Random projections in mathematical programming

Leo Liberti, CNRS LIX Ecole Polytechnique

with: P.L. Poirion, Ky Vu , B. Manca, C. D'Ambrosio, A. Oustry

Danniversary @ MIP22

@ @ &

Section 1

Random projections

The gist of random projections

- Let A be a $m \times n$ data matrix (n col. vectors $\in \mathbb{R}^m$, $m \gg 1$)
- $T = (T_{ij})$ with density σ, with $T_{ij} \sim \mathsf{N}(0, \frac{1}{\sqrt{k\sigma}})$ and $\boxed{k = O(\ln n)}$

columnwise: $$\underbrace{\begin{pmatrix} \cdots & \cdots \\ \cdots & \cdots \\ \cdots & \cdots \end{pmatrix}}_{T \text{ is } k \times m} \underbrace{\begin{pmatrix} \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix}}_{A \text{ is } m \times n} = \underbrace{\begin{pmatrix} \vdots & \vdots & \vdots & \vdots \end{pmatrix}}_{TA \text{ is } k \times n}$$

row-wise: $$\forall i \le m \qquad \sum_{j \le n} T_{ij} \mathsf{row}_j(A) = \mathsf{row}_i(TA)$$

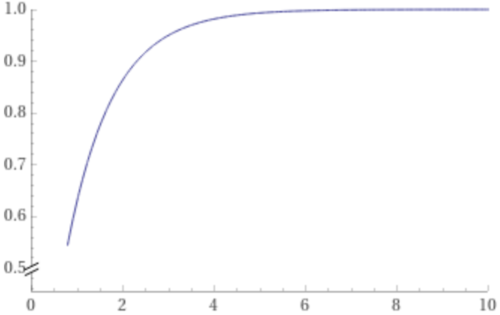
- We have $\forall i, j \le n \quad \|A_i - A_j\|_2 \approx \|TA_i - TA_j\|_2$ "WAHP"
- Notation: $k \times m$ *random projection* matrix $T \sim \mathsf{RP}(k, m)$

WAHP

"WAHP" = "with arbitrarily high probability"

the probability of E_k (depending on some parameter k)
approaches 1 *"exponentially fast"* as k increases

$$\mathbf{P}(E_k) \geq 1 - O(e^{-k})$$



Johnson-Lindenstrauss Lemma

Existence of RPs

Thm.

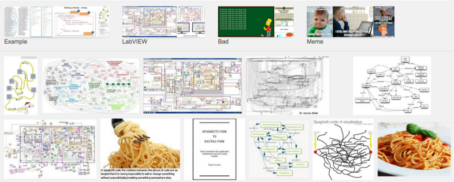
Given $\epsilon > 0$, $A \subseteq \mathbb{R}^m$ with $|A| = n$, $\exists k = O(\frac{1}{\epsilon^2} \ln n)$ and $T \sim \mathsf{RP}(k, m)$ s.t.
$\forall i < j \leq n \quad (1-\epsilon)\|A_i - A_j\|_2 \;\leq\; \|TA_i - TA_j\|_2 \;\leq\; (1+\epsilon)\|A_i - A_j\|_2$
WAHP

Equivalently: ϵ-distortion $(1-\epsilon) \;\leq\; \frac{\|TA_i - TA_j\|_2}{\|A_i - A_j\|_2} \;\leq\; (1+\epsilon)$

Remarks

- JLL Proof implies:
  (i) $\forall i < j \leq n \quad \mathbb{E}_T(\|TA_i - TA_j\|) = \|A_i - A_j\|$
  (ii) large discrepancy from the mean is unlikely
- Empirically, sample T very few times (e.g. once will do!)
  error will only significantly impact few pairs
- Surprising fact: k is independent of number of dimensions m
- JLL $\Rightarrow \exists O(e^k)$ almost orthogonal vectors in $\mathbb{R}^k$
  Pf.: approx. congruences almost preserve angles, apply T to std basis of $\mathbb{R}^m$

Example: clustering data



- k-means without random projections

```
VHimg = Map[Flatten[ImageData[#]] &, Himg];
```

Out[28]=

```
VHcl = Timing[ClusteringComponents[VHimg, 3, 1]]
Out[29]= {0.405908, {1, 2, 2, 2, 2, 2, 3, 2, 2, 2, 3}}
```

