STAT 3690 Lecture Note

Part VI: Linear model

Zhiyang Zhou (zhiyang.zhou@umanitoba.ca, zhiyanggeezhou.github.io)

2023/Mar/22 11:15:48

Multivariate linear regression

What is a linear model?

- Responses are linear functions with respect to unknown parameters.

Univariate/multiple linear regression (J&W Sec. 7.2–7.5)

- Model (population version):

$$Y \mid X_1, \ldots, X_q \sim \left(\sum_{j=1}^{q} X_j \beta_j, \sigma^2\right)$$

  - Equiv. $Y = \sum_{j=1}^{q} X_j \beta_j + \varepsilon$ with $\varepsilon \perp\!\!\!\perp [X_1, \ldots, X_q]^\top$ and $\varepsilon \sim (0, \sigma^2)$
  - Univariate linear regression: $q = 2$ with $X_1 = 1$
  - Multiple linear regression: $q > 2$ with $X_1 = 1$
- Model (sample version):

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

  - $\boldsymbol{Y} = [Y_1, \ldots, Y_n]^\top$
  - Design matrix

$$\boldsymbol{X} = \begin{bmatrix} X_{11} & \cdots & X_{1q} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{nq} \end{bmatrix}_{n \times q}$$

    - $\mathrm{rk}(\boldsymbol{X}) = q$
  - $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_q]^\top$
  - $\boldsymbol{\varepsilon} = [\varepsilon_1, \ldots, \varepsilon_n]^\top \sim (\mathbf{0}_n, \sigma^2 \mathbf{I}_n)$, independent of $\boldsymbol{X}$

- Least squares (LS) estimation (no need of normality)
  - $\hat{\boldsymbol{\beta}}_{\mathrm{LS}} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{Y}$
    - $\mathrm{E}(\hat{\boldsymbol{\beta}}_{\mathrm{LS}} \mid \boldsymbol{X}) = \boldsymbol{\beta}$
  - $\hat{\sigma}^2_{\mathrm{LS}} = (n-q)^{-1} (\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_{\mathrm{LS}})^\top (\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_{\mathrm{LS}}) = (n-q)^{-1} \boldsymbol{Y}^\top (\mathbf{I} - \mathbf{H}) \boldsymbol{Y}$
    - $n \times n$ hat matrix $\mathbf{H} = \boldsymbol{X}(\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top$
    - $\mathrm{E}(\hat{\sigma}^2_{\mathrm{LS}} \mid \boldsymbol{X}) = \sigma^2$

- ML estimation (under normality)

- $\hat{\boldsymbol{\beta}}_{\text{ML}} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1}\boldsymbol{X}^\top \boldsymbol{Y} = \hat{\boldsymbol{\beta}}_{\text{LS}}$
  - $\hat{\boldsymbol{\beta}}_{\text{ML}} \mid \boldsymbol{X} \sim \text{MVN}_q(\boldsymbol{\beta}, \sigma^2(\boldsymbol{X}^\top \boldsymbol{X})^{-1})$
- $\hat{\sigma}^2_{\text{ML}} = n^{-1}\boldsymbol{Y}(\mathbf{I} - \mathbf{H})\boldsymbol{Y} = n^{-1}(n-q)\hat{\sigma}^2_{\text{LS}}$
  - Given $\boldsymbol{X}$, $n\hat{\sigma}^2_{\text{ML}}/\sigma^2 = (n-q)\hat{\sigma}^2_{\text{LS}}/\sigma^2 \sim \chi^2(n-q)$

- Inference (under normality)
  - To infer $\boldsymbol{a}^\top\boldsymbol{\beta}$, given $\boldsymbol{a} \in \mathbb{R}^q$ (e.g., to compare β_1 and β_2 by checking $\boldsymbol{a}^\top\boldsymbol{\beta} = \beta_1 - \beta_2$ with $\boldsymbol{a} = [1, -1, 0, \ldots, 0]^\top$)
    - Estimator: $\boldsymbol{a}^\top\hat{\boldsymbol{\beta}}_{\text{ML}}$
    - $100 \times (1-\alpha)\%$ confidence interval for $\boldsymbol{a}^\top\boldsymbol{\beta}$:

$$\boldsymbol{a}^\top\hat{\boldsymbol{\beta}}_{\text{ML}} \pm \hat{\sigma}_{\text{LS}} \cdot t_{1-\alpha/2,n-q}\sqrt{\boldsymbol{a}^\top(\boldsymbol{X}^\top\boldsymbol{X})^{-1}\boldsymbol{a}}$$

