2.5. The Log-Transformation

Specific learning objectives:

2.5.1. Assess the need for a log-transformation of the response.
2.5.2. Implement a linear regression fit in R and interpret the results in terms of the problem at hand.

When to perform a Log transform

- When there is no linearity found in Y vs. X. May be found through
  - Lack of fit test. The Global F-Test may show lack of linearity when there could be another kind of functional relationship (e.g. quadratic, monotonic).
  - Scatter plots of Y vs. X show non-linear relationship.
  - Residual plots. E.g., Scatter plots of residuals show a trend other than random noise. The distribution of the residuals looks skewed, lack of constant variance, etc. (more on this on Residuals section.)
- Sometimes we simply know this by previous theoretical knowledge and/or experience of the phenomenon the governs the data.
- Lack of normality in dependent variable. When the normality assessment shows a right skewed distribution (via histogram, Q-Q Normal plots.)

Rules of logarithms:
log(X*Y) = log(X) + log(Y)
log(e^x)=X

An intrinsically linear function:

$$Y_i = \beta_0\ \exp(\beta_1 X_i)\ \varepsilon_i, \qquad \boxed{\varepsilon_i \sim \text{logNormal}}$$

which is transformed to a straight line by a logarithmic transformation:

$$\ln Y_i = \ln\beta_0 + \beta_1 X_i + \ln\varepsilon_i$$

or

$$Y_i' = \beta_0' + \beta_1 X_i + \varepsilon_i' \qquad \boxed{\ln\varepsilon_i = \varepsilon_i' \sim \text{Normal}}$$

Where the transformed errors are Normally independently distributed by the definition of the logNormal random variable:

$$\ln\varepsilon \sim \text{Normal} \quad \leftrightarrow \quad \varepsilon \sim \text{logNormal}$$

Rules of logarithms:
log(X*Y) = log(X) + log(Y)
log(e^x)=X

Example of intrinsically linear function

For a drug that has linear kinetics and elimination occurs from the central compartment then:

$$AUC_i = \frac{D_i \times F_i}{CL_i} \varepsilon_i,$$

$$\boxed{\varepsilon_i \sim \text{logNormal}}$$

where D is dose and CL/F is apparent oral clearance.

Transformation to a straight line by a logarithms gives:

$$\ln AUC_i = \ln D_i - \ln CL_i + \ln F_i + \varepsilon_i'$$

$$\boxed{\varepsilon_i' = \ln \varepsilon_i \sim \text{Normal}}$$

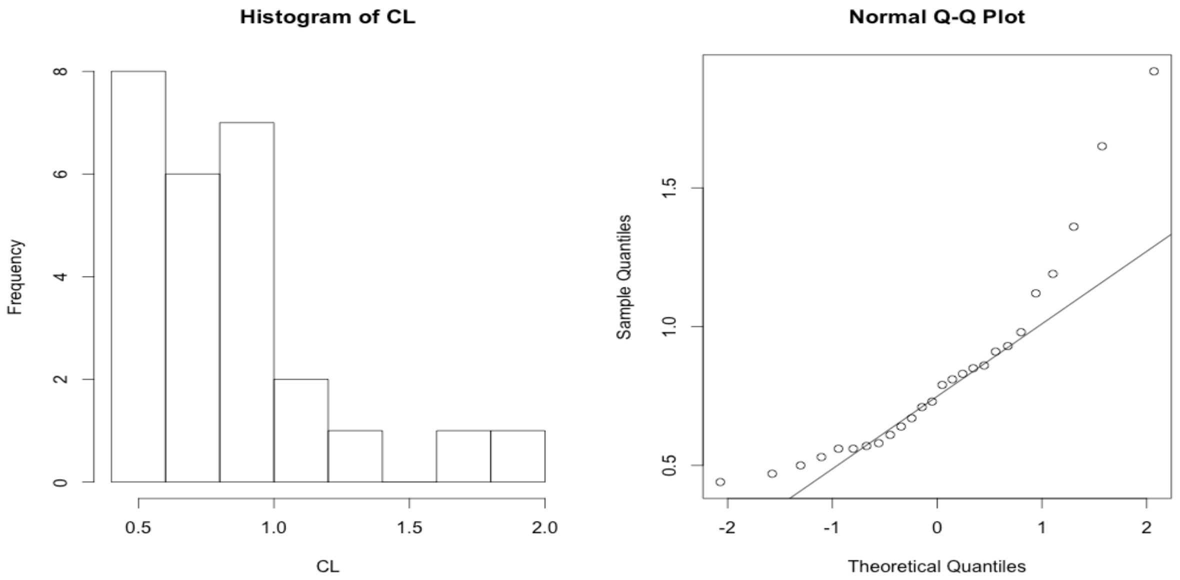
5-FU Example of Lack of Normality in Dependent Variable*

- Sample of 26 patients with advanced carcinomas.
- 5-Fluorouracil (5-FU) administered under a variety of doses and treatment schedules.
- Combination therapy with Methotrexate (MTX) was given once in 2-3 weeks.
- Serial blood samples were collected on Day 1 and **5-FU clearance** (L/min) was determined by non-compartmental analysis.
- Covariates measured:
  - Age (years),
  - Sex, (males=1, females=0)
  - BSA(m2),
  - 5-FU dose (mg), and
  - presence or absence of MTX.
- Of interest: determine whether a useful model relating 5-FU clearance and patient demographics could be developed for possible use in future individualized dosing regimens.

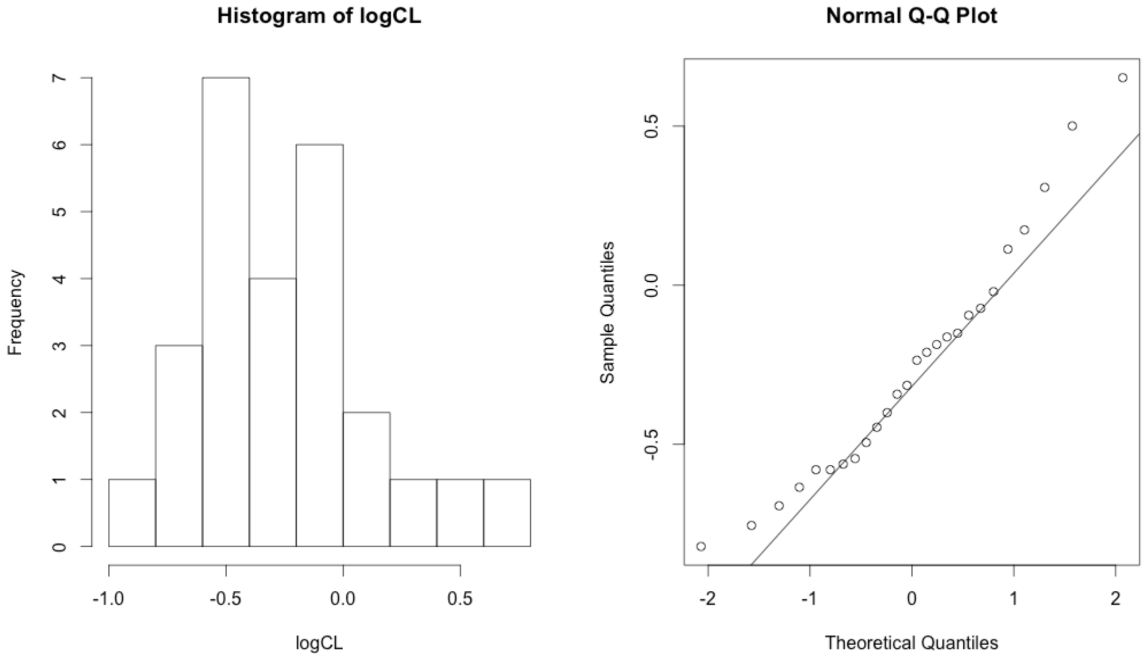
* Port et al. (1991) “Relative importance of dose, BSA, sex and age in 5-FU clearance”. Oncology. 48:227-281.

Data adapted from Pete Bonate (2nd ed)

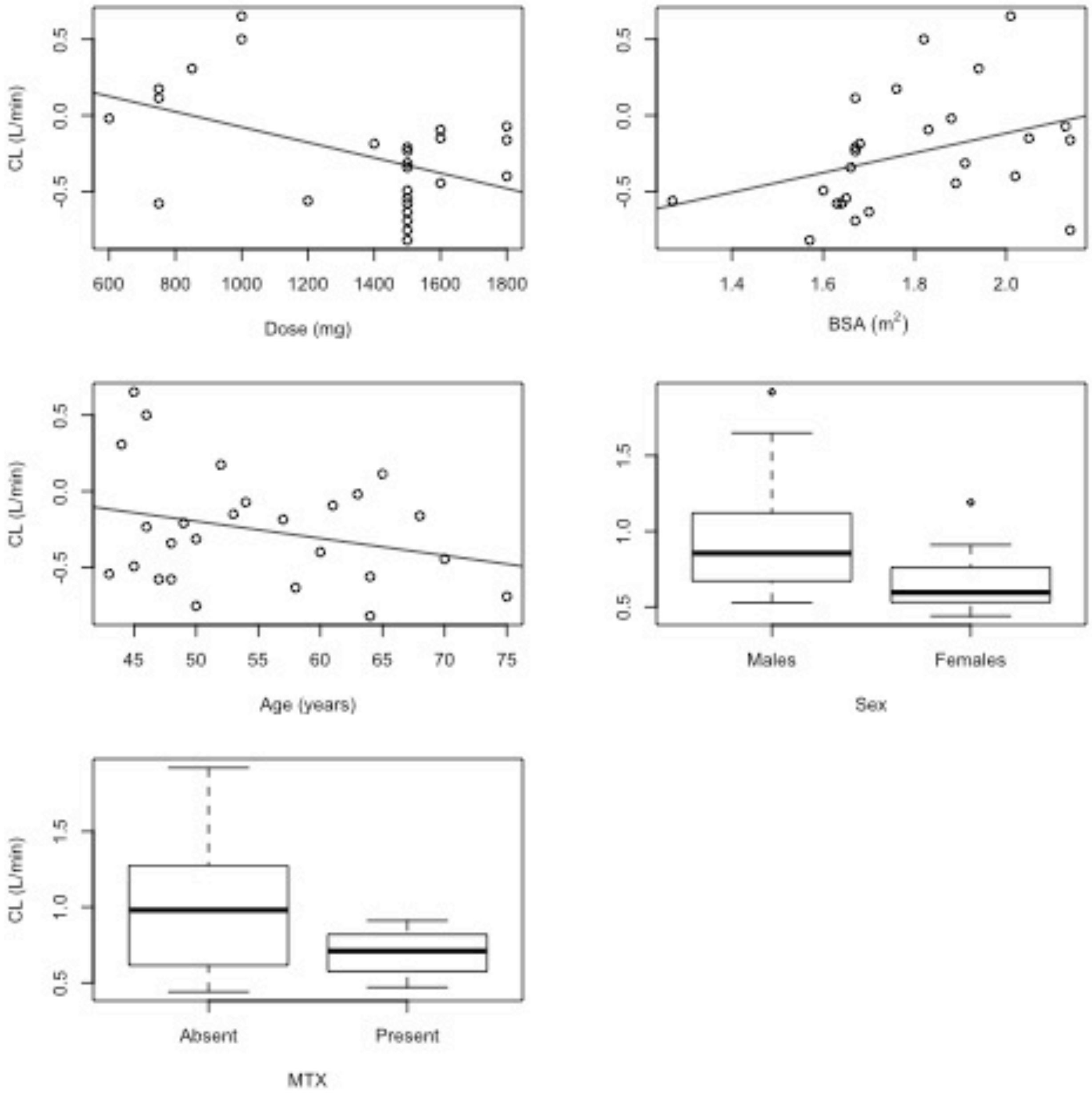
Raw CL data



Log-transformed CL data



Descriptive plots CL (L/min) vs. Covariates



R Code for previous slide’s array of plots

```
par(mfrow=c(3,2))    # plots drawn in an array of 3 cols x 2 rows.

par(mar=c(4,4,2,2))  # the number of lines of margin to be specified on
                     # the four sides of the plot: c(bottom, left, top, right)

plot(Dose,logCL,ylim=range(logCL),xlab="Dose (mg)",ylab="CL (L/min)")

abline(lm(logCL~Dose,data=dat))


plot(BSA,logCL,ylim=range(logCL),
                     xlab=expression(paste("BSA ",(m^2),sep=" ")),ylab="")
abline(lm(logCL~BSA,data=dat))


plot(Age,logCL,ylim=range(logCL), xlab="Age (years)",ylab="CL (L/min)")

abline(lm(logCL~Age,data=dat))


boxplot(CL~Sex,xlab="Sex", names=c("Males","Females"),data=dat)


boxplot(CL~MTX,xlab="MTX", names=c("Absent","Present"),
                          ylab="CL (L/min)",data=dat)
```

Results from simple linear regressions log 5-FU CL vs. potential covariates

```
              Estimate Std. Error Pr(>|t|) R2adj
(Intercept)     -0.092      0.092    0.324  0.000
Sex             -0.346      0.135    0.017  0.216

(Intercept)      0.367      0.453    0.426  0.000
Age             -0.011      0.008    0.179  0.074

(Intercept)     -1.416      0.619    0.031  0.000
BSA              0.649      0.343    0.071  0.130

(Intercept)      0.428      0.264    0.118  0.000
Dose            -0.001      0.000    0.014  0.228

(Intercept)     -0.076      0.107    0.481  0.000
MTX             -0.305      0.140    0.040  0.164
```

R Code to construct the table

Results from simple linear regressions of log 5-FU CL vs. potential covariates

```
# simple linear regressions by variable
# ---------------------------------------
fit.sex <- lm(logCL~ Sex,data=dat)
a <- cbind(summary(fit.sex)$coef[,c(1,2,4)],R2adj=summary(fit.sex)$r.squared)

fit.age <- lm(logCL~ Age,data=dat)
b <- cbind(summary(fit.age)$coef[,c(1,2,4)],R2adj=summary(fit.age)$r.squared)

fit.bsa <- lm(logCL~ BSA,data=dat)
c <- cbind(summary(fit.bsa)$coef[,c(1,2,4)],R2adj=summary(fit.bsa)$r.squared)

fit.dose <- lm(logCL~ Dose,data=dat)
d <- cbind(summary(fit.dose)$coef[,c(1,2,4)],R2adj=summary(fit.dose)$r.squared)

fit.mtx <- lm(logCL~ MTX,data=dat)
e <- cbind(summary(fit.mtx)$coef[,c(1,2,4)],R2adj=summary(fit.mtx)$r.squared)

ests <- rbind(a,b,c,d,e)
ests[,4] <- ests[,4]*rep(c(0,1),5)
round(ests,3)
```

Backwards elimination procedure
5-FU Clearance

```
> full.mod <- lm(logCL ~ Sex + Age + BSA + Dose + MTX, data=dat)
> summary(full.mod)

Call:
lm(formula = logCL ~ Sex + Age + BSA + Dose + MTX, data = dat)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.50699 -0.13936  0.01754  0.15127  0.47805 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)  
(Intercept)  0.1037931  0.6962062   0.149   0.8830  
Sex         -0.2465598  0.1230943  -2.003   0.0589 .
Age         -0.0098008  0.0064744  -1.514   0.1457  
BSA          0.5439696  0.3217032   1.691   0.1064  
Dose        -0.0004790  0.0002118  -2.262   0.0350 *
MTX         -0.0608639  0.1459008  -0.417   0.6810  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2763 on 20 degrees of freedom
Multiple R-squared:  0.575, Adjusted R-squared:  0.4688 
F-statistic: 5.412 on 5 and 20 DF,  p-value: 0.002624
```

Backwards elimination procedure, 5-FU Clearance

```
> red.mod1 <- update(full.mod,.~. -MTX)
> summary(red.mod1)

Call:
lm(formula = logCL ~ Sex + Age + BSA + Dose, data = dat)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.53581 -0.10467  0.02892  0.13430  0.46014 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)   
(Intercept)  0.0244981  0.6564512   0.037  0.97058   
Sex         -0.2462768  0.1206474  -2.041  0.05399 . 
Age         -0.0090441  0.0060916  -1.485  0.15249   
BSA          0.5877800  0.2980382   1.972  0.06190 . 
Dose        -0.0005354  0.0001597  -3.351  0.00302 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1 

Residual standard error: 0.2708 on 21 degrees of freedom
Multiple R-squared:  0.5713,    Adjusted R-squared:  0.4897 
F-statistic: 6.997 on 4 and 21 DF,  p-value: 0.0009605
```

