スパース正則化学習の理論とアルゴリズム

冨岡 亮太
東京大学 大学院情報理工学系研究科 数理情報学専攻

概要. このサーベイ論文では近年，信号処理，情報理論，機械学習の分野をまたいで注目されているスパース性を導く様々な正則化の方法を加法的なスパース正則化と構造的なスパース正則化の観点から分類し，それぞれに対する具体的な最適化アルゴリズムを与える． 具体的には加法的なスパース正則化に対しては（加速付き）近接勾配法および著者が提案する双対拡張ラグランジュ法を紹介する．双対拡張ラグランジュ法は加法的なスパース性から生じる条件数の悪化に対して有効であることを議論する．一方，構造的なスパース正則化に対しては交互方向乗数法を紹介する．交互方向乗数法は線形演算で表現される構造とスパース正則化項を分離することが可能で，構造的なスパース正則化に対して非常に有効な手法である．

Theory and algorithms for sparse learning

Ryota Tomioka
Department of Mathematical Informatics, The University of Tokyo

Abstract. In this survey, we review various regularization techniques that induce different types of sparsity, which has attracted considerable interest recently in signal processing, information theory, and machine learning. We categorize these techniques into *additive sparse regularization* and *structural sparse regularization* and discuss optimization algorithms for the two classes. More precisely, we discuss (accelerated) proximal gradient methods and dual augmented Lagrangian (DAL) method for the former. DAL method is particularly suited for poorly conditioned problems that may arise from the additive sparsity formulation. For the latter we discuss the alternating direction method of multipliers. This method is particularly attractive because it allows to separate the structure represented by a matrix from the sparsity inducing norms.

1. はじめに

機械学習は理学，工学からビジネスまでありとあらゆるデータを解析し，そこにひそむパターンを見つけ出し，予測を行うための方法論である．

機械学習の研究でもっとも重要なのは，いかに特定の対象をモデル化するかという側面である．このサーベイではデータにひそむ様々な構造を明らかにするためのアプローチとしてのスパース推定を扱う．現実の高次元で複雑なデータは，必ずしもその生成過程は自明ではなく，それに立ち向かうには予測的なモデリングが重要である．

例えば，腫瘍に関する遺伝子発現データを考える．予測的なモデリングは腫瘍のメカニズムを直接モデル化するのではなく，腫瘍のタイプ（例えば良性／悪性）を遺伝子発現か

ら**予測**するようなモデルを考える．腫瘍から特定の遺伝子発現に至る過程は複雑であり，ひとつのデータセットから何かを言えるとは限らないが，目的を腫瘍のタイプの予測に限れば問題はずっと解きやすいものになる．さらにその際に**どの遺伝子が腫瘍のタイプの予測に重要であるか**という情報が得られれば結果として間接的に腫瘍のメカニズム関する知識を得ることができる．

このように予測的なモデリングにおいては予測問題にひそむ**構造**を明らかにするような手法が重要である．上で述べた，すべての変数を使って予測するのではなく，重要な変数のみで予測を行うということもこのような構造の一種であり，この論文で扱うスパース正則化はこのような構造を導くために用いられる手法である．このサーベイでは，単純な疎性だけでなく，グループ単位の疎性や行列の低ランク性などより一般的な構造を導くための手法を扱うためにスパース性というより緩い言葉を用いる．

機械学習のもうひとつの側面として，このようなモデルをいかに最適化問題として定式化し，解を求めるかという側面がある．このサーベイでは凸最適化に基づくスパース正則化の様々なバリエーションを扱い，それらを解くための具体的なアルゴリズムを与える．

より具体的には，この論文では機械学習で頻繁に現れる，以下の形を持つ最適化問題のためのアルゴリズムを議論する:

$$\underset{\boldsymbol{w}\in\mathbb{R}^n}{\text{minimize}} \quad L(\boldsymbol{w}) + R(\boldsymbol{w}). \tag{1.1}$$

ここで $\boldsymbol{w} \in \mathbb{R}^n$ が求めたいパラメータベクトルであり，$L(\boldsymbol{w})$ は損失関数と呼ばれ，$R(\boldsymbol{w})$ は正則化項と呼ばれる．また，損失関数 $L(\boldsymbol{w})$，正則化項 $R(\boldsymbol{w})$ ともに凸関数であるとする．ここで損失関数は与えられたデータへのあてはまりの良さを定量化する項であり，正則化項は疎性などの問題にひそむ構造を取り出すべく設計するものである．

なぜ正則化項として凸関数を考えるかというと，まず機械学習で現れる多くの損失関数は凸であるということがある．従って，凸損失関数に凸正則化項を組み合わせることによって解きやすくかつ見通しやすく様々な問題を扱うことができる．経験ベイズの枠組みから導かれる非凸な正則化項については Wipf et al. [77, 78]，劣モジュラ関数を用いた凸ではないより一般的なスパース正則化に対するアプローチに関しては Bach [3] を参照して頂きたい．

Rudin et al. [61] によると，ℓ_1 正則化あるいはスパース正則化の歴史はガリレオ (1632) やラプラス (1793) に遡るという．近年では，90 年代に信号処理や画像処理の文脈 [18, 61] で扱われて以降，統計学 [69]，情報理論 [16] で注目され，現在では機械学習を中心としてこれらの領域にまたがる一分野をなしている．2000 年以降の特徴としては以下で紹介するような様々なスパース性のバリエーションが提案されている点と言えるかもしれない．

現実におけるスパースな信号の例として，自然画像のウェーブレット変換をあげる．図 1 に 1 つの自然画像とその 2 次元ウェーブレット変換を示す．右図から 2 次元ウェーブレット係数はほとんどがゼロであることがわかる．

もしある画像 $\boldsymbol{w}$ に雑音が加わって $\boldsymbol{y} = \boldsymbol{w} + \boldsymbol{\xi}$ が観測されたとしよう．$\boldsymbol{w}$ のウェーブレッ

Fig. 1. “Cameraman” image and its wavelet transform. The right pannel shows the absolute coefficients of the wavelet transformation of the image on the left with the haar wavelet. The wavelet coefficients are mostly very close to zero, corresponding to the right panel being mostly black.

ト変換 $\boldsymbol{\Psi w}$ がスパースであるという仮定のもとで，真の画像を $\boldsymbol{w}$ を求める問題は

$$\underset{\boldsymbol{w}\in\mathbb{R}^n}{\text{minimize}} \quad \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{w}\|_2^2 \quad \text{subject to} \quad |\text{supp}(\boldsymbol{\Psi w})| \le k, \tag{1.2}$$

と表現することができる．ここで，$|\text{supp}(\cdot)|$ は非ゼロ要素の数を表す．行列 $\boldsymbol{\Psi}$ が直交行列の場合 ($\boldsymbol{\Psi}^\top\boldsymbol{\Psi} = \boldsymbol{I}$) は，変数変換 $\boldsymbol{w}' = \boldsymbol{\Psi w}$ を行うことにより，上記最適化問題は

$$\underset{\boldsymbol{w}'\in\mathbb{R}^n}{\text{minimize}} \quad \frac{1}{2}\|\boldsymbol{\Psi y}-\boldsymbol{w}'\|_2^2 \quad \text{subject to} \quad |\text{supp}(\boldsymbol{w}')| \le k$$

と等価であるので，$\boldsymbol{w}$ の非ゼロ要素数が k 個以下という制約のもとでこれを最小化するには，観測 $\boldsymbol{y}$ のウェーブレット係数 $\boldsymbol{\Psi y}$ のうちもっとも大きいものから k 個を選んでくるのが最適である．従って，最適解は

$$\hat{w}'_j = \begin{cases} w_j^+ & (|w_j| \ge \theta_k), \\ 0 & (\text{otherwise}) \end{cases} \qquad (j = 1,\dots,n)$$

と与えられる．ここで，$w_j^+ = (\boldsymbol{\Psi y})_j$ とおいた．また，θ_k は w_j^+ の絶対値で k 番目に大きいものの値である．

最適化問題 (1.2) は信号処理に限らず機械学習でも自然に現れるものである．実際，$\boldsymbol{x}$ を入力として出力 y を予測する問題を考える．ϵ_i $(i = 1,\dots,m)$ を独立同一の正規分布に従う確率変数として，m 個の入出力対 $(\boldsymbol{x}_i, y_i)$ が正規雑音モデル

$$y_i = \langle \boldsymbol{x}_i, \boldsymbol{w}\rangle + \epsilon_i \qquad (i = 1,\dots,m) \tag{1.3}$$

に従って得られているとき，2 乗損失

$$L(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{X w}\|_2^2 \tag{1.4}$$

を考えるのが自然である [10]．この場合行列 $\boldsymbol{X} \in \mathbb{R}^{m\times n}$ は入力ベクトルを行方向にならべた行列であり，一般に正則行列ではない．このような場合に最適化問題 (1.2) は組み合わせ最適化問題であり，上で見たように解析的に解くことはできない．

この論文ではこのような組み合わせ最適化問題の凸緩和として得られる正則化項を扱う．具体例として，以下の最適化問題で表される推定量は統計学では lasso [69]，信号処理では basis pursuit [18] として知られている:

$$\underset{\boldsymbol{w}\in\mathbb{R}^n}{\text{minimize}} \quad \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{X}\boldsymbol{w}\|_2^2+\lambda\|\boldsymbol{w}\|_1. \tag{1.5}$$

ここで，$\|\boldsymbol{w}\|_1$ は ℓ_1 ノルムであり，ベクトル $\boldsymbol{w}$ の係数の絶対値の線形和として定義される（2.1 節を参照）．また，信号処理の分野で最近圧縮センシング [16] というキーワードで注目されている手法は，図 1 のように信号がある基底の上でスパースであるという仮定のもとに，少ない数の線形な観測から信号を復元する問題として

$$\underset{\boldsymbol{w}\in\mathbb{R}^n}{\text{minimize}} \quad \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{\Omega}\boldsymbol{w}\|_2^2+\lambda\|\boldsymbol{\Phi}\boldsymbol{w}\|_1 \tag{1.6}$$

と定式化することができる．ここで観測の数を $m(\ll n)$ として，行列 $\boldsymbol{\Omega}\in\mathbb{R}^{m\times n}$ は信号 $\boldsymbol{w}$ に対する m 回の線形な観測に対応し，$\boldsymbol{\Phi}$ は信号 $\boldsymbol{w}$ がスパースとなる基底への変換行列である．この問題はウェーブレットの場合のように $\boldsymbol{\Phi}$ に逆行列が存在すれば $\boldsymbol{X}=\boldsymbol{\Omega}\boldsymbol{\Phi}^{-1}$ とおくことで最適化問題 (1.5) に帰着することに注意する．

機械学習の文脈でスパース性の利点を繰り返すと，スパースに表現することにより単に予測できるだけでなく，予測に対してどの特徴量が効果的であるのかを専門家に伝えることができる．また，バイオインフォマティクスなどで現れるサンプル数 m が特徴量の次元 n より少ない場合，訓練データは m 次元の部分空間に散布しているため，サンプル数 m より多くの特徴量を用いることは直感的にもあまり効果的ではない [34].

スパース推定の理論に関しては日本語では田中 [84] や英語では Bühlmann & van de Geer [13], Negahban et al. [51], Bickel et al. [9] などの優れたレビューや論文があるので，本サーベイではこのような推定を実現するための最適化アルゴリズムを扱う．以下 2 節で様々なスパース性を導く正則化項を紹介する．このとき加法的なスパース性と構造的なスパース性に分類し，3 節でそれら両者のための最適化アルゴリズムを議論する．

2. 様々なスパースモデル

この節では様々なスパース性に対応する正則化の方法を概観する．はじめに 2.1 節で 3 種類の基本的なスパース正則化項について述べたあと，それらの組み合わせで得られる様々なスパース正則化項を議論する．ここで組み合わせ方には大きく分類して 2 通りの方法があり，ここではそれを加法的なスパース正則化（2.2 節）と構造的なスパース正則化（2.3 節）と呼ぶ.