  - To predict $Y_0 = \boldsymbol{X}_0^\top\boldsymbol{\beta} + \varepsilon_0$ with $\boldsymbol{X}_0$ different from each row of $\boldsymbol{X}$
    - Prediction: $\hat{Y}_0 = \boldsymbol{X}_0^\top\hat{\boldsymbol{\beta}}_{\text{ML}}$
    - $100 \times (1-\alpha)\%$ prediction interval for Y_0

$$\boldsymbol{X}_0^\top\hat{\boldsymbol{\beta}}_{\text{ML}} \pm \hat{\sigma}_{\text{LS}} \cdot t_{1-\alpha/2,n-q}\sqrt{1 + \boldsymbol{X}_0^\top(\boldsymbol{X}^\top\boldsymbol{X})^{-1}\boldsymbol{X}_0}$$

Multivariate linear regression

- Model (population version):

$$Y_1, \ldots, Y_p \mid X_1, \ldots, X_q \sim ([X_1, \ldots, X_q]\mathbf{B}, \boldsymbol{\Sigma})$$

  - Equiv. $[Y_1, \ldots, Y_p] = [X_1, \ldots, X_q]\mathbf{B} + \boldsymbol{\varepsilon}^\top$ with p-vector $\boldsymbol{\varepsilon} \perp\!\!\!\perp [X_1, \ldots, X_q]$ and $\boldsymbol{\varepsilon} \sim (\mathbf{0}_p, \boldsymbol{\Sigma})$
    - Unknown coefficients

$$\mathbf{B} = \begin{bmatrix} b_{11} & \cdots & b_{1p} \\ \vdots & \ddots & \vdots \\ b_{q1} & \cdots & b_{qp} \end{bmatrix}_{q\times p} = \begin{bmatrix} \boldsymbol{b}_{1\cdot}^\top \\ \vdots \\ \boldsymbol{b}_{q\cdot}^\top \end{bmatrix} = \begin{bmatrix} \boldsymbol{b}_{\cdot 1} & \cdots & \boldsymbol{b}_{\cdot p} \end{bmatrix}$$

      - $\boldsymbol{b}_{i\cdot}^\top$: the ith row of $\mathbf{B}$
      - $\boldsymbol{b}_{\cdot j}$: the jth column of $\mathbf{B}$
- Model (sample version):

$$\underset{n\times p}{\boldsymbol{Y}} = \underset{n\times q}{\boldsymbol{X}}\,\underset{q\times p}{\mathbf{B}} + \underset{n\times p}{\boldsymbol{E}}$$

  - Response

$$\boldsymbol{Y} = \begin{bmatrix} Y_{11} & \cdots & Y_{1p} \\ \vdots & \ddots & \vdots \\ Y_{n1} & \cdots & Y_{np} \end{bmatrix}_{n\times p}$$

  - Design matrix

$$\boldsymbol{X} = \begin{bmatrix} X_{11} & \cdots & X_{1q} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{nq} \end{bmatrix}_{n\times q}$$

    - $\text{rk}(\boldsymbol{X}) = q < p + q \leq n$
  - Error

$$\boldsymbol{E} = \begin{bmatrix} e_{11} & \cdots & e_{1q} \\ \vdots & \ddots & \vdots \\ e_{n1} & \cdots & e_{nq} \end{bmatrix}_{n\times q} = \begin{bmatrix} \boldsymbol{e}_{1\cdot}^\top \\ \vdots \\ \boldsymbol{e}_{n\cdot}^\top \end{bmatrix}$$

