Optimization

Aymeric DIEULEVEUT

EPFL, Lausanne

January 26, 2018

Journées YSP

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Outline

1. **General context and examples.**
2. **What makes optimization hard ?**

Outline

1. General context and examples.
2. What makes optimization hard ?

   In the context of supervised machine learning:

3. Minimizing Empirical Risk.

Outline

1. General context and examples.
2. What makes optimization hard ?

In the context of supervised machine learning:

3. Minimizing Empirical Risk.
4. Minimizing Generalization Risk.

General context

What is optimization about ?

$$\min_{\theta \in \Theta} f(\theta)$$

With θ a parameter, and f a cost function.

General context

What is optimization about ?

$$\min_{\theta \in \Theta} f(\theta)$$

With θ a parameter, and f a cost function.

Why ?

General context

What is optimization about ?

$$\min_{\theta \in \Theta} f(\theta)$$

With θ a parameter, and f a cost function.

Why ?
We formulate our problem as an optimization problem.
3 examples:

- Supervised machine learning
- Signal Processing
- Optimal transport

Some Examples

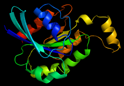
Example 1: Supervised Machine Learning

Goal: predict a phenomenon from "explanatory variables", given a set of observations.

Some Examples

Example 1: Supervised Machine Learning

Goal: predict a phenomenon from "explanatory variables", given a set of observations.



Bio-informatics

Input: DNA/RNA sequence,
Output: Drug responsiveness

Image classification

Input: Images,
Output: Digit

Supervised Machine Learning

Example 1: Supervised Machine Learning

Consider an input/output pair $(X, Y) \in \mathcal{X} \times \mathcal{Y}$, $(X, Y) \sim \rho$.

Goal: function $\theta : \mathcal{X} \to \mathbb{R}$, s.t. $\theta(X)$ good prediction for Y.

Supervised Machine Learning

Example 1: Supervised Machine Learning

Consider an input/output pair $(X, Y) \in \mathcal{X} \times \mathcal{Y}$, $(X, Y) \sim \rho$.

Goal: function $\theta : \mathcal{X} \to \mathbb{R}$, s.t. $\theta(X)$ good prediction for Y.

Here, as a linear function $\langle \theta, \Phi(X) \rangle$ of features $\Phi(X) \in \mathbb{R}^d$.

Supervised Machine Learning

Example 1: Supervised Machine Learning

Consider an input/output pair $(X, Y) \in \mathcal{X} \times \mathcal{Y}$, $(X, Y) \sim \rho$.

Goal: function $\theta : \mathcal{X} \to \mathbb{R}$, s.t. $\theta(X)$ good prediction for Y.

Here, as a linear function $\langle \theta, \Phi(X) \rangle$ of features $\Phi(X) \in \mathbb{R}^d$.

Consider a loss function $\ell : \mathcal{Y} \times \mathbb{R} \to \mathbb{R}_+$

Define the Generalization risk :

$$\mathcal{R}(\theta) := \mathbb{E}_\rho \left[\ell(Y, \langle \theta, \Phi(X) \rangle) \right].$$

Empirical Risk minimization (I)

Data: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \ldots, n$, i.i.d.

Empirical risk (or training error):

$$\hat{\mathcal{R}}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, \langle \theta, \Phi(x_i) \rangle).$$

Empirical Risk minimization (I)

Data: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \ldots, n$, i.i.d.

Empirical risk (or training error):

$$\hat{\mathcal{R}}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, \langle \theta, \Phi(x_i) \rangle).$$

Empirical risk minimization (ERM) : find $\hat{\theta}$ solution of

$$\min_{\theta \in \mathbb{R}^d} \quad \frac{1}{n} \sum_{i=1}^{n} \ell\big(y_i, \langle \theta, \Phi(x_i) \rangle\big) \quad + \quad \mu \Omega(\theta).$$

convex data fitting term + regularizer

Empirical Risk minimization (II)

For example, least-squares regression:

$$\min_{\theta \in \mathbb{R}^d} \quad \frac{1}{2n} \sum_{i=1}^{n} \big(y_i - \langle \theta, \Phi(x_i) \rangle\big)^2 \quad + \quad \mu \Omega(\theta),$$

Empirical Risk minimization (II)

For example, least-squares regression:

$$\min_{\theta \in \mathbb{R}^d} \quad \frac{1}{2n} \sum_{i=1}^{n} \big(y_i - \langle \theta, \Phi(x_i) \rangle\big)^2 \quad + \quad \mu \Omega(\theta),$$

and logistic regression:

$$\min_{\theta \in \mathbb{R}^d} \quad \frac{1}{n} \sum_{i=1}^{n} \log \big(1 + \exp(-y_i \langle \theta, \Phi(x_i) \rangle)\big) \quad + \quad \mu \Omega(\theta).$$

Some Examples

Example 2: Signal processing

Observe a signal $Y \in \mathbb{R}^{n \times q}$, try to recover the source $B \in \mathbb{R}^{p \times q}$, knowing the "forward matrix" $X \in \mathbb{R}^{n \times p}$. (multi-task regression)

$$\min_{\beta} \|X\beta - Y\|_F^2$$

Some Examples

Example 2: Signal processing

Observe a signal $Y \in \mathbb{R}^{n \times q}$, try to recover the source $B \in \mathbb{R}^{p \times q}$, knowing the “forward matrix” $X \in \mathbb{R}^{n \times p}$. (multi-task regression)

$$\min_{\beta} \|X\beta - Y\|_F^2 \quad + \quad \lambda\Omega(\beta)$$

Ω sparsity inducing regularization.

Some Examples

Example 2: Signal processing

Observe a signal $Y \in \mathbb{R}^{n \times q}$, try to recover the source $B \in \mathbb{R}^{p \times q}$, knowing the "forward matrix" $X \in \mathbb{R}^{n \times p}$.
(multi-task regression)

$$\min_{\beta} \|X\beta - Y\|_F^2 \quad + \quad \lambda \Omega(\beta)$$

Ω sparsity inducing regularization.

How to choose λ?

Some Examples

Example 3: Optimal transport

$$\min_{\pi \in \Pi} \int c(x,y) \mathrm{d}\pi(x,y)$$

Π set of probability distributions $c(x,y)$ “distance” from x to y.

$+$ regularization

Kantorovic formulation of OT.

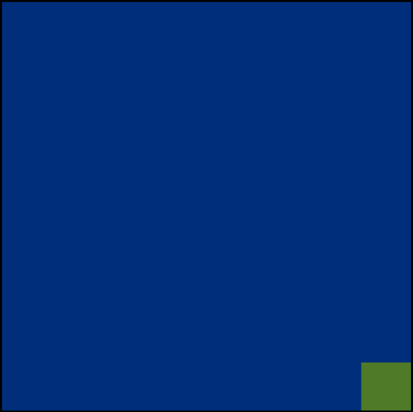
Is it a (hard) problem?

for convex optimization, in 99 % of the cases, no.

Is it a (hard) problem?

for convex optimization, in 99 % of the cases, no.

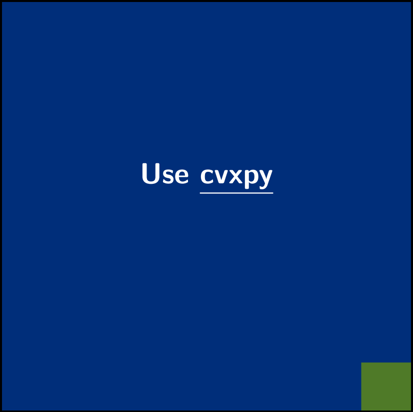
In other words:



Is it a (hard) problem?

for convex optimization, in 99 % of the cases, no.

