2008年度春季ゼミ 学習理論入門　担当：秋田

0　このゼミの概要

春季休業を利用して，学習理論の入門的な講義を 4 回行います．ベイズ推定の話を中心に以下のようなトピックスを扱います：

第 1 回（2/23） 多項式による曲線近似

第 2 回（3/02） 線形回帰

第 3 回（3/09） 分類問題

第 4 回（3/16） サポートベクタマシン入門

このゼミの教科書および参考書として以下のものを挙げておきます：

教科書: 平岡裕章, 応用数理 II 講義ノート（コピーを初回に配布予定）

参考書: Bishop, C. *Pattern Recognition and Machine Learning*, Springer, (2006)

また，簡単なレジュメのようなものを作成して配布する予定です.

1　多項式による曲線近似

今回および次回は N 個の既知データ $(x_1, t_1), \ldots, (x_N, t_N)$ があるときに，未知の $\hat{x}$ に対する出力 $\hat{t}$ を予測する問題を考察する.

今回は，既知データ (x_i, t_i) を近似する多項式 $y(x, \boldsymbol{w}) = w_0 + w_1 x + w_2 x^2 + \cdots + w_M x^M$ の係数を推定して，未知の $\hat{x}$ に対する出力 $\hat{t}$ を $\hat{t} = y(\hat{x}, \boldsymbol{w})$ と予測することを考えよう.

1.1　最小二乗法

方針：エラー関数

$$\mathrm{Err}(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}(y(x_n, \boldsymbol{w}) - t_n)^2$$

を最小にするような$\boldsymbol{w}$を求める.

解法：Err は$\boldsymbol{w}$に関して 2 次なので

$$\frac{\partial}{\partial \boldsymbol{w}}\mathrm{Err}(\boldsymbol{w}) = \begin{pmatrix} \frac{\partial}{\partial w_0}\mathrm{Err}(\boldsymbol{w}) \\ \frac{\partial}{\partial w_1}\mathrm{Err}(\boldsymbol{w}) \\ \vdots \\ \frac{\partial}{\partial w_N}\mathrm{Err}(\boldsymbol{w}) \end{pmatrix} = \boldsymbol{0}$$

を解くことによって$\boldsymbol{w}$= $(w_0, w_1, \ldots, w_N)'$ が推定できる.

問題点：過剰適合の問題がある（資料 1 参照）.

1.2 最尤推定法

ここではベイズ推定への準備としてデータが

$$t_i = y(x_i, \boldsymbol{w}) + \varepsilon_i, \quad \varepsilon_i \sim N(0, \beta^{-1})$$

によって生成されていると仮定し，最尤法によって$\boldsymbol{w}$を推定することを考える．

注意：出力 t の確率密度関数は

$$p(t \,|\, x, \boldsymbol{w}, \beta) = \mathcal{N}(t \,|\, y(x|\boldsymbol{w}), \beta^{-1}), \quad \text{where} \quad \mathcal{N}(t \,|\, \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

である．

方針：尤度関数

$$p(\boldsymbol{t} \,|\, \boldsymbol{x}, \boldsymbol{w}, \beta) = \mathcal{N}(t_1 \,|\, y(x_1, \boldsymbol{w}), \beta^{-1}) \cdots \mathcal{N}(t_N \,|\, y(x_N, \boldsymbol{w}), \beta^{-1})$$

を最大にする$\boldsymbol{w}$=$\boldsymbol{w}_{\mathrm{ML}}$ を求める．

解法：対数尤度関数が

$$\log p(\boldsymbol{t} \,|\, \boldsymbol{x}, \boldsymbol{w}, \beta) = -\frac{\beta}{2} \sum_{n=1}^{N} (y(x_n, \boldsymbol{w}) - t_n)^2 + \frac{N}{2} \log \beta - \frac{N}{2} \log(2\pi)$$

と変形できるので，$\boldsymbol{w}_{\mathrm{ML}}$ は $\mathrm{Err}(\boldsymbol{w})$ を最小化にすることによって得られる．

1.3 ベイズ推定法

資料 1 によると，過剰適合している$\boldsymbol{w}$がとても大きいことが分かる．そこで事前に$\boldsymbol{w}$はあまり大きな値はとらないという情報を与えて$\boldsymbol{w}$を推定しよう．

