機械学習とデータマイニングの基礎（大阪大学）

オンライン学習法の汎化

Matthew J. Holland

大阪大学 産業科学研究所

大阪大学
OSAKA UNIVERSITY

目次

目次

1. 今回の趣旨
2. オンライン学習と集中不等式
3. オンラインからバッチへ
4. バッチ学習法の導出
5. 確率勾配法との関係
6. まとめ

今回の趣旨

問題領域がふたつ

標本外の汎化

$$\mathsf{L}(h) \mapsto \mathrm{R}(h) \in \mathbb{R} \tag{1}$$

（典型例：$\mathrm{R}(h) = \mathbf{E}_{\mu}\,\mathsf{L}(h)$）

逐次損失の累積

$$(\ell(h_1; z_1), \ldots, \ell(h_T; z_T)) \mapsto \mathrm{R}_T(h) \in \mathbb{R} \tag{2}$$

（典型例：$\mathrm{R}_T(h_1, \ldots, h_T) = \sum_{t=1}^{T} \ell(h_t; z_t)$）

今回の趣旨

この標本外の汎化と逐次損失の累積の「架け橋」となる技法を見ていく．

- 汎用的な技法として「online-to-batch (OTB) 変換」として知られる．
- ここで「online」とは「逐次損失の累積」問題における学習法のこと．
（以降，オンライン学習法と呼ぶ）
- 一方の「batch」とは，「標本外の汎化」問題における学習法のこと．
（以降，バッチ学習法と呼ぶ）

ざっくり，「良いオンライン学習法から良いバッチ学習法を導き出す方法」を学ぶ．

目次

オンライン学習の概要

オンライン学習法のアウトプット

$$(z_1, z_2, \ldots) \mapsto (h_1, h_2, \ldots) \tag{3}$$

オンライン学習法の性能保証

$$\mathrm{Regret}_T \coloneqq \mathrm{R}_T - \mathrm{R}_T^* = \sum_{t=1}^{T} (\ell(h_t; z_t) - \ell(h^*; z_t)) \leq \mathsf{Bound}(T) \tag{4}$$

基本的に欲しいのは，$\mathsf{Bound}(T) = o(T)$ である（“sub-linear regret” という）．

オンライン学習の概要

学習データの不確実性を加味する

$$(\mathsf{Z}_1, \mathsf{Z}_2, \ldots) \mapsto (\mathsf{H}_1, \mathsf{H}_2, \ldots) \tag{5}$$

標本外の汎化との接点（直感）

以下の近似が良いなら助かるが，無条件ではあり得ない．

$$\mathrm{R}_T(\mathsf{H}_1, \ldots, \mathsf{H}_T) \approx \sum_{t=1}^{T} \mathrm{R}(\mathsf{H}_t) \tag{6}$$

左辺は学習データのみに依存．右辺は評価用データ $\mathsf{Z} \sim \mu$ にも依存．

新たな集中不等式の出番

データをめぐる仮定

$Z_1, Z_2, \ldots$ および Z はいずれも μ からの独立標本であるとする．

問題点

$\ell(H_1; Z_1), \ldots, \ell(H_T; Z_T)$ は独立ではない $\Longrightarrow$ 従来の Hoeffding は使えない

解決策の案

H_t は $\mathbf{Z}_{t-1}$ のみに依存し，Z_t 以降からは独立．$\rightarrow$ 条件つき独立性を活用しよう．

新たな集中不等式の出番

条件つき独立性の活用

まず，(6) の差分の部分和を以下のように表記する．

$$M_t := \sum_{i=1}^{t} \mathrm{R}(H_i) - \mathrm{R}_t(H_1, \ldots, H_t) \tag{7}$$

その差の条件つき期待値をとってみる．

$$\begin{aligned}
\mathbf{E}\left[M_t \mid \mathbf{Z}_{t-1}\right] &= \sum_{i=1}^{t} \left(\mathbf{E}\left[\mathrm{R}(H_i) \mid \mathbf{Z}_{t-1}\right] - \mathbf{E}\left[\ell(H_i; Z_i) \mid \mathbf{Z}_{t-1}\right]\right) \\
&= \sum_{i=1}^{t} \left(\mathbf{E}\left[\mathrm{R}(H_i) \mid \mathbf{Z}_{i-1}\right] - \mathbf{E}\left[\ell(H_i; Z_i) \mid \mathbf{Z}_{i-1}\right]\right) \\
&= \sum_{i=1}^{t} \left(\mathbf{E}\left[\mathrm{R}(H_i) \mid \mathbf{Z}_{i-1}\right] - \mathbf{E}\left[\mathrm{R}(H_i) \mid \mathbf{Z}_{i-1}\right]\right) \\
&= 0
\end{aligned}$$

