Advanced Machine Learning
Midterm Assignment

提出日：2025 年 7 月 22 日

系／学科／類：情報工学系

学籍番号：22B30282

E メール：kitamura.k.c2ea@m.isct.ac.jp

氏名：北村 要

目次

1　選択した問題

以下の問題を選択し、それぞれについて理論解析と実装を行った。

- **Problem 1**
- **Problem 2**
- **Problem 3**
- **Problem 4**

2　実装・コード

実装コードは、以下の GitHub リポジトリに格納した。

- `https://github.com/KitaKana2/AML`

3 Problem 1

3.1 問 1

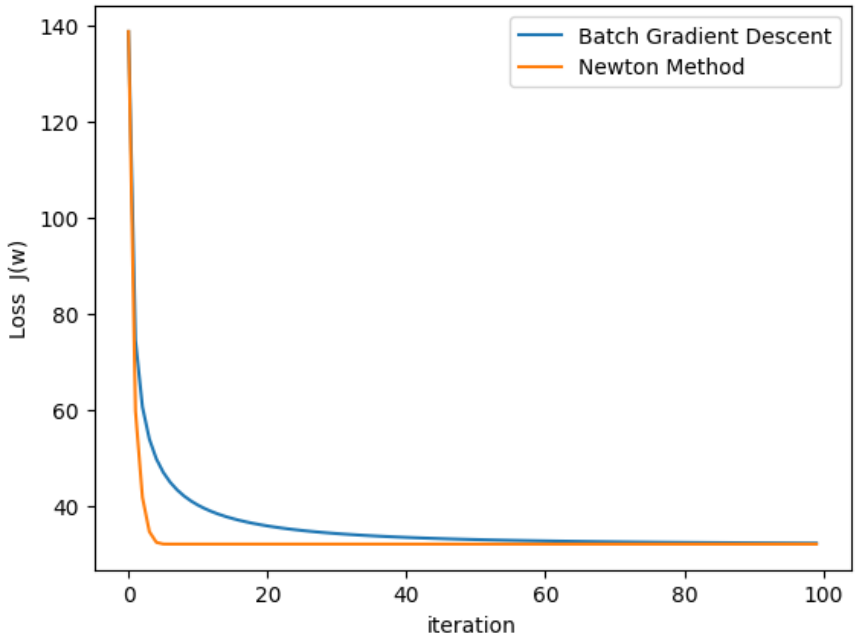

図 1　目的関数値の収束比較（$\left|J(w^{(t)}) - J(w^*)\right|$）

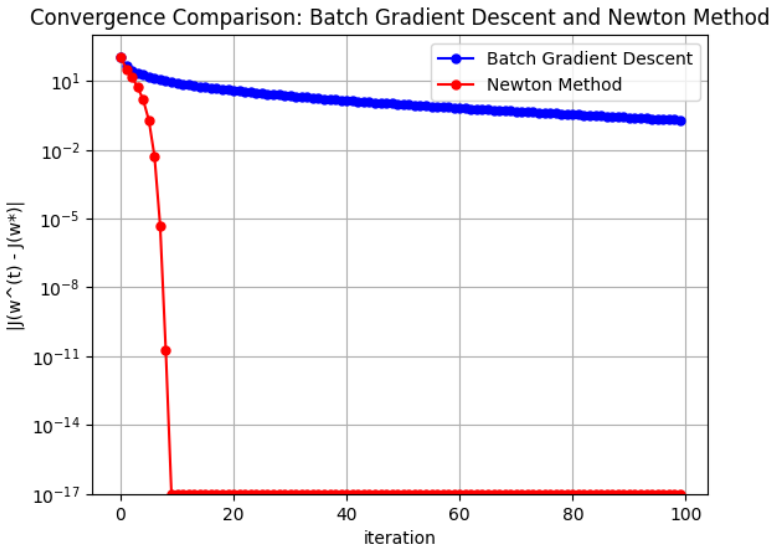

図 2　損失関数 $J(w)$ の各手法における変化

3.2 実装

指定されたデータセット（dataset 4）を用いた。
Batch steepest gradient method の学習率（learning rate）は、リプシッツ定数の逆数 $\eta = \frac{1}{L}$ に設定した。本問題におけるリプシッツ定数は $L = 0.25 \cdot \lambda_{\max}(X^\top X)$ である。終了条件は、勾配のノルム $\|\nabla J(w)\|_2$ が 10^{-8} 未満となることであり、最大で 100 回のイテレーションを行った。
Newton based method においては、学習率は使用せず、ヘッセ行列と勾配に基づいて逐次的にパラメータを更新した。終了条件は同様に、勾配のノルム $\|\nabla J(w)\|_2$ が 10^{-8} 未満となることであり、最大で 100 回のイテレーションを行った。

