機械学習における最適化理論と学習理論的側面

第一部：近接勾配法と確率的勾配降下法

鈴木大慈

東京大学大学院情報理工学系研究科数理情報学専攻
理研 AIP

2020 年 8 月 6 日
@組合せ最適化セミナー 2020 (COSS2020)

本セミナーのアウトライン

1. 第一部：近接勾配法と確率的最適化 (凸，有限次元)
2. 第二部：非凸最適化と再生核ヒルベルト空間における最適化
3. 第三部：深層学習の最適化 (非凸，無限次元)

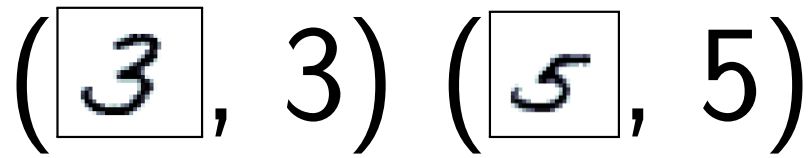
- 汎化誤差を考慮した最適化手法の設計
- シンプルな解法による「軽い」学習の実現：ビッグデータ解析
- 深層学習という解析の難しい対象の最適化理論：非凸最適化に現れる “凸性”

Outline

Outline

機械学習の問題設定

- **教師あり学習**
  データが入力とそれに対するラベルの組で与えられる.
  新しい入力が来た時に対応するラベルを予測する問題.
  問題の例：回帰，判別

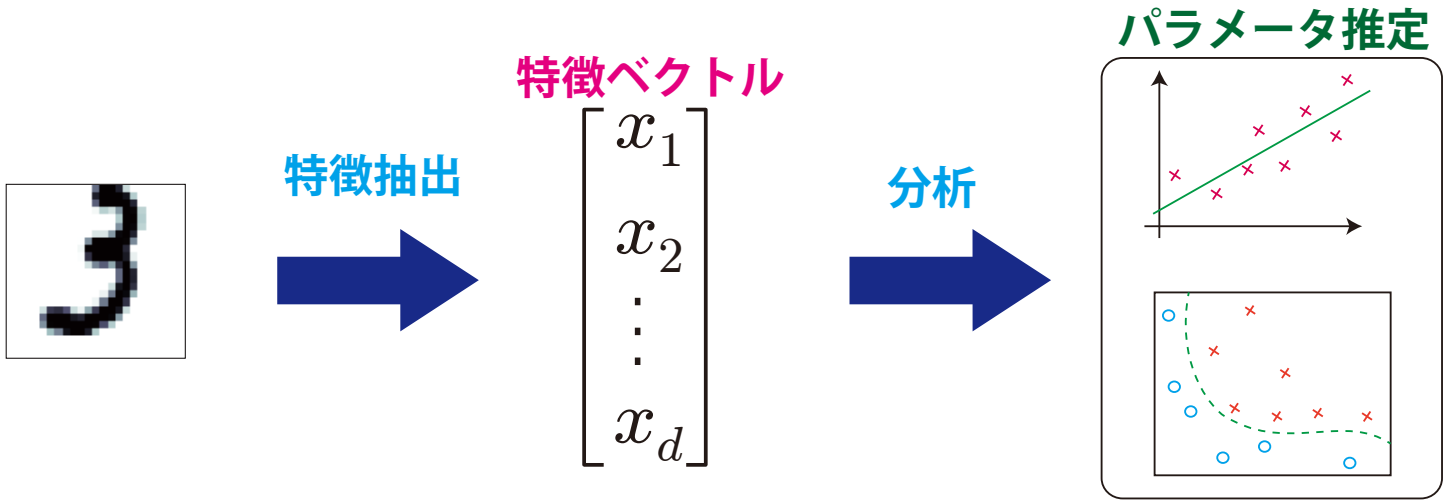


- **教師なし学習**
  データにラベルが付いていない.
  問題の例：クラスタリング，音源分離，異常検知

- **半教師あり学習**
  一部のデータにラベルが付いている.
- **強化学習**
  試行錯誤しながら自分でデータを集める.

機械学習の流れ

- **特徴抽出:** 画像などの対象を何らかの方法でベクトルに変換．(分野ごとのノウハウ)
- 一度特徴ベクトルに変換してしまえばあとは統計の問題．



予測モデルの構築 (θ: モデルのパラメータ)

（教師有り学習） $y = f(x; \theta)$

※深層学習は特徴抽出の部分をネットワーク構造を工夫することで学習に組み込んでいる．

損失関数を用いた定式化

教師あり/なし学習，いずれも損失関数の最小化として定式化できる．

- データの構造を表すパラメータ $\theta \in \Theta$ (Θ は仮説集合 (モデル))
  ← 「学習」$\approx \theta$ の推定
- 損失関数: パラメータ θ がデータ z をどれだけよく説明しているか;

$$\ell(z, \theta).$$

汎化誤差 (期待誤差)：損失の期待値 → 汎化誤差最小化 ≈「学習」

$$\min_{\theta \in \Theta} \mathbb{E}_Z[\ell(Z, \theta)].$$

訓練誤差（経験誤差）：観測されたデータで代用,

$$\min_{\theta \in \Theta} \frac{1}{n} \sum_{i=1}^{n} \ell(z_i, \theta).$$

※ 訓練誤差と汎化誤差に差があることが機械学習における最適化の特徴.

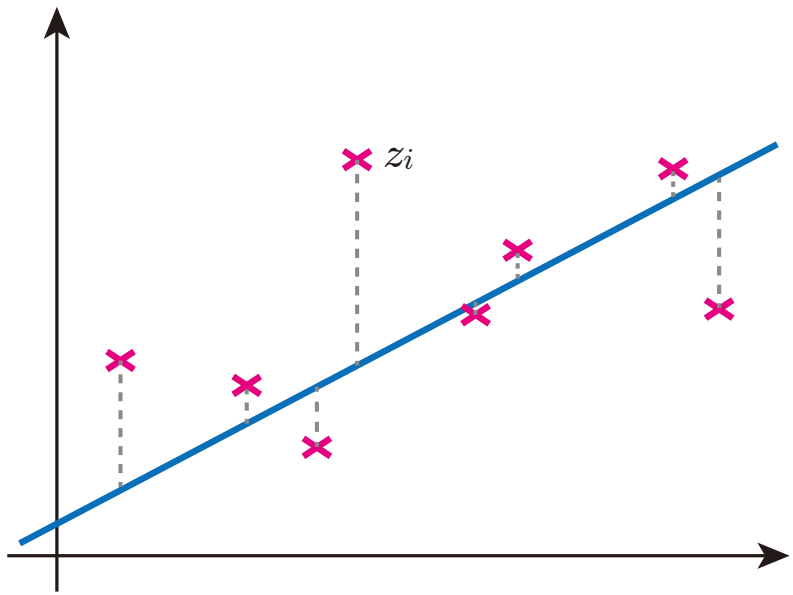
モデルの例 (教師あり)

- 回帰

$$z = (x, y) \in \mathbb{R}^{d+1}$$

$$\ell(z, \theta) = (y - \theta^\top x)^2 \qquad \text{(二乗誤差)}$$

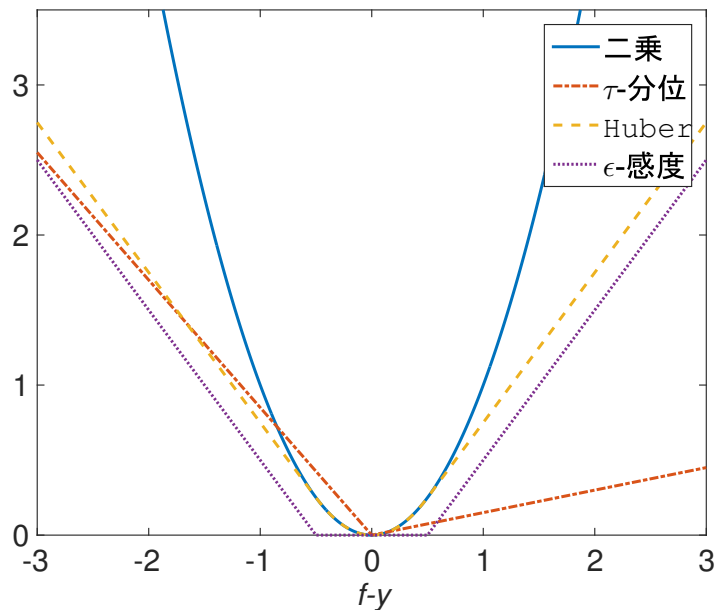
$$\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell(z_i, \theta) = \min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} (y_i - \theta^\top x_i)^2 \quad \text{(最小二乗法)}$$



教師あり学習の損失関数（回帰）

のデータ $z = (x, y)$ における $f = x^\top \theta$ の損失.

- 二乗損失: $\ell(y, f) = \frac{1}{2}(y - f)^2$.
- τ-分位点損失: $\ell(y, f) = (1 - \tau)\max\{f - y, 0\} + \tau \max\{y - f, 0\}$.
  ただし，$\tau \in (0, 1)$. 分位点回帰に用いられる.
- ϵ-感度損失: $\ell(y, f) = \max\{|y - f| - \epsilon, 0\}$,
  ただし，$\epsilon > 0$. サポートベクトル回帰に用いられる.

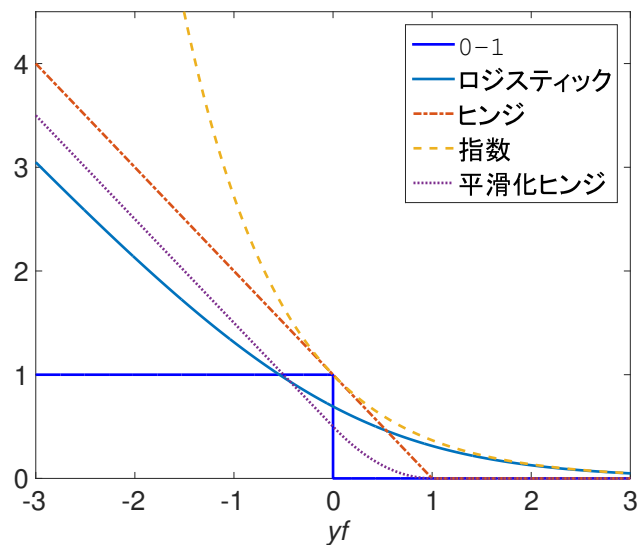


教師あり学習の損失関数（判別）

$y \in \{\pm 1\}$

- ロジスティック損失: $\ell(y, f) = \log((1 + \exp(-yf))/2)$.
- ヒンジ損失: $\ell(y, f) = \max\{1 - yf, 0\}$.
- 指数損失: $\ell(y, f) = \exp(-yf)$.
- 平滑化ヒンジ損失:

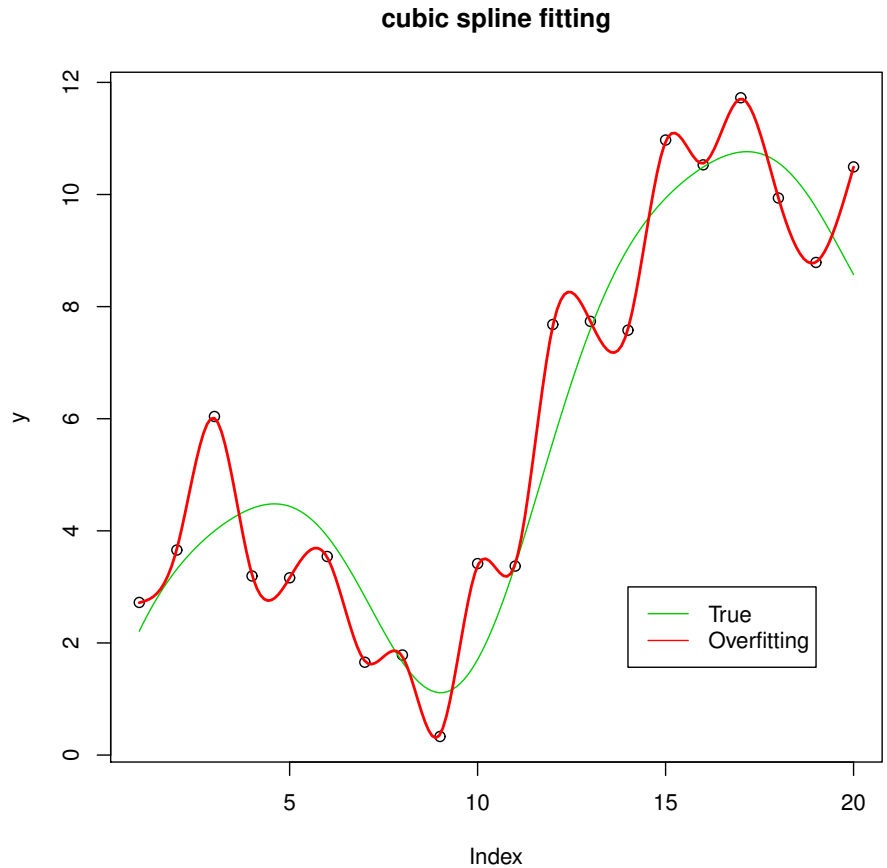
$$\ell(y, f) = \begin{cases} 0, & (yf \geq 1), \\ \frac{1}{2} - yf, & (yf < 0), \\ \frac{1}{2}(1 - yf)^2, & (\text{otherwise}). \end{cases}$$



過学習

経験誤差最小化と汎化誤差最小化には大きなギャップがある.

単なる経験誤差最小化は「過学習」を引き起こす.



正則化法

普通のロス関数 (負の対数尤度) 最小化:

$$\min_{\beta} \sum_{i=1}^{n} \ell(y_i, \beta^\top x_i).$$

正則化付き損失関数最小化:

$$\min_{\beta} \sum_{i=1}^{n} \ell(y_i, \beta^\top x_i) + \underbrace{\psi(\beta)}_{\text{正則化項}}.$$

正則化項の例:

- リッジ正則化 (ℓ_2-正則化): $\psi(\beta) = \lambda \|\beta\|_2^2$
- ℓ_1-正則化: $\psi(\beta) = \lambda \|\beta\|_1$
- トレースノルム正則化: $\psi(W) = \mathrm{Tr}[(W^\top W)^{1/2}]$ $(W \in \mathbb{R}^{N \times M}$: 行列)

正則化法

普通のロス関数 (負の対数尤度) 最小化:

$$\min_{\beta} \sum_{i=1}^{n} \ell(y_i, \beta^\top x_i).$$

正則化付き損失関数最小化:

$$\min_{\beta} \sum_{i=1}^{n} \ell(y_i, \beta^\top x_i) + \underbrace{\psi(\beta)}_{\text{正則化項}}.$$

正則化項の例:

- リッジ正則化 (ℓ_2-正則化): $\psi(\beta) = \lambda \|\beta\|_2^2$
- ℓ_1-正則化: $\psi(\beta) = \lambda \|\beta\|_1$
- トレースノルム正則化: $\psi(W) = \mathrm{Tr}[(W^\top W)^{1/2}]$ ($W \in \mathbb{R}^{N \times M}$: 行列)

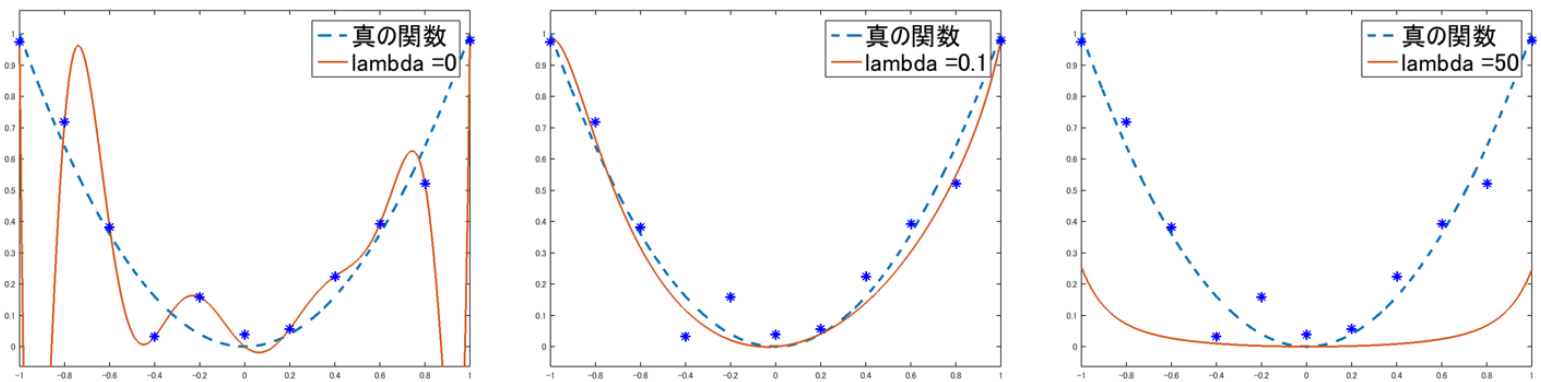
- 正則化項により分散が抑えられ，過学習が防がれる.
- その分，バイアスが乗る.

→ 適切な正則化の強さ (λ) を選ぶ必要がある.

正則化の例： リッジ正則化と過学習

多項式回帰（15 次多項式）

$$\min_{\theta \in \mathbb{R}^{15}} \frac{1}{n} \sum_{i=1}^{n} \{y_i - (\beta_1 x_i + \beta_2 x_i^2 + \cdots + \beta_{15} x_i^{15})\}^2 + \lambda \|\beta\|_2^2$$



$\lambda = 0$ $\lambda = 0.1$ $\lambda = 50$

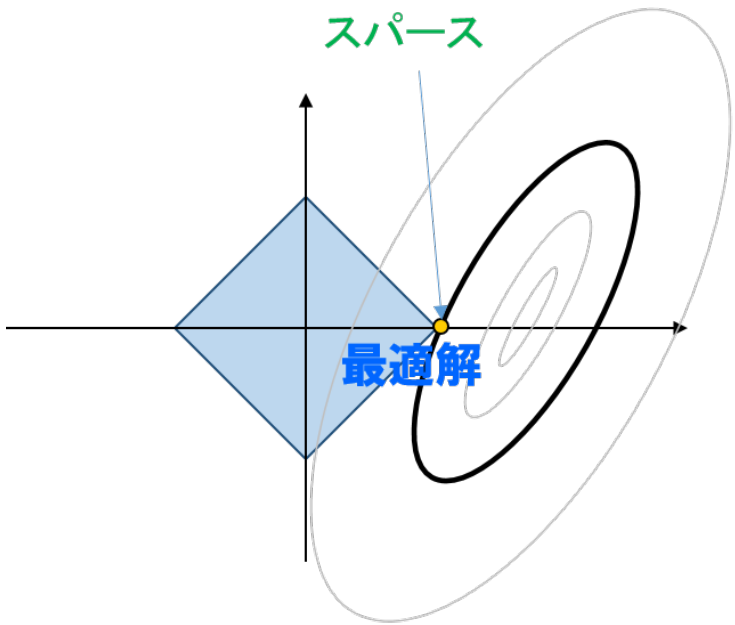
過学習 良い推定 過小学習

正則化の例：ℓ_1-正則化（スパース推定）

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^p}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} (y_i - x_i^\top \beta)^2 + \lambda \sum_{j=1}^{p} |\beta_j|.$$

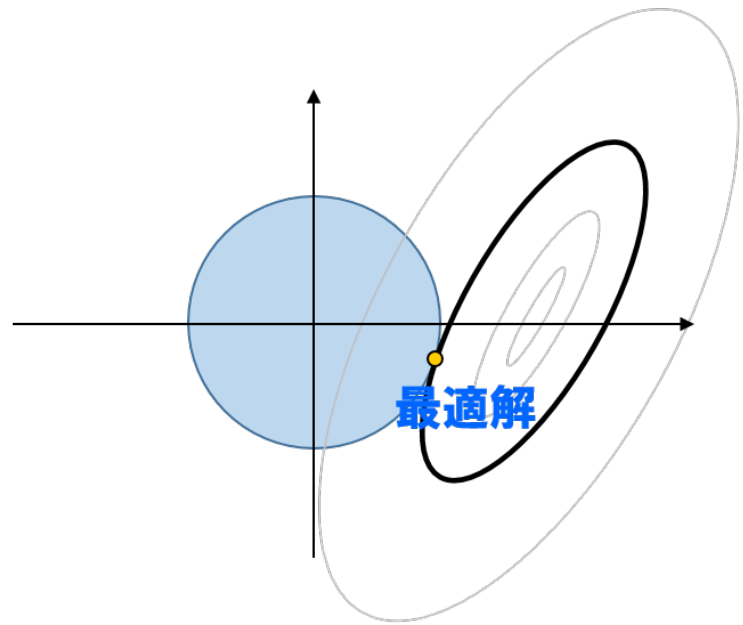
L1正則化

$$\|\beta\|_1 = |\beta_1| + \cdots + |\beta_p|$$



L2正則化（リッジ正則化）

$$\|\beta\|_2^2 = \beta_1^2 + \cdots + \beta_p^2$$



座表軸の上に乗りやすい

R. Tsibshirani (1996). Regression shrinkage and selection via the lasso. J. Royal. Statist. Soc B., Vol. 58, No. 1, pages 267–288.

スパース性の恩恵

$y_i = x_i^\top \beta^* + \epsilon_i$ $(i = 1, \ldots, n)$. β^*：真のベクトル.

$$\hat{\beta} = \arg\min_{\beta \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} (y_i - x_i^\top \beta)^2 + \lambda \sum_{j=1}^{p} |\beta_j|.$$

$x_i \in \mathbb{R}^p$ (p 次元), $d = \|\beta^*\|_0$ (真の非 0 要素の数) とする.

Theorem (Lasso の収束レート)

ある条件のもと，ある定数 C が存在して

$$\|\hat{\beta} - \beta^*\|_2^2 \leq C \frac{d \log(p)}{n}.$$

※全体の次元 p はたかだか $O(\log(p))$ でしか影響しない！
実質的次元 d が支配的.

$$\text{(Lasso)} \quad \frac{d \log(p)}{n} \ll \frac{p}{n} \quad \text{(最小二乗法)}$$

制限固有値条件 (Restricted eigenvalue condition)

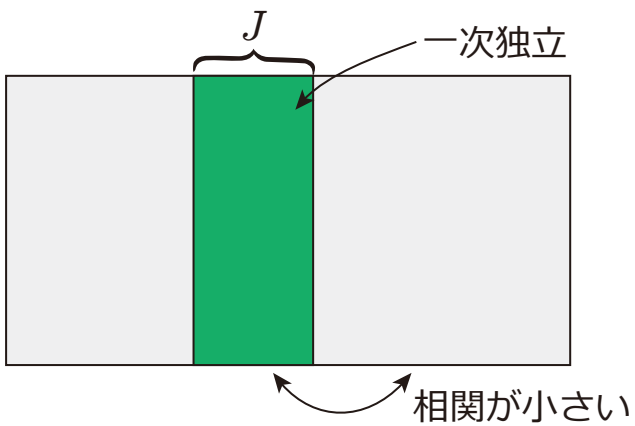
$A = \frac{1}{n} X^\top X$ とする.

Definition (制限固有値条件 (RE(k', C)))

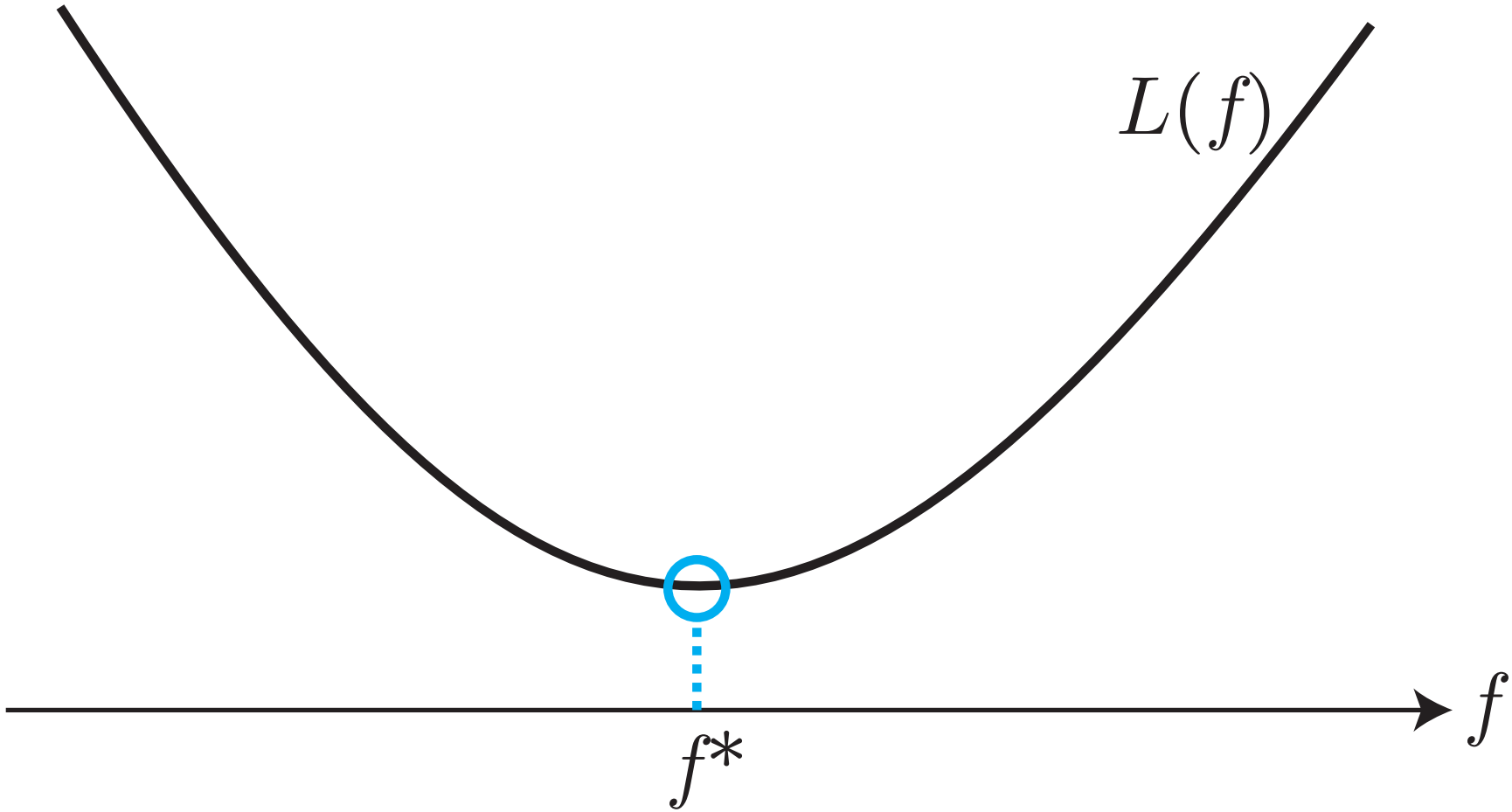
$$\phi_{\mathrm{RE}}(k', C) = \phi_{\mathrm{RE}}(k', C, A) := \inf_{\substack{J \subseteq \{1,\dots,n\}, v \in \mathbb{R}^p : \\ |J| \leq k', C\|v_J\|_1 \geq \|v_{J^c}\|_1}} \frac{v^\top A v}{\|v_J\|_2^2}$$

に対し，$\phi_{\mathrm{RE}} > 0$ が成り立つ.

