

6章カーネル法

これまで扱ってきたモデルは、基底 $\phi(\mathbf{x})$ に対してパラメータベクトル \mathbf{w} をかけ合わせる形のモデルを扱ってきた。この形式のモデルは有限次元の基底しか扱えないというデメリットがある。**カーネル法**は、パラメータ \mathbf{w} を保持する代わりに、学習データの全体あるいは一部を保持する形のモデルであり、**カーネル関数**を用いることで実質無限の基底を扱うことができるようになる。また、文字列などの記号的オブジェクトを扱える点もカーネル法の利点の一つである。

6.1、6.2 節はカーネル関数の構成方法を取り扱い、6.3、6.4 節でカーネル法を使った予測モデルを扱う。ここまでは全ての学習データを保持したモデルであるが、7章で扱うSVMは一部の学習データのみ保持するカーネル法であり、（既にご存知の通り）広く使われている予測モデルである。

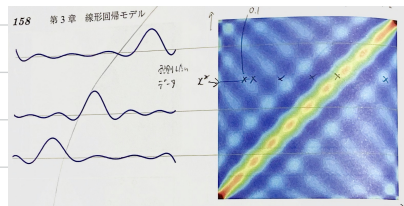
おさらい 上巻3.3節 (p.157) ベイズ線形回帰で登場したカーネル

ここで登場したカーネル関数とは、「データ点の近さ」を表すものだった。

ベイズ線形回帰で得られたパラメータベクトル \mathbf{m}_N の事後分布の平均 \mathbf{m}_N を、回帰モデルに代入すると、次の式が得られた。

$$y(\mathbf{x}, \mathbf{m}_N) = \sum_{n=1}^N k(\mathbf{x}, \mathbf{x}_n) t_n$$

この式の解釈としては、学習データのラベル集合 t を、特徴 \mathbf{x}_n と入力 \mathbf{x} の近さで重みづけたものと解釈できる。図3.10はこのカーネル関数 $k(\mathbf{x}, \mathbf{x}_n)$ が近いデータに対して高い数値を出力することを表している。



6.1 節 双対表現

ここでは、カーネル法の理解を深めるために、これまで扱ってきた二乗和誤差最小化の線形回帰問題が、双対表現によってカーネル関数で表せることを示す。操作自体は無意味であるが、基底関数 $\phi()$ の代わりにカーネル関数で回帰モデルを表現できていることを確かめる。カーネル関数は、入力 x に対する基底関数を $\phi(x)$ とした時に、次のように表される。

$$k(x, x') = \phi(x)^T \phi(x')$$

これまで扱ってきた、線形回帰モデルの正則化最小二乗法は、次の誤差関数の最小化に相当する。

$$J(w) = \frac{1}{2} \sum_{n=1}^N \left\{ w^T \phi(x_n) - t_n \right\}^2 + \frac{\lambda}{2} w^T w$$

$$J'(w) = 0 \text{ より,}$$

$$0 = \sum_{n=1}^N \left\{ w^T \phi(x_n) - t_n \right\} \phi(x_n) + \lambda w$$

左辺にそろえる

$$\Leftrightarrow w = - \frac{1}{\lambda} \sum_{n=1}^N \left\{ w^T \phi(x_n) - t_n \right\} \phi(x_n)$$

nに依存するスカラー値 $\equiv \alpha_n$ とおく

$$= \sum_{n=1}^N \alpha_n \phi(x_n)$$

$$= \begin{pmatrix} \phi(x_1) & \phi(x_2) & \dots & \phi(x_N) \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_N \end{pmatrix}$$

$$= \begin{pmatrix} \phi(x_1)^T \\ \phi(x_2)^T \\ \vdots \\ \phi(x_N)^T \end{pmatrix}^T \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_N \end{pmatrix}$$

