# Introduction to Biostatistics The 2nd Lecture

Hiroyosh IWATA hiroiwata@g.ecc.u-tokyo.ac.jp

2020/4/24

## Simple regression analysis

Changes in one variable may affect another, such as the relationship between breeding and cultivation conditions and the growth of animals and plants. One of the statistical methods to model the relationship between these variables is regression analysis. By statistically modeling the relationship between variables, it becomes possible to understand the causal relationship that exists between variables, and to predict one variable from another. Here, first, we will discuss simple regression analysis that models the relationship between two variables as a "linear relationship". In this case, the mechanism of single regression analysis will be explained using the analysis of rice data (Zhao et al. 2011, Nature Communications 2: 467) as an example.

First, read the rice data in the same way as before. Before entering the following command, change your R working directory to the directory (folder) where the two input files (RiceDiversityPheno.csv, RiceDiversityLine.csv) are located.

```
# this data set was analyzed in Zhao 2011 (Nature Communications 2:467)
pheno <- read.csv("RiceDiversityPheno.csv")
line <- read.csv("RiceDiversityLine.csv")
line.pheno <- merge(line, pheno, by.x = "NSFTV.ID", by.y = "NSFTVID")
head(line.pheno)[,1:12]
```

```
##   NSFTV.ID GSOR.ID        IRGC.ID Accession.Name Country.of.origin  Latitu
de
## 1        1  301001 To be assigned       Agostano             Italy 41.8719
40
## 2        3  301003         117636  Ai-Chiao-Hong             China 27.9025
27
## 3        4  301004         117601       NSF-TV 4             India 22.9030
81
## 4        5  301005         117641       NSF-TV 5             India 30.4726
64
## 5        6  301006         117603       ARC 7229             India 22.9030
81
## 6        7  301007 To be assigned          Arias         Indonesia -0.7892
75
##   Longitude Sub.population     PC1     PC2     PC3     PC4
## 1  12.56738            TEJ -0.0486  0.0030  0.0752 -0.0076
## 2 116.87256            IND  0.0672 -0.0733  0.0094 -0.0005
## 3  87.12158            AUS  0.0544  0.0681 -0.0062 -0.0369
## 4  75.34424       AROMATIC -0.0073  0.0224 -0.0121  0.2602
## 5  87.12158            AUS  0.0509  0.0655 -0.0058 -0.0378
## 6 113.92133            TRJ -0.0293 -0.0027 -0.0677 -0.0085
```

Prepare analysis data by extracting only the data used for simple regression analysis from the read data. Here we analyze the relationship between plant height (Plant.height) and flowering timing (Flowering.time.at.Arkansas). In addition, principal component scores (PC1 to PC4)

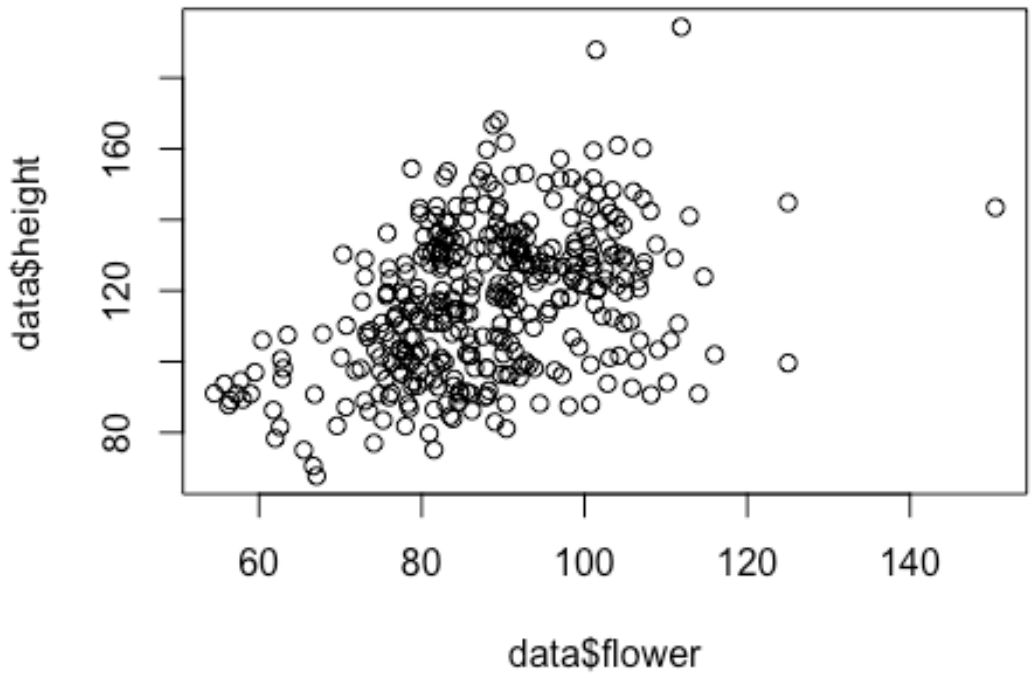
representing the genetic background to be used later are also extracted. Also, remove samples with missing values in advance.

```
# extract variables for regression analysis
data <- data.frame(
    height = line.pheno$Plant.height,
    flower = line.pheno$Flowering.time.at.Arkansas,
    PC1 = line.pheno$PC1,
    PC2 = line.pheno$PC2,
    PC3 = line.pheno$PC3,
    PC4 = line.pheno$PC4)
data <- na.omit(data)
head(data)
```

```
##     height    flower     PC1     PC2     PC3     PC4
## 1 110.9167  75.08333 -0.0486  0.0030  0.0752 -0.0076
## 2 143.5000  89.50000  0.0672 -0.0733  0.0094 -0.0005
## 3 128.0833  94.50000  0.0544  0.0681 -0.0062 -0.0369
## 4 153.7500  87.50000 -0.0073  0.0224 -0.0121  0.2602
## 5 148.3333  89.08333  0.0509  0.0655 -0.0058 -0.0378
## 6 119.6000 105.00000 -0.0293 -0.0027 -0.0677 -0.0085
```

First, visualize the relationship between two variables.

```
# look at the relationship between plant height and flowering time
plot(data$height ~ data$flower)
```



As shown in the above figure, the earlier the flowering time, the shorter the plant height, while the later the flowering time, the taller the plant height.

Let's create a simple regression model that explains the variation in plant height by the difference in flowering timing.

```
# perform single linear regression
model <- lm(height ~ flower, data = data)
```

The result of regression analysis (estimated model) is assigned to "model". Use the function "summary" to display the result of regression analysis.

```
# show the result
summary(model)
```

```
##
## Call:
## lm(formula = height ~ flower, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -43.846 -13.718   0.295  13.409  61.594
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 58.05464    6.92496   8.383 1.08e-15 ***
## flower       0.67287    0.07797   8.630  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19 on 371 degrees of freedom
## Multiple R-squared:  0.1672, Adjusted R-squared:  0.1649
## F-statistic: 74.48 on 1 and 371 DF,  p-value: < 2.2e-16
```

I will explain the results displayed by executing the above command in order.

Call:
lm(formula = height ~ flower, data = data)

This is a repeat of the command you entered earlier. If you get this output right after you type it, it does not seem to be useful information. However, if you make multiple regression models and compare them as described later, it may be useful because you can reconfirm the model employed in the analysis. Here, assuming that the plant height is $y_i$ and the flowering timing is $x_i$, the regression analysis is performed with the model

$$y_i = \mu + \beta x_i + \epsilon_i$$

As mentioned earlier, $x_i$ is called independent variable or explanatory variable, and $y_i$ is called dependent variable or response variable. $\mu$ and $\beta$ are called parameters of the regression model, and $\epsilon_i$ is called error. Also, $\mu$ is called population intercept and $\beta$ is called population regression coefficient.

In addition, since it is not possible to directly know the true values of the parameters $\mu$ and $\beta$ of the regression model, estimation is performed based on samples. The estimates of the parameters $\mu$ and $\beta$, which are estimated from the sample, are called sample intercept and sample regression coefficient, respectively. The values of $\mu$ and $\beta$ estimated from the samples are denoted by $m$ and $b$, respectively. Since $m$ and $b$ are values estimated from the samples,

they are random variables that vary depending on the samples selected by chance. Therefore, it follows a probability distribution. Details will be described later.

Residuals:
Min 1Q Median 3Q Max
-43.846 -13.718 0.295 13.409 61.594

This output gives an overview of the distribution of residuals. You can use this information to check the regression model. For example, the model assumes that the expected value (average) of the error is 0. You can check whether the median is close to it. You can also check whether the distribution is symmetrical around 0, i.e., whether the maximum and minimum or the first and third quantiles have almost the same value. In this example, the maximum value is slightly larger than the minimum value, but otherwise no major issues are found.

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) 58.05464 6.92496 8.383 1.08e-15 ***
flower 0.67287 0.07797 8.630 < 2e-16 ***
—
Signif. codes: 0 ‘’ ***0.001*** ‘’ *0.01* ‘’ 0.05 ‘.’ 0.1 ‘ ’ 1

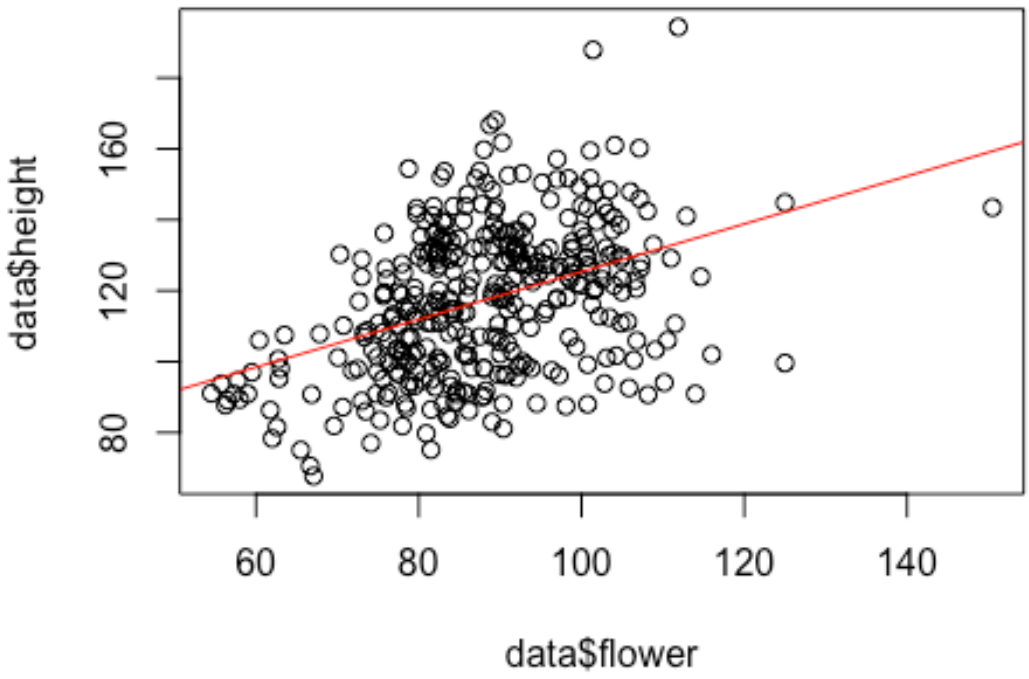
The estimates of parameters $\mu$ and $\beta$, i.e., $m$ and $b$, and their standard errors, $t$ values and $p$ values are shown. Asterisks at the endo of each line represent significance levels. One star represents 5%, two stars 1%, and three stars 0.1%.

