
ガウス過程と機械学習入門

持橋大地 大羽成征

統計数理研究所 京都大学

daichi@ism.ac.jp oba@i.kyoto-u.ac.jp

大阪大学 数理・データ科学教育研究センター(MMDS)
スプリングキャンプ
2019-3-11 (月)

今日の講義と実習のスケジュール

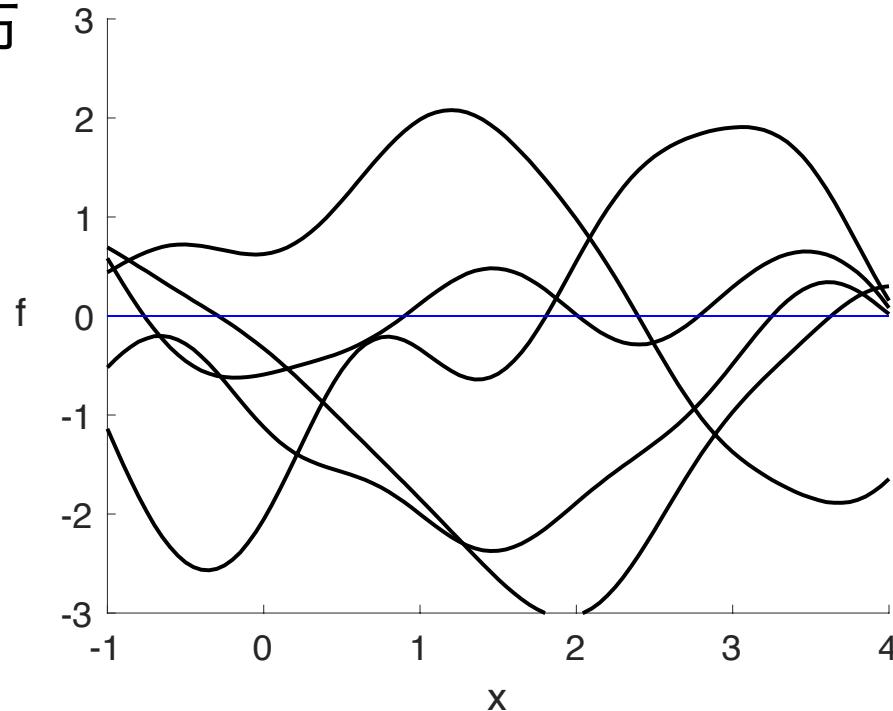
- 1限目：イントロ（大羽）、線形回帰モデル（持橋）
[演習] 線形回帰モデルの計算と評価
- 2限目：確率的生成モデルとベイズ推定（大羽）
[演習] ベイズ推定の基礎、多変量ガウス分布
- 3限目：ガウス過程、ハイパーパラメータの学習（持橋）
[演習] ガウス過程回帰モデルの計算、ハイパーパラメータ最適化
- 4限目：ガウス過程回帰の高速化（大羽）
補助変数法・変分ベイズ法
[演習] GPyを使った補助変数法の演習

実習用Slack

- <https://gpsc2019.slack.com/>
- 質問があれば #questions に、
何か他の参加者にも役立つ情報があれば #general に
書いて下さい
- 参加者同士で、上手く情報交換しましょう！
(集まって受講する意義はそこになります！)
- 演習の際には、まわりの人とどんどん相談して下さい

ガウス過程とは

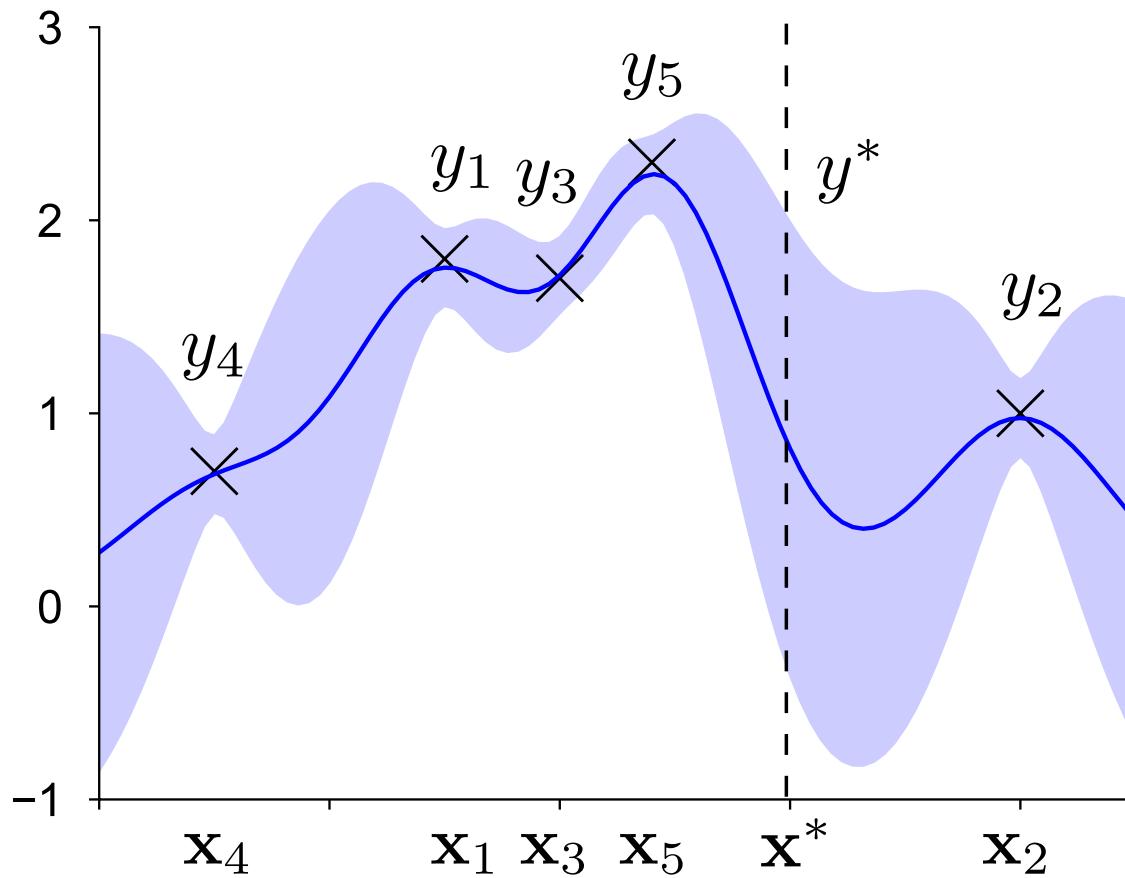
- ガウス過程 (Gaussian process)... 関数を生成する確率分布



- 関数：入力 $x \mapsto y$ への写像
- 時間 t の関数としてみれば、軌跡
- 数学的には、関数解析で扱われる対象

代表的な使い方: ガウス過程回帰

- Gaussian process regression (GPR)
 - (x, y) のペアが与えられた時、新しい x^* に対して y を予測



回帰とは

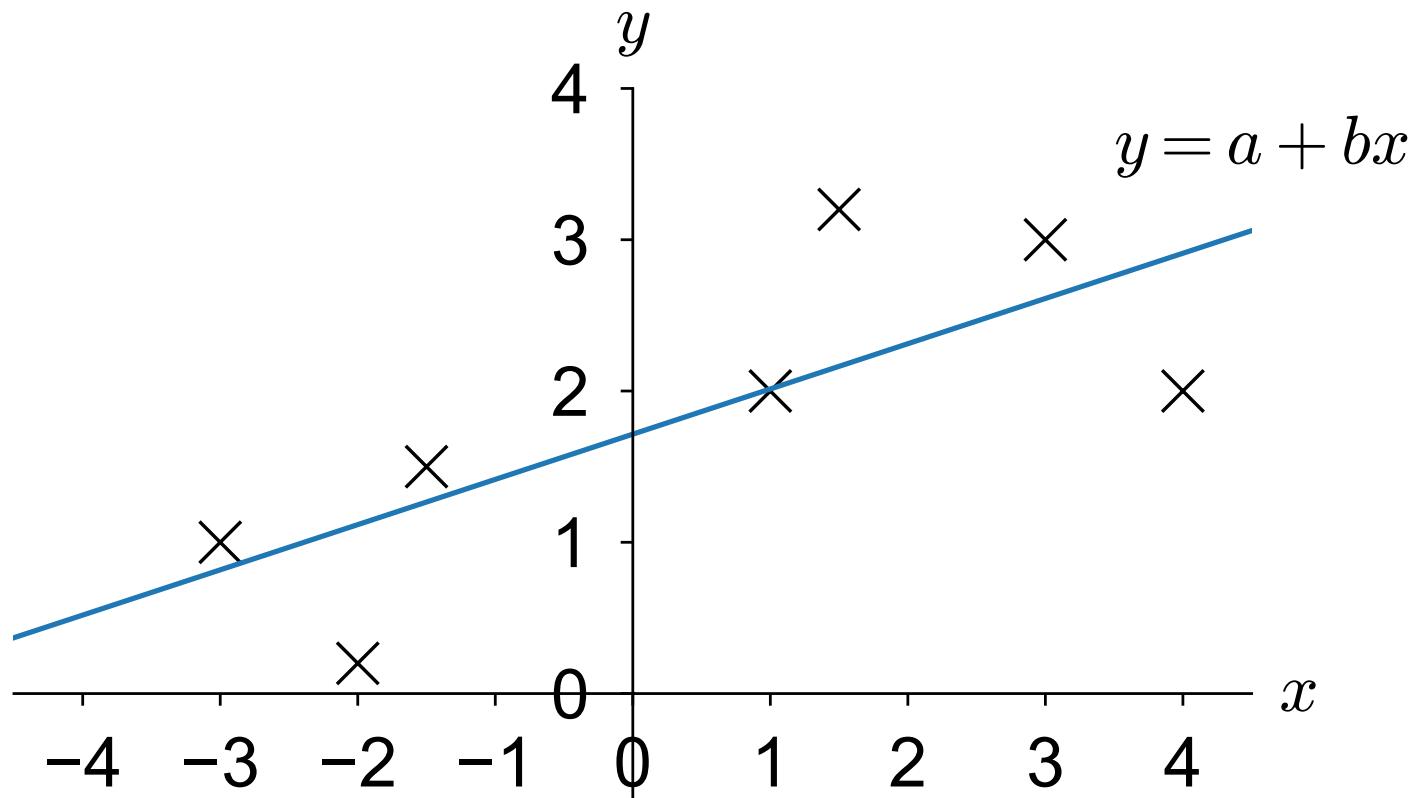
- 回帰 (regression) : 入力 x から出力 y を予測すること
 - 入力 $x \in \mathcal{X}$ は何でもよい
 - 典型的には、ベクトル $x \in \mathbb{R}^D$
 - グラフや文字列でも原理的にはOK (例: 化学構造)
 - ガウス過程なら、カーネルが定義できればよい
- 出力 $y \in \mathbb{R}$ は連続値
 - 出力がベクトルの場合は複雑なので今回は扱わない
 - 単純には、出力の次元ごとに別の関数を考えればOK
 - 出力が離散的なカテゴリ $y \in \{1, 2, \dots, K\}$ の場合は分類あるいは識別といい、別の問題になる
 - ガウス過程を使ったガウス過程識別モデルもある (教科書参照)

回帰の例

- 薬の投与後の時間から、血圧を予測
- 国の様々な経済指標から、GDPを予測
- 東京圏の座標(x,y)から、地価を予測
- 天体の吸収スペクトルから、未観測の波長の値を予測
- ... (皆さんの方が詳しいはず！)

単回帰モデル (simple regression)

- 最も単純な回帰 : $y = a + bx$
- aとbをどうやって決める?



単回帰モデル (2)

- データ $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$ があったとする.
- 各 x_n に対する予測値 \hat{y}_n は、一次式

$$\hat{y}_n = a + b x_n$$

- 観測値との差は

$$y_n - \hat{y}_n = y_n - (a + b x_n)$$

- これを最小にしたい！

単回帰モデル (3)

- $n=1,2,\dots,N$ について、
誤差 = $y_n - \hat{y}_n$ → 誤差の総和を最小にしたい
- 誤差は負のこともあるので、二乗した二乗誤差を最小化 (最小二乗法) :

$$E = \sum_{n=1}^N (y_n - \hat{y}_n)^2 = \sum_{n=1}^N (y_n - (a + bx_n))^2$$

を最小にする a, b を求める

単回帰モデル (4)

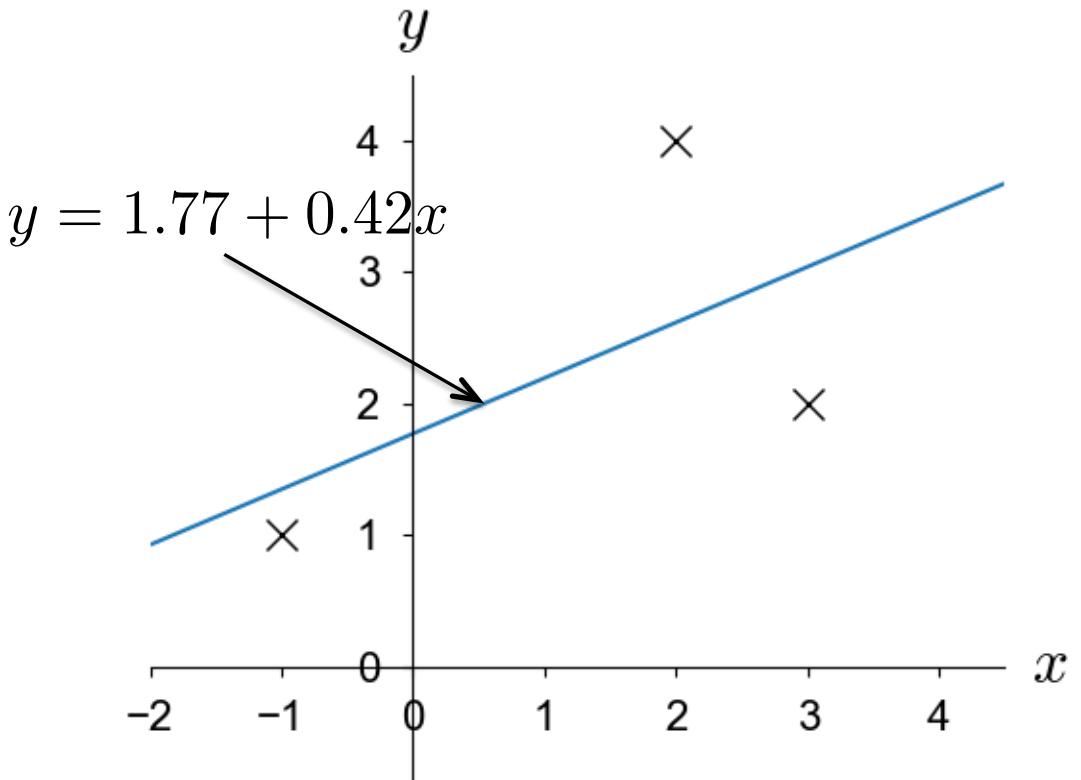
- E の極小点では a, b についての偏微分は0になるので、

$$\begin{aligned}\frac{\partial E}{\partial a} &= \frac{\partial}{\partial a} \sum_{n=1}^N (y_n - (a + bx_n))^2 \\ &= \frac{\partial}{\partial a} \sum_{n=1}^N (y_n^2 + a^2 + b^2x_n^2 - 2ay_n - 2abx_n + 2bx_ny_n) = 0\end{aligned}$$

$$\begin{aligned}\frac{\partial E}{\partial b} &= \frac{\partial}{\partial b} \sum_{n=1}^N (y_n - (a + bx_n))^2 \\ &= \frac{\partial}{\partial b} \sum_{n=1}^N (y_n^2 + a^2 + b^2x_n^2 - 2ay_n - 2abx_n + 2bx_ny_n) = 0\end{aligned}$$