* $\boldsymbol{e}_{i\cdot} \perp\!\!\!\perp [X_{i1}, \ldots, X_{iq}]$
* $\boldsymbol{e}_{i\cdot} \overset{\text{iid}}{\sim} (\mathbf{0}_p, \boldsymbol{\Sigma})$

- Relationship with MANOVA
  - MANOVA models can be expressed as multivariate linear regression with a carefully selected $\boldsymbol{X}$.
- Exercise 6.1: rephrase the following one-way MANOVA model

$$\boldsymbol{Y}_{ij} = \boldsymbol{\mu} + \boldsymbol{\tau}_i + \boldsymbol{E}_{ij}, \quad j = 1, \ldots, n_i, \quad i = 1, \ldots, m$$

into a multivariate linear regression model, where $\boldsymbol{E}_{ij} \overset{\text{iid}}{\sim} \text{MVN}_p(\mathbf{0}, \boldsymbol{\Sigma})$ and $\sum_i \boldsymbol{\tau}_i = 0$.

- LS estimation (no need of normality)
  - $\hat{\mathbf{B}}_{\text{LS}} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{Y}$
    * $\text{E}(\hat{\mathbf{B}}_{\text{LS}} \mid \boldsymbol{X}) = \mathbf{B}$
  - $\hat{\boldsymbol{\Sigma}}_{\text{LS}} = (n-q)^{-1}(\boldsymbol{Y} - \boldsymbol{X}\hat{\mathbf{B}}_{\text{LS}})^\top (\boldsymbol{Y} - \boldsymbol{X}\hat{\mathbf{B}}_{\text{LS}}) = (n-q)^{-1} \boldsymbol{Y}^\top (\mathbf{I} - \mathbf{H}) \boldsymbol{Y}$
    * $\text{E}(\hat{\boldsymbol{\Sigma}}_{\text{LS}} \mid \boldsymbol{X}) = \boldsymbol{\Sigma}$
- ML estimation (under normality)
  - $\hat{\mathbf{B}}_{\text{ML}} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{Y} = \hat{\mathbf{B}}_{\text{LS}}$
  - $\hat{\boldsymbol{\Sigma}}_{\text{ML}} = n^{-1} \boldsymbol{Y}^\top (\mathbf{I} - \mathbf{H}) \boldsymbol{Y} = n^{-1}(n-q)\hat{\boldsymbol{\Sigma}}_{\text{LS}}$
    * Given $\boldsymbol{X}$, $n\hat{\boldsymbol{\Sigma}}_{\text{ML}} \sim W_p(\boldsymbol{\Sigma}, n-q)$

- Inference (under normality)
  - To infer $\mathbf{B}^\top \boldsymbol{a}$, given $\boldsymbol{a} \in \mathbb{R}^q$ (e.g., to compare the 1st and 2nd rows of $\mathbf{B}$, i.e., $\boldsymbol{b}_{1\cdot}$ and $\boldsymbol{b}_{2\cdot}$, by checking $\mathbf{B}^\top \boldsymbol{a} = \boldsymbol{b}_{1\cdot} - \boldsymbol{b}_{2\cdot}$ with $\boldsymbol{a} = [1, -1, 0, \ldots, 0]^\top$)
    * Estimator: $\hat{\mathbf{B}}_{\text{ML}}^\top \boldsymbol{a}$
    * $100 \times (1-\alpha)\%$ confidence region for $\mathbf{B}^\top \boldsymbol{a}$

$$\left\{ \boldsymbol{u} \in \mathbb{R}^p : (\boldsymbol{u} - \hat{\mathbf{B}}_{\text{ML}}^\top \boldsymbol{a})^\top \hat{\boldsymbol{\Sigma}}_{\text{LS}}^{-1} (\boldsymbol{u} - \hat{\mathbf{B}}_{\text{ML}}^\top \boldsymbol{a}) \le \boldsymbol{a}^\top (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{a} \cdot \frac{p(n-q)}{n-p-q+1} F_{1-\alpha, p, n-p-q+1} \right\}$$

