計画数理学特論
～第7回：勾配法の基礎・応用～

担当：蓮池隆　(経営システム工学科)
e-mail: thasuike@waseda.jp

本日の講義について

- 勾配法の基礎として，凸2次関数を例とした最急降下法の収束速度を考察
- 勾配法の応用として，アダブーストと座標降下法との関わり，多層パーセプトロンと勾配法との関わりを紹介(詳細までは踏み込まない予定．詳細を知りたい方は，それぞれの専門書を参照すること)

(再掲)直線探索を用いる反復法

(反復法の流れ)

STEP0：初期点：$\boldsymbol{x}_0 \in \mathbb{R}^n$を設定し，$k \leftarrow 0$とする.

STEP1：停止条件が満たされるならば，$\boldsymbol{x}_k$を解として出力し，終了.

STEP2：降下方向として$\boldsymbol{d}_k$を設定.

STEP3：$\phi(\alpha) = f(\boldsymbol{x}_k + \alpha \boldsymbol{d}_k)$, $\alpha \geq 0$に対する直線探索により，ステップ幅α_kを計算.

STEP4：$\boldsymbol{x}_{k+1} \leftarrow \boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k$ と更新.

STEP5：$k \leftarrow k + 1$として，STEP1へ戻る.

(再掲)座標降下法のアルゴリズム

STEP0：初期点：$\boldsymbol{x}_0 \in \mathbb{R}^n$を設定し，$k \leftarrow 0$とする.

STEP1：停止条件が満たされるならば，$\boldsymbol{x}_k$を解として出力し，終了.

STEP2：降下方向として$\boldsymbol{d}_k$を，$\pm\boldsymbol{e}_1, \pm\boldsymbol{e}_2, \ldots, \pm\boldsymbol{e}_n$ の中から選択する.

STEP3：$\phi(\alpha) = f(\boldsymbol{x}_k + \alpha\boldsymbol{d}_k)$, $\alpha \geq 0$に対する直線探索により，ステップ幅α_kを計算.

STEP4：$\boldsymbol{x}_{k+1} \leftarrow \boldsymbol{x}_k + \alpha_k\boldsymbol{d}_k$ と更新.

STEP5：$k \leftarrow k + 1$として，STEP1へ戻る.

反復法で，STEP２の内容が座標降下法に合った内容に変わっただけ

(再掲)最急降下法のアルゴリズム

STEP0：初期点：$\boldsymbol{x}_0 \in \mathbb{R}^n$を設定し，$k \leftarrow 0$とする.

STEP1：停止条件が満たされるならば，$\boldsymbol{x}_k$を解として出力し，終了.

STEP2：降下方向として$\boldsymbol{d}_k = -\nabla f(\boldsymbol{x}_k)$を設定する.

STEP3：$\phi(\alpha) = f(\boldsymbol{x}_k + \alpha \boldsymbol{d}_k)$, $\alpha \geq 0$に対する直線探索により，ステップ幅α_kを計算.

STEP4：$\boldsymbol{x}_{k+1} \leftarrow \boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k$ と更新.

STEP5：$k \leftarrow k + 1$として，STEP1へ戻る.

反復法で，STEP２の内容が最急降下法に合った内容に変わっただけ

最急降下法の収束速度

(例) 凸2次関数：$f(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}^*)^t Q(\boldsymbol{x} - \boldsymbol{x}^*)$ (ただし，Qは正定値対称行列)

→ $\boldsymbol{x} = \boldsymbol{x}^*$が最適解となり，$f(\boldsymbol{x}^*) = 0$

・これを最急降下法で求めていくことを考えると…

・$\phi(\alpha) = f(\boldsymbol{x}_k + \alpha \boldsymbol{d}_k)$の最適解$\alpha$を厳密に求めると，最急降下法の$\boldsymbol{d}_k$は

$\boldsymbol{d}_k = -\nabla f(\boldsymbol{x}_k) = -Q(\boldsymbol{x}_k - \boldsymbol{x}^*)$より

最急降下法の収束速度

(例) 凸2次関数：$f(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}^*)^t Q(\boldsymbol{x} - \boldsymbol{x}^*)$ (ただし，Qは正定値対称行列)

→ $\boldsymbol{x} = \boldsymbol{x}^*$が最適解となり，$f(\boldsymbol{x}^*) = 0$

・ $f(\boldsymbol{x}_{k+1})$を求めると

$$f(\boldsymbol{x}_{k+1}) = \left\{1 - \frac{\|\nabla f(\boldsymbol{x}_k)\|^4}{\nabla f(\boldsymbol{x}_k)^t Q \nabla f(\boldsymbol{x}_k) \cdot \nabla f(\boldsymbol{x}_k)^t Q^{-1} \nabla f(\boldsymbol{x}_k)}\right\} f(\boldsymbol{x}_k)$$

