Unit 7 – Predictive Modelling Techniques

Contents

1. Hypothesis testing 4
1.1. Statistical Hypotheses 4
2. t test 14
2.1. Example: One-tailed, one-sample t-test 15
2.2. Example: two-sample t-test 16
2.3. Example: Paired t-test 18
3. Non-Parametric tests 19
3.1. Mann-Whitney U test 20
3.2. Example - Mann-Whitney U test 20
3.3. Wilcoxon Signed Rank 21
3.4. Example - Wilcoxon signed- Rank test 21
3.5. Kruskal Wallis 22
3.6. Example - Kruskal Wallis 22
3.7. Friedman tests 23
3.8. Example - Friedman tests 23
4. Introduction to Chi-Squared Test 25
5. Chi-squared test for independence 25
5.1. Example – Pearson's Chi-squared test 26
5.2. Example – Pearson's Chi-squared test 27
5.3. Example – Yates' continuity correction 28
5.4. Example – Fisher's exact test 29
5.5. Example McNemar's Test 30
6. Chi-squared test for goodness of fit 31
6.1. Example Multinomial Goodness of Fit 31
7. Introduction to ANOVA 32
7.1. One-Way ANOVA 32
7.2. One Way and Two way ANOVA 32
7.3. Example - ANOVA One-way 33

8. Monte Carlo simulation 35
8.1. Monte Carlo simulation 35
8.2. Steps used in Monte Carlo methods 35
8.3. Business Planning sample 36
8.4. Solution using R 37
9. Correlation 40
9.1. Scatter Diagram 40
9.2. Karl Pearson – correlation coefficient 45
9.3. Rank Correlation 46
9.4. Example – correlation coefficient 47
9.5. Example – Rank correlation Coefficient 48
9.6. Spurious Correlation 48
9.7. Probable Error 49
9.8. Exercise Probable Error 50
10. Regression 51
11. Various Regression models 51
12. Statistical Models in R 53
13. Least square method – Line of best fit 55
14. Simple Linear Regression 57
14.1. Estimated Simple Regression Equation 60
14.2. Coefficient of Determination, r^2 or R^2 61
14.3. Significance Test for Linear Regression 62
14.4. Confidence Interval for Linear Regression 63
14.5. Prediction Interval for Linear Regression 64
14.6. Residual Plot 64
14.7. Standard Residual 66
14.8. Normal Probability Plot of Residuals 67
15. Multiple Linear Regression 69
15.1. Estimated Multiple Regression Equation 69
15.2. Multiple Coefficient of Determinatiion 71
15.3. Adjusted Coefficient of Determination 72

15.4. Significant Test for Multiple Linear Regression 73

15.5. Confidence Interval for Multiple Linear Regression 75

15.6. Prediction Interval for Multiple Linear Regression 75

16. Polynomial Regression 76

17. Logistic Regression 80

17.1. Estimated Logistic Regression Equation 80

17.2. Significance Test for Logistic Regression Equation 81

17.3. Another example for Logistic Regression Equation 82

18. Clustering in R 84

19. Possible applications 85

20. Clustering algorithms 85

20.1. K-Means clustering 85

20.2. Example 1 86

20.3. Example 2 91

1. Hypothesis testing

A statistical hypothesis is an assumption about a population parameter. This assumption may or may not be true. Hypothesis testing refers to the formal procedures used by statisticians to accept or reject statistical hypothesis.

1.1. Statistical Hypotheses

The best way to determine whether a statistical hypothesis is true would be to examine the entire population. Since that is impractical, researchers typically examine a random sample from the population. If sample data are not consistent with the statistical hypothesis is rejected.
There are two types of statistical hypotheses.

Null Hypothesis

The null hypothesis, denoted by H_0, is usually the hypothesis that sample observations result purely from chance.

Alternative hypothesis

The alternative hypothesis, denoted by H1 is the hypothesis that sample observations are influenced by some non-random cause.

For example, suppose we wanted to determine whether a coin was fair and unbalanced. A null hypothesis might be that half the flips would result in Heads and half in Tails. The alternative hypothesis might be that the number of Heads and Tails would be very different. Symbolically, these hypotheses would be expressed as

H_0: P = 0.05
H_1: P ≠ 0.05

Hypothesis Tests

Statisticians follow a formal process to determine whether to reject a null hypothesis, based on sample data. This process, called hypothesis testing consists of four steps:

1. State the hypothesis

This involves stating the null and alternative hypotheses. The hypotheses are stated in such a way that they are mutually exclusive. That is, if one is true, the other must be false.

2. Formulate an analysis plan

The analysis plan describes how to use sample data to evaluate the null hypothesis. The evaluation often focuses around a single test statistic.

3. Analyze sample data

Find the value of the test statistic (mean score, proportion, t-score, z-score, etc.) described in the analysis plan.

4. Interpret results

Apply the decision rule described in the analysis plan. If the value of the test statistic is unlikely, based on the null hypothesis, reject the null hypothesis.

Decision Errors

Two types of errors can result from a hypothesis test.

Type I error Type I error occurs when the researcher rejects anull hypothesis when it is true. The probability of committing a Type I error is called the significance level. This probability is also called alpha and is often denoted by α.

Type II error A Type II error occurs when the researcher fails to reject a null hypothesis that is false. The probability of committing a Type II error is called Beta, and is often denoted by β. The probability of committing a Type II error is called the Power of the test.

Decision Rules

The analysis plan includes decision rules for rejecting the null hypothesis. In practice, statisticians describe these decision rules in two ways- with reference to a P-value or with reference to a region of acceptance.

P-value The strength of evidence in support of a null hypothesis is measured by the P-value. Suppose the test statistic is equal to S. The P-value is the probability of observing a test statistic as extreme as S, assuming the null hypothesis is true. If the P-value is less than the significance level, we reject the null hypothesis.

Region of acceptance The region of acceptance is a range of values. If the test statistic falls within the region of acceptance, the null hypothesis is not rejected. The region of acceptance is defined so that the chance of making a Type I error is equal to the significance level.

The set of values outside the region of acceptance is called the region of rejection. If the test statistic falls within the region of rejection, the null hypothesis is rejected. In such cases, we say that the hypothesis has been rejected at the α-level of significance.

One-tailed and Two-tailed Tests

On-tailed test A test of a statistical hypothesis, where the region of rejection is on only one side of the sampling distribution, is called a one-tailed test. For example, suppose the null hypothesis states that the mean is less than or equal to 10. The alternative hypothesis would be that the mean is greater than 10. The region of rejection would consist of a range of numbers located on the right side of sampling distribution, which is a set of numbers greater than 10.

Two-tailed test A test of statistical hypothesis, where the region of rejection is on both sides of the sampling distribution is called a two-tailed test. For example, suppose the null hypothesis states that the means is equal to 10. The alternative hypothesis would be that the mean is less than 10 or greater than 10. The region of rejection would consist of a range of numbers located on both sides of sampling distribution; that is the region of rejection would consist partly of numbers that were less than 10 and partly of the numbers that were greater than 10.

1. Lower Tail Test of Population Mean with known variance

The null hypothesis of the lower tail test of the population mean can be expressed as follows:

$\mu \geq \mu_0$

where μ_0 is a hypothesized lower bound of the true population mean μ.

Let us define the test statistic z in terms of the sample mean, the sample size and the population standard deviation δ:

$$z = \frac{\bar{x} - \mu_0}{\sigma / \sqrt{n}}$$

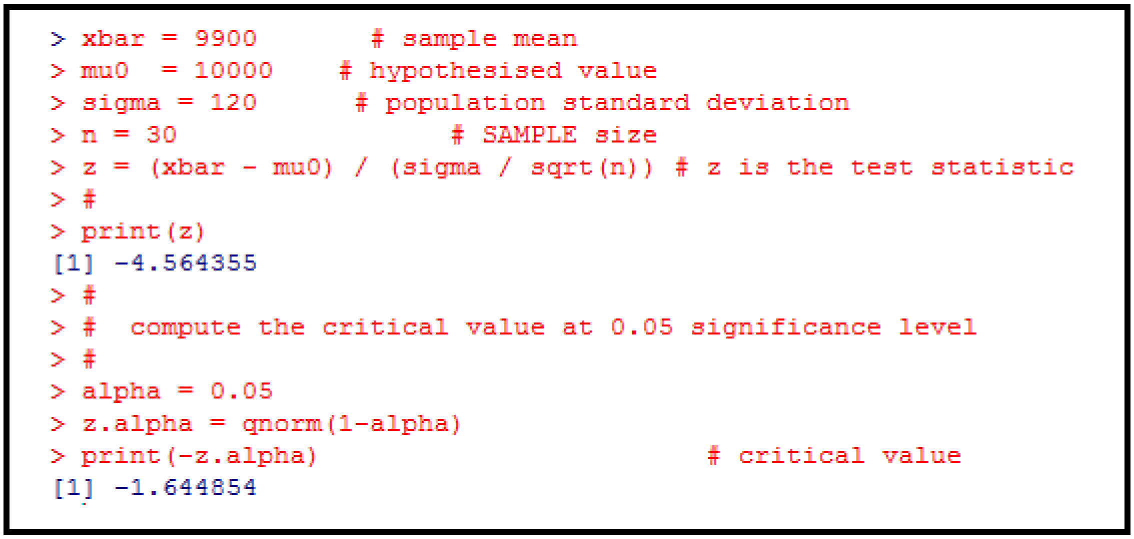
Then the null hypothesis of the lower tail test is to be rejected if $z \leq z_0$, where z_0 is the 100 (1 – α) percentile of the standard normal distribution.

Problem:

Suppose the manufacturer claims that the mean lifetime of a light bulb is more than 10,000 hours. In a sample of 30 light bulbs, it was found that they last only 9,900 hours on average. Assume the population standard deviation is 120 hours. At 0.05 significance level, can we reject the claim by the manufacturer?

Solution:

The null hypothesis is that $\mu \geq$ 10,000. We begin with computing the test statistic.



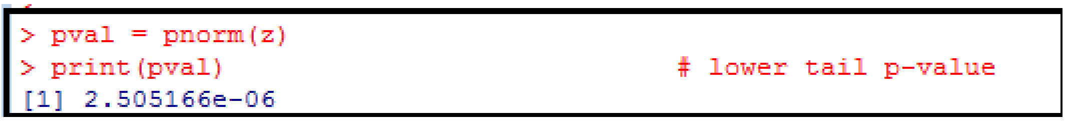
```
> xbar = 9900       # sample mean
> mu0  = 10000    # hypothesised value
> sigma = 120      # population standard deviation
> n = 30                 # SAMPLE size
> z = (xbar - mu0) / (sigma / sqrt(n)) # z is the test statistic
> #
> print(z)
[1] -4.564355
> #
> #  compute the critical value at 0.05 significance level
> #
> alpha = 0.05
> z.alpha = qnorm(1-alpha)
> print(-z.alpha)                        # critical value
[1] -1.644854
```

Conclusion

The test statistic -4.564355 is less than the critical value of -1.6449. Hence, at 0.05 significance level, we reject the claim that the mean lifetime of a light bulb is above 10,000 hours.

Alternative solution

Instead of using the critical value, we apply the pnorm function to compute the lower tail p-value of the test statistic. As it turns out to be less than the 0.05 significance level, we reject the null hypothesis that mean lifetime of a light bulb is above 10,000 hours.



```
> pval = pnorm(z)
> print(pval)                          # lower tail p-value
[1] 2.505166e-06
```

2. Upper Tail Test of Population Mean with known variance

The null hypothesis of the upper tail test of the population mean can be expressed as follows:

$\mu \le \mu_0$

where μ_0 is a hypothesized upper bound of the true population mean μ.

Let us define the test statistic z in terms of the sample mean, the sample size and the population standard deviation δ:

$$\boxed{z = \frac{\bar{x} - \mu_0}{\sigma / \sqrt{n}}}$$

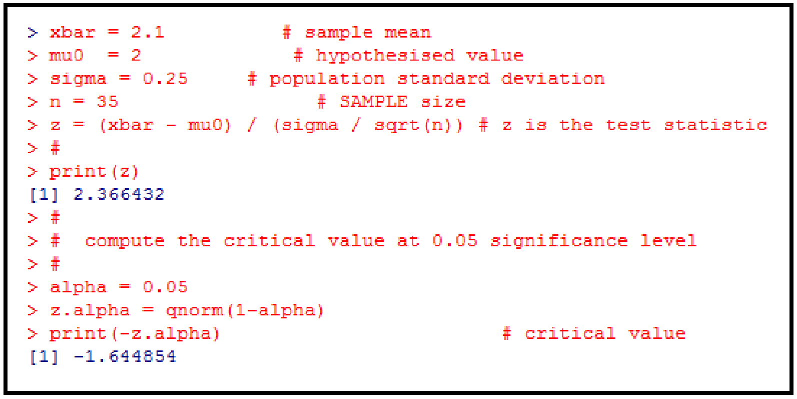
Then the null hypothesis of the upper tail test is to be rejected if $z \ge z_0$, where z_0 is the 100 (1 – α) percentile of the standard normal distribution.

Problem:

Suppose the food label on a cookie bag states that there is at most 2 grams of saturated fat in a single cookie. In a sample of 35 cookies, it is found that the mean amount of saturated fat per cookie is 2.1 grams. Assume that the population standard deviation is 0.25 grams. At 0.05 significance level, can we reject the claim on food label?

Solution

The null hypothesis is that $\mu \le 2$. We begin with computing the test statistic.



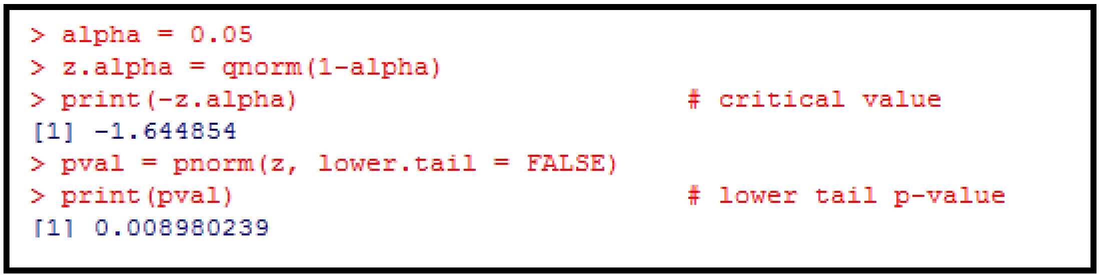
```
> xbar = 2.1          # sample mean
> mu0  = 2             # hypothesised value
> sigma = 0.25     # population standard deviation
> n = 35                 # SAMPLE size
> z = (xbar - mu0) / (sigma / sqrt(n)) # z is the test statistic
> #
> print(z)
[1] 2.366432
> #
> #  compute the critical value at 0.05 significance level
> #
> alpha = 0.05
> z.alpha = qnorm(1-alpha)
> print(-z.alpha)                    # critical value
[1] -1.644854
```

Conclusion

The test statistic 2.366432 is greater than the critical value of 1.644854. Hence at 0.05 significance level, we reject the claim that there is at most 2 grams of saturated fat in a cookie.