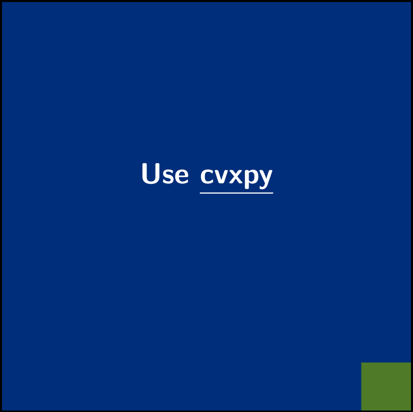
In other words:



Is it a (hard) problem?

for convex optimization, in 99 % of the cases, no.

In other words:

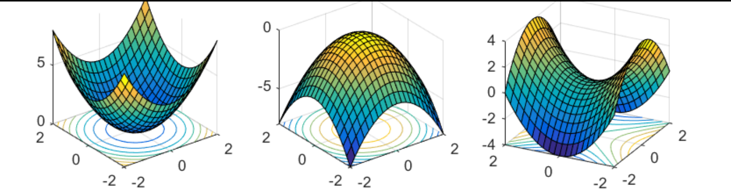


⇈

Interesting (or hard) problems

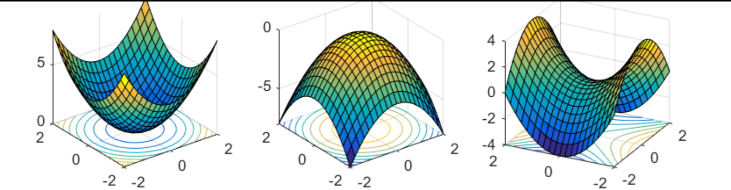
What makes it hard: 1. Convexity

Why?



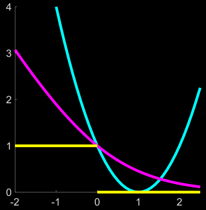
What makes it hard: 1. Convexity

Why?



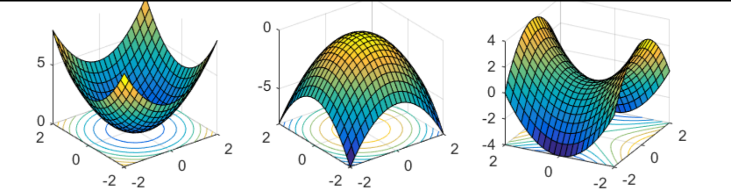
Typical non-convex problems:

Empirical risk minimization with 0-1 loss.



What makes it hard: 1. Convexity

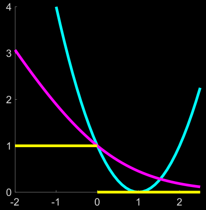
Why?



Typical non-convex problems:

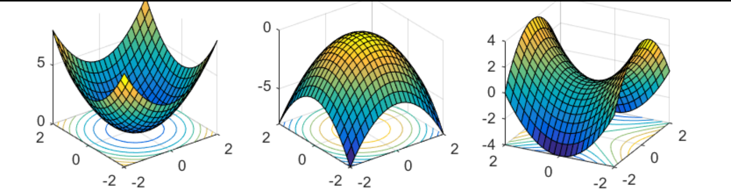
Empirical risk minimization with 0-1 loss.

$$\hat{\mathcal{R}}(\theta) = \frac{1}{n}\sum_{i=1}^{n} 1_{y_i \neq \mathrm{sign}\langle \theta, \Phi(x_i)\rangle}.$$



What makes it hard: 1. Convexity

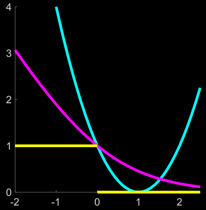
Why?



Typical non-convex problems:

Empirical risk minimization with 0-1 loss.

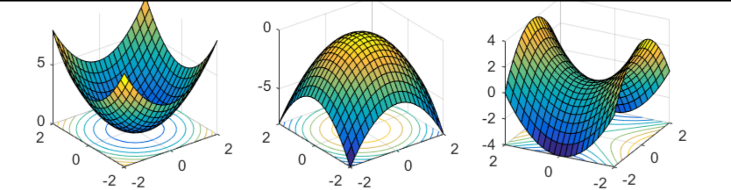
$$\hat{\mathcal{R}}(\theta) = \frac{1}{n}\sum_{i=1}^{n} 1_{y_i \neq \text{sign}\langle\theta,\Phi(x_i)\rangle}.$$



Matrix factorization $\min_{Y,W} \|X - YW\|_F^2$
↬ **not jointly convex.**

What makes it hard: 1. Convexity

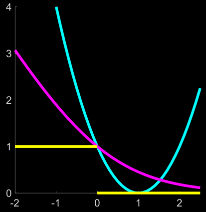
Why?



Typical non-convex problems:

Empirical risk minimization with 0-1 loss.

$$\hat{\mathcal{R}}(\theta) = \frac{1}{n} \sum_{i=1}^{n} 1_{y_i \neq \mathrm{sign}\langle \theta, \Phi(x_i) \rangle}.$$



Matrix factorization $\min_{Y,W} \|X - YW\|_F^2$
↬ **not jointly convex.**

Neural networks: parametric non-convex functions.

What makes it hard: 2. Regularity of the function

a. Smoothness

- A function $g : \mathbb{R}^d \to \mathbb{R}$ is L-smooth if and only if it is twice differentiable and

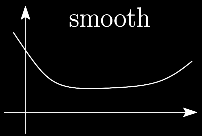
$$\forall \theta \in \mathbb{R}^d, \text{ eigenvalues}\big[g''(\theta)\big] \leqslant L$$

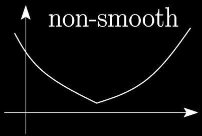
What makes it hard: 2. Regularity of the function

a. Smoothness

- A function $g : \mathbb{R}^d \to \mathbb{R}$ is L-smooth if and only if it is twice differentiable and

$$\forall \theta \in \mathbb{R}^d, \ \text{eigenvalues}\big[g''(\theta)\big] \leqslant L$$





For all $\theta \in \mathbb{R}^d$:

$$g(\theta) \leq g(\theta') + \langle g(\theta'), \theta - \theta' \rangle + L \left\| \theta - \theta' \right\|^2$$

What makes it hard: 2. Regularity of the function

b. Strong Convexity

- A twice differentiable function $g : \mathbb{R}^d \to \mathbb{R}$ is μ-strongly convex if and only if

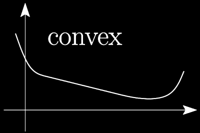
$$\forall \theta \in \mathbb{R}^d, \ \text{eigenvalues}\big[g''(\theta)\big] \geqslant \mu$$

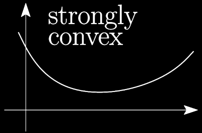
What makes it hard: 2. Regularity of the function

b. Strong Convexity

- A twice differentiable function $g : \mathbb{R}^d \to \mathbb{R}$ is μ-strongly convex if and only if

$$\forall \theta \in \mathbb{R}^d, \text{ eigenvalues}\left[g''(\theta)\right] \geqslant \mu$$





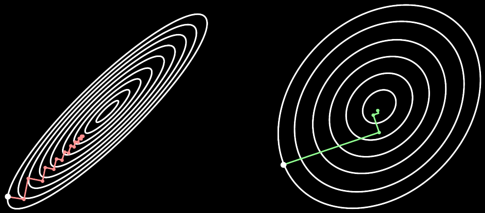
For all $\theta \in \mathbb{R}^d$:

$$g(\theta) \geq g(\theta') + \langle g(\theta'), \theta - \theta' \rangle + \mu \left\|\theta - \theta'\right\|^2$$

What makes it hard: 2. Regularity of the function

Why?