基本的なスパース正則化は，何らかの意味で分解可能 (separable) な構造を有しており，扱い易いため，このように呼んでいる．代表的なものが ℓ_1 ノルムに基づく正則化項

$$R(\boldsymbol{w})=\lambda\|\boldsymbol{w}\|_1$$

である．これはベクトルの成分ごとに分解する構造を有している．異なる基本的なスパース正則化項を用いると様々なスパース性を導くことができる．

加法的なスパース正則化および構造的なスパース正則化は，線形変換 $\boldsymbol{\Phi}$ と，基本的なスパース正則化項 $\|\cdot\|_\star$ によって定義される．

加法的なスパース正則化は，信号 $\boldsymbol{w}$ がスパースなベクトル $\boldsymbol{z}$ から，適当な線形変換 $\boldsymbol{\Phi}$ を用いて，$\boldsymbol{w} = \boldsymbol{\Phi}^\top \boldsymbol{z}$ のように得られている場合を指す．言い換えればスパースなものの**和で予測するモデル**が加法的なスパース正則化である．具体的には，

$$\underset{\boldsymbol{z}\in\mathbb{R}^p}{\text{minimize}} \quad L(\boldsymbol{\Phi}^\top \boldsymbol{z}) + \lambda\|\boldsymbol{z}\|_\star \tag{2.1}$$

のような場合である．ここで，一般の基本的なスパース正則化を表すために $\star$ のついたノルムを用いる．このとき，例えば lasso の最適化問題 (1.5) は

$$\underset{\boldsymbol{z}\in\mathbb{R}^p}{\text{minimize}} \quad \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\Phi}^\top \boldsymbol{z}\|_2^2 + \lambda\|\boldsymbol{z}\|_1$$

と書き換えられるので，$\boldsymbol{X}' = \boldsymbol{X}\boldsymbol{\Phi}^\top$ と定義すれば良いだけのように考えられるかもしれない．ただし機械学習の文脈では $\boldsymbol{X}$ と $\boldsymbol{\Phi}$ を分けておくメリットとして，以下の 2 点をあげる．1 つ目は，$\boldsymbol{X}$ は入力データから決まるものである一方，$\boldsymbol{\Phi}$ はモデルに期待する構造に基いて設計するものであるため，$\boldsymbol{\Phi}$ に関する仮定と $\boldsymbol{X}$ に関する仮定は異なる意味を持つという点である．2 つ目は，$\boldsymbol{\Phi}$ はしばしば（スパースであるなどの）構造を持っているので，その構造を利用することで行列-ベクトル積 $\boldsymbol{\Phi}^\top \boldsymbol{z}$ を効率的に計算できるという点である．

構造的なスパース正則化は，圧縮センシング (1.6) のように，信号そのものはスパースではなく，その線形変換 $\boldsymbol{\Phi}\boldsymbol{w}$ がスパースであるという場合を指す．具体的には

$$\underset{\boldsymbol{z}\in\mathbb{R}^p}{\text{minimize}} \quad L(\boldsymbol{w}) + \lambda\|\boldsymbol{\Phi}\boldsymbol{w}\|_\star \tag{2.2}$$

のような場合である．

なお、$\boldsymbol{\Phi} = \boldsymbol{I}$（単位行列）とすると，加法的なスパース正則化 (2.1)，構造的なスパース正則化 (2.2) のどちらも基本的なスパース正則化に帰着する点に注意する．

2.1 基本的なスパース正則化

疎性を導く正則化項のなかで代表的なものに ℓ_1 正則化がある．ℓ_1 正則化項は回帰ベクトル $\boldsymbol{w}$ の係数の絶対値の線形和として

$$\|\boldsymbol{w}\|_\star = \|\boldsymbol{w}\|_1 = \sum_{j=1}^{n} |w_j| \tag{2.3}$$

と定義される．ℓ_1 正則化項と最適化問題 (1.2) で考えた非ゼロ要素数の間の関係を図 2 に示す．1 次元に $|x|^p$ の概形を示している．この場合，非ゼロ要素の個数は $x = 0$ のとき

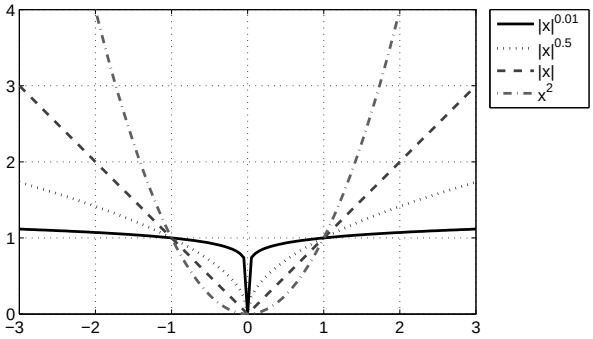


Fig. 2. Cardinality as a limit of ℓ_p^p pseudo norms. ℓ_1 norm is the tightest convex lower bound of the cardinality function.

ゼロでそれ以外の場合に 1 を取る関数であり，$p \to 0$ の極限に対応する．$|x|$ は凸関数の中で最もタイトな近似になっていることがわかる．実際，ℓ_1 ノルムは非ゼロ要素の個数に対する ℓ_∞ ノルムに関して最もタイトな凸緩和になっている．これは劣モジュラ関数の文脈では，ℓ_1 ノルム $\|\boldsymbol{w}\|_1$ は非ゼロ要素数 $|\mathrm{supp}(\boldsymbol{w})|$ の Lovász 拡張であることに対応する [3, Prop. 2.6].

ℓ_1 正則化を用いるとなぜスパースな（疎な）解が得られるかを直感的に理解するために，最適化問題 (1.5) において $\boldsymbol{X} = \boldsymbol{I}$ の場合を考える．この最小化は prox 作用素 (proximal operator) として知られており，定義としては凸関数 R に関する prox 作用素 [50, 58] は

$$\mathrm{prox}_R(\boldsymbol{z}) = \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(\frac{1}{2}\|\boldsymbol{z}-\boldsymbol{w}\|_2^2 + R(\boldsymbol{w}) \right) \tag{2.4}$$

と与えられる．このサーベイで扱ういくつかの正則化項に関する prox 作用素を表 1 に示す．

具体的には ℓ_1 正則化項 (2.3) に関する prox 作用素は，n 変数に関する最小化問題 (2.4) が

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

のように n 個の 1 次元の最小化問題に分解されることから，解析的に書くことができて

$$\mathrm{prox}_\lambda^{\ell_1}(z_j) = \begin{cases} z_j + \lambda & (z_j < -\lambda), \\ 0 & (-\lambda \le y_j \le \lambda), \\ z_j - \lambda & (z_j > \lambda) \end{cases} \qquad (j = 1, \ldots, n) \tag{2.5}$$

と与えられる．この関数の概形を図 3 に示す．信号処理の分野ではこの関数はソフト閾値関数 (soft-threshold function) と呼ばれている [21, 23, 28]．この関数は入力 y_j が絶対値で λ 以下の場合はゼロに打ち切り，絶対値が λ 以上の場合は原点方向に λ だけ縮小する．このように有限の正則化定数 λ で一部の成分がゼロとなる点が ℓ_1 およびこの論文でサーベイするより一般的なスパース正則化の特徴である．

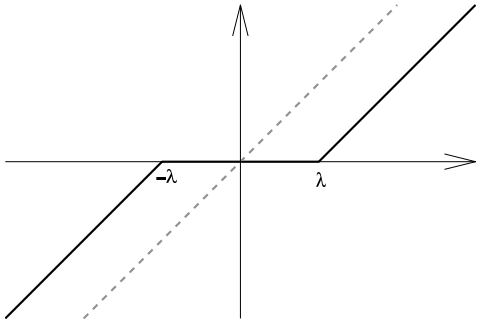


Fig. 3. Soft-threshold function.

以上では直感的に理解するために $\boldsymbol{X}$ が単位行列の場合に限定して議論したが，$\boldsymbol{X}$ が単位行列でない場合も同様である．劣微分を考えることにより，最小化問題 (1.5) の最適解 $\hat{\boldsymbol{w}}$ は

$$\hat{\boldsymbol{w}} = \mathrm{prox}_{\lambda}^{\ell_1}\left(\hat{\boldsymbol{w}} - g(\hat{\boldsymbol{w}})\right)$$

という方程式を満たすことを示すことができる [20]（この式が 3.1.1 項で述べる近接勾配法のひとつの導出方法になっている）．ここで $\mathrm{prox}_{\lambda}^{\ell_1}$ は図 3 に示すソフト閾値関数 (2.5) であり，$g(\boldsymbol{w}) = \boldsymbol{X}^{\top}(\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y})$ とおいた．

具体的にいくつの成分がゼロとなるかは正則化定数 λ に依存する．もちろん λ を十分大きく取ればすべての係数がゼロとなるが，回帰問題 (1.3) に対する当てはまりは悪くなる．

グループ ℓ_1 正則化

グループ ℓ_1 ノルム [5,46,82] は

$$\|\boldsymbol{w}\|_{\star} = \sum_{\mathfrak{g}\in\mathfrak{G}} \|\boldsymbol{w}_{\mathfrak{g}}\|_p. \tag{2.6}$$

のように定義される．ここで，$\mathfrak{G}$ はインデックス集合 $\{1,\dots,n\}$ の 1 つの分割（$\cup_{\mathfrak{g}\in\mathfrak{G}}\mathfrak{g} = \{1,\dots,n\}$ であり，変数のグループの間に重複はない，すなわち任意の $\mathfrak{g} \neq \mathfrak{g}' \in \mathfrak{G}$ に関して $\mathfrak{g} \cap \mathfrak{g}' = \emptyset$ であるとする．また，$\|\cdot\|_p$ は ℓ_p ノルムで $p = 2$ あるいは ∞ が多く使われる．$p = 1$ の場合は ℓ_1 ノルム (2.3) に帰着する．

このような正則化項のもっとも簡単な例としては行列の行あるいは列単位でスパースにしたいという場合がある．例えばマルチタスク学習や多入力多出力の問題においては，r 個の解くべき回帰問題に対する回帰ベクトル $\boldsymbol{w}_1,\dots,\boldsymbol{w}_r$ を並べて得られる行列 $\boldsymbol{W}$ のひとつの行をひとつのグループだと考えると，グループ ℓ_1 正則化を用いることによって，すべてのタスクに共通して有用な変数を抽出することができる [54].

グループ ℓ_1 正則化項 (2.6) に関する prox 作用素 (2.4) は

$$\lambda \sum_{\mathfrak{g}\in\mathfrak{G}} \|\boldsymbol{w}\|_p + \frac{1}{2}\|\boldsymbol{z} - \boldsymbol{w}\|_2^2 = \sum_{\mathfrak{g}\in\mathfrak{G}} \left(\lambda\|\boldsymbol{w}\|_p + \frac{1}{2}\|\boldsymbol{z}_{\mathfrak{g}} - \boldsymbol{w}_{\mathfrak{g}}\|_2^2\right)$$

のように分解できることから，変数のグループ $\mathfrak{g}$ ごとにゼロとなるか非ゼロとなるかが決まることがわかる．さらに，上の式の劣微分を取ることにより $\boldsymbol{w}_\mathfrak{g}$ がゼロとなるのは，$q = \frac{p}{p-1}$ として，$\boldsymbol{z}_\mathfrak{g}$ が原点を中心とする半径 λ の ℓ_q ノルム球に入っている場合であることがわかる．非ゼロとなる場合の具体的な形は p の値に依存する．表 1 に $p = 2$ の場合の具体的な式を示す．

なお，上では簡単のために正則化定数 λ をすべてのグループで共通としたが，例えば大きさの異なるグループが混在する場合は，グループごとに正則化の強さを変えて

$$\|\boldsymbol{w}\|_\star = \sum_{\mathfrak{g}\in\mathfrak{G}} \gamma_\mathfrak{g} \|\boldsymbol{w}_\mathfrak{g}\|_p$$

とすることもある．例えば Yuan & Lin [82] では $\gamma_\mathfrak{g} = \sqrt{|\mathfrak{g}|}$ とし，大きなグループほど正則化が強くなるようにしている．

トレースノルム正則化

トレースノルム [26, 37, 66] は行列に対して定義されるノルムで，$\boldsymbol{W}$ を行列として，正則化項としては

$$\|\boldsymbol{W}\|_\star = \|\boldsymbol{W}\|_{\mathrm{tr}} = \sum_{j=1}^{r} \sigma_j(\boldsymbol{W}) \tag{2.7}$$

と定義される．ここで r は行列 $\boldsymbol{W}$ のランクであり，$\sigma_j(\boldsymbol{W})$ は行列 $\boldsymbol{W}$ の j 番目に大きい特異値を表す．トレースノルムは nuclear norm [40] あるいは Ky-Fan r ノルム [37] という名前でも知られている（ [15, 81] も参照）．

トレースノルムは様々な面で ℓ_1 ノルムの行列への拡張ととらえることができる．例えばトレースノルムは行列のスペクトル（特異値の全体）に関する ℓ_1 ノルムであり，トレースノルムの双対ノルム[*1]は作用素ノルム（行列の最大特異値）である．また，トレースノルムは行列のランクに対する作用素ノルムに関して最もタイトな凸緩和である [57].

