## Parametric Methods: Sections 15-28

How to Run a One-Sample t-Test

How to Run a Two-Sample t-Test with Independent Samples

How to Run a Two-Sample t-Test with Paired Data

How to Perform a One-Tailed Hypothesis Test

How to Test a Claim about a Single Population Variance

How to Perform an F-Test to Compare Two Variances

How to Do One-Way ANOVA

How to Run a Repeated Measures ANOVA

How to Run a Two-Way ANOVA with or without Interactions

How to Run Mauchly's Test for Sphericity

How to Check Pairs of Data Values for Linear Correlation

How to Run a Simple Linear Regression

How to Obtain Residuals and Fits from a Regression Line and Check the Assumptions

# How to Run a Basic Multiple Regression

## Section 15: How to Run a One-Sample T-Test

(Uses data file: MO Life Expectancy.txt)

This example will test the hypothesis that the mean overall U.S. life expectancy is 76 years. You will use the Missouri life expectancy by county as your sample. This may not be random enough to provide a representative sample for the whole country, but proceed as if it is for this example.

First read in the data table, and check it for normality – see Section 10: "How to Check for Normality Using a Normal Probability Plot." A portion of the data set is displayed there. There you will find the graphs for this data set. You can see there that this data set looks fairly normal, although there is some doubt about the lower end. But for purposes of this example, assume normality holds.

Now your hypotheses are: $H_0$: Mean US life expectancy is 76 years
$H_1$: Mean US life expectancy is not 76 years
Pre-select your level of significance; this example uses alpha (α) =.05

Run a t-test. In the command, you have to specify the variable name and the hypothesized mean. In this example, the column containing the overall MO life expectancies by county is called "Total" in the data table. The hypothesized mean is 76. The test is intended as a two-tail test.

```
> t.test (Total, mu = 76)
```

The resulting output is as shown at left. Explanations are added on the right.

| | |
|---|---|
| One Sample t-test | |
| data: Total | |
| t = 3.8405, df = 114, p-value = 0.0002022 | ←-This line tells you that the test statistic is 3.8405. |
| | df = sample size -1 = 115 -1 = 114 in this example. |
| | The p-value is the probability of getting a sample at least as extreme as yours if the null hypothesis is true. Note that this p-value is much smaller than your α level. |
| alternative hypothesis: true mean is not equal to 76 | ← Indicates a two-tail test; a one-tail would specify "greater than 76" or "less than 76" |
| 95 percent confidence interval: | |
| 76.29051 76.90949 | ← Tells you that you can be 95% sure that the the true mean lies between about 76.3 and 76.9. |
| sample estimates: | |
| mean of x 76.6 | ← Calculated sample mean |

The R output does not produce the t-critical value(s), but you can do that now. Specify the quantiles desired and the degrees of freedom to be used. In this example, since it is intended to be a two-tail test with $\alpha$=0.05, split $\alpha$ in half and use half for each of the lower and upper critical values. The lower critical value will be the $2.5^{th}$ percentile; the upper critical value will be the $97.5^{th}$ percentile. The degrees of freedom (df) value appears in the output above, and is 114 in this case.

To get the lower critical value, use the following.

```
> LCV = qt (.025, 114)
> LCV
```

The result that is returned is:

```
-1.980992
```

The next two lines will get you the upper critical value.

```
> UCV = qt (.975, 114)
> UCV
```

The output is:

```
1.980992
```

Since the test statistic 3.8405 falls outside the interval between the two critical values, you will reject the null hypothesis. Also, since the p-value = 0.0002022 is less than $\alpha$, you will reject the null hypothesis.

NOTE: Comparing the test statistic against the t-critical value(s), and comparing the p-value to $\alpha$, should both yield the same conclusion.

INTERPRETATION: The sample provides significant evidence that the US overall life expectancy is not 76 years. It does not, by itself, indicate whether the national mean is higher or lower than that.

## Section 16: How to Run Two-Sample T-Tests (Independent Samples)

(Uses data file: Life Exp by Gender – nonpaired.txt)

This is an example of a t-test to compare means of two independent (unpaired) samples. The context is as follows. Male life expectancy in MO was recorded for 20 randomly selected counties. Female life expectancy in MO was recorded for 20 randomly selected counties, not necessarily the same ones. That is, the male and female data **is not** in pairs matched by county

The hypotheses to be tested are that mean life expectancy is the same for both genders vs. the alternative that it is not. Set up your null and alternative hypotheses and your level of significance.

Now your hypotheses are: $H_0$: Mean MO life expectancies by county for both genders are equal.
$H_1$: Mean MO life expectancy by county is not the same for both genders.
Pre-select your level of significance; this example uses alpha (α) =.05

```
> LifebyGender = read.table ("E:/Data Files/Life Exp by Gender - nonpaired.txt", header=TRUE)
> attach (LifebyGender)
> LifebyGender
```

A portion of the data set is shown below.

```
   Gender  Years

1      M    75.4
2      M    75.1
3      M    75.5
4      M    73.3
:
20     M    72.6
21     F    79.6
22     F    81.4
23     F    78.5
24     F    80.7
:
40     F    79.6
```

In general, the command to run a t-test to compare means of two groups is: "t.test (variable ~ group )." The ~ symbol means that the variable is being considered as a function of the group. So here, since the variable is Years and the group is Gender, you use the following statement.

```
> t.test (Years~Gender)
```

The resulting output is on the left; interpretation of the output is added on the right. Note that name of the two-sample t-test for independent samples is the "Welch Two Sample t-test."

Welch Two Sample t-test

data: Years by Gender
t = 9.9765, df = 35.6, p-value = 7.479e-12 ← Test statistic is 9.9765

df is calculated by a more complicated formula than in most textbooks.

p-value is about $7.5 \times 10^{-12} < \alpha$.

alternative hypothesis: true difference in means is not equal to 0 ← Indicates a two-tailed test. A one-tail test would specify "greater than" or "less than."

95 percent confidence interval:
4.134533 6.245467 ← Tells you that you can be 95% confident that the interval between about 4.1 and 6.2 contains the true mean difference in life expectancy by gender.

sample estimates:
mean in group F mean in group M
79.295 74.105 ← Means for the two samples.

Note that the df value that would be calculated by most textbooks is simpler, using the formula
(Male sample size -1) + (Female sample size – 1) = (20 - 1) + (20 - 1) = 38.
The value produced by R involves a more complicated calculation but the two df values should be close.

The output does not provide the t-critical values, but you can make R find them. Put half of α in each tail, since this is a two-tailed test. Use the df value from the output above, that is, use 35.6.

Therefore, you use the following two lines to get the lower critical value.

```
> LCV = qt (.025, 35.6)
> LCV
```

The resulting lower critical value is:

```
-2.028886
```

Similarly, you use the next two lines to get the upper critical value.

```
> UCV = qt (.975, 35.6)
> UCV
```

Then R returns the following upper critical value.

```
2.028886
```

Since the test statistic 9.9765 is outside the interval bounded by these two critical values, you will reject the null hypothesis. Since the p-value is smaller than $\alpha$, this also indicates that you will reject the null hypothesis.

INTERPRETATION: The sample provides sufficient evidence to conclude that mean life expectancy by county in Missouri is not the same for both genders. Note that this was a two-tailed test, so it does not indicate the direction of the difference. But you can compare the two sample means and observe the sample direction – on average, women live longer than men.

## Section 17: How to Run Two-Sample T-Tests with Paired Data

(Uses data file: Life Exp by Gender – paired.txt)

This is an example of a t-test to compare the means of paired data. The comparison is done by running a one-sample t-test on the difference between them. The null hypothesis is that the mean difference is zero, i.e., the means are equal.

The context of the example is a comparison of male and female life expectancy in Missouri in the period from 2004 through 2012. A random sample of twenty-four counties is taken, and the male and female life expectancies for these counties are recorded in pairs. The difference is calculated: Female expectancy minus male expectancy.

Note that you could also calculate the difference the other way: Male expectancy minus female expectancy. But once you decide on a direction, you have to be consistent.

The null hypothesis to be tested is that the difference is zero, vs. the alternative that the difference is non-zero.

Now your hypotheses are: $H_0$: Mean MO life expectancies by county for both genders are equal.
(Equivalently, the mean difference is zero.)
$H_1$: Mean MO life expectancy by county is not the same for both genders.
(Equivalently, the mean difference is not zero.)