Alternative Solution

Instead of using the critical value, we apply the pnorm function to compute the upper tail p-value of the test statistic. As it turns out to be less than the 0.05 significance level, we reject the null hypothesis that μ ≤ 2.



```
> alpha = 0.05
> z.alpha = qnorm(1-alpha)
> print(-z.alpha)                          # critical value
[1] -1.644854
> pval = pnorm(z, lower.tail = FALSE)
> print(pval)                              # lower tail p-value
[1] 0.008980239
```

3. Two- Tailed Test of Population Mean with known Variance

The null hypothesis of the upper tail test of the population mean can be expressed as follows:

$\mu = \mu_0$

where μ_0 is a hypothesized upper bound of the true population mean μ.

Let us define the test statistic z in terms of the sample mean, the sample size and the population standard deviation δ:

$$z = \frac{\bar{x} - \mu_0}{\sigma / \sqrt{n}}$$

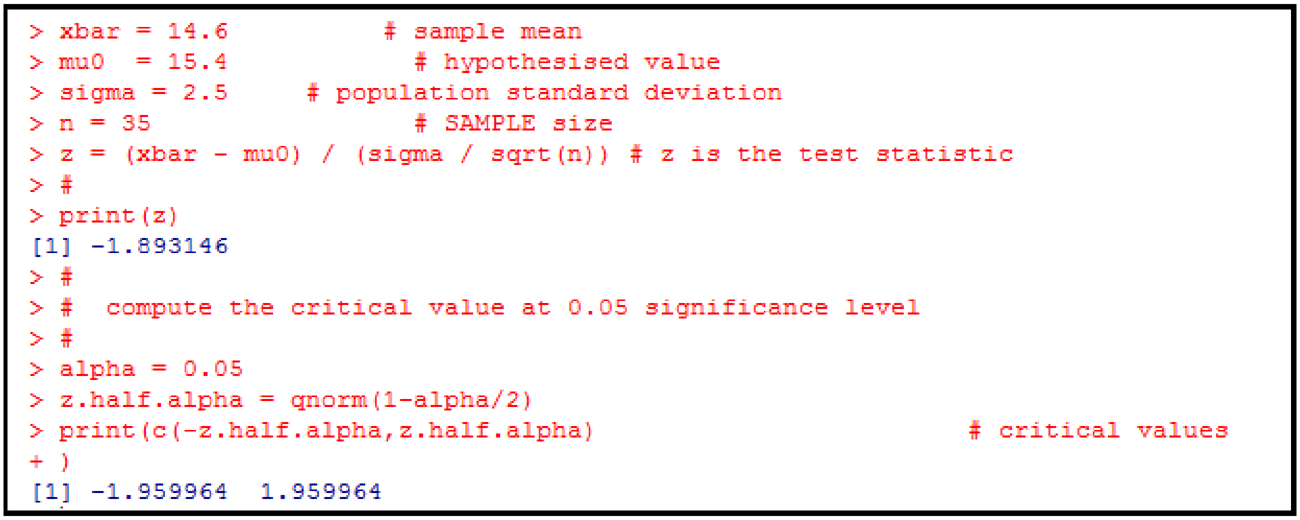
Then the null hypothesis of the two-tailed test is to be rejected if $z \leq z_{0/2}$ or $z \geq z_{0/2}$, where $z_{0/2}$ is the 100 (1 – α/2) percentile of the standard normal distribution.

Problem:

Suppose the mean weight of King Penguins found in Antarctic colony last year was 15.4 kg. In a sample of 35 penguins same time this year in the same colony, it is found that the mean penguin weight is 14.6 kg. Assume that the population standard deviation is 2.5 kg. At 0.05 significance level, can we reject the null hypothesis that the mean penguin weight does not differ from last year?

Solution

The null hypothesis is that μ = 15.4. We begin with computing the test statistic.



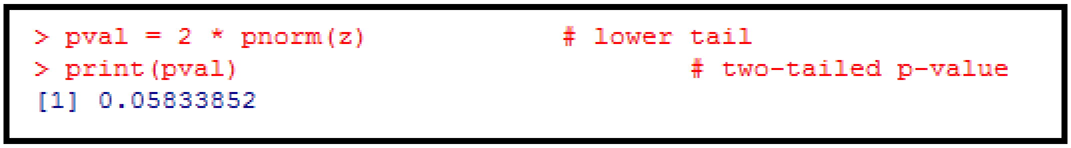
```
> xbar = 14.6           # sample mean
> mu0  = 15.4             # hypothesised value
> sigma = 2.5      # population standard deviation
> n = 35                  # SAMPLE size
> z = (xbar - mu0) / (sigma / sqrt(n)) # z is the test statistic
> #
> print(z)
[1] -1.893146
> #
> #  compute the critical value at 0.05 significance level
> #
> alpha = 0.05
> z.half.alpha = qnorm(1-alpha/2)
> print(c(-z.half.alpha,z.half.alpha)                       # critical values
+ )
[1] -1.959964  1.959964
```

Conclusion

The test statistic -1.893146 lies between the critical values -1.959964 and 1.959964. Hence at 0.05 significance level, we do not reject the null hypothesis that the mean penguin weight does not differ from last year.

Alternative Solution

Instead of using the critical value, we apply the pnorm function to compute the two-tailed p-value of the test statistic. It doubles the lower tail p-value as the sample mean is less than the hypothesized value.
As it turns out to be greater than the 0.05 significance level, we do not reject the null hypothesis that μ = 15.4



```
> pval = 2 * pnorm(z)               # lower tail
> print(pval)                               # two-tailed p-value
[1] 0.05833852
```

4. Lower Tail Test of Population Mean with Unknown variance

The null hypothesis of the lower tail test of the population mean can be expressed as follows:

$\mu \geq \mu_0$

where μ_0 is a hypothesized lower bound of the true population mean μ.

Let us define the test statistic t in terms of the sample mean, the sample size and the population standard deviation *s*:

$$t = \frac{\bar{x} - \mu_0}{s / \sqrt{n}}$$

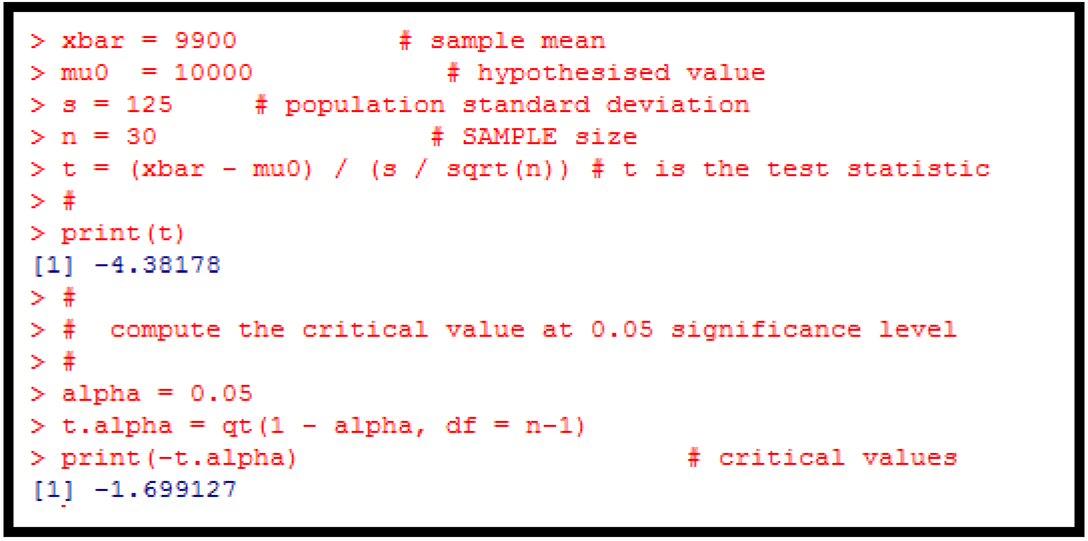
Then the null hypothesis of the lower tail test is to be rejected if $t \le t_{0,}$ where t_0 is the 100 (1 – α/2) percentile of the Student t distribution with n – 1 degrees of freedom.

Problem:

Suppose the manufacturer claims that the mean lifetime of a light bulb is more than 10,000 hours. In a sample of 30 light bulbs, it is found that they only last 9,900 hours on average. Assume that the sample standard deviation is 125 hours. At 0.05 significance level, can we reject the claim by the manufacturer?

Solution

The null hypothesis is that μ ≥ 10000. We begin with computing the test statistic.



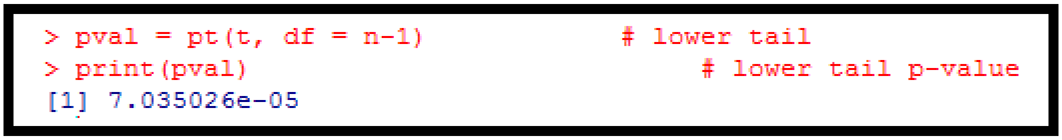
```
> xbar = 9900          # sample mean
> mu0  = 10000            # hypothesised value
> s = 125      # population standard deviation
> n = 30                 # SAMPLE size
> t = (xbar - mu0) / (s / sqrt(n)) # t is the test statistic
> #
> print(t)
[1] -4.38178
> #
> #  compute the critical value at 0.05 significance level
> #
> alpha = 0.05
> t.alpha = qt(1 - alpha, df = n-1)
> print(-t.alpha)                      # critical values
[1] -1.699127
```

Conclusion

The test statistic -4.38178 is less than the critical value of -1.6991. Hence at 0.05 significance level, we can reject the null hypothesis that the mean life time of a light bulb is above 10,000 hours.

Alternative Solution

Instead of using the critical value, we apply the pt function to compute the lower tail p-value of the test statistic. As it turns out to be less than the 0.05 significance level, we can reject the null hypothesis that μ ≥ 10000.



```
> pval = pt(t, df = n-1)            # lower tail
> print(pval)                           # lower tail p-value
[1] 7.035026e-05
```

5. Upper Tail Test of Population Mean with Unknown variance

The null hypothesis of the upper tail test of the population mean can be expressed as follows:

$\mu \leq \mu_0$

where μ_0 is a hypothesized upper bound of the true population mean μ.

Let us define the test statistic t in terms of the sample mean, the sample size and the population standard deviation *s*:

$$\boxed{t = \frac{\bar{x} - \mu_0}{s / \sqrt{n}}}$$

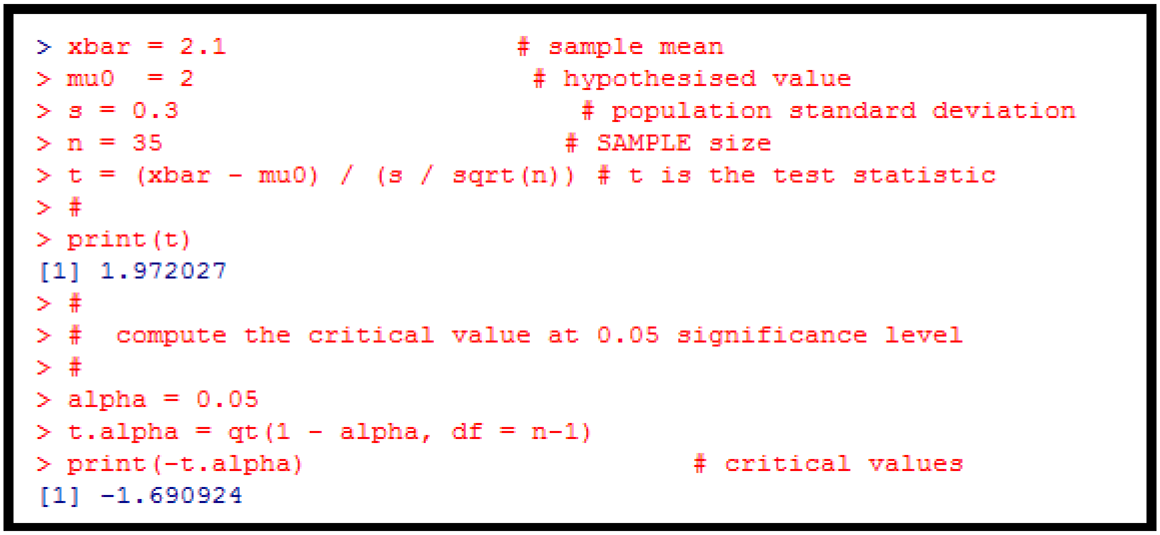
Then the null hypothesis of the upper tail test is to be rejected if $t \geq t_0$, where t_0 is the 100 (1 – α/2) percentile of the Student t distribution with n – 1 degrees of freedom.

Problem:

Suppose the local food label on a cookie bag states that there is at most 2 grams of saturated fat in a single cookie. In a sample of 35 cookies, it is found that the mean amount of saturated fat per cookie is 2.1 gram. Assume that the sample standard deviation is 0.3 grams. At 0.05 significance level, can we reject the claim on food label?

Solution

The null hypothesis is that $\mu \leq 2$. We begin with computing the test statistic.



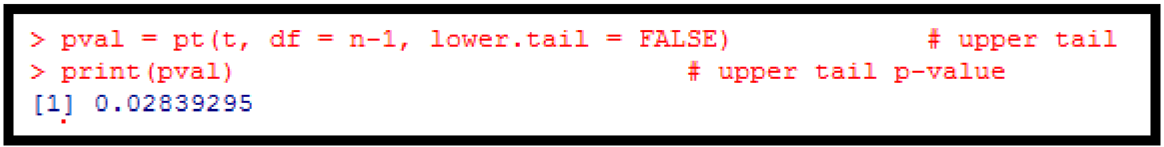
```
> xbar = 2.1                     # sample mean
> mu0  = 2                        # hypothesised value
> s = 0.3                           # population standard deviation
> n = 35                           # SAMPLE size
> t = (xbar - mu0) / (s / sqrt(n)) # t is the test statistic
> #
> print(t)
[1] 1.972027
> #
> #  compute the critical value at 0.05 significance level
> #
> alpha = 0.05
> t.alpha = qt(1 - alpha, df = n-1)
> print(-t.alpha)                        # critical values
[1] -1.690924
```

Conclusion

The test statistic 1.9720 is greater than the critical value of 1.6991. Hence at 0.05 significance level, we can reject the null hypothesis that the there is at most 2 grams of saturated fat in a single cookie.

