Chapter 5 - Linear Models

2016.10.25 장수경

목차

1. **Linear Models Introduction**
2. **Solving Linear Models**
3. **Linear Models as Matrix Multiplication**
4. **Interaction effect**
5. **실습**
    1. **Expressing design formula in R**
    2. **Linear models in practice**
    3. **Standard errors**
    4. **Interactions and contrasts**
    5. **Co-linearity**
    6. **The QR decomposition**

1. Linear Models Introduction

	일반선형모델 General linear model	일반화선형모델 Generalized linear model
모델구하는 수학적 방법	Least squares Best linear unbiased prediction	Maximum likelihood Bayesian
이 부류에 속하는 통계방법들	ANOVA ANCOVA MANOVA MANCOVA Linear regression Mixed model	Linear regression Logistic regression Poisson regression Gamma regression
SPSS에서 사용하는 방법	regression, glm	genlin, logistic regression
Matlab에서 사용하는 방법	mvregress()	glmfit()
R에서 사용하는 방법	lm()	glm()
SAS에서 사용하는 방법	PROC GLM, PROC MIXED	PROC GENMOD (특히 PROC LOGISTIC를 logistic regression할때만 사용하기도 함)
Stata에서 사용하는 방법	regress	glm

http://dogmas.tistory.com/392

2. Solving Linear Models

Solving linear models

simple, fast matrix multiplication:

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$

2. Solving Linear Models

- 최소제곱법과 matrix notation

Why? Ease of working with formula, e.g.:

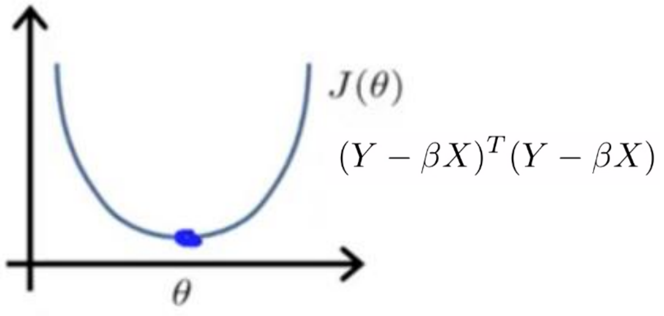
$$\sum_{i=1}^{N}(Y_i - \beta_0 - \beta_1 x_{i1} - \beta_2 x_{i2} - \beta_3 x_{i3} - \beta_4 x_{i4})^2$$

$$(Y - \beta X)^T (Y - \beta X)$$

Matrix notation makes writing formulas easy,
it also makes computation easy, it makes mathematics easy.

2. Solving Linear Models

- 최소제곱법의 계산(정규방정식,normal equation)



도함수가 0인 지점을 찾는 방식

$$2\mathbf{X}^\top(\mathbf{Y} - \mathbf{X}\hat{\beta}) = 0$$

$$\mathbf{X}^\top\mathbf{X}\hat{\beta} = \mathbf{X}^\top\mathbf{Y}$$

$$\hat{\beta} = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{Y}$$

(참고) 최소제곱법의 계산 -분석적(analytic) 방법

Define the ith **residual** to be

$$r_i = y_i - \sum_{j=1}^{n} X_{ij}\beta_j.$$

Then S can be rewritten

$$S = \sum_{i=1}^{m} r_i^2.$$

Given that S is convex, it is minimized when its gradient vector is zero (This follows by definition: if the gradient vector is not zero, there is a direction in which we can move to minimize it further - see maxima and minima.) The elements of the gradient vector are the partial derivatives of S with respect to the parameters:

$$\frac{\partial S}{\partial \beta_j} = 2\sum_{i=1}^{m} r_i \frac{\partial r_i}{\partial \beta_j} \ (j = 1, 2, \dots, n).$$

The derivatives are

$$\frac{\partial r_i}{\partial \beta_j} = -X_{ij}.$$

https://en.wikipedia.org/wiki/Linear_least_squares_(mathematics)#Derivation_of_the_normal_equations

(참고) 최소제곱법의 계산 -분석적(analytic) 방법

Substitution of the expressions for the residuals and the derivatives into the gradient equations gives

$$\frac{\partial S}{\partial \beta_j} = 2\sum_{i=1}^{m}\left(y_i - \sum_{j=1}^{n} X_{ij}\beta_j\right)(-X_{ij}) \ (j = 1, 2, \dots, n).$$