Backwards elimination procedure, 5-FU Clearance

```
> red.mod2 <- update(red.mod1,.~. -Age)
> summary(red.mod2)

Call:
lm(formula = logCL ~ Sex + BSA + Dose, data = dat)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.51925 -0.19700  0.04354  0.12039  0.46258 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)   
(Intercept) -0.5224608  0.5580009  -0.936   0.3593   
Sex         -0.2191216  0.1224733  -1.789   0.0874 . 
BSA          0.6428539  0.3037065   2.117   0.0458 * 
Dose        -0.0005799  0.0001611  -3.599   0.0016 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1 

Residual standard error: 0.2781 on 22 degrees of freedom
Multiple R-squared:  0.5263,    Adjusted R-squared:  0.4617 
F-statistic: 8.148 on 3 and 22 DF,  p-value: 0.0007835
```

Backwards elimination procedure, 5-FU Clearance

```
> red.mod3 <- update(red.mod2,.~. -Sex)
> summary(red.mod3)

Call:
lm(formula = logCL ~ BSA + Dose, data = dat)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.71547 -0.11646  0.04638  0.15246  0.51302 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)   
(Intercept) -1.0036257  0.5117744  -1.961  0.06209 . 
BSA          0.8864430  0.2841725   3.119  0.00482 **
Dose        -0.0006219  0.0001669  -3.727  0.00111 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1 

Residual standard error: 0.2911 on 23 degrees of freedom
Multiple R-squared:  0.4574,    Adjusted R-squared:  0.4102 
F-statistic: 9.694 on 2 and 23 DF,  p-value: 0.0008842
```

The 5-FU Clearance Final Model

$$\ln\ CL_i = \ln \beta_0 + \beta_1 BSA_i + \beta_2 Dose_i + \ln \varepsilon_i$$

or

$$\ln\ CL_i = \beta_0' + \beta_1 BSA_i + \beta_2 Dose_i + \varepsilon_i'$$

which is transformed back by taking anti-logs:

$$\boxed{CL = \beta_0' e^{\beta_1 BSA_i + \beta_2 Dose_i} \varepsilon_i',\ i = 1,\ldots,26.}$$

And the estimated model is written as:

$$\ln \hat{CL}_i = -1.004 + 0.886\ BSA_i - 0.000622\ Dose_i$$

$$\hat{CL}_i = \exp(-1.004 + 0.886\ BSA_i - 0.000622\ Dose_i)$$

$$\boxed{\hat{CL}_i = 0.366\ \exp(0.886\ BSA_i - 0.000622\ Dose_i)}$$

Forward selection procedure
5-FU Clearance

Recall previous results from simple linear regressions
log 5-FU CL vs. potential covariates

```
             Estimate  Std. Error Pr(>|t|) R2adj
(Intercept)   -0.092      0.092    0.324     -
Sex           -0.346      0.135    0.017   0.216

(Intercept)    0.367      0.453    0.426     -
Age           -0.564      0.408    0.179   0.074

(Intercept)   -1.416      0.619    0.031     -
BSA            1.188      0.628    0.071   0.130

(Intercept)    0.428      0.264    0.118     -
Dose          -0.505      0.190    0.014   0.228

(Intercept)   -0.076      0.107    0.481     -
MTX           -0.305      0.140    0.040   0.164
```

Forward selection procedure, 5-FU Clearance

Results from separately fitting models with:

Dose + each remaining variable

```
        Estimate Std. Error t value Pr(>|t|) R2.adj
sex     -0.335       0.117  -2.855    0.009  0.430
age     -0.420       0.375  -1.118    0.275  0.268
bsa      1.622       0.520   3.119    0.005  0.457
mtx     -0.152       0.163  -0.933    0.360  0.256
```

Dose + BSA + each remaining variable

```
       Estimate Std. Error t value Pr(>|t|) R2.adj
sex     -0.219       0.122  -1.789    0.087  0.526
age     -0.358       0.322  -1.112    0.278  0.486
mtx     -0.008       0.151  -0.056    0.956  0.457
```

At a 5% level of significance, there are no other variables that are significant.

The final model, same as with Backwards Elimination:

$$\ln\ CL_i = \ln\beta_0 + \beta_1 BSA_i + \beta_2 Dose_i + \ln\varepsilon_i$$

Forward selection procedure, 5-FU Clearance

R Code to obtain results from separately fitting models with Dose + each remaining variable

```
fit.sex <- lm(logCL ~ Dose + Sex, data=dat)
fit.age <- lm(logCL ~ Dose + Age, data=dat)
fit.bsa <- lm(logCL ~ Dose + BSA, data=dat)
fit.mtx <- lm(logCL ~ Dose + MTX, data=dat)

ests.w.dose <- rbind(
summary(fit.sex)$coef[3,],
summary(fit.age)$coef[3,],
summary(fit.bsa)$coef[3,],
summary(fit.mtx)$coef[3,])

R2.adj <- c(
  summary(fit.sex)$r.squared,
  summary(fit.age)$r.squared,
  summary(fit.bsa)$r.squared,
  summary(fit.mtx)$r.squared)

mat.ests.w.dose <- cbind(ests.w.dose,R2.adj)
dimnames(mat.ests.w.dose)[[1]] <- c("sex","age","bsa","mtx")

round(mat.ests.w.dose,3)
```

Interpretation of covariate model coefficients
5-FU CL Example

$$\ln \overset{\wedge}{CL}_i = -1.004 + 0.886\ BSA_i - 0.000622\ Dose_i$$

$$\overset{\wedge}{CL}_i = 0.367 e^{0.886\ BSA_i} e^{-0.000622\ Dose_i}$$

ON THE EFFECT OF BSA

The estimated coefficient of BSA is $\hat{\beta}_1 = 0.886$ so we would say that:

On average and while holding Dose constant, a one unit increase in BSA would result in a significant increase of exp(0.886) = 2.425 times the value in CL (p-val=0.005).

Or, by stating a percent change of BSA in CL:

On average and while holding Dose constant, a one unit increase in BSA would result in a significant percent increase in CL of 142.5% (p-value=0.005.)

$$\left(e^{\hat{\beta}_1} - 1\right) \times 100\% = 142.5\%$$

Interpretation of covariate model coefficients 5-FU CL Example

$$\ln \hat{CL}_i = -1.004 + 0.886\ BSA_i - 0.000622\ Dose_i$$

$$\hat{CL}_i = 0.367 e^{0.886\ BSA_i} e^{-0.000622\ Dose_i}$$

ON THE EFFECT OF DOSE

On average and while holding BSA constant, a one unit increase in BSA would result in a significant decrease of exp(-0.000622) = 0.999 times the value of CL (p-val=0.005).

On average and while holding BSA constant, a one unit increase in Dose would result in a significant 0.062% decrease in CL (p-val=0.001), since:

$$\left(e^{\hat{\beta}_2} - 1\right) \times 100\% = -0.0622\%$$

Is this reduction clinically significant?

Model Interpretation, 5-FU Data Example

Why percent change?

Our model is: $\ln CL = \beta_0 + \beta_1 BSA + \beta_2 Dose$

Model with one unit increase in BSA: $\ln CL^{+} = \beta_0 + \beta_1 (BSA + 1) + \beta_2 Dose$

We subtract the models: $\ln CL^{+} - \ln CL = \beta_1$

Is the change in CL for a one unit increase in BSA as in linear regression.

Note that: $\ln CL^{+} - \ln CL = \ln \frac{CL^{+}}{CL}$

Taking the antilog on both sides: $\exp\left(\ln \frac{CL^{+}}{CL}\right) = e^{\beta_1}$

$$\frac{CL^{+}}{CL} = e^{\beta_1}$$

Model Interpretation, 5-FU Data Example

Why percent change?

On average, the estimated percent change in CL for one unit increase in BSA while holding Dose fixed is:

$$\left(\frac{\widehat{CL}^{+}}{\widehat{CL}} - 1 \right) \times 100\% = \left(e^{\hat{\beta}_1} - 1 \right) \times 100\%$$

The difference in ***arithmetic means in lnCL*** for a one unit change in BSA

$$\ln \widehat{CL}^{+} - \ln \widehat{CL} = \hat{\beta}_1$$

The ratio of ***geometric means in CL*** for a one unit change in BSA

$$\frac{\widehat{CL}^{+}}{\widehat{CL}} = e^{\hat{\beta}_1}$$

While holding Dose constant.

Arithmetic Mean (AM) of lnCL : $\frac{1}{n}\sum_{i=1}^{n} \ln CL_i$

Geometric Mean (GM) of CL : $\left(\prod_{i=1}^{n} CL_i\right)^{1/n}$

Model Interpretation, 5-FU Data Example

Geometric Mean vs. Arithmetic Mean

$$\text{Arithmetic Mean (AM) of lnCL}: \quad \frac{1}{n}\sum_{i=1}^{n} \ln CL_i$$

$$\text{Geometric Mean (GM) of CL}: \quad \left(\prod_{i=1}^{n} CL_i\right)^{1/n}$$

- The GM is yet another measure of central tendency, employed with skewed distributions.
- The AM is sensitive to extreme values so not informative with skewed distributions.
- The GM = Median for the theoretical LogNormal distribution.
- The relationship between GM and AM is:

$$\ln\ GM(CL) = AM(\ln CL)$$

$$\ln\left(\prod_{i=1}^{n} CL_i\right)^{1/n} = \frac{1}{n}\sum_{i=1}^{n} \ln CL_i$$

Interpretation of scaled covariate model coefficients
5-FU CL Example

1. Suppose that prior to fitting the model, BSA and Dose are standardized (Bonate, 2011):

   BSA is standardized to BSA* = BSA / 1.83
   Dose is standardized to Dose* = Dose / 1000

2. And we fit the model with BSA* and Dose*:

$$\ln \hat{CL}_i = -1.004 + 1.622\, BSA^*{}_i - 0.622\, Dose^*{}_i$$

$$\hat{CL}_i = 0.367\, e^{1.622\, BSA^*{}_i}\, e^{-0.622\, Dose^*{}_i}$$

- One unit increase in BSA* or 1.83 m^2 increase in BSA leads to 406% increase in CL
- One unit increase in Dose* or 1000 mg increase in Dose leads to 46% reduction in CL

I.e. BSA=1.83 m^2 -> BSA*=1. Dose=1000 mg -> Dose*=1

Interpretation of the intercept

- The intercept becomes interesting when continuous variables are centered or dummy variables are in the model.

- Linear scale interpretation: - just like in usual linear regression -

  β_0 represents the ***arithmetic mean of lnY*** when the predictors X are zero, can be useful in some cases where X=0 makes sense (e.g., dummy variables, centered continuous variables).

- Non-linear scale interpretation:

  $e^{\beta 0}$ represents the ***geometric mean of Y*** when the predictors X are zero (or e^X=1 with X=0) also equivalent to the median of Y.

Interpretation of the intercept, 5-FU CL example

Fitting a intercept-only model with lm() on lnCL is equivalent to calculating the arithmetic mean of lnCL:

```
> summary(lm(logCL~1,data=dat))
. . .
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.25208    0.07435   -3.39  0.00232 **
---

> # arithmetic mean of logCL
> mean(dat$logCL)
[1] -0.252081


> # geometric mean of CL
> exp(-0.251)
[1] 0.7780224

> # median of CL
> median(dat$CL)
[1] 0.76
```

For practical purposes, we can interpret the geometric mean of CL as the median of CL.

Interpretation of coefficients with one dummy variable.

logCL ~ Sex, Sex=1 if female

The model for females is: $\ln CL^F = \beta_0 + \beta_1$

The model for males is: $\ln CL^M = \beta_0$

We subtract the models:

$$\ln CL^F - \ln CL^M = \ln \frac{CL^F}{CL^M} = \beta_1$$

Taking the antilog on both sides:

$$\exp\left(\ln \frac{CL^F}{CL^M}\right) = e^{\beta_1}$$

$$\frac{CL^F}{CL^M} = e^{\beta_1}$$

The estimated percent change in CL for females vs. males is:

$$\left(\frac{\overset{\wedge}{CL^F}}{\overset{\wedge}{CL^M}} - 1\right) \times 100\% = \left(e^{\hat{\beta}_1} - 1\right) \times 100\%$$

Interpretation of coefficients with one dummy variable.

```
> summary(lm(logCL~Sex,data=dat))
. . .
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.09219    0.09157  -1.007   0.3241
Sex2females -0.34642    0.13479  -2.570   0.0168 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
. . .
```

$\exp(\hat{\beta}_0)$ is the estimated geometric mean for the **male** patients group.

$$\hat{CL}^{males} = \exp(-0.092) = 0.912$$

$\exp(\hat{\beta}_1)$ for **females** is the ratio of the estimated GM for the female group over the estimated GM for the male group.

$$\hat{CL}^{females} = \exp(\hat{\beta}_0 + \hat{\beta}_1) = 0.645 \qquad \frac{\hat{CL}^{females}}{\hat{CL}^{males}} = \frac{0.645}{0.912} = 0.707 = \exp(\hat{\beta}_1)$$

CL for females will be ~30% lower than for males:

$(e^{-0.346}-1)*100\% = -29.3\%$.

Comments on transforming covariates
Centering/standardizing/scaling

- Centering can be done based on the researcher’s interest, with respect to the mean, median or any particular value of research relevance. In this case the intercept makes sense for the new centered covariate = 0.

- Scaling can be done to ease interpretation with respect to the scale of a variable. E.g. to express the effect of Dose in 1000 mg rather than 1 mg.

- Will change the estimated model coefficients (β’s) but will not change the p-values, hence the conclusions of the study remain the same.

- Is to be done in practice in the beginning of the modeling process, that is, since it may aid in decision making during the variable selection (not done here for illustration purposes).

- It may also be used as a remedy for collinearity (more on this later).

2.6. Collinearity

Specific learning objectives:

2.6.1. Perform a collinearity assessment in R via:

a) Correlations and tests.
b) Variance inflation factor.
c) Condition number.

2.6.2. Apply methods for mitigating collinearity.

Collinearity

- AKA multicollinearity or ill-conditioning.

- “Collinear” implies that there is correlation or linear dependencies among the independent variables. Suppose that:

  - x_1 and x_2 are regressed against Y, and

  - x_1 and x_2 are correlated, such that x_2 does not provide any more information than x_1, and viceversa.

  - as cor($x_{1,}$ x_2) increases it becomes more difficult to isolate the effect due to x_1 from the effect due to x_2, such that the parameters estimates become unstable.

- i.e., the parameter estimates become extremely sensitive to small changes in the X values and depend on the particular data set that generated them.

- There are complex geometric reasons for its effect on parameter estimation, please refer to Bonate for details.

(Has to do with inversion of matrices, mainly the problem of singularities, like dividing 1/x where x is almost 0).

In summary…

Causes:

- Subset of the predictors are highly correlated, effects are difficult to isolate.
- Influential observations, i.e., outliers.
- Poor scaling of covariates (leading to numbers close to zero).

Effects on estimates:

- Too sensitive to changes in which predictors/observations are included in the model
- Vary greatly from one data set to another, defeating the scientific purpose of reproducibility and are poor predictive tools.

Collinearity

How to detect (clues):

- Variables that are expected to be important are not found statistically significant.
- Estimates change drastically if a subject or covariate is discarded (e.g., change in sign).
- Inflated variance estimates.
- Order in which covariates are excluded from the model during the variable selection process affects their significance.
- Collinearity diagnostics (more on this).

Collinearity diagnostics

- Sample correlation matrix between covariates.

- Variance inflation factor (post-modeling):

$$VIF = \frac{1}{1-R_i^{\,2}} \left\{ \begin{array}{l} >5 : \text{possible} \\ >10 : \text{almost certain} \end{array} \right.$$

Where R_i^2 is the coefficient of determination of X_i regressed against all other X.

Collinearity diagnostics

- Condition number (K) defined as the ratio of the largest to the smallest eigenvalues of the correlation matrix.

$$K = \frac{l_l}{l_p}$$

Used in R, see ?kappa() function.
Example of usage:

```
mod.mat <- model.matrix(~ x1 + x2 + x3)
kappa(mod.mat)
```

Where l_l and l_p are the largest and smallest eigenvalues of the correlation matrix.

K = 1 means perfect stability; K → ∞ means perfect instability.

The difficulty with the use of the condition number is that it fails to identify which columns are collinear and simply indicates that collinearity is present.

Bonate's guideline:

$K < 10^4$: no collinearity

$10^4 < K < 10^6$: moderate collinearity

$K > 10^6$: severe collinearity

Some collinearity remedies:

- Transforming covariates: these remedies have to do with avoiding singularities when inverting a matrix (i.e. 1/0)

Centering to the mean $X^{*}_{ji} = X_{ji} - \overline{X}_{j}$

Sample mean of j-th covariate

Scaling to the mean $X^{*}_{ji} = X_{ji} / \overline{X}_{j}$

Standardizing

$$X^{*}_{ji} = \frac{X_{ji} - \overline{X}_{j}}{s_{j}}$$

Standard deviation of j-th covariate

- Using surrogate variables (e.g. use BSA as a function of weight and height)

Note: Ridge regression and Principal Component Analysis are more sophisticated remedies however will not be covered here.

