Regression Analysis

MIT 18.642

Dr. Kempthorne

Fall 2024

Multiple Linear Regression: Setup

Data Set

- n cases $i = 1, 2, \ldots, n$
- 1 Response (dependent) variable
$$y_i,\ i = 1, 2, \ldots, n$$
- p Explanatory (independent) variables
$$\mathbf{x}_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,p})^T,\ i = 1, 2, \ldots, n$$

Goal of Regression Analysis:

- Extract/exploit relationship between y_i and $\mathbf{x}_i$.

Examples

- Prediction
- Causal Inference
- Approximation
- Functional Relationships

General Linear Model: For each case i, the conditional distribution $[y_i \mid x_i]$ is given by

$$y_i = \hat{y}_i + \epsilon_i$$

where

- $\hat{y}_i = \beta_1 x_{i,1} + \beta_2 x_{i,2} + \cdots + \beta_{i,p} x_{i,p}$
- $\beta = (\beta_1, \beta_2, \ldots, \beta_p)^T$ are p regression parameters (constant over all cases)
- ϵ_i Residual (error) variable (varies over all cases)

Extensive breadth of possible models

- Polynomial approximation ($x_{i,j} = (x_i)^j$, explanatory variables are different powers of the same variable $x = x_i$)
- Fourier Series: ($x_{i,j} = sin(jx_i)$ or $cos(jx_i)$, explanatory variables are different sin/cos terms of a Fourier series expansion)
- Time series regressions: time indexed by i, and explanatory variables include lagged response values.

Note: *Linearity* of $\hat{y}_i$ (in regression parameters) maintained with non-linear x.

Steps for Fitting a Model

(1) Propose a model in terms of

- Response variable Y (specify the scale)
- Explanatory variables $X_1, X_2, \ldots X_p$ (include different functions of explanatory variables if appropriate)
- Assumptions about the distribution of ϵ over the cases

(2) Specify/define a criterion for judging different estimators.

(3) Characterize the best estimator and apply it to the given data.

(4) Check the assumptions in (1).

(5) If necessary modify model and/or assumptions and go to (1).

Specifying Assumptions in (1) for Residual Distribution

- Gauss-Markov: zero mean, constant variance, uncorrelated
- Normal-linear models: ϵ_i are i.i.d. $N(0, \sigma^2)$ r.v.s
- Generalized Gauss-Markov: zero mean, and general covariance matrix (possibly correlated,possibly heteroscedastic)
- Non-normal/non-Gaussian distributions (e.g., Laplace, Pareto, Contaminated normal: some fraction $(1 - \delta)$ of the ϵ_i are i.i.d. $N(0, \sigma^2)$ r.v.s the remaining fraction (δ) follows some contamination distribution).

Specifying Estimator Criterion in (2)

- Least Squares
- Maximum Likelihood
- Robust (Contamination-resistant)
- Bayes (assume β_j are r.v.'s with known *prior* distribution)
- Accommodating incomplete/missing data

Case Analyses for (4) Checking Assumptions

- Residual analysis
  - Model errors ϵ_i are unobservable
  - Model residuals for fitted regression parameters $\tilde{\beta}_j$ are:
$$e_i = y_i - [\tilde{\beta}_1 x_{i,1} + \tilde{\beta}_2 x_{i,2} + \cdots + \tilde{\beta}_p x_{i,p}]$$
- Influence diagnostics (identify cases which are highly 'influential'?)
- Outlier detection

Ordinary Least Squares Estimates

Least Squares Criterion: For $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_p)^T$, define

$$\begin{aligned} Q(\boldsymbol{\beta}) &= \sum_{i=1}^{N} [y_i - \hat{y}_i]^2 \\ &= \sum_{i=1}^{N} [y_i - (\beta_1 x_{i,1} + \beta_2 x_{i,2} + \cdots + \beta_{i,p} x_{i,p})]^2 \end{aligned}$$

Ordinary Least-Squares (OLS) estimate $\hat{\boldsymbol{\beta}}$: minimizes $Q(\boldsymbol{\beta})$.

Matrix Notation

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \quad \mathbf{X} = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,p} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{p,n} \end{bmatrix} \quad \boldsymbol{\beta} = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}$$

Solving for OLS Estimate $\hat{\boldsymbol{\beta}}$

$$\hat{\mathbf{y}} = \begin{pmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_n \end{pmatrix} = \mathbf{X}\boldsymbol{\beta} \text{ and}$$

$$\begin{aligned} Q(\boldsymbol{\beta}) &= \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = (\mathbf{y} - \hat{\mathbf{y}})^T(\mathbf{y} - \hat{\mathbf{y}}) \\ &= (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \end{aligned}$$

OLS $\hat{\boldsymbol{\beta}}$ solves $\frac{\partial Q(\boldsymbol{\beta})}{\partial \beta_j} = 0, \quad j = 1, 2, \ldots, p$

$$\begin{aligned} \frac{\partial Q(\boldsymbol{\beta})}{\partial \beta_j} &= \frac{\partial}{\partial \beta_j}\left(\sum_{i=1}^{n}[y_i - (x_{i,1}\beta_1 + x_{i,2}\beta_2 + \cdots x_{i,p}\beta_p)]^2\right) \\ &= \sum_{i=1}^{n} 2(-x_{i,j})[y_i - (x_{i,1}\beta_1 + x_{i,2}\beta_2 + \cdots x_{i,p}\beta_p)] \\ &= -2(\mathbf{X}_{[j]})^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \quad \text{where } \mathbf{X}_{[j]} \text{ is the } j\text{th column of } \mathbf{X} \end{aligned}$$

Solving for OLS Estimate $\hat{\boldsymbol{\beta}}$

$$\frac{\partial Q}{\partial \boldsymbol{\beta}} = \begin{bmatrix} \frac{\partial Q}{\partial \beta_1} \\ \frac{\partial Q}{\partial \beta_2} \\ \vdots \\ \frac{\partial Q}{\partial \beta_p} \end{bmatrix} = -2 \begin{bmatrix} \mathbf{X}_{[1]}^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \\ \mathbf{X}_{[2]}^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \\ \vdots \\ \mathbf{X}_{[p]}^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \end{bmatrix} = -2\mathbf{X}^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$$

So the OLS Estimate $\hat{\boldsymbol{\beta}}$ solves the **"Normal Equations"**

$$\begin{aligned} & \mathbf{X}^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) = \mathbf{0} \\ \iff \quad & \mathbf{X}^T\mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}^T\mathbf{y} \\ \implies \quad & \hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \end{aligned}$$

N.B. For $\hat{\boldsymbol{\beta}}$ to exist (uniquely)

$(\mathbf{X}^T\mathbf{X})$ must be invertible

$\iff$ $\mathbf{X}$ must have Full Column Rank

(Ordinary) Least Squares Fit

OLS Estimate:

$$\hat{\boldsymbol{\beta}} = \begin{pmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \\ \vdots \\ \hat{\beta}_p \end{pmatrix} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

Fitted Values:

$$\hat{\mathbf{y}} = \begin{pmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_n \end{pmatrix} = \begin{pmatrix} x_{1,1}\hat{\beta}_1 + \cdots + x_{1,p}\hat{\beta}_p \\ x_{2,1}\hat{\beta}_1 + \cdots + x_{2,p}\hat{\beta}_p \\ \vdots \\ x_{n,1}\hat{\beta}_1 + \cdots + x_{n,p}\hat{\beta}_p \end{pmatrix}$$

$$= \mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} = \mathbf{H}\mathbf{y}$$

Where $\mathbf{H} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ is the $n \times n$ "Hat Matrix"

(Ordinary) Least Squares Fit

The Hat Matrix $\mathbf{H}$ projects R^n onto the column-space of $\mathbf{X}$

Residuals: $\hat{\epsilon}_i = y_i - \hat{y}_i,\ i = 1, 2, \ldots, n$

$$\hat{\boldsymbol{\epsilon}} = \begin{pmatrix} \hat{\epsilon}_1 \\ \hat{\epsilon}_2 \\ \vdots \\ \hat{\epsilon}_n \end{pmatrix} = \mathbf{y} - \hat{\mathbf{y}} = (\mathrm{I}_\mathbf{n} - \mathbf{H})\mathbf{y}$$

Normal Equations: $\mathbf{X}^T(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}) = \mathbf{X}^T\hat{\boldsymbol{\epsilon}} = \mathbf{0}_p = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}$

N.B. The Least-Squares Residuals vector $\hat{\boldsymbol{\epsilon}}$ is orthogonal to the column space of $\mathbf{X}$

$$\Longrightarrow \hat{\boldsymbol{\epsilon}} \perp \hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}}$$

Normal Linear Regression Models

Probability Model:

$$\begin{aligned} Y_i &= x_{i,1}\beta_1 + x_{i,2}\beta_2 + \cdots x_{i,p}\beta_p + \epsilon_i \\ &= \mu_i + \epsilon_i \end{aligned}$$

Assume $\{\epsilon_1, \epsilon_2, \ldots, \epsilon_n\}$ are i.i.d $N(0, \sigma^2)$.

$$\Longrightarrow [Y_i \mid x_{i,1}, x_{i,2}, \ldots, x_{i,p}, \boldsymbol{\beta}, \sigma^2] \sim N(\mu_i, \sigma^2),$$

independent over $i = 1, 2, \ldots n$.

