スパース推定: 手を動かしてみる

鈴木 譲

2018 年 11 月 4 日

1 線形回帰

1. 説明変数と目的変数の N 個の組 $(x_1, y_1), \cdots, (x_N, y_N)$ から、$S := \sum_{i=1}^{N}(y_i - \beta_0 - \beta_1 x_i)^2$ を最小にする切片 β_0 と傾き β_1 をもとめたい。その解 $\hat{\beta}_0, \hat{\beta}_1$ について、$\hat{\beta}_0 + \bar{x}\hat{\beta}_1 = \bar{y}$ が成立することを示せ。次に、$\bar{x} := \dfrac{1}{N}\sum_{i=1}^{N} x_i$, $\overline{x^2} := \dfrac{1}{N}\sum_{i=1}^{N} x_i^2$, $\bar{y} := \dfrac{1}{N}\sum_{i=1}^{N} y_i$, $\overline{xy} := \dfrac{1}{N}\sum_{i=1}^{N} x_i y_i$ とおき、x_i, y_i を $x_i - \bar{x}, y_i - \bar{y}$, $i = 1, \cdots, N$ でおきかえるとき、$\hat{\beta}_0 = 0$, $\hat{\beta}_1 = \overline{xy}/\overline{x^2}$ となることを示せ。

以下では、p 個の説明変数と目的変数の N 個の組 $(x_{1,1}, \cdots, x_{1,p}, y_1), \cdots, (x_{N,1}, \cdots, x_{N,p}, y_N)$ から、切片 β_0 と各変数の傾き $\beta = (\beta_1, \cdots, \beta_p)$ を推定したい。各 $x_{i,j}$ から $\bar{x}_j = \dfrac{1}{N}\sum_{i=1}^{N} x_{i,j}$, $j = 1, \cdots, p$ をひき、各 y_i から $\bar{y} = \dfrac{1}{N}\sum_{i=1}^{N} y_i$ をひき、それぞれの平均を 0 にしてから β を推定する (推定値を $\hat{\beta}$ とする)。最後に $\hat{\beta}_0 = \bar{y} - \sum_{j=1}^{p} \bar{x}_j \hat{\beta}_j$ とする。また、$X = (x_{i,j})_{1 \le i \le N, 1 \le j \le p} \in \mathbb{R}^{N \times p}$, $y = (y_i)_{1 \le i \le N} \in \mathbb{R}^N$ とおくものとする。

2. $X^T X$ が正則のときに [1][2]、
$$||y - X\beta||_2^2 := \sum_{i=1}^{N}(y_i - \beta_1 x_{i,1} - \cdots - \beta_p x_{ip})^2$$
を最小にする β が $\hat{\beta} = (X^T X)^{-1} X^T y$ であたえられることを示せ。

3. 行列 $X \in \mathbb{R}^{N \times p}$ とベクトル $y \in \mathbb{R}^N$ を入力し、切片の推定値 $\hat{\beta}$ と傾きの推定値 $\hat{\beta} \in \mathbb{R}^p$ の推定値を出力する関数 `linear` を R 言語で構成したい。以下の (1)(2) をうめよ。

```
inner.prod=function(x,y)sum(x*y)
linear = function(X,y){
n=nrow(X); p=ncol(X);
X=as.matrix(X); x.bar=array(dim=p); for(j in 1:p)x.bar[j]=mean(X[,j]);
     for(j in 1:p)X[,j]=X[,j]-x.bar[j];                    ## X の中心化
y=as.vector(y); y.bar=mean(y); y=y-y.bar                    ## y の中心化
beta=as.vector(## ここをうめる (1)
beta.0=## ここをうめる (2)
return(list(beta=beta,beta.0=beta.0))
}
```

[1] T で転置をあらわす。

[2] $z = [z_1, \cdots, z_N]^T \in \mathbb{R}^N$ に対して、$||z||_2 := \sqrt{z_1^2 + \cdots + z_N^2}$ とかくものとする。

4. λ を非負定数として、

$$\frac{1}{2N}||y - X\beta||_2^2 + \frac{\lambda}{2}||\beta||_2^2 := \frac{1}{2N}\sum_{i=1}^{N}(y_i - \beta_1 x_{i,1} - \cdots - \beta_p x_{ip})^2 + \frac{\lambda}{2}\sum_{j=1}^{p}\beta_j^2$$