- k-means with random projections

```
VKimg = JohnsonLindenstrauss[VHimg, 0.1];
VKcl = Timing[ClusteringComponents[VKimg, 3, 1]]
Out[34]= {0.002232, {1, 2, 2, 2, 2, 2, 3, 2, 2, 2, 3}}
```

Projecting formulations

- The set-up
  - Given MP formulation $F(p,x)$ and a RP T consider $F(Tp,x)$
  - Aim at proving $\mathsf{val}(F(p,x)) \approx \mathsf{val}(F(Tp,x))$ WAHP
- Difficulties:
  - JLL only applies to finite sets, decision vars represent infinite sets
  - Need to prove approximate congruence ⇒ approx. feasibility/optimality
  - Often need a solution retrieval method

- For now, $\mathscr{C}$ is…
  - LP, SOCP, SDP [Vu et al. MOR 2018; L. et al. LAA 2021]
  - QP, QP over a ball constraint [Vu et al. IPCO 2019; D'Ambrosio et al. MPB 2020]
  - Min Sum-of-squares Clustering MINLP [L. & Manca JOGO 2021]
- In preparation:
  - QCQP
  - ILP

Section 2

Projected formulations: survey

Linear Programming: formulations

- Standard form LP

$$\left.\begin{array}{rrcl} \min\limits_{x\in\mathbb{R}^n} & c\cdot x & & \\ & Ax & = & b \\ & x & \geq & 0 \end{array}\right\} (\mathsf{LP})$$

- Projected standard form LP *(projecting the constraints)*

$$\left.\begin{array}{rrcl} \min\limits_{x\in\mathbb{R}^n} & c\cdot x & & \\ & TAx & = & Tb \\ & x & \geq & 0 \end{array}\right\} (T\mathsf{LP})$$

- LP has $k = O(\epsilon^{-2}\ln n) \ll m$ constraints, can be solved faster
  reducing the constraint number from m to k
- Constraint projection: $\forall h \leq k \; (TAx - Tb)_h = \sum\limits_{i\leq m} T_{hi}(A_i\cdot x - b_i)$

Linear Programming: theory

- Approximate feasibility:
  - we prove LP feasible $\Leftrightarrow$ TLP feasible WAHP
  - (A, b) feasible $\Rightarrow$ (TA, Tb) feasible by linearity of T
  - converse is harder and WAHP (e.g. by projected separating hyperplane)
- Approximate optimality:
  - let: x^* opt. of LP, $\bar{x}$ opt. of TLP
  - boundedness assumption: $\mathbf{1} \cdot x^* \le \theta$
  - "sandwich theorem":
    $\exists\delta$ s.t. $\mathsf{val}(\mathsf{LP}) - \delta \le \mathsf{val}(T\mathsf{LP}) \le \mathsf{val}(\mathsf{LP})$ WAHP
  - δ depends on ϵ, θ and a lot of other quantities e.g. opt. dual soln y^* (!!)
- Solution retrieval: $\tilde{x} = \bar{x} + A^\top (AA^\top)^{-1}(b - A\bar{x})$
  projection of $\bar{x}$ on subspace $Ax = b$
  $\tilde{x}$ satisfies $Ax = b$ by construction, has small error $x \not\geq 0$ WAHP
- See [Vu et al. MOR 2018]

Semidefinite Programming: formulations

- Standard form SDP

$$\left.\begin{array}{rrcl} \min_{X \in \mathbb{S}^n} & \langle C, X \rangle & & \\ \forall i \le m & \langle A_i, X \rangle & = & b_i \\ & X & \succeq & 0 \end{array}\right\} (\mathsf{SDP})$$

- Denote $A \odot X = (\langle A_i, X \rangle \mid i \le m)$ where $A = (\mathsf{vec}(A_i) \mid i \le m)$
- SDP equivalent to:

$$\left.\begin{array}{rrcl} \min_{X \in \mathbb{S}^n} & \langle C, X \rangle & & \\ & A \odot X & = & b \\ & X & \succeq & 0 \end{array}\right\}$$

- Projected SDP (*projecting the constraints*)

$$\left.\begin{array}{rrcl} \min_{X \in \mathbb{S}^n} & \langle C, X \rangle & & \\ & TA \odot X & = & Tb \\ & X & \succeq & 0 \end{array}\right\} (T\mathsf{SDP})$$

Semidefinite programming: theory

- Approximate feasibility:
  similar to projecting constraints in LP, but in a Jordan Algebra

- Approximate optimality:
  - let: X^* opt. of SDP, $\bar{X}$ opt. of TSDP
  - based on boundedness assumption $\langle \mathbf{1}, X^* \rangle \leq \theta$
  - "sandwich theorem":
    $\mathsf{val}(\mathsf{SDP}) - E \leq \mathsf{val}(T\mathsf{SDP}) \leq \mathsf{val}(\mathsf{SDP})$ WAHP
  - E depends on A, b, ϵ, θ and soln of SDP dual (!!)

