EPSRC Pioneering research and skills

MATHEMATICS FOR VAST DIGITAL RESOURCES

LONDON MATHEMATICAL SOCIETY 150 YEARS

NAIS Centre for Numerical Algorithms and Intelligent Software

amazon.com

Google

EDINBURGH MATHEMATICAL SOCIETY

Training Machine Learning Models via Empirical Risk Minimization (Lecture 1)

Peter Richtárik

THE UNIVERSITY OF EDINBURGH

THE ALAN TURING INSTITUTE

The 41st Woudschoten Conference - October 5-7, 2016

Part 1
Training Linear Predictors

The Idea

Statistical Nature of Data

$$(A_i, y_i) \sim Distribution$$

DATA


The New York Times

WEDNESDAY, MAY 10, 2006

Above, finalists from Nick at Nite's "Funniest Mom in America 2." Right, Darlene Westgor, the 2005 winner.

A Mom's Riff on the Parodies of Parenthood

$A_i \in \mathbb{R}^{d \times m}$

LABEL

"politics"

$y_i \in \mathbb{R}^m$

Prediction of Labels from Data

Find $w \in \mathbb{R}^d$ ← Linear predictor

such that when a (data, label) pair is drawn from the distribution

$$(A_i, y_i) \sim Distribution$$

then

$$A_i^\top w \approx y_i$$

Predicted label → $A_i^\top w$ ← True label

Measure of Success

$$loss(a, b)$$

Predicted label

True label

We want the expected loss (=risk) to be small:

$$\mathbf{E}\left[loss(A_i^\top w, y_i)\right]$$

$$(A_i, y_i) \sim Distribution$$

Replace Expectation by Average

Draw i.i.d. data samples from the distribution

$$(A_1, y_1), (A_2, y_2), \ldots, (A_n, y_n) \sim Distribution$$

Output predictor which minimizes the Empirical Risk:

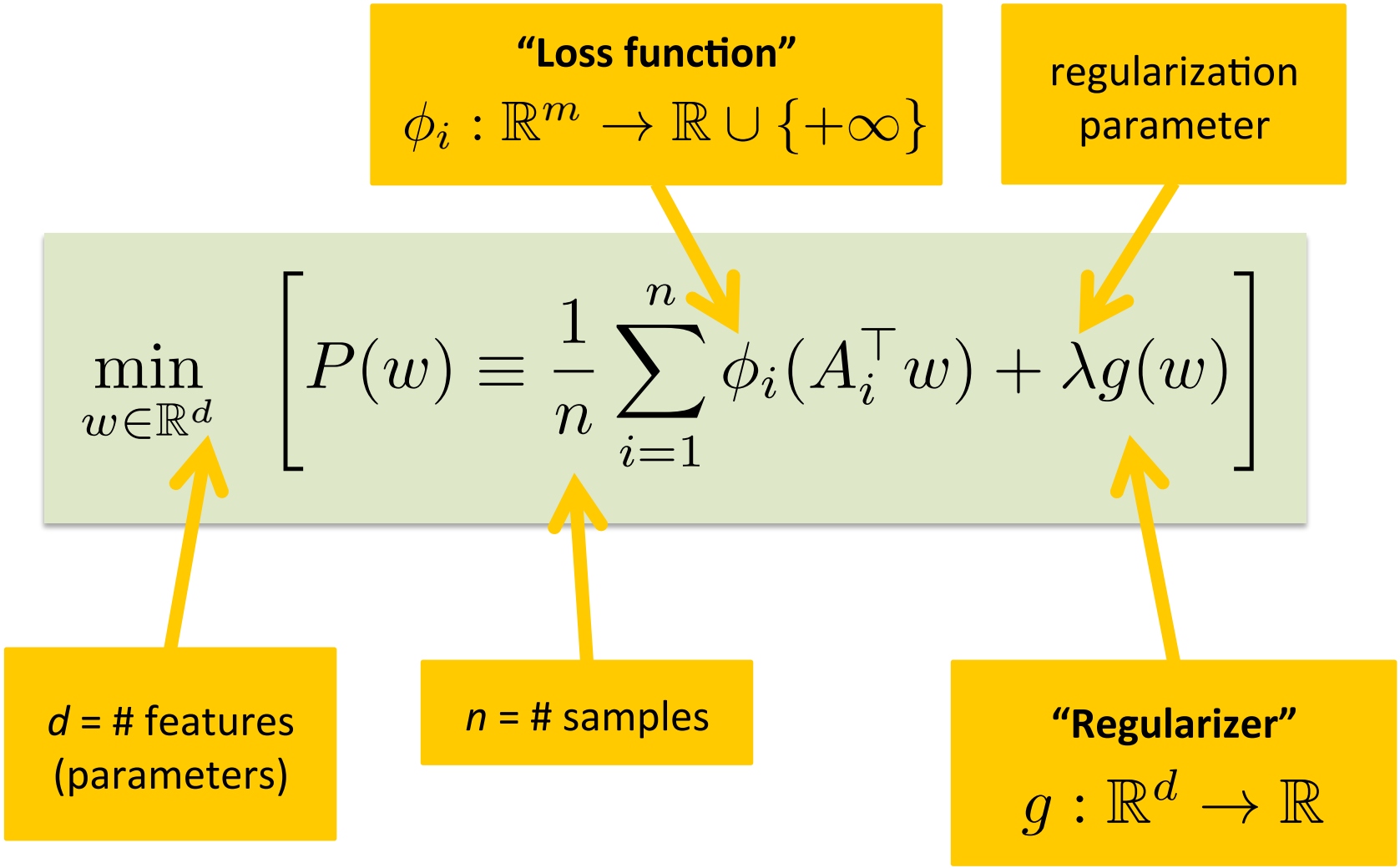
$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} loss(A_i^\top w, y_i)$$

Includes:

- Linear systems
- Linear regression
- Logistic regression
- Binary classification
- Neural networks

Empirical Risk Minimization

Primal Problem



Dual Problem

$$D(\alpha) \equiv -\lambda g^* \left(\underbrace{\frac{1}{\lambda n} \sum_{i=1}^{n} A_i \alpha_i}_{\in \mathbb{R}^d} \right) - \frac{1}{n} \sum_{i=1}^{n} \phi_i^*(\underbrace{-\alpha_i}_{\in \mathbb{R}^m})$$

Fenchel conjugate

Fenchel conjugate

$$g^*(w') = \max_{w \in \mathbb{R}^d} \left\{ (w')^\top w - g(w) \right\}$$

$$\phi_i^*(a') = \max_{a \in \mathbb{R}^m} \left\{ (a')^\top a - \phi_i(a) \right\}$$

$$\max_{\alpha = (\underbrace{\alpha_1}_{\in \mathbb{R}^m}, \ldots, \underbrace{\alpha_n}_{\in \mathbb{R}^m}) \in \mathbb{R}^N = \mathbb{R}^{nm}} D(\alpha)$$

Concave

Duality

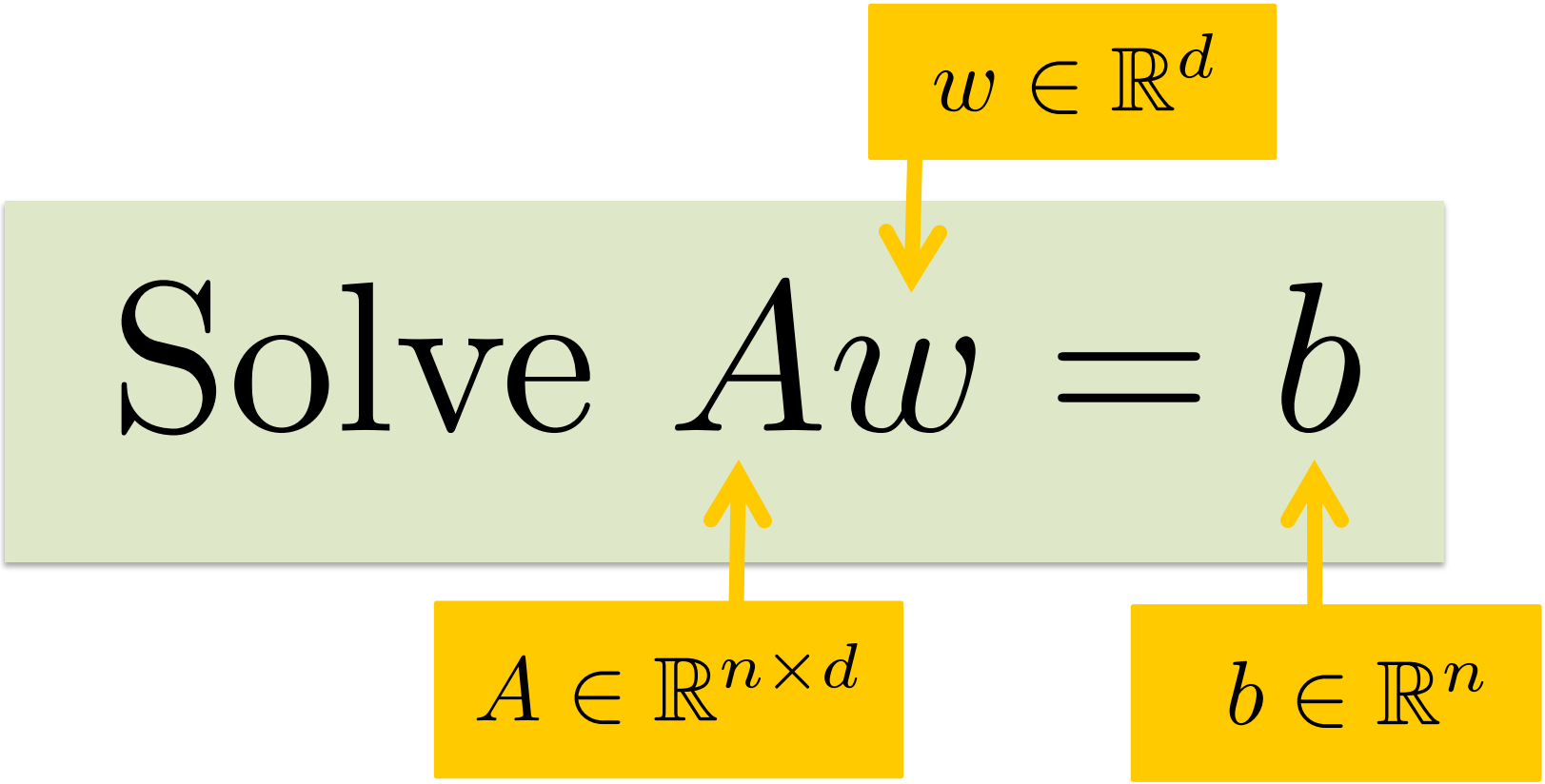
Weak Duality: $P(w) \geq D(\alpha) \quad \forall w \in \mathbb{R}^d,\ \alpha \in \mathbb{R}^N$

Strong Duality: $P(w^*) = D(\alpha^*)$ (Under suitable assumptions)

$$w^* = \frac{1}{\lambda n} \sum_{i=1}^{n} A_i \alpha_i^*$$

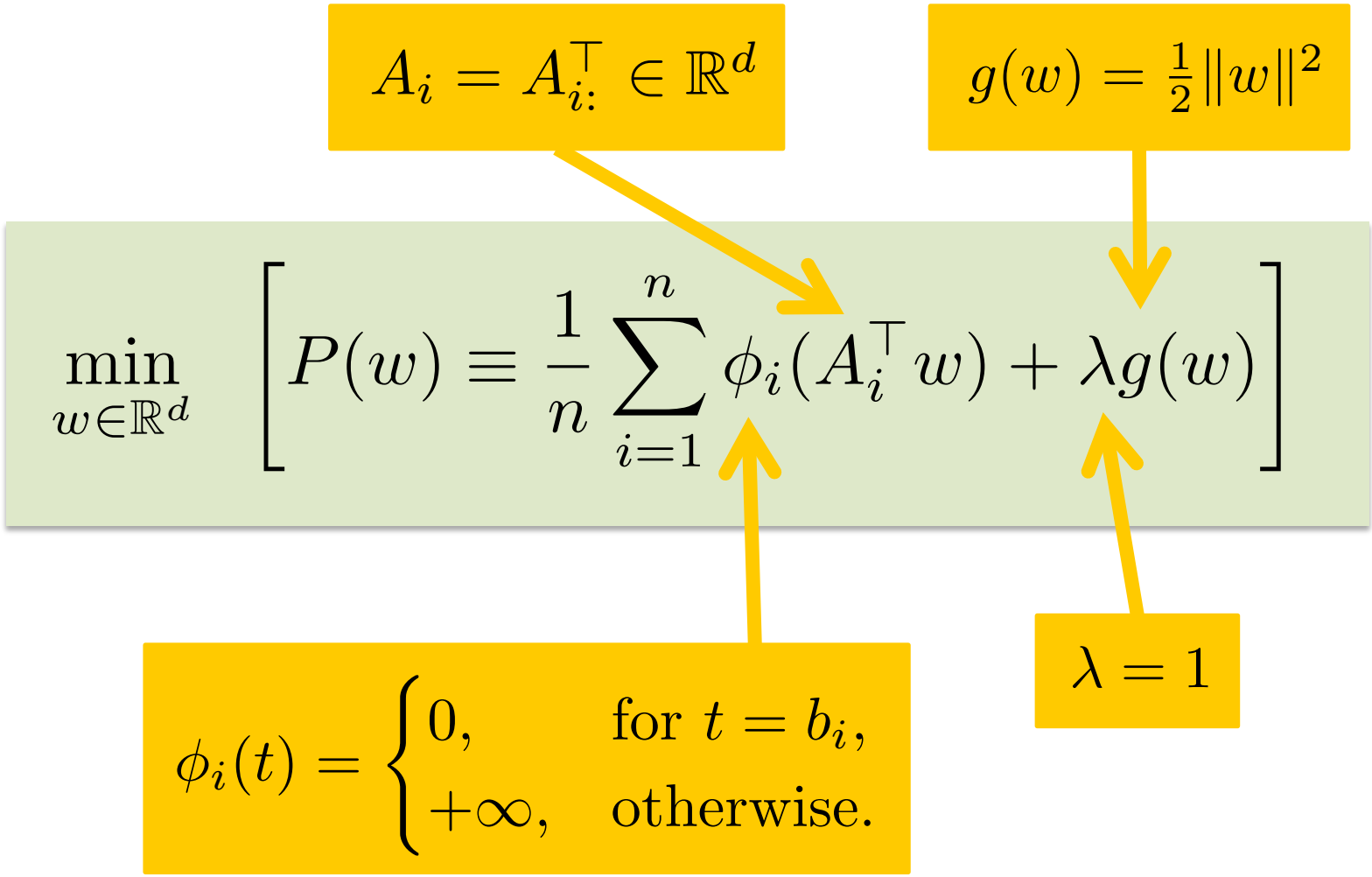
Setup 1:
Solving Linear Systems

Solving Linear Systems (with Many Rows)



Think: $n \gg d$

Linear System: Primal ERM Problem



Primal Problem

Find the least-norm solution of $Aw = b$:

$$\min_{w \in \mathbb{R}^d} \frac{1}{2}\|w\|^2$$

$$\text{subject to } Aw = b$$

Linear System: Dual ERM Problem

Unconstrained (non-strongly) concave quadratic maximization:

$$\max_{\alpha \in \mathbb{R}^n} D(\alpha) \equiv \frac{1}{n} b^\top \alpha - \frac{1}{2n^2} \|A^\top \alpha\|^2$$

$$\phi_i^*(t) = b_i t$$

$$g^*(w) = \tfrac{1}{2}\|w\|^2$$

Further Reading

Primal View:

Robert M. Gower and P.R.
Randomized Iterative Methods for Linear Systems
SIAM J. on Matrix Analysis and Applications 36(4), 1660-1690, 2015

2nd Most Downloaded Paper on SIMAX

Dual View:

Robert M. Gower and P.R.
Stochastic Dual Ascent for Solving Linear Systems
arXiv:1512.06890, 2015

Inverting Matrices & Connection to Quasi-Newton Methods:

Robert M. Gower and P.R.
Randomized Quasi-Newton Updates are Linearly Convergent Matrix Inversion Algorithms
arXiv:1602.01768, 2016

Further Reading

Application of Stochastic Matrix Inversion to General ERM:

Robert M. Gower, Donald Goldfarb and P.R.
Stochastic Block BFGS: Squeezing More Curvature out of Data
ICML, 2016

Acceleration:

P.R. and Martin Takáč
Stochastic Reformulations of Linear Systems and Fast Stochastic Iterative Methods
Manuscript, 2016