Conditioning on $\mathbf{X}$, $\boldsymbol{\beta}$, and σ^2

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \text{ where } \boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix} \sim N_n(\mathbf{O}_n, \sigma^2 \mathbf{I}_n)$$

Mean Vector and Covariance Matrix

$$\boldsymbol{\mu} = \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_n \end{pmatrix} = E(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2) = \mathbf{X}\boldsymbol{\beta}$$

$$\boldsymbol{\Sigma} = Cov(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2) = \begin{bmatrix} \sigma^2 & 0 & 0 & \cdots & 0 \\ 0 & \sigma^2 & 0 & \cdots & 0 \\ 0 & 0 & \sigma^2 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & & \sigma^2 \end{bmatrix} = \sigma^2 \mathbf{I_n}$$

That is, $\boldsymbol{\Sigma}_{i,j} = Cov(Y_i, Y_j \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2) = \sigma^2 \times \delta_{i,j}$.

$$\text{where } \delta_{i,j} = \begin{cases} 1 & if \quad i = j \\ 0 & if \quad i \neq j \end{cases}$$

Multivariate Gaussian Distributions

Apply Moment-Generating Functions (MGFs) to derive

Joint distribution of $\mathbf{Y} = (Y_1, Y_2, \ldots, Y_n)^T$

Joint distribution of $\hat{\boldsymbol{\beta}} = (\hat{\beta}_1, \hat{\beta}_2, \ldots, \hat{\beta}_p)^T$.

MGF of Y

For the n-variate r.v. $\mathbf{Y}$, and constant $n-$vector $\mathbf{t} = (t_1, \ldots, t_n)^T$,

$$\begin{aligned} M_{\mathbf{Y}}(\mathbf{t}) &= E(e^{\mathbf{t}^T\mathbf{Y}}) = E(e^{t_1Y_1+t_2Y_2+\cdots t_nY_n}) \\ &= E(e^{t_1Y_1}) \cdot E(e^{t_2Y_2}) \cdots E(e^{t_nY_n}) \\ &= M_{Y_1}(t_1) \cdot M_{Y_2}(t_2) \cdots M_{Y_n}(t_n) \\ &= \prod_{i=1}^n e^{t_i\mu_i+\frac{1}{2}t_i^2\sigma^2} \\ &= e^{\sum_{i=1}^n t_i\mu_i+\frac{1}{2}\sum_{i,k=1}^n t_i\boldsymbol{\Sigma}_{i,k}t_k} = e^{\mathbf{t}^T\mathbf{u}+\frac{1}{2}\mathbf{t}^T\boldsymbol{\Sigma}\mathbf{t}} \end{aligned}$$

$\Longrightarrow \mathbf{Y} \sim N_n(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

Multivariate Normal with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$

Multivariate Gaussian Distributions

MGF of $\hat{\beta}$

For the p-variate r.v. $\hat{\boldsymbol{\beta}}$, and constant $p-$vector $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_p)^T$,

$$M_{\hat{\boldsymbol{\beta}}}(\boldsymbol{\tau}) = E(e^{\boldsymbol{\tau}^T \hat{\boldsymbol{\beta}}}) = E(e^{\tau_1 \hat{\beta}_1 + \tau_2 \hat{\beta}_2 + \cdots \tau_p \beta_p})$$

Defining $\mathbf{A} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ we can express

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T y = \mathbf{AY}$$

and

$$\begin{aligned} M_{\hat{\boldsymbol{\beta}}}(\boldsymbol{\tau}) &= E(e^{\boldsymbol{\tau}^T \hat{\boldsymbol{\beta}}}) \\ &= E(e^{\boldsymbol{\tau}^T \mathbf{AY}}) \\ &= E(e^{\mathbf{t}^T \mathbf{Y}}), \text{ with } \mathbf{t} = \mathbf{A}^T \boldsymbol{\tau} \\ &= M_{\mathbf{Y}}(\mathbf{t}) \\ &= e^{\mathbf{t}^T \mathbf{u} + \frac{1}{2}\mathbf{t}^T \boldsymbol{\Sigma} \mathbf{t}} \end{aligned}$$

MGF of $\hat{\boldsymbol{\beta}}$

For

$$\begin{aligned} M_{\hat{\boldsymbol{\beta}}}(\boldsymbol{\tau}) &= E(e^{\boldsymbol{\tau}^T \hat{\boldsymbol{\beta}}}) \\ &= e^{\mathbf{t}^T \mathbf{u} + \frac{1}{2}\mathbf{t}^T \boldsymbol{\Sigma} \mathbf{t}} \end{aligned}$$

Plug in:

$$\begin{aligned} \mathbf{t} &= \mathbf{A}^T \boldsymbol{\tau} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\boldsymbol{\tau} \\ \boldsymbol{\mu} &= \mathbf{X}\boldsymbol{\beta} \\ \boldsymbol{\Sigma} &= \sigma^2 \mathbf{I}_n \end{aligned}$$

Gives:

$$\begin{aligned} \mathbf{t}^T\boldsymbol{\mu} &= \boldsymbol{\tau}^T\boldsymbol{\beta} \\ \mathbf{t}^T\boldsymbol{\Sigma}\mathbf{t} &= \boldsymbol{\tau}^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T[\sigma^2\mathbf{I}_n]\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\boldsymbol{\tau} \\ &= \boldsymbol{\tau}^T[\sigma^2(\mathbf{X}^T\mathbf{X})^{-1}]\boldsymbol{\tau} \end{aligned}$$

So the MGF of $\hat{\boldsymbol{\beta}}$ is

$$\begin{aligned} M_{\hat{\boldsymbol{\beta}}}(\boldsymbol{\tau}) &= e^{\boldsymbol{\tau}^T\boldsymbol{\beta} + \frac{1}{2}\boldsymbol{\tau}^T[\sigma^2(\mathbf{X}^T\mathbf{X})^{-1}]\boldsymbol{\tau}} \\ \Longleftrightarrow \qquad \hat{\boldsymbol{\beta}} &\sim N_p(\boldsymbol{\beta}, \sigma^2(\mathbf{X}^T\mathbf{X})^{-1}) \end{aligned}$$

Marginal Distributions of Least Squares Estimates

Marginal distribution of each $\hat{\beta}_j$

$$\hat{\beta}_j \sim N(\beta_j, \sigma^2 C_{j,j})$$

where $C_{j,j} = j$th diagonal element of $(\mathbf{X}^T\mathbf{X})^{-1}$

Proof: For $j = 1$, compute the MGF of $\hat{\beta}_1$
by setting $\boldsymbol{\tau} = t(1, 0, \ldots, 1)$ in the MGF of $\hat{\boldsymbol{\beta}}$
$\hat{\boldsymbol{\beta}} \sim N_p(\boldsymbol{\beta}, \sigma^2(\mathbf{X}^T\mathbf{X})^{-1})$

The MGF of $\hat{\boldsymbol{\beta}}$ is

$$\begin{aligned} M_{\hat{\boldsymbol{\beta}}_j}(t) = M_{\hat{\boldsymbol{\beta}}}(\boldsymbol{\tau}) &= e^{\boldsymbol{\tau}^T\boldsymbol{\beta} + \frac{1}{2}\boldsymbol{\tau}^T[\sigma^2(\mathbf{X}^T\mathbf{X})^{-1}]\boldsymbol{\tau}} \\ &= e^{t\beta_1 + \frac{t^2}{2}\sigma^2[(\mathbf{X}^T\mathbf{X})^{-1}]_{1,1}} \end{aligned}$$

More Distribution Theory

Assume $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$, where $\{\epsilon_i\}$ are i.i.d. $N(0, \sigma^2)$, i.e.,

$$\begin{aligned} \boldsymbol{\epsilon} &\sim N_n(\mathbf{0}_n, \sigma^2 \mathrm{I}_\mathbf{n}) \\ \text{or } \mathbf{y} &\sim N_n(\mathbf{X}\boldsymbol{\beta}, \sigma^2 \mathrm{I}_\mathbf{n}) \end{aligned}$$

Theorem* For any $(m \times n)$ matrix $\mathbf{A}$ of rank $m \leq n$, the random normal vector $\mathbf{y}$ transformed by $\mathbf{A}$,

$$\mathbf{z} = \mathbf{A}\mathbf{y}$$

is also a random normal vector:

$$\mathbf{z} \sim N_m(\boldsymbol{\mu}_\mathbf{z}, \boldsymbol{\Sigma}_\mathbf{z})$$

where $\boldsymbol{\mu}_\mathbf{z} = \mathbf{A}E(\mathbf{y}) = \mathbf{A}\mathbf{X}\boldsymbol{\beta}$,

and $\boldsymbol{\Sigma}_\mathbf{z} = \mathbf{A}Cov(\mathbf{y})\mathbf{A}^T = \sigma^2\mathbf{A}\mathbf{A}^T$.

Above, $\mathbf{A} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ yields the distribution of $\hat{\boldsymbol{\beta}} = \mathbf{A}\mathbf{y}$

Different definitions of $\mathbf{A}$ (and $\mathbf{z}$) give easy proofs of:

More Distribution Theory

Theorem For the normal linear regression model

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon},$$

where

$\mathbf{X}$ $(n \times p)$ has rank p and

$\boldsymbol{\epsilon} \sim N_n(\mathbf{0}_n, \sigma^2 \mathrm{I}_\mathbf{n})$.