トレースノルム最小化をシステム同定の文脈で低ランク行列の推定に用いたのは Fazel et al. [26] の研究であるが，機械学習で有名なのは Srebro et al. [65] の協調フィルタリングへの応用である．協調フィルタリングはおもに商品の推薦などの文脈で，部分的に観測された商品 × ユーザーの行列から未観測の要素を補完するという問題で，問題の背後にある真の行列が低ランクであるという仮定がしばしば用いられる．この低ランク性を制約として陽に扱うのではなく，トレースノルムを最小化するのがこれらの研究のアイディアである．

[*1] ノルム $\|\boldsymbol{x}\|$ の双対ノルム $\|\boldsymbol{y}\|_*$ は $\|\boldsymbol{y}\|_* = \sup_{\boldsymbol{x}} \langle \boldsymbol{x}, \boldsymbol{y} \rangle$ s.t. $\|\boldsymbol{x}\| \le 1$ と定義される．例えば，ℓ_1 ノルムの双対ノルムは ℓ_∞ ノルムであり，行列のスペクトルに関する ℓ_1 ノルムがトレースノルム，スペクトルに関する ℓ_∞ ノルムが作用素ノルムである．

Table 1. 基本的な正則化関数とその prox 作用素．ここで，λ は正則化定数であり，トレースノルム正則化（3行目）において行列 $\boldsymbol{Y}$ の特異値分解を $\boldsymbol{Y} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top$ とした．

名前	正則化関数	prox 作用素
ℓ_1 正則化	$\lambda \sum_{j=1}^n \lvert w_j \rvert$	$\left(\max(\lvert y_j \rvert - \lambda, 0)\dfrac{y_j}{\lvert y_j \rvert}\right)_{j=1}^n$
グループ ℓ_1 正則化	$\lambda \sum_{\mathfrak{g}\in\mathfrak{G}} \lVert \boldsymbol{w}_\mathfrak{g} \rVert_2$	$\left(\max(\lVert \boldsymbol{y}_\mathfrak{g} \rVert_2 - \lambda, 0)\dfrac{\boldsymbol{y}_\mathfrak{g}}{\lVert \boldsymbol{y}_\mathfrak{g} \rVert_2}\right)_{\mathfrak{g}\in\mathfrak{G}}$
トレースノルム正則化	$\lambda \sum_{j=1}^r \sigma_j(\boldsymbol{W})$	$\boldsymbol{U}\max(\boldsymbol{S} - \lambda, 0)\boldsymbol{V}^\top$

トレースノルムに関する prox 作用素 (2.4) は与えられた行列 $\boldsymbol{Y}$ の特異値分解 $\boldsymbol{Y} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top$ を用いて

$$\operatorname{prox}_\lambda^{\mathrm{tr}}(\boldsymbol{Y}) = \boldsymbol{U}\max(\boldsymbol{S} - \lambda, 0)\boldsymbol{V}^\top \tag{2.8}$$

と与えられる．すなわち，特異値の中で λ より大きいものは原点方向に λ 縮小し，それ以下のものはゼロで打ち切るという操作であり，特異値に対するソフト閾値処理 (2.5) と見ることができる．

トレースノルムは上で述べた ℓ_1 ノルムおよびグループ ℓ_1 ノルムを特殊な場合として含んでいる．例えば，ベクトル $\boldsymbol{w}$ の要素を対角に並べた行列を考えると，そのトレースノルムはベクトル $\boldsymbol{w}$ の ℓ_1 ノルムに一致する．また，グループ ℓ_1 ノルムの場合はグループごとに対角にブロック対角に並べた行列のトレースノルムはグループ ℓ_1 ノルムに等しい．ただし，このように考えることは正則化の間の関係を理解するためには有用だが，最適化の上ではこのような変換を行う利点はない．

2.2 加法的なスパース正則化

Jacob et al. [38] はマイクロアレイデータの解析において関連する遺伝子のグループを同時に選択するような正則化を考えた．ここで，関連する遺伝子の集合は重複があるので，単純にグループ ℓ_1 正則化を行うことはできない点に注意する．彼らは遺伝子の集合 $\mathfrak{g} \in \mathfrak{G}$（重複を許す）をグループとし，グループの大きさ $|\mathfrak{g}|$ 次元ベクトル $\boldsymbol{z}_\mathfrak{g}$ を考え，その重ねあわせ $\boldsymbol{w} = \sum_{\mathfrak{g}\in\mathfrak{G}} \boldsymbol{I}_\mathfrak{g}\boldsymbol{z}_\mathfrak{g}$ で回帰ベクトル $\boldsymbol{w}$ を表現することを考えた．ここで，$\boldsymbol{I}_\mathfrak{g}$ は $n \times |\mathfrak{g}|$ 行列で，グループ $\mathfrak{g}$ に対応する行の部分に単位行列を持ちそれ以外はゼロであるとする．すなわち式 (2.1) で言えば，$\boldsymbol{z}$ は $\boldsymbol{z}_\mathfrak{g}$ を並べたものであり，$\boldsymbol{\Phi}^\top$ は $\boldsymbol{I}_\mathfrak{g}$ を横に並べたものである．また，$\star$ ノルムはグループ ℓ_1 ノルム (2.6) である．最適化問題としては

$$\underset{(\boldsymbol{z}_\mathfrak{g})_{\mathfrak{g}\in\mathfrak{G}}}{\text{minimize}} \quad L\left(\sum\nolimits_{\mathfrak{g}\in\mathfrak{G}} \boldsymbol{I}_\mathfrak{g}\boldsymbol{z}_\mathfrak{g}\right) + \lambda \sum_{\mathfrak{g}\in\mathfrak{G}} \lVert \boldsymbol{z}_\mathfrak{g} \rVert \tag{2.9}$$

と与えられる．スパース性はグループ（遺伝子の集合）単位で考えている点に注意する．また，グループが重複しているため，ある変数がゼロになるのはその変数を含むすべての

グループがゼロになる場合であることに注意する．グループは，例えば，遺伝子の作用する経路がわかっている場合にそれを用いて定義することができる．

Jalali et al. [39] はマルチタスク学習において，すべてのタスクで共通に使われる変数とタスク固有の変数を抽出するために，タスク固有係数 $\boldsymbol{z}^{(1)}$ と，タスク共通係数 $\boldsymbol{z}^{(2)}$ の和で予測するモデルを考え，タスク固有係数に対しては ℓ_1 正則化項を，タスク共通係数に関してはグループ ℓ_1 正則化項を用いた．この場合，$\boldsymbol{z} = [\boldsymbol{z}^{(1)\top}, \boldsymbol{z}^{(2)\top}]^\top$，$\boldsymbol{\Phi}^\top = [\boldsymbol{I}_n, \boldsymbol{I}_n]$ と定義される．また，$\star$ ノルムは $2n$ 次元のベクトルに対するノルムで最初の n 係数については ℓ_1 正則化 (2.3)，次の n 変数に対しては異なるタスクで共通の変数を選択するようなグループ ℓ_1 正則化 (2.6) である．

マルチカーネル学習 [4, 45, 49] も加法的なスパース学習の一例である．カーネル法 [63] は，主に判別や回帰などの教師付き学習において非線形な回帰関数を学習するための枠組みだが，どのようなカーネル関数を用いるか，あるいは等価にどのような再生核ヒルベルト空間を考えるか，ということが常に問題となる．マルチカーネル学習は基底カーネル関数 $k_l(x, x')$ $(l = 1, \dots, L)$ が与えられたもとで，それらをカーネル重み d_l を用いてを線形結合したカーネル関数 $\bar{k}(x, x') = \sum_{l=1}^L d_l k_l(x, x')$ を用いてカーネル関数と予測器を同時に学習する問題として定式化することができる．

このとき，$\bar{\mathcal{H}}$ をカーネル関数 $\bar{k}$ から定まる再生核ヒルベルト空間とし，$\|\bar{f}\|_{\mathcal{H}_{\bar{k}}}$ を $\mathcal{H}_{\bar{k}}$ のノルムとすると，λ および，μ_l $(l = 1, \dots, L)$ を正則化定数として，最適化問題

$$\underset{(d_l)_{l=1}^L, \bar{f} \in \mathcal{H}_{\bar{k}}}{\text{minimize}} \quad \sum_{i=1}^m \ell(y_i, \bar{f}(x_i)) + \frac{\lambda}{2}\left(\|\bar{f}\|_{\mathcal{H}_{\bar{k}}}^2 + \sum\nolimits_{l=1}^L \mu_l d_l\right)$$

が最適化問題

$$\underset{(f_l \in \mathcal{H}_{k_l})_{l=1}^L}{\text{minimize}} \quad \sum_{i=1}^m \ell\left(y_i, \sum\nolimits_{l=1}^L f_l(x_i)\right) + \lambda \sum_{l=1}^L \sqrt{\mu_l}\, \|f_l\|_{\mathcal{H}_{k_l}}$$

と等価であるという事実 [2] から，マルチカーネル学習も加法的なスパース正則化の一種であることがわかる．ただしここで $\|f_l\|_{\mathcal{H}_{k_l}}$ は基底カーネル関数 k_l に対応する再生核ヒルベルト空間 $\mathcal{H}_{k_l}$ のノルムを表す．詳細は Tomioka & Suzuki [73] を参照して頂きたい．

より最近の話題としてはロバスト主成分分析 [14] も加法的なスパース正則化の一種である．通常の主成分分析は与えられたデータ行列 $\boldsymbol{Y}$（ここで $\boldsymbol{Y}$ は中心化されていて，$\boldsymbol{Y}^\top \boldsymbol{Y}/m$ が共分散行列であると仮定する）およびランク k に対して，最適化問題

$$\underset{\boldsymbol{W} \in \mathbb{R}^{m \times n}}{\text{minimize}} \|\boldsymbol{Y} - \boldsymbol{W}\|_F^2 \quad \text{s.t.} \quad \text{rank}(\boldsymbol{W}) \le k$$

の解として与えられ，具体的には $\boldsymbol{Y}$ の特異値分解の上位 k 番目までの特異値および特異ベクトルを計算することで得られる（ウェーブレット雑音除去 (1.2) の行列版と考えることができる）．ロバスト主成分分析はこの問題の誤差関数を ℓ_1 損失に変えることで外れ値

に対して頑健になるようにしたもので，最適化問題

$$\underset{\boldsymbol{L},\boldsymbol{S}\in\mathbb{R}^{m\times n}}{\text{minimize}} \quad \lambda\|\boldsymbol{L}\|_{\text{tr}} + \|\boldsymbol{S}\|_{\ell_1} \quad \text{s.t.} \quad \boldsymbol{Y} = \boldsymbol{L} + \boldsymbol{S}$$

の解として与えられる．ここで，行列 $\boldsymbol{L}$ は低ランク部分，$\boldsymbol{S}$ はスパースなノイズに対応し，$\|\cdot\|_{\text{tr}}$ はトレースノルム (2.7)，$\|\cdot\|_{\ell_1}$ は行列に対する要素ごとの ℓ_1 ノルムを表す．ここで，ランク制約のもとで ℓ_1 損失を最小化することは困難なので，ランク制約がトレースノルム正則化に置き換わっていることに注意する．

ここでもやはり，上の例と同様に，低ランク行列 $\boldsymbol{L}$ とスパース行列 $\boldsymbol{S}$ の**和で予測する**形になっていることに注意する．

Chandrasekaran et al. [17] は加法的なスパース性の観点から多くのスパース正則化の手法に対する統一的な理論を展開している．この論文ではスパース性を考える上で基本となる要素からなる集合「アトミック集合」$\mathcal{A}$ をまずはじめに定義する．その上で，アトミック集合の元の非負結合

$$\boldsymbol{w} = \sum_{a\in\mathcal{A}} c_a \boldsymbol{a} \quad (c_a \geq 0)$$

で書くことができるベクトル $\boldsymbol{w}$ に対するアトミックノルム $\|\boldsymbol{w}\|_{\mathcal{A}}$ を

$$\|\boldsymbol{w}\|_{\mathcal{A}} = \min_{(c_a)_{a\in\mathcal{A}}} \sum_{a\in\mathcal{A}} c_a \quad \text{s.t.} \quad \boldsymbol{w} = \sum_{a\in\mathcal{A}} c_a \boldsymbol{a},$$

のように定義する．例えばアトミック集合 $\mathcal{A}$ を座標系の標準基底をなす $\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_n$ とその符号反転の $2n$ 個のベクトルからなる集合とすると，アトミックノルムは ℓ_1 ノルムに一致する．また，$\mathcal{A}$ をフロベニウスノルムが 1 のランク 1 行列の集合とすると，アトミックノルムはトレースノルムと一致する．このようにアトミック集合は有限集合であってもよいし，無限集合であってもよい．

その他の加法的なスパース正則化の例として，Argyriou et al. [1] らの k-サポート・ノルムがある．

2.3 構造的なスパース正則化

構造的なスパース正則化の代表例として，重複のあるグループ ℓ_1 正則化 [41] を挙げる．最適化問題としては

$$\underset{\boldsymbol{w}\in\mathbb{R}^n}{\text{minimize}} \quad L(\boldsymbol{w}) + \lambda \sum_{\mathfrak{g}\in\mathfrak{G}} \|\boldsymbol{w}_{\mathfrak{g}}\|_2 \tag{2.10}$$

のように与えられる．ただし，$\mathfrak{g} \subseteq \{1, \ldots, n\}$ はインデックスの部分集合であり，$\boldsymbol{w}_{\mathfrak{g}}$ は $\mathfrak{g}$ で指定されるベクトル $\mathfrak{g}$ の要素からなる $|\mathfrak{g}|$ 次元ベクトルである．(2.6) とは異なり，$\mathfrak{g}$ は互いに重複することができる．

構造的なスパース正則化 (2.2) の一例として理解するには，前節の加法的な重複つきグループ ℓ_1 正則化と同様に，$|\mathfrak{g}| \times n$ 行列 $\boldsymbol{I}_{\mathfrak{g}}{}^{\top}$ を行方向にならべてに $\boldsymbol{\Phi}$ を定義し，$\star$ ノルムはグループ ℓ_1 正則化 (2.6) とすればよい．

