データ解析

Rによる多変量解析入門

(7) 最尤法，モデル選択

lec20021121 下平英寿 shimo@is.titech.ac.jp

最尤法

尤度 (likelihood)

- 重回帰モデル

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \epsilon_i; \quad i = 1, \ldots, n$$

$$\epsilon_1, \ldots, \epsilon_n \sim N(0, \sigma^2) \quad \textbf{i.i.d.}$$

- 確率密度関数

$$f(\boldsymbol{y}; \boldsymbol{x}, \boldsymbol{\beta}, \sigma^2)$$

$$= \frac{1}{\sqrt{(2\pi\sigma^2)^n}} \exp\left[-\sum_{i=1}^{n} \frac{\left(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} - y_i\right)^2}{2\sigma^2} \right]$$

- 尤度

$$L(\boldsymbol{\beta}, \sigma^2; \boldsymbol{x}, \boldsymbol{y}) = f(\boldsymbol{y}; \boldsymbol{x}, \boldsymbol{\beta}, \sigma^2)$$

最尤法 (maximum likelihood method)

- 一般にパラメタベクトルを θ，データを $\mathcal{X}$ と書く

$$L(\boldsymbol{\theta}; \mathcal{X}) = f(\mathcal{X}; \boldsymbol{\theta})$$

- 対数尤度 (log-likelihood)

$$\ell(\boldsymbol{\theta}; \mathcal{X}) = \log L(\boldsymbol{\theta}; \mathcal{X}) = \log f(\mathcal{X}; \boldsymbol{\theta})$$

- パラメタの最尤推定量 (MLE) を $\hat{\theta}$ と書く

$$\max_{\boldsymbol{\theta} \in \Theta} \ell(\boldsymbol{\theta}; \mathcal{X}) = \ell(\hat{\boldsymbol{\theta}}; \mathcal{X})$$

$$\left. \frac{\partial \ell(\boldsymbol{\theta}; \mathcal{X})}{\partial \boldsymbol{\theta}} \right|_{\hat{\boldsymbol{\theta}}} = 0$$

重回帰モデルの最尤推定

$$\ell(\boldsymbol{\beta}, \sigma^2; \boldsymbol{x}, \boldsymbol{y}) = -\frac{n}{2} \log(2\pi\sigma^2) - \sum_{i=1}^{n} \frac{\left(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} - y_i\right)^2}{2\sigma^2}$$

$$\frac{\partial \ell}{\partial \beta_k} = \sum_{i=1}^{n} \frac{\left(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} - y_i\right) x_{ik}}{\sigma^2}$$

$$\frac{\partial \ell}{\partial (\sigma^2)} = -\frac{n}{2\sigma^2} + \sum_{i=1}^{n} \frac{\left(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} - y_i\right)^2}{2\sigma^4}$$

最尤推定量 $\hat{\boldsymbol{\beta}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{y}, \quad \hat{\sigma}^2 = \dfrac{\|\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}\|^2}{n}$

c.f. σ^2 **の不偏推定量** $\hat{\sigma}^2 = \dfrac{\|\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}\|^2}{n-p-1}, \quad E(\hat{\sigma}^2) = \sigma^2$

モデル選択

モデル選択

- 確率モデルの候補が複数ある場合

$$f_1(\mathcal{X}; \boldsymbol{\theta}_1), \ldots, f_m(\mathcal{X}; \boldsymbol{\theta}_m)$$