(a) $\hat{\boldsymbol{\beta}} = [(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T]\mathbf{y} \sim N_p(\boldsymbol{\beta}, \sigma^2(\mathbf{X}^T\mathbf{X})^{-1})$

(b) $\hat{\boldsymbol{\epsilon}} = [\mathbf{I}_n - \mathbf{H}]\mathbf{y} \sim N_n(\mathbf{0}, \sigma^2(\mathbf{I}_n - \mathbf{H}))$

(c) $\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$ and $\hat{\boldsymbol{\epsilon}} = \mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}$ are independent r.v.s

$$A = \begin{bmatrix} (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T \\ \mathbf{I}_n - \mathbf{H} \end{bmatrix}$$ and note that

Joint MGF equals Product of MGFs

(proving independence, like pdfs)

Estimating σ^2

$$\begin{aligned} E[\textstyle\sum_{i=1}^n \hat{\epsilon}_i^2] &= E[\hat{\boldsymbol{\epsilon}}^T\hat{\boldsymbol{\epsilon}}] = trace[Cov(\hat{\boldsymbol{\epsilon}})] \\ &= trace[\sigma^2(\mathbf{I}_n - \mathbf{H})] = \sigma^2(n - trace[\mathbf{H}[) \\ &= (n-p)\sigma^2 \\ \Longrightarrow \quad \hat{\sigma}^2 &= (\textstyle\sum_{i=1}^n \hat{\epsilon}_i^2)/(n-p) \text{ is unbiased} \end{aligned}$$

More Distribution Theory

t-Distributions for Standardized Estimates

For each $j = 1, 2, \ldots, p$

$$\hat{t}_j = \frac{\hat{\beta}_j - \beta_j}{\hat{\sigma} C_{j,j}} \sim t_{n-p} \ (t- \text{ distribution})$$

where $\hat{\sigma}^2 = \frac{1}{n-p} \sum_{i=1}^{n} \hat{\epsilon}_i^2$

$$C_{j,j} = [(\mathbf{X}^T\mathbf{X})^{-1}]_{j,j}$$

F-Test of $\beta_j = 0$, $j = k+1, \ldots, p$

Set: $RSS_1 = \hat{\epsilon}^T\hat{\epsilon}$ from full model
$(\hat{\beta} = [(\mathbf{X}^T\mathbf{X})^{-1}]\mathbf{X}^T\mathbf{Y}$ and $\hat{\epsilon} = \mathbf{y} - \mathbf{X}\hat{\beta})$

$RSS_0 = \hat{\epsilon}_0^T\hat{\epsilon}_0$ from sub model
(use only first k columns of $\mathbf{X}$)

Compute: $$F = \frac{(RSS_0 - RSS_1)/(p-k)}{RSS_1/(n-p)}$$

When *Null* model True: $F \sim F_{df_1, df_2}$ distribution with
$df_1 = (p-k)$ and $df_2 = (n-k)$

Data Set: prostate

```
> library(faraway)
> data("prostate")
```

Data Set: *prostate* in library *faraway*

A data frame with 97 observations on the following 10 variables.

Variable	Description
lcavol	log cancer volume
lweight	log prostate weight
age	in years
lbph	log of the amount of benign prostatic hyperplasia
svi	seminal vesicle invasion
lcp	log of capsular penetration
gleason	a numeric vector
pgg45	percent of Gleason score 4 or 5
lpsa	response
train	a logical vector

Objective: Predict response (*lpsa*) given covariates
(Use cases *train* == *TRUE*, numbering 72)

Prostate Data: Summary Statistics

```
> round(apply(prostate,2,summary),digits=3)
```

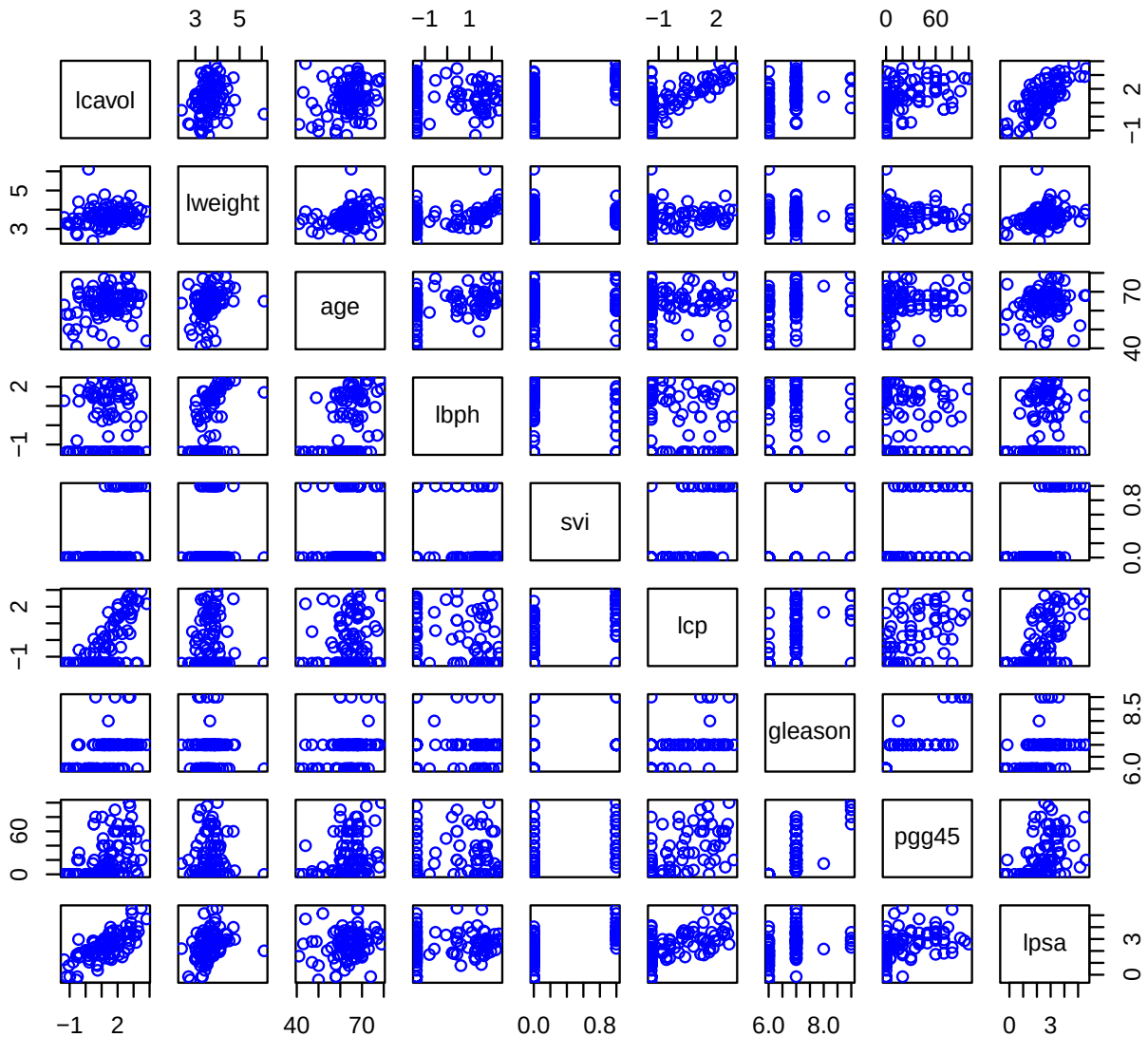
	lcavol	lweight	age	lbph	svi	lcp	gleason	pgg45	lpsa
Min.	-1.347	2.375	41.000	-1.386	0.000	-1.386	6.000	0.000	-0.431
1st Qu.	0.513	3.376	60.000	-1.386	0.000	-1.386	6.000	0.000	1.732
Median	1.447	3.623	65.000	0.300	0.000	-0.799	7.000	15.000	2.592
Mean	1.350	3.653	63.866	0.100	0.216	-0.179	6.753	24.381	2.478
3rd Qu.	2.127	3.878	68.000	1.558	0.000	1.179	7.000	40.000	3.056
Max.	3.821	6.108	79.000	2.326	1.000	2.904	9.000	100.000	5.583

```
> round(cor( prostate[,1:8]),digits=3)
```

	lcavol	lweight	age	lbph	svi	lcp	gleason	pgg45
lcavol	1.000	0.194	0.225	0.027	0.539	0.675	0.432	0.434
lweight	0.194	1.000	0.308	0.435	0.109	0.100	-0.001	0.051
age	0.225	0.308	1.000	0.350	0.118	0.128	0.269	0.276
lbph	0.027	0.435	0.350	1.000	-0.086	-0.007	0.078	0.078
svi	0.539	0.109	0.118	-0.086	1.000	0.673	0.320	0.458
lcp	0.675	0.100	0.128	-0.007	0.673	1.000	0.515	0.632
gleason	0.432	-0.001	0.269	0.078	0.320	0.515	1.000	0.752
pgg45	0.434	0.051	0.276	0.078	0.458	0.632	0.752	1.000

