スパース正則化およびマルチカーネル学習のための最適化アルゴリズムとCV・PRへの応用

冨岡 亮太[1], 鈴木 大慈[1], 杉山 将[2]

[1] 東京大学
[2] 東京工業大学

2009-08-31 @ PRMU/CVIM 仙台

http://www.ibis.t.u-tokyo.ac.jp/ryotat/prmu09/

Outline

Outline

Outline

Lasso 回帰 (1/3)

- **入出力の組み** $(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)$**.** $\boldsymbol{x}_i \in \mathbb{R}^n$**.**
- **仮定：**

$$y_i = \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \sigma^2).$$

- **経験誤差：**

$$L(\boldsymbol{w}) = \frac{1}{2} \sum_{i=1}^{m} (y_i - \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle)^2 = \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|^2.$$

- **動機 1：現実には多くの場合** $m < n$**.**
- **動機 2：なるべく少ない数の変数で説明したい！**
  （$\boldsymbol{w}$ **の非ゼロ要素の数** $\|\boldsymbol{w}\|_0$ **が少なければ少ないほどよい）**
- 問題 1:

$$\text{minimize} \quad \|\boldsymbol{w}\|_0, \quad \text{subject to } L(\boldsymbol{w}) \leq C.$$

- 問題 2:

$$\text{minimize} \quad L(\boldsymbol{w}), \quad \text{subject to } \|\boldsymbol{w}\|_0 \leq C'.$$

どちらも NP 困難!!

Lasso 回帰 (1/3)

- **入出力の組み** $(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)$**.** $\boldsymbol{x}_i \in \mathbb{R}^n$**.**
- **仮定：**

$$y_i = \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \sigma^2).$$

- **経験誤差：**

$$L(\boldsymbol{w}) = \frac{1}{2} \sum_{i=1}^{m} (y_i - \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle)^2 = \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|^2.$$

- **動機 1：現実には多くの場合** $m < n$**.**
- **動機 2：なるべく少ない数の変数で説明したい！**
  （$\boldsymbol{w}$ **の非ゼロ要素の数** $\|\boldsymbol{w}\|_0$ **が少なければ少ないほどよい）**
- **問題** 1:

$$\text{minimize} \quad \|\boldsymbol{w}\|_0, \quad \text{subject to } L(\boldsymbol{w}) \leq C.$$

- **問題** 2:

$$\text{minimize} \quad L(\boldsymbol{w}), \quad \text{subject to } \|\boldsymbol{w}\|_0 \leq C'.$$

どちらも NP 困難!!

Lasso 回帰 (1/3)

- **入出力の組み** $(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)$**.** $\boldsymbol{x}_i \in \mathbb{R}^n$**.**
- **仮定：**

$$y_i = \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \sigma^2).$$

- **経験誤差：**

$$L(\boldsymbol{w}) = \frac{1}{2} \sum_{i=1}^{m} (y_i - \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle)^2 = \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|^2.$$

- **動機 1：現実には多くの場合** $m < n$**.**
- **動機 2：なるべく少ない数の変数で説明したい！**
  （$\boldsymbol{w}$ **の非ゼロ要素の数** $\|\boldsymbol{w}\|_0$ **が少なければ少ないほどよい）**
- **問題** 1:

$$\text{minimize} \quad \|\boldsymbol{w}\|_0, \quad \text{subject to } L(\boldsymbol{w}) \leq C.$$

- **問題** 2:

$$\text{minimize} \quad L(\boldsymbol{w}), \quad \text{subject to } \|\boldsymbol{w}\|_0 \leq C'.$$

どちらも NP 困難!!

Lasso 回帰 (1/3)

- **入出力の組み** $(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)$**.** $\boldsymbol{x}_i \in \mathbb{R}^n$**.**
- **仮定：**

$$y_i = \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \sigma^2).$$

- **経験誤差：**

$$L(\boldsymbol{w}) = \frac{1}{2} \sum_{i=1}^{m} (y_i - \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle)^2 = \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|^2.$$

- **動機 1：現実には多くの場合** $m < n$**.**
- **動機 2：なるべく少ない数の変数で説明したい！**
  （$\boldsymbol{w}$ **の非ゼロ要素の数** $\|\boldsymbol{w}\|_0$ **が少なければ少ないほどよい）**
- **問題** 1:

$$\text{minimize} \quad \|\boldsymbol{w}\|_0, \quad \text{subject to } L(\boldsymbol{w}) \leq C.$$

- **問題** 2:

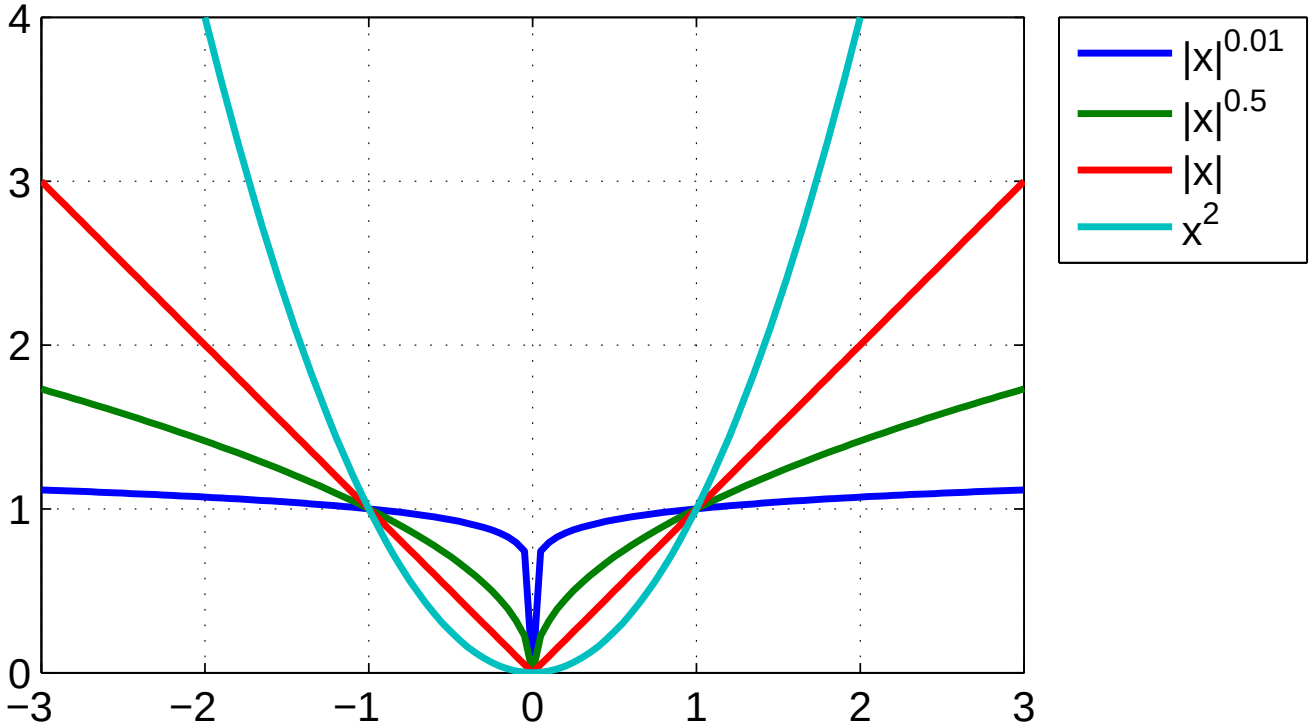
$$\text{minimize} \quad L(\boldsymbol{w}), \quad \text{subject to } \|\boldsymbol{w}\|_0 \leq C'.$$

どちらも NP 困難!!

Lasso 回帰 (2/3)

- *p*-ノルムの *p* 乗（のようなもの）

$$\|\boldsymbol{w}\|_p^p = \sum_{j=1}^{n} |w_j|^p : \begin{cases} p \geq 1 \text{ ならば凸} \\ p < 1 \text{ ならば非凸} \end{cases}$$

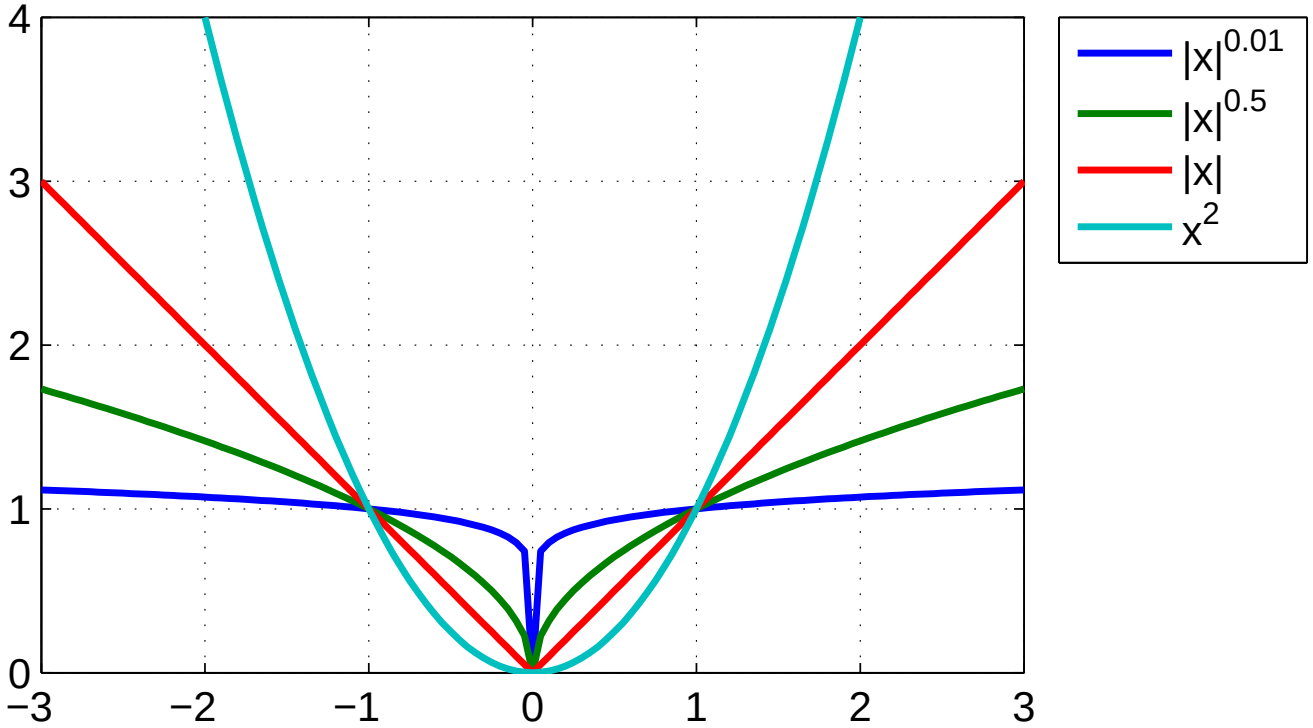


$\|w\|_1$ の正則化は凸の中ではもっとも $\|w\|_0$ の正則化に近い！

Lasso 回帰 (2/3)

- p-ノルムの p 乗（のようなもの）

$$\|\boldsymbol{w}\|_p^p = \sum_{j=1}^{n} |w_j|^p : \begin{cases} p \geq 1 \text{ ならば凸} \\ p < 1 \text{ ならば非凸} \end{cases}$$



$\|w\|_1$ の正則化は凸の中ではもっとも $\|w\|_0$ の正則化に近い！

Lasso 回帰 (3/3)

- **問題 1:**

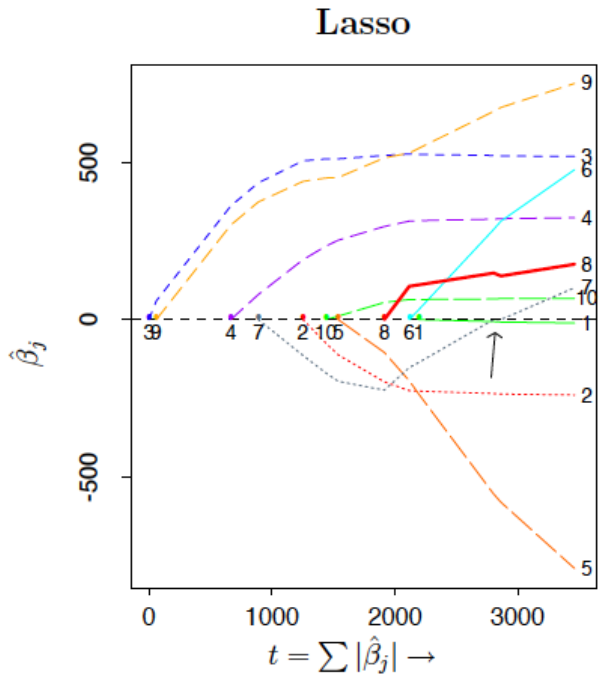
  $$\text{minimize} \quad \|\boldsymbol{w}\|_1, \quad \text{subject to } L(\boldsymbol{w}) \leq C.$$

- **問題 2:**

  $$\text{minimize} \quad L(\boldsymbol{w}), \quad \text{subject to } \|\boldsymbol{w}\|_1 \leq C'.$$

- **問題 3:**

  $$\text{minimize} \quad L(\boldsymbol{w}) + \lambda \|\boldsymbol{w}\|_1.$$



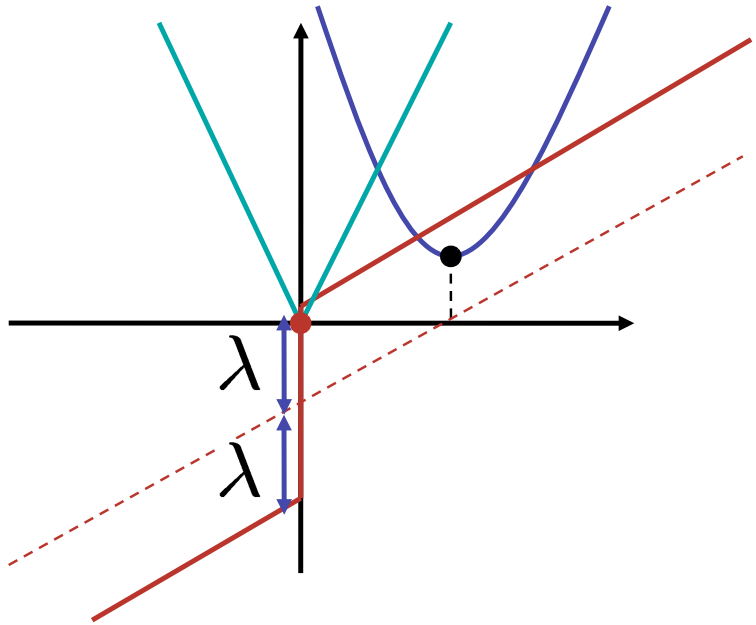
[From Efron et al. (2003)]

