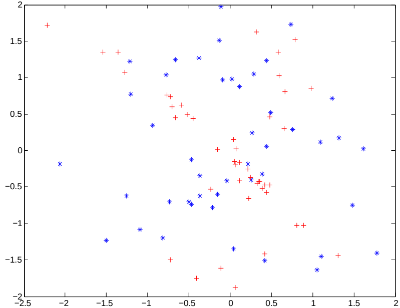
Robust low-rank facial cuts

20% noise ($n = 45,\ r = 2$):



Robust low-rank facial cuts

20% noise ($n = 45$, $r = 2$):

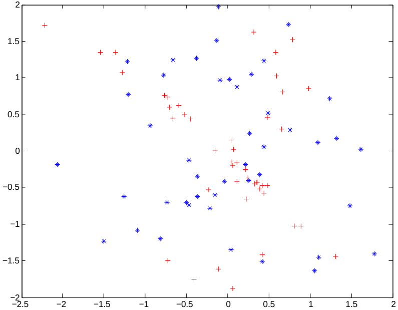


Eigenvalues of $\mathcal{M}$ (multiples of 10^{-5}):

$$0 \quad 1 \quad 3 \quad 6 \quad 40 \quad 60 \quad 80 \quad \ldots$$

Robust low-rank facial cuts

20% noise ($n = 45,\ r = 2$):



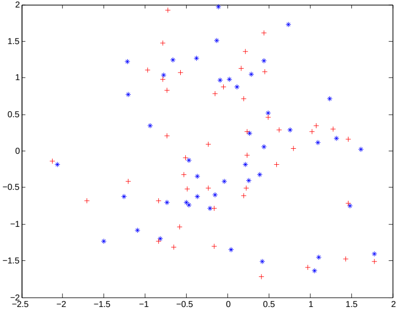
Eigenvalues of $\mathcal{M}$ (multiples of 10^{-5}):

$$0 \qquad 1 \qquad 3 \qquad 6 \qquad 40 \qquad 60 \qquad 80 \qquad \ldots$$

Idea: use the lowest 4 eigenvalues and then project to rank 2.

Robust low-rank facial cuts

20% noise ($n = 45$, $r = 2$):



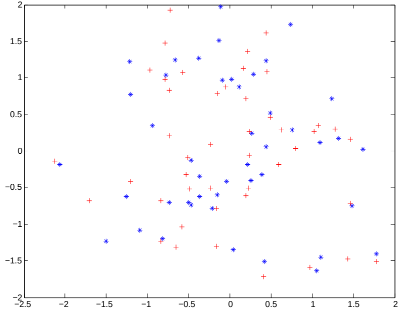
Eigenvalues of $\mathcal{M}$ (multiples of 10^{-5}):

$$0 \qquad 1 \qquad 3 \qquad 6 \qquad 40 \qquad 60 \qquad 80 \qquad \ldots$$

Idea: use the lowest 4 eigenvalues and then project to rank 2.

Robust low-rank facial cuts

20% noise ($n = 45$, $r = 2$):



Eigenvalues of $\mathcal{M}$ (multiples of 10^{-5}):

$$0 \quad 1 \quad 3 \quad 6 \quad 40 \quad 60 \quad 80 \quad \ldots$$

Idea: use the lowest 4 eigenvalues and then project to rank 2.
Better idea: initialize gradient descent on the rank 2 manifold.

Conclusion & Open questions

- Alternating projections: simple, powerful, versatile (when it works).
- Alternating projections can fail for structured problems.
- This can be an advantage.
  - Illustration: noisy, low-rank Euclidean distance completions.

Main open question:
What is the theoretical basis for the noisy EDM algorithm?

Thank you.