これは 2.2 節で紹介した Jacob et al. [38] の最適化問題 (2.9) と一見類似しているが，異なる点はグループごとにスパースなベクトルの和で予測するのではなく，1 つのベクトル $\boldsymbol{w}$ が複数のグループに関して同時にスパースとなるように正則化している点である．すなわち最適化問題 (2.10) で回帰ベクトル $\boldsymbol{w}$ の j 番目の要素が複数のグループに属している場合，w_j が非ゼロとなるのは j を含むすべてのグループが非ゼロになる場合であり，ひとつでもゼロとなるグループがある場合には w_j はゼロとなる．一方，最適化問題 (2.9) では j 番目の要素を含む任意の $\boldsymbol{w}_{\mathfrak{g}}$ が非ゼロとなれば，$\sum_{\mathfrak{g}\in\mathfrak{G}} \boldsymbol{w}_{\mathfrak{g}}$ の j 番目の要素は非ゼロとなる．

また，信号処理においてよく知られているスパースモデルに信号そのものではなく差分がスパースというものがある．その代表例は fused lasso [70] と total variation [61] である．

Fused lasso [70] は特徴量の間に 1 次元の順序関係がある場合に，各特徴量に対する係数がスパースとなるような正則化項と，隣り合う特徴量に対する係数がなるべく変化しないようにする正則化項を用いて，以下のように定式化される:

$$\underset{\boldsymbol{w}\in\mathbb{R}^n}{\text{minimize}} \quad L(\boldsymbol{w}) + \lambda_1 \|\boldsymbol{w}\|_1 + \lambda_2 \sum_{j=1}^{n-1} \left| w_{j+1} - w_j \right|. \tag{2.11}$$

ここで上の最適化問題で $\lambda_1 = 0$ の場合は，変数変換を行って差分 $w_{j+1} - w_j$ を新たな変数として取れば，2.1 項の単純スパース正則化問題に帰着することに注意する．

構造的なスパース正則化 (2.2) の一例としてみるには，$2n-1 \times n$ 行列 $\boldsymbol{\Phi}$ を以下のように定義する:

$$\boldsymbol{\Phi} = \begin{bmatrix} 1 & 0 & \dots & \dots & 0 \\ 0 & 1 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 1 & 0 \\ 0 & \dots & \dots & 0 & 1 \\ -1 & 1 & 0 & \dots & 0 \\ 0 & -1 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & -1 & 1 \end{bmatrix}.$$

また，$\star$ ノルムは $2n-1$ 次元ベクトルに対する ℓ_1 ノルムである．

Total variation [61] は上記の差分に基づく正則化項の 2 次元版と考えられ（ただしより以前に提案されている），画像修復や圧縮センシングで有効な手法である．具体的には，2

次元座標 (x, y) で指定される画像 $w(x, y)$ に対して最適化問題

$$\underset{\boldsymbol{w}}{\text{minimize}} \quad \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{w}\|_2^2 + \lambda_1\|\boldsymbol{\Psi}\boldsymbol{w}\|_1 + \lambda_2 \sum_{x,y} \|\nabla w(x, y)\|_2 \tag{2.12}$$

を考える．ここで第 1 項が観測との誤差に対応する損失項，第 2 項がウェーブレット基底に基づく正則化項（(1.2) 参照），第 3 項が total variation に基づく正則化項である．

∇ は離散微分演算子であり，total variation の直感的な意味は画像 I の微分ベクトル場 $\nabla I(x, y)$ がベクトル場としてスパースになるように正則化を行っているということになる．このような正則化は MRI や CT などの医療画像のように平坦な領域が多い画像には適していると言える．なお，上記定義式で ℓ_2 ノルムの代わりに ℓ_1 ノルムを用いると，x 方向微分と y 方向微分を別々に扱うことになるが，その場合，x 方向微分と y 方向微分が独立にゼロ／非ゼロの値を取るので，画像の回転に対する不変性がなくなり，軸に平行なエッジのアーティファクトが生じる可能性がある．

構造的なスパース正則化 (2.2) の一例として見るには $\boldsymbol{\Phi}$ を上半分にウェーブレット変換，下半分に 2 次元 1 階差分演算子を持つ行列と考え，$\star$ ノルムは最初の n 変数に対しては ℓ_1 正則化，残りの $2n$ 変数に対してはグループ ℓ_1 正則化と定義すればよい．このように，一般に $\boldsymbol{\Phi}$ によって変換された変数の次元は正則化の重複を反映してもとの変数の次元より大きくなることに注意する（つまり $\boldsymbol{\Phi}$ は縦長の行列となることが多い）．

最後に，低ランクテンソル補完への構造的なスパース正則化に基づくアプローチについて述べる．テンソル（あるいは多次元配列）は行列の多次元版と考えることができ，計量心理学 (psychometrics)，計量化学 (chemometrics)，信号処理などの分野をはじめとして多くの分野で重要なデータの形式である [44].

テンソルのランクを定義することは行列の場合と異なり必ずしも容易ではないが，ここではモード k ランクというランクの定義を考える．テンソルのモード k ランクはテンソルに対してある軸方向に**行列化**した際の行列としてのランクである．より具体的には $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times \cdots n_K}$ を K 階のテンソルとする．$N = \prod_{k=1}^{K} n_k$ をその要素数とする．テンソル $\boldsymbol{\mathcal{X}}$ のモード k 行列化 $\boldsymbol{X}_{(k)}$ を $\boldsymbol{\mathcal{X}}$ のモード k ファイバー，すなわち n_k 次元ベクトル $(X_{i_1 i_2 \cdots i_k \cdots i_n})_{i_k=1}^{n_k} \in \mathbb{R}^{n_k}$ $(i_1 = 1, \ldots, n_1, \ldots, i_K = 1, \ldots, n_K)$ を列方向に並べた $n_k \times N/n_k$ 行列とする．この時，$\boldsymbol{\mathcal{X}}$ のモード k ランクはモード k 行列化 $\boldsymbol{X}_{(k)}$ の行列としてのランクに等しい．言い換えればモード k ランクはモード k ファイバーの張る空間の次元に等しい．

上記定義から，M 個の要素が観測されたもとで，テンソルのモード k ランクを同時に最小化する最適化問題が

$$\underset{\boldsymbol{\mathcal{W}} \in \mathbb{R}^{n_1 \times \cdots \times n_K}}{\text{minimize}} \quad \frac{1}{2}\|\boldsymbol{y} - \Omega(\boldsymbol{\mathcal{W}})\|_2^2 + \lambda \sum_{k=1}^{K} \|\boldsymbol{W}_{(k)}\|_{\text{tr}} \tag{2.13}$$

と与えられる [31, 48, 64, 72]．ここで，$\boldsymbol{y} \in \mathbb{R}^M$ は観測値で，$\Omega : \mathbb{R}^{n_1 \times \cdots \times n_K} \to \mathbb{R}^M$ は観測された M 個の要素を取り出す作用素である．また $\|\cdot\|_{\text{tr}}$ は行列に対するトレースノルム

(2.7) であり，すべてのモード k に対し，モード k 行列化が**同時に**低ランクとなるように正則化していることになる．ここで，モード k 行列化の操作が要素の並べ替えという線形操作であることに注意すると，構造的なスパース正則化 (2.2) の一種と言える（詳しくは 3.2 節を参照）．

3. アルゴリズム

前節で見てきたように，凸最適化に基づくスパース推定は，基本的なスパース正則化関数および，それを加法的／構造的に組み合わせたものに分類することができる．

基本的なスパース正則化および，加法的なスパース正則化は一般に以下の最適化問題として定式することができる:

$$\underset{\boldsymbol{w}\in\mathbb{R}^n}{\text{minimize}} \quad \underbrace{f_\ell(\boldsymbol{A}\boldsymbol{w})}_{L(\boldsymbol{w})} + \lambda\|\boldsymbol{w}\|_\star. \tag{3.1}$$

ここで，$\boldsymbol{A}\in\mathbb{R}^{m\times n}$ はデザイン行列，f_ℓ は損失関数であり，例えば f_ℓ として，2 乗誤差

$$f_\ell(\boldsymbol{z}) = \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{z}\|_2^2$$

を考え，$\boldsymbol{A}=\boldsymbol{X}$ とすると，最適化問題 (3.1) は lasso(1.5) であり．また，y_i を +1 あるいは-1 の 2 値を取る出力変数として，ロジスティック損失

$$f_\ell(\boldsymbol{z}) = \sum_{i=1}^{m}\log(1+\exp(-z_i))$$

を考え，$\boldsymbol{A}=\mathrm{diag}(\boldsymbol{y})\boldsymbol{X}$ とすると，ℓ_1 正則化付きロジスティック回帰を考えていることになる．

デザイン行列 $\boldsymbol{A}$ を損失関数 f_ℓ と区別して書いたのは，機械学習において一般にデザイン行列は入力データから構成されており，デザイン行列の性質に仮定を置くことが難しいためである．また，加法的なスパース正則化を最適化問題 (3.1) の形に定式化するには，デザイン行列 $\boldsymbol{A}$ の列を重複させるなどの操作が必要になり，デザイン行列の性質（条件数など）はかなり悪い可能性がある，ということを意識するためにも (3.1) のようにデザイン行列を分離しておくことは有用である．ただし，以下で，分離することを陽に利用しない場合には煩雑なのでまとめて $L(\boldsymbol{w})=f_\ell(\boldsymbol{A}\boldsymbol{w})$ と書く．

一方，構造的なスパース正則化は一般に以下の最適化問題として定式化することができる:

$$\underset{\boldsymbol{w}\in\mathbb{R}^n}{\text{minimize}} \quad \underbrace{\frac{1}{\lambda}f_\ell(\boldsymbol{A}\boldsymbol{w})}_{L(\boldsymbol{w})} + \|\boldsymbol{\Phi}\boldsymbol{w}\|_\star. \tag{3.2}$$

ここで行列 $\boldsymbol{\Phi}$ は fused lasso(2.11) や total variation(2.12) であれば離散微分演算子であり，重複のあるグループ ℓ_1 正則化 (2.10) や低ランクテンソル補完 (2.13) であれば，適切に要素を並び替える置換演算子を複数並べたものになる．また，正則化定数 λ を正則化項ではなく，損失項の前に $1/\lambda$ のように付けたのはこの節で述べる交互方向乗数法 (alternating direction method of multipliers) を用いると，2 乗損失の場合，$\lambda \to 0$ の極限である制約付き最小化問題

$$\begin{aligned} &\underset{\boldsymbol{w}\in\mathbb{R}^n}{\text{minimize}} && \|\boldsymbol{\Phi}\boldsymbol{w}\|_\star, \\ &\text{subject to} && \boldsymbol{A}\boldsymbol{w} = \boldsymbol{y} \end{aligned} \tag{3.3}$$

を同一の枠組みで扱うことができて便利であるからである．

3.1 加法的なスパース正則化のためのアルゴリズム

3.1.1 近接勾配法およびその加速

2 乗誤差関数や，ロジスティック損失関数のように微分可能な誤差関数に対する最も基本的な方法は近接勾配法 (proximal gradient method) である．近接勾配法という名称は主に最適化のコミュニティで使われる名称で，別名として forward-backward splitting [19,20,47] あるいは iterative-shrinkage thresholding (IST) algorithm [21,27,28] という名前でも知られている．

この方法の特徴は微分可能な損失項と微分不可能な正則化項を区別して扱う点にある．

アルゴリズムとしては，繰り返しアルゴリズムで，現在の点 $\boldsymbol{w}^t$ において損失関数 $L(\boldsymbol{w})$ を線形近似したものに近接項 (proximity term) を加え，

$$\boldsymbol{w}^{t+1} = \underset{\boldsymbol{w}}{\operatorname{argmin}} \left(\left\langle \nabla L(\boldsymbol{w}^t), \boldsymbol{w} - \boldsymbol{w}^t \right\rangle + \lambda \|\boldsymbol{w}\|_\star + \frac{1}{2\eta_t} \left\| \boldsymbol{w} - \boldsymbol{w}^t \right\|_2^2 \right) \tag{3.4}$$

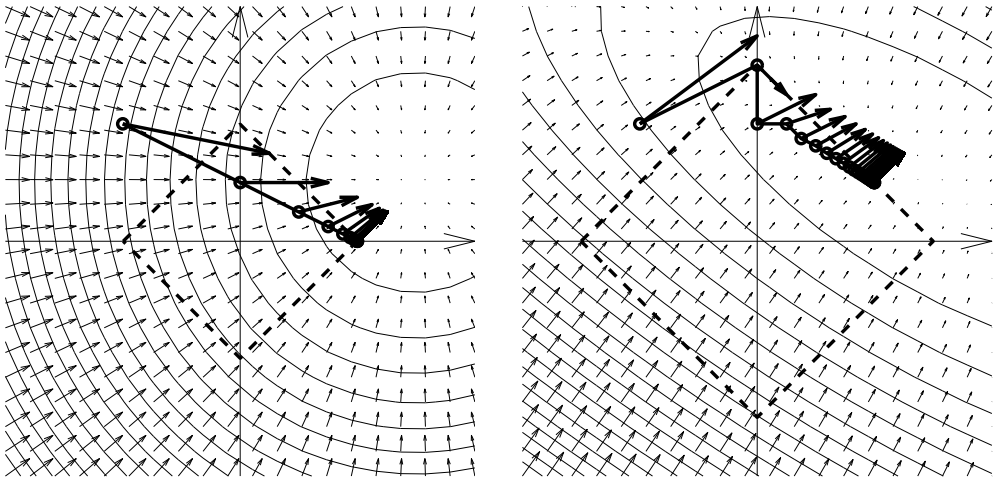
のように更新を行う．ここで第 3 項が近接項である．第 1 項と第 3 項をまとめることにより直ちにこの更新式は prox 作用素 (2.4) を用いて

$$\boldsymbol{w}^{t+1} = \operatorname{prox}_{\eta_t \lambda} \left(\boldsymbol{w}^t - \eta_t \nabla L(\boldsymbol{w}^t) \right) \tag{3.5}$$

のように書きなおすことができる．また，この形から，グループ ℓ_1 正則化，トレースノルム正則化のように prox 作用素を陽に書くことができる正則化項に対してはまったく同じアルゴリズムを適用できることがわかる．

更新式 (3.5) の直感的な意味は，損失関数の勾配方向に 1 ステップ進んだのちに，prox 作用素を用いて正則化項の効果を取り込んでいると言える（図 4(a) 参照）．また，正則化項がゼロの場合，prox 作用素は恒等写像になるので，上記アルゴリズムは単純な勾配法に帰着する．

定数 η_t は近接項の強さを決めるもので，上記更新式からは勾配ステップのステップサイズと見ることもできる．この定数の決め方としては損失関数の微分のリプシッツ定数 H



(a) Well conditioned problem. (b) Poorly conditioned problem.