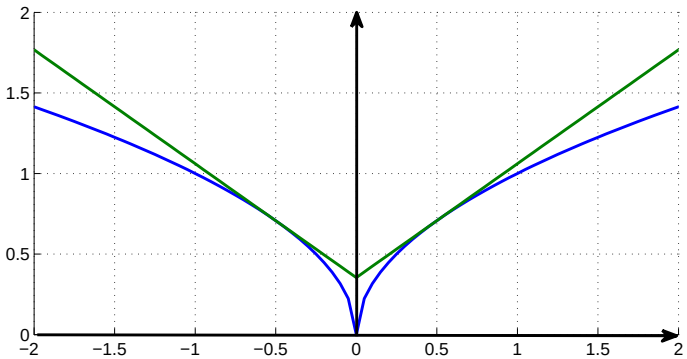
注意：

- **上の 3 つの問題はいずれも等価．**
- **正則化項やロス項に単調な非線形変換をしても等価．**
- **この発表では問題 3 を扱う．**

なぜ ℓ_1-正則化か？

- 凸の中で最も $\|\cdot\|_0$ に近い．
- 原点で微分不可能（有限の λ でゼロに打ち切ることができる．）
- 凸でない正則化
  $\longrightarrow$ 繰り返し（重み付き）ℓ_1-正則化問題を解けばよい．
- ベイズ周辺化尤度最大化
  $\longrightarrow$（特殊な場合に）繰り返し（重み付き）ℓ_1-正則化問題を解けばよい．
  (Wipf&Nagarajan, 08)



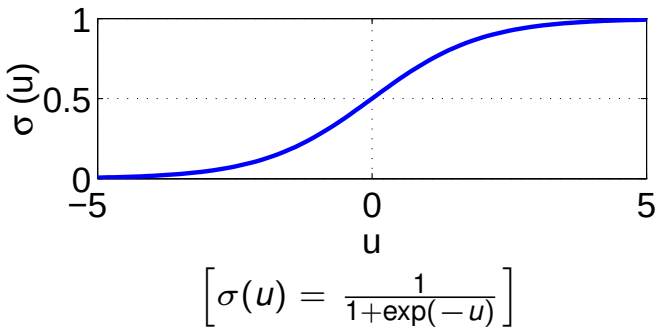


一般化

- **損失項を一般化 $\cdots$ 例：ℓ_1-ロジスティック回帰**

$$\underset{\boldsymbol{w}\in\mathbb{R}^n}{\text{minimize}} \quad \sum_{i=1}^{m} -\log P(y_i|\boldsymbol{x}_i;\boldsymbol{w}) + \lambda\|\boldsymbol{w}\|_1$$

ただし， $P(y|\boldsymbol{x};\boldsymbol{w}) = \sigma\left(y\left\langle \boldsymbol{w},\boldsymbol{x}\right\rangle\right)$ $(y\in\{-1,+1\})$



$\left[\sigma(u) = \frac{1}{1+\exp(-u)}\right]$

- **正則化項を一般化 $\cdots$ 例：グループラッソー** (Yuan&Lin,06)

$$\underset{\boldsymbol{w}\in\mathbb{R}^n}{\text{minimize}} \quad L(\boldsymbol{w}) + \lambda\sum_{\mathfrak{g}\in\mathfrak{G}}\|\boldsymbol{w}_{\mathfrak{g}}\|_2$$

ただし， $\mathfrak{G}$ **は** $\{1,\ldots,n\}$ **の適当な分割で，** $\boldsymbol{w} = \begin{pmatrix} (\boldsymbol{w}_{\mathfrak{g}_1}) \\ (\boldsymbol{w}_{\mathfrak{g}_2}) \\ \vdots \\ (\boldsymbol{w}_{\mathfrak{g}_q}) \end{pmatrix}$**，** $q = |\mathfrak{G}|$**.**

さらに一般化：カーネルを導入

マルチカーネル学習 (Multiple Kernel Learning, MKL): (Lanckriet, Bach, et al., 04)

$\mathcal{H}_1, \mathcal{H}_2, \ldots, \mathcal{H}_n$ を RKHS，$K_1, K_2, \ldots, K_n$ をそれに付随するカーネル関数とする．$f = \underbrace{f_1}_{\in \mathcal{H}_1} + \underbrace{f_2}_{\in \mathcal{H}_2} + \cdots + \underbrace{f_n}_{\in \mathcal{H}_n}$ で予測することで性能を上げよう！

$$\underset{f_j \in \mathcal{H}_j, b \in \mathbb{R}}{\text{minimize}} \quad L(f_1 + f_2 + \cdots + f_n + b) + \lambda \sum_{j=1}^{n} \|f_j\|_{\mathcal{H}_j}$$

$$\Downarrow \quad \text{（表現定理）}$$

$$\underset{\boldsymbol{\alpha}_j \in \mathbb{R}^m, b \in \mathbb{R}}{\text{minimize}} \quad f_\ell\left(\sum_{j=1}^{n} \boldsymbol{K}_j \boldsymbol{\alpha}_j + b\mathbf{1}\right) + \lambda \sum_{j=1}^{n} \|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}$$

ただし，$\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j} = \sqrt{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}$.

⋯ カーネル行列で重み付けされたグループラッソー．

さらに一般化：カーネルを導入

マルチカーネル学習 (Multiple Kernel Learning, MKL): (Lanckriet, Bach, et al., 04)

$\mathcal{H}_1, \mathcal{H}_2, \ldots, \mathcal{H}_n$ を RKHS，$K_1, K_2, \ldots, K_n$ をそれに付随するカーネル関数とする．$f = \underbrace{f_1}_{\in \mathcal{H}_1} + \underbrace{f_2}_{\in \mathcal{H}_2} + \cdots + \underbrace{f_n}_{\in \mathcal{H}_n}$ で予測することで性能を上げよう！

$$\underset{f_j \in \mathcal{H}_j, b \in \mathbb{R}}{\text{minimize}} \quad L(f_1 + f_2 + \cdots + f_n + b) + \lambda \sum_{j=1}^{n} \|f_j\|_{\mathcal{H}_j}$$

$\Downarrow$ （表現定理）

$$\underset{\boldsymbol{\alpha}_j \in \mathbb{R}^m, b \in \mathbb{R}}{\text{minimize}} \quad f_\ell\left(\sum_{j=1}^{n} \boldsymbol{K}_j \boldsymbol{\alpha}_j + b\mathbf{1}\right) + \lambda \sum_{j=1}^{n} \|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}$$

ただし，$\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j} = \sqrt{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}$.

⋯ カーネル行列で重み付けされたグループラッソー．

絞り込み

損失項は（多くの場合）損失関数 f_ℓ とデザイン行列 $\boldsymbol{A}$ に分解可能．

- **2 乗損失**

$$f_\ell^{\mathrm{Q}}(\boldsymbol{z}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{z}\|^2, \qquad \boldsymbol{A} = \begin{pmatrix} \boldsymbol{x}_1{}^\top \\ \vdots \\ \boldsymbol{x}_m{}^\top \end{pmatrix}$$

$$f_\ell^{\mathrm{Q}}(\boldsymbol{A}\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|^2$$

- **ロジスティック損失**

$$f_\ell^{\mathrm{L}}(\boldsymbol{z}) = \sum_{i=1}^{m} \log(1 + \exp(-y_i z_i)), \qquad \boldsymbol{A} = \begin{pmatrix} \boldsymbol{x}_1{}^\top \\ \vdots \\ \boldsymbol{x}_m{}^\top \end{pmatrix}$$

$$f_\ell^{\mathrm{L}}(\boldsymbol{A}\boldsymbol{w}) = \sum_{i=1}^{m} -\log \sigma(y_i \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle)$$

Outline

問題設定

以下の最適化問題を効率良く解くためのアルゴリズムが求められている．

$$\underset{\boldsymbol{w}\in\mathbb{R}^n}{\text{minimize}} \quad f_\ell(\boldsymbol{A}\boldsymbol{w}) + \phi_\lambda(\boldsymbol{w}).$$

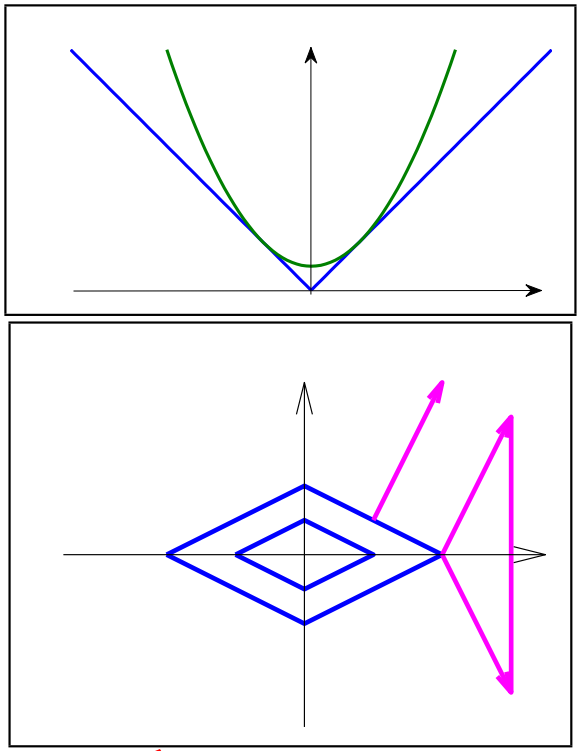
ただし，

- $\boldsymbol{A} \in \mathbb{R}^{m\times n}$（$m$: サンプル数，$n$: 未知変数の数）．
- f_ℓ は凸で 2 回微分可能．
- $\phi_\lambda(\boldsymbol{w})$ は例えば，$\phi_\lambda(\boldsymbol{w}) = \lambda\|\boldsymbol{w}\|_1$ など，凸だが，微分不可能であってもよい．また，$\eta\phi_\lambda = \phi_{\eta\lambda}$ を仮定．
- 特定の f_ℓ に依存したアルゴリズム（LARS など）ではもの足りない．
- No Free Lunch – 観測の数が変数の数より少ない場合 ($m \ll n$) や $\boldsymbol{A}$ のコンディションが悪い場合が応用上重要．

どこが難しいか？

今までの見方: $\phi_\lambda(\boldsymbol{w})$ **の微分不可能性が原因．**

- **正則化項を微分可能な関数で上から押さえる．**
  - FOCUSS
    (Rao & Kreutz-Delgado, 99)
  - Majorization-Minimization
    (Figueiredo et al., 07)
- **微分不可能性を陽に考慮する．**
  - Sub-gradient L-BFGS (Andrew & Gao, 07; Yu et al., 08)



我々の見方: ***A*** **が変数の間にからみを導入するのが原因．**

どこが難しいか？

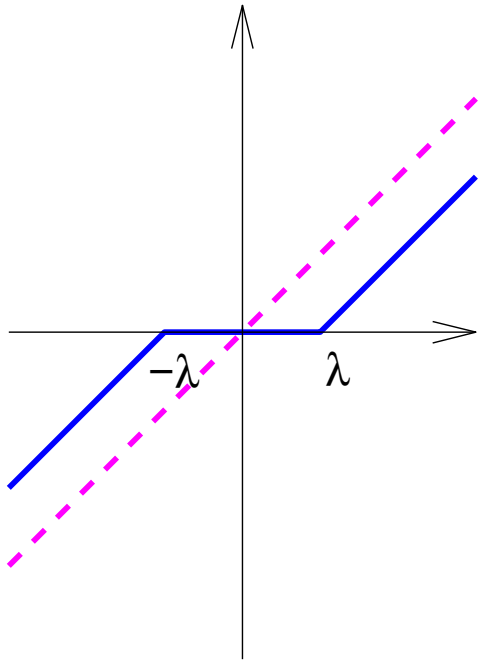
我々の見方:　A が変数の間にからみを導入するのが原因．

$A = I_n$ （単位行列の場合）

$$\min_{\boldsymbol{w}\in\mathbb{R}^n}\left(\frac{1}{2}\|\boldsymbol{y}-\boldsymbol{w}\|_2^2+\lambda\|\boldsymbol{w}\|_1\right)=\sum_{j=1}^{n}\min_{w_j\in\mathbb{R}}\left(\frac{1}{2}(y_j-w_j)^2+\lambda|w_j|\right).$$

$$\Rightarrow \quad w_j^* = \mathrm{ST}_\lambda(y_j)$$
$$= \begin{cases} y_j-\lambda & (\lambda \le y_j), \\ 0 & (-\lambda \le y_j \le \lambda), \\ y_j+\lambda & (y_j \le -\lambda). \end{cases}$$

解析的に解ける！



本発表では ϕ_λ として，上の最小化が解析的に求められるもののみを扱う．

先行研究

Iterative Shrinkage/Thresholding (IST) 法 (Figueiredo&Nowak, 03; Daubechies et al., 04,...):

アルゴリズム

1. 適当に初期解 $\boldsymbol{w}^1$ を決める．
2. 停止条件が満たされるまで反復：

$$\boldsymbol{w}^{t+1} \leftarrow \underset{\boldsymbol{w}\in\mathbb{R}^n}{\operatorname{argmin}} \left(Q_{\eta_t}(\boldsymbol{w}; \boldsymbol{w}^t) + \phi_\lambda(\boldsymbol{w})\right)$$

ただし，

$$Q_\eta(\boldsymbol{w}; \boldsymbol{w}^t) = \underbrace{L(\boldsymbol{w}^t) + \nabla L^\top(\boldsymbol{w}^t)(\boldsymbol{w} - \boldsymbol{w}^t)}_{\text{(1) 損失項を 1 次近似}} + \frac{1}{2\eta} \underbrace{\|\boldsymbol{w} - \boldsymbol{w}^t\|_2^2}_{\text{(2) 前の反復からの距離}^2 \text{にペナルティ}} .$$