Using surrogate variables

Example: CL vs. Weight and Height

Bonate, 2011

- Height and weight are often highly correlated

- Body Surface Area (BSA): a measure of the overall surface area on an individual, computed based on Weight and Height:

$$\mathrm{BSA} = 0.0235 \times \left(\mathrm{Wt}^{0.51456}\right) \times \left(\mathrm{Ht}^{0.42246}\right)$$

- Data: apparent oral clearance, weight and height was obtained from 65 individuals.

Using surrogate variables, Example: CL vs. Weight and Height. Bonate, 2011

Results from a Pearson type correlation and significance test
CL vs. Wt vs. Ht

```
> dat <- read.csv("Data/ClWtHt.csv")
> rcorr(as.matrix(dat[,1:3]),type="pearson")
     CL   Wt   Ht
CL 1.00 0.57 0.23
Wt 0.57 1.00 0.55
Ht 0.23 0.55 1.00

n= 65


P
   CL     Wt     Ht    
CL        0.0000 0.0635
Wt 0.0000        0.0000
Ht 0.0635 0.0000       
```

H_0: X,Y are not correlated
H_1: X,Y are correlated

p-val=0.000 => Very small p-values
And we reject H_0.

Pearson Population correlation: $\rho_{X,Y} = \dfrac{Cov(X,Y)}{\sigma_X \sigma_Y}$

Pearson Sample correlation: $r_{X,Y} = \dfrac{\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{s_X s_Y}$

Using surrogate variables, Example: CL vs. Weight and Height. Bonate, 2011

```
> fit.wt <- lm(CL ~ Wt,data=dat)
> summary(fit.wt)
. . .
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 19595.31    6535.10   2.998  0.00388 **
Wt            248.77      45.51   5.466 8.38e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6486 on 63 degrees of freedom
Multiple R-squared:  0.3217,  Adjusted R-squared:  0.3109
F-statistic: 29.88 on 1 and 63 DF,  p-value: 8.385e-07
```

```
> fit.ht <- lm(CL ~ Ht,data=dat)
> summary(fit.ht)
. . .
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  17536.5    19877.6   0.882   0.3810
Ht             539.9      285.8   1.889   0.0635 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7661 on 63 degrees of freedom
Multiple R-squared:  0.05361, Adjusted R-squared:  0.03859
F-statistic: 3.569 on 1 and 63 DF,  p-value: 0.06347
```

Using surrogate variables, Example: CL vs. Weight and Height. Bonate, 2011

```
> fit.both <- lm(CL ~ Wt + Ht, data=dat)
> summary(fit.both)
. . .
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 34809.80   17179.54   2.026    0.047 *
Wt            277.81      54.71   5.078 3.74e-06 ***
Ht           -278.56     290.86  -0.958    0.342
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6490 on 62 degrees of freedom
Multiple R-squared:  0.3316,  Adjusted R-squared:   0.31
F-statistic: 15.38 on 2 and 62 DF,  p-value: 3.77e-06
```

First sign of collinearity of Wt vs. Ht:
The effect of Ht is now negative.

Condition number: (10^4 < K < 10^6 : moderate collinearity)

```
> mod.mat <- model.matrix(CL~Wt+Ht,data=dat)
> kappa(mod.mat)
[1] 2579.437
```

Using surrogate variables, Example: CL vs. Weight and Height. Bonate, 2011

Fitting surrogate variable BSA

```
> dat$bsa <- 0.0235 * dat$Wt^0.51456 * dat$Ht^0.42246
> fit.bsa <- lm(CL ~ bsa, data=dat)
> summary(fit.bsa)
. . .

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)     1695      10848   0.156    0.876
bsa            29536       5988   4.933 6.23e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6689 on 63 degrees of freedom
Multiple R-squared:  0.2786,  Adjusted R-squared:  0.2672
F-statistic: 24.33 on 1 and 63 DF,  p-value: 6.233e-06
```

Note R^2_{adj}=0.31 if fit with Wt and Ht vs. R^2_{adj}=0.27 with BSA.

2.7. Residual Checks and outliers

Specific learning objectives:

2.7.1. State the assumptions of the model.
2.7.2. Identify the types of plots needed to perform the residual checks.
2.7.3. Implement a residual check via R and interpret it.
2.7.4. Implement plots and measures to identify outliers and influential points in R.

Residual Checks

Major assumptions in linear regression:

1. The relationship between the response and the regressors is linear, at least approximately.
2. The error term has zero mean
3. The error term has constant variance (homoscedastic)
4. The errors are uncorrelated
5. The errors are normally distributed

Gross violations lead to an unstable model: a different sample could lead to a totally different model with opposite conclusions.

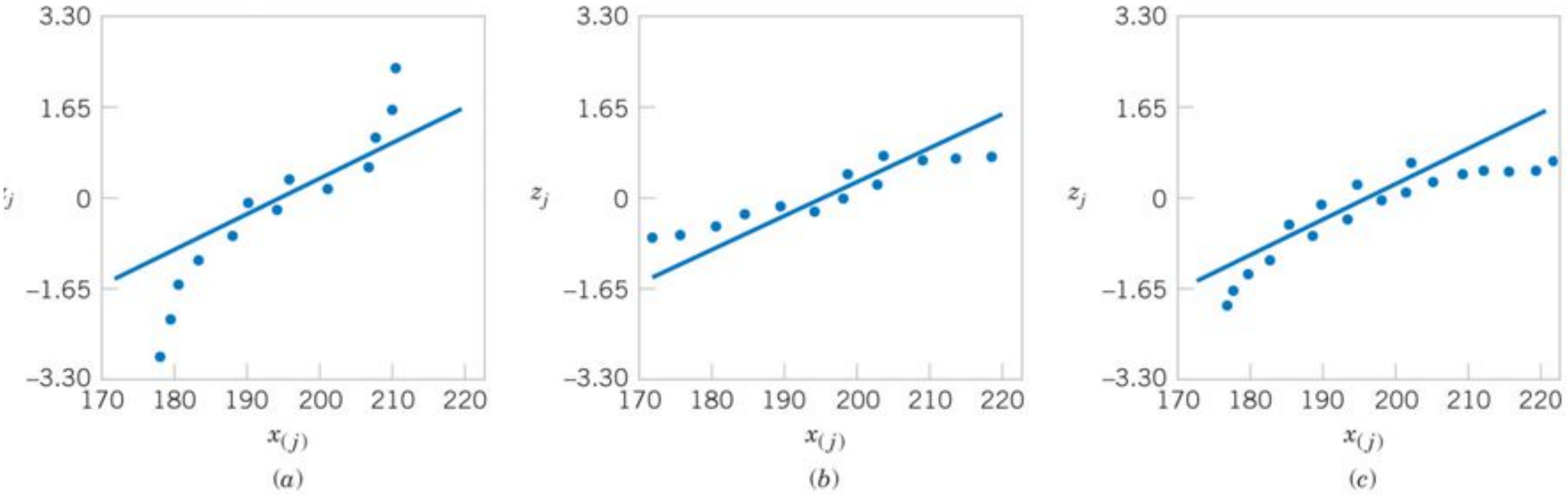
Residual plots

- Normal Quantile – Quantile plot
- Residuals vs. fitted values
- Residuals vs. covariates
- Residuals in time sequence

Quantile-Quantile Normal Residual Plot

- Plot to compare the distribution of residuals vs. the Standard Normal
- It should look like a straight line which is usually determined visually,

with emphasis on the central values.



Light tailed

Heavy tailed

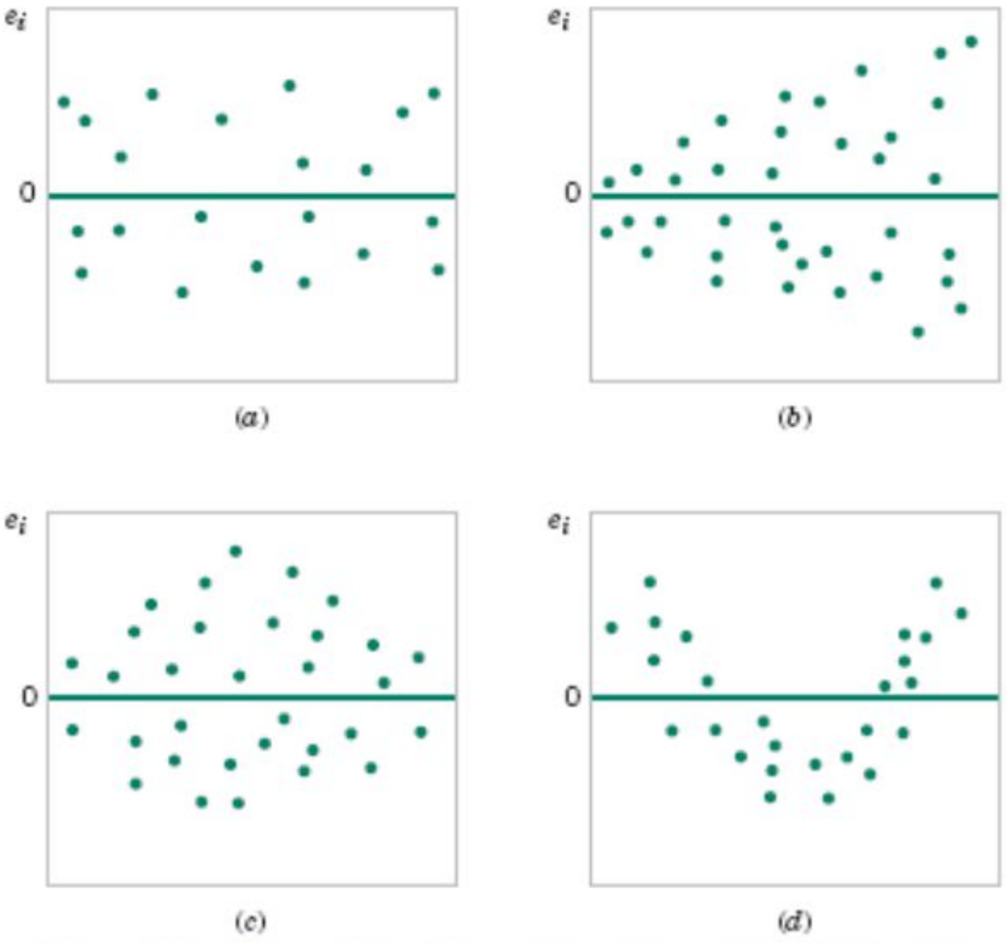
Right skewed

Small sample sizes (n<16) often produce plots that deviate substantially from normality. Larger sample sizes (n>32) are better behaved.

© John Wiley & Sons, Inc. *Applied Statistics and Probability for Engineers*, by Montgomery and Runger.

Residuals vs. Fitted Values

- Used to check for constant variance of residuals.
- Residuals should be contained within a horizontal band around the value of zero as in (a) below.



(a)

(b)

(c)

(d)

(b) Outward/inward-opening funnel pattern (variance increasing/ decreasing function of Y)

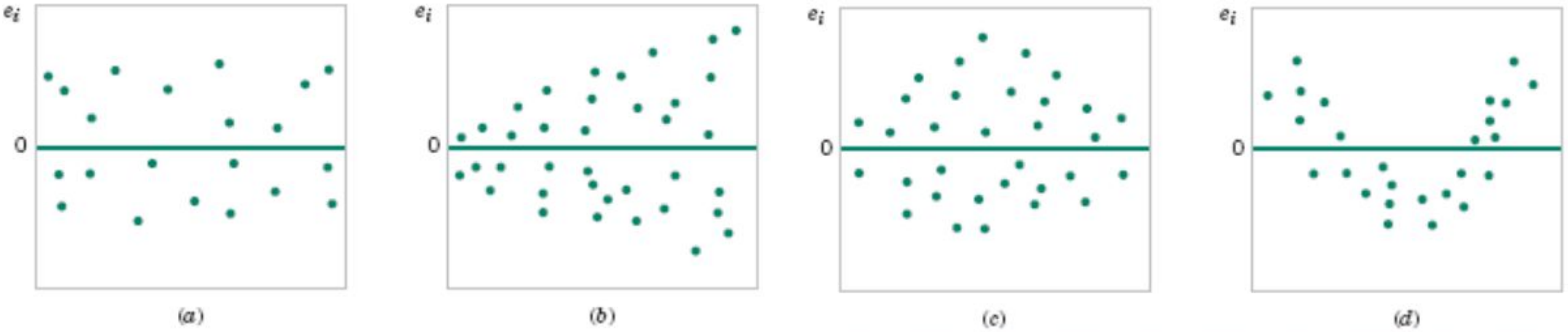
(c) Doble-bow pattern often when Y is a proportion between 0 and 1.

(d) Curved plot: indicates non-linearity. Other covariates could be needed in the model, e.g. a squared term.

Adapted from Montgomery et al. 2012.

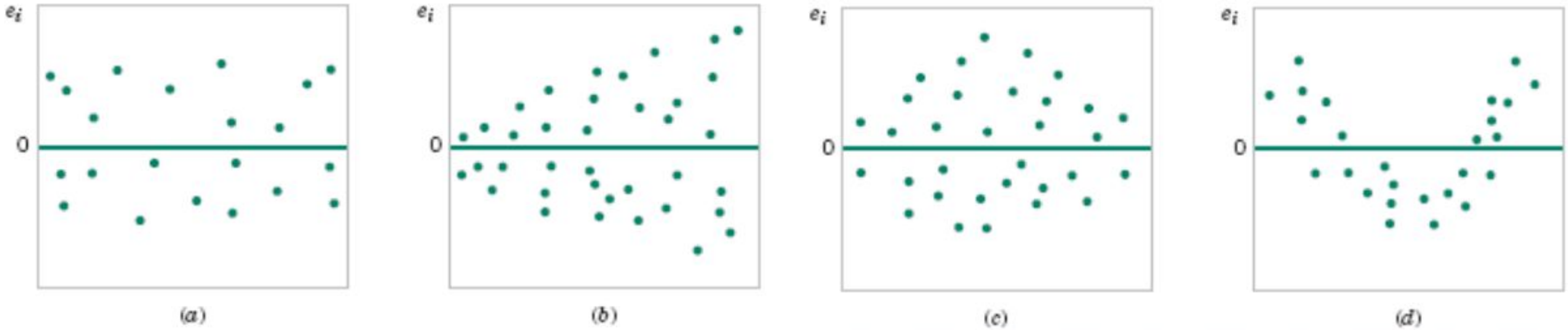
Residuals vs. Covariates
For multiple linear regression

- Used to check for constant variance and linearity of Y vs. each covariate separately.
- Similar patterns as in residuals vs. fitted only the horizontal scale corresponds to a covariate.
- An impression of a horizontal band is desirable, patterns indicative of non-constant variance as explained earlier apply.



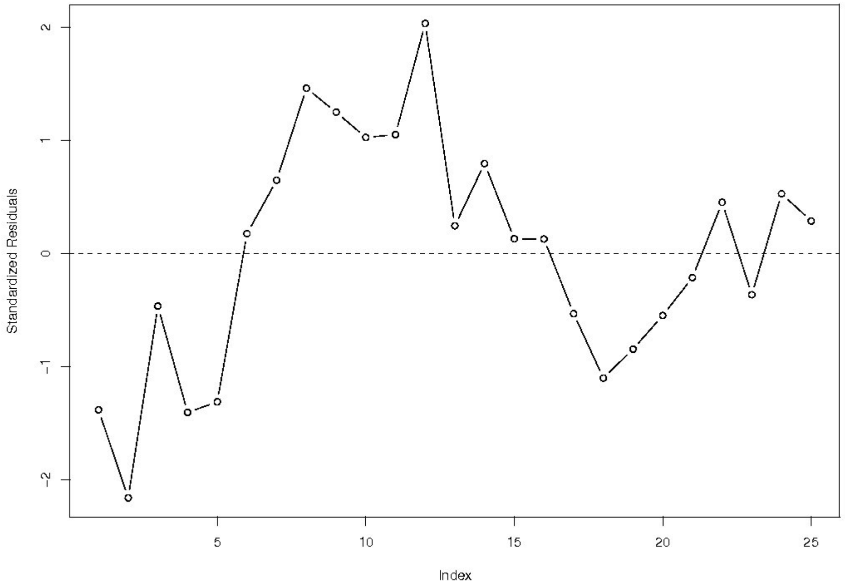
Residuals in time sequence plots

- Useful when time sequence in which data were collected is known.
- Random fluctuation around zero should look like a horizontal band (a).
- Any departure or pattern may be indicative of heterogeneous variance (linear or quadratic terms should be added to the model) – (b-d),



Residuals in time sequence plots

- Departures may suggest autocorrelation of residuals, i.e., positive or negative correlation.
- Potentially serious violation of the independence assumption.



