School of Mathematics

Recent Advances in Iterative Solvers for Interior Point Methods

Jacek Gondzio

Email: J.Gondzio@ed.ac.uk
http://www.maths.ed.ac.uk/~gondzio/

Outline

- 1984: IPMs were born
- Key ideas
- IPMs today
- Beyond the obvious:
  - How much IPM in IPM?
  - Direct vs Iterative Methods $\longrightarrow$ *Inexact*$^{\mathbf{2}}$ *IPM*
  - Primal-Dual Newton Conjugate Gradient Method
  - Sparse Approximations with IPMs

1984 Do not think of **George Orwell**!

Think like **Frank Sinatra**:
"When I was 24 ... it was a very good year"

Narendra Karmarkar (AT&T Bell Labs) published the paper:
A New Polynomial–time Algorithm for Linear Programming,
Combinatorica 4 (1984) 373–395.

Shocking mathematical concept:

Take *linear* optimization problem
and add *nonlinear* function to the objective.

A step against common sense and
the centuries of mathematical practice:

"nonlinearize" the linear problem

Primal-Dual Pair of Linear Programs

Primal

$$\begin{array}{ll} \min & c^T x \\ \text{s.t.} & Ax = b, \\ & x \geq 0; \end{array}$$

Dual

$$\begin{array}{ll} \max & b^T y \\ \text{s.t.} & A^T y + s = c, \\ & s \geq 0. \end{array}$$

Lagrangian

$$L(x, y) = c^T x - y^T (Ax - b) - s^T x.$$

Optimality Conditions

$$\begin{aligned} Ax &= b, \\ A^T y + s &= c, \\ XSe &= 0, \quad (\text{ i.e., } x_j \cdot s_j = 0 \quad \forall j), \\ (x, s) &\geq 0, \end{aligned}$$

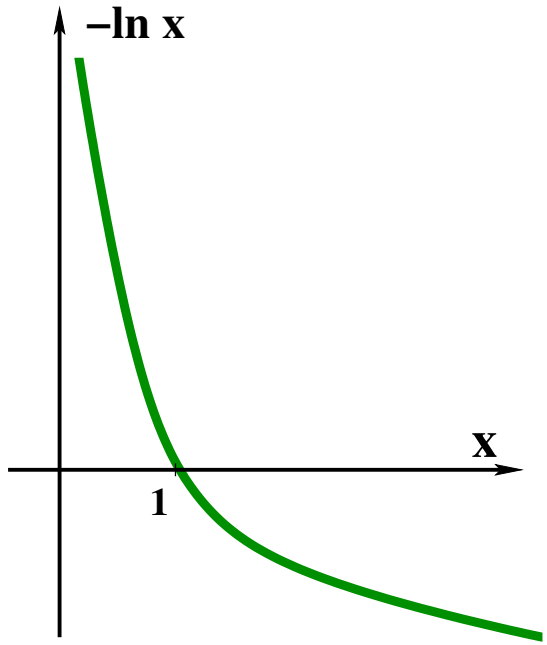
$X = diag\{x_1, \cdots, x_n\}$, $S = diag\{s_1, \cdots, s_n\}$, $e = (1, \cdots, 1) \in \mathcal{R}^n$.

Logarithmic barrier

$$-\ln x_j$$

"replaces" the inequality

$$x_j \geq 0 \ .$$



Observe that

$$\min \ \mathrm{e}^{-\sum_{j=1}^{n} \ln x_j} \quad \Longleftrightarrow \quad \max \prod_{j=1}^{n} x_j$$

The minimization of $-\sum_{j=1}^{n} \ln x_j$ is equivalent to the maximization of the product of distances from all hyperplanes defining the positive orthant: it prevents all x_j from approaching zero.

Logarithmic barrier

Replace the **primal** LP

$$\begin{array}{lll} \min & c^T x & \\ \text{s.t.} & Ax & = b, \\ & x \geq 0, & \end{array}$$

with the **primal barrier program**

$$\begin{array}{ll} \min & c^T x - \mu \sum_{j=1}^{n} \ln x_j \\ \text{s.t.} & Ax = b. \end{array}$$

Lagrangian: $$L(x, y, \mu) = c^T x - y^T (Ax - b) - \mu \sum_{j=1}^{n} \ln x_j.$$

Conditions for a stationary point of the Lagrangian

$$\begin{aligned} \nabla_x L(x,y,\mu) &= c - A^T y - \mu X^{-1} e = 0 \\ \nabla_y L(x,y,\mu) &= \qquad\qquad\qquad\quad Ax - b = 0, \end{aligned}$$

where $X^{-1} = diag\{x_1^{-1}, x_2^{-1}, \cdots, x_n^{-1}\}$.

Let us denote

$$s = \mu X^{-1} e, \qquad \text{i.e.} \qquad XSe = \mu e.$$

The **First Order Optimality Conditions** are:

$$\begin{aligned} Ax &= b, \\ A^T y + s &= c, \\ XSe &= \mu e, \\ (x,s) &> 0. \end{aligned}$$

Approximate FOC

The first order optimality conditions for the barrier problem

$$\begin{aligned} Ax &= b, \\ A^T y + s &= c, \\ XSe &= \boldsymbol{\mu} e, \\ (x, s) &\geq 0 \end{aligned}$$

approximate the first order optimality conditions for the LP

$$\begin{aligned} Ax &= b, \\ A^T y + s &= c, \\ XSe &= 0, \\ (x, s) &\geq 0 \end{aligned}$$

more and more closely as μ goes to zero.

Central Trajectory

Parameter μ controls the distance to optimality.

$$c^T x - b^T y = c^T x - x^T A^T y = x^T (c - A^T y) = x^T s = n\boldsymbol{\mu}.$$

Analytic centre (μ-centre): a (unique) point

$$(x(\mu), y(\mu), s(\mu)), \quad x(\mu) > 0, \; s(\mu) > 0$$

that satisfies FOC.