Residual standard error: 19 on 371 degrees of freedom
Multiple R-squared: 0.1672, Adjusted R-squared: 0.1649
F-statistic: 74.48 on 1 and 371 DF, p-value: < 2.2e-16

The first line shows the standard deviation of the residuals. This is the value represented by $s$, where $s^2$ is the estimated value of the error variance $\sigma^2$. The second line is the determination coefficient $R^2$. This index and the adjusted $R^2$ represent how well the regression explain the variation of y. The third line is the result of the F test that represents the significance of the regression model. It is a test under the hypothesis (null hypothesis) that all regression coefficients are 0, and if this $p$ value is very small, the null hypothesis is rejected and the alternative hypothesis (regression coefficient is not 0) is taken to be adopted.

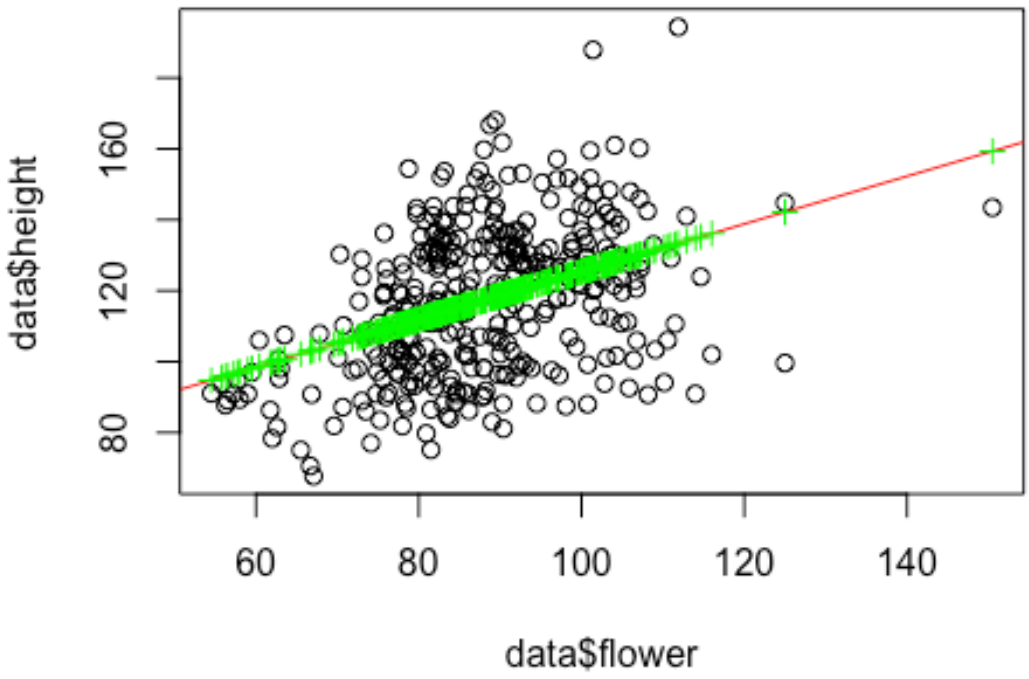
Let’s look at the results of regression analysis graphically. First, draw a scatter plot and draw a regression line.

```
# again, plot the two variables
plot(data$height ~ data$flower)
abline(model, col = "red")
```




Next, calculate and plot the value of $y$ when the data is fitted to the regression model.

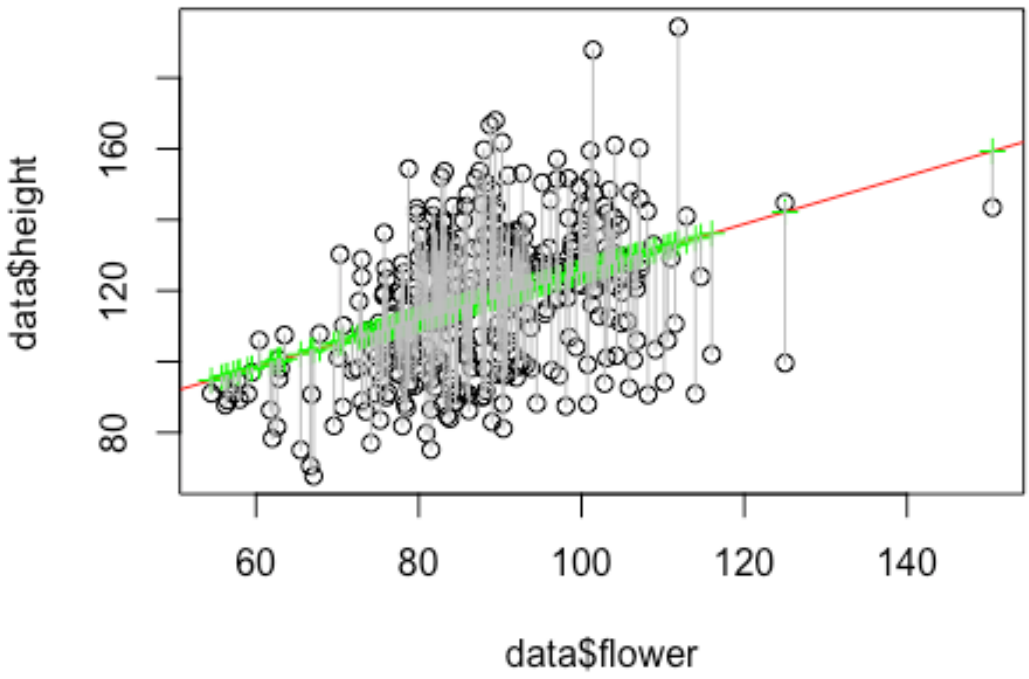
```
# calculate fitted values
height.fit <- fitted(model)
plot(data$height ~ data$flower)
abline(model, col = "red")
points(data$flower, height.fit, pch = 3, col = "green")
```




The values of $y$ calculated by fitting the model all lie on a straight line.

An observed value $y$ is expressed as the sum of the variation explained by the regression model and the error which is not explained by the regression. Let's visualize the error in the figure and check the relationship.

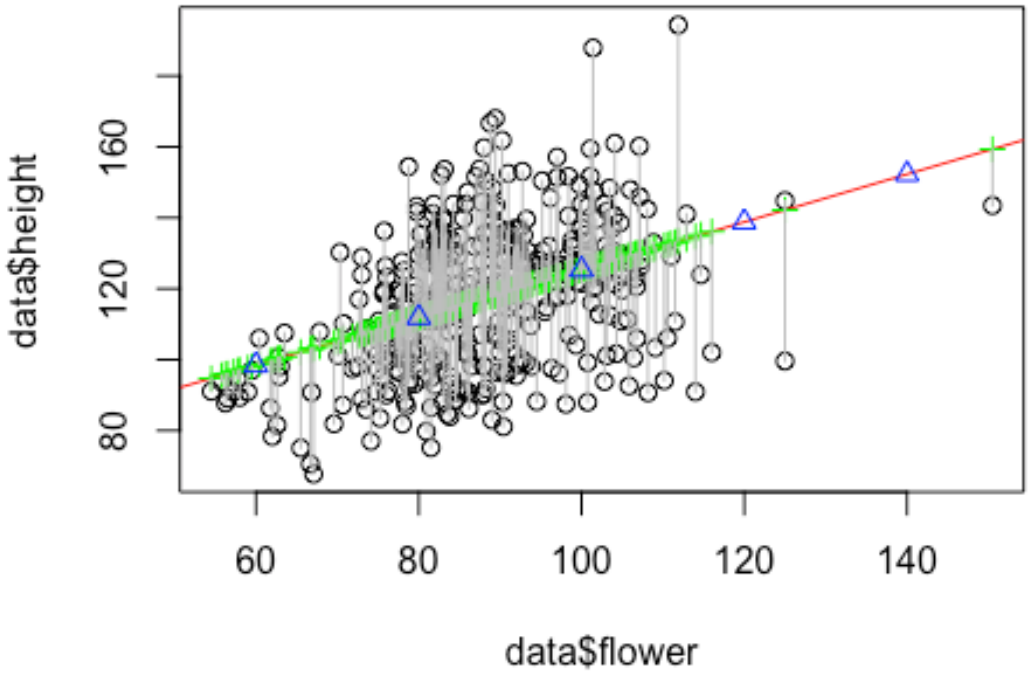
```
# plot residuals
plot(data$height ~ data$flower)
abline(model, col = "red")
points(data$flower, height.fit, pch = 3, col = "green")
segments(data$flower, height.fit,
         data$flower, height.fit + resid(model), col = "gray")
```




The value of $y$ is expressed as the sum of the values of $y$ calculated by fitting the model (green points) and the residuals of the model (gray line segments)

Let's use a regression model to predict $y$ for $x = (60,80,\ldots,140)$, which are not actually observed.

```
# predict unknown data
height.pred <- predict(model, data.frame(flower = seq(60, 140, 20)))

plot(data$height ~ data$flower)
abline(model, col = "red")
points(data$flower, height.fit, pch = 3, col = "green")
segments(data$flower, height.fit,
         data$flower, height.fit + resid(model), col = "gray")
points(seq(60, 140, 20), height.pred, pch = 2, col = "blue")
```




All the predicted values will locate again on the line.

### Quiz 1

Now, let's solve a practice question here. Practice questions will be presented in the lecture.

Go to https://www.menti.com/ and type in the number I will tell you in the lecture. Then, register your nickname and wait for the quiz to start.

## Method for calculating the parameters of a regression model

Here we will explain how to calculate a regression model. Also, let's calculate the regression coefficients while actually using the R command.

As mentioned earlier, the simple regression model is

$$y_i = \mu + \beta x_i + \epsilon_i$$

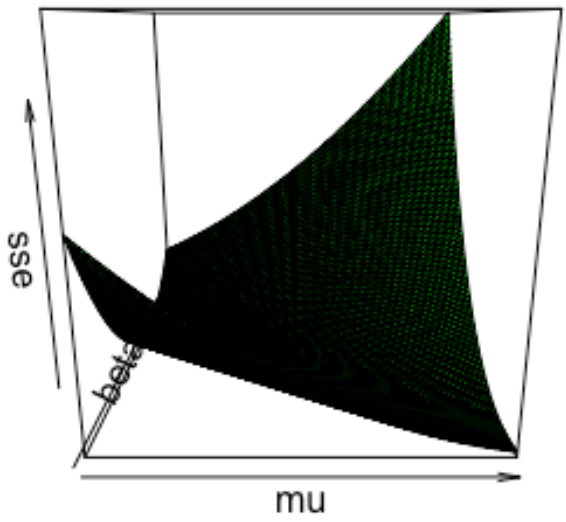
There are various criteria for what is considered "optimal", but here we consider minimizing the error $\epsilon_i$ across the data. Since errors can take both positive and negative values, errors cancels each other in their simple sum. So, we consider minimizing the sum of squared error (SSE). That is, consider $\mu$ and $\beta$ that minimize the following equation:

$$SSE = \sum_{i=1}^{n} \epsilon_i^2 = \sum_{i=1}^{n} (y_i - (\mu + \beta x_i))^2$$

式(1)

The following figure shows the change in SSE for various values of $\mu$ and $\beta$. The commands to draw the figure is a little complicated, but they are as follows:

```
#visualize the plane for optimization
x <- data$flower
y <- data$height
mu <- seq(0, 100, 1)
beta <- seq(0, 2, 0.02)
sse <- matrix(NA, length(mu), length(beta))
for(i in 1:length(mu)) {
    for(j in 1:length(beta)) {
        sse[i, j] <- sum((y - mu[i] - beta[j] * x)^2)
    }
}
persp(mu, beta, sse, col = "green")
```




Draw the graph using “plotly” package

```
# draw the figure with plotly
df <- data.frame(mu, beta, sse)
plot_ly(data = df, x = ~mu, y = ~beta, z = ~sse) %>%
add_surface()
```

It should be noted that at the point where *SSE* becomes the minimum in Figure 3, *SSE* should not change (the slope of the tangent is zero) even when $\mu$ or $\beta$ changes slightly. Therefore, the

coordinates of the minimum point can be determined by partially differentiating the equation (1) with $\mu$ and $\beta$, and setting the value to zero. That is,

$$\frac{\partial SSE}{\partial \mu} = 0, \frac{\partial SSE}{\partial \beta} = 0$$

We should obtain the values of $\mu$ and $\beta$ to satisfy these. The method of calculating the parameters of a regression model through minimizing the sum of squares of errors in this way is called the least squares method.

