Chapter 4. Linear Models for Classification

2022 年 2 月 22 日

目次

この章では、D-dimensional input space を disjoint な K 個の decision regions (決定領域) $\mathcal{C}_k$ $(k = 1, \cdots, K)$ に分ける decision boundaries/decision surfaces (決定境界/決定面) を決定するための線形モデルについて考える[*1]。

特別な場合を除き、本章では正解変数を

- 二クラス分類 t : $\{0, 1\}$
- 多クラス分類 $\mathbf{t}$: 1-of-K coding[*2]

で表すこととし、以下の 3 つの手法について説明する。

1. 入力ベクトルから直接クラスを推定する discrimination function (識別関数) を用いた方法
2. 条件付き確率分布 $p(\mathcal{C}_k|\mathbf{x})$ をモデル化してその分布を利用する方法
   (a) $p(\mathcal{C}_k|\mathbf{x})$ を直接モデル化する
   (b) $p(\mathbf{x}|\mathcal{C}_k)$ と $p(\mathcal{C}_k)$ をモデル化して Bayes' theorem から $p(\mathcal{C}_k|\mathbf{x})$ を求める

また、本章では 3 章の線形回帰を activation function (活性化関数) $f(\cdot)$ を用いて確率に拡張した generalized linear model (一般化線形モデル)

$$y(\mathbf{x}) = f\left(\mathbf{w}^T\mathbf{x} + w_0\right), \tag{4.1}$$

のみについて説明するが、3 章で見たように $\phi(\mathbf{x})$ を基底に用いることで容易に議論を拡張できる。

以降の一連の流れとして、まず二クラス分類の議論をした後にそれを K クラス分類に一般化するという形で進める。

[*1] 線形の決定面で完全に分離可能なデータを linearly separable (線形分離可能) と言う。

[*2] 例えば $K = 4$ で class 2 に分類されるときの target は $\mathbf{t} = (0, 1, 0, 0)^T$

4.1 Discriminant Functions

まずは識別関数を用いた分類のお話。

4.1.1 Two classes

ノート参照。

4.1.7 The perceptron algorithm

2 クラス分類の有名な線形識別モデルであるパーセプトロンモデルを最後に簡単に説明する。

パーセプトロンモデルは、適当な非線形変換 $\phi(\mathbf{x})$ の下での一般化線形モデル

$$y(\mathbf{x}) = f\left(\mathbf{w}^T \phi(\mathbf{x})\right), \quad (\text{including bias component } \phi_0(\mathbf{x}) = 1) \tag{4.2}$$

where

$$f(a) = \begin{cases} +1, & a \geq 0 \\ -1, & a \leq 0 \end{cases}. \tag{4.3}$$

である。さらに、正解ラベルの値として $t \in \{+1, -1\}$ を用いることで議論が簡略化されている。

誤差関数としては、$\mathcal{M}$ を誤分類されたものの集合として perceptron criterion (パーセプトロン基準) と呼ばれる

$$E_P(\mathbf{w}) = -\sum_{n \in \mathcal{M}} \mathbf{w}^T \phi_n t_n, \tag{4.4}$$

を用いる[*3]。この誤差関数を用いて、$\mathbf{w}$ を確率的勾配降下法

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta \nabla E_P(\mathbf{w}) = \mathbf{w}^{(\tau)} + \eta \phi_n t_n. \tag{4.5}$$

によって更新していく。また、パーセプトロンモデルの出力は $\mathbf{w}$ の定数倍で変化しないので、一般性を失わずに $\eta = 1$ とできる。従って、パーセプトロンモデルの学習プロセスでは、パーセプトロン関数を評価した後、謝って分類された場合のみ重みベクトル $\mathbf{w}$ に $\phi_n t_n$ を足すという過程を繰り返すものであるということがわかる (図. 4.7)。

このプロセスは、

$$-\mathbf{w}^{(\tau+1)T} \phi_n t_n = -\mathbf{w}^{(\tau)T} \phi_n t_n - (\phi_n t_n)^T \phi_n t_n < -\mathbf{w}^{(\tau)T} \phi_n t_n. \tag{4.6}$$

より誤分類されたパターンからの誤差への寄与を減少させることができる一方、それ以外のデータの誤差にどう影響を与えるかについては何も言っていない。さらには、総誤差関数を減少させることを保証してもいない。