3.3 考察

図 1 および図 2 は、それぞれ Batch steepest gradient method と Newton based method の収束挙動を比較したものである。図 1 では、縦軸に $\left|J(w^{(t)}) - J(w^*)\right|$ を対数スケールでプロットし、収束速度の違いを明示的に示している。

Batch steepest gradient method では、初期の数ステップで急激に損失が低下した後、ゆるやかに収束していく様子が確認できる。一方で、Newton based method は初期から急速に損失が減少し、わずか 10 ステップ以内で収束に達している。これは、Newton based method がヘッセ行列を用いることで 2 次情報を活用し、より効率的に最小値を探索できるためである。

図 2 における損失関数 $J(w)$ の推移も、同様の傾向を示しており、Newton based method が高速に最適解に到達していることが分かる。

以上の結果から、精度・収束速度の両面において、Newton based method は Batch steepest gradient method よりも優れた性能を発揮していることが確認された。ただし、Newton based method ではヘッセ行列の計算と逆行列の計算コストがかかるため、計算資源が限られる場合には Batch steepest gradient method の方が適している場合もある。

4 Problem 2

4.1 問 1

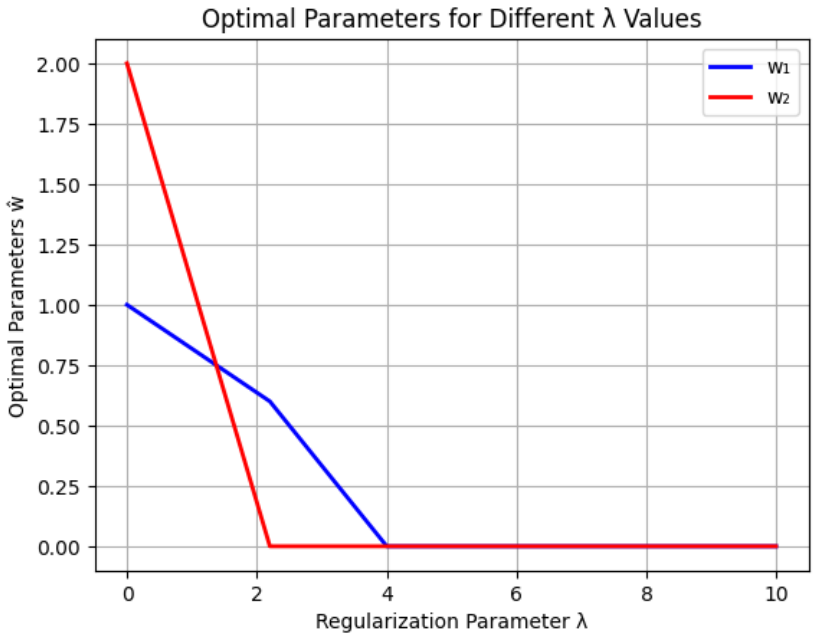

図 3　正則化パラメータ λ に対する最適パラメータ $\hat{w}$ の変化

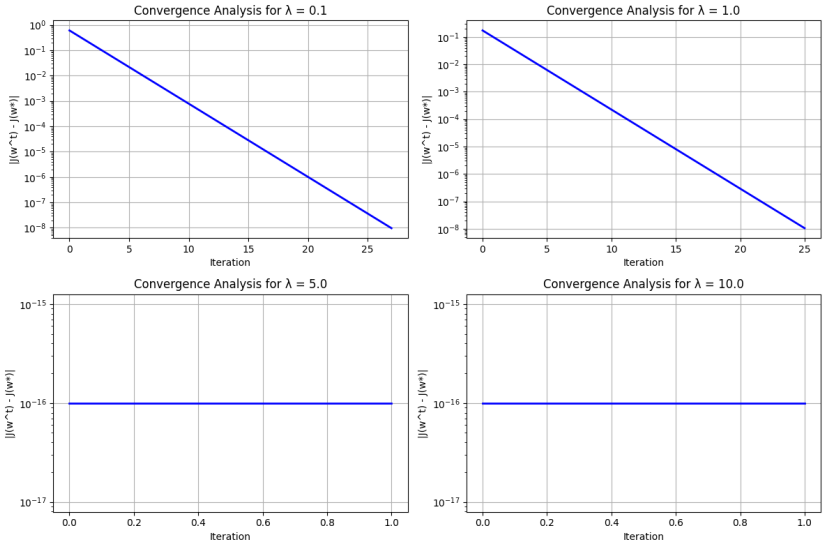

図 4　異なる λ に対する Proximal Gradient 法の収束挙動

4.2 問 2

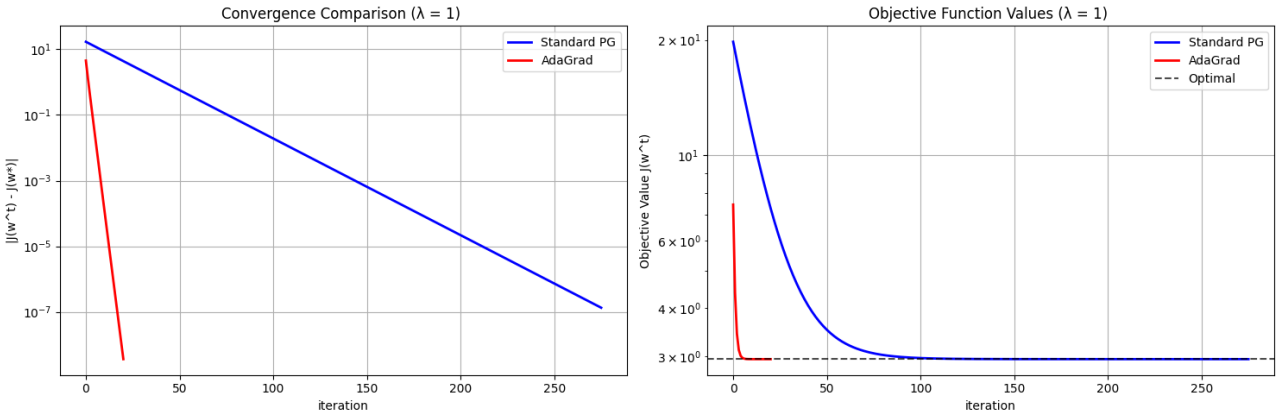

図 5　標準 PG 法と AdaGrad 法の比較（$\lambda = 1$ の場合）

4.3 実装

指定された目的関数

$$\hat{w} = \arg\min_{w} \left\{ \frac{1}{2}(w-\mu)^\top A(w-\mu) + \lambda \|w\|_1 \right\}$$

に対して、Proximal Gradient 法を用いた実装を行った。
標準的な Proximal Gradient method においては、学習率 (learning rate) を滑らかな部分の勾配のリプシッツ定数 L の逆数 $\eta = \frac{1}{L}$ に設定した。ここで、L は行列 A の最大固有値 $\lambda_{\max}(A)$ に一致し、$L = \lambda_{\max}(A)$ を用いて学習率を $\eta = 1/L$ と定めた。終了条件は、目的関数値 $J(w^{(t)})$ の変化が 10^{-8} 未満になることとし、最大で 1000 回のイテレーションを行った。

