Machine Learning:

Dynamics, Economics and Stochastics

Michael I. Jordan
University of California, Berkeley

December 16, 2018

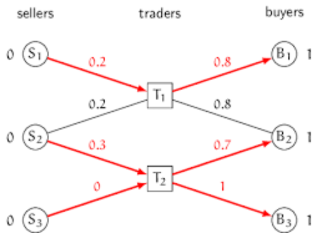
What Intelligent Systems Currently Exist?

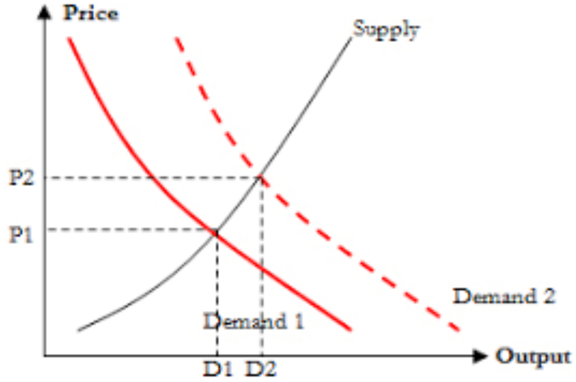
- Brains and Minds

What Intelligent Systems Currently Exist?

- Brains and Minds

- Markets





Chapter 1: History and Perspective

Machine Learning (aka, AI) Successes

- First Generation (‘90-’00): the backend
  - e.g., fraud detection, search, supply-chain management
- Second Generation (‘00-’10): the human side
  - e.g., recommendation systems, commerce, social media
- Third Generation (‘10-now): pattern recognition
  - e.g., speech recognition, computer vision, translation
- Fourth Generation (emerging): decisions and markets
  - not just one agent making a decision or sequence of decisions
  - rather, a huge interconnected web of data, agents, decisions
  - many new challenges!

Perspectives on AI

- The classical “human-imitative” perspective
  - cf. AI in the movies, interactive home robotics
- The “intelligence augmentation” (IA) perspective
  - cf. search engines, recommendation systems, natural language translation
  - the system need not be intelligent itself, but it reveals patterns that humans can make use of
- The “intelligent infrastructure” (II) perspective
  - cf. transportation, intelligent dwellings, urban planning
  - large-scale, distributed collections of data flows and loosely-coupled decisions

M. Jordan (2018), “Artificial Intelligence: The Revolution Hasn’t Happened Yet”, *Medium.*

Human-Imitative AI Isn't the Right Goal

- Problems studied from the "human-imitative" perspective aren't necessarily the same as those that arise in the IA or II perspectives
  - unfortunately, the "AI solutions" being deployed for the latter are often those developed in service of the former

- *"Autonomy" shouldn't be our main goal; rather our goal should be the development of small intelligences that work well with each other and with humans*

- *To make an overall system behave intelligently, it is neither necessary or sufficient to make each component of the system be intelligent*

Near-Term Challenges in II

- Error control for multiple decisions
- Systems that create markets
- Designing systems that can provide meaningful, calibrated notions of their uncertainty
- Achieving real-time performance goals
- Managing cloud-edge interactions
- Designing systems that can find abstractions quickly
- Provenance in systems that learn and predict
- Designing systems that can explain their decisions
- Finding causes and performing causal reasoning
- Systems that pursue long-term goals, and actively collect data in service of those goals
- Achieving fairness and diversity
- Robustness in the face of unexpected situations
- Robustness in the face of adversaries
- Sharing data among individuals and organizations
- Protecting privacy and issues of data ownership

Multiple Decisions: The Load-Balancing Problem

- In many II problems, a system doesn't make just a single decision, or a sequence of decisions, but huge numbers of linked decisions in each moment
  - those decisions often interact

Multiple Decisions: The Load-Balancing Problem

- In many II problems, a system doesn’t make just a single decision, or a sequence of decisions, but huge numbers of linked decisions in each moment
  - those decisions often interact
  - they interact when there is a scarcity of resources

Multiple Decisions: The Load-Balancing Problem

- In many II problems, a system doesn’t make just a single decision, or a sequence of decisions, but huge numbers of decentralized decisions in each moment
  - those decisions often interact
  - they interact when there is a scarcity of resources
- To manage scarcity of resources in large-scale decision making, “AI” isn’t enough; we need concepts from market design

Classical Recommendation Systems

- A record is kept of each customer’s purchases
- Customers are “similar” if they buy similar sets of items
- Items are “similar” are they are bought together by multiple customers

Classical Recommendation Systems

- A record is kept of each customer's purchases
- Customers are "similar" if they buy similar sets of items
- Items are "similar" are they are bought together by multiple customers
- Recommendations are made on the basis of these similarities
- In existing systems, recommendations are made *independently*

Classical Recommendation Systems

- A record is kept of each customer’s purchases
- Customers are “similar” if they buy similar sets of items
- Items are “similar” are they are bought together by multiple customers
- Recommendations are made on the basis of these similarities
- In existing systems, recommendations are made *independently*

- That won’t work in the real world!

Multiple Decisions: Load Balancing

- Suppose that recommending a certain movie is a good business decision (e.g., because it’s very popular)
- Is it OK to recommend the same movie to everyone?

Multiple Decisions: Load Balancing

- Suppose that recommending a certain movie is a good business decision (e.g., because it's very popular)
- Is it OK to recommend the same movie to everyone?
- Is it OK to recommend the same book to everyone?

Multiple Decisions: Load Balancing

- Suppose that recommending a certain movie is a good business decision (e.g., because it's very popular)
- Is it OK to recommend the same movie to everyone?
- Is it OK to recommend the same book to everyone?
- Is it OK to recommend the same restaurant to everyone?

Multiple Decisions: Load Balancing

- Suppose that recommending a certain movie is a good business decision (e.g., because it's very popular)
- Is it OK to recommend the same movie to everyone?
- Is it OK to recommend the same book to everyone?
- Is it OK to recommend the same restaurant to everyone?
- Is it OK to recommend the same street to every driver?

Multiple Decisions: Load Balancing

- Suppose that recommending a certain movie is a good business decision (e.g., because it's very popular)
- Is it OK to recommend the same movie to everyone?
- Is it OK to recommend the same book to everyone?
- Is it OK to recommend the same restaurant to everyone?
- Is it OK to recommend the same street to every driver?
- Is it OK to recommend the same stock purchase to everyone?

Multiple Decisions: Load Balancing

- Suppose that recommending a certain movie is a good business decision (e.g., because it's very popular)
- Is it OK to recommend the same movie to everyone?
- Is it OK to recommend the same book to everyone?
- Is it OK to recommend the same restaurant to everyone?
- Is it OK to recommend the same street to every driver?
- Is it OK to recommend the same stock purchase to everyone?