方針：事前確率を例えば $p(\boldsymbol{w}) = \mathcal{N}(w_0 \,|\, 0, \alpha^{-1}) \cdots \mathcal{N}(w_M \,|\, 0, \alpha^{-1})$ で導入し，事後確率

$$p(\boldsymbol{w} \,|\, \boldsymbol{x}, \boldsymbol{t}, \beta) = \frac{p(\boldsymbol{t} \,|\, \boldsymbol{x}, \boldsymbol{w}, \beta) p(\boldsymbol{w})}{p(\boldsymbol{t} \,|\, \boldsymbol{x}, \beta)}$$

を最大にする$\boldsymbol{w}$=$\boldsymbol{w}_{\mathrm{MAP}}$ を求める．

解法：事後確率は

$$p(\boldsymbol{w} \,|\, \boldsymbol{x}, \boldsymbol{t}, \beta) = C \exp\left\{ -\left(\frac{\beta}{2} \sum_{n=1}^{N} (y(x_n, \boldsymbol{w}) - t_n)^2 + \frac{\alpha}{2} \sum_{i=0}^{M} w_i^2 \right) \right\}$$

と変形できるので,

$$\widetilde{\mathrm{Err}}(\boldsymbol{w}) = \frac{1}{2} \sum_{n=1}^{N} (y(x_n, \boldsymbol{w}) - t_n)^2 + \frac{\alpha}{2\beta} \sum_{i=0}^{M} w_i^2$$

を最小にするような$\boldsymbol{w}$を求めればよい．

2　線形回帰

前回の多項式による曲線近似を少し一般化して，

$$y(x, \boldsymbol{w}) = w_0 + \sum_{j=1}^{M-1} w_j \phi_j(\boldsymbol{x}) = \boldsymbol{w}'\boldsymbol{\phi}(\boldsymbol{x})$$
$$\boldsymbol{\phi}(\boldsymbol{x}) = (\phi_0(\boldsymbol{x}), \ldots, \phi_{M-1}(\boldsymbol{x})), \quad \phi_0(\boldsymbol{x}) = 1, \quad \boldsymbol{x} \in \mathbb{R}^D, \quad \boldsymbol{w} \in \mathbb{R}^M$$

とおいて，入力$\boldsymbol{x}$に対し出力tが

$$t = y(\boldsymbol{x}, \boldsymbol{w}) + \varepsilon, \quad \varepsilon \sim N(0, \beta^{-1})$$

で与えられているときの係数$\boldsymbol{w}$の推定を考える．

2.1　最尤推定法

方針：対数尤度関数

$$p(\boldsymbol{t} \,|\, \boldsymbol{X}, \boldsymbol{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n \,|\, \boldsymbol{w}'\boldsymbol{\phi}(\boldsymbol{x}_n), \beta^{-1}), \quad \boldsymbol{t} = (t_1, \ldots, t_N)', \quad \boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)$$

の最大点$\boldsymbol{w}_{\mathrm{ML}}$, β_{ML} を求める．（以下では，$\boldsymbol{X}$を省略する．）

解法：対数尤度関数は

$$\log p(\boldsymbol{t} \,|\, \boldsymbol{w}, \beta) = \frac{N}{2} \log \beta - \frac{N}{2} \log(2\pi) - \beta \mathrm{Err}(\boldsymbol{w}), \quad \mathrm{Err}(\boldsymbol{w}) = \frac{1}{2} \sum_{n=1}^{N} (t_n - \boldsymbol{w}'\boldsymbol{\phi}(\boldsymbol{x}_n))^2$$

と変形できるので$\boldsymbol{w}_{\mathrm{ML}}$, β_{ML} を求めるには

$$\frac{\partial}{\partial \boldsymbol{w}} \log p(\boldsymbol{t} \,|\, \boldsymbol{w}, \beta) = \boldsymbol{0}, \quad \frac{\partial}{\partial \beta} \log p(\boldsymbol{t} \,|\, \boldsymbol{w}, \beta) = 0$$

を解けばよい．これらはそれぞれ

$$\left(\sum_{n=1}^{N} \boldsymbol{\phi}(\boldsymbol{x}_n) \boldsymbol{\phi}(\boldsymbol{x}_n)' \right) \boldsymbol{w} = \sum_{n=1}^{N} \boldsymbol{\phi}(\boldsymbol{x}_n) t_n, \quad \frac{N}{2\beta} - \mathrm{Err}(\boldsymbol{w}) = 0 \qquad (\spadesuit)$$