$A = \begin{bmatrix} 300 & 0.5 \\ 0.5 & 10 \end{bmatrix}$ の条件に対しては、標準的な PG 法に加えて、AdaGrad による適応的な学習率を用いた手法も実装した。

4.4 考察

図 3 は、正則化パラメータ λ の値を変化させたときの、最適パラメータ $\hat{w}$ の変化を示している。λ の増加に伴い、各成分が段階的にゼロへと収束する様子が確認できる。特に、$\lambda \geq 4$ では両方のパラメータがゼロとなっており、L1 正則化が効果的にスパースな解を導出していることが分かる。

図 4 は、異なる λ に対する Proximal Gradient 法の収束速度を示したものである。$\lambda = 0.1$ および 1.0 の場合、目的関数値 $|J(w^{(t)}) - J(w^*)|$ は指数的に減少し、良好な収束が観測されている。一方、$\lambda = 5.0$ および 10.0 のケースでは、初期ステップですでに最適解（すなわち $\hat{w} = 0$）に達しており、目的関数値も機械的精度の範囲で収束している。このことから、λ が大きい場合、早期にスパースな解へ収束する特性があることが分かる。

図 5 は、$\lambda = 1$ において標準的な PG 法と AdaGrad 法を比較した結果である。左図では、AdaGrad 法が標準的な PG 法と比べて急激に収束しており、収束速度に明確な優位性があることが示されている。また、右図の目的関数値の推移からも、AdaGrad 法が非常に少ない反復回数で最適値付近に到達していることが確認できる。

以上の結果より、L1 正則化は解にスパース性を導入する上で有効であり、また、学習率の調整が重要であることが分かった。特に、適応的な手法である AdaGrad は、標準的な PG 法と比較して高速な収束を実現する有効な手法であるといえる。

5 Problem 3

5.1 問 1

L2 正則化付きヒンジ損失に基づく 2 値分類問題において、元の最適化問題は次式で表される：

$$\hat{\boldsymbol{w}} = \underset{\boldsymbol{w}\in\mathbb{R}^d}{\operatorname{argmin}} \left(\sum_{i=1}^{n} \max(0, 1 - y_i \boldsymbol{w}^\top \boldsymbol{x}_i) + \frac{\lambda}{2} \|\boldsymbol{w}\|_2^2 \right) \tag{1}$$