- ほぼスパースなベクトルに制限して定義した最小固有値.
- $k' = 2d$ で成り立っていればよい.
- ランダムな X に対して高確率で成り立つことが示せる: Johnson Lindenstrauss の補題 (Johnson et al., 1986, Dasgupta and Gupta, 1999, Rudelson and Zhou, 2013).

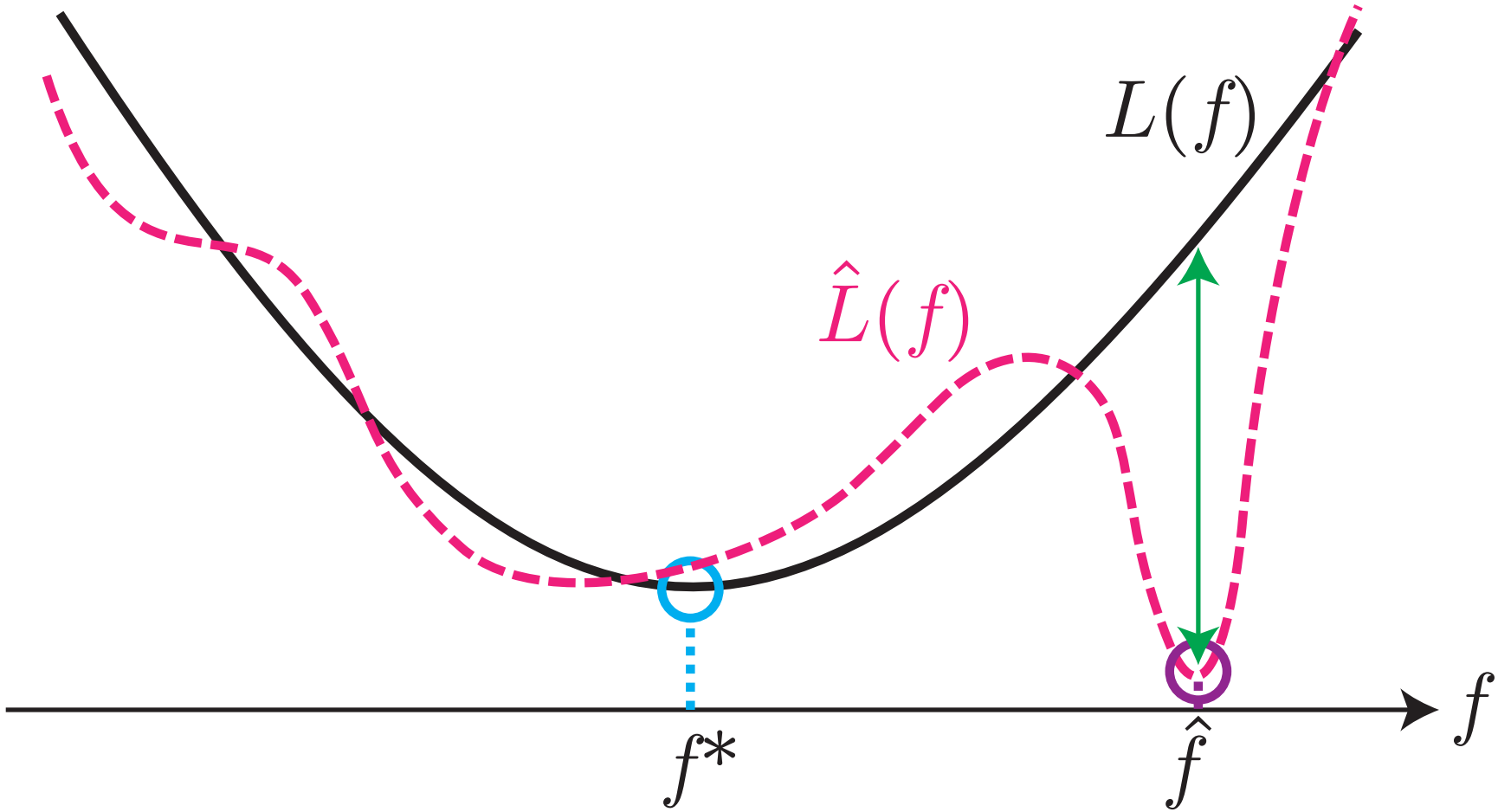


一様バウンド



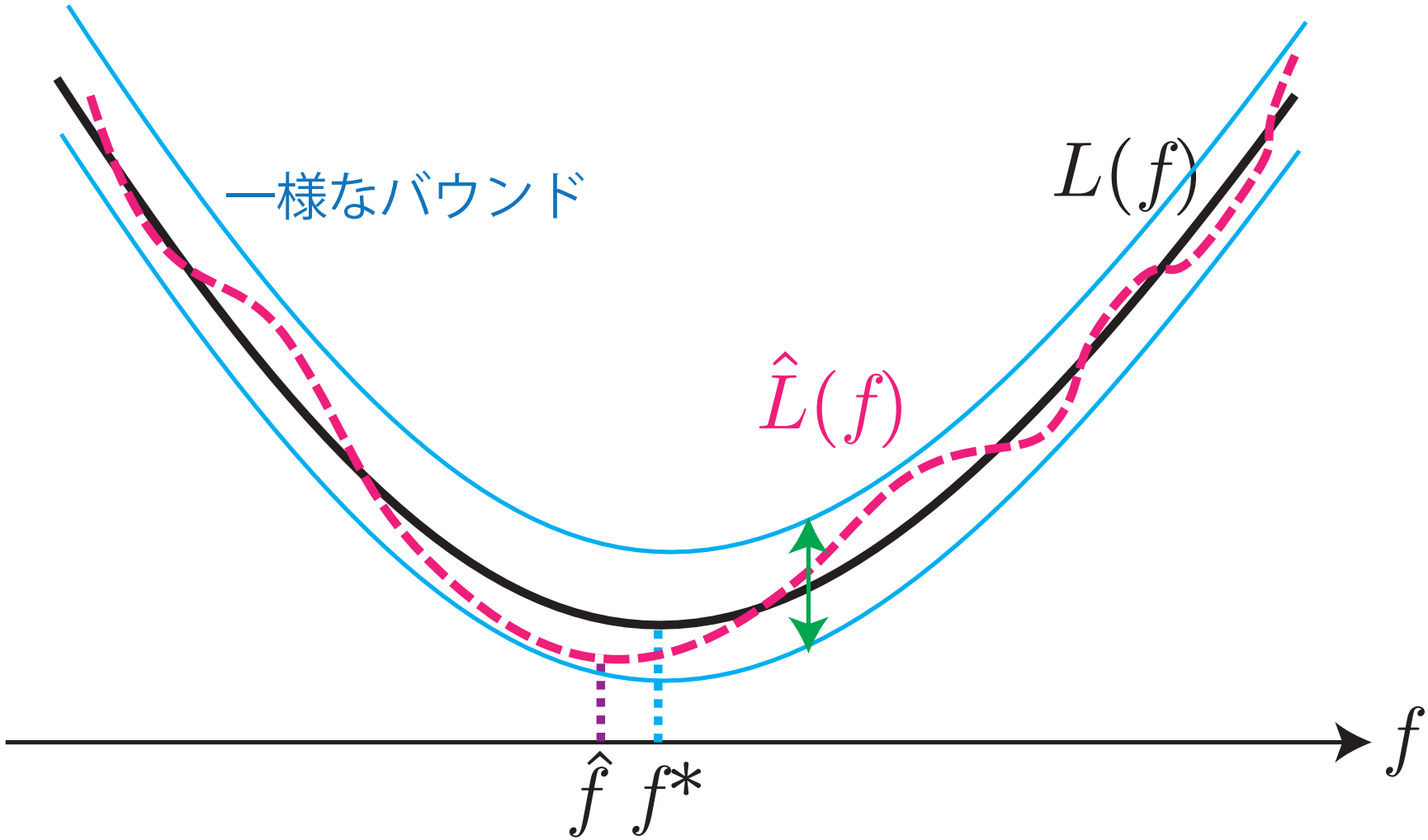
$$L(f) = \mathbb{E}[\ell(Y, f(X)),\ \widehat{L}(f) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f(x_i))]$$

一様バウンド



“たまたま” うまくいくやつがいる (過学習) かもしれない.
実際，$\mathcal{F}$ が複雑な場合収束しない例が

一様バウンド



一様なバウンドによって「たまたまうまくいく」が (ほとんど) ないことを保証
(経験過程の理論)

Rademacher 複雑度

(一様バウンド)　$L(\hat{f}) - \hat{L}(\hat{f}) \leq \sup_{f \in \mathcal{F}} \left\{ L(f) - \hat{L}(f) \right\} \leq (?)$

Rademacher 複雑度:
$\epsilon_1, \epsilon_2, \ldots, \epsilon_n$: Rademacher 変数, i.e., $P(\epsilon_i = 1) = P(\epsilon_i = -1) = \frac{1}{2}$.

$$R(\ell \circ \mathcal{F}) := \mathbb{E}_{\{\epsilon_i\}, \{x_i\}} \left[\sup_{f \in \mathcal{F}} \frac{1}{n} \left| \sum_{i=1}^{n} \epsilon_i \ell(y_i, f(x_i)) \right| \right].$$

対称化:

(期待値のバウンド)　$\mathbb{E}\left[\sup_{f \in \mathcal{F}} |\widehat{L}(f) - L(f)| \right] \leq 2R(\ell \circ \mathcal{F}).$

Rademacher 複雑さを抑えれば一様バウンドが得られる！

基本的に，$R(\ell \circ \mathcal{F}) \leq O(1/\sqrt{n})$ で抑えられる.
例： $\mathcal{F} = \{f(x) = x^\top \beta \mid \beta \in \mathbb{R}^d,\ \|\beta\| \leq 1\}$ かつ ℓ が 1-Lipshitz 連続な時,

$$R(\ell \circ \mathcal{F}) \leq O(\sqrt{d/n}).$$

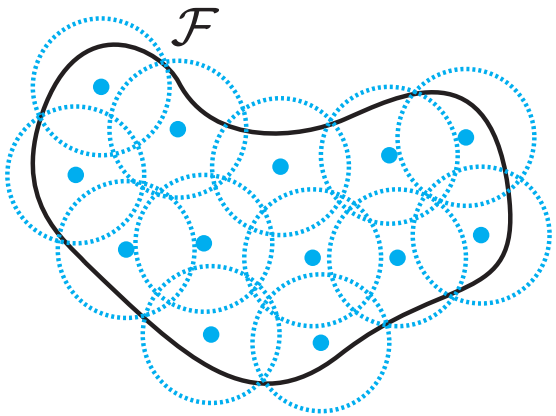
カバリングナンバー（参考）

Rademacher 複雑度を抑えるために有用.
カバリングナンバー: 仮説集合 $\mathcal{F}$ の複雑さ・容量.

ϵ-カバリングナンバー

$$N(\mathcal{F}, \epsilon, d)$$

ノルム d で定まる半径 ϵ のボールで $\mathcal{F}$ を覆うために必要な最小のボールの数.



有限個の元で $\mathcal{F}$ を近似するのに最低限必要な個数.

Theorem (Dudley 積分)

$\|f\|_n^2 := \frac{1}{n}\sum_{i=1}^n f(x_i)^2$ とすると,

$$R(\mathcal{F}) \leq \frac{C}{\sqrt{n}}\mathbb{E}_{D_n}\left[\int_0^\infty \sqrt{\log(N(\mathcal{F}, \epsilon, \|\cdot\|_n))}\mathrm{d}\epsilon\right].$$

局所 Rademacher 複雑さ（参考）

局所 Rademacher 複雑さ: $R_\delta(\mathcal{F}) := R(\{f \in \mathcal{F} \mid \mathbb{E}[(f - f^*)^2] \leq \delta\})$.

次の条件を仮定してみる.

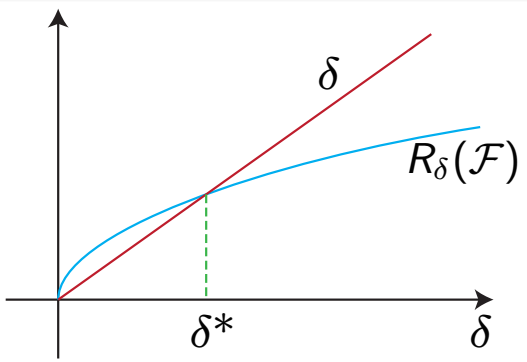
- $\mathcal{F}$ は 1 で上から抑えられている: $\|f\|_\infty \leq 1$ $(\forall f \in \mathcal{F})$.
- ℓ は Lipschitz 連続かつ強凸:
  $\mathbb{E}[\ell(Y, f(X))] - \mathbb{E}[\ell(Y, f^*(X))] \geq B\mathbb{E}[(f - f^*)^2]$ $(\forall f \in \mathcal{F})$.

Theorem (Fast learning rate (Bartlett et al., 2005))

$\delta^* = \inf\{\delta \mid \delta \geq R_\delta(\mathcal{F})\}$ とすると，確率 $1 - e^{-t}$ で

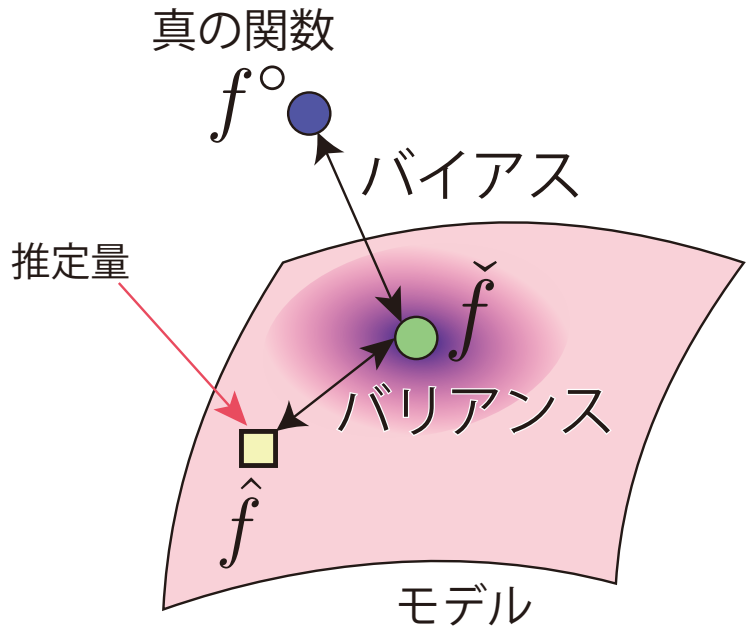
$$L(\hat{f}) - L(f^*) \leq C\left(\delta^* + \frac{t}{n}\right).$$

$\delta^* \leq R(\mathcal{F})$ は常に成り立つ (右図参照).
これを Fast learning rate と言う.

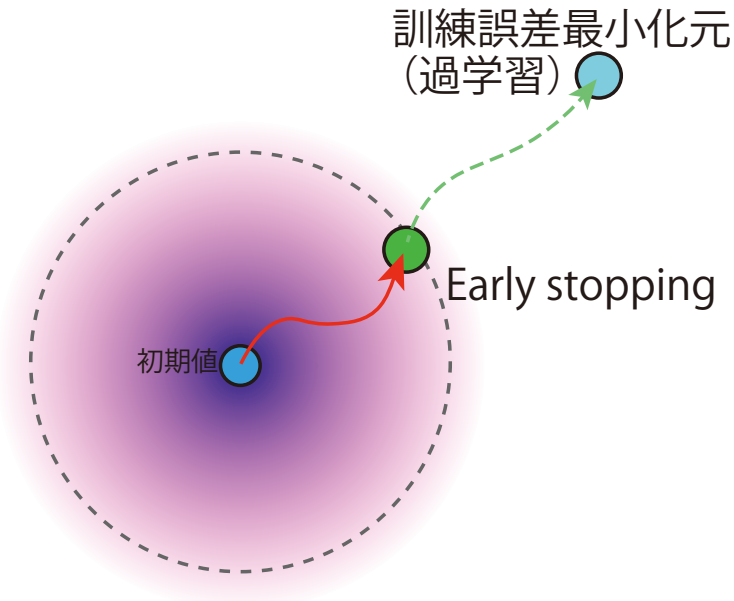


正則化と最適化

モデルの制限による正則化



Early stopping による正則化

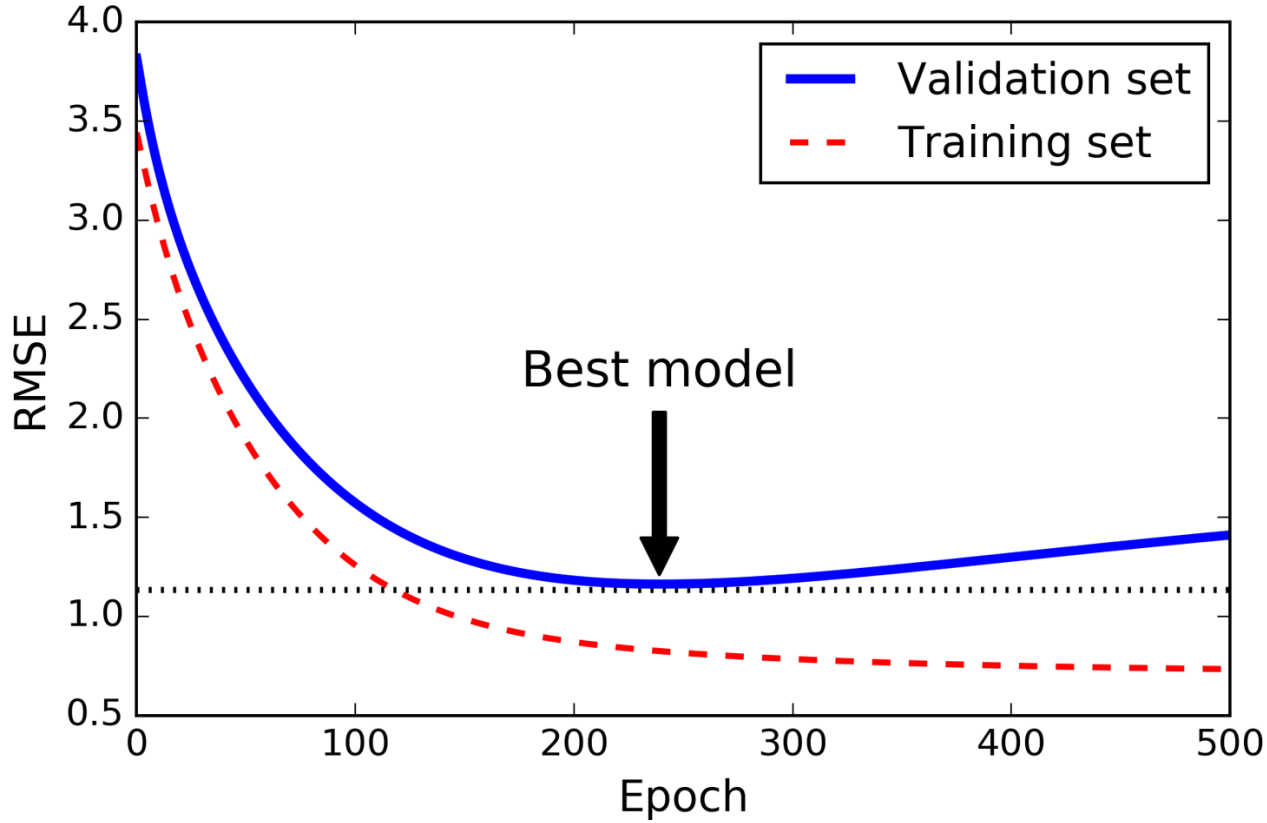


バイアス-バリアンス分解

$$\underbrace{\|f^{\circ} - \hat{f}\|_{L_2(P_X)}}_{\text{Estimation error}} \leq \underbrace{\|f^{\circ} - \check{f}\|_{L_2(P_X)}}_{\substack{\text{Approximation error} \\ \text{(bias)}}} + \underbrace{\|\check{f} - \hat{f}\|_{L_2(P_X)}}_{\substack{\text{Sample deviation} \\ \text{(variance)}}}$$

訓練誤差最小化元に達する前に止める (early stopping) ことで正則化が働く.
→ 深層学習，Boosting の常套手段.

Early stopping による過学習の回避



Hands-On Machine Learning with Scikit-Learn and TensorFlow by Aurlien Gron.
Chapter 4. Training Models.
https://www.oreilly.com/library/view/hands-on-machine-learning/9781491962282/ch04.html

機械学習の最適化の特徴

- 汎化誤差を小さくすることが重要．必ずしも最適化問題を完全に解く必要はない．
- 目的に応じて最適化しやすいように問題を変えて良い．
  例: スパース推定 (組合せ最適化を凸最適化に緩和)．
- 大規模・高次元データ．
  → なるべく楽して最適化したい．一次最適化法，確率的最適化法．

Outline

正則化学習法

- 訓練誤差最小化:

$$\min_{x \in \mathbb{R}^p} \quad \frac{1}{n} \sum_{i=1}^{n} \ell(z_i, x).$$

- 正則化付き訓練誤差最小化:

$$\min_{x \in \mathbb{R}^p} \quad \frac{1}{n} \sum_{i=1}^{n} \ell(z_i, x) + \psi(x).$$

しばらく ℓ と ψ は凸関数であると仮定.

平滑性と強凸性

Definition

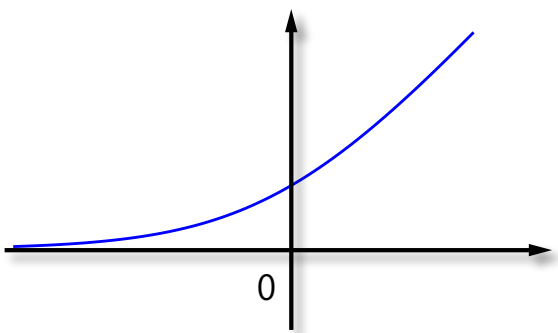
- 平滑性: 勾配がリプシッツ連続:

$$\|\nabla f(x) - \nabla f(x')\| \leq L\|x - x'\|.$$

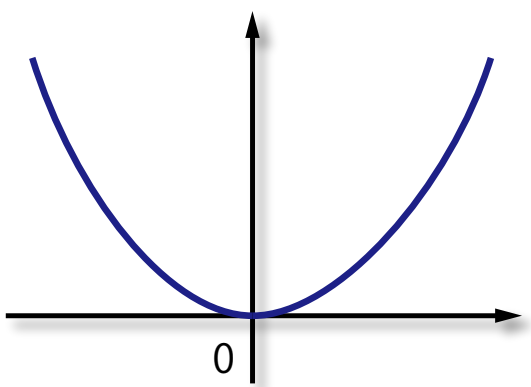
$$\Rightarrow f(x) \leq f(y) + \langle x - y, \nabla f(y)\rangle + \frac{L}{2}\|x - y\|^2.$$

- 強凸性: $\forall \theta \in (0,1), \forall x, y \in \mathrm{dom}(f)$,

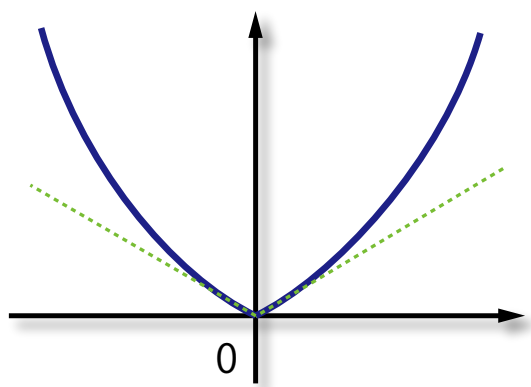
$$\frac{\mu}{2}\theta(1-\theta)\|x - y\|^2 + f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta)f(y).$$



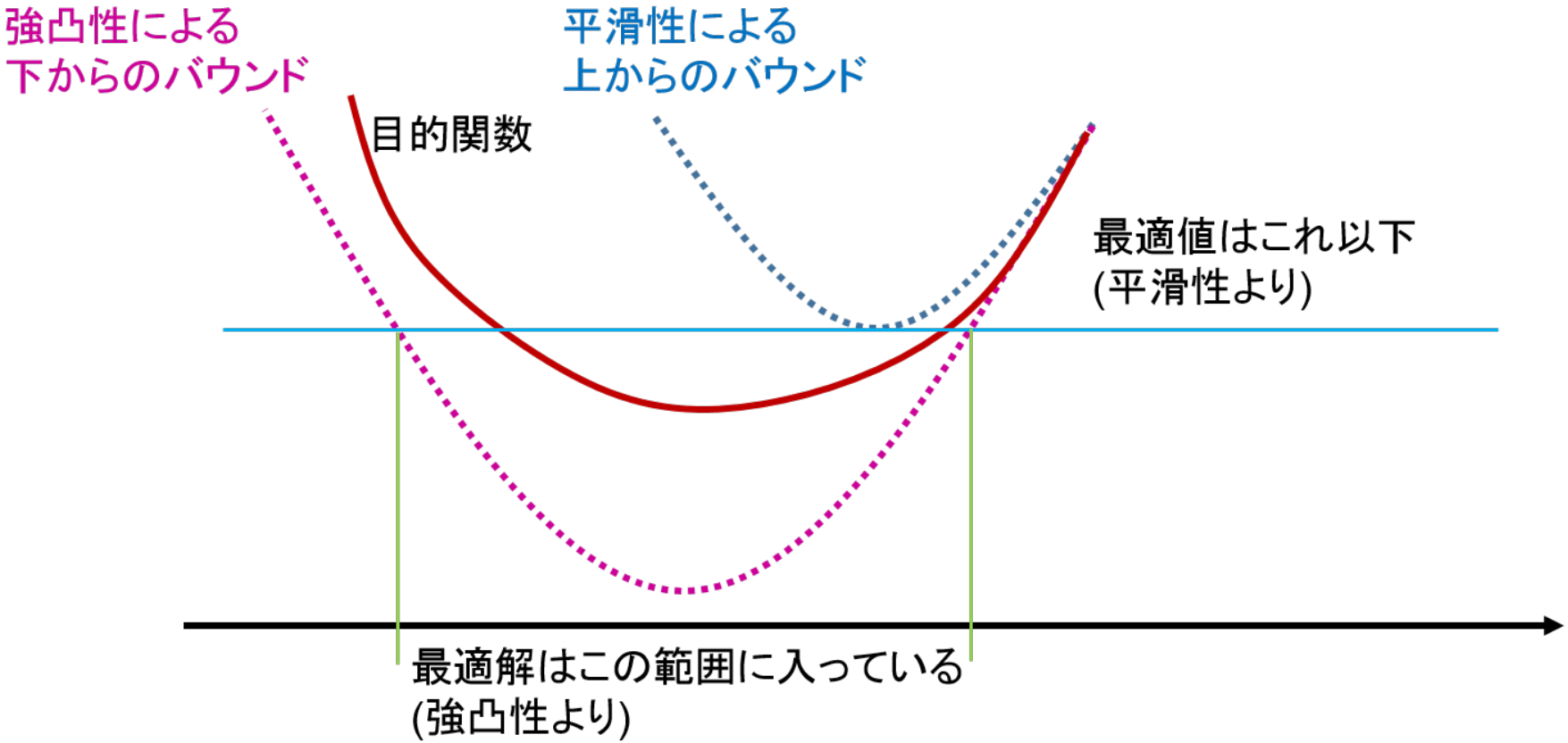
平滑だが
強凸ではない



平滑かつ
強凸



強凸だが
平滑ではない



- 平滑性→最適値を上から抑えられる.
- 強凸性→最適値の範囲を限定できる.