と変形できる．ここで

$$\Phi = \begin{pmatrix} \phi_0(\boldsymbol{x}_1) & \cdots & \phi_{M-1}(\boldsymbol{x}_1) \\ \vdots & \ddots & \vdots \\ \phi_0(\boldsymbol{x}_N) & \cdots & \phi_{M-1}(\boldsymbol{x}_N) \end{pmatrix}$$

を用いると ($\spadesuit$) の前者は

$$\Phi'\Phi\boldsymbol{w} = \Phi'\boldsymbol{t}$$

と変形できるので，結局$\boldsymbol{w}_{\mathrm{ML}}$, β_{ML} は

$$\boldsymbol{w}_{\mathrm{ML}} = (\Phi'\Phi)^{-1}\Phi\boldsymbol{t}, \quad \beta_{\mathrm{ML}} = \frac{N}{\sum_{n=1}^{N} (t_n - \boldsymbol{w}'_{\mathrm{ML}}\boldsymbol{\phi}(\boldsymbol{x}_n))^2}$$

によって得られる．

2.2 ベイズ推定法

前回見たように，最尤推定法では過剰適合の問題がある．そこで前回同様に$\boldsymbol{w}$の事前分布を与えることによって，係数を推定する．

方針：事前確率を

$$p(\boldsymbol{w} \,|\, \alpha) = \mathcal{N}(\boldsymbol{w} \,|\, \boldsymbol{0}, \alpha^{-1} I)$$

で導入し，事後確率

$$p(\boldsymbol{w} \,|\, \boldsymbol{t}, \alpha, \beta) = c p(t \,|\, \boldsymbol{w}, \beta) p(\boldsymbol{w} \,|\, \alpha)$$

を最大化する．

解法：事後確率は

$$p(\boldsymbol{w} \,|\, \boldsymbol{t}, \alpha, \beta) = c' \exp \varphi(\boldsymbol{w}), \quad \varphi(\boldsymbol{w}) = -\frac{\beta}{2} \sum_{n=1}^{N} (\boldsymbol{w}' \boldsymbol{\phi}(\boldsymbol{x}_n) - t_n)^2 - \frac{\alpha}{2} \boldsymbol{w}' \boldsymbol{w}$$

と変形できる．さらに，$\varphi(\boldsymbol{w})$ は

$$\varphi(\boldsymbol{w}) = -\frac{1}{2} (\boldsymbol{w} - \boldsymbol{a})' B^{-1} (\boldsymbol{w} - \boldsymbol{a}), \quad B^{-1} = \alpha I + \beta \Phi' \Phi, \quad \boldsymbol{a} = \beta B \Phi' \boldsymbol{t}$$

と変形できるのでベイズ推定により推定値は

$$\boldsymbol{w}_{\mathrm{MAP}} = \boldsymbol{a} = \beta B \Phi' \boldsymbol{t}$$

で与えられる．

2.3 時系列ベイズ推定

前節は N 個全てのデータを用いて一度に推定する方法であった．ここでは N 個のデータ $(\boldsymbol{x}_1, t_1), \ldots, (\boldsymbol{x}_N, t_N)$ が順次手に入るとき，その都度$\boldsymbol{w}$を学習していく方法を考える．

アイデア：事後確率を次のステップの事前確率に使う．

尤度関数		事後確率 $p(\boldsymbol{w})$			
$p(t_1 \mid \boldsymbol{w})$	$\to$	$p(\boldsymbol{w} \mid t_1)$	$=$	$c_1 p(t_1 \mid \boldsymbol{w}) p(\boldsymbol{w})$	ステップ 1
$p(t_2 \mid \boldsymbol{w})$	$\to$	$p(\boldsymbol{w} \mid t_1, t_2)$	$=$	$c_2 p(t_2 \mid \boldsymbol{w}) p(\boldsymbol{w} \mid t_1)$	ステップ 2
$\vdots$		$\vdots$	$\vdots$		$\vdots$
$p(t_k \mid \boldsymbol{w})$	$\to$	$p(\boldsymbol{w} \mid t_1, \ldots, t_k)$	$=$	$c_k p(t_k \mid \boldsymbol{w}) p(\boldsymbol{w} \mid t_1, \ldots, t_{k-1})$	ステップ k