ここで、$\boldsymbol{x}_i \in \mathbb{R}^d$ は i 番目の入力、$y_i \in \{\pm 1\}$ はラベル、$\boldsymbol{w} \in \mathbb{R}^d$ はパラメータベクトル、$\lambda > 0$ は正則化項の係数である。

式 (1) を制約付き最適化問題として次のように書き換える：

$$\text{minimize}_{\boldsymbol{w},\boldsymbol{\xi}} \quad \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \sum_{i=1}^{n} \xi_i \tag{2}$$

$$\text{subject to} \quad \xi_i \geq 0, \quad \xi_i \geq 1 - y_i \boldsymbol{w}^\top \boldsymbol{x}_i, \quad i = 1, \ldots, n \tag{3}$$

ラグランジュ乗数 $\alpha_i \geq 0$, $\mu_i \geq 0$ を導入してラグランジアンを構成する：

$$\mathcal{L} = \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \sum_i \xi_i + \sum_i \alpha_i (1 - y_i \boldsymbol{w}^\top \boldsymbol{x}_i - \xi_i) - \sum_i \mu_i \xi_i$$

$\boldsymbol{w}$ および $\boldsymbol{\xi}$ について微分し、最適条件を得る：

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{w}} = \lambda \boldsymbol{w} - \sum_i \alpha_i y_i \boldsymbol{x}_i = 0 \quad \Rightarrow \quad \boldsymbol{w} = \frac{1}{\lambda} \sum_i \alpha_i y_i \boldsymbol{x}_i \tag{4}$$

$$\frac{\partial \mathcal{L}}{\partial \xi_i} = 1 - \alpha_i - \mu_i = 0 \quad \Rightarrow \quad \mu_i = 1 - \alpha_i \tag{5}$$

$\mu_i \geq 0$ かつ $\alpha_i \geq 0$ の制約より、式 (5) から $0 \leq \alpha_i \leq 1$ が得られる。
式 (4) をラグランジアンに代入すると：

$$\mathcal{L} = \frac{\lambda}{2} \left\| \frac{1}{\lambda} \sum_i \alpha_i y_i \boldsymbol{x}_i \right\|^2 + \sum_i \xi_i + \sum_i \alpha_i (1 - y_i \boldsymbol{w}^\top \boldsymbol{x}_i - \xi_i) - \sum_i \mu_i \xi_i \tag{6}$$

$$= \frac{1}{2\lambda} \sum_{i,j} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j + \sum_i \xi_i + \sum_i \alpha_i - \sum_i \alpha_i y_i \boldsymbol{w}^\top \boldsymbol{x}_i - \sum_i \alpha_i \xi_i - \sum_i \mu_i \xi_i \tag{7}$$

式 (4) より $\sum_i \alpha_i y_i \boldsymbol{w}^\top \boldsymbol{x}_i = \frac{1}{\lambda} \sum_{i,j} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j$ であり、式 (5) より $\sum_i \xi_i - \sum_i \alpha_i \xi_i - \sum_i \mu_i \xi_i = \sum_i \xi_i (1 - \alpha_i - \mu_i) = 0$ である。

したがって：

$$\mathcal{L} = \frac{1}{2\lambda}\sum_{i,j}\alpha_i\alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j + \sum_i \alpha_i - \frac{1}{\lambda}\sum_{i,j}\alpha_i\alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j \tag{8}$$

$$= -\frac{1}{2\lambda}\sum_{i,j}\alpha_i\alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j + \sum_i \alpha_i \tag{9}$$

$$= -\frac{1}{2\lambda}\boldsymbol{\alpha}^\top \boldsymbol{K}\boldsymbol{\alpha} + \boldsymbol{\alpha}^\top \mathbf{1} \tag{10}$$

双対問題は主問題のラグランジアンの最大化であるため：

$$\text{maximize}_{\boldsymbol{\alpha}\in\mathbb{R}^n} \quad -\frac{1}{2\lambda}\boldsymbol{\alpha}^\top \boldsymbol{K}\boldsymbol{\alpha} + \boldsymbol{\alpha}^\top \mathbf{1} \tag{11}$$

$$\text{subject to} \quad \mathbf{0} \le \boldsymbol{\alpha} \le \mathbf{1} \tag{12}$$

ここで、$\boldsymbol{K} \in \mathbb{R}^{n\times n}$ は対称行列で、その (i,j) 要素は $K_{ij} = y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j$ で定義される。

