1.3 モデル選択

平成 28 年 9 月 10 日

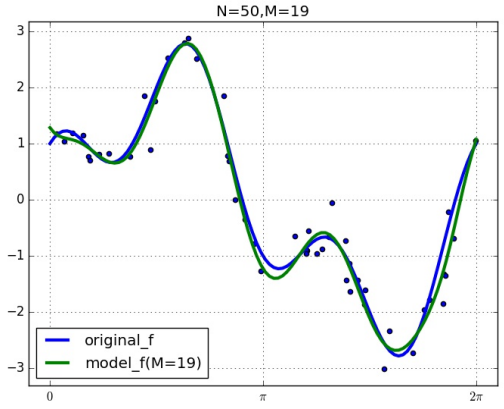
概 要

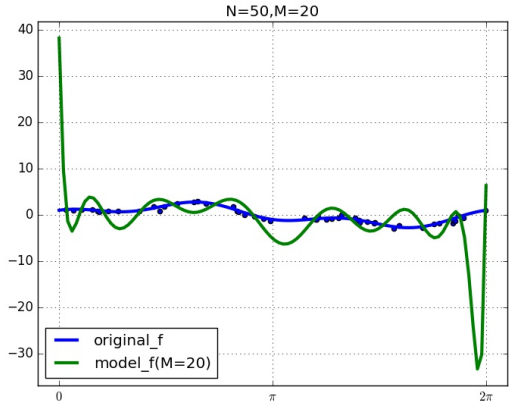
PRML の「1.3 モデル選択」についての実装と考察

目 次

1　問題設定

「多項式曲線フィッテイング」の例を用いて, モデル選択について考察する.





確認用集合で評価した平均二乗平方根誤差は 0.27, 4.91 となることが分かっている.
モデルの違いで学習が成功したり過学習したりするが, ちょうどよく学習できるモデルを見つけ評価したい.

2　確認用集合

2.1　アルゴリズム

全データを訓練用集合と確認用集合とに分ける. いくつかのモデルで訓練用集合を用い訓練し, できたモデルを確認用集合で比較する.

ここで, 注意するのは訓練用集合も確認用集合も十分なデータ数を持つ必要があるということ. 確認用集合のデータが少なければ学習は成功せず, 訓練用集合のデータが少なければモデル選択は失敗するからである.

2.2　結果

まず, 訓練集合で誤差 (平均二乗平方根誤差) を評価すると

M \ N	5	10	50	100	500
2	0.51	0.89	1.00	1.15	1.17
4	0.09	0.71	0.69	0.72	0.81
10	0.00	0.00	0.39	0.41	0.38
20	0.00	0.00	1.99	0.32	0.32
50	0.00	0.00	0.25	0.31	0.32

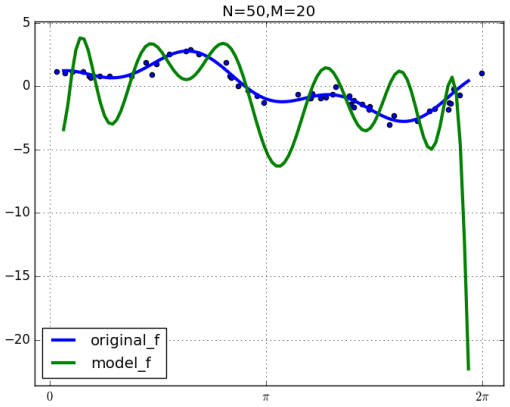
表 1: E_{RMS} と N,M の関係

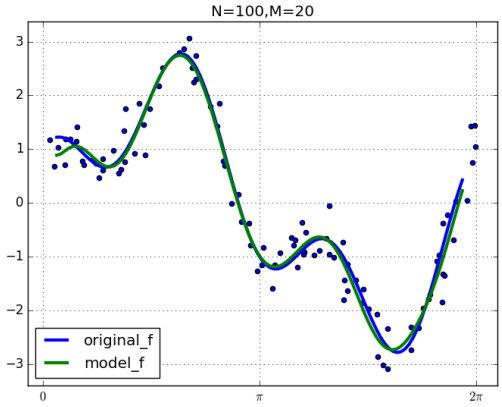
となり, 誤差 $= 0$ となる箇所があることが分かる.
次に, 確認用集合を用いて誤差を評価する. ここでは確認用集合として新たな 50 個のデータを用いて誤差関数を評価する.