- これを解いて、
$$a = \frac{\sum_n x_n^2 \sum_n y_n - \sum_n x_n \sum_n x_n y_n}{N \sum_n x_n^2 - (\sum_n x_n)^2}$$
$$b = \frac{N \sum_n x_n y_n - \sum_n x_n \sum_n y_n}{N \sum_n x_n^2 - (\sum_n x_n)^2}$$

単回帰モデルの計算例



一番単純な場合：
データD=
 $\{(3,2),(2,4),(-1,1)\}$

$$\sum_{n=1}^3 x_n = 3 + 2 - 1 = 4$$

$$\sum_{n=1}^3 y_n = 2 + 4 + 1 = 7$$

$$\sum_{n=1}^3 x_n^2 = 9 + 4 + 1 = 14$$

$$\sum_{n=1}^3 x_n y_n = 3 \cdot 2 + 2 \cdot 4 + (-1) \cdot 1 = 13$$

- 公式に代入して、
 $a = 1.77, b = 0.42$

重回帰モデル

- 入力 x が多次元なら? → 重回帰 (multiple regression)

$$\mathbf{x} = (x_1, x_2, \dots, x_D)^T \text{ のとき、}$$

$$y = w_0 + w_1 x_1 + w_2 x_2 + \dots + w_D x_D$$

- 二乗誤差は、

$$(y - \hat{y})^2 = (y - (w_0 + w_1 x_1 + w_2 x_2 + \dots + w_D x_D))^2$$

- これを最小化

→ $E = \sum_{n=1}^N (y_n - \hat{y}_n)^2$ を w_0, w_1, \dots, w_D について
微分して0とおき、連立方程式を解けばよい。

もっと見通しよく!

- \mathbf{x} を新しく $\mathbf{x} = (1, x_1, x_2, \dots, x_D)$ 、
重みベクトルを $\mathbf{w} = (w_0, w_1, w_2, \dots, w_D)$ と表せば、

$$\begin{aligned}\hat{y} &= w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_D x_D \\ &= (w_0, w_1, w_2, \dots, w_D) \begin{pmatrix} 1 \\ x_1 \\ \vdots \\ x_D \end{pmatrix} \\ &= \mathbf{w}^T \mathbf{x}\end{aligned}$$

もっと見通しよく! (2)

- よって、 $n=1,2,\dots,N$ について縦に並べれば、

$$\begin{pmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_N \end{pmatrix} = \begin{pmatrix} \mathbf{w}^T \mathbf{x}_1 \\ \mathbf{w}^T \mathbf{x}_2 \\ \vdots \\ \mathbf{w}^T \mathbf{x}_N \end{pmatrix} = \begin{pmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_N^T \end{pmatrix} \mathbf{w}$$

計画行列
という

- つまり、

$$\begin{pmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_N \end{pmatrix} = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1D} \\ 1 & x_{21} & x_{22} & \cdots & x_{2D} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{N1} & x_{N2} & \cdots & x_{ND} \end{pmatrix} \begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \\ w_D \end{pmatrix}$$

$\hat{\mathbf{y}} = \mathbf{X}\mathbf{w}$
と書ける!

$\hat{\mathbf{y}}$

\mathbf{X}

\mathbf{w}

行列・ベクトル表現

$$E = \sum_{n=1}^N (y_n - \hat{y}_n)^2 = (y_1 - \hat{y}_1, \dots, y_N - \hat{y}_N) \begin{pmatrix} y_1 - \hat{y}_1 \\ \vdots \\ y_N - \hat{y}_N \end{pmatrix}$$

- なので、

$$\begin{aligned} E &= (\mathbf{y} - \mathbf{X}\mathbf{w})^T(\mathbf{y} - \mathbf{X}\mathbf{w}) \\ &= \mathbf{y}^T(\mathbf{y} - \mathbf{X}\mathbf{w}) - (\mathbf{X}\mathbf{w})^T(\mathbf{y} - \mathbf{X}\mathbf{w}) \\ &= \mathbf{y}^T\mathbf{y} - 2\mathbf{w}^T(\mathbf{X}^T\mathbf{y}) + \mathbf{w}^T\mathbf{X}^T\mathbf{X}\mathbf{w} \end{aligned}$$

- よって、 $\frac{\partial E}{\partial \mathbf{w}} = \mathbf{0}$ を求めればよい。

ベクトルによる微分

$$\begin{cases} \mathbf{w} = (w_1, w_2, \dots, w_D) \\ \mathbf{x} = (x_1, x_2, \dots, x_D) \end{cases}$$

- のとき、 $\mathbf{w}^T \mathbf{x} = w_1 x_1 + w_2 x_2 + \dots + w_D x_D$ を
 - w_1 で微分すると、 $\frac{\partial}{\partial w_1} \mathbf{w}^T \mathbf{x} = x_1$
 - w_2 で微分すると、 $\frac{\partial}{\partial w_2} \mathbf{w}^T \mathbf{x} = x_2$
 -
- よって、
$$\frac{\partial}{\partial \mathbf{w}} \mathbf{w}^T \mathbf{x} = \mathbf{x} .$$

二次形式の微分

$$\mathbf{w}^T \mathbf{A} \mathbf{w} = \begin{pmatrix} \mathbf{w}^T \end{pmatrix} \begin{pmatrix} & \\ & \mathbf{A} \end{pmatrix} \begin{pmatrix} \mathbf{w} \end{pmatrix} = \sum_{i=1}^D \sum_{j=1}^D A_{ij} w_i w_j$$

- このとき、

$$\frac{\partial}{\partial \mathbf{w}} \mathbf{w}^T \mathbf{A} \mathbf{w} = (\mathbf{A} + \mathbf{A}^T) \mathbf{w}$$

- 証明は、教科書を参照

重回帰モデルの解

$$E = \mathbf{y}^T \mathbf{y} - 2\mathbf{w}^T (\mathbf{X}^T \mathbf{y}) + \mathbf{w}^T \mathbf{X}^T \mathbf{X} \mathbf{w}$$

- を \mathbf{w} で微分すれば、

$$\frac{\partial}{\partial \mathbf{w}} \mathbf{w}^T \mathbf{y} = \mathbf{y}, \quad \frac{\partial}{\partial \mathbf{w}} \mathbf{w}^T \mathbf{X}^T \mathbf{X} \mathbf{w} = 2\mathbf{X}^T \mathbf{X} \mathbf{w} \quad \text{より}$$

$$\frac{\partial E}{\partial \mathbf{w}} = -2\mathbf{X}^T \mathbf{y} + 2\mathbf{X}^T \mathbf{X} \mathbf{w} = \mathbf{0}$$

- よって

$$\mathbf{X}^T \mathbf{X} \mathbf{w} = \mathbf{X}^T \mathbf{y} \quad (\text{正規方程式})$$

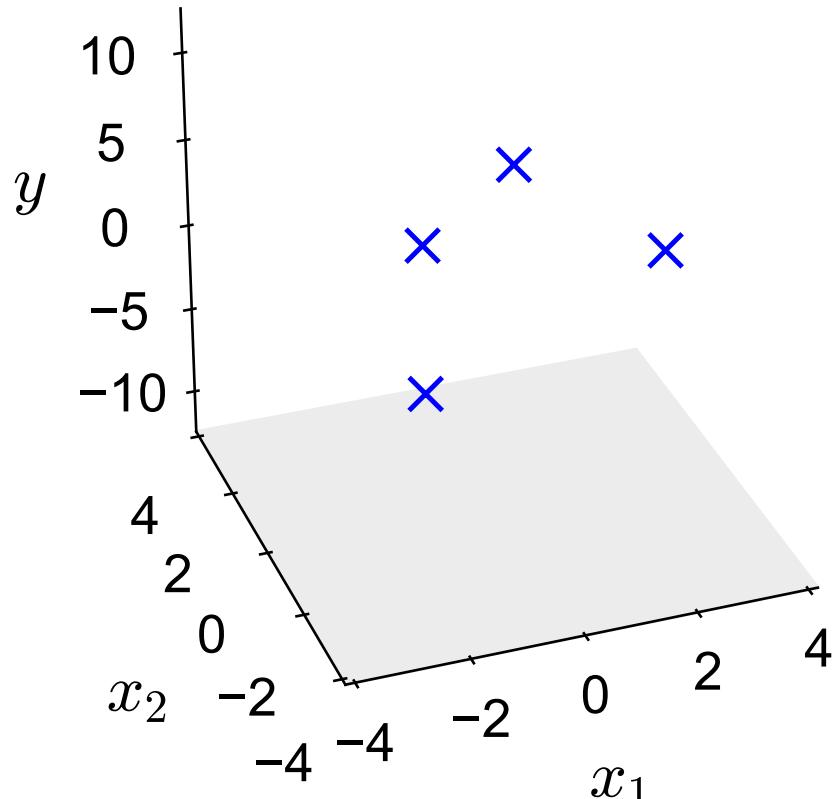
$$\therefore \mathbf{w} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}.$$

重回帰モデルの解

重回帰モデルの計算例

- データが下のとき、

$$\mathcal{D} = \{((1, 2), 4), ((-1, 1), 2), ((3, 0), 1), ((-2, -2), -1)\}$$



x_1	x_2	y
1	2	4
-1	1	2
3	0	1
-2	-2	-1

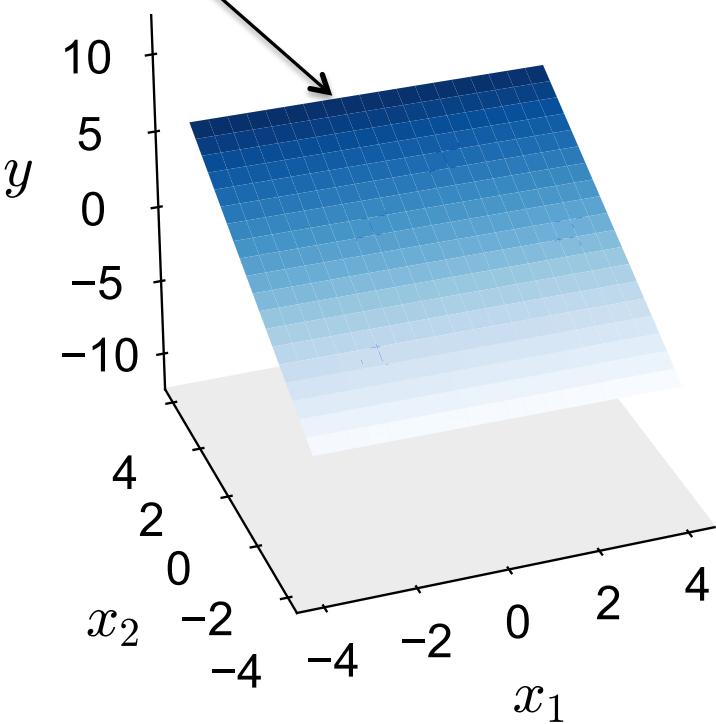
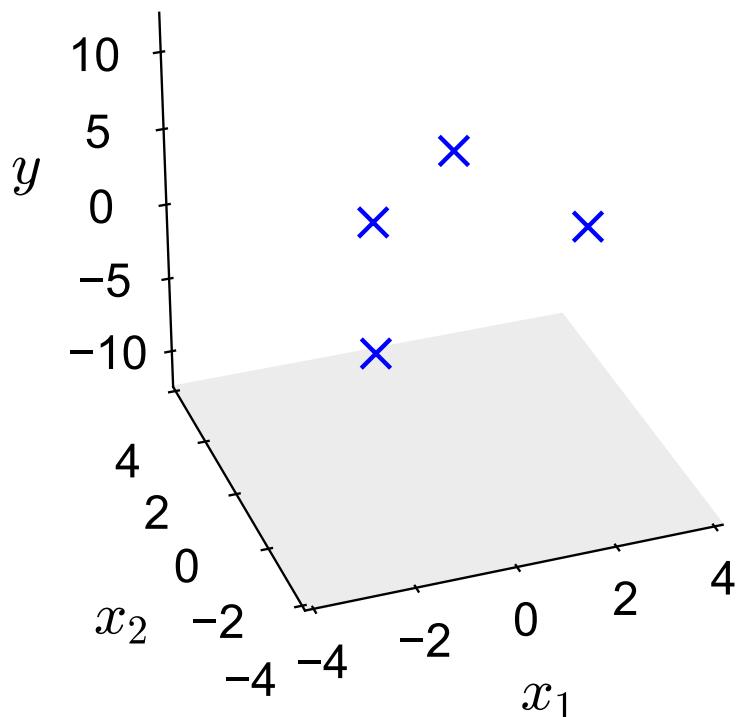
$$\mathbf{X} = \begin{pmatrix} 1 & 1 & 2 \\ 1 & -1 & 1 \\ 1 & 3 & 0 \\ 1 & -2 & -2 \end{pmatrix}$$

重回帰モデルの計算例 (2)