- Solution retrieval: $\tilde{X} = \bar{X} + A^\top (AA^\top)^{-1} \odot (b - A \odot \bar{X})$
  projection of $\bar{X}$ on subspace $A \odot X = b$
  $\tilde{X}$ satisfies $A \odot X = b$ by construction, has small error $\tilde{X} \not\succeq 0$ WAHP

- Proof techniques different from LP case
  cones modelled with formally real Jordan algebras

- See [L. et al. LAA 2021]

Projecting variables: an idea from duality

- Consider the dual LP in canonical form

$$\left.\begin{array}{rrcl} \max\limits_{y\in\mathbb{R}^m} & y\cdot b & & \\ & yA & \leq & c \end{array}\right\} (\mathsf{dLP})$$

- Consider the dual of $T\mathsf{LP} \equiv \min\{c\cdot x \mid TAx = Tb \wedge x \geq 0\}$

$$\left.\begin{array}{rrcl} \max\limits_{u\in\mathbb{R}^k} & u\cdot Tb & & \\ & uTA & \leq & c \end{array}\right\} (T\mathsf{dLP})$$

- LP has $k = O(\epsilon^{-2}\ln n) \ll m$ vars, can be solved faster
  reducing the number of variables from n to k
  projecting the variables instead of the constraints

Projecting variables: constructing the formulation

- ~~Approximate~~ Exact feasibility:
  $c \geq u(TA) = (uT)A$, hence uT feasible in dLP

- Solution retrieval: let $\quad y = uT \quad (\star)$

- What happens in the original variables:

  - $TT^\top \approx I_k \quad$ WAHP $\quad$ [Zhang et al. COLT 2013]
  - by $(\star)$ we have $\quad yT^\top = uTT^\top \approx uI_k = u \quad$ WAHP
  - $\Rightarrow \max\{y \cdot T^\top Tb \mid yT^\top TA \leq c\} \approx \max\{u \cdot Tb \mid uTA \leq c\}$
    $= \mathsf{val}(T\mathsf{dLP}) \quad$ WAHP
  - Gives us an idea for projecting variables

    (i) $u = Ty$ $\qquad$ (ii) $A \to TA$ and $b \to Tb$

Canonical form LP: formulations

- Original LP (canonical form)

$$\left.\begin{array}{lrcl} \max\limits_{x\in\mathbb{R}^n} & c\cdot x & & \\ & Ax & \leq & b \end{array}\right\}(\mathsf{cLP})$$

- Projecting the variables

$$\left.\begin{array}{lrcl} \max\limits_{x\in\mathbb{R}^n} & Tc\cdot Tx & & \\ & AT^\top(Tx) & \leq & b \end{array}\right\}$$

- Projected LP (canonical form)

$$\left.\begin{array}{lrcl} \max\limits_{u\in\mathbb{R}^k} & \bar{c}\cdot u & & \\ & \bar{A}u & \leq & b \end{array}\right\}(T\mathsf{cLP})$$

where $\bar{c} = Tc$, $\bar{A} = AT^\top$ and $u = Tx$

Canonical form LP: theory

- Feasibility: uT feasible in original problem
  Solution retrieval: $x = T^\top u$ (as shown above)
- Approximate optimality:
  - "sandwich theorem": for radius r of ball containing $Ax \leq b$
    $\mathsf{val}(T\mathsf{cLP}) \leq \mathsf{val}(\mathsf{cLP}) \leq \mu_\epsilon^r \mathsf{val}(T\mathsf{cLP}) + r\epsilon\|c\|_2$ WAHP
    for a certain $\mu_\epsilon^r < 1$ depending on many other quantities
- Proved using additive distortions:
  - *Notation:* $x \in y \pm \alpha \;\Leftrightarrow\; x \in [y - \alpha, y + \alpha]$
  - Given $\epsilon \in (0,1)$ $\exists$ constant $\mathcal{C}$ and $T \sim \mathsf{RP}(k,n)$ s.t.
    1. $\forall x, y \in \mathbb{R}^n \quad (Tx)^\top Ty \in x^\top y \pm \epsilon \|x\|_2 \, \|y\|_2$ WAHP
    2. $\forall x \in \mathbb{R}^n \quad AT^\top Tx \in Ax \pm \epsilon \|x\| \mathbf{1}$ WAHP

