ART.T458 Advanced Machine Learning Midterm Assignment (Final report assigned by Shimosaka)

Masamichi Shimosaka Department of Computer Science Tokyo Institute of Technology

2021 年 7 月 30 日

Introduction

- 好みの問題を選び、選ばれた問題を解く。
- Tex/MSword を使用.
- 2021 年 7 月 30 日 17:00 までに、A4 サイズ／レターサイズのレポート（pdf 形式のファイル）を t2schola で提出してください。
- 各問題で獲得できる最大ポイントは、各セクションのタイトルに表示されています。
- i 番目の問題で獲得したスコアを $s_i \geq 0$ とすると、評価に使用する最終スコア q は次のように処理されます。

$$q = \min(30, \Sigma_i s_i)$$

- 上記の qpt に加えて、本講座の前半で履修したい機械学習のトピックを、線形モデル、スパース学習、最適化、後半の内容（深層学習）と同様に示していただければ、2pt（q とは無関係）を獲得できます。アンサンブルモデル（決定付きブースティング、回帰トレス、バギングなど)、ベイズ法（ノンパラメトリックベイズ、マルコフ連鎖モンテカルロ法、変分ベイズなど)、シーケンスモデリング（カルマンフィルタ、マルコフモデル、CRF など)、応用的な観点からの課題（レコメンダーシステム（ランキング学習、因子化マシン）など)、数値最適化（内包点アルゴリズム、拡張ラグランジアン、ADMM、逐次二次計画など）など、お好みのキーワードを取り上げていただければと思っています。前半・後半の講義の内容を参考にすることはご遠慮ください。また，スライドに誤字・脱字があった場合は，「x 回目の講義のスライド y ページ目の z の式に誤りがあり，w のように修正すべきです」という提案を，ボーナスとして 2pt 追加することができます。個人的には，コースの質を向上させるための提案も，もちろん歓迎します。
- 報告書には、実装のソースコードを記載する必要はありません。ただし、github などの公開されているリポジトリサービスを利用して、コードの URL を示すことは歓迎されます。
- 重要：本レポートでは，SciPy, scikit-learn, (Py)-Torch, Tensor Flow, Matlab の Neural network toolbox などの高度な機械学習ライブラリを使用することはできませんが，NumPy などの基本的な線形代数ライブラリと基本的な Matlab 言語機能を使用することができます．なお、本報告書の主な目的は、ブラックボックス化した ML ライブラリの使い方ではなく、実装による ML の数学的観点を理解することにあります。

- 英語だけでなく、日本語も使えます。
- https://bit.ly/32gbpRt の Jupyter notebook にあるいくつかの参考コードは、この課題を推進するのに役立つかもしれませんし、また頻繁に更新されるかもしれません。このスクリプトを使うのではなく、最初から自分でコードを作っても構いません。もちろん、Python だけでなく、Matlab も使用可能です。
  注：*、**、***は、それぞれ各問題の難易度を示しています。と表示されている問題は、ほとんど解けると思います。機械学習の数学的な側面を強化したい場合は、**, ***を選択することをお勧めします。なお，ML ソフトを実装していなくても 30pt を獲得することができます :-).

Problem 1 (10pts)*

ここでは、線形ロジスティック回帰による二値分類を考える。$x \in \mathbb{R}_d$ を d 次元の入力ベクトル、$w \in \mathbb{R}_d$ をモデルのパラメータとします。分類器は，$f(x) = 2[\![w^\top x]\!] > 0 - 1$ で表され，c は，c が真であれば 1 を，そうでなければ 0 を返す指示関数を表します．この最適化問題は次のように書くことができます。教師付きデータセット $\boldsymbol{x}_i, y_i{}_{i=1}^n$ を用いて，ロジスティック回帰の最適化問題を考えます．この最適化問題は次のように書くことができます。

$$\hat{\boldsymbol{w}} = \underset{\boldsymbol{w}}{\operatorname{argmin}} J(\boldsymbol{w})$$
$$J(\boldsymbol{w}) := \sum_{i=1}^{n} (\ln(1 + \exp(-y_i \boldsymbol{w}^\top \boldsymbol{x}_i))) + \lambda \boldsymbol{w}^\top \boldsymbol{w}.$$