Pre-select your level of significance; this example uses alpha (α) =.05

```
>  LifeExp = read.table ("E:/Data Files/Life Exp by Gender - paired.txt" , header = TRUE)
> attach (LifeExp)
> LifeExp
```

A portion of data set appears as follows.

| | County | Male | Female | Difference |
|---|---|---|---|---|
| 1 | Adair | 75.4 | 79.9 | 4.5 |
| 2 | Andrew | 75.1 | 80.3 | 5.2 |
| 3 | Cole | 76.1 | 80.5 | 4.4 |
| 4 | Cooper | 75.3 | 79.4 | 4.1 |
| 5 | Franklin | 74.1 | 79.3 | 5.2 |
| | : | | | |
| 23 | Worth | 75.3 | 81.8 | 6.4 |
| 24 | Wright | 72.7 | 78.2 | 5.5 |

Now you can run the t-test with the hypothesized mean of Difference = 0.

```
> t.test (Difference, mu=0)
```

The resulting output is on the left; interpretive comments are added at right.

One Sample t-test

data: Difference
t = 24.9683, df = 23, p-value < 2.2e-16 ← Test statistic is 24.9683.

df=number of pairs – 1 = 24 – 1 = 23.

The p-value = $2.2 \times 10^{-16}$, which is smaller than α.

alternative hypothesis: true mean is not equal to 0 ← Indicates that this is a two-tail test. A one-tail test would specify "greater than" or "less than."

95 percent confidence interval: ← So you that you can be 95% confident that the
5.071067 5.987266 interval between about 5.07 and 5.99 contains the true population mean difference.

sample estimates:
mean of x 5.529167 ← Calculated sample mean difference.

The output does not include the t-critical values. You can obtain these in the usual way.

The following two lines will get you the lower critical value.

```
> LCV = qt (.025, 23)
> LCV
```

R returns the following:

```
-2.068658
```

Similarly, the next two lines will get you the upper critical value.

```
> UCV = qt (.975, 23)
> UCV
```

R returns the following:

```
2.068658
```

The test statistic t = 24.9683 is outside the interval bounded by the critical values, so you should reject the null hypothesis. Also, the p-value is less than α, so the null hypothesis is rejected on that basis.

INTERPRETATION: There is sufficient evidence to indicate that average (by county) male and female life expectancies are different in Missouri.

COMMENT: As presented here, the difference is already in the data set as its own column. If it was not, before running the test, you would have to make R calculate it by typing:

```
> Difference = Female – Male.
```

## Section 18: How to Perform a One-Tailed Hypothesis Test

(Uses data files: Life Exp by Gender – paired.txt, Life Exp by Gender – nonpaired2.txt)

The default alternative hypothesis in R is always "not equal to," that is, R defaults to a two-tailed hypothesis test. You can override this by specifying a direction. This goes into the "test" command after stating the value to be used for the null hypothesis. Also, if you are using a one-tailed test to compare two populations, you should arrange your data in two columns instead of one.

Here are two examples that you have already seen as two-tailed tests, but they are now presented as one-tailed tests. The advantage to the latter is that the direction of the difference in means is specified as part of the test.

EXAMPLE A (Paired Data)

The context of the first example is a comparison of male and female life expectancy in Missouri (2004-2012). A random sample of twenty counties is taken, and the male and female life expectancies for these counties are recorded. Then the difference is calculated: female expectancy minus male expectancy. The hypotheses to be tested are that the difference is zero, vs. the alternative that the difference is greater than zero. You can see the data set partially displayed in Section 17.

Your hypotheses are: $H_0$: Mean MO life expectancies by county for both genders are equal.
(Equivalently, the mean difference is zero.)
$H_1$: Mean MO life expectancy by county is greater for women than men.
(Equivalently, the mean difference is greater than zero.)
Pre-select your level of significance; this example uses alpha (α) =.05

```
> t.test (Difference, mu=0, alternative = "greater")
```

The output is as shown below on the left; interpretative comments are added on the right.

```
One Sample t-test

data:  Difference
t = 24.9683, df = 23, p-value < 2.2e-16        ← Test statistic is 24.9683.

                                                  Degrees of freedom=23.

                                                  The p-value is some value < 2.2 x 10^-16

alternative hypothesis: true mean is greater than 0   ← Indicates upper tail test.

95 percent confidence interval:                ← One-sided confidence interval: you can be 95%
 5.149634      Inf                                sure that the true mean difference is > 5.149634

sample estimates:                              ← Calculated sample mean difference.
mean of x          5.529167
```

The output, as usual, does not include the t-critical value. You can obtain this in the usual way, but remember that all of α = .05 belongs in the upper tail.

```
> UCV=qt (.95, 23)
> UCV
```

The critical value for the upper tail test is then returned by R.

```
1.713872
```

The test statistic of 24.9683 exceeds the upper critical value, and the p-value is less than α. Both of these indicate that you should reject the null hypothesis.

INTERPRETATION: There is sufficient evidence to indicate that the mean female life expectancy is greater than the mean male life expectancy in Missouri.

COMMENT: For a lower-tailed test, you would have specified: alternative = ”less.”

EXAMPLE B (Unpaired, i.e., Independent Data)

This is an example of a t-test to compare means of two independent (unpaired) samples. The context is as follows. Male life expectancy in MO was recorded for 20 randomly selected counties. Female life expectancy in MO was recorded for 20 randomly selected counties, not necessarily the same ones. That is, the male/female data is not in pairs matched by county.

You can read in and display the data set in the usual way.

```
 >  LifeExp = read.table ("E:/Data Files/Life Exp by Gender – nonpaired2.txt" , header = TRUE)
> attach (LifeExp)
> LifeExp
```

A portion of the data set is shown below.

| MYears | FYears |
|---|---|
| 75.4 | 79.6 |
| 75.1 | 81.4 |
| 75.5 | 78.5 |
| 73.3 | 80.7 |
| 73.1 | 78.9 |
| 75.5 | 80.5 |
| : | |
| 75.9 | 79.3 |
| 73.6 | 79.0 |
| 69.5 | 81.3 |
| 75.9 | 80.3 |
| 72.6 | 79.6 |

Your hypotheses are: $H_0$: Mean MO life expectancies by county for both genders are equal.
$H_1$: Mean MO life expectancy by county is less for men than women.
Pre-select your level of significance; this example uses alpha (α) =.05

```
> t.test (MYears, FYears, alternative="less")
```

The resulting output, with comments added at right, is as follows.

```
Welch Two Sample t-test

data:  MYears and FYears

t = -9.9765, df = 35.6, p-value = 3.739e-12        ← Test statistic is -9.9765.

                                                       Degrees of freedom = 35.6

                                                       The p-value=3.739 x 10^-12

alternative hypothesis: true difference in means is less than 0    ← Indicates lower-tailed test.

95 percent confidence interval:                    ← Tells you that you can be 95% confident that the
-Inf      -4.311453                                   mean male life expectancy is at least 4.311453 years
                                                      less than the mean female life expectancy.
sample estimates:
mean of x      mean of y
  74.105         79.295                            ← Calculated sample means.
```

The output, as usual, does not include the t-critical value. You can obtain this in the usual way, but remember that all of α = .05 belongs in the lower tail since this was a lower-tailed test.

```
> LCV = qt (.05, 35.6)
> LCV
```

R returns a critical value for a lower tail test.

```
-1.688799
```

The test statistic exceeds (negatively) the lower critical value, and the p-value is less than α. Both of these indicate that you should reject the null hypothesis.

INTERPRETATION: There is sufficient evidence to indicate that the mean male life expectancy is less than the mean female life expectancy in Missouri.

## Section 19: How to Test a Claim about a Single Population Variance

(Uses data file: Life Exp by Gender – nonpaired.txt)

Sometimes you will want to test a claim about a single population variance. For example, the National Bureau of Economic Research estimates the standard deviation in adult life spans in the United States as being around 15 years. Equivalently, the variance is approximately $15^2 = 225$.

Lifespans in the state of Missouri might be more variable, less variable, or about the same as lifespans nationally. You can use the data set "Life Exp by Gender – unpaired" to test this. Assume you want a two-sided test, with level of significance alpha (α) = .05. Since 1-.05 = .95, this corresponds to a 95% confidence level for the confidence interval.

As of this writing, the single population variance test is not included in base R, so you will first have to install and load the package that contains the appropriate test. The package you want is: EnvStats. See Section 3: "How to Find, Install and Load R Packages."

Once that is done, you proceed as usual to read in and attach the data set. In this test, you will be using the entire column Years without differentiating by gender. You are simply going to compare the variance in the overall Missouri life expectancy to the national value of 225.

```
> Data = read.table ("E:/Data Files/Life Exp by Gender - nonpaired.txt", header = TRUE)
> attach (Data)
> Data
```

A portion of the data is shown below.

```
   Gender  Years
1      M   75.4
2      M   75.1
3      M   75.5
4      M   73.3
:
:
38     F   81.3
39     F   80.3
40     F   79.6
```

Assuming you have installed and loaded the EnvStats package, you are ready to run the variance test. In the general case, this looks like:

```
> varTest (Variable Name, form of alternative, confidence level, hypothesized variance from the null)
```

Notice that the command requires the confidence level, not the level of significance. To be clear, you should plan your test.