M \ N	5	10	50	100	500
2	1.15	1.12	1.15	1.20	1.17
4	1.18	0.84	0.78	0.79	0.77
10	5.37	55.31	0.43	0.47	0.39
20	22.75	15.90	4.91	0.32	0.27
50	443.21	89932	0.84	0.39	0.27

表 2: E_{RMS} の N,M との関係

訓練用集合で誤差を評価したとき 誤差 $= 0$ となった箇所で誤差が大きくなっていることが分かる. ここでは, 学習が不十分であったことが分かる.
ただし, 確認用集合として 50 個のデータを用いたが, $N = 50, M = 20$ と $N = 100, M = 20$ を比べれば確認用集合を用いずすべてのデータを訓練用集合として用いた方がよいことが分かる.





3 交差確認

3.1 アルゴリズム

全データを訓練用集合と確認用集合とに分ける. いくつかのモデルで訓練用集合を用い訓練し, できたモデルを確認用集合で比較する.

交差確認

1. データ集合を S 個に分ける.
2. $S - 1$ 個を訓練に用いて, 残りの 1 個で誤差評価する.
3. 確認用集合を変えながら 2 を S 回繰り返し, 一番誤差の小さかったモデルを採用する.

3.2 コード

表示, プロット部分は省略する. メインはデータの分割と交差確認のアルゴリズム.(cross_validation.py)

```
#交差確認
S=50
L=ceil(N/S)

for i in range(S*L-N):
   x=np.append(x,None)
   t=np.append(t,None)

x_set=x.reshape(S,L)
t_set=t.reshape(S,L)

"""Wの最適化"""
M=20

A=np.zeros((M,M))
W=np.zeros(M)
T=np.zeros(M)

for s in range(S):
   for a in range(S):
      if a!=s:
         #二乗和誤差の最小化
         for i in range(M):
            for j in range(M):
               for l in range(L):
                  if x_set[a][l]!=None:
                     A[i][j]+=np.power(x_set[a][l],i+j)
                  for l in range(L):
                     if t_set[a][l]!=None:
                       T[i]+=np.power(x_set[a][l],i)*t_set[a][l]

   #線形方程式(AW=T)を解くことでパラメータWを求める
   W=np.linalg.solve(A,T)
   #求まったパラメータからモデル関数を作り
   def model_f(x):
      sum=0
      for m in range(M):
         sum+=W[m]*x**m
         return sum
```

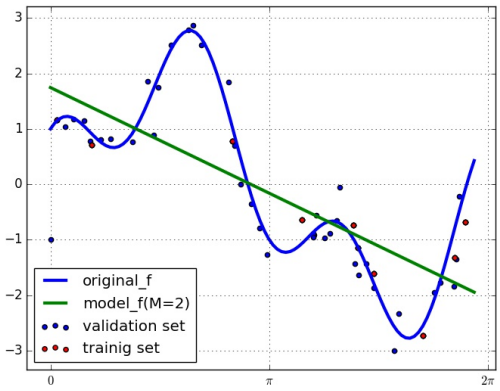
3.3 結果

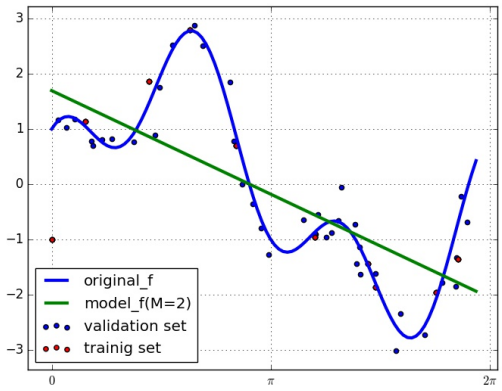
この実験では 50 番目のデータを改ざんし明らかな例外点を含んでいる.
データ数 $N = 50$ に対して $S = 5$ としたところ

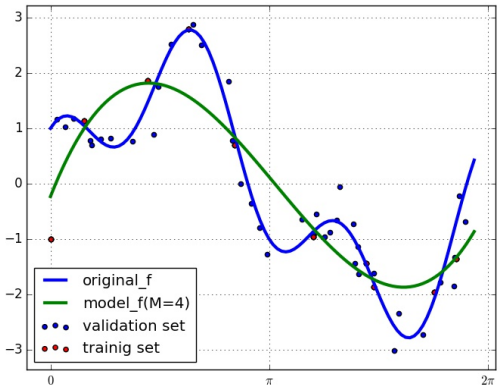
s \ M+1	2	4	10	20	50
0	0.40	0.37	0.23	0.15	0.15
1	0.30	0.26	0.13	0.17	0.15
2	0.47	0.34	0.17	0.15	0.14
3	0.44	0.33	0.23	0.20	0.40
4	0.55	0.21	0.10	0.10	0.09