ただし、perceptron convergence theorem (パーセプトロンの収束定理) では、学習データ集合が線形分離可能な場合パーセプトロン学習アルゴリズムは有限回の繰り返しで厳密解に収束することが保証されている。しかし、その繰り返しの回数がかなり多いため、現実では線形分離できない問題なのか収束が遅いだけなのか判断がつかないことが多い。また、線形分離可能な場合でもパラメータの初期値やデータの入力順に応じて様々な解に収束してしまうといった問題もある。

4.2 Probabilistic Generative Models

次に、クラス条件付 $p(\mathcal{C}_k|\mathbf{x})$ 分布を用いて決定境界を定める generative な手法を用いることで、単純な仮定の下線形な決定境界が現れることを見てみる。

[*3] 最も単純な誤差関数は誤分類されたデータの数であるが、そうすると誤差関数が $\mathbf{w}$ に関して不連続になってしまい勾配降下法が使えないため、より数学的に扱いやすい形になっている。

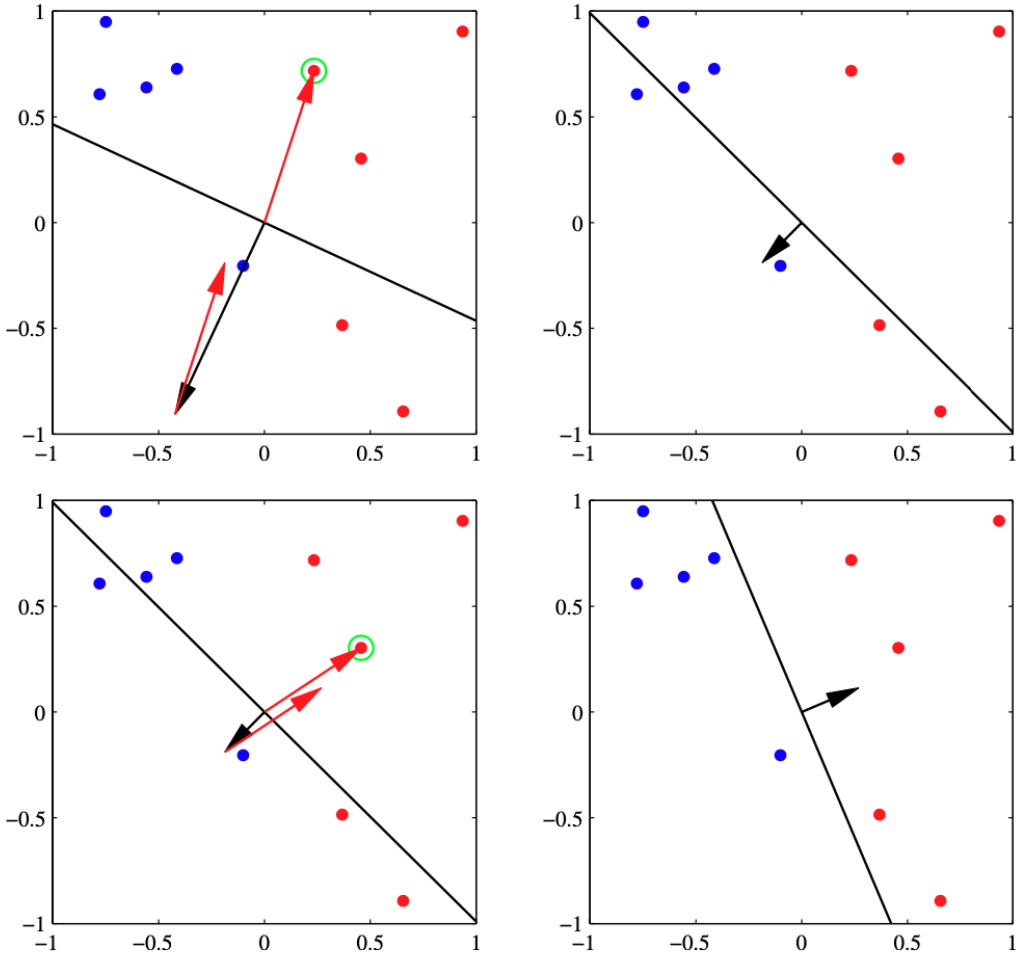

Figure 4.7 Illustration of the convergence of the perceptron learning algorithm, showing data points from two classes (red and blue) in a two-dimensional feature space (ϕ_1, ϕ_2). The top left plot shows the initial parameter vector $\mathbf{w}$ shown as a black arrow together with the corresponding decision boundary (black line), in which the arrow points towards the decision region which classified as belonging to the red class. The data point circled in green is misclassified and so its feature vector is added to the current weight vector, giving the new decision boundary shown in the top right plot. The bottom left plot shows the next misclassified point to be considered, indicated by the green circle, and its feature vector is again added to the weight vector giving the decision boundary shown in the bottom right plot for which all data points are correctly classified.