Quadratic Programming: formulations

- Original QP

$$\left.\begin{array}{rrcl} \max\limits_{x\in\mathbb{R}^n} & x^\top Qx & + & c\cdot x \\ & Ax & \leq & b \end{array}\right\}(\mathsf{QP})$$

- Projected reformulation (projecting variables)

$$\left.\begin{array}{rrcl} \max\limits_{x\in\mathbb{R}^n} & x^\top T^\top TQT^\top Tx & + & Tc\cdot Tx \\ & AT^\top(Tx) & \leq & b \end{array}\right\}$$

- Projected QP

$$\left.\begin{array}{rrcl} \max\limits_{u\in\mathbb{R}^k} & u^\top \bar{Q}u & + & \bar{c}\cdot u \\ & \bar{A}u & \leq & b \end{array}\right\}(T\mathsf{QP})$$

where $\bar{Q} = TQT^\top$, $\bar{c} = Tc$, $\bar{A} = AT^\top$ and $u = Tx$

Quadratic Programming: theory

- Feasibility: uT feasible in original problem
  Solution retrieval: $x = T^\top u$ (as for cLP)
- Approximate optimality
  - "sandwich theorem": for radius r of ball containing $Ax \le b$,
    $\mathsf{val}(T\mathsf{QP}) \le \mathsf{val}(\mathsf{QP}) \le \mu_\epsilon^r \mathsf{val}(T\mathsf{QP})) + r\epsilon(3r\|Q\|_F + \|c\|_2)$ WAHP
    for a certain $\mu_\epsilon^r < 1$ depending on many other quantities
- Proved using additive distortions:
  - *Notation:* $x \in y \pm \alpha \;\Leftrightarrow\; x \;\in\; [y-\alpha, y+\alpha]$
  - Given $\epsilon \in (0,1)$ $\exists$ constant $\mathcal{C}$ and $T \sim \mathsf{RP}(k,n)$ s.t.
    1. scalar products as for cLP
    2. linear forms as for cLP
    3. $\forall x, y \in \mathbb{R}^n \quad x^\top T^\top T Q T^\top T y \;\in\; x^\top Q y \pm 3\epsilon\|x\|\,\|y\|\,\|Q\|_F$
       WAHP
- See [D'Ambrosio et al. MPB 2020]

Section 3

Projected formulations: new directions

Improved solution retrieval for constraint projection

- RPs on conic programs (LP, SDP) project $Ax = b$ to $TAx = Tb$
- Solution retrieval: $\tilde{x} = \bar{x} + A^\top (AA^\top)^{-1}(b - A\bar{x})$
  geometric interpretation: project $\bar{x}$ onto $Ax = b$ subspace
- Issue: small negativity error $x \not\geq 0$, $X \not\succeq 0$ WAHP
- *Use alternating projection method (APM):*

  1. $\tilde{x} = \bar{x}$
  2. project $\tilde{x}$ onto $Ax = b$
  3. project result onto $x \geq 0$
  4. if errors of updated $\tilde{x}$ small w.r.t. $Ax = b$, $x \geq 0$, stop
  5. repeat a given number of times
  6. return $\tilde{x}$ as retrieved solution

  (respectively $X \succeq 0$ if SDP) [L. et al., AIRO-ODS22]

- Convergence properties well-studied for projections on convex sets
- *Projection on $x \geq 0$:* $\tilde{x} = (\max(x_j, 0) \mid j \leq n)$
- *Projection on $X \succeq 0$:* for eigenval vector λ and eigenvect matrix P of X,

$$\tilde{X} = P\,\mathsf{diag}(\max(\lambda_j, 0) \mid j \leq n) P^\top$$

Quadratically constrained sets

- Recall $A \odot X = (\langle A_i, X\rangle \mid i \le m)$ where $A = (\mathsf{vec}(A_i) \mid i \le m)$
- Original pure-feasibility equality QCP