5.2 問 2

KKT 条件より、双対最適解 $\boldsymbol{\alpha}$ に対して主問題の最適解 $\hat{\boldsymbol{w}}$ は以下のように表される：

$$\hat{\boldsymbol{w}} = \frac{1}{\lambda}\sum_{i=1}^{n}\alpha_i y_i \boldsymbol{x}_i \tag{13}$$

これは式 (4) から得られる勾配条件と一致しており、双対問題の解から主問題の解を再構築できることを示している。

5.3　問 3

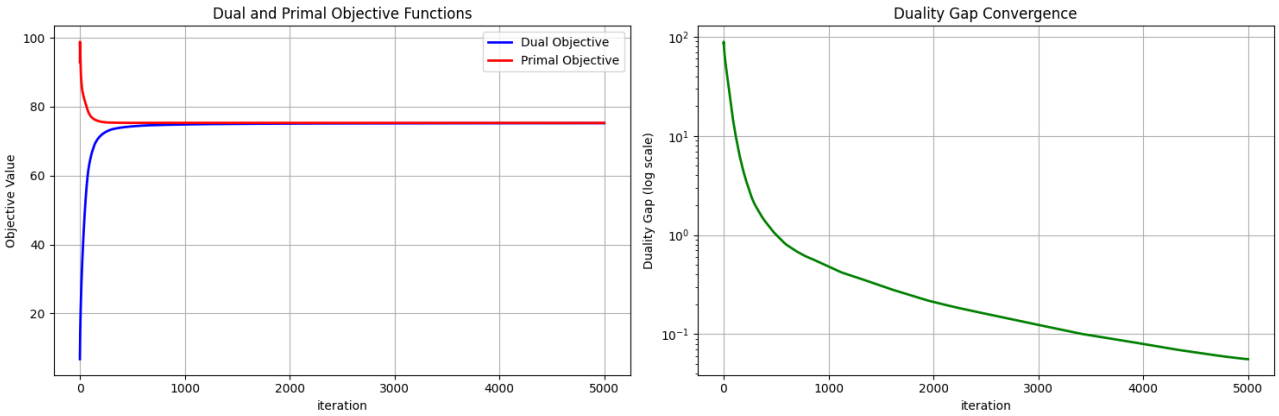

図 6　左：双対目的関数と主目的関数の推移，右：主・双対目的関数の差（収束過程）

5.4　問 4

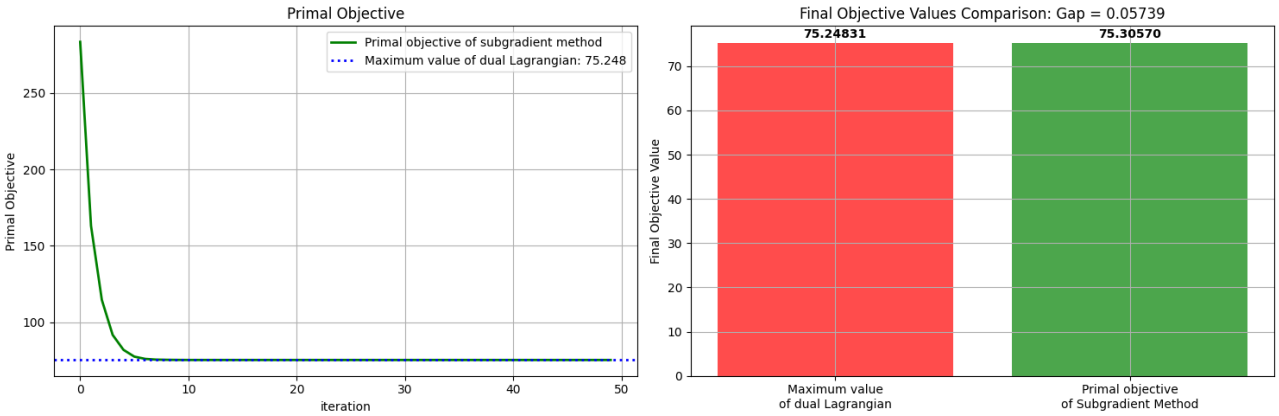

図 7　左：Subgradient 法における主目的関数の推移，右：最終目的関数値の比較

5.5 実装

データセット（dataset 2）において、データサイズ $n = 4000$ の中からランダムに 200 点を抽出して学習に用いた。

Projected Gradient 法により、双対ラグランジュ関数を最大化した。

各イテレーション t において、学習率は $\eta_t = \frac{\eta_0}{\sqrt{1+t}}$ として適応的に減衰させた。 $\eta_0 = 0.05$ に設定した。収束判定は、主問題の目的関数値と双対問題の目的関数値の差が 10^{-6} 未満になることとし、最大で 5000 回のイテレーションを実行した。

