講義概要

a.アウトライン

b.数学的定式化

- i.非線形回帰モデルの説明
- ii.パラメータの推定
- iii.バイアスバリアンスのトレードオフ
- iv.クロスバリデーション

非線形回帰モデル

複雑な非線形構造を内在する現象に対して、非線形回帰モデリングを実施。

・データの構造を線形でとらえられる場合は限られる。

・非線形な構造をとらえられる仕組みが必要。

■基底展開法

・回帰関数として、既定関数と呼ばれる既知の非線形関数とパラメータベクトルの線形結合を使用。

・未知パラメータは、線形回帰モデルと同様に最小二乗法や最尤法により推定。

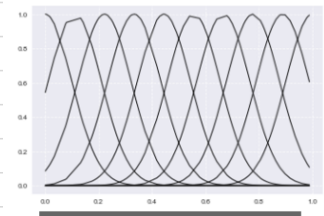
$$y_i = f(\boldsymbol{x}_i) + \varepsilon_i \qquad y_i = w_0 + \sum_{i=1}^{m} w_j \phi_j(\boldsymbol{x}_i) + \varepsilon_i$$

最尤法： **観測されたデータからそれを生んだ母集団を説明しようとする際に広く用いられる。**

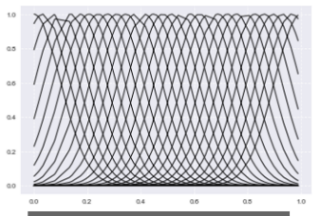
パラメータ θ が不明な確率分布fDに従う母集団から標本が得られたとき、データを良く説明する良いθ は何か

よく使われる基底関数

- ・多項式関数
- ・ガウス型既定関数
- ・スプライン関数/Bスプライン関数



ガウス型基底 (10)

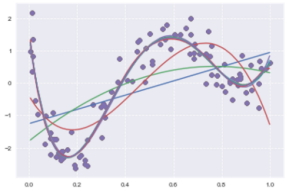


ガウス型基底 (30)

1次元の基底関数に基づく非線形回帰

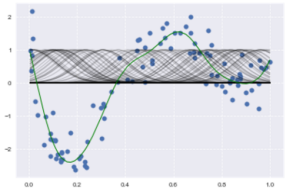
多項式 (1~9次)

$$\phi_j = x^j$$



ガウス型基底

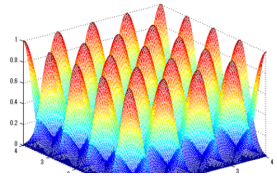
$$\phi_j(x) = \exp\left\{\frac{(x-\mu_j)^2}{2h_j}\right\}$$

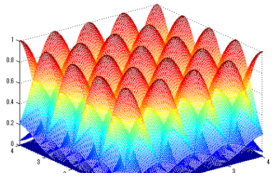


2次元の基底関数に基づく非線形回帰

2次元ガウス型基底関数

$$\phi_j(\boldsymbol{x}) = \exp\left\{\frac{(\boldsymbol{x}-\boldsymbol{\mu}_j)^T(\boldsymbol{x}-\boldsymbol{\mu}_j)}{2h_j}\right\}$$





未学習と過学習

学習データに対して、十分小さな誤差が得られないモデル→未学習

小さな誤差は得られたけど、テスト集合誤差との差が大きいモデル→過学習

不要な基底関数を削除

基底関数の数、位置、バンド幅によりモデルの複雑さが変化。

複数の基底関数を設定すると過学習の原因に。

正則化

罰則化項を設けた損失関数を最小化する方法

$$S_\gamma = (\boldsymbol{y} - \overset{n \times k}{\Phi}\boldsymbol{w})^T(\boldsymbol{y} - \Phi\boldsymbol{w}) + \gamma R(\boldsymbol{w}) \qquad \gamma(>0)$$

基底関数の数(k)が増加するとパラメータが増加し、残差は減少 (モデルが複雑化)