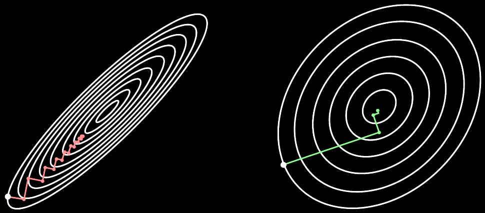
Rates typically depend on the condition number $\kappa = \frac{L}{\mu}$:



What makes it hard: 2. Regularity of the function

Why?

Rates typically depend on the condition number $\kappa = \frac{L}{\mu}$:



Large κ
harder to optimize

Small κ
easier to optimize

Smoothness and strong convexity in ML

We consider an a.s. convex loss in θ. Thus $\hat{\mathcal{R}}$ and $\mathcal{R}$ are convex.

Smoothness and strong convexity in ML

We consider an a.s. convex loss in θ. Thus $\hat{\mathcal{R}}$ and $\mathcal{R}$ are convex.

Hessian of $\hat{\mathcal{R}} \approx$ covariance matrix $\frac{1}{n}\sum_{i=1}^{n}\Phi(x_i)\Phi(x_i)^\top$

Smoothness and strong convexity in ML

We consider an a.s. convex loss in θ. Thus $\hat{\mathcal{R}}$ and $\mathcal{R}$ are convex.

Hessian of $\hat{\mathcal{R}} \approx$ covariance matrix $\frac{1}{n}\sum_{i=1}^{n}\Phi(x_i)\Phi(x_i)^\top$

If ℓ is smooth, and $\mathbb{E}[\|\Phi(X)\|^2] \leq r^2$, $\mathcal{R}$ is smooth.

If ℓ is μ-strongly convex, and data has an invertible covariance matrix (low correlation/dimension), $\mathcal{R}$ is strongly convex.

Smoothness and strong convexity in ML

We consider an a.s. convex loss in θ. Thus $\hat{\mathcal{R}}$ and $\mathcal{R}$ are convex.

Hessian of $\hat{\mathcal{R}} \approx$ covariance matrix $\frac{1}{n}\sum_{i=1}^{n} \Phi(x_i)\Phi(x_i)^\top$

If ℓ is smooth, and $\mathbb{E}[\|\Phi(X)\|^2] \leq r^2$, $\mathcal{R}$ is smooth.

If ℓ is μ-strongly convex, and data has an invertible covariance matrix (low correlation/dimension), $\mathcal{R}$ is strongly convex.

Importance of regularization: provides strong convexity, and avoids overfitting.

Smoothness and strong convexity in ML

We consider an a.s. convex loss in θ. Thus $\hat{\mathcal{R}}$ and $\mathcal{R}$ are convex.

Hessian of $\hat{\mathcal{R}} \approx$ covariance matrix $\frac{1}{n}\sum_{i=1}^{n} \Phi(x_i)\Phi(x_i)^\top$

If ℓ is smooth, and $\mathbb{E}[\|\Phi(X)\|^2] \leq r^2$, $\mathcal{R}$ is smooth.

If ℓ is μ-strongly convex, and data has an invertible covariance matrix (low correlation/dimension), $\mathcal{R}$ is strongly convex.

Importance of regularization: provides strong convexity, and avoids overfitting.

Note: when considering dual formulation of the problem:

- L-smoothness $\leftrightarrow$ $1/L$-strong convexity.
- μ-strong convexity $\leftrightarrow$ $1/\mu$-smoothness

What makes it hard: 3. Set Θ, complexity of f

a. **Set Θ:** (if Θ is a convex set.)

- May be described implicitly (via equations):
  $\Theta = \{\theta \in \mathbb{R}^d$ s.t. $\|\theta\|_2 \leq R$ and $\langle \theta, 1 \rangle = r\}$.

What makes it hard: 3. Set Θ, complexity of f

a. Set Θ: (if Θ is a convex set.)

- May be described implicitly (via equations):
  $\Theta = \{\theta \in \mathbb{R}^d \text{ s.t. } \|\theta\|_2 \leq R \text{ and } \langle \theta, 1 \rangle = r\}$.
  $\looparrowright$ Use dual formulation of the problem.

What makes it hard: 3. Set Θ, complexity of f

a. Set Θ: (if Θ is a convex set.)

- May be described implicitly (via equations):
  $\Theta = \{\theta \in \mathbb{R}^d \text{ s.t. } \|\theta\|_2 \leq R \text{ and } \langle \theta, 1 \rangle = r\}$.
  $\looparrowright$ Use dual formulation of the problem.
- Projection might be difficult or impossible.

What makes it hard: 3. Set Θ, complexity of f

a. Set Θ: (if Θ is a convex set.)

- May be described implicitly (via equations):
  $\Theta = \{\theta \in \mathbb{R}^d \text{ s.t. } \|\theta\|_2 \leq R \text{ and } \langle \theta, 1 \rangle = r\}$.
  $\looparrowright$ Use dual formulation of the problem.
- Projection might be difficult or impossible.
  $\looparrowright$ use algorithms requiring linear minimization oracle instead of quadratic oracles (Frank Wolfe)

What makes it hard: 3. Set Θ, complexity of f

a. Set Θ: (if Θ is a convex set.)

- May be described implicitly (via equations):
  $\Theta = \{\theta \in \mathbb{R}^d \text{ s.t. } \|\theta\|_2 \leq R \text{ and } \langle \theta, 1 \rangle = r\}$.
  $\looparrowright$ Use dual formulation of the problem.
- Projection might be difficult or impossible.
  $\looparrowright$ use algorithms requiring linear minimization oracle instead of quadratic oracles (Frank Wolfe)
- Even when $\Theta = \mathbb{R}^d$, d might be very large (typically millions)

What makes it hard: 3. Set Θ, complexity of f

a. Set Θ: (if Θ is a convex set.)

- May be described implicitly (via equations):
  $\Theta = \{\theta \in \mathbb{R}^d \text{ s.t. } \|\theta\|_2 \leq R \text{ and } \langle \theta, 1 \rangle = r\}$.
  ↬ Use dual formulation of the problem.
- Projection might be difficult or impossible.
  ↬ use algorithms requiring linear minimization oracle instead of quadratic oracles (Frank Wolfe)
- Even when $\Theta = \mathbb{R}^d$, d might be very large (typically millions)
  ↬ use only first order methods

What makes it hard: 3. Set Θ, complexity of f

a. Set Θ: (if Θ is a convex set.)

- May be described implicitly (via equations):
  $\Theta = \{\theta \in \mathbb{R}^d \text{ s.t. } \|\theta\|_2 \leq R \text{ and } \langle \theta, 1 \rangle = r\}$.
  $\looparrowright$ Use dual formulation of the problem.
- Projection might be difficult or impossible.
  $\looparrowright$ use algorithms requiring linear minimization oracle instead of quadratic oracles (Frank Wolfe)
- Even when $\Theta = \mathbb{R}^d$, d might be very large (typically millions)
  $\looparrowright$ use only first order methods

b. Structure of f. If $f = \hat{\mathcal{R}}(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, \langle \theta, \Phi(x_i) \rangle)$, computing a gradient has a cost proportional to n.