毎回の更新において、双対目的関数値・主目的関数値・双対目的関数値と主目的関数値の差（duality gap）を評価し、収束の進行を観察した。

主問題に対しては、Subgradient 法により、直接重みベクトル w を最適化した。目的関数は、ヒンジ損失と L2 正則化の和として定義される。ヒンジ損失のサブグラディエントは、$y_i w^\top x_i < 1$ の場合に $-y_i x_i$ を持ち、それ以外では 0 となる。正則化項については通常通り λw を勾配として加える。これらを組み合わせたサブグラディエントに基づいて、各イテレーションでパラメータを更新した。

各イテレーション t において、 学習率は $\eta_t = \frac{\eta_0}{\sqrt{1+t}}$ として適応的に減衰させた。$\eta_0 = 0.04$ に設定した。収束判定は、目的関数の変化量が 10^{-6} 未満となることとし、最大で 50 回のイテレーションを実行した。

5.6 考察

図 6 は、Projected Gradient 法を用いて双対問題を最適化した結果を示している。左図では、反復回数の増加に伴い、双対目的関数（青）と主目的関数（赤）が徐々に一致していく様子が確認できる。右図では、その差（主・双対の目的関数の差）が対数スケールでプロットされており、明確に単調減少していることから、適応的学習率を用いた Projected Gradient 法が安定して収束していることがわかる。最終的には 10^{-2} 以下に達しており、両者の目的関数の差はほとんどなくなっている。

図 7 左図では、Subgradient 法を用いて主問題の目的関数を最適化した結果を示しており、初期値から急激に目的関数値が減少し、早期に収束している様子が確認できる。また、右図では、最終的な主目的関数値と双対ラグランジュ関数の最大値を棒グラフとして比較しており、両者の差（目的関数の差）は約 0.03 程度と非常に小さいことが示されている。

これらの結果から、Projected Gradient 法による双対最適化と、Subgradient 法による主最適化がほぼ同じ解を導き出していることが確認でき、アルゴリズムの妥当性および双対定理の成立が数値的に検証された。

6 Problem 4

6.1 問 1

次の最適化問題が与えられている：

$$\hat{\boldsymbol{w}} = \underset{w \in \mathbb{R}^d}{\operatorname{argmin}} \left(\sum_{i=1}^{n} \max(0, 1 - y_i w^\top x_i) + \lambda \|w\|_1 \right) \tag{14}$$

この問題を線形計画問題に変換するために、以下の補助変数を導入する：

- ヒンジ損失に対して：スラック変数 $\xi_i \geq \max(0, 1 - y_i w^\top x_i)$, すなわち $\xi_i \geq 0$ かつ $\xi_i \geq 1 - y_i w^\top x_i$
- L1 ノルムに対して：各 w_j に対して $e_j \geq |w_j|$、すなわち $e_j \geq w_j$, $e_j \geq -w_j$

これらの補助変数を用いて、問題 (14) は次のような線形計画問題に書き換えられる：

$$\begin{aligned}
\text{minimize}_{w \in \mathbb{R}^d,\, \xi \in \mathbb{R}^n,\, e \in \mathbb{R}^d} \quad & \sum_{i=1}^{n} \xi_i + \lambda \sum_{j=1}^{d} e_j && (15) \\
\text{subject to} \quad & \xi_i \geq 1 - y_i w^\top x_i, \quad \forall i = 1, \ldots, n && (16) \\
& \xi_i \geq 0, \quad \forall i = 1, \ldots, n && (17) \\
& e_j \geq w_j, \quad \forall j = 1, \ldots, d && (18) \\
& e_j \geq -w_j, \quad \forall j = 1, \ldots, d && (19)
\end{aligned}$$

このように、目的関数 (15) および制約条件 (16)〜(19) はすべて線形で構成されており、与えられた問題を線形計画問題として定式化することができる。

線形計画問題の一般的な形式は以下のように表される：

$$\begin{aligned}
\text{minimize}_{\boldsymbol{z}} \quad & \boldsymbol{c}^\top \boldsymbol{z} && (20) \\
\text{subject to} \quad & A\boldsymbol{z} \leq \boldsymbol{b} && (21)
\end{aligned}$$

ここで、全変数をまとめたベクトル $\boldsymbol{z} \in \mathbb{R}^{2d+n}$ は以下のように定義される：

$$\boldsymbol{z} = \begin{bmatrix} \boldsymbol{w} \in \mathbb{R}^d \\ \boldsymbol{\xi} \in \mathbb{R}^n \\ \boldsymbol{e} \in \mathbb{R}^d \end{bmatrix}$$

目的関数の係数ベクトル $\boldsymbol{c} \in \mathbb{R}^{2d+n}$ は以下のように与えられる：

$$\boldsymbol{c}^\top = \begin{bmatrix} \mathbf{0}_d^\top & \mathbf{1}_n^\top & \lambda \cdot \mathbf{1}_d^\top \end{bmatrix}$$