Setup 2:
Ridge Regression

Primal Problem

Loss: quadratic

$$\phi_i(t) = \tfrac{1}{2}(t - b_i)^2$$

Regularizer: Tikhonov (L2)

$$g(w) = \tfrac{1}{2}\|w\|^2$$

$$\min_{w \in \mathbb{R}^d} \left[P(w) \equiv \frac{1}{n} \sum_{i=1}^{n} \phi_i(A_i^\top w) + \lambda g(w) \right]$$

$$P(w) = \tfrac{1}{2n}\|Aw - b\|^2 + \tfrac{\lambda}{2}\|w\|^2$$

i^{th} row of A: $A_{i:} = A_i^\top$

Setup 3: Something More General …

Primal Problem

Loss: convex & $1/\gamma$ - smooth

$$\|\nabla \phi_i(t) - \nabla \phi_i(t')\| \leq \gamma^{-1} \cdot \|t - t'\|$$

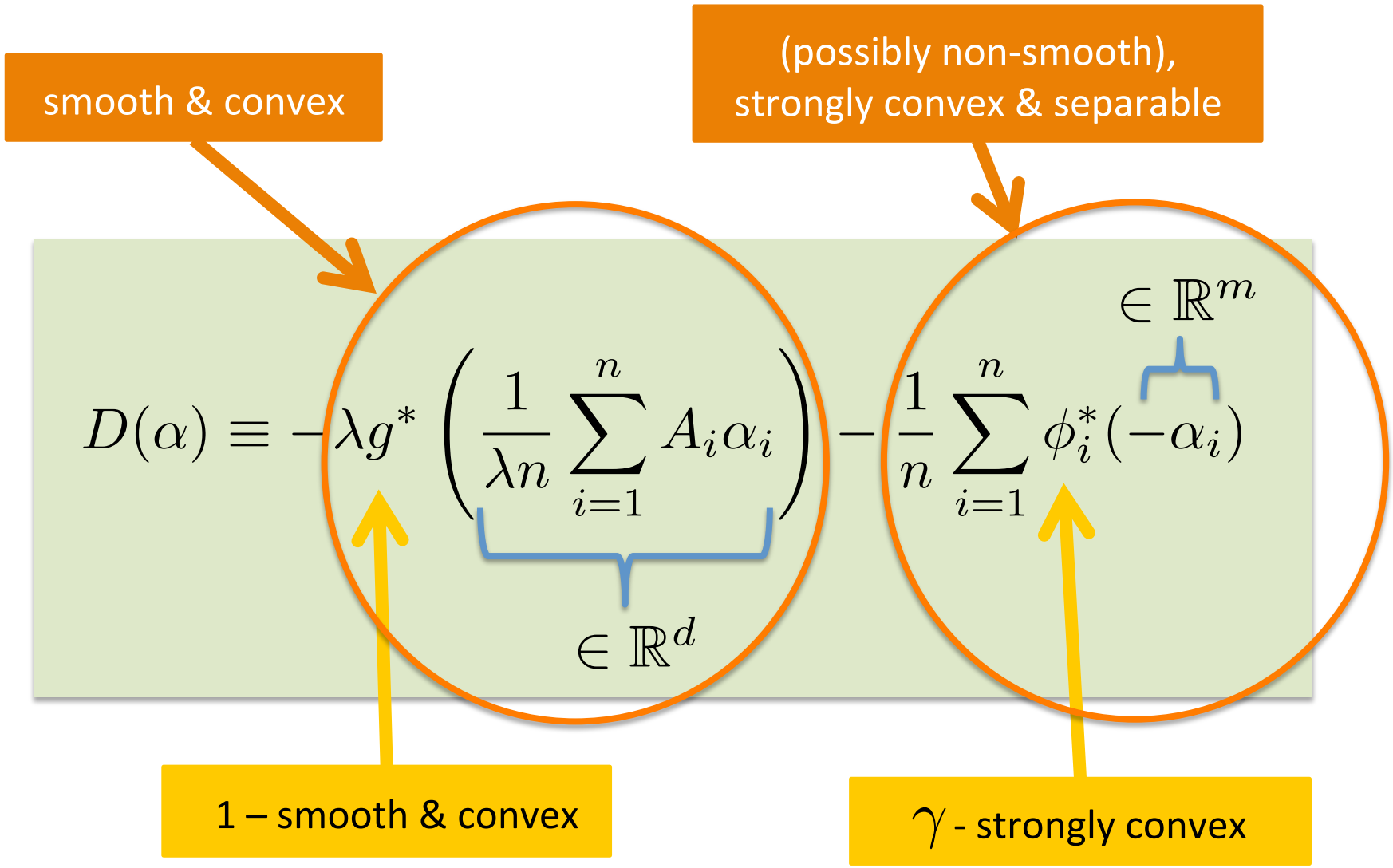
Lipschitz constant

$$\min_{w \in \mathbb{R}^d} \left[P(w) \equiv \frac{1}{n} \sum_{i=1}^{n} \phi_i(A_i^\top w) + \lambda g(w) \right]$$

Regularizer: 1-strongly convex

$$g(w) \geq g(w') + \langle \nabla g(w'), w - w' \rangle + \frac{1}{2}\|w - w'\|^2$$

Dual Problem



Part 2
Tools

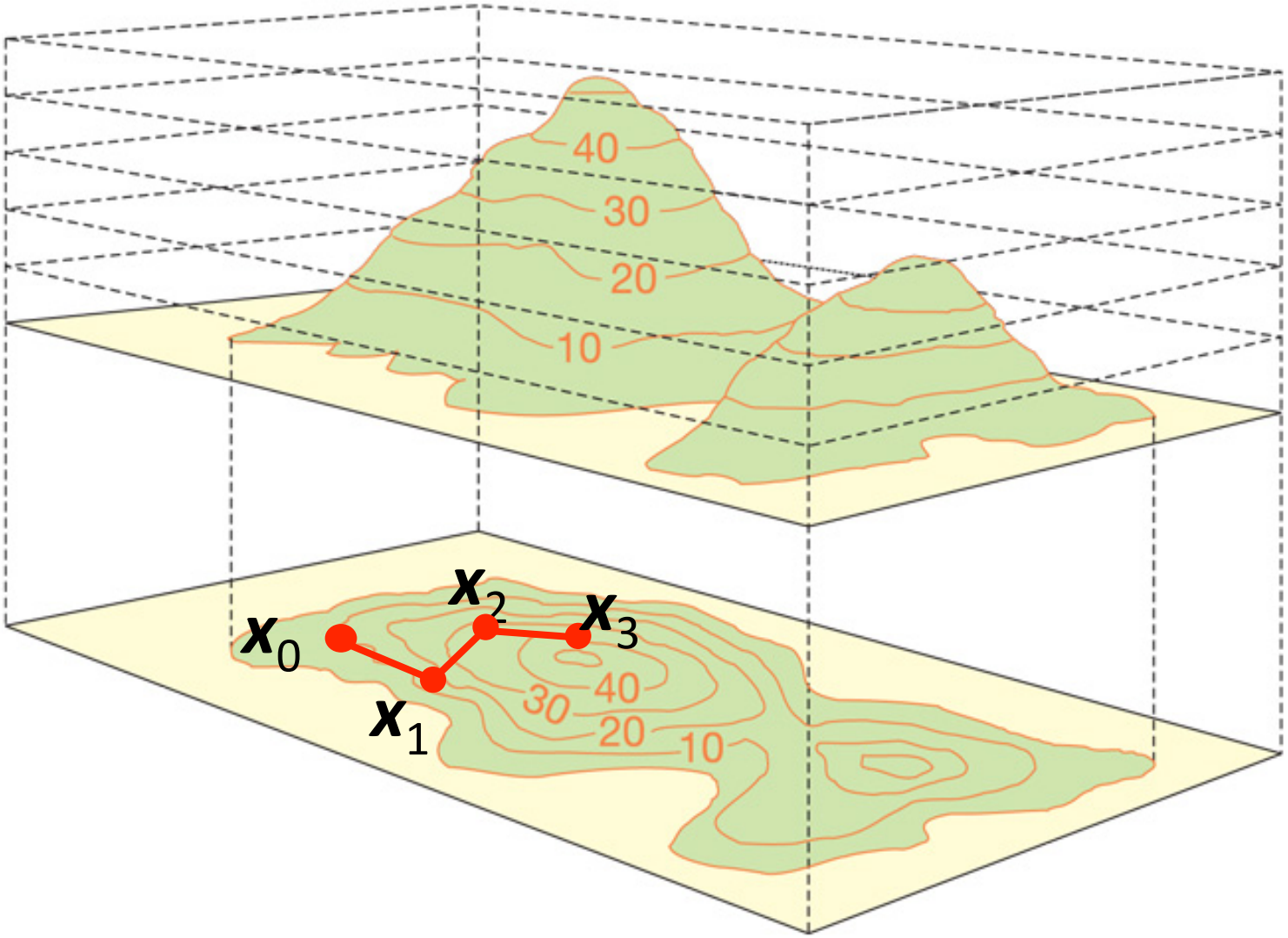
Optimization with Big Data

= Extreme* Mountain Climbing

* in a billion dimensional space on a foggy day

God's Algorithm = Teleportation

Mortals Have to Walk…



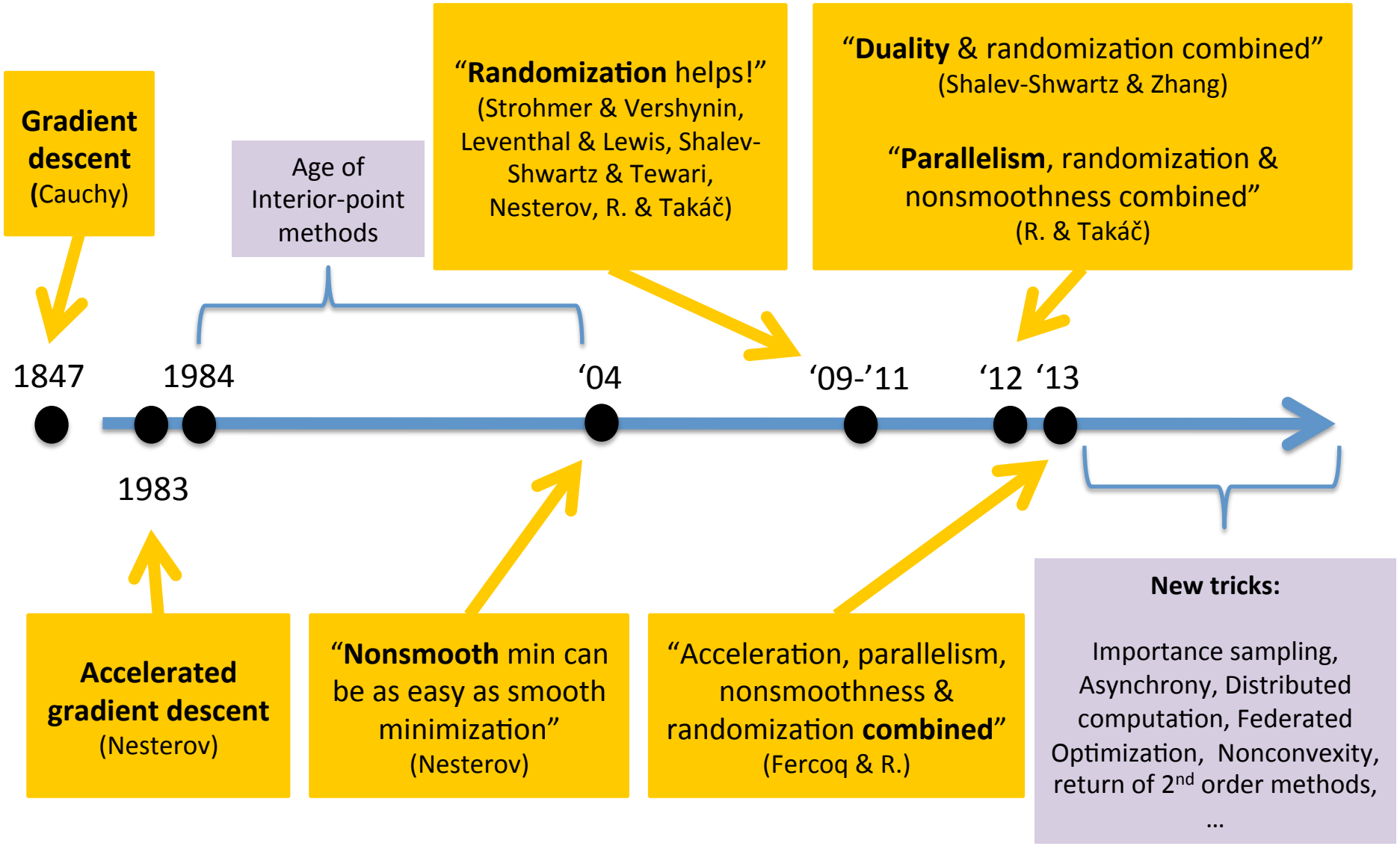
Algorithmic Tools

1. Gradient descent
2. Handling nonsmoothness via the proximal trick
3. Acceleration
4. Randomized decomposition
5. Parallelism / minibatching & sparsity

All these tools can be combined!

More tools: importance sampling, asynchrony, curvature, variance reduction, distributed sampling, …

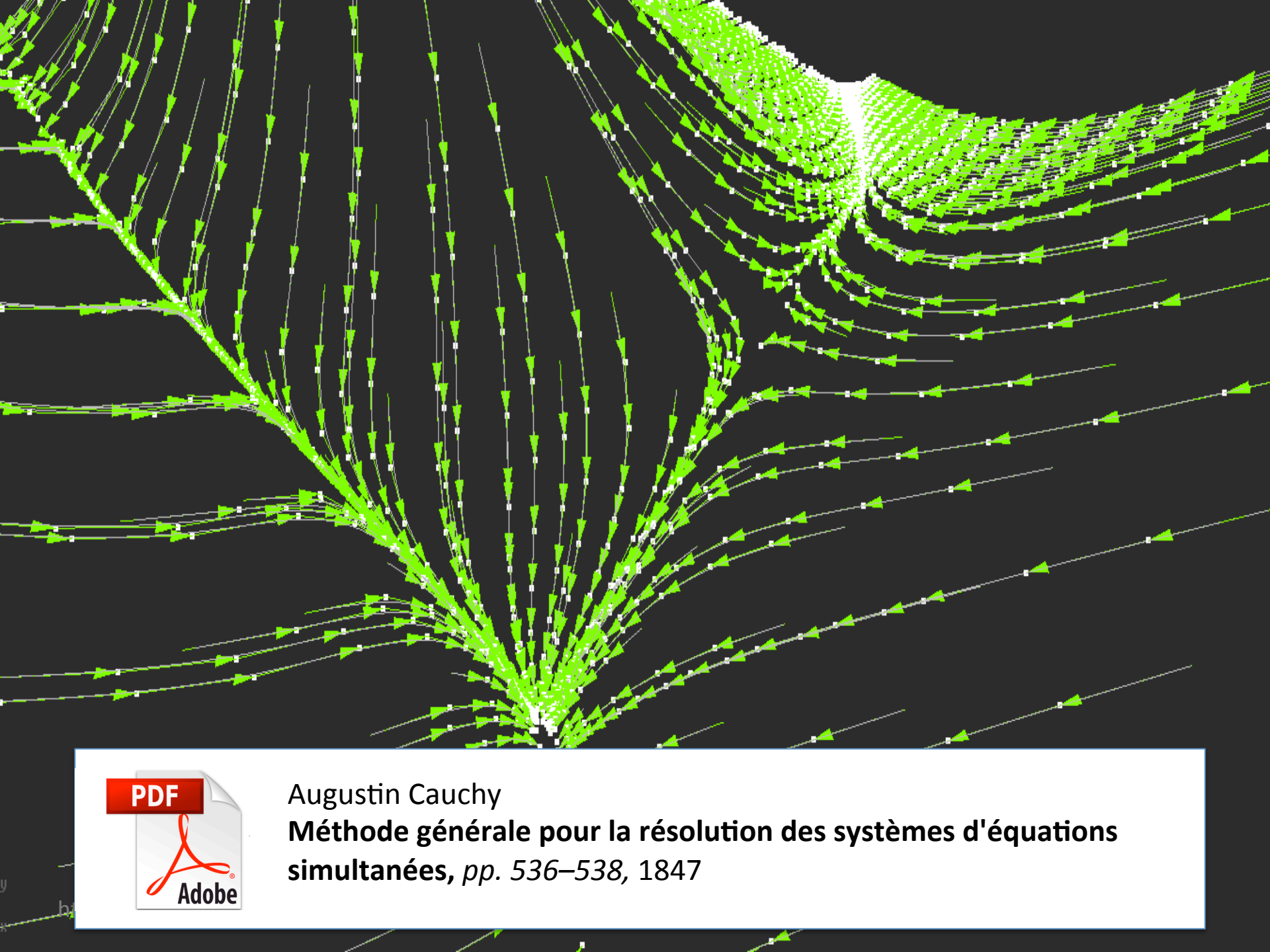
Brief, Biased and Severely Incomplete History of Big Data Optimization



Tool 1

Gradient Descent (1847)

“Just follow a ball rolling down the hill”



Augustin Cauchy

Méthode générale pour la résolution des systèmes d'équations simultanées, *pp. 536–538,* 1847

The Problem

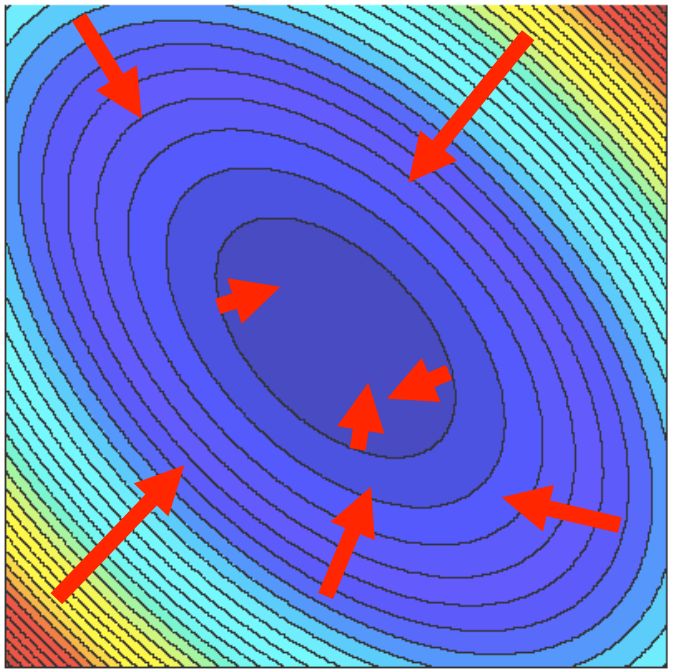
$$\min_{x \in \mathbb{R}^d} F(x)$$

L-smooth, μ-strongly convex

$$F(x) + \langle \nabla F(x), h \rangle + \tfrac{\mu}{2}\|h\|^2 \leq F(x+h) \leq F(x) + \langle \nabla F(x), h \rangle + \tfrac{L}{2}\|h\|^2$$

Gradient Descent (GD)

$$x_{k+1} = x_k - \frac{1}{L}\nabla F(x_k)$$



iterations

condition number of *F*

$$k \geq \frac{L}{\mu}\log(1/\epsilon)$$

$$F(x_k) - F(x_*) \leq \epsilon$$

Tool 2

Acceleration (1983/2003)

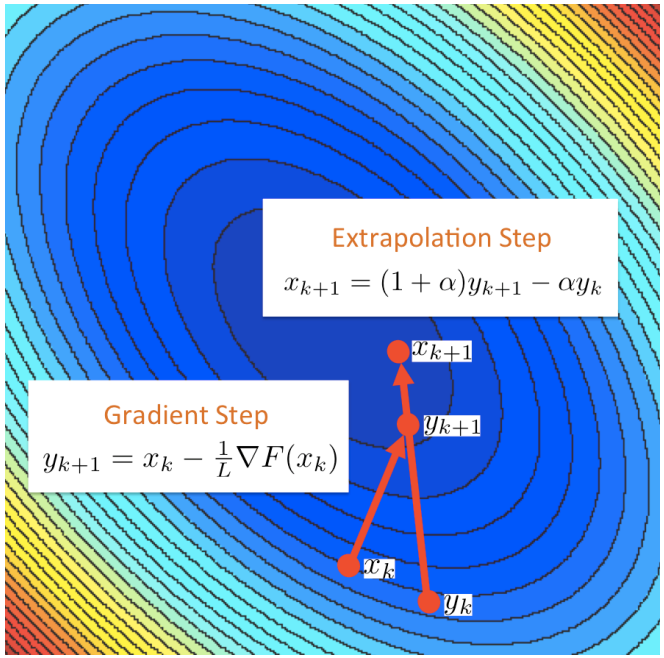
"Gradient descent can be made much faster!"