Note that $\mu$ minimizing SSE is

$$\frac{\partial SSE}{\partial \mu} = -2\sum_{i=1}^{n}(y_i - \mu - \beta x_i) = 0$$

$$\Leftrightarrow \sum_{i=1}^{n} y_i - n\mu - \beta\sum_{i=1}^{n} x_i = 0$$

$$\Leftrightarrow \mu = \frac{\sum_{i=1}^{n} y_i}{n} - \beta\frac{\sum_{i=1}^{n} x_i}{n} = \overline{y} - \beta\overline{x}$$

Also,$\beta$ minimizing $SSE$ is

$$\frac{\partial SSE}{\partial \beta} = -2\sum_{i=1}^{n} x_i\,(y_i - \mu - \beta x_i) = 0$$

$$\Leftrightarrow \sum_{i=1}^{n} x_i\,y_i - \mu\sum_{i=1}^{n} x_i - \beta\sum_{i=1}^{n} x_i^2 = 0$$

$$\Leftrightarrow \sum_{i=1}^{n} x_i\,y_i - n(\overline{y} - \beta\overline{x})\overline{x} - \beta\sum_{i=1}^{n} x_i^2 = 0$$

$$\Leftrightarrow \sum_{i=1}^{n} x_i\,y_i - n\overline{xy} - \beta(\sum_{i=1}^{n} x_i^2 - n\overline{x}^2) = 0$$

$$\Leftrightarrow \beta = \frac{\sum_{i=1}^{n} x_i\,y_i - n\overline{xy}}{\sum_{i=1}^{n} x_i^2 - n\overline{x}^2} = \frac{SSXY}{SSX}$$

Here, $SSXY$ and $SSX$ are sum of products of deviation in $x$ and $y$ and deviation the sum of squares of deviation in $x$, respectively.

$$SSXY = \sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})$$

$$= \sum_{i=1}^{n} x_i\,y_i - \overline{x}\sum_{i=1}^{n} y_i - \overline{y}\sum_{i=1}^{n} x_i + n\overline{xy}$$

$$= \sum_{i=1}^{n} x_i\,y_i - n\overline{xy} - n\overline{yx} + n\overline{xy}$$

$$= \sum_{i=1}^{n} x_i y_i - n\overline{xy}$$

$$SSX = \sum_{i=1}^{n} (x_i - \overline{x})^2$$

$$= \sum_{i=1}^{n} x_i^2 - 2\overline{x} \sum_{i=1}^{n} x_i - n\overline{x}^2$$

$$= \sum_{i=1}^{n} x_i y_i - 2n\overline{x}^2 + n\overline{x}^2$$

$$= \sum_{i=1}^{n} x_i y_i - n\overline{x}^2$$

The values of $\mu$ and $\beta$ minimizing $SSE$ are the estimates of the parameters, and let the estimates be represented by $m$ and $b$. That is,

$$b = \frac{SSXY}{SSX}$$

$$m = \overline{y} - b\overline{x}$$

Now let's calculate the regression coefficients based on the above equation. First, calculate the sum of products f deviation and the sum of squares of deviation.

```
# calculate sum of squares (ss) of x and ss of xy
n <- length(x)
ssx <- sum(x^2) - n * mean(x)^2
ssxy <- sum(x * y) - n * mean(x) * mean(y)
```

First we calculate the slope $b$.

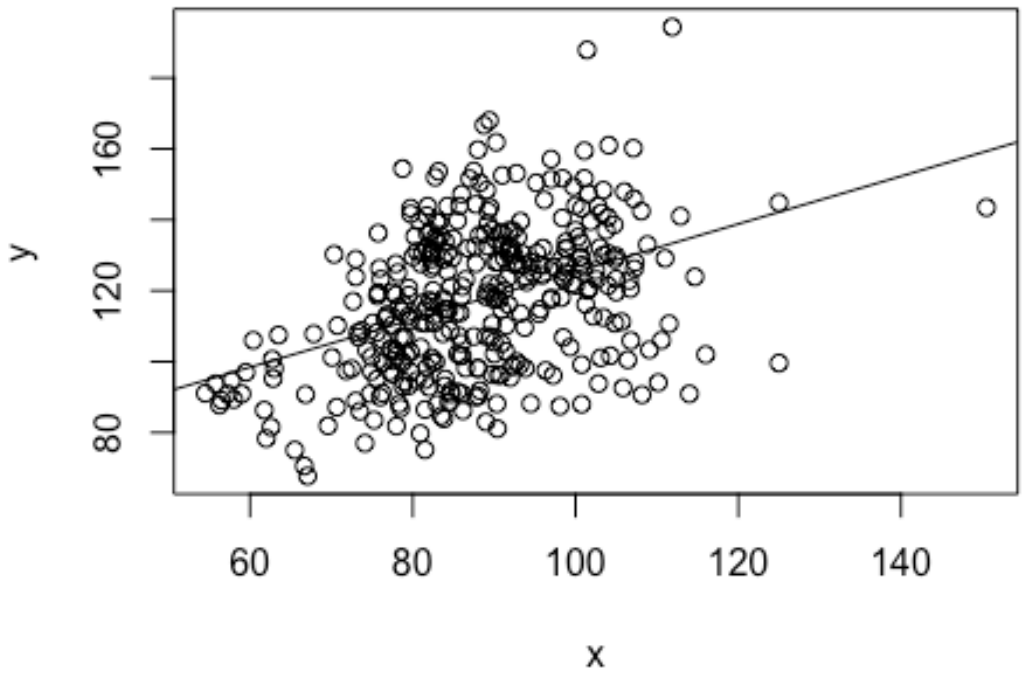
```
# calculate b
b <- ssxy / ssx
b

## [1] 0.6728746
```

Then calculate the intercept $m$

```
# calculate m
m <- mean(y) - b * mean(x)
m

## [1] 58.05464
```

Let's draw a regression line based on the calculated estimates.

```
# draw scatter plot and regression line
plot(y ~ x)
abline(m, b)
```



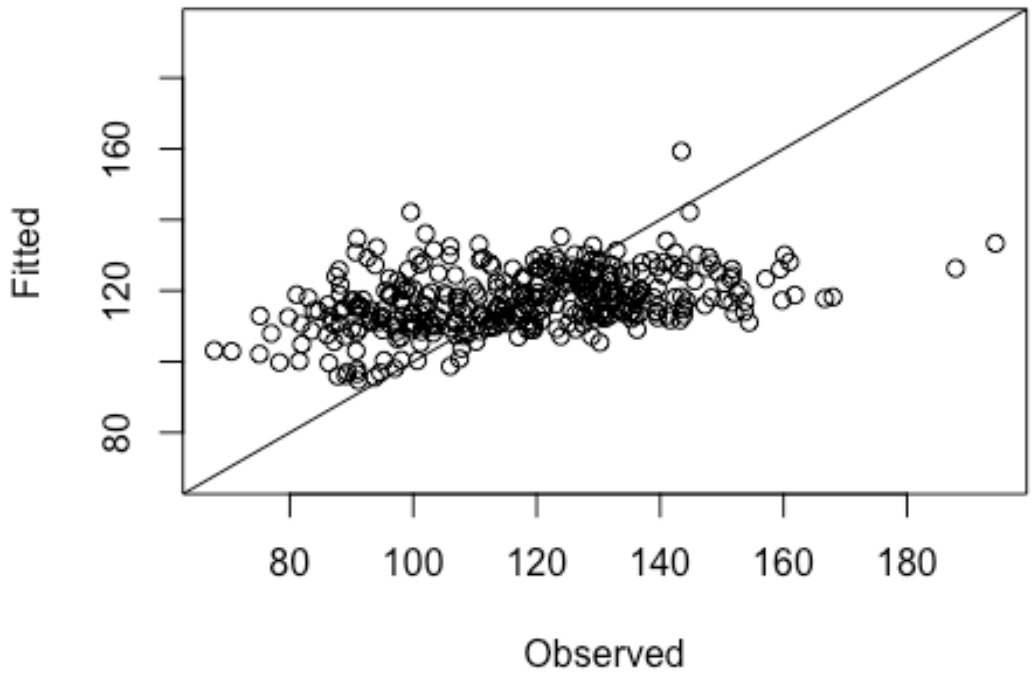

Let's make sure that the same regression line was obtained as the function lm which we used earlier.

Note that once the regression parameters $\mu$ and $\beta$ are estimated, it is possible to calculate $\hat{y}_i$, which is the value of $y$ corresponding to a given $x_i$. That is,

$$\hat{y}_i = m + bx_i$$

This makes it possible to calculate the value of $y$ when the model is fitted to the observed $x$, or to predict $y$ if only the value of $x$ is known. Here, let's calculate the value of $y$ when the model is fitted to the observed $x$, and draw scatter points on the figure drawn earlier.

```
# calculate fitted values
y.hat <- m + b * x
lim <- range(c(y, y.hat))
plot(y, y.hat, xlab = "Observed", ylab = "Fitted", xlim = lim, ylim = lim)
abline(0, 1)
```




Let's calculate the correlation coefficient between the two to find the degree of agreement between the observed and fitted values.

```
# calculate correlation between observed and fitted values
cor(y, y.hat)

## [1] 0.408888
```

In fact, the square of this correlation coefficient is the proportion of the variation of $y$ explained by the regression (coefficient of determination, $R^2$ value). Let's compare these two statistics.

```
# compare the square of the correlation and R2
cor(y, y.hat)^2

## [1] 0.1671894

summary(model)

##
## Call:
## lm(formula = height ~ flower, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -43.846 -13.718   0.295  13.409  61.594
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 58.05464    6.92496   8.383 1.08e-15 ***
## flower       0.67287    0.07797   8.630  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19 on 371 degrees of freedom
## Multiple R-squared:  0.1672, Adjusted R-squared:  0.1649
## F-statistic: 74.48 on 1 and 371 DF,  p-value: < 2.2e-16
```

### Quiz 2

Now, let's solve a practice question here. Practice questions will be presented in the lecture.