Logistic sigmoid function

二クラス分類の場合、例えば $\mathcal{C}_1$ に分類される確率は Bayes の定理より

$$p(\mathcal{C}_1|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{C}_1)\,p(\mathcal{C}_1)}{p(\mathbf{x}|\mathcal{C}_1)\,p(\mathcal{C}_1) + p(\mathbf{x}|\mathcal{C}_2)\,p(\mathbf{x}|\mathcal{C}_2)}, \tag{4.7}$$

$$= \frac{1}{1+\exp(-a)}, \tag{4.8}$$

$$\equiv \sigma(a). \tag{4.9}$$

where

$$\sigma(a) \equiv \frac{1}{1+\exp(-a)}, \quad a \equiv \ln\frac{p(\mathbf{x}|\mathcal{C}_1)\,p(\mathcal{C}_1)}{p(\mathbf{x}|\mathcal{C}_2)\,p(\mathcal{C}_2)}, \tag{4.10}$$

と表され、この σ を logistic sigmoide function と呼ぶ。

定義から明らかなように

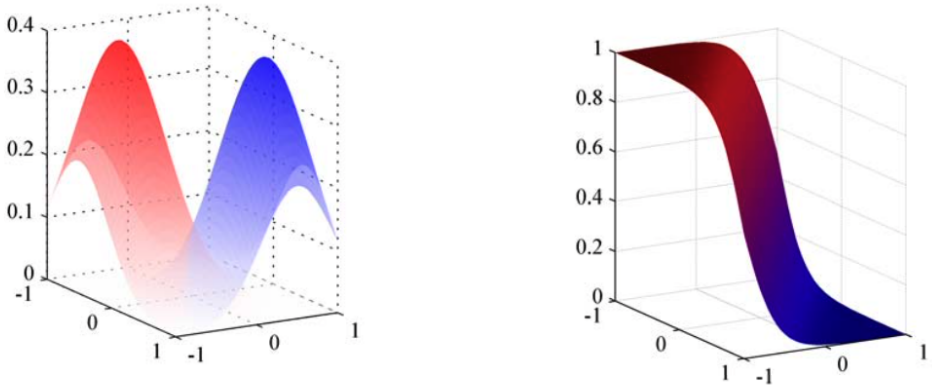

Figure 4.10 The left-hand plot shows the class-conditional densities for two classes, denoted red and blue. On the right is the corresponding posterior probability $p(\mathcal{C}_1|\mathbf{x})$, which is given by a logistic sigmoid of a linear function of $\mathbf{x}$. The surface in the right-hand plot is coloured using a proportion of red ink given by $p(\mathcal{C}_1|\mathbf{x})$ and a proportion of blue ink given by $p(\mathcal{C}_2|\mathbf{x}) = 1 - p(\mathcal{C}_1|\mathbf{x})$.

$$\sigma(-a) = 1 - \sigma(a), \quad a = \ln\left(\frac{\sigma}{1-\sigma}\right), \tag{4.11}$$

の性質があり、a は logit function/log odds と呼ばれる。

Normalized exponential function

$K > 2$ の他クラス分類の場合、事後確率は normalized exponential function (正規化指数関数)

$$p(\mathcal{C}_k|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{C}_k)\,p(\mathcal{C}_k)}{\sum_j p(\mathbf{x}|\mathcal{C}_j)\,p(\mathbf{x}|\mathcal{C}_j)} = \frac{\exp(a_k)}{\sum_j \exp(a_j)}, \tag{4.12}$$

where

$$a_k \equiv \ln p(\mathbf{x}|\mathcal{C}_k)\,p(\mathcal{C}_k), \tag{4.13}$$

で表され、a_k は max 関数と似ていることより softmax function と呼ばれる[*4]。

4.2.1 Continuous Inputs

まず、クラス条件付分布が Gaussian で表され、クラス間の共分散行列が同じ場合

$$p(\mathbf{x}|\mathcal{C}_k) = \frac{1}{(2\pi)^{D/2}}\frac{1}{|\mathbf{\Sigma}|^{1/2}}\exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T\mathbf{\Sigma}(\mathbf{x}-\boldsymbol{\mu}_k)\right], \tag{4.14}$$