・ここで，Qの固有値：$0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$とすると

$$\nabla f(\boldsymbol{x}_k)^t Q \nabla f(\boldsymbol{x}_k) \cdot \nabla f(\boldsymbol{x}_k)^t Q^{-1} \nabla f(\boldsymbol{x}_k)$$
$$\leq \lambda_n \|\nabla f(\boldsymbol{x}_k)\|^2 \cdot \frac{1}{\lambda_1} \|\nabla f(\boldsymbol{x}_k)\|^2 = \frac{\lambda_n}{\lambda_1} \|\nabla f(\boldsymbol{x}_k)\|^4$$

最急降下法の収束速度

(例) 凸2次関数：$f(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}^*)^t Q(\boldsymbol{x} - \boldsymbol{x}^*)$ (ただし，Qは正定値対称行列)

→ $\boldsymbol{x} = \boldsymbol{x}^*$が最適解となり，$f(\boldsymbol{x}^*) = 0$

・つまり，Qの最小固有値をλ_1，最大固有値をλ_nとするとき，

最急降下法の収束速度の意味

・最急降下法の収束速度と等高線・点列生成との関連性

機械学習への応用①

座標降下法とブースティング

- T個のデータ：$(\boldsymbol{a}_1, b_1), (\boldsymbol{a}_2, b_2), \cdots, (\boldsymbol{a}_T, b_T)$, $\boldsymbol{a}_t \in \mathbb{R}^d, b_t \in \{+1, -1\}$
  (つまり，入力$\boldsymbol{a}_t$が与えられたときに，1か−1をb_tの値として出力する)
- 新たなデータ：$\boldsymbol{a} \in \mathbb{R}^d$が得られたとき，出力$b \in \{+1, -1\}$を以下の関数
  $H_{\boldsymbol{x}}(\boldsymbol{a}) = \sum_{i=1}^{n} x_i h_i(\boldsymbol{a})$の符号で判定 → $H_{\boldsymbol{x}}(\boldsymbol{a}) > 0$なら$b = 1, H_{\boldsymbol{x}}(\boldsymbol{a}) < 0$なら$b = -1$
  (ここで，$h_1(\boldsymbol{a}), h_2(\boldsymbol{a}), \ldots, h_n(\boldsymbol{a})$は$\{+1, -1\}$の値をとる関数(基底関数とよぶ))

- 単純な基底関数を逐次的に複数組み合わせて，高い予測精度を達成する判別器を構成する方法：**ブースティング**

機械学習への応用①

・新たなデータ：$\boldsymbol{a} \in \mathbb{R}^d$が得られたとき，出力$b \in \{+1, -1\}$を以下の関数

$H_{\boldsymbol{x}}(\boldsymbol{a}) = \sum_{i=1}^{n} x_i h_i(\boldsymbol{a})$の符号で判定 → $H_{\boldsymbol{x}}(\boldsymbol{a}) > 0$なら$b = 1, H_{\boldsymbol{x}}(\boldsymbol{a}) < 0$なら$b = -1$

Q：$H_{\boldsymbol{x}}(\boldsymbol{a})$の係数$\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$を観測データからどのように定めるか？

→ $b_t H_{\boldsymbol{x}}(\boldsymbol{a}_t) > 0$なら，関数$H_{\boldsymbol{x}}(\boldsymbol{a})$で正しく判定できているので，できるだけ多くの$(\boldsymbol{a}_t, b_t)$に対して，$b_t H_{\boldsymbol{x}}(\boldsymbol{a}_t) > 0$が成り立つことが重要

・損失関数を導入：$f(\boldsymbol{x}) = \sum_{t=1}^{T} e^{-b_t H_x(\boldsymbol{a}_t)}$
→ これを最小化するときに，座標降下法を学習アルゴリズムとして用いる
(**アダブースト**(Adaboost))

機械学習への応用①

Q：どのようにして座標降下法を用いるか？

A：$f(\boldsymbol{x}) = \sum_{t=1}^{T} e^{-b_t H_x(\boldsymbol{a}_t)}$の勾配を計算する．(このとき$w_t = e^{-b_t H_x(\boldsymbol{a}_t)}$とする)

$$\frac{\partial f}{\partial x_i}(\boldsymbol{x}) = -\sum_{t=1}^{T} w_t b_t h_i(\boldsymbol{a}_t) = 2\sum_{t=1}^{T} w_t \mathbf{1}[b_t \neq \pm h_i(\boldsymbol{a}_t)] - \sum_{t=1}^{T} w_t$$

