DATA 581

Modeling and Simulation II

Lecture 3: Multiple Regression Models

What We Discuss Today

- Introduction
- Developing multiple regression models
- Matrix notation for multiple regression models
- QR decomposition
- Parameter estimation via QR decomposition
- Inference
- Variable selection
  - Traditional methods
  - Modern approaches

Introduction

- **Study of an event or phenomena will have various factors causing its occurrence.**

- **Multiple regression is a statistical technique dedicated to draw out a relationship between one response or dependent variable and multiple independent variables.**

$$y \sim g(x_1, x_2, \ldots, x_n)$$

House Price Prediction

Example: There are 24 observations on 9 variables are recorderd in the dataframe `table.b4` from the *MPV* package.

```
y sale price of the house (in thousands of dollars)
x1 taxes (in thousands of dollars)
x2 number of baths
x3 lot size (in thousands of square feet)
x4 living space (in thousands of square feet)
x5 number of garage stalls
x6 number of rooms
x7 number of bedrooms
x8 age of the home (in years)
x9 number of fireplaces
```

Question: Can we use predict the sale price of a house based on these collected variables (a.k.a features, covariates)?

Developing Multiple Regression models

Remember the simple linear regression model

$$y = \beta_0 + \beta_1 x + \epsilon\epsilon \sim N(0, \sigma^2)$$

Where we use only one variable to model y.

We extend this model to a multiple regression model by adding more variables.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + \epsilon$$

where k is the number of covariates included in the model.

For the house sale price example, if we included all covariates in the model, we would have $k = 9$ terms in the model in addition to the intercept.

不太重要

Developing multiple regression models

The multiple regression framework includes special types of nonlinear models.

For example, we could try to model the response as a kth degree polynomial.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_1^2 + \cdots + \beta_k x_1^k + \epsilon$$

- **We could also include polynomial terms in several different variables, such as and x_2^2,**
- **As well as interaction terms, such as $x_1 x_2$ or $x_1^2 x_2$.**

Therefore, being able to construct multiple regression models gives a great deal of flexibility to how data can be analyzed.

Matrix Notation for Multiple Regression Analysis

Note: Handling multiple covariates is more convenient if we use vectors and matrices.

The multiple regression model can be written as

$$\mathbf{Y} = \mathbf{X}\beta + \epsilon \tag{1}$$

where $\mathbf{Y}$ is the column vector of responses:

转置

$$\mathbf{Y} = [y_1 \; y_2 \; \cdots \; y_n]^{\mathsf{T}}.$$

We also have two column vectors for the coefficents and error term.

$$\beta = [\beta_0 \; \beta_1 \; \cdots \; \beta_k]^{\mathsf{T}}$$

and

$$\epsilon = [\varepsilon_1 \; \varepsilon_2 \; \cdots \; \varepsilon_n]^{\mathsf{T}}.$$

Matrix Formation Multiple Regression Analysis

X is an $n \times p$ matrix, called the model matrix or design matrix:

$$\mathrm{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nk} \end{bmatrix}$$

where n is the number of obsevations and

$$p = \begin{cases} k+1 & \text{if the selected model has intercept} \\ k & \text{otherwise} \end{cases} \tag{2}$$

k is the number of selected covariates to include in the model.

Note: First column of the design matrix is always 1 if the regression model have an intercept

Matrix form

The following special cases can help to illustrate the notation.

Example1: A simple linear regression model with no intercept (regression through the origin):

$$y = \beta_1 x + \varepsilon$$

is

$$\mathbf{Y} = \mathbf{X}\beta + \epsilon$$

with

$$\mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad \mathbf{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \epsilon = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix} \text{ and } \beta = [\beta_1]$$

Matrix form

Example2: **Simple linear regression (with intercept):**

$$y = \beta_0 + \beta_1 x + \varepsilon$$

is

$$\mathbf{Y} = \mathbf{X}\beta + \epsilon$$

with

$$\begin{matrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{matrix}, \quad \mathbf{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \epsilon = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix} \quad \text{and } \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}$$

Matrix form

Example3: Quadratic regression:

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \varepsilon$$

is

$$\mathbf{Y} = \mathbf{X}\beta + \epsilon$$

with

$$\mathbf{X} = \begin{bmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 \end{bmatrix}, \quad \mathbf{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \epsilon = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix} \quad \text{and } \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix}$$