How to access residuals and fitted values in R

```
red.mod3 <- lm(logCL ~ BSA + Dose, data=dat)
```

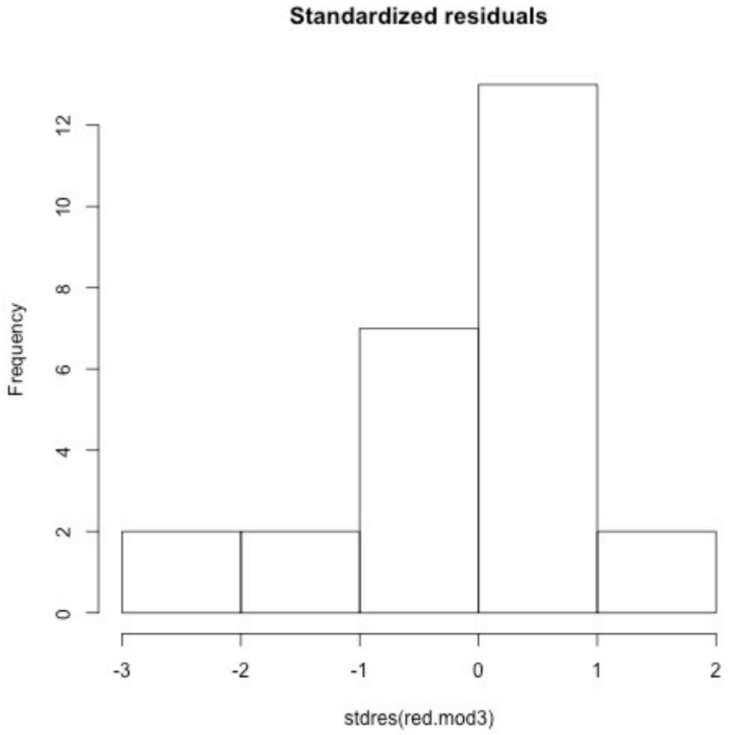
Raw residuals	`residuals(red.mod3)`
	`red.mod3$res`

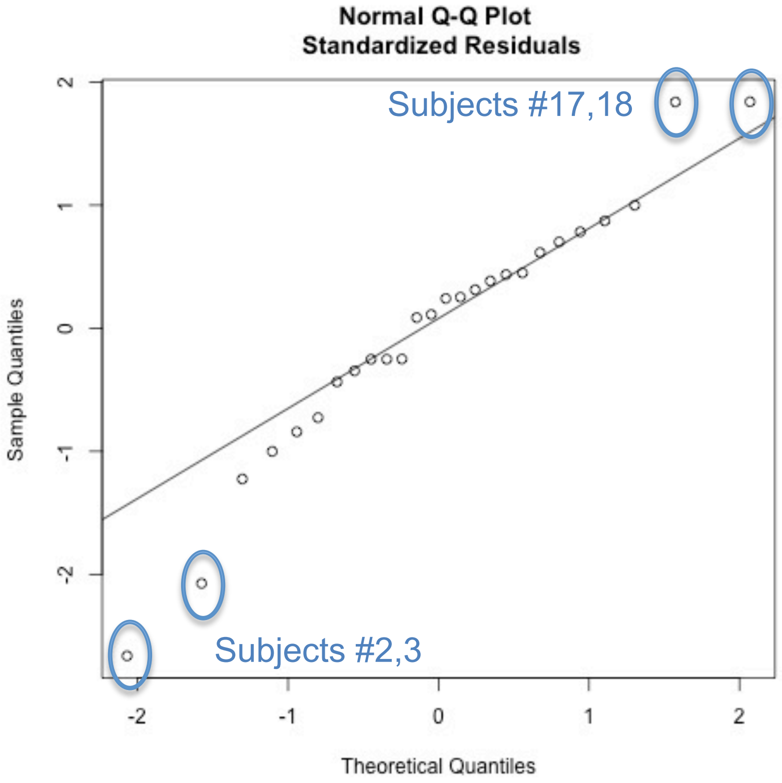
Standardized residuals	`stdres(red.mod3)`
Studentized residuals	`studres(red.mod3)`

Fitted Y's	`fitted(red.mod3)`
	`red.mod3$fitted`

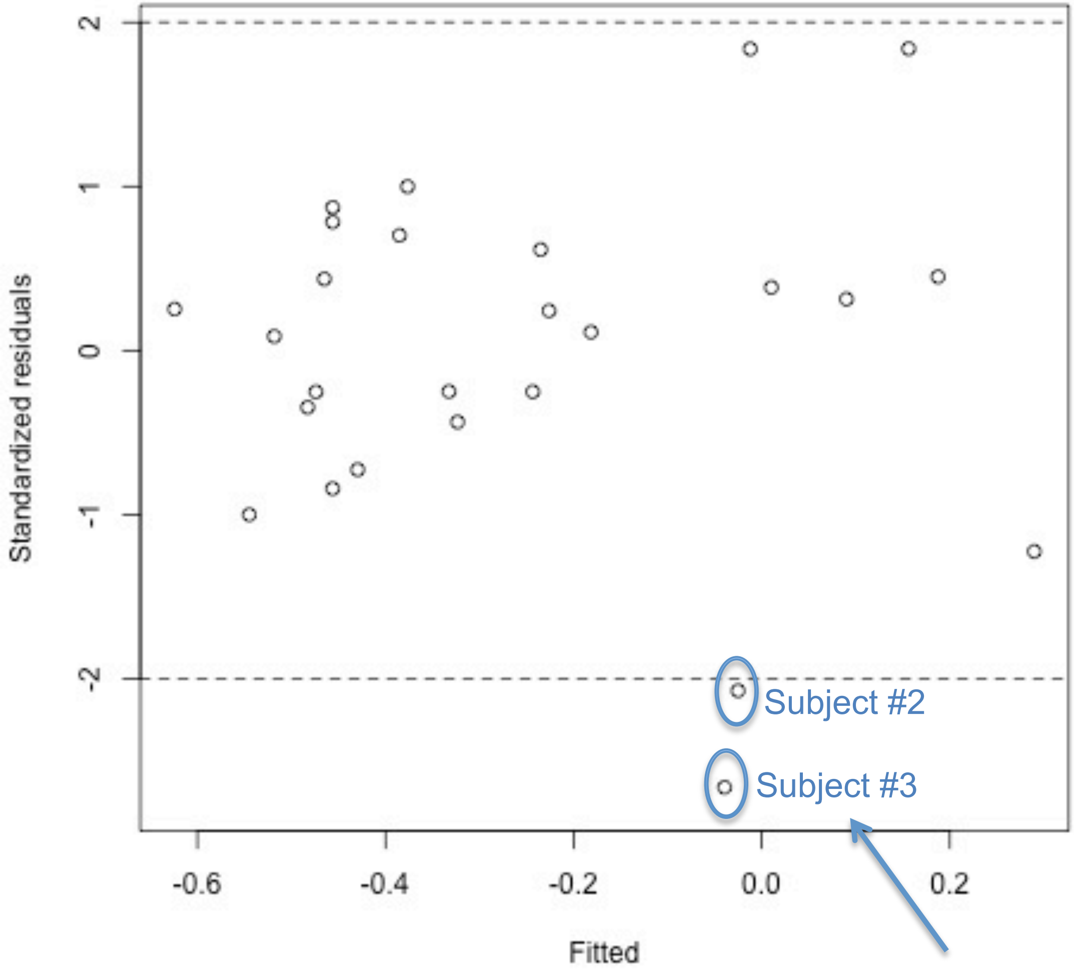
Example 5-FU CL

- The distribution of standardized residuals are usually easier to interpret.
- Since we assume that residuals are ~N(0,σ^2), standardized residuals will be ~N(0,1). In regression, MSE is an estimate of σ^2.



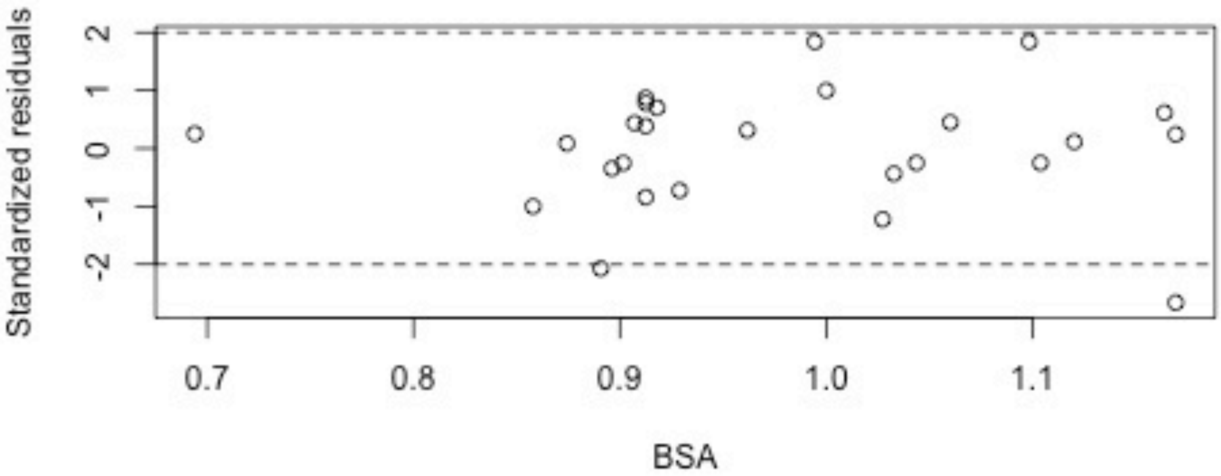


```
hist(stdres(red.mod3),main="Standardized residuals”)
qqnorm(stdres(red.mod3),main="Normal Q-Q Plot \n Standardized Residuals")
qqline(stdres(red.mod3))
```

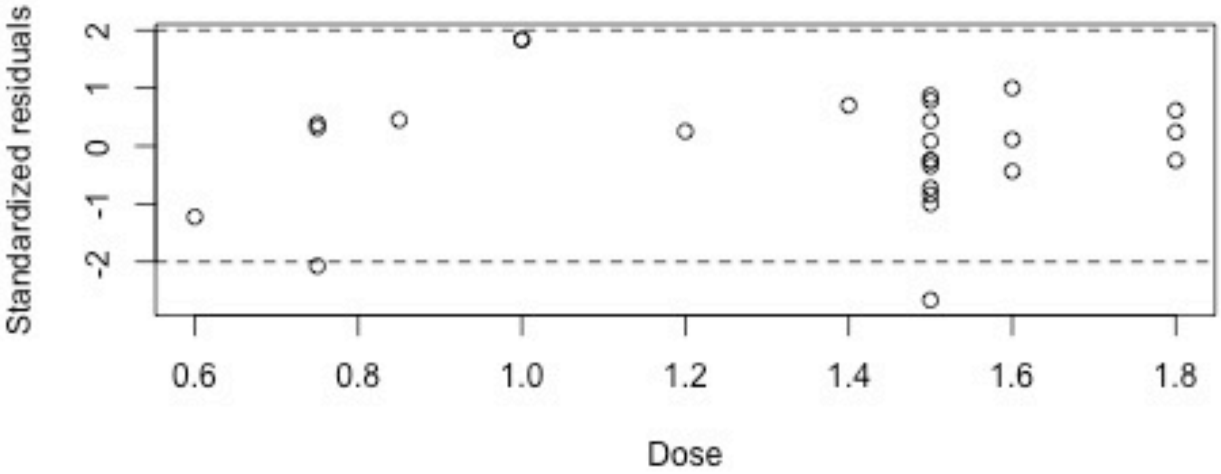


```
> dat$Subject[stdres(red.mod3)< -2]
[1] 2 3
```

```
plot(red.mod3$fitted,stdres(red.mod3),
    ylim=range(stdres(red.mod3)+c(-.08,.08)),
    xlab="Fitted",ylab="Standardized residuals")
abline(h=2,lty=2); abline(h=-2,lty=2)
```



BSA vs. Residuals



Dose vs. Residuals

R Code for plots of BSA and Dose vs. Standardized Residuals

```
par(mfrow=c(2,1))

plot(BSA,stdres(red.mod3),xlab="BSA",
               ylim=range(stdres(red.mod3)+c(-.08,.08)),
               ylab="Standardized residuals",main="BSA vs. Residuals")

abline(h=c(-2,2),lty=2)

plot(Dose,stdres(red.mod3),xlab="Dose",
                ylim=range(stdres(red.mod3)+c(-08,.08)),
                ylab="Standardized residuals",main="Dose vs. Residuals")


abline(h=2,lty=2)
abline(h=c(-2,2),lty=2)
```

Outliers

- Observations that differ considerably from the rest of the data.
- In simple regression, they may fall far from the line implied by the rest of the data.
- May be “a bad value” resulting from some recording or measurement error, or may be a highly useful piece of evidence concerning the process under study.
- There should be strong evidence that the outlier is a bad value before it is discarded.
- Effect of outliers may be checked by dropping these points and refitting the regression equation.

Influential points and influence diagnostics

Influential point:

- An observation which individually or together with other observations has a larger impact on estimates, than other observations (e.g. estimates: slope, standard error, test statistics).
- Can be influential on the x-direction or y-direction.

Influence diagnostics:

- Provide rational and objective measures to assess the impact individual data points may have on estimates.
- Gives impartial measures by which to either remove a data point or give it a weight to decrease its influence.

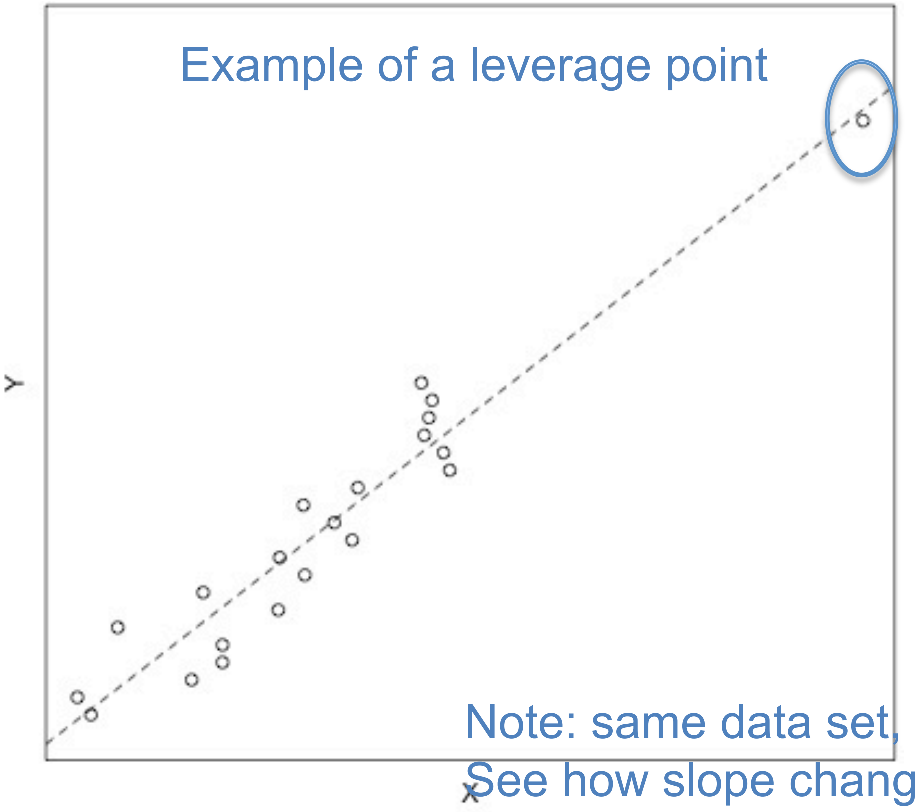
Not all outliers are influential and not all influential observations are outliers.