The path

$$\{(x(\mu), y(\mu), s(\mu)) : \mu > 0\}$$

is called the **primal-dual central trajectory**.

Central Path Neighbourhood

Define $\mathcal{F}^0 := \{(x, y, s) : Ax = b,\ c - A^T y - s = 0,\ x, s > 0\}$.
Assume a primal-dual strictly feasible solution $(x, y, s) \in \mathcal{F}^0$ lying in a neighbourhood of the central path is given; namely (x, y, s) satisfies:

$$\begin{aligned} Ax &= b, \\ A^T y + s &= c, \\ XSe &\approx \mu e. \end{aligned}$$

We define a θ**-neighbourhood** of the central path $N_2(\theta)$, a set of primal-dual strictly feasible solutions $(x, y, s) \in \mathcal{F}^0$ that satisfy:

$$\|XSe - \mu e\| \leq \theta\mu,$$

where $\theta \in (0, 1)$ and the barrier μ satisfies:

$$x^T s = n\mu.$$

Hence $N_2(\theta) = \{(x, y, s) \in \mathcal{F}^0 \mid \|XSe - \mu e\| \leq \theta\mu\}$.

Polynomial Complexity Result

Main ingredients of the polynomial complexity result for the short-step path-following algorithm:

Stay close to the central path:
all iterates stay in the $N_2(\theta)$ neighbourhood of the central path.

Make (slow) progress towards optimality:
reduce systematically duality gap

$$\mu^{k+1} = \sigma \mu^k,$$

where

$$\sigma = 1 - \beta/\sqrt{n},$$

for some $\beta \in (0, 1)$.

$\mathcal{O}(\sqrt{n})$ complexity result

Note that since at one iteration duality gap is reduced $1 - \beta/\sqrt{n}$ times, after $\sqrt{n}$ iterations the reduction achieves:

$$(1 - \beta/\sqrt{n})^{\sqrt{n}} \approx e^{-\beta}.$$

After $C \cdot \sqrt{n}$ iterations, the reduction is $e^{-C\beta}$. For sufficiently large constant C the reduction can thus be arbitrarily large (i.e. the duality gap can become arbitrarily small).

Hence this algorithm has iteration complexity $\mathcal{O}(\sqrt{n})$.

This should be understood as follows:

> *"after the number of iterations proportional to $\sqrt{n}$ the algorithm solves the problem".*

From LP via QP to NLP, SOCP and SDP

For the quadratic cone

$$K_q = \{(x,t) : x \in \mathcal{R}^{n-1}, t \in \mathcal{R},\ t^2 \geq \|x\|^2,\ t \geq 0\},$$

define the *logarithmic barrier function*, $f : \mathcal{R}^n \mapsto \mathcal{R}$

$$f(x,t) = \begin{cases} -\ln(t^2 - \|x\|^2) & \text{if } \|x\| < t \\ +\infty & \text{otherwise.} \end{cases}$$

For the cone $\mathcal{SR}_+^{n\times n}$ of positive definite matrices,
define the *logarithmic barrier function*, $f : \mathcal{SR}_+^{n\times n} \mapsto \mathcal{R}$

$$f(X) = \begin{cases} -\ln\det X & \text{if } X \succ 0 \\ +\infty & \text{otherwise.} \end{cases}$$

LP: Replace $x \geq 0$ with $-\mu \sum_{j=1}^n \ln x_j$.

SDP: Replace $X \succeq 0$ with $-\mu \sum_{j=1}^n \ln \lambda_j = -\mu \ln(\prod_{j=1}^n \lambda_j)$.

Computational View

Apply Newton Method to the FOC

The first order optimality conditions for the barrier problem form a large system of nonlinear equations

$$f(x,y,s) = \begin{bmatrix} Ax - b \\ A^T y + s - c \\ XSe - \mu e \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

Thus, for a given point (x,y,s) we find the Newton direction $(\Delta x, \Delta y, \Delta s)$ by solving the system of linear equations:

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S & 0 & X \end{bmatrix} \cdot \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \begin{bmatrix} b - Ax \\ c - A^T y - s \\ \mu e - XSe \end{bmatrix}.$$

Linear Algebra Perspective

Every IPM iteration requires solving a linear system:

$$\begin{bmatrix} -Q - \Theta_P^{-1} & A^T \\ A & \Theta_D \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix}.$$

Details depend on a class of problem and the presence of inequalities:

LP	**QP**	**NLP**
$\begin{bmatrix} -\Theta^{-1} & A^T \\ A & 0 \end{bmatrix}$	$\begin{bmatrix} -Q - \Theta^{-1} & A^T \\ A & 0 \end{bmatrix}$	$\begin{bmatrix} Q(x,y) & A(x)^T \\ A(x) & \Theta \end{bmatrix}$

- Optimizers call it a **reduced KKT system**
- PDE community calls it a **saddle point system**

Interior Point Methods:

- Unified view of optimization
  $\rightarrow$ from LP via QP to NLP, SOCP and SDP
- Predictable behaviour
  $\rightarrow$ small number of iterations
- Unequalled efficiency
  - competitive for small problems ($n \leq 10^6$)
  - beyond competition for large problems ($n \geq 10^6$)

Problem of size $\mathbf{10^9}$ solved in 2005.

Object-Oriented Parallel IPM Solver (OOPS):
`http://www.maths.ed.ac.uk/~gondzio/parallel/solver.html`

Gondzio and Grothey, Parallel IPM solver for structured QPs: application to financial planning problems, *Annals of Operations Research* 152 (2007) 319-339.

Improvements

Use 2nd-order information (Newton direction).