Prostate Data: Pairs Plot

```
> pairs( prostate[,1:9], col="blue" )
```



Prostate Data: Initial Regression

```
> library(tidyverse)
> set.seed(1) #  Make train_id reproducible
> prostate_id<-data.frame(irow=1:nrow(prostate))
> train_id<-slice_sample(prostate_id, prop=.75)
> train<- prostate[train_id$irow,]
> test  <- prostate[-train_id$irow,]
> summary(lm(lpsa ~ ., data=train))
Call:
lm(formula = lpsa ~ ., data = train)

Residuals:
     Min       1Q   Median       3Q      Max
-1.96222 -0.37818 -0.02121  0.42628  1.49784

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.501484   1.626900   0.308  0.75891
lcavol       0.543473   0.104384   5.206 2.25e-06 ***
lweight      0.360912   0.195254   1.848  0.06924 .
age         -0.009482   0.014579  -0.650  0.51779
lbph         0.081743   0.072298   1.131  0.26249
svi          0.876592   0.297203   2.949  0.00446 **
lcp         -0.042324   0.114845  -0.369  0.71371
gleason      0.053957   0.195489   0.276  0.78344
pgg45        0.003135   0.005241   0.598  0.55183
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7236 on 63 degrees of freedom
Multiple R-squared:  0.6298,        Adjusted R-squared:  0.5828
F-statistic:  13.4 on 8 and 63 DF,  p-value: 4.198e-11
```

Issue: scale of parameter estimates

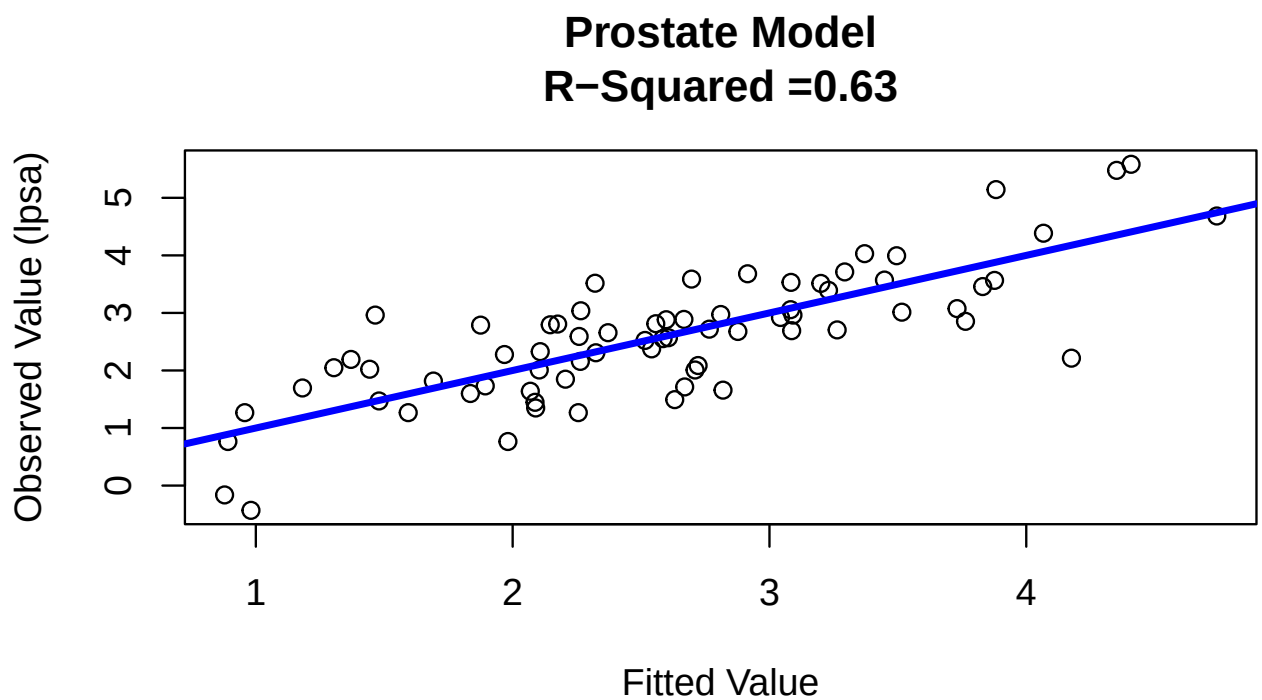
Regression on Standardized Covariates

```
> train0 <-train ; train0[,1:8]<-scale(train[,1:8])
> summary(lm(lpsa ~ ., data=train0))

Call:
lm(formula = lpsa ~ ., data = train0)

Residuals:
     Min       1Q   Median       3Q      Max
-1.96222 -0.37818 -0.02121  0.42628  1.49784

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.59101    0.08528  30.383  < 2e-16 ***
lcavol       0.60803    0.11678   5.206 2.25e-06 ***
lweight      0.18440    0.09976   1.848  0.06924 .
age         -0.06348    0.09760  -0.650  0.51779
lbph         0.11568    0.10231   1.131  0.26249
svi          0.34937    0.11845   2.949  0.00446 **
lcp         -0.05591    0.15172  -0.369  0.71371
gleason      0.03388    0.12273   0.276  0.78344
pgg45        0.08027    0.13418   0.598  0.55183
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7236 on 63 degrees of freedom
Multiple R-squared:  0.6298,        Adjusted R-squared:  0.5828
F-statistic:  13.4 on 8 and 63 DF,  p-value: 4.198e-11
```

Resolution: Estimate Scale in St Dev. Units of Covariate

R-Squared

```
> fit<-lm(lpsa~., data=train0); fit.summary<-summary(fit)
> names(fit.summary)
 [1] "call"          "terms"         "residuals"     "coefficients"
 [5] "aliased"       "sigma"         "df"            "r.squared"
 [9] "adj.r.squared" "fstatistic"    "cov.unscaled"
> plot(fit$fitted.values, train0$lpsa,
+      xlab="Fitted Value", ylab="Observed Value (lpsa)",
+      main=paste(c("Prostate Model","\nR-Squared =",
+         as.character(round(fit.summary$r.squared,digits=2))),
+         collapse=""))
> abline(lm(train0$lpsa ~ fit$fitted.values),col="blue",lwd=3)
```



Regression Diagnostics

R Functions (stats package)

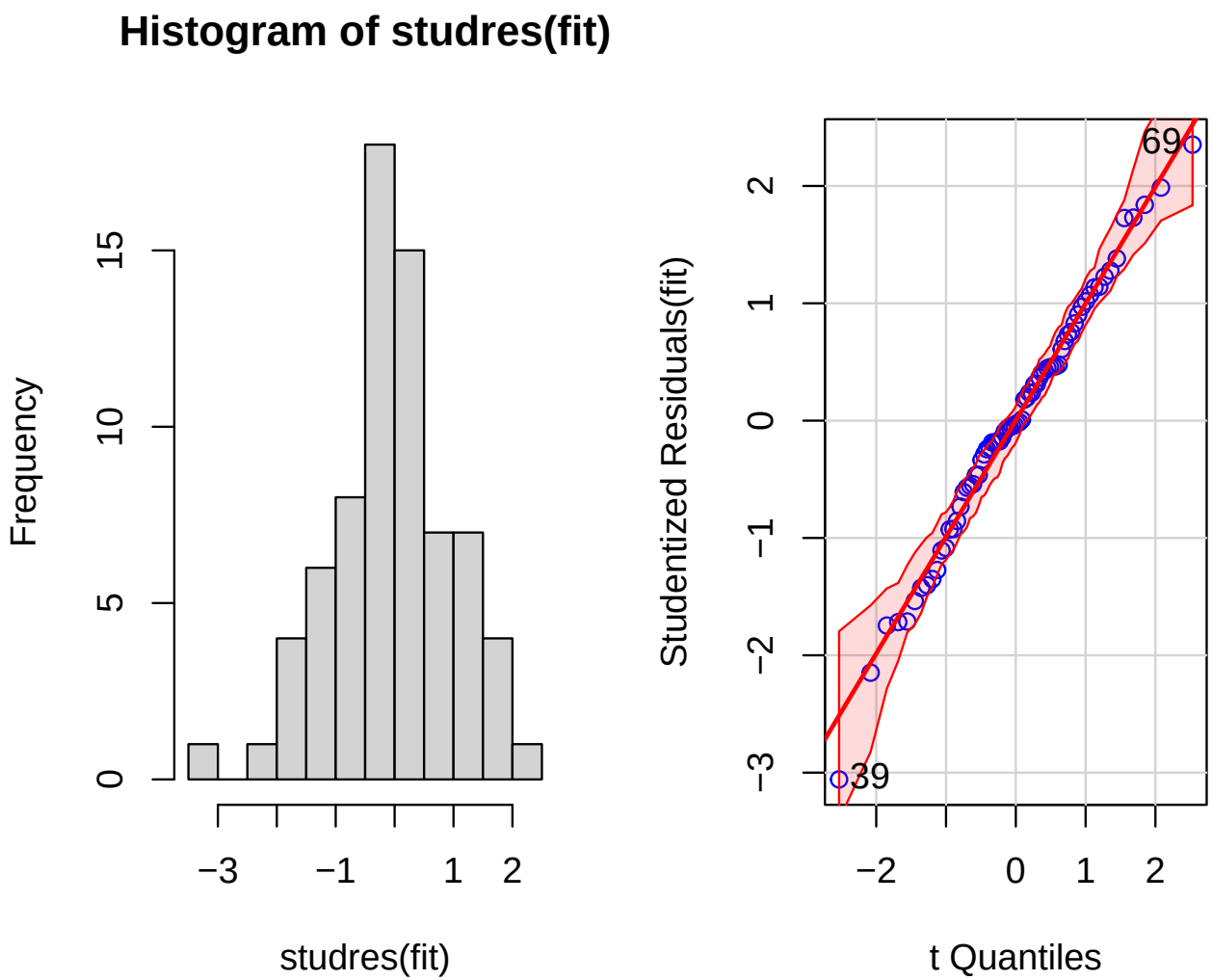
Function	Description
influence.measures(model)	High-level function; table of measures
plot(model)	Panel-plot of measures
rstudent(model, ...)	Studentized residuals*
dffits(model, infl = , res =)	Delta fitted Values*
dfbeta(model, ...)	Delta parameter estimates*
covratio(model,...)	Covariance matrix volume*
cooks.distance(model, ...)	Standarized delta parameter estimates*
hatvalues(model, ...)	Leverage values (diagonal of Hat matrix)
	* (Leave-one-out deletion diagnostics)