Your hypotheses are: $H_0$: Variance of MO life expectancies by county equals 225.
$H_1$: Variance of MO life expectancy by county does not equal 225.
Pre-select your level of significance; this example uses alpha (α) =.05, so confidence level is .95.

Specifically then, for this example with a two-tailed test, the command you want would therefore be:

```
> varTest (Years, alternative = "two.sided", conf.level = .95, sigma.squared = 225)
```

The output is shown below. Actual output appears on the left; explanatory comments are on the right.

```
Results of Hypothesis Test

Null Hypothesis:          Variance = 225                    ← Hypothesized population variance.
Alternative Hypothesis:   True variance is not equal to 225 ← Indicates two-sided test.

Test Name:                Chi-Squared Test on Variance
Estimated Parameter(s):   variance = 9.54359                ← This is the sample variance.
Data:                     Years                             ← Tells you the variable name.
Test Statistic:           Chi-Squared = 1.654222
Test Statistic Parameter: df = 39          ← Chi-squared distribution uses degrees of freedom n-1.
P-value:                  4.161694e-20     ← Given in scientific notation, this means 4.2 x 10^-20.
```

Note that the p-value is much less than $\alpha$, so the null hypothesis is rejected.

You also get the following two lines at the end of the output.

```
95% Confidence Interval:  LCL =  6.403985
                          UCL = 15.734966
```

These last two lines of output represent the endpoints of a 95% confidence interval for the standard deviation of the population lifespan, based on the Missouri data. Note that it <u>does</u> include the hypothesized standard deviation of sigma=15 that corresponds to the hypothesized value of the variance, although just barely. This would seem to contradict the conclusion based on the p-value. It is probably an indication that the sample data is not a representative random sample.

This are several issues with the data set that may be contributing to this discrepancy. For one thing, the data was obtained by county and not weighted at all according to the size of the county. Further, this particular data set is borderline in terms of normality, and the variance test has normality as an assumption. And finally, the actual test is for the variance but the confidence interval is for the standard deviation.

## Section 20: How to Perform an F-Test to Compare Two Variances

(Uses data file: Life Exp by Gender – nonpaired.txt)

Sometimes you will want to test whether or not two samples come from populations with a common variance. Usually this is because you ultimately want to use a two independent sample t-test to compare the means, and one of the assumptions for that test is that the populations have a common variance.

The variance test creates a ratio of one variance over the other. If the ratio is "close to" one, the variances are considered to be equal; otherwise they are not. The distribution is an F distribution with degrees of freedom given by:

df(numerator) = first sample size – 1

df(denominator) = second sample size – 1.

The example here compares the variances of male and female life expectancies by county in Missouri. An earlier example used a t-test for two independent samples to compare their means.

Your hypotheses are: $H_0$: Variances of MO life expectancies by county are the same for both genders.
$H_1$: Variances are not the same for both genders.
Pre-select your level of significance; this example uses alpha (α) =.05.

As usual, read in and attach the data set. You can display it if you wish.

```
> LifebyGender = read.table ("E:/Data Files/Life Exp by Gender - nonpaired.txt", header = TRUE)
> attach (LifebyGender)
> LifebyGender
```

A portion of the data set is shown below.

```
   Gender  Years
1      M   75.4
2      M   75.1
3      M   75.5
4      M   73.3
:
:
38     F   81.3
39     F   80.3
40     F   79.6
```

Now you are ready to run the test to compare the two variances.

```
> var.test (Years ~ Gender)
```

The resulting output is on the left; explanatory comments have been added on the right.

F test to compare two variances

data: Years by Gender

F = 0.5878, num df = 19, denom df = 19, p-value = 0.2557 ← Test statistic is 0.5878.

Both samples have n=20 so both df = 19.

alternative hypothesis: true ratio of variances is not equal to 1 ← Alternative says variances unequal.

95 percent confidence interval:
0.2326461 1.4849694 ← This is a confidence interval for the ratio of the variances

sample estimates:
ratio of variances 0.5877689 ← Calculated ratio of sample variances.

The p-value of 0.2557 is greater than α. This tells you to fail to reject the null hypothesis. You do not have evidence to believe that the variances are unequal.

You can also have R find the .025 and .975 quantiles of the F(19,19) distribution, as follows.

```
> qf (.025, 19, 19)
```

R returns the following.

```
0.3958122
```

Now for the other quantile.

```
> qf (.975, 19 ,19)
```

This time R returns the following.

```
2.526451
```

These are lower and upper critical values for a confidence interval for the ratio of the population variances. The test statistic falls between these two quantiles, which confirms your "fail to reject" conclusion.

# Section 21: How to Do One-Way ANOVA
## Compare Means of Multiple Populations
(Uses data file: Anxiety.txt)

You use one-way ANOVA when you want to compare the means of more than two populations, where the populations are distinguished by a single characteristic. It is similar to a t-test but uses sample data from three or more populations.

This example uses a fictional data set of "anxiety scores" of 156 college students in four types of institutions (coded as LPR = large private university, SPR = small private university, STA = state university and COM = community college). The goal is to determine whether or not the mean score is the same or different by type of institution. The null hypothesis is that all four types have the same mean anxiety score; the alternative is that at least one mean is different. Use level of significance alpha (α) =.05.

First read in the data table giving the scores and name it "Anxiety." Be sure to "attach" it and display it if you wish.

```
> Anxiety = read.table ("E:/Data Files/Anxiety.txt", header = TRUE)
> attach (Anxiety)
> Anxiety
```

A partial display of the output is as follows.

```
     Type  AnxScore

1     LPR        25
2     SPR        11
3     STA         8
4     COM        17
       :
153   SPR        33
154   SPR        30
155   COM        21
156   COM        19
```

Before trying to run the test, you should plan it.

Your hypotheses are: $H_0$: Mean anxiety scores are the same for all four types of institutions.
$H_1$: At least one type has a different mean anxiety score.
Pre-select your level of significance; this example uses alpha (α) =.05.

You then have to create a linear model of the score as a function of the institution type. This is required because ANOVA can only be performed on a model, not on the data table itself. In the following command, "lm" stands for "linear model."

```
> AnxModel = lm (AnxScore ~ Type)
```

Then you can run the "anova" command on the model.

```
> anova (AnxModel)
```

The output is below. For space reasons, the explanatory comments are in a paragraph below the output (instead of on the right as usual).

```
Analysis of Variance Table

Response: AnxScore

            Df     Sum Sq    Mean Sq  F value  Pr(>F)
Type         3      261.2     87.076   0.8578  0.4645
Residuals  152    15429.9    101.513
```

Here is what the output tells you. The F-value is the test statistic. The p-value is the value in the column on the far right, labelled "Pr(>F)." The other values in the output are used in calculating F. Since the p-value is larger than $\alpha$, you fail to reject the null hypothesis.

INTERPRETATION: The evidence is not sufficient to conclude that any institution type has a different mean anxiety score.

Notice, however, that the output does not give you the individual institution type sample means. If you want those, you have a couple of choices.

(1) Create another data file in which the scores for different types are listed in separate columns, and then find the mean for each column.

OR

(2) Apply a logical function to the original data set to create four vectors, each containing the data for a specific type of institution.

Here is how to do option (2). The double-equal sign (==) is a logical operator that tells R to compare each type listed in the data set to the item on the right, to see whether they match. The results are stored in the variables LPRScores, SPRScores, STAScores and COMScores, respectively.

```
> LPRScores = AnxScore*(Type =="LPR")
> SPRScores = AnxScore*(Type =="SPR")
> STAScores = AnxScore*(Type=="STA")
> COMScores = AnxScore*(Type=="COM")
```

Now total the values in LPRScores.

```
> sum (LPRScores)
```

The sum of the large private institution scores returned is:

```
764
```

Do the others similarly:

```
> sum (SPRScores)          ← R returns 921.
> sum (STAScores)          ← R returns 825.
> sum (COMScores)          ← R returns 801.
```

Now get the counts by institution type. The "summary" command will get the counts.

```
> summary (Type)
```

The resulting output is:

```
COM  LPR  SPR  STA
 39    39   40   38
```

You can now use the score sums and counts per institution type to obtain the four sample means. For instance, you know that the sum of the scores for all the LPR students is 764, and you know there are 39 students from LPR's. So the LPR sample mean is 764/39 = 19.589 (to three decimal places).