平滑性と強凸性の双対性

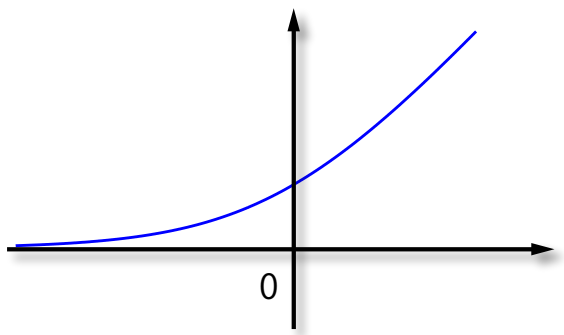
平滑性と強凸性は互いに双対の関係にある.

Theorem

$f : \mathbb{R}^p \to \bar{\mathbb{R}}$ を真閉凸関数であるとする．その時，以下が成り立つ:

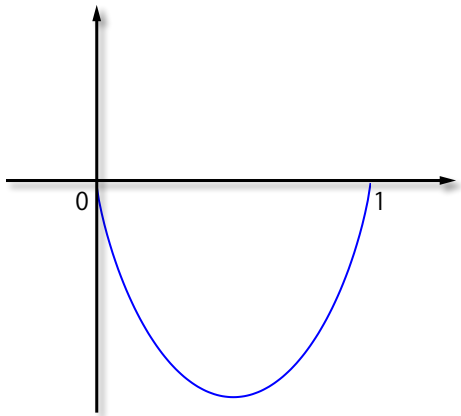
$$f \text{ が } L\text{-平滑} \iff f^* \text{ が } 1/L\text{-強凸}.$$

logistic loss



Smooth but
not strongly convex

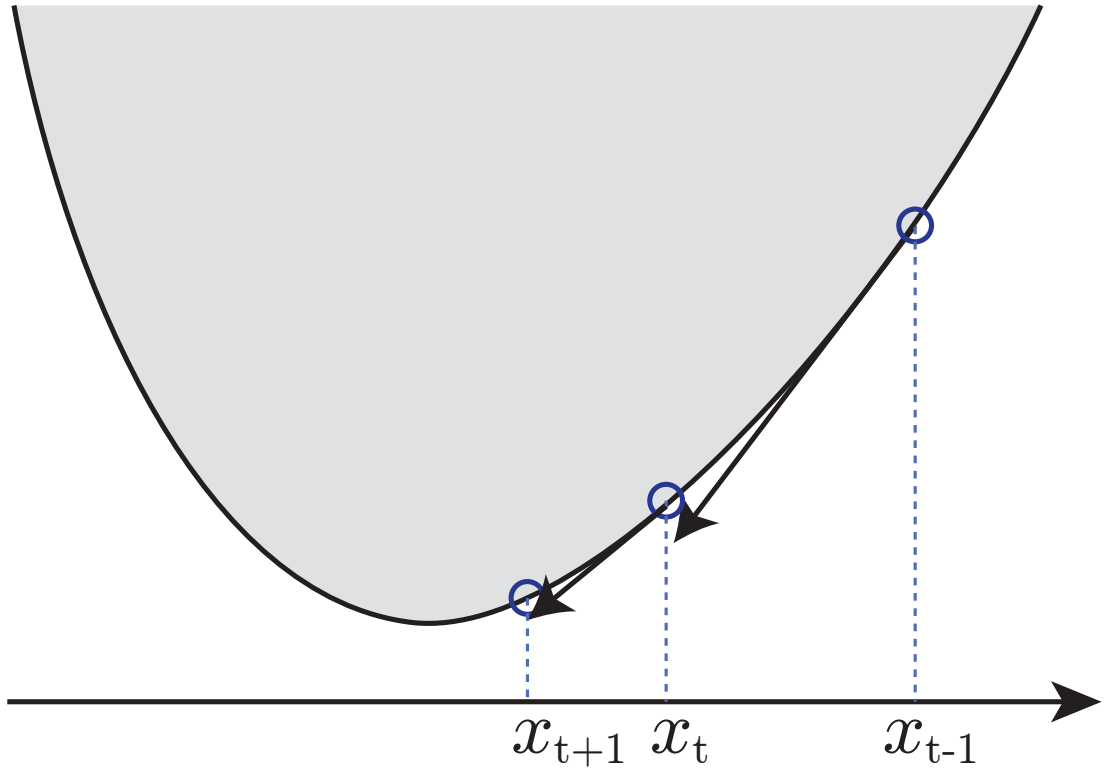
its dual function



Strongly convex but
not smooth
(gradient $\to \infty$)

$$f^*(y) = \sup_{x \in \mathbb{R}^p} \{\langle x, y \rangle - f(x)\}.$$

一次最適化法



- 関数値 $f(x)$ と勾配 $g \in \partial f(x)$ の情報のみを用いた最適化手法.
- 一回の更新にかかる計算量が軽く，高次元最適化問題に有用.
- ニュートン法は**二次最適化手法**.

最急降下法

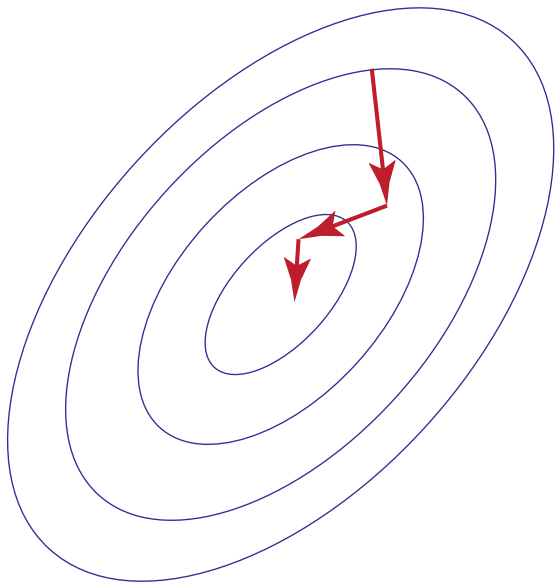
$f(x) = \sum_{i=1}^{n} \ell(z_i, x)$ とする.

$$\min_x \quad f(x).$$

(劣) 勾配法

微分可能な $f(x)$:

$$x_t = x_{t-1} - \eta_t \nabla f(x_{t-1}).$$



最急降下法

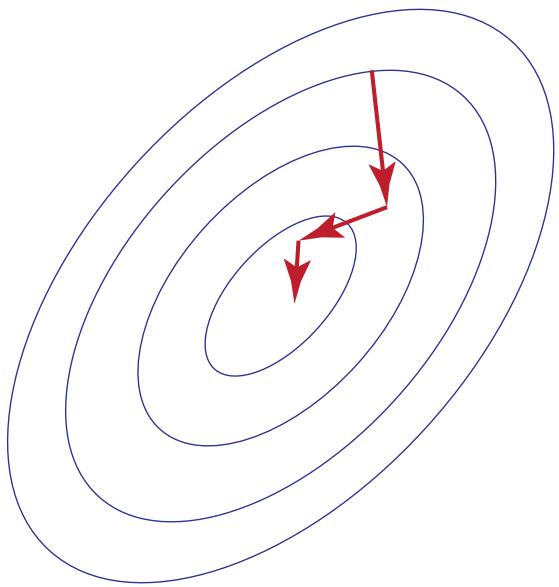
$f(x) = \sum_{i=1}^{n} \ell(z_i, x)$ とする.

$$\min_x \quad f(x).$$

(劣) 勾配法

劣微分可能な $f(x)$:

$$g_t \in \partial f(x_{t-1}),$$
$$x_t = x_{t-1} - \eta_t g_t.$$



最急降下法

$f(x) = \sum_{i=1}^{n} \ell(z_i, x)$ とする.

$$\min_x \quad f(x).$$

(劣) 勾配法 (同値な表現)

劣微分可能な $f(x)$:

$$x_t = x_{t-1} - \eta_t g_t = \operatorname*{argmin}_x \left\{ \frac{1}{2\eta_t} \|x - (x_{t-1} - \eta_t g_t)\|^2 \right\}$$

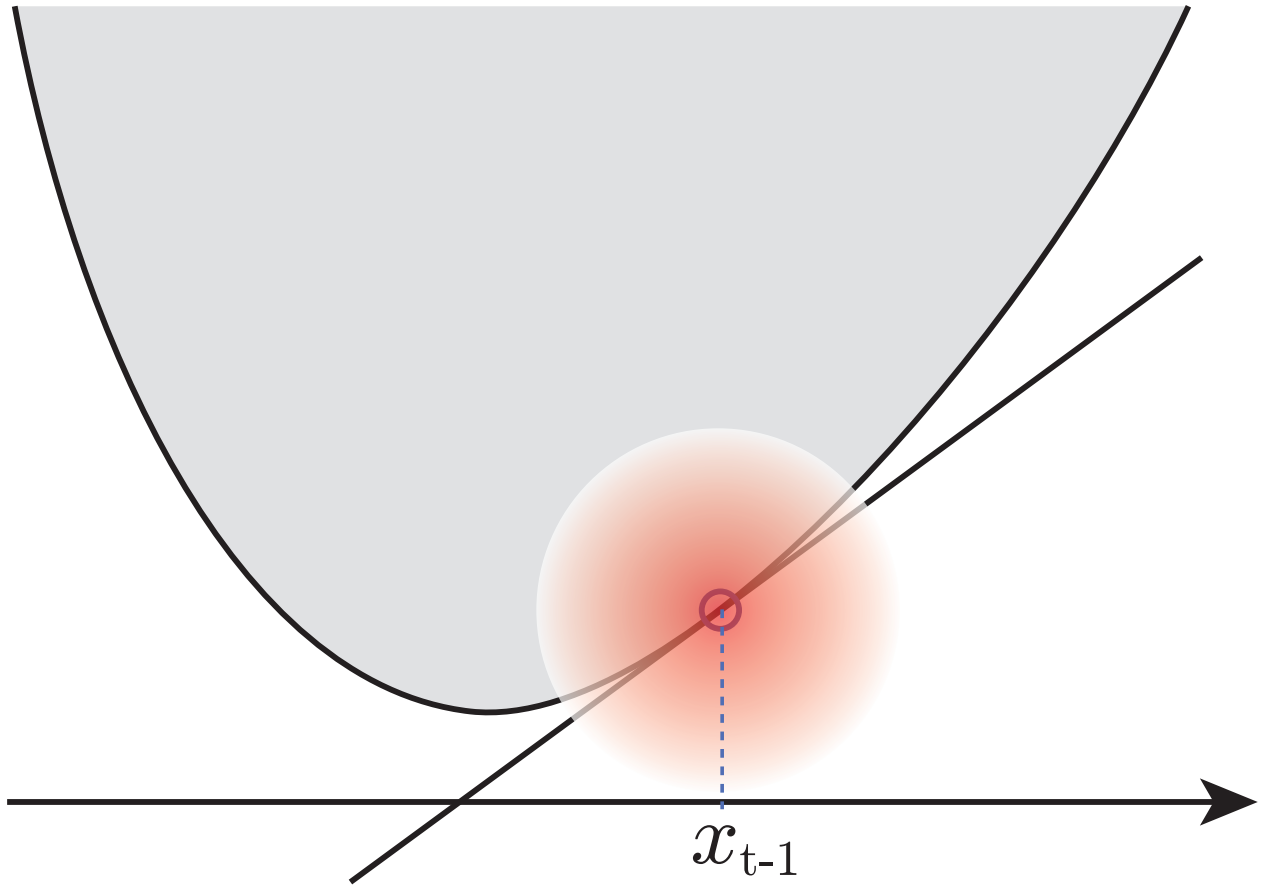
$$= \operatorname*{argmin}_x \left\{ \langle x, g_t \rangle + \frac{1}{2\eta_t} \|x - x_{t-1}\|^2 \right\},$$

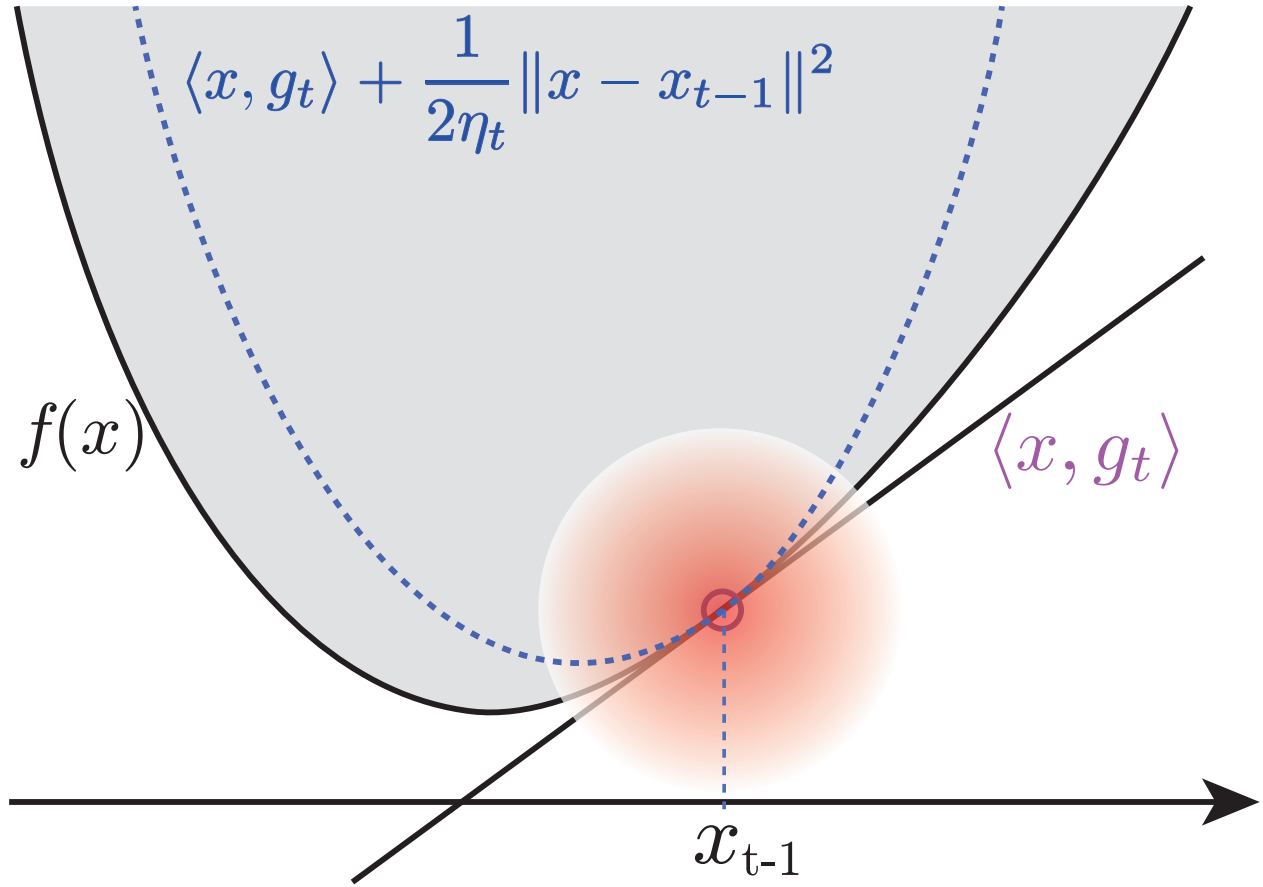
ただし, $g_t \in \partial f(x_{t-1})$.

近接点アルゴリズム:

$$x_t = \operatorname*{argmin}_x \left\{ f(x) + \frac{1}{2\eta_t} \|x - x_{t-1}\|^2 \right\}.$$

- 一般の場合: $f(x_t) - f(x^*) \leq \frac{1}{2\sum_{k=1}^{t} \eta_k} \|x_0 - x^*\|$.
- $f(x)$ が強凸の場合: $f(x_t) - f(x^*) \leq \frac{1}{2\eta} \left(\frac{1}{1+\sigma\eta} \right)^{t-1} \|x_0 - x^*\|^2$.


$x_{\text{t-1}}$


$\langle x, g_t \rangle + \frac{1}{2\eta_t} \|x - x_{t-1}\|^2$
$f(x)$
$\langle x, g_t \rangle$
$x_{\text{t-1}}$

★ 近接勾配法

$f(x) = \sum_{i=1}^{n} \ell(z_i, x)$ とする.

$$\min_x \quad f(x) + \psi(x).$$

近接勾配法

$$\begin{aligned} x_t &= \operatorname*{argmin}_x \left\{ \langle x, g_t \rangle + \psi(x) + \frac{1}{2\eta_t} \|x - x_{t-1}\|^2 \right\} \\ &= \operatorname*{argmin}_x \left\{ \eta_t \psi(x) + \frac{1}{2} \|x - (x_{t-1} - \eta_t g_t)\|^2 \right\} \end{aligned}$$

ただし, $g_t \in \partial f(x_{t-1})$.

更新則は近接写像で与えられる:

$$\operatorname{prox}(q|\tilde{\psi}) = \operatorname*{argmin}_x \left\{ \tilde{\psi}(x) + \frac{1}{2} \|x - q\|^2 \right\}$$

→ 近接写像により正則化項の悪い性質の影響を回避 (微分不可能性など).

近接写像の例 1

- L_1 正則化: $\psi(x) = \lambda\|x\|_1$.

$$x_t = \operatorname*{argmin}_x \left\{ \lambda\eta_t\|x\|_1 + \frac{1}{2}\|x - \underbrace{(x_{t-1} - \eta_t g_t)}_{=:q_t}\|^2 \right\}$$

$$= \operatorname*{argmin}_x \left\{ \sum_{j=1}^{p} \left[\lambda\eta_t|x_j| + \frac{1}{2}(x_j - q_{t,j})^2 \right] \right\}$$

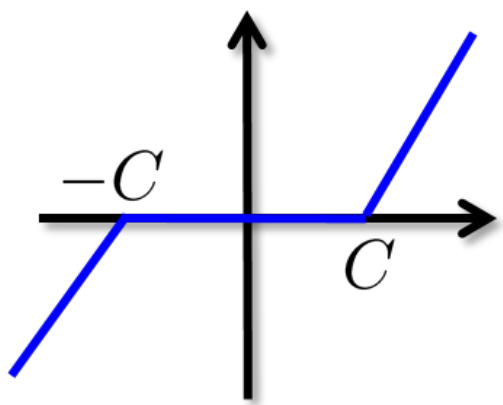
座標ごとに分かれている！

$$x_{t,j} = \mathrm{ST}_{\lambda\eta_t}(q_{t,j}) \quad (j \text{ 番目の要素})$$

ただし ST は Soft-Thresholding function と呼ばれる:

$$\mathrm{ST}_C(q) = \mathrm{sign}(q)\max\{|q| - C, 0\}.$$

$\rightarrow$ 重要でない要素を 0 にする.



近接写像の例 2

- トレースノルム: $\psi(X) = \lambda\|X\|_{\mathrm{tr}} = \lambda \sum_j \sigma_j(X)$ (特異値の和).

$$X_{t-1} - \eta_t G_t = U \,\mathrm{diag}(\sigma_1, \ldots, \sigma_d) V,$$

と特異値分解すると,

$$X_t = U \begin{pmatrix} \mathrm{ST}_{\lambda\eta_t}(\sigma_1) & & \\ & \ddots & \\ & & \mathrm{ST}_{\lambda\eta}(\sigma_d) \end{pmatrix} V.$$

近接勾配法の収束

強凸性と平滑性が収束レートを決める.

$$x_t = \mathrm{prox}(x_{t-1} - \eta_t g_t | \eta_t \psi(x)).$$

f の性質	μ-強凸	非強凸
γ-平滑	$\exp\left(-t\frac{\mu}{\gamma}\right)$	$\frac{\gamma}{t}$
非平滑	$\frac{1}{\mu t}$	$\frac{1}{\sqrt{t}}$

- ステップ幅 η_t は適切に選ぶ必要がある.

η_t の設定	強凸	非強凸
平滑	$\frac{1}{\gamma}$	$\frac{1}{\gamma}$
非平滑	$\frac{2}{\mu t}$	$\frac{1}{\sqrt{t}}$

- 最適な収束レートを得るためには適宜 $\{x_t\}_t$ の平均を取る必要がある.
  Polyak-Ruppert 平均化, 多項式平均化.

近接勾配法の収束

強凸性と平滑性が収束レートを決める.

$$x_t = \mathrm{prox}(x_{t-1} - \eta_t g_t | \eta_t \psi(x)).$$

f の性質	μ-強凸	非強凸
γ-平滑	$\exp\left(-t\sqrt{\frac{\mu}{\gamma}}\right)$	$\frac{\gamma}{t^2}$
非平滑	$\frac{1}{\mu t}$	$\frac{1}{\sqrt{t}}$

- ステップ幅 η_t は適切に選ぶ必要がある.

η_t の設定	強凸	非強凸
平滑	$\frac{1}{\gamma}$	$\frac{1}{\gamma}$
非平滑	$\frac{2}{\mu t}$	$\frac{1}{\sqrt{t}}$

- 最適な収束レートを得るためには適宜 $\{x_t\}_t$ の平均を取る必要がある.
  Polyak-Ruppert 平均化, 多項式平均化.
- 平滑な損失なら Nesterov の加速法により収束を改善できる.
  → 最適な収束レート

Nesterov の加速法 (非強凸)

$\min_x\{f(x)+\psi(x)\}$
仮定: $f(x)$ は γ-平滑.

Nesterov の加速法

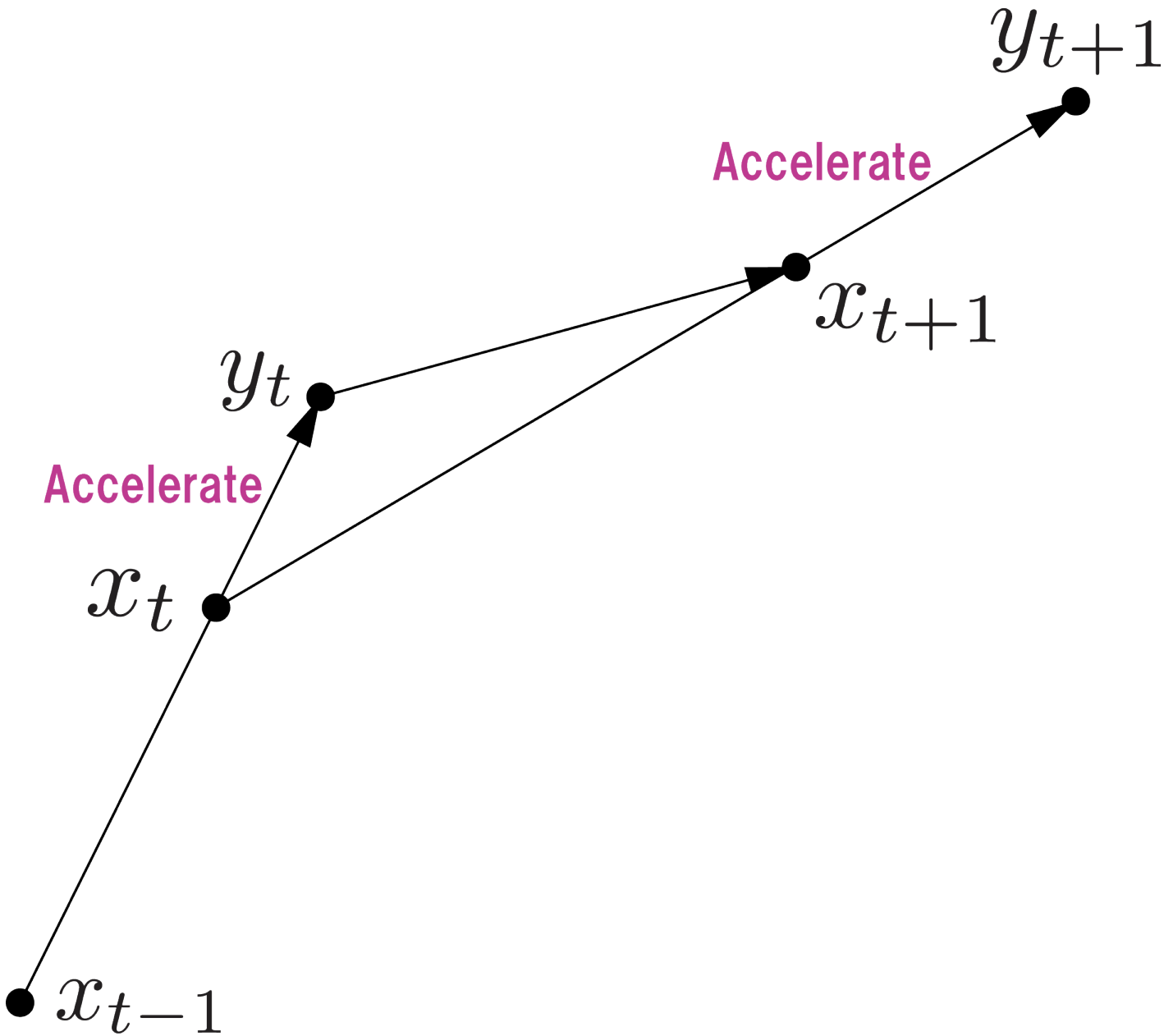
$s_1=1$, $\eta=\frac{1}{\gamma}$ とする. $t=1,2,\ldots$ で以下を繰り返す:

1. $g_t=\nabla f(y_t)$ として $x_t=\mathrm{prox}(y_t-\eta g_t|\eta\psi)$ と更新.
2. $s_{t+1}=\frac{1+\sqrt{1+4s_t^2}}{2}$ と設定.
3. $y_{t+1}=x_t+\left(\frac{s_t-1}{s_{t+1}}\right)(x_t-x_{t-1})$ と更新.

f が γ-平滑ならば,

$$f(x_t)-f(x^*)\leq\frac{2\gamma\|x_0-x^*\|^2}{t^2}.$$

- Fast Iterative Shrinkage Thresholding Algorithm (FISTA) (Beck and Teboulle, 2009) とも呼ばれている.
- ステップサイズ $\eta=1/\gamma$ はバックトラッキングで決定できる.
- 深層学習で使われている "モーメンタム" 法も似たような方法 (Sutskever et al., 2013).


y_{t+1}
Accelerate
x_{t+1}
y_t
Accelerate
x_t
x_{t-1}

Nesterov の加速法の解釈

加速法の解釈は様々な方向からなされてきた．その中でも，Ahn (2020) による最近の結果は理解しやすい．

- **近接点アルゴリズム:** $x_{t+1} = \mathrm{argmin}_x\{f(x) + \frac{1}{2\eta_{t+1}}\|x - x_t\|^2\}$ (良い収束).
  $\to$ 2 種類の近似: $g_t = \nabla f(y_t)$,

$$f(y_t) + \langle g_t, x - y_t\rangle \le f(x) \le f(y_t) + \langle g_t, x - y_t\rangle + \frac{\gamma}{2}\|x - y_t\|^2.$$