But, do not waste time on computing *exact* direction.

Use Inexact Newton Method

Dembo, Eisenstat and Steihaug,
Inexact Newton Methods,
SIAM J. on Numerical Analysis 19 (1982) 400–408.

Bellavia, Inexact Interior Point Method,
Journal of Optimization Theory and Appls 96 (1998) 109–121.

Main Tool: Inexact Newton Method

Replace an *exact* Newton direction

$$\nabla^2 f(x)\Delta x = -\nabla f(x)$$

with an *inexact* one:

$$\nabla^2 f(x)\Delta x = -\nabla f(x) + \boldsymbol{r},$$

where the *error* $\boldsymbol{r}$ is small: $\|\boldsymbol{r}\| \leq \boldsymbol{\eta}\|\nabla f(x)\|, \ \boldsymbol{\eta} \in (0,1)$.

Theorem: Suppose the feasible IPM for QP is used.

If the method operates in the *small* neighbourhood

$$\mathcal{N}_2(\theta) := \{(x, y, s) \in \mathcal{F}^0 : \|XSe - \mu e\|_2 \leq \theta\mu\}$$

and uses the *inexact* Newton direction with $\boldsymbol{\eta} = 0.3$, then it converges in at most

$$K = \mathcal{O}(\sqrt{\boldsymbol{n}}\ \ln(1/\epsilon)) \quad \text{iterations.}$$

If the method operates in the *symmetric* neighbourhood

$$\mathcal{N}_S(\gamma) := \{(x, y, s) \in \mathcal{F}^0 : \gamma\mu \leq x_i s_i \leq (1/\gamma)\mu\}$$

and uses the *inexact* Newton direction with $\boldsymbol{\eta} = 0.05$, then it converges in at most

$$K = \mathcal{O}(\boldsymbol{n}\ \ln(1/\epsilon)) \quad \text{iterations.}$$

Gondzio, Convergence Analysis of an Inexact Feasible IPM for Convex Quadratic Programming, *SIAM Journal on Optimization* 23 (2013) No 3, pp. 1510-1527.

Overarching Feature of IPMs

They possess an unequalled ability to identify
the **"essential subspace"**
in which the optimal solution is hidden.

Beyond the Obvious

- How much IPM in IPM?
  $\rightarrow$ do we need to be so rigid?
  work with **S. Bellavia, M. Porcelli, S. Pougkakiotis**

- *Inexact*$^{\mathbf{2}}$ IPMs
  $\rightarrow$ IPM-tuned stopping criteria for Krylov methods,
  work with **F. Zanetti**

- Primal-Dual Newton Conjugate Gradient Method
  $\rightarrow$ homotopy similar to IPM
  work with **K. Fountoulakis**

- Sparse Approximations with IPMs
  $\rightarrow$ ℓ_1-regularized problems, work with
  V.De Simone, D.di Serafino, S.Pougkakiotis, M.Viola

Relaxed (?) Interior Point Method

SDP with IPMs still remains a challenge

Wishes:

- Remove memory bottleneck
- Accelerate (if possible)

Redesign IPMs for SDP:

- "Relax" the rigid structure of the method
- Replace *exact* Newton Method
  with *inexact* Newton Method
- Work in *matrix-free* and *limited-memory* regime
- Using a preconditioner is essential

SDP in standard form

- Primal form

$$\begin{array}{ll} \min & C \bullet X \\ \text{s.t.} & A_i \bullet X = b_i \quad i = 1, \ldots, m \\ & X \succeq 0, \end{array}$$

where $A_i \in S\mathbb{R}^{n\times n}, C \in S\mathbb{R}^{n\times n}$, $b \in \mathbb{R}^m$ and $X \in S\mathbb{R}^{n\times n}$.

- Dual form

$$\begin{array}{ll} \max & b^T y \\ \text{s.t.} & S = C - \sum_{i=1}^m y_i A_i \\ & S \succeq 0, \end{array}$$

where $y \in \mathbb{R}^m$ and $S \in S\mathbb{R}^{n\times n}$.

The operation $A \bullet B = trace(A^T B)$.

Special interest in S sparse.
S is the linear combination: $S = C - \sum_{i=1}^m y_i A_i$

Dual Path-Following Interior-Point Algorithm

- Dual barrier problem parametrized by $\mu > 0$

$$\begin{array}{ll} \max & b^T y + \mu \ln(det(S)), \\ \text{s.t.} & \sum_{i=1}^m y_i A_i + S = C \quad (S \succeq 0). \end{array}$$

- Let $X = \mu S^{-1} \succeq 0$, then the first-order optimality conditions for this problem are given by:

$$F_\mu(X, y, S) = \begin{pmatrix} \sum_{i=1}^m y_i A_i + S - C \\ A_i \bullet X - b_i \quad i = 1, \ldots, m \\ X - \mu S^{-1} \end{pmatrix} = 0.$$

Primal-dual complementarity condition: $XS = \mu I$.

Dual Path-Following IPM (cont'd)

Choose a dual strictly feasible pair (S, y) and a scalar $\mu > 0$.

Outer **Interior-Point** iterations:
Update (reduce) $\mu := \sigma\mu$ until it is sufficiently small.

Inner **Newton** iterations:
Perform (damped) steps in Newton direction $(\Delta X, \Delta S, y)$ for

$$F_\mu(X, y, S) = 0$$

until the following proximity criteria is satisfied:

$$\|S^{-1/2} \Delta S \, S^{-1/2}\|_F \leq \tau < 1.$$

Bellavia, Gondzio and Porcelli, An inexact dual logarithmic barrier method for solving sparse semidefinite programs, *Mathematical Programming*, 178 (2019), pp 109–143.