The following commands do this for all four institution types.

```
> meanLPR = sum (LPRScores)/39;  meanLPR          ← R returns 19.58974
> meanSPR = sum (SPRScores)/40;  meanSPR          ← R returns 23.025
> meanSTA = sum (STAScores)/38;  meanSTA          ← R returns 21.71053
> meanCOM = sum( COMScores)/39;  meanCOM          ← R returns 20.53846
```

Now for a bit of assumption checking. ANOVA is based on an underlying assumption that the variance is the same across all categories, in this case, for all four institution types. There are several standard tests for homogeneity of variance, one of them due to Bartlett. The null hypothesis is that all of the variances are equal. The alternative is that at least one variance is different from the others. Here alpha (α) =.05 is used as the level of significance.

```
> bartlett.test (AnxScore ~ Type)
```

The resulting output is as follows.

```
Bartlett test of homogeneity of variances

data:  AnxScore by Type
Bartlett's K-squared = 0.55801, df = 3, p-value = 0.906
```

Since the p-value is greater than α, you fail to reject the null hypothesis about the variances. Therefore, it appears that the variances are the same across the four institution types, and consequently the use of ANOVA as the main test is acceptable. If you had been forced to reject the common variance assumption, you would need to find an alternative approach to your main test comparing the means. One option would be the Kruskal-Wallis test that is presented in Section 33 of this Manual.

# Section 22: How to Run a Repeated Measures ANOVA
## Compare Multiple Population Means in the Case of Repeated Measures
(Uses data file: AnxietyRepeat.txt)

The example uses the data set AnxietyRepeat.txt. This data set contains scores on an Anxiety Test, repeated three times on thirty-six (fictional) students. The test is first given during their first term in college (labelled Fall1), repeated second term (labelled Spr1) and then again in their third term (labelled Fall2). The same students are tested all three times. The goal of the testing is to see whether or not, on average, students' anxiety levels are consistent or change over time as they adjust to college.

First read in data table giving the anxiety scores and attach it. You can then display the data set if you wish.

```
> Data = read.table ("E:/Data Files/AnxietyRepeat.txt", header = TRUE)
> attach (Data)
> Data
```

A partial display of the output is as follows.

```
     ID  Test.Session  Anx.Score
1    S1      Fall1         25
2    S1      Spr1          22
3    S1      Fall2         33
4    S2      Fall1         11
5    S2      Spr1           5
6    S2      Fall2         20
:
:
103  S35     Fall1         22
104  S35     Spr1          27
105  S35     Fall2         17
106  S36     Fall1         24
107  S36     Spr1          30
108  S36     Fall2         37
```

As a first look, you may want to obtain the sample means for each test session. You therefore need to "split out" each session from the whole data set. To see how to do this, refer to Section 6: "How to Extract Particular Data Items or Sequences of Them." If you follow example D in that section, you will obtain the summary information for each test session.

| Test Session | Sample Mean Anxiety Score | Sample Variance in Anxiety Score |
|---|---|---|
| Fall1 | 19.17 | 90.77 |
| Spr1 | 23.78 | 142.92 |
| Fall2 | 21.36 | 90.24 |

The summary shows that the sample means are different but you don't know whether or not the population means are different. If you choose to run a two-way ANOVA to test whether or not the population means are the same or different, you have two options. You can run it as:

(1) Anx.Score as a function of person (ID) and Test.Session, OR
(2) Anx.Score as a function of person(ID) and Test.Session, plus an interaction term (like a practice effect).

First set up your hypotheses; assume your chosen level of significance is alpha (α) = .05.

$H_0$: The means of the anxiety scores are the same for all test sessions.
$H_1$: The mean anxiety score for at least one test session is different.

As shown below, this example has no interaction term, because there is no reason to think that anxiety levels are affected by repetition of the testing process. The first command creates a linear model with Anx.Score as a function of Test.Session and ID (individual person). The second command runs an analysis of variance on the model.

```
> Repeat.Model = lm (Anx.Score ~ Test.Session + ID)
> anova (Repeat.Model)
```

The resulting output is below. For space reasons, the explanatory comments are in the paragraph below the box, rather than inside it as usual.

```
Analysis of Variance Table
Response: Anx.Score

             Df  Sum Sq  Mean Sq  F value   Pr(>F)
Test.Session  2   383.0  191.509   2.1971  0.11874
ID           35  5235.9  149.597   1.7162  0.02781 *
Residuals    70  6101.6   87.166
```

INTERPRETATION:

The F-values are in the next-to-last column. They are actually the ratios:

(Mean Squares of Test.Session)/(Mean Squares of Residuals) = 191.509/87.166 = 2.1971
and
(Mean Squares of ID)/(Mean Squares of Residuals) = 149.597/87.166 = 1.7162

These F values are the test statistics, and the entries in the last column are their corresponding p-values. Here, the p-value for the test session is 0.119 (rounded), which is larger than α. So there is no reason to believe that mean anxiety scores change over time. (What this might mean: adjustment to college is not a major source of anxiety. Other factors, such as course load in a particular semester or personal issues, may produce anxiety more randomly over time.)

You might notice, however, that the p-value for ID (i.e., individual student) is about 0.028. That is less than α, and therefore is small enough to be statistically significant. While you did not have a hypothesis

about it, it tells you that the data suggest that mean anxiety level varies by student. This should not be a surprise, in this case, since some people have more anxiety than others.

Note: The model above has no interaction term. If the test had been a test of some kind of skill, rather than anxiety, there might very well have been a practice effect. This could have been tested by including an interaction term in the model. The command would then be as follows, with the interaction term showing up as a "product" expression after the last "+" in the command, highlighted below.

```
> Repeat.Model = lm (Anx.Score ~ Test.Session + ID + Test.Session*ID)
> anova (Repeat.Model)
```

This kind of model, with an interaction term, is discussed in more detail in Section 23.

# Section 23: How to Run a Two-Way ANOVA
## Compare Multiple Population Means with Two Factors and Potential Interaction
(Uses data file: Anxiety By Major and Term.txt)

This data contains scores on an Anxiety Test. The test is given to a randomly selected group of twelve statistics majors, twelve engineering majors and twelve physics majors during fall of their first year in college (Fall1). The test is then given to another randomly selected group of twelve students from each major during their second term (Spr1). Finally, this is done again with a third randomly selected group of twelve students from each major in their third term (labelled Fall2). The goal of the testing is to see whether or not, on average, students' anxiety levels are consistent or change over time as they adjust to college. Also, because some semesters have a heavier course load than other semesters in the same major, it is believed that there may be an interaction effect between Major and Test.Session. Note that this is NOT a repeated measurement process because different students are involved at each session.

First read in data table giving the anxiety scores, attach it and display it if you wish.

```
> Data = read.table ("E:/Data Files/Anxiety By Major and Term.txt", header = TRUE)
> attach (Data)
> Data
```

A partial display of the data is as follows.

```
      ID      Major        Test.Session   Anx.Score
1     S1      Statistics       Fall1          25
2     S2      Statistics       Fall1          11
3     S3      Statistics       Fall1           8
:
:
13    S13     Engineering      Fall1          12
14    S14     Engineering      Fall1          45
15    S15     Engineering      Fall1           6
:
:
25    S25     Physics          Fall1          27
26    S26     Physics          Fall1          27
27    S27     Physics          Fall1          18
:
:
                (and then similar for all of Spr1 followed by all of Fall2)
```

First set up your hypotheses; there are actually three sets of hypotheses here.

Set 1: $H_0$: The mean anxiety score is the same for all majors.
$H_1$: The mean anxiety score for at least one major is different.

Set 2: $H_0$: The mean anxiety score is the same for all test sessions.
$H_1$: The mean anxiety score for at least one test session is different.

Set 3: $H_0$: There is no interaction between a student's major and the test session.
$H_1$: There is an interaction, i.e., students in at least one major exhibit higher or lower anxiety at a particular test session.

Assume your level of significance is alpha (α) = .05.

You want to create a linear model that has the anxiety scores as a linear function of major, test session and an interaction term. You need two lines of code. The first line creates the linear model. Notice that the interaction term appears as a "product" expression after the last "+" in the line creating the model; this term is highlighted below. The second line runs an analysis of variance on the model.

```
> Model = lm (Anx.Score ~ Major + Test.Session + Major*Test.Session)
> anova (Model)
```

The output of the anova is below.