$$\mathsf{eQCP} \equiv \{(x, X) \mid A \odot X = b \wedge X = xx^\top\}$$

- Projected pure-feasibility equality QCP

$$T\mathsf{eQCP} \equiv \{(x, X) \mid TA \odot X = Tb \wedge X = xx^\top\}$$

- Thm.: Let QCP be feasible and $(\bar{x}, \bar{X})$ soln of TQCP
  Then $\exists$ const $\mathcal{C}$ s.t. $\forall u > 0$

$$\|A \odot \bar{X} - b\|_2 \le (\mathcal{C} \max_{j \le n^2} \|A^j\|_2 \sqrt{\ln n} + 2u\|A\|_2)\theta^2/\sqrt{k}$$

with probability $\ge 1 - 2e^{-u^2}$

A^j is the j-th col. of A and θ s.t. $\|\bar{x}\|_1 \le \theta$

[L. et al. DMD22]

A QCQP class: formulations

- Original QCQP

$$\left.\begin{array}{rrcl} \min\limits_{x\in\mathbb{R}^n} & x^\top Q^0 x & + & c\cdot x \\ \forall i\le m & x^\top Q^i x & = & a_i \\ & Ax & = & b \\ & x & \ge & 0 \end{array}\right\} \text{(QCQP)}$$

- Compact notation and linearization

$$\left.\begin{array}{llcl} \min\limits_{\substack{x\in\mathbb{R}^n \\ X\in\mathbb{S}^n}} & \langle Q^0, X\rangle & + & c\cdot x \\ & Q\odot X = a & \wedge & Ax = b \\ & X = xx^\top & \wedge & x\ge 0 \end{array}\right\}$$

- Projected QCQP

$$\left.\begin{array}{llcl} \min\limits_{\substack{x\in\mathbb{R}^n \\ X\in\mathbb{S}^n}} & \langle Q^0, X\rangle & + & c\cdot x \\ & SQ\odot X = Sa & \wedge & TAx = Tb \\ & X = xx^\top & \wedge & x\ge 0 \end{array}\right\} ((S,T)\text{QCQP})$$

A QCQPs class: theory

- Theory: union bound lemma on previous probabilistic results on SDP and QCP — we are working on this
- Solution retrieval:
  - on convex problems: APM
  - on nonconvex problems:
    use solution $(\bar{x}, \bar{X})$ of (S, T)QCQP as a starting point for a local solver deployed on QCQP

    denoted $(\tilde{x}, \tilde{X}) = \text{locSlv}(\mathsf{QCQP}, (\bar{x}, \bar{X}))$

A QCQPs class: projecting constraints *and* variables

- From RP4QP theory, can bound the projected value error on $x^\top Q x$
  $\Rightarrow$ can replace Q with $\bar{Q}$, c with $\bar{c}$ and x with u
- Obtain projected QCQP w.r.t. $R \sim \mathsf{RP}(r, n)$

$$\left.\begin{array}{rrcl} \min\limits_{u \in \mathbb{R}^r} & u^\top \bar{Q}^0 u & + & \bar{c} \cdot u \\ \forall e \in E & u^\top \bar{Q}^e u & = & \alpha_e \\ & \bar{A} u & = & a \end{array}\right\}$$

  where $\bar{Q}^e = R Q^e R^\top$ for $e \in E \cup \{0\}$, $\bar{A} = A R^\top$, $\bar{c} = c R^\top$, $u = Rx$
- Now apply $(S, T)\mathsf{QCQP}$ projection, get

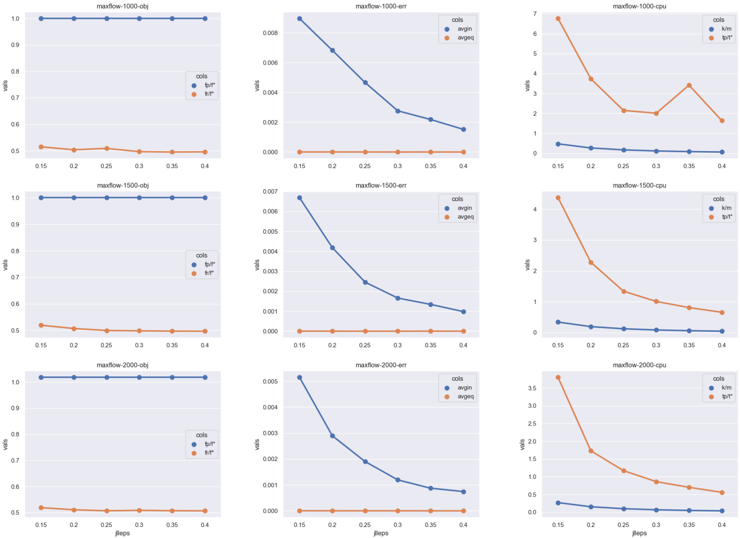
$$\left.\begin{array}{rrcl} \min\limits_{\substack{u \in \mathbb{R}^r \\ U \in \mathbb{R}^{r \times r}}} & \langle \bar{Q}^0, U \rangle & + & \bar{c} \cdot u \\ & S \bar{Q}^E \odot U & = & S \alpha^E \\ & T \bar{A} u & = & T a \\ & U & = & u u^\top \end{array}\right\} \quad ((R, S, T)\mathsf{QCQP})$$