Accelerated Gradient Descent (AGD)

Gradient step: $y_{k+1} = x_k - \frac{1}{L}\nabla F(x_k)$

Extrapolation: $x_{k+1} = (1+\alpha)y_{k+1} - \alpha y_k$

$\alpha = \frac{\sqrt{L/\mu}-1}{\sqrt{L/\mu}+1}$

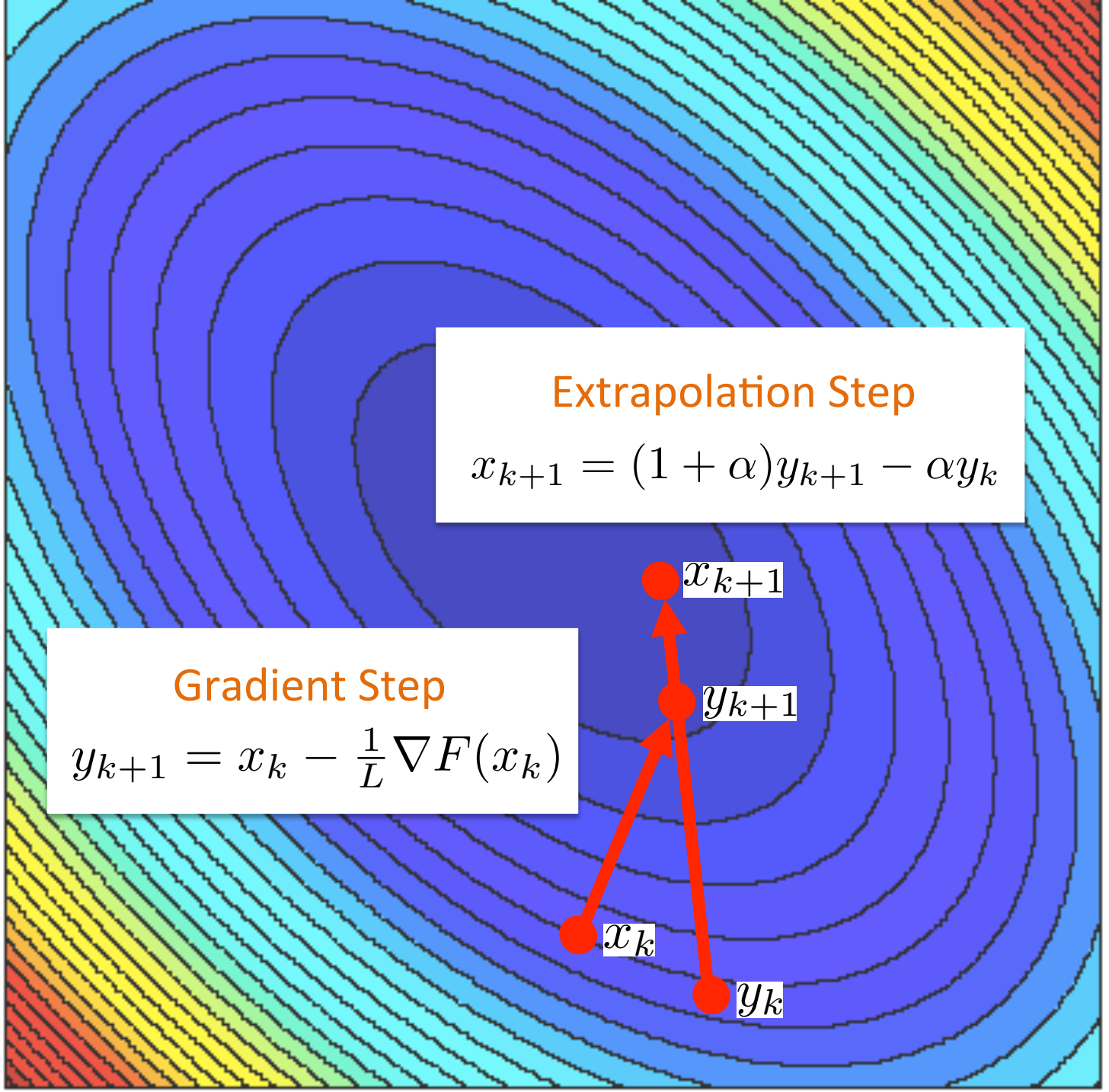


iterations

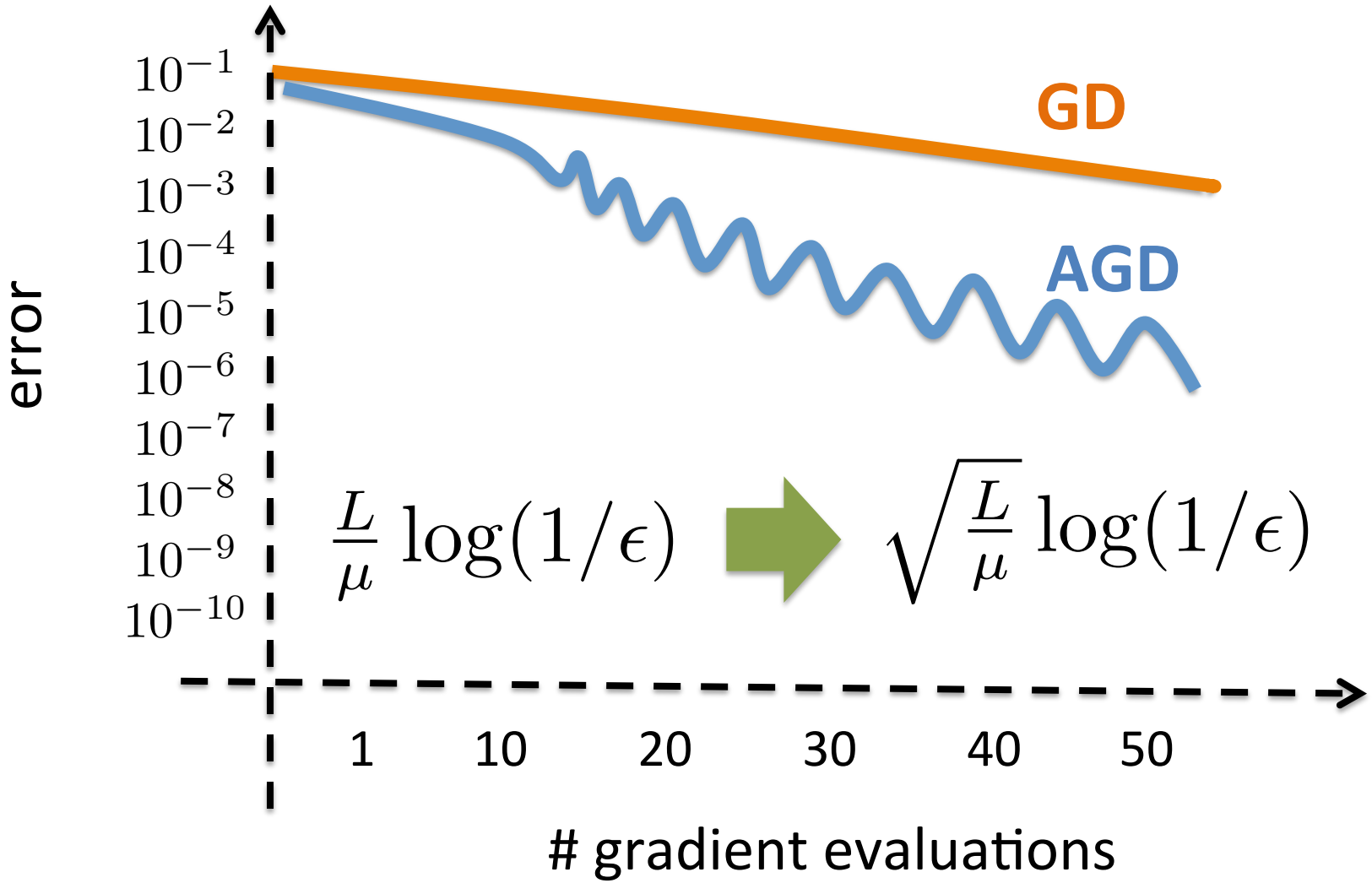
Square root of the condition number of *F*

$$k \geq \sqrt{\frac{L}{\mu}} \cdot \log(c/\epsilon)$$

$$F(x_k) - F(x_*) \leq \epsilon$$


Extrapolation Step
$x_{k+1} = (1+\alpha)y_{k+1} - \alpha y_k$
Gradient Step
$y_{k+1} = x_k - \frac{1}{L}\nabla F(x_k)$
x_{k+1}
y_{k+1}
x_k
y_k

Acceleration Works (Somewhat Mysteriously)



Acceleration and ODEs

ODE for Gradient Descent

$$\dot{X}(t) + \nabla f(X(t)) = 0$$

ODE for Accelerated Gradient Descent

$$\ddot{X}(t) + \frac{3}{t}\dot{X}(t) + \nabla f(X(t)) = 0$$

Weijie Su, Stephen Boyd and Emmanuel J. Candes
A Differential Equation for Modeling Nesterov's Accelerated Gradient Method: Theory and Insights
NIPS, 2014

Acceleration

- Reignited interest in gradient methods
- Called momentum in deep neural networks literature
- Oscillation can be tamed (e.g., by restarting)

Yurii Nesterov
Introductory Lectures on Convex Optimization: a Basic Course
Kluwer, Boston, 2003

Yurii Nesterov
A Method for Unconstrained Convex Minimization Problem with the Rate of Convergence O(1 / k^2)
Soviet Math. Doklady 269, 543-547, 1983

Tool 3

Proximal Trick (2004)

"Some nonsmooth problems are as easy as smooth problems"

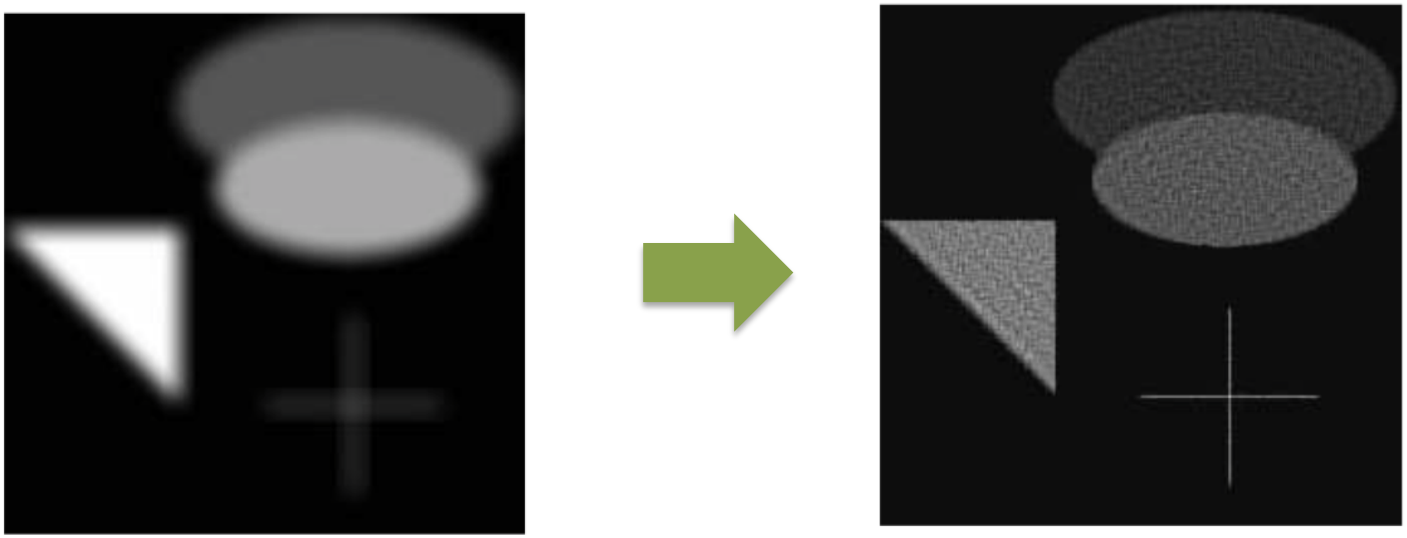
The Problem

$$\min_{x \in \mathbb{R}^d} F(x) + G(x)$$

L-smooth, μ-strongly convex

Convex,
but can be
nonsmooth

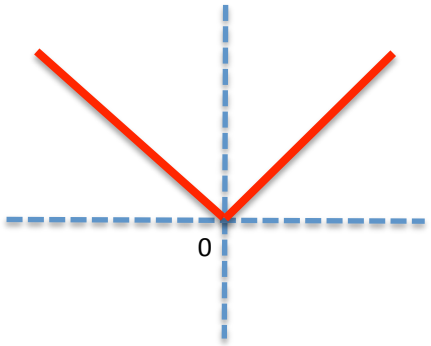
Image Deblurring



Amir Beck and Marc Teboulle. **A Fast Iterative Shrinking-Thresholding Algorithm for Linear Inverse Problems.** *SIAM J. Imaging Sciences* 2(1), 183-202, 2009

Jakub Konečný, Jie Liu, P.R., Martin Takáč. **Mini-Batch Semi-Stochastic Gradient Descent in the Proximal Setting.** *IEEE Journal of Selected Topics in Signal Processing* 10(2), 242-255, 2016

Image Deblurring: "LASSO" Problem



blurred image

$$\min_{x \in \mathbb{R}^n} \frac{1}{2}\|Ax - b\|_2^2 + \lambda\|x\|_1$$

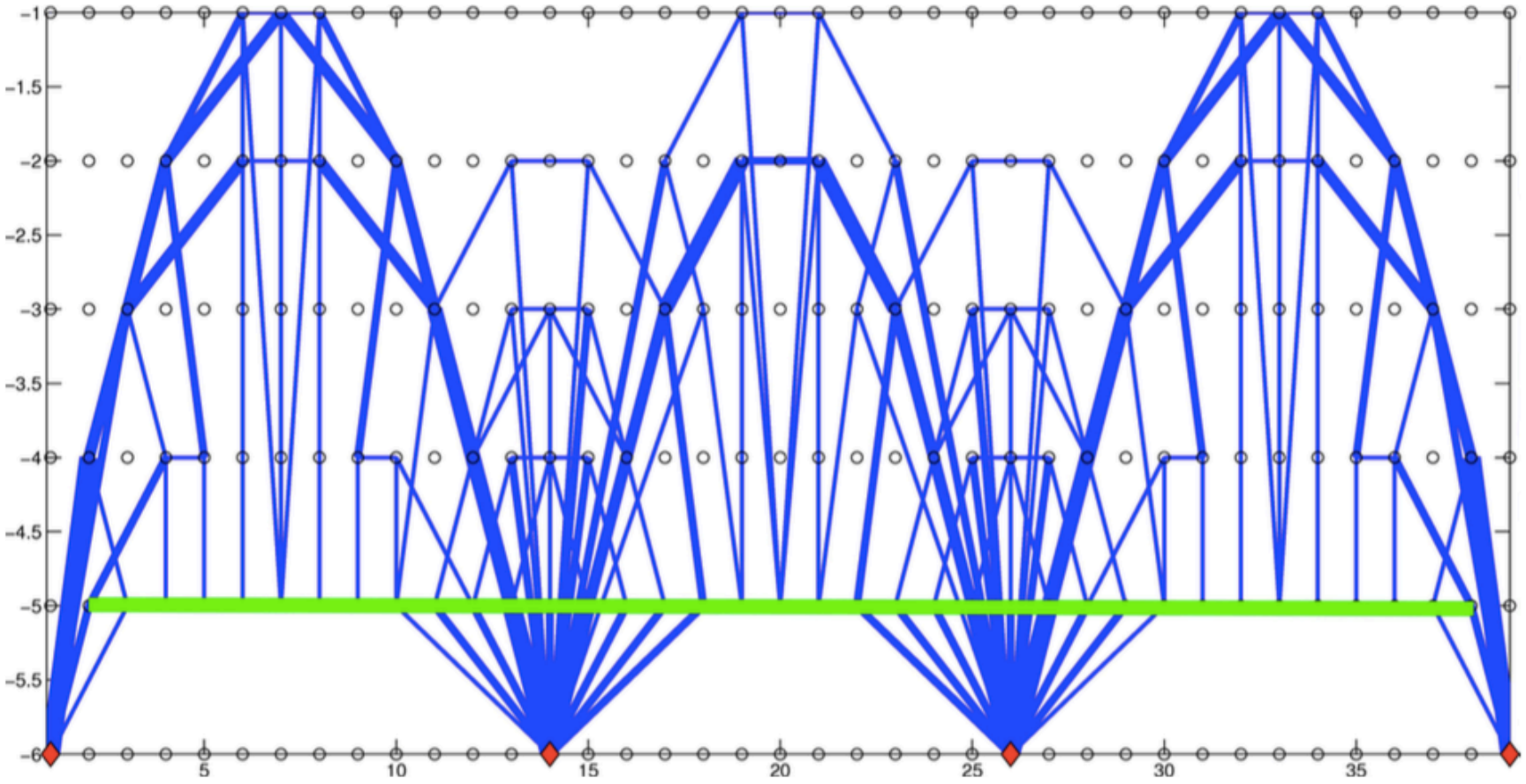
image

pixels in the image

Blurring matrix multiplied by a wavelet basis matrix

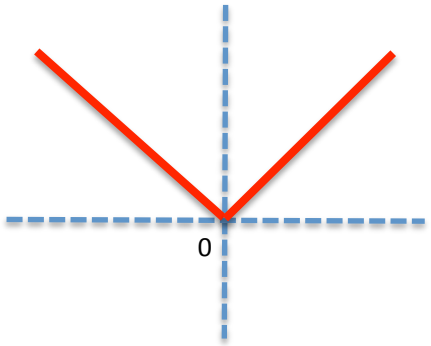
Encourages sparsity in the wavelet basis