Optimization

Take home

- We express problems as minimizing a function over a set
- Most convex problems are solved
- Difficulties come from non-convexity, lack of regularity, complexity of the set Θ (or high dimension), complexity of computing gradients

Optimization

Take home

- We express problems as minimizing a function over a set
- Most convex problems are solved
- Difficulties come from non-convexity, lack of regularity, complexity of the set Θ (or high dimension), complexity of computing gradients

What happens for supervised machine learning ?

Optimization

Take home

- We express problems as minimizing a function over a set
- Most convex problems are solved
- Difficulties come from non-convexity, lack of regularity, complexity of the set Θ (or high dimension), complexity of computing gradients

What happens for supervised machine learning ? Goals:

- present algorithms (convex, large dimension, high number of observations)

Optimization

Take home

- We express problems as minimizing a function over a set
- Most convex problems are solved
- Difficulties come from non-convexity, lack of regularity, complexity of the set Θ (or high dimension), complexity of computing gradients

What happens for supervised machine learning ? Goals:

- present algorithms (convex, large dimension, high number of observations)
- show how rates depend onsmoothness and strong convexity

Optimization

Take home

- We express problems as minimizing a function over a set
- Most convex problems are solved
- Difficulties come from non-convexity, lack of regularity, complexity of the set Θ (or high dimension), complexity of computing gradients

What happens for supervised machine learning ? Goals:

- present algorithms (convex, large dimension, high number of observations)
- show how rates depend onsmoothness and strong convexity
- show how we can use the structure

Optimization

Take home

- We express problems as minimizing a function over a set
- Most convex problems are solved
- Difficulties come from non-convexity, lack of regularity, complexity of the set Θ (or high dimension), complexity of computing gradients

What happens for supervised machine learning ? Goals:

- present algorithms (convex, large dimension, high number of observations)
- show how rates depend onsmoothness and strong convexity
- show how we can use the structure
- not forgetting the initial problem...!

Stochastic algorithms for ERM

$$\min_{\theta \in \mathbb{R}^d} \left\{ \hat{\mathcal{R}}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, \langle \theta, \Phi(x_i) \rangle) \right\}.$$

Two fundamental questions: (a) computing (b) analyzing $\hat{\theta}$.

Stochastic algorithms for ERM

$$\min_{\theta \in \mathbb{R}^d} \left\{ \hat{\mathcal{R}}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, \langle \theta, \Phi(x_i) \rangle) \right\}.$$

Two fundamental questions: (a) computing (b) analyzing $\hat{\theta}$.

"Large scale" framework: number of examples n and the number of explanatory variables d are both large.

1. High dimension d $\Longrightarrow$ First order algorithms

Gradient Descent (GD) :

$$\boxed{\theta_k = \theta_{k-1} - \gamma_k \hat{\mathcal{R}}'(\theta_{k-1})}$$

Stochastic algorithms for ERM

$$\min_{\theta \in \mathbb{R}^d} \left\{ \hat{\mathcal{R}}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, \langle \theta, \Phi(x_i) \rangle) \right\}.$$

Two fundamental questions: (a) computing (b) analyzing $\hat{\theta}$.

"Large scale" framework: number of examples n and the number of explanatory variables d are both large.

1. High dimension $d \implies$ First order algorithms

Gradient Descent (GD) :

$$\boxed{\theta_k = \theta_{k-1} - \gamma_k \hat{\mathcal{R}}'(\theta_{k-1})}$$

Problem: computing the gradient costs $O(dn)$ per iteration.

2. Large $n \implies$ Stochastic algorithms

Stochastic Gradient Descent (SGD)

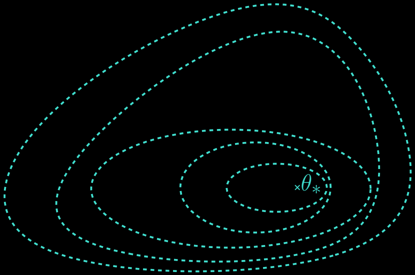
Stochastic Gradient descent

- **Goal:**

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

  given unbiased gradient estimates f'_n

- $\theta_* := \mathrm{argmin}_{\mathbb{R}^d} f(\theta)$.

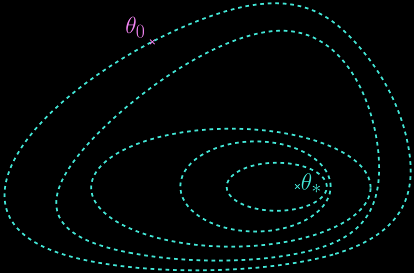


Stochastic Gradient descent

- **Goal:**

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

  given unbiased gradient estimates f'_n



- $\theta_* := \mathrm{argmin}_{\mathbb{R}^d} f(\theta)$.
- Key algorithm: Stochastic Gradient Descent (SGD) (Robbins and Monro, 1951):

$$\boxed{\theta_k = \theta_{k-1} - \gamma_k f'_k(\theta_{k-1})}$$

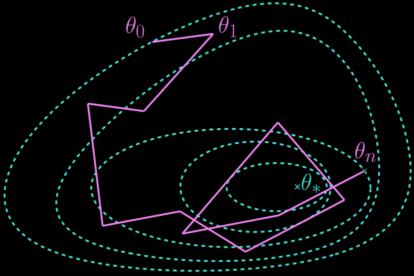
- $\mathbb{E}[f'_k(\theta_{k-1}) | \mathcal{F}_{k-1}] = f'(\theta_{k-1})$ for a filtration $(\mathcal{F}_k)_{k \geq 0}$, θ_k is $\mathcal{F}_k$ measurable.

Stochastic Gradient descent

- **Goal:**

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

  given unbiased gradient estimates f'_n
- $\theta_* := \mathrm{argmin}_{\mathbb{R}^d} f(\theta)$.



- **Key algorithm: Stochastic Gradient Descent (SGD) (Robbins and Monro, 1951):**

$$\boxed{\theta_k = \theta_{k-1} - \gamma_k f'_k(\theta_{k-1})}$$

- $\mathbb{E}[f'_k(\theta_{k-1})|\mathcal{F}_{k-1}] = f'(\theta_{k-1})$ **for a filtration $(\mathcal{F}_k)_{k\geq 0}$, θ_k is $\mathcal{F}_k$ measurable.**

SGD for ERM: $f = \hat{\mathcal{R}}$

Loss for a single pair of observations, for any $j \le n$:

$$f_j(\theta) := \ell(y_j, \langle \theta, \Phi(x_j) \rangle).$$

One observation at each step $\implies$ complexity $O(d)$ per iteration.

SGD for ERM: $f = \hat{\mathcal{R}}$

Loss for a single pair of observations, for any $j \leq n$:

$$f_j(\theta) := \ell(y_j, \langle \theta, \Phi(x_j) \rangle).$$

One observation at each step $\implies$ complexity $O(d)$ per iteration.

For the empirical risk $\hat{\mathcal{R}}(\theta) = \frac{1}{n} \sum_{k=1}^{n} \ell(y_k, \langle \theta, \Phi(x_k) \rangle)$.

