1　線形基底関数モデル

1.1　逐次学習

バッチ手法とオンライン学習の違いを理解する。読み合わせ（以下箇条書きを口頭確認する）。

・バッチ手法；全ての訓練データを一度に処理してモデルのパラメータを構築する（例. 最尤推定）。

・オンライン学習；新しく入ってきたデータ点 1 つだけを用いてモデルのパラメータを更新する（例. SGD）。

念のため（3.23）を導出しておく。

1.2　正則化最小二乗法

過学習を防ぐための手法である「誤差関数に正則化項を加える」手法を学ぶ。まず、正則化項を加えると、パラメタ $\boldsymbol{w}$ が小さくなることを 1.1 節の議論によってリマインドする（Ref. p.7-10）。続いて、正則化関数の最小化を考える意義について理解する。最後に、「正則化項を加える」ことが正則化していない二乗和誤差に与える影響について、Ridge 回帰と Lasso 回帰のそれぞれの場合において幾何学的な考察を行う。

リマインドパートは口頭で確認するのみ。

一般的な正則化誤差関数の min 化を考えるにあたって、不等式制約下でのラグランジュ未定乗数法の使い方について学ぶ必要がある（web サイト参照）。ポイントは元の関数の停留値（今回は $\boldsymbol{w}$）が制約式の範囲内に入っているか、否かということになる。

このポイントについて、今から紹介するお碗と紐の例で理解を深めたい。初めの例ではお椀の底（元の関数の極小値を与える停留点）が紐の内部（制約式の範囲内）に入っているため、実は制約式を設けても設けなくても停留値は変化しないことがわかる。次の例ではお椀の底（元の関数の極小値を与える停留点）が紐の内部[*1]（制約式の範囲内）に入っていないため、停留値は制約式の境界上に存在することになる。この場合は制約条件を与えることが停留値の選択に影響を与えていると言える。まとめると

・元の関数の停留値が制約不等式の範囲内ならば、制約条件を設けた場合と設けなかった場合で停留値は変化しない、つまり元の関数の停留値問題を解けばいい。

・元の関数の停留値が制約不等式の範囲外ならば、制約条件を設けた場合、つまり制約式の境界上（等号成立）で、停留値を考えれば良い。もっと言うと従来学んできたラグランジュ未定乗数法を適用すれば良いと言うことである。

以上の説明をもとに、正則化誤差関数（3.29）の min 化について考える（対応＞演習 3.5；Go to 頁 4[*2]）。

次に Ridge 回帰（q=2；境界が円）と Lasso 回帰（q=1；境界が正方形）について、それぞれに対応する正則化項を元の誤差関数に加えることが、停留条件にどのような影響を与えるか図 3.4 を用いて理解しよう（口頭で済ます）。

[*1] 初めの例と見た目が大きく異なっている。紐はビヨーンとなっているように見えるのだが、ちゃんと見れば $x_1^2 + x_2^2 = 1$ を満たしていることが確かめられる。見た目でビビらないように注意。

[*2] 編集の都合上、ページが飛んでしまうが勘弁。

1.3　出力変数が多数の場合

多次元の目標変数（目標ベクトル）を推定する手法として、共通の基底関数を用いることで、目標ベクトルの全ての要素に対してモデル化する手法を学ぶ。

ここよくわかっていない[*3]。

2　バイアスバリアンス分解

決定理論（ベイズ的なアプローチ）において、事後期待損失を最小化することが最適なパラメータを求めることであることは既に学んだ。ここでは事後期待損失の式を変形し「バイアス項」と「バリアンス項」と呼ばれる項に分解する。このバイアス項とバリアンス項の大きさのバランスによって、モデルの予測性能の良さ[*4]が決定することが知られており、これらのトレードオフ関係について、線形基底関数モデルを例にとって理解する。

まず、事後期待損失を（3.37）の形で得たい（Ref.（1.90））。最小化すべきは第一項となる（∵データ $y(\boldsymbol{x})$ に直接依存している）。第二項はノイズと呼ばれ、事後期待損失の最小値となりうる。

今、（3.37）内の $(y(\boldsymbol{x}) - h(\boldsymbol{x}))^2$ は当然、データ集合 D によって変わってしまう。そこで、この期待値を考え、その値を学習の性能としたい[*5]期待値の導出自体は非常に簡単なので口頭で済ますが、（3.40）について各項の名称を確認しておく。第一項がバイアス、第二項がバリアンスと呼ばれる。これら 2 つの量が何なのかは、このあと例を用いて説明する。このバイアス、バリアンス、そして先ほど出てきたノイズを用いることで、事後期待損失は（3.41）の形で表せる。ただし、（3.41-43）を満たす。