Bellavia, Gondzio and Porcelli, A relaxed interior point method for low-rank semidefinite programming problems with applications to matrix completion (revised in Mar 2021) `http://arxiv.org/abs/1909.06099`

Inexact^2 IPM

Standard *Inexact Newton Method*

$$\nabla^2 f(x)\Delta x = -\nabla f(x) + \boldsymbol{r},$$

where the *error* $\boldsymbol{r}$ is small: $\|\boldsymbol{r}\| \leq \boldsymbol{\eta}\|\nabla f(x)\|,\ \boldsymbol{\eta} \in (0,1)$
is *disappointingly conservative* when applied in IPMs!

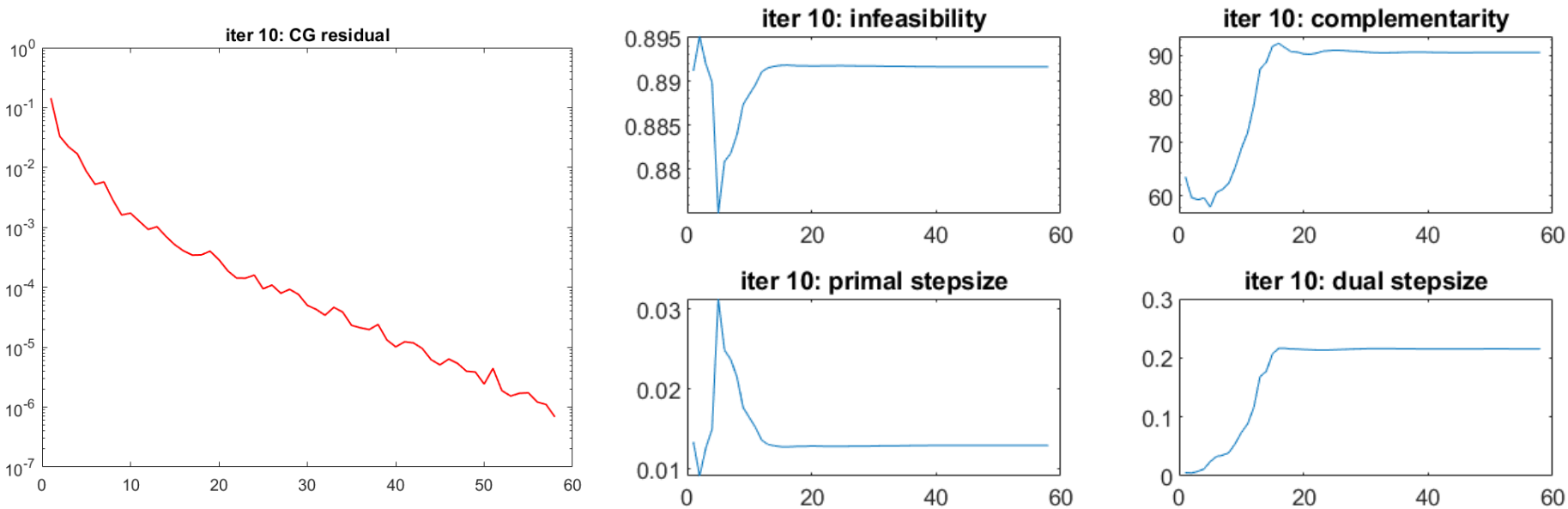
Newton direction $(\Delta x, \Delta y, \Delta s)$ comes from the system of linear equations:

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S & 0 & X \end{bmatrix} \cdot \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \begin{bmatrix} \xi_P \\ \xi_D \\ \xi_\mu \end{bmatrix} = \begin{bmatrix} b - Ax \\ c - A^T y - s \\ \sigma\mu e - XSe \end{bmatrix}.$$

Full step in Newton direction ($\alpha = 1$) would immediately reach primal feasibility and dual feasibility.
In practice such steps rarely happen.
Why should we waste time on computing accurate directions?

Intriguing observation

What happens at a particular IPM iteration?



The accuracy required from the inner solver **does not change** the quality of Newton direction!

Stop inner solver as soon as the **stagnation** occurs.

Inexact^2 IPM

Accept the direction produced by the inner solver as soon as

$$\max_j \left| \frac{\Delta x_j^k}{x_j^k} \right| \leq M, \qquad \max_j \left| \frac{\Delta s_j^k}{s_j^k} \right| \leq M$$

and

$$\|\xi_P^{k+1}\| \leq \eta_k \|\xi_P^k\|, \quad \|\xi_D^{k+1}\| \leq \eta_k \|\xi_D^k\|,$$