を最小にする β をもとめよ (Ridge 回帰)。

5. 問題 3 の関数にさらに、入力としてさらに非負定数 λ を加えて、問題 4 の処理を行う R 言語の関数 `ridge` を求めよ。

6. `https://web.stanford.edu/~hastie/StatLearnSparsity/data.html` の米国犯罪データをダウンロード (その画面で Ctrl-a, Ctrl-c でコピーして、各自 PC のエディタに新規ファイルオープンして、Ctrl-v でペースト) して、crime.txt というファイルに格納し、問題 3, 問題 5 で作成したプログラムを実行せよ。

列	説明/目的	変数の意味
1	目的	人口 100 万人あたりの犯罪率
2		(今回は用いない)
3	説明	警察官の年間給与
4	説明	25 歳以上で高校を卒業した人の割合
5	説明	16-19 歳で高校に通っていない人の割合
6	説明	18-24 歳で大学生の割合
7	説明	25 歳以上で 4 年制大学を卒業した人の割合

R を実行する際には、そのファイルが格納されているディレクトリにアクセスできるようにする。

```
> crime=read.table("crime.txt")
> X=crime[,3:7]
> y=crime[,1]
> linear(X,y)
$beta
[1] 10.9806703 -6.0885294  5.4803042  0.3770443  5.5004712
$beta.0
[1] 489.6486
> ridge(X,y)
$beta
[1] 10.9806703 -6.0885294  5.4803042  0.3770443  5.5004712
$beta.0
[1] 489.6486
> ridge(X,y,100)
$beta
[1]  5.32672839 -1.38451516  1.43075509 -0.81986481  0.09200358
$beta.0
[1] 599.4411
```

7. $p = 2$ のとき、X^TX の各成分を、$v_1 := \sqrt{\frac{1}{N}\sum_{i=1}^{N} x_{i,1}^2}$, $v_2 := \sqrt{\frac{1}{N}\sum_{i=1}^{N} x_{i,2}^2}$, $\rho := \frac{\frac{1}{N}\sum_{i=1}^{N} x_{i,1}x_{i,2}}{v_1 v_2}$ を用いて表示せよ。

8. X^TX の固有値が $\gamma_1, \cdots, \gamma_p$ のとき、$X^TX + N\lambda I$ の固有値が $\gamma_1 + N\lambda, \cdots, \gamma_p + N\lambda$ となることを示せ。また、X^TX に逆行列が存在しない条件を $\gamma_1, \cdots, \gamma_p$ を用いてあらわせ。さらに、$\lambda > 0$ である限り、$X^TX + N\lambda I$

には逆行列が必ず存在することを示せ。ただし、非負定値[3]の固有値がすべて非負となることは証明しないで用いてよい。

9. 関数 $f:\mathbb{R}^p \to \mathbb{R}$ が、任意の $0<\alpha<1$ と $u_1, u_2 \in \mathbb{R}^p$ について

$$f(\alpha u_1 + (1-\alpha)u_2) \leq \alpha f(u_1) + (1-\alpha) f(u_2)$$

であるとき、関数 f は凸であるという。この定義に基づいて、下記の各関数が凸であることを示せ。

(a) $f(u) = u^2$

(b) $f(u,v) = u^2 + v^2$

(c) $f(u) = |u|$

10. 以下の各関数 $f:\mathbb{R} \to \mathbb{R}$ と $u_0 \in \mathbb{R}$ について、

$$f(u) \geq f(u_0) + z(u-u_0)\ ,\ u \in \mathbb{R}$$

が成立する $z \in \mathbb{R}$ の集合、および $f(u)$ の $u=u_0$ での微分係数 $f'(u_0)$ を求めよ。今後、この集合[4]を $\partial f(u)$ とかくものとする (劣勾配)。

(a) $f(u) = u$, $u_0 = 0$

(b) $f(u) = u^2 + 2u$, $u_0 = 1$

また、$u=u_0$ で微分可能であれば、$\partial f(u)$ は、その微分係数となることを証明せよ。

11. $f(u) = |u|$ が $u=0$ で微分できないこと、および

$$\partial f(u_0) = \begin{cases} -1, & u_0 < 0 \\ [-1,1], & u_0 = 0 \\ 1, & u_0 > 0 \end{cases}$$