具体例については資料 2 を参照のこと．

3　分類問題

このゼミの後半の 2 回は入力された $\boldsymbol{x} \in \mathbb{R}^D$ に対して，それをどちらかのクラス C_1, C_2 に割り当てる分類問題を考察する．

方針：既知データを用いて $\mathbb{R}^D$ を $\mathbb{R}^D = C_1 \cup C_2$ と分割する．

3.1　非確率的手法を用いた解法

クラス数が 2 の分類問題は超平面 $y(\boldsymbol{x}) = \boldsymbol{w}'\boldsymbol{x} + w_0 = 0$ を 1 つ定め，$y(\boldsymbol{x}) \geq 0$ のとき $\boldsymbol{x} \in C_1$，$y(\boldsymbol{x}) < 0$ のとき $\boldsymbol{x} \in C_2$ と分類すればよい．しかしクラス数が 3 以上のときに拡張できるように次の分類方法を考える．

分類方法：2 個の線形関数

$$y_1(\boldsymbol{x}) = \boldsymbol{w}_1'\boldsymbol{x} + w_{10}, \quad y_2(\boldsymbol{x}) = \boldsymbol{w}_2'\boldsymbol{x} + w_{20}$$

を導入し，

$$y_1(\boldsymbol{x}) \geq y_2(\boldsymbol{x}) \Leftrightarrow \boldsymbol{x} \in C_1, \quad y_1(\boldsymbol{x}) < y_2(\boldsymbol{x}) \Leftrightarrow \boldsymbol{x} \in C_2$$

で分類分けを考える．

記号：簡略化のため $\boldsymbol{t} = \boldsymbol{t}(\boldsymbol{x})$ を，$\boldsymbol{t} = (1,0)' \Leftrightarrow \boldsymbol{x} \in C_1$, $\boldsymbol{t} = (0,1)' \Leftrightarrow \boldsymbol{x} \in C_2$ で定める．

即ち，考える問題は既知データ $(\boldsymbol{x}_1,\boldsymbol{t}_1), \ldots, (\boldsymbol{x}_N,\boldsymbol{t}_N)$ を用いて

$$\widetilde{W} = \begin{pmatrix} w_{10} & w_{20} \\ \boldsymbol{w}_1 & \boldsymbol{w}_2 \end{pmatrix} = \begin{pmatrix} \widetilde{\boldsymbol{w}}_1 & \widetilde{\boldsymbol{w}}_2 \end{pmatrix} \in M_{D+1,2}(\mathbb{R})$$

を学習する問題になる．ここで

$$y_k > y_j \Leftrightarrow (y_k - 1)^2 + y_j^2 < y_k^2 + (y_j - 1)^2$$

が成り立つことを使うと，任意の i 番目のデータに対して

$$\sum_{j=1}^{2} (y_j(\boldsymbol{x}_i) - t_{ij})^2 = (y_1(\boldsymbol{x}_i) - t_{i1})^2 + (y_2(\boldsymbol{x}_i) - t_{i2})^2$$

が最小になるような $\widetilde{W}$ は既知データの正しい分類を与えることがわかる．

方針：エラー関数

$$\mathrm{Err}(\widetilde{W}) = \frac{1}{2}\mathrm{tr}\{(\widetilde{X}\widetilde{W} - T)(\widetilde{X}\widetilde{W} - T)'\} \left(= \frac{1}{2}\sum_{n=1}^{N}\sum_{j=1}^{2}(y_j(\boldsymbol{x}_i) - t_{ij})^2 \right)$$

$$\widetilde{X} = \begin{pmatrix} 1 & \cdots & 1 \\ \boldsymbol{x}_1 & \cdots & \boldsymbol{x}_N \end{pmatrix}' = \begin{pmatrix} \widetilde{\boldsymbol{x}}_1 & \cdots & \widetilde{\boldsymbol{x}}_N \end{pmatrix}', \quad T = \begin{pmatrix} \boldsymbol{t}_1 & \cdots & \boldsymbol{t}_N \end{pmatrix}'$$