新たな集中不等式の出番

部分和の条件つき期待値

$$\mathbf{P}\left\{\mathbf{E}\left[M_t \mid \mathbf{Z}_{t-1}\right] = 0\right\} = 1, \quad \forall t \in [T] \tag{8}$$

このような M_t を構成する R と ℓ の差分を martingale differences という．

マルチンゲール版の集中不等式[1]

確率 1 で $|\mathrm{R}(H_t) - \ell(H_t; Z_t)| \leq c$ が成り立つとする（$t \in [T]$）．
このとき，以下の不等式が成り立つ．

$$\mathbf{P}\left\{M_T > \varepsilon\right\} \leq \exp\left(\frac{-2\varepsilon^2}{Tc^2}\right) \tag{9}$$

[1] Cesa-Bianchi and Lugosi [2, Lem. A.7] を参照．Hoeffding-Azuma の不等式と呼ばれる．

目次

オンラインからバッチへ

先ほどの (9) から，

$$\mathrm{M}_T = \sum_{t=1}^{T} \mathrm{R}(\mathsf{H}_t) - \mathrm{R}_T(\mathsf{H}_1, \ldots, \mathsf{H}_T) \leq c\sqrt{\frac{T}{2}\log\left(\frac{1}{\delta}\right)} \tag{10}$$

という上界が確率 $1-\delta$ 以上で成り立つ．

言い換えれば，以下のこともわかる．

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\mathrm{R}(\mathsf{H}_t) &\leq \frac{\mathrm{R}_t(\mathsf{H}_1,\ldots,\mathsf{H}_t)}{T} + c\sqrt{\frac{1}{2T}\log\left(\frac{1}{\delta}\right)} \\ &= \frac{\mathrm{Regret}_T}{T} + \frac{1}{T}\sum_{t=1}^{T}\ell(h^*;\mathsf{Z}_t) + c\sqrt{\frac{1}{2T}\log\left(\frac{1}{\delta}\right)} \end{aligned} \tag{11}$$

オンラインからバッチへ

さらに，従来の Hoeffding の不等式（one-sided 版）から以下のこともわかる．

$$\frac{1}{T}\sum_{t=1}^{T}\ell(h^*;\mathsf{Z}_t) - \mathrm{R}(h^*) \leq c\sqrt{\frac{1}{2T}\log\left(\frac{1}{\delta}\right)} \tag{12}$$

これも確率 $1-\delta$ 以上である．

ランダム事象の整理

ここで 2 つの事象を抱えている．

- 不等式 (11) は確率 $1-\delta$ 以上で成り立つ．
- 不等式 (12) は確率 $1-\delta$ 以上で成り立つ．

同じ事象とは限らないので，union bound を使う．

オンラインからバッチへ

Online-to-batch の起点 [2]

- (11) と (12) の両方が成り立つことを望んでいる．
- これが起こらないことは，少なくともどちらかが不成立だったことを意味する．
- 従って union bound より，$1-2\delta$ の確率で以下が成り立つ（係数に注目）．

$$\frac{1}{T}\sum_{t=1}^{T}\mathrm{R}(\mathsf{H}_t) - \mathrm{R}(h^*) \leq \frac{\mathrm{Regret}_T}{T} + c\sqrt{\frac{2}{T}\log\left(\frac{1}{\delta}\right)} \tag{13}$$

バッチ変換するには

まだ $(\mathsf{H}_1,\ldots,\mathsf{H}_T)$ に対する保証しかない… 一つの候補に絞りたい．

[2] これに至る論法は Cesa-Bianchi et al. [1] が開拓した．

目次

バッチ学習法の導出

変換の方策（凸性がある場合）

- $\mathcal{H}$ は凸集合である．
- $\mathrm{R}(\cdot)$ は凸関数である．

上記の 2 つの条件の下，合理的な変換はすぐに見つかる．

$$\overline{\mathsf{H}}_T := \frac{1}{T}\sum_{t=1}^{T}\mathsf{H}_t \tag{14}$$

$\mathcal{H}$ の凸性によって $\overline{\mathsf{H}}_T \in \mathcal{H}$ が担保される上に，以下も成り立つ（確率 $1-2\delta$）．

$$\mathrm{R}(\overline{\mathsf{H}}_T) \leq \frac{1}{T}\sum_{t=1}^{T}\mathrm{R}(\mathsf{H}_t) \leq \mathrm{R}(h^*) + \frac{\mathrm{Regret}_T}{T} + c\sqrt{\frac{2}{T}\log\left(\frac{1}{\delta}\right)}$$