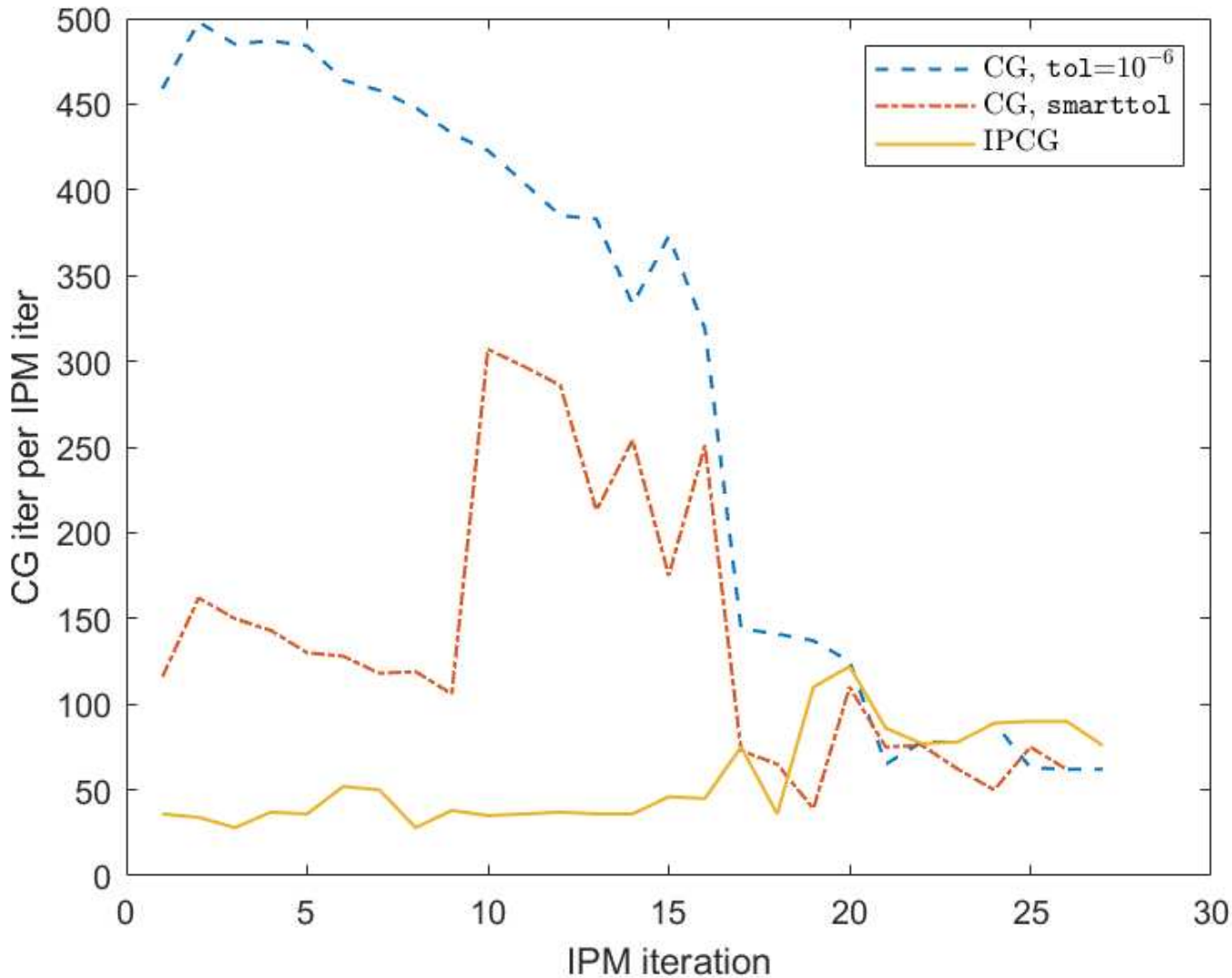
where $\eta_k \geq 1 - \alpha_k$.

- Implemented with CG and MINRES;
- Prevents IPM from "over-solving" of the linear systems
  $\rightarrow$ 70%-90% reduction of the number of Krylov iterations;
- Worst-case complexity drops from $\mathcal{O}(n)$ to $\mathcal{O}(n^2)$.

F. Zanetti and J. Gondzio, (`arXiv: 2106.16090`)

A new stopping criterion for Krylov solvers applied in interior point methods, June 2021 (submitted).

Inexact² IPM



Multi-Energy X-ray Tomography

$$\min_{x \geq 0} \ \|h - \mathcal{A}x\|_2^2 + \alpha\|x\|_2^2 + \beta x^T S x,$$

where $S = \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}$ is an *inner product regularizer* which promotes material separation (note $x_1 \geq 0, x_2 \geq 0$, keep $x_1^T x_2$ small).

	CG, $tol = 10^{-6}$			IPCG, $\varepsilon = 10^{-2}$		
Size	IPM	PCG	Time	IPM	PCG	Time
2,048	18	3,810	7.46	19	586	1.44
8,192	20	6,301	35.04	24	1,149	6.29
32,768	23	9,249	140.91	26	1,366	23.02
131,072	26	15,115	817.45	32	1,763	106.36
524,288	29	25,112	5,174.26	49	2,639	639.92

J. Gondzio, S.-M. Latva-Äijö, S.M Siltanen, M. Lassas, F. Zanetti, (`arXiv: 2107.03535`)
Material-separating regularizer for multi-energy X-ray tomography, June 2021 (submitted).