を考える。このとき、(4.9), (4.10) より

$$p(\mathcal{C}_1|\mathbf{x}) = \sigma\left(\mathbf{w}^T\mathbf{x} + w_0\right), \tag{4.15}$$

where

$$\mathbf{w} = \mathbf{\Sigma}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2), \tag{4.16}$$

$$w_0 = -\frac{1}{2}\boldsymbol{\mu}_1^T\mathbf{\Sigma}^{-1}\boldsymbol{\mu}_1 + \frac{1}{2}\boldsymbol{\mu}_2^T\mathbf{\Sigma}^{-1}\boldsymbol{\mu}_2 + \ln\frac{p(\mathcal{C}_1)}{p(\mathcal{C}_2)}. \tag{4.17}$$

従って、事後分布は $\mathbf{x}$ の 1 次関数となり図 4.10 の右のように図示される。

[*4] 例えば、もし $a_k \gg a_j$ for all $j \neq k$ なら $p(\mathcal{C}_k|\mathbf{x}) \simeq 1$ and $p(\mathcal{C}_j|\mathbf{x})$ となる。

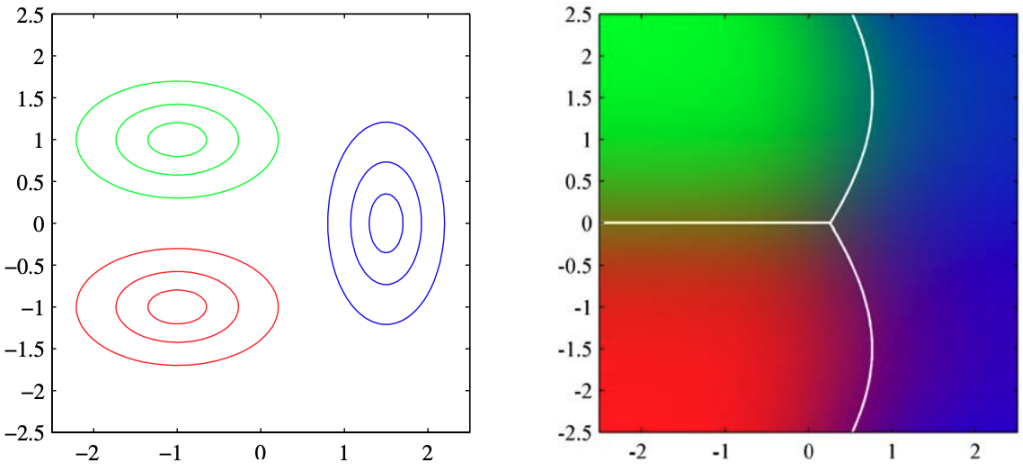

Figure 4.11 The left-hand plot shows the class-conditional densities for three classes each having a Gaussian distribution, coloured red, green, and blue, in which the red and green classes have the same covariance matrix. The right-hand plot shows the corresponding posterior probabilities, in which the RGB colour vector represents the posterior probabilities for the respective three classes. The decision boundaries are also shown. Notice that the boundary between the red and green classes, which have the same covariance matrix, is linear, whereas those between the other pairs of classes are quadratic.

一般の K クラス分類の場合も同様に (4.12), (4.13) より、

$$a_k \equiv \ln p\left(\mathbf{x}|\mathcal{C}_k\right) p\left(\mathcal{C}_k\right) = \mathbf{w}_k^T \mathbf{x} + w_{k0}, \tag{4.18}$$

where

$$\mathbf{w}_k = \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k, \tag{4.19}$$

$$w_0 = -\frac{1}{2} \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k + \ln p\left(\mathcal{C}_k\right), \tag{4.20}$$

となり、これも $\mathbf{x}$ の線形関数になることがわかる。また、1.5.1 で見たように誤分類を最小にする決定境界は事後分布の確率が等しい場合に相当するため、決定境界も同様に $\mathbf{x}$ の線形関数になる (図 4.11)。

一方、クラスごとの共分散行列が同じではない場合、分母分子の cancel がなくなるため quadratic discriminant (2 次判別関数) となる (図 4.11)。