を示せ。

12. λ を非負定数として

$$f(u) = \frac{1}{2N}\sum_{i=1}^{N}(y_i - u x_{i,1})^2 + \lambda |u|$$

について

$$\partial f(u_0) = \begin{cases} -\frac{1}{N}\sum_{i=1}^{N} x_{i,1}(y_i - u_0 x_{i,1}) - \lambda, & u_0 < 0 \\ -\frac{1}{N}\sum_{i=1}^{N} x_{i,1} y_i + \lambda[-1,1], & u_0 = 0 \\ -\frac{1}{N}\sum_{i=1}^{N} x_{i,1}(y_i - u_0 x_{i,1}) + \lambda, & u_0 > 0 \end{cases}$$

となることを示せ。また、$0 \in \partial f(0)$ となる条件を求めよ。ただし、劣勾配の場合でも微分と同様、関数の和の劣勾配がそれぞれの劣勾配の和になること、関数の定数倍の劣勾配がもとの関数の劣勾配の定数倍になることは、証明無しで用いて良い。

13. $p=1$, $\frac{1}{N}\sum_{i=1}^{N} x_{i,1}^2 = 1$ のとき、λ を非負定数として、

$$\frac{1}{2N}||y - X\beta||^2 + \lambda||\beta||_1 := \frac{1}{2N}\sum_{i=1}^{N}(y_i - \beta_1 x_{i,1})^2 + \lambda|\beta_1|$$

を最小にする β_1 をもとめよ (Lasso 回帰)。

---

[3] $X^T X$ のように、ある行列 A があって $A^T A$ とかける行列として定義される。

[4] 要素が 1 個しかない場合、集合ではなく ({,} をつけず) 要素だけを書く場合がある。

14. λ を非負定数として、関数

$$S_\lambda(x) := \begin{cases} x-\lambda, & x > \lambda \\ 0, & |x| \le \lambda \\ x+\lambda, & x < -\lambda \end{cases}$$

が、関数 $(x)_+ = \max\{x, 0\}$,

$$\mathrm{sign}(x) = \begin{cases} -1, & x < 0 \\ 0, & x = 0 \\ 1, & x > 0 \end{cases}$$

を用いて、$S_\lambda(x) = \mathrm{sign}(x)(|x| - \lambda)_+$ とかけることを示せ。

15. $\lambda > 0, x \in \mathbb{R}$ を入力として、$S_\lambda(x)$ を出力する関数 `soft.th` を R 言語でかけ。さらに、下記を実行し、関数の動作が正しいことを確認せよ (出力を pdf で提出する)。

```
curve(soft.th(5,x),-10,10)
```

場合分けで関数を定義したり、`max` を用いると、R 言語ではグラフを描いてくれない。`sign`, `abs`, `pmax` を用いよ。

16. 問題 13 の解を関数 $S_\lambda(\cdot)$ を用いてかけ。また、R 言語で記述するとき、下記の空欄をうめよ。

```
linear.lasso.1=function(x,y,lambda){
x.bar=mean(x); y.bar=mean(y); y=y-y.bar; N=length(y)
x.scale=scale(x) # 中心化するだけでなく、分散が 1 になるようにする
beta=## ここをうめる
beta.0=y.bar-beta*x.bar
return(list(beta=beta,beta.0=beta.0)
}
```