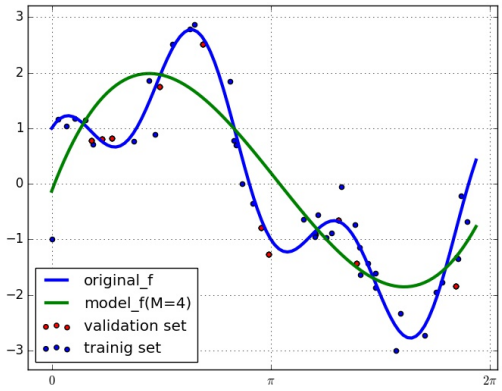
表 3: E_{RMS} の s, M との関係

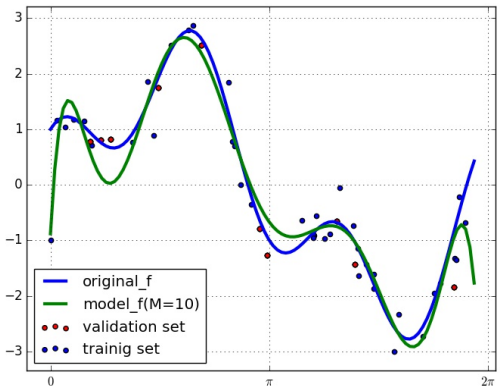
$M = 1$ の場合を除いて, 確認用集合を $s = 4$(例外点を含むもの) となった. 例外点は, たとえ例外点を含む確認用集合で誤差を評価したとしても, 訓練用集合に含まない方が良いと思われる. 各 M に対し誤差の最小, 最大となったもののプロットを示す.