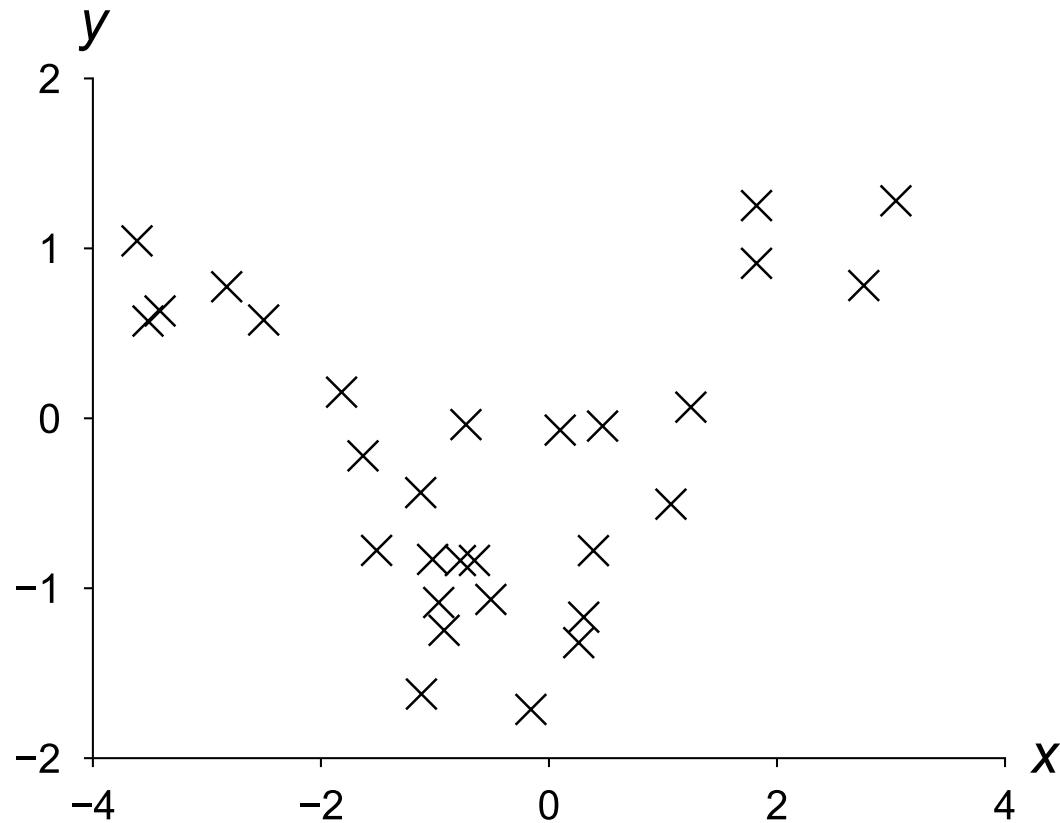
- よって、重みベクトルwの解は

$$\mathbf{w} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} = (1.202 \ -0.016 \ 1.209)^T$$

$$y = 1.202 - 0.016x_1 + 1.209x_2$$



もっと複雑にしたい！



- 直線や平面で表せない関係も多いのでは？
- 

関数をもっと複雑にすればよい！

線形回帰モデル

$$y = w_0 + w_1 x + w_2 x^2 \quad y = w_0 + w_1 x + w_2 \sin(x)$$
$$= \underbrace{(w_0 \quad w_1 \quad w_2)}_{\mathbf{w}^T} \underbrace{\begin{pmatrix} 1 \\ x \\ x^2 \end{pmatrix}}_{\phi(x)} \quad = \underbrace{(w_0 \quad w_1 \quad w_2)}_{\mathbf{w}^T} \underbrace{\begin{pmatrix} 1 \\ x \\ \sin(x) \end{pmatrix}}_{\phi(x)}$$

- どれも、係数ベクトルの線形式として書ける！
 - $\mathbf{x} \mapsto \phi(\mathbf{x})$ の変換は一意に決まる
 - $y = \mathbf{w}^T \phi(\mathbf{x}) \cdots$ 線形回帰モデル (linear regression model)
- 注: 一般化線形モデル(GLM)では、さらにリンク関数 f があって $y = f(\mathbf{w}^T \phi(\mathbf{x}))$ となる

線形回帰モデル (2)

$$y = \mathbf{w}^T \phi(\mathbf{x}) \quad (= \phi(x)^T \mathbf{w})$$

- は、 x が $\phi(x)$ に変わっただけで重回帰モデル $y = \mathbf{w}^T \mathbf{x}$ と同じなので、たとえば $\phi(x) = (1, x, x^2, x^3)$ のとき、上をN個並べれば

$$\underbrace{\begin{pmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_N \end{pmatrix}}_{\hat{\mathbf{y}}} = \underbrace{\begin{pmatrix} \phi(\mathbf{x}_1)^T \\ \phi(\mathbf{x}_2)^T \\ \vdots \\ \phi(\mathbf{x}_N)^T \end{pmatrix}}_{\Phi} \mathbf{w} = \underbrace{\begin{pmatrix} 1 & x_1 & x_1^2 & x_1^3 \\ 1 & x_2 & x_2^2 & x_2^3 \\ \vdots & & & \vdots \\ 1 & x_N & x_N^2 & x_N^3 \end{pmatrix}}_{\Phi} \underbrace{\begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \end{pmatrix}}_{\mathbf{w}}$$

線形回帰モデル (3)

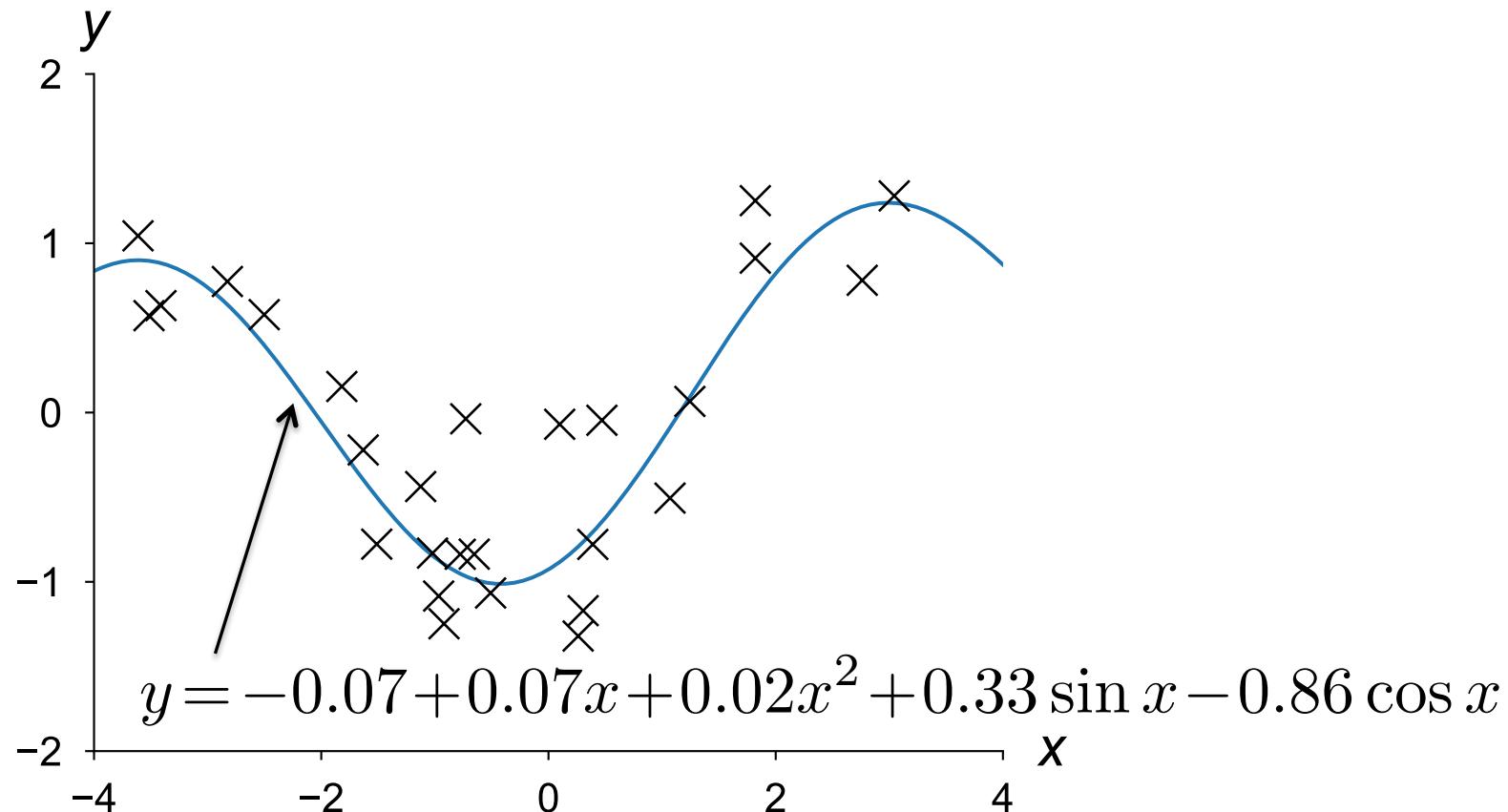
- つまり一般に、線形回帰モデルは以下のように書ける

$$\underbrace{\begin{pmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_N \end{pmatrix}}_{\hat{\mathbf{y}}} = \underbrace{\begin{pmatrix} 1 & \phi_1(x_1) & \cdots & \phi_H(x_1) \\ 1 & \phi_1(x_2) & \cdots & \phi_H(x_2) \\ \vdots & \vdots & & \vdots \\ 1 & \phi_1(x_N) & \cdots & \phi_H(x_N) \end{pmatrix}}_{\Phi} \underbrace{\begin{pmatrix} w_0 \\ w_1 \\ \vdots \\ w_H \end{pmatrix}}_{\mathbf{w}}$$

- 計画行列 Φ を使って、 $\hat{\mathbf{y}} = \Phi \mathbf{w}$ と書ける
- $\mathbf{X} \mapsto \Phi$ 以外は重回帰と同じなので、 $\mathbf{w} = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{y}$

線形回帰モデルの例

- 特徴ベクトルを $\phi(x) = (1, x, x^2, \sin x, \cos x)^T$ として先ほどのデータに適用すると、
 $w = (-0.065, 0.068, 0.022, 0.333, -0.863)^T$ が解



モデルの評価

- どのモデルを選べばよいか?
- 2つのモデルを比べたとき、誤差Eが小さいほどよいモデルとは限らない。

モデルの評価 (2)

- どうやってモデルを評価するべきか?



学習データに使わなかつたテストデータをうまく予測できるか (連續値なら平均二乗誤差 (MSE)でよい)

- テストデータの選び方

- **内挿** : 学習データのうちランダムな20%をテストデータとし、残りの80%から学習して予測できるか
- **外挿** : データの端の点を隠しておき、それを予測できるか

- 1次元では「端」が自明だが、高次元ではそうではない
- 目的により、未来の予測精度が重要な場合はOK

- **クロスバリデーション** : データをN分割してそれぞれをテストデータとしてN回学習し、予測誤差を平均する

リッジ回帰

またはその定数倍

- 重回帰や線形回帰で、 \mathbf{X} に同じ列があると $(\mathbf{X}^T \mathbf{X})^{-1}$ が存在せず、 \mathbf{w} が求められない

例:

$$\mathbf{X} = \begin{pmatrix} 1 & 2 & 4 \\ 1 & 3 & 6 \\ 1 & 4 & 8 \end{pmatrix} \quad \mathbf{X} = \begin{pmatrix} 1 & 2 & 4 \\ 1 & 3 & 6.1 \\ 1 & 4 & 7.9 \end{pmatrix}$$

- 厳密に同じでなくとも、ほとんど同じだと \mathbf{w} の値が極端になる

$$(\mathbf{X}^T \mathbf{X})^{-1} = \begin{pmatrix} 6.33 & 18.00 & -10.00 \\ 18.00 & 254.00 & -130.00 \\ -10.00 & -130.00 & 66.67 \end{pmatrix}$$

- $y = [1 \ -2 \ 3]$ のとき、 $\mathbf{w} = (1.67 \ 53.0 \ -26.67)$

リッジ回帰 (2)

- 重みベクトル w の大きさにペナルティを加える

$$|\mathbf{y} - \mathbf{X}\mathbf{w}|^2 + \alpha|\mathbf{w}|^2 \rightarrow \text{最小化}$$

- $|\mathbf{w}|^2 = \mathbf{w}^T \mathbf{w}$ だから、 \mathbf{w} で微分して0とおくと同様に

$$\mathbf{w} = (\mathbf{X}^T \mathbf{X} + \alpha \mathbf{I})^{-1} \mathbf{X}^T \mathbf{y}$$

- $\mathbf{X}^T \mathbf{X}$ がフルランクでなくとも、対角要素に α が足されているために $\mathbf{X}^T \mathbf{X} + \alpha \mathbf{I}$ はフルランクになり、逆行列が存在
- \mathbf{w} が極端な値にならない

リッジ回帰 (3)

- $y=[1 \ 2 \ 3]$, $\alpha=0.1$ のとき

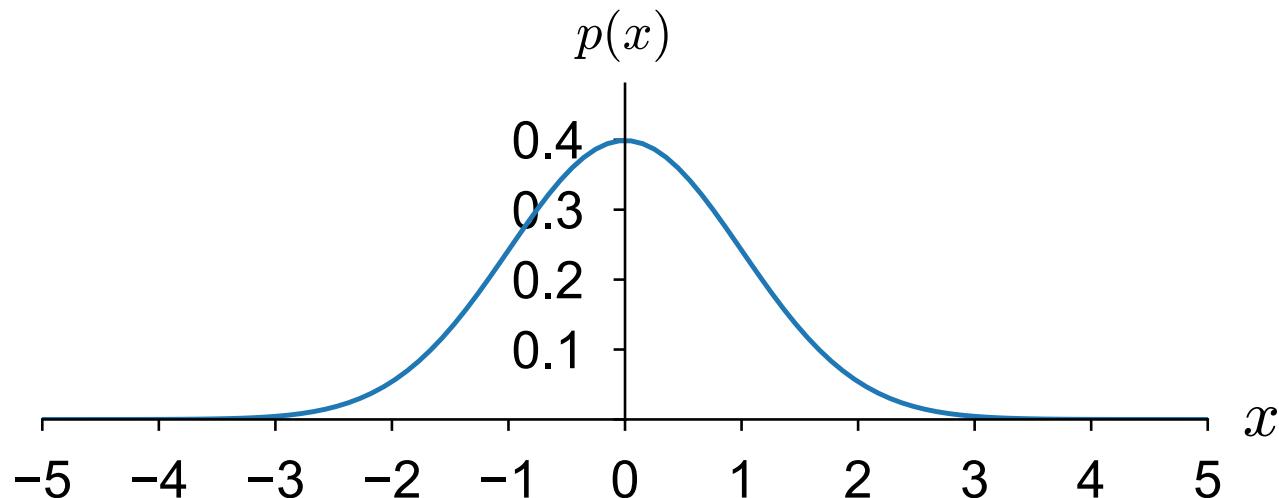
$$\mathbf{w} = (\mathbf{X}^T \mathbf{X} + \alpha \mathbf{I})^{-1} \mathbf{X}^T \mathbf{y}$$

$$= \left(\begin{pmatrix} 3 & 9 & 18 \\ 9 & 29 & 58 \\ 18 & 58 & 116 \end{pmatrix} + 0.1 \times \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \right)^{-1} \begin{pmatrix} 1 & 1 & 1 \\ 2 & 3 & 4 \\ 4 & 6 & 8 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

$$= \begin{pmatrix} -0.656 \\ 0.179 \\ 0.357 \end{pmatrix}$$

ガウス分布

- ガウス分布 (Gaussian distribution) または 正規分布 (normal distribution)：最も基本的な確率分布



$$p(x) = \mathcal{N}(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

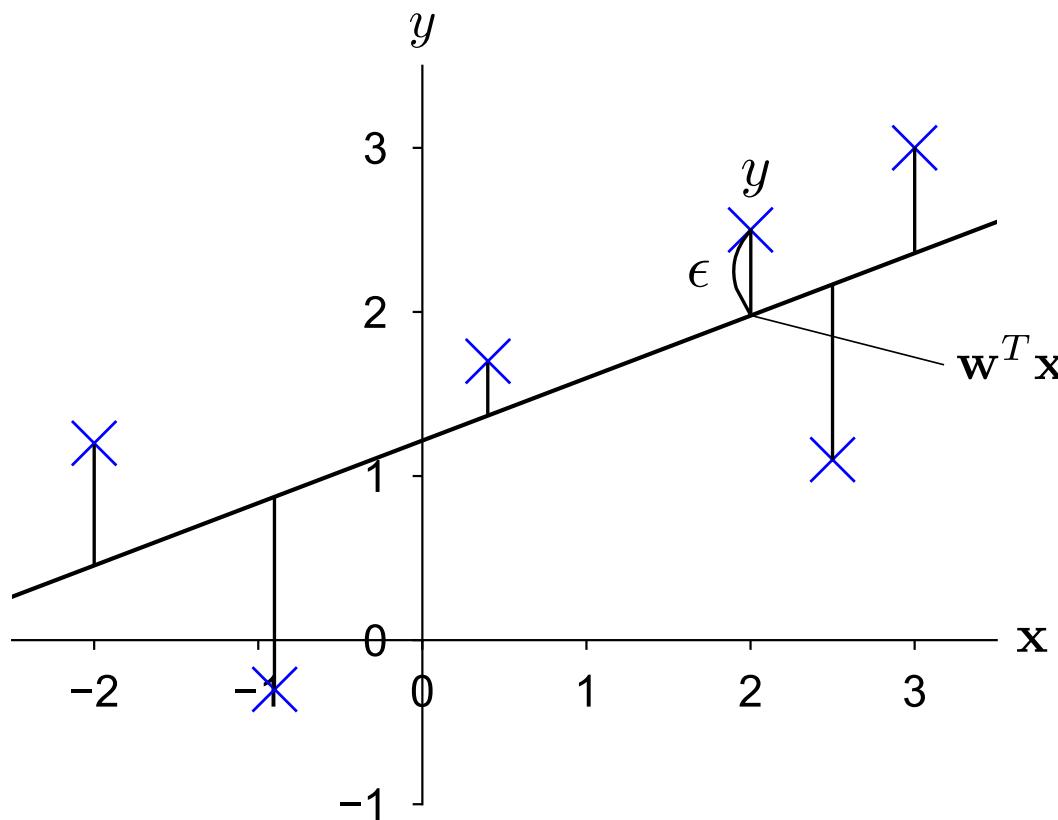
ガウス分布 (2)