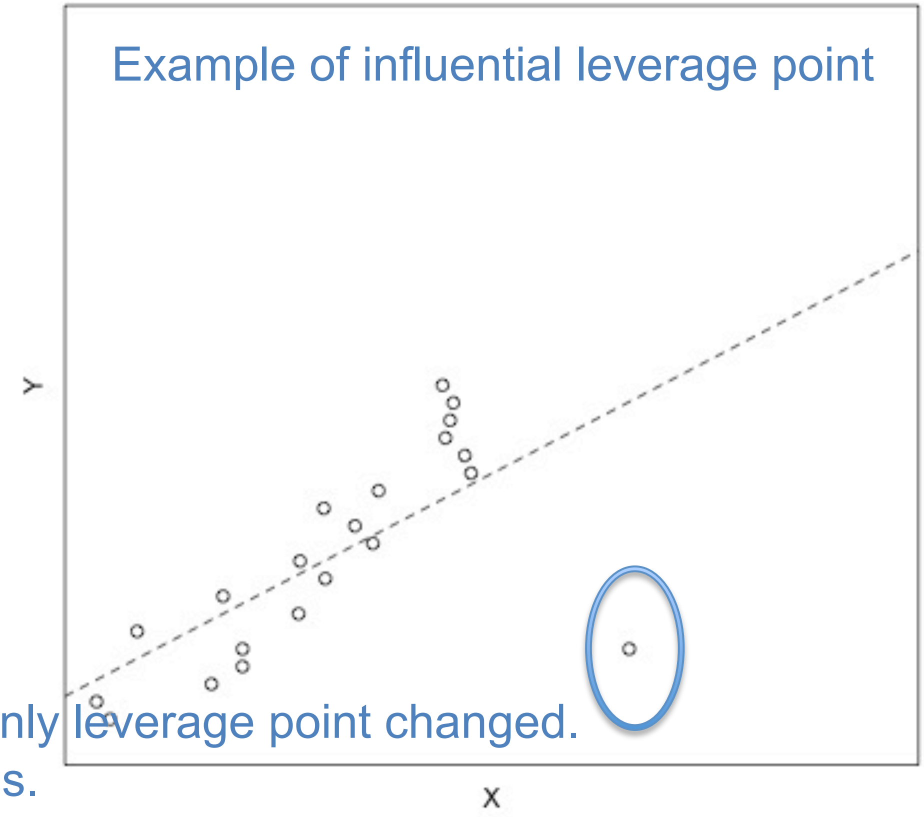
Influence in the x-direction and leverage

- Remote points in the “x-space” can have impact on the model estimates since they act as a “leverage” on the regression line.
- Measure of leverage: “Hat” value, to be seen shortly.

Not all leverage points are necessarily influential. This point has a large hat value, but it has almost no effect on the regression coefficients.

Observations with large hat values **and** large residuals are likely to be influential.


Example of a leverage point

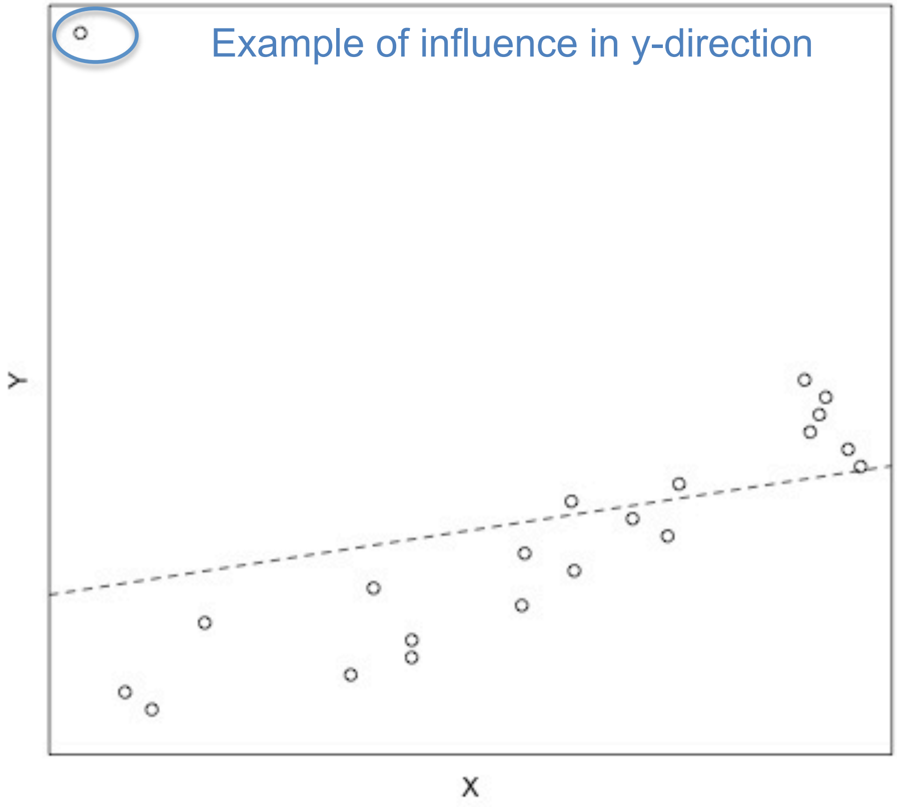

Example of influential leverage point

Note: same data set, only leverage point changed. See how slope changes.

Influence in the y-direction

When a single observation is discordant from the others in the y-direction and has an influence in the regression estimates.

Influential points in this direction are often detected by visual examination or by residual analysis.



Influence measure in the x-direction

The measure of leverage is called “hat” value, denoted by h_i:

- It is contained in the diagonal elements of the “Hat matrix” H and is associated to each observed value Y_i.
- It can be seen as a value used to “ponder” in importance each one of the observed values of Y based on their X values.

$$\hat{Y} = HY$$

H is calculated with X values

Rule of thumb: an independent variable has greater leverage than other observations when

$$h_i > \frac{2p}{n},$$

where p is the number of parameters in the model.

Studentized and “hat” values vs. Standardized Residuals

- Standardized residuals are scaled to the mean squared error (MSE, the variance of the residuals), **suspect of values > ±2.**

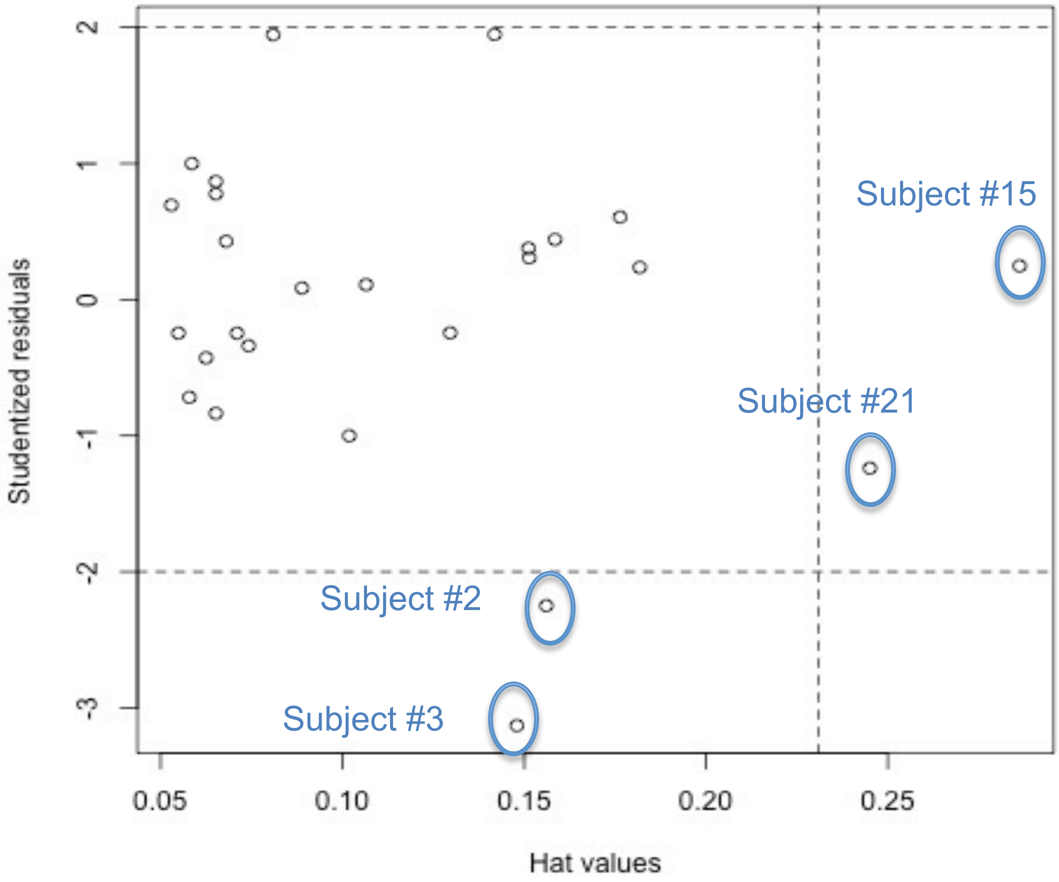
- Studentized residuals are scaled to the MSE with a term that considers the hat values, **suspect of values > ±2.**

- Studentized residuals are more useful in finding influential values because of the hat values that contribute to their construction.

```
In R, use

fit <- lm(y~x1+x2)
studres(fit)
```

```
> p <- 3
> rule.thumb <- 2*p/nrow(dat)
> rule.thumb
[1] 0.2307692
```



Plot shows that points #2,3 are possibly not influential after all, but rather points 15 and 21.

How to access hat values and plot in R

```
red.mod3 <- lm(logCL ~ BSA + Dose, data=dat)
hs <- hatvalues(red.mod3)

p <- 3
rule.thumb <- 2*p/nrow(dat)



# graph
plot(hs,studres(red.mod3),ylab="Studentized residuals",
                xlab="Hat values")
abline(h=-2,lty=2)
abline(h=2,lty=2)
abline(v=rule.thumb,lty=2)

# identify observations greater than 2p/n
eval.hatvalues <- (hs>rule.thumb)*1
hs[eval.hatvalues==1]
```

Some measures of influence in the y-direction (AKA Deletion Diagnostics)

DFFITS: measures the number of standard errors that the i-th predicted value changes if that observation is deleted from the data set.

DFBETAS: measures the number of standard errors that a parameter estimate changes with the the i-th observation deleted from the data set.

For small/moderate sample sizes, DFFITS or DFBETAS greater than +/- 1 are indicative of influential observations. See Bonate for more details.

Some measures of influence in the y-direction (AKA Deletion Diagnostics)

COOK'S DISTANCE:

- Represents a distance measure between the vector of parameter estimates before and after deletion of the i-th observation.
- Summarizes the DFBETAS information in one composite score that assesses the influence an observation has on the set of regression parameters.

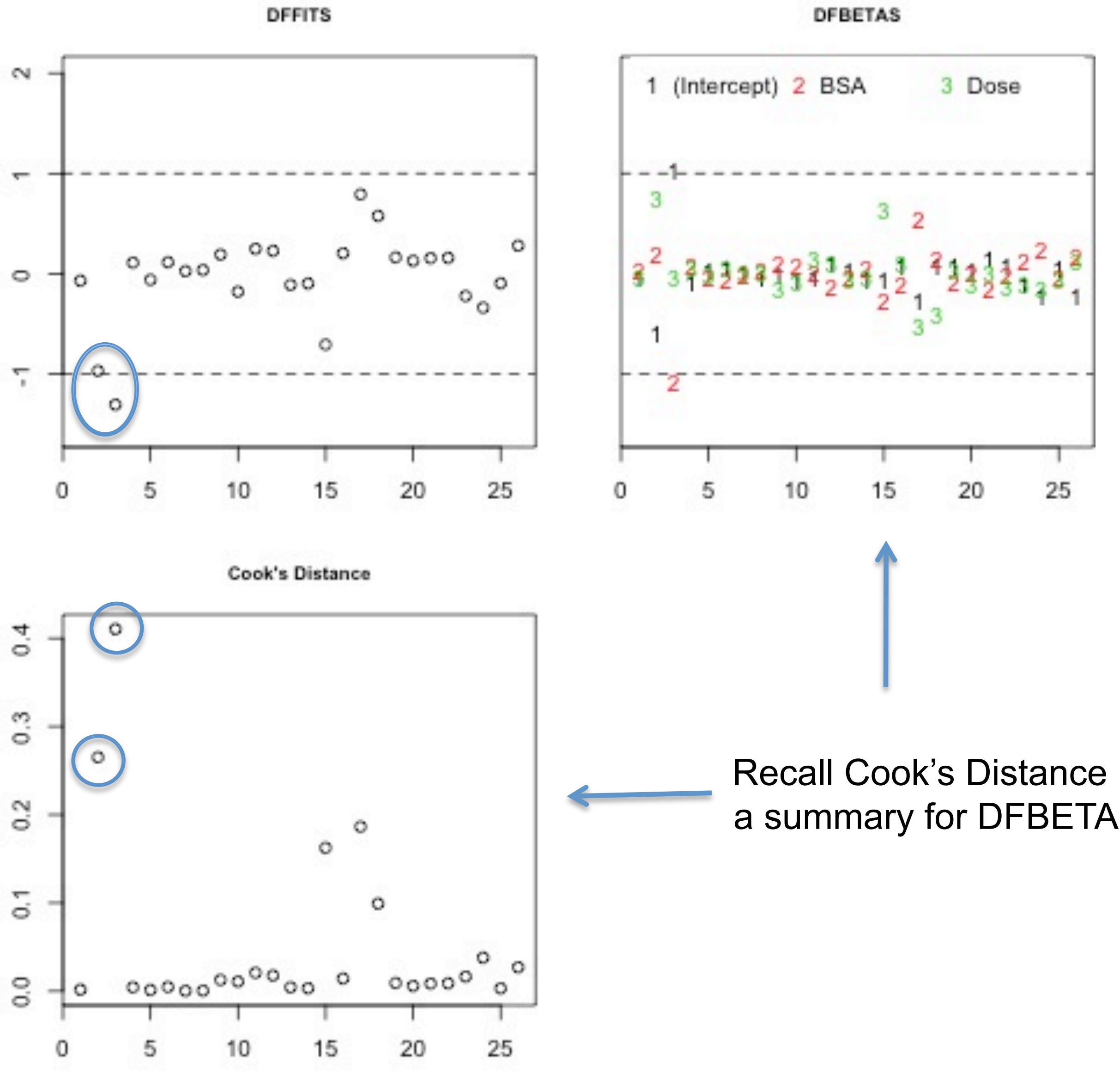
$$D_i = \frac{\sum_{i=1}^{n} \left(\hat{Y}_j - \hat{Y}_{j(i)} \right)^2}{p\ MSE}$$

MSE is based on the fit with all the data points
p is the number of parameters (k covariates + intercept)

$\hat{Y}_j$ is the prediction from the full regression model for observation j;

$\hat{Y}_{j(i)}$ is the prediction for observation j from a refitted regression model in which observation i has been omitted;

Influence diagnostics by residual index
Log 5-FU Data



Recall Cook's Distance is a summary for DFBETAS

When deletion of outliers is not justified:

- May consider that the model is misspecified.

- Try a weighted linear regression using hat values as inverse weights, this way influential observations are given less weight than remaining data.

- Regression model results before and after deletion of outliers should be reported and discussed.

Summary of steps in regression modeling

1. Summarize data to
   a) Identify potential outliers,
   b) Perform a preliminary comparison of groups,
   c) Assess the viability of inclusion of categorical covariates (in terms of sample size and resulting precision of estimates),

   Use:
   - Scatter plots, box plots, histograms, Normal Q-Q plots
   - Frequency tables among categorical variables to assess no. units within cross-classification.
   - Sample descriptive statistics.

2. Select an appropriate model based on a variable selection method.
   - Use t-tests for individual coefficients significance level.
   - Use ANOVA F-test for joint significance if dummies are present.
   - Employ Goodness of Fit tests (ANOVA F, R^2_{adj})

3. Check model assumptions via residual analysis.

Sometimes an iterative process.

4. Draw conclusions.

Some comments on Multiple Regression

- When having a large number of potential covariates is is recommended to fit univariate models separately. These give more direct results than correlation tests when dealing with categorical covariates.

- Multiple regression does not work very well for small data sets.

- Some rules of thumb on the number of covariates: square root of n. Also, sometimes it is suggested no more than n/10 variables.

Some comments on Multiple Regression

- Common sense is needed in specifying the significance level, especially when accumulation of evidence shows that a particular variable is important for the outcome.

E.g., sometimes a p-value of 0.07 may be good enough.

- Consider also interaction terms only if they make sense.

- To reduce the risk that the model is over-optimistic, it is desirable to assess the predictive capability of a model on a new, independent set of data, though not always possible.

2.8. Matrix representation and properties of E(Y) and Var(Y)

Two main features of regression models need to be understood in order to learn the second half of the course:

1. Matrix representation of the models:
   - The design matrix
   - The variance-covariance matrix of residuals.
2. Assumed expectation and variance of residuals and its relationship with the distribution of the response variable.

Also, objects in R have vector/matrix structure and calculations are internally processed in matrix form.