- At each step $k \in \mathbb{N}^*$, sample $I_k \sim \mathcal{U}\{1, \ldots n\}$, and use:

$$f'_{I_k}(\theta_{k-1}) = \ell'(y_{I_k}, \langle \theta_{k-1}, \Phi(x_{I_k}) \rangle)$$

SGD for ERM: $f = \hat{\mathcal{R}}$

Loss for a single pair of observations, for any $j \leq n$:

$$f_j(\theta) := \ell(y_j, \langle \theta, \Phi(x_j) \rangle).$$

One observation at each step $\implies$ complexity $O(d)$ per iteration.

For the empirical risk $\hat{\mathcal{R}}(\theta) = \frac{1}{n} \sum_{k=1}^{n} \ell(y_k, \langle \theta, \Phi(x_k) \rangle)$.

- At each step $k \in \mathbb{N}^*$, sample $I_k \sim \mathcal{U}\{1, \ldots n\}$, and use:

$$f'_{I_k}(\theta_{k-1}) = \ell'(y_{I_k}, \langle \theta_{k-1}, \Phi(x_{I_k}) \rangle)$$

$$\mathbb{E}[f'_{I_k}(\theta_{k-1}) | \mathcal{F}_{k-1}] = \frac{1}{n} \sum_{k=1}^{n} \ell'(y_k, \langle \theta, \Phi(x_k) \rangle)$$

SGD for ERM: $f = \hat{\mathcal{R}}$

Loss for a single pair of observations, for any $j \leq n$:

$$f_j(\theta) := \ell(y_j, \langle \theta, \Phi(x_j) \rangle).$$

One observation at each step $\implies$ complexity $O(d)$ per iteration.

For the empirical risk $\hat{\mathcal{R}}(\theta) = \frac{1}{n} \sum_{k=1}^{n} \ell(y_k, \langle \theta, \Phi(x_k) \rangle)$.

- At each step $k \in \mathbb{N}^*$, sample $I_k \sim \mathcal{U}\{1, \dots n\}$, and use:

$$f'_{I_k}(\theta_{k-1}) = \ell'(y_{I_k}, \langle \theta_{k-1}, \Phi(x_{I_k}) \rangle)$$

$$\mathbb{E}[f'_{I_k}(\theta_{k-1}) | \mathcal{F}_{k-1}] = \frac{1}{n} \sum_{k=1}^{n} \ell'(y_k, \langle \theta, \Phi(x_k) \rangle) = \hat{\mathcal{R}}'(\theta_{k-1}).$$

with $\mathcal{F}_k = \sigma((x_i, y_i)_{1 \leq i \leq n}, (I_i)_{1 \leq i \leq k})$.

Analysis: behaviour of $(\theta_n)_{n\geq 0}$

$$\boxed{\theta_k = \theta_{k-1} - \gamma_k f_k'(\theta_{k-1})}$$

Importance of the learning rate $(\gamma_k)_{k\geq 0}$.

For smooth and strongly convex problem, $\theta_k \to \theta_*$ a.s. if

$$\sum_{k=1}^{\infty} \gamma_k = \infty \qquad \sum_{k=1}^{\infty} \gamma_k^2 < \infty.$$

Analysis: behaviour of $(\theta_n)_{n\geq 0}$

$$\boxed{\theta_k = \theta_{k-1} - \gamma_k f_k'(\theta_{k-1})}$$

Importance of the learning rate $(\gamma_k)_{k\geq 0}$.

For smooth and strongly convex problem, $\theta_k \to \theta_*$ a.s. if

$$\sum_{k=1}^{\infty} \gamma_k = \infty \qquad \sum_{k=1}^{\infty} \gamma_k^2 < \infty.$$

And asymptotic normality $\sqrt{k}(\theta_k - \theta_*) \overset{d}{\to} \mathcal{N}(0, V)$, for $\gamma_k = \frac{\gamma_0}{k}$, $\gamma_0 \geq \frac{1}{\mu}$.

Analysis: behaviour of $(\theta_n)_{n\geq 0}$

$$\boxed{\theta_k = \theta_{k-1} - \gamma_k f_k'(\theta_{k-1})}$$

Importance of the learning rate $(\gamma_k)_{k\geq 0}$.

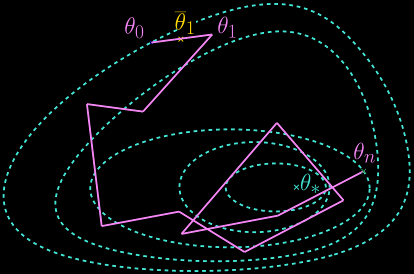
For smooth and strongly convex problem, $\theta_k \to \theta_*$ a.s. if

$$\sum_{k=1}^{\infty} \gamma_k = \infty \qquad \sum_{k=1}^{\infty} \gamma_k^2 < \infty.$$

And asymptotic normality $\sqrt{k}(\theta_k - \theta_*) \stackrel{d}{\to} \mathcal{N}(0, V)$, for $\gamma_k = \frac{\gamma_0}{k}$, $\gamma_0 \geq \frac{1}{\mu}$.

- Limit variance scales as $1/\mu^2$
- Very sensitive to ill-conditioned problems.
- μ generally unknown...

Polyak Ruppert averaging



Introduced by Polyak and Juditsky (1992) and Ruppert (1988):

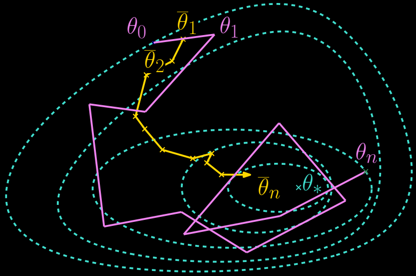
$$\bar{\theta}_k = \frac{1}{k+1}\sum_{i=0}^{k}\theta_i.$$

- **off line averaging reduces the noise effect.**

Polyak Ruppert averaging

Introduced by Polyak and Juditsky (1992) and Ruppert (1988):

$$\bar{\theta}_k = \frac{1}{k+1}\sum_{i=0}^{k}\theta_i.$$



- off line averaging reduces the noise effect.
- on line computing: $\bar{\theta}_{k+1} = \frac{1}{k+1}\theta_{k+1} + \frac{k}{k+1}\bar{\theta}_k$.

Convex stochastic approximation: convergence

Known global minimax rates for non-smooth problems

- Strongly convex: $O((\mu k)^{-1})$
  Attained by averaged stochastic gradient descent with $\gamma_k \propto (\mu k)^{-1}$
- Non-strongly convex: $O(k^{-1/2})$
  Attained by averaged stochastic gradient descent with $\gamma_k \propto k^{-1/2}$

Convex stochastic approximation: convergence

Known global minimax rates for non-smooth problems

- **Strongly convex:** $O((\mu k)^{-1})$
  Attained by averaged stochastic gradient descent with $\gamma_k \propto (\mu k)^{-1}$
- **Non-strongly convex:** $O(k^{-1/2})$
  Attained by averaged stochastic gradient descent with $\gamma_k \propto k^{-1/2}$

For smooth problems

- **Strongly convex:** $O(\mu k)^{-1}$
  for $\gamma_k \propto k^{-1/2}$: adapts to strong convexity.

Convergence rate for $f(\tilde{\theta}_k) - f(\theta_*)$, smooth f.