制約条件は以下の 4 つの種類から構成される：

- ヒンジ損失による制約（$\xi_i \geq 1 - y_i \boldsymbol{w}^\top \boldsymbol{x}_i$）：

$$-y_i \boldsymbol{x}_i^\top \boldsymbol{w} - \xi_i \leq -1 \quad (\forall i)$$

- 非負制約（$\xi_i \geq 0$）：

$$-\xi_i \leq 0 \quad (\forall i)$$

- L1 正則化の線形化（$e_j \geq w_j$）：

$$w_j - e_j \leq 0 \quad (\forall j)$$

- L1 正則化の線形化（$e_j \geq -w_j$）：

$$-w_j - e_j \leq 0 \quad (\forall j)$$

これらの制約を行ベクトルとしてまとめた行列 $A \in \mathbb{R}^{(2n+2d)\times(2d+n)}$、および右辺ベクトル $\boldsymbol{b} \in \mathbb{R}^{2n+2d}$ を定義することで、問題は完全に線形計画の標準形に落とし込まれる。

$$A = \begin{bmatrix} \text{(i) ヒンジ損失} & : & -y_i \boldsymbol{x}_i^\top & -1 & \boldsymbol{0}^\top & \text{for } i = 1, \ldots, n \\ \text{(ii) } \xi_i \geq 0 & : & \boldsymbol{0}^\top & -\boldsymbol{e}_i^\top & \boldsymbol{0}^\top & \text{for } i = 1, \ldots, n \\ \text{(iii) } e_j \geq w_j & : & \boldsymbol{e}_j^\top & \boldsymbol{0}^\top & -\boldsymbol{e}_j^\top & \text{for } j = 1, \ldots, d \\ \text{(iv) } e_j \geq -w_j & : & -\boldsymbol{e}_j^\top & \boldsymbol{0}^\top & -\boldsymbol{e}_j^\top & \text{for } j = 1, \ldots, d \end{bmatrix} \in \mathbb{R}^{(2n+2d)\times(2d+n)}$$

$$\boldsymbol{b} = \begin{bmatrix} -1 \\ \vdots \\ -1 \quad (n \text{ entries}) \\ 0 \\ \vdots \\ 0 \quad (n \text{ entries}) \\ 0 \\ \vdots \\ 0 \quad (d \text{ entries}) \\ 0 \\ \vdots \\ 0 \quad (d \text{ entries}) \end{bmatrix} \in \mathbb{R}^{2n+2d}$$

ここで：- $\boldsymbol{e}_j$ は長さ d の単位ベクトルで、j 番目のみ 1。- $\boldsymbol{x}_i \in \mathbb{R}^d$ は i 番目の入力データ点。- $y_i \in \{\pm 1\}$ は対応するラベル。

このように構成することで、すべての制約を 1 つの行列 A とベクトル $\boldsymbol{b}$ にまとめることができ、標準的な線形計画問題の形式：

$$\text{minimize}_{\boldsymbol{z}} \quad \boldsymbol{c}^\top \boldsymbol{z} \tag{22}$$