ここでは，$d-1$ 次元の特徴空間でのオフセットに分類器を適応させるために，$\boldsymbol{x}_i$ に定数値 1 が含まれていると仮定する．いくつかの人工的なデータセット（Toy Dataset の項、Dataset IV を参照）を用いて、以下のようにいくつかの最適化手法を実装することを考えます。

1. バッチ式最優先勾配法 [1] の導入.
2. ニュートンベースの手法を導入。
3. 上記 2 つの最適化手法の性能を比較するために、$|J(\boldsymbol{w}^{(t)}) - J(\hat{\boldsymbol{w}})|$ w.r.t. t (ここで、$\boldsymbol{w}^{(t)}$ は、t 番目の繰り返しにおけるパラメータを表し、$\hat{\boldsymbol{w}}$ は、2 つの方法で得られた J の最小値に達する最適なパラメータを表します) を半対数プロットで表します。
4. マルチクラス版ロジスティック回帰（Toy データセット V を使用）にニュートン法と単純な最勾配法を導入し、上記のバイナリーロジスティック回帰と同じ実験を行う。

Problem 2 (5 pts)*

我々は、線形回帰に平方損失と L1 正則化を採用した lasso を考える。この問題では、近似勾配法（PG）を採用します。議論を簡単にするために、以下の目的を使用する。

$$\hat{\boldsymbol{w}} = \underset{w}{\operatorname{argmin}} \left((\boldsymbol{w} - \boldsymbol{\mu}) \top \boldsymbol{A} (\boldsymbol{w} - \boldsymbol{\mu}) + \lambda ||\boldsymbol{w}||_1 \right).$$

lasso の PG を実装し、いくつかの条件で結果を示してください。この問題では、同じ学習率 $\eta_t = L_{-1}$ を使用します。ここで、L は目的の勾配のリプシッツ定数を表し、ヘシアン行列 $2\boldsymbol{A}$ から導かれます（つまり、学

習率の逆数として $2\boldsymbol{A}$ の最大固有値を使用します：$\eta_t^{-1} - 1$)。

1. PG の結果を、繰り返し回数に応じて $||\boldsymbol{w}^{(t)} - \hat{\boldsymbol{w}}||$ で示す。セミログプロットを使用。次の条件を使用します。

$$A = \begin{pmatrix} 3 & 0.5 \\ 0.5 & 1 \end{pmatrix}, \mu^\top = (12).$$

L1 正則化の特性を確認するために、$\lambda = 2, 4, 6$ で実験を行います。数値結果は、講義で使用したスライドに記載されていることを思い出してください。また、cvx (matlab) / cvxopt (python) を使って結果を確認することもできます。

Problem 3 (15 pts)*

ここでは，サポート・ベクトル・マシンの双対（L2 正則化されたヒンジ・ロスに基づくバイナリ分類器）を考える。この分類の元々の最適化問題は，次のように表すことができる。

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

ここで $\boldsymbol{x}_i \in \mathbb{R}^d$, $\boldsymbol{w} \in \mathbb{R}^d$, $y_i \in \pm 1$, $\lambda > 0$ はそれぞれ i 番目の入力変数、パラメータベクトル、i 番目の入力データのラベル、正則化項の係数を表す。

1. この最適化の双対ラグランジュ関数が次のように書けることを検証する。

$$\begin{aligned} &\text{maximize}_{\boldsymbol{\alpha} \in \mathbb{R}_n} \ -\frac{1}{4\lambda} \boldsymbol{\alpha}^\top \boldsymbol{K} \boldsymbol{\alpha} + \boldsymbol{\alpha}^\top \mathbf{1}, \\ &\text{subject to } \mathbf{0} \leq \boldsymbol{\alpha} \leq \mathbf{1} \end{aligned} \tag{2}$$