Alternative Solution

Instead of using the critical value, we apply the pt function to compute the upper tail p-value of the test statistic. As it turns out to be less than the 0.05 significance level, we can reject the null hypothesis that μ ≤ 2.



```
> pval = pt(t, df = n-1, lower.tail = FALSE)          # upper tail
> print(pval)                                # upper tail p-value
[1] 0.02839295
```

6. Two- Tailed Test of Population Mean with Unknown Variance

The null hypothesis of the two-tailed test of the population mean can be expressed as follows:

$\mu = \mu_0$

where μ_0 is a hypothesized upper bound of the true population mean μ.

Let us define the test statistic t in terms of the sample mean, the sample size and the population standard deviation *s*:

$$t = \frac{\bar{x} - \mu_0}{s / \sqrt{n}}$$

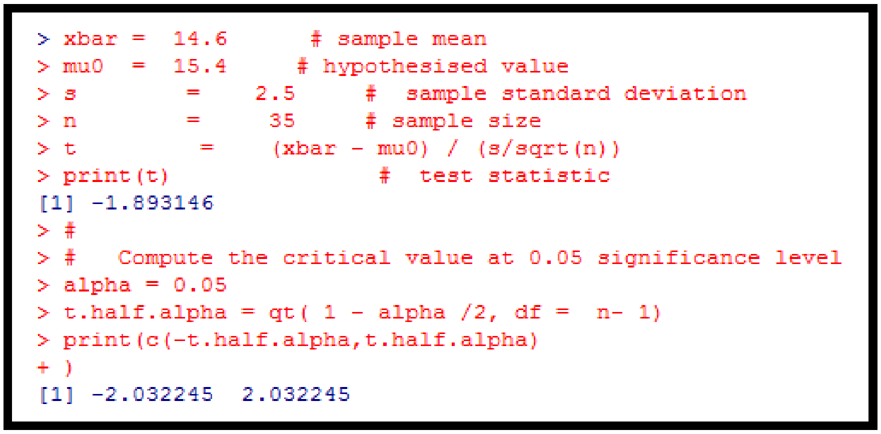
Then the null hypothesis of the two-tailed test is to be rejected if $t \leq t_{0/2}$ or $t \geq t_{0/2,}$ where $t_{0/2}$ is the 100 (1 – α/2) percentile of the Student t distribution with n – 1 degrees of freedom.

Problem:

Suppose the mean of King Penguins found in an Antarctic colony last year was 15.4 kg. In a sample of 35 penguins same time this year in the same colony, the mean penguin weight is 14.6 kg. Assume that the sample standard deviation is 2.5 grams. At 0.05 significance level, can we reject the null hypothesis that the mean penguin weight does not differ from last year?

Solution

The null hypothesis is that μ = 15.4. We begin with computing the test statistic.



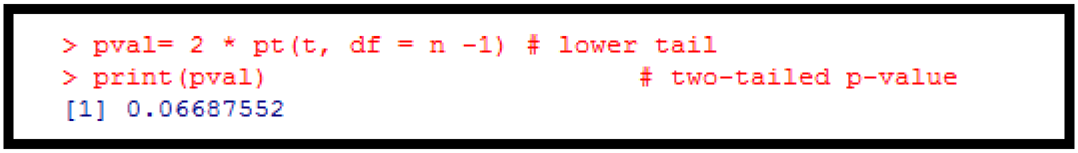
```
> xbar =  14.6      # sample mean
> mu0  =  15.4     # hypothesised value
> s         =    2.5     #  sample standard deviation
> n         =     35     # sample size
> t          =    (xbar - mu0) / (s/sqrt(n))
> print(t)                 #  test statistic
[1] -1.893146
> #
> #   Compute the critical value at 0.05 significance level
> alpha = 0.05
> t.half.alpha = qt( 1 - alpha /2, df =  n- 1)
> print(c(-t.half.alpha,t.half.alpha)
+ )
[1] -2.032245  2.032245
```

Conclusion

The test statistic -1.893146 lies between the critical values of -2.032245 and 2.032245. Hence at 0.05 significance level, we do not reject the null hypothesis that the mean penguin weight does not differ from last year.

Alternative Solution

Instead of using the critical value, we apply the pt function to compute the two-tailed p-value of the test statistic. It doubles the lower tail p-value as the sample mean is less than the hypothesized value. Since it turns out to be greater than the 0.05 significance level, we do not reject the null hypothesis that μ = 15.4.



```
> pval= 2 * pt(t, df = n -1) # lower tail
> print(pval)                         # two-tailed p-value
[1] 0.06687552
```

2. t test

A t-test is used to test hypotheses about the mean value of a population from which a sample is drawn. A t-test is suitable if the data is believed to be drawn from a normal distribution, or if the sample size is large.

The function t.test() can be used to perform both one and two sample t-tests on vectors of data. The function contains a variety of options and can be called as follows:

t.test(x,y = NULL, alternative = c("two sided","less","greater"), mu= 0, paired= FALSE, var.equal= FALSE, conf.level = 0.95)

Here, x is a numeric vector of data values and y is an optional numeric vector of data values. If y is excluded, the function performs a one-sample t-test on the data contained in x, if it is included it performs a two-sample t-tests using both x and y.

The option mu provides a number indicating the true value of the mean (or difference in means if you are performing a two-sample test) under the null hypothesis. The option alternative is a character string specifying the alternative hypothesis, and must be one of the following: "two sided" (which is default), "greater" or "less" depending on whether the alternative hypothesis is that the mean is different than, greater than or less than mu, respectively.

The option paired indicates whether or not you want a paired t-test (TRUE= yes and FALSE = no). If you leave this option out it defaults to FALSE.

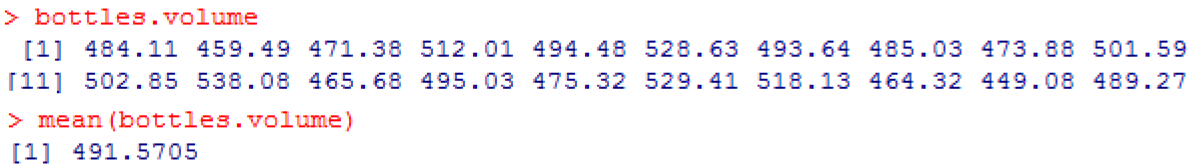
The option var.equal is a logical variable indicating whether or not to assume the two variances as being equal when performing a two-sample t-test. If TRUE then the pooled variance is used to estimate the variance otherwise the Welch (or Satterthwaite) approximation to the degree of freedom is used. If you leave this option out it defaults to FALSE.

Finally, the option conf.level determines the confidence level of the reported confidence interval for μ_0 in the one-sample case and μ_1- μ_2 in the two-sample case.

2.1. Example: One-tailed, one-sample t-test

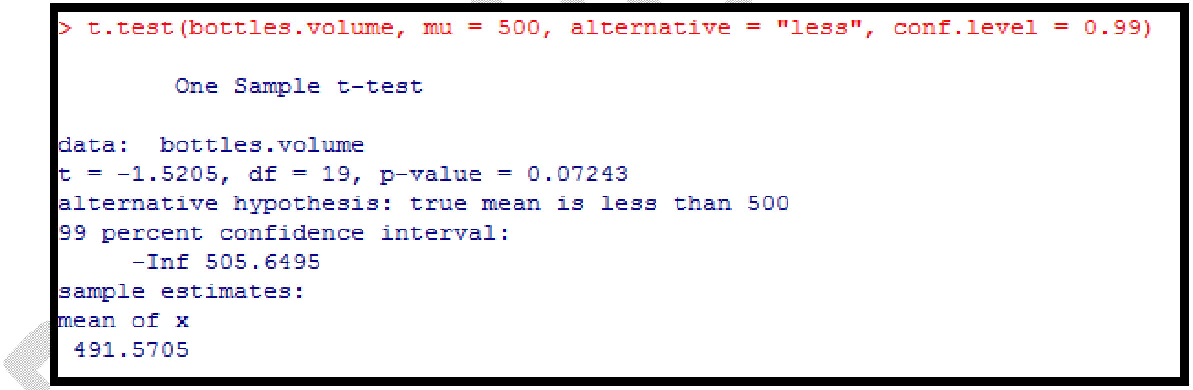
A bottle filling machine is set to fill bottles with drinking water to a volume of 500 ml. The actual volume is known to follow a normal distribution. The manufacturer believes the machine is under filling bottles. A sample of 20 bottles is taken and the volume of liquid is measured. The results are given in the bottles dataset, which is available here.

bottles.volume <- c(484.11, 459.49, 471.38, 512.01, 494.48, 528.63, 493.64, 485.03, 473.88, 501.59, 502.85, 538.08, 465.68, 495.03, 475.32, 529.41, 518.13, 464.32, 449.08, 489.27)



```
> bottles.volume
 [1] 484.11 459.49 471.38 512.01 494.48 528.63 493.64 485.03 473.88 501.59
[11] 502.85 538.08 465.68 495.03 475.32 529.41 518.13 464.32 449.08 489.27
> mean(bottles.volume)
[1] 491.5705
```

Suppose you want to use a one-sample t-test to determine whether the bottles are being consistently under-filled, or whether the low mean volume for the sample is purely the result of random variation. A one-sided test is suitable because the manufacturer is specifically interested in knowing whether the volume is less than 500 ml. The test has the null hypothesis that the mean filling volume is equal to 500 ml, and the alternative hypothesis that the mean filling volume is less than 500 ml. A significance level of 0.01 is to be used.



```
> t.test(bottles.volume, mu = 500, alternative = "less", conf.level = 0.99)

        One Sample t-test

data:  bottles.volume
t = -1.5205, df = 19, p-value = 0.07243
alternative hypothesis: true mean is less than 500
99 percent confidence interval:
     -Inf 505.6495
sample estimates:
mean of x 
 491.5705
```

From the output, we can see that the mean bottle volume for sample is 491.6 ml. The one-sided 99 % confidence interval tells us that the mean filling volume is likely to be less than 505.6 ml. The p-value of 0.07243 tells us that if the mean filling volume of the machine were 500 ml, the probability of selecting a sample with a mean volume less than or equal to this one would be approximately 7 %.

Since the p-value is not less than the significance level of 0.01, we cannot reject the null hypothesis that the mean filling volume is equal to 500 ml. This means that there is no evidence that the bottles are being under-filled.

2.2. Example: two-sample t-test

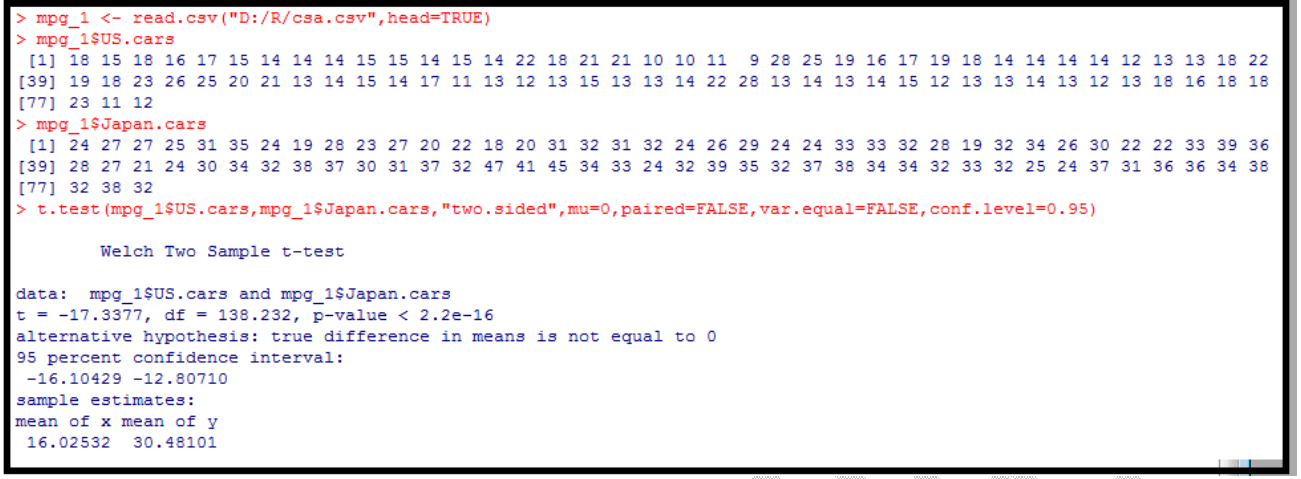
The following is the data used for the two-sample t-test example. The first column is miles per gallon for US cars and the second column is miles per gallon for Japanese cars.

We are testing the hypothesis that the population means are equal for the two samples. We assume that the variances for the two samples are equal.