  - To predict $\boldsymbol{Y}_0 = \mathbf{B}^\top \boldsymbol{X}_0 + \boldsymbol{E}_0$ with newly observed $\boldsymbol{X}_0 \in \mathbb{R}^q$
    * Prediction: $\hat{\boldsymbol{Y}}_0 = \mathbf{B}_{\text{ML}}^\top \boldsymbol{X}_0$
    * $100 \times (1-\alpha)\%$ prediction region for $\boldsymbol{Y}_0$

$$\left\{ \boldsymbol{u} \in \mathbb{R}^p : (\boldsymbol{u} - \hat{\boldsymbol{Y}}_0)^\top \hat{\boldsymbol{\Sigma}}_{\text{LS}}^{-1} (\boldsymbol{u} - \hat{\boldsymbol{Y}}_0) \le \{1 + \boldsymbol{X}_0^\top (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}_0\} \cdot \frac{p(n-q)}{n-p-q+1} F_{1-\alpha, p, n-p-q+1} \right\}$$

  - To infer $\boldsymbol{a}^\top \boldsymbol{Y}_0 = \boldsymbol{a}^\top (\mathbf{B}^\top \boldsymbol{X}_0 + \boldsymbol{E}_0)$, given $\boldsymbol{a} \in \mathbb{R}^p$ and newly observed $\boldsymbol{X}_0 \in \mathbb{R}^q$
    * Prediction: $\boldsymbol{a}^\top \hat{\boldsymbol{Y}}_0 = \boldsymbol{a}^\top \mathbf{B}_{\text{ML}}^\top \boldsymbol{X}_0$
    * $100 \times (1-\alpha)\%$ prediction interval for $\boldsymbol{a}^\top \boldsymbol{Y}_0$

$$\boldsymbol{a}^\top \hat{\boldsymbol{Y}}_0 \pm \sqrt{\boldsymbol{a}^\top \hat{\boldsymbol{\Sigma}}_{\text{LS}} \boldsymbol{a} \cdot \{1 + \boldsymbol{X}_0^\top (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}_0\}} \cdot t_{1-\alpha/2, n-q}$$

  - $100 \times (1-\alpha)\%$ simultaneous prediction intervals for $\boldsymbol{a}_k^\top \boldsymbol{Y}_0$, $k = 1, \ldots, m$, given $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m \in \mathbb{R}^p$ and newly observed $\boldsymbol{X}_0 \in \mathbb{R}^q$
    * (Bonferroni)

$$\boldsymbol{a}_k^\top \hat{\boldsymbol{Y}}_0 \pm \sqrt{\boldsymbol{a}_k^\top \hat{\boldsymbol{\Sigma}}_{\text{LS}} \boldsymbol{a}_k \cdot \{1 + \boldsymbol{X}_0^\top (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}_0\}} \cdot t_{1-\alpha/(2m), n-q}$$

    * (Scheffé's)