- 平均と分散が一定の下で、エントロピー最大となる確率分布
- 任意の同じ分布に従う確率変数の総和は、中心極限定理によりガウス分布に従う→誤差の分布

観測誤差の分布

- 線形回帰モデルで、観測値と予測値との誤差 ϵ がガウス分布に従うとしてみる

$$\epsilon = y - \hat{y} = y - \mathbf{w}^T \phi(\mathbf{x}) \sim \mathcal{N}(0, \sigma^2)$$



よって、

$$y \sim \mathcal{N}(\mathbf{w}^T \phi(\mathbf{x}), \sigma^2)$$

確率モデル
になった！

線形回帰モデルとガウス分布

- よって、 $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)$ が与えられた下での $\mathbf{y} = (y_1, y_2, \dots, y_N)$ の確率は、観測が独立だとすれば

$$p(\mathbf{y}|\mathbf{X}) = \prod_{n=1}^N p(y_n|\mathbf{x}_n) = \prod_{n=1}^N \mathcal{N}(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n), \sigma^2)$$

- 対数をとって、

$$\begin{aligned}\log p(\mathbf{y}|\mathbf{X}) &= \sum_{n=1}^N \log \mathcal{N}(y_n | \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n), \sigma^2) \\ &= \sum_{n=1}^N \left[-\log(\sqrt{2\pi}\sigma) - \frac{1}{2\sigma^2} (y_n - \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n))^2 \right] \\ &= |\mathbf{y} - \Phi \mathbf{w}|^2 + \text{const.}\end{aligned}$$

最小二乗法と同じ！

ガウス分布とリッジ回帰

- w の各要素 w_i がそれぞれ、ガウス分布

$$w_i \sim \mathcal{N}(0, \eta^2)$$

に従うとすれば、

- y と w の同時確率の対数は

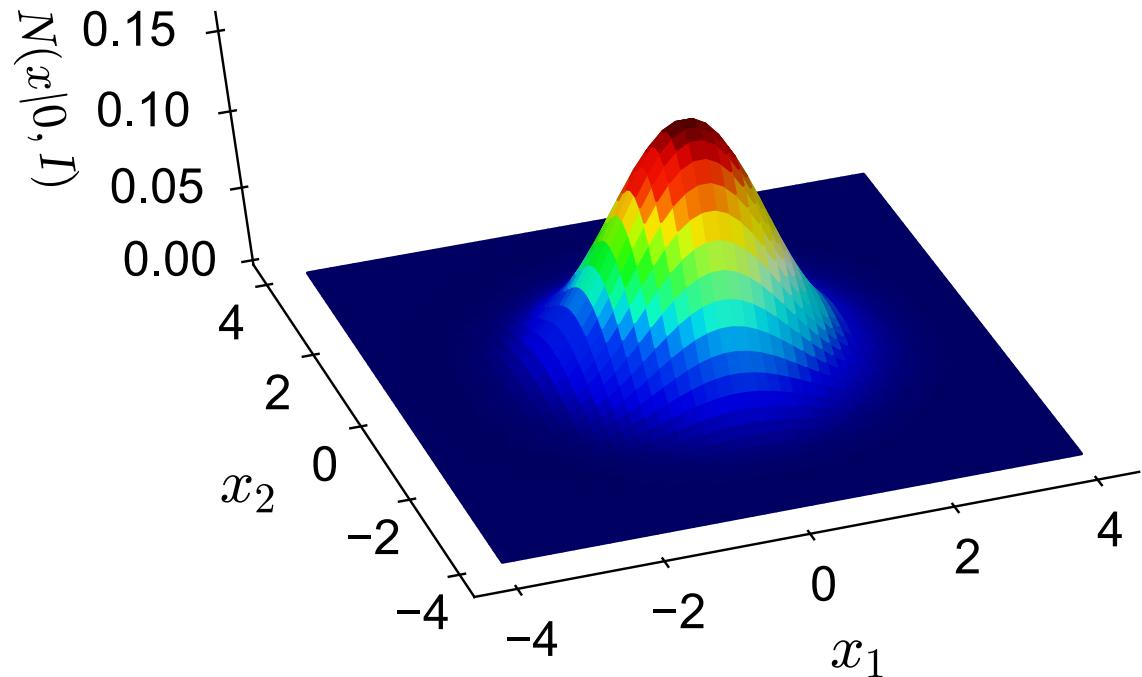
$$\log p(\mathbf{y}, \mathbf{w} | \mathbf{X}) = \log p(\mathbf{y} | \mathbf{X}, \mathbf{w}) p(\mathbf{w})$$

$$\begin{aligned} &= \sum_{n=1}^N \left[-\frac{1}{2\sigma^2} (y_n - \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n))^2 \right] - \frac{1}{2\eta^2} \sum_{i=1}^D w_i^2 \\ &= -(|\mathbf{y} - \boldsymbol{\Phi}\mathbf{w}|^2 + \alpha \mathbf{w}^T \mathbf{w}) \end{aligned}$$

リッジ回帰と同じ！

演習 (線形回帰モデルとガウス分布)

多変量ガウス分布

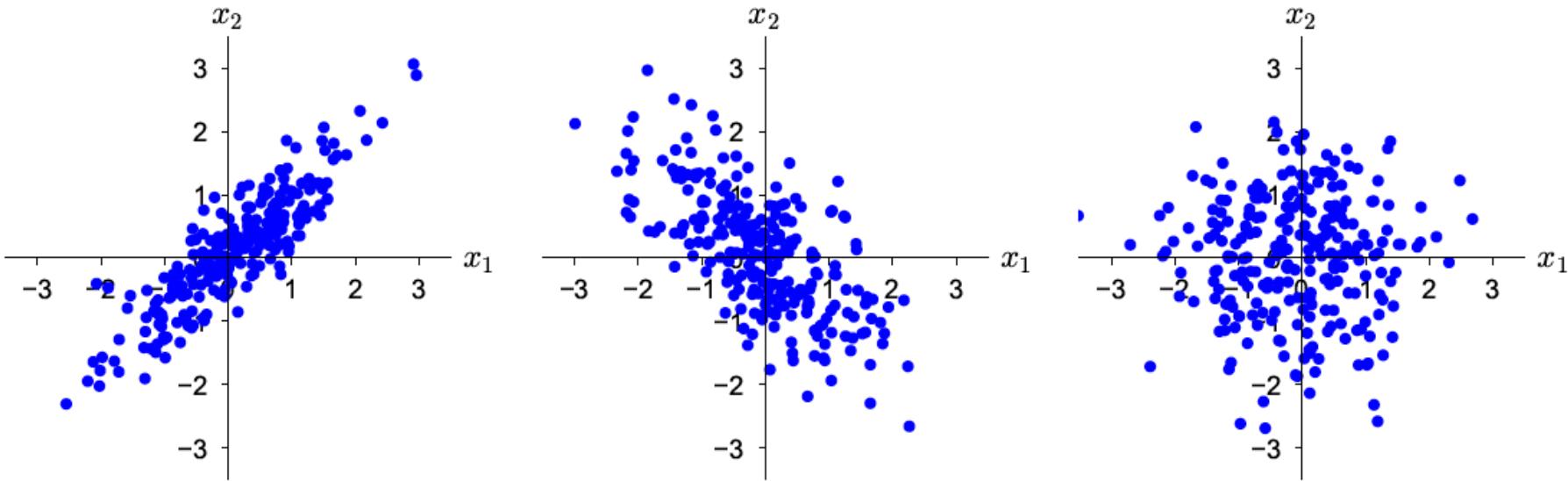


- $p(\mathbf{x}|\boldsymbol{\mu}, \Sigma) = \frac{1}{(\sqrt{2\pi})^D \sqrt{|\Sigma|}} \exp \left(-\frac{1}{2} (\mathbf{x}-\boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x}-\boldsymbol{\mu}) \right)$

$$\boldsymbol{\mu} = \mathbb{E}[\mathbf{x}] \quad \text{(平均ベクトル)}$$

$$\Sigma = \mathbb{E}[\mathbf{x}\mathbf{x}^T] - \mathbb{E}[\mathbf{x}]\mathbb{E}[\mathbf{x}]^T \quad \text{(共分散行列)}$$

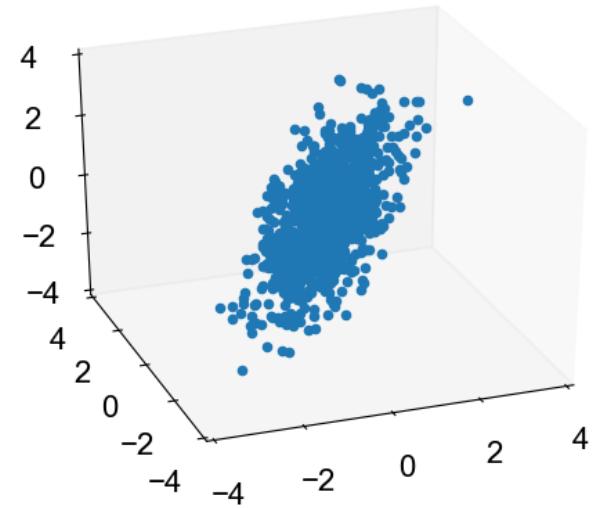
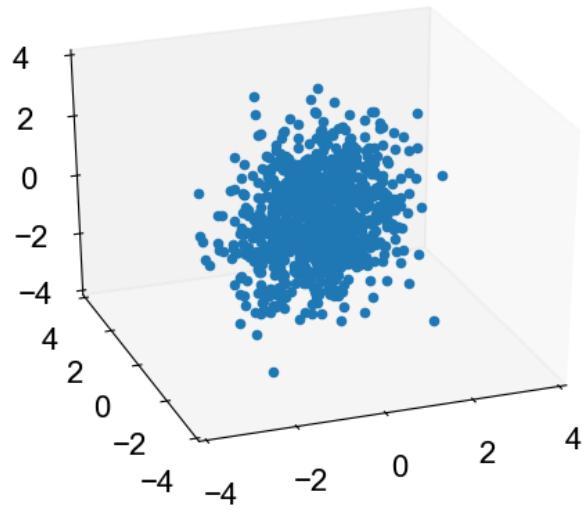
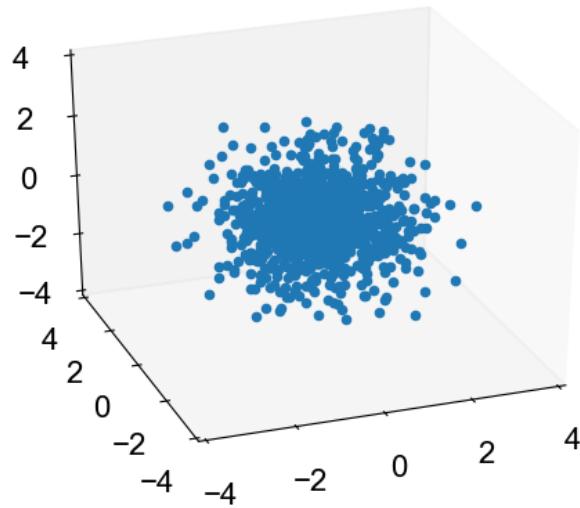
多変量ガウス分布からのサンプル



$$(a) \Sigma = \begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix} \quad (b) \Sigma = \begin{pmatrix} 1 & -0.7 \\ -0.7 & 1 \end{pmatrix} \quad (c) \Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

- 共分散行列の要素の値が大きい(共分散が大)と、類似した値がサンプルされる
 - 負では逆相関、0ならば、無相関

多変量ガウス分布からのサンプル (2)



$$\Sigma = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} 1 & 0.5 & 0.2 \\ 0.5 & 1 & 0.5 \\ 0.2 & 0.5 & 1 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} 1 & 0.8 & 0.6 \\ 0.8 & 1 & 0.8 \\ 0.6 & 0.8 & 1 \end{pmatrix}$$

- 3次元の場合

線形回帰モデルと基底関数

$$y = \mathbf{w}^T \phi(\mathbf{x})$$

$$\mathbf{w} = (w_0, w_1, w_2, \dots, w_H)$$

$$\phi(\mathbf{x}) = (\underbrace{\phi_0(\mathbf{x}), \phi_1(\mathbf{x}), \phi_2(\mathbf{x}), \dots, \phi_H(\mathbf{x})}_{=1})$$