$$= \Phi^T \alpha$$

これを使って $J(w)$ を α で表現し直すと、

$$\begin{aligned}
J(w) &= \frac{1}{2} \sum_{n=1}^N \left\{ w^T \phi(x_n) \phi(x_n)^T w - 2 w^T \phi(x_n) t_n + t_n^2 \right\} + \frac{\lambda}{2} w^T w \\
&= \frac{1}{2} w^T \underbrace{\sum_{n=1}^N \phi(x_n) \phi(x_n)^T}_{=\Phi^T \Phi \text{ にあたる? ①}} w - w^T \underbrace{\sum_{n=1}^N \phi(x_n) t_n}_{=\begin{pmatrix} \phi(x_1)^T \\ \vdots \\ \phi(x_N)^T \end{pmatrix}^T \begin{pmatrix} t_1 \\ \vdots \\ t_N \end{pmatrix}} + \frac{1}{2} \sum_{n=1}^N t_n^2 + \frac{\lambda}{2} w^T w
\end{aligned}$$

$$= \frac{1}{2} \alpha^T \Phi \Phi^T \Phi \Phi^T \alpha - \alpha^T \Phi \Phi^T t + \frac{1}{2} t^T t + \frac{\lambda}{2} \alpha^T \Phi \Phi^T \alpha \quad (6.5)$$

①について、 i, j 成分を考えると、成り立ちそうに $\phi(x_i) \dots \phi(x_j)$?

$$\left(\Phi^T \Phi \right)_{ij} = \phi(x_i)^T \phi(x_j)$$

\uparrow
 $\begin{pmatrix} \phi(x_1)^T \\ \vdots \\ \phi(x_N)^T \end{pmatrix}^T$

$$\begin{aligned}
\left(\sum_{n=1}^N \phi(x_n) \phi(x_n)^T \right)_{ij} &= \sum_{n=1}^N \left(\phi(x_n) \phi(x_n)^T \right)_{ij} \\
&= \sum_{n=1}^N \phi_i(x_n) \phi_j(x_n)
\end{aligned}$$

次にグラム行列 $K = \Phi \Phi^T$ を導入する。 K は、 n, m 要素が $k(x_n, x_m)$ となるような行列である。

$$\begin{aligned}
K = \Phi \Phi^T &= \begin{pmatrix} \phi(x_1)^T \\ \phi(x_2)^T \\ \vdots \\ \phi(x_N)^T \end{pmatrix} \begin{pmatrix} \phi(x_1) & \phi(x_2) & \dots & \phi(x_N) \end{pmatrix} \\
&= \begin{pmatrix} \phi(x_1)^T \phi(x_1) & \dots & \phi(x_1)^T \phi(x_N) \\ \vdots & & \vdots \\ \phi(x_N)^T \phi(x_1) & \dots & \phi(x_N)^T \phi(x_N) \end{pmatrix} \\
&= \begin{pmatrix} k(x_1, x_1) & \dots & k(x_1, x_N) \\ \vdots & & \vdots \\ k(x_N, x_1) & \dots & k(x_N, x_N) \end{pmatrix}
\end{aligned}$$

すなわち、 $J(w) = J(\alpha)$ は、次の式になる。

$$J(\alpha) = \frac{1}{2} \alpha^T K K \alpha - \alpha^T K t + \frac{1}{2} t^T t + \frac{\lambda}{2} \alpha^T K \alpha \quad (6.7)$$


(6.7) の表現は、 w を解く代わりに α を解くという点で、双対表現と呼ばれている。 w の次元は基底関数の次元で、 α の次元はデータサイズとなっているので、多くの場合、解の次元が大きくなる点に注意。

最後に、 α の解を求める。教科書だと、 $J(w)$ の w に関する停留点を求める式 (6.3) を使って解いているが、ここではせっかく双対表現を求めたので、(6.7) の停留点を求める形で導出する。

$$\begin{aligned}
 0 &= J'(\alpha) = K K \alpha - K t + \lambda K \alpha \\
 \Leftrightarrow 0 &= K \alpha - t + \lambda \alpha && (K^{-1} \text{は存在するとします...}) \\
 \Leftrightarrow (K + \lambda I) \alpha &= t \\
 \Leftrightarrow \alpha &= (K + \lambda I)^{-1} t && (6.8)
 \end{aligned}$$

双対表現における解を得られたので、回帰モデル自体も w を使わない形で表現し直すことができる。回帰モデルは次のようになる。

$$\begin{aligned}
 y(x) &= w \phi(x) = \alpha^T \Phi \phi(x) = \phi^T(x) \Phi^T \alpha \\
 &= \phi^T(x) (\phi(x_1) \phi(x_2) \dots \phi(x_n)) \alpha \\
 &= \begin{pmatrix} \phi^T(x) \phi(x_1) \\ \vdots \\ \phi^T(x) \phi(x_n) \end{pmatrix}^T (K + \lambda I_n)^{-1} t \\
 &= k(x)^T (K + \lambda I_n)^{-1} t && (6.9)
 \end{aligned}$$