If you close the page of the quiz, go to https://www.menti.com/ and type in the number I will tell you in the lecture.

## Significance test of a regression model

When the linear relationship between variables is strong, a regression line fit well to the observations, and the relationship between both variables can be well modeled by a regression line. However, when a linear relationship between variables is not clear, modeling with a regression line does not work well. Here, as a method to objectively confirm the goodness of fit of the estimated regression model, we will explain a test using analysis of variance.

First, let's go back to the simple regression again.

```
model <- lm(height ~ flower, data = data)
```

The significance of the obtained regression model can be tested using the function anova.

```
# analysis of variance of regression
anova(model)
```

```
## Analysis of Variance Table
##
## Response: height
##            Df Sum Sq Mean Sq F value    Pr(>F)
## flower      1  26881 26881.5  74.479 < 2.2e-16 ***
## Residuals 371 133903   360.9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

As a result of the analysis of variance, the term of flowering time is highly significant ($p < 0.001$), and the goodness of fit of the regression model that the flowering timing influences plant height is confirmed.

Analysis of variance for regression models involves the following calculations: First of all, "Sum of squares explained by regression" can be calculated as follows.

$$SSR = \sum_{i=1}^{n} (\hat{y}_i - \overline{y})^2$$

$$= \sum_{i=1}^{n} (\mu + bx_i - (\mu + b\overline{x}))^2$$

$$= b^2 \sum_{i=1}^{n} (x_i - \overline{x})^2$$

$$= b^2 \cdot SSX = b \cdot SSXY$$

Also, the sum of squares of deviation from the mean of the observed values $y$ is expressed as the sum of the sum of squares $SSR$ explained by the regression and the residual sum of squares $SSE$. That is,

$$SSY = \sum_{i=1}^{n} (y_i - \overline{y})^2$$

$$= \sum_{i=1}^{n} (y_i - \hat{y}_i + \hat{y}_i - \overline{y})^2$$

$$= \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \sum_{i=1}^{n} (\hat{y}_i - \overline{y})^2$$

$$= SSE + SSR$$

$$\because 2 \sum_{i=1}^{n} (y_i - \hat{y}_i)(\hat{y}_i - \overline{y})$$

$$= 2 \sum_{i=1}^{n} (y_i - m - bx_i)(m + bx_i - (m + b\overline{x}))$$

$$= 2b \sum_{i=1}^{n} (y_i - (\overline{y} - b\overline{x}) - bx_i)(x_i - \overline{x})$$

$$= 2b \sum_{i=1}^{n} (y_i - \overline{y} - b(x_i - \overline{x}))(x_i - \overline{x})$$

$$= 2b(SSXY - b \cdot SSX) = 0$$

Let's actually calculate it using the above equation. First, calculate $SSR$ and $SSE$.

```
# calculate sum of squares of regression and error
ssr <- b * ssxy
ssr

## [1] 26881.49

ssy <- sum(y^2) - n * mean(y)^2
sse <- ssy - ssr
sse

## [1] 133903.2
```

Next, calculate the mean squares, which is of the sum of squares divided by the degrees of freedom.

```
# calculate mean squares of regression and error
msr <- ssr / 1
msr

## [1] 26881.49

mse <- sse / (n - 2)
mse

## [1] 360.9251
```

Finally, the mean square of the regression is divided by the mean square of the error to calculate the *F* value. Furthermore, calculate the *p* value corresponding to the calculated F value.

```
# calculate F value
f.value <- msr / mse
f.value

## [1] 74.47943

# calculate p value for the F value
1 - pf(f.value, 1, n - 2)

## [1] 2.220446e-16
```

The results obtained are in agreement with the results calculated earlier using the function anova.

The results of regression analysis are included in the results of regression analysis displayed using the function “summary”.

```
# check the summary of the result of regression analysis
summary(model)

##
## Call:
## lm(formula = height ~ flower, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -43.846 -13.718   0.295  13.409  61.594
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 58.05464    6.92496   8.383 1.08e-15 ***
## flower       0.67287    0.07797   8.630  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19 on 371 degrees of freedom
## Multiple R-squared:  0.1672, Adjusted R-squared:  0.1649
## F-statistic: 74.48 on 1 and 371 DF,  p-value: < 2.2e-16
```

“Residual standard error” is the square root of the mean square of the residual.

```
# square root of mse
sqrt(mse)

## [1] 18.99803
```

"Multiple R-squared" ($R^2$) is a value called the coefficient of determination, which is the ratio of $SSR$ to $SSY$.

```
# R squared
ssr / ssy

## [1] 0.1671894
```

"Adjusted R-squared" ($R^2_{adj}$) is a value called the adjusted coefficient of determination, which can be calculated as follows.

```
# adjusted R squared
(ssy / (n - 1) - mse) / (ssy / (n - 1))

## [1] 0.1649446
```

Also, "F-statistic" matches the $F$ value and its $p$ value which are expressed as the effect of flowering time in the analysis of variance. In addition, the t value calculated for the regression coefficient of the flowering time term is squared to obtain the $F$ value (8.6302 = 74.477).

$R^2$ and $R^2_{adj}$ can also be expressed using $SSR$, $SSY$, and $SSE$ as follows.

$$R^2 = \frac{SSR}{SSY} = 1 - \frac{SSE}{SSY}$$

$$R^2_{adj} = 1 - \frac{n-1}{n-p} \cdot \frac{SSE}{SSY}$$

Here, $p$ is the number of parameters included in the model, and $p = 2$ for a simple regression model. It can be seen that $R^2_{adj}$ has a larger amount of adjustment (the residual sum of squares is underestimated) as the number of parameters included in the model increases.

### Quiz 3

Now, let's solve a practice question here. Practice questions will be presented in the lecture.

If you close the page of the quiz, go to https://www.menti.com/ and type in the number I will tell you in the lecture.

## Distribution that estimated value of regression coefficient follows

As mentioned earlier, the estimates $b$ and $m$ of the regression coefficients $\mu$ and $\beta$ are values estimated from samples and are random variables that depend on the samples chosen by chance. Thus, estimates $b$ and $m$ have probabilistic distributions. Here we consider the distributions that the estimates follow.

The estimate $b$ follows the normal distribution:

$$b \sim N(\beta, \frac{\sigma^2}{SSX})$$

Here, $\sigma^2$ is the residual variance $\sigma^2 = Var(y_i) = Var(e_i)$.

The esimate $m$ follows the normal distribution:

$$m \sim N(\mu, \sigma^2[\frac{1}{n} + \frac{\overline{x}^2}{SSX}])$$

Although the true value of the error variance $\sigma^2$ is unknown, it can be replaced by the residual variance $s^2$. That is,

$$s^2 = \frac{SSE}{n-2}$$

This value is the mean square of the residuals calculated during the analysis of variance.

At this time, statistics on $b$

$$t = \frac{b - \beta_0}{s/\sqrt{SSX}}$$

follows the t distribution with n – 2 degrees of freedom under the null hypothesis:

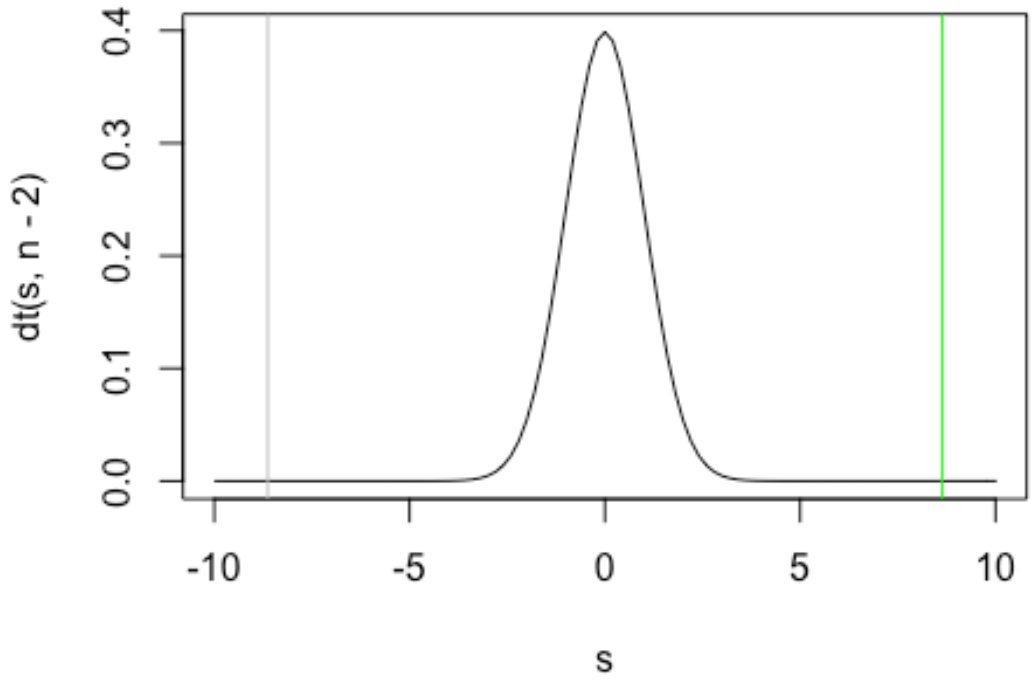
$$H_0: \beta = \beta_0$$

First, test for null hypothesis $H_0: \beta = 0$ for $b$.

```
# test beta = 0
t.value <- (b - 0) / sqrt(mse/ssx)
t.value

## [1] 8.630147
```

This statistic follows the t distribution with 371 degrees of freedom under the null hypothesis. Draw a graph of the distribution

```
# visualize the t distribution under H0
s <- seq(-10, 10, 0.2)
plot(s, dt(s, n - 2), type = "l")
abline(v = t.value, col = "green")
abline(v = - t.value, col = "gray")
```




From the distribution that follows under the null hypothesis, the value of $t$ that is being obtained appears to be large. Let's perform a $t$ test. Note that it is a two-tailed test.

```
# perform t test
2 * (1 - pt(abs(t.value), n - 2)) # two-sided test

## [1] 2.220446e-16
```

The results of this test were already displayed as regression analysis results.

```
# check the summary of the model
summary(model)

##
## Call:
## lm(formula = height ~ flower, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -43.846 -13.718   0.295  13.409  61.594
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 58.05464    6.92496   8.383 1.08e-15 ***
## flower       0.67287    0.07797   8.630  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 19 on 371 degrees of freedom
## Multiple R-squared:  0.1672, Adjusted R-squared:  0.1649
## F-statistic: 74.48 on 1 and 371 DF,  p-value: < 2.2e-16
```

The hypothesis test performed above can be performed for any $\beta_0$. For example, let's test for null hypothesis $H_0: \beta = 0.5$.

```
# test beta = 0.5
t.value <- (b - 0.5) / sqrt(mse/ssx)
t.value

## [1] 2.217253

2 * (1 - pt(abs(t.value), n - 2))

## [1] 0.02721132
```

The result is significant at the 5% level.

Now let us test for $m$. First, let's test the null hypothesis $H_0: m = 0$.