バッチ学習法の導出

変換の方策（無作為に選ぶ）[3]

$$\mathsf{T} \sim \mathsf{Uniform}\{1,\ldots,T\} \text{ として，} \mathsf{H}_{\mathsf{T}} \text{を採る} \tag{15}$$

- 利点：$\mathrm{R}(\cdot)$ や $\mathcal{H}$ の凸性を必要としない．
- 難点：平均的な保証にとどまることが多い．
- 難点：T は学習前から決める必要がある（対策はいろいろあるが）．

[3]非凸関数の確率的最適化でも重宝される random stopping の一例である（Ghadimi and Lan [3]）．

バッチ学習法の導出

この平均化によって，汎化誤差とリグレットの接点は明確である．

$$\begin{aligned}
\mathbf{E}\left[\mathrm{R}(\mathsf{H}_{\mathsf{T}}) - \mathrm{R}(h^*)\right] &= \mathbf{E}\left(\mathbf{E}\left[\mathrm{R}(\mathsf{H}_{\mathsf{T}}) - \mathrm{R}(h^*) \,|\, \mathbf{Z}_T\right]\right) && (16)\\
&= \frac{1}{T}\sum_{t=1}^{T}\mathbf{E}\left[\mathrm{R}(\mathsf{H}_t) - \mathrm{R}(h^*)\right] && (17)\\
&= \frac{1}{T}\sum_{t=1}^{T}\mathbf{E}\left(\mathbf{E}\left[\ell(\mathsf{H}_t;\mathsf{Z}_t) - \ell(h^*;\mathsf{Z}_t) \,|\, \mathbf{Z}_{t-1}\right]\right) && (18)\\
&= \frac{1}{T}\mathbf{E}\left[\sum_{t=1}^{T}\left(\ell(\mathsf{H}_t;\mathsf{Z}_t) - \ell(h^*;\mathsf{Z}_t)\right)\right] && (19)\\
&= \frac{\mathbf{E}\left[\mathrm{Regret}_T\right]}{T} && (20)
\end{aligned}$$

バッチ学習法の導出

変換の方策（データをかけた検証）

偶数 T 個のデータ $(\mathsf{Z}_1,\ldots,\mathsf{Z}_T)$ の半分を検証用に使う．

$$\mathsf{R}_{\mathsf{val}}(h) := \frac{2}{T}\sum_{t=T+1}^{2T}\ell(h;\mathsf{Z}_t) \tag{21}$$

この検証用の目的関数をもって，最良の候補を選定する．

$$\star \in \operatorname*{arg\,min}_{t \leq T/2} \mathsf{R}_{\mathsf{val}}(\mathsf{H}_t) \tag{22}$$

この学習法は $\mathsf{H}_\star$ を出力する．

バッチ学習法の導出（データをかけた検証）

直感

検証用のデータが十分にあれば $\mathsf{R}_{\mathrm{val}}(\cdot)$ に強い候補は $\mathrm{R}(\cdot)$ にも強い（はず）．

議論の入り口

各指標における最良の候補を整理しておく．

$$\mathsf{H}_\star \in \operatorname*{arg\,min}_{h \in \{\mathsf{H}_1,\ldots,\mathsf{H}_{T/2}\}} \mathsf{R}_{\mathrm{val}}(h), \qquad \mathsf{H}^\ast \in \operatorname*{arg\,min}_{h \in \{\mathsf{H}_1,\ldots,\mathsf{H}_{T/2}\}} \mathrm{R}(h) \tag{23}$$

これを踏まえて，入り口が見えてくる．

$$\mathrm{R}(\mathsf{H}_\star) - \mathrm{R}(h^\ast) = \mathrm{R}(\mathsf{H}_\star) - \mathrm{R}(\mathsf{H}^\ast) + \mathrm{R}(\mathsf{H}^\ast) - \mathrm{R}(h^\ast) \tag{24}$$

$$\leq \mathrm{R}(\mathsf{H}_\star) - \mathrm{R}(\mathsf{H}^\ast) + \underbrace{\frac{2}{T}\sum_{t=1}^{T/2} \mathrm{R}(\mathsf{H}_t) - \mathrm{R}(h^\ast)}_{\text{see (13)}} \tag{25}$$

バッチ学習法の導出（データをかけた検証）

前半のデータを条件として，オンライン学習法からの $(\mathsf{H}_1,\ldots,\mathsf{H}_{T/2})$ は固定値になる．

$$\mathbf{E}\left[\mathsf{R}_{\mathrm{val}}(\mathsf{H}_t) \,|\, \mathbf{Z}_{T/2}\right] = \mathrm{R}(\mathsf{H}_t), \quad \forall t \in [T/2] \tag{26}$$