データ $\mathcal{X}$ を最もよく説明するモデルを選ぶ

[例] 多項式回帰の次数選択，重回帰分析の説明変数の選択

- モデルの良さを表す規準を $C(f_i, \mathcal{X})$ と書く

$$C(f_1, \mathcal{X}), \ldots, C(f_m, \mathcal{X})$$

を最大（または最小）にするモデル f_i を選択する

- 様々な規準 $C(f_i, \mathcal{X})$ が提案されている

[例] 修正決定係数，赤池情報量規準

修正決定係数

- 回帰分析の決定係数 R^2
  「観測データへのモデルの当てはまりの良さ」を測る
- 決定係数の問題点
  モデルが大きくなるほど大きな値を取る（多項式回帰では最大次数のモデルが R^2 を最大にする）
- 説明変数の数 p が増えるに従い推定誤差が増える影響を考慮して決定係数を修正する
- 修正決定係数 $\bar{R}^2$ を最大にするモデルを選ぶ

$$\bar{R}^2 = 1 - \frac{n-1}{n-p-1}(1 - R^2)$$

赤池情報量規準 (Akaike Information Criterion)

- モデル選択の「最尤法」
  (回帰分析に限らずさまざまなモデルに適用可能)
- モデル候補 $1, \ldots, m$ の最大対数尤度

$$\hat{\ell}_i = \log f(\mathcal{X}; \hat{\boldsymbol{\theta}}_i), \quad i = 1, \ldots, m$$

- パラメタ数 $p_i = \dim \theta_i$ の大きいモデルが $\hat{\ell}_i$ を大きくする
- p_i の増加に伴い推定誤差が増える影響を考慮して $\hat{\ell}_i$ を修正する

$$\mathrm{AIC}_i = -2 \times (\hat{\ell}_i - p_i)$$

AICを最小にするモデルを選ぶ

- 回帰分析の場合，最大対数尤度は

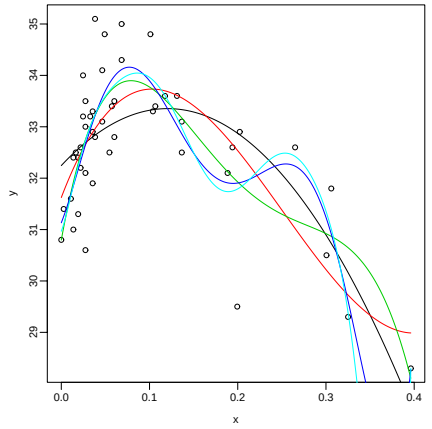
$$\hat{\ell} = -\frac{n}{2}\{1 + \log(2\pi\hat{\sigma}^2)\}$$

多項式回帰の次数選択

```
> ## 次数1,..,10で回帰分析
> ax <- "B02304"; ay <- "B02102"
> X2000$jitem[c(ax,ay)]
                                    B02304
                        "雪　　日　　数（年　間）　"
                                    B02102
"最　高　気　温（日最高気温の月平均の最高値）　"
> x <- X2000$x[,ax]/366  # 雪が降る頻度にしておく
> y <- X2000$x[,ay]
> m0 <- 10
> xx0 <- apply(as.matrix(1:m0),1,function(i) x^i) # データ行列
> xx0[1:3,]
             [,1]       [,2]       [,3]        [,4]        [,5]          [,6]
Hokkaido 0.3961749 0.15695452 0.06218144 0.024634722 0.009759658 0.003866531(
Aomori   0.3251366 0.10571382 0.03437143 0.011175411 0.003633535 0.001181395(
Iwate    0.3005464 0.09032817 0.02714781 0.008159178 0.002452212 0.000737003(
                [,7]         [,8]          [,9]        [,10]
```