Truss Topology Design



P.R. and Martin Takáč. **Efficient Serial and Parallel Coordinate Descent Methods for Huge-Scale Truss Topology Design.** *Operations Research Proceedings*, pp 27-32, 2012

Truss Topology Design: "LASSO" Problem



Encodes all potential bars

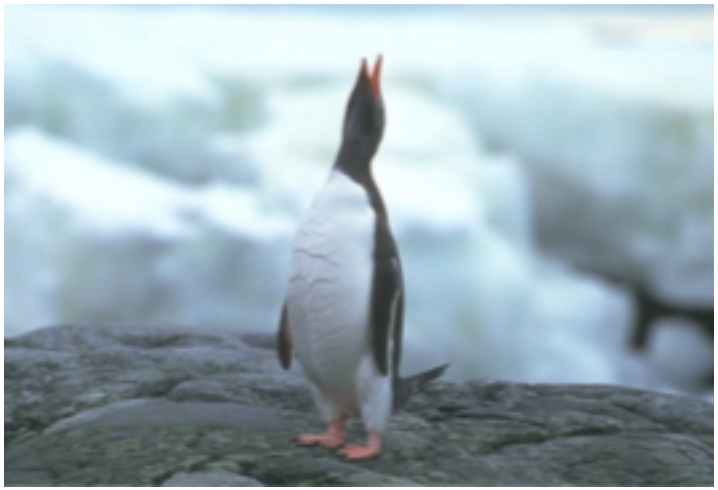
$$\min_{x \in \mathbb{R}^n} \frac{1}{2}\|Ax - b\|_2^2 + \lambda \|x\|_1$$

potential bars (quadratic in mesh size)

Least-squares (convex, smooth, quadratic)

L1 norm (convex, nonsmooth, but "simple")

Image Segmentation



Olivier Fercoq and P.R. **Accelerated, Parallel and Proximal Coordinate Descent.** *SIAM Journal on Optimization* 25(4), 1997-2023, 2015

Alina Ene and Huy L. Nguyen. **Random Coordinate Descent Methods for Minimizing Decomposable Submodular Functions.** *ICML* 2015

Image Segmentation: Reformulated Submodular Optimization

$$\text{minimize} \quad \frac{1}{2}\left\|\sum_{i=1}^{n} x^i\right\|^2$$

Smooth, convex, quadratic

$$\text{subject to} \quad x^i \in P^i,\ i = 1, 2, \ldots, n$$

polytope

grows with the image size

Proximal Gradient Descent (PGD)

STEP 1: Pretend there is no *G*

$$z_{k+1} = x_k - \frac{1}{L}\nabla F(x_k)$$

STEP 2: Take a "proximal" step with respect to *G*

$$x_{k+1} = \arg\min_x \frac{1}{2}\|x - z_{k+1}\|^2 + \frac{1}{L}G(x)$$

1. Gradient Descent is a special case for $G = 0$

2. Even though this is a nonsmooth problem,
   # steps is the same as for Gradient Descent!!!

$$\frac{L}{\mu}\log(1/\epsilon)$$

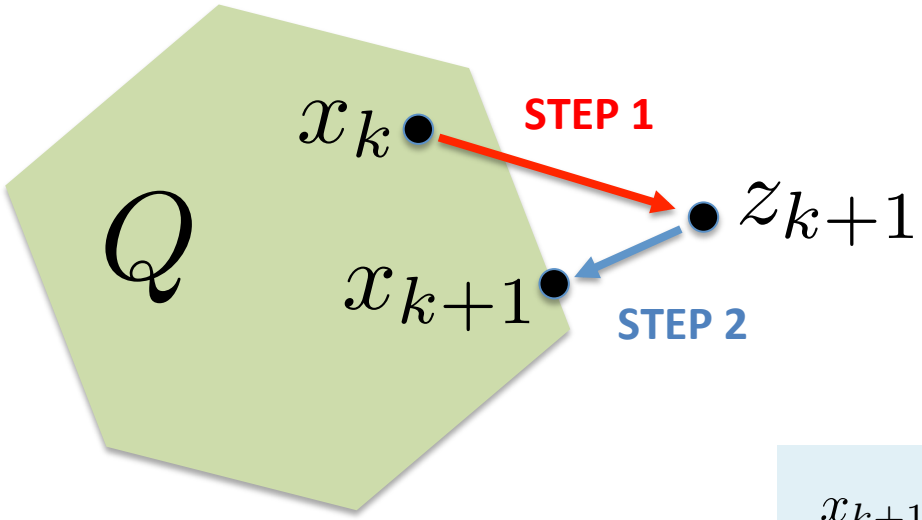
3. Efficient if Step 2 is easy to do

Example: Projected Gradient Descent

$$\min_{x \in Q} F(x) \quad \Leftrightarrow \quad \min_x F(x) + G(x)$$

Convex set

$$G(x) = \begin{cases} 0 & x \in Q \\ +\infty & x \notin Q \end{cases}$$



$$z_{k+1} = x_k - \tfrac{1}{L} \nabla F(x_k)$$

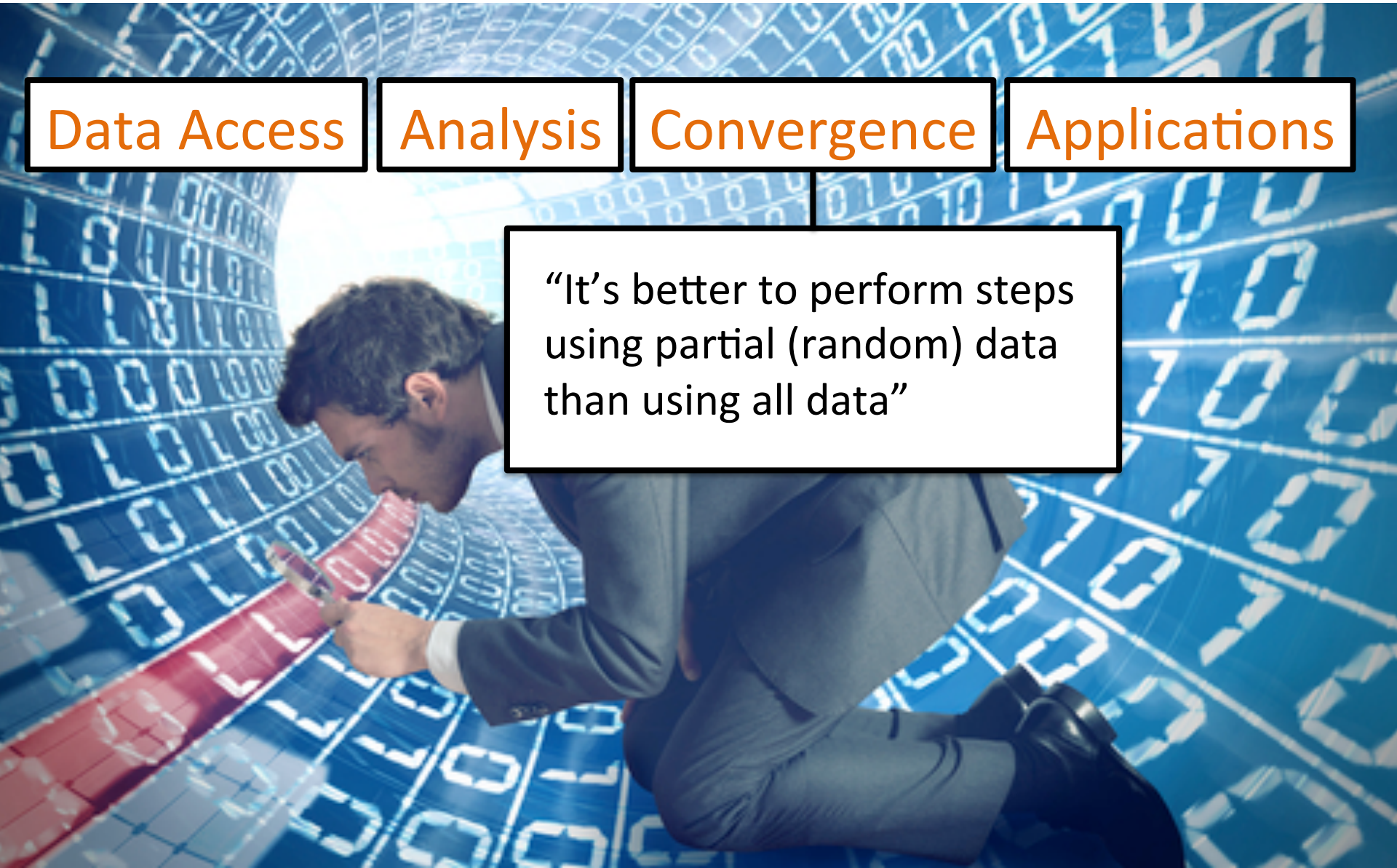
$$x_{k+1} = \arg\min_x \tfrac{1}{2} \|x - z_{k+1}\|^2 + \tfrac{1}{L} G(x)$$

Tool 4

Randomized Decomposition

"Doing many simple decisions is better than doing a few smart ones"

Why Randomize?



Primal ERM Problem: Stochastic Gradient Descent

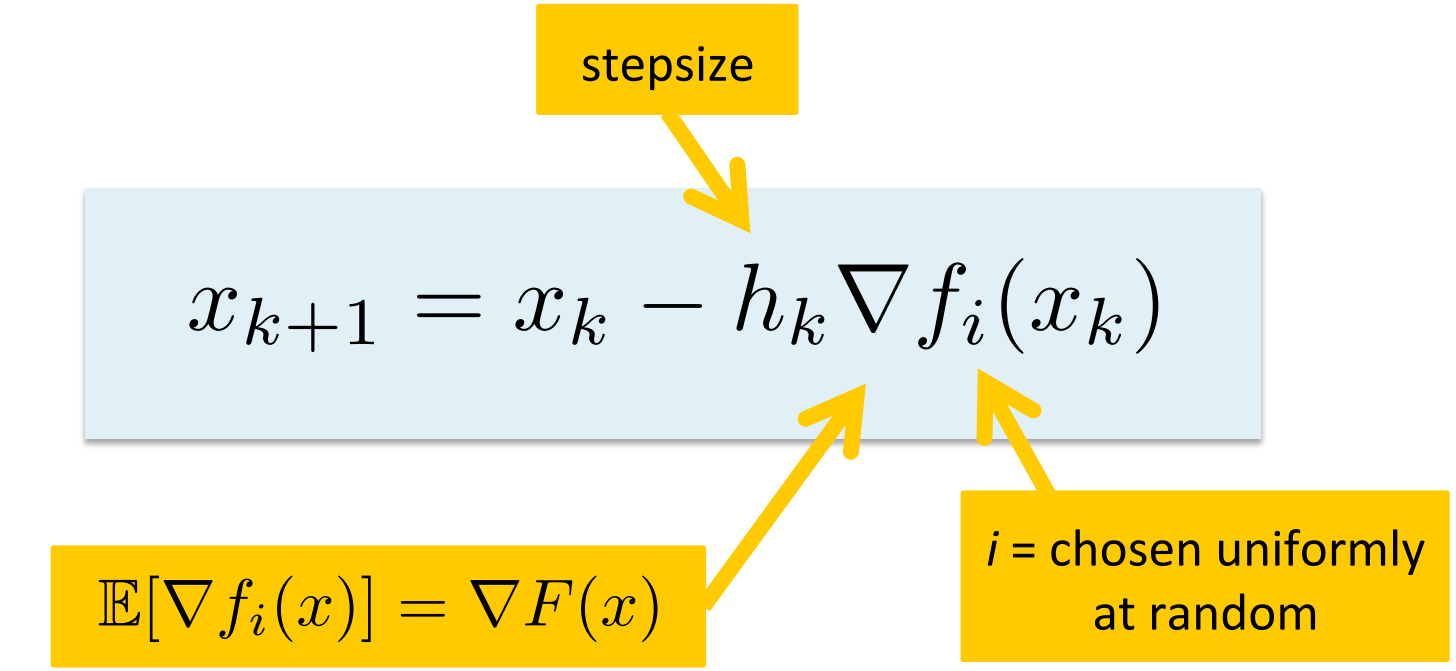
H. Robbins and S. Monro
A Stochastic Approximation Method
Annals of Mathematical Statistics 22, pp. 400–407, 1951

The Problem

n is big

$$\min_{x \in \mathbb{R}^d} \left\{ F(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x) \right\}$$

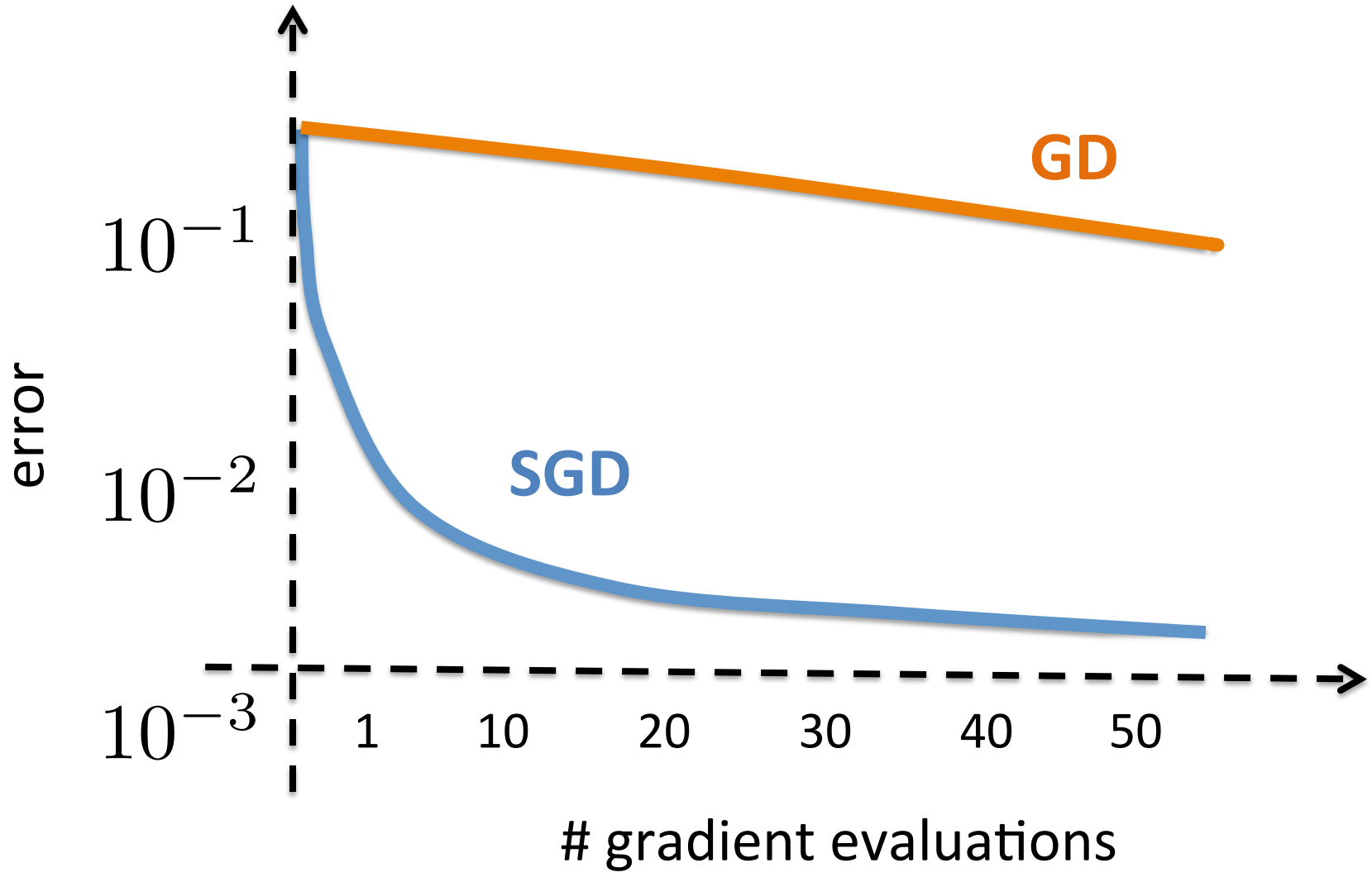
Stochastic Gradient Descent (SGD)



Unbiased estimate of the gradient

Each iteration *n* times cheaper than 1 iteration of GD

Stochastic Gradient Descent vs Gradient Descent



Dual ERM Problem: Randomized Coordinate Descent

Yurii Nesterov
Efficiency of Coordinate Descent Methods on Huge-Scale Optimization Problems
SIAM Journal on Optimization, 22(2), 341–362, 2012

P.R. and Martin Takáč
Iteration Complexity of Randomized Block Coordinate Descent Methods for Minimizing a Composite Function
Mathematical Programming 144(2), 1-38, 2014 *(arXiv:1107.2848)*