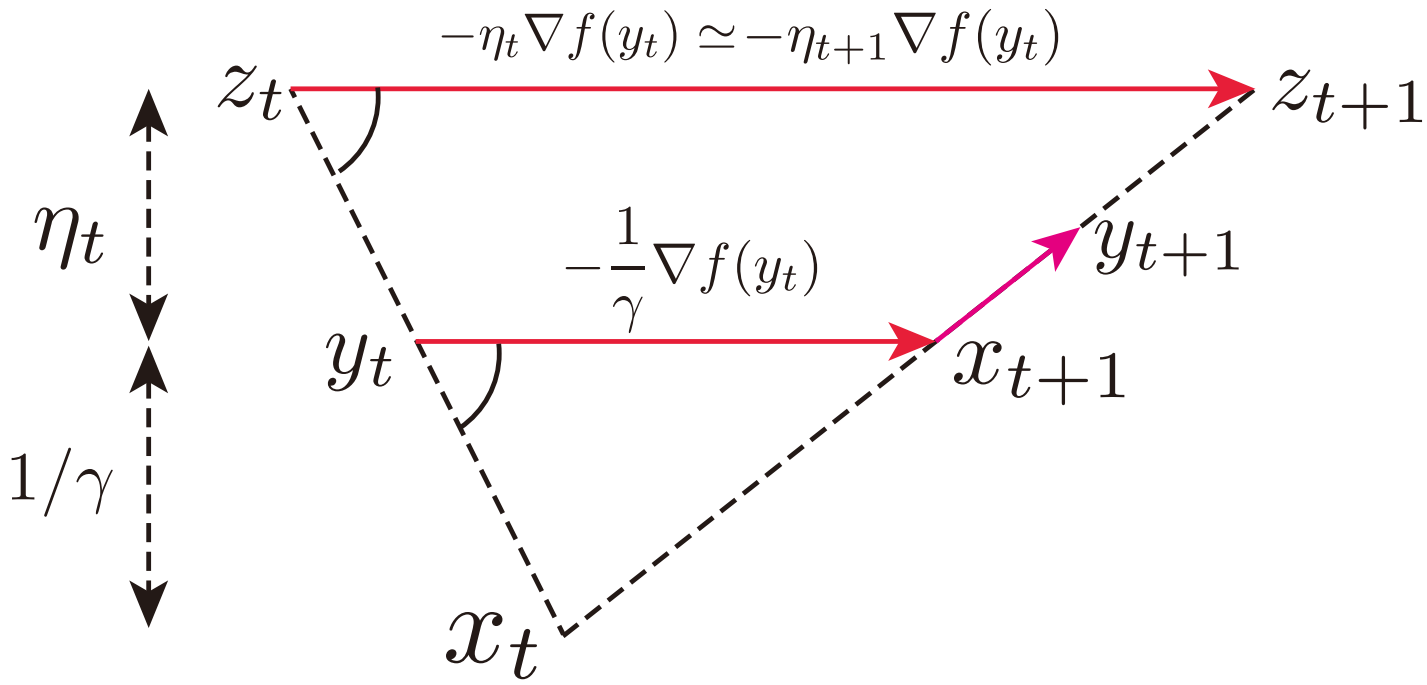
- 2 種類の近似を用いた交互最適化:

$$z_{t+1} = \mathrm{argmin}_z \left\{ f(y_t) + \langle \nabla f(y_t), z - y_t\rangle + \frac{1}{2\eta_{t+1}}\|z - z_t\|^2 \right\},$$

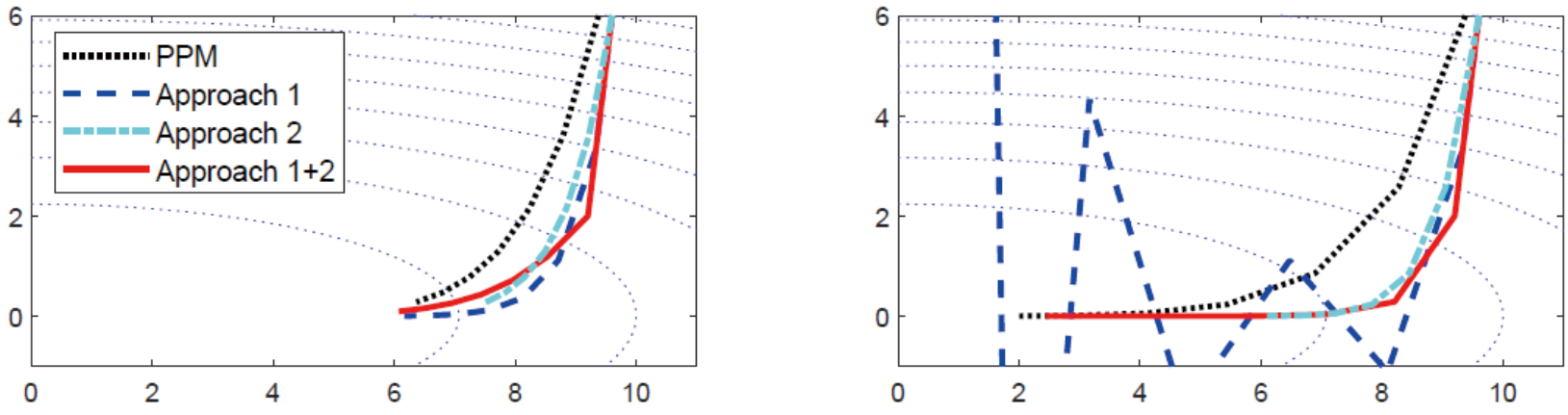
$$y_{t+1} = \mathrm{argmin}_y \left\{ f(y_t) + \langle \nabla f(y_t), y - y_t\rangle + \frac{\gamma}{2}\|y - y_t\|^2 + \frac{1}{2\eta_{t+1}}\|y - z_{t+1}\|^2 \right\}.$$

$$\begin{aligned} y_t &= \tfrac{1/\gamma}{1/\gamma + 1/\eta_t} z_t + \tfrac{1/\gamma}{1/\gamma + 1/\eta_t} x_t, \\ z_{t+1} &= z_t - \eta_{t+1}\nabla f(y_t), \\ x_{t+1} &= y_t - \tfrac{1}{\gamma}\nabla f(y_t). \end{aligned} \quad \simeq \quad \begin{aligned} y_t &= \tfrac{1/\gamma}{1/\gamma + 1/\eta_t} z_t + \tfrac{1/\gamma}{1/\gamma + 1/\eta_t} x_t, \\ x_{t+1} &= y_t - \tfrac{1}{\gamma}\nabla f(y_t), \\ z_{t+1} &= x_{t+1} + \gamma\eta_t(x_{t+1} - x_t). \end{aligned}$$

- $(\gamma\eta_{t+1} + 1/2)^2 = (\gamma\eta_t + 1)^2 + 1/4$ とすれば，元の更新式を得る ($\eta_t = \Theta(t)$).
- 左の更新式でも $O(1/t^2)$ を達成.



$\eta_t = \Theta(t)$ とする．つまり，t が増大するにつれ，下界に関する更新が強調される．強凸度合いが強い方向へ先に収束して（上界の方），後から強凸具合が弱い方向（下界の方）を収束させる動きになる．



(a) $\eta_t \equiv 1/3$. (b) $\eta_t = t/3$.

Figure 1: Iterates comparison between PPM (1.1), the first approach (3.1), the second approach (3.3), and the combined approach (4.1). For the setting, we choose $f(x, y) = 0.1x^2 + y^2$ and $x_0 = (10, 10)$.

加速法の軌道．Ahn (2020) より．

Approach 1: 下界のみ．Approach 2: 上界のみ．Approach 1+2: 加速法．

Nesterov の加速法 (強凸)

リスタート法

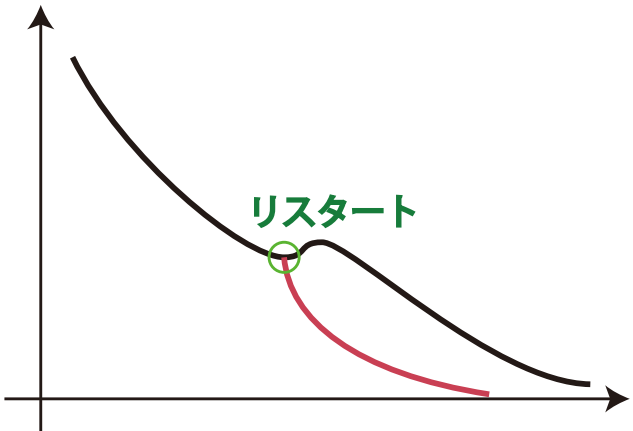
ある程度 Nesterov の加速法で更新を繰り返したら，初期化しなおしてリスタートする．

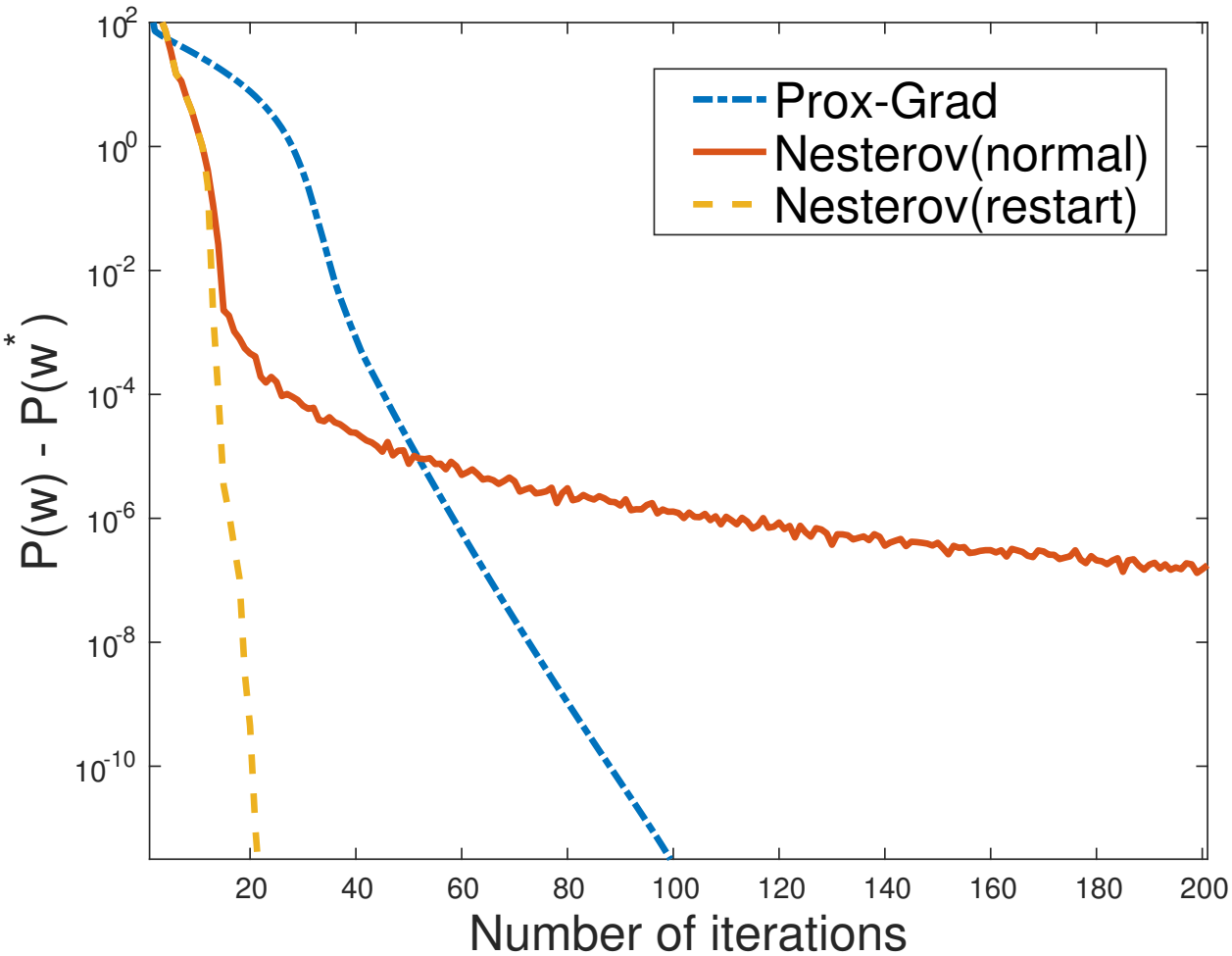
直接加速するバージョンもあるが，条件が弱くて済む (一点強凸性)， リスタート版の方が見通しがよい，実装も楽．

リスタートの規準

- $t \geq \sqrt{\frac{8\gamma}{\mu}}$ 回更新したらリスタート (Excess risk が 1/2 になるため)
- 目的関数が一度上昇したらリスタート
- $(y_{t+1} - x_{t-1})^\top (x_t - x_{t-1}) \geq 0$ となったらリスタート

第二，第三の方法はヒューリスティクス．$\exp\left(-t\sqrt{\frac{\mu}{\gamma}}\right)$ の収束レート．





Nesterov's acceleration vs. normal gradient descent

Lasso: $n = 8,000$, $p = 500$.

Outline

Outline

機械学習における確率的最適化の歴史

1951	**Robbins and Monro**	**Stochastic approximation** 零点問題
1957	Rosenblatt	パーセプトロン
1978 1983	Nemirovskii and Yudin	滑らかでない関数における ロバストな方策および最適性
1988 1992	Ruppert Polyak and Juditsky	滑らかな関数におけるロバストな ステップサイズや平均化の方策
1998 2004	Bottou Bottou and LeCun	オンライン型確率的最適化による 大規模機械学習
2009- 2012	Singer and Duchi; Duchi et al.; Xiao	FOBOS, AdaGrad, RDA
2012- 2013	Le Roux et al. Shalev-Shwartz and Zhang Johnson and Zhang	バッチ型手法, 線形収束 (SAG,SDCA,SVRG)
2016	Allen-Zhu	Katyusya バッチ型手法の加速
2017-		各種非凸最適化手法の発展

確率的最適化法とは

目的関数: $F(x) = \mathbb{E}_Z[\ell(Z, x)]$

F 自体ではなく，$\ell(Z, x)$ をサンプリングすることしかできない状況で F を最小化する問題（確率的計画問題）を解く手法，または意図的にランダムネスを用いて F を最適化する手法．機械学習では F が陽に計算できる状況でもわざとランダムネスを利用して解くことも多い．

- オンライン型
  - データは次から次へと来る．
  - 基本的に各訓練データは一回しか使わない．

$$\min_x \ \mathbb{E}_Z[\ell(Z, x)]$$

- バッチ型
  - データ数固定．
  - 訓練データは何度も使って良いが，n が大きい状況を想定．$\sum_{i=1}^n \cdot$ はなるべく計算したくない．

$$\min_x \ \frac{1}{n} \sum_{i=1}^{n} \ell(z_i, x)$$

オンライン型確率的最適化の目的関数

$\ell(z, x)$ を観測 z に対するパラメータ x の損失.

(期待損失) $$L(x) = \mathbb{E}_Z[\ell(Z, x)]$$

or

(正則化付き期待損失) $$L_\psi(x) = \mathbb{E}_Z[\ell(Z, x)] + \psi(x)$$

観測値 Z の分布は状況によっていろいろ

- 真の分布
  (つまり $L(x) = \int \ell(Z, x) \mathrm{d}P(Z)$ の時)
  $\rightarrow$ $L(x)$ は汎化誤差.

  オンライン型最適化はそれ自体が学習!
- 巨大ストレージに記憶されているデータの経験分布
  (つまり $L(x) = \frac{1}{n} \sum_{i=1}^{n} \ell(z_i, x)$ の時)
  $\rightarrow$ L (または L_ψ) は (正則化ありの) 訓練誤差.

Outline

確率的勾配降下法

(Stochastic Gradient Descent, SGD)

SGD (正則化なし)

- $z_t \sim P(Z)$ を観測．$\ell_t(x) := \ell(z_t, x)$ とする．
  （ここだけが普通の勾配法と違う点）
- 損失関数の劣微分を計算:

$$g_t \in \partial_x \ell_t(x_{t-1}).$$

- x を更新:

$$x_t = x_{t-1} - \eta_t g_t.$$

- 各反復で一個のデータ z_t を観測すれば良い．
  $\rightarrow$ 各反復ごとに $O(1)$ の計算量 (全データ使う勾配法は $O(n)$).
- データ全体 $\{z_i\}_{i=1}^n$ を使わないで良い．

Reminder: $\mathrm{prox}(q|\psi) := \mathrm{argmin}_x \left\{\psi(x) + \frac{1}{2}\|x - q\|^2\right\}$.

確率的勾配降下法

(Stochastic Gradient Descent, SGD)

SGD (正則化あり)

- $z_t \sim P(Z)$ を観測．$\ell_t(x) := \ell(z_t, x)$ とする．
  （ここだけが普通の勾配法と違う）
- 損失関数の劣微分を計算:

$$g_t \in \partial_x \ell_t(x_{t-1}).$$

- x を更新:

$$x_t = \mathrm{prox}(x_{t-1} - \eta_t g_t | \eta_t \psi).$$

- 各反復で一個のデータ z_t を観測すれば良い．
  $\to$ 各反復ごとに $O(1)$ の計算量 (全データ使う勾配法は $O(n)$).
- データ全体 $\{z_i\}_{i=1}^n$ を使わないで良い．

Reminder: $\mathrm{prox}(q|\psi) := \mathrm{argmin}_x \left\{\psi(x) + \frac{1}{2}\|x - q\|^2\right\}$.

重要な点

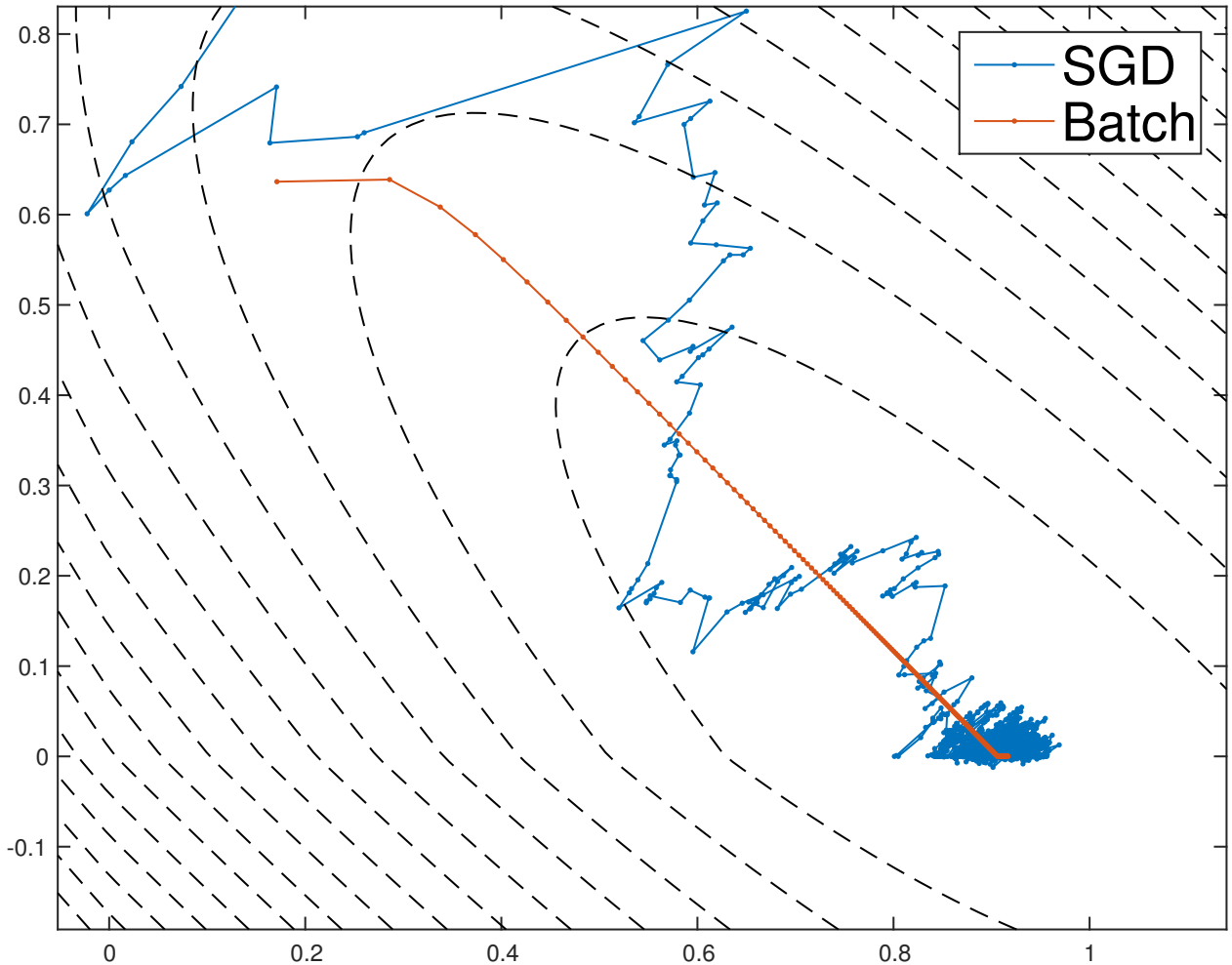
確率的勾配の期待値は本当の勾配

$g_t = \nabla \ell_t(x_{t-1})$ より，

$$\mathbb{E}_{z_t}[g_t] = \mathbb{E}_{z_t}[\nabla \ell(Z, x_{t-1})] = \nabla \mathbb{E}_{z_t}[\ell(Z, x_{t-1})] = \nabla L(x_{t-1})$$

$\Rightarrow$ 確率的勾配は本当の勾配の不偏推定量

SGD の振る舞い



SGD の収束解析

仮定

(A1) $\mathbb{E}[\|g_t\|^2] \le G^2$.
(A2) $\mathbb{E}[\|x_t - x^*\|^2] \le D^2$.

(仮定 (A2) は $\{x \mid \|x\| \le D/2\}$ なる集合に定義域を限ることで保証)

Theorem

$\bar{x}_T = \frac{1}{T+1}\sum_{t=0}^{T} x_t$ *(Polyak-Ruppert 平均化)* とする.
ステップサイズを $\eta_t = \frac{\eta_0}{\sqrt{t}}$ とすれば

$$\mathbb{E}_{z_{1:T}}[L_\psi(\bar{x}_T) - L_\psi(x^*)] \le \frac{\eta_0 G^2 + D^2/\eta_0}{\sqrt{T}}.$$

- $\eta_0 = \frac{D}{G}$ とすれば, 期待誤差の上界は

$$\frac{2GD}{\sqrt{T}}.$$

- これはミニマックス最適 (定数倍を除いて).
- G は ψ と関係ない. →近接写像のおかげ.
  ※ L_1 正則化では $\|\partial\psi(x)\| \le C\sqrt{p}$ である.

SGD の収束解析 (平滑な目的関数)

仮定

(A1') L は γ-平滑，$\mathbb{E}[\|g_t - \mathbb{E}[g_t]\|^2] = \sigma^2$.

(A2) $\mathbb{E}[\|x_t - x^*\|^2] \leq D^2$.

(仮定 (A2) は $\{x \mid \|x\| \leq D/2\}$ なる集合に定義域を限ることで保証)

Theorem

$\bar{x}_T = \frac{1}{T+1}\sum_{t=0}^{T} x_t$ (*Polyak-Ruppert* 平均化) とする.
$\forall \mu' > 0$,

$$\mathbb{E}_{z_{1:T}}[L_\psi(\bar{x}_T) - L_\psi(x^*)] \leq \frac{1}{2T}\sum_{t=1}^{T}\left(\frac{1}{\eta_{t+1}} - \frac{1}{\eta_t}\right)D^2 + \frac{1}{2T\eta_1}D^2 + \frac{1}{2}\sum_{t=1}^{T}\left(\gamma - \frac{1}{2\eta_t}\right)\|\beta_t - \beta_{t+1}\|^2 + \frac{\sigma^2}{T}\sum_{t=1}^{T}\eta_t$$

- $\eta_t = \frac{D}{\sigma}\frac{1}{\sqrt{t}}$ としてかつ $\frac{1}{2\gamma} > \eta_t$ なら, 期待誤差の上界は $O(\frac{\sigma D}{\sqrt{T}})$.
- $\sigma^2 = 0$ (ノイズなし) なら，$\eta_t = 1/(2\gamma)$ とすることで期待誤差 $= O(\frac{\gamma}{T}D^2)$ が得られ，通常の勾配法のレートが復元される.

証明

$\psi = 0$ で示す.

- $L(x_t) \le L(x_{t-1}) + \nabla^\top L(x_{t-1})(x_t - x_{t-1}) + \frac{\gamma}{2}\|x_t - x_{t-1}\|^2$
- $L(x_{t-1}) + \nabla^\top L(x_{t-1})(x^* - x_{t-1}) \le L(x^*)$

に注意する. この二式を足すことで,

$$
\begin{aligned}
& L(x_t) - L(x^*) \\
& \le \nabla^\top L(x_{t-1})(x_t - x^*) + \frac{\gamma}{2}\|x_t - x_{t-1}\|^2 = \langle g_t + \epsilon_t, x_t - x^* \rangle + \frac{\gamma}{2}\|x_t - x_{t-1}\|^2 \\
& \le -\frac{1}{\eta_t}\langle x_t - x_{t-1}, x_t - x^* \rangle + \langle \epsilon_t, x_t - x_{t-1} + x_{t-1} - x^* \rangle + \frac{\gamma}{2}\|x_t - x_{t-1}\|^2 \\
& \le -\frac{1}{\eta_t}\langle x_t - x_{t-1}, x_t - x^* \rangle + \frac{1}{4\eta_t}\|x_t - x_{t-1}\|^2 + \eta_t\|\epsilon_t\|^2 \\
& \quad + \frac{\gamma}{2}\|x_t - x_{t-1}\|^2 + \langle \epsilon_t, x_{t-1} - x^* \rangle \\
& = \frac{1}{2\eta_t}\left(\|x_{t-1} - x^*\|^2 - \|x_{t-1} - x_t\|^2 - \|x_t - x^*\|^2\right) \\
& \quad + \frac{1}{4\eta_t}\|x_t - x^*\|^2 + \eta_t\|\epsilon_t\|^2 + \frac{\gamma}{2}\|x_t - x_{t-1}\|^2 \\
& = \frac{1}{2\eta_t}\left(\|x_{t-1} - x^*\|^2 - \|x_t - x^*\|^2\right) + \frac{1}{2}\left(\gamma - \frac{1}{2\eta_t}\right)\|x_{t-1} - x_t\|^2 \\
& \quad + [\eta_t\|\epsilon_t\|^2 + \langle \epsilon_t, x_{t-1} - x^* \rangle] \quad (\leftarrow [\cdot] \text{ の期待値} \le \eta_t\sigma^2 + 0).
\end{aligned}
$$

証明 (続)

あとは両辺期待値取って，$t = 1, \ldots, T$ で足し合わせればよい．
ほぼ通常の勾配法の評価方法と同じだが，ノイズが乗った分だけ $\frac{\sigma^2}{T} \sum_{t=1}^{T} \eta_t$ だけずれる．
この後出てくる分散縮小勾配降下法なども基本はこの評価式．

SGD の収束解析 (強凸)

仮定

(A1) $\mathbb{E}[\|g_t\|^2] \leq G^2$.