Matrix form -Exam

Example4: Regression with 2 predictor variables

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \varepsilon$$

is

$$\mathbf{Y} = \mathbf{X}\beta + \epsilon$$

with

$$\mathbf{X} = \begin{bmatrix} 1 & x_{11} & x_{12} \\ 1 & x_{21} & x_{22} \\ \vdots & \vdots & \vdots \\ 1 & x_{n1} & x_{n2} \end{bmatrix}, \quad \mathbf{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \epsilon = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix} \quad \text{and } \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix}$$

QR Decomposition of the Design Matrix

QR decomposition of the design matrix

For any $n \times p$ matrix X whose columns are linearly independent ;

- there exists an $n \times n$ orthogonal matrix Q 正交矩阵
  - Orthogonality of Q means that $\mathrm{Q}^{\intercal}\mathrm{Q} = \mathbf{I}$. I：单位矩阵
- and exists an invertible $n \times p$ upper triangular matrix R
  - bottom $n \times p$ rows of R consist of 0's and the top p rows of R constitute a nonsingular upper triangular matrix U.

$$\mathrm{R} = \begin{bmatrix} \mathbf{U} \\ \mathbf{0} \end{bmatrix}$$

such that

$$\mathrm{X} = \mathrm{QR}$$

In order for the columns of X to be linearly independent, it is necessary that $p \leq n$.

Example – house price data

We can easily obtaines QR decomposition of a matrix in R.

For the house price data, we can obtain these quantities in the following way:

```r
n <- nrow(table.b4) # this counts the number of observations
X <- table.b4[,-1] # removes the y vector
X <- cbind(x0=rep(1, n), X) # this adds the vector of
##1's to the design matrix as we defined the model
## with intercept.
X.QR <- qr(X)
R <- qr.R(X.QR, complete=TRUE)
Q <- qr.Q(X.QR, complete=TRUE)
```

Parameter Estimation via the QR decomposition

Parameter Estimation via the QR decomposition

Consider the usual multiple regression model again

$$\mathbf{Y} = \mathbf{X}\beta + \epsilon$$

- The columns of X are assumed to be linearly independent.
- if an intercept is in the model, the first column is a vector of 1's.

- The elements of ϵ are assumed to be uncorrelated random variables with mean 0 and common variance σ^2.

Thus, the expected value and variance of the vector $\mathbf{Y}$ are

$$E\left[\mathbf{Y}|\mathbf{X}\right] = E[\mathbf{X}\beta] + E[\epsilon]$$

and

$$Var\left[\mathbf{Y}|\mathbf{X}\right] = Var[\mathbf{X}\beta] + Var(\epsilon)$$

where $Var(\epsilon) = \sigma^2 I$ and I in an $n \times n$ identity matrix.

Parameter Estimation via the QR decomposition

- **The least-squares approach is a common technique that is used to estimate the parameters, β, in the regression model by minimizing Residual Sum of Squares**

- **It finds β such that $\sum |\epsilon^2|$ is minized.**

Estimate of the parameters can be done by using calculus and QR method.

β estimation via QR

Model: $\mathbf{Y} = \mathrm{X}\beta + \epsilon = \mathrm{QR}\beta + \epsilon = \mathrm{Q}_1\mathrm{U} + \epsilon$

Where we have partitioned Q as $[Q_1 \;\; Q_2]$.

- **Q_1 represents the first p columns of Q**
- **Q_2 is the remaining np columns of Q.**

Multiplying the model equation on the left by $\mathrm{Q}^\intercal$ gives

$$\mathrm{Q}^\intercal \mathbf{Y} = \mathrm{R}\beta + \mathrm{Q}^\intercal \varepsilon$$

which can be re-written as the summation of two statements.