繰り返しになるが、我々は事後期待損失を min 化したい。ところが、バイアスとバリアンスはトレードオフの関係にあり、これらをバランス良く小さくすることが要求される[*6]。このバランスの取り方を三角関数のデータ生成モデルの例を用いて理解する。設定は教科書を参照されたい。

結論から言うと、図 3.5 の中でバリアンスとバイアスの大きさについてバランスが良いものは、すなわち性能の良い予測関数は中段のモデルであると考えられる[*7]。上段のモデルは、データセットによる予測関数の違いは生じにくいものの、真の sin 関数上からは大きく外れてしまっている。一方、下段のモデルは、（平均化された）予測関数が真の sin 関数上に乗っているものの、1 つのデータセットから作られるフィッティング曲線については真の sin 関数の振る舞いとは大きく異なってしまっている。図 3.5 のサブプロットごとに、バイアス-バリアンスの大小関係を表 1 にまとめる。

バイアスバリアンス分解の欠点は、観測データのセットが 1 組しか得られない時には適用できないことである。なぜなら、このような場合は、複数のデータセットの平均量を計算できなくなってしまうからである[*8]。いくつかのデータセットがあって、それらの平均が存在するときに初めてこの手法が使えるようになる。実際

[*3] （3.35）のありがたみがイマイチ理解できなかった。一緒に読み合わせで理解させてほしい。

[*4] 要するに汎化性能のことである。

[*5] そういった意味で「学習アルゴリズムの性能をデータ集合の取り方に関する平均の意味で評価する」と教科書は謳っているのである。この「平均的な云々かんぬん」という話は、2.8 節において「パラメタの事後分散の平均が事前分散より小さくなる」といったところでも出ていた。リマインドまで。

[*6] もっとも、バランスの良さについて定量的な指標を得ることはできないのだが...

[*7] 教科書に結論が書かれていないので、よくわからん。

[*8] 図 3.5 の下段左の曲線群の中で 1 本しか作れないということである。これでは真の sin 関数の振る舞いとはかけ離れてしまうことは自明であろう。

表 1　バイアス-バリアンス大小まとめ

図の位置	バイアス大小	バリアンス大小
上	大（回帰関数との GAP 大）	小（データ集合によらず変動小）
中	中（上と下の中庸）	中（上と下の中庸）
下	小（回帰関数との GAP 小）	大（データ集合によって変動大）

のところ、そんなにデータが得られる恵まれた環境は生じづらいということである。

次の節ではそのような制限下でも過学習を回避し、かつ、訓練データのみ[*9]でモデルの複雑さを自動的に決定できる方法を学ぶ。

3　ベイズ線形回帰

線形回帰モデルの作り方について、最尤推定の特徴や課題を振り返る。最尤推定は最も単純な手法であり尤度関数を最大化するというものである。課題としてコインの例でも見たように大きな過学習を引き起こしてしまうことがある。また、計算に使えるデータは限られているため、交差検証を用いて学習を行なっていくが、この手法は計算量が多くなるという課題もある[*10]。先の説の最後でも述べたように、本節で学ぶベイズ線形回帰は次のような特徴を持つ。

・過学習を起こさないこと

・テスト用データだけでモデルを作れること

3.1　パラメタの分布

線形回帰モデルのベイズ的取り扱いについて学ぶ。パラメタの事後分布を得たいが、まずは、モデルパラメータの事前分布 $\boldsymbol{w}$ を定義する。そして、適当な尤度を準備し、事後分布を計算すれば良い。しかしながら、ここではベイズの公式を用いることで、事後分布を直接求めることができる。従い、事前分布と尤度の積を計算した後に、正規化を行う必要はない。

では、実際に事前分布（3.48）と尤度関数（3.10）を掛けて、さらにベイズの公式を用いて事後分布を求める計算を実行する（Go to 頁 5）。

次に、データ点が逐次的に与えられる場合[*11]について、既に N 個のデータが観測されているとしたときの事後分布の表式を考える。（対応＞演習 3.8 ; Go to 頁 6）

[*9] テスト用データをとっておく必要がないということである。本来データをテスト用にデータを割いてしまうことは、望ましくない。学習に使うデータは大変貴重なものであるからだ。

[*10] 1.3 節で学んだように、S 個のホールドアウト集合がある場合には、計算コストは S 倍となってしまう。

[*11] データが次々に更新されるときのモデルを構成することこそがベイズ線形回帰を考える目的であった。

準備として

$$F(w) = \frac{1}{2}\sum_{n=1}^{N}\{t_n - w^T\phi(x_n)\}^2,\quad G(w) = F(w) + \frac{\lambda}{2}\left(\sum_{j=1}^{M}|w_j|^q - \eta\right)$$