(A3) L_ψ は μ-強凸.

Theorem

$\bar{x}_T = \frac{1}{T+1}\sum_{t=0}^{T} x_t$ とする.
ステップサイズを $\eta_t = \frac{1}{\mu t}$ とすれば,

$$\mathbb{E}_{z_{1:T}}[L_\psi(\bar{x}_T) - L_\psi(x^*)] \leq \frac{G^2 \log(T)}{T\mu}.$$

- 非強凸な場合よりも速い収束.
- しかし, これはミニマックス最適ではない.
- 上界自体はタイト (Rakhlin et al., 2012).

強凸目的関数における多項式平均化

仮定

(A1) $\mathbb{E}[\|g_t\|^2] \leq G^2$.

(A3) L_ψ は μ-強凸.

更新則を

$$x_t = \mathrm{prox}\left(x_{t-1} - \eta_t \frac{t}{t+1} g_t | \eta_t \psi \right),$$

とし，重み付き平均を取る: $\bar{x}_T = \frac{2}{(T+1)(T+2)} \sum_{t=0}^{T} (t+1) x_t$.

Theorem

$\eta_t = \frac{2}{\mu t}$ に対し，　$\mathbb{E}_{z_{1:T}}[L_\psi(\bar{x}_T) - L_\psi(x^*)] \leq \frac{2G^2}{T\mu}$ である.

$\log(T)$ が消えた.

これはミニマックス最適.

一般化したステップサイズと荷重方策

s_t $(t = 1, 2, \ldots, T+1)$ を $\sum_{t=1}^{T+1} s_t = 1$ なる数列とする.

$$x_t = \mathrm{prox}\left(x_{t-1} - \eta_t \frac{s_t}{s_{t+1}} g_t | \eta_t \psi \right) \quad (t = 1, \ldots, T)$$

$$\bar{x}_T = \sum_{t=0}^{T} s_{t+1} x_t.$$

仮定: (A1) $\mathbb{E}[\|g_t\|^2] \leq G^2$, (A2) $\mathbb{E}[\|x_t - x^*\|^2] \leq D^2$, (A3) L_ψ は μ-強凸.

Theorem

$$\mathbb{E}_{z_{1:T}}[L_\psi(\bar{x}_T) - L_\psi(x^*)]$$
$$\leq \sum_{t=1}^{T} \frac{s_{t+1}\eta_{t+1}}{2} G^2 + \sum_{t=0}^{T-1} \frac{\max\{\frac{s_{t+2}}{\eta_{t+1}} - s_{t+1}(\frac{1}{\eta_t} + \mu), 0\} D^2}{2}$$

ただし $t = 0$ では $1/\eta_0 = 0$ とする.

特別な例

重み s_t をステップサイズ η_t に比例させてみる (ステップサイズを重要度とみなす):

$$s_t = \frac{\eta_t}{\sum_{\tau=1}^{T+1} \eta_\tau}.$$

この設定では，前述の定理より

$$\mathbb{E}_{z_{1:T}}[L_\psi(\bar{x}_T) - L_\psi(x^*)] \leq \frac{\sum_{t=1}^{T} \eta_t^2 G^2 + D^2}{2\sum_{t=1}^{T} \eta_t}$$

$$\sum_{t=1}^{\infty} \eta_t = \infty$$

$$\sum_{t=1}^{\infty} \eta_t^2 < \infty$$

ならば収束．遠くまで到達できて，かつ適度に減速.

確率的最適化による学習は「速い」

計算量と汎化誤差の関係

- 強凸な汎化誤差の最適な収束レートは $O(1/n)$ (n はデータ数).
- $O(1/n)$ なる汎化誤差を達成するには，訓練誤差も $O(1/n)$ まで減少させないといけない.

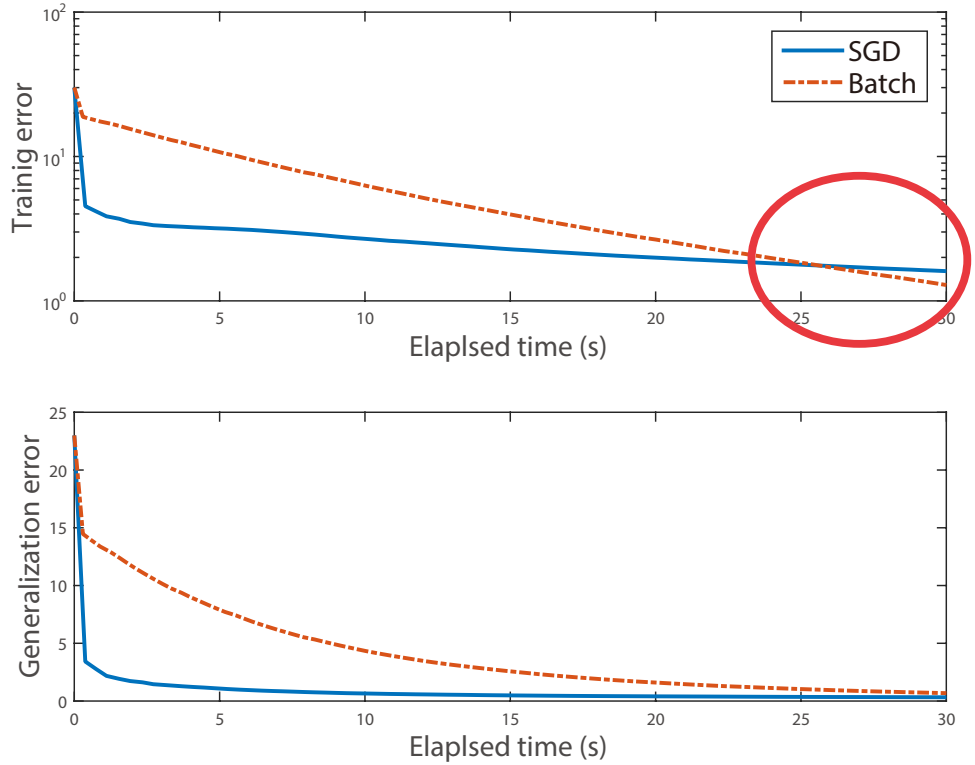
	通常の勾配法	SGD
反復ごとの計算量	n	1
誤差 ϵ までの反復数	$\log(1/\epsilon)$	$1/\epsilon$
誤差 ϵ までの計算量	$n\log(1/\epsilon)$	$1/\epsilon$
誤差 $1/n$ までの計算量	$n\log(n)$	n

(Bottou, 2010)

SGD は $O(\log(n))$ だけ通常の勾配法よりも「速い」.

「n 個データ見るまで減少せず」　vs　「n 個データ見れば $1/n$ まで減少」

典型的な振る舞い



Normal gradient descent vs. SGD

L_1 正則化付きロジスティック回帰: $n = 15,000$, $p = 1,000$.

最初 SGD は訓練誤差と汎化誤差をともに早く減少させる．訓練誤差は途中で全データを使う勾配法に追いつかれ抜かれる．汎化誤差は SGD が優位なことが多い．

SGD に対する Nesterov の加速法

仮定:

- 期待誤差 $L(x)$ は γ-平滑.
- 勾配の分散は σ^2:

$$\mathbb{E}_Z[\|\nabla_\beta \ell(Z,\beta) - \nabla L(\beta)\|^2] \leq \sigma^2.$$

$\rightarrow$ Nesterov の加速によって，より速い収束を実現.

- SGD の加速: Hu et al. (2009)
- SDA の加速: Xiao (2010), Chen et al. (2012)
- 非凸関数も含めたより一般的な枠組み: Lan (2012), Ghadimi and Lan (2012, 2013)

(非強凸) $$\mathbb{E}_{z_{1:T}}[L_\psi(x^{(T)})] - L_\psi(x^*) \leq C\left(\frac{\sigma D}{\sqrt{T}} + \frac{D^2\gamma}{T^2}\right)$$

(強凸) $$\mathbb{E}_{z_{1:T}}[L_\psi(x^{(T)})] - L_\psi(x^*) \leq C\left(\frac{\sigma^2}{\mu T} + \exp\left(-C\sqrt{\frac{\mu}{\gamma}}T\right)\right)$$

加速 SGD のミニバッチ法による加速

$$\mathbb{E}_{z_{1:T}}[L_\psi(x^{(T)})] - L_\psi(x^*) \leq C\left(\frac{\sigma D}{\sqrt{T}} + \frac{D^2\gamma}{T^2}\right)$$

σ^2 は勾配の推定量の分散:

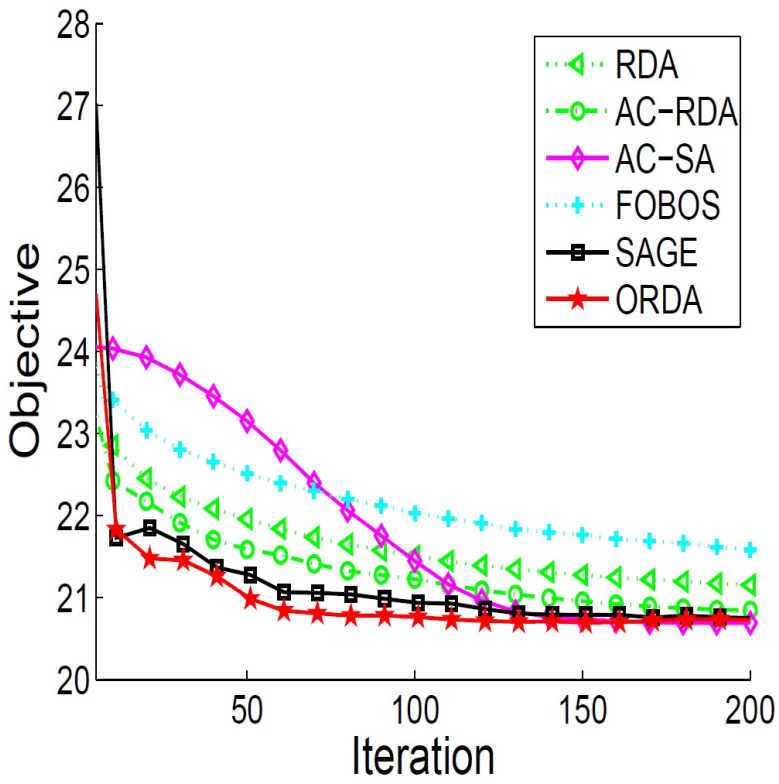
$$\mathbb{E}_Z[\|\nabla_\beta \ell(Z,\beta) - \nabla L(\beta)\|^2] \leq \sigma^2.$$

分散はミニバッチ法を用いることで減少させることが可能:

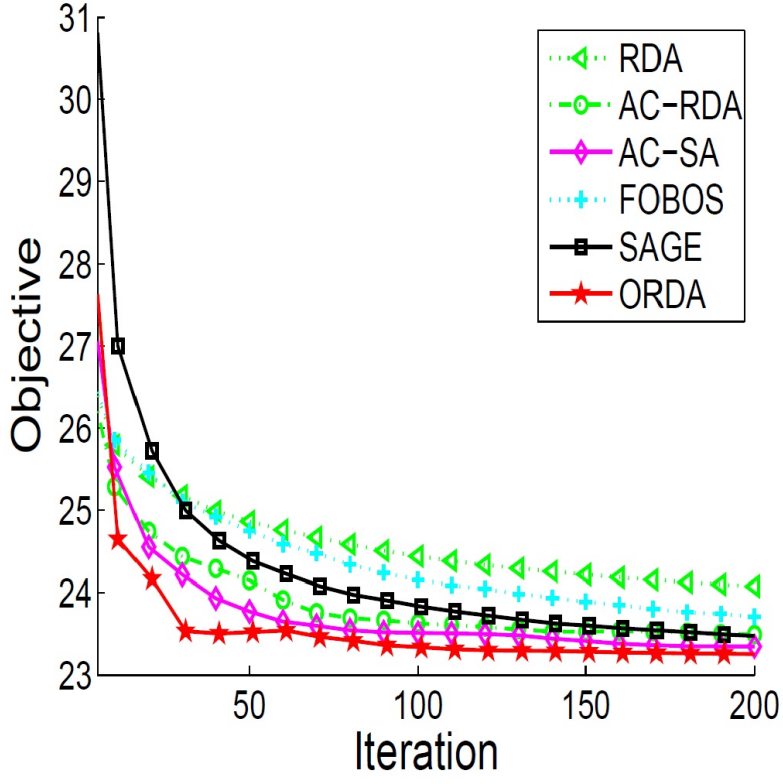
$$\boxed{g = \nabla \ell(z, x^{(t-1)})} \quad \Rightarrow \quad \boxed{g = \frac{1}{K}\sum_{k=1}^{K} \nabla \ell(z_k, x^{(t-1)})}$$

(分散) σ^2 σ^2/K

- K 個の勾配を計算するのは並列化できる.
- $\sigma \to 0$ できれば，上記の上界は $O(1/T^2)$ となる．これは非確率的な Nesterov の加速法と同じ収束速度.



(a) Objective for L_1



(b) Objective for Elastic-net

人工データにおける数値実験 (a) L_1 正則化 (Lasso)，(b) エラスティックネット正則化 (figure is from Chen et al. (2012)).

SAGE: Accelerated SGD (Hu et al., 2009), AC-RDA: Accelerated stochastic RDA (Xiao, 2010), AC-SA: Accelerated stochastic approximation Ghadimi and Lan (2012), ORDA: Optimal stochastic RDA (Chen et al., 2012)

オンライン型確率的最適化のまとめ

- 各更新でたった一つのデータを見るだけで良い.
- ある程度の汎化誤差を得るまでが早い.
- Nesterov の加速法との組合せも可能: Hu et al. (2009), Xiao (2010), Chen et al. (2012).
- AdaGrad (Duchi et al., 2011) などによる計量の自動調節法もある.
- 期待誤差の収束レート:
  - $\frac{GR}{\sqrt{T}}$ (非平滑, 非強凸) Polyak-Ruppert 平均化
  - $\frac{G^2}{\mu T}$ (非平滑, 強凸) 多項式平均化
  - $\frac{\sigma R}{\sqrt{T}} + \frac{R^2 L}{T^2}$ (平滑, 非強凸) 加速
  - $\frac{\sigma^2}{\mu T} + \exp\left(-\sqrt{\frac{\mu}{L}} T\right)$ (平滑, 強凸) 加速

G: 勾配のノルムの上界, R: 定義域の半径, L: 平滑性, μ: 強凸性, σ: 勾配の分散

確率的一次最適化法のミニマックス最適レート

$$\min_{x\in\mathcal{B}} L(x) = \min_{x\in\mathcal{B}} \mathbb{E}_Z[\ell(Z,x)]$$

Condition

- $\hat{g}_x \in \partial_x \ell(Z,x)$ の期待値は有界: $\|\mathbb{E}[\hat{g}_x]\| \le G$ $(\forall x \in \mathcal{B})$.
- ドメイン $\mathcal{B}$ は半径 R の球を含む.
- $L(x)$ は μ-強凸 ($\mu = 0$ も許す).

Theorem (ミニマックス最適レート (Agarwal et al., 2012, Nemirovsky and Yudin, 1983))

任意の確率的一次最適化法に対し，上記の条件を満たすある損失関数 ℓ と分布 $P(Z)$ が存在して，

$$\mathbb{E}[L(x^{(T)}) - L(x^*)] \ge c \min\left\{\frac{GR}{\sqrt{T}}, \frac{G^2}{\mu T}, \frac{GR}{\sqrt{p}}\right\}.$$

SGD はこの最適レートを達成する.
一次最適化法: 損失関数の値およびクエリ点 x における勾配 $(\ell(Z,x), \hat{g}_x)$ にのみ依存するアルゴリズム. (SGD は含まれている)

Outline

オンラインとバッチの違い

バッチ型ではデータは固定. ただ単に訓練誤差を最小化する:

$$P(x) = \frac{1}{n}\sum_{i=1}^{n} \ell_i(x) + \psi(x).$$

バッチ型手法

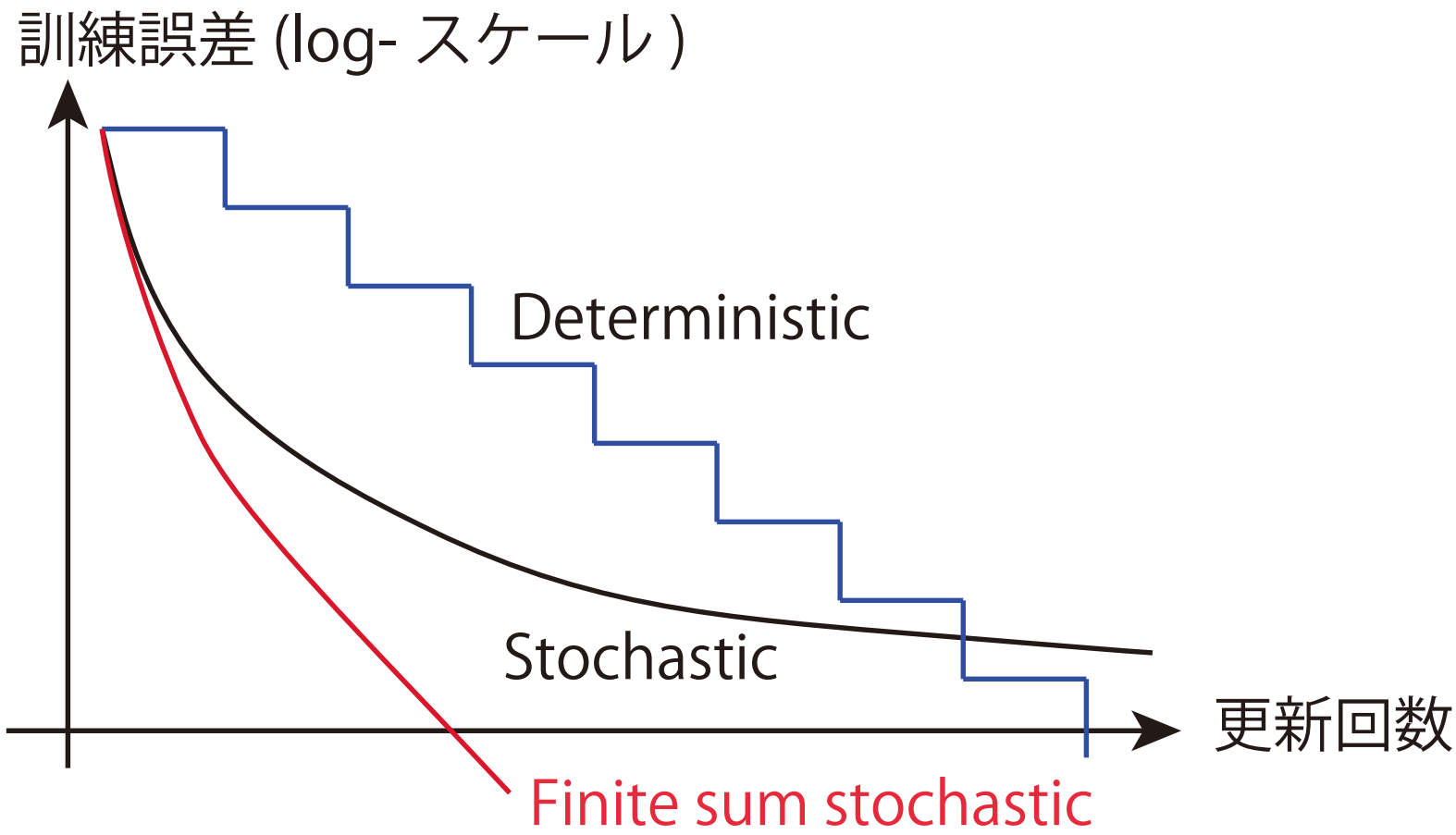
- 1 回の更新に 1 データのみ利用 (オンライン型と同じ)
- 線形収束する (オンライン型と違う):

$$T > (n + \gamma/\lambda)\log(1/\epsilon)$$

回の更新で ϵ 誤差を達成. ただし γ-平滑な損失と λ-強凸な正則化を仮定.

※通常の勾配法は $n\gamma/\lambda \log(1/\epsilon)$ なる計算量.

確率的手法と非確率的手法の良いとこどりをしたい.


訓練誤差 (log- スケール)
Deterministic
Stochastic
更新回数
Finite sum stochastic

代表的な三つの方法

- 確率的分散縮小勾配法
  Stochastic Variance Reduced Gradient descent, SVRG (Johnson and Zhang, 2013, Xiao and Zhang, 2014)
- 確率的平均勾配法
  Stochastic Average Gradient descent, SAG (Le Roux et al., 2012, Schmidt et al., 2013, Defazio et al., 2014)
- 確率的双対座標降下法
  Stochastic Dual Coordinate Ascent, SDCA (Shalev-Shwartz and Zhang, 2013)

- SAG と SVRG は主問題を解く方法.
- SDCA は双対問題を解く方法.

問題設定

$$P(x) = \underbrace{\frac{1}{n}\sum_{i=1}^{n} \ell_i(x)}_{\text{平滑}} + \underbrace{\psi(x)}_{\text{強凸}}$$

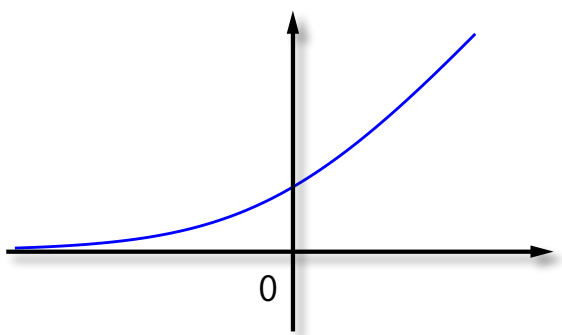
仮定:

- ℓ_i: 損失関数は γ-平滑.
- ψ: 正則化関数は λ-強凸. 典型的には $\lambda = O(1/n)$ または $O(1/\sqrt{n})$.

例:

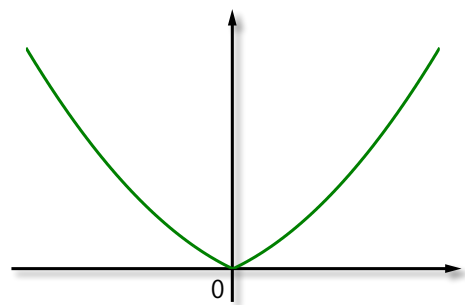
損失関数

- 平滑化ヒンジ損失
- ロジスティック損失



正則化関数

- L_2 正則化
- エラスティックネット正則化
- $\tilde{\psi}(x) + \lambda\|x\|^2$



Outline

主問題を扱う方法

アイディア: 勾配の分散を小さくする.

$$\underbrace{\frac{1}{n}\sum_{i=1}^{n}\ell_i(x)}_{\text{どうやって近似するか?}} + \psi(x)$$

主問題を扱う方法

アイディア: 勾配の分散を小さくする.

$$\underbrace{\frac{1}{n}\sum_{i=1}^{n}\ell_i(x)}_{\text{どうやって近似するか?}} + \psi(x)$$

$\langle g, x \rangle$

オンライン型手法: $\hat{i} \in \{1, \ldots, n\}$ をランダムに選択し，線形近似.

$$g = \nabla \ell_{\hat{i}}(x) \quad \Rightarrow \quad \mathbb{E}[g] = \tfrac{1}{n}\textstyle\sum_{i=1}^{n} \nabla \ell_i(x)$$

つまり勾配の不偏推定量.

分散は?

→ 分散が問題!

→ バッチの設定では分散を縮小させることが容易.

確率的分散縮小勾配法

SVRG (Johnson and Zhang, 2013, Xiao and Zhang, 2014)

$$\min_x\{L(x)+\psi(x)\} = \min_x\{\tfrac{1}{n}\textstyle\sum_{i=1}^n \ell_i(x)+\psi(x)\}$$

ある x に近い参照点 $\hat{x}$ に対し，分散を縮小した勾配を次のように求める:

$$g = \nabla\ell_i(x) - \nabla\ell_i(\hat{x}) + \underbrace{\tfrac{1}{n}\textstyle\sum_{j=1}^n \nabla\ell_j(\hat{x})}_{\nabla L(\hat{x})}.$$

偏り: 不偏

$$\mathbb{E}[g] = \frac{1}{n}\sum_{i=1}^n [\nabla\ell_i(x) - \nabla\ell_i(\hat{x}) + \nabla L(\hat{x})] = \frac{1}{n}\sum_{i=1}^n \nabla\ell_i(x) = \nabla L(x).$$

分散は?