R Functions (car package)

Function	Description
qqPlot()	QQ Plot of studentized residuals
influencePlot()	Studentized residuals vs hat values
leveragePlots()	Added-variable plots
residualPlots()	Residual plots with curvature tests
durbinWatsonTest()	Residual autocorrelations and Generalized DW test

qqPlot() in Package car

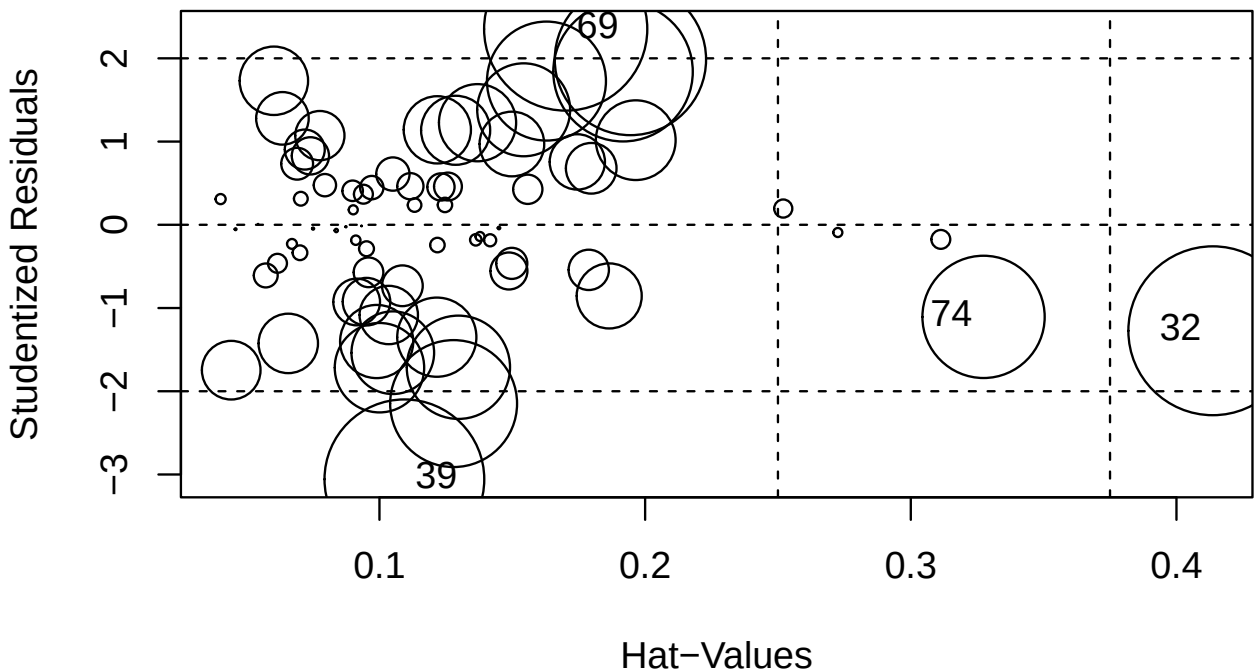
```
;
> par(mfcol=c(1,2));hist(studres(fit),class=25);qqPlot(fit,col="blue",col.lines="red")
39 69
 2 65
```



influencePlot() in Package car

```
> library(car);influencePlot(lm(lpsa~., data=train0))
```

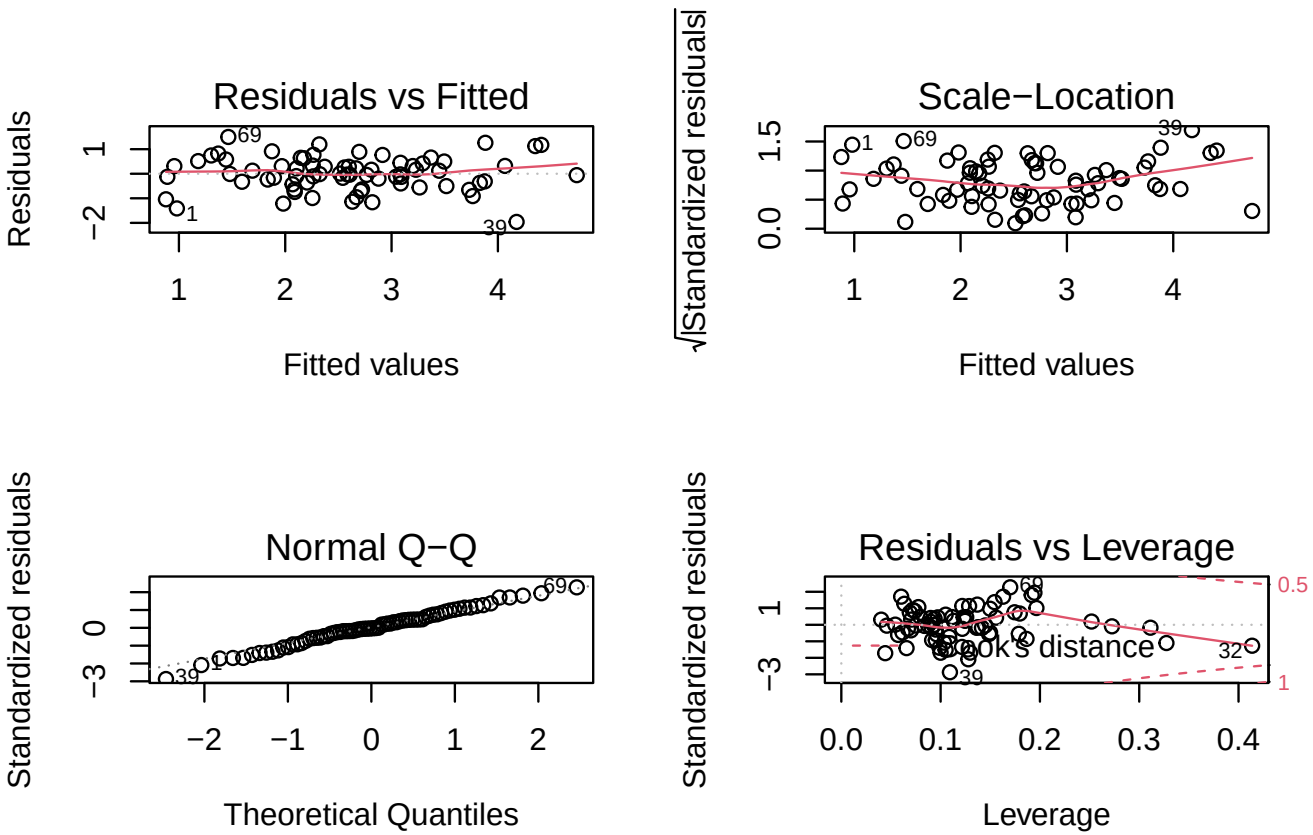
```
     StudRes       Hat      CookD
39 -3.057931 0.1093825 0.11267049
74 -1.106432 0.3273886 0.06597263
32 -1.273374 0.4136773 0.12587260
69  2.352556 0.1701000 0.11757924
```