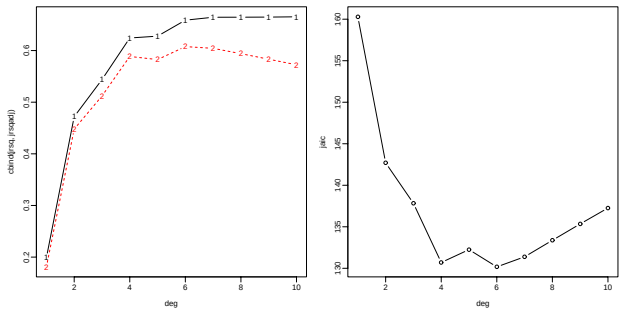


```
Hokkaido 0.0015318224 6.068695e-04 2.404265e-04 9.525092e-05
Aomori   0.0003841149 1.248898e-04 4.060625e-05 1.320258e-05
Iwate    0.0002215038 6.657218e-05 2.000803e-05 6.013343e-06
> f0 <- mylsfit(xx0,y)  # １０次回帰
> f0$tsummary
, ,  = Y


               Estimate      Std.Err    t-value     Pr(>|t|)
Intercept  3.096691e+01 7.902722e-01 39.1851242 4.031380e-31
X1         4.430242e+01 1.840984e+02  0.2406453 8.111953e-01
X2         2.848865e+03 1.462952e+04  0.1947341 8.466960e-01
X3        -1.273483e+05 5.121129e+05 -0.2486723 8.050275e-01
X4         2.457604e+06 9.500205e+06  0.2586896 7.973482e-01
X5        -2.701828e+07 1.027694e+08 -0.2629020 7.941249e-01
X6         1.788248e+08 6.785100e+08  0.2635551 7.936255e-01
X7        -7.220250e+08 2.763475e+09 -0.2612743 7.953700e-01
X8         1.739417e+09 6.763122e+09  0.2571914 7.984954e-01
X9        -2.297849e+09 9.104277e+09 -0.2523922 8.021734e-01
X10        1.281302e+09 5.174020e+09  0.2476414 8.058189e-01

> f0$fsummary
  Mean Sum Sq R Squared  F-value Df 1 Df 2       Pr(>F)
Y   0.9423033 0.6653465 7.157395   10   36 4.427039e-06
> jf <- vector(m0,mode="list") # 部分回帰の結果をしまう場所
> jna <- paste("deg",1:m0,sep="") # 名前をつける
> jna
 [1] "deg1"  "deg2"  "deg3"  "deg4"  "deg5"  "deg6"  "deg7"  "deg8"  "deg9"
[10] "deg10"
> names(jf) <- jna
> for(i in 1:m0) jf[[i]] <- mylsfit(xx0[,1:i],y) # 次数1,...,10
> ## 散布図に曲線を描き加える
> xs <- seq(min(x),max(x),length=300)
> xx1 <- apply(as.matrix(1:m0),1,function(i) xs^i) # データ行列
> xx1 <- cbind(1,xx1) # 定数項の列を加える
> plot(x,y) # 散布図
> a <- c(2,3,4,6,10) # この次数の曲線だけ描く
> for(i in seq(along=a)) lines(xs,xx1[,1:(a[i]+1)] %*% jf[[a[i]]]$coef,col=i)
```




```
> ## 決定係数によるモデル比較
> jrsq <- sapply(jf,function(a) a$fsummary[[1,2]]) # rsqだけ取り出してベクトルにする
> jrsq
     deg1      deg2      deg3      deg4      deg5      deg6      deg7      de
0.1988971 0.4719236 0.5437978 0.6243143 0.6278945 0.6588093 0.6644755 0.66456
     deg9     deg10
0.6647764 0.6653465
> ## 修正決定係数によるモデル比較
> jrsqadj <- sapply(jf,function(a) a$rsqadj[[1]]) # rsqadjだけ取り出してベクトルにする
> jrsqadj
     deg1      deg2      deg3      deg4      deg5      deg6      deg7      de
0.1810948 0.4479201 0.5119697 0.5885347 0.5825158 0.6076307 0.6042531 0.59394
     deg9     deg10
0.5832355 0.5723872
> ## aicによるモデル比較
```

```
> jaic <- sapply(jf,function(a) a$aic[[1]]) # aicだけ取り出してベクトルにする
> jaic
    deg1     deg2     deg3     deg4     deg5     deg6     deg7     deg8
160.2884 142.7013 137.8249 130.6983 132.2483 130.1717 131.3846 133.3719
    deg9    deg10
135.3424 137.2624
> ## プロット
> matplot(1:m0,cbind(jrsq,jrsqadj),type="b",xlab="deg")
> plot(1:m0,jaic,type="b",xlab="deg")
```




```
> ## 結果をまとめる
> cbind(jrsq,jrsqadj,jaic)
           jrsq   jrsqadj     jaic
deg1  0.1988971 0.1810948 160.2884
deg2  0.4719236 0.4479201 142.7013
deg3  0.5437978 0.5119697 137.8249
deg4  0.6243143 0.5885347 130.6983
deg5  0.6278945 0.5825158 132.2483
deg6  0.6588093 0.6076307 130.1717
deg7  0.6644755 0.6042531 131.3846
deg8  0.6645659 0.5939482 133.3719
deg9  0.6647764 0.5832355 135.3424
deg10 0.6653465 0.5723872 137.2624
> ordval <- function(x) { # 数値の順位を返す関数を定義
+ a <- 1:length(x)
+ a[order(x)] <- a
+ a
+ }
```

```
> order(jaic)
 [1]  6  4  7  5  8  9 10  3  2  1
> ordval(jaic)
 [1] 10  9  8  2  4  1  3  5  6  7
> cbind(jrsq,ordval(-jrsq),jrsqadj,ordval(-jrsqadj),jaic,ordval(jaic))
           jrsq      jrsqadj        jaic
deg1  0.1988971 10 0.1810948 10 160.2884 10
deg2  0.4719236  9 0.4479201  9 142.7013  9
deg3  0.5437978  8 0.5119697  8 137.8249  8
deg4  0.6243143  7 0.5885347  4 130.6983  2
deg5  0.6278945  6 0.5825158  6 132.2483  4
deg6  0.6588093  5 0.6076307  1 130.1717  1
deg7  0.6644755  4 0.6042531  2 131.3846  3
deg8  0.6645659  3 0.5939482  3 133.3719  5
deg9  0.6647764  2 0.5832355  5 135.3424  6
deg10 0.6653465  1 0.5723872  7 137.2624  7
```