Fig. 4. Behavior of the proximal gradient method in two dimensions. Concentric curves show the contours of the loss function $L(\boldsymbol{w})$, and the arrows show the gradient directions. The diamond shapes shown with dashed lines show the ℓ_1 norm of the optimal point in each scenario; thus they are not given as part of the optimization problems.

が有限の場合，すなわち

$$\|\nabla L(\boldsymbol{w}') - \nabla L(\boldsymbol{w})\|_2 \leq H\|\boldsymbol{w}' - \boldsymbol{w}\|_2 \quad (\forall \boldsymbol{w}, \boldsymbol{w}')$$

がとなるような定数 H が存在する場合，$\eta_t \leq 1/H$ のように取るのが最適である．この時，式 (3.4) の第 1 項と第 3 項を合わせたものが，損失関数の上界となることを示すことができる．定数 H の別の見方としては損失関数のヘシアン行列の最大固有値の上限である．

一般にリプシッツ定数 H はデザイン行列 $\boldsymbol{A}$（つまり入力データ）に依存し，あらかじめ仮定することができないので，実際には backtracking [6]，あるいは Barzilai-Borwein 法 [79] のような工夫を用いてステップサイズを選ぶことが多い．

近接勾配法の収束の速さは Tseng [76], Nesterov [53], Beck & Teboulle [6] らによって調べられており，損失関数 L が強凸かつその微分がリプシッツ連続の場合，線形収束すること，損失関数の微分がリプシッツ連続だが強凸とは限らない場合，ステップ数を k として

$$f(\boldsymbol{w}^k) - f(\boldsymbol{w}^*) \leq \frac{H\|\boldsymbol{w}^0 - \boldsymbol{w}^*\|_2^2}{2k}$$

のように $O(1/k)$ で収束することが知られている．ここで，$f(\boldsymbol{w}^*)$ は目的関数の最適値，H は損失関数の微分のリプシッツ定数，$\boldsymbol{w}^0$ は最適化を始める初期点である．またステップサイズは理想的に $\eta = 1/H$ とした．この結果は従来知られているなめらかな関数に対する勾配法の結果 [52] と同じであり，微分不可能な正則化項を損失項から区別して扱うことで，微分不可能な問題を微分可能な問題と同じように扱うことができるということを意味している．ただし，この結果は prox 作用素を厳密に計算できることを前提にして

Algorithm 1 Accelerated proximal gradient algorithm [6, 53].

1. Initialize $\boldsymbol{w}^0$ appropriately, $\boldsymbol{z}^1 = \boldsymbol{w}^0$, $s_1 = 1$.
2. Iterate until convergence:
   （a）Update $\boldsymbol{w}^t$:

$$\boldsymbol{w}^t = \mathrm{prox}_{\lambda\eta_t}\left(\boldsymbol{z}^t - \eta_t \nabla L(\boldsymbol{z}^t)\right).$$

   （b）Update $\boldsymbol{z}^t$:

$$\boldsymbol{z}^{t+1} = \boldsymbol{w}^t + \left(\frac{s_t - 1}{s_{t+1}}\right)\left(\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\right),$$

   where $s_{t+1} = \left(1 + \sqrt{1 + 4s_t^2}\right)/2$.

いる点に注意する．Prox 作用素の計算に誤差が含まれる場合は Combettes & Wajs [20], Schmidt et al. [62] などで扱われている．

近接勾配法の魅力は上で述べた手法の単純さに留まらない．実は上で述べた $O(1/k)$ の収束レートは最適ではなく，加速 (acceleration) というテクニックを使うことにより，最適な $O(1/k^2)$ を達成できることが知られている [6, 53].

具体的なアルゴリズムをアルゴリズム 1 に示す．アルゴリズムは理論的な解析から導出されているため，必ずしも直感的でないが，上述の単純な近接勾配法とほぼ同じ計算コストで収束レートを改善できる点が大きな魅力である．

加速付き近接勾配法をトレースノルム正則化に適用した例として Toh & Yun [71]，Ji & Ye [43] らの研究がある．

（加速付き）近接勾配法は正則化項が表 1 に示す基本的な正則化項の場合だけでなく，圧縮センシング (1.6) において，スパースとなる基底への変換行列 $\boldsymbol{\Phi}$ に逆行列が存在する場合（例えばウェーブレット基底）$\boldsymbol{A} = \boldsymbol{\Omega}\boldsymbol{\Phi}^{-1}$ と定義することにより，(3.1) の形に帰着することができる．この際，$\boldsymbol{\Phi}$ やその逆行列 $\boldsymbol{\Phi}^{-1}$ を陽に保持する必要はなく，例えばウェーブレット変換やその逆変換などの操作を行うことができれば十分である．

さらに，（加速付き）近接勾配法は構造的なスパース正則化項のように，prox 作用素を適用するステップが自明でない場合にも適用されている．その際に，繰り返しアルゴリズムを用いたり，双対問題の性質を利用したり [42] などの工夫が提案されている．

3.1.2 双対拡張ラグランジュ (dual augmented Lagrangian) 法

上述の（加速付き）近接勾配法は勾配法を滑らかでない正則化項を持つ問題に拡張したものであるが，そのために勾配法の持つ問題点も継承している．例えば，図 4(b) に示すようにデザイン行列 $\boldsymbol{A}$ の条件数が悪い場合，収束はかなり遅くなることがある．この問題

Table 2. Loss functions commonly used in machine learning and their convex conjugate functions. The functions are defined to be $+\infty$ outside their domains.

	loss function $f_\ell(\boldsymbol{y})$	convex conjugate $f_\ell^*(-\boldsymbol{\alpha})$
Squared loss	$\frac{1}{2}\|\boldsymbol{y}-\boldsymbol{z}\|_2^2$	$\frac{1}{2}\|\boldsymbol{\alpha}\|_2^2 - \langle \boldsymbol{\alpha}, \boldsymbol{y}\rangle$
Huber loss	$\sum_{i=1}^m \begin{cases} \frac{1}{2}(y_i-z_i)^2 & (\lvert y_i-z_i\rvert \le \epsilon), \\ \epsilon\lvert y_i-z_i\rvert - \epsilon^2/2 & (\text{otherwise}) \end{cases}$	$\frac{1}{2}\|\boldsymbol{\alpha}\|_2^2 - \langle \boldsymbol{\alpha}, \boldsymbol{y}\rangle$ $(-\epsilon \le \alpha_i \le \epsilon)$
Logistic loss	$\sum_{i=1}^m \log\left(1+\exp(-y_i z_i)\right)$	$\sum_{i=1}^m ((\alpha_i y_i)\log(\alpha_i y_i)$ $+(1-\alpha_i y_i)\log(1-\alpha_i y_i))$ $(0 \le \alpha_i y_i \le 1)$
Hyperbolic secant	$\sum_{i=1}^m \log\left(e^{y_i - z_i} + e^{-y_i+z_i}\right)$	$\frac{1}{2}\sum_{i=1}^m ((1-\alpha_i)\log(1-\alpha_i)$ $+(1+\alpha_i)\log(1+\alpha_i) - 2\alpha_i y_i)$ $(-1 \le \alpha_i \le 1)$

は加法的なスパース正則化を扱う際にデザイン行列 $\boldsymbol{A}$ の列が繰り返す場合には非常に顕著になる（例えば [33]）．

デザイン行列 $\boldsymbol{A}$ の性質が悪い場合を考えるためにはデザイン行列と損失関数の性質を分離することが重要である．このような分離を行うと，Fenchel 双対定理 [58] により最適化問題 (3.1) の双対問題は

$$\underset{\boldsymbol{\alpha}\in\mathbb{R}^m}{\text{maximize}} \quad -f_\ell^*(-\boldsymbol{\alpha}) - \delta_{\|\cdot\|_{\star^*}\le\lambda}(\boldsymbol{A}^\top\boldsymbol{\alpha}) \tag{3.6}$$

のように得られる．ここで f_ℓ^* は誤差関数 f_ℓ の凸共役である（表 2 に機械学習で頻繁に現れる誤差関数の凸共役対を示す）．また，$\delta_{\|\cdot\|_{\star^*}\le\lambda}$ は

$$\delta_{\|\cdot\|_{\star^*}\le\lambda}(\boldsymbol{v}) = \begin{cases} 0 & (\|\boldsymbol{v}\|_{\star^*} \le \lambda), \\ +\infty & (\text{otherwise}) \end{cases}$$

と定義される半径 λ の双対ノルム $\|\cdot\|_{\star^*}$ 球の指示関数であり，正則化項 $\lambda\|\cdot\|_\star$ の凸共役である．例えば ℓ_1 ノルムであれば双対ノルムは ℓ_∞ ノルムである．

筆者が提案する双対拡張ラグランジュ (dual augmented Lagrangian, DAL) 法は上記双対問題に対する拡張ラグランジュ法 (augmented Lagrangian method [36, 56]) である．

双対問題 (3.6) に拡張ラグランジュ法を適用するには，まず，第 2 項の中にある線形演算を陽に制約として書きなおし，以下の制約付き最小化問題を得る:

$$\begin{aligned} \underset{\boldsymbol{\alpha}\in\mathbb{R}^m, \boldsymbol{v}\in\mathbb{R}^n}{\text{minimize}} \quad & f_\ell^*(-\boldsymbol{\alpha}) + \delta_{\|\cdot\|_{\star^*}\le\lambda}(\boldsymbol{v}), \\ \text{subject to} \quad & \boldsymbol{A}^\top\boldsymbol{\alpha} = \boldsymbol{v}. \end{aligned}$$

ここで，説明の便宜上，最大化問題 (3.6) を符号反転して最小化問題に変換した．

上記線形制約付き最小化問題に対する拡張ラグランジュ関数 (augmented Lagrangian function) は以下のように与えられる:

$$\mathcal{L}_\eta(\boldsymbol{\alpha}, \boldsymbol{v}, \boldsymbol{w}) = f_\ell^*(-\boldsymbol{\alpha}) + \delta_{\|\cdot\|_\star^* \le \lambda}(\boldsymbol{v}) + \boldsymbol{w}^\top(\boldsymbol{A}^\top\boldsymbol{\alpha} - \boldsymbol{v}) + \frac{\eta}{2}\left\|\boldsymbol{A}^\top\boldsymbol{\alpha} - \boldsymbol{v}\right\|_2^2. \tag{3.7}$$

ここで，最後の項が「拡張」と呼ばれる部分であり，$\eta = 0$ の時，拡張ラグランジュ関数は通常のラグランジュ関数であることに注意する．また，ラグランジュ乗数 $\boldsymbol{w}$ に主問題 (3.1) の変数と同じ記号を用いたのは，まさに双対問題に関するラグランジュ乗数が主問題の変数と一致するためである．

拡張ラグランジュ法は拡張ラグランジュ関数 (3.7) の最小として定義される双対関数

$$d_\eta(\boldsymbol{w}) = \min_{\boldsymbol{\alpha}\in\mathbb{R}^m, \boldsymbol{v}\in\mathbb{R}^n} \mathcal{L}_\eta(\boldsymbol{\alpha}, \boldsymbol{v}, \boldsymbol{w})$$

に対する最急勾配法であり，上の最小化を達成する変数を

$$(\boldsymbol{\alpha}^t, \boldsymbol{v}^t) = \operatorname*{argmin}_{\boldsymbol{\alpha}\in\mathbb{R}^m, \boldsymbol{v}\in\mathbb{R}^n} \mathcal{L}_\eta(\boldsymbol{\alpha}, \boldsymbol{v}, \boldsymbol{w}) \tag{3.8}$$