Note: circle size proportional to Cook's Distance

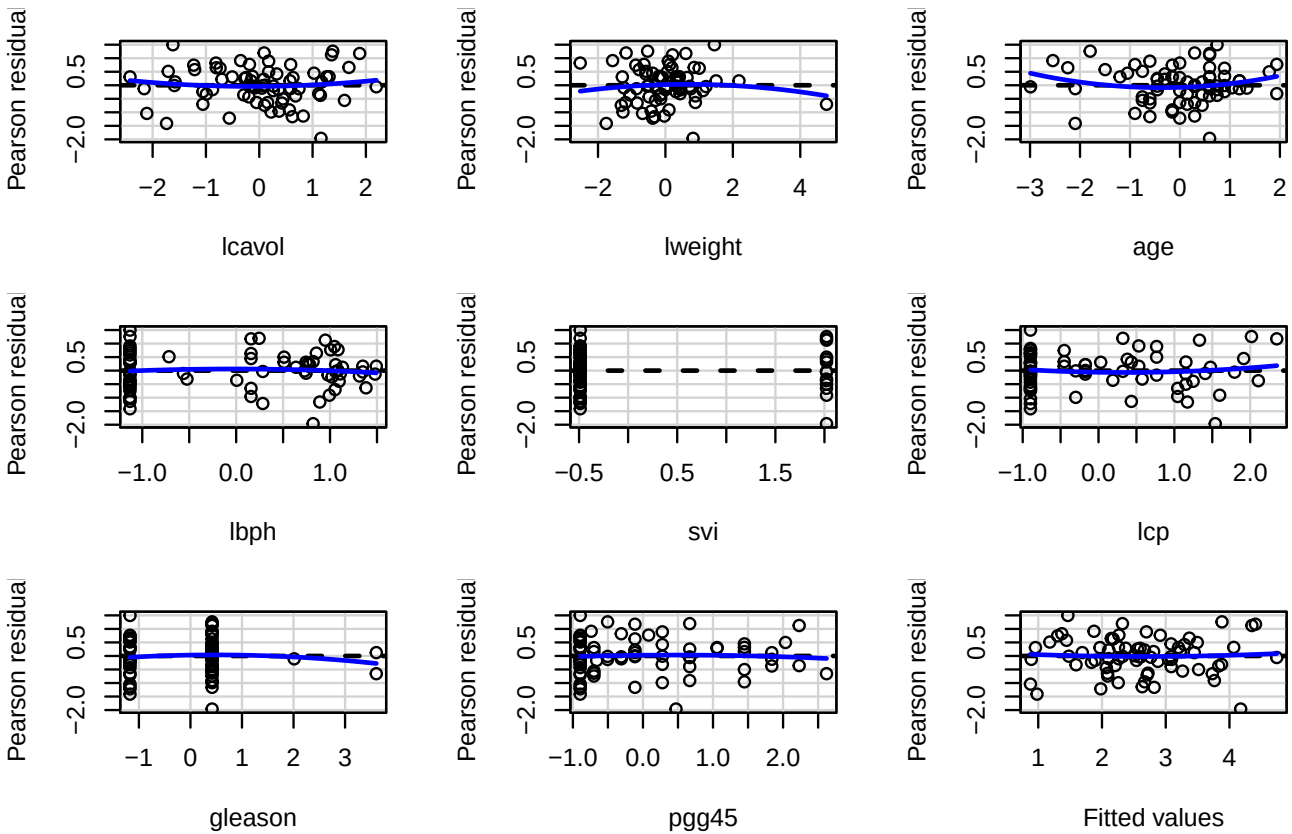
plot.lm() in STATS

```
> #  The R function $plot.lm()$ generates a 2x2 display of plots
> oldpar=par(no.readonly=TRUE)
> layout(matrix(c(1,2,3,4),2,2)) # optional 4 graphs/page
> plot(fit) ; par(oldpar,no.readonly=TRUE)
```



residualPlots() in Package car

```
> library(car) ; residualPlots(lm(lpsa~., data=train0),tests=FALSE)
```



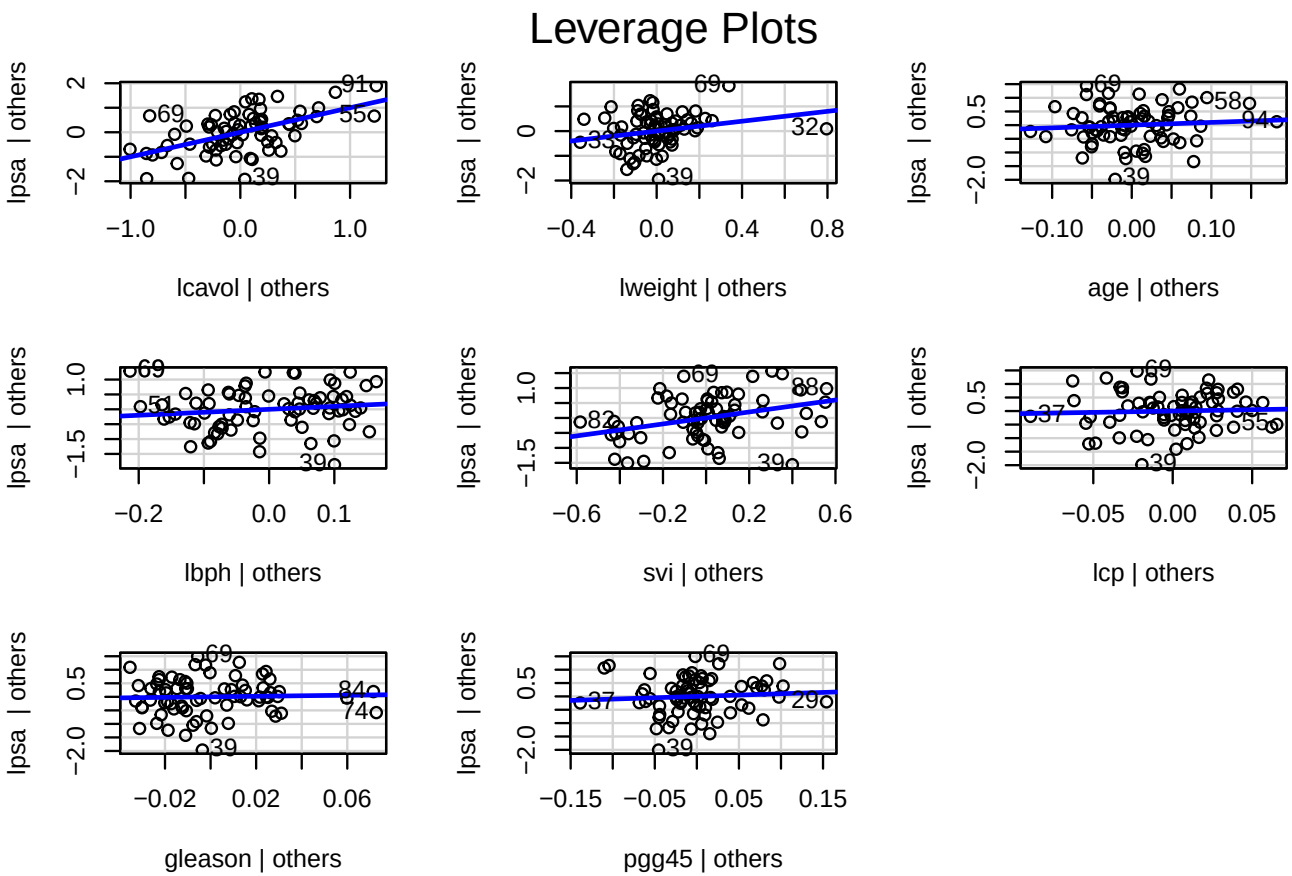
residualPlots() in Package car

```
> library(car)
> residualPlots(lm(lpsa~., data=train0),tests=TRUE,plot=FALSE)
           Test stat Pr(>|Test stat|)
lcavol        0.7273           0.46977
lweight      -0.7810           0.43775
age           1.3759           0.17380
lbph         -0.4307           0.66815
svi          -1.6716           0.09963 .
lcp           0.6207           0.53707
gleason      -0.7582           0.45123
pgg45        -0.3076           0.75938
Tukey test    0.3931           0.69424
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Test: curvature test for each plot
quadratic term added and tested whether coef is zero

leveragePlots() in Package car