#	Mpg of US Cars	Mpg of Japan cars
1	18	24
2	15	27
3	18	27
4	16	25
5	17	31
6	15	35
7	14	24
8	14	19
9	14	28
10	15	23
11	15	27
12	14	20
13	15	22
14	14	18
15	22	20
16	18	31
17	21	32
18	21	31
19	10	32
20	10	24
21	11	26
22	9	29
23	28	24
24	25	24
25	19	33
26	16	33
27	17	32
28	19	28
29	18	19
30	14	32
31	14	34
32	14	26
33	14	30
34	12	22
35	13	22

#	Mpg of US Cars	Mpg of Japan cars
36	13	33
37	18	39
38	22	36
39	19	28
40	18	27
41	23	21
42	26	24
43	25	30
44	20	34
45	21	32
46	13	38
47	14	37
48	15	30
49	14	31
50	17	37
51	11	32
52	13	47
53	12	41
54	13	45
55	15	34
56	13	33
57	13	24
58	14	32
59	22	39
60	28	35
61	13	32
62	14	37
63	13	38
64	14	34
65	15	34
66	12	32
67	13	33
68	13	32
69	14	25
70	13	24
71	12	37
72	13	31
73	18	36
74	16	36
75	18	34
76	18	38
77	23	32
78	11	38

#	Mpg of US Cars	Mpg of Japan cars
79	12	32



```
> mpg_1 <- read.csv("D:/R/csa.csv",head=TRUE)
> mpg_1$US.cars
 [1] 18 15 18 16 17 15 14 14 14 15 15 14 15 14 22 18 21 21 10 10 11  9 28 25 19 16 17 19 18 14 14 14 14 12 13 13 18 22
[39] 19 18 23 26 25 20 21 13 14 15 14 17 11 13 12 13 15 13 13 14 22 28 13 14 13 14 15 12 13 13 14 13 12 13 18 16 18 18
[77] 23 11 12
> mpg_1$Japan.cars
 [1] 24 27 27 25 31 35 24 19 28 23 27 20 22 18 20 31 32 31 32 24 26 29 24 24 33 33 32 28 19 32 34 26 30 22 22 33 39 36
[39] 28 27 21 24 30 34 32 38 37 30 31 37 32 47 41 45 34 33 24 32 39 35 32 37 38 34 34 32 33 32 25 24 37 31 36 36 34 38
[77] 32 38 32
> t.test(mpg_1$US.cars,mpg_1$Japan.cars,"two.sided",mu=0,paired=FALSE,var.equal=FALSE,conf.level=0.95)

        Welch Two Sample t-test

data:  mpg_1$US.cars and mpg_1$Japan.cars
t = -17.3377, df = 138.232, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -16.10429 -12.80710
sample estimates:
mean of x mean of y
 16.02532  30.48101
```

H_0: $\mu_1 = \mu_2$

H_1: $\mu_1 != \mu_2$

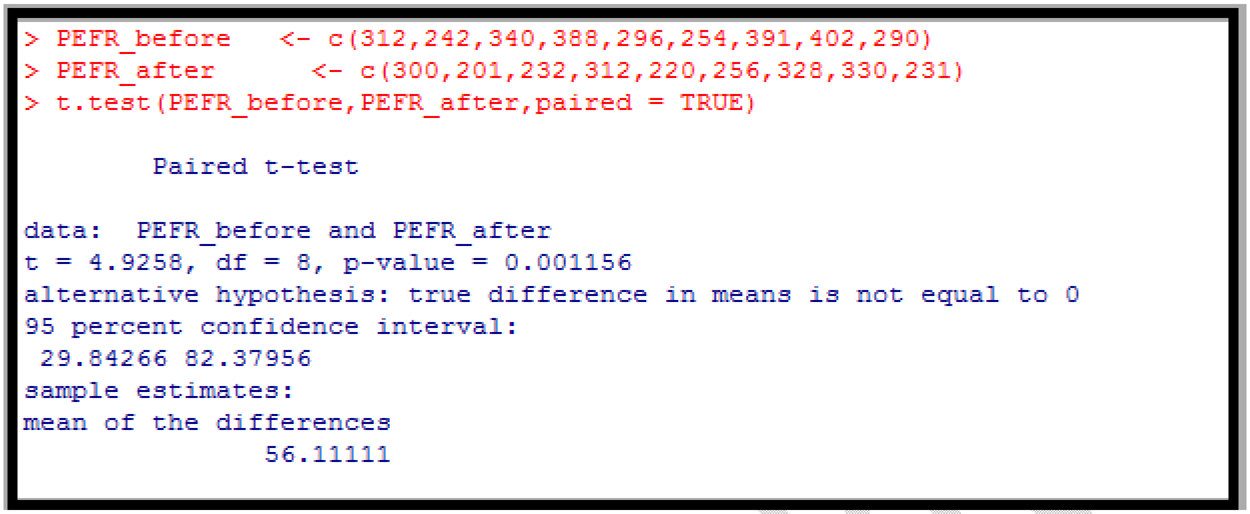
*Since the p-value is very much less than the significance level of 0.05, we **reject** the null hypothesis and conclude that the **two population means are different** at the 0.05 significance level.*

2.3. Example: Paired t-test

Nine asthmatic patients are randomly selected for a walk on a cold winter day. Comparison of peak expiratory flow rate before and after a walk on a cold winter's day is made.

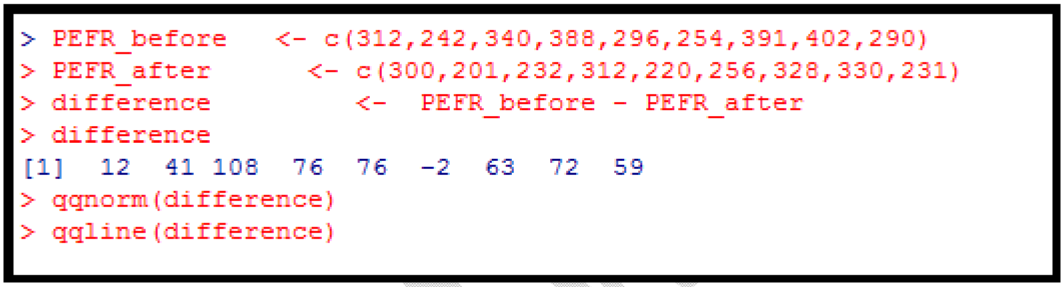
Patient	Before	After
1	312	300
2	242	201
3	340	232
4	388	312
5	296	220
6	254	256
7	391	328
8	402	330
9	290	231

Check if there is any significance difference between the means.

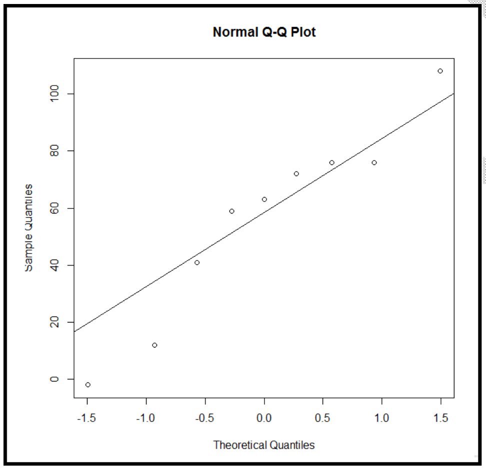


```
> PEFR_before   <- c(312,242,340,388,296,254,391,402,290)
> PEFR_after      <- c(300,201,232,312,220,256,328,330,231)
> t.test(PEFR_before,PEFR_after,paired = TRUE)

        Paired t-test

data:  PEFR_before and PEFR_after
t = 4.9258, df = 8, p-value = 0.001156
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 29.84266 82.37956
sample estimates:
mean of the differences
               56.11111
```

Since the p-value 0.001156 is less than the 5 % significance level, we reject the null hypothesis of no difference between the means is clearly rejected.



```
> PEFR_before   <- c(312,242,340,388,296,254,391,402,290)
> PEFR_after      <- c(300,201,232,312,220,256,328,330,231)
> difference         <-  PEFR_before - PEFR_after
> difference
[1]  12  41 108  76  76  -2  63  72  59
> qqnorm(difference)
> qqline(difference)
```



3. Non-Parametric tests

Nonparametric statistics are statistics not based on parameterized families of probability distributions. They include both descriptive and inferential statistics. The typical parameters are the mean, variance, etc. Unlike parametric statistics, nonparametric statistics make no assumptions about the probability distributions of the variables being assessed.

Non-parametric methods are widely used for studying populations that take on a ranked order (such as movie reviews receiving one to four stars). The use of non-parametric methods may be necessary when having a ranking but no clear numerical interpretation, such as when assessing preferences. In terms of levels of measurement, non-parametric methods result in "ordinal" data.

R provides functions for carrying out the following nonparametric tests of group differences:

a. Mann-Whitney U

b. Wilcoxon Signed Rank

c. Kruskal Wallis

d. Friedman tests

3.1. Mann-Whitney U test

Two data samples are independent if they come from distinct populations and the samples do not affect each other. Using the Mann-Whitney-Wilcoxon Test, we can decide whether the population distributions are identical without assuming them to follow the normal distribution.

Unlike the independent-samples t-test, the Mann-Whitney U test allows you to draw different conclusions about your data depending on the assumptions you make about your data's distribution. These conclusions can range from simply stating whether the two populations differ through to determining if there are differences in medians between groups.

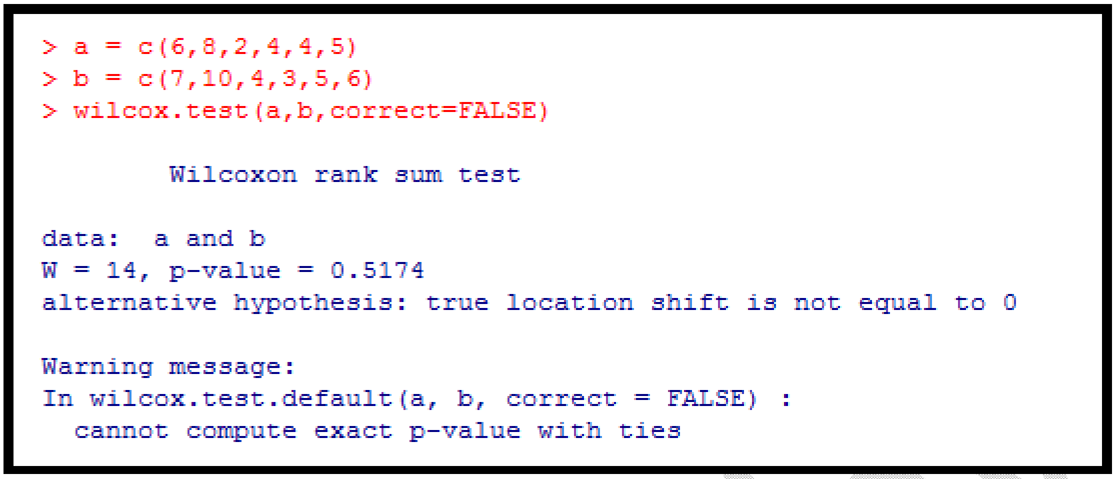
3.2. Example - Mann-Whitney U test

You would like to test if the mean of goals suffered by two football teams over the years is the same.

Team A	6	8	2	4	4	5
Team B	7	10	4	3	5	6

The Wilcoxon-Matt-Whitney test (or Wilcoxon rank sum test or Mann-Whitney U- test) is used when you are asked to compare the means of two groups that do not follow a normal distribution: it is a non-parametrical test. It is the equivalent of t-test, applied for independent samples.

Solution:



```
> a = c(6,8,2,4,4,5)
> b = c(7,10,4,3,5,6)
> wilcox.test(a,b,correct=FALSE)

        Wilcoxon rank sum test

data:  a and b
W = 14, p-value = 0.5174
alternative hypothesis: true location shift is not equal to 0

Warning message:
In wilcox.test.default(a, b, correct = FALSE) :
  cannot compute exact p-value with ties
```

Conclusion:

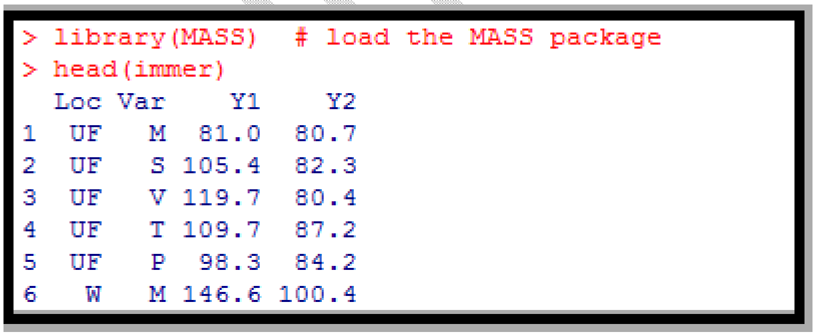
The p-value is 0.5174 is greater than 0.05, so we can accept the hypothesis H_0 of statistical equality of the means of two groups.

3.3. Wilcoxon Signed Rank

Two data samples are matched if they come from repeated observations of the same subject. Using the Wilcoxon signed- Rank test, we can decide whether the corresponding data population distributions are identical without assuming them to follow the normal distribution.

3.4. Example - Wilcoxon signed- Rank test

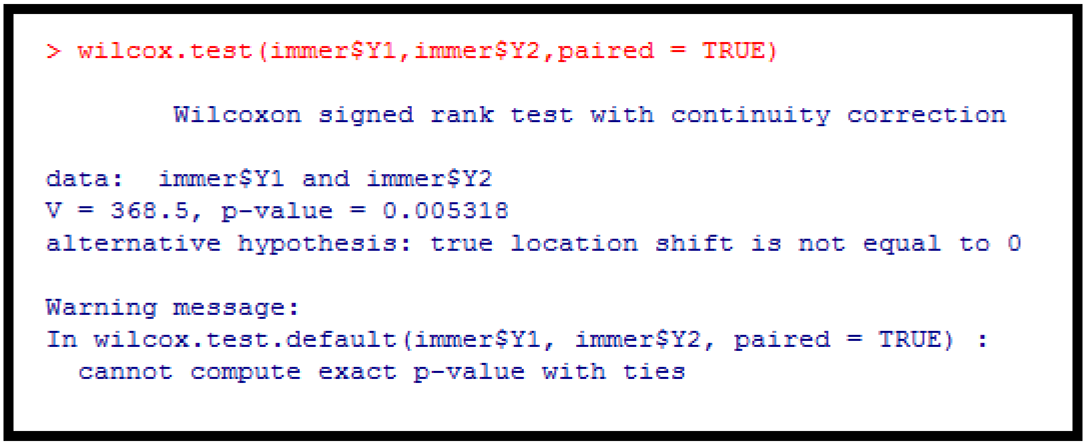
In the built-in data set named immer, the barley yield in years 1931 and 1932 of the same field are recorded. The yield data are presented in the data frame columns Y1 and Y2.



```
> library(MASS)   # load the MASS package
> head(immer)
  Loc Var    Y1    Y2
1  UF   M  81.0  80.7
2  UF   S 105.4  82.3
3  UF   V 119.7  80.4
4  UF   T 109.7  87.2
5  UF   P  98.3  84.2
6   W   M 146.6 100.4
```

Without assuming the data to have normal distribution, test at 0.05 significance level if the barley yields of 1931 and 1932 in data set immer have identical data distributions.

Solution:



```
> wilcox.test(immer$Y1,immer$Y2,paired = TRUE)

        Wilcoxon signed rank test with continuity correction

data:  immer$Y1 and immer$Y2
V = 368.5, p-value = 0.005318
alternative hypothesis: true location shift is not equal to 0

Warning message:
In wilcox.test.default(immer$Y1, immer$Y2, paired = TRUE) :
  cannot compute exact p-value with ties
```

Conclusion

We find that the P-value (0.005318) < significance level 0.05.