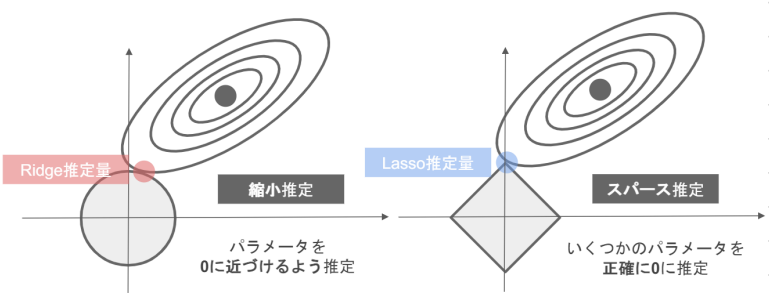
モデルの複雑さに伴う罰則

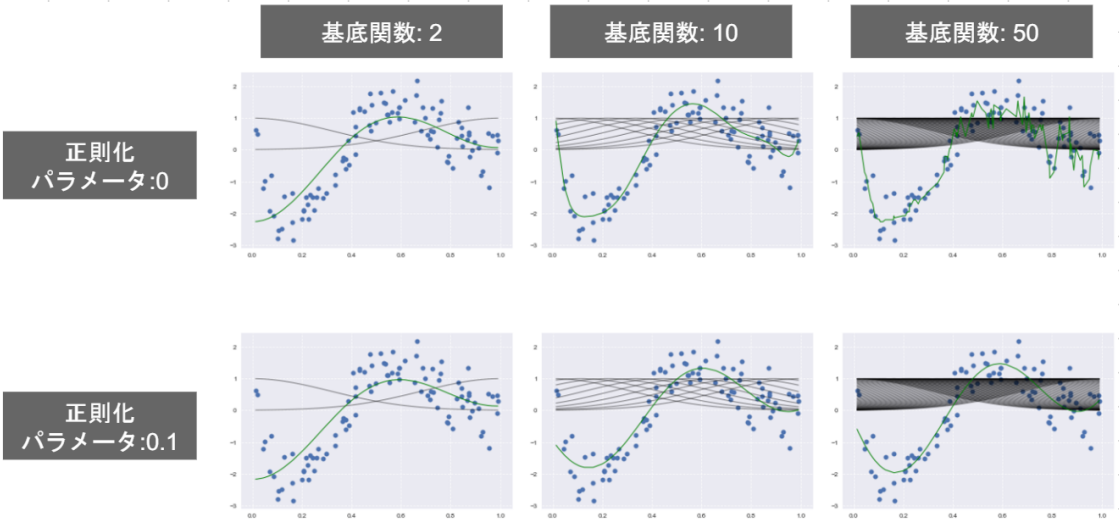
Ridge推定:L2ノルムを利用・・・縮小推定:パラメータを0に近づけるように推定