- よって、 $y = w_0 + w_1 \phi_1(\mathbf{x}) + w_2 \phi_2(\mathbf{x}) + \dots + w_H \phi_H(\mathbf{x})$ は
関数 $y = \phi_0(\mathbf{x}) (= 1)$

$$y = \phi_1(\mathbf{x})$$

$$y = \phi_2(\mathbf{x})$$

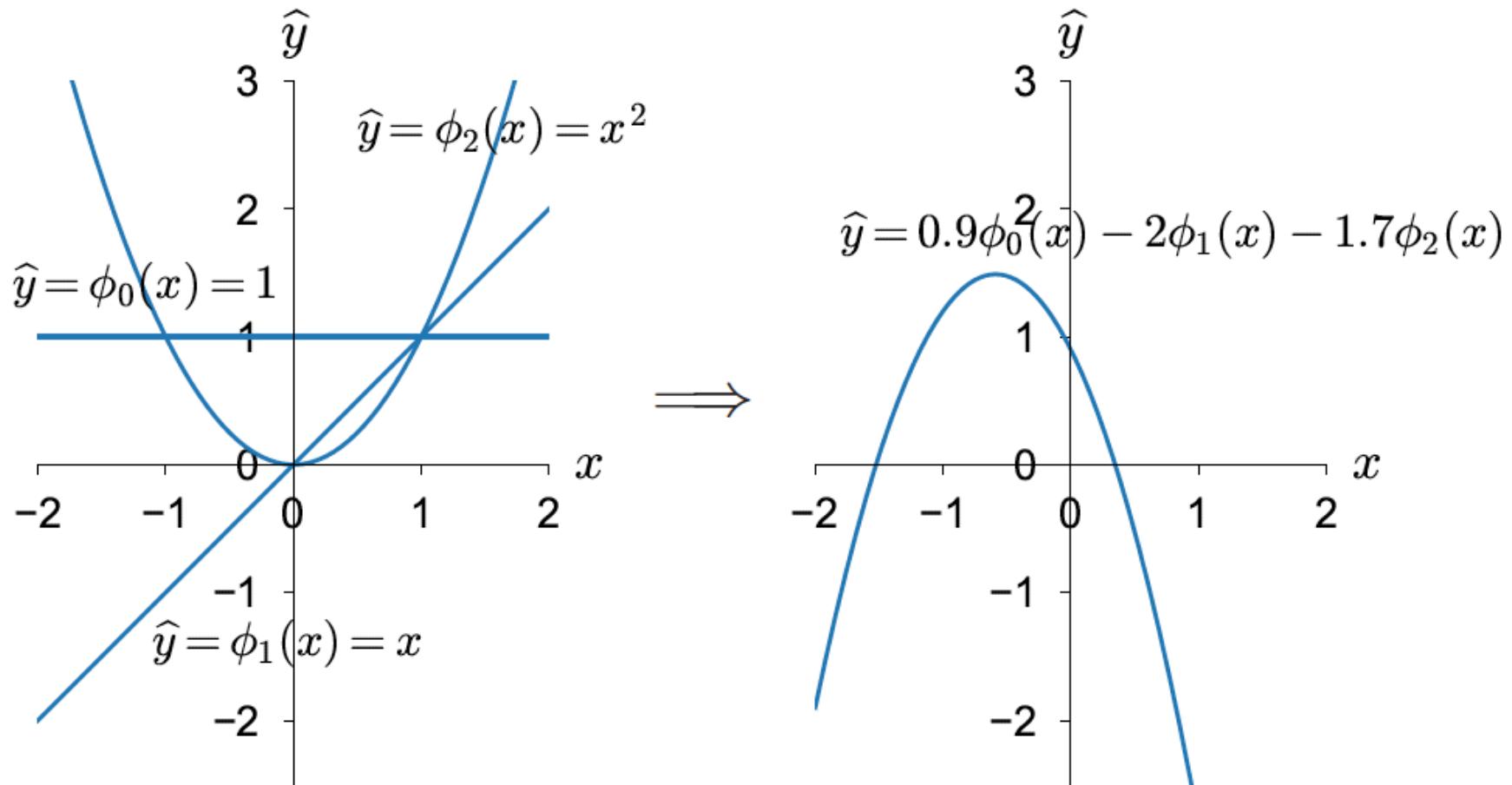
⋮

$$y = \phi_H(\mathbf{x})$$

基底関数
という

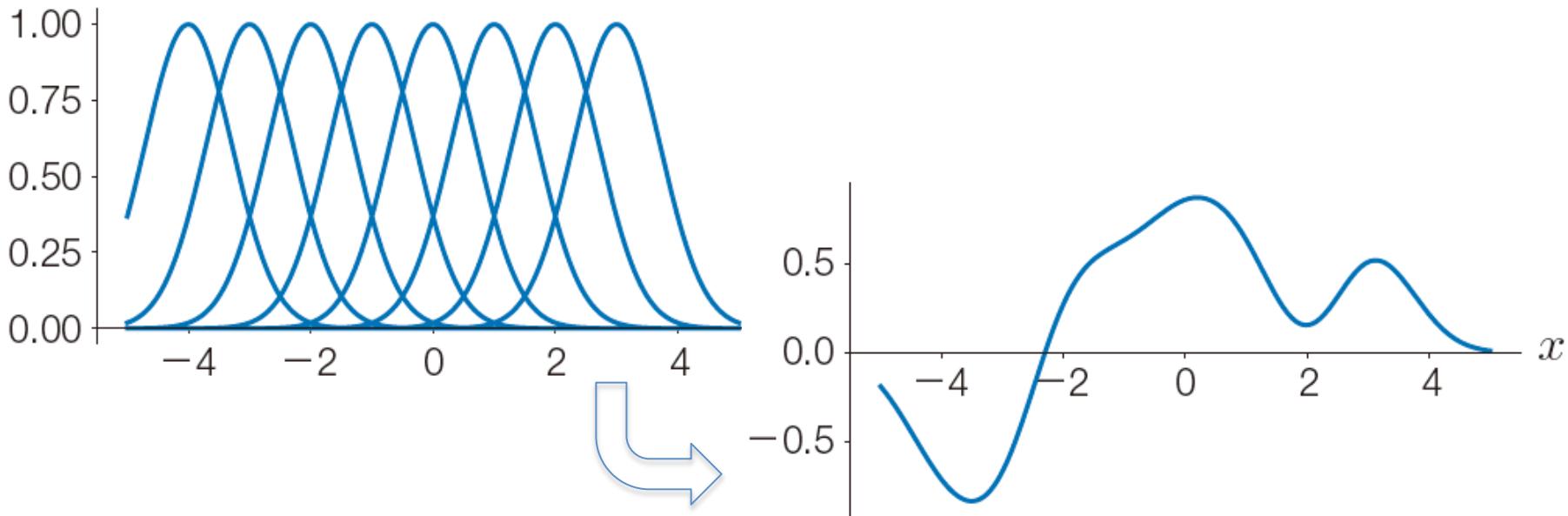
の重みつき和(線形結合)とみなせる

線形回帰モデルと基底関数



- 2次関数 $y = -1.7x^2 - 2x + 0.9$ は、関数 $y = 1$, $y = x$, $y = x^2$ の線形和

動径基底関数回帰



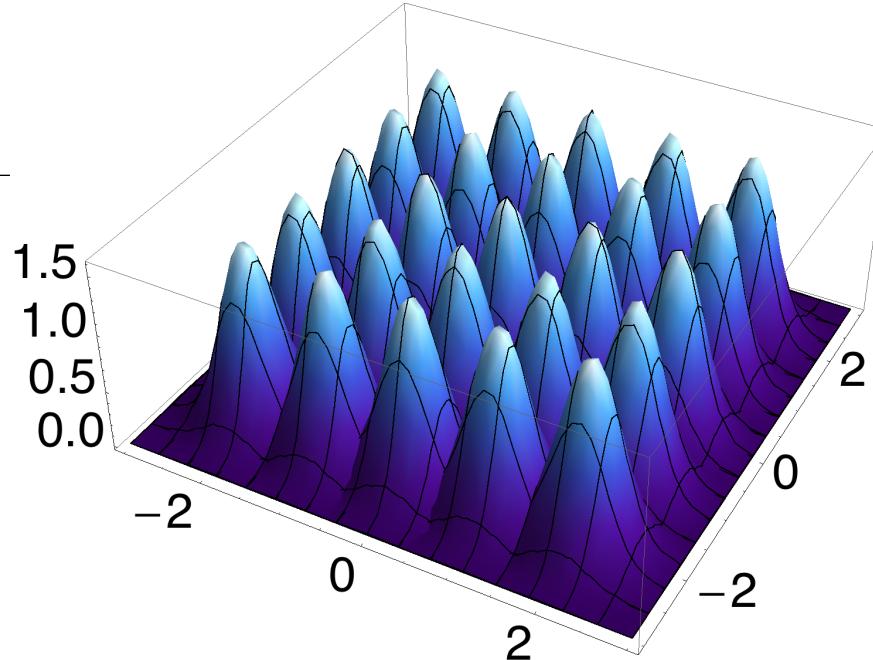
- それなら、 $\phi_h(x) = \exp\left(-\frac{(x-\mu_h)^2}{\sigma^2}\right)$ をたくさん用意すれば、任意の関数が表せるのでは？



動径基底関数回帰 (radial basis function regression)

次元の呪い

- しかし...
 - 動径基底関数回帰に必要な基底関数の数(=パラメータの数)は、 x の次元が増えると指数的に増加
 - 1.0おきに基底関数をとると、 $[-10,10]$ で1次元では21個
 - 2次元では $21^2=441$ 個
 - 10次元では $21^{10}=16,679,880,978,201$ 個！
- 次元の呪い (curse of dimensionality)



どうするか?

- 線形回帰モデル $\mathbf{y} = \Phi \mathbf{w}$ において、リッジ回帰と同様に、重みベクトル \mathbf{w} がガウス分布
$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \alpha \mathbf{I})$$
に従っているとする。

$$\begin{pmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_N \end{pmatrix} = \underbrace{\begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \cdots & \phi_H(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \cdots & \phi_H(\mathbf{x}_2) \\ \vdots & & & \vdots \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \cdots & \phi_H(\mathbf{x}_N) \end{pmatrix}}_{\Phi} \underbrace{\begin{pmatrix} w_0 \\ w_1 \\ \vdots \\ w_H \end{pmatrix}}_{\mathbf{w}}$$

重みwの積分消去

- このとき、 Φ は定数行列なので、 w を定数行列で変換した $y = \Phi w$ もガウス分布に従い、
 - 平均 $\mu = \mathbb{E}[y] = \mathbb{E}[\Phi w] = \Phi \mathbb{E}[w] = 0$
 - 共分散 $\Sigma = \mathbb{E}[yy^T] - \mathbb{E}[y]\mathbb{E}[y]^T$ $= \mathbb{E}[(\Phi w)(\Phi w)^T] = \Phi \mathbb{E}[ww^T] \Phi^T$ $= \alpha \Phi \Phi^T$
- すなわち、 y は全体として、
$$y \sim \mathcal{N}(0, \alpha \Phi \Phi^T)$$
のガウス分布に従う。

ガウス過程

$$p(\mathbf{y}|\mathbf{X}) = \mathcal{N}(\mathbf{0}, \alpha \Phi \Phi^T)$$

は、どんな入力 $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$ についても成り立つ
→ ガウス過程

- どんな入力 $(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)$ についても、対応する出力 $\mathbf{y} = (y_1, y_2, \dots, y_N)$ がガウス分布に従うとき、 \mathbf{y} はガウス過程に従う、という
 - ガウス過程 = 無限次元のガウス分布
 - 線形回帰モデルで、重み w を積分消去したもの
- $\mathbf{K} = \alpha \Phi \Phi^T$ の要素を与えるカーネル関数
$$K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j) = \alpha \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$$
だけでガウス分布が定まる (カーネル法)

ガウス過程 (2)

$$\mathbf{K} = \lambda^2 \Phi \Phi^T = \lambda^2 \underbrace{\begin{pmatrix} \vdots \\ \boxed{\phi(\mathbf{x}_n)^T} \\ \vdots \end{pmatrix}}_{\Phi} \underbrace{\begin{pmatrix} \cdots & \boxed{\phi(\mathbf{x}_{n'})} & \cdots \end{pmatrix}}_{\Phi^T}$$

- x_n と $x_{n'}$ が近ければ、共分散行列 \mathbf{K} の要素 $K_{nn'}$ も大きい
 \downarrow
 $y_n, y_{n'}$ が近い値をとる
- ガウス過程は、 x が似ていれば y も似ている ことを数学的に表すための確率過程。

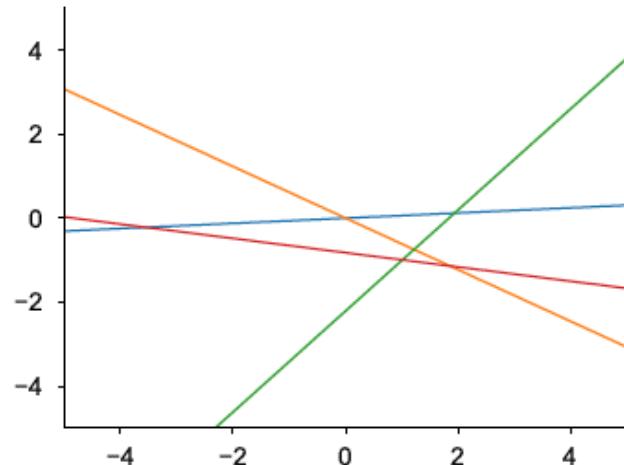
カーネルと特徴ベクトル

- ガウス過程では、 $K_{ij} = \alpha \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$ だけが必要
↓
 $\phi(\mathbf{x}_i), \phi(\mathbf{x}_j)$ を直接求める必要はない
 - 例： $k(\mathbf{x}, \mathbf{x}') = (x_1 x'_1 + x_2 x'_2 + 1)^2$ のとき
 $\phi(\mathbf{x}) = (x_1^2, x_2^2, \sqrt{2}x_1 x_2, \sqrt{2}x_1, \sqrt{2}x_2, 1)$ となるが、この $\phi(\mathbf{x})$ を計算する必要はない
- $\phi(\mathbf{x})$ を求めると、無限次元になることもある
(=無限次元の線形回帰モデルに相当)
- これをカーネルトリックという

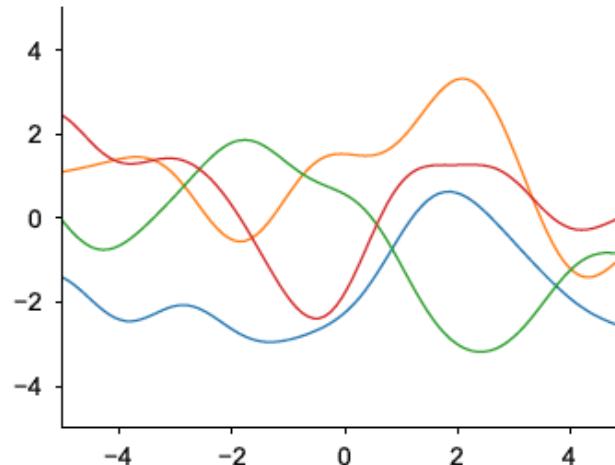
さまざまなカーネル

- 線形カーネル: $k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T \mathbf{x}'$
 - $\phi(\mathbf{x}) = \mathbf{x}$ を意味する → ガウス過程は、重回帰を包含
- 指数カーネル: $k(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{|\mathbf{x} - \mathbf{x}'|}{\theta}\right)$
- 周期カーネル: $k(\mathbf{x}, \mathbf{x}') = \exp\left(\cos \theta_1 \left(\frac{|\mathbf{x} - \mathbf{x}'|}{\theta_2}\right)\right)$
- Matérn カーネル:
$$k(\mathbf{x}, \mathbf{x}') = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}r}{\theta}\right)^\nu K_\nu\left(\frac{\sqrt{2\nu}r}{\theta}\right) \quad (r = |\mathbf{x} - \mathbf{x}'|)$$
 - $\nu = \frac{1}{2}, \frac{3}{2}, \frac{5}{2}$ のときが有名 (Matérn 1/3/5)