Thus if $\hat{\beta}$ minimizes S, we have

$$2\sum_{i=1}^{m}\left(y_i - \sum_{j=1}^{n} X_{ij}\hat{\beta}_j\right)(-X_{ij}) = 0 \ (j = 1, 2, \dots, n).$$

Upon rearrangement, we obtain the **normal equations**:

$$\sum_{i=1}^{m}\sum_{j=1}^{n} X_{ij}X_{ij}\hat{\beta}_j = \sum_{i=1}^{m} X_{ij}y_i \ (j = 1, 2, \dots, n).$$

The normal equations are written in matrix notation as

$(\mathbf{X}^{\mathrm{T}}\mathbf{X})\hat{\boldsymbol{\beta}} = \mathbf{X}^{\mathrm{T}}\mathbf{y}$ (where X^{T} is the matrix transpose of X).

The solution of the normal equations yields the vector $\hat{\boldsymbol{\beta}}$ of the optimal parameter values.

https://en.wikipedia.org/wiki/Linear_least_squares_(mathematics)#Derivation_of_the_normal_equations

3. Linear Models as Matrix Multiplication

Linear modeling in general

model parameters

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \varepsilon_i, \quad i \in 1, \ldots, N$$

0s or 1s, for now

observed data

3. Linear Models as Matrix Multiplication

Rewriting two groups

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

$$\begin{pmatrix} Y_1 \\ Y_2 \\ Y_3 \\ Y_4 \\ Y_5 \\ Y_6 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \\ \varepsilon_6 \end{pmatrix}$$

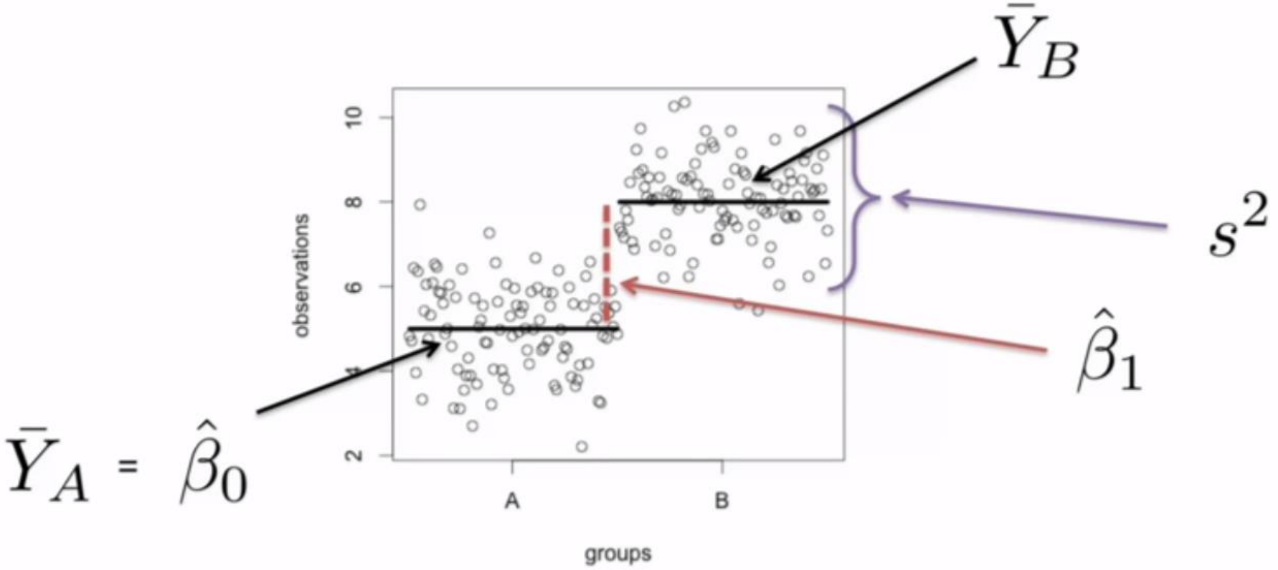
3. Linear Models as Matrix Multiplication

Rewriting two groups

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

$$\begin{pmatrix} Y_1 \\ Y_2 \\ Y_3 \\ Y_4 \\ Y_5 \\ Y_6 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \\ \varepsilon_6 \end{pmatrix}$$