Big Data Optimization

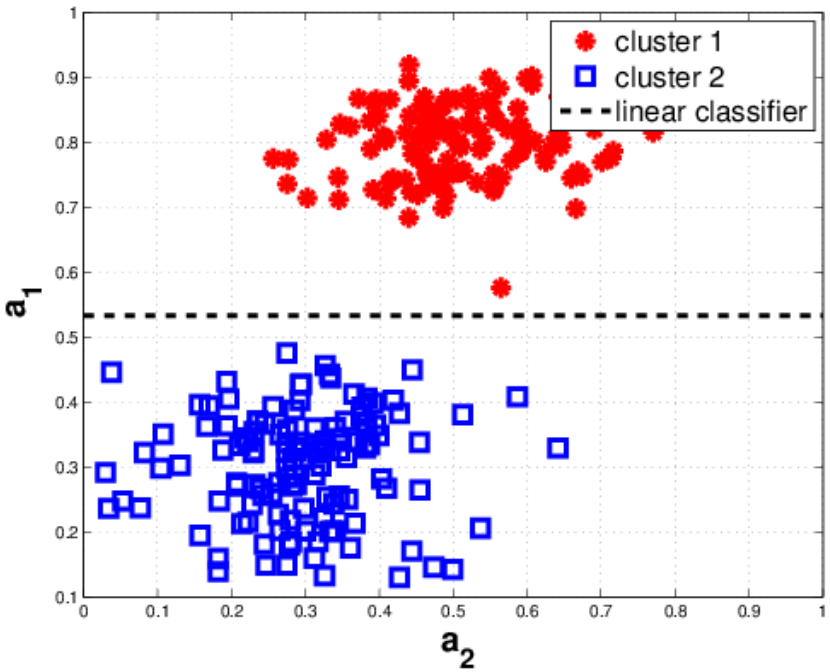
Sparse Approximation

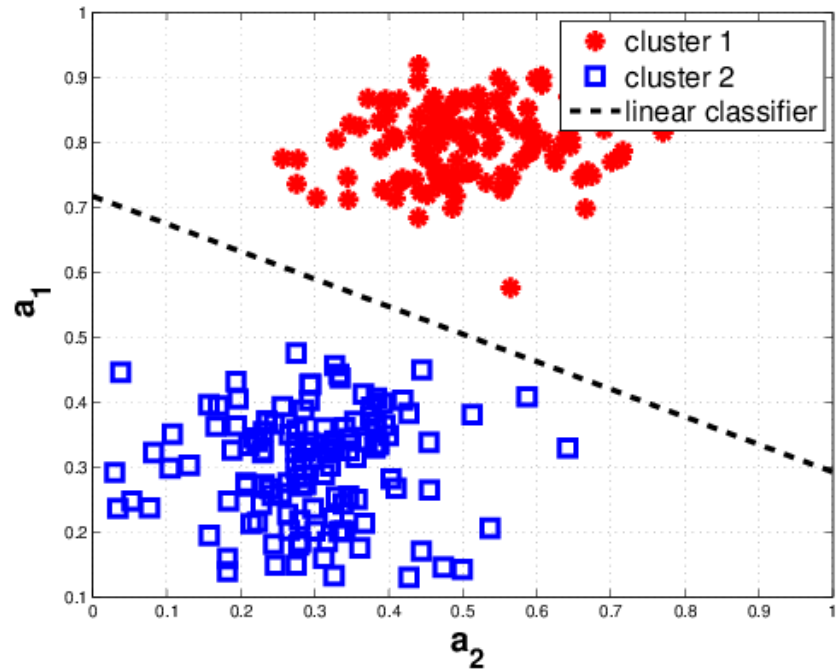
- Machine Learning: Classification with SVMs
- Statistics: Estimate x from observations
- Wavelet-based signal/image reconst. & restoration
- Compressed Sensing (Signal Processing)

All such problems lead to the same dense, possibly very large QP.

Binary Classification

$$\min \tau\|x\|_1 + \sum_{i=1}^{m} \log(1+e^{-b_i x^T a_i}) \qquad \min \tau\|x\|_2^2 + \sum_{i=1}^{m} \log(1+e^{-b_i x^T a_i})$$





ℓ_1-regularization

$$\min_x \quad \tau\|x\|_1 + \phi(x).$$

think of LASSO:

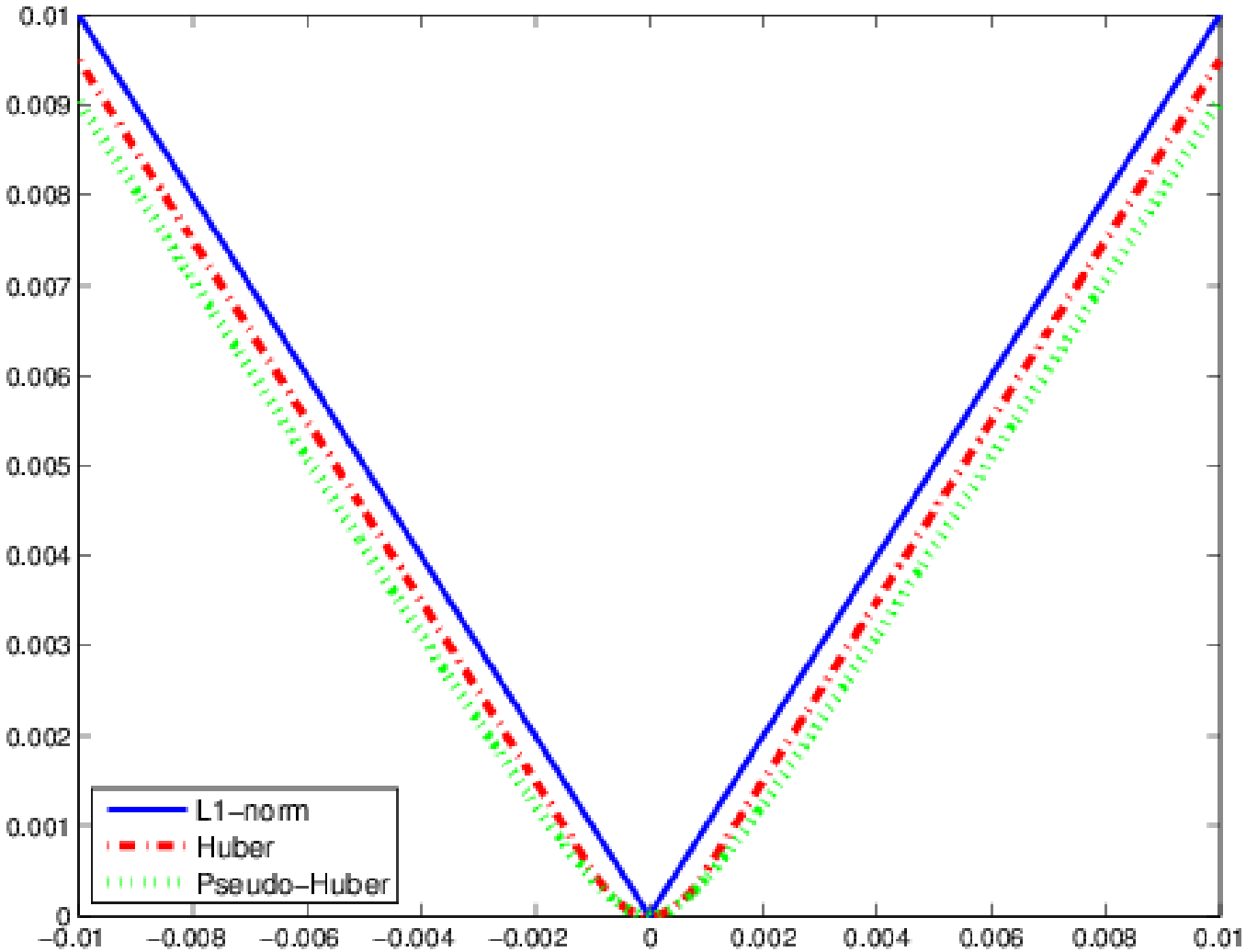
$$\min_x \; f(x) = \tau\|x\|_1 + \|Ax - b\|_2^2$$