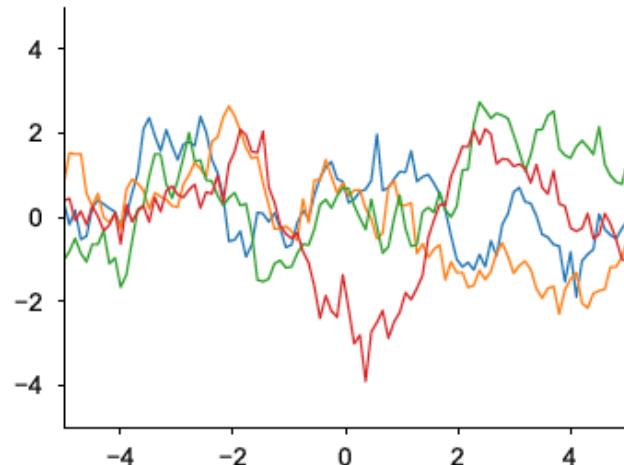
さまざまなカーネル (2)



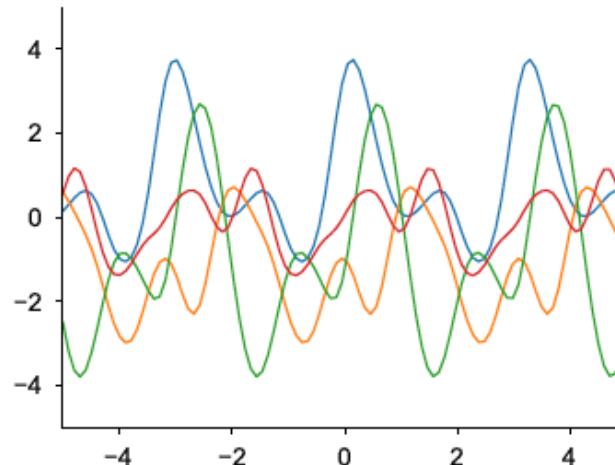
(a) 線形カーネル: $\mathbf{x}^T \mathbf{x}'$



(b) ガウスカーネル: $\exp(-|\mathbf{x} - \mathbf{x}'|^2/\theta)$

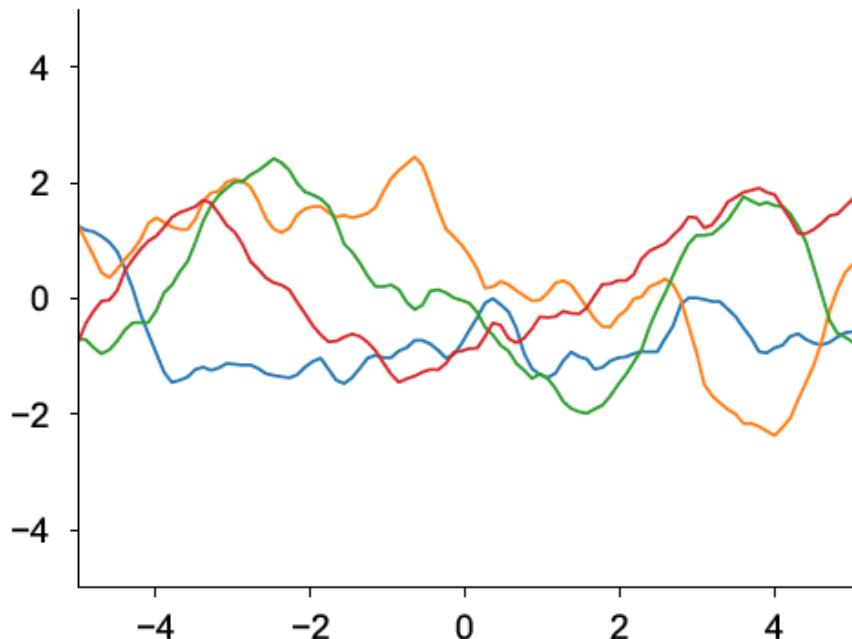


(c) 指数カーネル: $\exp(-|\mathbf{x} - \mathbf{x}'|/\theta)$
(Ornstein-Uhlenbeck 過程)

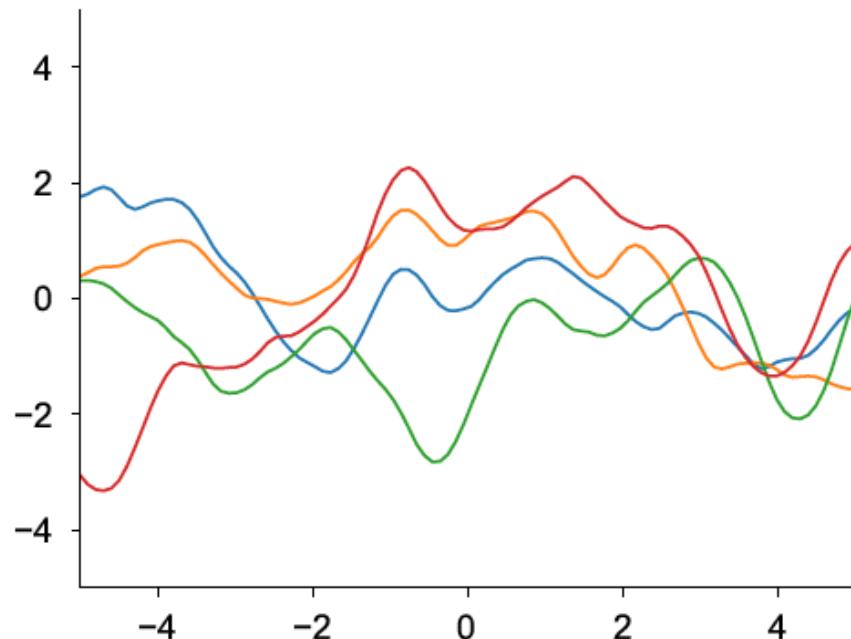


(d) 周期カーネル: $\exp(\theta_1 \cos(|\mathbf{x} - \mathbf{x}'|/\theta_2))$

さまざまなカーネル (3)



(a) $\nu = 3/2$ の場合 (Matérn3)

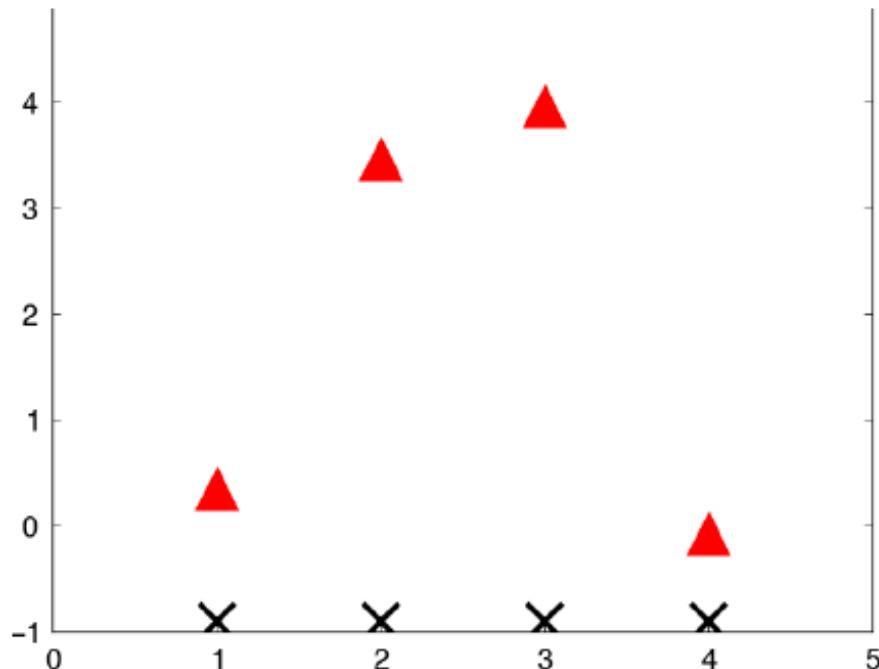


(b) $\nu = 5/2$ の場合 (Matérn5)

- $\lfloor \nu \rfloor$ は、関数を微分できる回数を表す ($C^{\lfloor \nu \rfloor}$ 級)
 - ガウスカーネルは、 $\nu = \infty$
 - 指数カーネルとガウスカーネルの中間の滑らかさを持つ

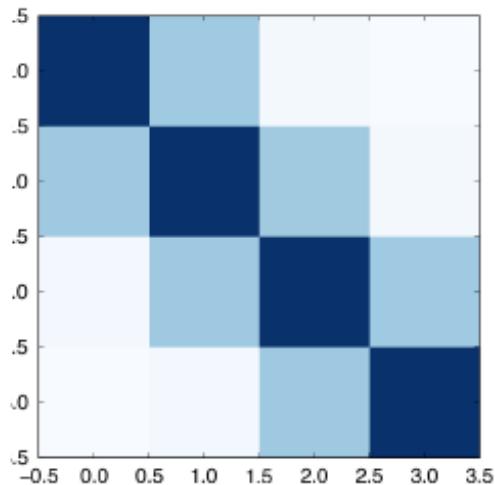
直感的理解

- 相関のある多変量ガウス分布



ガウス分布からのサンプル

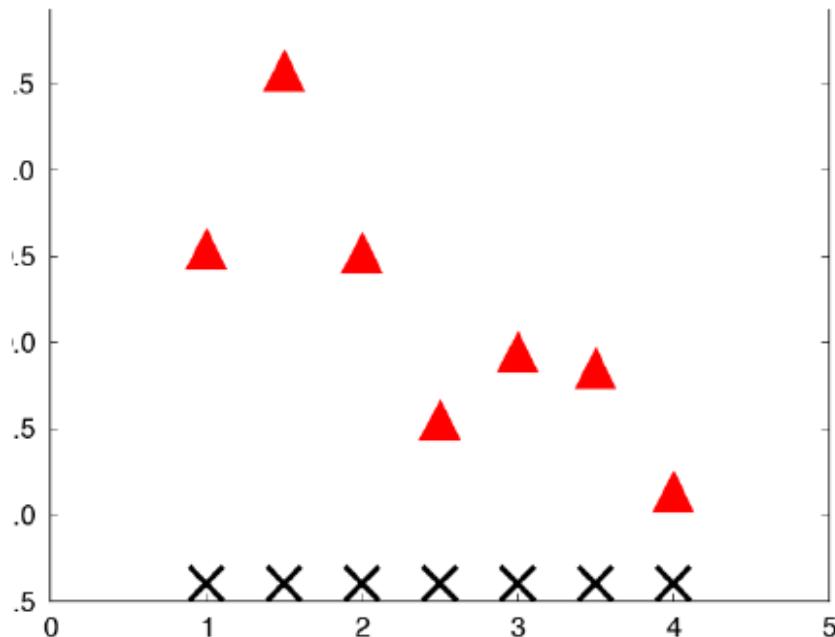
$$K =$$



分散・共分散行列

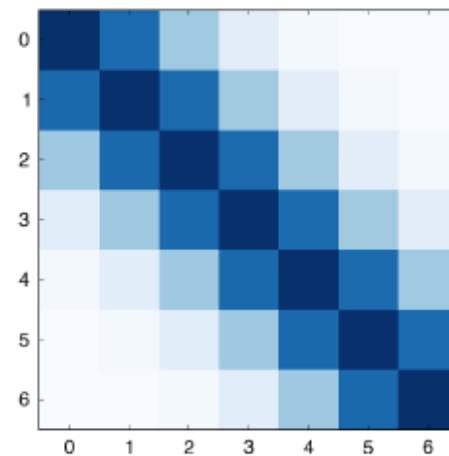
直感的理解

- 相関のある多変量ガウス分布



ガウス分布からのサンプル

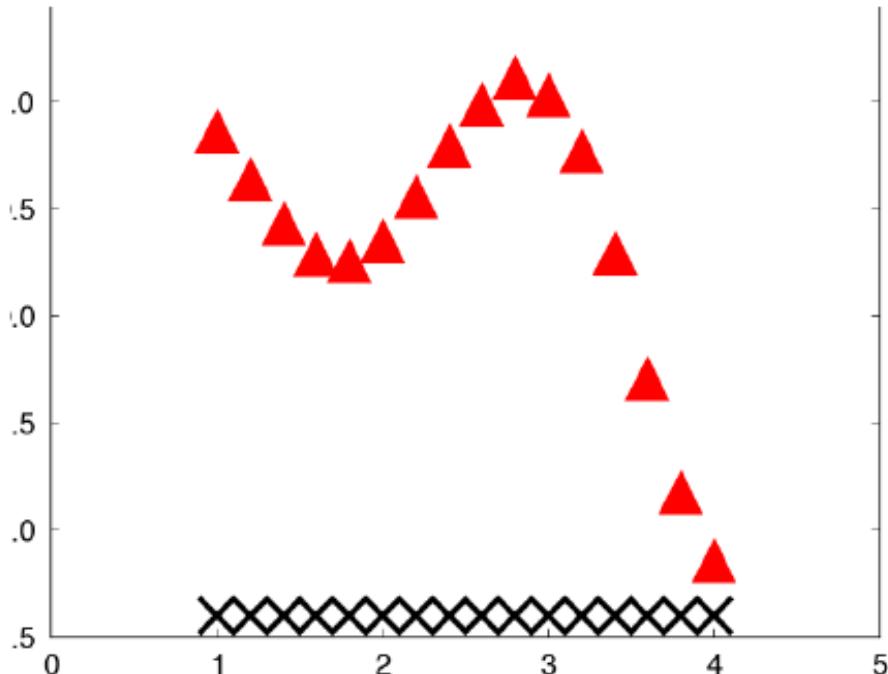
$$K =$$



分散・共分散行列

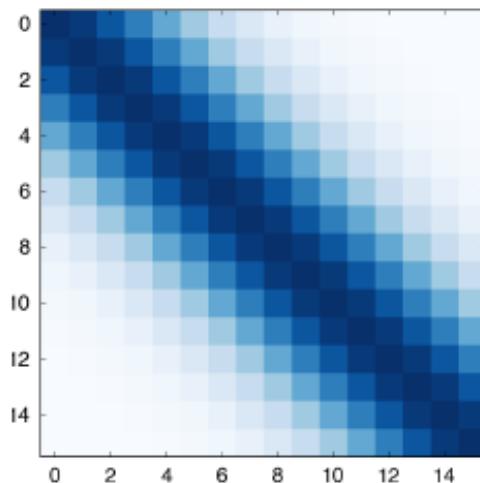
直感的理解

- 相関のある多変量ガウス分布



ガウス分布からのサンプル

$$K =$$



分散・共分散行列

「基底関数」の消去

- 点 h での基底関数

$$\phi_h(x) = \tau \exp\left(-\frac{(x - h/H)^2}{r^2}\right)$$

を考えてみる

- $H \rightarrow \infty$ にしてグリッドを無限に細かくすると、

$$\begin{aligned} k(x, x') &= \lim_{H \rightarrow \infty} \sum_{h=-H^2}^{H^2} \phi_h(x) \phi_h(x') \\ &\rightarrow \int_{-\infty}^{\infty} \tau^2 \exp\left(-\frac{(x - h)^2}{r^2}\right) \exp\left(-\frac{(x' - h)^2}{r^2}\right) dh \\ &= \tau^2 \sqrt{\pi r^2 / 2} \exp\left(-\frac{1}{2r^2}(x - x')^2\right) \\ &\equiv \theta_1 \exp\left(-\frac{1}{\theta_2}(x - x')^2\right) \quad \text{ガウスカーネル!} \end{aligned}$$