- Such problems are best approached via the creation of markets
  - restaurants bid on customers
  - street segments bid on drivers

The Consequences

- By creating a market based on the data flows, new jobs are created!
- So here's a way that AI can be a job creator, and not (mostly) a job killer
- This can be done in a wide range of other domains, not just music
  - entertainment
  - information services
  - personal services
  - etc

Near-Term Challenges in II

- Error control for multiple decisions
- Systems that create markets
- Designing systems that can provide meaningful, calibrated notions of their uncertainty
- Achieving real-time performance goals
- Managing cloud-edge interactions
- Designing systems that can find abstractions quickly
- Provenance in systems that learn and predict
- Designing systems that can explain their decisions
- Finding causes and performing causal reasoning
- Systems that pursue long-term goals, and actively collect data in service of those goals
- Achieving fairness and diversity
- Robustness in the face of unexpected situations
- Robustness in the face of adversaries
- Sharing data among individuals and organizations
- Protecting privacy and issues of data ownership

Chapter 2: In the Engine Room

Algorithmic and Theoretical Progress

- Nonconvex optimization
  - avoidance of saddle points
  - rates that have dimension dependence
  - acceleration, dynamical systems and lower bounds
  - statistical guarantees from optimization guarantees
- Computationally-efficient sampling
  - nonconvex functions
  - nonreversible MCMC
  - links to optimization
- Market design
  - approach to saddle points
  - recommendations and two-way markets

Computation and Statistics

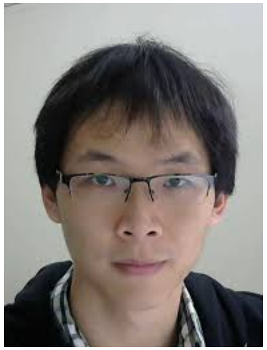
- A Grand Challenge of our era: tradeoffs between statistical inference and computation
  - most data analysis problems have a time budget
  - and often they're embedded in a control problem
- Optimization has provided the computational model for this effort (computer science, not so much)
  - it's provided the algorithms and the insight
- On the other hand, modern large-scale statistics has posed new challenges for optimization
  - millions of variables, millions of terms, sampling issues, nonconvexity, need for confidence intervals, parallel/distributed platforms, etc

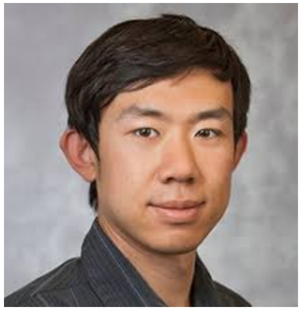
Computation and Statistics (cont)

- Modern large-scale statistics has posed new challenges for optimization
  - millions of variables, millions of terms, sampling issues, nonconvexity, need for confidence intervals, parallel/distributed platforms, etc
- Current algorithmic focus: what can we do with the following ingredients?
  - gradients
  - stochastics
  - acceleration
- Current theoretical focus: placing lower bounds from statistics and optimization in contact with each other

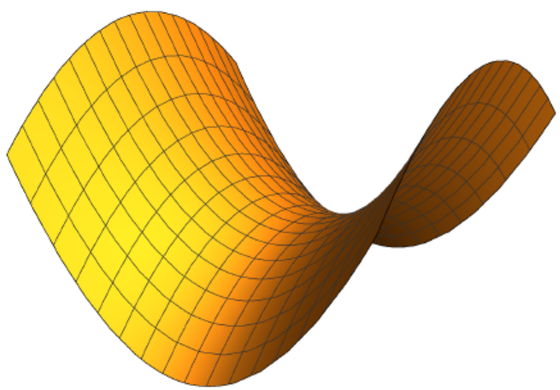
Part I: How to Escape Saddle Points Efficiently

with Chi Jin, Praneeth Netrapalli, Rong Ge, and Sham Kakade

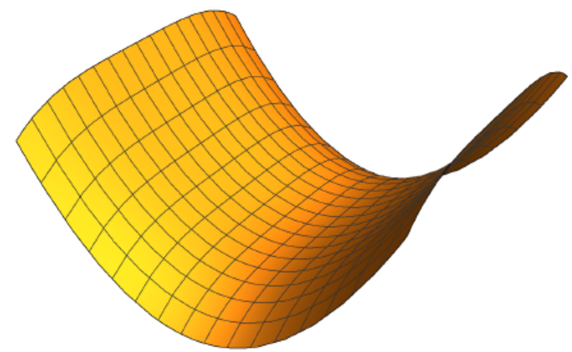





The Importance of Saddle Points



Strict saddle point



Non-strict saddle point

- How to escape?
  - need to have a negative eigenvalue that's strictly negative
- How to escape efficiently?
  - in high dimensions how do we find the direction of escape?
  - should we expect exponential complexity in dimension?

Some Well-Behaved Nonconvex Problems

- PCA, CCA, Matrix Factorization
- Orthogonal Tensor Decomposition (Ge, Huang, Jin, Yang, 2015)
- Complete Dictionary Learning (Sun et al, 2015)
- Phase Retrieval (Sun et al, 2015)
- Matrix Sensing (Bhojanapalli et al, 2016; Park et al, 2016)
- Symmetric Matrix Completion (Ge et al, 2016)
- Matrix Sensing/Completion, Robust PCA (Ge, Jin, Zheng, 2017)

- The problems have no spurious local minima and all saddle points are strict

A Few Facts

- Gradient descent will asymptotically avoid saddle points (Lee, Simchowitz, Jordan & Recht, 2017)
- Gradient descent can take exponential time to escape saddle points (Du, Jin, Lee, Jordan, & Singh, 2017)
- Stochastic gradient descent can escape saddle points in polynomial time (Ge, Huang, Jin & Yuan, 2015)
  - but that’s still not an explanation for its practical success
- Can we prove a stronger theorem?

Optimization

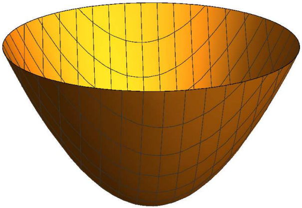
Consider problem:

$$\min_{\mathbf{x}\in\mathbb{R}^d} \; f(\mathbf{x})$$

Gradient Descent (GD):

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta\nabla f(\mathbf{x}_t).$$

Convex: converges to global minimum; **dimension-free** iterations.