```
# test mu = 0
t.value <- (m  - 0) / sqrt(mse * (1/n + mean(x)^2 / ssx))
t.value

## [1] 8.383389

2 * (1 - pt(abs(t.value), n - 2))

## [1] 1.110223e-15
```

This result was also already calculated.

```
# check the summary of the model again
summary(model)

##
## Call:
## lm(formula = height ~ flower, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -43.846 -13.718   0.295  13.409  61.594
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 58.05464    6.92496   8.383 1.08e-15 ***
## flower       0.67287    0.07797   8.630  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19 on 371 degrees of freedom
## Multiple R-squared:  0.1672, Adjusted R-squared:  0.1649
## F-statistic: 74.48 on 1 and 371 DF,  p-value: < 2.2e-16
```

(It may be due to the rounding error that the p value does not match completely)

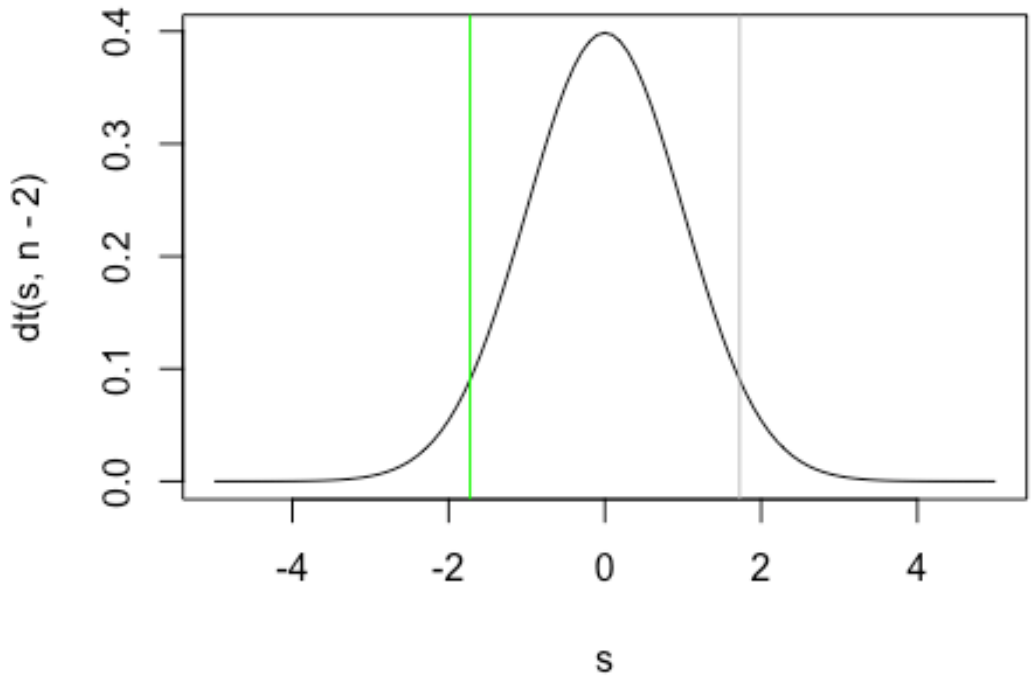
Finally, let's test for the null hypothesis $H_0$: $m = 50$.

```
# test mu = 70
t.value <- (m  - 70) / sqrt(mse * (1/n + mean(x)^2 / ssx))
t.value

## [1] -1.724971

2 * (1 - pt(abs(t.value), n - 2))

## [1] 0.08536545
```

The result was not significant even at the 5% level.

Draw a graph of the distribution unser the null hypothesis.

```
# visualize the t distribution under H0
s <- seq(-5, 5, 0.1)
plot(s, dt(s, n - 2), type = "l")
abline(v = t.value, col = "green")
abline(v = - t.value, col = "gray")
```




### Quiz 4

Now, let's solve a practice question here. Practice questions will be presented in the lecture.

If you close the page of the quiz, go to https://www.menti.com/ and type in the number I will tell you in the lecture.

## Confidence intervals for regression coefficients and fitted values

Function "predict" has various functions. First, let's use the function with the estimated regression model. Then, the values of y when the model fitted to observed data are calculated. The values $\hat{y}$ are exactly the same as calculated by the function "fitted".

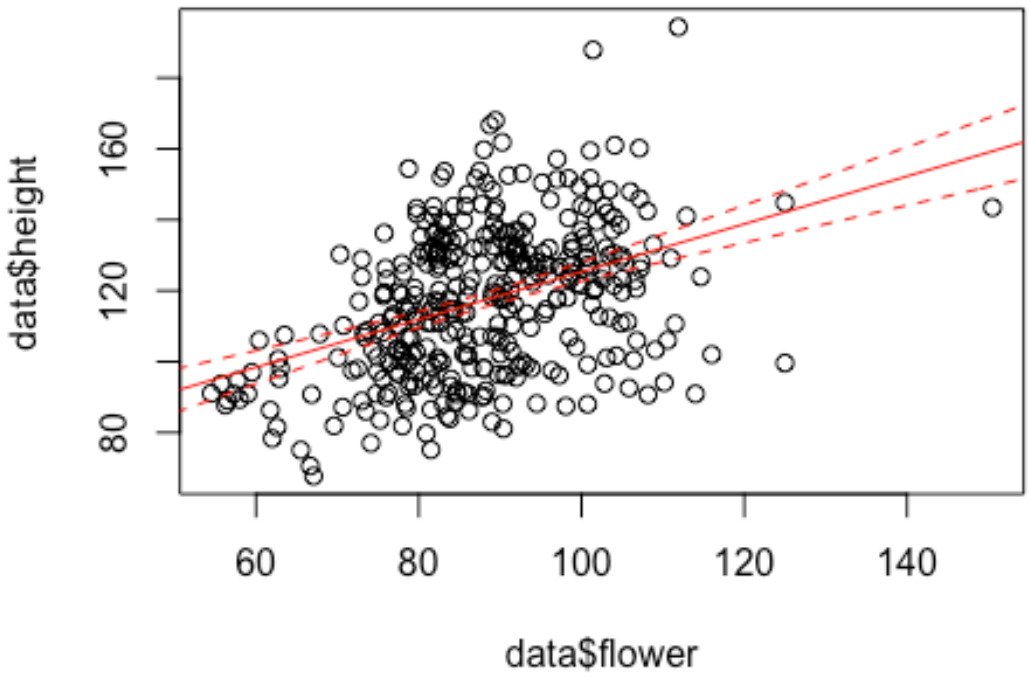
```
# fitted values
pred <- predict(model)
head(pred)

##        1        2        3        4        5        6
## 108.5763 118.2769 121.6413 116.9312 117.9966 128.7065

head(fitted(model))

##        1        2        3        4        5        6
## 108.5763 118.2769 121.6413 116.9312 117.9966 128.7065
```

By setting the options "interval" and "level", you can calculate the confidence interval (the 95 confidence interval at the default setting) of y at the specified significance level when fitting the model.

```
# calculate confidence interval
pred <- predict(model, interval = "confidence", level = 0.95)
head(pred)

##        fit      lwr      upr
## 1 108.5763 105.8171 111.3355
## 2 118.2769 116.3275 120.2264
## 3 121.6413 119.4596 123.8230
## 4 116.9312 114.9958 118.8665
## 5 117.9966 116.0540 119.9391
## 6 128.7065 125.4506 131.9623
```

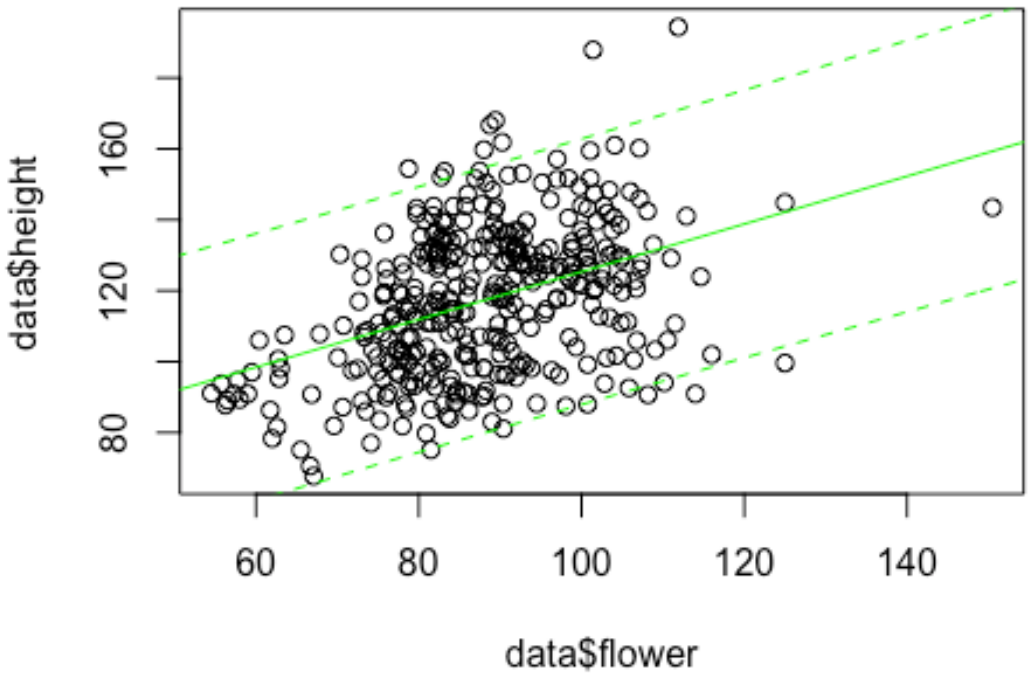
Let's vidualize the 95% confidence interval of y using the function "predict".

```
# draw confidence bands
pred <- data.frame(flower = 50:160)
pc <- predict(model, interval = "confidence", newdata = pred)
plot(data$height ~ data$flower)
matlines(pred$flower, pc, lty = c(1, 2, 2), col = "red")
```




Next, we will consider the case of predicting unknown data. Let us use the predict function to draw the 95% confidence interval of the predicted value, i.e., the value of $y$ that would be observed for the unknown data, i.e., the predicted value $\tilde{y}$, is subject to additional variation due to error.

```
pc <- predict(model, interval= "prediction", newdata = pred)
plot(data$height ~ data$flower)
matlines(pred$flower, pc, lty = c(1, 2, 2), col = "green")
```




With the added error, the variability of the predicted $\tilde{y}$ is greater than the estimated $\hat{y}$.

Note that for a particular $x$, the confidence intervals for the estimated y, i.e., $\hat{y}$, and the predicted y, i.e., $\tilde{y}$, can be found as follows. Here, we find the *99%* confidence interval when $x =$ 120.