Linear model of 2 groups = t-test



$$\bar{Y}_A = \hat{\beta}_0$$

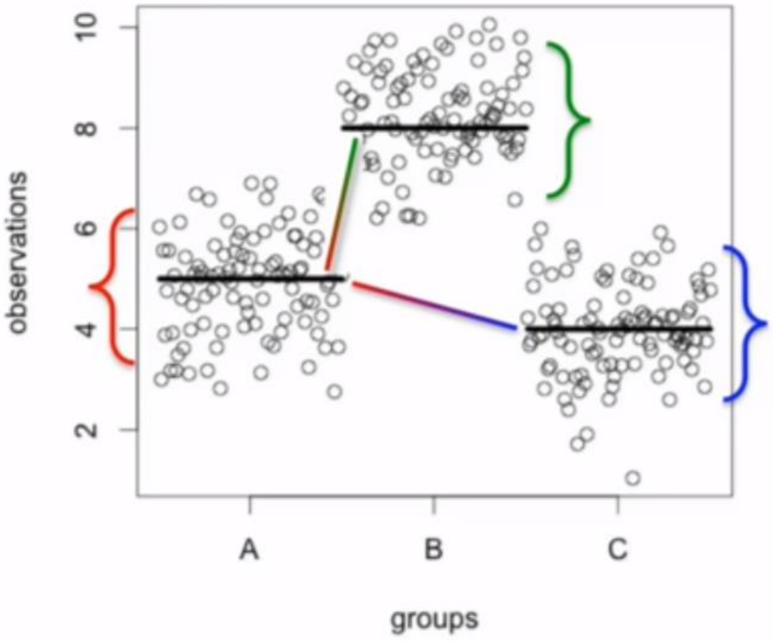
$$\bar{Y}_B = \hat{\beta}_0 + \hat{\beta}_1$$

3. Linear Models as Matrix Multiplication

Rewriting three groups

$$\begin{pmatrix} Y_1 \\ Y_2 \\ Y_3 \\ Y_4 \\ Y_5 \\ Y_6 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \\ \varepsilon_6 \end{pmatrix}$$