を最小にする $\widetilde{W}_*$ を求める．

注意：行列 $\widetilde{X}\widetilde{W}$ の (i,j) 成分は $y_j(\boldsymbol{x}_i)$ である．

解法：次の性質

$$\frac{\partial}{\partial A}\mathrm{tr}(A'BA) = (B+B')A, \quad \frac{\partial}{\partial A}\mathrm{tr}(A'B) = B$$

を用いると，$\widetilde{W}_*$ は具体的に

$$\widetilde{W}_* = (\widetilde{X}'\widetilde{X})^{-1}\widetilde{X}'T$$

と書ける．

3.2 確率的手法を用いた解法

ここでは，確率的手法を用いた分類を考える．要するに $p(C_1|\boldsymbol{x})$ を求めて，$p(C_1|\boldsymbol{x}) = 1/2$ となる超曲面を $\mathbb{R}^D$ 内で定め，$\boldsymbol{x} \in C_1 \Leftrightarrow p(C_1|\boldsymbol{x}) \geq 1/2$ によって分類を与える．

方針：以下のような流れで分類分けを定める境界を求める．

(1) データの発生について

$$p(\boldsymbol{x}|C_k) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_k)'\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_k)\right\} \quad (k=1,2),$$
$$p(C_1) = \pi, \quad p(C_2) = 1-\pi, \quad (0 \leq \pi \leq 1)$$

を仮定し既知データ $(\boldsymbol{x}_1, t_1), \ldots, (\boldsymbol{x}_N, t_N)$ を用いてパラメータ $\boldsymbol{\mu}_1$,$\boldsymbol{\mu}_2, \Sigma, \pi$ を学習する．ここで t_i は，$t_i = 1 \Leftrightarrow \boldsymbol{x}_i \in C_1$，$t_i = 0 \Leftrightarrow \boldsymbol{x}_i \in C_2$ であるとする．

(2) ベイズの定理を用いて

$$p(C_1|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|C_1)p(C_1)}{p(\boldsymbol{x})} = \frac{p(\boldsymbol{x}|C_1)p(C_1)}{p(\boldsymbol{x}|C_1)p(C_1) + p(\boldsymbol{x}|C_2)p(C_2)} \qquad (\heartsuit)$$

を求める．（このとき，$p(C_2|\boldsymbol{x}) = 1 - p(C_1|\boldsymbol{x})$ である．）

(3) $p(C_1|\boldsymbol{x}) = 1/2$ を定める超平面が C_1 と C_2 の境界として定まる．

解法：まず次に注意する：

$$\boldsymbol{x}_n \in C_1 \Rightarrow p(\boldsymbol{x}_n, t_n = 1) = \pi\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_1, \Sigma),$$
$$\boldsymbol{x}_n \in C_2 \Rightarrow p(\boldsymbol{x}_n, t_n = 0) = (1-\pi)\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_2, \Sigma)$$

これより $\boldsymbol{t} = (t_1, \ldots, t_N)$，$X = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)$ とおいたとき，尤度関数が

$$p(\boldsymbol{t}, \boldsymbol{X}|\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \Sigma, \pi) = \prod_{n=1}^{N}\{\pi\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_1, \Sigma)\}^{t_n}\{(1-\pi)\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_2, \Sigma)\}^{1-t_n}$$

となる．対数尤度関数を各パラメータで微分することにより，最大点がそれぞれ

$$\pi^* = \frac{1}{N}\sum_{n=1}^{N} t_n, \quad \boldsymbol{\mu}_1^* = \frac{1}{N_1}\sum_{n\in C_1}\boldsymbol{x}_n, \quad \boldsymbol{\mu}_2^* = \frac{1}{N_2}\sum_{n\in C_2}\boldsymbol{x}_n, \quad \Sigma^* = \frac{N_1}{N}S_1 + \frac{N_2}{N}S_2$$

であることが分かる．ここで記号の意味は以下の通り．

$$S_1 = \frac{1}{N_1}\sum_{n\in C_1}(\boldsymbol{x}_n - \boldsymbol{\mu}_1^*)(\boldsymbol{x}_n - \boldsymbol{\mu}_1^*)', \quad N_1 = \sum_{n=1}^{N} t_n,$$

$$S_2 = \frac{1}{N_2}\sum_{n\in C_2}(\boldsymbol{x}_n - \boldsymbol{\mu}_2^*)(\boldsymbol{x}_n - \boldsymbol{\mu}_2^*)', \quad N_2 = \sum_{n=1}^{N}(1 - t_n)$$