重回帰分析のモデル選択

```
> ## 説明変数のすべての組み合わせで回帰分析
> save.image()
> ax <- c("E09504","A0410302","C01301","B02101"); x <- X2000$x[,ax]
> ay <- "A05203"; y <- X2000$x[,ay]
> X2000$jitem[c(ax,ay)]
                                E09504                                 A0410302
"最終学歴が大学・大学院卒の者の割合 " "未　婚　者　割　合［20～24歳・女］ "
                                C01301                                 B02101
          "県民１人当たり県民所得　"                "年　平　均　気　温　"
                                A05203
            "合　計　特　殊　出　生　率　"
> p0 <- ncol(x) # 説明変数の数
> p0
[1] 4
> f0 <- mylsfit(x,y)  # すべての変数
> f0$tsummary
, ,  = Y
```

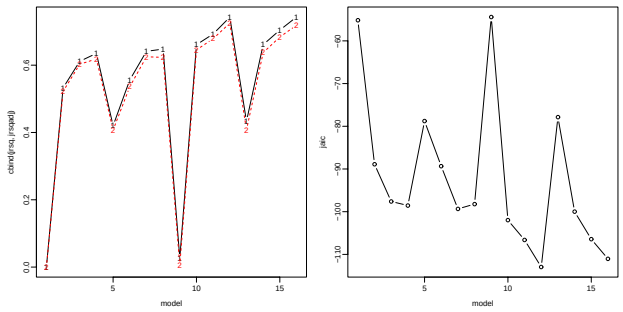


```
                Estimate      Std.Err    t-value     Pr(>|t|)
  Intercept  3.636118e+00 6.108140e-01  5.9529052 4.642519e-07
  E09504    -1.655659e-02 6.570354e-03 -2.5198932 1.562857e-02
  A0410302  -2.727500e-02 7.438516e-03 -3.6667262 6.850351e-04
  C01301     1.228960e-05 4.635665e-05  0.2651098 7.922217e-01
  B02101     2.012511e-02 5.098161e-03  3.9475245 2.951848e-04

> f0$fsummary
  Mean Sum Sq R Squared  F-value Df 1 Df 2       Pr(>F)
Y  0.07061925 0.7431179 30.37478    4    42 6.678249e-12
> genbit <- function(m) {
+   n <- 2^m
+   x <- matrix(logical(n*m),m)
+   for(i in 1:m) {
+     k <- 2^(i-1)
+     x[i,] <- c(rep(F,k),rep(T,k))
+   }
+   x
+ }
>
> jb <- genbit(p0) # ４個の説明変数の組み合わせ
> jb
      [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10] [,11] [,12]
[1,] FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE
[2,] FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE
[3,] FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE
[4,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE
     [,13] [,14] [,15] [,16]
[1,] FALSE  TRUE FALSE  TRUE
[2,] FALSE FALSE  TRUE  TRUE
[3,]  TRUE  TRUE  TRUE  TRUE
[4,]  TRUE  TRUE  TRUE  TRUE
> jna <- apply(jb,2,function(x) paste((1:p0)[x],collapse="")) # 名前
> jna <- paste("(",jna,")",sep="")
> jna
 [1] "()"     "(1)"    "(2)"    "(12)"   "(3)"    "(13)"   "(23)"   "(123)"
 [9] "(4)"    "(14)"   "(24)"   "(124)"  "(34)"   "(134)"  "(234)"  "(1234)"
> m0 <- dim(jb)[2] # 組み合わせの数 = 2^4 = 16
> m0
[1] 16
> jf <- vector(m0,mode="list") # 回帰分析の結果をしまう場所
> names(jf) <- jna
> x1 <- cbind(1,x) # 定数項の列も加える
> for(j in 1:m0) { # すべての組み合わせで回帰分析
+ jx <- x1[,c(T,jb[,j])]
+ jf[[j]] <- mylsfit(jx,y,int=F)
+ }
> ## 決定係数によるモデル比較
> jrsq <- sapply(jf,function(a) a$fsummary[[1,2]]) # rsqだけ取り出してベクトルにする
> jrsq
       ()       (1)       (2)      (12)       (3)      (13)      (23)      (12
0.9920675 0.9962908 0.9969184 0.9971052 0.9954002 0.9964788 0.9971539 0.9972
      (4)      (14)      (24)     (124)      (34)     (134)     (234)     (123
0.9922747 0.9973095 0.9975616 0.9979589 0.9955045 0.9973100 0.9976542 0.9979
```

```
> # 上で求めた決定係数はまちがい（int=Fとして計算したため）
> jrsq <- sapply(jf,function(a) cor(a$pred,y)^2)
Warning message:
The standard deviation is zero in: cor(x, y, na.method)