4.2.2 Maximum likelihood solution

最尤推定法を使ってパラメータの値を確認してみる。

まず、共通の共分散で Gaussian の事前分布を持つ 2 クラス分類を考えデータセットを $\{\mathbf{x}_n, t_n\}$ $(n = 1, \cdots, N)$ と表す。また、事前分布を

$$p\left(\mathcal{C}_1\right) = \pi, \ p\left(\mathcal{C}_2\right) = 1 - \pi, \tag{4.21}$$

とおくと、事後分布はそれぞれ

$$p\left(\mathbf{x}_n, \mathcal{C}_1\right) = p\left(\mathcal{C}_1\right) p\left(\mathbf{x}_n|\mathcal{C}_1\right) = \pi \mathcal{N}\left(\mathbf{x}_n|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}\right), \tag{4.22}$$

$$p\left(\mathbf{x}_n, \mathcal{C}_2\right) = p\left(\mathcal{C}_2\right) p\left(\mathbf{x}_n|\mathcal{C}_2\right) = (1 - \pi) \mathcal{N}\left(\mathbf{x}_n|\boldsymbol{\mu}_2, \boldsymbol{\Sigma}\right). \tag{4.23}$$

$$\therefore \quad p\left(\mathbf{t}|\pi, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}\right) = \prod_{n=1}^{N} \left[\pi \mathcal{N}\left(\mathbf{x}_n|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}\right)\right]^{t_n} \left[(1 - \pi) \mathcal{N}\left(\mathbf{x}_n|\boldsymbol{\mu}_2, \boldsymbol{\Sigma}\right)\right]^{1 - t_n}. \tag{4.24}$$

where $\mathbf{t} = \left(t_1, \cdots, t_N\right)^T$.

まず、π について log-likelihood を最大化してみると、

$$\sum_{n=1}^{N} \left[t_n \ln \pi + (1 - t_n) \ln (1 - \pi) \right]. \tag{4.25}$$

$$\therefore \quad \frac{\partial \ln p}{\partial \pi} = 0 \quad \Leftrightarrow \quad \pi = \frac{1}{N} \sum_{n=1}^{N} t_n = \frac{N_1}{N_1 + N_2}. \tag{4.26}$$

となり、単純に各クラスに含まれるデータ数との比で表されることがわかる。隣全く同様にして、K クラス分類の場合も N_k/N となる。

$\boldsymbol{\mu}_1$ については、

$$\sum_{n=1}^{N} t_n \mathcal{N} \left(\mathbf{x}_n | \boldsymbol{\mu}_1, \boldsymbol{\Sigma} \right) = -\frac{1}{2} \sum_{n=1}^{N} t_n \left(\mathbf{x}_n - \boldsymbol{\mu}_1 \right)^T \Sigma^{-1} \left(\mathbf{x}_n - \boldsymbol{\mu}_1 \right) + \text{const.} \tag{4.27}$$

$$\therefore \quad \frac{\partial \ln p}{\partial \boldsymbol{\mu}_1} = 0 \quad \Leftrightarrow \quad \boldsymbol{\mu}_1 = \frac{1}{N_1} \sum_{n=1}^{N} t_n \mathbf{x}_n. \tag{4.28}$$

となり[*5]、input vector $\mathbf{x}$ の単純平均となる。$\boldsymbol{\mu}_2$ についても同様。

最後に shared covariance matrix $\boldsymbol{\Sigma}$ について考えると、

$$-\frac{1}{2} \sum_{n=1}^{N} t_n \ln |\boldsymbol{\Sigma}| - \frac{1}{2} \sum_{n=1}^{N} t_n \left(\mathbf{x}_n - \boldsymbol{\mu}_1 \right)^T \boldsymbol{\Sigma}^{-1} \left(\mathbf{x}_{\boldsymbol{n}} - \boldsymbol{\mu}_1 \right), \tag{4.29}$$

$$= -\frac{1}{2} \sum_{n=1}^{N} (1 - t_n) \ln |\boldsymbol{\Sigma}| - \frac{1}{2} \sum_{n=1}^{N} (1 - t_n) \left(\mathbf{x}_n - \boldsymbol{\mu}_1 \right)^T \boldsymbol{\Sigma}^{-1} \left(\mathbf{x}_{\boldsymbol{n}} - \boldsymbol{\mu}_1 \right), \tag{4.30}$$