$$\boldsymbol{a}_k^\top \hat{\boldsymbol{Y}}_0 \pm \sqrt{\boldsymbol{a}_k^\top \hat{\boldsymbol{\Sigma}}_{\text{LS}} \boldsymbol{a}_k \cdot \{1 + \boldsymbol{X}_0^\top (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}_0\} \cdot \frac{p(n-q)}{n-p-q+1} F_{1-\alpha, p, n-p-q+1}}$$

```r
options(digits = 4)
tear <- c(
  6.5, 6.2, 5.8, 6.5, 6.5, 6.9, 7.2, 6.9, 6.1, 6.3,
  6.7, 6.6, 7.2, 7.1, 6.8, 7.1, 7.0, 7.2, 7.5, 7.6
)
gloss <- c(
  9.5, 9.9, 9.6, 9.6, 9.2, 9.1, 10.0, 9.9, 9.5, 9.4,
  9.1, 9.3, 8.3, 8.4, 8.5, 9.2, 8.8, 9.7, 10.1, 9.2
)
rate <- factor(gl(2,10,length=length(tear)), labels=c("Low", "High"))

# Model fitting
fit <- lm(cbind(tear, gloss) ~ rate)
summary(fit)

# Prediction
(Obs_new <- data.frame(rate = factor(c("High"), levels = c("Low", "High"))))
(prediction <- t(predict(fit, newdata = Obs_new)))

# Prediction region
n = nrow(model.matrix(fit))
p = ncol(coef(fit))
q = ncol(model.matrix(fit))-1
(X <- model.matrix(fit))
(X0 <- t(model.matrix(~rate, Obs_new)))
(SigmaHatLS <- crossprod(resid(fit))/(n-q))
quad_form <- drop(t(X0) %*% solve(crossprod(X)) %*% X0)
fvalue = p*(n-q)/(n-p-q+1)*qf(0.95, p, n-p-q+1)

# 95% prediction region for Y0
c1 = sqrt((1 + quad_form)*fvalue)
car::ellipse(center = as.vector(prediction), shape = SigmaHatLS, radius = c1, add = F,
             xlab = "tear", ylab = "gloss", col = 'blue')

# 95% confidence region for t(B)X0
c2 = sqrt(quad_form*fvalue)
car::ellipse(center = as.vector(prediction), shape = SigmaHatLS, radius = c2, add = T,
             xlab = "tear", ylab = "gloss", col = 'red')

# 95% Scheffé's simultaneous prediction intervals for entries of Y0
a1 = c(1,0)
c(
  t(a1) %*% prediction - (t(a1) %*% SigmaHatLS %*% a1)^.5 * c1,
  t(a1) %*% prediction + (t(a1) %*% SigmaHatLS %*% a1)^.5 * c1
) # for tear
a2 = c(0,1)
c(
  t(a2) %*% prediction - (t(a2) %*% SigmaHatLS %*% a2)^.5 * c1,
  t(a2) %*% prediction + (t(a2) %*% SigmaHatLS %*% a2)^.5 * c1
) # for gloss
```

Testing for nested models

- $H_0 : \mathrm{E}(\boldsymbol{Y} \mid \boldsymbol{X}) = \boldsymbol{X}_{(0)}\mathbf{B}_{(0)}$ (nested model) vs. $H_1 : \mathrm{E}(\boldsymbol{Y} \mid \boldsymbol{X}) = \boldsymbol{X}_{(0)}\mathbf{B}_{(0)} + \boldsymbol{X}_{(1)}\mathbf{B}_{(1)}$ (larger model)
  - When $\boldsymbol{X}_{(0)}$ has only the column of ones, the model under H_0 is the empty/null model (i.e., only the intercept).
  - When $\boldsymbol{X}_{(1)}$ only involves one explanatory variable (i.e., is of $n \times 1$), we are testing the significance of that variable.
- Likelihood ratio

$$\lambda = \left(\frac{\det \hat{\boldsymbol{\Sigma}}_{\mathrm{ML},H_0}}{\det \hat{\boldsymbol{\Sigma}}_{\mathrm{ML}}} \right)^{-n/2} = \left[\det \left\{ (\hat{\boldsymbol{\Sigma}}_{\mathrm{ML},H_0} - \hat{\boldsymbol{\Sigma}}_{\mathrm{ML}}) \hat{\boldsymbol{\Sigma}}_{\mathrm{ML}}^{-1} + \mathbf{I} \right\} \right]^{-n/2}$$