Unconstrained optimization $\Rightarrow$ easy
Serious Issue: nondifferentiability of $\|.\|_1$

Two possible tricks:

- Splitting $x = u - v$ with $u, v \geq 0$
- Smoothing with pseudo-Huber approximation replaces $\|x\|_1$ with $\psi_\mu(x) = \sum_{i=1}^n (\sqrt{\mu^2 + x_i^2} - \mu)$

Huber:


0.01
0.009
0.008
0.007
0.006
0.005
0.004
0.003
0.002
0.001
0
−0.01
−0.008
−0.006
−0.004
−0.002
0
0.002
0.004
0.006
0.008
0.01
L1−norm
Huber
Pseudo−Huber

Continuation

Embed inexact Newton Method into a *homotopy* approach:

- Inequalities $u \geq 0$, $v \geq 0$ $\longrightarrow$ use **IPM**

  replace $z \geq 0$ with $-\mu \, log z$ and drive μ to zero.

- pseudo-Huber regression $\longrightarrow$ use **continuation**

  replace $|x_i|$ with $\mu(\sqrt{1+\frac{x_i^2}{\mu^2}}-1)$ and drive μ to zero.

Questions:

- Theory?
- Practice?

Compressed Sensing and Continuation

Replace $$\min_x \quad f(x) = \tau\|W^T x\|_1 + \frac{1}{2}\|Ax - b\|_2^2, \qquad \longrightarrow \boldsymbol{x_\tau}$$

with $$\min_x \quad f_\mu(x) = \tau\psi_\mu(W^T x) + \frac{1}{2}\|Ax - b\|_2^2, \qquad \longrightarrow \boldsymbol{x_{\tau,\mu}}$$

Solve approximately a family of problems for a (short) decreasing sequence of μ's: $\boldsymbol{\mu_0 > \mu_1 > \mu_2 \cdots}$

Theorem (Brief description)

There exists a $\tilde{\mu}$ such that $\forall \mu \leq \tilde{\mu}$ the difference of the two solutions satisfies

$$\|x_{\tau,\mu} - x_\tau\|_2 = \mathcal{O}(\mu^{1/2}) \quad \forall\, \tau, \mu.$$

Primal-Dual Newton Conjugate Gradient Method:

Fountoulakis and Gondzio, A Second-order Method for Strongly Convex ℓ_1-regularization Problems, *Mathematical Programming*, 156 (2016) 189–219.

Dassios, Fountoulakis and Gondzio, A Preconditioner for a Primal-Dual Newton Conjugate Gradient Method for Compressed Sensing Problems, *SIAM J on Scientific Computing*, 37 (2015) A2783–A2812.

Simple test example for ℓ_1-regularization

$$\min_x \ \tau\|x\|_1 + \|Ax - b\|_2^2$$

Special matrix given in SVD form $A = U\Sigma V^T$, where U and V are products of Givens rotations. The user controls:

- the condition number $\kappa(A)$,
- the sparsity of matrix A.

Matlab generator:
http://www.maths.ed.ac.uk/ERGO/trillion/

Fountoulakis and Gondzio

Performance of First- and Second-Order Methods for ℓ_1-regularized Least Squares Problems,

Computational Optimization and Applications 65 (2016) 605–635.

Excessive Computational Tests (4 mths of CPU)

- FISTA (Fast Iterative Shrinkage-Thresholding Algorithm)
- PCDM (Parallel Coordinate Descent Method)
- PSSgb (Projected Scaled Subgradient, Gafni-Bertsekas)
- pdNCG (primal-dual Newton Conjugate Gradient)

The **1st order** methods:

- work well if the condition number $\kappa(A) \leq 10^2$,
- struggle when $\kappa(A) \geq 10^3$,
- stall when $\kappa(A) \geq 10^4$.

The **2nd order** method (pdNCG, diagonal preconditioner):

- works well if the condition number $\kappa(A) \leq 10^6$.

Let us go big: a trillion (2^{40}) variables

n (billions)	Processors	Memory (TB)	time (s)
1	64	0.192	1923
4	256	0.768	1968
16	1024	3.072	1986
64	4096	12.288	1970
256	16384	49.152	1990
1,024	65536	196.608	2006

ARCHER (ranked 25 on `top500.com`, 11 March 2015)

Linpack Performance (Rmax) 1,642.54 TFlop/s
Theoretical Peak (Rpeak) 2,550.53 TFlop/s

More Sparse Approximations

Problems of the form

$$\begin{array}{ll} \min & f(x) + \tau_1 \|x\|_1 + \tau_2 \|Lx\|_1 \\ \text{s.t.} & Ax = b. \end{array}$$

- Sparse portfolio selection
  comparison with Split Bregman method
- Classification models for funct'l Magnetic Resonance Imaging
  comparison with FISTA and ADMM
- TV-based Poisson Image Restoration
  comparison with PDAL
- Linear Classification via Regularized Logistic Regression
  comparison with newGLMNET and ADMM