観測ノイズ

- 実際の観測値は、誤差が乗っていることが普通

$$\mathbf{f} \sim \mathcal{N}(\mathbf{0}, \mathbf{K})$$

のとき、観測値yは

$$y_n = f(\mathbf{x}_n) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma^2)$$

- すなわち $p(\mathbf{y}|\mathbf{f}) = \mathcal{N}(\mathbf{f}, \sigma^2 \mathbf{I})$ なので、 \mathbf{f} を積分消去すれば

$$\begin{aligned} p(\mathbf{y}|\mathbf{X}) &= \int p(\mathbf{y}, \mathbf{f}|\mathbf{X}) d\mathbf{f} = \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{X}) d\mathbf{f} \\ &= \int \mathcal{N}(\mathbf{y}|\mathbf{f}, \sigma^2 \mathbf{I}) \mathcal{N}(\mathbf{f}|\mathbf{0}, \mathbf{K}) d\mathbf{f} \end{aligned}$$

観測ノイズ (2)

- これは独立なガウス分布の畠み込みなので、結果は共分散行列を単に足し合わせればよく

$$\begin{aligned} p(\mathbf{y}|\mathbf{X}) &= \int \mathcal{N}(\mathbf{y}|\mathbf{f}, \sigma^2 \mathbf{I}) \mathcal{N}(\mathbf{f}|\mathbf{0}, \mathbf{K}) d\mathbf{f} \\ &= \mathcal{N}(\mathbf{0}, \mathbf{K} + \sigma^2 \mathbf{I}) \end{aligned}$$

- つまり、 \mathbf{y} はカーネルを新しく

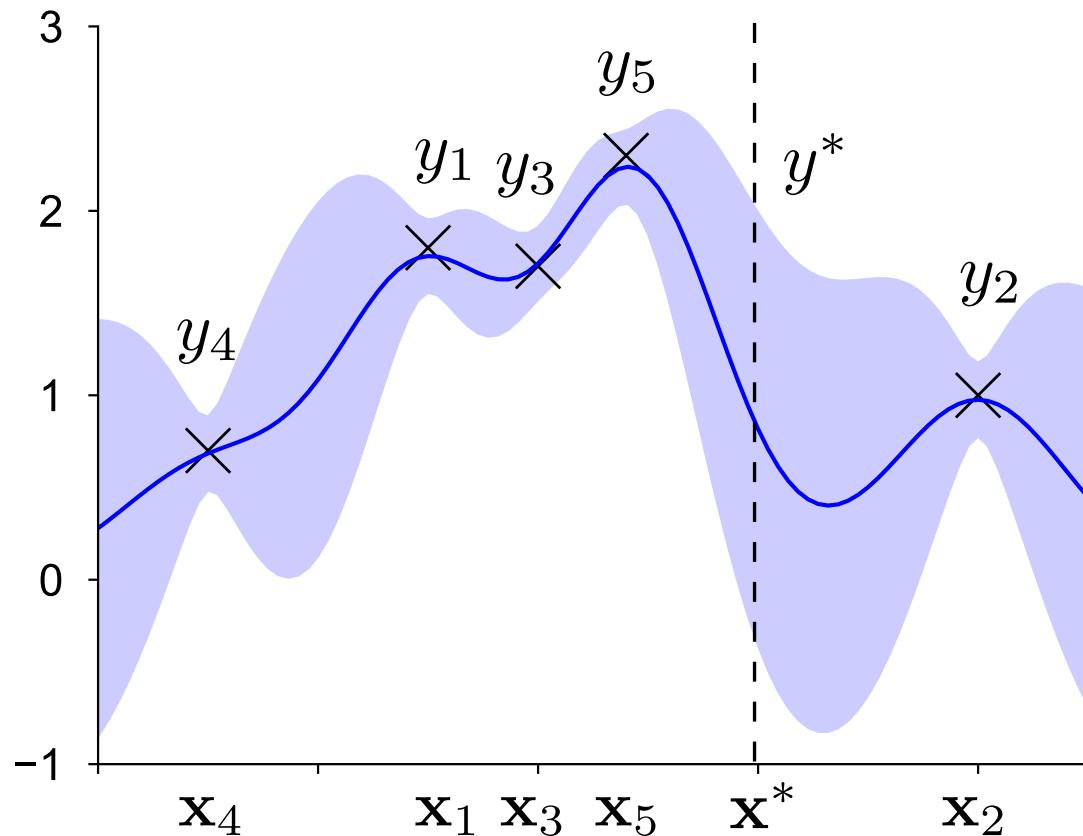
$$k'(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_i, \mathbf{x}_j) + \sigma^2 \delta(i, j)$$

と定義したガウス過程に従う

- カーネル行列の対角要素に σ^2 を足すだけ
- リッジ回帰の「ガウス過程版」

ガウス過程回帰モデル

- 新しい入力点 x^* での出力 y^* の分布はどうなるか？



ガウス過程回帰モデル (2)

- 学習データの y に y^* を加えた $y' = (y, y^*)$ が、学習データの X に x^* を加えた X' から計算される行列を共分散行列としたガウス分布に従うので

$$\begin{pmatrix} y_1 \\ \vdots \\ y_N \\ y^* \end{pmatrix} \sim \mathcal{N} \left(\begin{pmatrix} 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}, \begin{matrix} \mathbf{x}_1 & \cdots & \mathbf{x}_N & \mathbf{x}^* \\ \vdots & & \vdots & \\ \mathbf{K} & & \mathbf{k}_* \\ \mathbf{k}_*^T & & k_{**} \end{matrix} \right)$$

ここで $\mathbf{k}_* = (k(x^*, x_1), k(x^*, x_2), \dots, k(x^*, x_N))$
 $k_{**} = k(x^*, x^*)$

ガウス過程回帰モデル (3)

- 数式で簡潔に書くと

$$\begin{pmatrix} \mathbf{y} \\ y^* \end{pmatrix} \sim \mathcal{N} \left(\mathbf{0}, \begin{pmatrix} \mathbf{K} & \mathbf{k}_* \\ \mathbf{k}_*^T & k_{**} \end{pmatrix} \right)$$

- なので、多変量ガウス分布の条件つき分布の公式から

$$p(y^* | \mathbf{x}^*, \mathbf{X}, \mathbf{y}) = \mathcal{N} \left(\mathbf{k}_*^T \mathbf{K}^{-1} \mathbf{y}, k_{**} - \mathbf{k}_*^T \mathbf{K}^{-1} \mathbf{k}_* \right)$$

- よって、その期待値は

$$\mathbb{E}[y^* | \mathbf{x}^*, \mathbf{X}, \mathbf{y}] = \mathbf{k}_*^T \mathbf{K}^{-1} \mathbf{y}$$

ガウス過程回帰のアルゴリズム

```
1: [mu,var] = gpr (xtest, xtrain, ytrain, kernel)
2: N = length (ytrain)
3: for n = 1 … N do
4:   for n' = 1 … N do
5:     K[n,n'] = kernel (xtrain[n],xtrain[n'])
6:   end for
7: end for
8: yy = K-1 * ytrain
9: for m = 1 … M do
10:  for n = 1 … N do
11:    k[n] = kernel (xtrain[n],xtest[m])
12:  end for
13:  s = kernel (xtest[m],xtest[m])
14:  mu[m] = k * yy
15:  var[m] = s - k * K-1 * kT
16: end for
```

入力:

xtrain = $[x_1, \dots, x_N]$
– 入力 $x \in \mathbb{R}^D$ を N 個並べたベクトル.

ytrain = $[y_1, \dots, y_N]^T$
– 出力 $y \in \mathbb{R}$ を N 個並べたベクトル.

xtest = $[x'_1, \dots, x'_M]$
– 回帰したい入力 x' を M 個並べたベクトル.

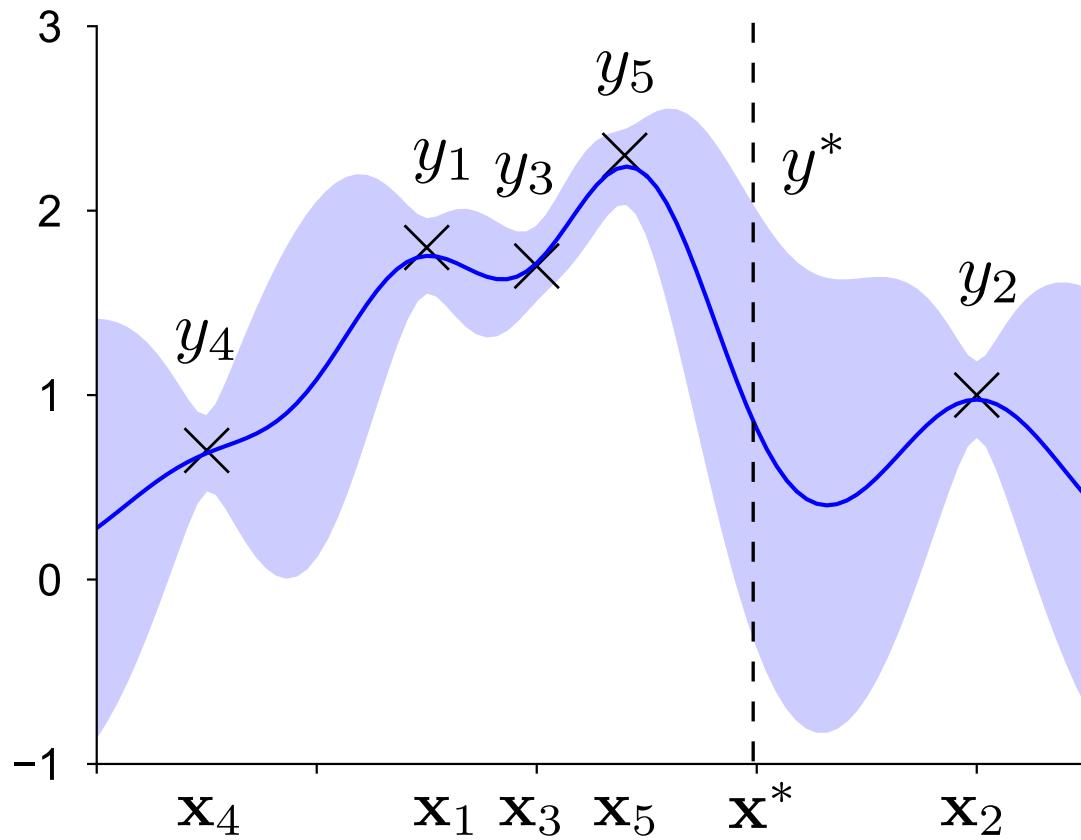
出力:

mu : xtest に対応する y の期待値

var : xtest に対応する y の分散

- (xtrain,ytrain)が与えられたとき、xtestの各点について平均muと分散varを出力

ガウス過程回帰の例



- 新しい入力 x^* での予測値 y^* の分布は、ガウス分布
- 青線は期待値、水色の領域は $\pm 2\sigma$ のエリア

ハイパーパラメータの最適化

$$k(\mathbf{x}_i, \mathbf{x}_j) = \theta_1 \exp\left(-\frac{|\mathbf{x}_i - \mathbf{x}_j|^2}{\theta_2}\right) + \theta_3 \delta(i, j)$$

- カーネルのハイパーパラメータを $\theta = (\theta_1, \theta_2, \theta_3)$ とおくと、 \mathbf{y} の確率はガウス分布なので、 θ に依存して

$$\begin{aligned} p(\mathbf{y} | \mathbf{X}, \boldsymbol{\theta}) &= \mathcal{N}(\mathbf{y} | \mathbf{0}, \mathbf{K}_{\boldsymbol{\theta}}) \\ &= \frac{1}{(2\pi)^{N/2}} \frac{1}{|\mathbf{K}_{\boldsymbol{\theta}}|^{1/2}} \exp\left(-\frac{1}{2} \mathbf{y}^T \mathbf{K}_{\boldsymbol{\theta}}^{-1} \mathbf{y}\right) \end{aligned}$$

- すなわち、

$$\log p(\mathbf{y} | \mathbf{X}, \boldsymbol{\theta}) = \log |\mathbf{K}_{\boldsymbol{\theta}}| - \mathbf{y}^T \mathbf{K}_{\boldsymbol{\theta}}^{-1} \mathbf{y} + \text{const.}$$

- これを最大にする θ を求めればよい。

ハイパーパラメータの最適化 (2)

$$L = \log p(\mathbf{y} | \mathbf{X}, \boldsymbol{\theta}) = \log |\mathbf{K}_{\boldsymbol{\theta}}| - \mathbf{y}^T \mathbf{K}_{\boldsymbol{\theta}}^{-1} \mathbf{y} + \text{const.}$$

- ある $\theta \in \boldsymbol{\theta}$ について、微分の連鎖則から

$$\frac{\partial L}{\partial \theta} = \frac{\partial L}{\partial \mathbf{K}_{\boldsymbol{\theta}}} \frac{\partial \mathbf{K}_{\boldsymbol{\theta}}}{\partial \theta} = \sum_{i=1}^N \sum_{j=1}^N \frac{\partial L}{\partial K_{ij}} \frac{\partial K_{ij}}{\partial \theta}$$

- ここで

$$\frac{\partial}{\partial \theta} \log |\mathbf{K}_{\boldsymbol{\theta}}| = \text{tr} \left(\mathbf{K}_{\boldsymbol{\theta}}^{-1} \frac{\partial \mathbf{K}_{\boldsymbol{\theta}}}{\partial \theta} \right)$$

$$\frac{\partial}{\partial \theta} \mathbf{K}_{\boldsymbol{\theta}}^{-1} = -\mathbf{K}_{\boldsymbol{\theta}}^{-1} \frac{\partial \mathbf{K}_{\boldsymbol{\theta}}}{\partial \theta} \mathbf{K}_{\boldsymbol{\theta}}^{-1}$$

なので、後は $\frac{\partial K_{ij}}{\partial \theta}$ を使っているカーネル毎に計算すればよい。

ハイパーパラメータの最適化 (3)

- $\frac{\partial \mathbf{K}_\theta}{\partial \theta}$ は?

→ 各 K_{ij} を θ で微分して並べた行列.