Linear model of 3 groups ≠ 2 t-tests

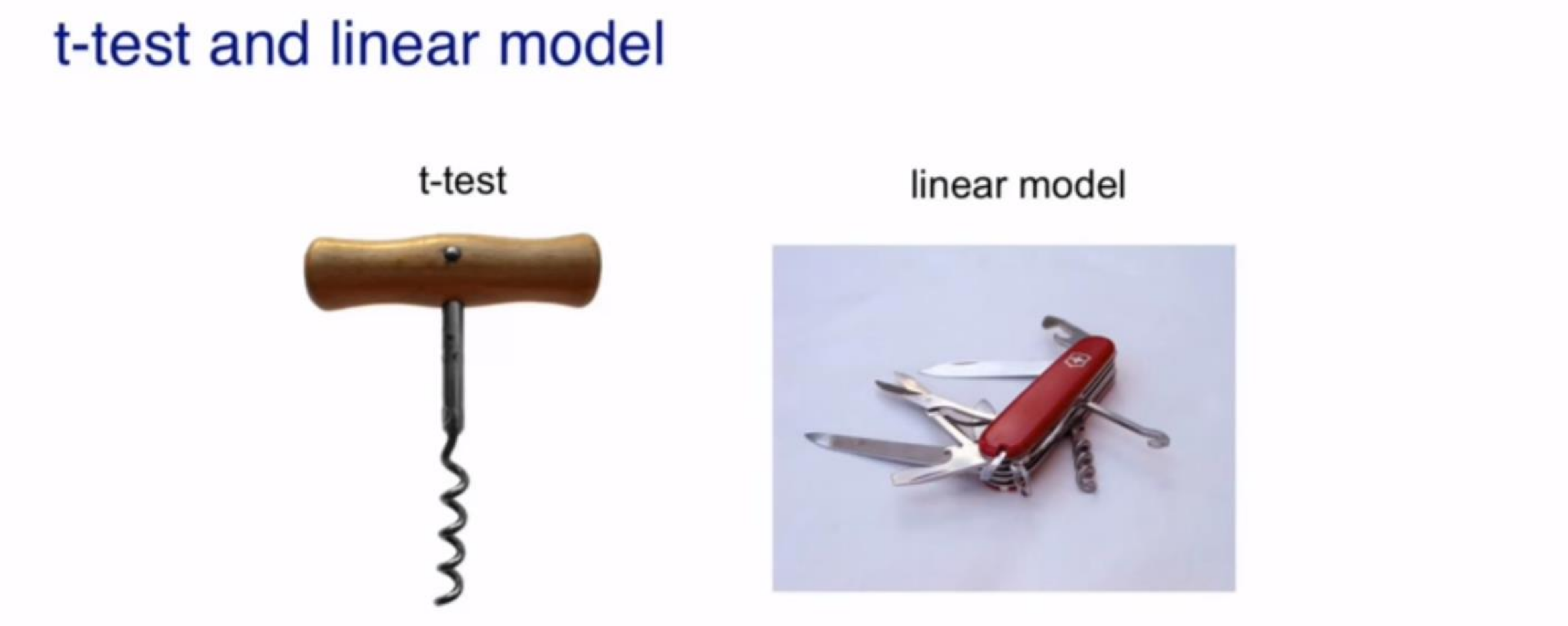


$$\bar{Y}_A = \hat{\beta}_0$$

$$\bar{Y}_B = \hat{\beta}_0 + \hat{\beta}_1$$

$$\bar{Y}_C = \hat{\beta}_0 + \hat{\beta}_2$$

3. Linear Models as Matrix Multiplication



The t-test that we described in a previous module
is actually something can be derived from the linear model machinery.
You can think of the t-test as a corkscrew,
while linear models are much more applicable and much more general,
so you can think of it as a Swiss Army knife.

실습

실습1. Expressing design formula in R

- **The Design Matrix**

```
group <- factor(c("control","control","highfat","highfat"))
model.matrix(~ group)
```

```
##   (Intercept)   grouphighfat
## 1 1             0
## 2 1             0
## 3 1             1
## 4 1             1
```

- **More variables**

```
diet <- factor(c(1,1,1,1,2,2,2,2))
sex <- factor(c("f","f","m","m","f","f","m","m"))
model.matrix(~ diet + sex)
```

$$Y_i = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \varepsilon_i$$

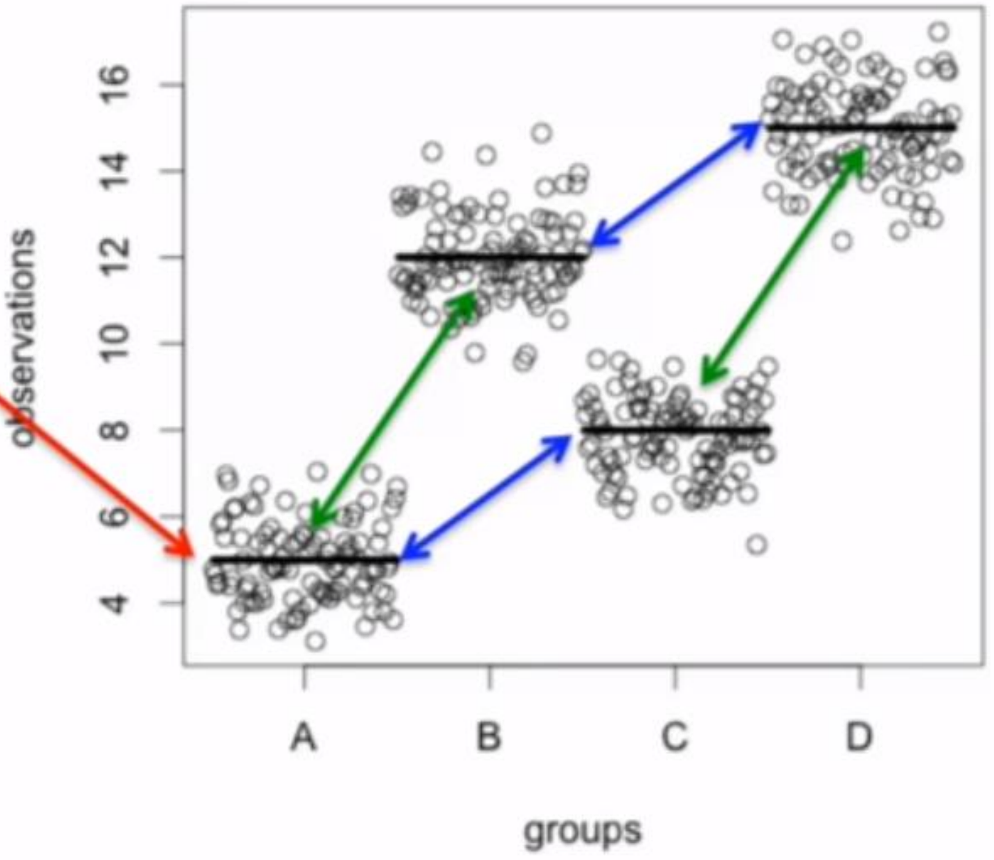
  - **The interaction model**

```
   model.matrix(~ diet + sex + diet:sex)
or model.matrix(~ diet*sex)
```