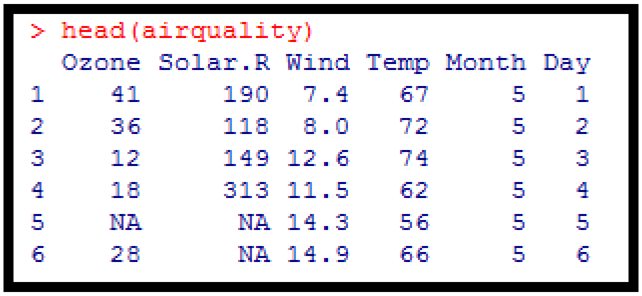
At 0.05 significance level, we conclude that the barley yields of 1931 and 1932 from the data set immer are non-identical populations.

3.5. Kruskal Wallis

When you have more than two samples to compare you would usually attempt to use analysis of variance. However, if the data are not normally distributed (i.e., not parametric) then an alternative must be sought. One such alternative is Kruskal-Wallis test, designed to test for significant differences in population medians when you have more than two samples. You can think of K-W test as a non-parametric version of one-way anova.

3.6. Example - Kruskal Wallis

In the built-in data set named **airquality**, the air quality measurements in New York, May to September 1973, are recorded. The ozone densities are presented in the data frame column ozone.

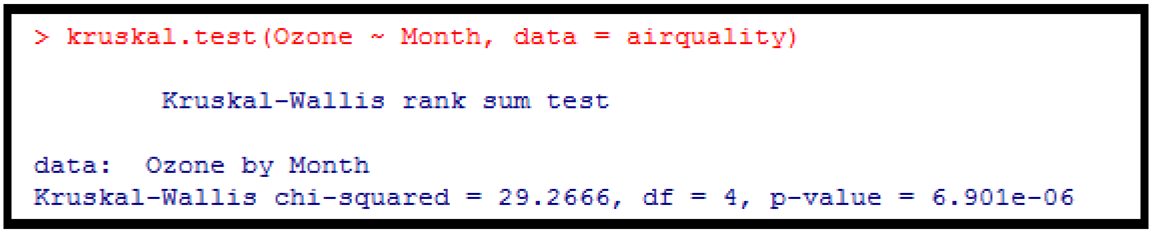


```
> head(airquality)
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6
```

Problem

Without assuming the data to have normal distribution, test at 0.05 significance level if the monthly ozone density in New York has identical data distributions from May to September 1973.

Solution:



```
> kruskal.test(Ozone ~ Month, data = airquality)

        Kruskal-Wallis rank sum test

data:  Ozone by Month
Kruskal-Wallis chi-squared = 29.2666, df = 4, p-value = 6.901e-06
```

The null hypothesis is that the monthly ozone density are identical populations. To test the hypothesis, we apply the kruskal.test function to compare the independent monthly data. The p-value turns out to be nearly zero (6.901e-06). Hence, we reject the null hypothesis.

Conclusion

At 0.05 significance level, we conclude that the monthly ozone density in New York from May to September 1973 are non-identical populations.

3.7. Friedman tests

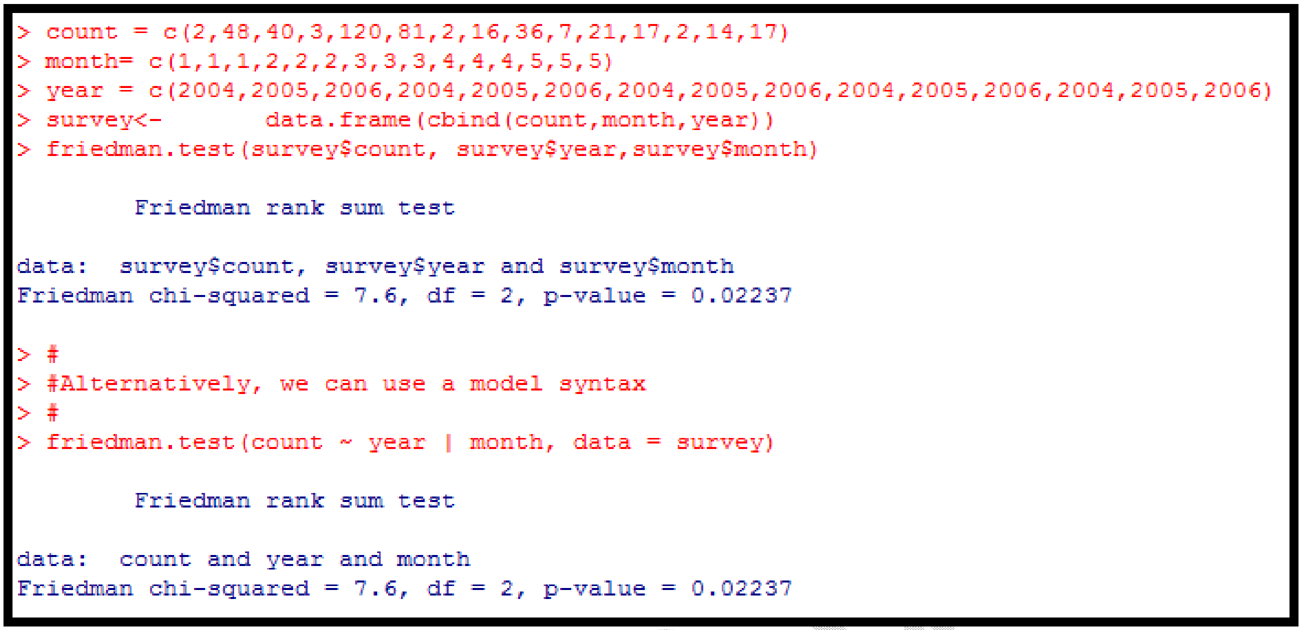
Friedman test can be used for analyzing unreplicated complete block design (i.e., there is exactly one observation in y for each combination of levels of groups and blocks) where the normality assumption may be violated.

The null hypothesis is that apart from an effect of blocks the location parameter of y is the same in each of the groups.

3.8. Example - Friedman tests

We have data on surveys of amphibians. The first column (count) represents the number of individuals captured. The final column is the year that the survey was conducted. The middle column (month) shows that for each year, there were 5 survey events in each year. What we have is a replicated block design. Each year is a treatment or group whilst the month variable represents a block. This is a common sort of experimental design; the blocks are set up to take care of any possible variation and to provide replication for the treatment. In this instance, we would like to know if there is any significant difference due to year.

Solution:



```
> count = c(2,48,40,3,120,81,2,16,36,7,21,17,2,14,17)
> month= c(1,1,1,2,2,2,3,3,3,4,4,4,5,5,5)
> year = c(2004,2005,2006,2004,2005,2006,2004,2005,2006,2004,2005,2006,2004,2005,2006)
> survey<-        data.frame(cbind(count,month,year))
> friedman.test(survey$count, survey$year,survey$month)

        Friedman rank sum test

data:  survey$count, survey$year and survey$month
Friedman chi-squared = 7.6, df = 2, p-value = 0.02237

> #
> #Alternatively, we can use a model syntax
> #
> friedman.test(count ~ year | month, data = survey)

        Friedman rank sum test

data:  count and year and month
Friedman chi-squared = 7.6, df = 2, p-value = 0.02237
```

Conclusion

The p-value turns out to be 0.02237 and it is < 0.05 level of significance. Hence, we reject the null hypothesis that there is no significant difference in amphibians due to year.

At 0.05 significance level, we conclude that the there is significant difference in amphibians due to year.

.

4. Introduction to Chi-Squared Test

Let the probabilities of various classes in a distribution be p_1, $p_{2, \ldots}$ p_k with observed frequencies m_1, $m_{2, \ldots}$ m_k.

$$\chi_s^2 = \sum_{i=1}^{k} \frac{(m_i - N\,p_i)^2}{N\,p_i}$$

This quantity is therefore a measure of the deviation of a sample from expectation, where N is the sample size.

Karl Pearson proved that the limiting distribution of χ_s^2 is a chi-squared distribution.

The syntax of the chisq.test () function is given below:

chisq.test(x, y = NULL, correct = TRUE, p= rep(1/length(x), length(x)), rescale.p = FALSE, simulate.p.value = FALSE, B= 2000)

This function is used for both the goodness of fit and the test of independence, and which test it does depends upon what kind of data you feed it.

If "x" is a numerical vector or a one-dimensional table of numerical values, a goodness of fit test will be done (or attempted), treating "x" as a vector of observed frequencies. If "x" is a 2-D table, array, or matrix, then it is assumed to be a contingency table of frequencies, and a test of independence will be done.

The correct = TRUE option applies the Yates continuity correction when "x" is a 2 X 2 table. Set this to FALSE, if the correction is not desired.

For the goodness of fit test, set "p" equal to the null hypothesized proportions or probabilities for each of the categories represented in the vector "x".

5. Chi-squared test for independence

A Chi-square test is a common test for nominal (categorical) data. One application of a Chi-Square test i a test for independence. In this case, the null hypothesis is that the occurrence of the outcomes for the two groups is equal.

Chi-square test can be used to determine whether observed frequencies are significantly different from expected frequencies.

Chi-square tests enable us to compare observed and expected frequencies objectively, since it is not always possible to tell just by looking at them whether they are “different enough” to be considered statistically significant. Statistical significance in this case implies that the differences are not due to chances alone, but instead may be indicative of other processes at work.

5.1. Example – Pearson's Chi-squared test

For example, you have three groups based on age (under 45, between 45 and 59 and over 60) and you have nominal data for each group - the frequency of regular health check up (yearly, occasionally, never). You are interested in figuring out whether the outcomes for the two groups were statistically equal. We can tally the frequency of medical check ups with the age group in the following table, known as **contingency table** of the two variables.

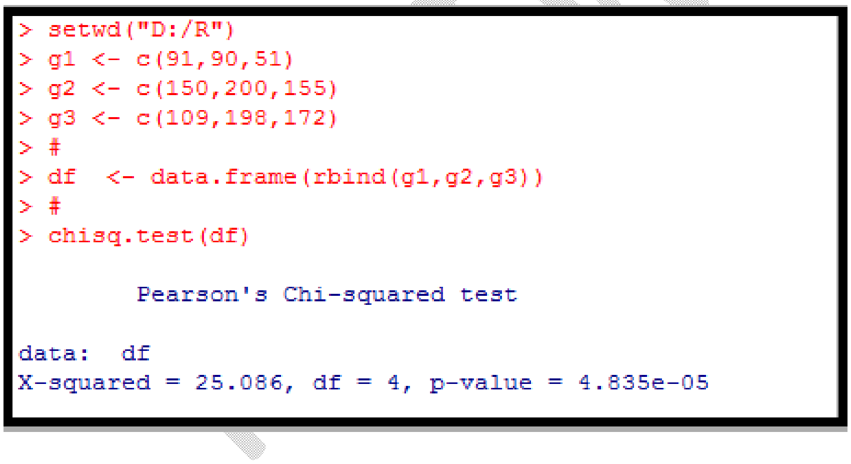
Age	Frequency of regular medical check ups		
	Yearly	Occasionally	Never
Under 45	91	90	51
45 – 59	150	200	155
60 and over	109	198	172

Test the hypothesis whether the frequency of the medical check-up is independent of the age group at 0.05 significance level.

H_0: Medical check-up is independent of the age group at 0.05 significance level
H_1: Medical check-up is dependent of the age group at 0.05 significance level

Solution:

We apply the chisq.test function to the contingency table (data frame,df) and found the p-value to be 4.835e-05.



```
> setwd("D:/R")
> g1 <- c(91,90,51)
> g2 <- c(150,200,155)
> g3 <- c(109,198,172)
> #
> df  <- data.frame(rbind(g1,g2,g3))
> #
> chisq.test(df)

        Pearson's Chi-squared test

data:  df
X-squared = 25.086, df = 4, p-value = 4.835e-05
```

As the p-value 0.00004835 is very much smaller than 0.05 significance level, we reject the null hypothesis that frequency of medical check-up is independent of the age group at 0.05 significance level.

Conclusion:

Frequency of medical check-up is dependent of the age group at 0.05 significance level.

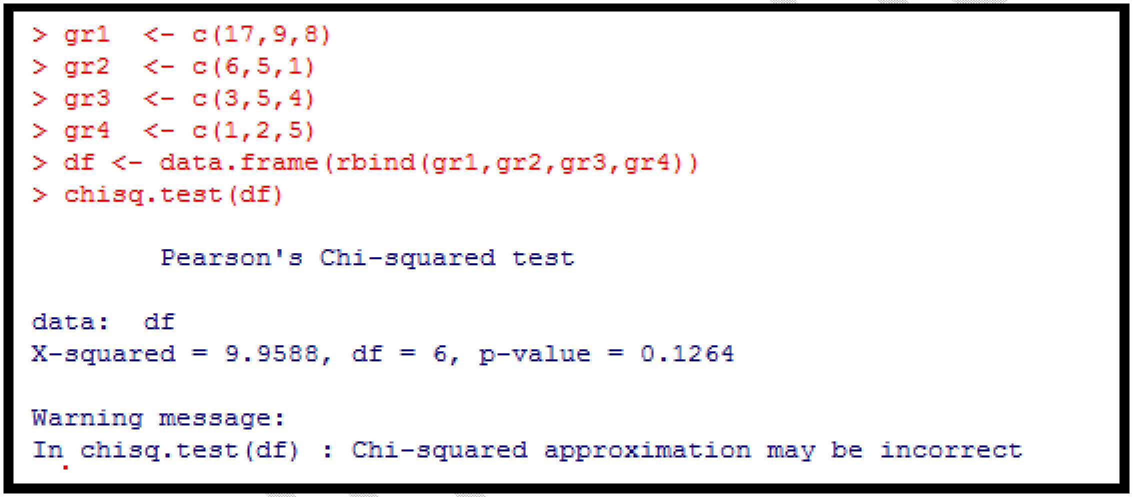
5.2. Example – Pearson's Chi-squared test

The following data describe the state of grief of 66 mothers who had suffered a neonatal death. This table relates this to the amount of support given to these women:

Grief State	Support		
	Good	Adequate	Poor
I	17	9	8
II	6	5	1
III	3	5	4
IV	1	2	5

Did the supporting these mothers help lessen their burden of grief?