INFORMS Computing Society Best Student Paper Prize (runner up), 2012

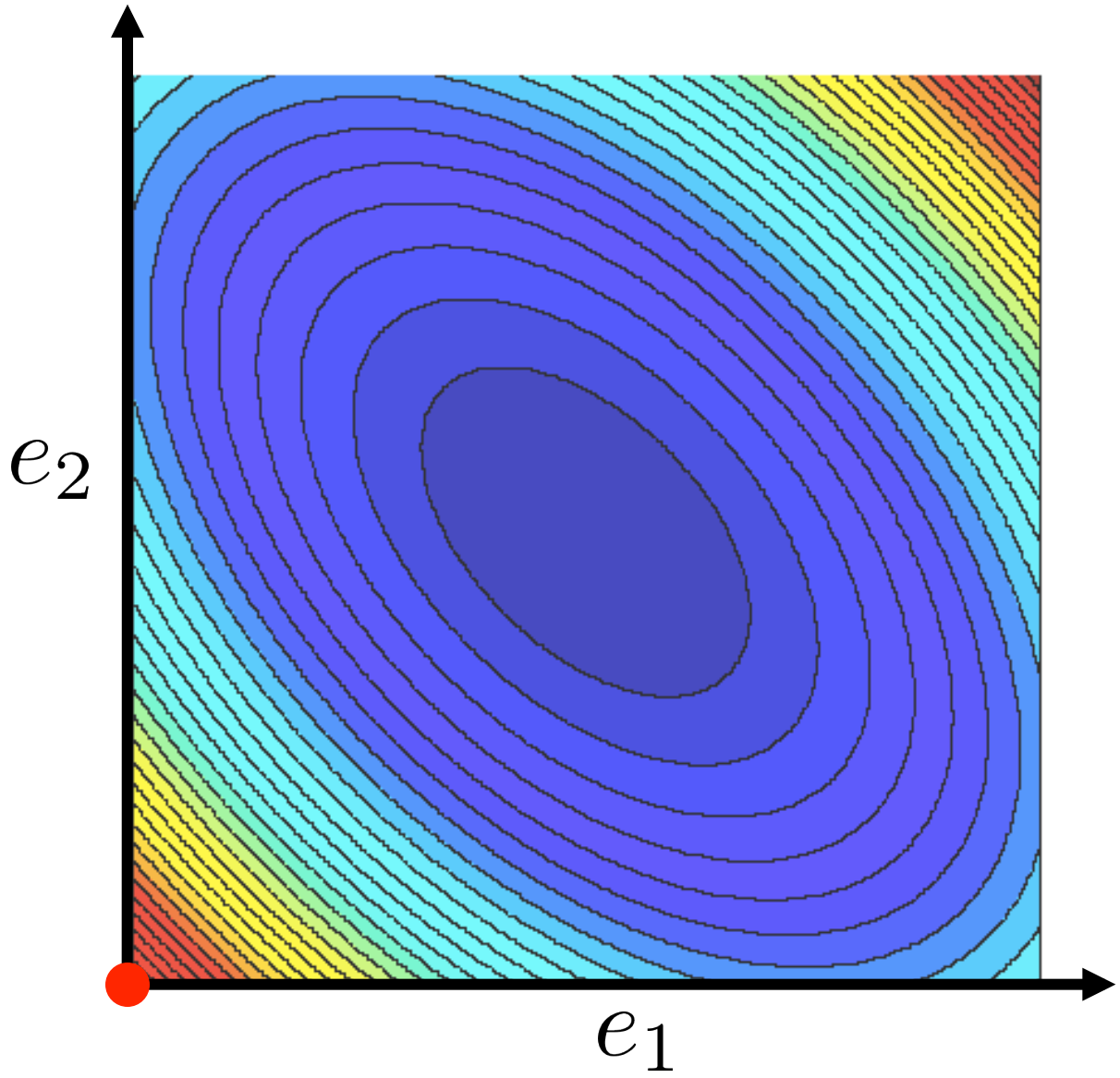
The Problem

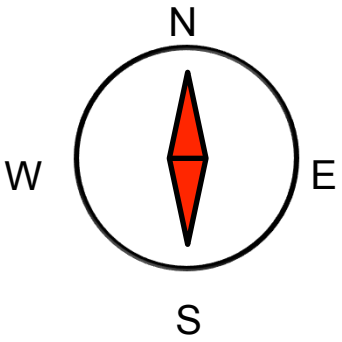
$$\min_{x \in \mathbb{R}^n} F(x)$$

n is BIG

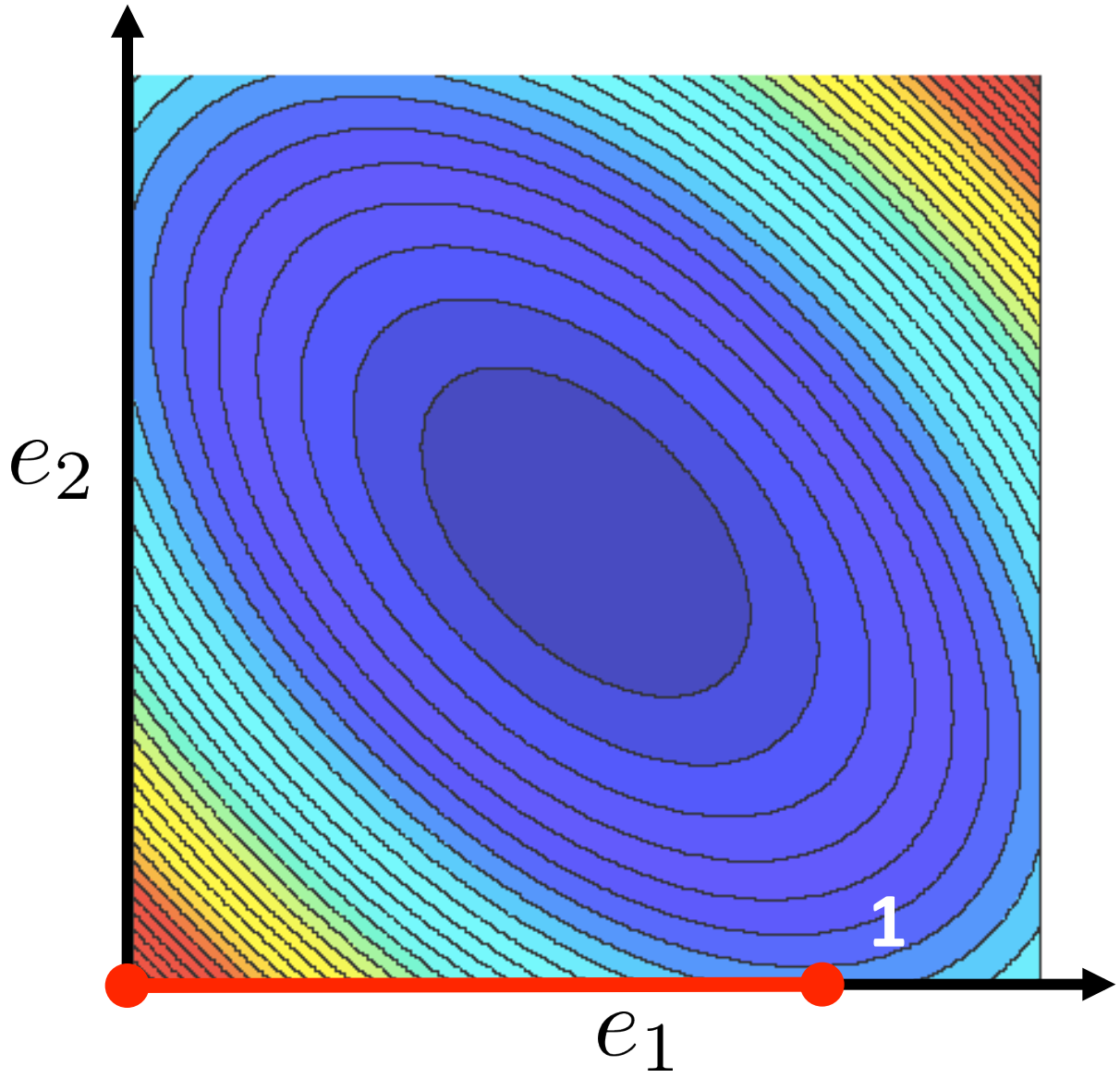
smooth, μ-strongly convex

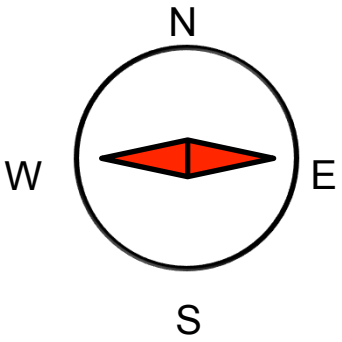
Randomized Coordinate Descent in 2D



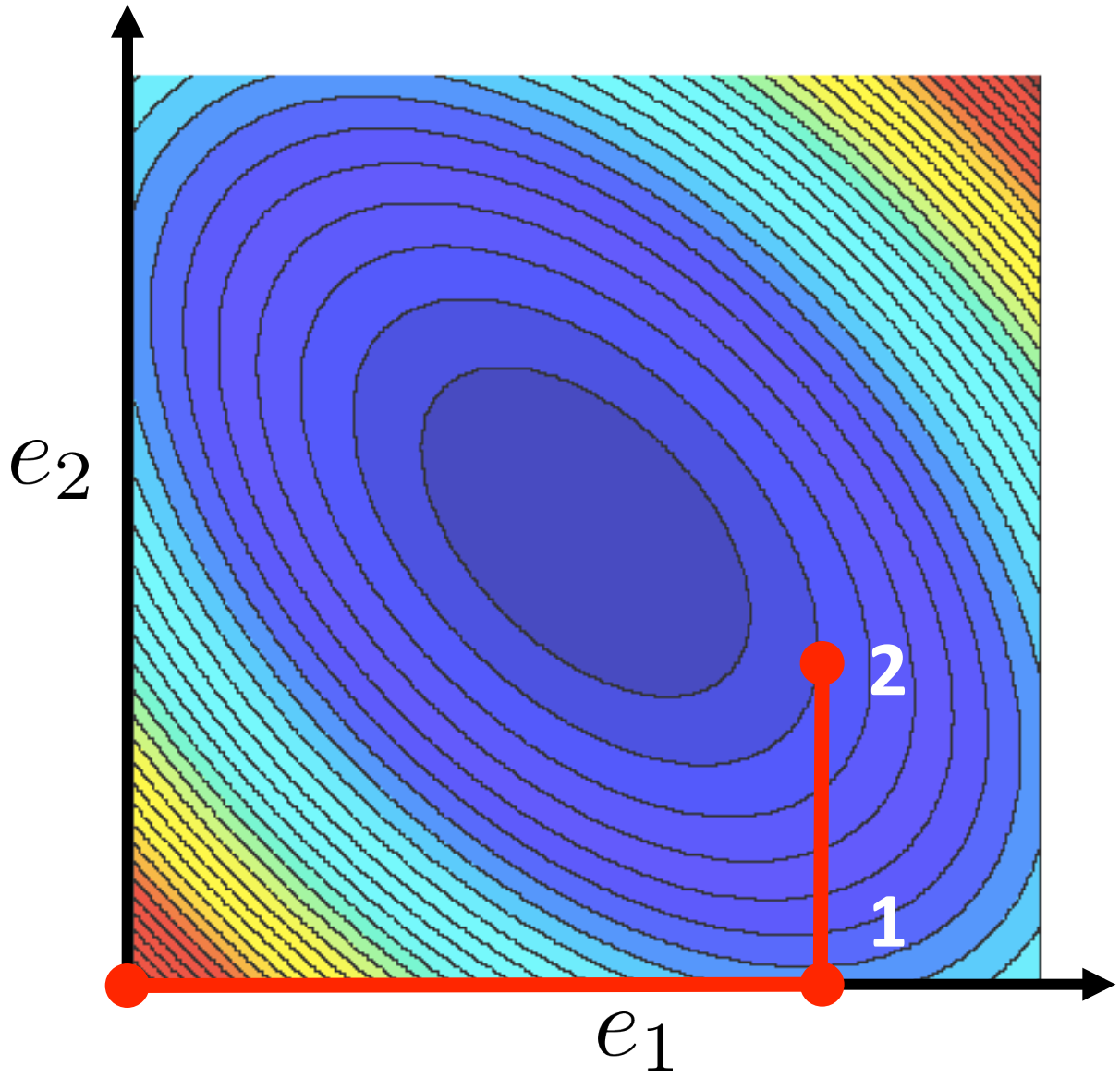


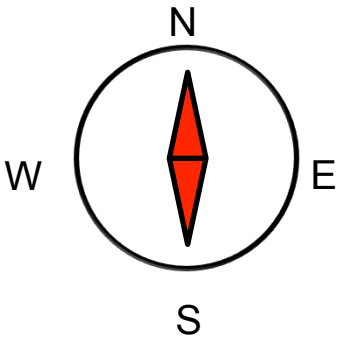
Randomized Coordinate Descent in 2D



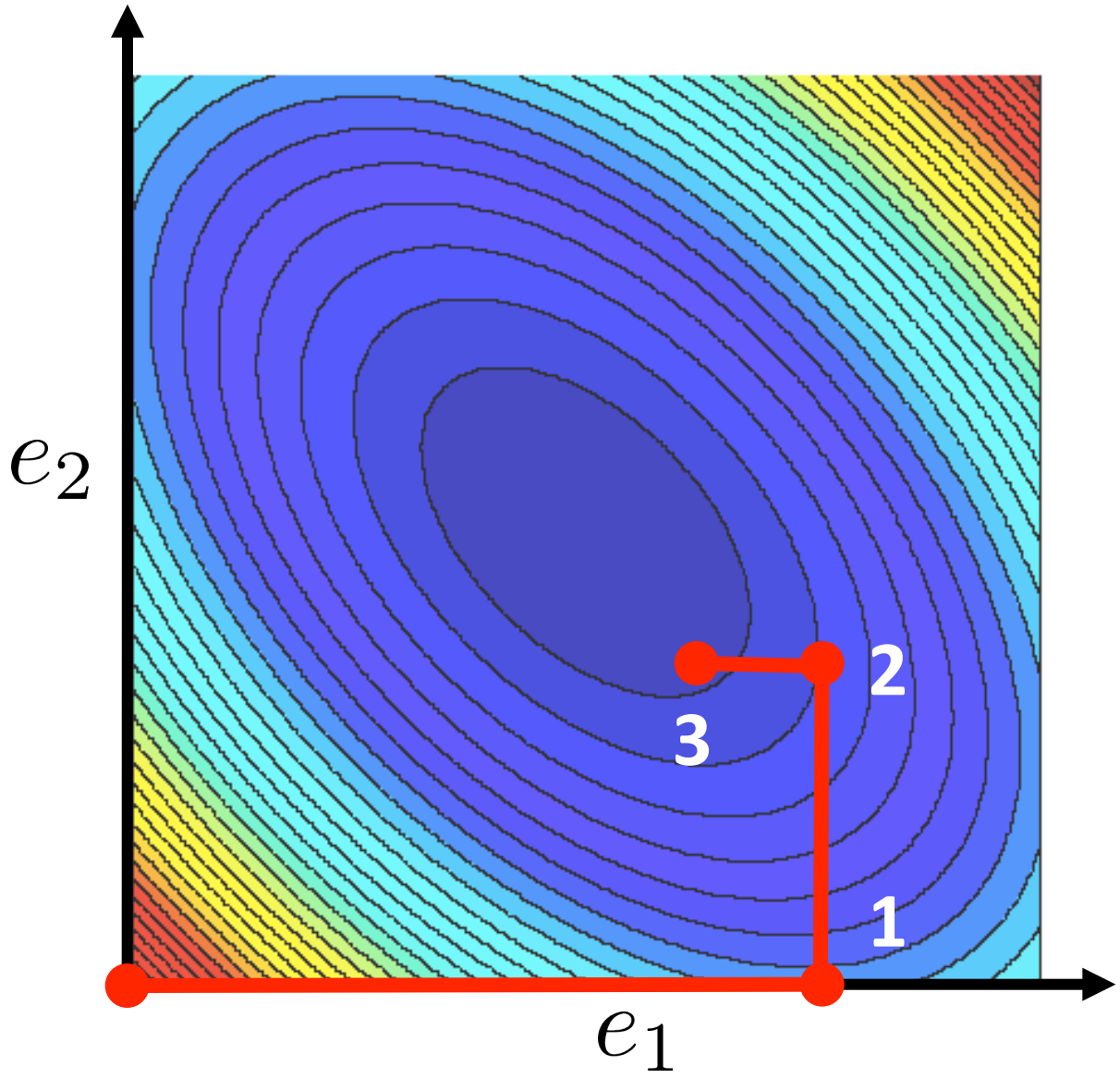


Randomized Coordinate Descent in 2D

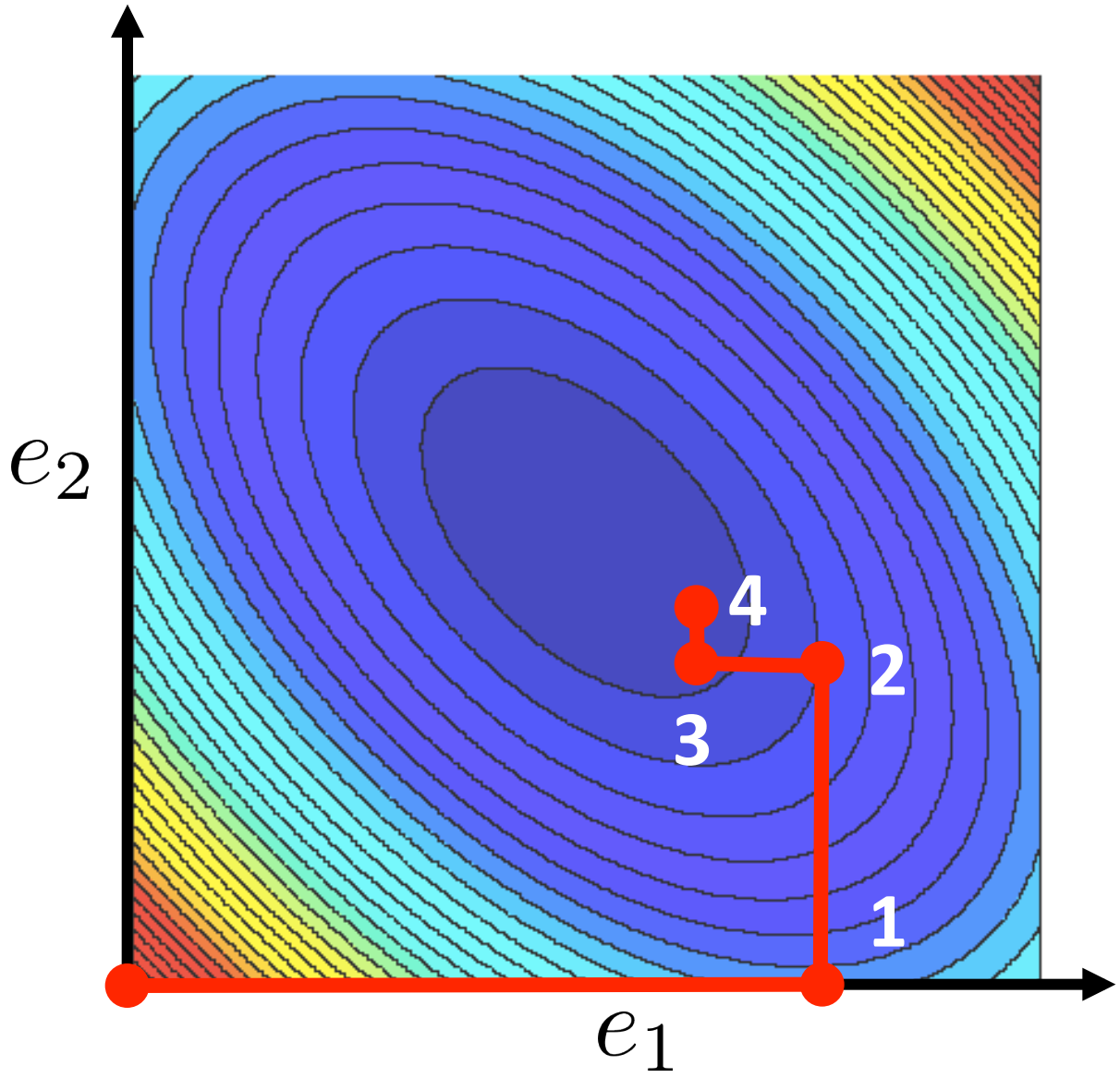




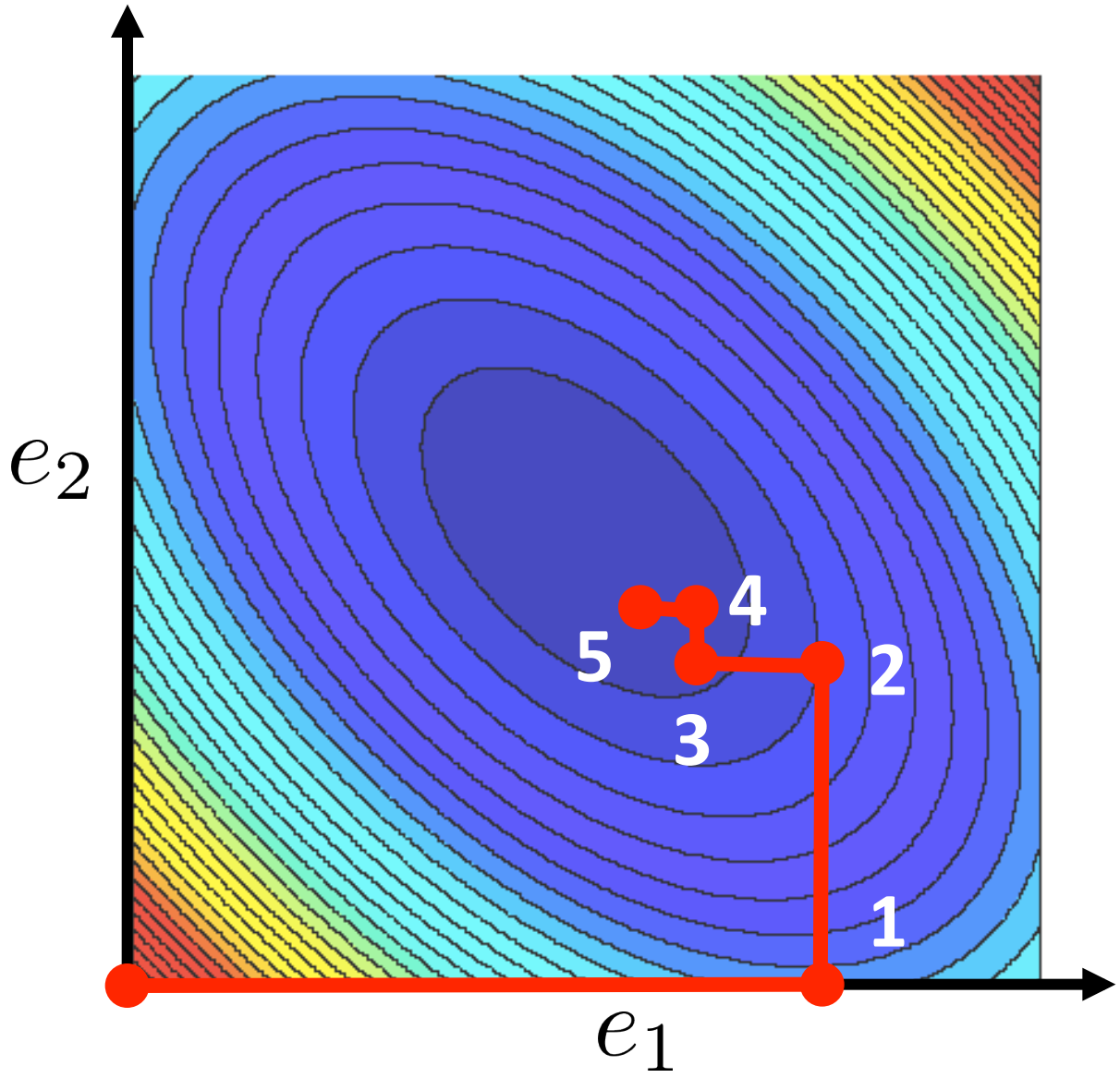
Randomized Coordinate Descent in 2D



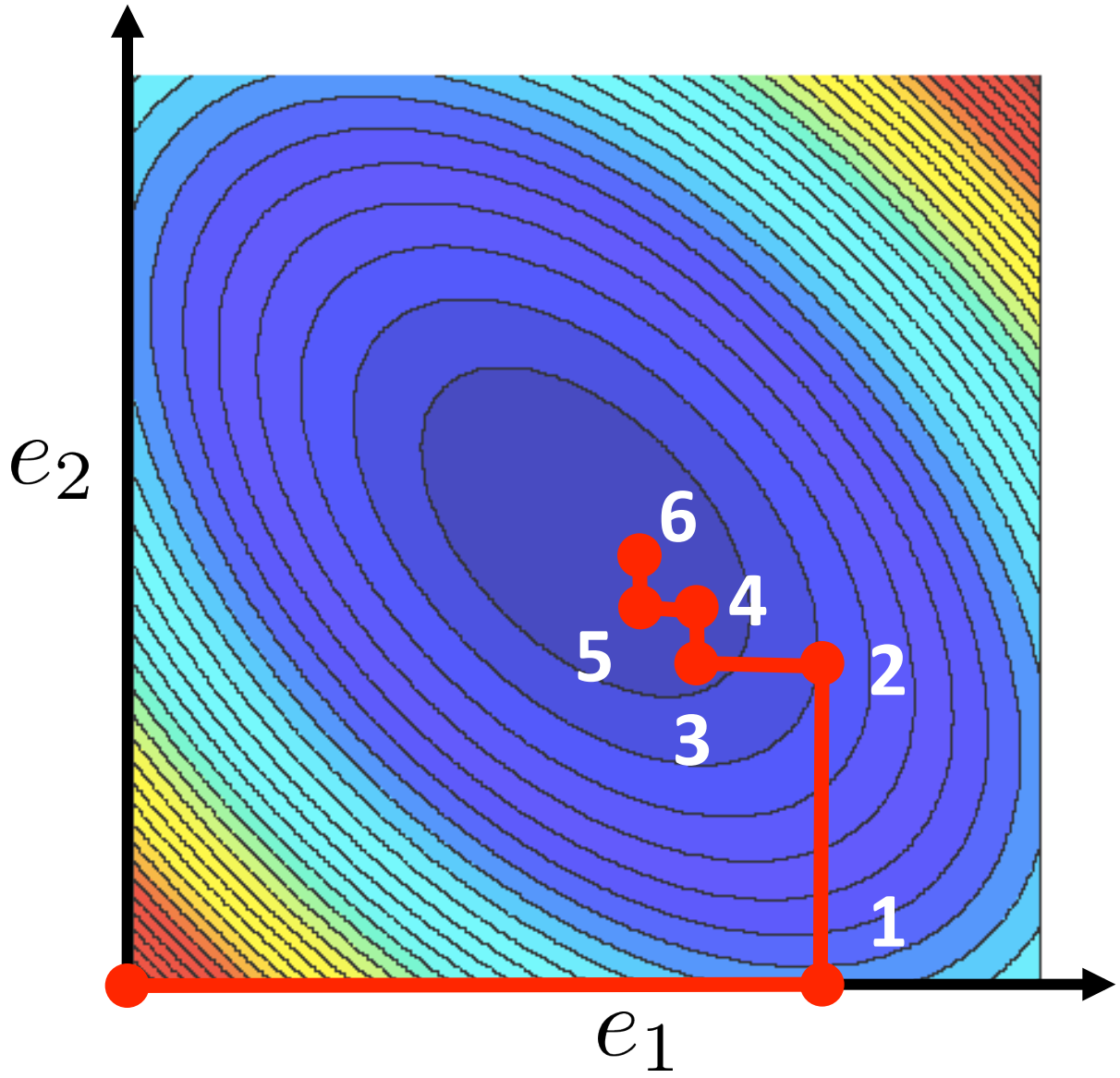