17. 下記は、j 番目の説明変数を固定して、残差

$$y_i - \sum_{k \neq j} x_{i,k}\hat{\beta}_k \ , \ i = 1, \cdots, N$$

を計算して、係数 $\hat{\beta}_j$ を推定するということを、$j = 1, \cdots, p$ で繰り返し、さらにそれら全体を繰り返して、$\hat{\beta}_1, \cdots, \hat{\beta}_p$ の値の収束をまつ処理 (Lasso 推定の座標降下法) である。下記の空欄 (残差を計算する箇所) をうめて、$\lambda = 0$ のときに、関数 `linear` に一致することを確認せよ。

```
covar=function(x,y)sum(x*y)/length(x)
linear.lasso=function(X, y, lambda=0, standardize=TRUE){
    X=as.matrix(X)
    p=ncol(X)
    X.bar=array(dim=p); scale=array(dim=p)
    for(j in 1:p){X.bar[j]=mean(X[,j]);X[,j]=X[,j]-X.bar[j];}
    y.bar=mean(y); y=y-y.bar
    beta=array(0, dim=p); beta.old=array(0, dim=p)
    eps=1
    if(standardize==TRUE)for(j in 1:p){scale[j]=sqrt(covar(X[,j],X[,j]));X[,j]=X[,j]/scale[j];}
    while(eps>0.001){
        for(j in 1:p){
            r= ##ここをうめる
```

```
            beta[j]=soft.th(lambda,covar(r,X[,j]))/covar(X[,j],X[,j])
        }
        eps=max(abs(beta-beta.old))
        beta.old=beta
    }
    if(standardize==TRUE)for(j in 1:p)beta[j]=beta[j]/scale[j]
    beta.0=y.bar-innerprod(X.bar,beta)
    return(list(beta=beta, beta.0=beta.0))
}
```

18. 下記は、犯罪のデータについて、λ の値を変えながら、各変数の係数がどのように変化するかを表示したものである。plot と lines の行の λ を $\log\lambda$ にかえて、グラフを表示させよ (出力を pdf で提出する)。

```
df=read.table("crime.txt")
x=df[,3:7]
y=df[,1]
p=ncol(x)
lambda.seq=seq(0,2000)
coef.seq=lambda.seq
plot(lambda.seq, coef.seq, xlim=c(0,2000), ylim=c(-12,12),
     xlab="lambda",ylab="係数",main="lambda と係数の変化を見る",
     type="n", col="red") ##この 1 文
for(j in 1:p){
coef.seq=NULL; for(lambda in lambda.seq)coef.seq=c(coef.seq,ridge(x,y,lambda)$beta[j])
par(new=TRUE)
lines(lambda.seq,coef.seq, col=j) ##この 1 文
}
legend("topright",legend=
c("警察官の年間給与","25 歳以上で高校を卒業した人の割合",
"16-19 歳で高校に通っていない人の割合","18-24 歳で大学生の割合",
"25 歳以上で 4 年制大学を卒業した人の割合"), col=1:p, lwd=2, cex =.8)
```

19. ライブラリ glmnet をインストールして、linear.lasso を glmnet におきかえて、犯罪のデータを実行せよ。glmet は、X,y とも行列でないと受け付けない。したがって、

```
X=as.matrix(X); y=as.vector(y)
result=glmnet(X,y)
```

のようにする。result は種々の結果がはいっているが、plot(result) を実行する。

20. 問題 18 の関数 ridge を linear.lasso にかえて実行し、グラフを表示させよ。問題 18 で行った変更と同様に、横軸は log(lambda.seq) で表示せよ。

21. 各 j で、$\sum_{i=1}^{N} x_{i,j}^2 = 1$ のとき、すべての変数の係数が 0 になるような λ の最小値が、$\lambda = \max_{1\le j\le p} \frac{1}{N}\sum_{i=1}^{N} x_{i,j}y_i$ であたえられることを示せ。

22. 問題 17 の関数 linear.lasso は、glmnet と同じオプション standardize=TRUE, standardize=FALSE を設けている。standardize=TRUE では、全体の処理の前に、X の各列 (第 j 列) をその大きさ s_j で割って Lasso の処理を行い、座標降下法が収束したら、最後に β_j に s_j で割っている。standardize=FALSE では、そのような処理を行わない。以下の 2 条件のそれぞれで、両者は一致することを確認し、その理由を述べよ。

(a) $\lambda = 0$ のとき

(b) X の各列をその大きさで割って (正規化して) から、`linear.lasso` を実行したとき

通常の線形回帰 ($\lambda = 0$) で、X の各列 j を α_j で割るということは、$1/\alpha_1, \cdots, 1/\alpha_p$ を成分にもつ対角行列 Λ を X の右からかけることに相当する。$Z = X\Lambda$ とおくと、

$$(Z^TZ)^{-1}Z^Ty = (\Lambda^TX^TX\Lambda)^{-1}\Lambda^TX^Ty = \Lambda^{-1}(X^TX)^{-1}X^Ty = \Lambda^{-1}\hat{\beta} = \begin{bmatrix} \alpha_1 & & \\ & \ddots & \\ & & \alpha_p \end{bmatrix} \begin{bmatrix} \hat{\beta}_1 \\ \vdots \\ \hat{\beta}_p \end{bmatrix} = \begin{bmatrix} \alpha_1\hat{\beta}_1 \\ \vdots \\ \alpha_p\hat{\beta}_p \end{bmatrix}$$