Analysis of Variance Table

Response: Anx.Score

| | Df | Sum Sq | Mean Sq | F value | Pr(>F) | |
|---|---|---|---|---|---|---|
| Major | 2 | 129.5 | 64.731 | 0.5762 | 0.5639 | ← p-value for Hypotheses Set 1. |
| Test.Session | 2 | 383.0 | 191.509 | 1.7048 | 0.1871 | ← p-value for Hypotheses Set 2. |
| Major:Test.Session | 4 | 87.0 | 21.745 | 0.1936 | 0.9412 | ← p-value for Hypotheses Set 3. |
| Residuals | 99 | 11121.1 | 112.334 | | | |

INTERPRETATION: The p-values are in the rightmost column. As you can see, none of them are smaller than α. Therefore, neither major, test session, nor interaction appears to have any significant effect on anxiety scores.

# Section 24: How to Run Mauchly's Test for Sphericity

(Uses data file: AnxietyRepeat2.txt)

In Section 22: "How to Run a Repeated Measures ANOVA," the example uses the data set AnxietyRepeat.txt. This data set contains scores on an Anxiety Test, repeated three times on thirty-six (fictional) students. The test is first given during their first term in college (labelled Fall1), repeated second term (labelled Spr1) and then again in their third term (labelled Fall2). The goal of the testing is to see whether or not, on average, students' anxiety levels are consistent from one term to the next, or whether they change over time as students adjust to college. In Section 22: "How to Run a Repeated Measures ANOVA," you will find the following hypothesis test.

$H_0$: The means of the anxiety scores are the same for all test sessions.
$H_1$: The mean anxiety score for at least one test session is different.

| | |
|---|---|
| > Repeat.Model = lm (Anx.Score ~ Test.Session + ID) | ← Creates a linear model with Anx.Score as a function of Test.Session and ID (individual person) |
| > anova (Repeat.Model) | |

The resulting output is repeated below for reference; see Section 22 for the detailed comments on it.

Analysis of Variance Table
Response: Anx.Score

| | Df | Sum Sq | Mean Sq | F value | Pr(>F) |
|---|---|---|---|---|---|
| Test.Session | 2 | 383.0 | 191.509 | 2.1971 | 0.11874 |
| ID | 35 | 5235.9 | 149.597 | 1.7162 | 0.02781 * |
| Residuals | 70 | 6101.6 | 87.166 | | |

One key assumption for repeated measures ANOVA to be valid is sphericity. Sphericity means that the variances of the differences between all possible pairs of "within subject" conditions are equal. In this example, it means that you would need to have equal variances for the three differences: Fall1 - Spring1, Fall1 - Fall2, and Spring1 - Fall2.

$H_0$: Data appears to satisfy the sphericity property.
$H_1$: Data does not appear to satisfy the sphericity property.
Preset your level of significance; for this example, assume that alpha ($\alpha$) = .05

You can test this sphericity property using a test called Mauchly's Test. In R, it is included in the package: car. Install this package if necessary. You can find an explanation of how to install a package in Section 3 of this Manual. Now you are ready to set up for the test.

Preliminary Step. Rearrange the data set so that it is in rows by student, columns by test session. But leave out the headings. Here that has been done for you and the rearranged data set is called: AnxietyRepeat2.txt. But if you are working on other data, you may want to put it into a spreadsheet to reconfigure it. Then proceed as follows.

Here is an overview of the steps you will follow. Details are supplied, step-by-step, after the overview.

Overview:

1. Read in this new data set.
2. Convert it from a table to a matrix.
3. Tell R the names to use in the model to indicate the repeated measurements.
4. Load the package: car, if not already done.
5. Create the linear model using the matrix.
6. Run the repeated measures analysis on the resulting model and save the results.

Step 1. Read in the rearranged data set, attach it, and have a look at it to be sure it is in the right form.

```
> Data = read.table ("E:/Data Files/AnxietyRepeat2.txt", header = FALSE)
> attach (Data)
> Data
```

The first few lines of data are now arranged as shown below. Notice that where you would ordinarily have headings in a table, R has written the labels V1, V2 and V3. This is OK, because you do not want the actual test session labels to go into the matrix. V1 is the first set of scores from Fall1, V2 is the set from Spring1, and V3 is the last set from Fall3.

The numbers on the far left are not part of the data; R is just numbering the lines as usual.

```
   V1  V2  V3
1  25  22  33
2  11   5  20
3   8  31  13
4  17  13  18
5   7  20  13
   :   :   :
   :   :   :
```

Step 2. Convert this new table to a matrix. The commands you need are as follows.

```
> AnxietyMatrix = as.matrix (Data, rownames.force = NA)
> AnxietyMatrix
```

The “as. matrix” command changes the table to a matrix. Since you called the table “Data” when you read it into R, that name is the first item in the parentheses. The second item, “rownames.force” is set to NA because you are going to supply your own names in the next step.

Step 3. Make a list that tells R what names you want to use for the repeated measurements (also called levels of the factor). There is a catch though; R will try to use the names for the “levels of the factor” in alphabetical order. Therefore, YOU DO NOT want to use Fall1, Spring1, Fall2 as the names in your matrix. That would result in Spring1 being put last, and it should not be. Instead use something like Term1, Term2 and Term3 that will stay in the right order when alphabetized.

You make the list by using the command " c ("Term1", "Term2", "Term3")" inside the "factor" command, as shown below. In order to be able to refer to this later, you need to give it a name; here it is called "design" but you can use a different name if you want.

```
> design = factor (c ("Term1", "Term2", "Term3"))
> design
```

The resulting output that shows what "design" is as follows:

```
[1]      Term1   Term2   Term3
Levels:  Term1   Term2   Term3
```

Step 4. Load the package: car. See Section 3: "How to Find, Install and Load R Packages."

Step 5. Create the linear model using the matrix. In step 2 of this example, you called it AnxietyMatrix. So that is what goes into the command. The last part of the command (that says ~1) is telling R what kind of matrix to compare your matrix against. Also, note that you won't see any output until Step 6.

```
> AnxietyModel = lm (AnxietyMatrix ~ 1)
```

Step 6. Run an ANOVA on this model. Note several things.

When you are working in the package "car", the first letter of the command to do an analysis of variance has to be capitalized. This is different from when you run the same test in the basic R program, where it is not capitalized.

The command is applied to the model you just created, which you named AnxietyModel.

The rest of the command (highlighted below) is telling R to use the names that you specified in "design" as the labels for the repeated measurements/levels of the factor. If you always use the name "design" for your labels, you can just copy this part of the command as written.

```
 > results = Anova (AnxietyModel, idata = data.frame(design), idesign = ~design, type = "III")
> summary (results, multivariate = FALSE)
```

Here is the output from these two lines of code. In the package "car", the output will not include the effect of the individual student by ID. It will only show the effect of the repeated test sessions. This is another difference from the version of the ANOVA command in basic R. It is fine because you are really running this analysis to get the results of Mauchly's sphericity test anyway.

The output is relatively long because it has several parts. The first part of it is the analysis of variance repeated all over again by the "car" package. The second part of it is the results of Mauchly's sphericity test.

Explanatory comments have been added on the right.

Univariate Type III Repeated-Measures ANOVA Assuming Sphericity ← This is just the anova all over again in "car."

| | SS | num Df | Error SS | den Df | F | Pr(>F) |
|---|---|---|---|---|---|---|
| (Intercept) | 49622 | 1 | 5235.9 | 35 | 331.7085 | <2e-16 *** |
| design | 383 | 2 | 6101.6 | 70 | 2.1971 | 0.1187 ← Same p-value as before. |

Mauchly Tests for Sphericity ← This is the start of the output about the sphericity test.

| | Test statistic | p-value | |
|---|---|---|---|
| design | 0.96142 | 0.5123 | ← p-value > α, so the data appears to satisfy sphericity. |

Greenhouse-Geisser and Huynh-Feldt Corrections for Departure from Sphericity ←Correction factors that can be used if sphericity fails; beyond the scope of this discussion.

| | GG eps | Pr(>F[GG]) |
|---|---|---|
| design | 0.96285 | 0.1209 |

| | HF eps | Pr(>F[HF]) |
|---|---|---|
| design | 1.017791 | 0.1187361 |

## Section 25: How to Check Pairs of Data Values for Linear Correlation (Pearson's r)

(Uses data file: HeightWeight.txt)

Frequently, you will have a set of data consisting of pairs of measurements on the same individual, such as a person's height and weight. You may want to determine whether or not these measurement pairs fall approximately on a line. To determine this, the standard method to use is linear correlation.