$$\mathrm{Q}_1^\intercal \mathbf{Y} = \mathrm{U}\beta + \mathrm{Q}_1^\intercal \epsilon \tag{3}$$

and

$$\mathrm{Q}_2^\intercal \mathbf{Y} = \mathrm{Q}_2^\intercal \epsilon \tag{4}$$

β estimation via QR

To estimate β, it is clear that model (3) contains all of the needed information. and the ordinary least-squares estimate must be the solution of the upper triangular linear system

$$\mathbf{U}\widehat{\beta} = \mathbf{Q}_1^{\intercal}\mathbf{Y}$$

so an estimate of β is obtained by:

$$\widehat{\beta} = \mathbf{U}^{\intercal}\mathbf{Q}_1^{\intercal}\mathbf{Y}$$

β Estimation via QR - Example

We illustrate this techniques on the house sales data in `table.b4`**.**

```r
X <-  table.b4[,-1] # remove "y" column to form model matrix
n <- nrow(table.b4); X1 <- rep(1, n) # column for intercept
X <- cbind(X1, X) # append 1's to create full model matrix
QR <- qr(X) # QR decomposition of X
Q <- qr.Q(QR, complete=TRUE) # complete=TRUE gives Q1 & Q2
p <- 10 # p = 9+1 for this problem
Q1 <- Q[, 1:p]
y <- table.b4[, 1]
Q1y <- t(Q1)%*%y
U <- qr.R(QR)  # if you want all of R, use complete=TRUE
betahat <- solve(U, Q1y)
```

β Estimation via QR - Example

```
betahat

##            [,1]
## X1 14.92764759
## x1  1.92472156
## x2  7.00053420
## x3  0.14917793
## x4  2.72280790
## x5  2.00668402
## x6 -0.41012376
## x7 -1.40323530
## x8 -0.03714908
## x9  1.55944663
```

So the fitted model:

$$\widehat{E[y|x]} = 14.9 + 1.9x_1 + 7x_2 + 0.1x_3 + 2.7x_4 + 2x_5 - 0.4x_6 - 1.4x_7 - 0.04x_8 + 1.6x_9$$

We can check the estimated parameters from:

```
y.lm <- lm(y ~ ., data = table.b4)
coef(y.lm)

## (Intercept)          x1          x2          x3          x4          x5
## 14.92764759  1.92472156  7.00053420  0.14917793  2.72280790  2.00668402
##          x6          x7          x8          x9
## -0.41012376 -1.40323530 -0.03714908  1.55944663
```

Estimating σ^2 with QR decompition method

To estimate σ^2, we can use model equation (4).

$$\mathrm{Q}_2^{\intercal}\mathbf{Y} = \mathrm{Q}_2^{\intercal}\epsilon$$

It is easily shown that

-
$$E[\mathrm{Q}_2^{\intercal}\mathbf{Y}|\mathrm{X}] = 0$$

-
$$\text{Var}(\mathrm{Q}_2^{\intercal}\mathbf{Y}|\mathrm{X}) = \mathrm{Q}_2^{\intercal}\sigma^2\mathbf{I}\mathrm{Q}_2 = \sigma^2\mathbf{I}.$$

So, $\mathrm{Q}_2^{\intercal}\mathbf{Y}$ is a vector of $n-p$ uncorrelated mean 0 and variance σ^2 random variables.

Parameter Estimation via QR

$\mathbf{Y}^\intercal \mathrm{Q}_2 \mathrm{Q}_2^\intercal \mathbf{Y}$ must be the sum of squares of such random variables.

Which means

$$\text{i.e.} \quad \mathbf{Y}^\intercal \mathrm{Q}_2 \mathrm{Q}_2^\intercal \mathbf{Y} = \sum_{i=1}^{n-p} Z_i^2 \tag{5}$$

where $Z_1, Z_2, \ldots, Z_{n-p}$ are uncorrelated random variables with mean 0 and variance Var$(Z_i) = \sigma^2$.

Therefore,

$$E[\mathbf{Y}^\intercal \mathrm{Q}_2 \mathrm{Q}_2^\intercal \mathbf{Y}] = \sum_{i=1}^{n-p} \mathbf{Var}(Z_i) = (n-p)\sigma^2,$$

$\rightsquigarrow$ **An unbiased estimator for σ^2 is**

$$\widehat{\sigma}^2 = \textbf{MSE} = \frac{\mathbf{Y}^\intercal \mathrm{Q}_2 \mathrm{Q}_2^\intercal \mathbf{Y}}{n-p}$$

Note that the residual sum of squares is

$$\text{SSE} = \mathbf{Y}^{\intercal}\mathrm{Q}_2\mathrm{Q}_2^{\intercal}\mathbf{Y}.$$

If we make the further assumption that $\mathbf{Y}$ is normal, then the discussion around (5) implies that the SSE is a sum of squares of uncorrelated normal random variables Z_i with mean 0 and variance Var$(Z_i) = \sigma^2$.