として

$$\boldsymbol{w}^{t+1} = \boldsymbol{w}^t + \eta(\boldsymbol{A}^\top\boldsymbol{\alpha}^t - \boldsymbol{v}^t) \tag{3.9}$$

と与えられる．ここで更新式 (3.9) のように，双対関数 $d_\eta(\boldsymbol{w})$ の勾配方向が拡張ラグランジュ関数 $\mathcal{L}_\eta$ を形式的に $\boldsymbol{w}$ で微分したものに，最小化を達成する $(\boldsymbol{\alpha}^t, \boldsymbol{v}^t)$ を代入したものとして与えられることに注意する（この事実は Danskin の定理として知られている [8]）．

上記更新式 (3.9),(3.8) に対して，prox 作用素の定義 (2.4) を用いて変数 $\boldsymbol{v}$ を取り除くと，$\boldsymbol{w}^t$ と $\boldsymbol{\alpha}^t$ だけを含む更新式が得られる．これをアルゴリズム 2 に示す．

上記アルゴリズムは一見複雑であるが，Fenchel 双対定理を用いると，もとの最小化問題 (3.1) に対する近接最小化 (proximal minimization)

$$\boldsymbol{w}^{t+1} = \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n}\left(f_\ell(\boldsymbol{A}\boldsymbol{w}) + \lambda\|\boldsymbol{w}\|_1 + \frac{1}{2\eta_t}\|\boldsymbol{w} - \boldsymbol{w}^t\|_2^2\right) \tag{3.13}$$

を行なっていることと等価であることを示すことができる [74, 75]．近接最小化 (3.13) は一見，近接勾配法の更新式 (3.4) と類似しているが，損失関数（第 1 項）を線形近似していない点で異なる．

近接最小化問題 (3.13) は最小化問題 (3.1) に近接項（第 3 項）を加えただけなので，そのままでは最小化は容易ではないが，その Fenchel 双対問題 (3.10) を考えると滑らかな最小化問題であり，ニュートン法，擬似ニュートン法などの方法で比較的容易に最小化することができる．また，その際，第 2 項の中の prox 作用素は式 (3.12) より，次のステップの非ゼロ要素に対応する項のみ計算すればよいから，解がスパースであればあるほど内部最小化問題 (3.10) は効率良く解くことができる点に注意する．

Algorithm 2 Dual augmented Lagrangian (DAL) method

1. Initialize $\boldsymbol{w}^0$ appropriately. Choose an non-decreasing sequence $\eta_0 \leq \eta_1 \leq \eta_2 \leq \cdots$.
2. Iterate until convergence:
   （a）Minimize the augmented Lagrangian function as

$$\boldsymbol{\alpha}^t \simeq \underset{\boldsymbol{\alpha}\in\mathbb{R}^m}{\operatorname{argmin}} \underbrace{\left(f_\ell^*(-\boldsymbol{\alpha}) + \frac{1}{2\eta_t} \left\| \operatorname{prox}_{\eta_t \lambda} \left(\boldsymbol{w}^t + \eta_t \boldsymbol{A}^\top \boldsymbol{\alpha} \right) \right\|_2^2 \right)}_{=\varphi(\boldsymbol{\alpha})} \tag{3.10}$$

   with the stopping criterion

$$\|\nabla \varphi(\boldsymbol{\alpha}^t)\| \leq \sqrt{\frac{\gamma}{\eta_t}} \left\| \operatorname{prox}_{\eta_t \lambda} \left(\boldsymbol{w}^t + \eta_t \boldsymbol{A}^\top \boldsymbol{\alpha}^t \right) - \boldsymbol{w}^t \right\|_2^2, \tag{3.11}$$

   where $1/\gamma$ is the Lipshitz constant of ∇f_ℓ.
   （b）Update $\boldsymbol{w}^t$:

$$\boldsymbol{w}^{t+1} = \operatorname{prox}_{\eta_t \lambda} \left(\boldsymbol{w}^t + \eta_t \boldsymbol{A}^\top \boldsymbol{\alpha}^t \right). \tag{3.12}$$

さらに，更新式 (3.12)，(3.10) では正則化項の性質は prox 作用素にのみ現れているので，グループ ℓ_1 正則化やトレースノルム正則化の場合も表 1 から適切な prox 作用素を用いればまったく同じアルゴリズムを用いることができる．また，係数ごとに異なる正則化定数を持つ場合や，バイアス項を持つ場合も同様である．

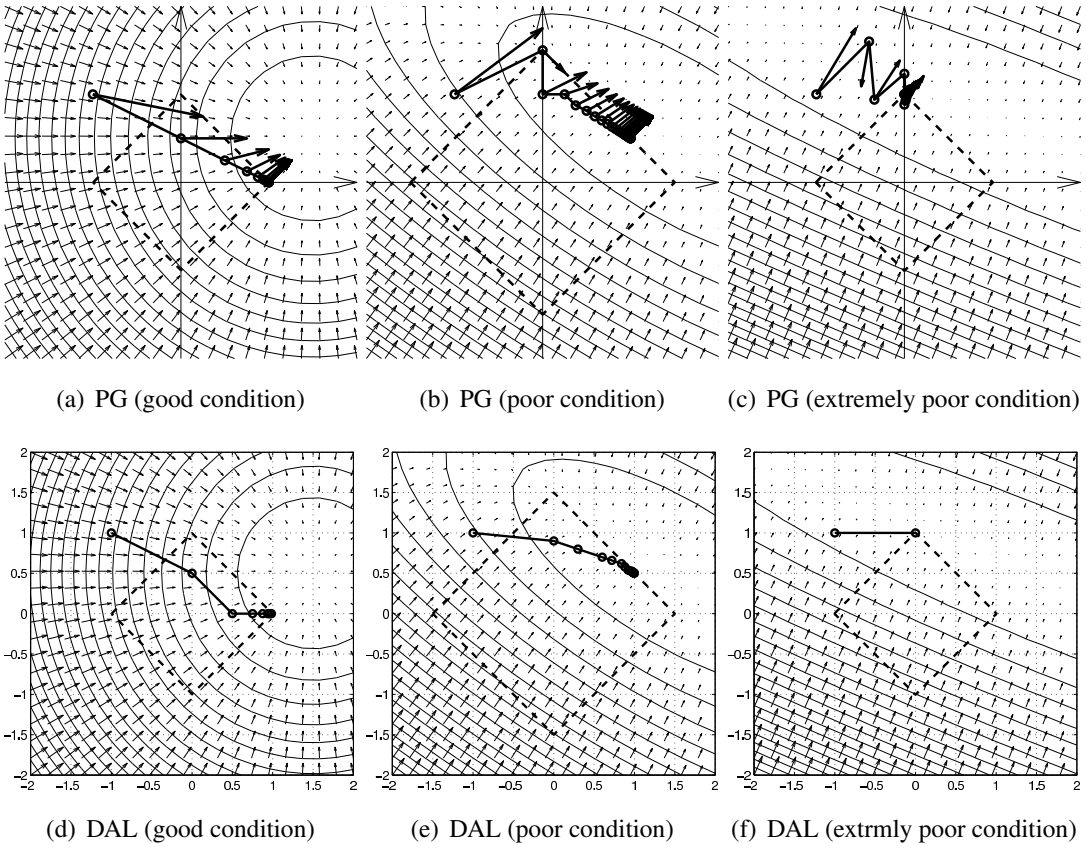
図 5 に 2 次元空間上での近接勾配法と DAL の振る舞いを比較する．1 ステップあたりの計算量が異なるので，必ずしも公平な比較とは言えないが，双対拡張ラグランジュ法 (DAL) の方が，近接勾配法よりも問題のスケーリングに影響されずに最適解に到達することができることがわかる．

双対拡張ラグランジュ法（アルゴリズム 2）は不等式 (3.11) のような現実的な停止基準のもとで**超 1 次収束**することが分かっている [75]．これを定理の形で以下に示す．

定理 1 $\boldsymbol{w}^1, \boldsymbol{w}^2, \ldots$ をアルゴリズム 2 が停止基準 (3.11) のもとで生成する解の系列とする．また，W^* を最小化問題 (3.1) の最小を達成する $\boldsymbol{w}$ の集合とする．さらに，最小化問題 (3.1) の目的関数 $f(\boldsymbol{w}) := f_\ell(\boldsymbol{A}\boldsymbol{w}) + \lambda \|\boldsymbol{w}\|_1$ とおき，以下の仮定 (A1)–(A3) をおく．

(A1)　正の定数 σ および α $(1 \leq \alpha \leq 2)$ が存在して

$$f(\boldsymbol{w}^{t+1}) - f(W^*) \geq \sigma \|\boldsymbol{w}^{t+1} - W^*\|_2^\alpha \qquad (t = 0, 1, 2, \ldots). \tag{3.14}$$



(a) PG (good condition) (b) PG (poor condition) (c) PG (extremely poor condition)

(d) DAL (good condition) (e) DAL (poor condition) (f) DAL (extrmly poor condition)

Fig. 5. Behaviors of proximal gradient (PG) method and dual augmented Lagrangian (DAL) method in 2D toy problems.

(A2) 損失関数 f_ℓ の微分 $\nabla f_\ell(z)$ は定数 $1/\gamma$ でリプシッツ連続である．すなわち

$$\|\nabla f_\ell(z') - \nabla f_\ell(z)\|_2 \leq \frac{1}{\gamma}\|z' - z\|_2^2 \quad (\forall z, z' \in \mathbb{R}^m). \tag{3.15}$$

(A3) Prox 作用素 prox_λ は厳密に計算することができる．

この時，不等式

$$\left\|\boldsymbol{w}^{t+1} - W^*\right\|_2^{\frac{1+\alpha\sigma\eta_t}{1+2\sigma\eta_t}} \leq \frac{1}{\sqrt{1+2\sigma\eta_t}}\|\boldsymbol{w}^t - W^*\|_2$$

が成立する．ただし，$\|\boldsymbol{w}^t - W^*\|$ は集合 W^* と点 $\boldsymbol{w}^t$ の最小距離 $\min_{\boldsymbol{w}\in W^*}\|\boldsymbol{w}^t - \boldsymbol{w}\|_2$ を表す．すなわち，$\alpha < 2$ あるいは $\alpha = 2$ かつ η_t が増加列なら，$\boldsymbol{w}^t$ は W^* に超 1 次収束する．

証明は文献 [75] を参照頂きたい．ここでは要点のみを述べる．

まず，仮定 (A1) の定数は反復回数が有限回の場合は必ず存在する．拡張ラグランジュ法の超 1 次収束は反復回数に関して漸近的な設定では Rockafellar [59], Bertsekas [7] らに

よって確立されており，上の定理はそれを有限回の反復の場合に補完していると解釈することができる．

また，仮定 (A2) のリプシッツ定数は，デザイン行列 $\boldsymbol{A}$ と損失関数 f_ℓ を分離しているため，入力データに無関係にあらかじめ解析的に求めることができる．

停止基準 (3.11) の右辺には σ, α などの未知の量は含まれず，既知の量だけで計算することができるため，現実的である．

仮定 (A3) の prox 作用素を厳密に計算できるという点は ℓ_1 正則化に関しては現実的であるが，トレースノルムなどへの適用を考えると必ずしも現実に即していると言えないかもしれない．これに関しては近接勾配法に関する Combettes & Wajs [20]，Schmidt et al. [62] らの結果を拡張することができると思われる．

最後に，DAL は内部最小化をある程度高い精度で解くため，ペナルティ項のパラメータ η_t を増加させることができる．これが定理 1 で見たように超 1 次収束を可能にする理由であり，次の節で述べる交互方向乗数法との決定的な違いである．

3.2 構造的なスパース正則化のためのアルゴリズム

ここでは交互方向乗数法 (alternating direction method of multipliers, ADMM) に基づく構造的なスパース正則化問題 (3.2) のためのアルゴリズムを議論する．

交互方向乗数法は拡張ラグランジュ法 [36, 56] の近似として Gabay & Mercier [29, 30] によって提案された方法で，最近ではスパース推定と関連して注目されたため Boyd et al. [11] によるチュートリアルが書かれている．

信号処理／機械学習の分野で，この方法を初めて適用したのは Goldstein & Osher [32] の研究だが，彼らの研究は Bregman iteration [55, 80] というまた異なる枠組みから導出されていて興味深い．

交互方向乗数法およびその基本となる拡張ラグランジュ法の基本的なアイディアは損失関数や，正則化項から線形演算を分離し，変数の間の絡みを取り除くことである．前節の双対拡張ラグランジュ法は正則化項の凸共役の中に含まれる項 $\boldsymbol{A}^\top\boldsymbol{\alpha}$ を分離することによって得られた．ここでも，以下のように構造的なスパース正則化問題 (3.2) の正則化項の中に含まれる項 $\boldsymbol{\Phi w}$ を線形制約として陽に表現する:

$$\begin{aligned} \underset{\boldsymbol{w}\in\mathbb{R}^n, \boldsymbol{z}\in\mathbb{R}^l}{\text{minimize}} \quad & L(\boldsymbol{w}) + \|\boldsymbol{z}\|_\star, \\ \text{subject to} \quad & \boldsymbol{z} = \boldsymbol{\Phi w}. \end{aligned}$$