Here is an example, using the heights (in inches) and weights (in pounds) of 25 fictional army recruits. The question is whether or not they are linearly related. First read in the table of data, make it available to R by "attaching" it and then display it.

```
> HWTable = read.table ("E:/Data Files/HeightWeight.txt", header = TRUE)
> attach (HWTable)
> HWTable
```

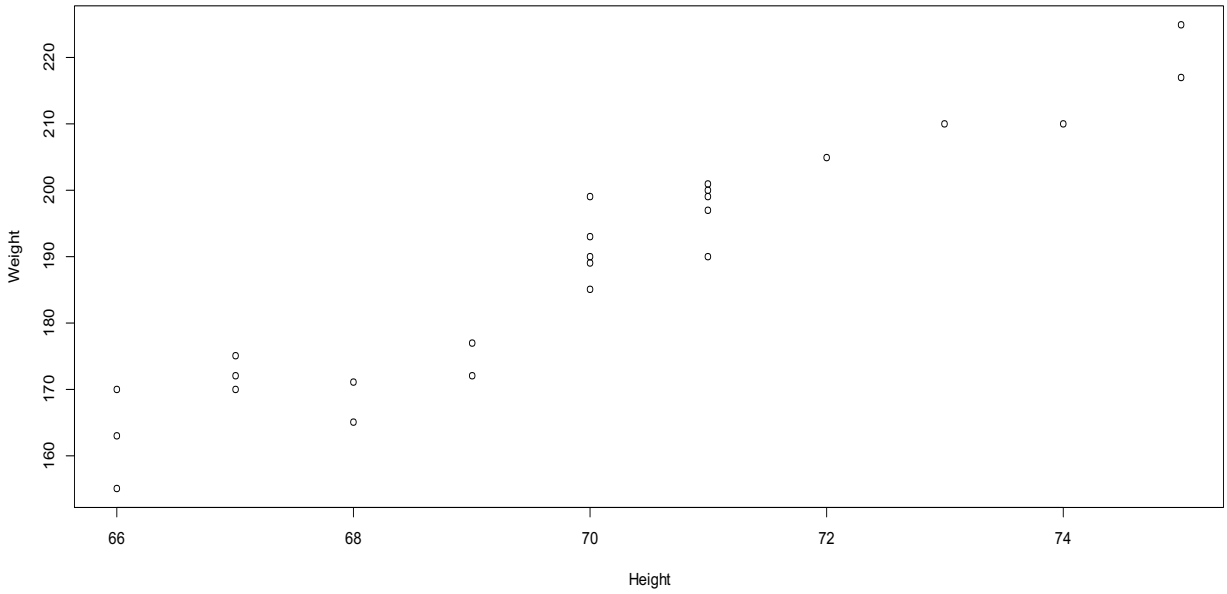
The resulting output is the following.

```
   Height Weight
1      67    175
2      73    210
3      70    189
4      75    225
5      70    193
6      71    197
7      71    200
8      75    217
9      68    165
10     74    210
11     67    170
12     71    201
13     66    163
14     68    171
15     71    190
16     66    170
17     67    172
18     69    172
19     70    199
20     69    177
21     71    199
22     66    155
23     72    205
24     70    190
25     70    185
```

Now create the scatterplot. This is a good idea to do first, as it will help you visualize your data. The first variable that you list will go on the horizontal axis; the second one will go on the vertical axis.

```
> plot (Height, Weight)
```

The resulting graph is as shown below.




This graph certainly looks as if a rising line would fit fairly well through the data set. To get confirmation for this, you should now find the correlation coefficient. The command to do this is "cor" with the names of the variables in parentheses.

```
> cor (Height, Weight)
```

R returns the sample correlation coefficient, called Pearson's r.

```
0.9540717
```

In this example, the value is positive and very close to one. It supports the belief that a rising line is a good way to represent the relationship between height and weight of army recruits.

# Section 26: How to Run a Simple Linear Regression

(Uses data file: HeightWeight.txt)

Suppose you have already made a scatter plot and run a correlation for a set of (X,Y) data pairs, and both indicate that a straight line will be a good fit to the data. Usually you then want to know the specific line that fits the data best, called the regression line.

Refer back to the example of army recruits' height-weight data. You have already seen the scatterplot and the correlation in Section 25 of this Manual. To get the regression line, you continue from your previous work. Here is a repetition of the code from the earlier work in case you need it.

```
> HWTable = read.table ("E:/Data Files/HeightWeight.txt", header = TRUE)
> attach (HWTable)
> HWTable
> plot (Height, Weight)
> cor (Height, Weight)
```

As displayed in the previous section, the scatterplot looked fairly linear and the sample correlation coefficient was 0.9540717, indicating a strong positive linear relationship.

Now you can move on to actually testing the hypothesis that the correlation is significant. That is, you will test the hypotheses:

$H_0$: The true population correlation is zero (no linear relationship).
$H_1$: The true population correlation is not zero (there is a linear relationship).
Preset your level of significance. This example uses alpha ($\alpha$) = .05.

```
> cor.test (Height, Weight)
```

The output from this test follows, with some interpretation added on the right-hand side. Note that it was not really necessary to run the correlation before, as you did in Section 25, because it is included as part of the output of the correlation hypothesis test.

```
Pearson's product-moment correlation

data:  Height and Weight

t = 15.2734,  df = 23,  p-value = 1.568e-13        ← p-value in scientific notation: 1.568 x 10^-13 ;
                                                     this is much less than α.

alternative hypothesis: true correlation is not equal to 0

95 percent confidence interval:                    ← You can be 95% sure that the true
  0.8971521  0.9798250                               population correlation coefficient lies between
                                                     about 0.897 and 0.980.

sample estimates:
    cor       0.9540717                            ← Calculated sample correlation coefficient.
```

INTERPRETATION: Since the p-value is less than α, that indicates that you should reject the null hypothesis. There is sufficient evidence to conclude that a linear relationship exists between height and weight.

Finally, fit the regression line. You use "lm" to create the linear model. The expression (Weight ~ Height) tells R that you want it to consider Weight the response, based on predictor Height. Name the model HWLine. Output includes the slope and the y-intercept.

```
> HWLine =  lm (Weight ~Height)
```

The following output results; one explanatory comment has been added on the right.

```
Call:

lm(formula = Weight ~ Height)

Coefficients:

(Intercept)      Height
-274.856         6.624          ←This tells you that the "y-intercept" is -274.856 and the slope is
                                   6.624. Slope is the coefficient of the predictor, in this case "Height."
```

This means that the resulting regression line is:

Weight = -274.856 + 6.624 * Height

For practical purposes, since you probably only want weight to the nearest pound anyway, you could round the terms to use the equation:

Weight = -274.9 + 6.6 * Height

# Section 27: How to Obtain Residuals and Fits from a Regression Line And Check the Assumptions

(Uses data file: HeightWeight.txt)

This example is a continuation of what you did in Section 26: "How to Run a Simple Linear Regression." First run all of the instructions in that section. The result is a regression line with equation:

| Weight = -274.856 + 6.624 * Height |
|---|

More information can be obtained about the model with the "summary" command, applied to the object HWLine – the name that you gave to the linear model in R if you worked through the previous section of this Manual.

```
> summary (HWLine)
```

The resulting output is at left, with explanatory comments added at right. Recall that the level of significance was alpha (α) = .05.

```
Call:
lm(formula = Weight ~ Height)

Residuals:
    Min      1Q  Median      3Q     Max                ← Summarizes distribution of the residuals (errors).
-10.548  -4.913   1.205   3.582  10.205

Coefficients:
             Estimate  Std. Error  t value  Pr(>|t|)
(Intercept) -274.8556   30.3257    - 9.063  4.73e-09 ***   ← p-value for intercept is less than α;
                                                             the Intercept is significant.

Height         6.6236   0.4337     15.273   1.57e-13 ***   ← p-value for coefficient of Height is less
                                                             than α; slope is significant.
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.632 on 23 degrees of freedom     ← Standard deviation of distribution of
                                                              the residuals is 5.632.

Multiple R-squared:  0.9103,    Adjusted R-squared:  0.9064  ← See discussion below this box.

F-statistic: 233.3 on 1 and 23 DF,      p-value: 1.566e-13  ←Overall p-value is less than α; thus the
                                                             regression line is a good representation
                                                             of the Height and Weight relationship.
```

The term "R-squared" is the square of the correlation coefficient. When you did the original regression, you found the correlation was 0.9540717. If you square that, you get 0.9103 as shown in the box above. This number tells you that about 91% of the variability in Weight is explained by the regression line model. This is a very large effect size, and so the model is very useful.