De Simone, di Serafino, Gondzio, Pougkakiotis, Viola,

Sparse Approximations with Interior Point Methods (released in February 2021)

`https://arxiv.org/abs/2102.13608`

Classification models for fMRI

Comparison of IPM, FISTA and ADMM (opt tol 10^{-5}). We report:

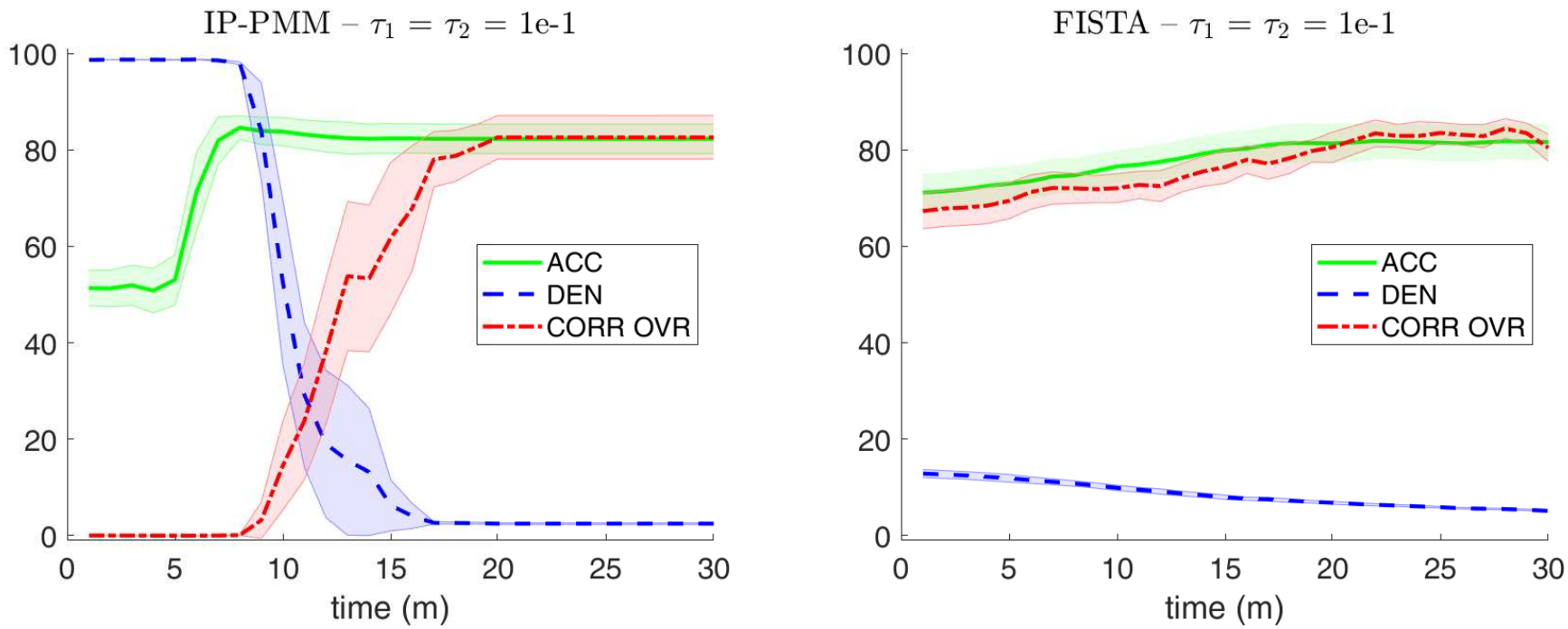
- *classification accuracy* (ACC),
- *corrected pairwise overlap* (CORR OVR); measures the "stability" of each voxel selection,
- *solution density* (DEN).

Algorithm	$\tau_1 = \tau_2$	**ACC**	**CORR OVR**	**DEN**
IP-PMM	10^{-2}	86.16 ± 7.11	43.47 ± 9.09	20.56 ± 6.63
	$5 \cdot 10^{-2}$	84.90 ± 4.80	62.70 ± 10.39	3.77 ± 0.84
	10^{-1}	82.29 ± 6.22	82.60 ± 9.24	2.49 ± 0.34
FISTA	10^{-2}	86.90 ± 5.01	5.43 ± 0.43	88.97 ± 0.71
	$5 \cdot 10^{-2}$	84.15 ± 5.92	65.50 ± 2.68	19.36 ± 0.86
	10^{-1}	81.62 ± 7.58	80.44 ± 5.72	5.14 ± 0.44
ADMM	10^{-2}	86.46 ± 6.91	0.03 ± 0.01	98.70 ± 0.03
	$5 \cdot 10^{-2}$	85.57 ± 5.37	0.15 ± 0.04	97.97 ± 0.05
	10^{-1}	82.07 ± 6.51	0.26 ± 0.13	97.50 ± 0.19

We want: ACC and CORR OVR *close to 100*, and *small* DEN.

Classification models for fMRI (cont'd)

Performance comparison in terms of elapsed time:



Evolution of ACC, DEN and CORR OVR with time;
IP-PMM (*left*) and FISTA (*right*).
We report average measures with 95% confidence intervals.

Conclusions

IPMs have revolutionized the optimization.

They are clearly the **2nd-order** methods,
but work well with the **inexact Newton method**
(this makes the **matrix-free** implementation possible).

Trick:

- use continuation
- find the **"essential subspace"** and
- exploit it to simplify the linear algebra
  - works in IPMs for **LP, QP, SOCP, SDP**
  - works in Newton CG for **ℓ_1-regularization**
  - works in IPMs for **sparse approximations**

Thank you for your attention

Stay safe, stay healthy!