注意：$Q_\eta(\boldsymbol{w}; \boldsymbol{w}^t)$ の最小化は普通の勾配ステップ（サイズ η）を与える．

先行研究：IST 法

$$\operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(Q_{\eta_t}(\boldsymbol{w};\boldsymbol{w}^t) + \phi_\lambda(\boldsymbol{w})\right)$$

$$= \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(\text{const.} + \nabla L^\top(\boldsymbol{w}^t)(\boldsymbol{w}-\boldsymbol{w}^t) + \frac{1}{2\eta_t}\|\boldsymbol{w}-\boldsymbol{w}^t\|_2^2 + \phi_\lambda(\boldsymbol{w})\right)$$

$$= \underbrace{\operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(\frac{1}{2\eta_t}\|\boldsymbol{w}-\tilde{\boldsymbol{w}}^t\|_2^2 + \phi_\lambda(\boldsymbol{w})\right)}_{\text{ここの最小化は解析的にできると仮定}} =: \mathrm{ST}_{\eta_t\lambda}(\tilde{\boldsymbol{w}}^t)$$

ただし，$\tilde{\boldsymbol{w}}^t = \boldsymbol{w}^t - \eta_t\nabla L(\boldsymbol{w}^t)$（勾配ステップ先）．
結局

$$\boldsymbol{w}^{t+1} \leftarrow \underbrace{\mathrm{ST}_{\eta_t\lambda}}_{\text{縮小}} \underbrace{\left(\boldsymbol{w}^t - \eta_t\nabla L(\boldsymbol{w}^t)\right)}_{\text{勾配ステップ}}$$

- 長所：簡単．
- 短所：A の悪スケーリングに弱い．

先行研究：IST 法

$$\begin{aligned}
&\operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(Q_{\eta_t}(\boldsymbol{w};\boldsymbol{w}^t)+\phi_\lambda(\boldsymbol{w})\right)\\
&=\operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(\text{const.}+\nabla L^\top(\boldsymbol{w}^t)(\boldsymbol{w}-\boldsymbol{w}^t)+\frac{1}{2\eta_t}\|\boldsymbol{w}-\boldsymbol{w}^t\|_2^2+\phi_\lambda(\boldsymbol{w})\right)\\
&=\underbrace{\operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(\frac{1}{2\eta_t}\|\boldsymbol{w}-\tilde{\boldsymbol{w}}^t\|_2^2+\phi_\lambda(\boldsymbol{w})\right)}_{\text{ここの最小化は解析的にできると仮定}} =: \mathrm{ST}_{\eta_t\lambda}(\tilde{\boldsymbol{w}}^t)
\end{aligned}$$

ただし，$\tilde{\boldsymbol{w}}^t = \boldsymbol{w}^t - \eta_t\nabla L(\boldsymbol{w}^t)$（勾配ステップ先）．

結局

$$\boldsymbol{w}^{t+1} \leftarrow \underbrace{\mathrm{ST}_{\eta_t\lambda}}_{\text{縮小}}\underbrace{\left(\boldsymbol{w}^t-\eta_t\nabla L(\boldsymbol{w}^t)\right)}_{\text{勾配ステップ}}$$

- 長所：簡単．
- 短所：A の悪スケーリングに弱い．

先行研究：IST 法

$$\operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(Q_{\eta_t}(\boldsymbol{w};\boldsymbol{w}^t)+\phi_\lambda(\boldsymbol{w})\right)$$

$$= \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(\text{const.} + \nabla L^\top(\boldsymbol{w}^t)(\boldsymbol{w}-\boldsymbol{w}^t) + \frac{1}{2\eta_t}\|\boldsymbol{w}-\boldsymbol{w}^t\|_2^2 + \phi_\lambda(\boldsymbol{w})\right)$$

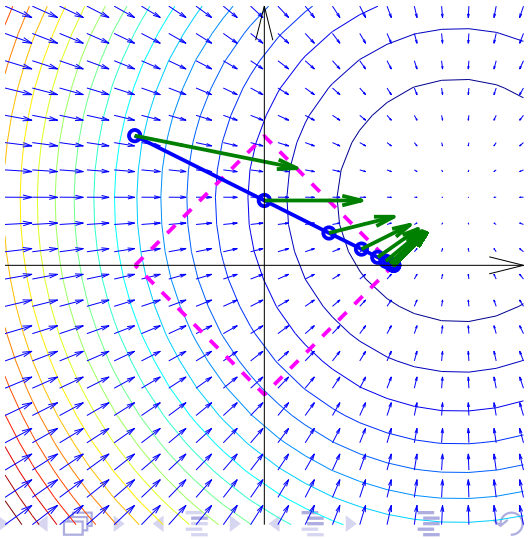
$$= \underbrace{\operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(\frac{1}{2\eta_t}\|\boldsymbol{w}-\tilde{\boldsymbol{w}}^t\|_2^2 + \phi_\lambda(\boldsymbol{w})\right)}_{\text{ここの最小化は解析的にできると仮定}} =: \mathrm{ST}_{\eta_t\lambda}(\tilde{\boldsymbol{w}}^t)$$

ただし，$\tilde{\boldsymbol{w}}^t = \boldsymbol{w}^t - \eta_t\nabla L(\boldsymbol{w}^t)$（勾配ステップ先）．

結局

$$\boldsymbol{w}^{t+1} \leftarrow \underbrace{\mathrm{ST}_{\eta_t\lambda}}_{\text{縮小}} \underbrace{\left(\boldsymbol{w}^t - \eta_t\nabla L(\boldsymbol{w}^t)\right)}_{\text{勾配ステップ}}$$

- **長所：簡単．**
- **短所：A の悪スケーリングに弱い．**



先行研究：IST 法

$$\operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(Q_{\eta_t}(\boldsymbol{w};\boldsymbol{w}^t)+\phi_\lambda(\boldsymbol{w})\right)$$

$$= \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(\text{const.} + \nabla L^\top(\boldsymbol{w}^t)(\boldsymbol{w}-\boldsymbol{w}^t) + \frac{1}{2\eta_t}\|\boldsymbol{w}-\boldsymbol{w}^t\|_2^2 + \phi_\lambda(\boldsymbol{w})\right)$$

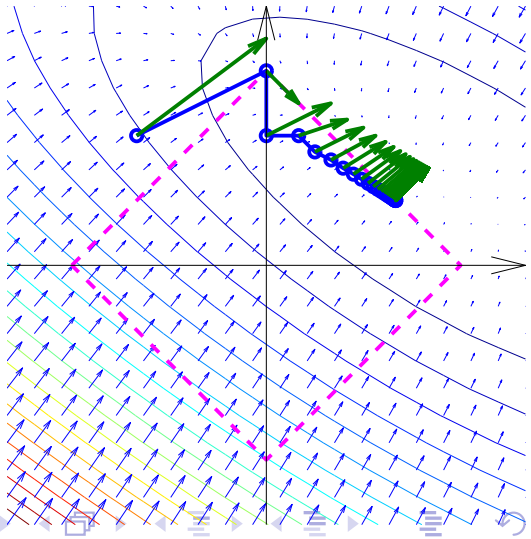
$$= \underbrace{\operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(\frac{1}{2\eta_t}\|\boldsymbol{w}-\tilde{\boldsymbol{w}}^t\|_2^2 + \phi_\lambda(\boldsymbol{w})\right)}_{\text{ここの最小化は解析的にできると仮定}} =: \mathrm{ST}_{\eta_t\lambda}(\tilde{\boldsymbol{w}}^t)$$

ただし，$\tilde{\boldsymbol{w}}^t = \boldsymbol{w}^t - \eta_t\nabla L(\boldsymbol{w}^t)$（勾配ステップ先）．

結局

$$\boldsymbol{w}^{t+1} \leftarrow \underbrace{\mathrm{ST}_{\eta_t\lambda}}_{\text{縮小}} \underbrace{\left(\boldsymbol{w}^t - \eta_t\nabla L(\boldsymbol{w}^t)\right)}_{\text{勾配ステップ}}$$

- **長所：簡単．**
- **短所：A の悪スケーリングに弱い．**



ここまでのまとめ

最適化問題：

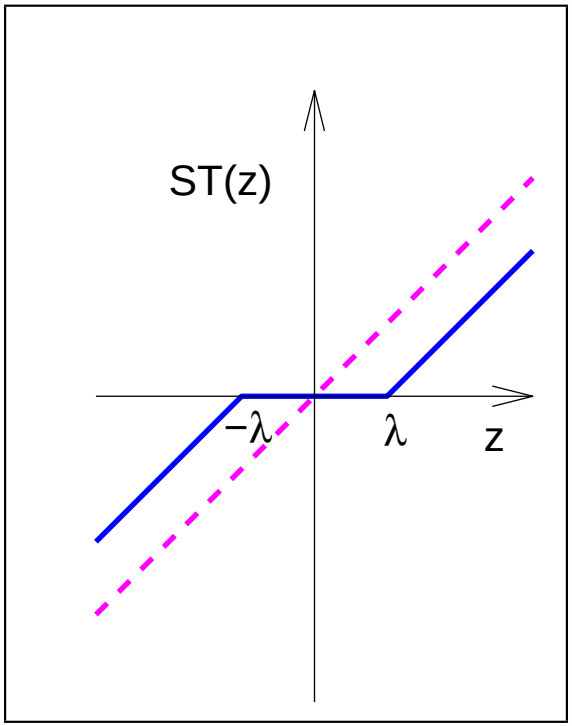
$$\underset{\boldsymbol{w}\in\mathbb{R}^n}{\text{minimize}} \quad f_\ell(\boldsymbol{A}\boldsymbol{w}) + \phi_\lambda(\boldsymbol{w}).$$

を解きたい．ただし，

- f_ℓ **は凸で 2 階微分可能．**
- $\phi_\lambda(\boldsymbol{w})$ **は例えば** $\phi_\lambda(\boldsymbol{w}) = \lambda\|\boldsymbol{w}\|_1$ **で，以下の最小化が陽に求まるもの．**

$$\mathrm{ST}_\lambda(\boldsymbol{z}) = \underset{\boldsymbol{w}\in\mathbb{R}^n}{\operatorname{argmin}} \left(\frac{1}{2}\|\boldsymbol{w}-\boldsymbol{z}\|_2^2 + \phi_\lambda(\boldsymbol{w}) \right)$$

- ϕ_λ **の微分不可能性** $\Rightarrow$ST **による打ち切り．スパース性が計算効率を高める！**
- $\boldsymbol{A}$ **のスケーリングにロバストにしたい．**



Outline

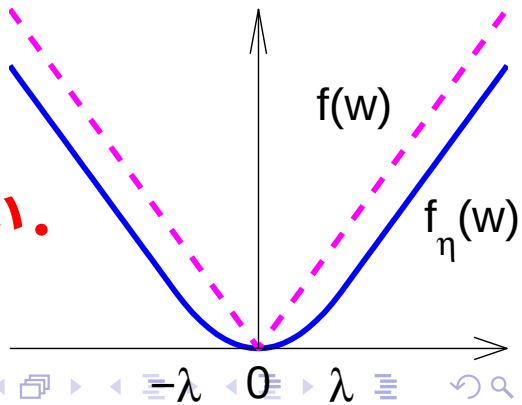
Outline

Proximal Minimization (Rockafellar, 1976)

1. 適当に初期解 $\boldsymbol{w}^1$ を決める．
2. 停止条件が満たされるまで反復：

$$\boldsymbol{w}^{t+1} \leftarrow \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \Big(\underbrace{f_\ell(\boldsymbol{A}\boldsymbol{w})}_{\text{線形近似しない}} + \phi_\lambda(\boldsymbol{w}) + \frac{1}{2\eta_t} \underbrace{\|\boldsymbol{w}-\boldsymbol{w}^t\|_2^2}_{\text{前の反復からの距離}^2\text{にペナルティ}} \Big)$$

- $f_\eta(\boldsymbol{w}) = \min_{\tilde{\boldsymbol{w}}\in\mathbb{R}^n} \left(f_\ell(\boldsymbol{A}\tilde{\boldsymbol{w}}) + \phi_\lambda(\tilde{\boldsymbol{w}}) + \frac{1}{2\eta}\|\tilde{\boldsymbol{w}}-\boldsymbol{w}\|_2^2 \right)$
  とおくと，
  - 事実 1: $f_\eta(\boldsymbol{w}) \le f(\boldsymbol{w}) = f_\ell(\boldsymbol{A}\boldsymbol{w}) + \phi_\lambda(\boldsymbol{w})$.
  - 事実 2: $f_\eta(\boldsymbol{w}^*) = f(\boldsymbol{w}^*)$.
- このままではもとの最適化問題と同程度に難しい．



IST 法（既存手法）

1. 適当に初期解 $\boldsymbol{w}^1$ を決める．
2. 停止条件が満たされるまで反復：

$$\boldsymbol{w}^{t+1} \leftarrow \mathrm{ST}_{\eta_t \lambda}\left(\boldsymbol{w}^t + \eta_t \boldsymbol{A}^\top(-\nabla f_\ell(\boldsymbol{A}\boldsymbol{w}^t))\right)$$

Dual Augmented Lagrangian 法（提案法）

1. 適当に初期解 $\boldsymbol{w}^1$ を決める．
2. 停止条件が満たされるまで反復：

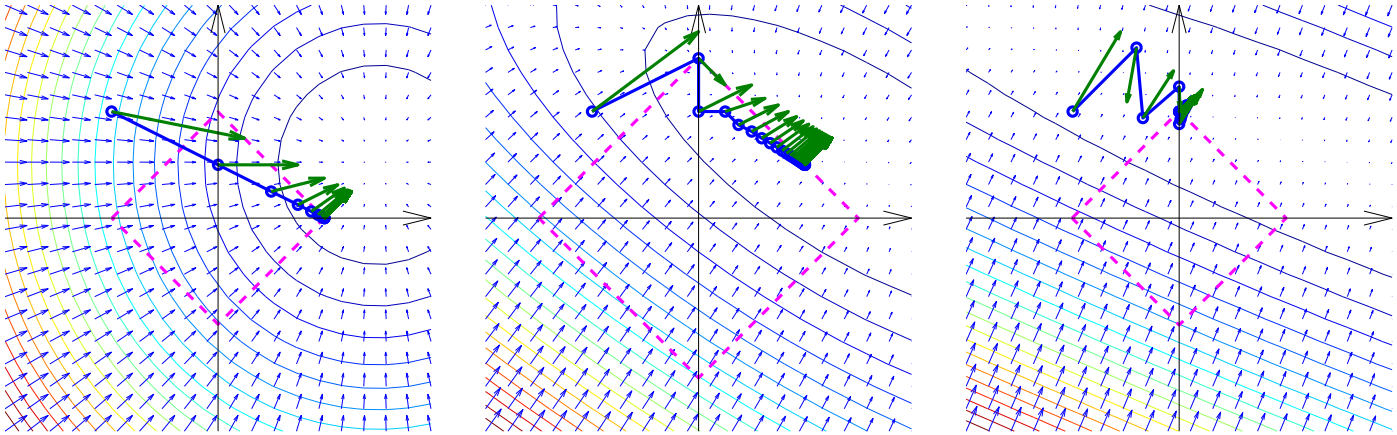
$$\boldsymbol{w}^{t+1} \leftarrow \mathrm{ST}_{\eta_t \lambda}\left(\boldsymbol{w}^t + \eta_t \boldsymbol{A}^\top \boldsymbol{\alpha}^t\right)$$