- Prevent linear infeas: choose $S \sim \mathsf{RP}(s, |E|)$, $T \sim \mathsf{RP}(t, m)$ s.t. $s, t < r$
- Retrieval: *vars:* $X = R^\top U R$, *constrs:* (x, X) start pt for locSlv(QCQP)

Section 4

Some computational results

LP: Maximum flow



$\bar{f}/f^*$ vs $\tilde{f}/f^*$

ineqerr vs eqerr

k/m vs $\tilde{t}/t^*$

[L. et al. SEA22]

normal retrieval

Semidefinite programming: ACOPF instances

ACOPF: decide voltage, power s.t. Ohm's and Kirchhoff's laws + technical constr

ACOPF SDP formulation:

$$\min\{F = \langle C, X\rangle \mid A \odot X = b \wedge L \le X \le U \wedge X \succeq 0\}$$

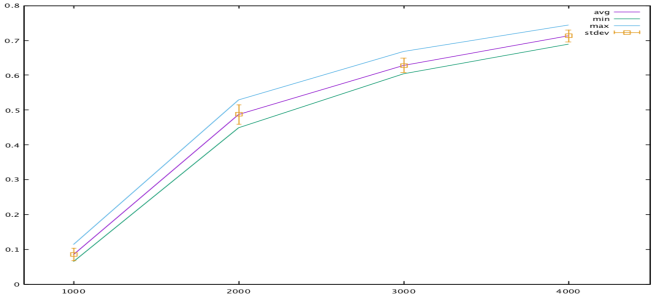
Instance						Objective		Feasibility			CPU
name	σ	m	k	n	d	$\bar{F}/F^*$	$\tilde{F}/F^*$	*infeas*	`rng`	$\lambda_{\min}$	$\tilde{t}/t^*$
`c57`	0.00366	3363	3	114	6555	0.0002	0.9152	0.000	0.007	0.092	0.74
`c73`	0.00205	5475	5	146	10731	0.0028	0.8228	0.000	0.018	0.094	0.58
`c89`	0.00377	8099	8	178	15931	0.0001	0.0147	0.000	0.281	0.022	0.14
`c118`	0.00136	14160	14	236	27966	0.0006	0.8456	0.000	0.010	0.092	0.35
`c162`	0.00072	26568	27	324	52650	0.0015	0.8868	0.000	0.012	0.099	0.32
`c179`	0.00059	32399	32	358	64261	0.0010	0.0632	0.000	0.261	0.017	0.28

[L. et al. AIRO-ODS22] improved retrieval

QP: Portfolio with short sells and investiment areas

- Relative obj fun ratio
  $\mathsf{r} = \frac{|f^* - \bar{f}|}{\max(|f^*|,|\bar{f}|)}$
- CPU ratio $\mathsf{c} = \bar{t}/t^*$
- Sizes: $1000 \leq n \leq 4000$
  $2300 \leq m \leq 8900$

	r	c
mean	0.478	0.83
stdev	0.242	0.42
min	0.065	0.30
max	0.744	1.66



bad news: error increases with size (probably $O(\sqrt{n})$)