従って，Hoeffding（one-sided 版）はそのまま使える．

$$\mathbf{P}\left\{ \mathrm{R}(\mathsf{H}_t) - \mathsf{R}_{\mathrm{val}}(\mathsf{H}_t) \leq c\sqrt{\frac{1}{T}\log\left(\frac{1}{\delta}\right)} \,\middle|\, \mathbf{Z}_{T/2} \right\} \geq 1 - \delta \tag{27}$$

同様に，$\mathsf{H}^\ast$ も固定になるので，以下のバウンドも成り立つ．

$$\mathbf{P}\left\{ \mathsf{R}_{\mathrm{val}}(\mathsf{H}^\ast) - \mathrm{R}(\mathsf{H}^\ast) \leq c\sqrt{\frac{1}{T}\log\left(\frac{1}{\delta}\right)} \right\} \geq 1 - \delta \tag{28}$$

バッチ学習法の導出（データをかけた検証）

先ほど用意した集中不等式 (27)–(28) を駆使していこう．

$$\mathrm{R}(\mathsf{H}_\star) - \mathrm{R}(\mathsf{H}^\ast) = \mathrm{R}(\mathsf{H}_\star) - \mathsf{R}_{\mathrm{val}}(\mathsf{H}_\star) + \mathsf{R}_{\mathrm{val}}(\mathsf{H}_\star) - \mathrm{R}(\mathsf{H}^\ast) \tag{29}$$

$$\leq \max_{t \leq T/2} \left(\mathrm{R}(\mathsf{H}_t) - \mathsf{R}_{\mathrm{val}}(\mathsf{H}_t)\right) + \mathsf{R}_{\mathrm{val}}(\mathsf{H}^\ast) - \mathrm{R}(\mathsf{H}^\ast) \tag{30}$$

$$\leq c\sqrt{\frac{1}{T}\log\left(\frac{T}{2\delta}\right)} + c\sqrt{\frac{1}{T}\log\left(\frac{1}{\delta}\right)} \tag{31}$$

$$\leq 2c\sqrt{\frac{1}{T}\log\left(\frac{T}{2\delta}\right)} \tag{32}$$

上記は $1 - 2\delta$ 以上の確率で成り立つ．

バッチ学習法の導出（データをかけた検証）

これまでの議論を整理しておく．誤差の水準を

$$\varepsilon_{T/2} := 2c\sqrt{\frac{1}{T}\log\left(\frac{T}{2\delta}\right)}$$

とおいたとき，前掲の (32) から

$$\mathbf{P}\left\{\mathrm{R}(\mathsf{H}_\star) - \mathrm{R}(\mathsf{H}^\ast) \leq \varepsilon_{T/2}\right\} = \mathbf{E}\left[\mathbf{P}\left\{\mathrm{R}(\mathsf{H}_\star) - \mathrm{R}(\mathsf{H}^\ast) \leq \varepsilon_{T/2} \,|\, \mathbf{Z}_{T/2}\right\}\right] \tag{33}$$

$$\geq \mathbf{E}\left[1 - 2\delta\right] \tag{34}$$

$$= 1 - 2\delta \tag{35}$$

これを (25) と (13) と一緒に使う．

バッチ学習法の導出（データをかけた検証）

確率 $1-4\delta$ 以上で，以下のバウンドが成り立つ．

$$\mathrm{R}(\mathsf{H}_\star) - \mathrm{R}(h^*) \leq 2c\sqrt{\frac{1}{T}\log\left(\frac{T}{2\delta}\right)} + \frac{2\,\mathrm{Regret}_{T/2}}{T} + 2c\sqrt{\frac{1}{T}\log\left(\frac{1}{\delta}\right)} \tag{36}$$

$$\leq 4c\sqrt{\frac{1}{T}\log\left(\frac{T}{2\delta}\right)} + \frac{2\,\mathrm{Regret}_{T/2}}{T} \tag{37}$$

重要な知見
「強いオンライン学習法 $\Longrightarrow$ 良い標本外の汎化性能」と言える．

目次

確率勾配法との関係

確率勾配法（いわゆる one-pass SGD）
学習データ $(\mathsf{Z}_1, \mathsf{Z}_2, \ldots)$ に対して，従来の確率勾配法とは

$$\mathsf{H}_{t+1} = \mathsf{H}_t - \alpha_t \nabla\ell(\mathsf{H}_t; \mathsf{Z}_t) \tag{38}$$