ここで，正則化定数を式 (3.2) のように誤差項に含めていることに注意する．次に，拡張ラグランジュ関数 (augmented Lagrangian function) を以下のように定義する:

$$\mathcal{L}_\eta(\boldsymbol{w}, \boldsymbol{z}, \boldsymbol{\alpha}) = L(\boldsymbol{w}) + \|\boldsymbol{z}\|_\star + \boldsymbol{\alpha}^\top(\boldsymbol{z} - \boldsymbol{\Phi w}) + \frac{\eta}{2}\|\boldsymbol{z} - \boldsymbol{\Phi w}\|_2^2. \tag{3.16}$$

3.1.2 項で見たように，拡張ラグランジュ法であれば，拡張ラグランジュ関数 $\mathcal{L}_\eta$ を $\boldsymbol{w}$

および z に関して同時に最適化し，それを用いて α を更新するのだが，交互方向乗数法はこれを Gauss-Seidel 的に近似して以下のように更新を行う:

$$w^{t+1} = \operatorname*{argmin}_{w\in\mathbb{R}^n} \mathcal{L}_\eta(w, z^t, \alpha^t), \tag{3.17}$$

$$z^{t+1} = \operatorname*{argmin}_{z\in\mathbb{R}^l} \mathcal{L}_\eta(w^{t+1}, z, \alpha^t), \tag{3.18}$$

$$\alpha^{t+1} = \alpha^t + \eta(z^{t+1} - \Phi w^{t+1}). \tag{3.19}$$

ここで，式 (3.17) で更新した w^{t+1} を z^t の更新式 (3.18) で用いていることに注意する．拡張ラグランジュ法の近似という観点からはこのことは些細なことのように考えられるかもしれないが，実は交互方向乗数法は最適化問題 (3.2) の双対問題に対する Douglas-Rachford splitting [24,47] という方法と等価であることが知られており [25,29]，この等価性にはこのステップが重要である．

交互方向乗数法の収束性に関しては上述の Douglas-Rachford splitting が近接点法 (proximal point algorithm) [60] の一種である [25] という事実から，収束性があることが分かっている．また，Boyd et al. [11] もより初等的な方法で収束性の議論をしているが，収束のレートに関しては未解決としている．ただ，より最近になって交互方向乗数法の収束レートに関する論文が出ている [22,35].

交互方向乗数法の更新式 (3.17) および (3.18) をより詳しく見てみる．

更新式 (3.17) は w に依存する項のみを残して具体的に書き下すと

$$w^{t+1} = \operatorname*{argmin}_{w\in\mathbb{R}^n} \left(\frac{1}{\lambda} f_\ell(Aw) + \frac{\eta}{2} \|z - \Phi w + \alpha/\eta\|_2^2 \right)$$

となる．この最小化は損失関数 f_ℓ が滑らかならばニュートン法や擬似ニュートン法などの方法で最小化することができる．

特に損失関数が 2 乗損失の場合，

$$w^{t+1} = \operatorname*{argmin}_{w\in\mathbb{R}^n} \left(\frac{1}{2} \|w\|_C^2 - \left(A^\top y/\lambda + \Phi^\top(\eta z + \alpha) \right)^\top w \right) \tag{3.20}$$

と書ける．ただし，$C = \frac{1}{\lambda} A^\top A + \eta \Phi^\top \Phi$ と定義し，ノルム $\|\cdot\|_C$ を $\|w\|_C = \sqrt{w^\top C w}$ と定義した．

2 乗損失の場合，最小二乗問題 (3.20) を各反復に 1 回ずつ，繰り返し解くことになるが，行列 C は反復ごとに変化しないので，予め行列 C のコレスキー分解を計算しておけば，各反復の計算量を非常に小さくすることができる．

別の方法としては，最小二乗問題 (3.20) を近接勾配法（3.1.1 項を参照）と同様に線形化するという方法がある．これに関しては Zhang et al. [83] が詳しい．

2 乗損失の場合の更新式 (3.20) をよく見ると，$\Phi^\top \Phi$ および $A(\Phi^\top \Phi)^{-1} A^\top$ が正則の場合に $\lambda \to 0$ の極限が存在して，

$$w^{t+1} = B(AB)^{-1} y + \left(I - B(AB)^{-1} A \right) w^+$$

と書くことができる．ただしここで，$\boldsymbol{B} = (\boldsymbol{\Phi}^\top\boldsymbol{\Phi})^{-1}\boldsymbol{A}^\top$, $\boldsymbol{w}^+ = (\boldsymbol{\Phi}^\top\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^\top(\boldsymbol{z}^t + \boldsymbol{\alpha}^t/\eta)$ とおいた．上の更新式は制約付き最小二乗問題

$$\begin{aligned} \underset{\boldsymbol{w}}{\text{minimize}} \quad & \frac{1}{2}\|\boldsymbol{\Phi}\boldsymbol{w} - \boldsymbol{z}^t - \boldsymbol{\alpha}^t/\eta\|_2^2, \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{w} = \boldsymbol{y} \end{aligned}$$

を解いていることに注意すると，構造的なスパース正則化問題 (3.2) の正則定数 $\lambda \to 0$ の極限である制約付き最小化問題 (3.3) に対する交互方向乗数法になっていることが確認できる．これが，(3.2) において正則化定数 λ を正則化項につけずに損失項の前に付けた理由である．

一方，$\boldsymbol{z}$ に関する更新式 (3.18) は prox 作用素を用いて

$$\boldsymbol{z}^{t+1} = \mathrm{prox}_{1/\eta}\left(\boldsymbol{\Phi}\boldsymbol{w}^{t+1} - \boldsymbol{\alpha}^t/\eta\right)$$

と書きなおすことができる．Prox 作用素として表 1 にあるもの（あるいはその組み合わせ）であればこの計算は直ちに行うことができる．いかに複雑な正則化項であってもそれを分離可能な正則化項と，線形演算 $\boldsymbol{\Phi}$ に分離できれば簡単に prox 作用素が計算できる点が交互方向乗数法の魅力である．また，正則化項が ℓ_1 だけでなく，total variation の場合のように ℓ_1 とブロック ℓ_1 の組み合わせであってもこの方法は適用できる点に注意する．

最後に低ランクテンソル補完への応用について述べる．テンソル $\boldsymbol{\mathcal{W}}$ をベクトル化したものを $\boldsymbol{w} \in \mathbb{R}^N$ とおく．観測要素を取り出す演算は行列 $\boldsymbol{\Omega} \in \mathbb{R}^{M\times N}$ で与えられる．モード k 行列化は置換行列 $\boldsymbol{P}_k$ $(k = 1, \dots, K)$ で与えられる．従って，解くべき問題は

$$\underset{\boldsymbol{w}\in\mathbb{R}^N}{\text{minimize}} \quad \frac{1}{2\lambda}\|\boldsymbol{y} - \boldsymbol{\Omega}\boldsymbol{w}\|_2^2 + \sum_{k=1}^K \|\boldsymbol{P}_k\boldsymbol{w}\|_{\mathrm{tr}} \tag{3.21}$$

と与えられる．ここでトレースノルムは $\boldsymbol{w}$ の要素を置換して得られる N 次元ベクトルに対して定義している．厳密には $\boldsymbol{P}_k\boldsymbol{w}$ の要素を適切に並べて作られる行列に対して定義するべきだが，ここでは不必要に記法を複雑にしないためこうした．

最適化問題 (3.21) に対する拡張ラグランジュ関数 (3.16) は

$$\mathcal{L}_\eta(\boldsymbol{w}, (\boldsymbol{z}_k)_{k=1}^K, (\boldsymbol{\alpha}_k)_{k=1}^K) = \frac{1}{2\lambda}\|\boldsymbol{y} - \boldsymbol{\Omega}\boldsymbol{w}\|_2^2 + \sum_{k=1}^K \|\boldsymbol{z}_k\|_{\mathrm{tr}} + \sum_{k=1}^K \left(\boldsymbol{\alpha}_k{}^\top(\boldsymbol{z}_k - \boldsymbol{P}_k\boldsymbol{w}) + \frac{\eta}{2}\|\boldsymbol{z}_k - \boldsymbol{P}_k\boldsymbol{w}\|_2^2\right)$$

と与えられる．従って，$\boldsymbol{w}$, $\boldsymbol{z}_k$, および $\boldsymbol{\alpha}_k$ に関する更新式は

$$\begin{aligned} \boldsymbol{w}^{t+1} &= \left(\boldsymbol{\Omega}^\top\boldsymbol{\Omega} + \lambda\eta K\boldsymbol{I}\right)^{-1}\left(\boldsymbol{\Omega}^\top\boldsymbol{y} + \lambda\textstyle\sum_{k=1}^K \boldsymbol{P}_k{}^\top(\eta\boldsymbol{z}_k^t + \boldsymbol{\alpha}_k^t)\right), \\ \boldsymbol{z}_k^{t+1} &= \mathrm{prox}_{1/\eta}^{\mathrm{tr}}\left(\boldsymbol{P}_k\boldsymbol{w}^{t+1} - \boldsymbol{\alpha}_k^t/\eta\right) \qquad (k = 1, \dots, K), \\ \boldsymbol{\alpha}_k^{t+1} &= \boldsymbol{\alpha}_k^t + \eta(\boldsymbol{z}_k^{t+1} - \boldsymbol{P}_k\boldsymbol{w}^{t+1}) \qquad (k = 1, \dots, K) \end{aligned}$$

と与えられる．とくに，$\boldsymbol{w}^t$ の更新式では $\boldsymbol{\Omega}^\top\boldsymbol{\Omega}$ が対角行列であるため，要素ごとの演算だけ計算できればよい点に注意する．また，$\boldsymbol{\Phi}^\top\boldsymbol{\Phi} = \eta K\boldsymbol{I}$ であるので，上の議論から $\lambda \to 0$ の極限を考えることができて，この場合の $\boldsymbol{w}^t$ に関する更新式は

$$w_i^{t+1} = \begin{cases} (\boldsymbol{\Omega}^\top \boldsymbol{y})_i & (i \in \Omega), \\ \left(\frac{1}{K}\sum_{k=1}^K \boldsymbol{P}_k{}^\top(\boldsymbol{z}_k^t + \boldsymbol{\alpha}_k^t/\eta)\right)_i & (i \notin \Omega) \end{cases}$$

と与えられる．ここで Ω を観測された要素に対応するインデックスの集合とした．上の式の意味は観測された要素に関しては観測値で現在の値を上書きし，観測されていない要素については $\boldsymbol{z}_k^t$ と $\boldsymbol{\alpha}_k^t$ で与えられる予測値の平均で更新するということになる．

4. おわりに

この論文では，様々なスパース性を導く正則化の方法をサーベイし，それらを加法的なスパース正則化と構造的なスパース正則化に分類した．その上で，それぞれに対する最適化アルゴリズムを具体的に与えた．

加法的なスパース正則化に対しては近接勾配法およびその加速版がよく研究されている．近接勾配法の魅力は微分可能な損失関数と正則化項を別々に扱い，正則化項に関しては prox 作用素を計算すればよい，という単純さである．Boyd [12] は ℓ_1 正則化は 21 世紀の最小ニ乗法であると言ったが，著者は近接勾配法は 21 世紀の勾配法であると言っても過言ではないと思う．

加法的なスパース正則化においてはグループの重複に伴って生じるデザイン行列の条件数の悪化が問題となる．これに対応するために双対問題に対する拡張ラグランジュ (dual augmented Lagrangian) 法を紹介した．なお，DAL 法のプログラムは `http://www.ibis.t.u-tokyo.ac.jp/ryotat/dal/` で公開している．

構造的なスパース正則化に関しては，「構造」に対応する行列 $\boldsymbol{\Phi}$ をいかに正則化項から分離するか，という観点から交互方向乗数法を紹介した．交互方向乗数法は非常に幅広く適用できる手法であり，近接勾配法よりは導出が複雑であるものの，アルゴリズムは非常にシンプルで実装しやすい場合が多い．

このサーベイでは機械学習の視点から損失関数 f_ℓ，デザイン行列 $\boldsymbol{A}$，期待する構造に起因する $\boldsymbol{\Phi}$ を分離して扱うことを強調した．損失関数 f_ℓ および，構造 $\boldsymbol{\Phi}$ は問題を考える側が設計するものであるので，その性質を仮定することは妥当であると考えられる．一方，デザイン行列 $\boldsymbol{X}$ はデータに基づくものであり，一般には仮定が少ない方がよい．このように問題を詳しく見ることで問題に即した最適化アルゴリズムを設計したり，問題に即した仮定のもとで性能評価を行うことが重要であると著者は考える．

最後にこの論文でカバーできなかった内容について補足する．アトミックノルム [17] で正則化される一般的な（加法的な）スパース正則化を対象とするアルゴリズムとして Tewari et al. [68] の研究がある．確率的最適化はデータが非常に大規模な場合に実用

的に重要な方法であるだけでなく学習理論の観点からも重要である．これは Srebro & Tewari [67] の国際会議 ICML2010 のチュートリアルのスライドが詳しい．

謝辞 著者は科研費 22700138, 23240019, 25870192, 最先端研究開発支援プログラム（FIRST 合原最先端数理モデルプロジェクト），および NTT コミュニケーション科学基礎研究所の温かい支援に心から感謝します．

参考文献

[1] A. Argyriou, R. Foygel, and N. Srebro. Sparse prediction with the k-support norm. In *Advances in NIPS 25*, pages 1466–1474, 2012.