```
# estimate the confidence intervals for the estimate and prediction of y
pred <- data.frame(flower = 120)
predict(model, interval = "confidence", newdata = pred, level = 0.99)

##        fit      lwr      upr
## 1 138.7996 131.8403 145.7589

predict(model, interval = "prediction", newdata = pred, level = 0.99)

##        fit      lwr      upr
## 1 138.7996 89.12106 188.4781
```

### Quiz 5

Now, let's solve a practice question here. Practice questions will be presented in the lecture.

If you close the page of the quiz, go to https://www.menti.com/ and type in the number I will tell you in the lecture.

## Polynomial regression model and multiple regression model

So far, we have applied to the data a regression model that represents the relationship between the two variables with a straight line. Let's extend the regression model a bit.

First, let's perform regression by a method called polynomial regression. In polynomial regression,

$$y_i = \mu + \beta_1 x_i + \beta_2 x_i^2 + \cdots + \beta_p x_i^p + \epsilon_i$$

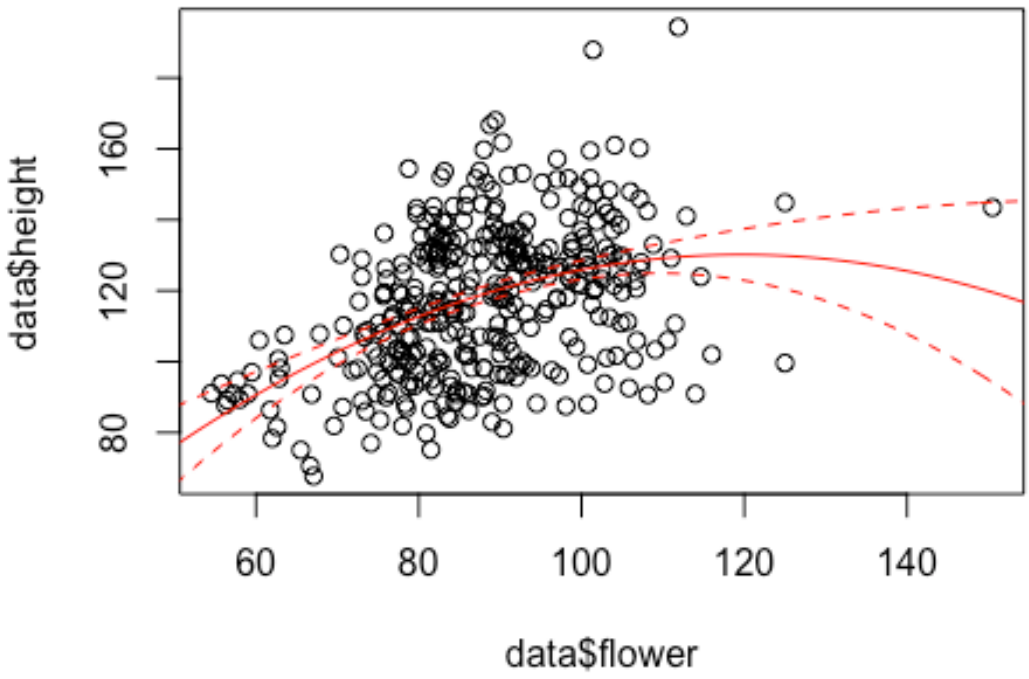
In this way, regression is performed using the second or higher order terms of $x$. First, let's perform regression using the first and second terms of $x$.

```
# polynomial regression
model.quad <- lm(height ~ flower + I(flower^2), data = data)
summary(model.quad)

##
## Call:
## lm(formula = height ~ flower + I(flower^2), data = data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -39.57 -13.60   0.97  12.91  64.83
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -29.082326  27.019440  -1.076 0.282473
## flower        2.662663   0.601878   4.424 1.28e-05 ***
## I(flower^2)  -0.011130   0.003339  -3.333 0.000945 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.74 on 370 degrees of freedom
## Multiple R-squared:  0.1915, Adjusted R-squared:  0.1871
## F-statistic: 43.81 on 2 and 370 DF,  p-value: < 2.2e-16
```

It can be seen that the proportion of variation of $y$(coefficient of determination $R^2$) explained by the polynomial regression model is larger than that of the simple regression model.

Although this will be mentioned later, you should not judge that the polynomial regression model is excellent only with this value. This is because a polynomial regression model has more parameters than a simple regression model, and you have more flexibility when fitting the model to data. It is easy to improve the fit of the model to the data by increasing the flexibility. In extreme cases, the model can be completely fitted to the data with as many parameters as the size of data (In that case, the coefficient pf determination R^2 completely matches 1). Therefore, careful selection of some statistical criteria is required when selecting the best model. This will be discussed later.

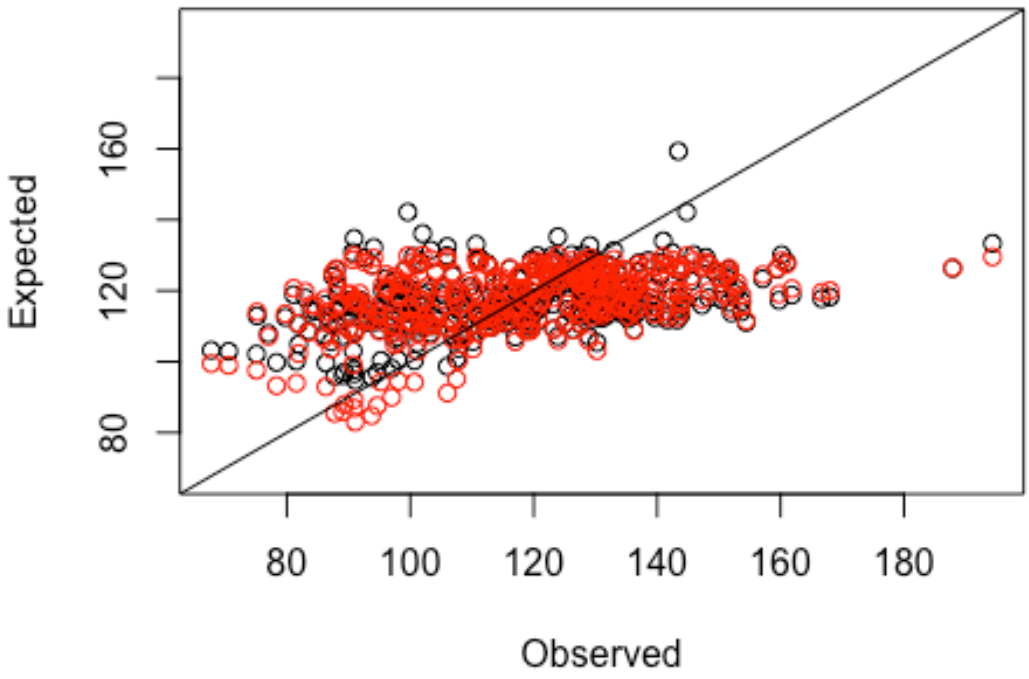
```
# plot(data$height ~ data$flower)
pred <- data.frame(flower = 50:160)
pc <- predict(model.quad, interval = "confidence", newdata = pred)
plot(data$height ~ data$flower)
matlines(pred$flower, pc, lty = c(1, 2, 2), col = "red")
```




When the timing of flowering is over 120 days after sowing, it can be seen that the reliability of the model is low.

Now let's draw the result of polynomial regression with confidence intervals.

```
# compare predicted and observed values
lim <- range(c(data$height, fitted(model), fitted(model.quad)))
plot(data$height, fitted(model),
        xlab = "Observed", ylab = "Expected",
        xlim = lim, ylim = lim)
points(data$height, fitted(model.quad), col = "red")
abline(0, 1)
```




The above figure represents relationship between fitted value and observed value of the simple regression model (black) and the second-order polynomial model (red).

Let's test if the improvement in the explanatory power of the second-order polynomial model is statistically significant. The significance is tested with $F$ test whether the difference between the residual sum of squares of the two models is sufficiently large compared to the residual sum of squares of the model containing the other (here, Model 2 contains Model 1).

```
# compare error variance between two models
anova(model, model.quad)

## Analysis of Variance Table
##
## Model 1: height ~ flower
## Model 2: height ~ flower + I(flower^2)
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    371 133903
## 2    370 129999  1    3903.8 11.111 0.0009449 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The results show that the difference in residual variance between the two models is highly significant ($p < 0.001$). In other words, Model 2 has significantly more explanatory power than Model 1.

Now let's fit a third-order polynomial regression model and test if it is significantly more descriptive than a second-order model.

```r
# extend polynormial regression model to a higher dimensional one...
model.cube <- lm(height ~ flower + I(flower^2) + I(flower^3), data = data)
summary(model.cube)
```

```
##
## Call:
## lm(formula = height ~ flower + I(flower^2) + I(flower^3), data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -39.699 -13.705   1.031  13.240  65.840
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.001e+02  8.541e+01  -1.172   0.2419
## flower       5.029e+00  2.765e+00   1.818   0.0698 .
## I(flower^2) -3.664e-02  2.929e-02  -1.251   0.2118
## I(flower^3)  8.898e-05  1.015e-04   0.877   0.3813
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.75 on 369 degrees of freedom
## Multiple R-squared:  0.1931, Adjusted R-squared:  0.1866
## F-statistic: 29.44 on 3 and 369 DF,  p-value: < 2.2e-16
```

```r
# compare error variance between two models
anova(model.quad, model.cube)
```

```
## Analysis of Variance Table
##
## Model 1: height ~ flower + I(flower^2)
## Model 2: height ~ flower + I(flower^2) + I(flower^3)
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    370 129999
## 2    369 129729  1    270.17 0.7685 0.3813
```

The 3rd-order model has a slightly better explanatory power than the 2nd-order model. However, the difference is not statistically significant. In other words, it turns out that extending a second-order model to a third-order model is not a good idea.

Finally, let's apply the multiple linear regression model:

$$y_i = \mu + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_p x_{pi} + \epsilon_i$$

In this way, regression is performed using multiple explanatory variables ($x_{1i}, x_{2i}, \ldots, x_{pi}$). In the first lecture, I confirmed in the graph that the height varies depending on the difference in genetic background. Here, we will create a multiple regression model that explains plant height using genetic backgrounds (PC1 to PC4) expressed as the scores of four principal components.

```r
# multi-linear regression with genetic background
model.wgb <- lm(height ~ PC1 + PC2 + PC3 + PC4, data = data)
summary(model.wgb)
```

```
##
## Call:
## lm(formula = height ~ PC1 + PC2 + PC3 + PC4, data = data)
```

```
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -45.89 -11.65   0.15  11.05  72.12
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 117.2608     0.8811 133.080  < 2e-16 ***
## PC1         181.6572    18.2977   9.928  < 2e-16 ***
## PC2          83.5334    17.9920   4.643 4.79e-06 ***
## PC3         -88.6432    18.1473  -4.885 1.55e-06 ***
## PC4         122.1351    18.2476   6.693 8.16e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17 on 368 degrees of freedom
## Multiple R-squared:  0.3388, Adjusted R-squared:  0.3316
## F-statistic: 47.14 on 4 and 368 DF,  p-value: < 2.2e-16
```

```
anova(model.wgb)
```

```
## Analysis of Variance Table
##
## Response: height
##            Df Sum Sq Mean Sq F value    Pr(>F)
## PC1         1  28881 28881.3  99.971 < 2.2e-16 ***
## PC2         1   5924  5924.2  20.506 8.040e-06 ***
## PC3         1   6723  6723.2  23.272 2.063e-06 ***
## PC4         1  12942 12942.3  44.799 8.163e-11 ***
## Residuals 368 106314   288.9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

You can see that the coefficient of determination of the regression model is higher than that of the polynomial regression model. The results of analysis of variance show that all principal components are significant and need to be included in the regression.

Finally, let's combine the polynomial regression model with the multiple regression model.