(ここで，$\mathbf{1}$[A]はAが真なら1，偽なら0をとる定義関数)

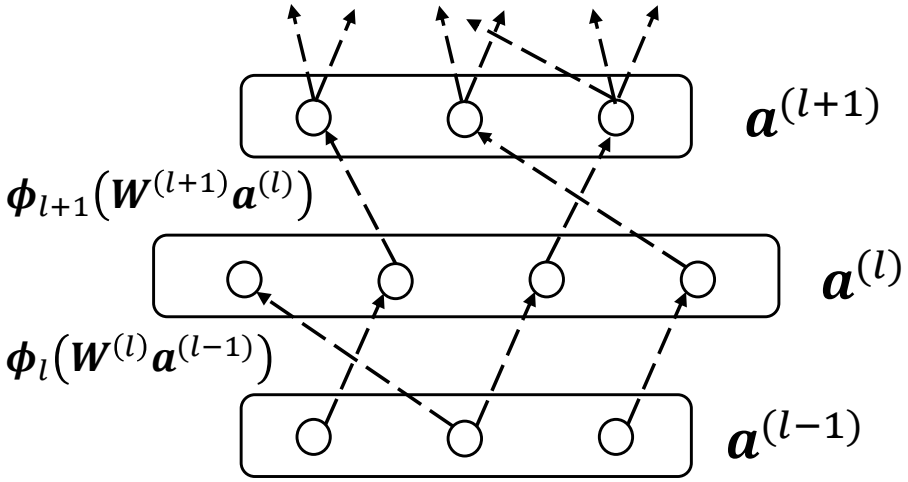
・よって，関数値が減少する座標軸の方向は，$\pm h_1, \pm h_2, \cdots, \pm h_n$の中で，

$\sum_{t=1}^{T} w_t \mathbf{1}[b_t \neq \pm h_i(\boldsymbol{a}_t)]$を最小にする方向に対応(重み付き誤り数と解釈可能)

(＝座標降下法と相性が良い)

機械学習への応用②

- **ニューラルネットワーク**(今回は**多層パーセプトロン**)
- N個のデータ：$(\boldsymbol{a}_1, \boldsymbol{b}_1), (\boldsymbol{a}_2, \boldsymbol{b}_2), \cdots, (\boldsymbol{a}_N, \boldsymbol{b}_N)$ において，$\boldsymbol{a}_i$と$\boldsymbol{b}_i$の関係を学習
  → 新たなデータ$\boldsymbol{a}$が入力されたときに，出力$\boldsymbol{b}$を精度よく予測できるか？



機械学習への応用②

多層パーセプトロン

・学習するパラメータ：$\boldsymbol{W}^{(l)}, l = 1, 2, \dots, L$

・入力$\boldsymbol{a}$，出力$\boldsymbol{b}$の間の関係 ($\boldsymbol{a}^{(0)} = \boldsymbol{a}$として)

$$\boldsymbol{a}^{(l)} = \boldsymbol{\phi}_l\left(\boldsymbol{W}^{(l)}\boldsymbol{a}^{(l-1)}\right), l = 1,2, \dots, L,$$

$$\boldsymbol{b} = \boldsymbol{a}^{(L)} = \boldsymbol{\phi}_L\left(\boldsymbol{W}^{(L)}\boldsymbol{\phi}_{L-1}\left(\boldsymbol{W}^{(L-1)} \cdots \boldsymbol{W}^{(2)}\boldsymbol{\phi}_1\left(\boldsymbol{W}^{(1)}\boldsymbol{a}^{(0)}\right)\cdots\right)\right) = \boldsymbol{\Phi}(\boldsymbol{a}, \boldsymbol{W})$$

$$(\boldsymbol{W} = \left(\boldsymbol{W}^{(1)}, \dots, \boldsymbol{W}^{(L)}\right)$$

ここで，$\boldsymbol{\phi}_l(\boldsymbol{z}) = \left(\psi(z_1), \psi(z_2), \dots, \psi\left(z_{D_l}\right)\right) = \left(\tanh(z_1), \tanh(z_2), \dots, \tanh\left(z_{D_l}\right)\right)$
などが用いられる.