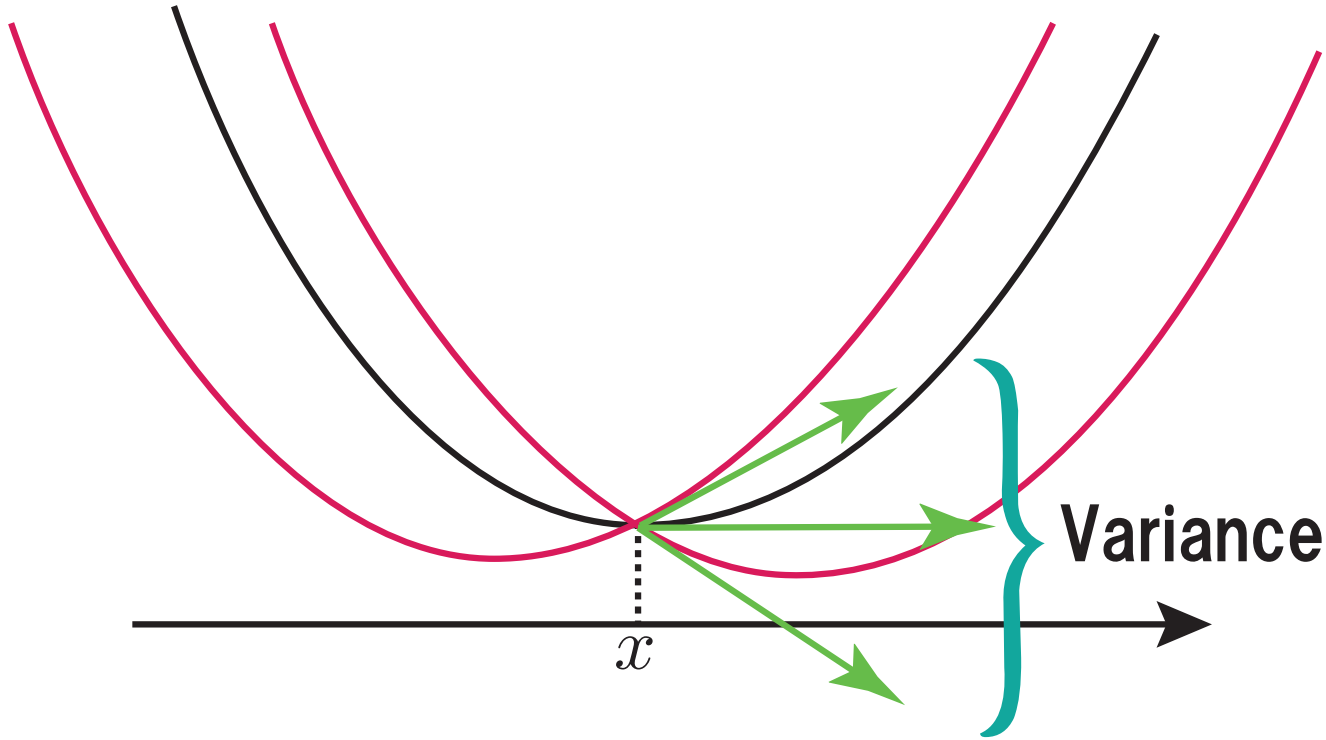
分散の縮小

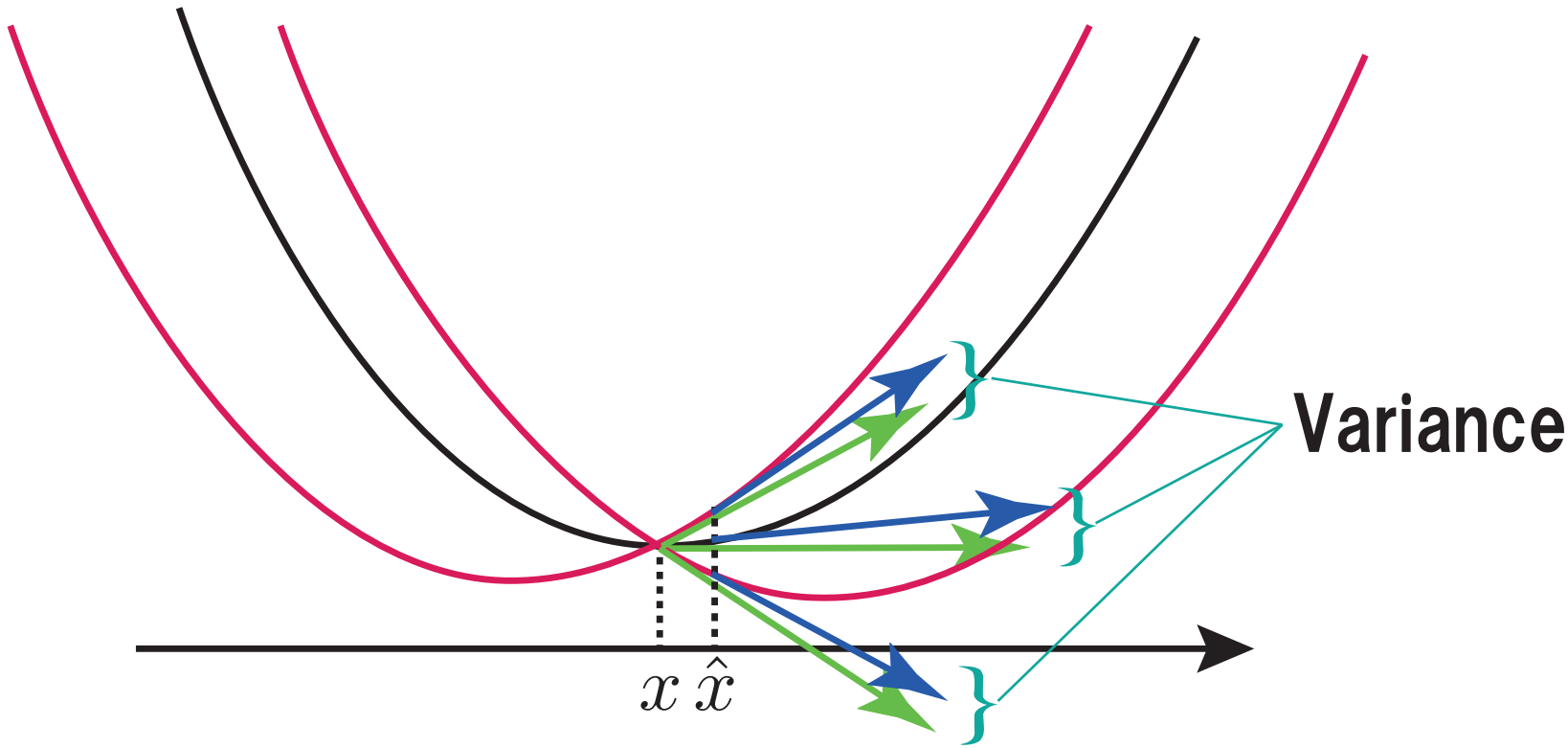
$$g = \nabla \ell_i(x) - \nabla \ell_i(\hat{x}) + \nabla L(\hat{x}).$$

分散:

$$\begin{aligned}
\mathrm{Var}[g] &= \tfrac{1}{n} \textstyle\sum_{i=1}^{n} \|\nabla \ell_i(x) - \nabla \ell_i(\hat{x}) + \nabla L(\hat{x}) - \nabla L(x)\|^2 \\
&= \tfrac{1}{n} \textstyle\sum_{i=1}^{n} \left\|\nabla \ell_i(x) - \nabla \ell_i(\hat{x})\|^2 - \|\nabla L(\hat{x}) - \nabla L(x)\right\|^2 \\
&\qquad (\because \mathrm{Var}[X] = \mathbb{E}[\|X\|^2] - \|\mathbb{E}[X]\|^2) \\
&\leq \tfrac{1}{n} \textstyle\sum_{i=1}^{n} \|\nabla \ell_i(x) - \nabla \ell_i(\hat{x})\|^2 \\
&\leq \gamma \|x - \hat{x}\|^2.
\end{aligned}$$

ℓ_i が平滑で，x と $\hat{x}$ が近ければ分散も小さい.


Variance
x


Variance
x
$\hat{x}$

SVRG の手順

SVRG

$t = 1, 2, \ldots$ において, 以下の手順を繰り返す:

1. $\hat{x} = \hat{x}^{(t-1)}$, $x_{[0]} = \hat{x}$ として,

$$\hat{g} = \nabla L(\hat{x}) = \frac{1}{n} \sum_{i=1}^{n} \nabla \ell_i(\hat{x}). \quad \text{(全勾配)}$$

2. $k = 1, \ldots, m$ において以下を実行:
   1. $i \in \{1, \ldots, n\}$ を一様にサンプル.
   2. 分散縮小された勾配を計算:

$$g = \nabla \ell_i(x_{[k-1]}) - \nabla \ell_i(\hat{x}) + \hat{g}. \quad \text{(分散縮小)}$$

   3. $x_{[k]}$ を次のように更新:

$$x_{[k]} = \mathrm{prox}(x_{[k-1]} - \eta g | \eta\psi).$$

3. $\hat{x}^{(t)} = \frac{1}{m} \sum_{k=1}^{m} x_{[k]}$ とする.

t 反復までの勾配の計算回数: $t \times (n + 2m)$.

収束解析

仮定: ℓ_i は γ-平滑で，ψ は λ-強凸.

Theorem

ステップサイズ η と内部反復回数 m が $\eta < 1/(4\gamma)$ を満たし，

$$\rho := \frac{\eta^{-1}}{\lambda(1-4\gamma\eta)m} + \frac{4\gamma(m+1)\eta}{(1-4\gamma\eta)m} < 1,$$

なら，T 反復後の目的関数値は次のように上から抑えられる:

$$\mathbb{E}[P(\hat{x}^{(T)}) - P(x^*)] \leq \rho^T (P(\hat{x}^{(0)}) - P(x^*)).$$

定理の仮定は m が次を満たすことを要請:

$$m \geq \Omega\left(\frac{\gamma}{\lambda}\right).$$

- 内部反復の計算量: $O(n+m)$
- 誤差 ϵ までの外部反復の回数: $T = O(\log(1/\epsilon))$

$\Rightarrow$ 全体的な計算量:

$$O\left((n+m)\log(1/\epsilon)\right) = O\left(\left(n + \frac{\gamma}{\lambda}\right)\log(1/\epsilon)\right)$$

非確率的手法との比較

$\mathbb{E}[P(x^{(T)}) - P(x^*)] \leq \epsilon$ までの計算量を比較.
$\kappa = \gamma/\lambda$ とする (条件数).

- SVRG:

$$\boxed{(n+\kappa)\log(1/\epsilon)}$$

$\Omega((n+\kappa)\log(1/\epsilon))$ 回の反復 $\times$ 1 反復ごと $\Omega(1)$

- 非確率的勾配法:

$$\boxed{n\kappa\log(1/\epsilon)}$$

$\Omega(\kappa\log(1/\epsilon))$ 回の反復 $\times$ 1 反復ごと $\Omega(n)$

データ数 $n = 100,000$, 正則化パラメータ $\lambda = 1/1000$, 平滑性 $\gamma = 1$:

$$n \times \kappa = 10^8, \quad n + \kappa = 10^5.$$

1000 倍

SVRG 等バッチ型手法の加速

Catalyst: SVRG, SAG, SAGA の加速 (汎用的な手法)

Catalyst (Lin et al., 2015)

For $t = 1, 2, \ldots$:

1. 強凸性を強めた以下の問題を解く:

$$x^{(t)} \simeq \mathrm{argmin}_x \left\{ P(x) + \frac{\alpha}{2} \|x - y^{(t-1)}\|^2 \right\} \quad (\text{up to } \epsilon_t \text{ precision}).$$

2. 解を "加速" する: $y^{(t)} = x^{(t)} + \beta_t (x^{(t)} - x^{(t-1)})$.

– Catalyst は近似的近接点法の加速法にあたる.
– $\epsilon_t = C(1 - \sqrt{\lambda/2(\lambda+\alpha)})^t$ とすれば,

$$P(x^{(t)}) - P(x^*) \leq C' \left(1 - \sqrt{\frac{\lambda}{2(\lambda+\alpha)}}\right)^t.$$

– $\alpha = \max\{c\frac{\gamma}{n}, \lambda\}$ として SVRG, SAG, SAGA を内部ループに適用すれば, 全体で $(n + \sqrt{\frac{\gamma n}{\lambda}}) \log(1/\epsilon)$ の計算量で済む.
– 汎用的な手法ではあるが, 内部ループの更新回数や α の選択に敏感.
– APPA (Frostig et al., 2015) も似たような手法.

Katyusha：SVRG の加速法

Katyusha

Katyusha = SVRG + Inner Acceleration + Katyusha momentum
(Allen-Zhu, 2016, 2017) (STOC2017)

Katyusha は **"negative momentum"** を内部ループで用いる:

$$
\begin{aligned}
y_k &= \theta\{\underbrace{x_{k-1} + \beta(x_{k-1} - x_{k-2})}_{\text{acceleration}}\} + (1-\theta)\ \underbrace{x_0}_{\text{magnet}} \\
&= x_{k-1} + \theta\beta(x_{k-1} - x_{k-2}) + \underbrace{(1-\theta)(x_0 - x_{k-1})}_{\text{Katyusha momentum}}
\end{aligned}
$$

Katyusha

Katyusha (Allen-Zhu (2016))

For $s = 0, \ldots, S-1$:

1. $\mu^{(s)} = \nabla L(\hat{x}^{(s)})$.
2. For $j = 0, \ldots, m-1$:
   1. $k = (sm) + j$.
   2. $x_{[k+1]} = \tau_1 z_k + \tau_2 \hat{x}^{(s)} + (1 - \tau_1 - \tau_2) y_{[k]}$ (加速ステップ)
   3. $i \in \{1, \ldots, n\}$ を一様にサンプリングし，分散縮小勾配を計算:

$$g = \mu^{(s)} + \nabla \ell_i(x_{[k+1]}) - \nabla \ell_i(\hat{x}^{(s)}) \qquad (\text{分散縮小勾配}).$$

   4. $z_{[k+1]} = \arg\min_z \{ \frac{1}{2\alpha} \|z - z_{[k]}\|^2 + \langle g, z \rangle + \psi(z) \}$.
   5. $y_{[k+1]} = \arg\min_y \{ \frac{3L}{2} \|y - x_{[k+1]}\|^2 + \langle g, y \rangle + \psi(y) \}$
3. $\hat{x}^{(s+1)} = \frac{1}{\sum_{j=0}^{m-1} (1+\alpha\sigma)^j} \left(\sum_{j=0}^{m-1} (1+\alpha\sigma)^j y_{sm+j+1} \right)$

- 通常の強凸関数に関する加速は $x_{[k+1]} = \tau_1 z_k + (1 - \tau_1) y_{[k]}$ とするが，$\hat{x}^{(s)}$ 成分も入れることで SVRG でも加速を達成.
  →以前の解に近づけるので「negative momentum」と言う.
- τ_1, τ_2, α は適切に設定（次のページ）

Katyusha の計算量 (強凸)

- ℓ_i が γ-平滑かつ，ψ が λ-強凸の時:
  $m = 2n$, $\tau_1 = \min\{\sqrt{\frac{m\lambda}{3\gamma}}, \frac{1}{2}\}$, $\tau_2 = \frac{1}{2}$, $\alpha = \frac{1}{3\tau_2\gamma}$ とすることで

$$\boxed{(n + \sqrt{n\tfrac{\gamma}{\lambda}})\log(1/\epsilon)}$$

  回の更新回数で誤差 ϵ まで到達.

SVRG $(n + \frac{\gamma}{\lambda})\log(1/\epsilon)$ $\rightarrow$ Katyusha $(n + \sqrt{n\frac{\gamma}{\lambda}})\log(1/\epsilon)$ に改善

近接勾配法の加速法は $O(n\sqrt{\gamma/\lambda}\log(1/\epsilon))$ なので**最大で $\sqrt{n}$ 倍速い.**
$n = 10^6$ なら 1000 倍速い.

Katyusha の計算量 (非強凸)

- ℓ_i が γ-平滑かつ，ψ が強凸とは限らない時:
  $m = 2n$, $\tau_1 = \frac{2}{s+4}$, $\tau_2 = \frac{1}{2}$, $\alpha = \frac{1}{3\tau_1\gamma}$ (τ_1 と α は各 s ごとに変化させる) とすれば

$$\frac{n}{\sqrt{\epsilon}}\sqrt{P(\hat{x}^{(0)} - P(x^*)} + \sqrt{\frac{n\gamma}{\epsilon}}\|\hat{x}^{(0)} - x^*\| = \Omega\left(\frac{n}{\sqrt{\epsilon}}\right)$$

  回の更新回数で誤差 ϵ まで達成.

さらに，AdaptReg (Allen-Zhu and Hazan (2016)) という方法と組み合わせることで

$$n\log(1/\epsilon) + \sqrt{\frac{n\gamma}{\epsilon}}$$

まで改善することができる.

$\Rightarrow$ 近接勾配法の加速法の計算量 $O(n\sqrt{\gamma/\epsilon})$ と比べて $\sqrt{n}$ 倍速い.

Katyusha の計算量まとめ

SVRG の加速を達成.

計算量の比較

	μ-strongly convex	Non-strongly convex
GD	$O\left(n\log\left(\frac{1}{\epsilon}\right)\right)$	$O(n\sqrt{\frac{L}{\epsilon}})$
SVRG	$O\left((n+\kappa)\log\left(\frac{1}{\epsilon}\right)\right)$	$O\left(\frac{n+L}{\epsilon}\right)$
Katyusha	$O\left(\left(n+\sqrt{n\kappa}\right)\log\left(\frac{1}{\epsilon}\right)\right)$	$O\left(n\log\left(\frac{1}{\epsilon}\right)+\sqrt{\frac{nL}{\epsilon}}\right)$

($\kappa = L/\mu$: 条件数)

強凸の加速レートは確率的座標降下法の APCG (Lin et al., 2014) や主双対型の SPDC (Zhang and Lin, 2015) でも達成できる.

さらに改善：ミニバッチ法 + 加速
（二重加速分散縮小法）

Doubly accelerated stochastic variance reduced dual averaging method (Murata and Suzuki, NIPS2017)

$\min_x P(x) = \frac{1}{n}\sum_{i=1}^n f_i(x) + \psi(x) = F(x) + \psi(x)$

ミニバッチ法: 各更新で $I \subset [n]$ $(|I| = b)$ なるミニバッチをランダムに選択

$$g_t = \frac{1}{|I|}\sum_{i\in I}(\nabla_x f_i(x_t) - \nabla_x f_i(\hat{x})) + \frac{1}{n}\sum_{i=1}^n \nabla_x f_i(\hat{x})$$

ミニバッチ **SVRG** + 二重加速法

Doubly accelerated stochastic variance reduced dual averaging method (Murata and Suzuki, NIPS2017)

$\min_x P(x) = \frac{1}{n}\sum_{i=1}^n f_i(x) + \psi(x) = F(x) + \psi(x)$

ミニバッチ法: 各更新で $I \subset [n]$ $(|I| = b)$ なるミニバッチをランダムに選択

$$g_t = \frac{1}{|I|}\sum_{i\in I}(\nabla_x f_i(x_t) - \nabla_x f_i(\hat{x})) + \frac{1}{n}\sum_{i=1}^n \nabla_x f_i(\hat{x})$$

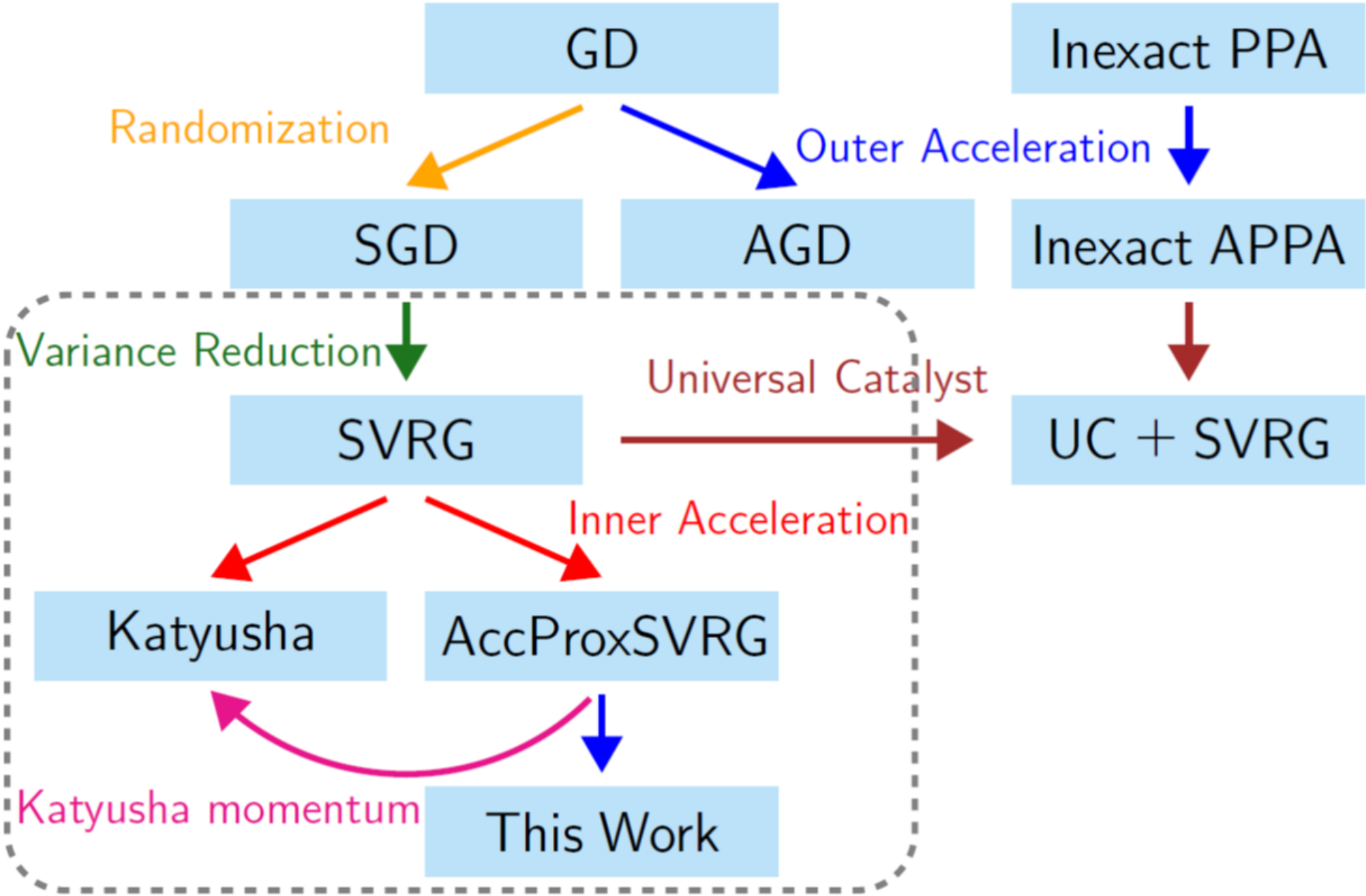
ミニバッチ SVRG + 二重加速法

- ミニバッチサイズへの依存性を改善.
- Katyusha と比べてチューニングパラメータが少ない.

誤差 ϵ までの計算量. ($b = |I|$ はミニバッチサイズ)

	Katyusha	**DASVRDA**
Strong conv.	$O\left(\left(n + \sqrt{bn\kappa}\right)\log(1/\epsilon)\right)$	$O\left(\left(n + \sqrt{n\kappa} + b\sqrt{\kappa}\right)\log(1/\epsilon)\right)$
Non-Strong	$O\left(n\log(1/\epsilon) + \sqrt{b\frac{nL}{\epsilon}}\right)$	$O\left(n\log(1/\epsilon) + \sqrt{\frac{nL}{\epsilon}} + b\sqrt{\frac{L}{\epsilon}}\right)$

最大で $\sqrt{b}$ 倍速い.



AccProxSVRG: Nitanda (2014)

Algorithm (Murata and Suzuki, 2017)

DASVRDA($\tilde{x}_0, \eta, m, b, S$)

Iterate the following for $s = 1, 2, \ldots, S$:

- $\tilde{y}_s = \tilde{x}_{s-1} + \frac{s-2}{s+1}(\tilde{x}_{s-1} - \tilde{x}_{s-2}) + \frac{s-1}{s+1}(\tilde{z}_{s-1} - \tilde{x}_{s-2})$ **(outer acceleration)**
- $(\tilde{x}_s, \tilde{z}_s) =$ One-Pass-AccSVRDA($\tilde{y}_s, \tilde{x}_{s-1}, \eta, \beta, m, b$)

Algorithm (Murata and Suzuki, 2017)

DASVRDA($\tilde{x}_0, \eta, m, b, S$)

Iterate the following for $s = 1, 2, \ldots, S$:

- $\tilde{y}_s = \tilde{x}_{s-1} + \frac{s-2}{s+1}(\tilde{x}_{s-1} - \tilde{x}_{s-2}) + \frac{s-1}{s+1}(\tilde{z}_{s-1} - \tilde{x}_{s-2})$ **(outer acceleration)**
- $(\tilde{x}_s, \tilde{z}_s) =$ One-Pass-AccSVRDA$(\tilde{y}_s, \tilde{x}_{s-1}, \eta, \beta, m, b)$

強凸の場合，さらに「リスタート法」を適用する．

DASVRDA$^{\mathrm{sc}}$($\check{x}_0, \eta, m, b, S, T$)

Iterate the following for $t = 1, 2, \ldots, T$:

- $\check{x}_t =$ DASVRDA$(\check{x}_{t-1}, \eta, m, b, S)$.

One-Pass-AccSVRDA($x_0, \tilde{x}, \eta, m, b$)

Iterate the following for $k = 1, 2, \ldots, m$:

- Sample $I_k \subset \{1, \ldots, n\}$ with size b uniformly.
- $y_k = x_{k-1} + \frac{k-1}{k+1}(x_{k-1} - x_{k-2})$ **(inner acceleration)**
- $v_k = \frac{1}{b}\sum_{i \in I_k}(\nabla f_i(y_k) - \nabla f_i(\tilde{x})) + \nabla F(\tilde{x})$ (variance reduction)
- $\bar{v}_k = \left(1 - \frac{2}{k+1}\bar{v}_{k-1} + \frac{2}{k+1}v_k\right)$ (dual averaging)
- $z_k = \mathrm{prox}(x_0 - \frac{\eta k(k+1)}{4}\bar{v}_k | \frac{\eta k(k+1)}{4}\psi)$ (prox. gradient descent)
- $x_k = \left(1 - \frac{2}{k+1}\right)x_{k-1} + \frac{2}{k+1}z_k$ **(inner acceleration)**

Output: (x_m, z_m)

アイディア: 内部加速 と 外部加速 の組み合わせ (Double acceleration)

- 外側の加速：APG (Accelerated Proximal Gradient)
  新しいモーメンタム項 $\frac{s-1}{s+1}(\tilde{z}_{s-1} - \tilde{x}_{s-2})$ を加える.
- 内側の加速：AccSVRDA = **AccSDA** (Xiao, 2009) + 分散縮小
  AccProxSVRG (Nitanda, 2014) の技法も援用→良いミニバッチ効率性，ミニバッチによる分散縮小を用いてより積極的に加速.

収束解析

仮定:
- 目的関数 P は μ-強凸.
- ℓ_i は γ-平滑.

Theorem (強凸)

$\eta = O\left(\frac{1}{(1+n/b^2)\gamma}\right)$, $S = O(1 + \frac{b}{n}\sqrt{\frac{\gamma}{\mu}} + \sqrt{\frac{\gamma}{n\mu}})$, $T = O(\log(1/\epsilon))$ とする.
$\mathrm{DASVRDA}^{\mathrm{sc}}(\check{x}_0, \eta, n/b, b, S, T)$ (+二段階加速法) は

$$O\left(\left(n + \sqrt{\frac{n\gamma}{\mu}} + b\sqrt{\frac{\gamma}{\mu}}\right)\log\left(\frac{1}{\epsilon}\right)\right)$$

回の勾配計算で $\mathbb{E}[P(\check{x}_T) - P(x^*)] \leq \epsilon$ を達成.

Katyusha	$O((n + \sqrt{nb}\sqrt{\kappa})\log(1/\epsilon))$
DASVRDA	$O((n + (\sqrt{n} + b)\sqrt{\kappa})\log(1/\epsilon))$

$(\kappa := \gamma/\mu)$

$\rightarrow \min\{\sqrt{b}, \sqrt{n/b}\}$ 倍速い.
$b = \max\{\sqrt{n}, n/\sqrt{\kappa}\}$ とすれば, 更新回数は $= \sqrt{\kappa}\log(1/\epsilon)$.

仮定: ℓ_i は γ-平滑.

Theorem (非強凸)

$\eta = O\left(\frac{1}{(1+n/b^2)\gamma}\right)$, $S = O(1 + \frac{b}{n}\sqrt{\frac{\gamma}{\epsilon}} + \sqrt{\frac{\gamma}{n\epsilon}})$ とすれば,
$\mathrm{DASVRDA}(x_0, \eta, n/b, b, S)$ は

$$O\left(n\log(1/\epsilon) + \sqrt{\frac{n\gamma}{\epsilon}} + b\sqrt{\frac{\gamma}{\epsilon}}\right)$$

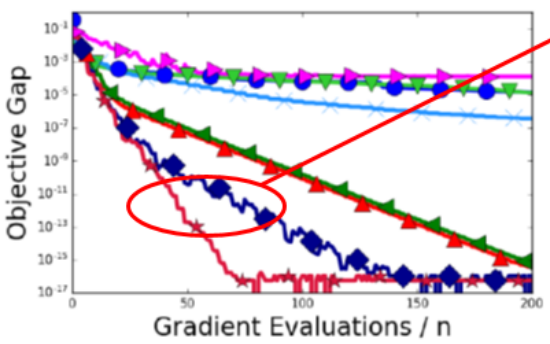
回の勾配計算で $\mathbb{E}[P(\tilde{x}_S) - P(x^*)] \leq \epsilon$ を達成.