Convergence to FOSP

Function $f(\cdot)$ is ℓ**-smooth (or gradient Lipschitz)**

$$\forall \mathbf{x}_1, \mathbf{x}_2,\ \|\nabla f(\mathbf{x}_1) - \nabla f(\mathbf{x}_2)\| \le \ell \|\mathbf{x}_1 - \mathbf{x}_2\|.$$

Point $\mathbf{x}$ is an ϵ**-first-order stationary point** (ϵ-FOSP) if

$$\|\nabla f(\mathbf{x})\| \le \epsilon$$

Theorem [GD Converges to FOSP (Nesterov, 1998)]

For ℓ-smooth function, GD with $\eta = 1/\ell$ finds ϵ-FOSP in iterations:

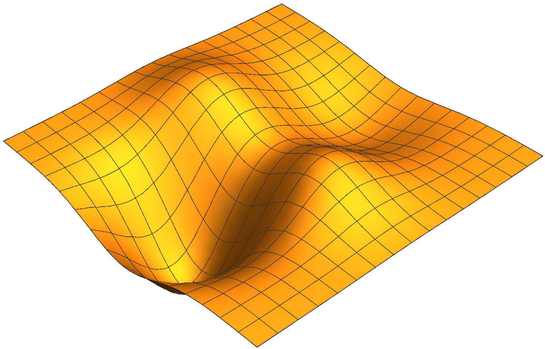
$$\frac{2\ell(f(\mathbf{x}_0) - f^\star)}{\epsilon^2}$$

*Number of iterations is dimension free.

Nonconvex Optimization

Non-convex: converges to Stationary Point (SP) $\nabla f(\mathbf{x}) = 0$.

SP : local min / local max / saddle points



Many applications: no spurious local min (see full list later).

Definitions and Algorithm

Function $f(\cdot)$ is ρ-**Hessian Lipschitz** if

$$\forall \mathbf{x}_1, \mathbf{x}_2,\ \|\nabla^2 f(\mathbf{x}_1) - \nabla^2 f(\mathbf{x}_2)\| \le \rho\|\mathbf{x}_1 - \mathbf{x}_2\|.$$

Point $\mathbf{x}$ is an ϵ-**second-order stationary point** (ϵ-SOSP) if

$$\|\nabla f(\mathbf{x})\| \le \epsilon, \qquad \text{and} \qquad \lambda_{\min}(\nabla^2 f(\mathbf{x})) \ge -\sqrt{\rho\epsilon}$$

Definitions and Algorithm

Function $f(\cdot)$ is ρ-**Hessian Lipschitz** if

$$\forall \mathbf{x}_1, \mathbf{x}_2, \ \|\nabla^2 f(\mathbf{x}_1) - \nabla^2 f(\mathbf{x}_2)\| \leq \rho \|\mathbf{x}_1 - \mathbf{x}_2\|.$$

Point $\mathbf{x}$ is an ϵ-**second-order stationary point** (ϵ-SOSP) if

$$\|\nabla f(\mathbf{x})\| \leq \epsilon, \qquad \text{and} \qquad \lambda_{\min}(\nabla^2 f(\mathbf{x})) \geq -\sqrt{\rho\epsilon}$$

Algorithm Perturbed Gradient Descent (PGD)

1. **for** $t = 0, 1, \ldots$ **do**
2. **if** perturbation condition holds **then**
3. $\mathbf{x}_t \leftarrow \mathbf{x}_t + \xi_t, \qquad \xi_t$ uniformly $\sim \mathbb{B}_0(r)$
4. $\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t - \eta \nabla f(\mathbf{x}_t)$

Adds perturbation when $\|\nabla f(\mathbf{x}_t)\| \leq \epsilon$; no more than once per T steps.

Main Result

Theorem [PGD Converges to SOSP]

For ℓ-smooth and ρ-Hessian Lipschitz function f, PGD with $\eta = O(1/\ell)$ and proper choice of r, T w.h.p. finds ϵ-SOSP in iterations:

$$\tilde{O}\left(\frac{\ell(f(\mathbf{x}_0) - f^\star)}{\epsilon^2}\right)$$

*Dimension dependence in iteration is $\log^4(d)$ (almost dimension free).

Main Result

Theorem [PGD Converges to SOSP]

For ℓ-smooth and ρ-Hessian Lipschitz function f, PGD with $\eta = O(1/\ell)$ and proper choice of r, T w.h.p. finds ϵ-SOSP in iterations:

$$\tilde{O}\left(\frac{\ell(f(\mathbf{x}_0) - f^\star)}{\epsilon^2}\right)$$

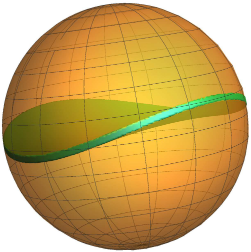
*Dimension dependence in iteration is $\log^4(d)$ (almost dimension free).

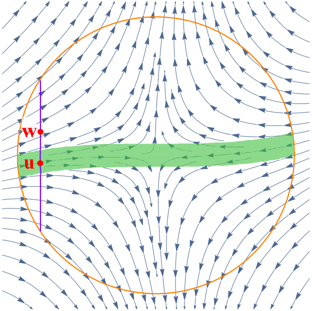
	GD(Nesterov 1998)	**PGD**(This Work)
Assumptions	ℓ-grad-Lip	ℓ-grad-Lip + ρ-Hessian-Lip
Guarantees	ϵ-FOSP	ϵ-SOSP
Iterations	$2\ell(f(\mathbf{x}_0) - f^\star)/\epsilon^2$	$\tilde{O}(\ell(f(\mathbf{x}_0) - f^\star)/\epsilon^2)$

Geometry and Dynamics around Saddle Points

Challenge: non-constant Hessian + large step size $\eta = O(1/\ell)$.

Around saddle point, **stuck region** forms a non-flat "pancake" shape.





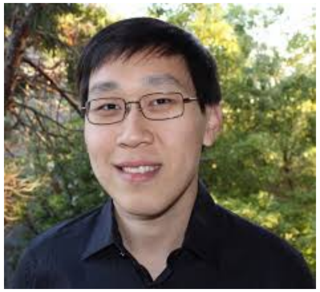
Key Observation: although we don't know its shape, we know it's thin!
(Based on an analysis of two nearly coupled sequences)

How Fast Can We Go?

- Important role of lower bounds (Nemirovski & Yudin)
  - strip away inessential aspects of the problem to reveal fundamentals
- The acceleration phenomenon (Nesterov)
  - achieve the lower bounds
  - second-order dynamics
  - a conceptual mystery
- Our perspective: it's essential to go to continuous time
  - the notion of ”acceleration” requires a continuum topology to support it

Part II: Variational, Hamiltonian and Symplectic Perspectives on Acceleration

with Andre Wibisono, Ashia Wilson and
Michael Betancourt



Accelerated gradient descent

Setting: Unconstrained convex optimization

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

- Classical gradient descent:

$$x_{k+1} = x_k - \beta \nabla f(x_k)$$

obtains a convergence rate of $O(1/k)$

Accelerated gradient descent

Setting: Unconstrained convex optimization

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

- Classical gradient descent:

$$x_{k+1} = x_k - \beta \nabla f(x_k)$$

obtains a convergence rate of $O(1/k)$

- Accelerated gradient descent:

$$\begin{aligned} y_{k+1} &= x_k - \beta \nabla f(x_k) \\ x_{k+1} &= (1 - \lambda_k) y_{k+1} + \lambda_k y_k \end{aligned}$$

obtains the (optimal) convergence rate of $O(1/k^2)$

Accelerated methods: Continuous time perspective

- Gradient descent is discretization of gradient flow

$$\dot{X}_t = -\nabla f(X_t)$$

(and mirror descent is discretization of natural gradient flow)

Accelerated methods: Continuous time perspective

- Gradient descent is discretization of gradient flow

$$\dot{X}_t = -\nabla f(X_t)$$

  (and mirror descent is discretization of natural gradient flow)

- Su, Boyd, Candes '14: Continuous time limit of accelerated gradient descent is a second-order ODE

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

Accelerated methods: Continuous time perspective

- Gradient descent is discretization of gradient flow

$$\dot{X}_t = -\nabla f(X_t)$$

(and mirror descent is discretization of natural gradient flow)

- Su, Boyd, Candes '14: Continuous time limit of accelerated gradient descent is a second-order ODE

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

- These ODEs are obtained by taking continuous time limits. Is there a deeper generative mechanism?