Matrix addition

$$A_{2x2} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}_{2x2}$$

A is a matrix of size 2x2 (no. rows x no. cols)
dim(A) = 2x2

$$B_{2x2} = \begin{bmatrix} e & f \\ g & h \end{bmatrix}_{2x2}$$

B is a matrix of size 2x2
dim(B) = 2x2

Addition rule:
For two matrices to be added, they have to be of the same size.

$$(A+B)_{2x2} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}_{2x2} + \begin{bmatrix} e & f \\ g & h \end{bmatrix}_{2x2} = \begin{bmatrix} a+e & b+f \\ c+g & d+h \end{bmatrix}_{2x2}$$

dim(A+B) = 2 x 2

And the same applies with subtraction.

Matrix multiplication

$$A_{2x2} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}_{2x2}$$

A is a matrix of dimensions 2x2 (no. rows x no. cols)

$$B_{2x2} = \begin{bmatrix} e & f \\ g & h \end{bmatrix}_{2x2}$$

B is a matrix with dimension 2x2

$$C_{2x1} = \begin{bmatrix} i \\ j \end{bmatrix}_{2x1}$$

C is a vector of dimension 2
Or a matrix with dimension 2x1

Multiplication rule:
For two matrices to be multiplied, the number of columns in the first equals the number of rows in the second, i.e. they are “conformable”.

$$(AC)_{2x1} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}_{2x2} \cdot \begin{bmatrix} i \\ j \end{bmatrix}_{2x1} = \begin{bmatrix} ai + bj \\ ci + dj \end{bmatrix}_{2x1}$$

dim(AC)=2x1
No.cols A x no. rows C

$$(AB)_{2x2} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}_{2x2} \cdot \begin{bmatrix} e & f \\ g & h \end{bmatrix}_{2x2} = \begin{bmatrix} ae + bg & af + bh \\ ce + dg & cf + dh \end{bmatrix}_{2x2}$$

Matrix multiplication and transpose of a vector

$$A_{2x2} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}_{2x2}$$

A is a matrix of dimensions 2x2 (no. rows x no. cols)

$$C_{2x1} = \begin{bmatrix} i \\ j \end{bmatrix}_{2x1}$$

C is a vector of dimension 2
Or a matrix with dimension 2x1

$$C^{T}{}_{1x2} = \begin{bmatrix} i & j \end{bmatrix}_{1x2}$$

This is the transpose of the vector, simply expressing it as a row instead of a column.

$$\left(C^{T}AC\right)_{1x1} = \begin{bmatrix} i & j \end{bmatrix}_{1\times2} \begin{bmatrix} a & b \\ c & d \end{bmatrix}_{2x2} \cdot \begin{bmatrix} i \\ j \end{bmatrix}_{2x1}$$

Note C needs to be transposed in order to be conformable with A.

$$= \left(\begin{bmatrix} i & j \end{bmatrix}_{1\times2} \begin{bmatrix} a & b \\ c & d \end{bmatrix}_{2x2} \right) \begin{bmatrix} i \\ j \end{bmatrix}_{2x1}$$

$$= \begin{bmatrix} ia + jc & ib + jd \end{bmatrix}_{1\times2} \begin{bmatrix} i \\ j \end{bmatrix}_{2x1} = i^{2}a + ijc + ijb + j^{2}d$$

Note this is a scalar, with quadratic terms i and j due to multiplying the C vector twice.

Matrix multiplication, transpose of a matrix, inverse of a matrix

$$A_{2x2} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}_{2x2}$$

A is a matrix of dimensions 2x2 (no. rows x no. cols)

$$B_{2x2} = \begin{bmatrix} e & f \\ g & h \end{bmatrix}_{2x2}$$

B is a matrix with dimension 2x2

$$B^{T}{}_{2x2} = \begin{bmatrix} e & g \\ f & h \end{bmatrix}_{2x2}$$

B^T is the transpose of the B matrix with dimension 2x2

$$\left(B^T B\right)_{2x2} = \begin{bmatrix} e & g \\ f & h \end{bmatrix}_{2x2} \begin{bmatrix} e & f \\ g & h \end{bmatrix}_{2x2} = \begin{bmatrix} e^2 + g^2 & ef + gh \\ ef + gh & f^2 + h^2 \end{bmatrix}_{2x2}$$

$$\left(B^T B\right)^{-1}$$

$(B^TB)^{-1}$ is called “the inverse” of the matrix B^TB and has the following property:

$$\left(B^T B\right)^{-1}\left(B^T B\right) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = I_2$$

I_2 is called the identity matrix

Summary

1. Dimensions of matrices are given by no. rows x no. columns.
2. A vector is a matrix with 1 column.
3. Addition of matrices: they have to be of the same size.
4. Multiplication of two matrices: no. cols of the first must match no. rows of the second.
5. The multiplication rule above means matrices are “conformable”.
6. Transpose of a vector: useful in making it conformable when multiplying with a matrix.
7. Matrix inversion: is the analogue of the multiplicative inverse for scalars: multiplying the inverse of $1/a=a^{-1}$ times a will give unity ($A^{-1}A=I$, identity matrix.)
8. The analogue of scalar multiplication $axa=a^2$ is of the form A^TA, using A transposed, where quadratic and crossed terms result.

Matrix representation for the simple linear model (case k=1)

$$Y_i = \beta_0 + \beta_1 X_{1i} + \varepsilon_i, \;\; i = 1,\ldots,n; \;\; \varepsilon_i \sim N(0,\sigma^2).$$

Rows of X correspond to subject i=1,…n

$$Y_{n\times 1} = X_{n\times 2}\beta_{2\times 1} + \varepsilon_{n\times 1}$$

Stack all Y observations into one vector

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix}_{n\times 1} = \begin{bmatrix} 1 & X_{11} \\ 1 & X_{12} \\ \vdots & \vdots \\ 1 & X_{1n} \end{bmatrix}_{n\times 2} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}_{2\times 1} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}_{n\times 1}$$

X_{nx2} is called the "design matrix"

Matrix notation, case k=1
Body composition %Fat example

$$Y_i = \beta_0 + \beta_1 \, Age_i + \varepsilon_i, \; i = 1,\ldots,25;$$

$$\varepsilon_i \sim N(0,\sigma^2).$$

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_{25} \end{bmatrix}_{25\times 1} = \underbrace{\begin{bmatrix} 1 & Age_1 \\ 1 & Age_2 \\ \vdots & \vdots \\ 1 & Age_{25} \end{bmatrix}_{25\times 2}}_{\text{Design matrix}} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}_{2\times 1} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_{25} \end{bmatrix}_{25\times 1}$$

Matrix representation for the multiple linear model (case k≥2)

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \cdots + \beta_k X_{ki} + \varepsilon_i$$

Rows of X correspond to subject i=1,…n

$$Y_{n\times 1} = X_{n\times p}\beta_{p\times 1} + \varepsilon_{n\times 1}$$

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix}_{n\times 1} = \begin{bmatrix} 1 & X_{11} & X_{21} & \cdots & X_{k1} \\ 1 & X_{12} & X_{22} & \cdots & X_{k2} \\ \vdots & & \vdots & \ddots & \vdots \\ 1 & X_{1n} & X_{2n} & \cdots & X_{kn} \end{bmatrix}_{n\times p} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix}_{p\times 1} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}_{n\times 1}$$

Matrix notation, case k=2
Body composition %Fat example

$$Y_i = \beta_0 + \beta_1\, Sex_i + \beta_2\, Age_i + \varepsilon_i;\;\; i = 1,\ldots,25,\;\; \varepsilon_i \sim N(0,\sigma^2).$$

$$Y_{25\times 1} = X_{25\times 3}\beta_{3\times 1} + \varepsilon_{25\times 1}$$

Sex variable

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_{15} \\ Y_{16} \\ \vdots \\ Y_{25} \end{bmatrix} = \begin{bmatrix} 1 & Sex_1 & Age_1 \\ \vdots & \vdots & \vdots \\ 1 & Sex_{15} & Age_{15} \\ 1 & Sex_{16} & Age_{16} \\ \vdots & \vdots & \vdots \\ 1 & Sex_{25} & Age_{25} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_{15} \\ \varepsilon_{16} \\ \vdots \\ \varepsilon_{25} \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 1 & Age_1 \\ \vdots & \vdots & \vdots \\ 1 & 1 & Age_{15} \\ 1 & 0 & Age_{16} \\ \vdots & \vdots & \vdots \\ 1 & 0 & Age_{25} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_{15} \\ \varepsilon_{16} \\ \vdots \\ \varepsilon_{25} \end{bmatrix}}$$

Subjects are grouped and listed according to gender: in this example, starting with women who have a code of 1 in the Sex variable (15 subjects) followed by men with a code of 0 (remaining 10 subjects).

How to access the design matrix in R

```
> fit <- lm(fat~age+sex,data=agefat)
> model.matrix(fit)
   (Intercept) age sexmale
1            1  24       1
2            1  37       1
3            1  41       1
4            1  60       1
5            1  31       0
6            1  39       0
7            1  58       1
8            1  23       1
9            1  23       0
10           1  27       1
```

```
   (Intercept) age sexmale
11           1  27       1
12           1  39       0
13           1  41       1
14           1  45       1
15           1  49       0
16           1  50       0
17           1  53       0
18           1  53       0
19           1  54       0
20           1  56       0
21           1  57       0
22           1  58       0
23           1  58       0
24           1  60       0
25           1  61       0
```

Matrix representation for the residuals' distribution

$$\varepsilon_{n\times 1} \sim N(\mu_{n\times 1}, \Sigma_{n\times n}),$$

Applies for both simple and multiple linear models.

Called null-vector

where

$$\mu_{n\times 1} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \text{ and } \Sigma_{nxn} = \begin{bmatrix} Var(\varepsilon_1) & Cov(\varepsilon_1,\varepsilon_2) & \cdots & Cov(\varepsilon_1,\varepsilon_n) \\ Cov(\varepsilon_2,\varepsilon_1) & Var(\varepsilon_2) & \cdots & Cov(\varepsilon_2,\varepsilon_n) \\ \vdots & \vdots & \ddots & \vdots \\ Cov(\varepsilon_n,\varepsilon_1) & Cov(\varepsilon_n,\varepsilon_2) & \cdots & Var(\varepsilon_n) \end{bmatrix}$$

$$= \begin{bmatrix} \sigma^2 & 0 & \cdots & 0 \\ 0 & \sigma^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma^2 \end{bmatrix} = diag(\sigma^2)_n\,.$$

Diagonal "variance-covariance matrix". Subjects are assumed uncorrelated, so off-diagonal elements are zero.

Matrix representation for the residuals' distribution

When units are uncorrelated, the variance-covariance matrix can also be expressed in terms of an "identity" matrix of dimension n (I_n):

$$\Sigma_{nxn} = \begin{bmatrix} \sigma^2 & 0 & \cdots & 0 \\ 0 & \sigma^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma^2 \end{bmatrix} = \sigma^2 \underbrace{\begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}}_{\text{The identity matrix is a matrix of ones in its diagonal and zeros in the off-diagonal.}} = \sigma^2 I_n$$

Matrix notation for residuals
Body composition %Fat example

$$\varepsilon_{25\times 1} \sim N(\mu_{25\times 1}, \Sigma_{25\times 25}),$$

where

$$\mu_{25\times 1} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \text{ and } \Sigma_{25x25} = \begin{bmatrix} Var(\varepsilon_1) & Cov(\varepsilon_1,\varepsilon_2) & \cdots & Cov(\varepsilon_1,\varepsilon_{25}) \\ Cov(\varepsilon_2,\varepsilon_1) & Var(\varepsilon_2) & \cdots & Cov(\varepsilon_2,\varepsilon_{25}) \\ \vdots & \vdots & \ddots & \vdots \\ Cov(\varepsilon_{25},\varepsilon_1) & Cov(\varepsilon_{25},\varepsilon_2) & \cdots & Var(\varepsilon_{25}) \end{bmatrix}$$

$$= \begin{bmatrix} \sigma^2 & 0 & \cdots & 0 \\ 0 & \sigma^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma^2 \end{bmatrix} = diag(\sigma^2)_{25}.$$

Diagonal “variance-covariance matrix”. Subjects are assumed uncorrelated, so off-diagonal elements are zero.

Revisiting Expectation and Variance

- Recall that expectation and variance are the population analogues to the sample mean and sample variance.

- For the theoretical Normal distribution, E(X) = μ and Var(X)=σ^2.

- These are usually unknown quantities which we aim to guess through the sample mean $\overline{X}$ and sample variance s^2.

- For continuous distributions, they are mathematically defined in terms of the PDF of X, f(x):

$$E(X) = \mu = \int x f(x) dx.$$

$$Var(X) = \sigma^2 = \int (x - \mu)^2 f(x) dx = E\left[(x - \mu)^2\right].$$

- Since these are integrals and functions of the PDF and the random variable X, they follow some rules when calculating the expectation and variance of arithmetic operations of random variables,
  - e.g. E(X+Y),Var(X+Y)

Rules of Expectation and Variance

Let X, Y represent random variables; a, b represent constants. The following rules apply for X, Y, a, b as vectors or scalars.

$$E(a) = a$$

$$E(a\,X) = a\,E(X)$$

$$E(X + b) = E(X) + b$$

$$E(X+Y) = E(X) + E(Y)$$

$$Var(a) = 0$$

$$Var(a\,X) = a^2\,Var(X),$$

$$= a^T\,Var(X)\,a,$$

In case a,X are vectors.
Note that if X is a vector, $Var(X)$ is a matrix

$$Var(X + Y) = Var(X) + Var(Y) + 2\,Cov(X,Y)$$ if X,Y not independent

$$Var(X + Y) = Var(X) + Var(Y)$$ if X,Y independent

Example, Simple linear regression (k=1)

$$Y_i = \beta_0 + \beta_1 X_{1i} + \varepsilon_i;$$

$$\varepsilon_i \sim N(0,\sigma^2) \Leftrightarrow E(\varepsilon_i) = 0, Var(\varepsilon_i) = \sigma^2.$$

$$\begin{aligned} E(Y_i) &= E(\beta_0 + \beta_1 X_{1i} + \varepsilon_i) \\ &= E(\beta_0) + E(\beta_1 X_{1i}) + E(\varepsilon_i) \\ &= \beta_0 + \beta_1 X_{1i}. \end{aligned}$$

Note: the proper notation here should be E(Y|X) – the conditional expectation of Y given X – since X can be random too. We will leave out the conditional notation for simplicity, since the rules remain unaltered when X is random.

$$\begin{aligned} Var(Y_i) &= Var(\beta_0 + \beta_1 X_{1i} + \varepsilon_i) \\ &= Var(\beta_0) + Var(\beta_1 X_{1i}) + Var(\varepsilon_i) \\ &= Var(\varepsilon_i) = \sigma^2. \end{aligned}$$

$$Y_i \sim N(\ \beta_0 + \beta_1 X_{1i}\ ,\ \sigma^2\ )$$

variable is Y, so X is a cosntant

Example, Multiple linear regression (k≥2)

$$Y_{n\times 1} = X_{n\times p}\beta_{p\times 1} + \varepsilon_{n\times 1};$$

$$\varepsilon_{n\times 1} \sim N(0_n, \sigma^2 I_n) \leftrightarrow E(\varepsilon_{n\times 1}) = 0_n,\ \ Var(\varepsilon_{n\times 1}) = \sigma^2 I_n .$$

Xβ is a constant

$$E(Y) = E(X\beta) + E(\varepsilon) = X\beta + E(\varepsilon) = X\beta.$$

$$Var(Y) = \underbrace{Var(X\beta)}_{0} + Var(\varepsilon) = \sigma^2 I_n .$$

$$Y_{n\times 1} \sim N\left(X\beta\ ,\ \sigma^2 I_n\right)$$

Regression coefficients and predicted response in matrix notation

$$\hat{\beta}_{p\times 1} = \left(X^T X\right)^{-1}_{p\times p} X^T_{p\times n} Y_{n\times 1}$$

Estimated regression coefficients for the k=1 case

$$\left\{\begin{aligned} \hat{\beta}_1 &= \frac{\sum_{i=1}^{n}\left(x_i - \bar{x}\right)\left(y_i - \bar{y}\right)}{\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2} \\ \hat{\beta}_0 &= \bar{y} - \hat{\beta}_1 \bar{x} \end{aligned}\right.$$

Predicted value of Y in terms of the "Hat" matrix H.

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

$$= HY$$

Regression coefficients
Matrix calculation in R
5-FU data example

$$\hat{\beta}_{p\times 1} = \left(X^T X\right)^{-1}_{p\times p} X^T_{p\times n} Y_{n\times 1}$$

```
red.mod3s <- lm(logCL ~ SBSA + SDose,data=dat)

mat <- model.matrix(red.mod3s)

# the %*% makes the matrix multiplication
xtx <- t(mat)%*%mat
xty <- t(mat)%*%dat$logCL
```

X*transpose*X

Xtranspose*y

```
# the solve() function calculates the rest
> solve(xtx,xty)
                  [,1]
(Intercept) -1.0036257
SBSA         1.7728859
SDose       -0.6219452
```

```
> summary(red.mod3s)$coef[,1]
(Intercept)        SBSA       SDose 
 -1.0036257   1.7728859  -0.6219452 
```

2.9. Estimation via Maximum Likelihood

The Likelihood Function

- The likelihood is a function that has the identical form of a PDF but has a a different use and interpretation (both discrete and continuous random variables).