QP: Usefulness

- Finding feasible solutions of huge QPs where solver fails

  cuberot, $n = 3000, m = 7000$, dens $= 0.9$

  cuberot, $n = 4000, m = 9000$, dens $= 0.9$

- Solver speed-up with starting point given by TQP

Instance set	n	m	$\hat{t}/t^*$
cuberot	4000	8100	0.63
cuberot	4000	8100	0.78
cuberot	4000	8100	0.78
cuberot	4000	9000	0.80
cuberot	4000	9000	0.77
cuberot	4000	9000	0.56
pairs	4000	2000	0.68
pairs	4000	2000	0.63
pairs	4000	2000	0.71
random	4000	1000	0.70
random	4000	1000	0.55
random	4000	1000	0.72

$$\hat{t} = \bar{t} + \mathsf{cpu}(\mathrm{locSlv}(\mathsf{QP}, \bar{x}))$$

DGP: Projecting both variables and constraints

$$\min\{\|y^+ + y^-\|_1 \mid \forall \{i,j\} \in E \quad \|x_i - x_j\|_2 = d_{ij}^2 + y_{ij}^+ - y_{ij}^-\} \quad \text{(slack)}$$

$$\min\{\|Y^+ + Y^-\|_1 \mid \forall \{i,j\} \in E \ X_{ii} + X_{jj} - 2X_{ij} = d_{ij}^2 \wedge X - xx^\top = Y^+ - Y^-\} \quad \text{(org)}$$

$$\min\{\|Y^+ + Y^-\|_1 \mid S\bar{Q}^E \odot U = S\,(d_{ij}^2 \mid \{i,j\} \in E)^\top \wedge U - uu^\top = Y^+ - Y^-\} \quad \text{(prj)}$$

[$\bar{Q}^E = RQ^E R^\top$ and $Q^E \odot xx^\top$ encodes the right-hand sides of the DGP constraints]

Algorithms: locSlv based on ipopt local NLP solver

- (org): $X^* = \text{locSlv}(\text{org}, 0), x^* = \text{PCA}(X^*, K)$
- (prj): $\underbrace{\bar{U} = \text{locSlv}(\text{prj}, 0)}_{\text{solve proj. prob}}, \underbrace{\bar{X} = R^\top \bar{U} R}_{\text{variable retrieval}}, \underbrace{\bar{x} = \text{PCA}(\bar{X}, K)}_{\text{dimensionality reduction}}, \underbrace{\tilde{x} = \text{locSlv}(\text{slack}, \bar{x})}_{\text{constraint retrieval}}$

Instance	$\lvert V\rvert$	r	$\lvert E\rvert$	mde*	$\overline{\text{mde}}$	$\widetilde{\text{mde}}$	lde*	$\overline{\text{lde}}$	$\widetilde{\text{lde}}$	t^*	$\tilde{t}$
`names`	87	9	849	1.707	3.486	3.486	4.582	4.999	4.999	58.36	1.68
`1guu`	150	12	955	2.456	2.805	0.081	4.936	4.979	1.186	502.66	4.81
`1guu-1`	150	12	959	2.411	2.437	0.065	4.904	5.851	1.942	65.75	15.48
`2kxa`	333	23	2711	2.203	2.422	0.224	4.707	15.274	4.033	619.00	79.64
`100d`	491	25	5741	2.917	2.268	0.290	4.970	16.198	4.736	1732.88	242.61
`water`	648	26	11939	3.221	2.354	0.461	4.969	12.256	4.512	3659.58	1246.65
`3al1`	681	20	17417	3.105	2.362	0.129	4.988	14.510	4.202	3820.72	560.31

The first successful projection of both variables and constraints!

Some references

- L. Liberti, B. Manca, P.-L. Poirion, *Random projections for the distance geometry problem*, Proceedings of Discrete Mathematics Days (DMD) 2022
- L. Liberti, B. Manca, A. Oustry, P.-L. Poirion, *Random projections for semidefinite programming*, Proceedings of AIRO-ODS, Springer, 2022
- L. Liberti, B. Manca, P.-L. Poirion, *Practical performance of random projections in linear programming*, Symposium of Experimental Algorithms (SEA), LIPICS, 2022
- L. Liberti, P.-L. Poirion, K. Vu, *Random projections for conic programs*, LAA 626:204-220, 2021
- C. D'Ambrosio, L. Liberti, P.-L. Poirion, K. Vu, *Random projections for quadratic programs*, MPB 183:619-647, 2020
- K. Vu, P.-L. Poirion, C. D'Ambrosio, L. Liberti, *Random projections for quadratic programs over a Euclidean ball*, in A. Lodi et al. (eds.), Integer Programming and Combinatorial Optimization (IPCO), LNCS 11480:442-452, 2019
- K. Vu, P.-L. Poirion, L. Liberti, *Gaussian random projections for Euclidean membership problems*, DAM 253:93-102, 2019
- K. Vu, P.-L. Poirion, L. Liberti, *Random projections for linear programming*, MOR 43(4):1051-1071, 2018