- 例:

$$K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j) = \theta_1 \exp\left(-\frac{|\mathbf{x}_i - \mathbf{x}_j|^2}{\theta_2}\right) + \theta_3 \delta(i, j)$$

のとき、

$\theta_1 > 0$ なので $\theta_1 = e^\tau \Leftrightarrow \tau = \log \theta_1$ とおけば、

$$\frac{\partial k(\mathbf{x}_i, \mathbf{x}_j)}{\partial \tau} = e^\tau \exp\left(-\frac{|\mathbf{x}_i - \mathbf{x}_j|^2}{e^\tau}\right) = k(\mathbf{x}_i, \mathbf{x}_j) - e^\tau \delta(i, j)$$

- θ_2, θ_3 についても同様

ハイパーパラメータの最適化 (4)

- 最適化アルゴリズム (Python, BFGSの場合)

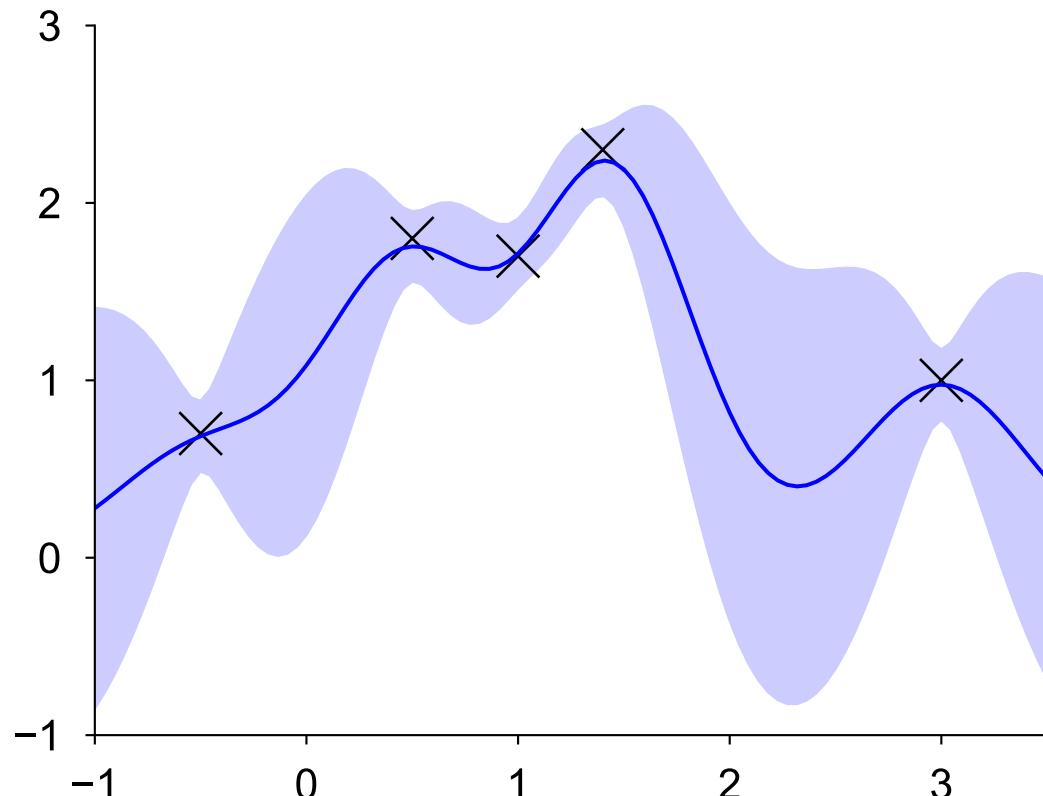
```
from scipy.optimize import minimize

def optimize (xtrain, ytrain, kernel, kgrad, init):
    res = minimize (loglik, init,
                    args = (xtrain,ytrain,kernel,kgrad),
                    jac = gradient, method = 'BFGS',
                    callback = printparam,
                    options = {'gtol' : 1e-4, 'disp' :
print res.message
return res.x
```

- loglik で目的関数(負の対数尤度)を、gradientで偏微分を並べたベクトルを計算

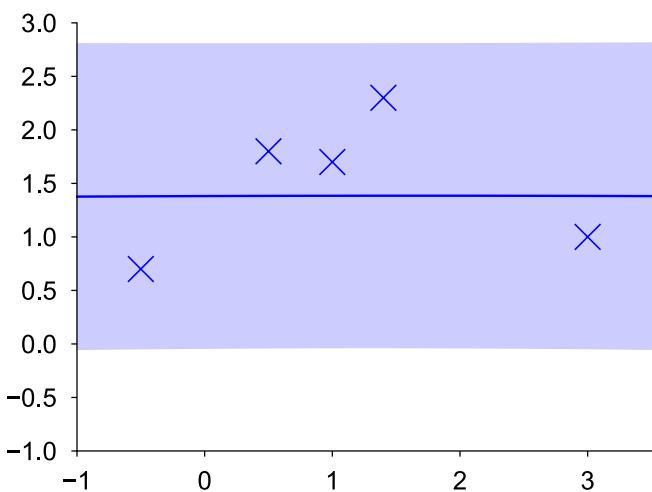
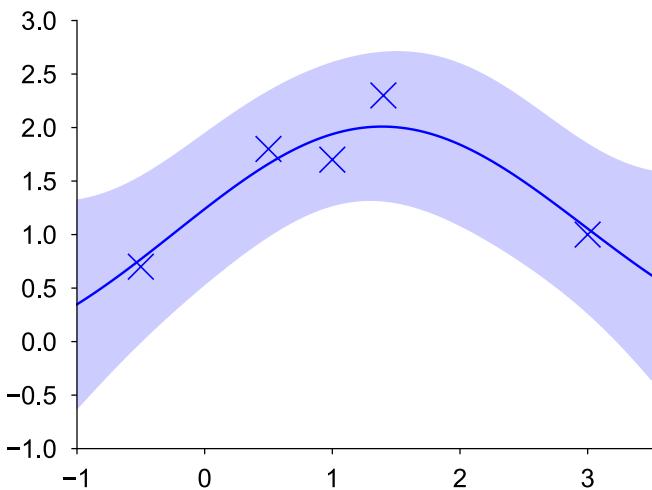
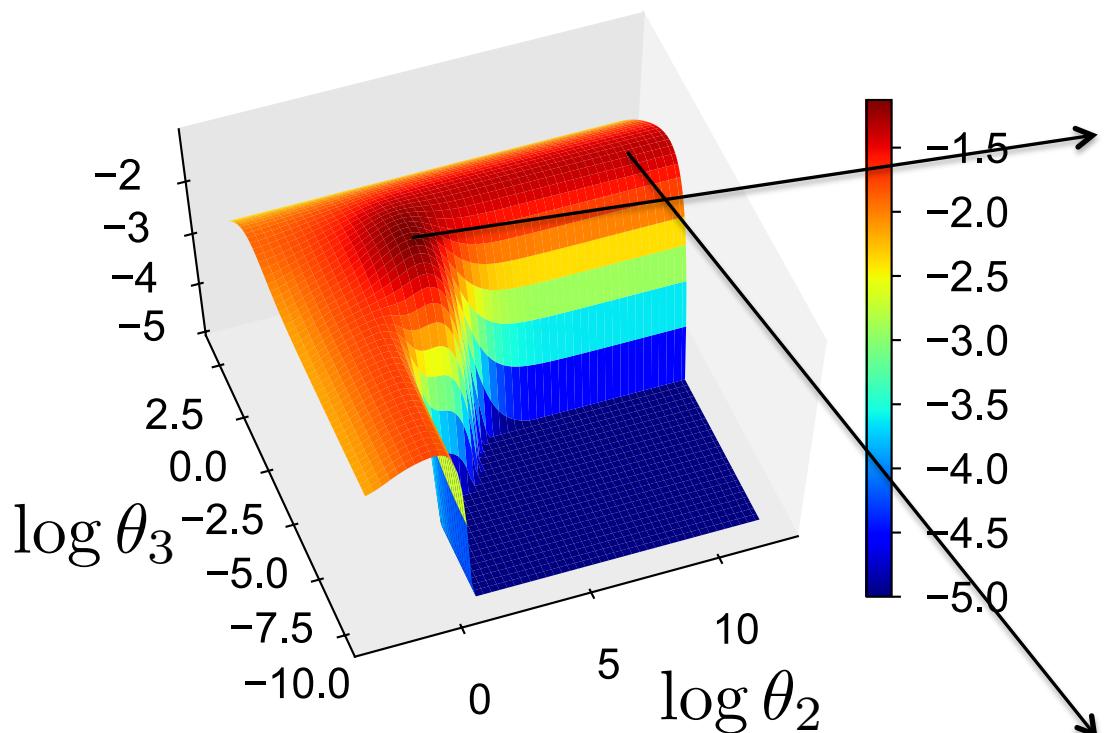
最適化ルーチンは最小化問題を解いているため

ハイパーパラメータの最適化 (5)



- カーネル $k(\mathbf{x}_i, \mathbf{x}_j) = \theta_1 \exp\left(-\frac{|\mathbf{x}_i - \mathbf{x}_j|^2}{\theta_2}\right) + \theta_3 \delta(i, j)$ で、 $\theta_1 = 1$ としてみる
- 上の画像は、観測点が少ないので若干オーバーフィット

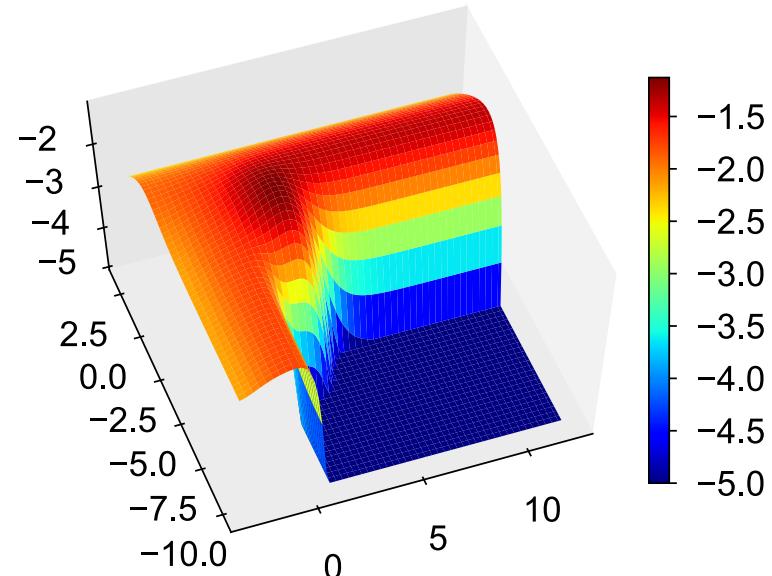
ハイパーパラメータの最適化 (6)



- $\log \theta_2, \log \theta_3$ の関数
- 複数の峰があり、最適化は初期値に依存する

ハイパーパラメータの最適化 (5)

- 最適化は初期値に依存
- 対策：
 - 初期値を様々に変えて、最適化された尤度を観察
 - MCMC法
 - ランダムウォーク Metropolis-Hastings
 - 勾配が計算できるので、Hamiltonian Monte Carloや NUTSで効率的に最適化
- 一般に、ハイパーパラメータの数が増えると高次元の探索問題になり、難しいことに注意する



カーネルの組み合わせ

- カーネル $k_1(\mathbf{x}, \mathbf{x}')$ と $k_2(\mathbf{x}, \mathbf{x}')$ の和や積も、正しいカーネル関数になる

- $\theta_1 k_1(\mathbf{x}, \mathbf{x}') + \theta_2 k_2(\mathbf{x}, \mathbf{x}')$
- $k_1(\mathbf{x}, \mathbf{x}')^p \cdot k_2(\mathbf{x}, \mathbf{x}')^q \quad (p, q \in \mathbb{N})$

などは、また有効なカーネル関数→GPML4章を参照

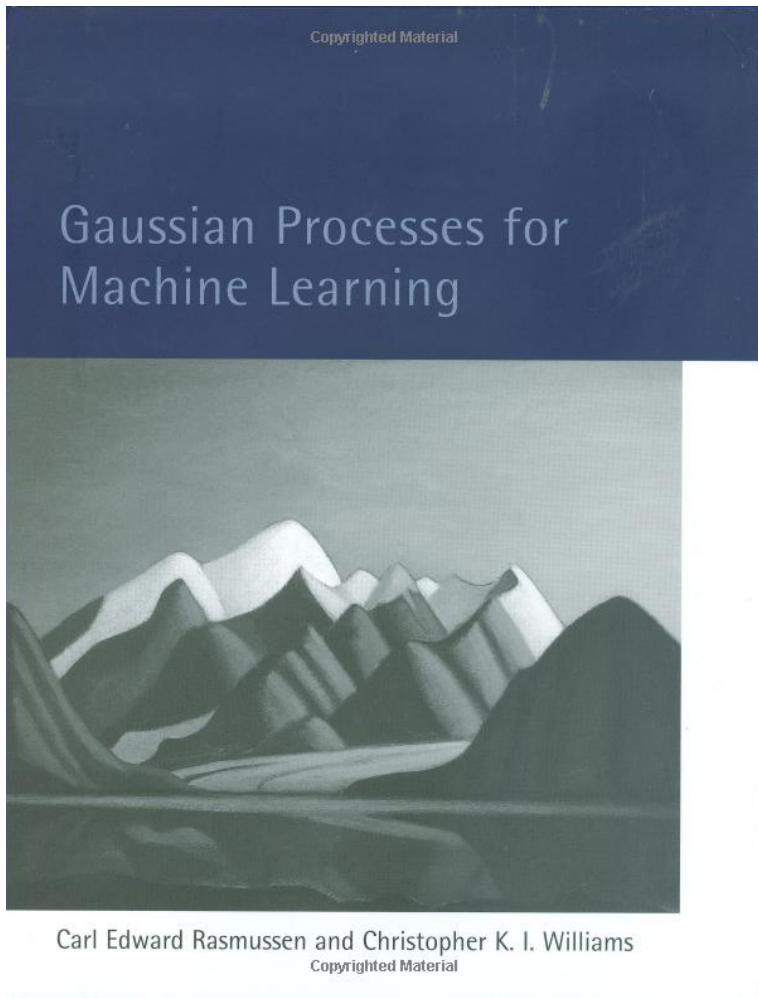
- カーネルとして、たとえば

$$k(\mathbf{x}, \mathbf{x}') = \theta_1 \mathbf{x}^T \mathbf{x}' + \theta_2 \exp\left(\theta_3 \cos\left(\frac{|\mathbf{x} - \mathbf{x}'|}{\theta_4}\right)\right) \quad (\theta_1, \theta_2, \theta_3, \theta_4 \geq 0)$$

を使って $\theta_1, \theta_2, \theta_3, \theta_4$ を最適化すれば、線形性と周期性を自動的に調節した回帰モデルが得られる!

参考文献

- GPML (“Gaussian Processes for Machine Learning”)



- 教科書はフリーでダウンロード可能
<http://www.gaussianprocesses.org/gpml/>
- 付属MATLABライブラリ：
<http://www.gaussianprocesses.org/gpml/code/>