$$= -\frac{N}{2} \ln |\boldsymbol{\Sigma}| - \frac{N}{2} \mathrm{Tr} \left[\boldsymbol{\Sigma}^{-1} \mathbf{S} \right]. \tag{4.31}$$

where

$$\mathbf{S} = \frac{N_1}{N} \mathbf{S}_1 + \frac{N_2}{N} \mathbf{S}_2, \tag{4.32}$$

$$\mathbf{S}_1 = \frac{1}{N_1} \sum_{n \in \mathcal{C}_1} \left(\mathbf{x}_n - \boldsymbol{\mu}_1 \right) \left(\mathbf{x}_n - \boldsymbol{\mu}_1 \right)^T, \tag{4.33}$$

$$\mathbf{S}_2 = \frac{1}{N_2} \sum_{n \in \mathcal{C}_2} \left(\mathbf{x}_n - \boldsymbol{\mu}_2 \right) \left(\mathbf{x}_n - \boldsymbol{\mu}_2 \right)^T. \tag{4.34}$$

$$\therefore \quad \frac{\partial \ln p}{\partial \boldsymbol{\Sigma}} = 0 \quad \Leftrightarrow \quad \boldsymbol{\Sigma} = \mathbf{S}. \tag{4.35}$$

この方法は K クラス分類の場合にも容易に一般化できる。しかし、正規分布を仮定した最尤推定は外れ値に頑強ではないことには注意した方が良い。

4.2.3 Discrete features

離散特徴量について考える。

まず、簡単のため D 次元の二値特徴量 $x_i \in \{0, 1\}$ を考えると、分布をパラメータの組み合わせは $2^D - 1$ 通り存在する。このままではパラメータ数が指数関数的に発散してしまうので、$\mathcal{C}_k$ の条件の下で各特徴量のパラメータが独立だと仮定する naive Bayes assumtion を用いると、

$$p \left(\mathbf{x} | \mathcal{C}_k \right) = \prod_{i=1}^{D} \mu_{ki}^{x_i} \left(1 - \mu_{ki} \right)^{1 - x_i}, \tag{4.36}$$

[*5] $\frac{d}{d\boldsymbol{x}} \left(\boldsymbol{x}^T A \boldsymbol{x} \right) = \left(A + A^T \right) \boldsymbol{x}$ などを用いる。

と表される。これを (4.13) に代入すると、

$$a(\mathbf{x}) = \sum_{i=1}^{D} \{x_i \ln \mu_{ki} + (1 - x_i) \ln (1 - \mu_{ki})\} + \ln p(\mathcal{C}_k), \tag{4.37}$$

となり再び線形関数で表されることがわかる。

4.2.4 Exponential family

Gaussian や離散データで事後分布が $\mathbf{x}$ に関して logistic sigmoid/softmax function を活性化関数とした generalized linear models になるという話は指数分布族の特別な場合で一般化できる。

クラス条件付確率分布を指数分布族

$$p(\mathbf{x}|\boldsymbol{\lambda}_k) = h(\mathbf{x})\, g(\boldsymbol{\lambda}_k) \exp\left[\boldsymbol{\lambda}_k^T \boldsymbol{u}(\mathbf{x})\right], \tag{4.38}$$

として $\boldsymbol{u}(\mathbf{x}) = \mathbf{x}$ に限り、scale invariance を導入してパラメータ s を設定すると[*6]、

$$p(\mathbf{x}|\boldsymbol{\lambda}_k, s) = \frac{1}{s} h\left(\frac{1}{s}\mathbf{x}\right) g(\boldsymbol{\lambda}_k) \exp\left[\frac{1}{s}\boldsymbol{\lambda}_k^T \mathbf{x}\right], \tag{4.39}$$

と表せる。これをそれぞれ (4.10), (4.13) に代入すれば

$$a(\mathbf{x}) = (\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2)^T \mathbf{x} + \ln g(\boldsymbol{\lambda}_1) - \ln g(\boldsymbol{\lambda}_2) + \ln p(\mathcal{C}_1) - \ln p(\mathcal{C}_2), \tag{4.40}$$

$$a_k(\mathbf{x}) = \boldsymbol{\lambda}_k^T \mathbf{x} + \ln g(\boldsymbol{\lambda}_k) \ln p(\mathcal{C}_k), \tag{4.41}$$

となり、どちらも $\mathbf{x}$ の線形関数になることがわかる。

[*6] Scale invariance を仮定しているのは Gaussian との analogy？