 中身は全てカーネル関数で表現されている。

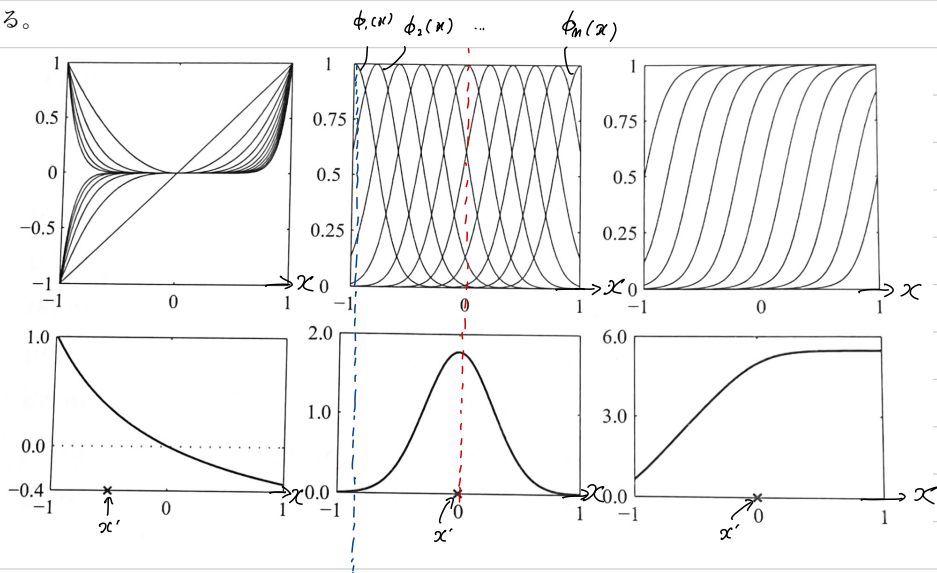
以上より、パラメータベクトル w の代わりに、カーネル関数（とグラム行列）でモデルを表現できることを示せた。

6.2節 カーネル関数の構成

ここまでは基底関数の構成方法を学ぶ。構成方法は大きく分けて二つあり、一つはこれまで扱ってきたように、既に基底関数が与えられている場合に、対応するカーネル関数を求める方法である。もう一つは、カーネル関数を直接定義し、カーネル関数の性質を満たしているかを確認するという方法である。

実は後者の方が重要で、基底関数を明に定めなくとも柔軟にモデルを構築できるようになるというメリットがある。例えば、後に扱うように、実質無限次元の基底関数や記号を対象とする基底関数などを背景においたモデルを構築できるようになる。

まずは基底関数からカーネル関数を構築する方法をおさらいする。図6.1はいくつかの基底関数を仮定したときのカーネル関数 $k(x, x')$ (x, x' は図示) のグラフである。



$x = -1$ あたりを考慮する。

$$\phi(x) = \begin{pmatrix} 0.1 \\ 0.7 \\ 0.2 \\ 0.0 \\ 0.0 \end{pmatrix} \quad \phi(x) = \begin{pmatrix} 0 \\ 0.6 \\ 0.6 \\ 0.1 \\ 0 \end{pmatrix}$$

$$\text{よって, } k(x, x') = \phi(x)^T \phi(x') = \langle \text{内積値} \rangle$$

(感想)

図の見方に注意が必要だったが、理解の仕方としては「こんなグラフになるのね」程度で良いかと思われる。似た入力に対して高い数値になる、と言ったが、同じ値の時に常に最大になるわけとも限らない点が興味深い。例えば6.12式のような内積の二乗をカーネルとする場合、「同じベクトル同士」よりも「同じベクトル方向を向いて、なおかつノルムが大きいベクトル」の方が類似度が高いという解釈になるみたい。