Our work: A general variational approach to acceleration

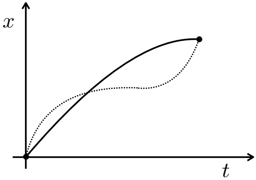
A systematic discretization methodology

Bregman Lagrangian

$$\mathcal{L}(x, \dot{x}, t) = e^{\gamma_t + \alpha_t} \left(D_h(x + e^{-\alpha_t} \dot{x}, x) - e^{\beta_t} f(x) \right)$$

Variational problem over curves:

$$\min_X \int \mathcal{L}(X_t, \dot{X}_t, t)\, dt$$



Optimal curve is characterized by **Euler-Lagrange** equation:

$$\frac{d}{dt} \left\{ \frac{\partial \mathcal{L}}{\partial \dot{x}}(X_t, \dot{X}_t, t) \right\} = \frac{\partial \mathcal{L}}{\partial x}(X_t, \dot{X}_t, t)$$

E-L equation for Bregman Lagrangian under ideal scaling:

$$\ddot{X}_t + (e^{\alpha_t} - \dot{\alpha}_t)\dot{X}_t + e^{2\alpha_t + \beta_t} \left[\nabla^2 h(X_t + e^{-\alpha_t} \dot{X}_t) \right]^{-1} \nabla f(X_t) = 0$$

Mysteries

- Why can’t we discretize the dynamics when we are using exponentially fast clocks?
- What happens when we arrive at a clock speed that we can discretize?
- How do we discretize once it’s possible?

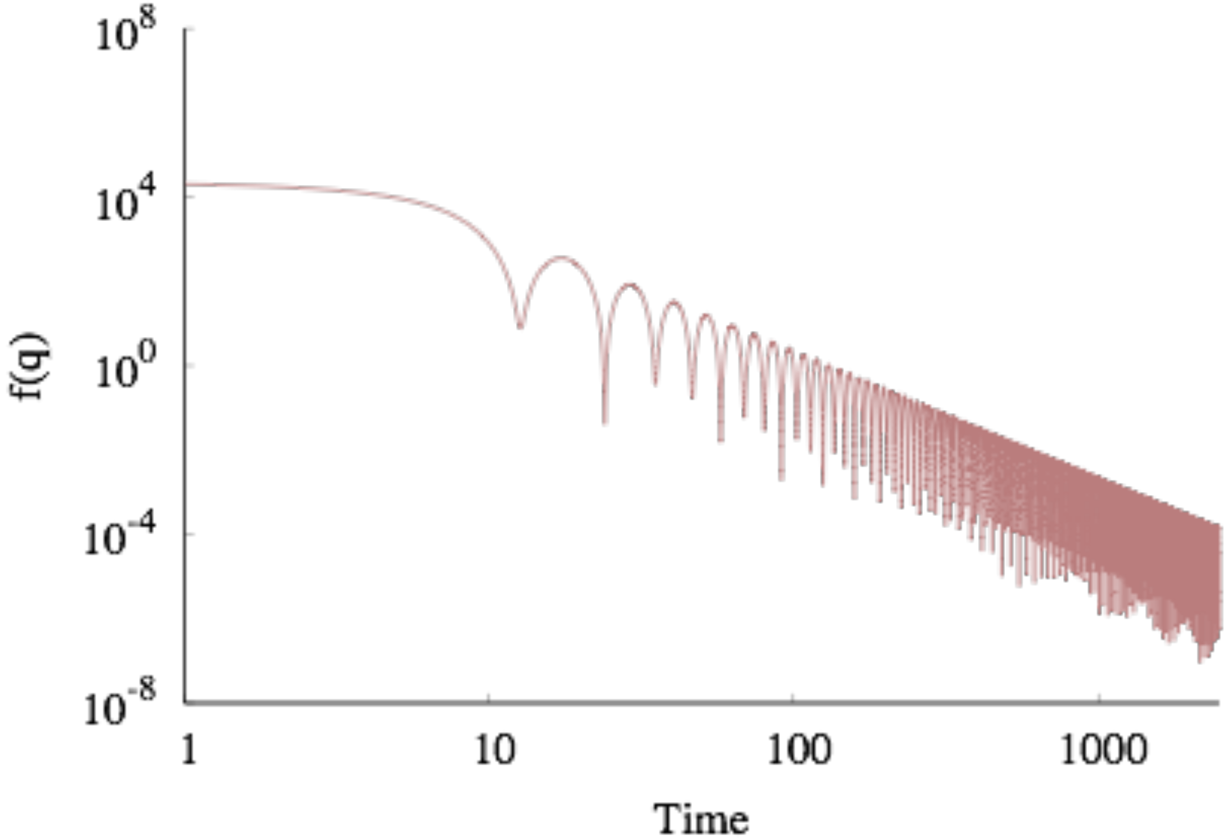
Towards A Symplectic Perspective

- We’ve discussed discretization of Lagrangian-based dynamics
- Discretization of Lagrangian dynamics is often fragile and requires small step sizes
- We can build more robust solutions by taking a Legendre transform and considering a *Hamiltonian* formalism:

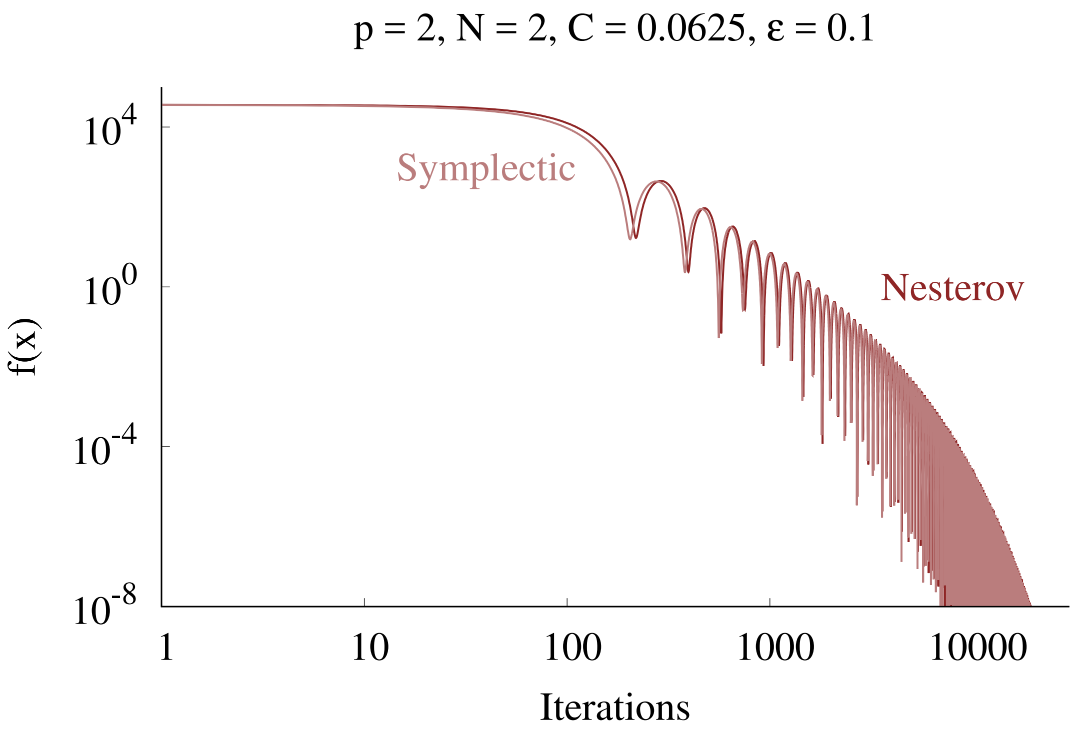
$$L(q, v, t) \to H(q, p, t, \mathcal{E})$$

$$\left(\frac{\mathrm{d}q}{\mathrm{d}t}, \frac{\mathrm{d}v}{\mathrm{d}t} \right) \to \left(\frac{\mathrm{d}q}{\mathrm{d}\tau}, \frac{\mathrm{d}p}{\mathrm{d}\tau}, \frac{\mathrm{d}t}{\mathrm{d}\tau}, \frac{\mathrm{d}\mathcal{E}}{\mathrm{d}\tau} \right)$$

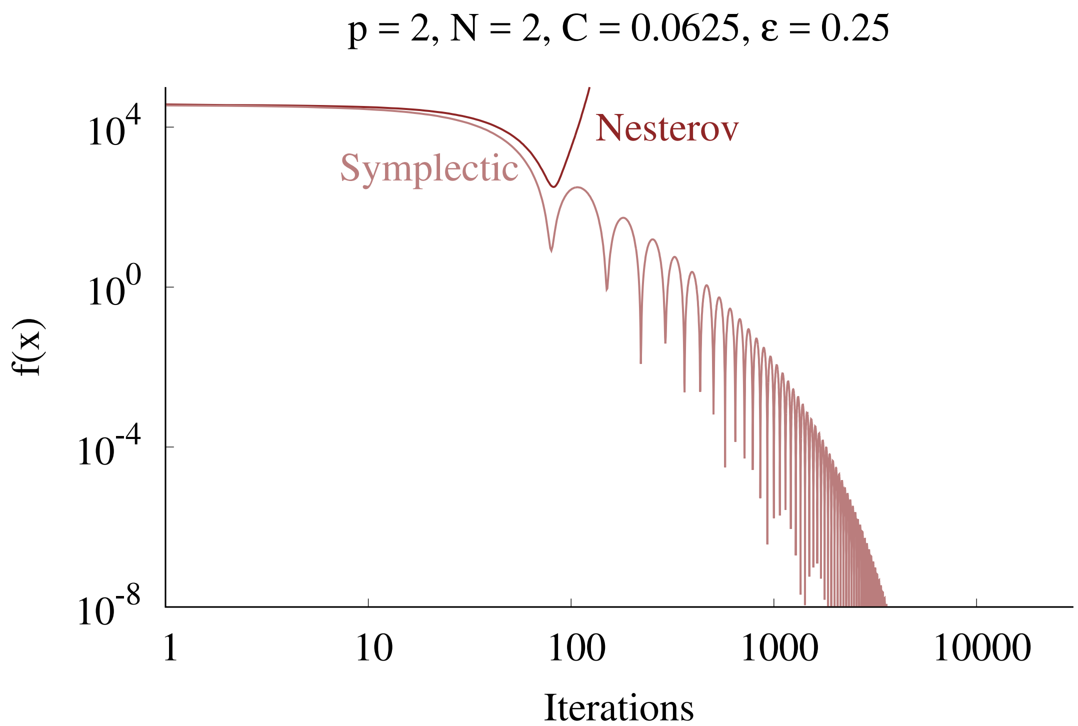
Symplectic Integration of Bregman Hamiltonian