Lasso推定:L1ノルムを利用・・・スパース推定:いくつかのパラメータを「正確に」0に近づけるように推定

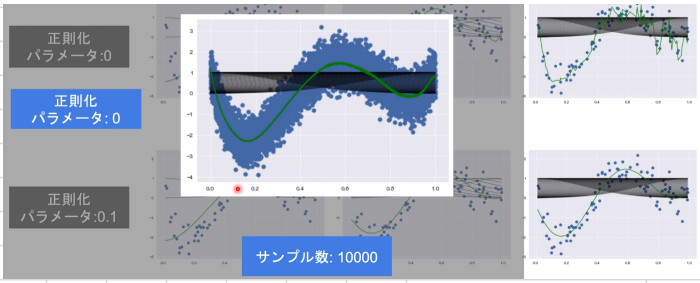
正則化パラメータを小さくすると・・・・制約が大きくなる

正則化パラメータを大きくすると・・・・制約が小さくなる



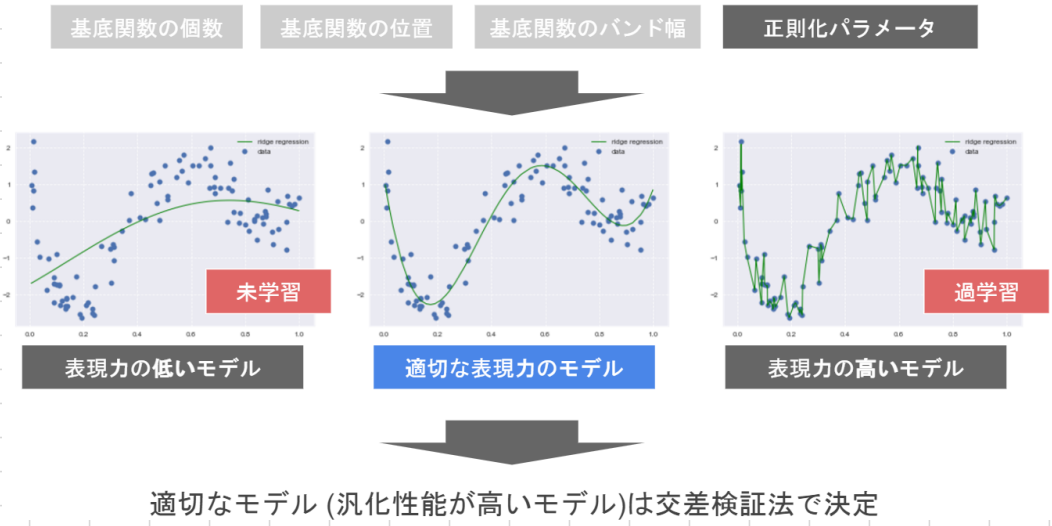


※サンプル数は100



※サンプル数10000

基底関数、正則化パラメータ、サンプル数が「未学習」「過学習」状態に影響を与えている。 →交差検証法により検出する。



汎化性能

バイアス・バリアンス分解

https://www.hellocybernetics.tech/entry/2017/01/24/100415

実際の機械学習では、学習しながらテストデータでテストを行う。

途中で汎化誤差が大きくなっていることが分かった場合、学習の継続は意味がない。

一般的には、正則化項を調整するほうが楽

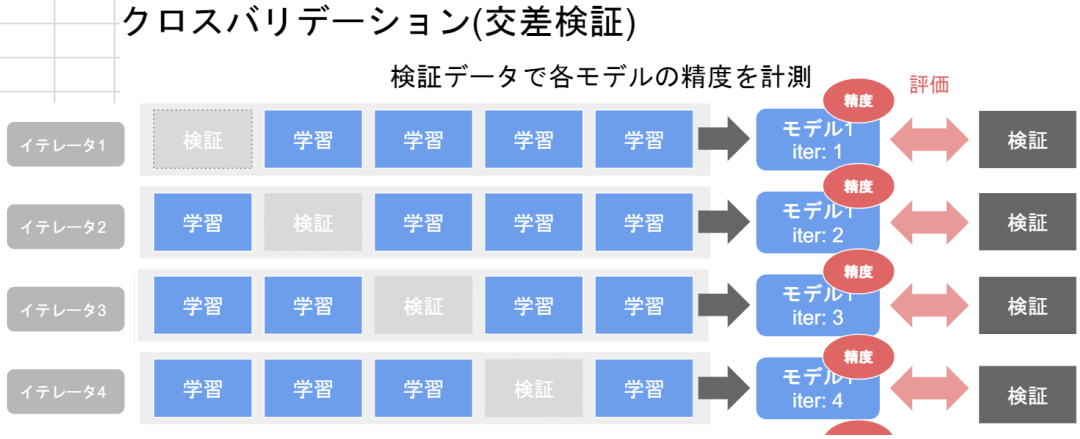
ホールドアウト法

データを学習用とテスト用のふたつに分割し「予測精度」や「誤り率」を推定するために使用。

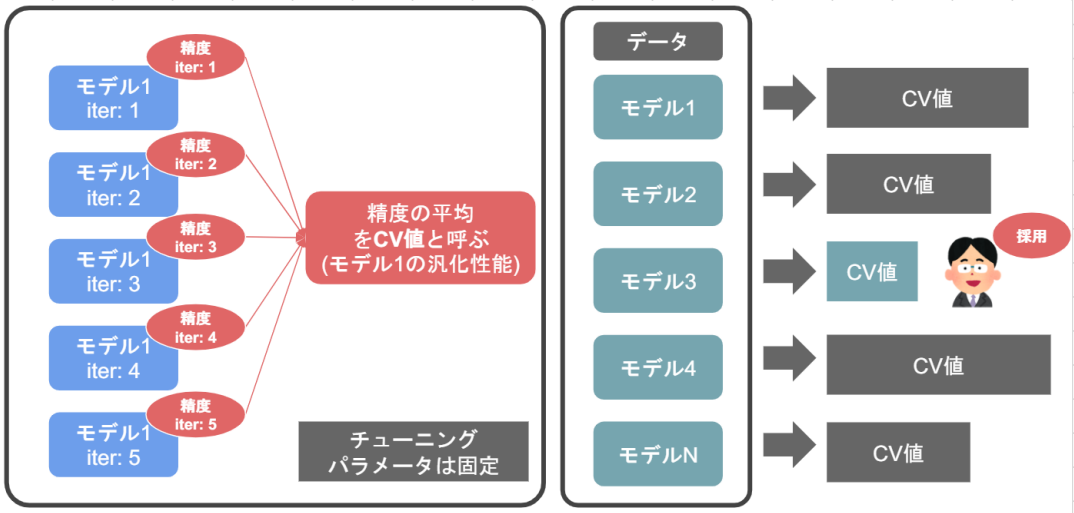
データ量が少ない場合、よい性能評価を出すことができない。