- (このレートを達成するには warm-start を最初に入れる必要がある.)

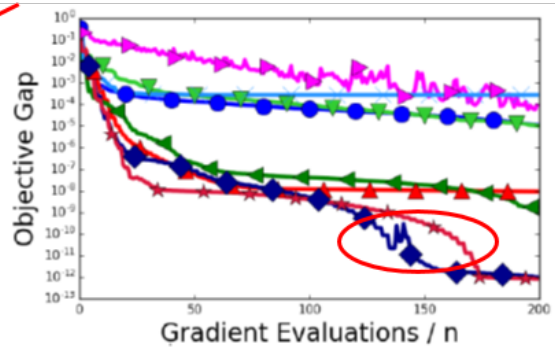
Katyusha	$O((n\log(1/\epsilon) + \sqrt{nb}\sqrt{\gamma/\epsilon}))$
DASVRDA	$O((n\log(1/\epsilon) + (\sqrt{n}+b)\sqrt{\gamma/\epsilon}))$

$\rightarrow \min\{\sqrt{b}, \sqrt{n/b}\}$ 倍速い.
$b = \sqrt{n}$ とすれば, 更新回数は $= \sqrt{\gamma/\epsilon}$.

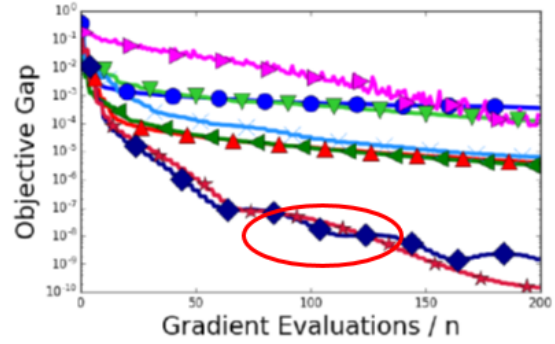
Proposed method



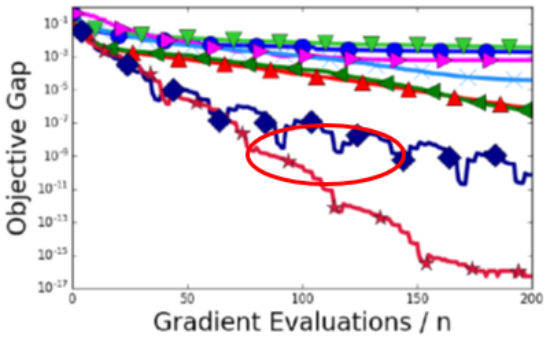
(a) a9a, $(\lambda_1, \lambda_2) = (10^{-4}, 0)$



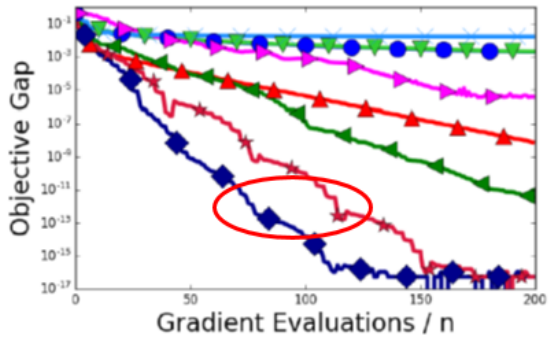
(b) a9a, $(\lambda_1, \lambda_2) = (10^{-4}, 10^{-6})$



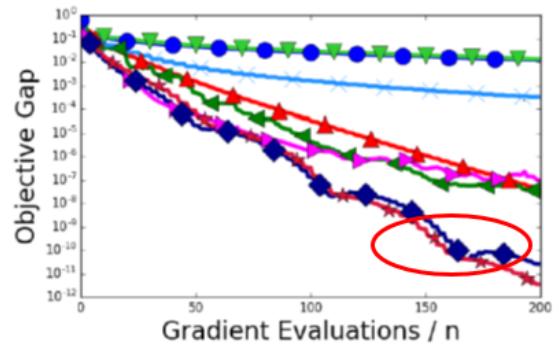
(c) a9a, $(\lambda_1, \lambda_2) = (0, 10^{-6})$



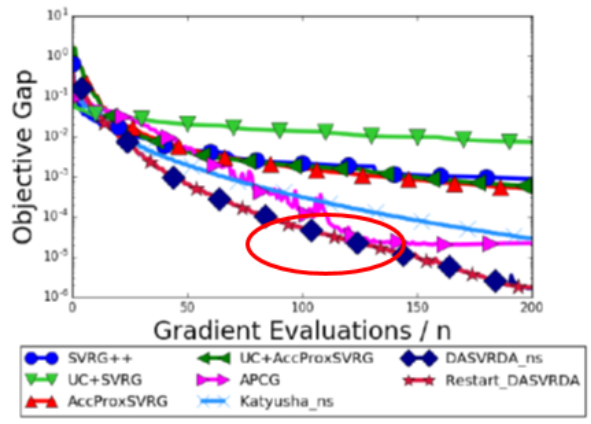
(d) rcv1, $(\lambda_1, \lambda_2) = (10^{-4}, 0)$



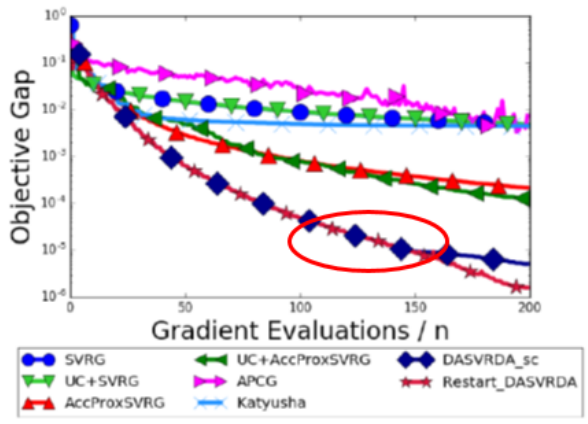
(e) rcv1, $(\lambda_1, \lambda_2) = (10^{-4}, 10^{-6})$



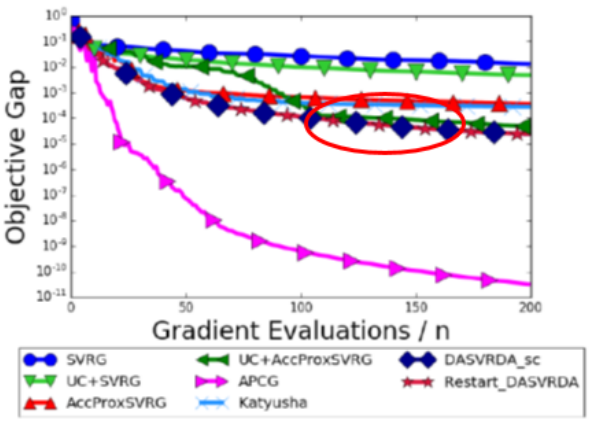
(f) rcv1, $(\lambda_1, \lambda_2) = (0, 10^{-6})$



(g) sido0, $(\lambda_1, \lambda_2) = (10^{-4}, 0)$



(h) sido0, $(\lambda_1, \lambda_2) = (10^{-4}, 10^{-6})$



(i) sido0, $(\lambda_1, \lambda_2) = (0, 10^{-6})$

ミニバッチサイズの最適性

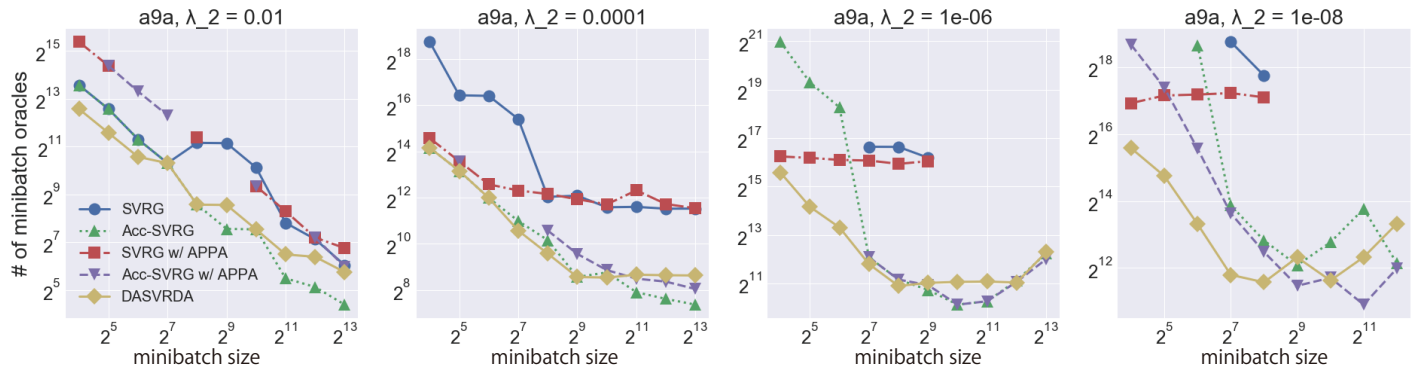
適切なミニバッチサイズを用いた DASVRDA は更新回数，全計算量ともにほぼ最適 (Nitanda, Murata, and Suzuki, 2019) (see also Nesterov (2004), Woodworth and Srebro (2016), Arjevani and Shamir (2016)).

全計算量 (勾配の計算回数) = ミニバッチサイズ × 更新回数

	全計算量	更新回数
GD	$n\sqrt{\kappa}\log(1/\epsilon)$	$\sqrt{\kappa}\log(1/\epsilon)$
DASVRDA (with b^*)	$O\left(\left(n+\sqrt{n\kappa}\right)\log(1/\epsilon)\right)$	$O\left(\sqrt{\kappa}\log(1/\epsilon)\right)$
Optimal	$\Omega(n+\sqrt{n\kappa}\log(1/\epsilon))$	$\Omega(\sqrt{\kappa}\log(1/\epsilon))$

(強凸の場合，最適ミニバッチサイズ $b^* = \sqrt{n} + \frac{n}{\sqrt{\kappa}\log(1/\epsilon)}$)

例えば $n \leq \kappa$ の時，ミニバッチサイズを $b \gg b^* = \Theta(\sqrt{n})$ としても無駄.
つまり，通常の勾配法のような大バッチサイズの手法は何にせよ得をしない.



非強凸の場合も同様.

バッチ型確率的最適化手法のまとめ

各種手法の性質

手法	SDCA	SVRG	SAG/SAGA
主/双対	双対	主	主
メモリ効率	✓	✓	△
加速 ($\mu > 0$)	✓	Katyusha/DASVRDA	Catalyst
その他制約	$\ell_i(\beta) = f_i(x_i^\top \beta)$	二重ループ	平滑な正則化

バッチ型確率的最適化手法のまとめ

各種手法の性質

手法	SDCA	SVRG	SAG/SAGA
主/双対	双対	主	主
メモリ効率	✓	✓	△
加速 ($\mu > 0$)	✓	Katyusha/DASVRDA	Catalyst
その他制約	$\ell_i(\beta) = f_i(x_i^\top \beta)$	二重ループ	平滑な正則化

収束レート (γ, μ の log 項は除く)

手法	$\mu > 0$	$\mu = 0$	加速法 ($\mu > 0$)
近接勾配法	$n\frac{\gamma}{\mu}\log(1/\epsilon)$	$n\sqrt{\gamma/\epsilon}$ (加速)	$n\sqrt{\frac{\gamma}{\mu}}\log(1/\epsilon)$
SDCA & SVRG	$(n+\frac{\gamma}{\mu})\log(1/\epsilon)$	$O(n\log(\frac{1}{\epsilon})+\sqrt{n\frac{\gamma}{\epsilon}})$	$(n+\sqrt{\frac{n\gamma}{\mu}})\log(1/\epsilon)$
SAG/SAGA	$(n+\frac{\gamma}{\mu})\log(1/\epsilon)$	$\gamma n/\epsilon$(※)	$(n+\sqrt{\frac{n\gamma}{\mu}})\log(1/\epsilon)$

※ $\mu = 0$ の時，Catalyst は $O((n+\sqrt{n\gamma/\epsilon})\log^2(1/\epsilon))$，Katyusha は AdaptReg と合わせて $O(n\log(1/\epsilon)+\sqrt{n\gamma/\epsilon})$ の収束レートを達成.

近接勾配法　　確率的最適化

$$n\sqrt{\frac{\gamma}{\mu}}\log(1/\epsilon) \quad \geq \quad (n+\sqrt{\frac{n\gamma}{\mu}})\log(1/\epsilon)$$

有用な文献

- Yurii Nesterov: *Introductory Lectures on Convex Optimization: A Basic Course*. Kluwer Academic Publishers, 2014.
- ★ Guanghui Lan: *First-order and Stochastic Optimization Methods for Machine Learning*. Springer Series in the Data Sciences, Springer, 2020.
- 金森 敬文，鈴木 大慈，竹内 一郎，佐藤 一誠:『機械学習のための連続最適化』．講談社，2016.
- 鈴木大慈:『確率的最適化』．講談社，2015.

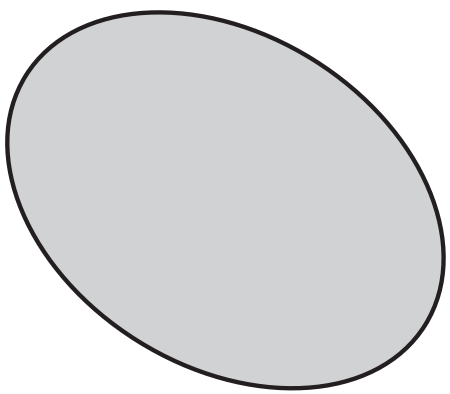
Outline

Convex set

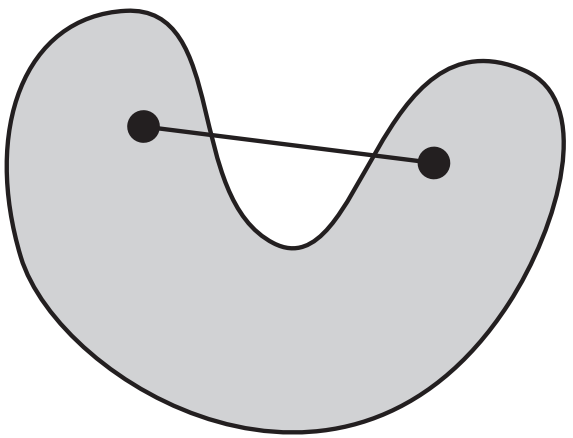
Definition (Convex set)

A convex set is a set that contains the segment connecting two points in the set:

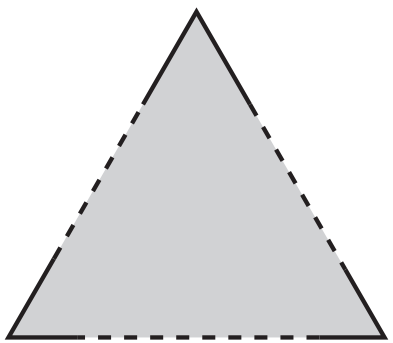
$$x_1, x_2 \in C \quad \Longrightarrow \quad \theta x_1 + (1 - \theta_2) x_2 \in C \quad (\theta \in [0, 1]).$$



凸集合



非凸集合



非凸集合

Epigraph and domain

Let $\bar{\mathbb{R}} := \mathbb{R} \cup \{\infty\}$.

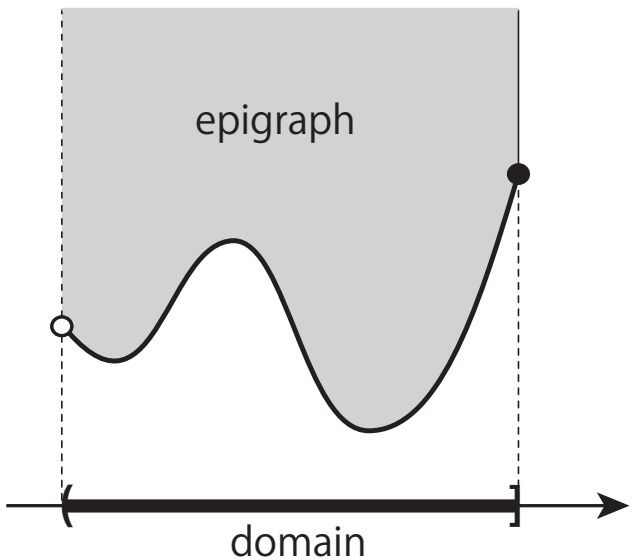
Definition (Epigraph and domain)

- The epigraph of a function $f : \mathbb{R}^p \to \bar{\mathbb{R}}$ is given by

$$\operatorname{epi}(f) := \{(x, \mu) \in \mathbb{R}^{p+1} : f(x) \leq \mu\}.$$

- The domain of a function $f : \mathbb{R}^p \to \bar{\mathbb{R}}$ is given by

$$\operatorname{dom}(f) := \{x \in \mathbb{R}^p : f(x) < \infty\}.$$



Convex function

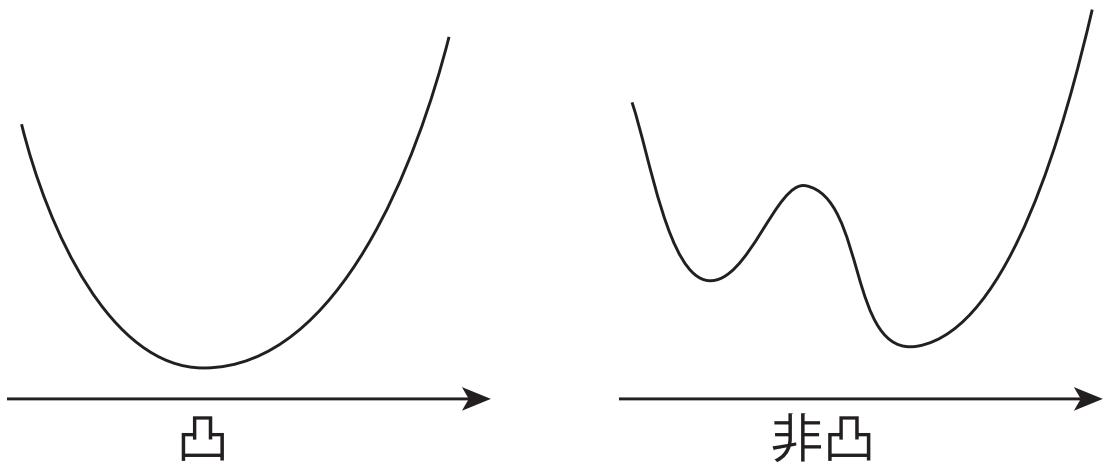
Let $\bar{\mathbb{R}} := \mathbb{R} \cup \{\infty\}$.

Definition (Convex function)

A function $f : \mathbb{R}^p \to \bar{\mathbb{R}}$ is a convex function if f satisfies

$$\theta f(x) + (1-\theta) f(y) \geq f(\theta x + (1-\theta) y) \quad (\forall x, y \in \mathbb{R}^p, \theta \in [0,1]),$$

where $\infty + \infty = \infty$, $\infty \leq \infty$.



- f is convex $\Leftrightarrow$ $\mathrm{epi}(f)$ is a convex set.

Proper and closed convex function

- If the domain of a function f is not empty ($\mathrm{dom}(f) \neq \emptyset$), f is called proper.
- If the epigraph of a convex function f is a closed set, then f is called closed.

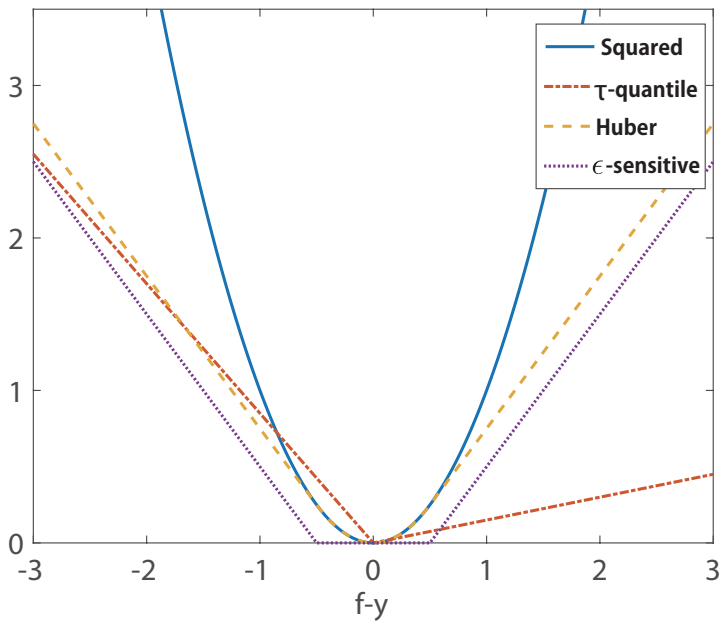
  (We are interested in only a proper closed function in this lecture.)

- Even if f is closed, it's domain is not necessarily closed (even for 1D).
- "f is closed" does not imply "f is continuous."
- Closed convex function is continuous on a segment in its domain.
- Closed function is "lower semicontinuity."

Convex loss functions (regression)

All well used loss functions are (closed) convex. The followings are convex w.r.t. u with a fixed label $y \in \mathbb{R}$.

- Squared loss: $\ell(y,u) = \frac{1}{2}(y-u)^2$.
- τ-quantile loss: $\ell(y,u) = (1-\tau)\max\{u-y,0\} + \tau\max\{y-u,0\}$. for some $\tau \in (0,1)$. Used for quantile regression.
- ϵ-sensitive loss: $\ell(y,u) = \max\{|y-u| - \epsilon, 0\}$ for some $\epsilon > 0$. Used for support vector regression.

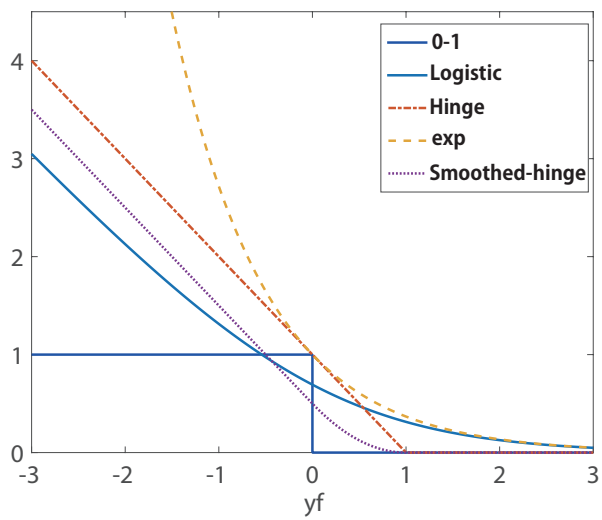


Convex surrogate loss (classification)

$y \in \{\pm 1\}$

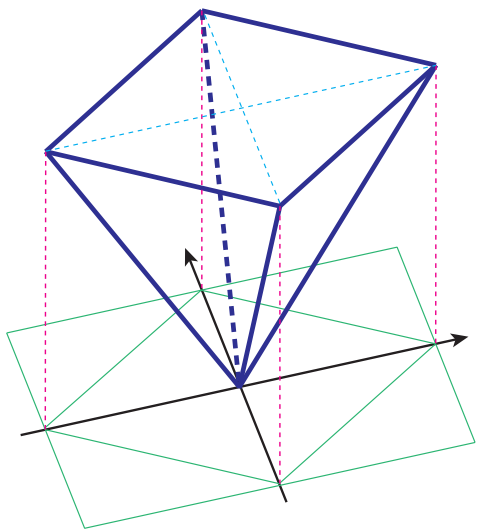
- Logistic loss: $\ell(y,u) = \log((1+\exp(-yu))/2)$.
- Hinge loss: $\ell(y,u) = \max\{1-yu, 0\}$.
- Exponential loss: $\ell(y,u) = \exp(-yu)$.
- Smoothed hinge loss:

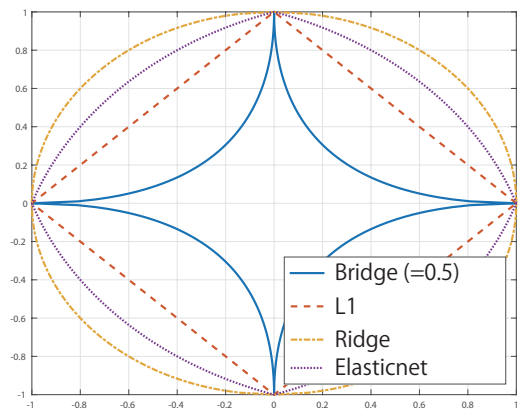
$$\ell(y,u) = \begin{cases} 0, & (yu \geq 1), \\ \frac{1}{2} - yu, & (yu < 0), \\ \frac{1}{2}(1-yu)^2, & (\text{otherwise}). \end{cases}$$



Convex regularization functions

- Ridge regularization: $R(x) = \|x\|_2^2 := \sum_{j=1}^p x_j^2$.
- L_1 regularization: $R(x) = \|x\|_1 := \sum_{j=1}^p |x_j|$.
- Trace norm regularization: $R(X) = \|X\|_{\rm tr} = \sum_{k=1}^{\min\{q,r\}} \sigma_j(X)$
  where $\sigma_j(X) \geq 0$ is the j-th singular value.





$\frac{1}{n}\sum_{i=1}^n (y_i - z_i^\top x)^2 + \lambda\|x\|_1$: Lasso
$\frac{1}{n}\sum_{i=1}^n \log(1+\exp(-y_i z_i^\top x)) + \lambda\|X\|_{\rm tr}$: Low rank matrix recovery

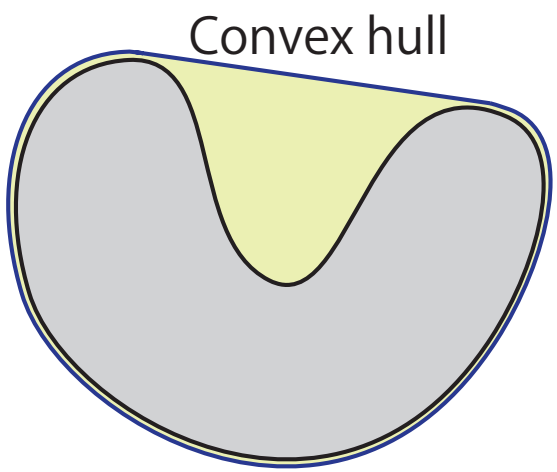
Other definitions of sets

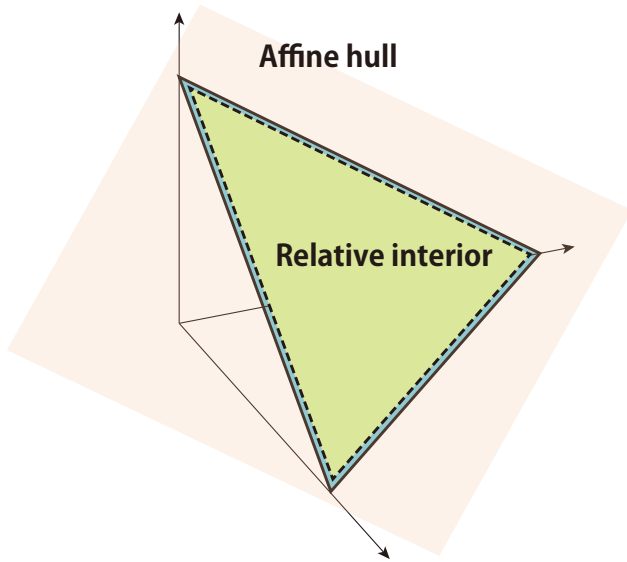
Convex hull: $\mathrm{conv}(C)$ is the smallest convex set that contains a set $C \subseteq \mathbb{R}^p$.