ただし，

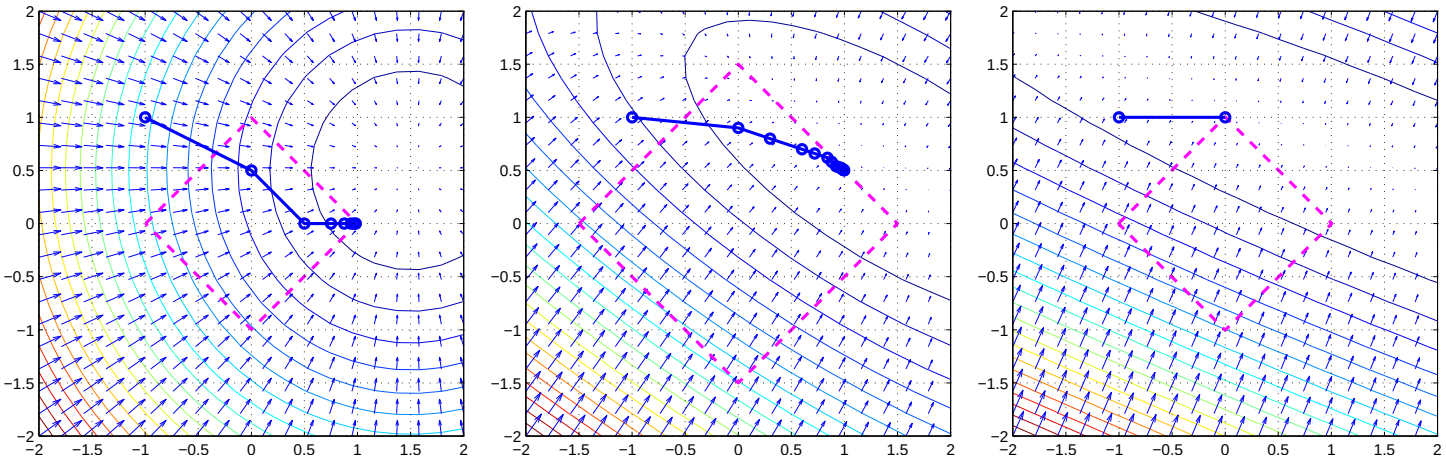
$$\boldsymbol{\alpha}^t = \operatorname*{argmin}_{\boldsymbol{\alpha} \in \mathbb{R}^m}\left(f_\ell^*(-\boldsymbol{\alpha}) + \frac{1}{2\eta_t}\|\mathrm{ST}_{\eta_t \lambda}(\boldsymbol{w}^t + \eta_t \boldsymbol{A}^\top \boldsymbol{\alpha})\|_2^2\right)$$

数値例

IST



DAL



Dual Augmented Lagrangian 法

1. 適当に初期解 $\boldsymbol{w}^1$ を決める．
2. 停止条件が満たされるまで反復：

$$\boldsymbol{w}^{t+1} \leftarrow \mathrm{ST}_{\eta_t \lambda}\left(\boldsymbol{w}^t + \eta_t \boldsymbol{A}^\top \boldsymbol{\alpha}^t\right)$$

ただし，

$$\boldsymbol{\alpha}^t = \operatorname*{argmin}_{\boldsymbol{\alpha} \in \mathbb{R}^m} \Bigg(\underbrace{f_\ell^*(-\boldsymbol{\alpha})}_{\text{損失関数 } f_\ell \text{ の凸共役}} + \frac{1}{2\eta_t} \|\mathrm{ST}_{\eta_t \lambda}(\boldsymbol{w}^t + \eta_t \boldsymbol{A}^\top \boldsymbol{\alpha})\|_2^2 \Bigg)$$

凸共役（Legendre 変換）

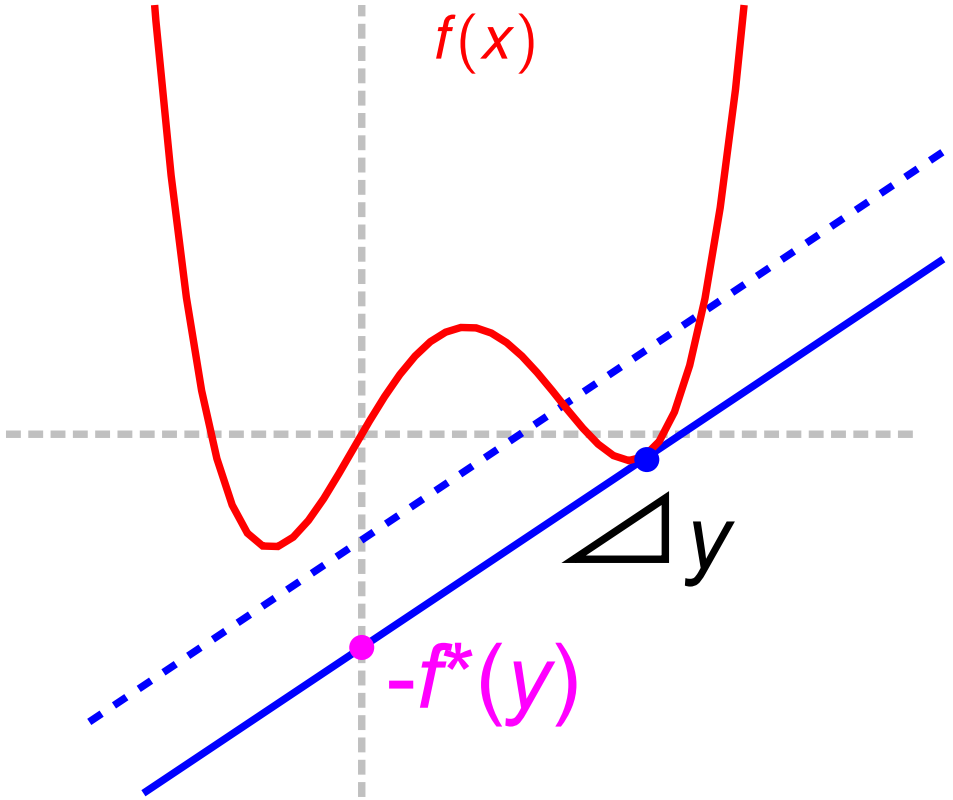
$$f^*(\boldsymbol{y}) = \sup_{\boldsymbol{x}} \left(\boldsymbol{y}^\top \boldsymbol{x} - f(\boldsymbol{x})\right)$$

Outline

Legendre 変換

関数 $f(x)$ を関数 $f^*(y)$ に移す変換（フーリエ変換のようなもの）

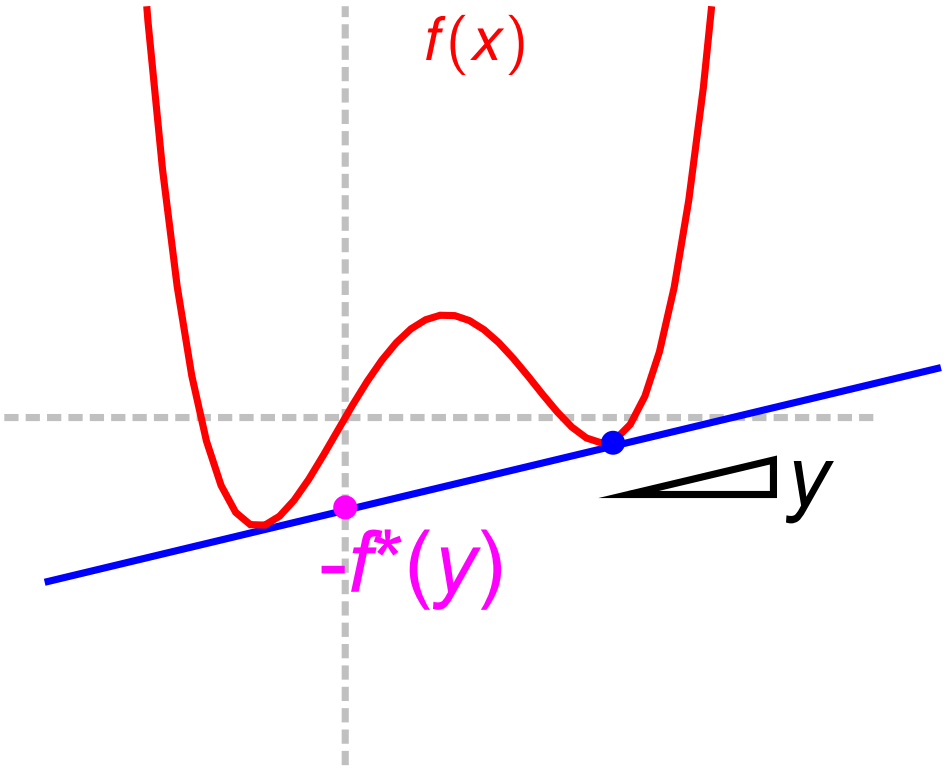
$$f^*(\boldsymbol{y}) = \sup_{\boldsymbol{x}} \left(\boldsymbol{y}^\top \boldsymbol{x} - f(\boldsymbol{x}) \right)$$



Legendre 変換

関数 $f(x)$ を関数 $f^*(y)$ に移す変換（フーリエ変換のようなもの）

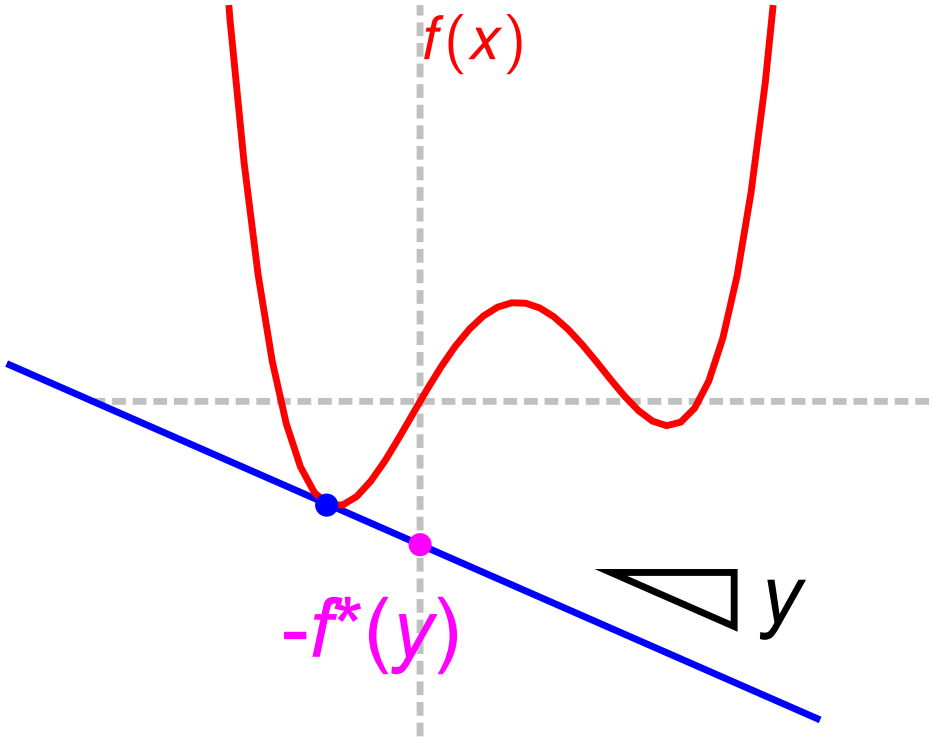
$$f^*(\boldsymbol{y}) = \sup_{\boldsymbol{x}} \left(\boldsymbol{y}^\top \boldsymbol{x} - f(\boldsymbol{x}) \right)$$



Legendre 変換

関数 $f(x)$ を関数 $f^*(y)$ に移す変換（フーリエ変換のようなもの）

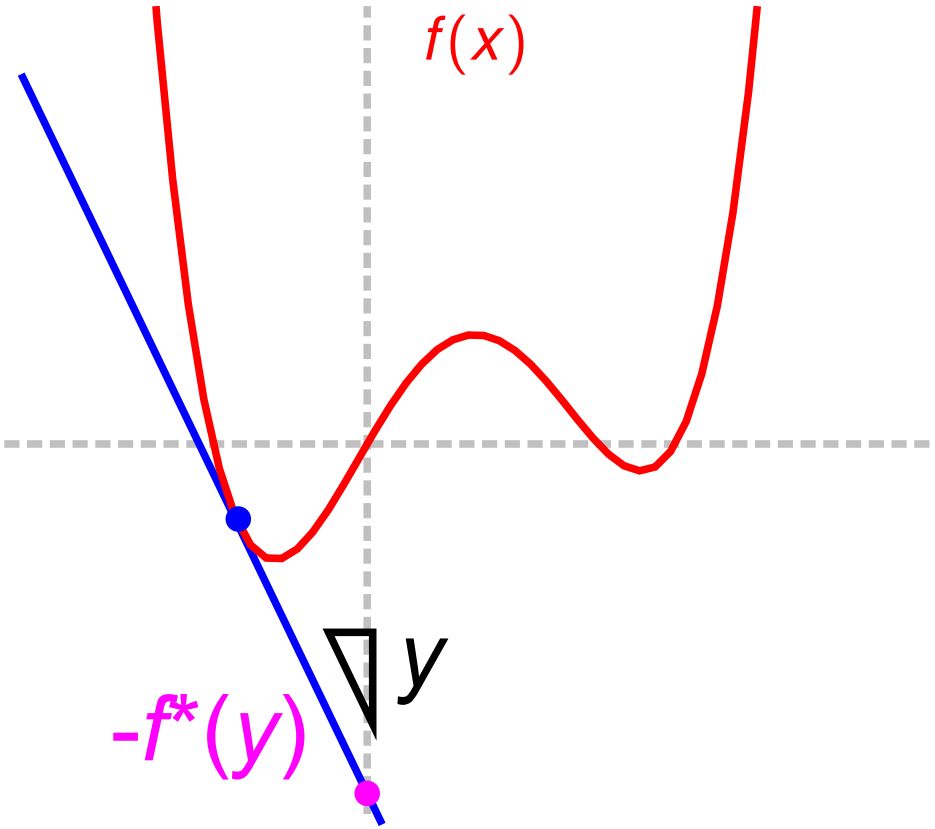
$$f^*(\boldsymbol{y}) = \sup_{\boldsymbol{x}} \left(\boldsymbol{y}^\top \boldsymbol{x} - f(\boldsymbol{x}) \right)$$