Parameter Estimation via QR - Example

Residual Sum of Squares:

```
Q2 <- Q[, -(1:p)]
Q2y <- t(Q2)%*%y

SSE <- t(Q2y)%*%Q2y
SSE

##           [,1]
## [1,] 121.7482
```

Variance Estimate and Root-MSE*:

```
MSE <- SSE/(n-p)
MSE

##           [,1]
## [1,] 8.696297
```

```
sqrt(MSE)

##           [,1]
## [1,] 2.948949
```

The root-MSE is part of the output from `summary(y.lm)`.

*This is the estimate of σ, the noise standard deviation.

Prediction error

We may also wish to predict a single response value y for given values of the explanatory variables, summarized by the row vector $\mathbf{x}_0$*.

- use the estimated response for the point prediction
- take into account not only error in the estimate of the mean response, but also for the prediction error associated with any variables or factors that have not been measured (i.e. the noise ε in the new response).

The variance of the prediction error is

$$\text{Var}(\mathbf{x}_0\widehat{\beta} + \varepsilon) = \underset{\text{error in mean estimate}}{\mathbf{x}_0(\mathrm{U}^{-1}\mathrm{U}^{-\top})\mathbf{x}_0^\top\sigma^2} + \underset{\text{noise in new response}}{\sigma^2}.$$

The standard deviation of the prediction error is thus

$$\sigma\sqrt{\mathbf{x}_0(\mathrm{U}^{-1}\mathrm{U}^{-\top})\mathbf{x}_0^\top + 1}.$$

*which is of length p and whose first component must be ‘1’, if an intercept has been included in the model

Estimating the mean response at $\mathbf{x}_0$ - Example

Estimate the mean sale price for a house with characteristics summarized in a 9-vector called `x0`:

```
##    x1   x2   x3   x4   x5   x6   x7   x8   x9
##   4.5  1.0  2.3  1.2  1.0  6.0  3.0 40.0  0.0
```

```
x0 <- c(1, x0)
Ey.hat <- x0%*%betahat  # estimated price in thousands
```

Estimated mean sale price:

```
1000*Ey.hat


##          [,1]
## [1,] 28050.18
```

Prediction error - Example

For a single house price, the standard error of this estimate as a prediction is

```
UTx0T <- solve(t(U), x0)
SEPred<- sqrt(MSE)*sqrt(1+ sum(UTx0T^2))
SEPred

##          [,1]
## [1,] 3.301332
```

We can multiply this value by $1000 to get the standard error of prediction in actual dollars:

```
1000*SEPred

##          [,1]
## [1,] 3301.332
```

Prediction interval for a new response

If ϵ is normally distributed,

Since $\widehat{\beta}$ and **MSE** $=$ **SSE**$/(n-p)$ are independent of each other. SE of the mean response ;

$$\frac{\mathbf{x}_0\widehat{\beta} - \mathbf{x}_0\beta}{\sqrt{\textbf{MSE}}\sqrt{1+\mathbf{x}_0(\mathrm{U}^{-1}\mathrm{U}^{-\mathsf{T}})\mathbf{x}_0^{\mathsf{T}}}}$$

must have a t distribution on $n-p$ degrees of freedom,

Therefore, A $1-\alpha$ prediction interval for $E[y|\mathbf{x}_0] = \mathbf{x}_0\beta$ is then

$$\mathbf{x}_0\widehat{\beta} \pm t_{n-p,\alpha/2} SE_{meanrepsonse}$$

$$= \mathbf{x}_0\widehat{\beta} \pm t_{n-p,\alpha/2}\sqrt{\textbf{MSE}}\sqrt{1+\mathbf{x}_0(\mathrm{U}^{-1}\mathrm{U}^{-\mathsf{T}})\mathbf{x}_0^{\mathsf{T}}}.$$

This interval should contain the new response $Y = \mathbf{x}_0\beta + \varepsilon$ with probability $1-\alpha$.