Symplectic vs Nesterov



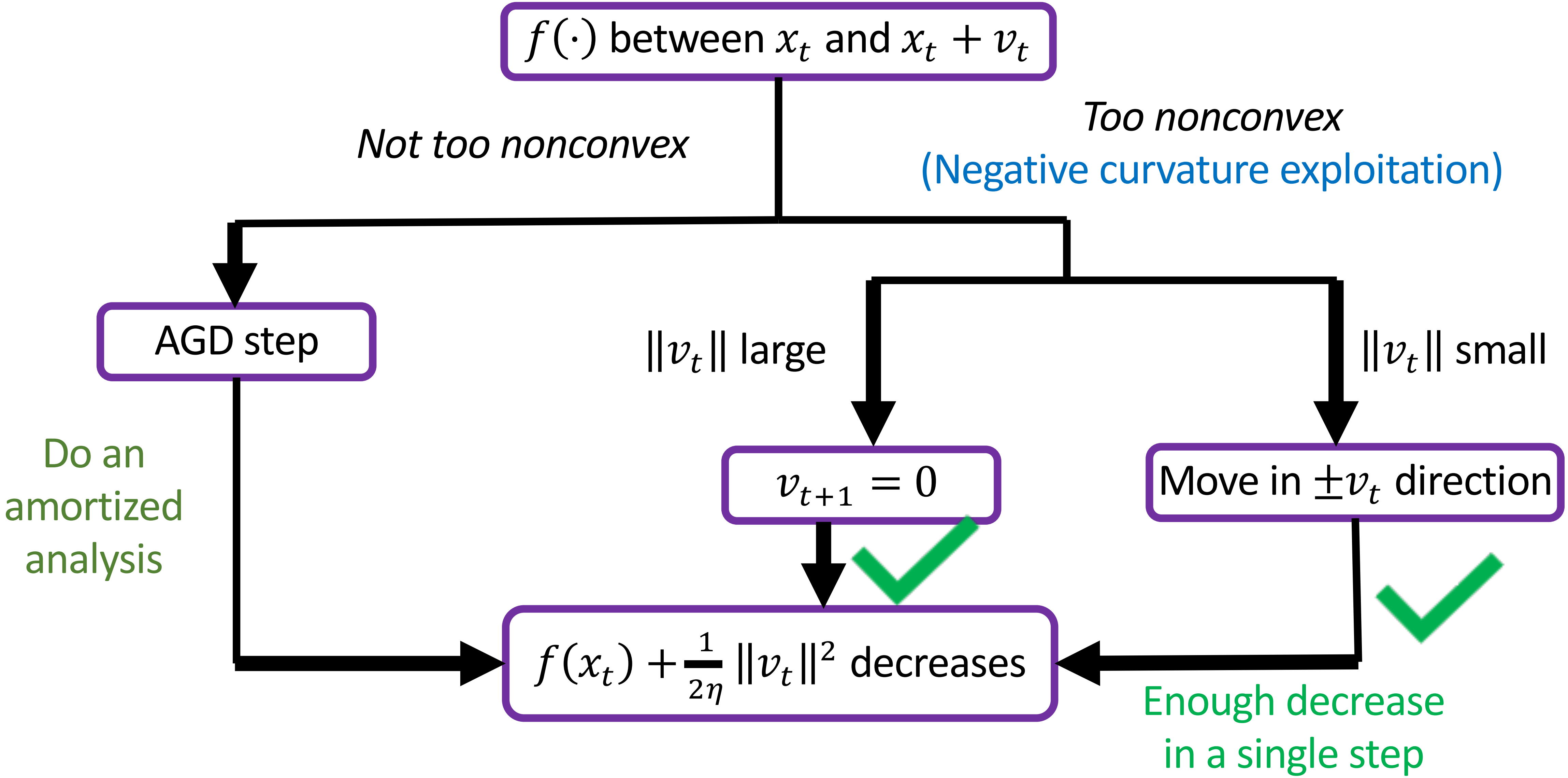
Symplectic vs Nesterov



Part III: Acceleration and Saddle Points

with Chi Jin and Praneeth Netrapalli

Hamiltonian Analysis



Convergence Result

PAGD Converges to SOSP Faster (Jin et al. 2017)

For ℓ-gradient Lipschitz and ρ-Hessian Lipschitz function f, PAGD with proper choice of $\eta, \theta, r, T, \gamma, s$ w.h.p. finds ϵ-SOSP in iterations:

$$\tilde{O}\left(\frac{\ell^{1/2}\rho^{1/4}(f(\mathbf{x}_0) - f^\star)}{\epsilon^{7/4}}\right)$$

	Strongly Convex	Nonconvex (SOSP)
Assumptions	ℓ-grad-Lip & α-str-convex	ℓ-grad-Lip & ρ-Hessian-Lip
(Perturbed) GD	$\tilde{O}(\ell/\alpha)$	$\tilde{O}(\Delta_f \cdot \ell/\epsilon^2)$
(Perturbed) AGD	$\tilde{O}(\sqrt{\ell/\alpha})$	$\tilde{O}(\Delta_f \cdot \ell^{\frac{1}{2}}\rho^{\frac{1}{4}}/\epsilon^{\frac{7}{4}})$
Condition κ	ℓ/α	$\ell/\sqrt{\rho\epsilon}$
Improvement	$\sqrt{\kappa}$	$\sqrt{\kappa}$

Part IV: Acceleration and Stochastics

with Xiang Cheng, Niladri Chatterji and Peter Bartlett

Acceleration and Stochastics

- Can we accelerate diffusions?
- There have been negative results…
- …but they've focused on classical overdamped diffusions

Acceleration and Stochastics

- Can we accelerate diffusions?
- There have been negative results…
- …but they've focused on classical overdamped diffusions
- Inspired by our work on acceleration, can we accelerate underdamped diffusions?

Overdamped Langevin MCMC

Described by the Stochastic Differential Equation (SDE):

$$dx_t = -\nabla U(x_t)dt + \sqrt{2}dB_t$$

where $U(x): R^d \to R$ and B_t is standard Brownian motion.

The stationary distribution is $p^*(x) \propto \exp\big(U(x)\big)$

Corresponding Markov Chain Monte Carlo Algorithm (MCMC):

$$\tilde{x}_{(k+1)\delta} = \tilde{x}_{k\delta} - \nabla U(\tilde{x}_{k\delta}) + \sqrt{2\delta}\xi_k$$

where δ is the *step-size* and $\xi_k \sim N(0, I_{d\times d})$

Guarantees under Convexity

Assuming $U(x)$ is L-smooth and m-strongly convex:

Dalalyan'14: Guarantees in Total Variation

If $n \geq O\left(\frac{d}{\epsilon^2}\right)$ then, $TV(p^{(n)}, p^*) \leq \epsilon$

Durmus & Moulines'16: Guarantees in 2-Wasserstein

If $n \geq O\left(\frac{d}{\epsilon^2}\right)$ then, $W_2(p^{(n)}, p^*) \leq \epsilon$

Cheng and Bartlett'17: Guarantees in KL divergence

If $n \geq O\left(\frac{d}{\epsilon^2}\right)$ then, KL$(p^{(n)}, p^*) \leq \epsilon$

Underdamped Langevin Diffusion

Described by the *second-order* equation:

$$dx_t = v_t dt$$
$$dv_t = -\gamma v_t dt + \lambda \nabla U(x_t) dt + \sqrt{2\gamma\lambda}\, dB_t$$

The stationary distribution is $p^*(x, v) \propto \exp\left(-U(x) - \frac{|v|_2^2}{2\lambda}\right)$

Intuitively, x_t is the position and v_t is the velocity

$\nabla U(x_t)$ is the force and γ is the drag coefficient

Quadratic Improvement

Let $p^{(n)}$ denote the distribution of $(\tilde{x}_{n\delta}, \tilde{v}_{n\delta})$. Assume $U(x)$ is strongly convex

Cheng, Chatterji, Bartlett, Jordan '17:

If $n \geq O\left(\frac{\sqrt{d}}{\epsilon}\right)$ then $W_2\left(p^{(n)}, p^*\right) \leq \epsilon$

Compare with Durmus & Moulines '16 (Overdamped)

If $n \geq O\left(\frac{d}{\epsilon^2}\right)$ then $W_2\left(p^{(n)}, p^*\right) \leq \epsilon$

Proof Idea: Reflection Coupling

Tricky to prove continuous-time process contracts. Consider two processes,

$$dx_t = -\nabla U(x_t)dt + \sqrt{2}\, dB_t^x$$
$$dy_t = -\nabla U(y_t)dt + \sqrt{2}\, dB_t^y$$

where $x_0 \sim p_0$ and $y_0 \sim p^*$. Couple these through Brownian motion

$$dB_t^y = \left[I_{d\times d} - \frac{2\cdot(x_t - y_t)(x_t - y_t)^\top}{|x_t - y_t|_2^2} \right] dB_t^x$$

"reflection along line separating the two processes"

Reduction to One Dimension

By Itô's Lemma we can monitor the evolution of the separation distance

$$d|x_t - y_t|_2 = -\left\langle \frac{x_t - y_t}{|x_t - y_t|_2}, \nabla U(x_t) - \nabla U(y_t) \right\rangle dt + 2\sqrt{2} dB_t^1$$

'Drift'

'1-d random walk'

Two cases are possible

1. If $|x_t - y_t|_2 \leq R$ then we have strong convexity; the drift helps.
2. If $|x_t - y_t|_2 \geq R$ then the drift hurts us, but Brownian motion helps stick*

Rates not exponential in d as we have a 1-d random walk

*Under a clever choice of Lyapunov function.