Legendre 変換

関数 $f(x)$ を関数 $f^*(y)$ に移す変換（フーリエ変換のようなもの）

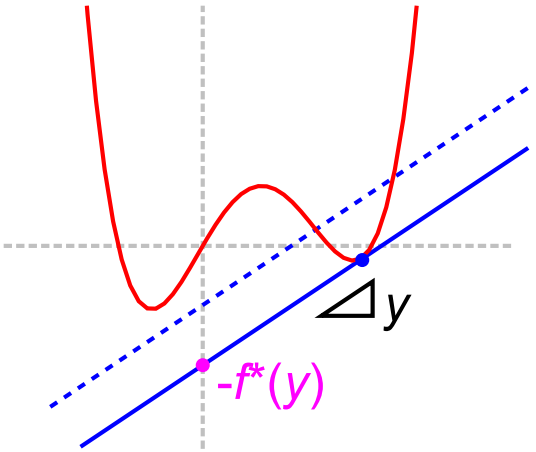
$$f^*(\boldsymbol{y}) = \sup_{\boldsymbol{x}} \left(\boldsymbol{y}^\top \boldsymbol{x} - f(\boldsymbol{x}) \right)$$



Legendre 変換

関数 $f(x)$ を関数 $f^*(y)$ に移す変換
（フーリエ変換のようなもの）

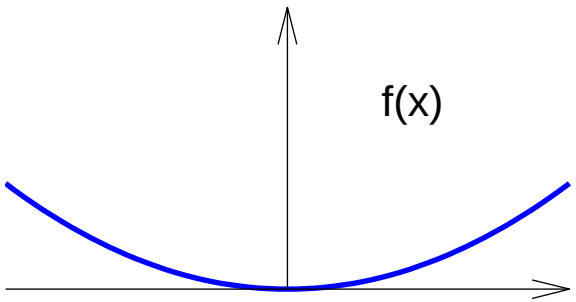
$$f^*(\boldsymbol{y}) = \sup_{\boldsymbol{x}} \left(\boldsymbol{y}^\top \boldsymbol{x} - f(\boldsymbol{x}) \right)$$

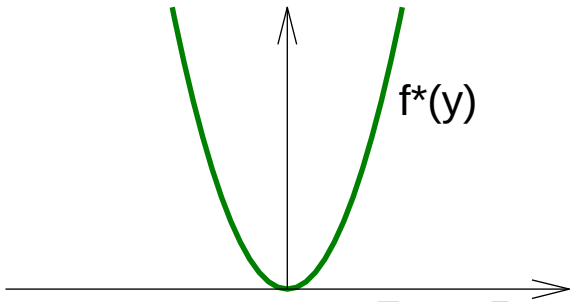


例 1（2 乗誤差関数）：

$$f(x) = \frac{x^2}{2\sigma^2} \quad \Rightarrow \quad f^*(y) = \sup_x \left(xy - \frac{x^2}{2\sigma^2} \right) = \left((\sigma^2 y) y - \frac{(\sigma^2 y)^2}{2\sigma^2} \right)$$

$$= \frac{\sigma^2 y^2}{2}$$





Legendre 変換の性質

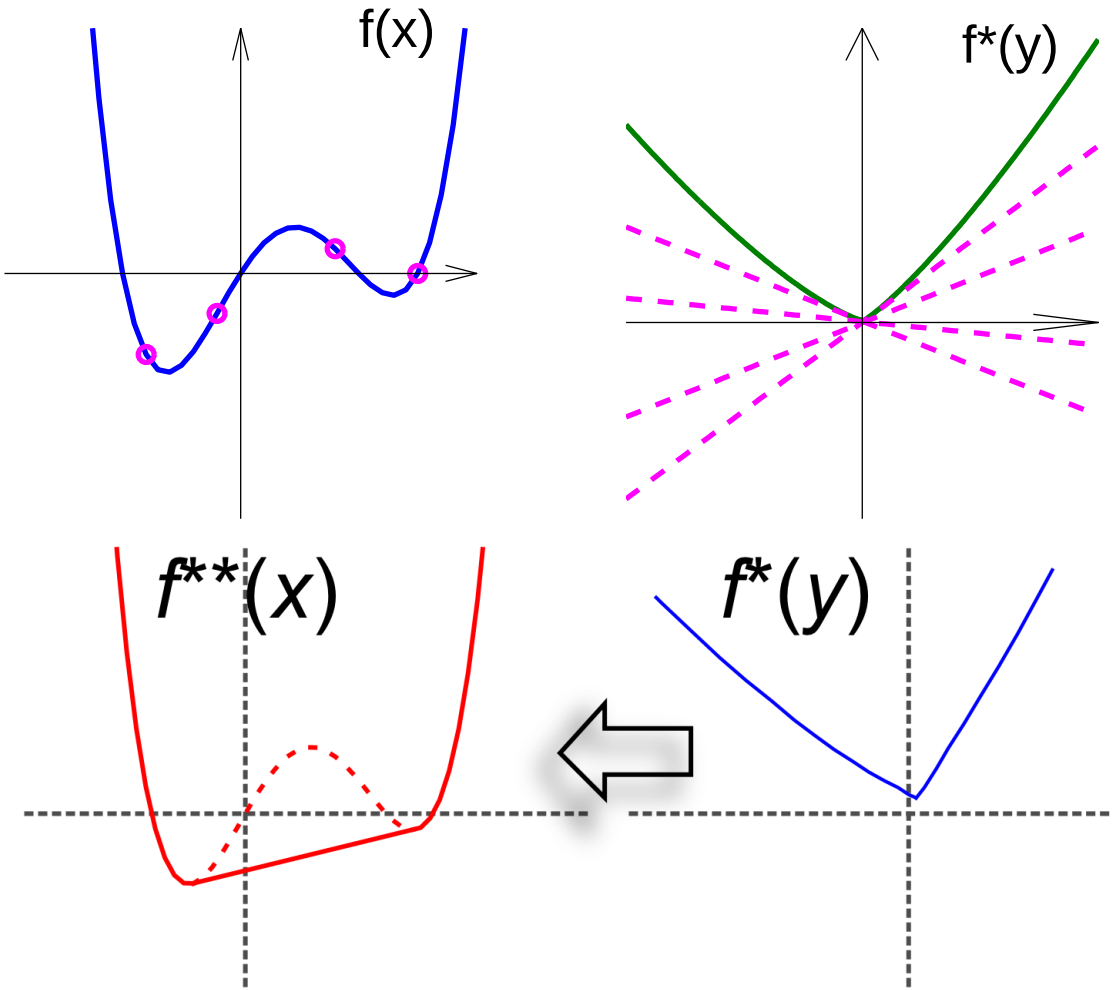
- $f^*(y)$ はいつも凸．

$$\because \quad f^*(y) \geq \underbrace{xy - f(x)}_{\text{線形な下限}}$$

- f が凸なら $f^{**}(x) = f(x)$．

$$\because \quad f(x) \geq xy - f^*(y)$$

（ただし等号が成り立つとは限らない）



Legendre 変換: $f^*(\boldsymbol{y}) = \sup_{\boldsymbol{x}} \left(\boldsymbol{y}^\top \boldsymbol{x} - f(\boldsymbol{x})\right)$

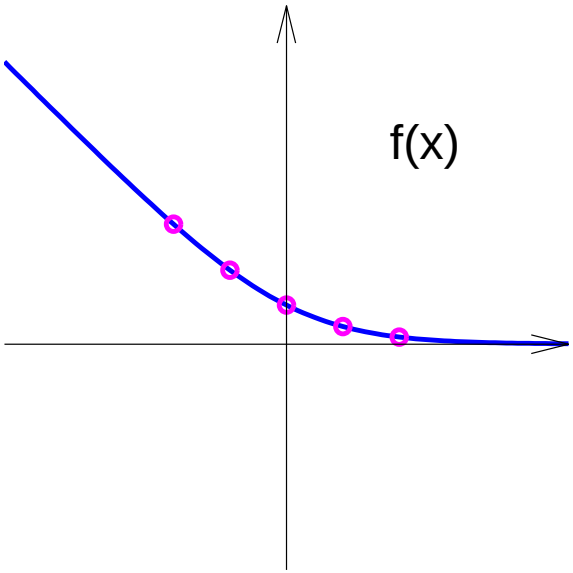
例 2（ロジスティック損失関数）：

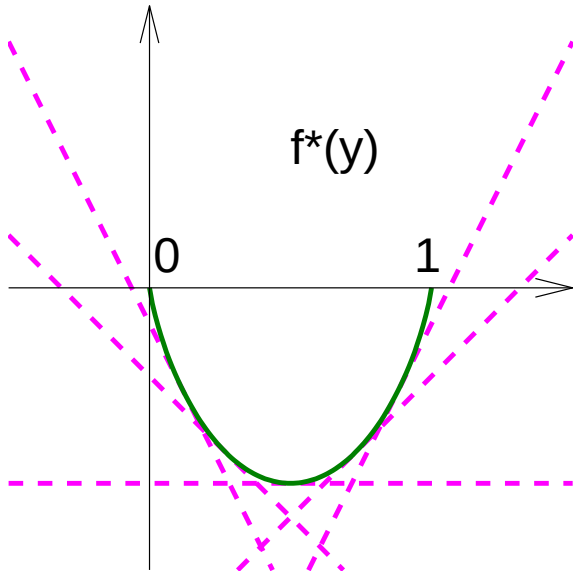
$$f(x) = \log(1+\exp(-x))$$

$$\Rightarrow \quad f^*(-y) = \sup_x \left(-xy - \log(1+\exp(-x))\right)$$

$$= \left(-(\log \tfrac{1-y}{y})y - \log(1+\tfrac{y}{1-y})\right)$$

$$= y\log(y) + (1-y)\log(1-y) \quad \text{（エントロピーの符号反転）}$$

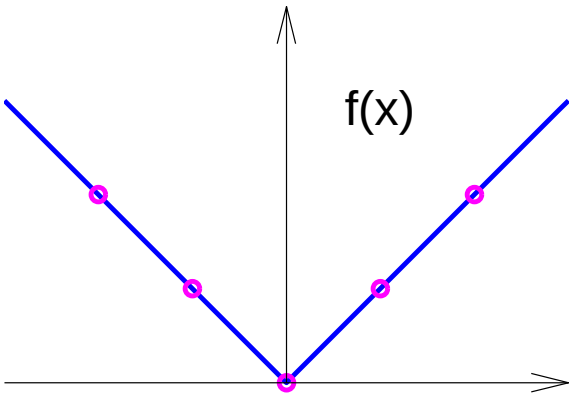


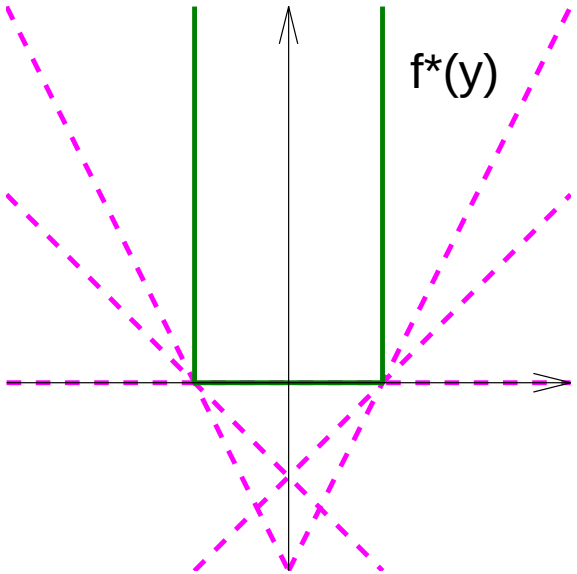


Legendre 変換: $f^*(\boldsymbol{y}) = \sup_{\boldsymbol{x}} \left(\boldsymbol{y}^\top \boldsymbol{x} - f(\boldsymbol{x})\right)$

例 3（ℓ_1 正則化関数）：

$$f(x) = |x| \quad \Rightarrow \quad f^*(y) = \sup_x (xy - |x|) = \begin{cases} 0 & (|y| \leq 1), \\ +\infty & (\text{otherwise}). \end{cases}$$





双対を使う

主問題の目的関数を双対を使って表現

$$f_\ell(\boldsymbol{A}\boldsymbol{w}) + \phi_\lambda(\boldsymbol{w}) = \max_{\boldsymbol{v}\in\mathbb{R}^n, \boldsymbol{\alpha}\in\mathbb{R}^m} \Big(\underbrace{\boldsymbol{w}^\top(\boldsymbol{v} - \boldsymbol{A}^\top\boldsymbol{\alpha}) - (f_\ell^*(-\boldsymbol{\alpha}) + \phi_\lambda^*(\boldsymbol{v}))}_{\boldsymbol{w}\text{ に関し線形な下限}} \Big)$$

$$\begin{aligned}
\because) \quad & \max_{\boldsymbol{v}\in\mathbb{R}^n, \boldsymbol{\alpha}\in\mathbb{R}^m} \Big(\boldsymbol{w}^\top(\boldsymbol{v} - \boldsymbol{A}^\top\boldsymbol{\alpha}) - f_\ell^*(-\boldsymbol{\alpha}) - \phi_\lambda^*(\boldsymbol{v}) \Big) \\
&= \max_{\boldsymbol{\alpha}\in\mathbb{R}^m} \Big(-\boldsymbol{w}^\top\boldsymbol{A}^\top\boldsymbol{\alpha} - f_\ell^*(-\boldsymbol{\alpha}) \Big) + \max_{\boldsymbol{v}\in\mathbb{R}^n} \Big(\boldsymbol{w}^\top\boldsymbol{v} - \phi_\lambda^*(\boldsymbol{v}) \Big) \\
&= f_\ell(\boldsymbol{A}\boldsymbol{w}) + \phi_\lambda(\boldsymbol{w})
\end{aligned}$$

Proximal minimization の式に代入

$$\boldsymbol{w}^{t+1} \leftarrow \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(f_\ell(\boldsymbol{A}\boldsymbol{w}) + \phi_\lambda(\boldsymbol{w}) + \frac{1}{2\eta_t}\|\boldsymbol{w}-\boldsymbol{w}^t\|_2^2 \right)$$

$$= \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left\{ \max_{\boldsymbol{v}\in\mathbb{R}^n,\boldsymbol{\alpha}\in\mathbb{R}^m} \left(-f_\ell^*(-\boldsymbol{\alpha}) - \phi_\lambda^*(\boldsymbol{v}) + \boldsymbol{w}^\top(\boldsymbol{v}-\boldsymbol{A}^\top\boldsymbol{\alpha}) \right) + \frac{1}{2\eta_t}\|\boldsymbol{w}-\boldsymbol{w}^t\|_2^2 \right\}$$

min と max の順番を交換し，あとは計算，計算...