と定義する。

$F(w)$ を以下制約式

$$-\frac{1}{2}\left(\sum_{j=1}^{M}|w_j|^q - \eta\right) \geq 0 \quad (\because (3.30))$$

のもとで min 化する。

$$F(w) - \lambda\cdot\left\{-\frac{1}{2}\left(\sum_{j=1}^{M}|w_j|^q - \eta\right)\right\}$$

$$= F(w) + \frac{\lambda}{2}\left(\sum_{j=1}^{M}|w_j|^q - \eta\right) = G(w)$$

$G(w)$ において

$\frac{\partial}{\partial w}\left(\frac{\lambda\eta}{2}\right) = 0$ なので $G(w)$ の min 化は

$F(w) + \frac{\lambda}{2}\sum_{j=1}^{M}|w_j|^q$、つまり (3.29) の min 化に等しい。

$F(w)$ を上記制約式で min 化する問題を考えているから、制約式の右辺を $g(w)$ とすると KKT 条件 (E.9-11) が成立する。(※ → w と読みかえよ。)

特に条件 (E.11) は

$$-\frac{\lambda}{2}\left(\sum_{j=1}^{M}|w_j|^q - \eta\right) = 0$$

$$\eta = \sum_{j=1}^{M}|w_j|^q$$

となり、この等式が $F(w)$ を min 化する w^* で成立する。

つまり

$$\eta = \sum_{j=1}^{M}|w_j^*|^q$$

となる。

※等号条件であることが、ヒモとおわんの例で説明した「ヒモ上で停留値が与えられる」ことに対応している。

Back to 頁1

$$p(w|t) \sim p(w)\,p(t|w)$$

$$= \prod_{n=1}^{N} \mathcal{N}(t_n \mid w^T\phi(x_n), \beta^{-1})\ \mathcal{N}(w \mid m_0, S_0) \quad (\because (3.10), (3.48))$$

$$\propto \exp\left(-\frac{\beta}{2}(t_1 - w^T\phi(x_1))^2\right)\cdots\exp\left(-\frac{\beta}{2}(t_N - w^T\phi(x_N))^2\right)$$
$$\exp\left(-\frac{1}{2}(w-m_0)^T S_0^{-1}(w-m_0)\right)$$

$$= \exp\left\{-\frac{\beta}{2}\left((t_1 - w^T\phi(x_1))^2 + \cdots + (t_N - w^T\phi(x_N))^2\right)\right\}\exp\left(-\frac{1}{2}(w-m_0)^T S_0^{-1}(w-m_0)\right)$$

$$= \exp\left\{-\frac{\beta}{2}\begin{pmatrix} t_1 - w^T\phi(x_1) \\ \vdots \\ t_N - w^T\phi(x_N)\end{pmatrix}^T\begin{pmatrix} t_1 - w^T\phi(x_1) \\ \vdots \\ t_N - w^T\phi(x_N)\end{pmatrix}\right\}\exp\left(-\frac{1}{2}(w-m_0)^T S_0^{-1}(w-m_0)\right)$$

$$= \exp\left\{-\frac{\beta}{2}\left(t - \begin{pmatrix}\phi^T(x_1)w \\ \vdots \\ \phi^T(x_N)w\end{pmatrix}\right)^T\left(t - \begin{pmatrix}\phi^T(x_1)w \\ \vdots \\ \phi^T(x_N)w\end{pmatrix}\right)\right\}\exp\left(-\frac{1}{2}(w-m_0)^T S_0^{-1}(w-m_0)\right)$$

$(\because \phi_n^T(x_n)w$ はスカラ$)$

$$t = \begin{pmatrix} t_1 \\ \vdots \\ t_N\end{pmatrix}$$

$$= \exp\left\{-\frac{\beta}{2}(t - \Phi w)^T(t - \Phi w)\right\}\exp\left(-\frac{1}{2}(w-m_0)^T S_0^{-1}(w-m_0)\right)$$

$(\because (3.16))$

$$= \exp\left[-\frac{1}{2}\left\{\beta t^T t - \beta t^T\Phi w - \beta(\Phi w)^T t + \beta(\Phi w)^T\Phi w + w^T S_0^{-1} w - w^T S_0^{-1} m_0 - m_0^T S_0^{-1} w + m_0^T S_0^{-1} m_0\right\}\right]$$