```
> library(car) ; leveragePlots(lm(lpsa~., data=train0))
```



Gauss-Markov Assumptions (for Non-Gaussian Errors)

Data $\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$ and $\mathbf{X} = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,p} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{p,n} \end{bmatrix}$

follow a linear model satisfying the **Gauss-Markov Assumptions** if $\mathbf{y}$ is an observation of random vector $\mathbf{Y} = (Y_1, Y_2, \ldots Y_N)^T$ and

- $E(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\beta}) = \mathbf{X}\boldsymbol{\beta}$, where $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots \beta_p)^T$ is the p-vector of regression parameters.
- $Cov(\mathbf{Y} \mid \mathbf{X}, \beta) = \sigma^2 \mathrm{I}_n$, for some $\sigma^2 > 0$.

Equivalently: $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$

where $E[\boldsymbol{\epsilon}] = \mathbf{0}$ and $Cov[\boldsymbol{\epsilon}] = \sigma^2 \mathrm{I}_n$

Gauss-Markov Theorem

For known constants $c_1, c_2, \ldots, c_p, c_{p+1}$, consider the problem of estimating

$$\theta = c_1\beta_1 + c_2\beta_2 + \cdots c_p\beta_p + c_{p+1}.$$

Under the Gauss-Markov assumptions, the estimator

$$\hat{\theta} = c_1\hat{\beta}_1 + c_2\hat{\beta}_2 + \cdots c_p\hat{\beta}_p + c_{p+1},$$

where $\hat{\beta}_1, \hat{\beta}_2, \ldots \hat{\beta}_p$ are the least squares estimates is

1) An **Unbiased Estimator** of θ

2) A **Linear Estimator** of θ, that is

$\hat{\theta} = \sum_{i=1}^{n} b_i y_i$, for some known (given $\mathbf{X}$) constants b_i.

Theorem: Under the Gauss-Markov Assumptions, the estimator $\hat{\theta}$ has the smallest (*Best*) variance among all *Linear Unbiased Estimators* of θ, i.e., $\hat{\theta}$ is *BLUE*.

Generalized Least Squares (GLS) Estimates

Consider generalizing the Gauss-Markov assumptions for the linear regression model to

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

where the random n-vector $\boldsymbol{\epsilon}$: $E[\boldsymbol{\epsilon}] = \mathbf{0}_n$ and $E[\boldsymbol{\epsilon}\boldsymbol{\epsilon}'] = \sigma^2\Sigma$.

- σ^2 is an unknown scale parameter
- Σ is a known $(n \times n)$ positive definite matrix specifying the relative variances and correlations of the component observations.

Transform the data $(\mathbf{Y}, \mathbf{X})$ to $\mathbf{Y}^* = \Sigma^{-\frac{1}{2}}\mathbf{Y}$ and $\mathbf{X}^* = \Sigma^{-\frac{1}{2}}\mathbf{X}$ and the model becomes

$$\mathbf{Y}^* = \mathbf{X}^*\boldsymbol{\beta} + \boldsymbol{\epsilon}^*, \text{ where } E[\boldsymbol{\epsilon}^*] = \mathbf{0}_n \text{ and } E[\boldsymbol{\epsilon}^*(\boldsymbol{\epsilon}^*)'] = \sigma^2\mathbf{I}_n$$

By the Gauss-Markov Theorem, the BLUE ('GLS') of $\boldsymbol{\beta}$ is

$$\hat{\beta} = [(\mathbf{X}^*)^T(\mathbf{X}^*)]^{-1}(\mathbf{X}^*)^T(\mathbf{Y}^*) = [\mathbf{X}^T\Sigma^{-1}\mathbf{X}]^{-1}(\mathbf{X}^T\Sigma^{-1}\mathbf{Y})$$

Maximum-Likelihood Estimation

Consider the normal linear regression model:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \text{ where } \{\epsilon_i\} \text{ are i.i.d. } N(0, \sigma^2), \text{ i.e.,}$$

$$\begin{aligned} \boldsymbol{\epsilon} &\sim N_n(\mathbf{0}_n, \sigma^2 \mathrm{I}_\mathbf{n}) \\ \text{or } \mathbf{y} &\sim N_n(\mathbf{X}\boldsymbol{\beta}, \sigma^2 \mathrm{I}_\mathbf{n}) \end{aligned}$$

Definitions:

- The **likelihood function** is
$$L(\boldsymbol{\beta}, \sigma^2) = p(\mathbf{y} \mid \mathbf{X}, \mathbf{B}, \sigma^2)$$
where $p(\mathbf{y} \mid \mathbf{X}, \mathbf{B}, \sigma^2)$ is the joint probability density function (pdf) of the conditional distribution of $\mathbf{y}$ given data $\mathbf{X}$, (known) and parameters $(\boldsymbol{\beta}, \sigma^2)$ (unknown).
- The **maximum likelihood** estimates of $(\boldsymbol{\beta}, \sigma^2)$ are the values maximizing $L(\boldsymbol{\beta}, \sigma^2)$, i.e., those which make the observed data $\mathbf{y}$ most likely in terms of its pdf.

Because the y_i are independent r.v.'s with $y_i \sim N(\mu_i, \sigma^2)$ where $\mu_i = \sum_{j=1}^{p} \beta_j x_{i,j}$,

$$\begin{aligned} L(\boldsymbol{\beta}, \sigma^2) &= \prod_{i=1}^{n} p(y_i \mid \beta, \sigma^2) \\ &= \prod_{i=1}^{n} \left[\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(y_i - \sum_{j=1} \beta_j x_{i,j})^2} \right] \\ &= \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-\frac{1}{2}(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})^T(\sigma^2 \mathrm{I_n})^{-1}(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})} \end{aligned}$$

The maximum likelihood estimates $(\hat{\boldsymbol{\beta}}, \hat{\sigma}^2)$ maximize the log-likeliood function (dropping constant terms)

$$\begin{aligned} logL(\boldsymbol{\beta}, \sigma^2) &= -\frac{n}{2} log(\sigma^2) - \frac{1}{2}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^T(\sigma^2 \mathrm{I_n})^{-1}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \\ &= -\frac{n}{2} log(\sigma^2) - \frac{1}{2\sigma^2} Q(\boldsymbol{\beta}) \end{aligned}$$

where $Q(\boldsymbol{\beta}) = (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$ ("Least-Squares Criterion"!)

- The OLS estimate $\hat{\boldsymbol{\beta}}$ is also the ML-estimate.
- The ML estimate of σ^2 solves

$$\frac{\partial log\ L(\hat{\boldsymbol{\beta}}, \sigma^2)}{\partial(\sigma^2)} = 0 \text{ ,i.e., } -\frac{n}{2}\frac{1}{\sigma^2} - \frac{1}{2}(-1)(\sigma^2)^{-2} Q(\hat{\boldsymbol{\beta}}) = 0$$

$$\Longrightarrow \hat{\sigma^2_{ML}} = Q(\hat{\boldsymbol{\beta}})/n = (\textstyle\sum_{i=1}^{n} \hat{\epsilon}_i^2)/n \quad (biased!)$$

Generalized M Estimation

For data $\mathbf{y}$, $\mathbf{X}$ fit the linear regression model

$$\mathbf{y}_i = \mathbf{x}_i^T \boldsymbol{\beta} + \epsilon_i,\ i = 1, 2, \ldots, n.$$

by specifying $\boldsymbol{\beta} = \hat{\boldsymbol{\beta}}$ to minimize

$$Q(\boldsymbol{\beta}) = \sum_{i=1}^{n} h(y_i, \mathbf{x}_i, \boldsymbol{\beta}, \sigma^2)$$

The choice of the function $h(\ )$ distinguishes different estimators.

(1) Least Squares: $h(y_i, \mathbf{x}_i, \boldsymbol{\beta}, \sigma^2) = (y_i - \mathbf{x}_i^T \boldsymbol{\beta})^2$

(2) Mean Absolue Deviation (MAD):
$h(y_i, \mathbf{x}_i, \boldsymbol{\beta}, \sigma^2) = |y_i - \mathbf{x}_i^T \boldsymbol{\beta}|$

(3) Maximum Likelihood (ML): Assume the y_i are independent with pdf's $p(y_i \mid \boldsymbol{\beta}, \mathbf{x}_i, \sigma^2)$,

$$h(y_i, \mathbf{x}_i, \boldsymbol{\beta}, \sigma^2) = -log\ p(y_i \mid \boldsymbol{\beta}, \mathbf{x}_i, \sigma^2)$$

(4) Robust $M-$Estimator: $h(y_i, \mathbf{x}_i, \boldsymbol{\beta}, \sigma^2) = \chi(y_i - \mathbf{x}_i^T \boldsymbol{\beta})$

$\chi(\ )$ is even, monotone increasing on $(0, \infty)$.

(5) Quantile Estimator: For $\tau : 0 < \tau < 1$, a fixed *quantile*

$$h(y_i, \mathbf{x}_i, \boldsymbol{\beta}, \sigma^2) = \begin{cases} \tau|y_i - \mathbf{x}_i^T\boldsymbol{\beta}|, & \text{if } y_i \geq \mathbf{x}_i\boldsymbol{\beta} \\ (1-\tau)|y_i - \mathbf{x}_i^T\boldsymbol{\beta}|, & \text{if } y_i < \mathbf{x}_i\boldsymbol{\beta} \end{cases}$$

- E.g., $\tau = 0.90$ corresponds to the 90th quantile / upper-decile.
- $\tau = 0.50$ corresponds to the *MAD* Estimator

Ridge Regression

Ridge Regression Estimate:

$$\hat{\beta}^{ridge} = \underset{\vec{\beta}}{\text{argmin}} \left(\sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right)$$

Note:

- $\lambda \geq 0$: complexity parameter
- $\hat{\beta}_0 = \bar{y} = \sum_1^n y_i / n$

Center/Standardize Variables