- Test statistics alternative to the likelihood ratio
  - Wilks' lambda: $\prod_i (1+\eta_i)^{-1}$
  - Pillai's trace: $\sum_i \{\eta_i (1+\eta_i)^{-1}\}$
  - Hotelling-Lawley trace: $\sum_i \eta_i$
  - Roy's largest root: $\eta_1 (1+\eta_1)^{-1}$
    * Suppose $\eta_1 \geq \cdots \geq \eta_p$ are eigenvalues of $(\hat{\boldsymbol{\Sigma}}_{\mathrm{ML},H_0} - \hat{\boldsymbol{\Sigma}}_{\mathrm{ML}}) \hat{\boldsymbol{\Sigma}}_{\mathrm{ML}}^{-1}$
    * When $\boldsymbol{X}_{(1)}$ has only one column (i.e., is of $n \times 1$), all the four tests are equivalent;
    * As n increases, all thess tests give similar results.

```r
options(digits = 4)
tear <- c(
  6.5, 6.2, 5.8, 6.5, 6.5, 6.9, 7.2, 6.9, 6.1, 6.3,
  6.7, 6.6, 7.2, 7.1, 6.8, 7.1, 7.0, 7.2, 7.5, 7.6
)
gloss <- c(
  9.5, 9.9, 9.6, 9.6, 9.2, 9.1, 10.0, 9.9, 9.5, 9.4,
  9.1, 9.3, 8.3, 8.4, 8.5, 9.2, 8.8, 9.7, 10.1, 9.2
)
opacity <- c(
  4.4, 6.4, 3.0, 4.1, 0.8, 5.7, 2.0, 3.9, 1.9, 5.7,
  2.8, 4.1, 3.8, 1.6, 3.4, 8.4, 5.2, 6.9, 2.7, 1.9
)
rate <- factor(gl(2,10,length=length(tear)), labels=c("Low", "High"))
additive <- factor(gl(2,5,length=length(tear)), labels=c("Low", "High"))

# Testing the necessity of interaction
fit0 <- lm(cbind(tear, gloss, opacity) ~ 1)
fit1 = lm(cbind(tear, gloss, opacity) ~ rate*additive)
anova(fit1, fit0, test='Wilks')
anova(fit1, fit0, test='Pillai')
anova(fit1, fit0, test='Hotelling')
anova(fit1, fit0, test='Roy')
```

Information criteria

- Akaike's information criterion (AIC)

$$-\ln \mathit{Likelihood} + 2 \times \text{number of parameters to estimate}$$

  - Number of parameters to estimate in $\mathbf{B}$ and $\boldsymbol{\Sigma}$: $pq + p(p+1)/2$

    - The smaller, the better.

- Bayesian information criterion (BIC)

$$-\ln Likelihood + \ln n \times \text{number of parameters to estimate}$$

- Model selection using information criteria proceeds as follows
    - Select models of interest, say $M_1, \ldots, M_K$, which do NOT need to be nested.
        * Candidate models should be selected using domain-specific expertise, if possible. Or, you can go through all possible models.
    - Compute the specific information criterion for each model.
    - Select the model with the smallest value of the information criterion.

```
options(digits = 4)
tear <- c(
  6.5, 6.2, 5.8, 6.5, 6.5, 6.9, 7.2, 6.9, 6.1, 6.3,
  6.7, 6.6, 7.2, 7.1, 6.8, 7.1, 7.0, 7.2, 7.5, 7.6
)
gloss <- c(
  9.5, 9.9, 9.6, 9.6, 9.2, 9.1, 10.0, 9.9, 9.5, 9.4,
  9.1, 9.3, 8.3, 8.4, 8.5, 9.2, 8.8, 9.7, 10.1, 9.2
)
opacity <- c(
  4.4, 6.4, 3.0, 4.1, 0.8, 5.7, 2.0, 3.9, 1.9, 5.7,
  2.8, 4.1, 3.8, 1.6, 3.4, 8.4, 5.2, 6.9, 2.7, 1.9
)
rate <- factor(gl(2,10,length=length(opacity)), labels=c("Low", "High"))
additive <- factor(gl(2,5,length=length(opacity)), labels=c("Low", "High"))

fit0 <- lm(cbind(tear, gloss, opacity) ~ rate)
logLik(fit0)
AIC(fit0)
BIC(fit0)

logLik.mlm <- function(object, ...) {
  resids <- residuals(object)
  Sigma_ML <- crossprod(resids)/nrow(resids)
  ans <- sum(mvtnorm::dmvnorm(resids, sigma = Sigma_ML,log = TRUE))
  df <- prod(dim(coef(object))) + choose(ncol(Sigma_ML) + 1, 2)
  attr(ans, "df") <- df
  class(ans) <- "logLik"
  return(ans)
}
logLik(fit0)
AIC(fit0)
BIC(fit0)

fit1 <- lm(cbind(tear, gloss, opacity) ~ additive)
AIC(fit1)
BIC(fit1)
```