Randomized Coordinate Descent in 2D

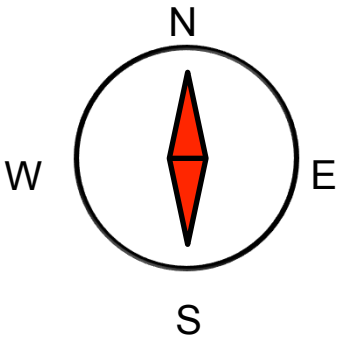


Randomized Coordinate Descent in 2D

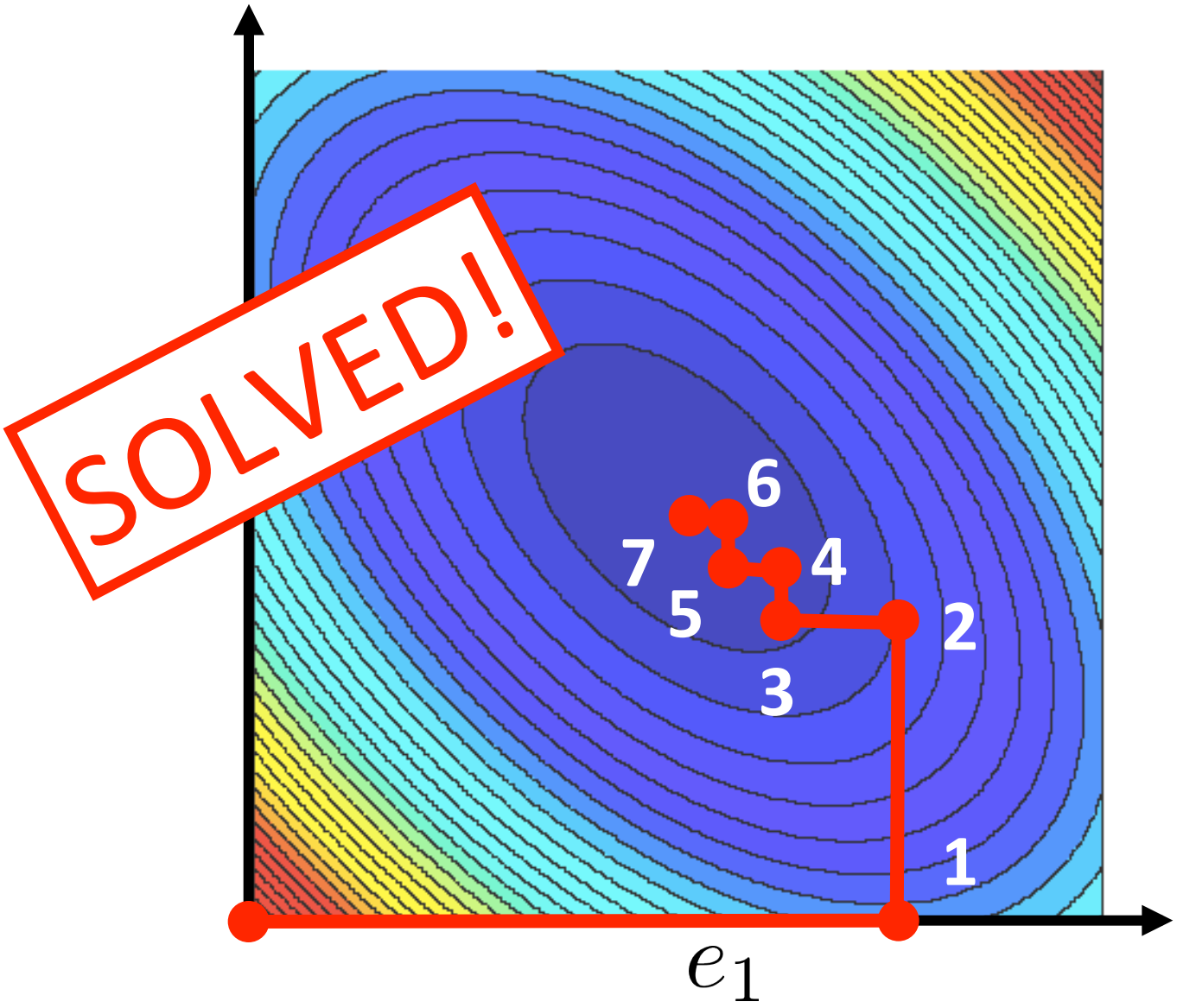


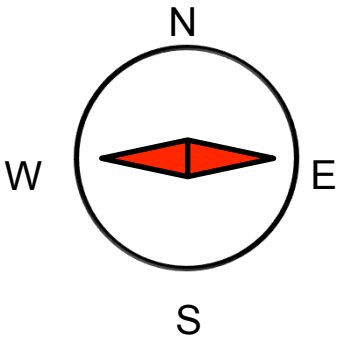
Randomized Coordinate Descent in 2D





Randomized Coordinate Descent in 2D





Randomized Coordinate Descent

Partial derivative of *F*

i^{th} standard unit basis vector in $\mathbb{R}^n$

$$x_{k+1} = x_k - \frac{1}{L_i} \nabla_i F(x_k) e_i$$

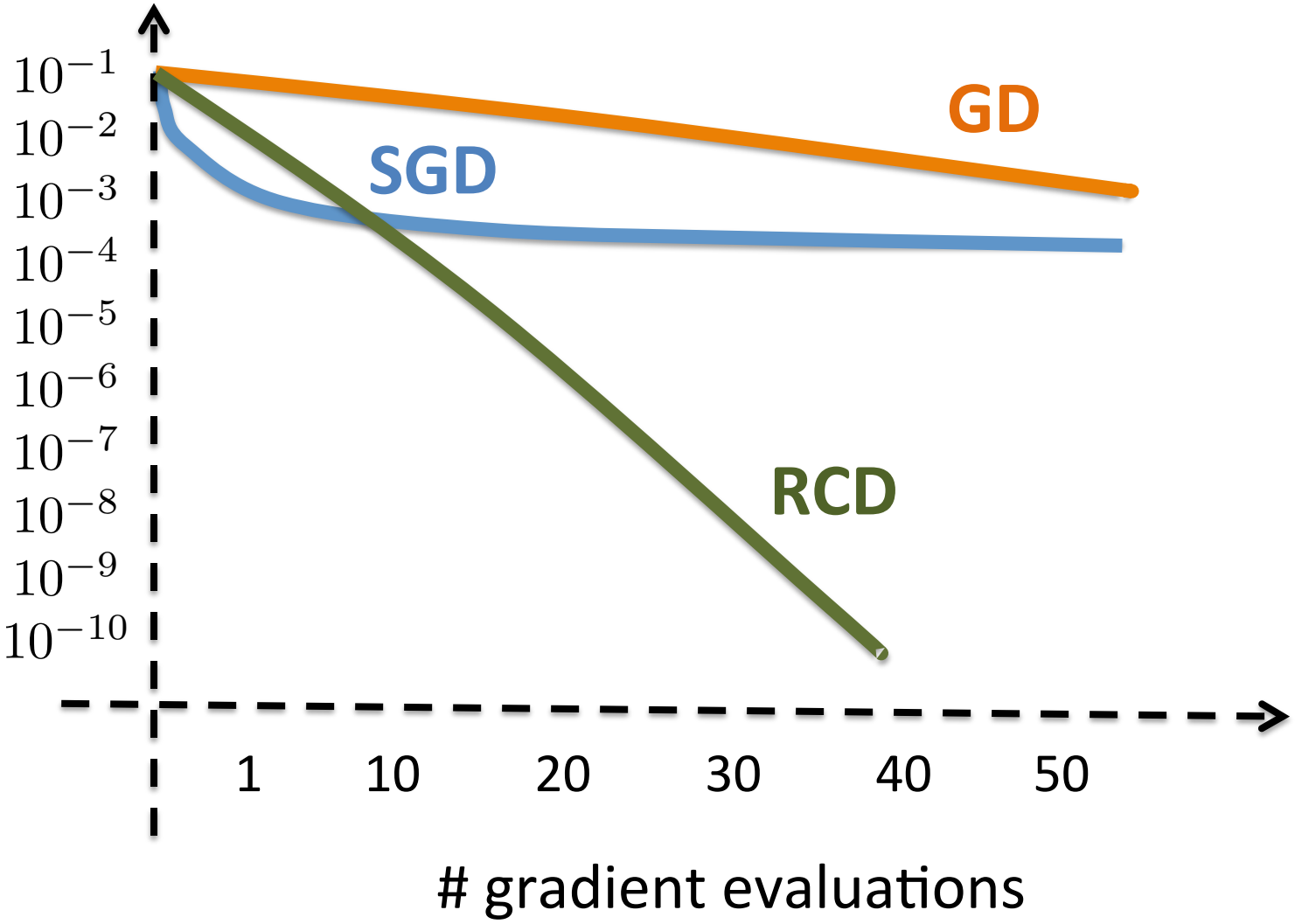
F is L_i-smooth along e_i

$$|\nabla_i F(x + te_i) - \nabla_i F(x)| \leq L_i |t|$$

Often, each iteration is *n* times cheaper.
However, complexity is not *n* times worse!
So, RCD is better than GD!