실습1. Expressing design formula in R

Crossed designs

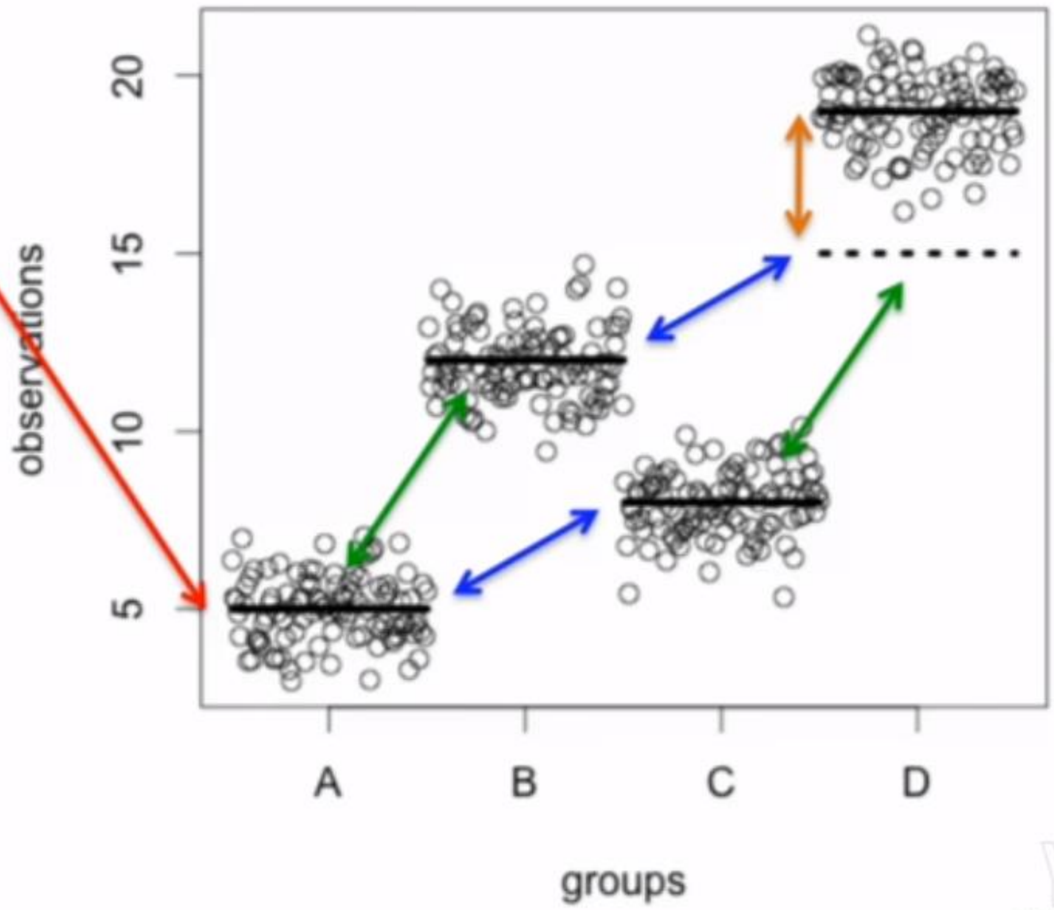
$$\begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix}$$



실습1. Expressing design formula in R

Crossed designs and interaction

$$
\begin{pmatrix}
1 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 \\
1 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 \\
1 & 0 & 1 & 0 \\
1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1
\end{pmatrix}
\begin{pmatrix}
\beta_0 \\
\beta_1 \\
\beta_2 \\
\beta_{1:2}
\end{pmatrix}
$$



HARVARD
SCHOOL OF PUBLIC HEALTH

실습2. Linear models in practice

- **The mouse diet example**

```
X <- model.matrix(~ Diet, data=dat)
fit <- lm(Bodyweight ~ Diet, data=dat)
```