Solution:



```
> gr1  <- c(17,9,8)
> gr2  <- c(6,5,1)
> gr3  <- c(3,5,4)
> gr4  <- c(1,2,5)
> df <- data.frame(rbind(gr1,gr2,gr3,gr4))
> chisq.test(df)

        Pearson's Chi-squared test

data:  df
X-squared = 9.9588, df = 6, p-value = 0.1264

Warning message:
In chisq.test(df) : Chi-squared approximation may be incorrect
```

As the p-value 0.1264 is greater than 0.05 significance level, we accept the null hypothesis that the grief state of the mothers is independent of the support they have received at 0.05 significance level.

Conclusion:

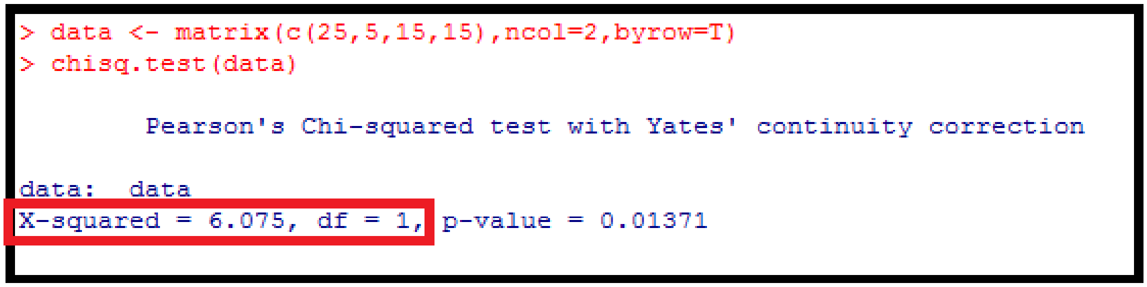
There is no significant reduction in grief because of the support, received by these mothers at 0.05 significance level.

5.3. Example – Yates' continuity correction

You have two user groups (Male, Female). You have nominal data for each group, for example, whether they use mobile devices or which OS they use.

	Own Device A	Don't own device A
Male	25	5
Female	15	15

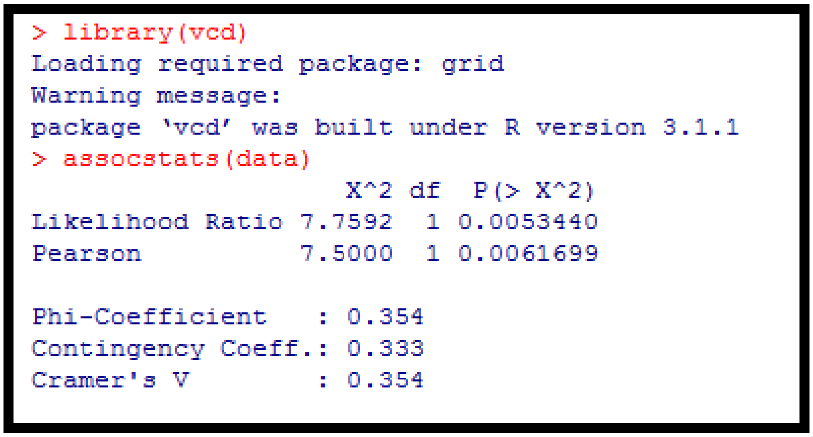
Solution:



```
> data <- matrix(c(25,5,15,15),ncol=2,byrow=T)
> chisq.test(data)

        Pearson's Chi-squared test with Yates' continuity correction

data:  data
X-squared = 6.075, df = 1, p-value = 0.01371
```

This example has a significant difference (p-value = 0.01371 > 0.05 significance level), which means the ownership of device A significantly differs between male and female users.

The effect size of the first test can be calculated with vcd package.



```
> library(vcd)
Loading required package: grid
Warning message:
package 'vcd' was built under R version 3.1.1
> assocstats(data)
                    X^2 df  P(> X^2)
Likelihood Ratio 7.7592  1 0.0053440
Pearson          7.5000  1 0.0061699

Phi-Coefficient   : 0.354
Contingency Coeff.: 0.333
Cramer's V        : 0.354
```

For a 2 X 2 table, you can also calculate the odds ratio, which gives the probability of the phenomena is affected by the dependent variable. This can be calculated as ad / bc. = (25 *15) / (5 * 15) = 5

	Own Device A	Don't own device A
Male	a = 25	b = 5
Female	c = 15	d = 15

Conclusion: Our Chi-Square test with Yates' continuity correction revealed that the percentage of the ownership of device A significantly differed by gender (χ^2 (1, N = 60) = 6.08, p < 0.05, ϕ = 0.35, the odds ratio is 5.0)

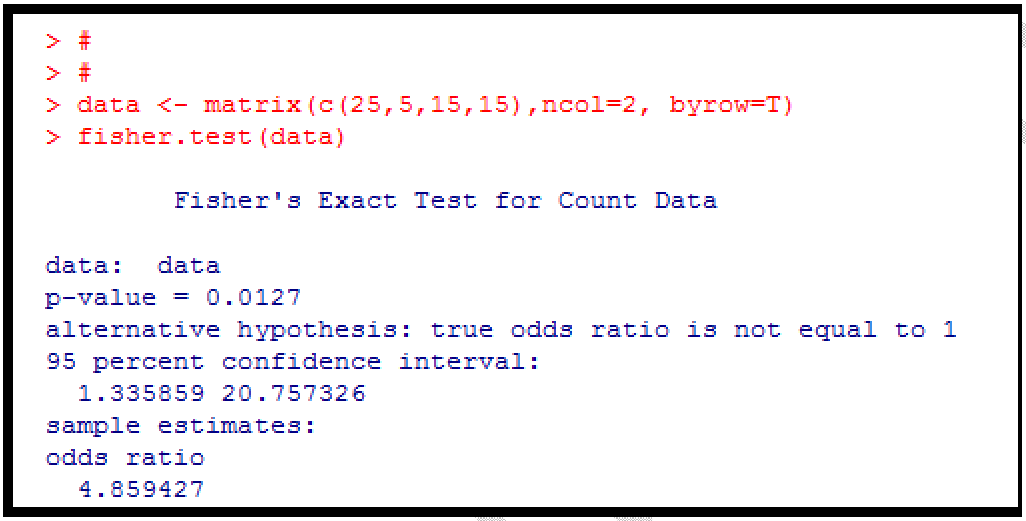
5.4. Example – Fisher’s exact test

You have two user groups (Male, Female). You have nominal data for each group, for example, whether they use mobile devices or which OS they use.

	Own Device A	Don’t own device A
Male	25	5
Female	15	15

One can use Fisher’s exact test if your sample size is small. In general, it is better to use a Fisher’s exact test than a Chi-square test when you have small than 10 in any cell of your data table (like the example, above).

Solution:



```
> #
> #
> data <- matrix(c(25,5,15,15),ncol=2, byrow=T)
> fisher.test(data)

        Fisher's Exact Test for Count Data

data:  data
p-value = 0.0127
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
  1.335859 20.757326
sample estimates:
odds ratio
  4.859427
```

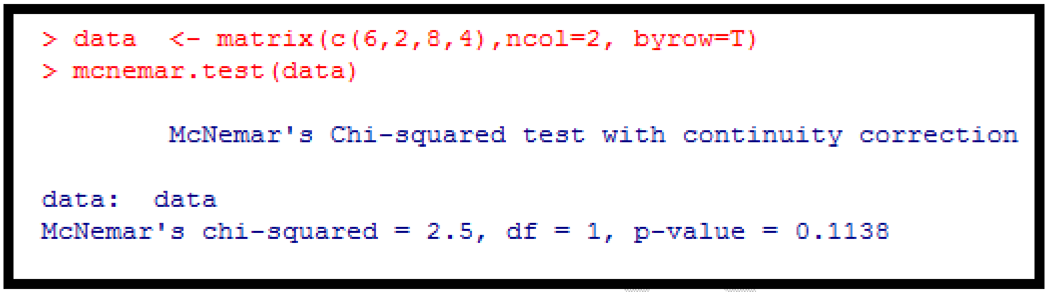
Conclusion: Our Fisher’s exact test revealed that the percentage of the ownership of device A significantly differed by gender (p = 0.0127 < 0.05, the odds ratio is 4.859427 and 95 % confidence interval for the mean extends from 1.34 to 20.76).

5.5. Example McNemar's Test

This test is basically a paired version of the Chi-squared test. If you would like to test whether the participants liked the device before and after the experiment.

	After Experiment	
Before Experiment	**Yes**	**No**
Yes	**6**	**2**
No	**8**	**4**

Solution:



```
> data  <- matrix(c(6,2,8,4),ncol=2, byrow=T)
> mcnemar.test(data)

        McNemar's Chi-squared test with continuity correction

data:  data
McNemar's chi-squared = 2.5, df = 1, p-value = 0.1138
```

Conclusion: Since p = 0.1138 > significance level 0.05, we accept the null hypothesis. It means that the number of the participants who liked the device were not significantly changed between before and after the experiment.

6. Chi-squared test for goodness of fit

Multinomial Goodness of Fit

A population is called multinomial if its data is categorical and belongs to a collection of discrete non-overlapping classes.

The null hypothesis for goodness of fit test for multinomial distribution is that the observed frequency f_i is equal to an expected count e_i in each category. It is to be rejected of the p-value of the following Chi-Squared test statistics is less than a given significance level α.

6.1. Example Multinomial Goodness of Fit

In the following example, a survey is conducted to study the student's smoking habits. There are four proper responses in the survey: "Heavy","Regularly","Occasionally" and "Never". The smoking data is multinominal. (Data taken from the built-in data set survey available in the library MASS)

The frequency distribution is given below for the smoking data.

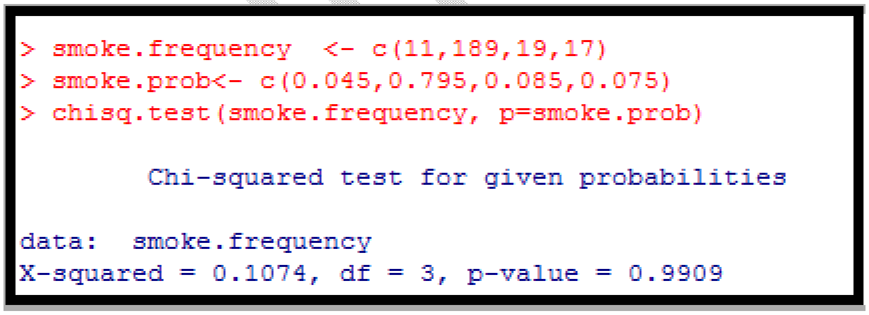
Heavy	Never	Occasionally	Regularly
11	189	19	17

As per the campus smoking statistics, we have

Heavy	Never	Occasionally	Regularly
4.5 %	79.5 %	8.5 %	7.5 %

Determine whether the sample data in survey supports it at 0.05 significance level.

Solution:



```
> smoke.frequency  <- c(11,189,19,17)
> smoke.prob<- c(0.045,0.795,0.085,0.075)
> chisq.test(smoke.frequency, p=smoke.prob)

        Chi-squared test for given probabilities

data:  smoke.frequency
X-squared = 0.1074, df = 3, p-value = 0.9909
```

Conclusion: As the p-value 0.9909 is greater than the 0.05 significance level, we do not reject the null hypothesis that the sample data in survey supports the campus-wide smoking statistics.

7. Introduction to ANOVA

(Ref: Explorable.com (Jun 6, 2009). ANOVA. Retrieved Sep 19, 2014 from Explorable.com: https://explorable.com/anova)

The Analysis of Variance, popularly known as the ANOVA, can be used in cases where there are more than two groups.

When we have only two samples we can use the t-test to compare the means of the samples but it might become unreliable in case of more than two samples. If we only compare two means, then the t-test (independent samples) will give the same results as the ANOVA.

7.1. One-Way ANOVA

Example: Suppose we want to test the effect of five different exercises. For this, we recruit 10 men and assign one type of exercise to 4-men (5 groups). Their weights are recorded after a few weeks. We may find out whether the effect of these exercises on them is significantly different or not and this may be done by comparing the weights of the 5 groups of 4 men each.

The above example is a case of one-way balanced ANOVA.

It has been termed as one-way as there is only one category whose effect has been studied and balanced as the same number of men has been assigned on each exercise. Thus the basic idea is to test whether the samples are all alike or not.

7.2. One Way and Two way ANOVA

Now some questions may arise as to what are the means we are talking about and why variances are analyzed in order to derive conclusions about means. The whole procedure can be made clear with the help of an experiment.

Let us study the effect of fertilizers on yield of wheat. We apply five fertilizers, each of different quality, on five plots of land each of wheat. The yield from each plot of land is recorded and the difference in yield among the plots is observed. Here, fertilizer is a factor and the different qualities of fertilizers are called levels.

This is a case of one-way or one-factor ANOVA since there is only one factor, fertilizer. We may also be interested to study the effect of fertility of the plots of land. In such a case we would have two factors, fertilizer and fertility. This would be a case of two-way or two-factor ANOVA. Similarly, a third factor may be incorporated to have a case of three-way or three-factor ANOVA.

Chance Cause and Assignable Cause

In the above experiment the yields obtained from the plots may be different and we may be tempted to conclude that the differences exist due to the differences in quality of the fertilizers.

But this difference may also be the result of certain other factors which are attributed to chance and which are beyond human control. This factor is termed as "error". Thus, the differences or variations that exist within a plot of land may be attributed to error.

Thus, estimates of the amount of variation due to assignable causes (or variance between the samples) as well as due to chance causes (or variance within the samples) are obtained separately and compared using an F-test and conclusions are drawn using the value of F.