```
# multi-linear regression with all information
model.all <- lm(height ~ flower + I(flower^2) + PC1 + PC2 + PC3 + PC4, data =
 data)
summary(model.all)
```

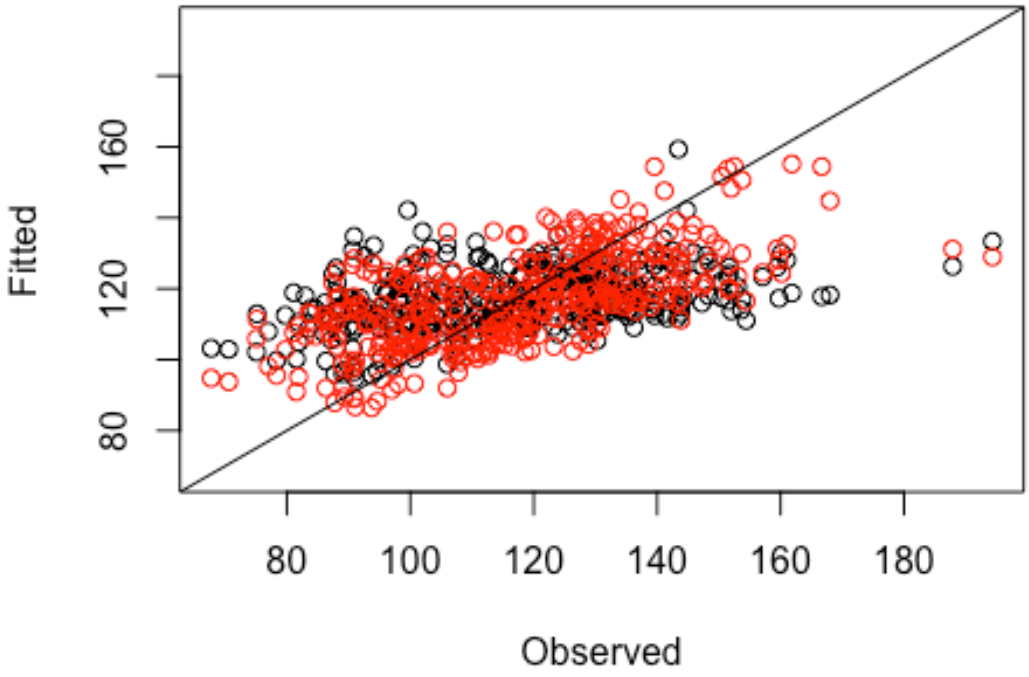
```
##
## Call:
## lm(formula = height ~ flower + I(flower^2) + PC1 + PC2 + PC3 +
##     PC4, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -37.589 -10.431   0.542  10.326  65.390
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  25.739160  24.725955   1.041  0.29857
## flower        1.633185   0.543172   3.007  0.00282 **
```

```
## I(flower^2)  -0.006598   0.002974  -2.219  0.02713 *
## PC1         141.214491  18.547296   7.614 2.29e-13 ***
## PC2          83.552448  17.231568   4.849 1.84e-06 ***
## PC3         -45.310663  18.647979  -2.430  0.01559 *
## PC4         119.638954  17.369423   6.888 2.48e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.17 on 366 degrees of freedom
## Multiple R-squared:  0.4045, Adjusted R-squared:  0.3947
## F-statistic: 41.43 on 6 and 366 DF,  p-value: < 2.2e-16

# compare error variance
anova(model.all, model.wgb)

## Analysis of Variance Table
##
## Model 1: height ~ flower + I(flower^2) + PC1 + PC2 + PC3 + PC4
## Model 2: height ~ PC1 + PC2 + PC3 + PC4
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
## 1    366  95753
## 2    368 106314 -2    -10561 20.184 4.84e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The effect of the genetic background on plant height is very large, but it can also be seen that the model's explanatory power improves if the effect of flowering timing is also added.

Lastly, let's compare the first regression model and the last created multiple regression model by plotting the scatter of the observed value and the fitted value.

```
# compare between the simplest and final models
lim <- range(data$height, fitted(model), fitted(model.all))
plot(data$height, fitted(model), xlab = "Observed", ylab = "Fitted", xlim = l
im, ylim = lim)
points(data$height, fitted(model.all), col = "red")
abline(0,1)
```




As a result, we can see that the explanatory power of the model is significantly improved by considering the genetic background and the second order terms. However, on the other hand, it can also be seen that the two varieties and lines whose flowering timing is late (after 180 days) can not be sufficiently explained even by the finally obtained model. There may be room to improve the model, such as adding new factors as independent variables.

### Quiz 6

Now, let's solve a practice question here. Practice questions will be presented in the lecture.

If you close the page of the quiz, go to https://www.menti.com/ and type in the number I will tell you in the lecture.

## Experimental design and analysis of variance

When trying to draw conclusions based on experimental results, it is always the presence of errors in the observed values. Errors are inevitable no matter how precise the experiment is, especially in field experiments, errors are caused by small environmental variations in the field. Therefore, experimental design is a method devised to obtain objective conclusions without being affected by errors.

First of all, what is most important in planning experiments is the Fisher's three principles:

1. Replication: In order to be able to perform statistical tests on experimental results, we repeat the same process. For example, evaluate one variety multiple times. The experimental unit equivalent to one replication is called a plot.

2. Randomization: An operation that makes the effect of errors random is called randomization. For example, in the field test example, varieties are randomly assigned to plots in the field using dice or random numbers.
3. Local control: Local control means dividing the field into blocks and managing the environmental conditions in each block to be as homogeneous as possible. In the example of the field test, the grouped area of the field is divided into small units called blocks to make the cultivation environment in the block as homogeneous as possible. It is easier to homogenize each block rather than homogenizing the cultivation environment of the whole field.

The experimental method of dividing the field into several blocks and making the cultivation environment as homogeneous as possible in the blocks is called the randomized block design. In the randomized block design, the field is divided into blocks, and varieties are randomly assigned within each block. The number of blocks is equal to the number of replications.

Next, I will explain the method of statistical test in the randomized block design through a simple simulation. First, let's set the "seed" of the random number before starting the simulation. A random seed is a source value for generating pseudorandom numbers.

```
# set a seed for random number generation
set.seed(12)
```

Let's start the simulation. Here, consider a field where 16 plots are arranged in 4 × 4. And think about the situation that there is a slope of the soil fertility in the field.

```
# The blocks have unequal fertility among them
field.cond <- matrix(rep(c(4,2,-2,-4), each = 4), nrow = 4)
field.cond
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    4    2   -2   -4
## [2,]    4    2   -2   -4
## [3,]    4    2   -2   -4
## [4,]    4    2   -2   -4
```

However, it is assumed that there is an effect of +4 where the soil fertility is high and -4 where it is low.

Here, we arrange blocks according to Fisher's three principles. The blocks are arranged to reflect the difference in the soil fertility well.

```
# set block to consider the heterogeneity of field condition
block <- c("I", "II", "III", "IV")
blomat <- matrix(rep(block, each = 4), nrow = 4)
blomat
```

```
##      [,1] [,2] [,3]  [,4]
## [1,] "I"  "II" "III" "IV"
## [2,] "I"  "II" "III" "IV"
## [3,] "I"  "II" "III" "IV"
## [4,] "I"  "II" "III" "IV"
```

Next, randomly arrange varieties in each block according to Fisher's three principles. Let's prepare for that first.

```
# assume that there are four varieties
variety <- c("A", "B", "C", "D")
```

```
# sample the order of the four varieties randomly
sample(variety)

## [1] "B" "D" "C" "A"

sample(variety)

## [1] "C" "B" "A" "D"
```

Let's allocate varieties randomly to each block.

```
# allocate the varieties randomly to each column of the field
varmat <- matrix(c(sample(variety), sample(variety),
            sample(variety), sample(variety)), nrow = 4)
varmat

##      [,1] [,2] [,3] [,4]
## [1,] "D"  "B"  "B"  "D"
## [2,] "B"  "A"  "D"  "A"
## [3,] "C"  "D"  "C"  "B"
## [4,] "A"  "C"  "A"  "C"
```

Consider the differences in genetic values of the four varieties. Let the genetic values of the A to D varieties be +4, +2, -2, -4, respectively.

```
# simulate genetic ability of the varieties
g.value <- matrix(NA, 4, 4)
g.value[varmat == "A"] <- 4
g.value[varmat == "B"] <- 2
g.value[varmat == "C"] <- -2
g.value[varmat == "D"] <- -4
g.value

##      [,1] [,2] [,3] [,4]
## [1,]   -4    2    2   -4
## [2,]    2    4   -4    4
## [3,]   -2   -4   -2    2
## [4,]    4   -2    4   -2
```

Environmental variations are generated as random numbers from a normal distribution with an average of 0 and a standard deviation of 2.5.

```
# simulate error variance (variation due to the heterogeneity of local enviro
nment)
e.value <- matrix(rnorm(16, sd = 2.5), 4, 4)
e.value

##           [,1]      [,2]       [,3]       [,4]
## [1,] -1.547611  2.232424  0.1911757  1.8861892
## [2,] -2.207789  3.909922 -2.1251164 -0.8860432
## [3,]  1.098536  2.524477 -3.2760349 -1.1552031
## [4,]  3.110199 -0.690938 -4.2153578  4.7493152
```

Although the above command generates random numbers, I think you will get the same value as the textbook. This is because the random numbers generated are pseudo random numbers and are generated according to certain rules. Note that if you change the value of the random

seed, the same value as above will not be generated. Also, different random numbers are generated each time you run.

Finally, the overall average, the gradient of soil fertility, the genetic values of varieties, and the variation due to the local environment are added together to generate a simulated observed value of the trait.

```
# simulate phenotypic values
grand.mean <- 50
simyield <- grand.mean + field.cond + g.value + e.value
simyield
```

```
##          [,1]     [,2]     [,3]     [,4]
## [1,] 48.45239 56.23242 50.19118 43.88619
## [2,] 53.79221 59.90992 41.87488 49.11396
## [3,] 53.09854 50.52448 42.72397 46.84480
## [4,] 61.11020 49.30906 47.78464 48.74932
```

Before performing analysis of variance, reshape data in the form of matrices into vectors and rebundle them.

```
# unfold a matrix to a vector
as.vector(simyield)
```

```
##  [1] 48.45239 53.79221 53.09854 61.11020 56.23242 59.90992 50.52448 49.309
06
##  [9] 50.19118 41.87488 42.72397 47.78464 43.88619 49.11396 46.84480 48.749
32
```

```
as.vector(varmat)
```

```
##  [1] "D" "B" "C" "A" "B" "A" "D" "C" "B" "D" "C" "A" "D" "A" "B" "C"
```

```
as.vector(blomat)
```

```
##  [1] "I"   "I"   "I"   "I"   "II"  "II"  "II"  "II"  "III" "III" "III" "II
I"
## [13] "IV"  "IV"  "IV"  "IV"
```