> jrsq # これでもモデル()でうまく行かない
       ()       (1)       (2)      (12)       (3)      (13)      (23)
       NA 0.53240778 0.61152634 0.63506910 0.42013785 0.55610091 0.64121051
     (123)       (4)      (14)      (24)     (124)      (34)     (134)
0.64780881 0.02612046 0.66082410 0.69261296 0.74268801 0.43327717 0.66088575
     (234)     (1234)
0.70428060 0.74311788
> jrsq <- sapply(jf,function(a) var(a$pred)/var(y))
> jrsq # これでやっと決定係数がすべてのモデルで計算できた
       ()       (1)       (2)      (12)       (3)      (13)      (23)
0.00000000 0.53240778 0.61152634 0.63506910 0.42013785 0.55610091 0.64121051
     (123)       (4)      (14)      (24)     (124)      (34)     (134)
0.64780881 0.02612046 0.66082410 0.69261296 0.74268801 0.43327717 0.66088575
     (234)     (1234)
0.70428060 0.74311788
```

```
> ## 修正決定係数によるモデル比較
> jrsqadj <- sapply(jf,function(a) 1-(length(y)-1)/(length(y)-length(a$coef))
> jrsqadj
        ()         (1)         (2)        (12)         (3)        (13)
0.000000000 0.522016839 0.602893591 0.618481327 0.407252028 0.535923676
      (23)       (123)         (4)        (14)        (24)       (124)
0.624901897 0.623237331 0.004478694 0.645407016 0.678640817 0.724736015
      (34)       (134)       (234)      (1234)
0.407517041 0.637226618 0.683649012 0.718652920
> ## aicによるモデル比較
> jaic <- sapply(jf,function(a) a$aic[[1]]) # aicだけ取り出してベクトルにする
> jaic
       ()        (1)        (2)       (12)        (3)       (13)       (23)
 -55.16929  -88.89675  -97.60920  -98.54751  -78.78264  -89.34072  -99.34521
     (123)        (4)       (14)       (24)      (124)       (34)      (134)
 -98.21760  -54.41327 -101.98740 -106.61273 -112.97019  -77.85988  -99.99595
     (234)     (1234)
-106.43147 -111.04878
```

```
> ## プロット
> matplot(1:m0,cbind(jrsq,jrsqadj),type="b",xlab="model")
> plot(1:m0,jaic,type="b",xlab="model")
```




```
> ## 結果をまとめる
> cbind(1:m0,jrsq,ordval(-jrsq),jrsqadj,ordval(-jrsqadj),jaic,ordval(jaic))
               jrsq         jrsqadj          jaic
()      1 0.00000000 16 0.000000000 16  -55.16929 15
(1)     2 0.53240778 12 0.522016839 12  -88.89675 12
(2)     3 0.61152634 10 0.602893591 10  -97.60920 10
(12)    4 0.63506910  9 0.618481327  9  -98.54751  8
(3)     5 0.42013785 14 0.407252028 14  -78.78264 13
(13)    6 0.55610091 11 0.535923676 11  -89.34072 11
(23)    7 0.64121051  8 0.624901897  7  -99.34521  7
(123)   8 0.64780881  7 0.623237331  8  -98.21760  9
(4)     9 0.02612046 15 0.004478694 15  -54.41327 16
(14)   10 0.66082410  6 0.645407016  5 -101.98740  5
(24)   11 0.69261296  4 0.678640817  4 -106.61273  3
(124)  12 0.74268801  2 0.724736015  1 -112.97019  1
(34)   13 0.43327717 13 0.407517041 13  -77.85988 14
(134)  14 0.66088575  5 0.637226618  6  -99.99595  6
(234)  15 0.70428060  3 0.683649012  3 -106.43147  4
(1234) 16 0.74311788  1 0.718652920  2 -111.04878  2
```