($\beta w^T\Phi^T t$, $\beta w^T\Phi^T\Phi w$)

$$= \exp\left[-\frac{1}{2}\left\{w^T S_0^{-1} w + \beta w^T\Phi^T\Phi w - w^T S_0^{-1} m_0 - \beta w^T\Phi^T t - m_0^T S_0^{-1} w - \beta t^T\Phi w + m_0^T S_0^{-1} m_0 + \beta t^T t\right\}\right]$$

$$= \exp\left[-\frac{1}{2}\left\{w^T(S_0^{-1} + \beta\Phi^T\Phi)w - w^T(S_0^{-1} + \beta\Phi^T t) - (m_0^T S_0^{-1} + \beta t^T\Phi)w + m_0^T S_0^{-1} m_0 + \beta t^T t\right\}\right]$$

($S_N^{-1}S_N$)

$$= \exp\left[-\frac{1}{2}\left\{w^T S_N^{-1} w - w^T S_N^{-1} m_N - (m_N^T S_N^{-1} w)^T + m_N^T S_N^{-1} m_N - m_N^T S_N^{-1} m_N + m_0^T S_0^{-1} m_0 + \beta t^T t\right\}\right]$$

$$= \exp\left[-\frac{1}{2}(w-m_N)^T S_N^{-1}(w-m_N)\right]\exp\left\{-\frac{1}{2}\left(m_0^T S_0^{-1} m_0 - m_N^T S_N^{-1} m_N + \beta t^T t\right)\right\}$$

$$\propto \exp\left[-\frac{1}{2}(w-m_N)^T S_N^{-1}(w-m_N)\right] = \mathcal{N}(w \mid m_N, S_N)$$

Back to 頁3

分布 $p(w \mid t_{N+1})$，事前分布 $p(w)$，尤度 $p(t_{N+1} \mid x_{N+1}, w)$ と定義する。
分布と尤度は以下のように与えられる。

$w) = N(w \mid m_N, S_N)$　(∵ 今回の設定)

$t_{N+1} \mid x_{N+1}, w) = N(t_{N+1} \mid y(x_{N+1}, w), \beta^{-1})$　(∵ (3.10) でデータ1コ取り出し)

ベイズの公式より

$$\mid t_{N+1}) \sim p(t_{N+1} \mid x_{N+1}, w)\, p(w)$$

$$\sim N(t_{N+1} \mid y(x_{N+1}, w), \beta^{-1})\, N(w \mid m_N, S_N)$$

$$\sim \exp\left\{-\frac{1}{2}\beta(t_{N+1} - w^T\phi(x_{N+1}))^2\right\} \exp\left\{-\frac{1}{2}(w - m_N)^T S_N^{-1}(w - m_N)\right\}$$

$$= \exp\left\{-\frac{1}{2}\beta(t_{N+1}^2 - 2t_{N+1}w^T\phi(x_{N+1}) + w^T\phi(x_{N+1})\phi(x_{N+1})^T w\right)$$
$$-\frac{1}{2}(w^T S_N^{-1} w - w^T S_N^{-1} m_N - m_N^T S_N^{-1} w + m_N^T S_N^{-1} m_N)\Big\}$$

$$= \exp\left\{-\frac{1}{2} w^T(S_N^{-1} + \phi(x_{N+1})\phi(x_{N+1})^T) w - 2\beta t_{N+1} w^T\phi(x_{N+1})\right.$$
$$- w^T S_N^{-1} m_N - w^T S_N^{-1} m_N$$
$$\left. + \text{const.}\right\}$$　(∵ S_N^{-1} は対称、const. は w を含まない)

$$= \exp\left\{-\frac{1}{2} w^T(S_N^{-1} + \phi(x_{N+1})\phi(x_{N+1})^T) w - 2w^T(\beta t_{N+1}\phi(x_{N+1}) + S_N^{-1} m_N) + \text{const.}\right\}$$

$$S_{N+1}^{-1} = S_N^{-1} + \phi(x_{N+1})\phi(x_{N+1})^T$$

$$m_{N+1} = S_{N+1}(S_N^{-1} m_N + \beta\phi(x_{N+1}) t_{N+1})$$

$$S_{N+1}^{-1} m_{N+1} = S_N^{-1} m_N + \beta\phi(x_{N+1}) t_{N+1}$$

整理すると

$$\mid t_{N+1}) \sim \exp\left(-\frac{1}{2} w^T S_{N+1}^{-1} w - 2w^T S_{N+1}^{-1} m_{N+1} + \text{const.}\right)$$

$$\sim p(w \mid m_{N+1}, S_{N+1})$$