従ってパラメータが推定できた．次に (♡) は

$$p(C_1|\boldsymbol{x}) = \frac{1}{1+\exp(-a)} =: \sigma(a), \quad a = \log\frac{p(\boldsymbol{x}|C_1)p(C_1)}{p(\boldsymbol{x}|C_2)p(C_2)}$$

と変形できる．ここで $\sigma(\cdot)$ は logistic sigmoid 関数と呼ばれる単調増加関数である．これに $p(\boldsymbol{x}|C_k), p(C_k)$ を代入して整理すると

$$a = \boldsymbol{w}'\boldsymbol{x} + w_0$$

の形になる．ここで

$$\boldsymbol{w} = \Sigma^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2), \quad w_0 = -\frac{1}{2}\boldsymbol{\mu}_1'\Sigma^{-1}\boldsymbol{\mu}_1 + \frac{1}{2}\boldsymbol{\mu}_2'\Sigma^{-1}\boldsymbol{\mu}_2 + \log\frac{\pi}{1-\pi}$$

である．結局 (♡) は $p(C_1|\boldsymbol{x}) = \sigma(\boldsymbol{w}'\boldsymbol{x} + w_0)$ となる．ところで $\sigma(a) = 1/2$ のとき $a = 0$ であるから

$$p(C_1|\boldsymbol{x}) = \sigma(\boldsymbol{w}'\boldsymbol{x} + w_0) = \frac{1}{2}$$

は $\boldsymbol{w}'\boldsymbol{x} + w_0 = 0$ となる．この超平面が分類分けを定める境界となる．

3.3 [補] 超平面の幾何的考察

$y(\boldsymbol{x}) = \boldsymbol{w}'\boldsymbol{x} + w_0 = 0$ は $\mathbb{R}^D$ の超平面を定める．これを P とおくと以下の性質が成り立つ：

- $\boldsymbol{x}_A, \boldsymbol{x}_B \in \mathrm{P}$ のとき $y(\boldsymbol{x}_A) = y(\boldsymbol{x}_B)$ なので $\boldsymbol{w}'(\boldsymbol{x}_A - \boldsymbol{x}_B) = 0$ となる．従って $\boldsymbol{w}$ は P に直交するベクトルである．
- 原点 O から超平面 P までの最短距離 r_n は，$\boldsymbol{x} \in \mathrm{P}$ をとると次のように表される：

$$r_n = \frac{\boldsymbol{w}'\boldsymbol{x}}{\|\boldsymbol{w}\|} = \frac{-w_0}{\|\boldsymbol{w}\|}$$

- $\boldsymbol{x} \in \mathbb{R}^D$ に対して，$\boldsymbol{x}_\perp \in \mathrm{P}$ を $\boldsymbol{x}$ の P への射影とすると，

$$\boldsymbol{x} = \boldsymbol{x}_\perp + r\frac{\boldsymbol{w}}{\|\boldsymbol{w}\|}, \quad r \in \mathbb{R}$$

と書ける．このとき r は次のように求められる．

$$y(\boldsymbol{x}) = \boldsymbol{w}'\boldsymbol{x}_\perp + r\frac{\boldsymbol{w}'\boldsymbol{w}}{\|\boldsymbol{w}\|} + w_0 = r\|\boldsymbol{w}\|, \quad \therefore r = \frac{y(\boldsymbol{x})}{\|\boldsymbol{w}\|}$$

これらの性質はサポートベクタマシンのときに使用する．

4　サポートベクタマシン入門

前回は既知データから超平面

$$y(\boldsymbol{x}) = \boldsymbol{w}'\boldsymbol{x} + w_0 = 0$$

を学習させてデータの分類を行った．今回は少し一般化して

$$y(\boldsymbol{x}) = \boldsymbol{w}'\boldsymbol{\phi}(\boldsymbol{x}) + w_0, \quad \boldsymbol{\phi}(\boldsymbol{x}) = (\phi_1(\boldsymbol{x}), \ldots, \phi_D(\boldsymbol{x}))' \qquad (\diamondsuit)$$