Multivariate influence measures

- Hat matrix $\mathbf{H} = [h_{ij}]_{n\times n} = \boldsymbol{X}(\boldsymbol{X}^\top\boldsymbol{X})^{-1}\boldsymbol{X}^\top$

- Leverage: the influence of the ith observation (i.e., the ith row of $\boldsymbol{Y}$), say $\boldsymbol{Y}_{i\cdot}^\top$, on $\hat{\boldsymbol{Y}}_{i\cdot}$ $(= h_{ii}\boldsymbol{Y}_{i\cdot} + \sum_{j\neq i} h_{ij}\boldsymbol{Y}_{j\cdot})$; specifically, $\boldsymbol{Y}_{i\cdot}$ is said to have a high leverage if h_{ii} is large compared to the other diagonal entries of hat matrix $\mathbf{H}$
- (Externally) Studentized residuals

$$T_i^2 = \frac{\hat{\boldsymbol{e}}_{i\cdot}^\top \hat{\boldsymbol{\Sigma}}_{\mathrm{LS},(-i)}^{-1} \hat{\boldsymbol{e}}_{i\cdot}}{1 - h_{ii}}$$

  - $\hat{\mathbf{e}}_{i\cdot}^\top$: the ith row of residual matrix $\hat{\boldsymbol{E}} = (\mathbf{I} - \mathbf{H})\boldsymbol{Y}$
  - $\hat{\boldsymbol{E}}_{(-i)\cdot}^\top$: the remaining part of $\hat{\boldsymbol{E}}$ with Row i removed
  - $\hat{\boldsymbol{\Sigma}}_{\mathrm{LS},(-i)} = (n - q - 1)^{-1}\hat{\boldsymbol{E}}_{(-i)\cdot}^\top \hat{\boldsymbol{E}}_{(-i)\cdot}$: LS estimator of $\boldsymbol{\Sigma}$ after removing Row i from the residual matrix
  - The ith observation may be considered as a potential outlier if

$$T_i^2 > \frac{p(n-q-1)}{n-p-q} F_{1-\alpha,p,n-q-1}$$

    * $F_{1-\alpha,p,n-q-1}$: the $1-\alpha$ quantile of $F(p, n-q-1)$
- (Multivariate) Cook's distance

$$D_i = \frac{h_{ii}}{q(1-h_{ii})^2} \hat{\boldsymbol{e}}_{i\cdot}^\top \hat{\boldsymbol{\Sigma}}_{\mathrm{LS}}^{-1} \hat{\boldsymbol{e}}_{i\cdot}$$

  - The Cut-off is far from unique even for multiple linear regression (i.e., the case with $p = 1$)
  - Pay attention to a small set of observations that have substantially higher values than the remaining observations

---

```
install.packages(c("car"))
options(digits = 4)
tear <- c(
  6.5, 6.2, 5.8, 6.5, 6.5, 6.9, 7.2, 6.9, 6.1, 6.3,
  6.7, 6.6, 7.2, 7.1, 6.8, 7.1, 7.0, 7.2, 7.5, 7.6
)
gloss <- c(
  9.5, 9.9, 9.6, 9.6, 9.2, 9.1, 10.0, 9.9, 9.5, 9.4,
  9.1, 9.3, 8.3, 8.4, 8.5, 9.2, 8.8, 9.7, 10.1, 9.2
)
opacity <- c(
  4.4, 6.4, 3.0, 4.1, 0.8, 5.7, 2.0, 3.9, 1.9, 5.7,
  2.8, 4.1, 3.8, 1.6, 3.4, 8.4, 5.2, 6.9, 2.7, 1.9
)
rate <- factor(gl(2,10,length=length(opacity)), labels=c("Low", "High"))
additive <- factor(gl(2,5,length=length(opacity)), labels=c("Low", "High"))

fit0 <- lm(cbind(tear, gloss, opacity) ~ rate*additive)
resids <- residuals(fit0)

# Leverage
X <- model.matrix(fit0)
H <- X %*% solve(crossprod(X)) %*% t(X)
(Hii = diag(H))
hist(Hii, 50)

# Externally Studentized residuals
n <- nrow(X)
```

```
p = ncol(resids)
T_square = numeric(n)
for (i in 1:n){
  SigmaHatLS_i <- crossprod(resids[-i,])/(n-1-ncol(X))
  T_square[i] = t(resids[i,]) %*% solve(SigmaHatLS_i) %*% resids[i,]
}
hist(T_square, 50)
which(T_square > p*(n-1-ncol(X))/(n-p-ncol(X))*qf(.95, p, n-1-ncol(X)))

# Cook's distance
SigmaHatLS <- crossprod(resids)/(n - ncol(X))
cookD <- Hii/((1 - Hii)^2*ncol(X)) * diag(resids %*% solve(SigmaHatLS) %*% t(resids))
hist(cookD, 50)
which(cookD>0.4)
```