- Center $\vec{y} \in R^n$ and the columns of $(n \times p)$ matrix X at their mean values; replace $\vec{y}$ with $H^* y$, and X with $H^* X$, where
$$H^* = \mathrm{I}_n - \frac{1}{n} \vec{1}\, \vec{1}^\top$$
- Rescale columns of X to have unit variance; replace
$$X \text{ with } S^{-\frac{1}{2}} X, \text{ where } S = \frac{1}{n} diag(X^\top X)$$

Matrix/Vector Formulation:

$$\begin{aligned} \vec{y} &= X\vec{\beta} + \vec{\epsilon}, \quad \text{where } \vec{\beta} \in R^p, \text{ and } \vec{\epsilon} \in R^n \\ \hat{\beta}^{ridge} &= \underset{\vec{\beta}}{\text{argmin}} \left(||\vec{y} - X\vec{\beta}||^2 + \lambda ||\vec{\beta}||^2 \right) \end{aligned}$$

Ridge Regression

Ridge-Regression Estimate

$$\begin{aligned}\hat{\beta}^{ridge} &= \underset{\vec{\beta}}{\text{argmin}} \left(||\vec{y} - X\vec{\beta}||^2 + \lambda ||\vec{\beta}||^2 \right) \\ &= (X^\top X + \lambda \mathrm{I}_p)^{-1} X^\top \vec{y}\end{aligned}$$

- $\hat{\beta}^{ridge}$ is linear in $\vec{y}$
- $\lambda = 0 \Longrightarrow \hat{\beta}^{ridge}$ equals Least-Squares $\hat{\beta} = (X^T X)^{-1} X^T \vec{y}$.

Ridge-Regression Fitted Values

$$\hat{y}^{ridge} = X\hat{\beta}^{ridge} = X(X^T X + \lambda \mathrm{I}_p)^{-1} X^T \vec{y}$$

Use SVD (Singular Value Decomposition): $X = UDV^T$, where

U: $(n \times p)$ orthonormal columns, $(U^T U) = \mathrm{I}_p$

V: $(p \times p)$ orthogonal, $V^T = V^{-1}$

D: $(p \times p)$ diagonal, $diag(d_1, \ldots, d_p)$

$$\begin{aligned}\hat{y}^{ridge} &= UD(D^2 + \lambda \mathrm{I}_p)^{-1} DU^T \vec{y} \\ &= UD^* U^T \vec{y} \quad \text{where } D^* = diag(d_j^2/(d_j^2 + \lambda))\end{aligned}$$

Fitted Values: Ridge vs LS Regression

Ridge-Regression Fitted Values

$$\begin{aligned} \Longrightarrow \quad \hat{y}^{ridge} &= UD(D^2 + \lambda \mathrm{I}_p)^{-1} DU^T \vec{y} \\ &= \sum_{j=1}^{p} \frac{d_j^2}{d_j^2+\lambda} \vec{u}_j \vec{u}_j^{\top} \vec{y} \text{ where } \vec{u}_j \text{ is } j\text{th column of } U \\ &= \sum_{j=1}^{p} (\frac{d_j^2}{d_j^2+\lambda}) c_j \vec{u}_j, \quad \text{where } c_j = \vec{u}_j^T \vec{y} \end{aligned}$$

Least-Squares Fitted Values

$$\begin{aligned} \hat{y} &= UU^T \vec{y} \\ &= \sum_{j=1}^{p} c_j \vec{u}_j \end{aligned}$$

- Ridge regression shrinks c_j in $\vec{u}_j$ direction
- Less shrinkage the larger d_j^2
- More shrinkage the smaller d_j^2

Principal Components Regression

Principal Components of X

- X: $(n \times p)$ with de-meaned columns
- SVD: $X = UDV^T$, where
  $U = [\vec{u}_1 \ \cdots \ \vec{u}_p]$: $(n \times p)$ orthonormal columns, $(U^T U)$
  $V = [\vec{v}_1 \ \cdots \ \vec{v}_p]$: $(p \times p)$ orthogonal, $V^T = V^{-1}$
  $D = diag(d_1, \ldots, d_p)$: $(p \times p)$ diagonal $(d_1 \geq \cdots d_p)$
- Sample covariance matrix of row-vectors of X:
  $S = \frac{1}{n} \sum_{i=1}^{n} \vec{x}_i \vec{x}_i^\top = \frac{1}{n} X^T X = \frac{1}{n} V D^2 V^T$
  $(p \times p)$ covariance matrix of $\{\vec{x}_i, 1 \leq n\}$
- $\vec{v}_j$: Eigen Vector of $X^T X$ and S
  $[X^T X]\vec{v}_j = d_j^2 \vec{v}_j$ and $S\vec{v}_j = \frac{d_j^2}{n} \vec{v}_j$
- Principal component (PC) variables of X:
  $\vec{z}_j = X\vec{v}_j, \quad (j = 1, \ldots, p)$

Principal Components / Regression

Properties of Principal Component Variables

- First PC variable maximizes sample variance of normalized linear combinations of columns of X

$$\begin{aligned} Var(\vec{z}_1) &= Var(X\vec{v}_1) = \frac{d_1^2}{n} \\ &= \max\{Var(X\vec{a}) : \ \vec{a} \in R^p \text{ and } |\vec{a}| = 1\} \end{aligned}$$

- jth PC variable maximizes sample variance of normalized linear combination subject to being orthogonal to $\vec{z}_1, \ldots, \vec{z}_{j-1}$

$$\begin{aligned} Var(\vec{z}_j) &= Var(X\vec{v}_j) = \frac{d_j^2}{n} \\ &= \max\{Var(X\vec{a}) : \ |\vec{a}| = 1, \text{ and } Cov(X\vec{a}, \vec{z}_i) = 0, i = 1 \end{aligned}$$

Principal Components Regression

- Fix $m \leq p$ (number of PC variables)
- Fit Regression of $\vec{y}$ on $\vec{z}_1, \ldots, \vec{z}_m$ (orthogonal covariates with highest variability)

Principal Components Regression

PC Regression Model

$$\vec{y} = Z_m\vec{\gamma} + \vec{\epsilon}, \text{ where}$$

- $Z_m = [\vec{z}_1 \ \cdots \vec{z}_m]$ (first m PC variables)
- $\vec{\gamma} \in R^m$, (PC regression coefficient)
- $\vec{\epsilon} \in R^n : E[\vec{\epsilon}] = \vec{0}$ and $Cov(\vec{\epsilon}) = \sigma^2 \mathrm{I}_n$ (same error vector)

PC Regression Estimate

$$\begin{aligned}
\text{Define} \quad V_m &= [\vec{v}_1 \ \cdots \ \vec{v}_m] \\
D_m &= diag(d_1, \cdots, d_m) \\
U_m &= [\vec{u}_1 \ \cdots \ \vec{u}_m] \\
\text{Then} \quad Z_m &= XV_m = U_m D_m \\
\hat{\gamma} &= [Z_m^T Z_m]^{-1} Z_m^T \vec{y} \ \text{ with } \hat{\gamma}_j = \frac{\vec{z}_j^T \vec{y}}{\vec{z}_j^T \vec{z}_j} = \frac{\vec{z}_j^T \vec{y}}{d_j^2} \\
\text{And} \quad Cov(\hat{\gamma}) &= \sigma^2 [Z_m^T Z_m]^{-1} \\
&= \sigma^2 diag(d_1^{-2}, d_2^{-2}, \cdots, d_m^{-2})
\end{aligned}$$

PC Regression Estimate

In terms of original variables:

$$\begin{aligned}\vec{y} &= Z_m\gamma + \vec{\epsilon} \\ &= XV_m\gamma + \vec{\epsilon} \\ &= X\vec{\beta}^{pc} + \vec{\epsilon}\end{aligned}$$

So:
$$\begin{aligned}\hat{\beta}^{pc} &= V_m\hat{\gamma} \\ Cov(\hat{\beta}^{pc}) &= V_m Cov(\hat{\gamma}) V_m^T \\ &= \sigma^2 V_m diag(d_1^{-2}, \ldots, d_m^{-2}) V_m^T\end{aligned}$$

Note: $Cov(\hat{\beta}^{pc}) \leq Cov(\hat{\beta})$

PC Regression Fitted Values

$$\begin{aligned}\hat{y}^{pc} &= Z_m\hat{\gamma} \\ &= Z_m[Z_m^T Z_m]^{-1} Z_m^T \vec{y} \\ &= U_m D_m [D_m^2]^{-1} D_m U_m^T \vec{y} \\ &= U_m U_m^T \vec{y} = \sum_{j=1}^{m} c_j \vec{u}_j \text{ where } c_j = \vec{u}_j^T \vec{y}\end{aligned}$$

Projection onto span of $\{\vec{z}_1, \ldots, z_m\}$

Comparison of Fitted Values

$$
\begin{aligned}
\text{Least Squares:} \quad \hat{y}^{LS} &= \sum_{j=1}^{p} c_j \vec{u}_j \\
\text{Ridge:} \quad \hat{y}^{Ridge} &= \sum_{j=1}^{p} \left(\frac{d_j^2}{d_j^2+\lambda}\right) c_j \vec{u}_j \\
\text{PC:} \quad \hat{y}^{PC} &= \sum_{j=1}^{m} c_j \vec{u}_j
\end{aligned}
$$

LASSO Regression

LASSO-Regression Estimate

$$\hat{\beta}^{LASSO} = \underset{\vec{\beta}}{\text{argmin}} \left(\sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right)$$

- Like Ridge Regression except
  $L_2 = \sum_{j=1}^{p} \beta_j^2$ replaced by $L_1 = \sum_{j=1}^{p} |\beta_j|$
- Same estimate: $\hat{\beta}_0 = \bar{y}$.
- Center/de-mean y and columns of X
- Scale columns of X to have unit variances.

Matrix/Vector Formulation:

$$\begin{aligned} \vec{y} &= X\vec{\beta} + \vec{\epsilon}, \quad \text{where } \vec{\beta} \in R^p, \text{ and } \vec{\epsilon} \in R^n \\ \hat{\beta}^{LASSO} &= \underset{\vec{\beta}}{\text{argmin}} \left(||\vec{y} - X\vec{\beta}||^2 + \lambda |\vec{\beta}| \right) \end{aligned}$$

Equivalent Constrained Optimization Problem

Minimize: $||\vec{y} - X\vec{\beta}||^2$
Subject to: $|\beta| \leq t$

LASSO Regression

- Constraint t in Optimization Problem binding only when
$$t < t_0 = |\hat{\beta}^{LS}|$$
- Regularization paths for LASSO Estimates:
$$\hat{\beta}^{LASSO} \text{ versus } \lambda, \text{ for } \lambda \geq 0$$
$$\hat{\beta}^{LASSO} \text{ versus } t \text{ for } 0 \leq t \leq t_0.$$

RStudio Cloud Project

See: ETF_casestudy.pdf in ETF_casestudy.zip

- Least Squares Regression

  Regression diagnostics from R package car

- Principal Components Regression

- Ridge Regression

- Lasso Regression

18.642 Topics in Mathematics with Applications in Finance

Fall 2024