- For example, take the joint Normal (μ,σ^2) PDF for a sample of independent observations y_1, y_2,…,y_n .

- As a PDF, we would say it is a function of the observed data and is useful in calculating the joint probability:

$$P(Y \leq y_1, Y \leq y_2, \ldots, Y \leq y_n).$$

$$\begin{aligned} f(y_1, y_2, \ldots, y_n) &= \prod_{i=1}^{n} f(y_i) \\ &= \left(2\pi\sigma^2\right)^{-n/2} \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left(y_i - \mu\right)^2\right\}. \end{aligned}$$

Note: under independence, the joint PDF is calculated as the product of the individual PDF’s.

The Likelihood Function

- The likelihood function for these data (under the Normal model) is given by:

$$L(\mu,\sigma^2) = \left(2\pi\sigma^2\right)^{-n/2} \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left(y_i - \mu\right)^2\right\}.$$

- The likelihood takes the data y_1, $y_2,\ldots,y_n$ as fixed (not random as the PDF does) and $\theta=(\mu,\sigma^2)$ as a varying parameter.

- Therefore, interpretation of $L(\theta)$ in general, is that it gives the plausibility for a set of values for θ, given the data at hand.

- The most likely value of θ is denoted by $\hat{\theta}^{ML} = \left(\hat{\mu}^{ML}, \hat{\sigma}^{2ML}\right)$ and is called the Maximum Likelihood Estimator, or MLE.

- That is, the MLE maximizes the function $L(\theta)$.

The Maximum Likelihood Estimation Method
(Due to Ronald A. Fisher, 1890-1962)

- It is an estimation method that aims to maximize the likelihood funciton $L(\theta)$ with respect to θ.

- The value that maximizes $L(\theta)$ is called Maximum Likelihood Estimator (MLE).

- The MLE competes with other estimators (e.g., OLS), and has some statistically desirable attributes involving accuracy and precision.

- **Accuracy (Bias):** describes systematic departure from the true value of θ.

- **Precision (Variability):** describes the sampling error of the MLE

(Recall that since the MLE is calculated with the data then it is random as well and as any random variable, it has variability – think of the SE of the mean).

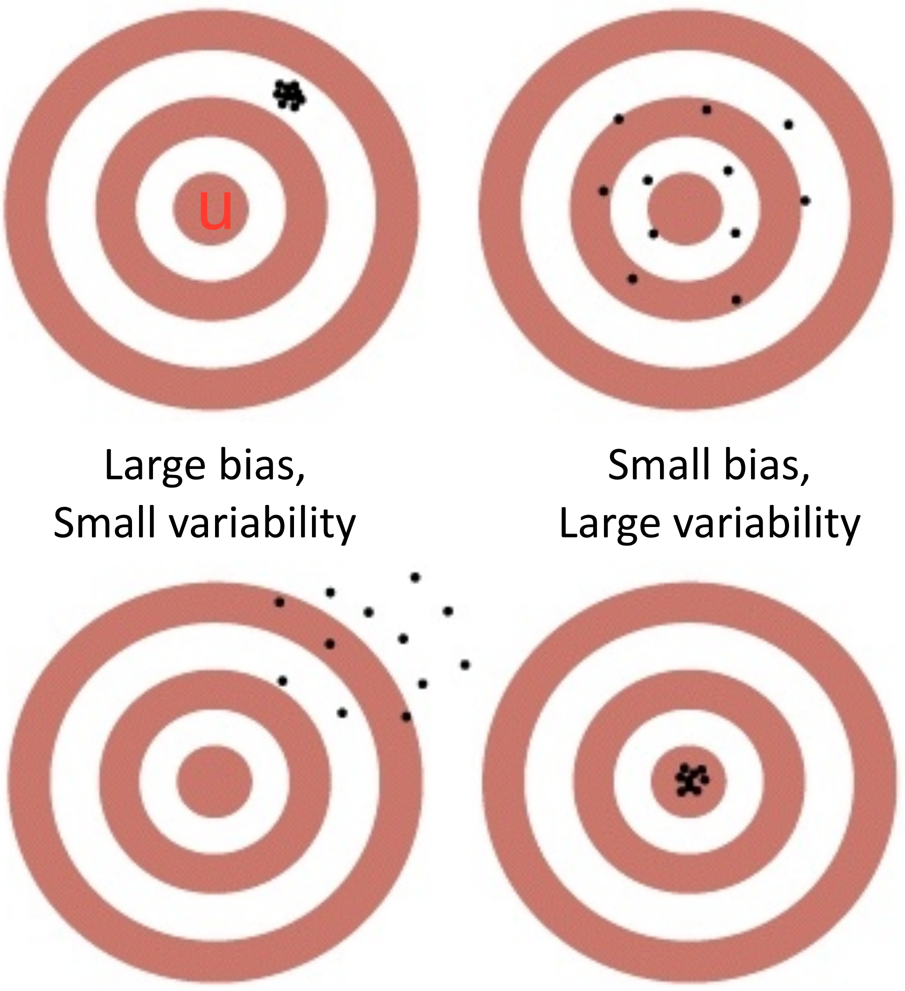
Examples of Bias and Variability

The target represents the true value of θ (population parameter).

The points represent different values of the estimator (MLE) obtained from different samples.

Bias is the average difference between an estimator (MLE) and the true value θ.

The variability of the estimator is represented by the dispersion of the points.



Large bias,
Small variability

Small bias,
Large variability

Large bias,
Large variability

Small bias,
Small variability

Some statistical properties of MLE’s

1. Asymptotically unbiased (i.e., increasingly accurate as n grows large).
2. Asymptotically minimum variance (i.e., increasingly precise as n grows large).
3. Scale invariance.

Notes:

- In theory, asymptotically means “when the sample size n goes to infinity” but in practice, n only needs to be moderately large.
- Scale invariance means that the estimates themselves are unchanged when both the measurements and the parameters are transformed.

Maximization of the likelihood function

Maximization of L(θ) can be done:

- Analytically, for likelihood functions that are mathematically tractable.
- Numerically, for more complex functions.

Steps when solved analytically:

1. Take logarithms of the likelihood function $l(\theta)$
2. Take derivatives of $l(\theta)$ with respect to the parameters
3. Solve the derivatives
4. Verify that the solution found in (3) is the overall maxima (i.e., is not a local maxima).

- Numerically, can be done via computer software (including R). Maximizing $L(\theta)$ is equivalent to minimizing $-l(\theta)=-\log L(\theta)$. Optimizers in statistical packages usually work by minimizing the result of a function.

Analytical maximization of the Normal likelihood

find mu and sigma to maximum L(theta)

Maximum likelihood in original scale

$$L(\theta) = \left(2\pi\sigma^2\right)^{\text{-n/2}} \exp\left\{-\frac{1}{2\sigma^2}\sum\nolimits_{i=1}^{n}\left(y_i-\mu\right)^2\right\},$$

Log-likelihood

$$l(\theta) = \log L(\theta) = -\frac{\text{n}}{2}\log\left(2\pi\right) - \frac{\text{n}}{2}\log\left(\sigma^2\right) - \frac{1}{2\sigma^2}\sum\nolimits_{i=1}^{n}\left(y_i-\mu\right)^2,$$

Partial derivative and solution wrt μ

$$\frac{\partial l(\theta)}{\partial \mu} = \frac{1}{\sigma^2}\sum\nolimits_{i=1}^{n}\left(y_i-\mu\right)^2 = 0 \qquad \hat{\mu}_{ML} = \frac{1}{n}\sum\nolimits_{i=1}^{n} y_i = \bar{y}$$

Partial derivative and solution wrt σ^2

$$\left.\frac{\partial l(\theta)}{\partial \sigma^2}\right|_{\mu=\hat{\mu}^{ML}} = -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4}\sum\nolimits_{i=1}^{n}\left(y_i-\bar{y}\right)^2 = 0$$

$$\hat{\sigma}_{ML}{}^2 = \frac{1}{n}\sum\nolimits_{i=1}^{n}\left(y_i-\bar{y}\right)^2 = s_{ML}{}^2$$

The MLE of θ

$$\hat{\theta}^{ML} = \left(\bar{y}, s_{ML}{}^2\right)$$

Verification that MLE is the overall maximum, double derivatives evaluated at the MLE;s must be < 0.

Analytical maximization of the Normal likelihood

Exercise:

Under the normality assumption, derive the MLE for (μ,σ^2) for

1. A sample of size one, $y=3$.
2. A sample of size two, $y_1=6$, $y_2=4$.

Analytical maximization of the Normal likelihood

$$l(\theta) = \log L(\theta) = -\frac{\mathrm{n}}{2}\log(2\pi) - \frac{\mathrm{n}}{2}\log\left(\sigma^2\right) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}\left(y_i - \mu\right)^2,$$

A sample of size one, y=3.

Log-likelihood

$$l(\theta) = -\frac{1}{2}\log\left(\sigma^2\right) - \frac{1}{2\sigma^2}\left(3-\mu\right)^2$$

Partial derivative and solution wrt μ

$$\frac{\partial l(\theta)}{\partial\mu} = -\frac{1}{2\sigma^2}2\left(3-\mu\right)(-1) = \frac{1}{\sigma^2}\left(3-\mu\right) = 0$$

$$\hat{\mu}^{ML} = y = 3.$$

Partial derivative and solution wrt σ^2

$$\left.\frac{\partial l(\theta)}{\partial\sigma^2}\right|_{\mu=\hat{\mu}^{ML}} = -\frac{1}{2\sigma^2} + \frac{1}{2\sigma^4}\left(3-3\right)^2 = 0$$

$$\sigma_{ML}{}^2 = 0.$$

Analytical maximization of the Normal likelihood

$$l(\theta) = \log L(\theta) = -\frac{\mathrm{n}}{2}\log(2\pi) - \frac{\mathrm{n}}{2}\log\left(\sigma^2\right) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}\left(y_i - \mu\right)^2,$$

A sample of size two, y_1=6, y_2=4.

Log-likelihood

$$l(\theta) = -\log\left(\sigma^2\right) - \frac{1}{2\sigma^2}\left\{(6-\mu)^2 + (4-\mu)^2\right\}$$

Partial derivative and solution wrt μ

$$\frac{\partial l(\theta)}{\partial \mu} = -\frac{1}{2\sigma^2}\left\{-2(6-\mu) - 2(4-\mu)\right\}$$

$$= \frac{1}{\sigma^2}\left\{10 - 2\mu\right\} = 0$$

$$\hat{\mu}_{ML} = \frac{10}{2} = \overline{y} = 5.$$

Partial derivative and solution wrt σ^2

$$\left.\frac{\partial l(\theta)}{\partial \sigma^2}\right|_{\mu=\hat{\mu}^{ML}} = -\frac{1}{\sigma^2} + \frac{1}{2\sigma^4}\left\{(1)^2 + (1)^2\right\}$$

$$= -\frac{1}{\sigma^2}\left(1 - \frac{1}{\sigma^2}\right) = 0$$

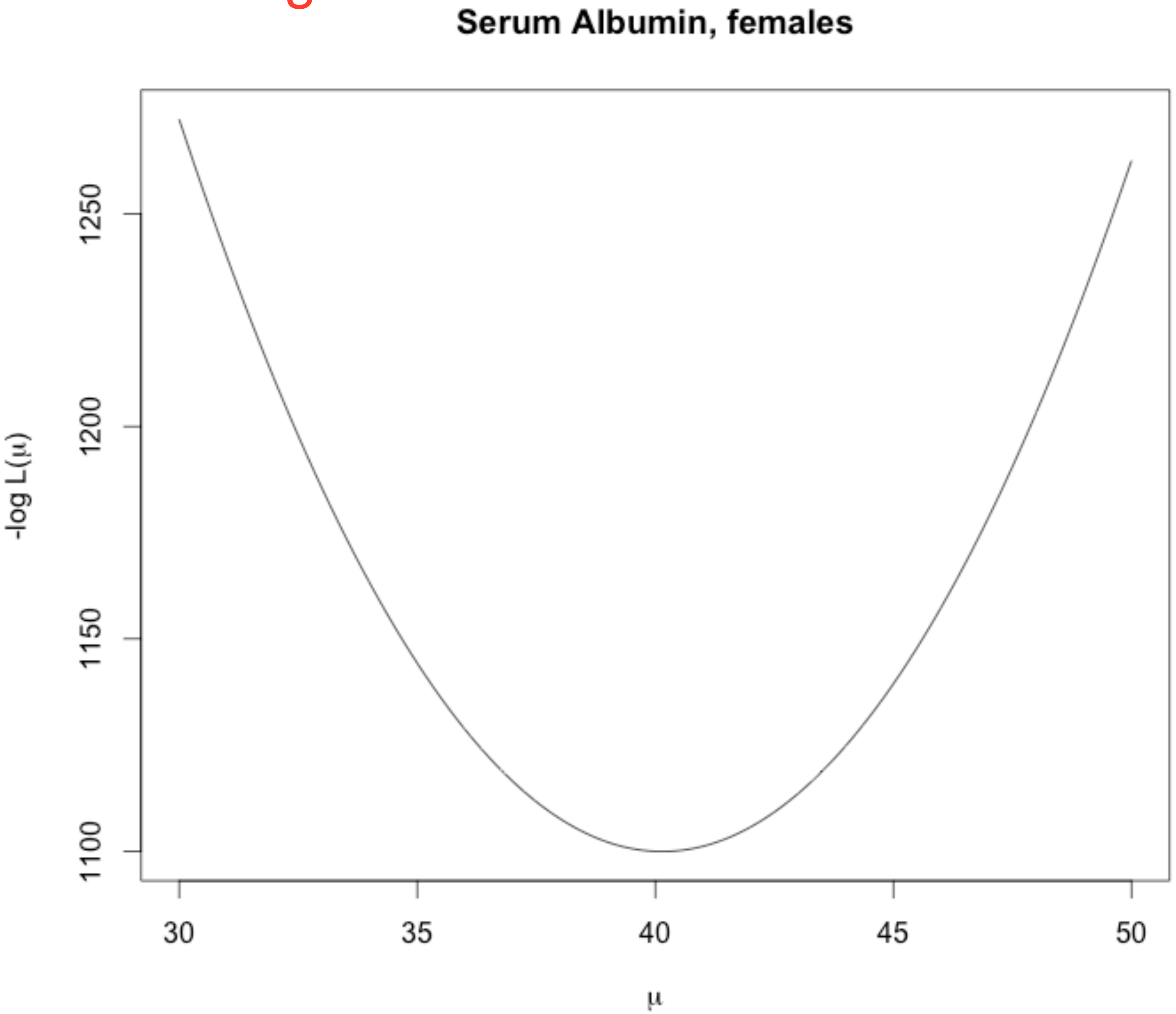
$$\sigma^2 - 1 = 0$$

$$\hat{\sigma}_{ML}{}^2 = 1$$

What is the sample variance?