以降はカーネル関数を直接定義する手法を紹介する。簡単な例として、次のようなカーネル関数を考える。

$$k(x, z) = (x^T z)^2$$

x, z が2次元ベクトルと仮定すると、これはカーネル関数の定義を満たすことを確認できる。

$$\begin{aligned} k(x, z) &= (x^T z)^2 = (x_1 z_1 + x_2 z_2)^2 \\ &= x_1^2 z_1^2 + 2 x_1 z_1 x_2 z_2 + x_2^2 z_2^2 \\ &= \begin{pmatrix} x_1^2 & \sqrt{2} x_1 x_2 & x_2^2 \end{pmatrix} \begin{pmatrix} z_1^2 & \sqrt{2} z_1 z_2 & z_2^2 \end{pmatrix}^T \\ &= \phi(x)^T \phi(z) \end{aligned} \quad (6.12)$$

基底関数の内積の形で表わされた。

カーネル関数の性質を満たしているかを確認するには、直接基底関数の内積で表されることを示す以外にも簡単な方法がある。既にカーネル関数とわかっている関数 $k_a(), k_b()$ を使って、(6.13-6.22) の式で表現できることを確かめられれば良い。

$$k(\mathbf{x}, \mathbf{x}') = ck_1(\mathbf{x}, \mathbf{x}') \quad (6.13)$$

$$k(\mathbf{x}, \mathbf{x}') = f(\mathbf{x})k_1(\mathbf{x}, \mathbf{x}')f(\mathbf{x}') \quad (6.14)$$

$$k(\mathbf{x}, \mathbf{x}') = q(k_1(\mathbf{x}, \mathbf{x}')) \quad (6.15)$$

$$k(\mathbf{x}, \mathbf{x}') = \exp(k_1(\mathbf{x}, \mathbf{x}')) \quad (6.16)$$

$$k(\mathbf{x}, \mathbf{x}') = k_1(\mathbf{x}, \mathbf{x}') + k_2(\mathbf{x}, \mathbf{x}') \quad (6.17)$$

$$k(\mathbf{x}, \mathbf{x}') = k_1(\mathbf{x}, \mathbf{x}')k_2(\mathbf{x}, \mathbf{x}') \quad (6.18)$$

$$k(\mathbf{x}, \mathbf{x}') = k_3(\phi(\mathbf{x}), \phi(\mathbf{x}')) \quad (6.19)$$

$$k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T \mathbf{A} \mathbf{x}' \quad (6.20)$$

$$k(\mathbf{x}, \mathbf{x}') = k_a(\mathbf{x}_a, \mathbf{x}'_a) + k_b(\mathbf{x}_b, \mathbf{x}'_b) \quad (6.21)$$

$$k(\mathbf{x}, \mathbf{x}') = k_a(\mathbf{x}_a, \mathbf{x}'_a)k_b(\mathbf{x}_b, \mathbf{x}'_b). \quad (6.22)$$

(6.12) では2次の項のみ含む基底関数が現れたが、 $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T \mathbf{x}' + c)^M$ のように構成すると、0次の項からM次の項まで全て現れるので、よく使われるカーネルである。

別のよく使われるカーネルとしてガウスカネルがある(scikit-learn の SVMクラスのデフォルトのカーネルである kernel='rbf' もガウスカネルである)。

$$k(\mathbf{x}, \mathbf{x}') = \exp(-\|\mathbf{x} - \mathbf{x}'\|^2 / 2\sigma^2)$$

これがカーネル関数であるかは次のように確かめられる。

$$\begin{aligned} k(\mathbf{x}, \mathbf{x}') &= \exp(-\|\mathbf{x} - \mathbf{x}'\|^2 / 2\sigma^2) \\ &= \exp\left\{-\frac{1}{2\sigma^2} (\mathbf{x}^T \mathbf{x} - 2\mathbf{x}^T \mathbf{x}' + \mathbf{x}'^T \mathbf{x}')\right\} \\ &= \underbrace{\exp\left(-\frac{\mathbf{x}^T \mathbf{x}}{2\sigma^2}\right)}_{f(\mathbf{x})} \underbrace{\exp\left(\frac{1}{\sigma^2} \mathbf{x}^T \mathbf{x}'\right)}_{\text{カーネル関数 (6.13)}} \underbrace{\exp\left(-\frac{\mathbf{x}'^T \mathbf{x}'}{2\sigma^2}\right)}_{f(\mathbf{x}')} \quad (6.23) \\ &\quad \underbrace{\hspace{10em}}_{\text{カーネル関数 (6.14)}} \end{aligned}$$