You can get the list of residuals (errors) with the "resid" command applied to the model (called HWLine above).

```
> resid (HWLine)
```

The output of residuals is as follows.

| 1 | 2 | 3 | 4 | 5 | 6 | …… | 25 |
|---|---|---|---|---|---|---|---|
| 6.0759013 | 1.3344402 | 0.2051708 | 3.0872865 | 4.2051708 | 1.5815939 | …… | -3.7948292 |

Similarly, you can also obtain the fits (predicted values) of Weight that result when the Heights are substituted into the regression equation.

```
> fitted (HWLine)
```

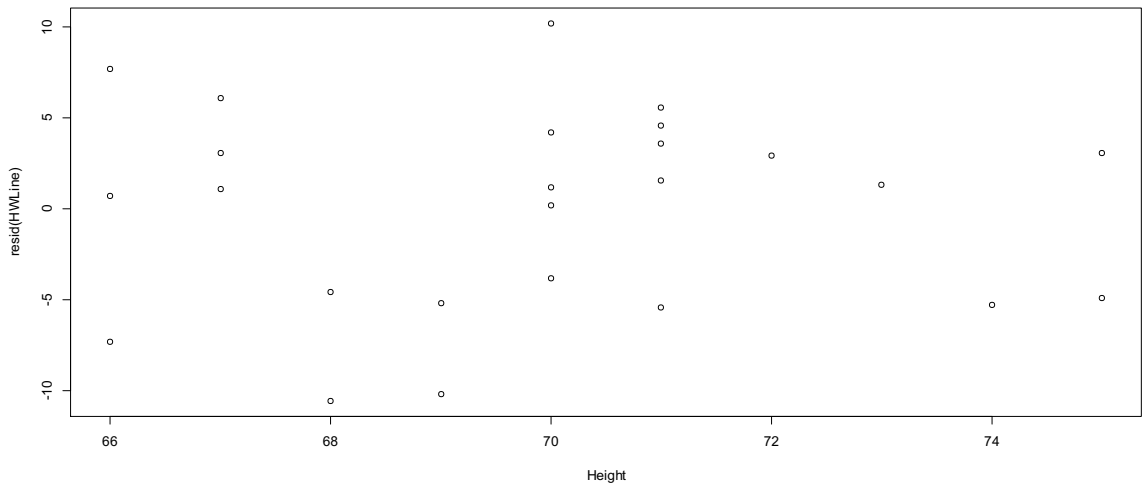
The output of predicted values is as follows.

| 1 | 2 | 3 | 4 | 5 | 6 | …… | 25 |
|---|---|---|---|---|---|---|---|
| 168.9241 | 208.6656 | 188.7948 | 221.9127 | 188.7948 | 195.4184 | …….. | 188.7948 |

This means that if you substitute the data value for Height (67 inches) into the regression equation, it predicts the weight should be 168.9 lbs. This is the first "fit" in the output of predicted values. The residual, or error term, is the difference between the observed Weight and the predicted Weight. You could calculate this yourself, using the first Weight in the data set and the first predicted value. However, it appears as the first value in the list of residuals, i.e., 6.08 lbs. The others work similarly.

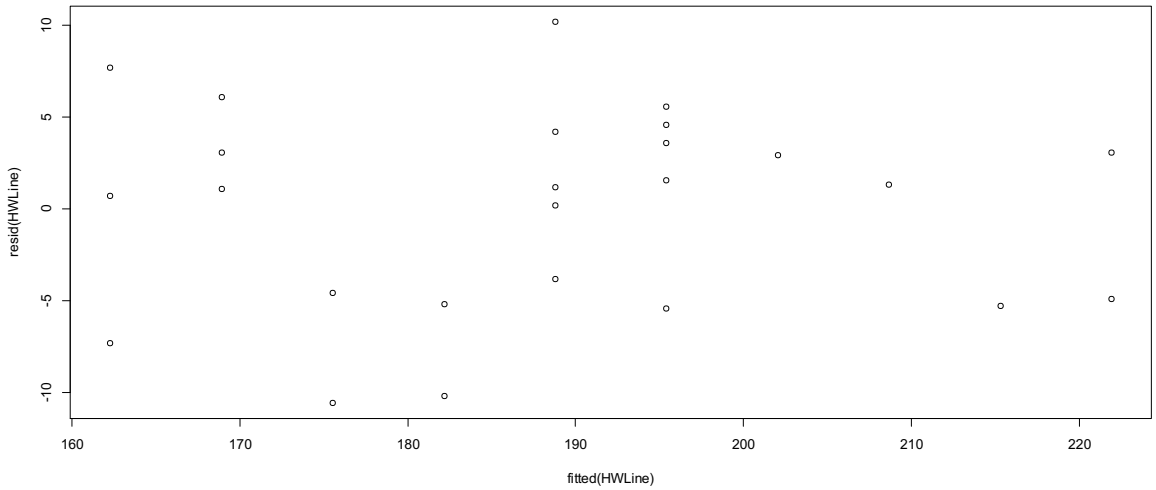
Now for assumption checking. You need to check for independence, common variance and normality. The necessary residual plots and the normality-test graph can be generated. If you want to keep the plots, enter one command at a time, get the graph and copy/paste it into Word. Then enter the next command to get the next graph, and so on. The first graph plots the residuals versus the predictor variable.

```
>plot (Height, resid (HWLine))
```



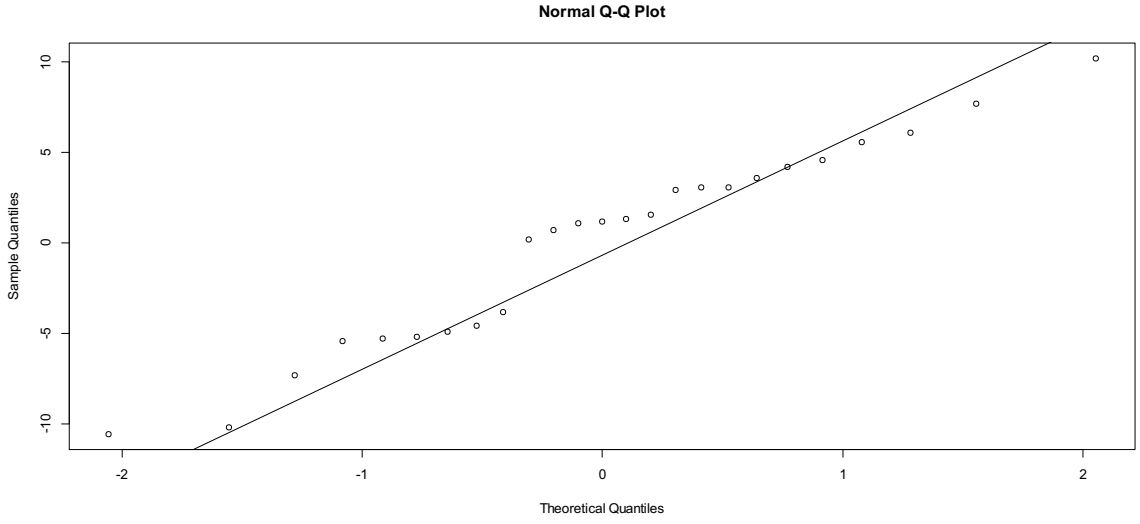
The second graph plots the residuals versus the fitted values.

```
> plot (fitted (HWLine), resid (HWLine))
```



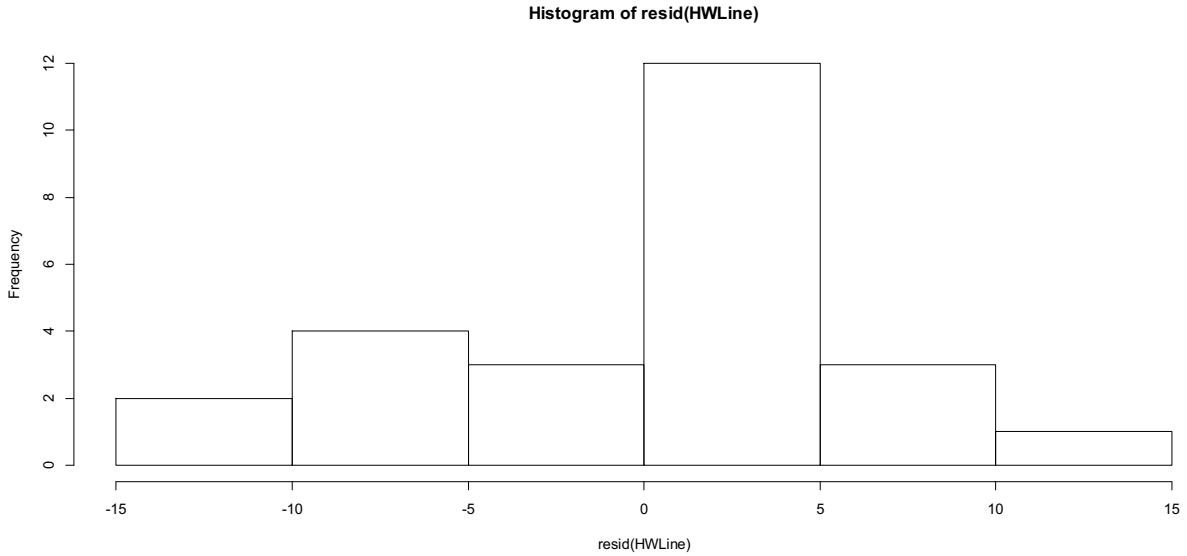

Neither plot shows any pattern, so you can generally believe that the common variance assumption and independence assumption hold for the data set. Now check for normality by creating a normal probability plot of the residuals. These are discussed in Section 10: "How to Check for Normality Using a Normal Probability Plot."

```
> qqnorm (resid (HWLine))
> qqline (resid (HWLine))
```




You may also want to look at a histogram of the residuals.

```
> hist (resid (HWLine))
```




The histogram of residuals looks approximately normal. In the "qq-plot," the points representing the residuals fall fairly well along the normal quantile line. Both of these plots therefore suggest that you can safely accept the normality assumption. All of the assumptions required in order to use regression have been met, so you can use your regression line with confidence.