Proximal minimization の式に代入

$$\boldsymbol{w}^{t+1} \leftarrow \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(f_\ell(\boldsymbol{A}\boldsymbol{w}) + \phi_\lambda(\boldsymbol{w}) + \frac{1}{2\eta_t}\|\boldsymbol{w}-\boldsymbol{w}^t\|_2^2 \right)$$

$$= \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left\{ \max_{\boldsymbol{v}\in\mathbb{R}^n,\boldsymbol{\alpha}\in\mathbb{R}^m} \left(-f_\ell^*(-\boldsymbol{\alpha}) - \phi_\lambda^*(\boldsymbol{v}) + \boldsymbol{w}^\top(\boldsymbol{v}-\boldsymbol{A}^\top\boldsymbol{\alpha}) \right) + \frac{1}{2\eta_t}\|\boldsymbol{w}-\boldsymbol{w}^t\|_2^2 \right\}$$

min と max の順番を交換し，あとは計算，計算...

Proximal minimization の式に代入

$$\boldsymbol{w}^{t+1} \leftarrow \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(f_\ell(\boldsymbol{A}\boldsymbol{w}) + \phi_\lambda(\boldsymbol{w}) + \frac{1}{2\eta_t}\|\boldsymbol{w}-\boldsymbol{w}^t\|_2^2 \right)$$

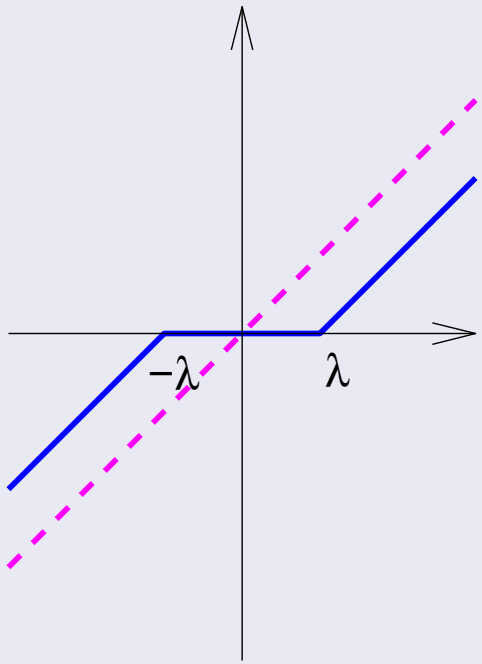
$$= \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left\{ \max_{\boldsymbol{v}\in\mathbb{R}^n,\boldsymbol{\alpha}\in\mathbb{R}^m} \left(-f_\ell^*(-\boldsymbol{\alpha}) - \phi_\lambda^*(\boldsymbol{v}) + \boldsymbol{w}^\top(\boldsymbol{v}-\boldsymbol{A}^\top\boldsymbol{\alpha}) \right) + \frac{1}{2\eta_t}\|\boldsymbol{w}-\boldsymbol{w}^t\|_2^2 \right\}$$

min と max の順番を交換し，あとは計算，計算...

Dual Augmented Lagrangian 法

(1) この計算は解析的にできると仮定．

$$\boldsymbol{w}^{t+1} \leftarrow \mathrm{ST}_{\eta_t \lambda}\left(\boldsymbol{w}^t + \boldsymbol{A}^\top \alpha^t\right)$$



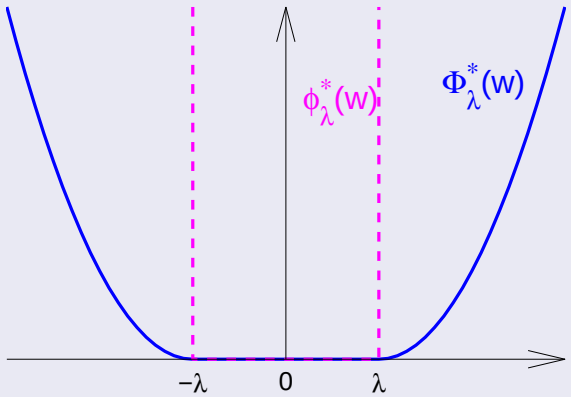
Dual Augmented Lagrangian 法

(1) この計算は解析的にできると仮定．

$$\boldsymbol{w}^{t+1} \leftarrow \mathrm{ST}_{\eta_t \lambda}\left(\boldsymbol{w}^t + \boldsymbol{A}^\top \alpha^t\right)$$

(2) この計算はスパースであればあるほど効率的 （**“アクティブな”未知変数の数に線形**）

$$\alpha^t = \operatorname*{argmin}_{\alpha \in \mathbb{R}^m}\left(f_\ell^*(-\alpha) + \frac{1}{2\eta_t}\|\mathrm{ST}_{\eta_t\lambda}(\boldsymbol{w}^t + \eta_t \boldsymbol{A}^\top \alpha)\|_2^2\right)$$



Dual Augmented Lagrangian 法

(1) **この計算は解析的にできると仮定．**

$$\boldsymbol{w}^{t+1} \leftarrow \mathrm{ST}_{\eta_t \lambda}\left(\boldsymbol{w}^t + \boldsymbol{A}^\top \alpha^t\right)$$

(2) **この計算はスパースであればあるほど効率的．** （**“アクティブな”未知変数の数に線形**）

$$\alpha^t = \underset{\alpha \in \mathbb{R}^m}{\mathrm{argmin}} \Big(\underbrace{f_\ell^*(-\alpha)}_{\boldsymbol{A}\text{のスケーリングの影響を受けない}} + \underbrace{\frac{1}{2\eta_t} \|\mathrm{ST}_{\eta_t\lambda}(\boldsymbol{w}^t + \eta_t \boldsymbol{A}^\top \alpha)\|_2^2}_{\begin{array}{l}\frac{\partial}{\partial \alpha} : \boldsymbol{A}\mathrm{ST}_{\eta_t\lambda}(\boldsymbol{w}^t + \eta_t \boldsymbol{A}^\top \alpha) \\ \frac{\partial}{\partial \alpha^2} : \eta_t \boldsymbol{A}_+ \boldsymbol{A}_+{}^\top\end{array}} \Big)$$

（$\boldsymbol{A}_+$ は $\boldsymbol{A}$ の“アクティブな”列からなる部分行列; ℓ_1-正則化の場合）

(3) **$\boldsymbol{A}$ のスケーリングの悪さの影響を受けにくい．**

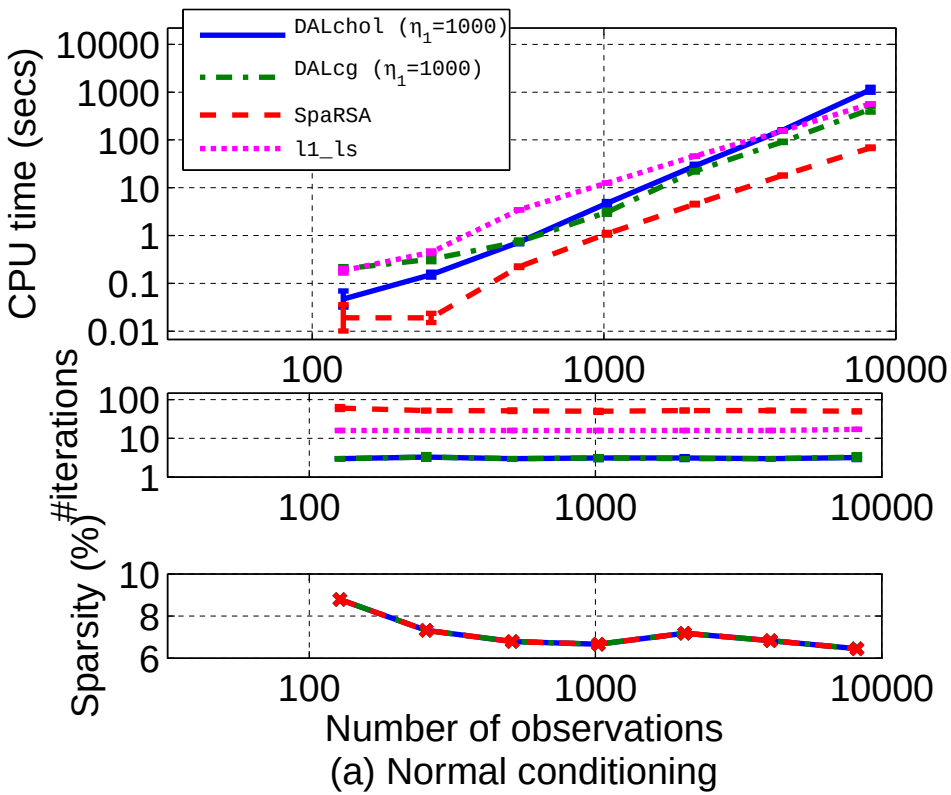
Outline

実験（設定）

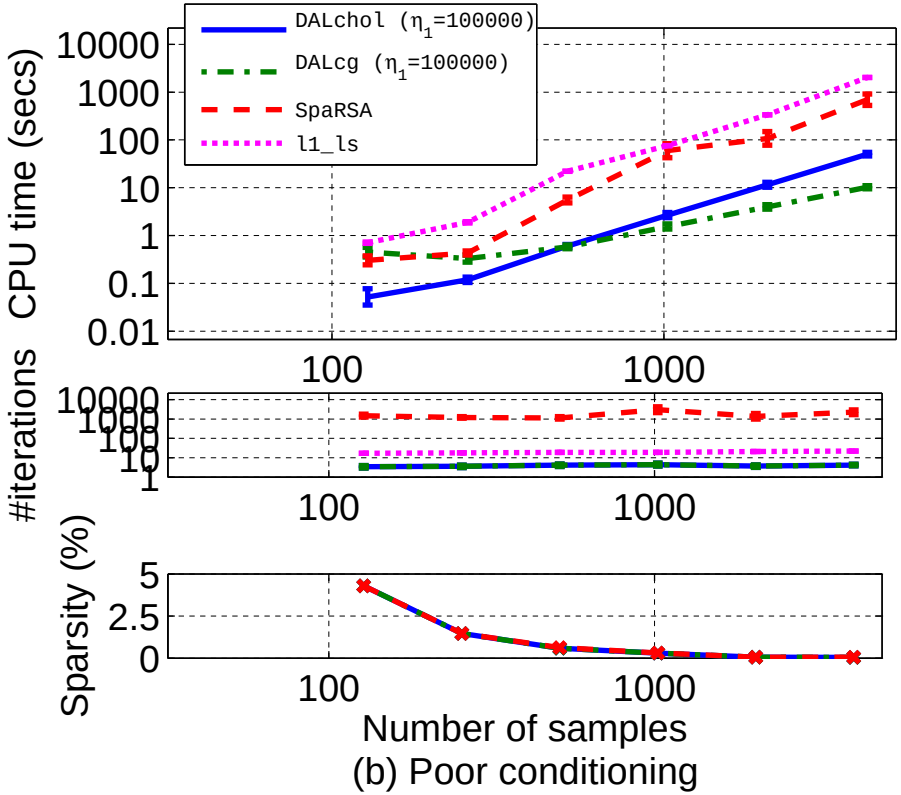
- **問題：** LASSO （2 乗損失＋ ℓ_1 正則化）
- **対抗手法：**
  - l1_ls （内点法）
  - SpaRSA （ステップサイズ改良 IST）
- **ランダムデザイン行列** $A \in \mathbb{R}^{m \times n}$ (m: **サンプル数** n: **未知変数の数**)
  - `A=randn(m,n);` （良条件数）
  - `A=U*diag(1./(1:m))*V';` (悪条件数)
- 2 **つの状況**
  - **中規模** ($n = 4m$, $n < 10000$)
  - **大規模** ($m = 1024$, $n < 1e + 6$)

結果（中規模）

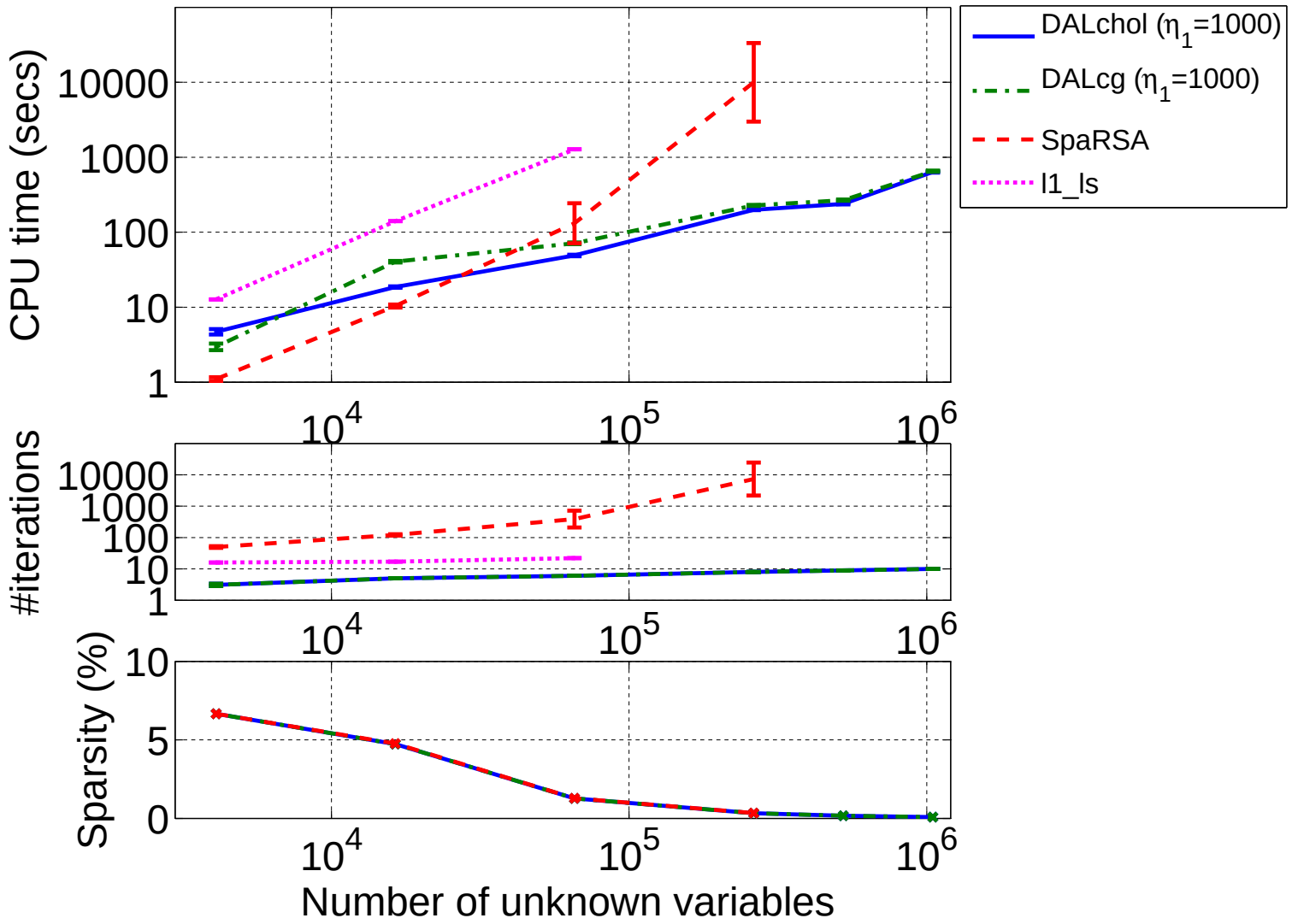
良条件数



悪条件数



結果（大規模）


CPU time (secs)
10000
1000
100
10
1
#iterations
10000
1000
100
10
1
Sparsity (%)
10
5
0
10^4
10^5
10^6
Number of unknown variables
DALchol (η_1=1000)
DALcg (η_1=1000)
SpaRSA
l1_ls