	$\min \hat{\mathcal{R}}$	
	SGD	GD
Convex	$O\left(\frac{1}{\sqrt{k}}\right)$	$O\left(\frac{1}{k}\right)$

Convergence rate for $f(\tilde{\theta}_k) - f(\theta_*)$, smooth f.

min $\hat{\mathcal{R}}$

	SGD	GD
Convex	$O\left(\frac{1}{\sqrt{k}}\right)$	$O\left(\frac{1}{k}\right)$
Stgly-Cvx	$O\left(\frac{1}{\mu k}\right)$	$O(e^{-\mu k})$

Convergence rate for $f(\tilde{\theta}_k) - f(\theta_*)$, smooth f.

$\min \hat{\mathcal{R}}$

	SGD	GD
Convex	$O\left(\frac{1}{\sqrt{k}}\right)$	$O\left(\frac{1}{k}\right)$
Stgly-Cvx	$O\left(\frac{1}{\mu k}\right)$	$O(e^{-\mu k})$

$\ominus$ Gradient descent update costs n times as much as SGD update.

Convergence rate for $f(\tilde{\theta}_k) - f(\theta_*)$, smooth f.

	$\min \hat{\mathcal{R}}$	
	SGD	**GD**
Convex	$O\left(\frac{1}{\sqrt{k}}\right)$	$O\left(\frac{1}{k}\right)$
Stgly-Cvx	$O\left(\frac{1}{\mu k}\right)$	$O(e^{-\mu k})$

$\ominus$ **Gradient descent update costs n times as much as SGD update.**

Can we get best of both worlds ?

Methods for finite sum minimization

- GD: at step k, use $\frac{1}{n}\sum_{i=0}^{n} f_i'(\theta_k)$

Methods for finite sum minimization

- GD: at step k, use $\frac{1}{n}\sum_{i=0}^{n} f_i'(\theta_k)$
- SGD: at step k, sample $i_k \sim \mathcal{U}[1;n]$, use $f_{i_k}'(\theta_k)$

Methods for finite sum minimization

- GD: at step k, use $\frac{1}{n}\sum_{i=0}^{n} f_i'(\theta_k)$
- SGD: at step k, sample $i_k \sim \mathcal{U}[1; n]$, use $f_{i_k}'(\theta_k)$
- **SAG**: at step k,
  - keep a "full gradient" $\frac{1}{n}\sum_{i=0}^{n} f_i'(\theta_{k_i})$, with $\theta_{k_i} \in \{\theta_1, \dots \theta_k\}$

Methods for finite sum minimization

- GD: at step k, use $\frac{1}{n}\sum_{i=0}^{n} f_i'(\theta_k)$
- SGD: at step k, sample $i_k \sim \mathcal{U}[1;n]$, use $f_{i_k}'(\theta_k)$
- SAG: at step k,
  - keep a "full gradient" $\frac{1}{n}\sum_{i=0}^{n} f_i'(\theta_{k_i})$, with $\theta_{k_i} \in \{\theta_1, \dots \theta_k\}$
  - sample $i_k \sim \mathcal{U}[1;n]$, use

$$\frac{1}{n}\left(\sum_{i=0}^{n} f_i'(\theta_{k_i}) - f_{i_k}'(\theta_{k_{i_k}}) + f_{i_k}'(\theta_k)\right),$$

Methods for finite sum minimization

- GD: at step k, use $\frac{1}{n}\sum_{i=0}^{n} f_i'(\theta_k)$
- SGD: at step k, sample $i_k \sim \mathcal{U}[1;n]$, use $f_{i_k}'(\theta_k)$
- SAG: at step k,
  - keep a "full gradient" $\frac{1}{n}\sum_{i=0}^{n} f_i'(\theta_{k_i})$, with $\theta_{k_i} \in \{\theta_1, \dots \theta_k\}$
  - sample $i_k \sim \mathcal{U}[1;n]$, use

$$\frac{1}{n}\left(\sum_{i=0}^{n} f_i'(\theta_{k_i}) - f_{i_k}'(\theta_{k_{i_k}}) + f_{i_k}'(\theta_k)\right),$$

↬ ⊕ update costs the same as SGD

↬ ⊖ needs to store all gradients $f_i'(\theta_{k_i})$ at "points in the past"

Some references:

- SAG Schmidt et al. (2013), SAGA Defazio et al. (2014a)
- SVRG Johnson and Zhang (2013) (reduces memory cost but 2 epochs...)
- FINITO Defazio et al. (2014b)
- S2GD Konečný and Richtárik (2013)...

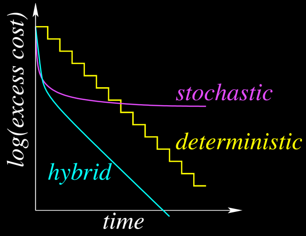
And many others... See for example Niao He's lecture notes for a nice overview.

Convergence rate for $f(\tilde{\theta}_k) - f(\theta_*)$, smooth objective f.

		$\min \hat{\mathcal{R}}$	
	SGD	GD	SAG
Convex	$O\left(\frac{1}{\sqrt{k}}\right)$	$O\left(\frac{1}{k}\right)$	

Convergence rate for $f(\tilde{\theta}_k) - f(\theta_*)$, smooth objective f.

	$\min \hat{\mathcal{R}}$		
	SGD	GD	SAG
Convex	$O\left(\frac{1}{\sqrt{k}}\right)$	$O\left(\frac{1}{k}\right)$	
Stgly-Cvx	$O\left(\frac{1}{\mu k}\right)$	$O(e^{-\mu k})$	$O\left(1 - (\mu \wedge \frac{1}{n})\right)^k$



GD, SGD, SAG (Fig. from Schmidt et al. (2013))

Take home

Stochastic algorithms for Empirical Risk Minimization.

- **Rates depend on the regularity of the function.**
- **Several algorithms to optimize empirical risk, most efficient ones are stochastic and rely on finite sum structure**

Take home

Stochastic algorithms for Empirical Risk Minimization.

- Rates depend on the **regularity of the function.**
- **Several algorithms** to optimize empirical risk, most efficient ones are **stochastic** and rely on **finite sum structure**
- **Stochastic algorithms** to optimize a **deterministic function**.

What about generalization risk

Initial problem: Generalization guarantees.

- Uniform upper bound $\sup_\theta \left| \hat{\mathcal{R}}(\theta) - \mathcal{R}(\theta) \right|$. (empirical process theory)
- More precise: localized complexities (Bartlett et al., 2002), stability (Bousquet and Elisseeff, 2002).

What about generalization risk

Initial problem: Generalization guarantees.

- Uniform upper bound $\sup_\theta \left| \hat{\mathcal{R}}(\theta) - \mathcal{R}(\theta) \right|$. (empirical process theory)
- More precise: localized complexities (Bartlett et al., 2002), stability (Bousquet and Elisseeff, 2002).

Problems for ERM:

- Choose regularization (overfitting risk)
- How many iterations (i.e., passes on the data)?
- Generalization guarantees generally of order $O(1/\sqrt{n})$, no need to be precise

What about generalization risk

Initial problem: **Generalization guarantees.**

- Uniform upper bound $\sup_\theta \left| \hat{\mathcal{R}}(\theta) - \mathcal{R}(\theta) \right|$. (empirical process theory)
- More precise: localized complexities (Bartlett et al., 2002), stability (Bousquet and Elisseeff, 2002).

Problems for ERM:

- Choose regularization (overfitting risk)
- How many iterations (i.e., passes on the data)?
- Generalization guarantees generally of order $O(1/\sqrt{n})$, no need to be precise

2 important insights:

1. No need to optimize below statistical error,
2. Generalization risk is more important than empirical risk.

What about generalization risk

Initial problem: Generalization guarantees.