Affine set: A set A is an affine set if and only if $\forall x, y \in A$, the line that intersects x and y lies in A: $\lambda x + (1-\lambda)y \quad \forall \lambda \in \mathbb{R}$.

Affine hull: The smallest affine set that contains a set $C \subseteq \mathbb{R}^p$.

Relative interior: $\mathrm{ri}(C)$. Let A be the affine hull of a convex set $C \subseteq \mathbb{R}^p$. $\mathrm{ri}(C)$ is a set of internal points with respect to the relative topology induced by the affine hull A.





Continuity of a closed convex function

Theorem

For a (possibly non-convex) function $f : \mathbb{R}^p \to \bar{\mathbb{R}}$, the following three conditions are equivalent to each other.

1. *f is lower semi-continuous.*
2. *For any converging sequence $\{x_n\}_{n=1}^{\infty} \subseteq \mathbb{R}^p$ s.t. $x_\infty = \lim_n x_n$, $\liminf_n f(x_n) \geq f(x_\infty)$.*
3. *f is closed.*

Remark: Any convex function f is continuous in $\mathrm{ri}(\mathrm{dom}(f))$. The continuity could be broken on the boundary of the domain.

Outline

Subgradient

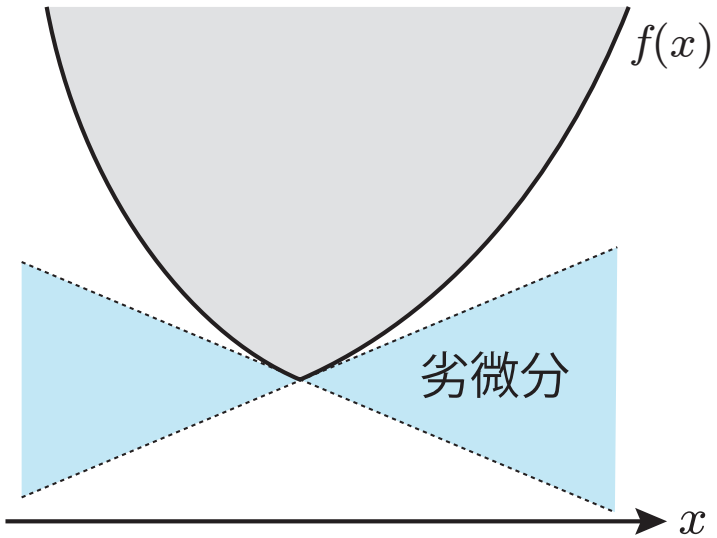
We want to deal with non-differentiable function such as L_1 regularization. To do so, we need to define something like gradient.

Definition (Subdifferential, subgradient)

For a proper convex function $f : \mathbb{R}^p \to \bar{\mathbb{R}}$, the **subdifferential** of f at $x \in \mathrm{dom}(f)$ is defined by

$$\partial f(x) := \{g \in \mathbb{R}^p \mid \langle x' - x, g \rangle + f(x) \leq f(x') \quad (\forall x' \in \mathbb{R}^p)\}.$$

An element of the subdifferential is called **subgradient**.



Properties of subgradient

- Subgradient does not necessarily exist ($\partial f(x)$ could be empty).
  $f(x) = x\log(x)$ $(x \geq 0)$ is proper convex but not subdifferentiable at $x = 0$.
- Subgradient always exists on $\mathrm{ri}(\mathrm{dom}(f))$.

Properties of subgradient

- Subgradient does not necessarily exist ($\partial f(x)$ could be empty).
  $f(x) = x\log(x)$ $(x \geq 0)$ is proper convex but not subdifferentiable at $x = 0$.
- Subgradient always exists on $\mathrm{ri}(\mathrm{dom}(f))$.
- If f is differentiable at x, its gradient is the unique element of subdiff.

$$\partial f(x) = \{\nabla f(x)\}.$$

Properties of subgradient

- Subgradient does not necessarily exist ($\partial f(x)$ could be empty).
  $f(x) = x\log(x)$ $(x \geq 0)$ is proper convex but not subdifferentiable at $x = 0$.
- Subgradient always exists on $\mathrm{ri}(\mathrm{dom}(f))$.
- If f is differentiable at x, its gradient is the unique element of subdiff.

$$\partial f(x) = \{\nabla f(x)\}.$$

- If $\mathrm{ri}(\mathrm{dom}(f)) \cap \mathrm{ri}(\mathrm{dom}(h)) \neq \emptyset$, then

$$\begin{aligned}\partial(f+h)(x) &= \partial f(x) + \partial h(x) \\ &= \{g + g' \mid g \in \partial f(x),\ g' \in \partial h(x)\} \\ &\quad (\forall x \in \mathrm{dom}(f) \cap \mathrm{dom}(h)).\end{aligned}$$

Properties of subgradient

- Subgradient does not necessarily exist ($\partial f(x)$ could be empty).
  $f(x) = x\log(x)$ $(x \geq 0)$ is proper convex but not subdifferentiable at $x = 0$.
- Subgradient always exists on $\mathrm{ri}(\mathrm{dom}(f))$.
- If f is differentiable at x, its gradient is the unique element of subdiff.

$$\partial f(x) = \{\nabla f(x)\}.$$

- If $\mathrm{ri}(\mathrm{dom}(f)) \cap \mathrm{ri}(\mathrm{dom}(h)) \neq \emptyset$, then

$$\begin{aligned}\partial(f+h)(x) &= \partial f(x) + \partial h(x) \\ &= \{g + g' \mid g \in \partial f(x),\ g' \in \partial h(x)\} \\ &\quad (\forall x \in \mathrm{dom}(f) \cap \mathrm{dom}(h)).\end{aligned}$$

- For all $g \in \partial f(x)$ and all $g' \in \partial f(x')$ $(x, x' \in \mathrm{dom}(f))$,

$$\langle g - g', x - x' \rangle \geq 0.$$

Legendre transform

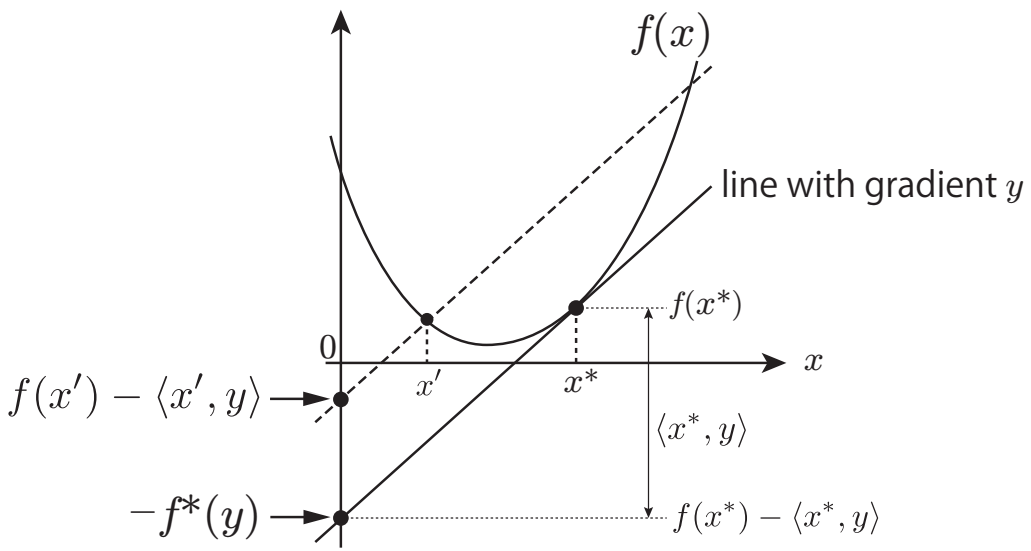
Defines the other representation on the **dual space** (the space of **gradients**).

Definition (Legendre transform)

Let f be a (possibly non-convex) function $f : \mathbb{R}^p \to \bar{\mathbb{R}}$ s.t. $\mathrm{dom}(f) \neq \emptyset$. Its **convex conjugate** is given by

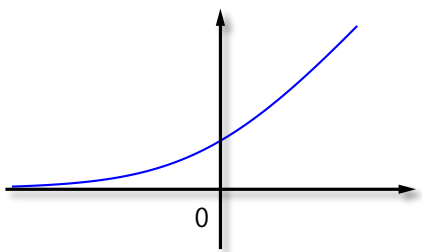
$$f^*(y) := \sup_{x \in \mathbb{R}^p} \{\langle x, y \rangle - f(x)\}.$$

The map from f to f^* is called Legendre transform.

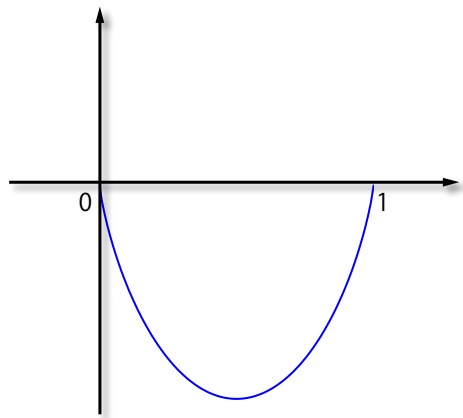


Examples

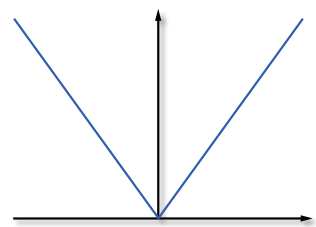
	$f(x)$	$f^*(y)$
Squared loss	$\frac{1}{2}x^2$	$\frac{1}{2}y^2$
Hinge loss	$\max\{1-x, 0\}$	$\begin{cases} y & (-1 \leq y \leq 0), \\ \infty & (\text{otherwise}). \end{cases}$
Logistic loss	$\log(1+\exp(-x))$	$\begin{cases} (-y)\log(-y) + (1+y)\log(1+y) & (-1 \leq y \leq 0), \\ \infty & (\text{otherwise}). \end{cases}$
L_1 regularization	$\|x\|_1$	$\begin{cases} 0 & (\max_j \lvert y_j \rvert \leq 1), \\ \infty & (\text{otherwise}). \end{cases}$
L_p regularization $(p > 1)$	$\sum_{j=1}^d \lvert x_j \rvert^p$	$\sum_{j=1}^d \frac{p-1}{p^{\frac{p}{p-1}}} \lvert y_j \rvert^{\frac{p}{p-1}}$



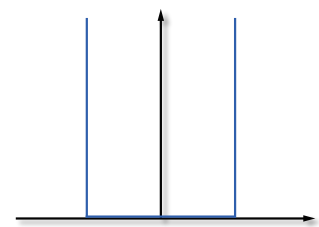
logistic



dual of logistic



L_1-norm



dual

Properties of Legendre transform

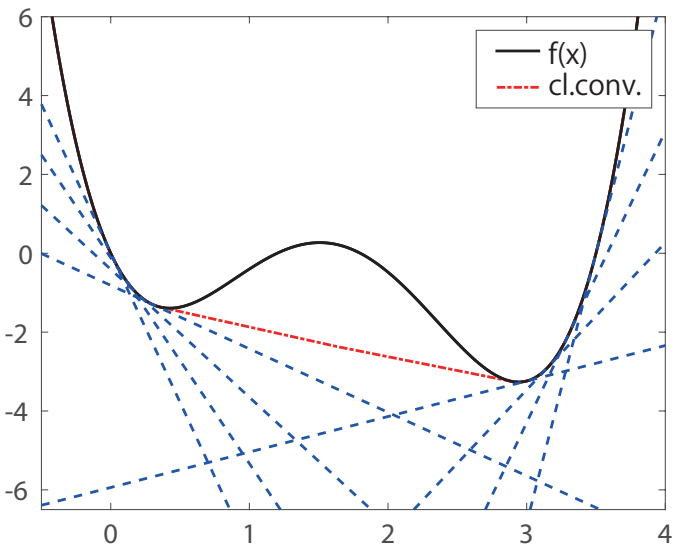
- f^* is a convex function even if f is not.
- f^{**} is the closure of the convex hull of f:

$$f^{**} = \mathrm{cl}(\mathrm{conv}(f)).$$

Corollary

Legendre transform is a bijection from the set of proper closed convex functions onto that defined on the dual space.

$$f \ \ (\textit{proper closed convex}) \quad \Leftrightarrow \quad f^* \ \ (\textit{proper closed convex})$$



Connection to subgradient

Lemma

$$y \in \partial f(x) \quad \Leftrightarrow \quad f(x) + f^*(y) = \langle x, y \rangle \quad \Leftrightarrow \quad x \in \partial f^*(y).$$

$$\because \quad y \in \partial f(x) \Rightarrow x = \underset{x' \in \mathbb{R}^p}{\operatorname{argmax}} \{\langle x', y \rangle - f(x')\}$$

(take the "derivative" of $\langle x', y \rangle - f(x')$)

$$\Rightarrow f^*(y) = \langle x, y \rangle - f(x).$$

Remark: By definition, we always have

$$f(x) + f^*(y) \geq \langle x, y \rangle.$$

$\rightarrow$ **Young-Fenchel's inequality**.

★ Fenchel's duality theorem

Theorem (Fenchel's duality theorem)

Let $f : \mathbb{R}^p \to \bar{\mathbb{R}}$, $g : \mathbb{R}^q \to \bar{\mathbb{R}}$ be proper closed convex, and $A \in \mathbb{R}^{q \times p}$. Suppose that either of condition (a) or (b) is satisfied, then it holds that

$$\inf_{x \in \mathbb{R}^p} \{f(x) + g(Ax)\} = \sup_{y \in \mathbb{R}^q} \{-f^*(A^\top y) - g^*(-y)\}.$$

(a) $\exists x \in \mathbb{R}^p$ s.t. $x \in \mathrm{ri}(\mathrm{dom}(f))$ and $Ax \in \mathrm{ri}(\mathrm{dom}(g))$.

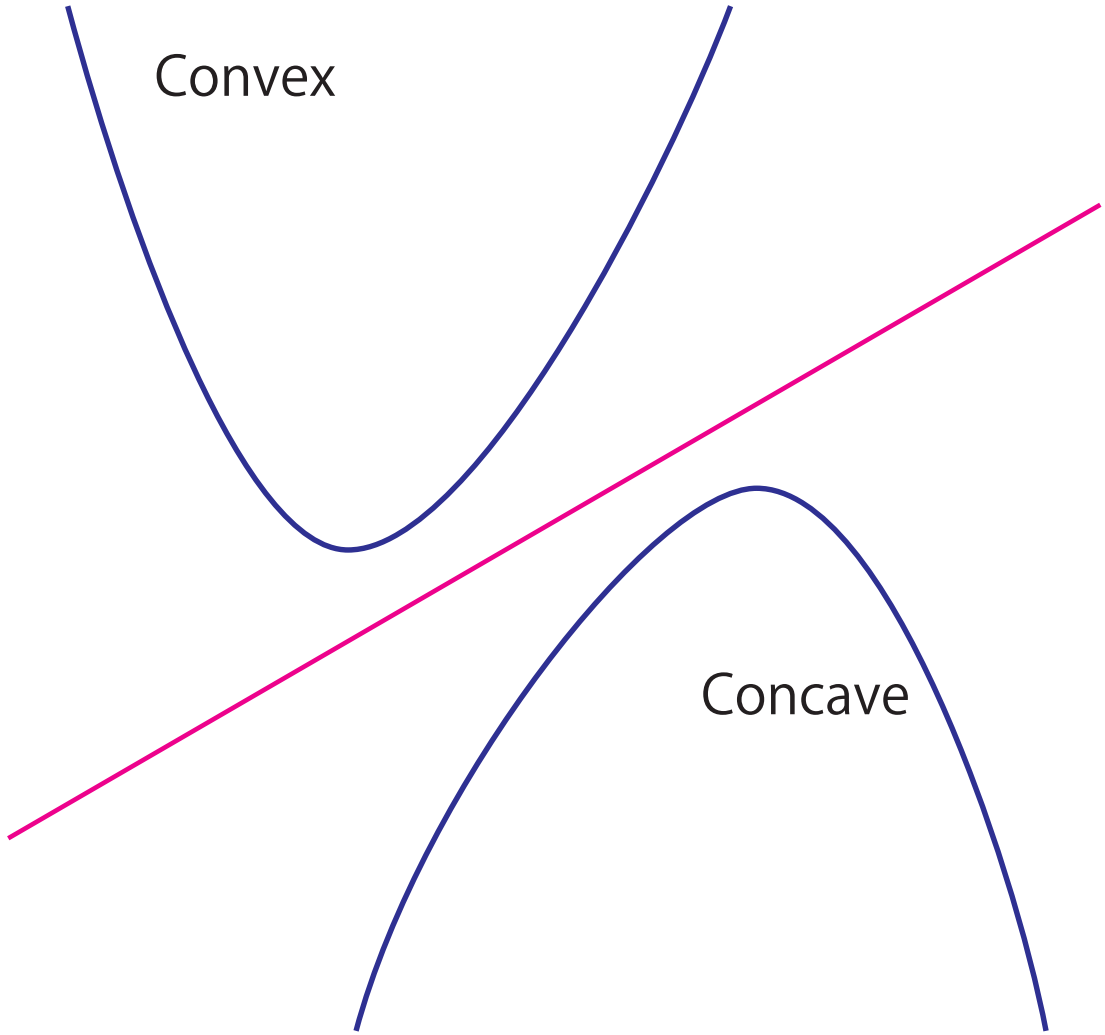
(b) $\exists y \in \mathbb{R}^q$ s.t. $A^\top y \in \mathrm{ri}(\mathrm{dom}(f^*))$ and $-y \in \mathrm{ri}(\mathrm{dom}(g^*))$.

If (a) is satisfied, there exists $y^* \in \mathbb{R}^q$ that attains sup of the RHS.
If (b) is satisfied, there exists $x^* \in \mathbb{R}^p$ that attains inf of the LHS.
Under (a) and (b), x^*, y^* are the optimal solutions of the each side iff

$$A^\top y^* \in \partial f(x^*), \quad Ax^* \in \partial g^*(-y^*).$$

$\to$ **Karush-Kuhn-Tucker condition**.

Equivalence to the separation theorem



Applying Fenchel's duality theorem to RERM

RERM (Regularized Empirical Risk Minimizatino):
Let $\ell_i(z_i^\top x) = \ell(y_i, z_i^\top x)$ where (z_i, y_i) is the input-output pair of the i-th observation.

(Primal) $$\inf_{x \in \mathbb{R}^p} \left\{ \underbrace{\sum_{i=1}^n \ell_i(z_i^\top x)}_{f(Zx)} + \psi(x) \right\}$$

[Fenchel's duality theorem]

$$\inf_{x \in \mathbb{R}^p} \{f(Zx) + \psi(x)\} = - \inf_{y \in \mathbb{R}^n} \{f^*(y) + \psi^*(-Z^\top y)\}$$

(Dual) $$\sup_{y \in \mathbb{R}^n} \left\{ \sum_{i=1}^n \ell_i^*(y_i) + \psi^*(-Z^\top y) \right\}$$

This fact will be used to derive dual coordinate descent alg.

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

K. Ahn. From proximal point method to nesterov's acceleration, 2020.

Z. Allen-Zhu. Katyusha: The first direct acceleration of stochastic gradient methods. arXiv preprint arXiv:1603.05953, 2016.

Z. Allen-Zhu. Katyusha: the first direct acceleration of stochastic gradient methods. In Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, pages 1200–1205. ACM, 2017.

Z. Allen-Zhu and E. Hazan. Optimal black-box reductions between optimization objectives. arXiv preprint arXiv:1603.05642, 2016.

Y. Arjevani and O. Shamir. Dimension-free iteration complexity of finite sum optimization problems. In Advances in Neural Information Processing Systems, pages 3540–3548, 2016.

P. Bartlett, O. Bousquet, and S. Mendelson. Local Rademacher complexities. The Annals of Statistics, 33:1487–1537, 2005.

A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM journal on imaging sciences, 2(1):183–202, 2009.

L. Bottou. Online algorithms and stochastic approximations. 1998. URL `http://leon.bottou.org/papers/bottou-98x`. revised, oct 2012.

L. Bottou. Large-scale machine learning with stochastic gradient descent. In Proceedings of COMPSTAT'2010, pages 177–186. Springer, 2010.

L. Bottou and Y. LeCun. Large scale online learning. In S. Thrun, L. Saul, and B. Schölkopf, editors, Advances in Neural Information Processing Systems 16. MIT Press, Cambridge, MA, 2004. URL http://leon.bottou.org/papers/bottou-lecun-2004.

X. Chen, Q. Lin, and J. Pena. Optimal regularized dual averaging methods for stochastic optimization. In F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, Advances in Neural Information Processing Systems 25, pages 395–403. Curran Associates, Inc., 2012.

S. Dasgupta and A. Gupta. An elementary proof of the johnson-lindenstrauss lemma. Technical Report 99–006, U.C. Berkeley, 1999.

A. Defazio, F. Bach, and S. Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, editors, Advances in Neural Information Processing Systems 27, pages 1646–1654. Curran Associates, Inc., 2014.

J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. Journal of Machine Learning Research, 12:2121–2159, 2011.

R. Frostig, R. Ge, S. Kakade, and A. Sidford. Un-regularizing: approximate proximal point and faster stochastic algorithms for empirical risk minimization. In International Conference on Machine Learning, pages 2540–2548, 2015.

S. Ghadimi and G. Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization I: A generic algorithmic framework. SIAM Journal on Optimization, 22(4):1469–1492, 2012.

S. Ghadimi and G. Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, II: shrinking procedures and optimal algorithms. SIAM Journal on Optimization, 23(4):2061–2089, 2013.

C. Hu, W. Pan, and J. T. Kwok. Accelerated gradient methods for stochastic optimization and online learning. In Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta, editors, Advances in Neural Information Processing Systems 22, pages 781–789. Curran Associates, Inc., 2009.

R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, editors, Advances in Neural Information Processing Systems 26, pages 315–323. Curran Associates, Inc., 2013.

W. B. Johnson, J. Lindenstrauss, and G. Schechtman. Extensions of lipschitz maps into banach spaces. Israel Journal of Mathematics, 54(2):129–138, 1986.

G. Lan. An optimal method for stochastic composite optimization. Mathematical Programming, 133(1-2):365–397, 2012.

N. Le Roux, M. Schmidt, and F. R. Bach. A stochastic gradient method with an exponential convergence rate for finite training sets. In F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, Advances in Neural Information Processing Systems 25, pages 2663–2671. Curran Associates, Inc., 2012.

H. Lin, J. Mairal, and Z. Harchaoui. A universal catalyst for first-order optimization. Technical report, 2015. arXiv:1506.02186.

Q. Lin, Z. Lu, and L. Xiao. An accelerated proximal coordinate gradient method. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, editors, Advances in Neural Information Processing Systems 27, pages 3059–3067. Curran Associates, Inc., 2014.

T. Murata and T. Suzuki. Doubly accelerated stochastic variance reduced dual averaging method for regularized empirical risk minimization. In Advances in Neural Information Processing Systems 30, pages 608–617. 2017.

A. Nemirovskii and D. Yudin. On cezari ' s convergence of the steepest descent method for approximating saddle points of convex-concave functions. Soviet Mathematics Doklady, 19(2):576–601, 1978.

A. Nemirovsky and D. Yudin. Problem complexity and method efficiency in optimization. John Wiley, New York, 1983.

Y. Nesterov. Introductory lectures on convex optimization: A basic course. Kluwer Academic Publishers, 2004.

A. Nitanda. Stochastic proximal gradient descent with acceleration techniques. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, editors, Advances in Neural Information Processing Systems 27, pages 1574–1582. Curran Associates, Inc., 2014.

A. Nitanda, T. Murata, and T. Suzuki. Sharp characterization of optimal minibatch size for stochastic finite sum convex optimization. In 2019 IEEE International Conference on Data Mining (ICDM), pages 488–497, 2019.

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

A. Rakhlin, O. Shamir, and K. Sridharan. Making gradient descent optimal for strongly convex stochastic optimization. In J. Langford and J. Pineau, editors, Proceedings of the 29th International Conference on Machine Learning, pages 449–456. Omnipress, 2012. ISBN 978-1-4503-1285-1.

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

F. Rosenblatt. The perceptron: A perceiving and recognizing automaton. Technical Report Technical Report 85-460-1, Project PARA, Cornell Aeronautical Lab., 1957.

M. Rudelson and S. Zhou. Reconstruction from anisotropic random measurements. IEEE Transactions of Information Theory, 39, 2013.

D. Ruppert. Efficient estimations from a slowly convergent robbins-monro process. Technical report, Cornell University Operations Research and Industrial Engineering, 1988.

M. Schmidt, N. Le Roux, and F. R. Bach. Minimizing finite sums with the stochastic average gradient, 2013. hal-00860051.

S. Shalev-Shwartz and T. Zhang. Stochastic dual coordinate ascent methods for regularized loss minimization. Journal of Machine Learning Research, 14: 567–599, 2013.

Y. Singer and J. C. Duchi. Efficient learning using forward-backward splitting. In Advances in Neural Information Processing Systems, pages 495–503, 2009.

I. Sutskever, J. Martens, G. Dahl, and G. Hinton. On the importance of initialization and momentum in deep learning. In Proceedings of the 30th international conference on machine learning (ICML-13), pages 1139–1147, 2013.

B. E. Woodworth and N. Srebro. Tight complexity bounds for optimizing composite objectives. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, Advances in Neural Information Processing Systems 29, pages 3639–3647. Curran Associates, Inc., 2016. URL http://papers.nips.cc/paper/6058-tight-complexity-bounds-for-optimizing-composite-objectives.pdf.

L. Xiao. Dual averaging methods for regularized stochastic learning and online optimization. In Advances in Neural Information Processing Systems 23. 2009.

L. Xiao. Dual averaging methods for regularized stochastic learning and online optimization. Journal of Machine Learning Research, 11:2543–2596, 2010.

L. Xiao and T. Zhang. A proximal stochastic gradient method with progressive variance reduction. SIAM Journal on Optimization, 24:2057–2075, 2014.

Y. Zhang and X. Lin. Stochastic primal-dual coordinate method for regularized empirical risk minimization. In F. Bach and D. Blei, editors, Proceedings of the 32nd International Conference on Machine Learning, volume 37 of Proceedings of Machine Learning Research, pages 353–361, Lille, France, 07–09 Jul 2015. PMLR. URL http://proceedings.mlr.press/v37/zhanga15.html.