という更新則によって特徴づけられる．

- この損失関数 ℓ を「本命」と位置づける．
- H_1 は適当に初期化する．
- $\alpha_t \geq 0$ は step size（学習率とも呼ばれる）．
- この (38) による $(\mathsf{H}_1, \mathsf{H}_2, \ldots)$ はオンライン学習法の出力にほかならない．

確率勾配法との関係

バッチ学習法の導出（凸関数の場合）
もし $h \mapsto \ell(h; z)$ が凸関数ならば，明確な切り口が見えてくる．

$$\mathrm{R}(\mathsf{H}_t) - \mathrm{R}(h^*) = \mathbf{E}_\mu\left[\ell(\mathsf{H}_t; \mathsf{Z}) - \ell(h^*; \mathsf{Z})\right] \tag{39}$$

$$\leq \mathbf{E}_\mu\left[\langle \nabla\ell(\mathsf{H}_t; \mathsf{Z}), \mathsf{H}_t - h^* \rangle\right] \tag{40}$$

$$= \mathbf{E}_\mu\langle \nabla\ell(\mathsf{H}_t; \mathsf{Z}), \mathsf{H}_t \rangle - \mathbf{E}_\mu\langle \nabla\ell(\mathsf{H}_t; \mathsf{Z}), h^* \rangle \tag{41}$$

$$= \langle g(\mathsf{H}_t), \mathsf{H}_t \rangle - \langle g(\mathsf{H}_t), h^* \rangle \tag{42}$$

ただし，ここで $g(h) \coloneqq \mathbf{E}_\mu \nabla\ell(h; \mathsf{Z})$ と定義している．

この右辺にあるのは，いわゆる linear loss の期待値の差．

確率勾配法との関係

さらにこの論法を突き詰めていく．$g(\cdot)$ に対して $\mathsf{G}_t := \nabla\ell(\mathsf{H}_t;\mathsf{Z}_t)$ とおく．

$$\langle g(\mathsf{H}_t),\mathsf{H}_t\rangle - \langle g(\mathsf{H}_t),h^*\rangle = \langle g(\mathsf{H}_t),\mathsf{H}_t - h^*\rangle \tag{43}$$

$$= \underbrace{\langle g(\mathsf{H}_t) - \mathsf{G}_t,\mathsf{H}_t - h^*\rangle}_{\text{martingale diff.}} + \langle \mathsf{G}_t,\mathsf{H}_t - h^*\rangle \tag{44}$$

したがって，

$$\sum_{t=1}^{T}[\mathrm{R}(\mathsf{H}_t) - \mathrm{R}(h^*)] \leq \underbrace{\sum_{t=1}^{T}\langle g(\mathsf{H}_t) - \mathsf{G}_t,\mathsf{H}_t - h^*\rangle}_{\text{use Hoeffding-Azuma}} + \underbrace{\sum_{t=1}^{T}\langle \mathsf{G}_t,\mathsf{H}_t - h^*\rangle}_{\text{linear }\mathrm{Regret}_T} \tag{45}$$

つまり OTB 変換の道筋が見えてきた．

確率勾配法との関係

これまでの結果に結びつける

$|\langle \mathsf{G}_t,\mathsf{H}_t - h^*\rangle| \leq \|\mathsf{H}_t - h^*\|\|\mathsf{Z}_t\| \leq c/2$ と仮定する

$\downarrow$

linear loss 版の (13) がそのまま通ずる（OTB 変換は可能）

$\downarrow$

更新則 (38) の特性を使って linear Regret_T を抑える

linear regret の上界を得る $\Longrightarrow$ 本命の期待損失の上界を得る

目次

まとめ

- オンライン学習法は原則として「未来」のデータが見られないため，被る損失値の間には条件つき独立性が生じやすい（IID データの場合）．
- 条件つき独立性を駆使する Hoeffding-Azuma の不等式が登場．
- 逐次損失の和が高い確率で「集中」するのは，その期待値の和である．
- この近似を活かして，期待損失の和を Regret_T などで抑えることが可能．
- OTB 変換とは，このバウンドを前提として「期待損失の和」以下の期待損失を持つ候補を一つ探求する作業（平均化，検証など）．
- オンライン学習法としての確率勾配法とその標本外の汎化の論法も確認した．

参考文献

[1] Cesa-Bianchi, N., Conconi, A., and Gentile, C. (2004). On the generalization ability of on-line learning algorithms. *IEEE Transactions on Information Theory*, 50(9):2050–2057.

[2] Cesa-Bianchi, N. and Lugosi, G. (2006). *Prediction, Learning, and Games.* Cambridge University Press.

[3] Ghadimi, S. and Lan, G. (2013). Stochastic first- and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368.