$$\text{subject to} \quad A\boldsymbol{z} \leq \boldsymbol{b} \tag{23}$$

に完全に対応する。

6.2　問 2

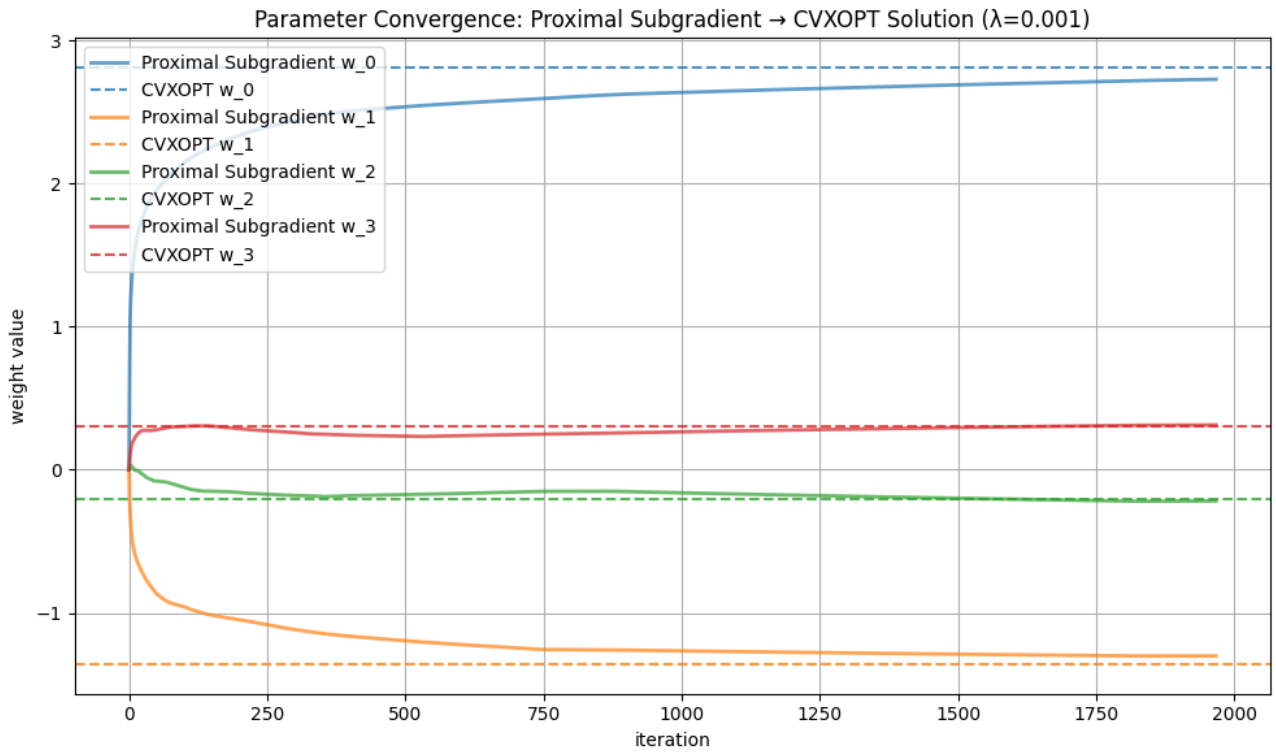

図 8　Proximal Subgradient 法によるパラメータの収束挙動（$\lambda = 0.001$）

6.3 実装

データセット（dataset 4）を用いて、学習を行った。

主問題に対しては、Proximal Subgradient 法により、直接重みベクトル $\boldsymbol{w}$ を最適化した。目的関数は、ヒンジ損失と L1 正則化の和として定義される。ヒンジ損失のサブグラディエントは、$y_i \boldsymbol{w}^\top \boldsymbol{x}_i < 1$ のときに $-y_i \boldsymbol{x}_i$ を取り、それ以外ではゼロとなる。

各イテレーション t において、学習率は $\eta_t = \frac{\eta_0}{\sqrt{1+t}}$ の形式で適応的に減衰させた。初期値は $\eta_0 = 1$ に設定した。正則化係数は $\lambda = 0.001$ に設定した。収束判定は、目的関数の変化量が 10^{-8} 未満となることとし、最大で 3000 回のイテレーションを実行した。

6.4 考察

図 8 は、CVXOPT を用いて線形計画問題として得られた最適解（破線）に対し、Proximal Subgradient 法によって得られる各重み成分（実線）がどのように収束していくかを比較している。

図より、すべての成分 w_j において、反復の進行に伴って Proximal Subgradient 法の解が CVXOPT 解に収束していく様子が明確に確認できる。

この結果は、適応的な学習率を用いた Proximal Subgradient 法が、主目的関数の最適化において安定的な収束性を持つことを示している。（正則化係数 $\lambda = 0.001$ を小さく設定したことで、L1 正則化によるスパース性は限定的である。）

以上のことから、CVX による厳密解と数値的最適化による近似解の収束挙動が整合しており、問題設定とアルゴリズムの実装が妥当であることが数値的に検証された。

7　学びたい機械学習のトピック

説明可能な AI について、関心がある。具体的には、機械学習モデルの予測に対して各特徴量がどのように寄与しているかを定量的に評価したり、個々の予測結果の根拠を可視化する技術について学びたいと考えている。

8　スライドの誤りについて

中間課題スライドの Problem 3 において、以下の誤りと思われる記述が見受けられた。

- 双対目的関数の $\alpha^\top K\alpha$ の係数が $-\frac{1}{4\lambda}$ となっているが、正しくは $-\frac{1}{2\lambda}$ 。
- KKT 条件から得られる $\hat{w}$ の式が $\frac{1}{2\lambda}\sum_i \alpha_i y_i x_i$ となっているが、正しくは $\frac{1}{\lambda}\sum_i \alpha_i y_i x_i$ 。
- Projected Gradient 法の更新式の勾配項も $\frac{1}{2\lambda}K\alpha - \mathbf{1}$ ではなく、$\frac{1}{\lambda}K\alpha - \mathbf{1}$ が正しいと考えられる。

これらはいずれも、主問題の正則化項 $\frac{\lambda}{2}\|w\|^2$ に対応する双対問題の導出と整合していないと思われる。