Prediction interval for a new response - Example

A 95% prediction interval for the house price for a single house having the characteristics described in x_0 is

```r
PI <- c(Ey.hat - qt(.975, df = n - p)*SEPred, Ey.hat +
    qt(.975, df = n - p)*SEPred)
PI

## [1] 20.96953 35.13083
```

In dollars, this is

```r
1000*PI

## [1] 20969.53 35130.83
```

Inference

Inference for the regression coefficients

It can also be deduced that $\widehat{\beta}$ is normally distributed with mean β and variance-covariance matrix $(\mathrm{U}^{-1}\mathrm{U}^{-\intercal})\sigma^2$.

Thus, each component of $\widehat{\beta}$ can be standardized as follows:

$$\frac{\widehat{\beta}_i - \beta_i}{\sigma\sqrt{(\mathrm{U}^{-1}\mathrm{U}^{-\intercal})_{i+1,i+1}}}$$

to become a standard normal random variable, for $i = 0, 1, 2, \ldots, p-1$.

It then follows, by the same reasoning as before, that

$$\frac{\widehat{\beta}_i - \beta_i}{\widehat{\sigma}\sqrt{(\mathrm{U}^{-1}\mathrm{U}^{-\intercal})_{i+1,i+1}}}$$

must have a t-distribution on $n-p$ degrees of freedom.

This result can be used to justify t-tests and related confidence intervals for individual regression coefficients.

A confidence interval for a regression coefficient

Let c_{jj} be the $j+1$th diagonal element of $(\mathrm{U}^\intercal\mathrm{U})^{-1}$. Then a $(1-\alpha)$ confidence interval for β_j is

$$\widehat{\beta}_j \pm t_{n-p,\alpha/2}\widehat{\sigma}\sqrt{c_{jj}} = \widehat{\beta}_j \pm t_{n-p,\alpha/2} SE_j$$

where we have denoted the Standard Error of the jth coefficient estimate as SE_j.

A confidence interval for a regression coefficient - Example

For the house price data, the coefficient standard errors can be computed using

```
Cii <- sqrt(diag(solve(t(U)%*%U)))
SEii <- Cii*as.numeric(sqrt(MSE))
SEii
```

```
##      X1      x1      x2      x3      x4      x5
## 5.91285 1.02990 4.30037 0.49039 4.35955 1.37351
##      x6      x7      x8      x9
## 2.37854 3.39554 0.06672 1.93750
```

Compare with the standard errors obtained from `summary(y.lm)`**.**

A 95% confidence interval for the 4th regression coefficient (β_3) is

```
betahat[4] + SEii[4]*qt(c(.025, .975), df = n-p)
```

```
## [1] -0.9026  1.2010
```

ADVICE package in R

You can also check ADVICE package in R which that summerize discussed matters and provides:

Accurate point and interval estimation methods for multiple linear regression coefficients, under classical normal and independent error assumptions, taking into account variable selection.

Variable Selection Methods

Criteria for selecting a model

- The selected model would be the model with the smallest residual sum of squares or largest likelihood.

- The more parameters allowed in the model, the more closely it will fit the response $\mathbf{Y}$.

- Too many parameters can result in over-fitting.

Question: Which of the covariates needs to be chosen in order to have the best fit?

Variable selection-traditional methods

- **Backward Selection**: This method begins with the fitting of a model which includes all given covariates. The variable with the largest p-value is removed, and the model with all remaining variables is re-fit. Variables are removed sequentially in this manner, until all remaining variables have satisfactorily small p-values.

- **Forward Selection**: This method builds models, adding terms sequentially, beginning with the one involving the variable giving the smallest residual sum of squares. At each stage, the variable added is the one that reduces the sum of squares most.

- **Stepwise Selection**: This method usually starts as in Backward Selection, but variables can be added back in to the model at later stages if a large enough reduction in residual sum of squares (or some other criterion) can be achieved.

- **All Subsets Regression**: This method involves the computation of all possible regression models. The one which is closest to meeting a particular criterion is chosen.