- Uniform upper bound $\sup_\theta \left| \hat{\mathcal{R}}(\theta) - \mathcal{R}(\theta) \right|$. (empirical process theory)
- More precise: localized complexities (Bartlett et al., 2002), stability (Bousquet and Elisseeff, 2002).

Problems for ERM:

- Choose regularization (overfitting risk)
- How many iterations (i.e., passes on the data)?
- Generalization guarantees generally of order $O(1/\sqrt{n})$, no need to be precise

2 important insights:

1. No need to optimize below statistical error,
2. Generalization risk is more important than empirical risk.

SGD can be used to minimize the generalization risk.

SGD for the generalization risk: $f = \mathcal{R}$

SGD: key assumption $\mathbb{E}[f_n'(\theta_{n-1})|\mathcal{F}_{n-1}] = f'(\theta_{n-1})$.

SGD for the generalization risk: $f = \mathcal{R}$

SGD: key assumption $\mathbb{E}[f_n'(\theta_{n-1})|\mathcal{F}_{n-1}] = f'(\theta_{n-1})$.

For the risk

$$\mathcal{R}(\theta) = \mathbb{E}_\rho\left[\ell(Y, \langle\theta, \Phi(X)\rangle)\right]$$

- At step $0 < k \leq n$, use a new point independent of θ_{k-1}:

$$f_k'(\theta_{k-1}) = \ell'(y_k, \langle\theta_{k-1}, \Phi(x_k)\rangle)$$

SGD for the generalization risk: $f = \mathcal{R}$

SGD: key assumption $\mathbb{E}[f_n'(\theta_{n-1})|\mathcal{F}_{n-1}] = f'(\theta_{n-1})$.

For the risk

$$\mathcal{R}(\theta) = \mathbb{E}_\rho\left[\ell(Y, \langle \theta, \Phi(X) \rangle)\right]$$

- At step $0 < k \leq n$, use a new point independent of θ_{k-1}:

$$f_k'(\theta_{k-1}) = \ell'(y_k, \langle \theta_{k-1}, \Phi(x_k) \rangle)$$

- For $0 \leq k \leq n$, $\mathcal{F}_k = \sigma((x_i, y_i)_{1 \leq i \leq k})$.

$$\mathbb{E}[f_k'(\theta_{k-1})|\mathcal{F}_{k-1}] \quad = \quad \mathbb{E}_\rho[\ell'(y_k, \langle \theta_{k-1}, \Phi(x_k) \rangle)|\mathcal{F}_{k-1}]$$

SGD for the generalization risk: $f = \mathcal{R}$

SGD: key assumption $\mathbb{E}[f_n'(\theta_{n-1})|\mathcal{F}_{n-1}] = f'(\theta_{n-1})$.

For the risk

$$\mathcal{R}(\theta) = \mathbb{E}_\rho\left[\ell(Y, \langle\theta, \Phi(X)\rangle)\right]$$

- At step $0 < k \leq n$, use a new point independent of θ_{k-1}:

$$f_k'(\theta_{k-1}) = \ell'(y_k, \langle\theta_{k-1}, \Phi(x_k)\rangle)$$

- For $0 \leq k \leq n$, $\mathcal{F}_k = \sigma((x_i, y_i)_{1\leq i\leq k})$.

$$\begin{aligned}\mathbb{E}[f_k'(\theta_{k-1})|\mathcal{F}_{k-1}] &= \mathbb{E}_\rho[\ell'(y_k, \langle\theta_{k-1}, \Phi(x_k)\rangle)|\mathcal{F}_{k-1}] \\ &= \mathbb{E}_\rho\left[\ell'(Y, \langle\theta_{k-1}, \Phi(X)\rangle)\right] = \mathcal{R}'(\theta_{k-1})\end{aligned}$$

SGD for the generalization risk: $f = \mathcal{R}$

SGD: key assumption $\mathbb{E}[f_n'(\theta_{n-1})|\mathcal{F}_{n-1}] = f'(\theta_{n-1})$.

For the risk

$$\mathcal{R}(\theta) = \mathbb{E}_\rho\left[\ell(Y, \langle \theta, \Phi(X)\rangle)\right]$$

- At step $0 < k \leq n$, use a new point independent of θ_{k-1}:

$$f_k'(\theta_{k-1}) = \ell'(y_k, \langle \theta_{k-1}, \Phi(x_k)\rangle)$$

- For $0 \leq k \leq n$, $\mathcal{F}_k = \sigma((x_i, y_i)_{1 \leq i \leq k})$.

$$\begin{aligned} \mathbb{E}[f_k'(\theta_{k-1})|\mathcal{F}_{k-1}] &= \mathbb{E}_\rho[\ell'(y_k, \langle \theta_{k-1}, \Phi(x_k)\rangle)|\mathcal{F}_{k-1}] \\ &= \mathbb{E}_\rho\left[\ell'(Y, \langle \theta_{k-1}, \Phi(X)\rangle)\right] = \mathcal{R}'(\theta_{k-1}) \end{aligned}$$

- Single pass through the data, Running-time $= O(nd)$,
- "Automatic" regularization.

SGD for the generalization risk: $f = \mathcal{R}$

SGD: key assumption $\mathbb{E}[f_n'(\theta_{n-1})|\mathcal{F}_{n-1}] = f'(\theta_{n-1})$.

For the risk

$$\mathcal{R}(\theta) = \mathbb{E}_\rho\left[\ell(Y, \langle\theta, \Phi(X)\rangle)\right]$$

- At step $0 < k \leq n$, use a new point independent of θ_{k-1}:

$$f_k'(\theta_{k-1}) = \ell'(y_k, \langle\theta_{k-1}, \Phi(x_k)\rangle)$$

- For $0 \leq k \leq n$, $\mathcal{F}_k = \sigma((x_i, y_i)_{1\leq i\leq k})$.

$$\begin{aligned}\mathbb{E}[f_k'(\theta_{k-1})|\mathcal{F}_{k-1}] &= \mathbb{E}_\rho[\ell'(y_k, \langle\theta_{k-1}, \Phi(x_k)\rangle)|\mathcal{F}_{k-1}] \\ &= \mathbb{E}_\rho\left[\ell'(Y, \langle\theta_{k-1}, \Phi(X)\rangle)\right] = \mathcal{R}'(\theta_{k-1})\end{aligned}$$

- Single pass through the data, Running-time $= O(nd)$,
- "Automatic" regularization.

SGD for the generalization risk: $f = \mathcal{R}$

	ERM minimization	Gen. risk minimization
	several passes : $0 \leq k$	One pass $0 \leq k \leq n$
x_i, y_i is	$\mathcal{F}_t$-measurable for any t	$\mathcal{F}_t$-measurable for $t \geq i$.

Convergence rate for $f(\tilde{\theta}_k) - f(\theta_*)$, smooth objective f.

	$\min \hat{\mathcal{R}}$			$\min \mathcal{R}$
	SGD	GD	SAG	SGD
Convex	$O\left(\frac{1}{\sqrt{k}}\right)$	$O\left(\frac{1}{k}\right)$		$O\left(\frac{1}{\sqrt{k}}\right)$

Convergence rate for $f(\tilde{\theta}_k) - f(\theta_*)$, smooth objective f.