となって、推定される $\hat{\beta}$ の各成分が α_j 倍される。したがって、X ではなく Z を用いて推定された $\hat{\beta}$ について、最後に各成分を α_j でわると、X のみを用いた場合と同じ $\hat{\beta}$ の値が求まる。

23. 座標降下法で、最初に λ の値を大きくしてすべての係数を 0 にし、λ の値を徐々に小さくしていく。そして、毎回、直前の λ での値を初期値として、次の λ の値を計算することを考える (warm start)。下記の (1)(2) をうめよ。

```
covar=function(x,y)sum(x*y)/length(x)
soft.th=function(lambda,x)sign(x)*pmax(abs(x)-lambda,0)

warm.start=function(X,y,lambda.max=100,standardize=TRUE){
     p=ncol(X); n=nrow(X);
     X=as.matrix(X); for(j in 1:p)X[,j]=X[,j]-mean(X[,j]);
     y=as.vector(y); y=y-mean(y);
     if(standardize==TRUE){
          scale=array(dim=p);
          for(j in 1:p){scale[j]=sqrt(covar(X[,j],X[,j]));X[,j]=X[,j]/scale[j];}
     }
     dec=round(lambda.max/50);
     lambda.seq=seq(lambda.max,1,-dec);
     r=length(lambda.seq);
     coef.seq=matrix(nrow=r,ncol=p);
     ## X[,3],...,X[,7] のそれぞれについて、係数の列を作る。最初は NULL
     beta=array(0, dim=p);
     k=0;
     for(lambda in lambda.seq){
          k=k+1;
          beta.old= ## (1) ここにいれる
          eps=1;
          while(eps>0.001){
               for(j in 1:p){
                    r= y- X[,-j] %*% beta[-j]
                    beta[j]=soft.th(lambda,covar(r,X[,j]))/covar(X[,j],X[,j])
               }
               eps=max(abs(beta-beta.old));
               beta.old=beta
          }
          if(standardize==TRUE)for(j in 1:p)beta[j]=beta[j]/scale[j];
```

```
            coef.seq[k,]= ## (2) ここにいれる
            ## 各 j=1,...,p に対して、coef.seq[[j]] の最後に係数を追加する。
      }
      return(coef.seq)
}

crime=read.table("crime.txt"); X=crime[,3:7]; y=crime[,1];
coef.seq=warm.start(X,y,300)
p=ncol(X)
lambda.max=100
dec=round(lambda.max/50);
lambda.seq=seq(lambda.max,1,-dec);
plot(log(lambda.seq),coef.seq[,1], xlab="log(lambda)", ylab="係数",
ylim=c(min(coef.seq),max(coef.seq)), type="n")
for(j in 1:p)lines(log(lambda.seq),coef.seq[,j], col=j)
```

24. 下記は、データセットから AIC で線形回帰の説明変数を選択する処理である。一般に説明変数が p 個ある場合に、AIC の値を何回計算して比較する必要があるか。また、実際に、犯罪率のデータに適用 (`crime=read.table("crime.txt"); X=crime[,3:7]; y=crime[,1]`) して、最適な変数の組を選択せよ。

```
crime=read.table("crime.txt")
X=as.matrix(crime[,3:7]); y=crime[,1]
p=ncol(X);n=length(y)
AIC=function(T){
      ss=T
      k=length(T);
      S=sum((lm(y~X[,T])$fitted.values-y)^2)/n;
      value=n*log(S)+2*k;
      print(ss); print(value);
      if(k==1)return(list(ss=ss,value=value));
      for(i in T){
            U=setdiff(T,i)
            info=AIC(U)
            if(info$value<value){ss=info$ss; value=info$value}
      }
      return(list(ss=ss,value=value))
}
AIC(1:p)
```

25. `X=as.matrix(X);y=as.vector(y);cv=cv.glmnet(X,y);plot(cv)` を実行して、出力を pdf でせよ。最上部 n の 0 から 5 までの数値は、どういう意味か。