Normality of residuals

- Check the normality of residuals following Lecture Note Part 3
- Apply Box-Cox transformation to colums of $\boldsymbol{Y}$

```
install.packages(c("car","EnvStats"))
options(digits = 4)
tear <- c(
  6.5, 6.2, 5.8, 6.5, 6.5, 6.9, 7.2, 6.9, 6.1, 6.3,
  6.7, 6.6, 7.2, 7.1, 6.8, 7.1, 7.0, 7.2, 7.5, 7.6
)
gloss <- c(
  9.5, 9.9, 9.6, 9.6, 9.2, 9.1, 10.0, 9.9, 9.5, 9.4,
  9.1, 9.3, 8.3, 8.4, 8.5, 9.2, 8.8, 9.7, 10.1, 9.2
)
opacity <- c(
  4.4, 6.4, 3.0, 4.1, 0.8, 5.7, 2.0, 3.9, 1.9, 5.7,
  2.8, 4.1, 3.8, 1.6, 3.4, 8.4, 5.2, 6.9, 2.7, 1.9
)
rate <- factor(gl(2,10,length=length(opacity)), labels=c("Low", "High"))
additive <- factor(gl(2,5,length=length(opacity)), labels=c("Low", "High"))

fit0 <- lm(cbind(tear, gloss, opacity) ~ rate*additive)

# Normal Q-Q plots of residuals
(res = residuals(fit0))
name = colnames(res)
op <- par(mfrow = c(2,2),
          oma = c(5,4,0,0),
          mar = c(1,1,2,2))
for (i in 1:ncol(res)){
  car::qqPlot(res[,i], main = name[i], id = F)
}
title(xlab = "Normal quantiles",
      ylab = "Sample quantiles",
      outer = TRUE, line = 3)
par(op)
```

```r
# Box-Cox transformation
fit1 = lm(tear ~ rate*additive)
fit2 = lm(gloss ~ rate*additive)
fit3 = lm(opacity ~ rate*additive)

(lambda1 = EnvStats::boxcox(fit1 , optimize=T, lambda=c(-10,10))$lambda)
tear.new = (tear^lambda1-1)/lambda1
(lambda2 = EnvStats::boxcox(fit2 , optimize=T, lambda=c(-10,10))$lambda)
gloss.new = (gloss^lambda2-1)/lambda2
(lambda3 = EnvStats::boxcox(fit3 , optimize=T, lambda=c(-10,10))$lambda)
opacity.new = (opacity^lambda3-1)/lambda3

fit0.new <- lm(cbind(tear.new, gloss.new, opacity.new) ~ rate*additive)

(res = residuals(fit0.new))
name = colnames(res)
op <- par(mfrow = c(2,2),
          oma = c(5,4,0,0),
          mar = c(1,1,2,2))
for (i in 1:ncol(res)){
  car::qqPlot(res[,i], main = name[i], id = F)
}
title(xlab = "Normal quantiles",
      ylab = "Sample quantiles",
      outer = TRUE, line = 3)
par(op)
```