第７回 課題

1.　修正決定係数と赤池情報量規準を計算する関数 `modelcrit` を書け．ただし入力は `lsfit` の出力としてリストの要素 `resid` と `coef` を用いる．`modelcrit` の出力はリストとし要素 `rsqadj` に修正決定係数，`aic` に赤池情報量規準を格納する．

```
modelcrit <- function(f) {
#  ここでf$residとf$coefからrsqadjとaicを計算する
 list(rsqadj,aic)
}
f <- lsfit(x,y)
a <- modelcrit(f)
```

2.　`X2000$x` から適当な項目を選んで重回帰分析をする．ただし説明変数は４個以上とする．`mylsfit` の出力する `tsummary` を示せ．自分で作成した `modelcrit` を用いて修正決定係数と赤池情報量規準を計算し，その値を `mylsfit` の出力する `rsqadj` と `aic` と比べ同じ値になっているを確認せよ．

3.　上記２で行った重回帰分析について，すべての説明変数の組み合わせについて部分回帰を計算する．もし説明変数がp個あれば組み合わせは2^p個である．すべての部分回帰に`modelcrit`を適用し修正決定係数と赤池情報量基準を計算せよ．修正決定係数と赤池情報量規準をもちいたモデル選択を行い，それぞれの規準によって選ばれた上位１０個のモデルを示せ．(順位，変数の組み合わせ，規準の値からなる３列の表が二つ)