(演習6.11)

ガウスカーネルに対応する基底関数が無限次元であることを確かめる。

(6.23) の直交中の項は次のように展開できる。

$$\begin{aligned} \exp\left(\frac{1}{2\sigma^2} x^\top x'\right) &= \sum_{n=0}^{\infty} \frac{1}{n!} \left(\frac{1}{2\sigma^2} x^\top x'\right)^n \\ &= \sum_{n=0}^{\infty} \left(\frac{1}{n!}\right) \left(\frac{1}{2\sigma^2}\right)^n \underbrace{(x^\top x')^n}_{\text{展開}} \end{aligned}$$

ここで、

$$\begin{aligned} (x^\top x')^n &= (x_1 x'_1 + x_2 x'_2 + \dots + x_d x'_d)^n \\ &= (x_1 x'_1)^n + (x_1 x'_1)^{n-1} (x_2 x'_2) + \dots + (x_d x'_d)^n \\ &\quad \underbrace{\text{d個の中から} n \text{個抜き出す全組み合わせ}} \\ &= \prod_{i=1}^d x_i^{\tilde{g}_1(i)} \cdot x_i^{\tilde{g}_1(i)} + \prod_{i=1}^d x_i^{\tilde{g}_2(i)} x_i'^{\tilde{g}_2(i)} + \dots + \prod_{i=1}^d x_i^{\tilde{g}_m(i)} x_i'^{\tilde{g}_m(i)} \\ &\quad \tilde{g}_1(i) \text{ は、組み合わせ} \tilde{g}_1 \text{ における } x_i \text{ の次数} \\ &= \sum_{\tilde{g} \in [d]^n} \left(\prod_{i=1}^d x_i^{\tilde{g}(i)} \right) \left(\prod_{i=1}^d x_i'^{\tilde{g}(i)} \right) \\ &\quad \underbrace{\text{d個から} n \text{個選ぶ全組み合わせ}} \\ &= \varphi_n(x)^\top \varphi_n(x') \quad \because \text{積の総和なので、内積の形で表せる。} \end{aligned}$$

よって、

$$\begin{aligned} \exp\left(\frac{1}{2\sigma^2} x^\top x'\right) &= \sum_{n=0}^{\infty} \left\{ \frac{1}{n!} \left(\frac{1}{2\sigma^2}\right)^n \right\}^{\frac{1}{2}} \varphi_n(x)^\top \varphi_n(x') \left\{ \frac{1}{n!} \left(\frac{1}{2\sigma^2}\right)^n \right\}^{\frac{1}{2}} \\ &\quad \text{さらに無限次元ベクトルの内積の形で表せる。} \\ &= \dot{\varphi}_n(x)^\top \dot{\varphi}_n(x') \end{aligned}$$

$\dot{\varphi}$ を使えば、 $k(x, x')$ は次のように表せる。

$$\begin{aligned} k(x, x') &= \exp\left(-\frac{x^\top x'}{2\sigma^2}\right) \dot{\varphi}_n(x)^\top \cdot \exp\left(-\frac{x'^\top x}{2\sigma^2}\right) \dot{\varphi}_n(x') \\ &= \phi(x)^\top \phi(x') \end{aligned}$$

以降ではいくつかの特殊なカーネル関数の形とその解釈を紹介する。

■ 記号を扱うカーネル関数

与えられるデータがたとえば集合であるとき、得られた二つの集合に対してもカーネル関数を定めることができる。例えば以下のようなものがある。

$$k(A_1, A_2) = 2^{|A_1 \cap A_2|} \quad (6.27)$$

これがカーネルであることは、基底関数 $\phi_v(A) = \begin{cases} 1 & v \subseteq A \text{ のとき} \\ 0 & \text{それ以外} \end{cases}$ を駆使して構成できる (演習6.12)。全集合を \mathcal{S} として、その冪集合を $2^{\mathcal{S}}$ とする。 \mathcal{S} の部分集合、 A における基底関数を次のように構成する。

$$\Phi(A) = \begin{pmatrix} \phi_{v_1}(A) \\ \phi_{v_2}(A) \\ \vdots \\ \phi_{v_n}(A) \end{pmatrix} \quad \text{ここで } v_1, v_2, \dots, v_n \text{ は } 2^{\mathcal{S}} \text{ の要素の列挙である。}$$