$$k \geq \left(\frac{\max_i L_i}{\mu} \right) \log(c/\epsilon) \quad \Rightarrow \quad \mathbf{E}[F(x_k) - F(x_*)] \leq \epsilon$$

SGD vs GD vs RCD



1 Billion Rows & 100 Million Variables

$$A \in \mathbf{R}^{10^9 \times 10^8}$$

k/n	$F(x_k) - F^*$	# nonzeros in x_k	time [s]
0.01	$< 10^{18}$	18,486	1.32
9.35	$< 10^{14}$	99,837,255	1294.72
11.97	$< 10^{13}$	99,567,891	1657.32
14.78	$< 10^{12}$	98,630,735	2045.53
17.12	$< 10^{11}$	96,305,090	2370.07
20.09	$< 10^{10}$	86,242,708	2781.11
22.60	$< 10^{9}$	58,157,883	3128.49
24.97	$< 10^{8}$	19,926,459	3455.80
28.62	$< 10^{7}$	747,104	3960.96
31.47	$< 10^{6}$	266,180	4325.60
34.47	$< 10^{5}$	175,981	4693.44
36.84	$< 10^{4}$	163,297	5004.24
39.39	$< 10^{3}$	160,516	5347.71
41.08	$< 10^{2}$	160,138	5577.22
43.88	$< 10^{1}$	160,011	5941.72
45.94	$< 10^{0}$	160,002	6218.82
46.19	$< 10^{-1}$	160,001	6252.20
46.25	$< 10^{-2}$	160,000	6260.20
46.89	$< 10^{-3}$	160,000	6344.31
46.91	$< 10^{-4}$	160,000	6346.99
46.93	$< 10^{-5}$	160,000	6349.69

Tool 5

Parallelism

“Work on random subsets”

The Problem

$$\min_{x \in \mathbb{R}^n} F(x)$$

n is BIG

Convex, smooth

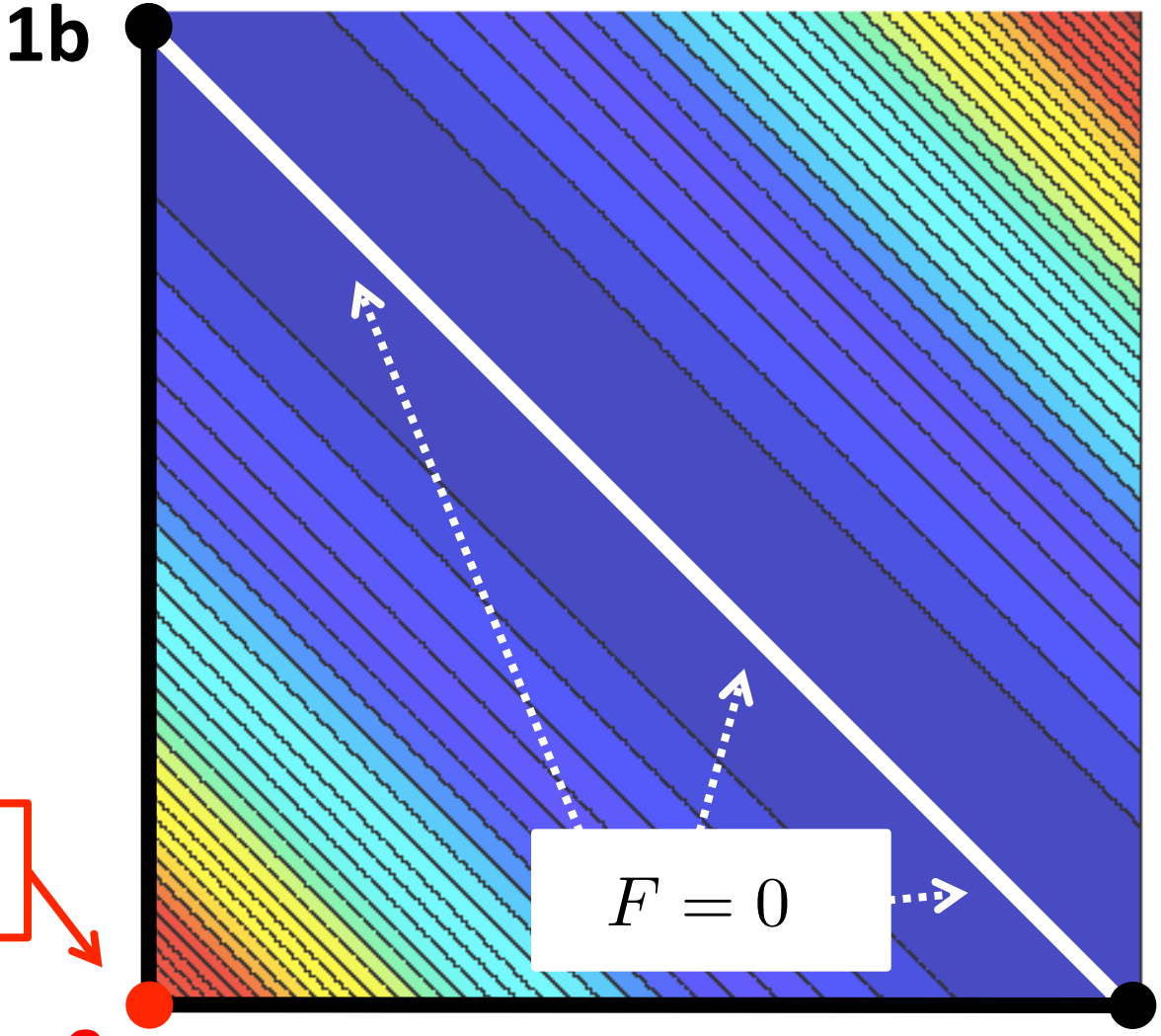
Parallel Randomized Coordinate Descent

P.R. and Martin Takáč
Parallel Coordinate Descent Methods for Big Data Optimization
Mathematical Programming 156(1), 433-484, 2016