Four basic Assumptions of ANOVA

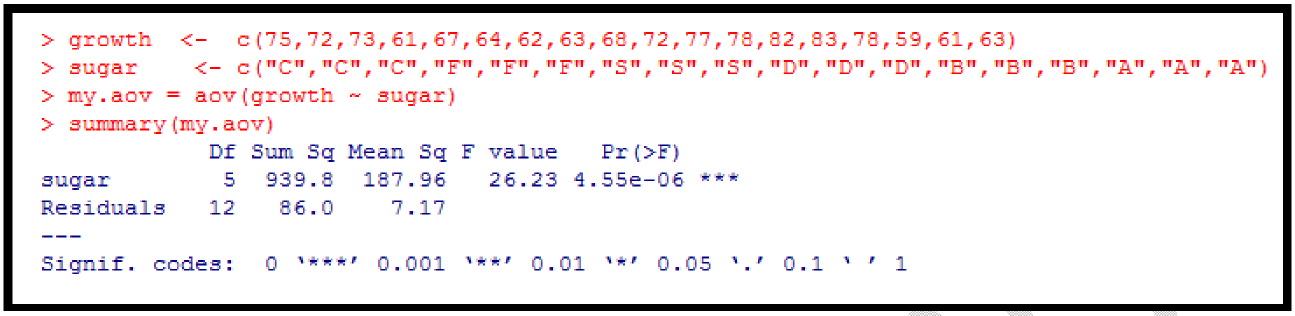
1. the expected values of the errors are zero
2. the variances of all errors are equal to each other
3. the errors are independent
4. they are normally distributed

7.3. Example - ANOVA One-way

The simplest ANOVA would be where we have a single dependent variable and one single factor. We may have raised broods of flies on various flies on various sugars. We measure the size of the individual flies and record the diet for each. Our data file would consist of two columns; one for growth and one for sugar. e.g.

growth	sugar
75	C
72	C
73	C
61	F
67	F
64	F
62	S
63	S
68	S
72	D
77	D
78	D
82	B
83	B
78	B
59	A
61	A
63	A

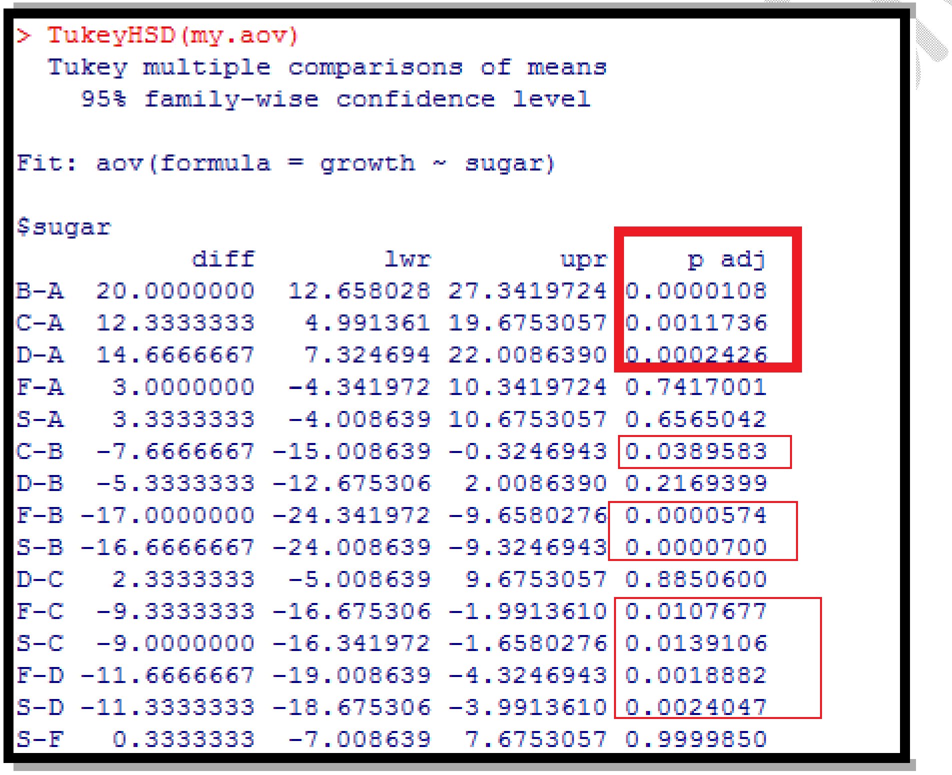
In this case we have a column for the dependent variable (growth) and a column for the dependent factor (sugar). The first column contains numeric data but the second contains letters.



```
> growth  <-  c(75,72,73,61,67,64,62,63,68,72,77,78,82,83,78,59,61,63)
> sugar    <- c("C","C","C","F","F","F","S","S","S","D","D","D","B","B","B","A","A","A")
> my.aov = aov(growth ~ sugar)
> summary(my.aov)
            Df Sum Sq Mean Sq F value   Pr(>F)    
sugar        5  939.8  187.96   26.23 4.55e-06 ***
Residuals   12   86.0    7.17                     
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Since p-value < 0.05 (significance level), we see that there is a significant effect of diet upon growth.

We need to conduct Post-hoc testing to know which of these treatments are significantly different from the controls and from other treatments.



```
> TukeyHSD(my.aov)
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = growth ~ sugar)

$sugar
           diff        lwr        upr     p adj
B-A  20.0000000  12.658028 27.3419724 0.0000108
C-A  12.3333333   4.991361 19.6753057 0.0011736
D-A  14.6666667   7.324694 22.0086390 0.0002426
F-A   3.0000000  -4.341972 10.3419724 0.7417001
S-A   3.3333333  -4.008639 10.6753057 0.6565042
C-B  -7.6666667 -15.008639 -0.3246943 0.0389583
D-B  -5.3333333 -12.675306  2.0086390 0.2169399
F-B -17.0000000 -24.341972 -9.6580276 0.0000574
S-B -16.6666667 -24.008639 -9.3246943 0.0000700
D-C   2.3333333  -5.008639  9.6753057 0.8850600
F-C  -9.3333333 -16.675306 -1.9913610 0.0107677
S-C  -9.0000000 -16.341972 -1.6580276 0.0139106
F-D -11.6666667 -19.008639 -4.3246943 0.0018882
S-D -11.3333333 -18.675306 -3.9913610 0.0024047
S-F   0.3333333  -7.008639  7.6753057 0.9999850
```

The following treatments are significantly different from the controls and other treatments as shown by rectangles drawn in red:

B – A, **C- A**, **D- A**, **C- B**, **F- B, S- B**, **F- C**, **S – C, F – D** and **S - D**

8. Monte Carlo simulation

- Statistical inference is the procedure of drawing conclusions about a population or process based on a sample. Characteristics of a population are known as parameters. Monte Carlo methods are widely used and increasingly important in statistical inference, both Frequentist and Bayesian.
- Frequentist inference is one of a number of possible techniques of formulating generally applicable schemes for making statistical inference. That implies of drawing conclusions from sample data by the emphasis on the frequency or proportion of the data. This is the inference framework in which the well-established methodologies of statistical hypothesis testing and confidence intervals are based.
- Bayesian inference is one of the two dominant approaches to statistical inference. Bayesian inference is a collection of statistical methods which are based on Bayes' formula. The distinctive aspect of Bayesian inference is that both parameters and sample data are treated as random quantities, while other approaches regard the parameters non-random.
- Refer to the book: Introducing Monte Carlo Methods with R
  Christian Robert, George Casella
  Springer Science & Business Media, 07-Dec-2009 - Computers - 284 pages

8.1. Monte Carlo simulation

- Monte Carlo simulation is a statistical approach which is concerned with experiments employing random numbers. The technique is used by professionals with applications in a variety of fields including Operations Research, Physics, Finance, Chemistry, Biology and Medicine.
- Monte Carlo methods are used to handle both probabilistic and deterministic problems according to whether or not they are directly concerned with the behaviour and outcome of a random process. In the case of a probabilistic problem a simple Monte Carlo approach is to observe random numbers, chosen in such a way that they directly simulate the physical random process of the original problem, and to infer the desired solution from the behaviour of these random numbers.
- Monte Carlo simulation has wide application in performing risk analysis by building models of possible results by substituting a range of values (a probability distribution) for any factor that has inherent uncertainty. It then calculates results over and over, each time using a different set of random values from the probability functions. Depending on the number of uncertainties and the ranges specified for them, a Monte Carlo simulation produces distributions of possible outcome values. By using probability distributions, variables can have different probabilities of different outcomes occurring. Probability distributions are a realistic way of describing uncertainty in variables of a risk analysis.

8.2. Steps used in Monte Carlo methods

1. *Define some domain of inputs. This just means we have some set of variables and what values they can take on, or we have some observations that are part of a dataset.*
2. *Generate inputs (the values of the variables or sets of observations) randomly, governed by some probability distribution.*
3. *Perform some computation on these inputs.*

4 *Repeat 2 and 3 over and over either an infinite number of times (a very large number of times usually >= 10000), or until convergence.*

5 *Aggregate the results from the previous step into some final computation.*

8.3. Business Planning sample

- There is an interesting article in the website “Frontline Solvers – Developers of the Excel Solver” (http://www.solver.com/monte-carlo-simulation-example) about a Business Planning sample using Monte Carlo Simulation.

- You, as a marketing manager of a firm are planning to introduce a new product. You need to estimate the first year profit from this product, which will depend on:
  - ○ Sales volume in units
  - ○ Price per unit
  - ○ Unit Cost
  - ○ Fixed Cost

- Net profit will be calculated as Net Profit = Sales Volume * (Selling Price – Unit Cost) – Fixed Cost.

- Fixed costs (for overhead, advertising, etc.) are known to be $120,000. But other factors all involve some uncertainty. Sales volume (in units) can over quite a range, and the selling price per unit will depend on competitor actions. Unit costs will also vary depending on vendor prices and production experience.

Uncertain variables

- To build a risk analysis model, you must identify the uncertain variables – also called random variables. While there’s some uncertainty in almost all variables in a business model, we want to focus on variables where the range of values is significant.

Sales and Price

- Based on the market research, you believe that there are equal chances that the market will be Slow, OK or Hot.

  - In the “Slow market” scenario, you expect to sell 50,000 units at an average selling price of $11 per unit.
  - In the “OK market” scenario, you expect to sell 75,000 units at an average selling price of $10 per unit.
  - In the “Hot market” scenario, you expect to sell 100,000 units at an average selling price of $8 per unit. In this scenario, your competitors will push your price down.

Unit Cost

- Another uncertain variable is unit cost. Your firm's production manager advises you that unit costs may vary anywhere between $5.50 to $7.50, with a most likely cost of $6.50. In this case, the most likely cost is also the average cost.

Uncertain Functions

Net Profit

- We now identify uncertain functions – also called functions of a random variable. Since Sales volume and Selling Price and Unit cost are all uncertain variables, the net profit calculated based on these variables is an uncertain function.

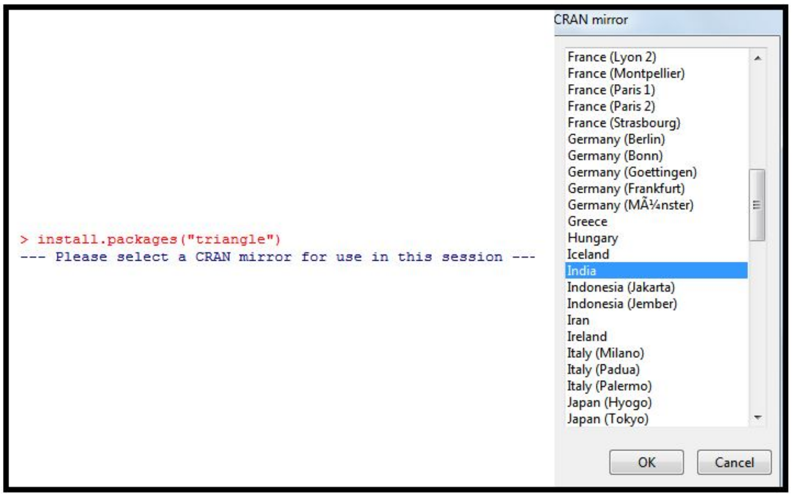
8.4. Solution using R

- Since there are equal chances that the market will be Slow, OK, or Hot, we want to create an uncertain variable that selects among these three possibilities by generating a random number – say 1 or 2 or 3 – with equal probability. We associate 1 with "Slow Market" state, 2 with "OK market" state and 3 with "Hot market" state. We generate this number easily by using the R function – "sample" and then base the Sales Volume and Selling Price of this uncertain variable.

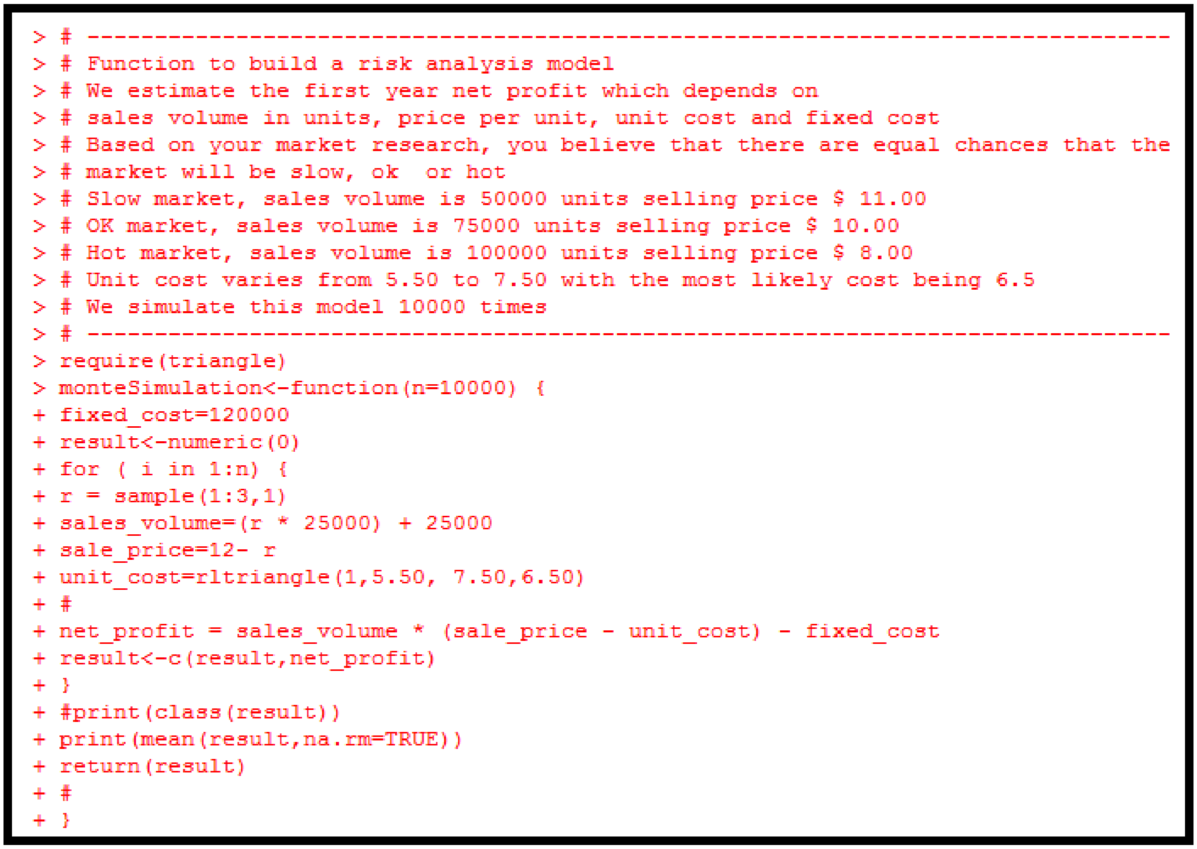
- Unit cost may vary anywhere from $5.50 to $7.50, with a most likely cost of $6.50. We use triangular distribution to generate this variable.

- *In probability theory and statistics, the triangular distribution is a continuous probability distribution with a lower limit a, upper limit b and mode c, where a < b and a ≤ c ≤ b.*

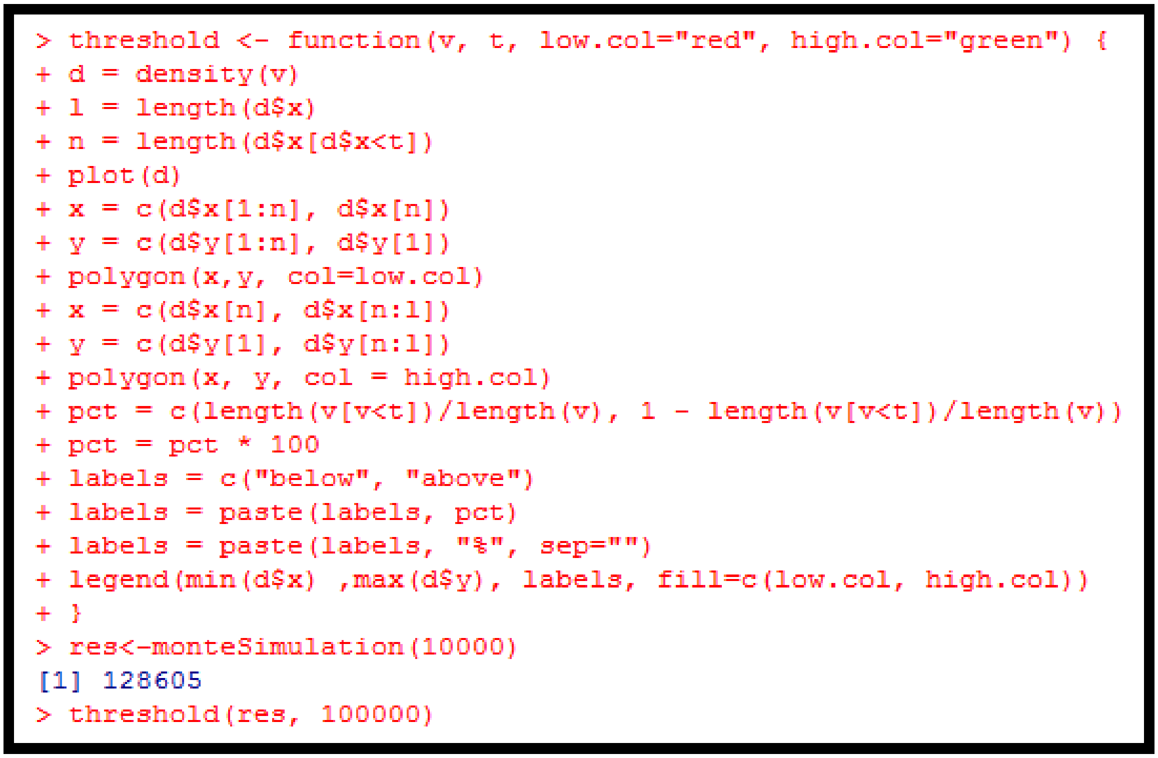
- We install "triangle" package first before using the R to solve this problem.



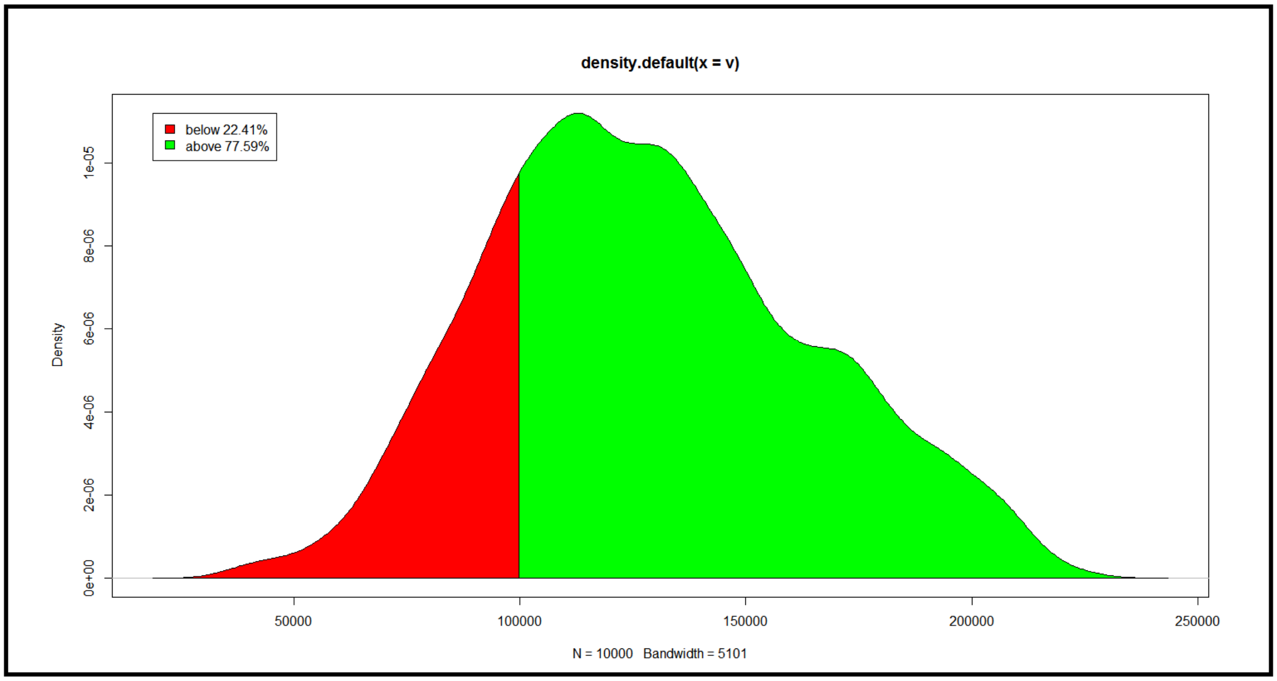
```
> install.packages("triangle")
--- Please select a CRAN mirror for use in this session ---
```