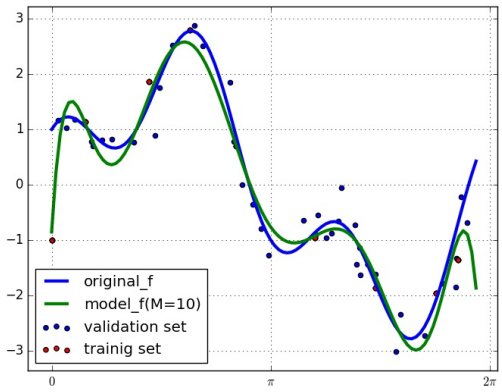


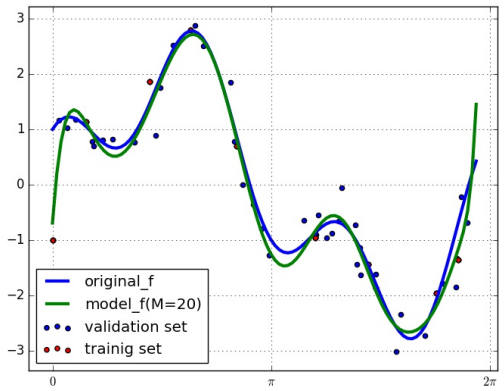


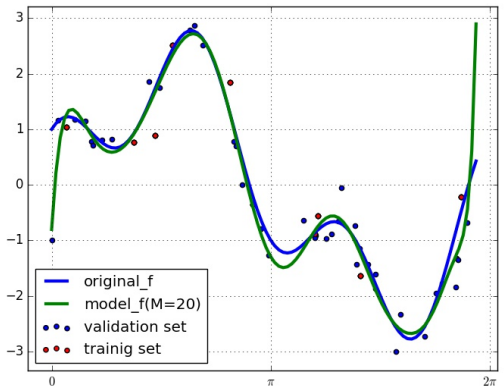


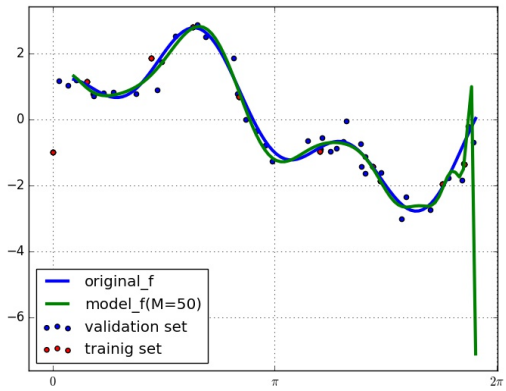


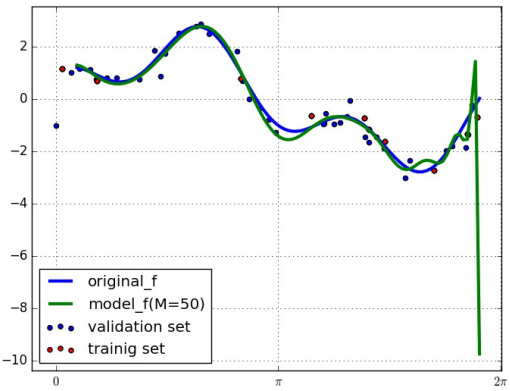








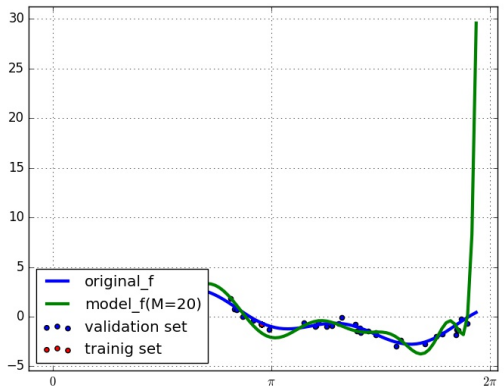


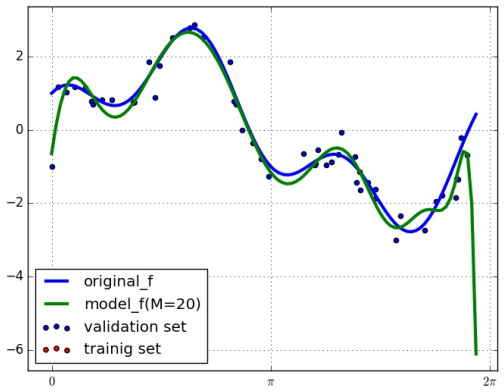


メリットとしては $M = 20$ のときに過学習を起こしていないことがある.

3.4 LOO 法

データが少ないときに行われる方法で $S = N$ とする方法である. $M = 20$ のとき誤差が最小, 最大となったもののプロットを示す.





誤差を評価する点が 1 つという時点で信憑性がない. 効果は薄いと感じる.

4 情報基準量

より複雑なモデルによる過学習を避ける罰金項を足すことによって最優推定のバイアスを修正しようというもの.

4.1 赤池情報基準量 (AIC)

4.1.1 アルゴリズム

AIC は

$$\ln p(D|\mathbf{w}_{ML}) - M \quad (1.73)$$

という量が最大になるモデルを選ぶというものである. ここで N は一定と考え

$$\begin{aligned} \ln p(D|\mathbf{w}_{ML}) - M &= -\frac{\beta}{2}\sum_{n=1}^{N}\{t_n - y(x_n, \mathbf{w})\}^2 + \frac{N}{2}\ln\beta - \frac{N}{2}\ln 2\pi - M \\ &= -\frac{\beta}{2}\sum_{n=1}^{N}\{t_n - y(x_n, \mathbf{w})\}^2 - M + const \end{aligned}$$

となる. ただし, $\beta = 1/(0.3)^2$ とする. (ベイズ曲線フィッテイング参照)

4.1.2 結果

上の定数項を除いた部分を AIC とし, これを評価する. $N = 100$ とすると

M	2	4	10	20	50
AIC	-732	-295	-102	-76	-102

表 4: AIC と M との関係

ここでは $M = 20$ の場合を選ぶ. $M = 50$ とならず過学習が抑制されていることが分かる.

4.2 ベイズ情報基準量 (BIC)

4.2.1 アルゴリズム

BIC は

$$\ln p(D|\mathbf{w}_{ML}) - \frac{1}{2}M\ln N \quad (4.139)$$

という量が最大になるモデルを選ぶというものである. ここで N は一定と考え

$$\ln p(D|\mathbf{w}_{ML}) - M\frac{1}{2}M\ln N = -\frac{\beta}{2}\sum_{n=1}^{N}\{t_n - y(x_n, \mathbf{w})\}^2 - \frac{1}{2}M\ln N + const$$

となる. ただし, $\beta = 1/(0.3)^2$ とする. (ベイズ曲線フィッテイング参照)

4.2.2 結果

上の定数項を除いた部分を BIC とし, これを評価する. $N = 100$ とすると

M	2	4	10	20	50
BIC	-734	-300	-115	-102	-168

表 5: BIC と M との関係

ここでは $M = 20$ の場合を選ぶ. AIC の場合同様過学習を抑制する結果となった.

5 まとめ

1. 交差確認はうまく動いたが, データ点数が少ないときの LOO 法は必要性をあまり感じない. 訓練集合にまわすべき.

2. AIC, BIC はこの問題ではよい結果を出したが, 論理的背景が薄いと感じる. 罰金項については定数倍の変動について説明できず, そこがけっこう肝心なところだと思う.