交差検証法（クロスバリデーション法、CV法）

データを複数に分割し、交互に一部を検証データとして利用する。



CV値が小さい最適なモデルを選択することができる。



※あるモデルをホールドアウトで検証したところ70%の精度、CVで65%だった場合でも、
汎化性能の推定としてはCVを利用するようにしてください。

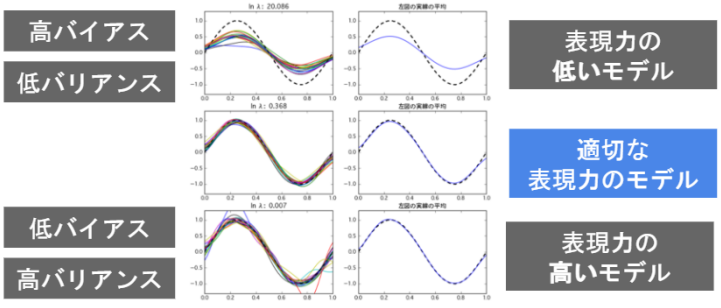
＜現場でありがちなこと＞

ホールドアウト法で70%誤差、CV法で65%誤差となった場合に、ホールドアウト値を採用してしまうNGケース。

グリッドサーチ：すべてのチューニングパラメータの組み合わせで評価値を算出する方法。

※最近は、ベイズ最適化でパラメータチューニングすることも多い。

- 汎化誤差はバイアス+バリアンス+誤差に分解できる
  - (汎化誤差を小さくするモデルを選びたい)
- バイアス-バリアンス分解
  - 学習に用いるサンプルが変化した場合
    - 表現力が**低い**モデル ▶ 低分散・高バイアス
      - 予測モデルは大きく変わらないが、誤差(真の値とのズレ)が大きい
    - 表現力の**高い**モデル ▶ 高分散・低バイアス
      - 予測モデルが大きく変わるが、誤差は小さい



参照：https://openbook4.me/projects/261/sections/1638