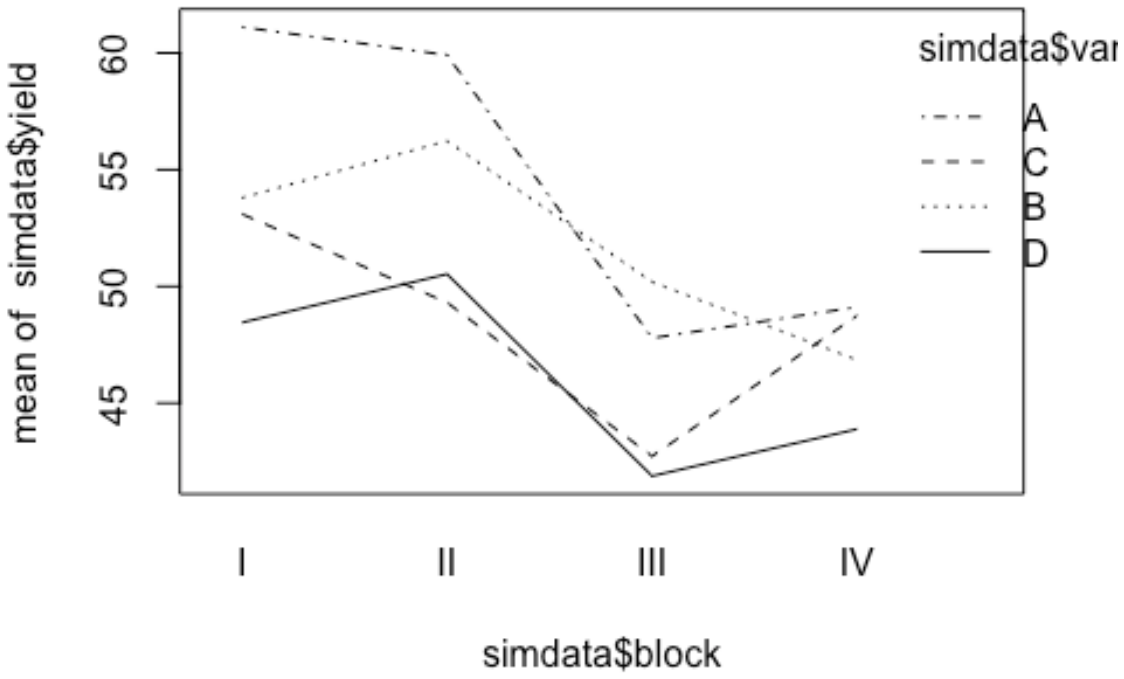
Below, the data is bundled as a data frame.

```
# create a dataframe for the analysis of variance
simdata <- data.frame(variety = as.vector(varmat), block = as.vector(blomat),
 yield = as.vector(simyield))
head(simdata, 10)
```

```
##    variety block    yield
## 1        D     I 48.45239
## 2        B     I 53.79221
## 3        C     I 53.09854
## 4        A     I 61.11020
## 5        B    II 56.23242
## 6        A    II 59.90992
## 7        D    II 50.52448
## 8        C    II 49.30906
## 9        B   III 50.19118
## 10       D   III 41.87488
```

Let's plot the created data using the function interaction.plot.

```
# draw interaction plot
interaction.plot(simdata$block, simdata$variety, simdata$yield)
```




It can be seen that the difference between blocks is as large as the difference between varieties

Let's perform an analysis of variance using the prepared data.

```
# perform the analysis of variance (ANOVA) with simulated data
res <- aov(yield ~ block + variety, data = simdata)
summary(res)

##             Df Sum Sq Mean Sq F value  Pr(>F)
## block        3 239.11   79.70  11.614 0.00190 **
## variety      3 159.52   53.17   7.748 0.00728 **
## Residuals    9  61.77    6.86
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

It can be seen that both the block and variety effects are highly significant. Note that the former is not the subject of verification, and is incorporated into the model in order to estimate the variety effect correctly.

The analysis of variance described above can also be performed using the function "lm" for estimating regression models.

```
# perform ANOVA with a linear model
res <- lm(yield ~ block + variety, data = simdata)
anova(res)

## Analysis of Variance Table
##
## Response: yield
##           Df  Sum Sq Mean Sq F value   Pr(>F)
## block      3 239.109  79.703 11.6138 0.001898 **
## variety    3 159.518  53.173  7.7479 0.007285 **
## Residuals  9  61.765   6.863
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

### Complete random block design and completely randomized design

The local control, one of Fisher's three principles, is very important for performing highly accurate experiments in fields under high heterogeneity between plots. Here, assuming the same environmental conditions as before, let's consider performing an experiment without setting up a block.

In the previous simulation experiment, we blocked each column and placed A, B, C, D randomly in that block. Here we will assign the plots with 4 varieties x 4 replicates completely randomly across the field. An experiment in which blocks are not arranged in the expliment and arranged completely randomly is called "completely randomized design."

```
# completely randomized the plots of each variety in a field
varmat.crd <- matrix(sample(varmat), nrow = 4)
varmat.crd

##      [,1] [,2] [,3] [,4]
## [1,] "A"  "C"  "C"  "D"
## [2,] "B"  "A"  "B"  "D"
## [3,] "A"  "C"  "B"  "A"
## [4,] "D"  "B"  "C"  "D"
```

This time, you should careful that the frequency of appearance of variety varies from row to row, since varieties are randomly assigned to the entire field.

The genetic effect is assigned according to the order of varieties in a completely random arrangement.

```
# simulate genetic ability of the varieties
g.value.crd <- matrix(NA, 4, 4)
g.value.crd[varmat.crd == "A"] <- 4
g.value.crd[varmat.crd == "B"] <- 2
g.value.crd[varmat.crd == "C"] <- -2
g.value.crd[varmat.crd == "D"] <- -4
g.value.crd

##      [,1] [,2] [,3] [,4]
## [1,]    4   -2   -2   -4
## [2,]    2    4    2   -4
## [3,]    4   -2    2    4
## [4,]   -4    2   -2   -4
```

As in the previous simulation experiment, the overall average, the gradient of soil fertility, the genetic effect of varieties, and the variation due to the local environment are summed up.

```
# simulate phenotypic values
simyield.crd <- grand.mean + g.value.crd + field.cond + e.value
simyield.crd
```

```
##          [,1]     [,2]     [,3]     [,4]
## [1,] 56.45239 52.23242 46.19118 43.88619
## [2,] 53.79221 59.90992 47.87488 41.11396
## [3,] 59.09854 52.52448 46.72397 48.84480
## [4,] 53.11020 53.30906 41.78464 46.74932
```

The data is bundled as a data frame.

```
# create a dataframe for the analysis of variance
simdata.crd <- data.frame(variety = as.vector(varmat.crd),
                          yield = as.vector(simyield.crd))
head(simdata.crd, 10)
```

```
##    variety    yield
## 1        A 56.45239
## 2        B 53.79221
## 3        A 59.09854
## 4        D 53.11020
## 5        C 52.23242
## 6        A 59.90992
## 7        C 52.52448
## 8        B 53.30906
## 9        C 46.19118
## 10       B 47.87488
```

Now let's perform analysis of variance on the data generated in the simulation. Unlike the previous experiment, we do not set blocks. Thus, we perform regression analysis with the model that only includes the varietal effect and does not include the block effect.

```
# perform ANOVA
res <- lm(yield ~ variety, data = simdata.crd)
anova(res)
```

```
## Analysis of Variance Table
##
## Response: yield
##           Df Sum Sq Mean Sq F value  Pr(>F)
## variety    3 218.12  72.705  3.1663 0.06392 .
## Residuals 12 275.55  22.962
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

In the above example, the varietal effect is not significant. This is considered to be due to the fact that the spatial heterogeneity in the field causes the error to be large and the genetic difference between varieties cannot be estimated with sufficient accuracy.

The above simulation experiment was repeated 100 times (shown on the next page). As a result, in the experiment using the random complete block design, the varietal effect was detected (the significance level was set to 5%) in 94 experiments out of 100, but it was detected only 66 times in the completely random arrangement. In addition, when the

significance level was set to 1%, the number of the varietal effect detected was 70 and 30, respectively (in the case of completely random arrangement, the varietal effect was missed 70 times!). From this result, it can be seen that the adoption of the random complete block design is effective when there is among-replication heterogeneity such as the slope of soil fertility. In order to make a time-consuming and labor-intensive experiment as efficient as possible, it is important to design the experiment properly.

```
# perform multiple simulations
n.rep <- 100
p.rbd <- rep(NA, n.rep)
p.crd <- rep(NA, n.rep)
for(i in 1:n.rep) {
    # experiment with randomized block design
    varmat <- matrix(c(sample(variety), sample(variety),
            sample(variety), sample(variety)), nrow = 4)
    g.value <- matrix(NA, 4, 4)
    g.value[varmat == "A"] <- 4
    g.value[varmat == "B"] <- 2
    g.value[varmat == "C"] <- -2
    g.value[varmat == "D"] <- -4
    e.value <- matrix(rnorm(16, sd = 2.5), 4, 4)
    simyield <- grand.mean + field.cond + g.value + e.value
    simdata <- data.frame(variety = as.vector(varmat),
            block = as.vector(blomat), yield = as.vector(simyield))
    res <- lm(yield ~ block + variety, data = simdata)
    p.rbd[i] <- anova(res)$Pr[2]

    # experiment with completed randomized design
    varmat.crd <- matrix(sample(varmat), nrow = 4)
    g.value.crd <- matrix(NA, 4, 4)
    g.value.crd[varmat.crd == "A"] <- 4
    g.value.crd[varmat.crd == "B"] <- 2
    g.value.crd[varmat.crd == "C"] <- -2
    g.value.crd[varmat.crd == "D"] <- -4
    simyield.crd <- grand.mean + g.value.crd + field.cond + e.value
    simdata.crd <- data.frame(variety = as.vector(varmat.crd),
                              yield = as.vector(simyield.crd))
    res <- lm(yield ~ variety, data = simdata.crd)
    p.crd[i] <- anova(res)$Pr[1]
}
sum(p.rbd < 0.05) / n.rep

## [1] 0.94

sum(p.crd < 0.05) / n.rep

## [1] 0.54

sum(p.rbd < 0.01) / n.rep

## [1] 0.74

sum(p.crd < 0.01) / n.rep

## [1] 0.21
```

## Report assignment

1. Using the number of seeds per panicle (Seed.number.per.panicle) as the dependent variable $y$ and panicle length (Panicle.length) as the independent variable $x$, fit a simple regression model ($y_i = \mu + \beta x_i + \epsilon_i$) and find the sample intercept $m$ and sample regression coefficient $b$, which are estimates of $\mu$ and $\beta$.
2. Test the null hypothesis $H_0: b = 0.02$.
3. Test the null hypothesis $H_0: m = 5$.
4. Answer the 95% confidence interval between the estimated y, i.e., $\hat{y}$ and the predicted y, i.e., $\tilde{y}$ for $x = 27$.
5. Fit the polynomial regression model $y_i = \mu + \beta_1 x_i + \beta_2 x_i^2 + \epsilon_i$) using the first- and second-order terms of $x$ and answer the determination coefficient $R^2$ and the adjusted coefficient of determination $R_{adj}^2$.
6. Compare the regression model in 5 with the regression model in 1 in an analysis of variance and consider the validity of including a second-order term of $x$ in the regression model.
7. Fit the polynomial regression model $y_i = \mu + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \epsilon_i$) using the first- to third-order terms of $x$ and answer the decision coefficient $R^2$ and the he adjusted coefficient of determination $R_{adj}^2$.
8. Compare the regression model in 7 with the regression model in 5 in an analysis of variance to examine the validity of extending the second-order polynomial regression model to a third-order polynomial regression model.

Submission method:

- Create a report as a pdf file and submit it to ITC-LMS.
- When you cannot submit your report to ITC-LMS with some issues, send the report to report@iu.a.u-tokyo.ac.jp
- Make sure to write the affiliation, student number, and name at the beginning of the report.
- The deadline for submission is May 8th.