(6.27) のカーネル関数がこの基底関数に対応することを次の簡単な例で確かめる。

$$\begin{aligned} \mathcal{S} &= \{0, 1, 2\} & A_1 &= \{0, 1\} & A_2 &= \{1\} \\ 2^{\mathcal{S}} &= \{ \emptyset, \{0\}, \{1\}, \{2\}, \{0, 1\}, \{0, 2\}, \{1, 2\}, \{0, 1, 2\} \} \\ \text{基底関数は次の形になる。} \end{aligned}$$

$$\Phi(A) = \begin{pmatrix} \phi_{\emptyset}(A) \\ \phi_{\{0\}}(A) \\ \vdots \\ \phi_{\{0, 1, 2\}}(A) \end{pmatrix}$$

$$\Phi(A_1)^T \Phi(A_2) = \sum_{v \in 2^{\mathcal{S}}} \underbrace{\phi_v(A_1) \phi_v(A_2)}_{v \subseteq A_1 \text{ かつ } v \subseteq A_2 \text{ のときのみ 1}}$$

$$= \sum_{v \in 2^{\mathcal{S}}} \phi_v(A_1 \cap A_2)$$

$v \in 2^{\mathcal{S}}$ のうち、 $A_1 \cap A_2$ に含まれるもののカウント = $A_1 \cap A_2$ の冪集合のサイズ

$$= 2^{|A_1 \cap A_2|} = k(A_1, A_2)$$

ちなみに、実際に計算してみても一致する。

$$\Phi(A_1)^T \Phi(A_2) = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}^T \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} = 2$$

$$k(A_1, A_2) = 2^{|1111|} = 2$$

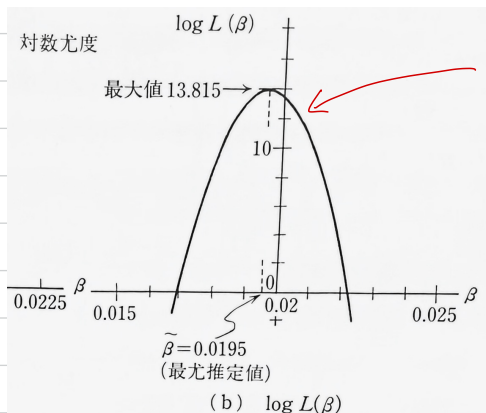
■ 生成モデルを使うカーネル関数

p.8 を読む形で対応させてください。

■ フィッシャーカーネル

教科書読むだけでとさっぱり意味がわからないので、とりあえず自然科学の統計学p120を参照する。ただそれでも結局理解できなかった。

自然科学の統計学では、フィッシャー情報量は推定量としての良さの指標と解釈できた。



$-\frac{\partial^2}{\partial \beta^2} \log L$ は放物線の曲率に相当する。値が大きい程急峻になり、推定値としての確信度が高まる。

カーネル関数の形だけ紹介する。詳細は読み合わせとし、なにかアイデアが浮かぶことを祈る。

$$k(x, x') = g(\phi(x))^T F^{-1} g(\phi, x')$$

$$\text{ここで, } g(\phi, x) = \nabla_{\phi} \ln p(x|\phi)$$

$$F = E_x(g(\phi, x) g(\phi, x)^T) \quad (\text{フィッシャー情報行列})$$

簡略版は、

$$k(x, x') = g(\phi, x)^T g(\phi, x')$$

■ シグモイドカーネル

実用的によく使用されているカーネルの一つ。詳細は教科書参照

$$k(x, x') = \tanh(a x^T x' + b)$$

(感想)

記号に対するカーネル関数までは、カーネル関数のやりたいことが見えていたが、生成モデルを使うカーネル関数あたりから、どういう時にこれらを使えばいいのかが不明になってきた。(そもそもカーネル関数を紹介している章なのでその理解までは必要とされていない気がするが...) 後半のガウス仮定では、ベイズ線形モデルでガウスカーネルを使用する背景が明確になるようなので、カーネルの選定基準についてはこの辺りで何かヒントを得られるといいなーと考えてます。