[2] N. Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68:337–404, 1950.

[3] F. Bach. Learning with submodular functions: A convex optimization perspective. Technical report, HAL 00645271, 2011.

[4] F. Bach, G. Lanckriet, and M. Jordan. Multiple kernel learning, conic duality, and the SMO algorithm. In *the 21st International Conference on Machine Learning*, pages 41–48, 2004.

[5] S. Bakin. *Adaptive regression and model selection in data mining problems*. PhD thesis, Australian National University, 1999.

[6] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.*, 2(1):183–202, 2009.

[7] D. P. Bertsekas. *Constrained Optimization and Lagrange Multiplier Methods*. Academic Press, 1982.

[8] D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, 1999. 2nd edition.

[9] P. J. Bickel, Y. Ritov, and A. B. Tsybakov. Simultaneous analysis of lasso and dantzig selector. *Ann. Stat.*, 37(4):1705–1732, 2009.

[10] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[11] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2010.

[12] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[13] P. Bühlmann and S. van de Geer. *Statistics for High-Dimensional Data: Methods, Theory, and Applications*. Springer, 2011.

[14] E. J. Candes, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? Technical report, arXiv:0912.3599, 2009.

[15] E. J. Candes and B. Recht. Exact matrix completion via convex optimization. *Found. Comput. Math.*, 9(6):717–772, 2009.

[16] E. J. Candes, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inform. Theory*, 52(2):489–509, 2006.

[17] V. Chandrasekaran, B. Recht, PA Parrilo, and A. Willsky. The convex geometry of linear inverse problems, prepint. Technical report, arXiv:1012.0621v2, 2010.

[18] S. Chen, D. Donoho, and M. Saunders. Atomic decomposition by basis pursuit. *SIAM J. Sci. Comput.*, 20(1):33–61, 1998.

[19] P. L. Combettes and J.-C. Pesquet. Proximal splitting methods in signal processing. In H. H. Bauschke, R. Burachik, P. L. Combettes, V. Elser, D. R. Luke, and H. Wolkowicz, editors, *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*. Springer, 2010.

[20] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Model. Sim.*, 4(4):1168–1200, 2005.

[21] I. Daubechies, M. Defrise, and C. De Mol. An Iterative Thresholding Algorithm for Linear Inverse Problems with a Sparsity Constraint. *Commun. Pur. Appl. Math.*, LVII:1413–1457, 2004.

[22] W. Deng and W. Yin. On the global and linear convergence of the generalized alternating direction method of multipliers. Technical Report TR12-14, CAAM, Rice University, 2012.

[23] D. L. Donoho. De-noising by soft-thresholding. *IEEE Trans. Inform. Theory*, 41(3):613–627, 1995.

[24] J. Douglas and H. H. Rachford. On the numerical solution of heat conduction problems in two and three space variables. *Trans. Amer. Math. Soc.*, 82(2):421–439, 1956.

[25] J. Eckstein and D. P. Bertsekas. On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 55(1):293–318, 1992.

[26] M. Fazel, H. Hindi, and S. P. Boyd. A Rank Minimization Heuristic with Application to Minimum Order System Approximation. In *Proc. of the American Control Conference*,

2001.

[27] M. A. T. Figueiredo, J. M. Bioucas-Dias, and R. D. Nowak. Majorization-minimization algorithm for wavelet-based image restoration. *IEEE Trans. Image Process.*, 16(12), 2007.

[28] M. A. T. Figueiredo and R. Nowak. An EM algorithm for wavelet-based image restoration. *IEEE Trans. Image Process.*, 12:906–916, 2003.

[29] D. Gabay. Applications of the method of multipliers to variational inequalities. In *Augmented Lagrangian Methods: Applications to the Numerical Solution of Boundary-Value Problems*, pages 299–331. Elsevier, 1983.

[30] D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers and Mathematics with Applications*, 2(1):17–40, 1976.

[31] S. Gandy, B. Recht, and I. Yamada. Tensor completion and low-n-rank tensor recovery via convex optimization. *Inverse Problems*, 27:025010, 2011.

[32] T. Goldstein and S. Osher. The split Bregman method for L1 regularized problems. *SIAM J. Imaging Sci.*, 2(2):323–343, 2009.

[33] S. Hara and T. Washio. Learning a common substructure of multiple graphical gaussian models. *Neural Networks*, 38:23–38, 2013.

[34] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer-Verlag, 2001.

[35] B. He and X. Yuan. On the o(1/n) convergence rate of the douglas-rachford alternating direction method. *SIAM J. Numer. Anal.*, 50(2):700–709, 2012.

[36] M. R. Hestenes. Multiplier and gradient methods. *J. Optim. Theory Appl.*, 4:303–320, 1969.

[37] R. A. Horn and C. R. Johnson. *Topics in matrix analysis*. Cambridge University Press, 1991.

[38] L. Jacob, G. Obozinski, and J.P. Vert. Group Lasso with overlap and graph Lasso. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 433–440. ACM, 2009.

[39] A. Jalali, P. Ravikumar, S. Sanghavi, and C. Ruan. A dirty model for multi-task learning. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, editors, *Advances in NIPS 23*, pages 964–972. 2010.

[40] G. J. O. Jameson. *Summing and nuclear norms in Banach space theory*. Cambridge

University Press, 1987.

[41] R. Jenatton, J.Y. Audibert, and F. Bach. Structured variable selection with sparsity-inducing norms. *J. Mach. Learn. Res.*, 12:2777–2824, 2011.

[42] R. Jenatton, J. Mairal, G. Obozinski, and F. Bach. Proximal methods for hierarchical sparse coding. *J. Mach. Learn. Res.*, 12:2297–2334, 2011.

[43] S. Ji and J. Ye. An accelerated gradient method for trace norm minimization. In *Proceedings of the 26th International Conference on Machine Learning (ICML2009)*, pages 457–464, New York, NY, 2009. ACM.

[44] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

[45] G. Lanckriet, N. Cristianini, L. El Ghaoui, P. Bartlett, and M. Jordan. Learning the kernel matrix with semi-definite programming. *J. Mach. Learn. Res.*, 5:27–72, 2004.

[46] Y. Lin and H. H. Zhang. Component selection and smoothing in smoothing spline analysis of variance models. *Ann. Stat.*, 34(5):2272–2297, 2006.

[47] P. L. Lions and B. Mercier. Splitting algorithms for the sum of two nonlinear operators. *SIAM J. Numer. Anal.*, 16(6):964–979, 1979.

[48] J. Liu, P. Musialski, P. Wonka, and J. Ye. Tensor completion for estimating missing values in visual data. In *Prof. ICCV*, 2009.

[49] C. A. Micchelli and M. Pontil. Learning the kernel function via regularization. *J. Mach. Learn. Res.*, 6:1099–1125, 2005.

[50] J. J. Moreau. Proximité et dualité dans un espace hilbertien. *Bulletin de la S. M. F.*, 93:273–299, 1965.

[51] S. Negahban, P. Ravikumar, M. J. Wainwright, and B. Yu. A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. *Stat. Sci.*, 27(4):538–557, 2012.

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

[53] Y. Nesterov. Gradient methods for minimizing composite objective function. Technical Report 2007/76, Center for Operations Research and Econometrics (CORE), Catholic University of Louvain, 2007.

[54] G. Obozinski, M. J. Wainwright, and M. I. Jordan. Support union recovery in high-dimensional multivariate regression. *Ann. Stat.*, 39(1):1–47, 2011.

[55] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin. An iterative regularization method

for total variation-based image restoration. *Multiscale Model. Sim.*, 4(2):460–489, 2005.

[56] M. J. D. Powell. A method for nonlinear constraints in minimization problems. In R. Fletcher, editor, *Optimization*, pages 283–298. Academic Press, London, New York, 1969.

[57] B. Recht, M. Fazel, and P.A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 52(3):471–501, 2010.

[58] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, 1970.

[59] R. T. Rockafellar. Augmented Lagrangians and applications of the proximal point algorithm in convex programming. *Math. of Oper. Res.*, 1:97–116, 1976.

[60] R. T. Rockafellar. Monotone operators and the proximal point algorithm. *SIAM J. Control Optimiz.*, 14:877–898, 1976.

[61] L.I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60(1-4):259–268, 1992.

[62] M. Schmidt, N. Le Roux, and F. Bach. Convergence rates of inexact proximal-gradient methods for convex optimization. Technical report, arXiv:1109.2415, 2011.

[63] B. Schölkopf and A. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT Press, Cambridge, MA, 2002.

[64] M. Signoretto, L. De Lathauwer, and J.A.K. Suykens. Nuclear norms for tensors and their use for convex multilinear estimation. Technical Report 10-186, ESAT-SISTA, K.U.Leuven, 2010.

[65] N. Srebro, J. D. M. Rennie, and T. S. Jaakkola. Maximum-margin matrix factorization. In Lawrence K. Saul, Yair Weiss, and Léon Bottou, editors, *Advances in NIPS 17*, pages 1329–1336. MIT Press, Cambridge, MA, 2005.

[66] N. Srebro and A. Shraibman. Rank, trace-norm and max-norm. In *COLT'05 Proceedings of the 18th annual conference on Learning Theory*, pages 599–764. Springer, 2005.

[67] N. Srebro and A. Tewari. Stochastic optimization for machine learning, 2010. http://ttic.uchicago.edu/˜nati/Publications/ICML10tut.pdf.

[68] A. Tewari, P. K. Ravikumar, and I. S. Dhillon. Greedy algorithms for structurally constrained high dimensional problems. In J. Shawe-Taylor, R.S. Zemel, P. Bartlett, F.C.N. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 882–890. 2011.

[69] R. Tibshirani. Regression shrinkage and selection via the lasso. *J. Roy. Stat. Soc. B*, 58(1):267–288, 1996.

[70] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B*, 67(1):91–108, 2005.

[71] K.-C. Toh and S. Yun. An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems. *Pacific Journal of Optimization*, 6:615–640, 2010.

[72] R. Tomioka, K. Hayashi, and H. Kashima. On the extension of trace norm to tensors. In *NIPS2010 Workshop: Tensors, Kernels and Machine Learning (TKML)*, 2010.

[73] R. Tomioka and T. Suzuki. Regularization strategies and empirical Bayesian learning for MKL. *J. Mach. Learn. Res.*, 2011. Accepted with minor revision.

[74] R. Tomioka, T. Suzuki, and M. Sugiyama. Augmented Lagrangian methods for learning, selecting, and combining features. In S. Sra, S. Nowozin, and S. J. Wright, editors, *Optimization for Machine Learning*. MIT Press, 2011.

[75] R. Tomioka, T. Suzuki, and M. Sugiyama. Super-linear convergence of dual augmented Lagrangian algorithm for sparse learning. *J. Mach. Learn. Res.*, 12:1537–1586, 2011.

[76] P. Tseng. Applications of a splitting algorithm to decomposition in convex programming and variational inequalities. *SIAM J. Control Optimiz.*, 29(1):119–138, 1991.

[77] D. Wipf and S. Nagarajan. A new view of automatic relevance determination. In *Advances in NIPS 20*, pages 1625–1632. MIT Press, 2008.

[78] D. P. Wipf, B. D. Rao, and S. Nagarajan. Latent variable bayesian models for promoting sparsity. *IEEE Trans. Inform. Theory*, 57(9):6236–6255, 2011.

[79] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo. Sparse reconstruction by separable approximation. *IEEE Trans. Signal Process.*, 2009.

[80] W. Yin, S. Osher, D. Goldfarb, and J. Darbon. Bregman Iterative Algorithms for L1-Minimization with Applications to Compressed Sensing. *SIAM J. Imaging Sci.*, 1(1):143–168, 2008.

[81] M. Yuan, A. Ekici, Z. Lu, and R. Monteiro. Dimension reduction and coefficient estimation in multivariate linear regression. *J. Roy. Stat. Soc. B*, 69(3):329–346, 2007.

[82] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *J. Roy. Stat. Soc. B*, 68(1):49–67, 2006.

[83] X. Zhang, M. Burger, and S. Osher. A unified primal-dual algorithm framework based on bregman iteration. *J. Sci. Comput.*, 46(1):20–46, 2010.

[84] 田中利幸. 圧縮センシングの数理. *IEICE Fundamentals Review*, 4(1):39–47, 2010.

冨岡 亮太 (会員)　〒113-8656 東京都文京区本郷 7-3-1

東京大学 大学院情報理工学系研究科 数理情報学専攻 助教．平成 20 年 3 月に同専攻から博士（情報理工学）の学位を得たあと，東京工業大学 大学院情報理工学研究科 計算工学専攻で研究員．平成 22 年 4 月から現職．専門は行列・テンソル的なモデリングに基づく機械学習，データマイニング，およびそのための最適化．