# Section 28: How to Run a Basic Multiple Regression

(Uses data file: FluData1.txt)

This example uses a data set called FluData1.txt. It contains information on various people who have influenza. The variables and the meaning of the values are:

Age (in years)
Gender (0 = Female, 1 = Male)
Vaccine (0 = No flu shot, 1 = Had flu shot)
Treatment (0 = Not treated, 1 = Treated)
Smoking (0 = Non-smoker, 1 = Smoker)
Temperature (body temperature in degrees Farenheit)
Cough.Severity (rated on a scale from 0 to 8, higher rating means more severe).

The goal is to find a linear equation that describes Cough.Severity as a function of the variables that make a difference, and omit those that do not. Preset the level of significance; in this example, you will use alpha ($\alpha$)= 0.05.

First read in the data table, and attach it so that R can work with it. Display it if you choose.

```
> Data = read.table ("E:/Data Files/FluData1.txt", header = TRUE)
> attach (Data)
> Data
```

A portion of the data set is as shown.

| | Age | Gender | Vaccine | Treatment | Smoking | Temperature | Cough.Severity |
|---|---|---|---|---|---|---|---|
| 1 | 55 | 1 | 0 | 1 | 1 | 98.0 | 1.5 |
| 2 | 17 | 0 | 0 | 1 | 0 | 98.6 | 0.8 |
| 3 | 36 | 1 | 1 | 0 | 0 | 97.9 | 1.1 |
| 4 | 30 | 0 | 0 | 0 | 0 | 97.9 | 0.0 |
| 5 | 63 | 0 | 0 | 0 | 0 | 98.2 | 2.8 |
| : | | | | | | | |
| : | | | | | | | |
| 475 | 14 | 1 | 0 | 1 | 0 | 102.8 | 6.6 |
| 476 | 5 | 0 | 1 | 0 | 0 | 103.2 | 4.2 |
| 477 | 8 | 1 | 0 | 1 | 0 | 101.5 | 4.4 |
| 478 | 5 | 1 | 0 | 0 | 0 | 100.6 | 7.7 |

If you have no idea which of the variables play a role in making a cough more or less severe, you could create a regression model for the response Cough.Severity with all five (except Temperature) other variables as predictors initially.

Comment: The rationale for leaving out Temperature is that it is generally also a response to having influenza, not a factor influencing the severity of other symptoms. However, if you think temperature might affect Cough.Severity, you could certainly include it as a predictor in the model too.

The first line of code creates the model; "lm" stands for linear model. Cough.Severity is on the left side of the ~ symbol as the response variable. The five predictors, separated by plus signs, are on the right. The "summary" command displays details about the resulting model.

```
> CoughModel1 = lm (Cough.Severity ~ Age + Gender + Vaccine + Treatment + Smoking)
> summary (CoughModel1)
```

The resulting output is as follows, with explanatory comments added on the right.

```
lm(formula = Cough.Severity ~ Age + Gender + Vaccine + Treatment + Smoking)

Residuals:
   Min       1Q   Median       3Q      Max
-5.8603  -1.0291   0.1955   1.3049   2.8370

Coefficients:
             Estimate   Std. Error  t value   Pr(>|t|)      ←The last column contains the p-values.
(Intercept)  5.750420   0.156116    36.834    < 2e-16 ***   ← p-value is less than α.
Age         -0.013480   0.004662     2.892    0.00401 **    ← p-value is less than α.
Gender      -0.032635   0.158750    -0.206    0.83721
Vaccine      0.005707   0.212573     0.027    0.97859
Treatment    0.258138   0.168351     1.533    0.12586
Smoking      0.490297   0.246478     1.989    0.04725 *     ← p-value is less than α.
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.72 on 472 degrees of freedom
Multiple R-squared:  0.02432,   Adjusted R-squared:  0.01398   ← Model only explains about 1-2% of
                                                                the variability in Cough.Severity.

F-statistic: 2.353 on 5 and 472 DF,  p-value: 0.03979          ← Overall p-value is less than α.
```

At this point, you might decide to abandon this approach, since the R-squared values indicate that the model explains so little. But for purposes of the example, suppose you decide you want to continue with building a multiple regression model, eliminating the variables from the model that did not show up as statistically significant (their p-values are not less than α). Therefore, you keep only Age and Smoking as predictors.

```
>  CoughModel2 = lm (Cough.Severity ~ Age + Smoking)
>  summary (CoughModel2)
```

This time, the output appears as follows, except that the highlighting has been added. Overall, the second model is simpler in that it contains fewer predictor variables. Also, the same terms continue to show up as being statistically significant. However, the overall performance of the model is still about the same, accounting for only about 1-2% of the variability in Cough.Severity.

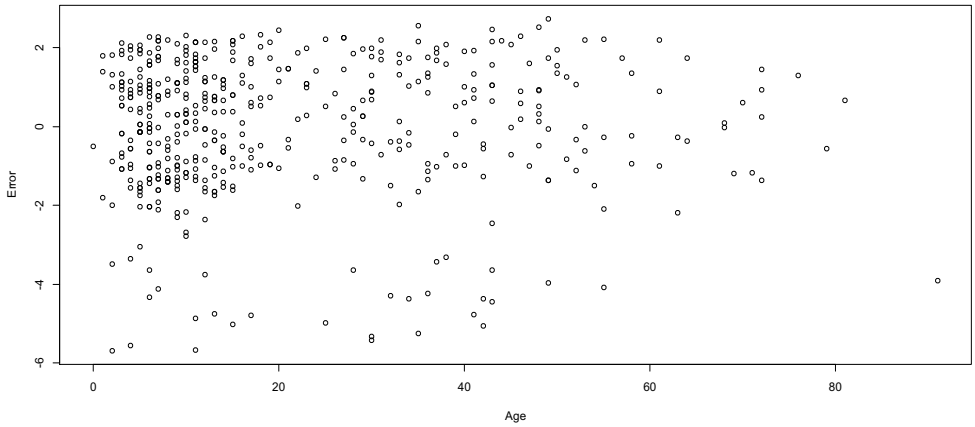
```
Call:
lm(formula = Cough.Severity ~ Age + Smoking)

Residuals:
    Min      1Q  Median      3Q     Max
-5.6874 -1.0199  0.2798  1.2971  2.7338


Coefficients:

             Estimate   Std. Error  t value  Pr(>|t|)
(Intercept)  5.813861   0.121409    47.886   < 2e-16 ***
Age          -0.013217  0.004639    -2.849   0.00457 **
Smoking      0.495108   0.245090    2.020    0.04393 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.719 on 475 degrees of freedom
Multiple R-squared:  0.0194,    Adjusted R-squared:  0.01527
F-statistic: 4.698 on 2 and 475 DF,  p-value: 0.009538
```

Using the estimates that are highlighted, the new, simpler equation is:

Cough.Severity = 5.8 - .013 Age + .495 Smoking

You may want to plot the residuals (error terms) vs. the predictors, and check for patterns. The first graph shows the residuals versus the predictor Age.

```
> Error = resid (CoughModel2)
> plot (Age, Error)
```



No particular pattern is detectable here, except that more young people get the flu than older people. You might speculate that this is probably because the older people have some immunity from prior exposures. But you cannot conclude that from the plot; that is only speculation.

The other residual plot, for Error vs. Smoking, could be done the same way, but it is not very productive to examine because Smoking was a yes/no variable.

You probably also want to examine a plot of the residuals versus the predicted values. First store the fitted values in a variable, such as Fits (used here). Plotting Error vs. Fits from the model can then be done, as follows.

```
> Fits = fitted (CoughModel2)
> plot (Fits, Error)
```

The resulting graph is shown below; again there is no particular pattern (except for the fact that the fitted Cough.Severity scores cluster around the center, showing that the average is around 5.7).