   ここで、$\mathbf{1}$, $\mathbf{0}$ は、それぞれ 1, 0 を要素とする n 次元のベクトルを表し、$K \in \mathbb{R}^{n \times n}$ は対称正方行列を表し、その i 番目の行と j 番目の列の要素は $y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j$ である。
2. KKT 条件から、α で与えられる最適な重みパラメータ $\boldsymbol{w}$ が次のように書けることを検証する。

$$\hat{\boldsymbol{w}} = \frac{1}{2\lambda} \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{x}_i \tag{3}$$

3. 負の双対ラグランジュ関数の最小化を、投影勾配を用いて実施する。妥当性については、双対ラグランジュ関数のスコア（(2) を使用）と、ヒンジ損失関数と正則化の和を、それぞれ反復回数（(1) と (3) を使用）に対して示す。双対性ギャップが 0 になると収束することを確認する。ここでは、投影勾配がちょうど計算されると仮定する。

$$\boldsymbol{\alpha}^{(t)} = P_{[0,1]^n} \left(\boldsymbol{\alpha}^{(t-1)} - \eta_t \left(\frac{1}{2\lambda} \boldsymbol{K} \boldsymbol{\alpha}^{(t-1)} - \mathbf{1} \right) \right),$$

   ここで、η_t は、t 番目の反復における学習率を表し、$P_{[0,1]^n}$ は、入力された各キャストを $[0, 1]$ に投影する投影演算子を表している。

Problem 4 (10 pts)**

ここでは、ヒンジ損失関数と L1 正則化を用いた二値分類問題を考える。また、$bmx\mathbb{R}^d$ を入力、$bmw\mathbb{R}^d$ をモデルのパラメータとする。ここでは、線形回帰による二値判別関数を $f(x) = 2[\![w^\top x \geq 0]\!] - 1$ とし、$[\![c]\!]$ は、c が真であれば 1 を返し、そうでなければ 0 を返すものとします。

$$\hat{\boldsymbol{w}} = \underset{\boldsymbol{w}\in\mathbb{R}^d}{\operatorname{argmax}}(0, 1 - y_i \boldsymbol{w}^\top x_i) + \lambda ||\boldsymbol{w}||_2^2, \tag{4}$$

ここで、y_i $in\pm 1$ は、i 番目の学習例のバイナリラベルを表します。

1. (4) から線形プログラムを導出する（補助変数 $\xi_i \geq \max(0, 1 - y_i \boldsymbol{w}^\top \boldsymbol{x}_i) \geq 0$, $e_i \geq |\boldsymbol{w}_i| \geq 0$)。線形プログラムは次のように書けることを思い出してください。

$$\begin{aligned} &\text{minimize } \boldsymbol{c}^\top \boldsymbol{z} \\ &\text{subjectto } \boldsymbol{A}\boldsymbol{z} \leq \boldsymbol{b} \end{aligned}$$

   (L1 正則化ヒンジロスモデルからの LP を扱う LpBoost を参照)．)
2. いくつかの人工的なデータセット（ToyDatasetsection, DatasetIV 参照）を用いて， cvx (in Matlab) / cvxopt (in python) (参考) と，（バッチ式の）近位劣位法によって，この問題を実装してください．そして，パラメータが適切に収束することを確認します．データセットの仕様は、報告書に記載する必要があります。

Problem 5 (10 pts + optional 5 pts)**

以下の目的を持つ行列最適化問題を考えます。

$$\underset{\boldsymbol{Z}\in\mathbb{R}^{x\times n}}{\operatorname{argmin}} \left(\sum_{i,j\notin Q} |Ai, j - Zi, j|2 + \lambda \ ||\boldsymbol{Z}||_* \right)$$