Modern approaches to variable selection — LASSO

A more effective way to select variables is the LASSO (Tibshirani, 1996)

Specifically, the estimate of the $k + 1$ element vector β is chosen to minimize

$$||y - X\beta||^2 + \lambda^2 \sum_{j=0}^{k} |\beta_j|.$$

- When $\lambda = 0$, this reduces to the ordinary least-squares problem.
- When λ is nonzero, this is a nonlinear optimization problem
- the LASSO optimization has the property that for large enough λ^2, at least one of the coefficient estimates becomes identically 0.
- As λ^2 increases, it can be shown that the coefficients all shrink toward 0, and ultimately, all become identically 0.

Ridge regression

An easier optimization problem would have been

$$||y - X\beta||^2 + \lambda^2 \sum_{j=0}^{k} |\beta_j|^2.$$

As λ^2 increases, the coefficients shrink to 0, but they do not become identically 0 in the same way as for the LASSO.

A direct method, based on QR

- If U happened to be a diagonal matrix, then the coefficient estimates would be uncorrelated with each other, and individual t-tests could be carried out, subject to appropriate multiple testing criteria.

- In general, U is not a diagonal matrix, so the t-tests are not simply compromised by multiple testing issues, but by the intercorrelations between the estimates themselves, a phenomenon referred to as multicollinearity and variance inflation.

  This problem only gets worse as the number of terms in a regression model increases.

A QR pre-test estimate for β

The elements of $\mathrm{Q}_1^\intercal \mathbf{Y}$ are uncorrelated, with mean $\mathrm{U}\beta$ and variance σ^2, since

$$\mathrm{Q}_1^\intercal \mathbf{Y} = \mathrm{U}\beta + \mathrm{Q}_1^\intercal \epsilon$$

The elements of $\mathrm{Q}_2^\intercal \mathbf{Y}$ are also uncorrelated, but with mean 0 and variance σ^2.

$$\mathrm{Q}_2^\intercal \mathbf{Y} = \mathrm{Q}_2^\intercal \epsilon$$

The elements of $\mathrm{Q}_1^\intercal \mathbf{Y}$ and $\mathrm{Q}_2^\intercal \mathbf{Y}$ are uncorrelated.

A QR pre-test estimate for β

If some covariates do not belong in the model, that is, some elements of $Q_1^\top \mathbf{Y}$ corresponding to elements of $U\beta$ are 0, then

- those elements of $\mathrm{Q}_1^\top \mathbf{Y}$ will have the same distribution as the elements of $\mathrm{Q}_2^\top \mathbf{Y}$ whose means are 0.

- Other elements of $\mathrm{Q}_1^\top \mathbf{Y}$ will necessarily follow a different distribution (i.e. having a different mean).

- **outliers on a QQ-plot** of $\mathrm{Q}_1^\top \mathbf{Y}$ against $\mathrm{Q}_2^\top \mathbf{Y}$ should identify elements of $\mathrm{U}\beta$ which are nonzero.

Mallow’s C_p

Mallows’ C_p also can be used to determine which regression model is best among multiple candidates.

$$\mathbf{C}_p = \frac{(y - \mathbf{X}\widehat{\beta})^T(y - \mathbf{X}\widehat{\beta})}{\sigma^2} + 2p - n$$

where p is the number of model parameters.

The model with the lowest C_p value that is close to $p + 1$ is the best fit from this data.

Mallow's C_p

You can calculate Mallows' Cp in R by using the ***ols_mallows_cp()*** function from the ***olsrr*** package.

Where you compare a candiate with selected p variables with a full model (includes all variables

- Every potential model that has a high Mallows' Cp value is likely to be missing important predictor variables.

- Use the model with the lowest Mallow's Cp value if several possible models have low values.

PRESS — special case of cross-validation

- **PRESS residuals:**

$$e_{(i)} = y_i - \widehat{y}_{(i)}.$$

Here y_i is the ith observed response and $\widehat{y}_{(i)}$ is the predicted value at the ith observation based on the regression of y against the xs, omitting the ith observation.

- **PRedicted Error Sum of Squares:**

$$\textbf{PRESS} = \sum_{i=1}^{n} e_{(i)}^2$$

$$= \sum_{i=1}^{n} \left(\frac{e_i}{1 - h_{ii}} \right)^2$$

This gives an idea of how well a regression model can predict new data. Small values of PRESS are desired.

Summary

- Matrix form for multiple regression models
- used QR decompostion methods to obtained estimated for parameters in linear regression
- used QR decompostion methods to build confidence interval for prediction
- Discussed some variable selection methods