- the t-statistic is the same:

```
ttest <- t.test(s[["hf"]], s[["chow"]], var.equal=TRUE)
```

실습3. Standard errors

Testing the significance

$$\frac{\text{signal}}{\text{noise}} = \frac{\hat{\beta}_i}{\mathrm{se}(\hat{\beta}_i)}$$

$$\mathrm{se}(\hat{\beta}_i) = \sqrt{s^2 (X^T X)^{-1}_{ii}}$$

실습3. Standard errors

- estimate standard errors of linear model estimates
- **Falling object example**
  - The constant is fixed, but our estimates are not. (a random variable)
  - To see this we can run a Monte Carlo simulation.
- **Father and son heights example**
- **Variance-covariance matrix** $\Sigma_{i,j} \equiv \mathrm{Cov}(Y_i, Y_j)$
  - estimate standard errors of linear model estimates(beta hat)
- **LSE standard errors** $\mathrm{var}(\hat{\beta}) = \sigma^2 (\mathbf{X}^\top \mathbf{X})^{-1}$

실습4. Interactions and contrasts

- **Contrasting coefficients**
  - a comparison which is not a single coefficient, but a combination of coefficients
  - L3vsL2 <- **contrast**(fitTL, list(leg="L3",type="pull"), list(leg="L2",type="pull"))
  - A *contrast* is a combination of estimated coefficient: $\mathbf{c}^\top \hat{\boldsymbol{\beta}}$, ,where **c** is a column vector with as many rows as the number of coefficients in the linear model.
  - The standard error of the contrast estimate : $\sqrt{\mathbf{c}^\top \hat{\boldsymbol{\Sigma}} \mathbf{c}}$ $\boldsymbol{\Sigma} = \sigma^2 (\mathbf{X}^\top \mathbf{X})^{-1}$

- **Linear Model with Interactions**
  - X <- model.matrix(~ type + leg + type:leg, data=spider)
  - fitX <- lm(friction ~ type + leg + type:leg, data=spider)

- **Analysis of variance**
  - anova(fitX)

실습5. Co-linearity(공선)

실습6. The QR decomposition

- QR 분해(QR decomposition, QR factorization)는 임의의 행렬을 **직교행렬**과 **상삼각행렬**의 곱으로 분해하는 방법이다.

$$\mathbf{X} = \mathbf{QR}$$

  - QR 분해는 선형 최소제곱법을 풀 때나 고유벡터를 구할 때 등의 상황에 사용되며, 그람-슈미트 직교정규화 혹은 하우스홀더의 방법 등을 사용한다.
  - **직교행렬**(orthogonal matrix): $Q^T = Q^{-1}$
  - **상삼각행렬**(upper triangular matrix): 대각항의 아래쪽 항들의 값이 모두 0인 정삼각행렬

$$\mathbf{R} = \begin{bmatrix} u_{1,1} & u_{1,2} & u_{1,3} & \cdots & u_{1,n} \\ & u_{2,2} & u_{2,3} & \cdots & u_{2,n} \\ & & \ddots & \ddots & \vdots \\ & & & \ddots & u_{n-1,n} \\ 0 & & & & u_{n,n} \end{bmatrix}$$

https://ko.wikipedia.org/wiki/QR_분해

실습6. The QR decomposition

$$\mathbf{X} = \mathbf{QR}$$

- **Finding LSE with QR**

$$\mathbf{X}^\top \mathbf{X}\beta = \mathbf{X}^\top \mathbf{Y}$$

$$(\mathbf{QR})^\top (\mathbf{QR})\beta = (\mathbf{QR})^\top \mathbf{Y}$$

$$\mathbf{R}^\top (\mathbf{Q}^\top \mathbf{Q})\mathbf{R}\beta = \mathbf{R}^\top \mathbf{Q}^\top \mathbf{Y}$$

$$\mathbf{R}^\top \mathbf{R}\beta = \mathbf{R}^\top \mathbf{Q}^\top \mathbf{Y}$$

$$(\mathbf{R}^\top)^{-1}\mathbf{R}^\top \mathbf{R}\beta = (\mathbf{R}^\top)^{-1}\mathbf{R}^\top \mathbf{Q}^\top \mathbf{Y}$$

$$\mathbf{R}\beta = \mathbf{Q}^\top \mathbf{Y}$$

```
QR <- qr(X)
betahat <- solve.qr(QR, y)
```

This gives us identical results to the lm function.