ここで $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ はデータ行列（推薦システムにおけるユーザー対映画の評価データ）を表し、Q にはヌル値も含まれます。また、$||\cdot|_*$ は行列の核ノルムを表し、$\lambda > 0$ は正則化のためのハイパーパラメータを表します。この最適化問題では、$\boldsymbol{Z}$ は不完全なデータ行列 $\boldsymbol{A}$ から復元されたデータに相当します。推薦システムのシナリオでは、推定された $\boldsymbol{Z}$ の位置 Q は、映画の評価の推定スコアに相当します。

1. 行列の核ノルムの定義を www から調べて記述しなさい。その定義に加えて、核ノルムを用いた近位演算を定義しなさい。(ヒント：特異値分解を使う)
2. いくつかのデータセット（Toy Dataset III 参照）を用いて、この機械学習問題に近位勾配法を実装し、復元されたデータ $\boldsymbol{Z}$ を曲面プロットで示します。
3. (Option: Additional 5 pts 獲得可能) $\boldsymbol{A}$ を復元するための代替アプローチとして非負行列因子分解を実装し、ハイパーパラメータを選択してパフォーマンスを比較する。ここで実装した 2 つの方法の長所と短所を説明してください。

Problem 6 (10 pts)***

L-Lipschitz の凸関数 f を最小化するための勾配降下法を考える。開始点から $\boldsymbol{w}^{(0)} \in \mathbb{R}_d$ を出発点とし、各反復において、パラメータ $\boldsymbol{w}$ を、各反復において、パラメータ $\boldsymbol{w}$ を $\boldsymbol{w}^{(t+1)} \leftarrow \boldsymbol{w}^{(t)} - \eta \nabla f|_{\boldsymbol{w}=\boldsymbol{w}^{(t)}}$ のように更新する（固定ステップサイズの勾配降下法による更新)。最後に、最適な値を選択します。

$$\hat{\boldsymbol{w}} = \underset{\boldsymbol{w}^{(0)},\boldsymbol{w}^{(1)},\ldots,\boldsymbol{w}^{(n)},}{\operatorname{argmin}} \{f(\boldsymbol{w}^{(0)}), f(\boldsymbol{w}^{(1)}), \ldots, f(\boldsymbol{w}^{(T)})\}$$

. ここでは、関数 f を最小化するための最適な値を $f(\hat{\boldsymbol{w}}) - f(\boldsymbol{w}^*) \leq \varepsilon$ としたときに成立する最小の反復回数 T^* を求めます。ここでは、関数 f を最小化するための最適値を $\boldsymbol{w}^*$ とします。このような T^* は、$O(1/\varepsilon^2)$ と書けることを証明してください。

Problem 7 (10 pts)***

問題 6 と同様に, 滑らかな凸関数 f を最小化するための勾配降下法を考えます。開始点から $\boldsymbol{w}^{(0)} \in \mathbb{R}_d$ としてパラメータを更新していく。$\boldsymbol{w}^{(t+1)} \leftarrow \boldsymbol{w}^{(t)} - \eta \nabla f|_{\boldsymbol{w}=\boldsymbol{w}^{(t)}}$ としてパラメータ $\boldsymbol{w}$ を更新します。 - と更新する（固定ステップサイズの勾配降下法)。最後に、最適な値を選択します。

$$\hat{\boldsymbol{w}} = \underset{\boldsymbol{w}^{(0)},\boldsymbol{w}^{(1)},\ldots,\boldsymbol{w}^{(n)},}{\operatorname{argmin}} \{f(\boldsymbol{w}^{(0)}), f(\boldsymbol{w}^{(1)}), \ldots, f(\boldsymbol{w}^{(T)})\}$$

. ここでは、$\eta = 1/\gamma$ を設定したときに $f(\hat{\boldsymbol{w}}) - f(\boldsymbol{w}_*) \leq \varepsilon$ が成立する最小の反復回数 T^* を推定したい。ここでは、関数 f を最小化するための最適値を $\boldsymbol{w}^*$ とします。このような T^* は、$O(1/\epsilon)$ と書けることを証明してください。