16th IMA Leslie Fox Prize (2nd), 2013

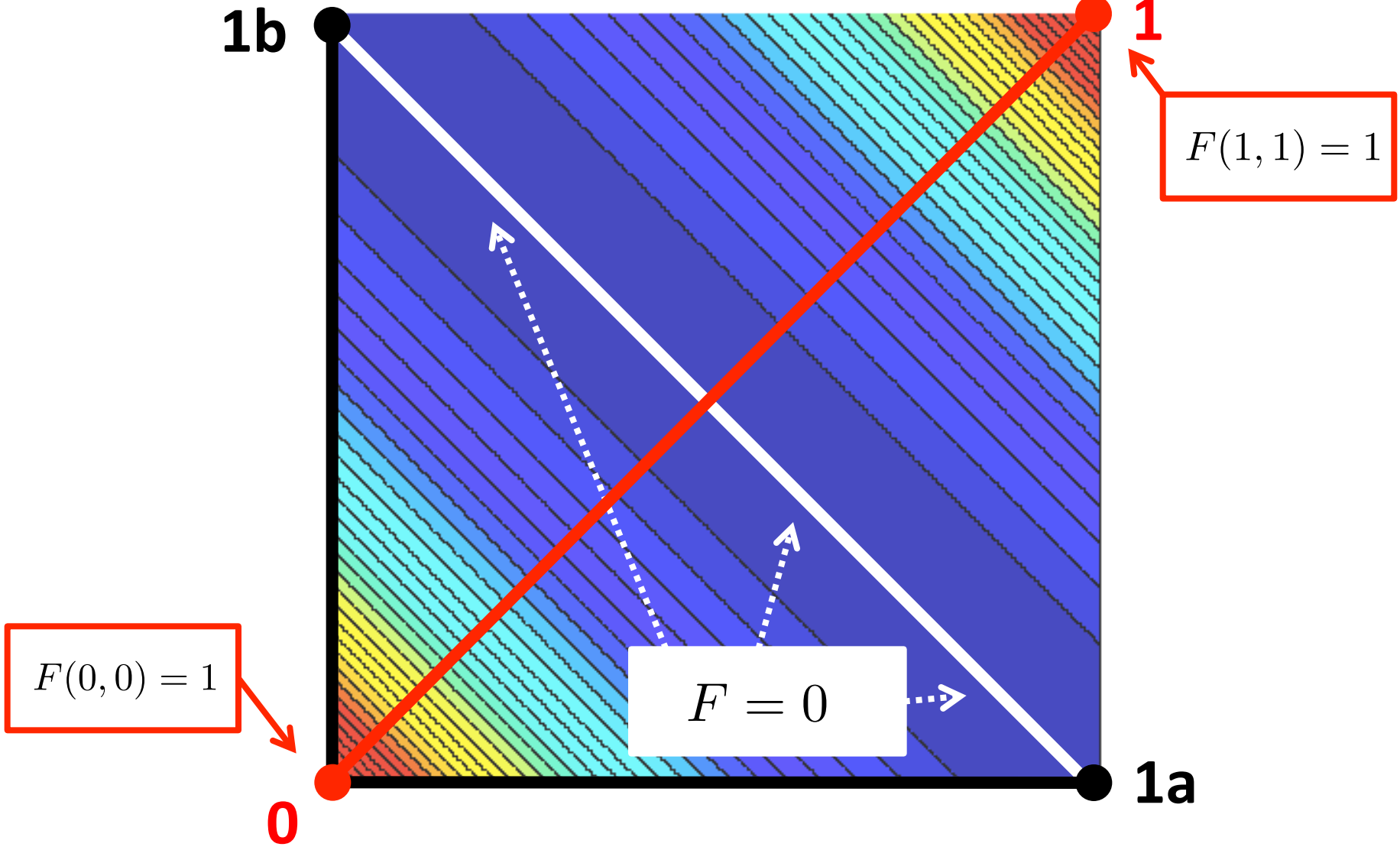
Additive Strategy

$$x = (x^1, x^2) \in \mathbb{R}^2, \quad F(x^1, x^2) = (x^1 + x^2 - 1)^2$$



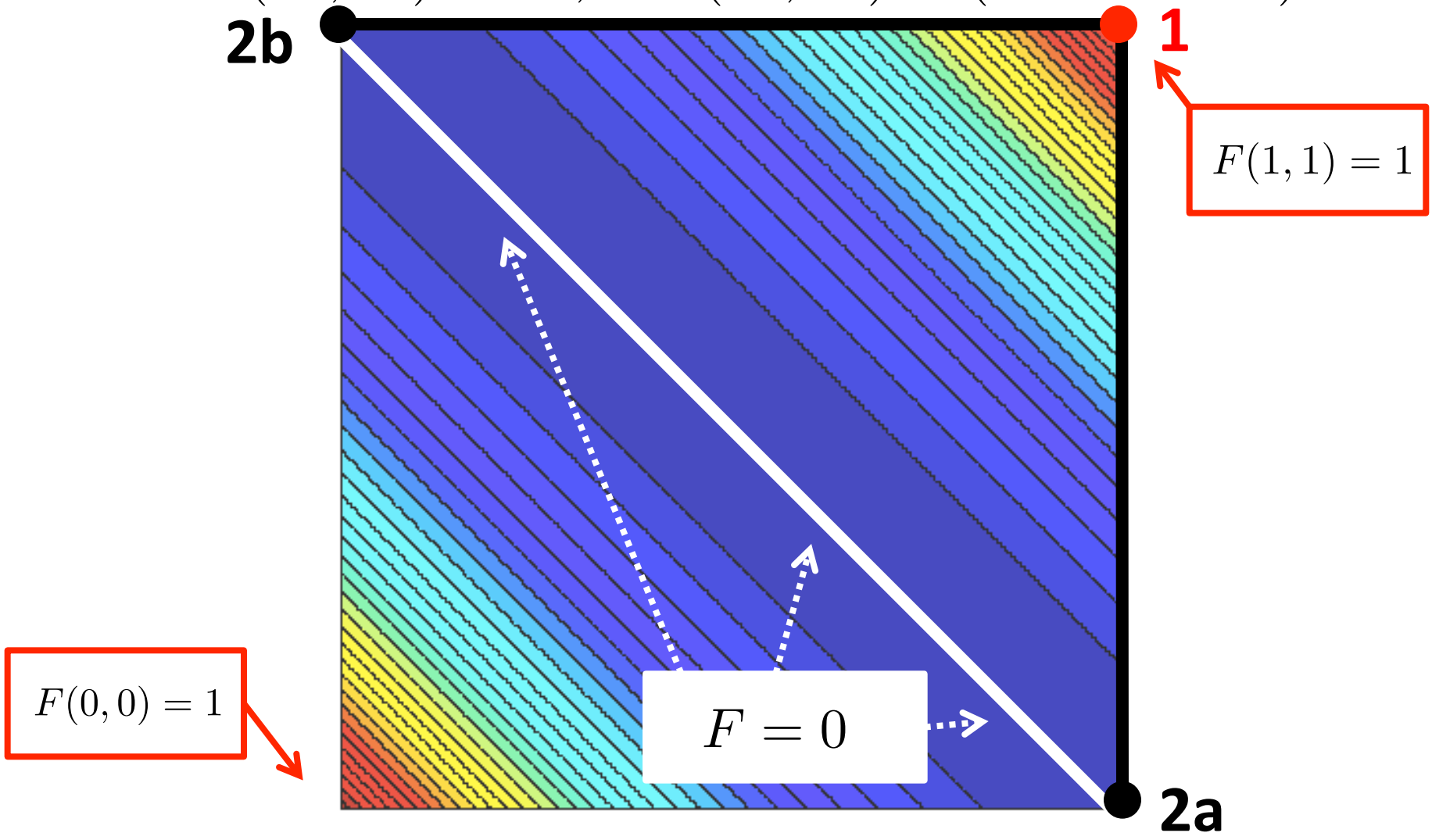
Additive Strategy

$$x = (x^1, x^2) \in \mathbb{R}^2, \quad F(x^1, x^2) = (x^1 + x^2 - 1)^2$$



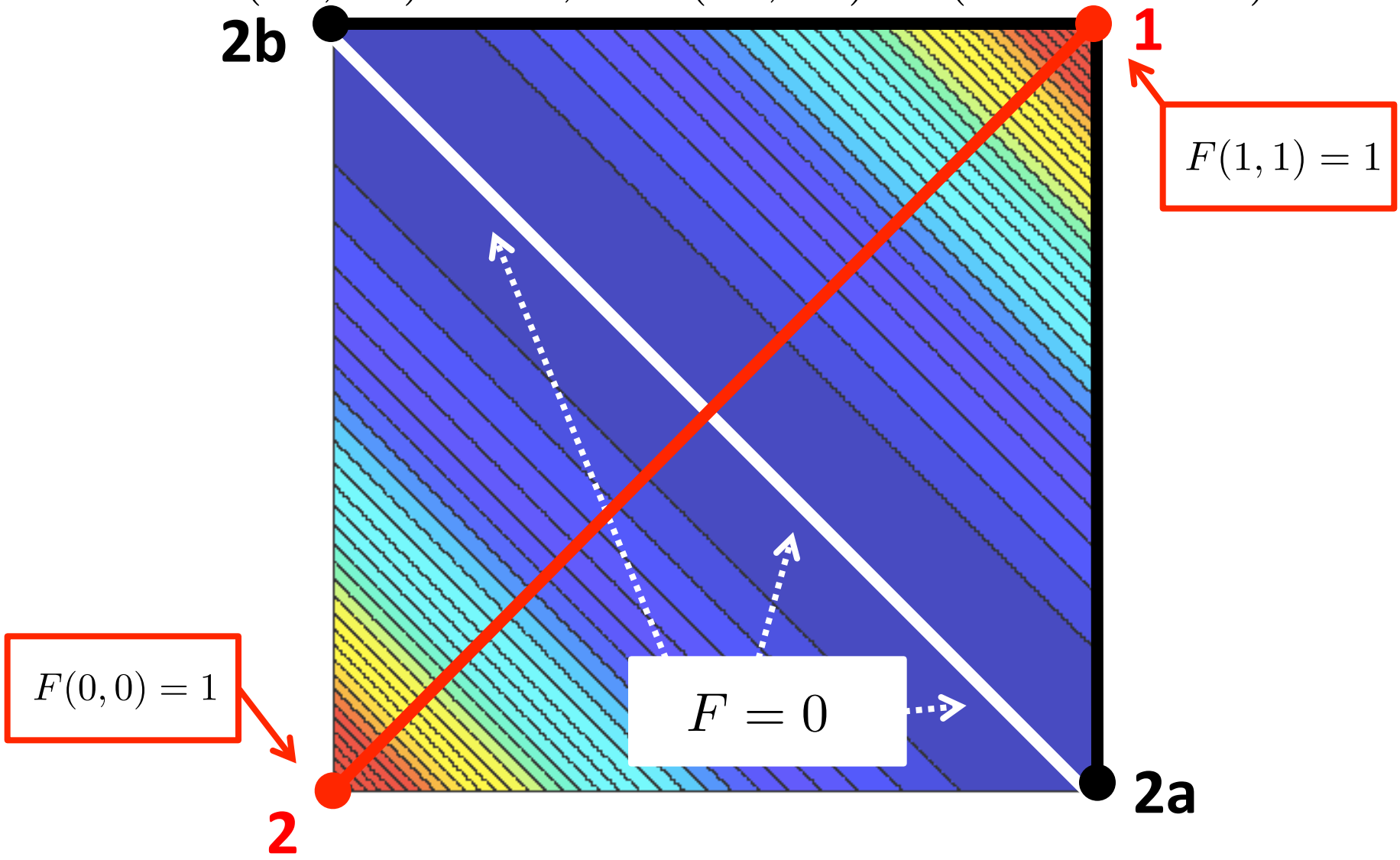
Additive Strategy

$$x = (x^1, x^2) \in \mathbb{R}^2, \quad F(x^1, x^2) = (x^1 + x^2 - 1)^2$$



Additive Strategy

$$x = (x^1, x^2) \in \mathbb{R}^2, \quad F(x^1, x^2) = (x^1 + x^2 - 1)^2$$

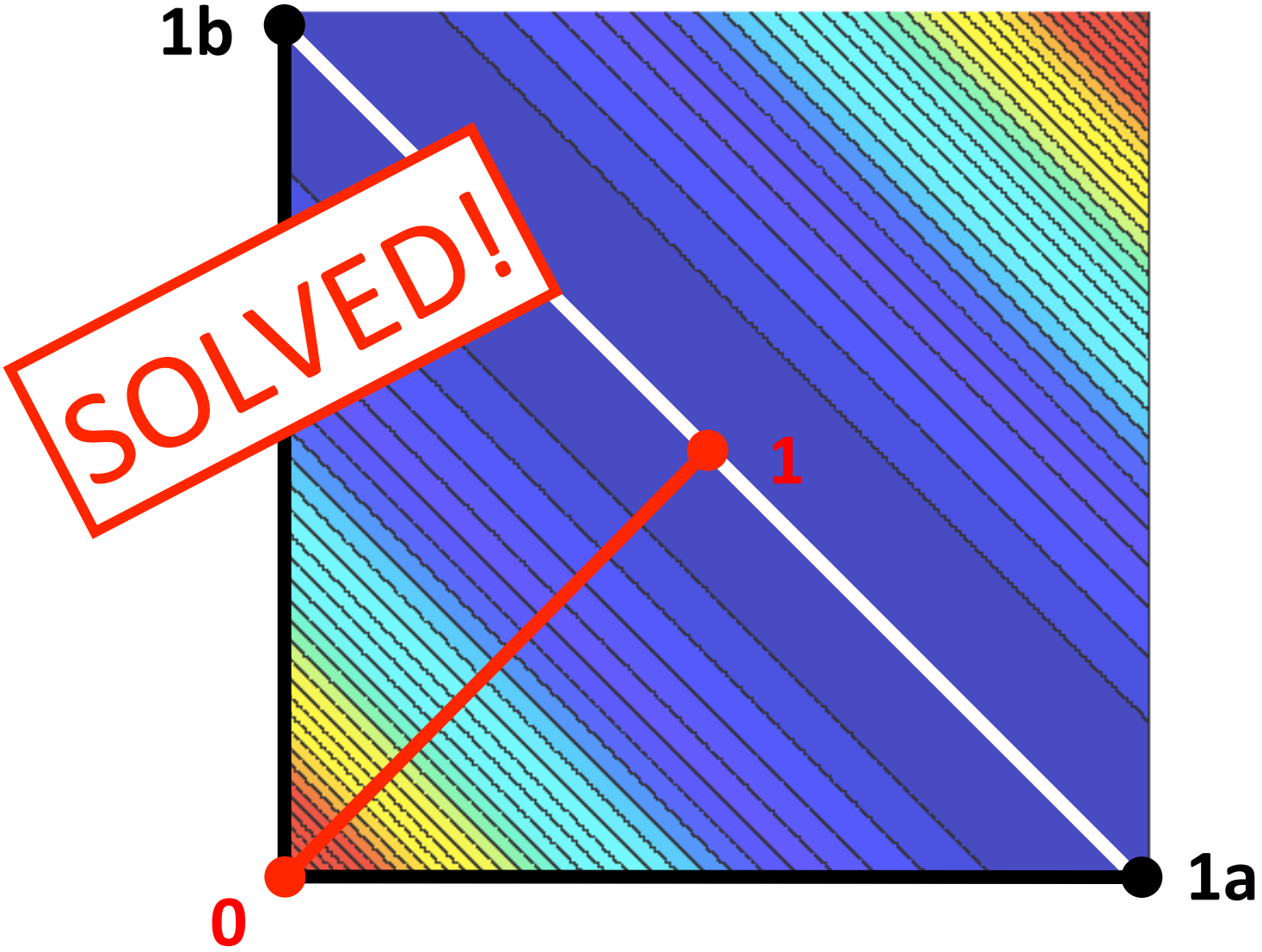


Additive Strategy

$$x = (x^1, x^2) \in \mathbb{R}^2, \quad F(x^1, x^2) = (x^1 + x^2 - 1)^2$$

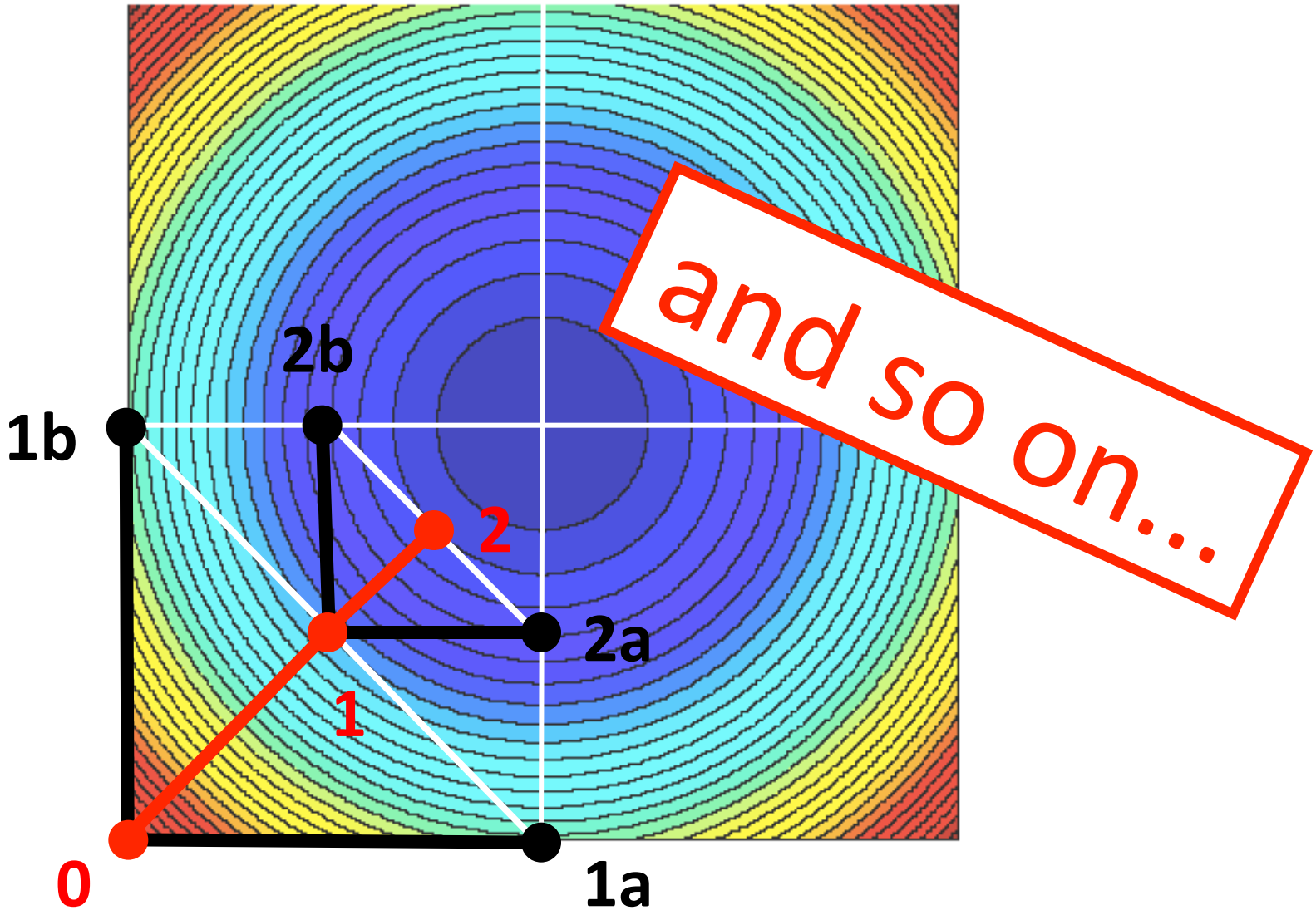
Averaging Strategy

$$x = (x^1, x^2) \in \mathbb{R}^2, \quad F(x^1, x^2) = (x^1 + x^2 - 1)^2$$



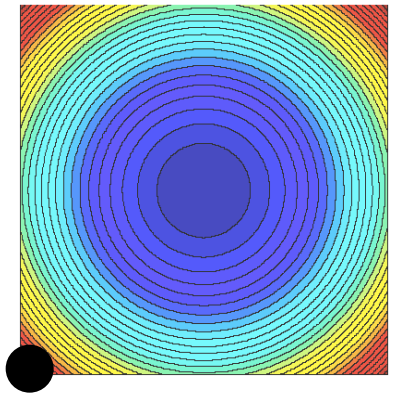
Averaging Can Be Bad, Too!

$$x = (x^1, x^2) \in \mathbb{R}^2, \quad F(x^1, x^2) = (x^1 - 1)^2 + (x^2 - 1)^2$$



Actually, Averaging Can Be Very Bad!

$$F(x) = (x^1 - 1)^2 + (x^2 - 1)^2 + \cdots + (x^n - 1)^2$$



$$x_0 = 0 \in \mathbb{R}^n \;\Rightarrow\; F(x_0) = n$$

BAD!!!

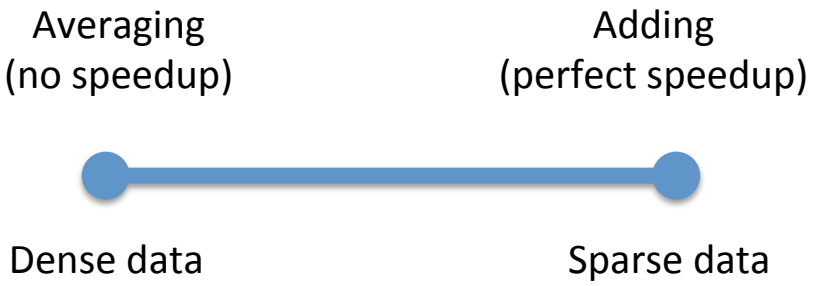
$$k \geq \frac{n}{2} \log\left(\frac{n}{\epsilon}\right)$$

$$F(x_k) = n\left(1 - \frac{1}{n}\right)^{2k} \leq \epsilon$$

WANT

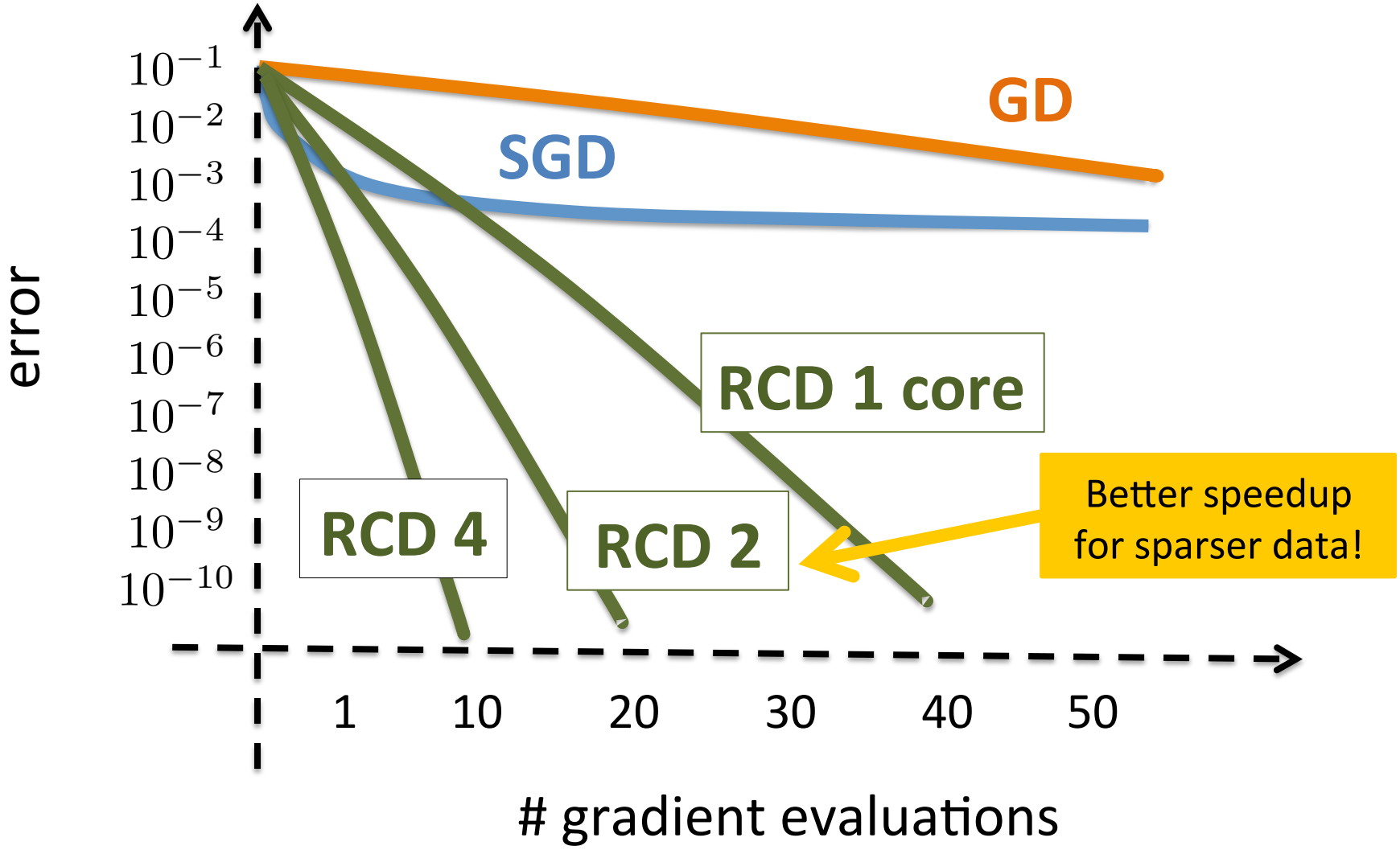
How to Combine the Updates?

- We should do data-dependent combination of the results obtained in parallel
- There is rich theory for this now



Zheng Qu and P.R.
Coordinate Descent with Arbitrary Sampling II: Expected Separable Overapproximation
Optimization Methods and Software 31(5), 858-884, 2016

Performance



Problem with 1 Billion Variables

	$F(x_k) - F^*$			Elapsed Time		
$(k \cdot \tau)/n$	1 core	8 cores	16 cores	1 core	8 cores	16 cores
0	6.27e+22	6.27e+22	6.27e+22	0.00	0.00	0.00
1	2.24e+22	2.24e+22	2.24e+22	0.89	0.11	0.06
2	2.25e+22	3.64e+19	2.24e+22	1.97	0.27	0.14
3	1.15e+20	1.94e+19	1.37e+20	3.20	0.43	0.21
4	5.25e+19	1.42e+18	8.19e+19	4.28	0.58	0.29
5	1.59e+19	1.05e+17	3.37e+19	5.37	0.73	0.37
6	1.97e+18	1.17e+16	1.33e+19	6.64	0.89	0.45
7	2.40e+16	3.18e+15	8.39e+17	7.87	1.04	0.53
⋮	⋮	⋮	⋮	⋮	⋮	⋮
26	3.49e+02	4.11e+01	3.68e+03	31.71	3.99	2.02
27	1.92e+02	5.70e+00	7.77e+02	33.00	4.14	2.10
28	1.07e+02	2.14e+00	6.69e+02	34.23	4.30	2.17
29	6.18e+00	2.35e-01	3.64e+01	35.31	4.45	2.25
30	4.31e+00	4.03e-02	2.74e+00	36.60	4.60	2.33
31	6.17e-01	3.50e-02	6.20e-01	37.90	4.75	2.41
32	1.83e-02	2.41e-03	2.34e-01	39.17	4.91	2.48
33	3.80e-03	1.63e-03	1.57e-02	40.39	5.06	2.56
34	7.28e-14	7.46e-14	1.20e-02	41.47	5.21	2.64
35	-	-	1.23e-03	-	-	2.72
36	-	-	3.99e-04	-	-	2.80
37	-	-	7.46e-14	-	-	2.87

Part 3
Conclusion

Summary

- Presented some of the key ideas, in isolation, behind big data optimization for ERM
- Focused on dual methods
  - Did not go into the theory
  - Just scratched the surface
  - Did not cover: importance sampling, adaptive sampling, arbitrary sampling, importance sampling for minibatches, …
- Left out primal methods (variance reduction): SAG, SAGA, SVRG, S2GD, MS2GD, Katyusha, …

The Tools Can Be Combined

Randomization + Acceleration + Parallelism + Proximal Trick

Olivier Fercoq and P.R.
Accelerated, Parallel and Proximal Coordinate Descent
SIAM Journal on Optimization 25(4), 1997-2023, 2015
SIAM Review 58(4), 2016

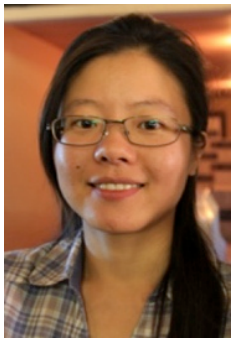
17th IMA Leslie Fox Prize (2nd), 2015
2nd Most Downloaded Paper on SIOPT

Above + Duality (ERM)

Qihang Lin, Zhaosong Lu and Lin Xiao
An Accelerated Randomized Proximal Coordinate Gradient Method and its Application to Regularized Empirical Risk Minimization
SIAM Journal on Optimization 25(4), 2244–2273, 2015

Martin Takáč
(Lehigh)

Jakub Mareček
(IBM)



Zheng Qu
(Hong Kong)

Olivier Fercoq
(Telecom ParisTech)

Rachael Tappenden
(Johns Hopkins)

Robert M Gower
(Edinburgh)

Virginia Smith
(Berkeley)

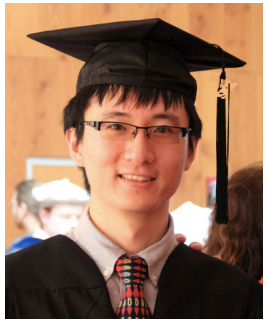
Jakub Konečný
(Edinburgh)

Jie Liu
(Lehigh)

Michael Jordan
(Berkeley)

Dominik Csba
(Edinburgh)

Tong Zhang
(Rutgers & Baidu)



Zeyuan Allen-Zhu
(Princeton)

Nati Srebro
(TTI Chicago)

Donald Goldfarb
(Columbia)

Chenxin Ma
(Lehigh)

Martin Jaggi
(ETH Zurich)