Outline

目的

本来やりたいこと：

$$\underset{f\in\mathcal{H},b\in\mathbb{R},\boldsymbol{d}\in\mathbb{R}^n}{\text{minimize}} \quad \sum_{i=1}^{m}\xi_i + \frac{\lambda}{2}\|f\|^2_{\mathcal{H}(d)}$$

$$\text{subject to} \quad y_i(f(x_i)+b) \geq 1-\xi_i \qquad (i=1,\ldots,m)$$

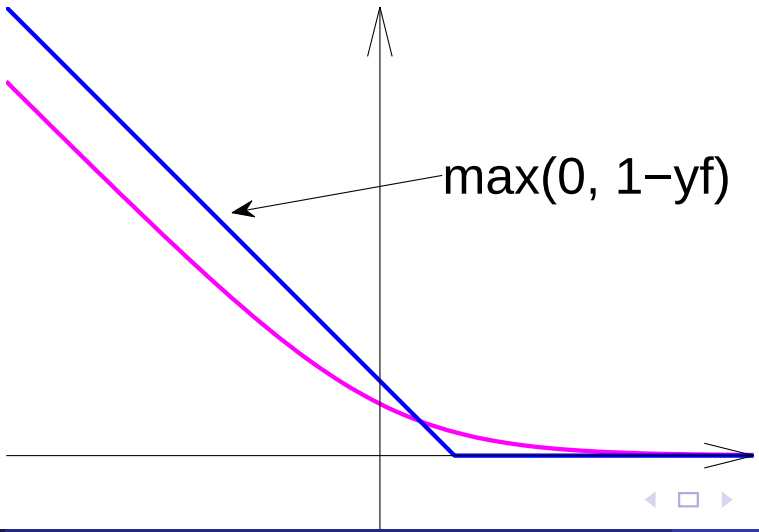
$$\boldsymbol{K}(d) = \sum_{i=1}^{n} d_j \boldsymbol{K}_j, \quad d_j \geq 0, \quad \sum_j d_j \leq 1.$$

表現定理を適用

$$\underset{\boldsymbol{\alpha}\in\mathbb{R}^m, b\in\mathbb{R}, \boldsymbol{d}\in\mathbb{R}^n}{\text{minimize}} \quad \sum_{i=1}^{m} \xi_i + \frac{\lambda}{2}\boldsymbol{\alpha}^\top \boldsymbol{K}(d)\boldsymbol{\alpha}$$

$$\text{subject to} \quad y_i((\boldsymbol{K}(d)\boldsymbol{\alpha})_i + b) \geq 1 - \xi_i \qquad (i = 1, \ldots, m)$$

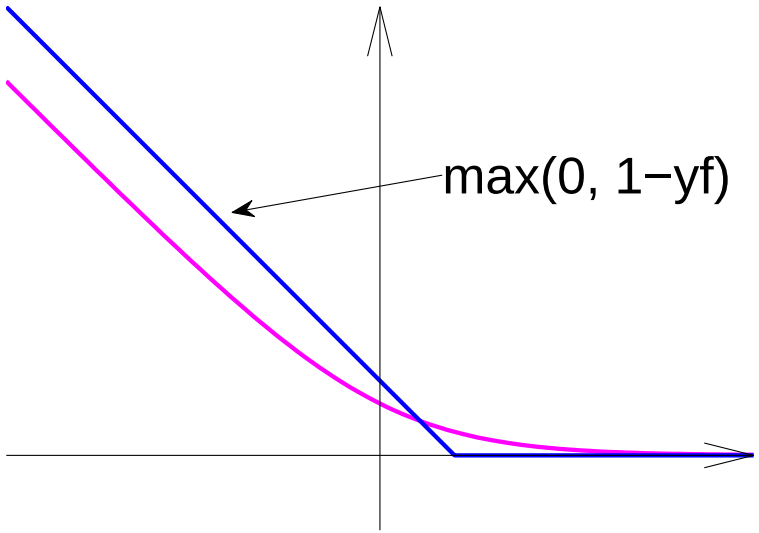
$$\boldsymbol{K}(d) = \sum_{i=1}^{n} d_j \boldsymbol{K}_j, \quad d_j \geq 0, \quad \sum_j d_j \leq 1.$$



損失関数を定義

$$\underset{\boldsymbol{\alpha}\in\mathbb{R}^m, b\in\mathbb{R}, \boldsymbol{d}\in\mathbb{R}^n}{\text{minimize}} \quad L(\boldsymbol{K}(d)\boldsymbol{\alpha} + b\mathbf{1}) + \frac{\lambda}{2}\boldsymbol{\alpha}^\top \boldsymbol{K}(d)\boldsymbol{\alpha}$$

$$\text{subject to} \quad \boldsymbol{K}(d) = \sum_{i=1}^{n} d_j \boldsymbol{K}_j, \quad d_j \geq 0, \quad \sum_j d_j \leq 1.$$



正則化項を緩和

$$\underset{\boldsymbol{\alpha}\in\mathbb{R}^m, b\in\mathbb{R}, \boldsymbol{\gamma}\in\mathbb{R}^n}{\text{minimize}} \quad L(\boldsymbol{K}(d)\boldsymbol{\alpha} + b\mathbf{1}) + \frac{\lambda}{2}\boldsymbol{\alpha}^\top \boldsymbol{K}(d)\boldsymbol{\alpha}$$

$$\text{subject to} \quad \boldsymbol{K}(d) = \sum_{i=1}^{n} d_j \boldsymbol{K}_j, \quad d_j \geq 0, \quad \sum_i d_j \leq 1.$$

補助変数 α_j $(j = 1, \ldots, n)$ を導入して正則化項を緩和

$$\boldsymbol{\alpha}^\top \boldsymbol{K}(d)\boldsymbol{\alpha} = \min_{\boldsymbol{\alpha}_j \in \mathbb{R}^m} \left(\sum_{j=1}^{n} \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j} \right) \qquad \text{subject to} \sum_{j=1}^{n} \boldsymbol{K}_j \boldsymbol{\alpha}_j = \boldsymbol{K}(d)\boldsymbol{\alpha}$$

ラグランジュ乗数 β を導入し，

$$\frac{1}{2} \sum_{j=1}^{n} \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j} + \beta^\top \left(\boldsymbol{K}(d)\boldsymbol{\alpha} - \sum_{j=1}^{n} \boldsymbol{K}_j \boldsymbol{\alpha}_j \right)$$

を最小化．$\alpha_j = d_j \beta$，$\beta = \alpha$ を得る．

正則化項を緩和

$$\underset{\boldsymbol{\alpha}\in\mathbb{R}^m, b\in\mathbb{R}, \boldsymbol{\gamma}\in\mathbb{R}^n}{\text{minimize}} \quad L(\boldsymbol{K}(d)\boldsymbol{\alpha} + b\mathbf{1}) + \frac{\lambda}{2}\boldsymbol{\alpha}^\top \boldsymbol{K}(d)\boldsymbol{\alpha}$$

$$\text{subject to} \quad \boldsymbol{K}(d) = \sum_{i=1}^{n} d_j \boldsymbol{K}_j, \quad d_j \geq 0, \quad \sum_j d_j \leq 1.$$

補助変数 α_j $(j = 1, \ldots, n)$ を導入して正則化項を緩和

$$\boldsymbol{\alpha}^\top \boldsymbol{K}(d)\boldsymbol{\alpha} = \min_{\boldsymbol{\alpha}_j \in \mathbb{R}^m} \left(\sum_{j=1}^{n} \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j} \right) \qquad \text{subject to} \sum_{j=1}^{n} \boldsymbol{K}_j \boldsymbol{\alpha}_j = \boldsymbol{K}(d)\boldsymbol{\alpha}$$

ラグランジュ乗数 β を導入し，

$$\frac{1}{2}\sum_{j=1}^{n} \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j} + \beta^\top \left(\boldsymbol{K}(d)\boldsymbol{\alpha} - \sum_{j=1}^{n} \boldsymbol{K}_j \boldsymbol{\alpha}_j \right)$$

を最小化．$\alpha_j = d_j \beta$，$\beta = \alpha$ を得る．

正則化項を緩和

$$\underset{\boldsymbol{\alpha}\in\mathbb{R}^m, b\in\mathbb{R}, \boldsymbol{\gamma}\in\mathbb{R}^n}{\text{minimize}} \quad L(\boldsymbol{K}(d)\boldsymbol{\alpha} + b\mathbf{1}) + \frac{\lambda}{2}\boldsymbol{\alpha}^\top \boldsymbol{K}(d)\boldsymbol{\alpha}$$

$$\text{subject to} \quad \boldsymbol{K}(d) = \sum_{i=1}^{n} d_j \boldsymbol{K}_j, \quad d_j \geq 0, \quad \sum_j d_j \leq 1.$$

補助変数 α_j $(j = 1, \ldots, n)$ を導入して正則化項を緩和

$$\boldsymbol{\alpha}^\top \boldsymbol{K}(d)\boldsymbol{\alpha} = \min_{\boldsymbol{\alpha}_j \in \mathbb{R}^m} \left(\sum_{j=1}^{n} \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j} \right) \qquad \text{subject to} \sum_{j=1}^{n} \boldsymbol{K}_j \boldsymbol{\alpha}_j = \boldsymbol{K}(d)\boldsymbol{\alpha}$$

ラグランジュ乗数 β を導入し，

$$\frac{1}{2} \sum_{j=1}^{n} \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j} + \beta^\top \left(\boldsymbol{K}(d)\boldsymbol{\alpha} - \sum_{j=1}^{n} \boldsymbol{K}_j \boldsymbol{\alpha}_j \right)$$

を最小化．$\alpha_j = d_j \beta$，$\beta = \alpha$ を得る．

上限を最小化

$$\underset{\boldsymbol{\alpha}_j \in \mathbb{R}^m, b \in \mathbb{R}, \boldsymbol{\gamma} \in \mathbb{R}^n}{\text{minimize}} \quad L\left(\sum_{j=1}^{n} \boldsymbol{K}_j \boldsymbol{\alpha}_j + b\mathbf{1}\right) + \frac{\lambda}{2} \sum_{j=1}^{n} \frac{{\boldsymbol{\alpha}_j}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j}$$

$$\text{subject to} \quad d_j \geq 0, \quad \sum_j d_j \leq 1.$$

$$\sum_{j=1}^{n} \frac{{\boldsymbol{\alpha}_j}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j} = \sum_{j=1}^{n} d_j \frac{{\boldsymbol{\alpha}_j}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j^2} = \sum_{j=1}^{n} d_j \left(\frac{\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}}{d_j}\right)^2$$

$$\geq \left(\sum_{j=1}^{n} d_j \frac{\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}}{d_j}\right)^2 \qquad \left(\begin{array}{c}\sum_j d_j = 1 \text{ より} \\ \text{Jensen の不等式}\end{array}\right)$$

$$= \left(\sum_{j=1}^{n} \|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}\right)^2$$

↑ノルムの線形和．

上限を最小化

$$\underset{\boldsymbol{\alpha}_j \in \mathbb{R}^m, b \in \mathbb{R}, \boldsymbol{\gamma} \in \mathbb{R}^n}{\text{minimize}} \quad L\left(\sum_{j=1}^{n} \boldsymbol{K}_j \boldsymbol{\alpha}_j + b\mathbf{1}\right) + \frac{\lambda}{2} \sum_{j=1}^{n} \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j}$$

$$\text{subject to} \quad d_j \geq 0, \quad \sum_j d_j \leq 1.$$

$$\sum_{j=1}^{n} \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j} = \sum_{j=1}^{n} d_j \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j^2} = \sum_{j=1}^{n} d_j \left(\frac{\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}}{d_j} \right)^2$$

$$\geq \left(\sum_{j=1}^{n} d_j \frac{\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}}{d_j} \right)^2 \qquad \left(\begin{array}{l} \textstyle\sum_j d_j = 1 \text{ より} \\ \text{Jensen の不等式} \end{array} \right)$$

$$= \left(\sum_{j=1}^{n} \|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j} \right)^2$$

↑ノルムの線形和．

上限を最小化

$$\underset{\boldsymbol{\alpha}_j \in \mathbb{R}^m, b \in \mathbb{R}, \boldsymbol{\gamma} \in \mathbb{R}^n}{\text{minimize}} \quad L\left(\sum_{j=1}^{n} \boldsymbol{K}_j \boldsymbol{\alpha}_j + b\mathbf{1}\right) + \frac{\lambda}{2} \sum_{j=1}^{n} \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j}$$

$$\text{subject to} \quad d_j \geq 0, \quad \sum_j d_j \leq 1.$$

$$\sum_{j=1}^{n} \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j} = \sum_{j=1}^{n} d_j \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j^2} = \sum_{j=1}^{n} d_j \left(\frac{\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}}{d_j} \right)^2$$

$$\geq \left(\sum_{j=1}^{n} d_j \frac{\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}}{d_j} \right)^2 \qquad \left(\begin{array}{l} \sum_j d_j = 1 \text{ より} \\ \text{Jensen の不等式} \end{array} \right)$$

$$= \left(\sum_{j=1}^{n} \|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j} \right)^2$$