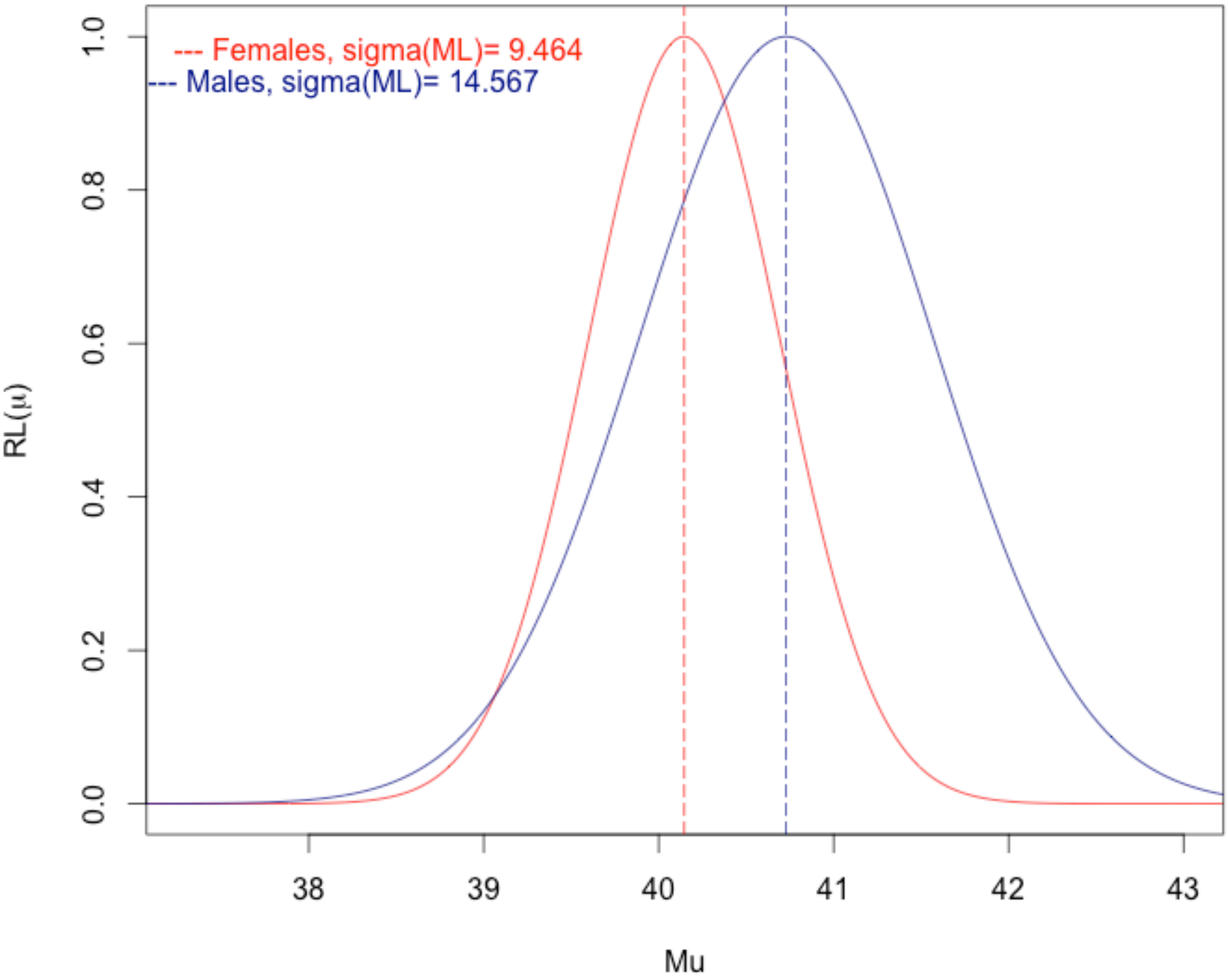
Example: Serum Albumin Data

fix sigma and observe the effect of u



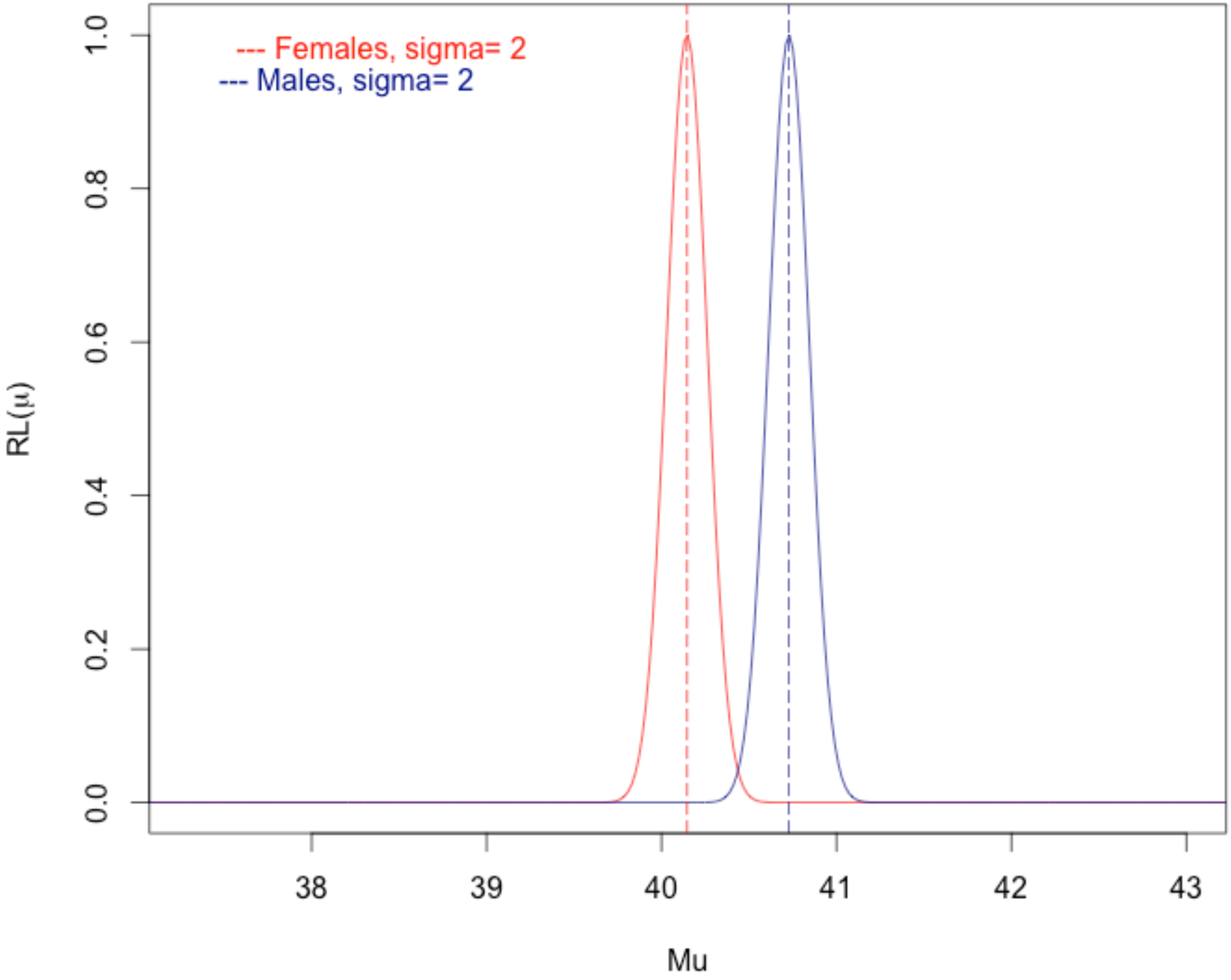
Example: Serum Albumin Data

The Relative Likelihood (RL) is the Likelihood scaled to be plotted within [0,1]



Example: Serum Albumin Data

The width of the curves will vary with respect of the value of sigma used, just like with the Normal PDF.



General Overview
Numerical Optimization of the Normal Likelihood in R

- Some optimizers available in R:
  - `mle()` . . . "stats4" package*
  - `nlminb()`
  - `nlm()`
  - `optim()` . . . Is the only maximizer

前三个找最小值，最后一个找最大值

- Use the help for more information

- They work in terms of an R function which is to be created separately.

- They need starting values: some of them will be more sensitive to them than others and in some cases may work more efficiently.

http://www.r-bloggers.com/fitting-a-model-by-maximum-likelihood/

Numerical Optimization of the Normal Likelihood in R

R function used by `mle()` for the Serum Albumin data for females.

```
LL.fem.fun <- function(mu,sigma) {
  # this function evaluates the -log Normal Likelihood
  # for the Serum Albumin data for females
  # it takes a list with two scalar values: mu and sigma
  # elaborated by D. Hajducek
  # last updated Feb 7, 2016
  var <- SerumAlbumin[Sex=="female"]
  pdf = dnorm(var, mu, sigma)
  return( -sum(log(pdf)) )
}

fit.mle <- mle(LL.fem.fun , start=list(mu=35,sigma=10))
```

Numerical Optimization of the Normal Likelihood in R

```
> fit.mle

Call:
mle(minuslogl = LL3.fem, start = list(mu = 35, sigma = 10))

Coefficients:
       mu     sigma 
40.145327  9.463997 
```

```
# From their analytical form,
# we know the true mle's
> c(mean.fem, sd.mle.fem)
[1] 40.145450  9.464037
```

```
> summary(fit.mle)
Maximum likelihood estimation

Call:
mle(minuslogl = LL3.fem, start = list(mu = 35, sigma = 10))

Coefficients:
       Estimate Std. Error
mu    40.145327  0.5464042
sigma  9.463997  0.3863637

-2 log L: 2199.863 
```

```
# We know the Std. Error of the
# estimate of mu as sigma/sqrt(n)

> sd.mle.fem/sqrt(sum(Sex=="female"))
[1] 0.5464041
```

Numerical Optimization of the Normal Likelihood in R

R function used by `nlminb()` for the Serum Albumin data for females.

```
LL.fem.fun2 <- function(vec) {
  # this function calculates the –log Normal Likelihood for
  # Serum Albumin in females
  # it takes as argument a vector with two entries: c(mu,sigma)
  # elaborated by D. Hajducek
  # last updated Feb 7, 2016

  mu <- vec[1]
  sigma <- vec[2]
  var <- SerumAlbumin[Sex=="female"]
  pdf = dnorm(var, mu, sigma)

  return(-sum(log(pdf))
}

nlminb(c(5,10),LL.fem.fun2)
```

Note: nlminb gave better results with very different starting values. The mle() with mu=5,sigma=10 did not converge.

Numerical Optimization of the Normal Likelihood in R

```
> nlminb(c(5,10),LL.fem.fun)
$par
[1] 40.145455  9.464037

$objective
[1] 1099.931

$convergence
[1] 0

$iterations
[1] 10

$evaluations
function gradient 
      15       27 

$message
[1] "relative convergence (4)"
```

```
> c(mean.fem,sd.mle.fem)
[1] 40.145450  9.464037
```

Maximum Likelihood vs. Least Squares Estimators
Linear Regression

- For moderately large samples and independent observations, the MLE's and LS estimators for $\beta_0, \beta_1, \sigma^2$ are identical.

- More complex models like random effects or NLME's will give different estimates.

- Different assumptions or a variance model that depends on the value of the observation would lead to different MLE's.

- For simple linear models,
  - LS requires only assumptions about the mean and variance of the random errors ("second-moment assumptions").
  - ML not only requires second-moment assumptions for the random errors, but also a distributional assumption (i.e., Normality.)

Maximum Likelihood Estimation (linear regression, e.g. simple case k=1)

Recall that earlier we had the Normal likelihood function:

$$L(\theta) = \left(2\pi\sigma^2\right)^{-n/2} \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left(y_i - \mu\right)^2\right\}, \qquad \theta = \left(\mu,\sigma^2\right)$$

In simple linear regression, it is assumed that $\varepsilon \sim N(0,\sigma^2)$, therefore Y is Normal with mean $\beta_0+\beta_1 X_{1i}$ and variance σ^2.

So the likelihood function for N ($\beta_0+\beta_1 X_{1i}$, σ^2) is given by:

$$L(\theta) = \left(2\pi\sigma^2\right)^{-n/2} \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left(y_i - (\beta_0 + \beta_1 X_{1i})\right)^2\right\},$$

$$\text{where} \qquad \theta = \left(\beta_0,\beta_1,\sigma^2\right).$$

Numerical Optimization of the Normal Simple Regression in R

R function used by `mle()` for Simple Linear Regression: Serum Albumin vs. Sex

```
LL.reg.fun <- function(b0,b1,sigma) {
  # this function calculates the —log Likelihood for
  # the normal residuals of a simple linear regression
  # Serum Albumin vs. Sex
  # it takes as argument a list with three components:
  # list=(b0,b1,sigma)
  # elaborated by D. Hajducek, last updated Feb 7, 2016

  var <- SerumAlbumin - b0 - b1 * (Sex=="female")
  pdf = dnorm(var, 0, sigma)
  -sum(log(pdf))
}

fit.mle <- mle(LL.reg.fun , start=list(b0=35,b1=1,sigma=10))
```

Numerical Optimization of the Normal Simple Regression in R

```
> fit.mle
Call:
mle(minuslogl = LL.reg.fun, start = list(b0 = 35, b1 = 1, sigma = 10))

Coefficients:
        b0         b1      sigma
40.7286783 -0.5832289 12.2837131

> summary(fit.mle)

Maximum likelihood estimation

Call:
mle(minuslogl = LL.reg.fun, start = list(b0 = 35, b1 = 1, sigma = 10))

Coefficients:
        Estimate Std. Error
b0    40.7286783  0.7092004
b1    -0.5832289  1.0029608
sigma 12.2837131  0.3546002

-2 log L: 4712.655
```

```
> summary(fit.ols <- lm(SerumAlbumin ~ Sex))
$coef[,1:2]
              Estimate Std. Error
(Intercept) 40.7286767  0.7103855
Sexfemale   -0.5832267  1.0046368>

summary(lm(SerumAlbumin ~ Sex))$sigma
[1] 12.30424
```

Numerical Optimization of the Normal Simple Regression in R

R function used by `nlminb()`for
Simple Linear Regression: Serum Albumin vs. Sex

```
LL.reg.fun2 <- function(vec) {
  # this function calculates the –log Likelihood for
  # the residuals of a simple linear regression
  # Serum Albumin vs. Sex
  # it takes as argument a vector with three entries:
  # c(b0,b1,sigma)
  # by D.Hajducek, last updated Feb 7, 2016

  b0 <- vec[1]
  b1 <- vec[2]
  sigma <- vec[3]
  res <- SerumAlbumin – (b0+b1*(Sex=="female"))
  pdf = dnorm(res, 0, sigma)
  -sum(log(pdf))
}

nlminb(c(35,1,10),LL.reg.fun2)
```

Numerical Optimization of the Normal Simple Regression in R

```
> nlminb(c(35,1,10),LL.reg.fun2)
$par
[1] 40.7286844 -0.5832444 12.2837154

$objective
[1] 2356.328

$convergence
[1] 0

$iterations
[1] 8

$evaluations
function gradient 
      10       36 

$message
[1] "relative convergence (4)"
```

```
> summary(lm(SerumAlbumin~Sex))$coef[,1]
(Intercept)   Sexfemale 
 40.7286767  -0.5832267 

> summary(lm(SerumAlbumin~Sex))$sigma
[1] 12.30424
```

In class R exercise: 5-FU Clearance data (analyzed in class)

Putting aside the correct physiological interpretation of the functional relationship of CL vs. BSA and Dose, examine the effect of log transforming CL in the residuals.

1. Fit the following models for the 5-FU Clearance data:

go to slide 50

$$\boxed{CL = e^{\beta_0 + \beta_1 BSA_i + \beta_2 Dose_i} \varepsilon_i,\ i = 1,\ldots,26.}$$

$$\boxed{CL = \beta_0' + \beta_1' BSA_i + \beta_2' Dose_i + \varepsilon'_i,\ i = 1,\ldots,26.}$$

2. Perform a residual analysis for both models and visually compare*.
   - Histogram of standardized residuals
   - Normal QQ plot for standardized residuals
   - Scatter plot of standardized residuals vs. fitted values
   - Scatter plots of covariates vs. standardized residuals.

- Perform influential diagnostics for the both models and compare.

*Note: the MASS package and library are required to access the standardized and studentized residuals.

In class R exercise: 5-FU Clearance data (analyzed in class)

3. Obtain the MLE’s for the regression coefficient for the first model by using the following code:

```
LL.reg.fun <- function(b0,b1,b2,sigma) {
  # this function calculates the –log Likelihood for
  # the normal residuals of a linear regression
  # 5-FU log CL vs. SBSA and SDose (SBSA=BSA/1.83, SDose=1/1000)
  # it takes as argument a list with four components:
  # list=(b0,b1,b2,sigma)
  # elaborated by D. Hajducek, last updated Feb 7, 2016

  var <- logCL- b0 - b1 * SBSA - b2 * SDose
  pdf = dnorm(var, 0, sigma)
  -sum(log(pdf))
}


fit.mle <- mle(LL.reg.fun ,
               start=list(b0=-1,b1=1.5,b2=-.5,sigma=.3),
               method="L-BFGS-B",lower=c(-Inf,-Inf,-Inf 0),
               upper=c(Inf,Inf,Inf,Inf))
```

4. Play with different starting values to see the stability of the estimates.

In class R exercise: 5-FU Clearance data (analyzed in class)

Notes on the `method, lower` and `upper` options in `mle():`

```
fit.mle <- mle(LL.reg.fun ,
               start=list(b0=-1,b1=1.5,b2=-.5,sigma=.3),
               method="L-BFGS-B",lower=c(-Inf,-Inf,-Inf 0),
               upper=c(Inf,Inf,Inf,Inf))
```

- These options are common in optimization algorithms.
- In mle:
  - method: there are several methods available. In particular "L-BFGS-B" will perform a constrained optimization. For example, we want sigma to only have positive values. In case of specifying this method, we also need to add the `lower` and `upper` options.

  - lower, upper: these options allow us to specify the parameter constraints. In our example we specified as –Inf for b0,b1,b2 and 0 for sigma, and Inf for all of them.