の形で分類問題を考える．このとき制約条件つき極値問題に帰着させるので，ラグランジュ未定乗数法について触れておく．

4.1　[補] ラグランジュ未定乗数法

ここでは 2 種類の制約条件つき極値問題を扱う．

(1) 制約条件：$g(\boldsymbol{x}) = 0$, $\boldsymbol{x} \in \mathbb{R}^D$ の下で関数 $f(\boldsymbol{x})$ を最大にする点 $\boldsymbol{x}^*$ を求める．

補題：$g(\boldsymbol{x}) = 0$ が定める超曲面を S とする．このとき $\boldsymbol{x} \in S$ に対し

$$\nabla g(\boldsymbol{x}) = \left(\frac{\partial g}{\partial x_1}, \ldots, \frac{\partial g}{\partial x_D} \right)$$

は S に直交する．

補題：$f(\boldsymbol{x})$ が $\boldsymbol{x}^* \in S$ で極大になるなら，$\nabla f(\boldsymbol{x}^*)$ は S に直交する．

上の 2 つの補題から，$\nabla f(\boldsymbol{x}^*) + \lambda \nabla g(\boldsymbol{x}^*) = \boldsymbol{0}$ となる $0 \neq \lambda \in \mathbb{R}$ が存在する．

従って，ラグランジュ関数を

$$L(\boldsymbol{x}, \lambda) = f(\boldsymbol{x}) + \lambda g(\boldsymbol{x})$$

で定義すると，問題 (1) が極大値を持つ必要条件は

$$\nabla_x L = \boldsymbol{0}, \quad \frac{\partial}{\partial \lambda} L = 0$$

である．これを満たす $(\boldsymbol{x}^*, \lambda^*)$ を見つける方法をラグランジュ未定乗数法という．

(2) 制約条件：$g(\boldsymbol{x}) \geq 0$, $\boldsymbol{x} \in \mathbb{R}^D$ の下で関数 $f(\boldsymbol{x})$ を最大にする点 $\boldsymbol{x}^*$ を求める．

制約条件は $g(\boldsymbol{x}^*) > 0$ か $g(\boldsymbol{x}^*) = 0$ かで場合分けをする．前者は無条件の極値問題であり，後者は (1) と同じ条件付極値問題となる．従ってラグランジュ関数

$$L(\boldsymbol{x}, \lambda) = f(\boldsymbol{x}) + \lambda g(\boldsymbol{x})$$

（但し $g(\boldsymbol{x}) \geq 0$, $\lambda \geq 0$, $\lambda g(\boldsymbol{x}) = 0$）の極値を与える $(\boldsymbol{x}^*, \lambda^*)$ を探す方法が (2) に対するラグランジュ未定係数法である．

(3) 制約条件が複数：$g_j(\boldsymbol{x}) = 0\,(j = 1, \ldots, J)$，$h_k(\boldsymbol{x}) \geq 0\,(k = 1, \ldots, K)$ のとき

$$L(\boldsymbol{x}, \{\lambda_j\}, \{\mu_k\}) = f(\boldsymbol{x}) + \sum_{j=1}^{J} \lambda_j g_j(\boldsymbol{x}) + \sum_{k=1}^{K} \mu_k h_k(\boldsymbol{x})$$

（但し $\mu_k \geq 0$，$\mu_k h_k(\boldsymbol{x}) = \boldsymbol{0}$） とすればよい．

4.2 サポートベクタマシン

既知データを $(\boldsymbol{x}_n, t_n), n = 1, \ldots, N$ とする．ここで

$$t_n = \begin{cases} 1 & y(\boldsymbol{x}_n) > 0 \quad \Leftrightarrow \quad \boldsymbol{x}_n \in C_1 \\ -1 & y(\boldsymbol{x}_n) < 0 \quad \Leftrightarrow \quad \boldsymbol{x}_n \in C_2 \end{cases}$$

である．サポートベクタマシンはデータを分類する曲面 $y(\boldsymbol{x}) = 0$ のうち，**最も近い既知データとの距離が最大になる曲面** を求める方法である．これは §3.3 と $|y(\boldsymbol{x}_n)| = t_n y(\boldsymbol{x}_n)$ より

$$\arg\max_{w,w_0} \left\{ \frac{1}{\|\boldsymbol{w}\|} \min_n [t_n(\boldsymbol{w}'\boldsymbol{\phi}(\boldsymbol{x}_n) + w_0)] \right\}$$