Problem 8 (10pts)*

線形回帰と正則化の効果について考えます。講義で示したように、最小二乗問題として知られる線形回帰の最適化は、以下の最適化問題として定義されます。

$$\hat{\boldsymbol{w}}_{LS} = \underset{\boldsymbol{w}}{\operatorname{argmin}} \frac{1}{2}||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}||_2^2,$$

ここで、設計行列、応答ベクトル、パラメータはそれぞれ $\boldsymbol{X} \in \mathbb{R}^{n \times d}$, $\boldsymbol{y} \in \mathbb{R}^n$, and $\boldsymbol{w} \in \mathbb{R}^d$ で表されます (必要に応じて、講義のスライドを確認し、ビデオをチェックしてください)。この最適化による回帰は，条件によっては正常に動作することもありますが，オーバーフィッティングが起こりやすいことも知られています。オーバーフィッティングの問題に対処する簡単な方法として、リッジ正則化は機械学習の分野でよく使われています。その結果、最適化問題は次のように定義されます。

$$\hat{\boldsymbol{w}}_{ridge} = \underset{\boldsymbol{w}}{\operatorname{argmin}} \frac{1}{2}||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}||_2^2 + \lambda||\boldsymbol{w}||_2^2.$$

1. それぞれ、$\hat{\boldsymbol{w}}_{LS}$、$\hat{\boldsymbol{w}}_{ridge}$ の解析解を得る。ここでは、$\boldsymbol{X}^\top\boldsymbol{X}$ が正則であること、すなわち、$(\boldsymbol{X}^\top\boldsymbol{X})$ - 1 が存在することを仮定する。

2. たとえ $\boldsymbol{X}^\top \boldsymbol{X}$ が規則的でなくても、$\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I}$ が規則的であることを証明してください。つまり、$\boldsymbol{w}_{ridge}$ の最適なパラメータは、$\boldsymbol{X}^\top \ bmX$ が正則であるかどうかに関わらず、常に利用できるということです。ここで、$\boldsymbol{I} \in \mathbb{R}^{d \times d}$ は、恒等行列を表し、$\lambda > 0$ は、正則化のハイパーパラメータを表します。
3. ここでは、$\boldsymbol{X}^t op \boldsymbol{X}$ が規則的であると仮定して、$||\boldsymbol{X}\hat{\boldsymbol{w}}_{LS}||_2^2 \geq ||\boldsymbol{X}\hat{\boldsymbol{w}}_{ridge}||_2^2$ を証明します。この結果は、機械学習におけるシュリンケージとも呼ばれています。機械学習において、縮退が有効に働く場面を説明してください。

Problem 9 (10 pts) **

1. ここでは、(x, y) から点 $\{x_i, y_i\}_{i=1}^n$ までの最大距離を最小化することを考えます。

$$\text{minimize}_{x \in \mathbb{R}, y \in \mathbb{R}} \quad \max_i \sqrt{(x_i - x)^2 + (y_i - y)^2}$$

この最適化問題を二次計画問題に変換します。二次関数を目的関数として使用し、制約条件には線形の等式と不等式を使用します。

2. 次の最適化を凸最適化に変換してください。ここでは、変数 x_1, x_2, x_3 をそれぞれ実数とする。ヒント：4 番目の制約条件に注目し、x_1, x_2, x_3 をそれぞれ他の変数で表してください。

$$\begin{aligned} \text{minimize}_{x_1, x_2, x_3} \ & x_2/x_1 \\ \text{subject to } & x_1^2 + x_2/x_3 \leq \sqrt{x}_2 \\ & x_1/x_2 = x_3^2 \\ & 2 \leq x_1 \leq 3, \\ & x_1, x_2, x_3 > 0 \end{aligned}$$