Part V: Optimization vs. Sampling

With Yi-An Ma, Yuansi Chen, Chi Jin and Nicolas Flammarion

Sampling vs. Optimization: The Tortoise and the Hare

- Folk knowledge: Sampling is slow, while optimization is fast
  - but sampling provides inferences, while optimization only provides point estimates
- But there hasn't been a clear theoretical analysis that establishes this folk knowledge as true

Sampling vs. Optimization: The Tortoise and the Hare

- Folk knowledge: Sampling is slow, while optimization is fast
  - but sampling provides inferences, while optimization only provides point estimates
- But there hasn’t been a clear theoretical analysis that establishes this folk knowledge as true
- Is it really true?

Sampling vs. Optimization: The Tortoise and the Hare

- Folk knowledge: Sampling is slow, while optimization is fast
  - but sampling provides inferences, while optimization only provides point estimates
- But there hasn't been a clear theoretical analysis that establishes this folk knowledge as true
- Is it really true?
- Define the *mixing time*:

$$\tau(\epsilon, p^0) = \min\{k \mid \|p^k - p^*\|_{\mathrm{TV}} \leq \epsilon\}$$

- We'll study the Unadjusted Langevin Algorithm (ULA) and the Metropolis-Adjusted Langevin Algorithm (MALA)

Sampling

Theorem. *For $p^* \propto e^{-U}$, we assume that U is m-strongly convex outside of a region of radius R and L-smooth. Let $\kappa = L/m$ denote the condition number of U. Let $p^0 = \mathcal{N}(0, \frac{1}{L}I)$ and let $\epsilon \in (0,1)$. Then ULA satisfies*

$$\tau_{ULA}(\epsilon, p^0) \leq \mathcal{O}\left(e^{32LR^2}\kappa^2 \frac{d}{\epsilon^2} \ln\left(\frac{d}{\epsilon^2}\right)\right).$$

For MALA,

$$\tau_{MALA}(\epsilon, p^0) \leq \mathcal{O}\left(e^{16LR^2}\kappa^{1.5}\left(d \ln \kappa + \ln\left(\frac{1}{\epsilon}\right)\right)^{3/2} d^{1/2}\right).$$

Optimization

Theorem. For any radius $R > 0$, Lipschitz and strong convexity constants $L \geq 2m > 0$, probability $0 < p \leq 1$, there exists an objective function $U(x)$ where $x \in \mathbb{R}^d$ and U is L-Lipschitz smooth and m-strongly convex for $\|x\|_2 > 2R$, such that for any optimization algorithm that inputs $\{U(x), \nabla U(x), \ldots, \nabla^n U(x)\}$, for some n, at least

$$K \geq \mathcal{O}\left(p \cdot (LR^2/\epsilon)^{d/2}\right)$$

steps are required for $\epsilon \leq \mathcal{O}(LR^2)$ so that $P(|U(x_K) - U(x^*)| < \epsilon) \geq p$.

Part VI: Acceleration and Sampling

With Yi-An Ma, Niladri Chatterji, and Xiang Cheng

Acceleration of SDEs

- *The underdamped Langevin stochastic differential equation is Nesterov acceleration on the manifold of probability distributions, with respect to the KL divergence* (Ma, et al., to appear)

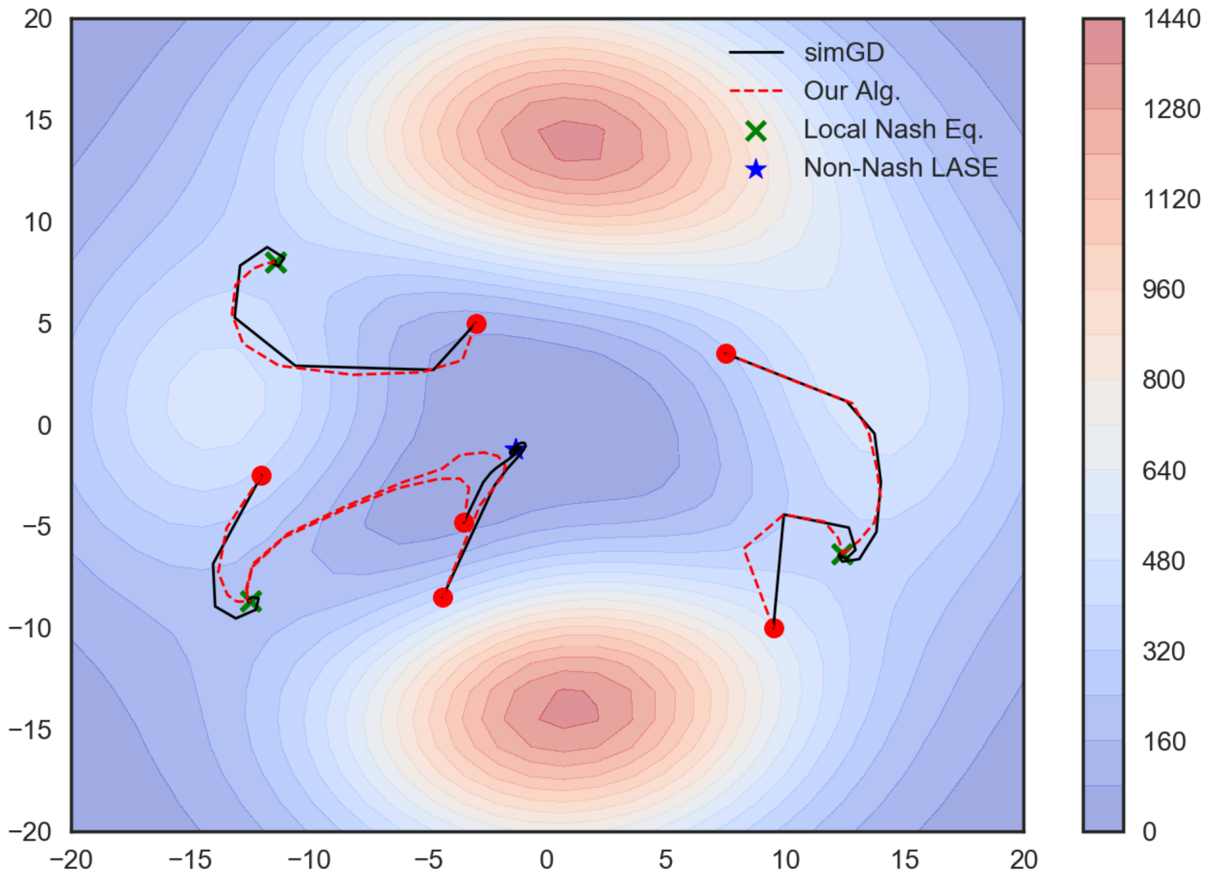
Part VII: Market Design Meets Gradient-Based Learning