で与えられる．ここで $\{\cdot\}$ は $\boldsymbol{w} \to \kappa\boldsymbol{w}$，$w_0 \to \kappa w_0\, (0 \neq \kappa \in \mathbb{R})$ で不変なので，曲面に最も近い点 $\boldsymbol{x}_n$ で

$$t_n(\boldsymbol{w}'\boldsymbol{\phi}(\boldsymbol{x}_n) + w_0 = 1$$

と規格化すると，次の制約条件付き極値問題になる：

制約条件：$t_i(\boldsymbol{w}'\boldsymbol{\phi}(\boldsymbol{x}_i) + w_0) \geq 1\, (i = 1, \ldots, N)$ のもとで

$$\arg\max_{w,w_0} \frac{1}{\|\boldsymbol{w}\|} = \arg\min_{w,w_0} \frac{1}{2}\|\boldsymbol{w}\|^2$$

を求める．

ラグランジュ関数を次で定義する：

$$L(\boldsymbol{w}, w_0, \boldsymbol{a}) = \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{n=1}^{N} a_n\{t_n(\boldsymbol{w}'\boldsymbol{\phi}(\boldsymbol{x}_n) + w_0) - 1\}$$

$a_n \geq 0$，$t_n y(\boldsymbol{x}_n) \geq 1$，$a_n(t_n y(\boldsymbol{x}_n) - 1) = 0$ の下で $L(\boldsymbol{w}, w_0, \boldsymbol{a})$ の極値を探せばよい．ここで

$$\frac{\partial L}{\partial \boldsymbol{w}} = \boldsymbol{w} - \sum_{n=1}^{N} a_n t_n \boldsymbol{\phi}(\boldsymbol{x}_n) = \boldsymbol{0}, \quad \frac{\partial L}{\partial w_0} = -\sum_{n=1}^{N} a_n t_n = 0$$

これより $\boldsymbol{w}^* = \sum_{n=1}^{N} a_n t_n \boldsymbol{\phi}(\boldsymbol{x}_n)$ なので，$L(\boldsymbol{w}, w_0, \boldsymbol{a})$ に代入すると次のようになる：

$$\widetilde{L}(\boldsymbol{a}) = \sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n a_m t_n t_m k(\boldsymbol{x}_n, \boldsymbol{x}_m), \quad k(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{\phi}(\boldsymbol{x})'\boldsymbol{\phi}(\boldsymbol{x}')$$

$a_n \geq 0$，$\sum_{n=1}^{N} a_n t_n = 0$ の下で極値 $\widetilde{L}(\boldsymbol{a}^*)$ が求まれば，$\boldsymbol{w}^* = \sum_{n=1}^{N} a_n^* t_n \boldsymbol{\phi}(\boldsymbol{x}_n)$ となる．これは MATLAB などの数式ソフトウェアで数値解を求めることが出来る．

学習解 $\boldsymbol{w}^*$ の性質：$(\diamondsuit)$ に $\boldsymbol{w}^*$ を代入し，さらに $a_n(t_n y(\boldsymbol{x}_n) - 1) = 0$ に注意すると

$$\begin{aligned} y(\boldsymbol{x}) &= \boldsymbol{w}^{*\prime}\boldsymbol{\phi}(\boldsymbol{x}) + w_0 = \sum_{n=1}^{N} a_n t_n k(\boldsymbol{x}_n, \boldsymbol{x}) + w_0 \\ &= \sum_{n\in\mathcal{S}} a_n t_n k(\boldsymbol{x}_n, \boldsymbol{x}) + w_0, \quad \text{where} \quad \mathcal{S} = \{n \,|\, t_n y(\boldsymbol{x}_n) = 1\} \end{aligned}$$

となる．$n \in \mathcal{S}$ となる $\boldsymbol{x}_n$ は分類曲面 $y(\boldsymbol{x}) = 0$ の最も近くにある既知データなので未知データの分類の計算量が大幅に減少する．この最も近い点のことをサポートベクトルという．サポートベクトルのみを使ってデータの分類を行うのがサポートベクタマシンの特徴である．