↑ノルムの線形和．

上限を最小化

$$\underset{\boldsymbol{\alpha}_j \in \mathbb{R}^m, b \in \mathbb{R}, \boldsymbol{\gamma} \in \mathbb{R}^n}{\text{minimize}} \quad L\left(\sum_{j=1}^{n} \boldsymbol{K}_j \boldsymbol{\alpha}_j + b\mathbf{1}\right) + \frac{\lambda}{2} \sum_{j=1}^{n} \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j}$$

$$\text{subject to} \quad d_j \geq 0, \quad \sum_j d_j \leq 1.$$

$$\sum_{j=1}^{n} \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j} = \sum_{j=1}^{n} d_j \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j^2} = \sum_{j=1}^{n} d_j \left(\frac{\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}}{d_j} \right)^2$$

$$\geq \left(\sum_{j=1}^{n} d_j \frac{\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}}{d_j} \right)^2 \qquad \left(\begin{array}{l} \sum_j d_j = 1 \text{ より} \\ \text{Jensen の不等式} \end{array} \right)$$

$$= \left(\sum_{j=1}^{n} \|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j} \right)^2$$

↑ノルムの線形和．

上限を最小化

$$\underset{\boldsymbol{\alpha}_j \in \mathbb{R}^m, b \in \mathbb{R}, \boldsymbol{\gamma} \in \mathbb{R}^n}{\text{minimize}} \quad L\left(\sum_{j=1}^{n} \boldsymbol{K}_j \boldsymbol{\alpha}_j + b\mathbf{1}\right) + \frac{\lambda}{2} \sum_{j=1}^{n} \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j}$$

$$\text{subject to} \quad d_j \geq 0, \quad \sum_j d_j \leq 1.$$

$$\sum_{j=1}^{n} \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j} = \sum_{j=1}^{n} d_j \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j^2} = \sum_{j=1}^{n} d_j \left(\frac{\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}}{d_j} \right)^2$$

$$\geq \left(\sum_{j=1}^{n} d_j \frac{\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}}{d_j} \right)^2 \qquad \left(\begin{array}{l} \sum_j d_j = 1 \text{ より} \\ \text{Jensen の不等式} \end{array} \right)$$

$$= \left(\sum_{j=1}^{n} \|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j} \right)^2$$

↑ノルムの線形和．

上限を最小化

$$\underset{\boldsymbol{\alpha}_j \in \mathbb{R}^m, b \in \mathbb{R}, \boldsymbol{\gamma} \in \mathbb{R}^n}{\text{minimize}} \quad L\left(\sum_{j=1}^{n} \boldsymbol{K}_j \boldsymbol{\alpha}_j + b\mathbf{1}\right) + \frac{\lambda}{2} \sum_{j=1}^{n} \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j}$$

$$\text{subject to} \quad d_j \geq 0, \quad \sum_j d_j \leq 1.$$

$$\sum_{j=1}^{n} \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j} = \sum_{j=1}^{n} d_j \frac{\boldsymbol{\alpha}_j{}^\top \boldsymbol{K}_j \boldsymbol{\alpha}_j}{d_j^2} = \sum_{j=1}^{n} d_j \left(\frac{\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}}{d_j} \right)^2$$

$$\geq \left(\sum_{j=1}^{n} d_j \frac{\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}}{d_j} \right)^2 \qquad \left(\begin{array}{l} \sum_j d_j = 1 \text{ より} \\ \text{Jensen の不等式} \end{array} \right)$$

$$= \left(\sum_{j=1}^{n} \|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j} \right)^2$$

↑ノルムの線形和．

2 つの定式化の同値性

ノルム線形和の 2 乗で正則化（一般的な定式化）

$$\underset{\boldsymbol{\alpha}_j \in \mathbb{R}^n, b \in \mathbb{R}}{\text{minimize}} \quad L\Big(\sum_{j=1}^{n} \boldsymbol{K}_j \boldsymbol{\alpha}_j + b\mathbf{1}\Big) + \frac{\lambda}{2}\Big(\sum_{j=1}^{n} \|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}\Big)^2 \qquad \text{(A)}$$

ノルム線形和で正則化（我々の定式化）

$$\Leftrightarrow \underset{\boldsymbol{\alpha}_j \in \mathbb{R}^n, b \in \mathbb{R}}{\text{minimize}} \quad L\Big(\sum_{j=1}^{n} \boldsymbol{K}_j \boldsymbol{\alpha}_j + b\mathbf{1}\Big) + \lambda' \sum_{j=1}^{n} \|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j} \qquad \text{(B)}$$

(A) **の最適性：** $\nabla_{\boldsymbol{\alpha}_j} L + \lambda\Big(\sum_{j=1}^{n} \|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}\Big) \partial_{\boldsymbol{\alpha}_j} \|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j} \ni 0$

(B) **の最適性：** $\nabla_{\boldsymbol{\alpha}_j} L + \lambda' \partial_{\boldsymbol{\alpha}_j} \|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j} \ni 0$

SpicyMKL

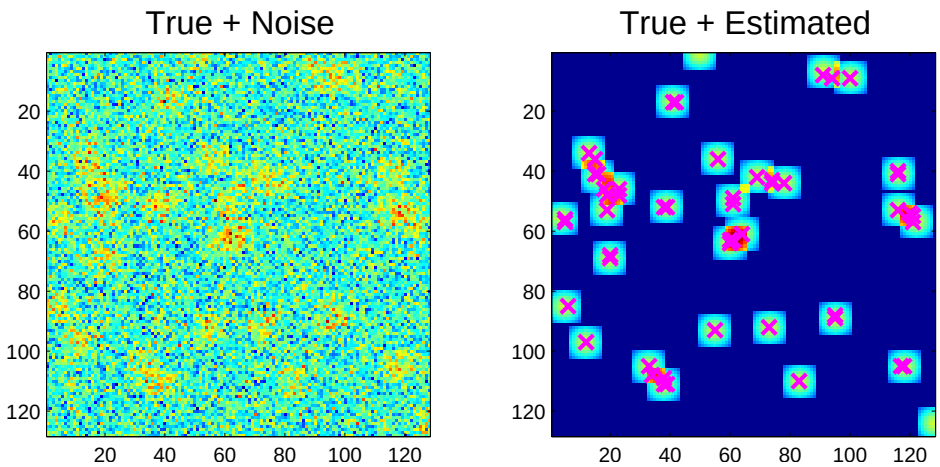
DAL + MKL = SpicyMKL (Sparse Iterative MKL)

- **基本的には** DAL **と同じ．**
- **バイアス項を扱う必要がある．**
- **ヒンジロスは微分できないので特別に扱う必要がる（今回の実験はロジスティック損失）．**
- Soft-thresholding **が（変数単位ではなく）カーネル単位でかかる．**

$$\mathrm{ST}_\lambda(\boldsymbol{\alpha}_j) = \begin{cases} 0 & (\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j} \leq \lambda) \\ \left(\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j} - \lambda\right) \frac{\boldsymbol{\alpha}_j}{\|\boldsymbol{\alpha}_j\|_{\boldsymbol{K}_j}} & (\text{otherwise}) \end{cases}$$

Outline

Demo1 – デコンボリューション



- **画像は** 128x128**.**
- **フィルタは** $\sigma = 5$ **のガウシアンぼかし.**
- **コマンド：**

```
[xx,stat]=dalsql1(zeros(m*n,1),H,Y(:),lambda,'eta',500,'solver','cg');
```

2乗ロス+L1正則化　初期値　畳み込み行列　入力画像　正則化定数　ペナルティーの強さ（の初期値）　インナーループの最適化にCG法を使う

Demo2 – バイオインフォマティクス

- 多発性硬化症に対する β インターフェロン療法の効果を検証.
- 53 人の患者の 70 遺伝子の発現データが投薬開始から最長 2 年間に渡って集められた.(t=0, 3, 6, 9, 12, 18, 24ヶ月後)
- 2 値分類問題(効果的/効果なし) $\rightarrow$ ロジスティック損失を使う.
- 2 つの設定
  - 時系列情報を扱うグループラッソーの問題.
  - 遺伝子の組みを探す MKL の問題.

Demo2.1 – グループラッソー

- **各遺伝子ごとに，**
  1. **（時間方向の）平均発現量**
  2. **時間方向 1 階差分の平均**
  3. **時間方向 2 階差分の平均**

  を計算．(3 × 70 次元特徴)

- **グループラッソー**: $\underset{\boldsymbol{w}\in\mathbb{R}^{3\times 70}, b\in\mathbb{R}}{\text{minimize}} \sum_{i=1}^{m} \ell^L(\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle + b) + \lambda \sum_{j=1}^{70} \|\boldsymbol{w}_j\|_2$
- Soft-threholding: $\mathrm{ST}_\lambda(\boldsymbol{w}_j) = \max(0, \|\boldsymbol{w}_j\|_2 - \lambda)\dfrac{\boldsymbol{w}_j}{\|\boldsymbol{w}_j\|_2}$
- **コマンド**

```
[ww,bb,stat]=dallrgl(zeros(ns,nc),F(:,:),Y(:), lambda);
```

重み　バイアス　ロジスティック損失 + グループラッソー正則化　初期値 (ns=3, nc=70)　特徴量　ラベル　正則化定数

Demo2.2 – MKL

- **時刻 0（治療開始時）のデータだけを利用．**
- Baranzini **らが見つけた遺伝子の 3 つ組 9 つにそれぞれ 2 次の多項式カーネル** $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = (1 + \boldsymbol{x}_i{}^\top \boldsymbol{x}_j)^2$ **を導入．**

```
opt=struct('loss','logit');
```
←ロジスティック損失を指定

```
[alpha,d,b,actset]=SpicyMKL(K,Y,lambda,opt);
```

Demo3 – 画像認識

- Caltech101 (Fei-Fei et al., 2004) の中から anchor, ant, cannon, chair, cup の 5 クラスを利用．
- 10 通りの 2 クラス分類問題．
- カーネル数 1,760 = 特徴抽出法（4 通り）× 領域分割（22 通り）× カーネル関数（20 通り）
  - 特徴抽出: van de Sande らのコードを利用．hsvsift, sift（スケール自動），sift（スケール 4px 固定）, sift（スケール 8px 固定）の 4 通り．
  - 領域分割と統合：画像全体，4 分割，16 分割し，それぞれの領域で visual words の出現頻度を計算，さらに，それらを spatial pyramid で統合したもの（計 22 通り）．
  - カーネル関数: ガウシアンカーネルと χ^2 カーネルをそれぞれ 10 通りのハイパーパラメータで用意．

Outline

まとめ

- ℓ_1-正則化：凸最適化だからといって終わりではない．まだまだ工夫の必要／余地がある．
- DAL：スパース性を計算の面でも積極的に使う．
- Legendre 変換：微分を取って線形化 →Legendre 変換で線形化
  ••• 困ったら下限を作ってみる．
- MKL：「最適化問題」を信用しない．同じ問題を表現する方法は無数にある．

謝辞

電通大の柳内先生には画像認識に関して詳細にアドバイス頂き，感謝しています．

陰勾配法としての提案法

$$\boldsymbol{w}^{t+1} \leftarrow \underset{\boldsymbol{w}}{\operatorname{argmin}} \left(f(\boldsymbol{w}) + \frac{1}{2\eta^t} \|\boldsymbol{w} - \boldsymbol{w}^t\|_2^2 \right)$$

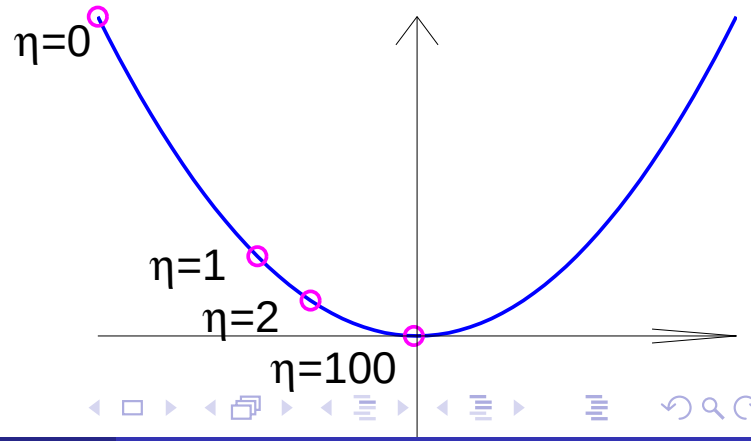
より，

$$\partial f(\boldsymbol{w}^{t+1}) + \frac{1}{\eta_t} \left(\boldsymbol{w}^{t+1} - \boldsymbol{w}^t \right) \ni 0$$

整理すると，

$$\boldsymbol{w}^{t+1} - \boldsymbol{w}^t \in -\eta_t \underbrace{\partial f(\boldsymbol{w}^{t+1})}_{\text{遷移先での勾配}}$$

$$w^{t+1} = \frac{w^t}{1 + \eta_t}$$



Convolution

Inf-convolution:

$$(f \circ g)(x) = \inf_y (f(x-y) + g(y))$$

畳み込みと Legendre 変換

$$(f \circ g)^*(\alpha) = f^*(\alpha) + g^*(\alpha)$$

$$\begin{aligned}
\because (f \circ g)^*(\alpha) &= \sup_x \left(\alpha x - \inf_y (f(x-y) + g(x)) \right) \\
&= \sup_x \sup_y (\alpha x - f(x-y) - g(y)) \\
&= f^*(\alpha) + \sup_y (\alpha y - g(y)) \\
&= f^*(\alpha) + g^*(\alpha)
\end{aligned}$$

Proximity Operation

$f(\boldsymbol{x})$ は凸で下半連続．$f^*(\boldsymbol{y}) = \sup_{\boldsymbol{x}}(\boldsymbol{x}^\top \boldsymbol{y} - f(\boldsymbol{x}))$ とする．

Proximity operator **（凸集合への射影の一般化）**

$$\mathrm{prox}_f(\boldsymbol{z}) = \inf_{\boldsymbol{x}} \left(\frac{1}{2}\|\boldsymbol{z} - \boldsymbol{x}\|_2^2 + f(\boldsymbol{x}) \right)$$

Moreau's decomposition (Moreau, 65; Combettes&Wajs, 05)

$$\mathrm{prox}_f(\boldsymbol{z}) + \mathrm{prox}_{f^*}(\boldsymbol{z}) = \boldsymbol{z}$$