	$\min \hat{\mathcal{R}}$			$\min \mathcal{R}$
	SGD	GD	SAG	SGD
Convex	$O\left(\frac{1}{\sqrt{k}}\right)$	$O\left(\frac{1}{k}\right)$		$O\left(\frac{1}{\sqrt{k}}\right)$
Stgly-Cvx	$O\left(\frac{1}{\mu k}\right)$	$O(e^{-\mu k})$	$O\left(1-(\mu \wedge \frac{1}{n})\right)^k$	$O\left(\frac{1}{\mu k}\right)$

Convergence rate for $f(\tilde{\theta}_k) - f(\theta_*)$, smooth objective f.

	$\min \hat{\mathcal{R}}$			$\min \mathcal{R}$
	SGD	GD	SAG	SGD
Convex	$O\left(\frac{1}{\sqrt{k}}\right)$	$O\left(\frac{1}{k}\right)$		$O\left(\frac{1}{\sqrt{n}}\right)$
Stgly-Cvx	$O\left(\frac{1}{\mu k}\right)$	$O(e^{-\mu k})$	$O\left(1 - (\mu \wedge \frac{1}{n})\right)^k$	$O\left(\frac{1}{\mu n}\right)$
		$0 \leq k$		$0 \leq k \leq n$

Convergence rate for $f(\tilde{\theta}_k) - f(\theta_*)$, smooth objective f.

	$\min \hat{\mathcal{R}}$			$\min \mathcal{R}$
	SGD	GD	SAG	SGD
Convex	$O\left(\frac{1}{\sqrt{k}}\right)$	$O\left(\frac{1}{k}\right)$		$O\left(\frac{1}{\sqrt{n}}\right)$
Stgly-Cvx	$O\left(\frac{1}{\mu k}\right)$	$O(e^{-\mu k})$	$O\left(1 - (\mu \wedge \frac{1}{n})\right)^k$	$O\left(\frac{1}{\mu n}\right)$
	$0 \leq k$			$0 \leq k \leq n$

Gradient is unknown

Least Mean Squares: rate independent of μ

Least-squares: $\mathcal{R}(\theta) = \frac{1}{2}\mathbb{E}\big[(Y - \langle \Phi(X), \theta\rangle)^2\big]$

Analysis for averaging and constant step-size $\gamma = 1/(4R^2)$ (Bach and Moulines, 2013)

- Assume $\|\Phi(x_n)\| \leqslant r$ and $|y_n - \langle \Phi(x_n), \theta_*\rangle| \leqslant \sigma$
- No assumption regarding lowest eigenvalues of the Hessian

$$\mathbb{E}\mathcal{R}(\bar{\theta}_n) - \mathcal{R}(\theta_*) \leqslant \frac{4\sigma^2 d}{n} + \frac{\|\theta_0 - \theta_*\|^2}{\gamma n}$$

Least Mean Squares: rate independent of μ

Least-squares: $\mathcal{R}(\theta) = \frac{1}{2}\mathbb{E}\big[(Y - \langle \Phi(X), \theta \rangle)^2\big]$

Analysis for averaging and constant step-size $\gamma = 1/(4R^2)$ (Bach and Moulines, 2013)

- Assume $\|\Phi(x_n)\| \leqslant r$ and $|y_n - \langle \Phi(x_n), \theta_* \rangle| \leqslant \sigma$
- No assumption regarding lowest eigenvalues of the Hessian

$$\mathbb{E}\mathcal{R}(\bar{\theta}_n) - \mathcal{R}(\theta_*) \leqslant \frac{4\sigma^2 d}{n} + \frac{\|\theta_0 - \theta_*\|^2}{\gamma n}$$

- Matches statistical lower bound (Tsybakov, 2003).
- Optimal rate with "large" step sizes

Take home

- SGD can be used to minimize the true risk directly
- Stochastic algorithm to minimize unknown function

Take home

- SGD can be used to minimize the true risk directly
- Stochastic algorithm to minimize unknown function
- No regularization needed, only one pass

Take home

- SGD can be used to minimize the true risk directly
- Stochastic algorithm to minimize unknown function
- No regularization needed, only one pass
- For Least Squares, with constant step, optimal rate .

Further references

Many stochastic algorithms not covered in this talk (coordinate descent, online Newton, composite optimization, non convex learning) ...

- Good introduction: Francis's lecture notes at Orsay
- Book:
  Convex Optimization: Algorithms and Complexity,
  Sébastien Bubeck

Agarwal, A., Bartlett, P. L., Ravikumar, P., and Wainwright, M. J. (2012). Information-theoretic lower bounds on the oracle complexity of stochastic convex optimization. IEEE Transactions on Information Theory, 58(5):3235–3249.

Agarwal, A. and Bottou, L. (2014). A Lower Bound for the Optimization of Finite Sums. ArXiv e-prints.

Arjevani, Y. and Shamir, O. (2016). Dimension-free iteration complexity of finite sum optimization problems. In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R., editors, Advances in Neural Information Processing Systems 29, pages 3540–3548. Curran Associates, Inc.

Bach, F. and Moulines, E. (2013). Non-strongly-convex smooth stochastic approximation with convergence rate O(1/n). Advances in Neural Information Processing Systems (NIPS).

Bartlett, P. L., Bousquet, O., and Mendelson, S. (2002). Localized Rademacher Complexities, pages 44–58. Springer Berlin Heidelberg, Berlin, Heidelberg.

Bousquet, O. and Elisseeff, A. (2002). Stability and generalization. Journal of Machine Learning Research, 2(Mar):499–526.

Defazio, A., Bach, F., and Lacoste-Julien, S. (2014a). Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In Advances in Neural Information Processing Systems, pages 1646–1654.

Defazio, A., Domke, J., and Caetano, T. (2014b). Finito: A faster, permutable incremental gradient method for big data problems. In Proceedings of the 31st international conference on machine learning (ICML-14), pages 1125–1133.

Fabian, V. (1968). On asymptotic normality in stochastic approximation. The Annals of Mathematical Statistics, pages 1327–1332.

Johnson, R. and Zhang, T. (2013). Accelerating stochastic gradient descent using predictive variance reduction. In Advances in neural information processing systems, pages 315–323.

Konečný, J. and Richtárik, P. (2013). Semi-stochastic gradient descent methods. arXiv preprint arXiv:1312.1666.

Nemirovsky, A. S. and Yudin, D. B. (1983). Problem complexity and method efficiency in optimization. A Wiley-Interscience Publication. John Wiley & Sons, Inc., New York. Translated from the Russian and with a preface by E. R. Dawson, Wiley-Interscience Series in Discrete Mathematics.

Nesterov, Y. (2004). Introductory Lectures on Convex Optimization: A Basic Course. Applied Optimization. Springer.

Polyak, B. T. and Juditsky, A. B. (1992). Acceleration of stochastic approximation by averaging. SIAM J. Control Optim., 30(4):838–855.

Robbins, H. and Monro, S. (1951). A stochastic approxiation method. The Annals of mathematical Statistics, 22(3):400–407.

Robbins, H. and Siegmund, D. (1985). A convergence theorem for non negative almost supermartingales and some applications. In Herbert Robbins Selected Papers, pages 111–135. Springer.

Ruppert, D. (1988). Efficient estimations from a slowly convergent Robbins-Monro process. Technical report, Cornell University Operations Research and Industrial Engineering.

Schmidt, M., Le Roux, N., and Bach, F. (2013). Minimizing finite sums with the stochastic average gradient. Mathematical Programming, 162(1-2):83–112.

Tsybakov, A. B. (2003). Optimal rates of aggregation. In Proceedings of the Annual Conference on Computational Learning Theory.