with Lydia Liu, Horia Mania and Eric Mazumdar

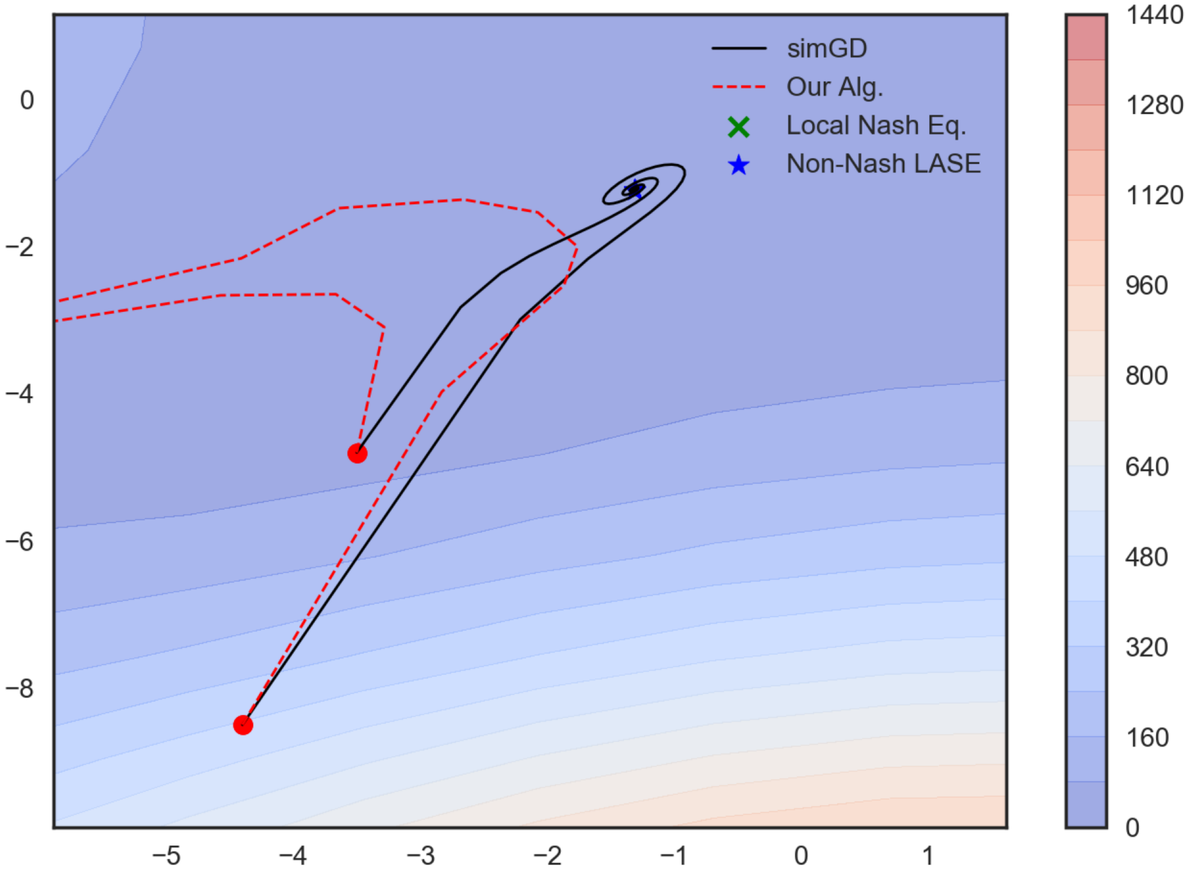
Two Examples of Current Projects

- How to find saddle points in high dimensions?
  - not just any saddle points; we want to find the Nash equilibria (and only the Nash equilibria)

- Competitive bandits and two-way markets
  - how to find the "best action" when supervised training data is not available, when other agents are also searching for best actions, and when there is conflict (e.g., scarcity)



simGD
Our Alg.
Local Nash Eq.
Non-Nash LASE
20
15
10
5
0
−5
−10
−15
−20
−20
−15
−10
−5
0
5
10
15
20
1440
1280
1120
960
800
640
480
320
160
0


simGD
Our Alg.
Local Nash Eq.
Non-Nash LASE
0
−2
−4
−6
−8
−5
−4
−3
−2
−1
0
1
1440
1280
1120
960
800
640
480
320
160
0

Chapter 3: Concluding Remarks

Machine Learning (aka, AI)

- First Generation ('90-'00): the backend
  - e.g., fraud detection, search, supply-chain management
- Second Generation ('00-'10): the human side
  - e.g., recommendation systems, commerce, social media
- Third Generation ('10-now): pattern recognition
  - e.g., speech recognition, computer vision, translation
- Fourth Generation (emerging): decisions and markets
  - not just one agent making a decision or sequence of decisions
  - but a huge interconnected web of data, agents, decisions
  - many new challenges!

- What do these developments have to do with "intelligence"?

AI = Data + Algorithms + Markets

- Computers are currently gathering huge amounts of data, for and about humans, to be fed into learning algorithms
  - often the goal is to learn to imitate humans
  - a related goal is to provide personalized services to humans
  - but there's a lot of guessing going on about what people want
- Services are best provided in the context of a market; market design can eliminate much of the guesswork
  - when data flows in a market, the underlying system can learn from that data, so that the market provides better services
  - fairness arises not from providing the same service to everyone, but by allowing individual utilities to be expressed
- Learning algorithms provide the glue between data and the market

Consequences for IT Business Models

- Many modern IT companies collect data as part of providing a service on a platform
  - often the value provided by these services is limited
  - so the monetization comes from advertising
  - i.e., many companies are in fact creating markets based on data and learning algorithms, but these markets only link the IT company and the advertisers
- Humans are treated as a product, not as a player in a market
  - the results (ads) are not based on the utility (happiness) of the providers of the data, and does not pay them for their data
- This is broken---humans should be able to participate fully in a market in which their data are being used
  - they should not be treated as mere product or mere observers

Executive Summary

- ML (AI) has come of age
- But it is far from being a solid engineering discipline that can yield robust, scalable solutions to modern data-analytic problems
- There are many hard problems involving uncertainty, inference, decision-making, robustness and scale that are far from being solved
  - not to mention economic, social and legal issues