機械学習への応用②

・入出力における2乗誤差を最小化する

$$f(\boldsymbol{W}) = \frac{1}{2N}\sum_{m=1}^{N} \|\boldsymbol{\Phi}(\boldsymbol{a}_m, \boldsymbol{W}) - \boldsymbol{b}_m\|^2$$

・探索方向を求めるためには，勾配の計算が必要

→ 誤差を出力層(最後の方)から入力層に伝搬させているように解釈できる
(**誤差逆伝搬法**(バック・プロパゲーション))

・$\boldsymbol{u}^{(l+1)} = \boldsymbol{W}^{(l+1)}\boldsymbol{a}^{(l)} = \boldsymbol{W}^{(l+1)}\boldsymbol{\phi}_l\left(\boldsymbol{u}^{(l)}\right)$を設定

機械学習への応用②

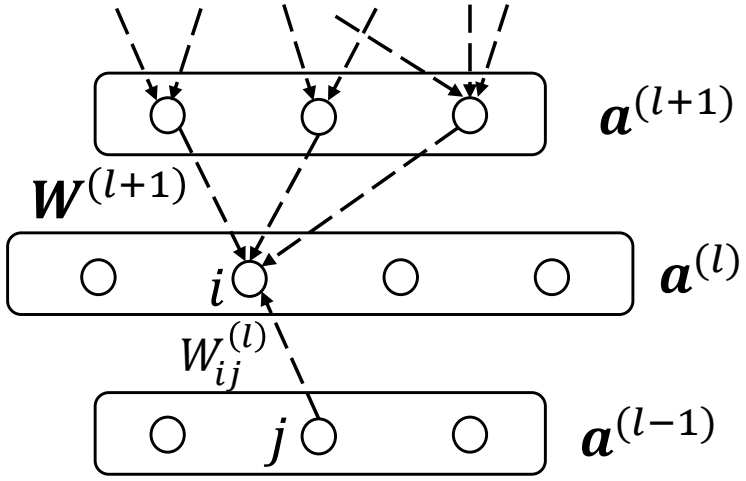
・次の微分を考える ($\boldsymbol{u}^{(l+1)} = \boldsymbol{W}^{(l+1)}\boldsymbol{\phi}_l\left(\boldsymbol{u}^{(l)}\right)$を利用して)

$$\frac{\partial \Phi_d}{\partial u_i^{(l)}} = \sum_k \frac{\partial \Phi_d}{\partial u_k^{(l+1)}} \frac{\partial u_k^{(l+1)}}{\partial u_k^{(l)}} = \sum_k \frac{\partial \Phi_d}{\partial u_k^{(l+1)}} W_{ki}^{(l+1)} \psi'\left(u_i^{(l)}\right)$$

つまり，$l+1$層目の微分からl層目の微分が求められる.

・また，$\boldsymbol{u}^{(l)} = \boldsymbol{W}^{(l)}\boldsymbol{a}^{(l-1)}$も利用すると

$$\frac{\partial \Phi_d}{\partial W_{ij}^{(l)}} = \sum_k \frac{\partial \Phi_d}{\partial u_k^{(l)}} \frac{\partial u_k^{(l)}}{\partial W_{ij}^{(l)}} = \frac{\partial \Phi_d}{\partial u_i^{(l)}} a_j^{(l-1)}$$



機械学習への応用②

・以上の結果から

$$\frac{\partial f}{\partial W_{ij}^{(l)}}(\boldsymbol{W}) = \frac{1}{N}\sum_{m=1}^{N}\sum_{d=1}^{D}\left(\Phi_d(\boldsymbol{a}_m, \boldsymbol{W}) - b_{m,d}\right)\frac{\partial \Phi_d}{\partial W_{ij}^{(l)}}(\boldsymbol{a}_m, \boldsymbol{W})$$

をfの最適化に用いる.

・確率的最適化のアイディアを用いると，$W_{ij}^{(l)}$は以下のように表記できる

$$W_{ij}^{(l)} \leftarrow W_{ij}^{(l)} - \varepsilon_t \cdot \sum_{d=1}^{D}\left(\Phi_d(\overline{\boldsymbol{a}}_t, \boldsymbol{W}) - \overline{b}_{t,d}\right)\frac{\partial \Phi_d}{\partial W_{ij}^{(l)}}(\overline{\boldsymbol{a}}_t, \boldsymbol{W})$$

(ここで，$(\overline{\boldsymbol{a}}_t, \overline{\boldsymbol{b}}_t)$はtステップ目で与えられたデータ，$\varepsilon_t$は適当な正の数)

今回のまとめ

- 勾配法の基礎として，凸2次関数を例とした最急降下法の収束速度を考察
  → 最急降下法を1次収束であるため，最適解付近ではやはり収束が鈍化
  → ニュートン法や準ニュートン法の利用
- 勾配法の応用として，アダブーストと座標降下法との関わり，多層パーセプトロンと勾配法との関わりを紹介(詳細までは踏み込まない予定．詳細を知りたい方は，それぞれの専門書を参照すること)
  → 先端手法の影の主役は，実は最適化