```
> # --------------------------------------------------------------------------------
> # Function to build a risk analysis model
> # We estimate the first year net profit which depends on
> # sales volume in units, price per unit, unit cost and fixed cost
> # Based on your market research, you believe that there are equal chances that the
> # market will be slow, ok  or hot
> # Slow market, sales volume is 50000 units selling price $ 11.00
> # OK market, sales volume is 75000 units selling price $ 10.00
> # Hot market, sales volume is 100000 units selling price $ 8.00
> # Unit cost varies from 5.50 to 7.50 with the most likely cost being 6.5
> # We simulate this model 10000 times
> # --------------------------------------------------------------------------------
> require(triangle)
> monteSimulation<-function(n=10000) {
+ fixed_cost=120000
+ result<-numeric(0)
+ for ( i in 1:n) {
+ r = sample(1:3,1)
+ sales_volume=(r * 25000) + 25000
+ sale_price=12- r
+ unit_cost=rltriangle(1,5.50, 7.50,6.50)
+ #
+ net_profit = sales_volume * (sale_price - unit_cost) - fixed_cost
+ result<-c(result,net_profit)
+ }
+ #print(class(result))
+ print(mean(result,na.rm=TRUE))
+ return(result)
+ #
+ }
```



```
> threshold <- function(v, t, low.col="red", high.col="green") {
+ d = density(v)
+ l = length(d$x)
+ n = length(d$x[d$x<t])
+ plot(d)
+ x = c(d$x[1:n], d$x[n])
+ y = c(d$y[1:n], d$y[1])
+ polygon(x,y, col=low.col)
+ x = c(d$x[n], d$x[n:l])
+ y = c(d$y[1], d$y[n:l])
+ polygon(x, y, col = high.col)
+ pct = c(length(v[v<t])/length(v), 1 - length(v[v<t])/length(v))
+ pct = pct * 100
+ labels = c("below", "above")
+ labels = paste(labels, pct)
+ labels = paste(labels, "%", sep="")
+ legend(min(d$x) ,max(d$y), labels, fill=c(low.col, high.col))
+ }
> res<-monteSimulation(10000)
[1] 128605
> threshold(res, 100000)
```



We have observed that the average net profit after 10000 trials is $128,605. From this graph, we see that 77.59% of the trials resulted in the net profit above $100,00 and 22.41% of the trials resulted in the net profit below $100,000.

9. Correlation

Correlation is a method of studying the relationship between two variables.

In statistical analysis, we come across the study of two variables wherein the change in the value of one variable produces a change in the value of another variable. In that case, we say that the variables are correlated or there is a correlation between the two variables.

Two variables may have a positive correlation, a negative correlation or they may be uncorrelated.

1. Positive Correlation

Two variables are said to be positively correlated if for an increase in the value of one variable there is also an increase in the value of the other variable or for the decrease in the value of one variable there is also a decrease in the value of the other variable; that is the two variables change in the same direction.

For example, the dividend and the premium of the share are positively correlated since as the premium of the share increases, the dividend also increases.

2. Negative correlation

Two variables are said to be negatively correlated if for an increase in the value of one variable there is a decrease in the value of the other variable; that is the two variables change in opposite direction.

For example, when the price increases the demand for the commodity decreases and when the price decreases the demand increases.

3. No correlation

Two variables are said to be uncorrelated if the change in the value of one variable has no connection with the change in the value of the other variable.

For example, we would expect zero correlation between weight of the person and colour of his / her hair.

9.1. Scatter Diagram

Let us consider a set of paired values of the variables x and y. For example, x represents the heights of persons and y their weights. Along the horizontal axis, we represent the height and along the vertical axis the weight. Plot the values (x,y) in a graph. We get a collection of dots, called as scatter diagram.

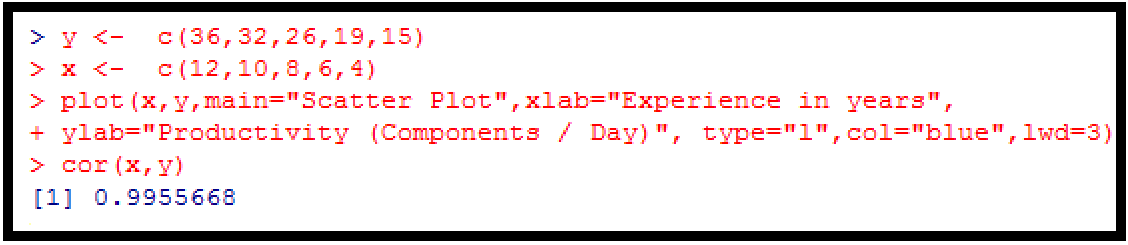
From the scatter diagram, we can obtain a rough idea of the correlation between the two variables x and y.

- If all these cluster around a line, the correlation is called a **linear correlation.**
- If the dots cluster around a curve, the correlation is called a non-linear or curve linear correlation.
- We can also get an idea of whether the correlation is positive or negative from the scatter diagram.

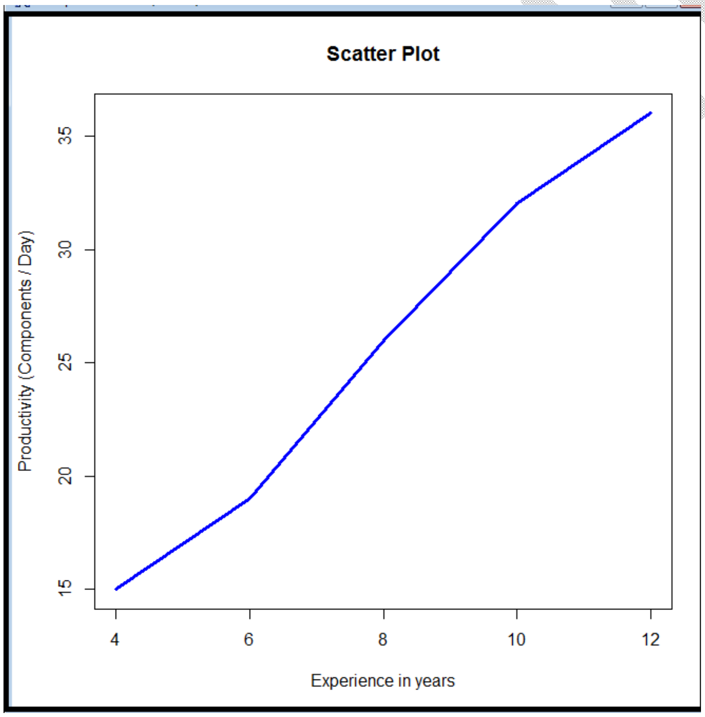
a. **Scatter plot- Positive linear correlation**

The data given below represents a sample of 5 workers in a particular factory. The productivity of each worker was measured at some point in time, and the worker's experience in years was noted. The response variable is productivity, measured in number of components per day, and the predictor variable is experience, measured in years. Using R draw a scatter plot.

Worker	y = Productivity (components / day)	x = Experience (years)
1	36	12
2	32	10
3	26	8
4	19	6
5	15	4



```
> y <-  c(36,32,26,19,15)
> x <-  c(12,10,8,6,4)
> plot(x,y,main="Scatter Plot",xlab="Experience in years",
+ ylab="Productivity (Components / Day)", type="l",col="blue",lwd=3)
> cor(x,y)
[1] 0.9955668
```

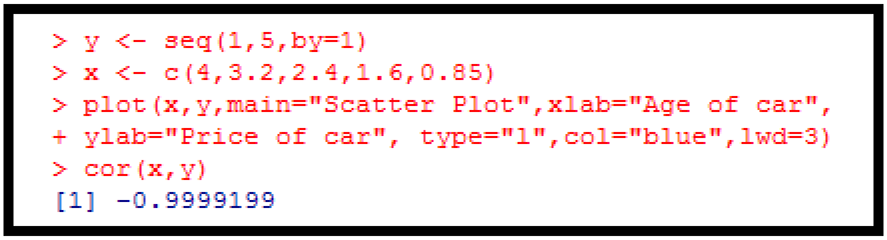


As the correlation coefficient is positive - closer to 1 (value is 0.9956), there is a positive relationship between x and y.

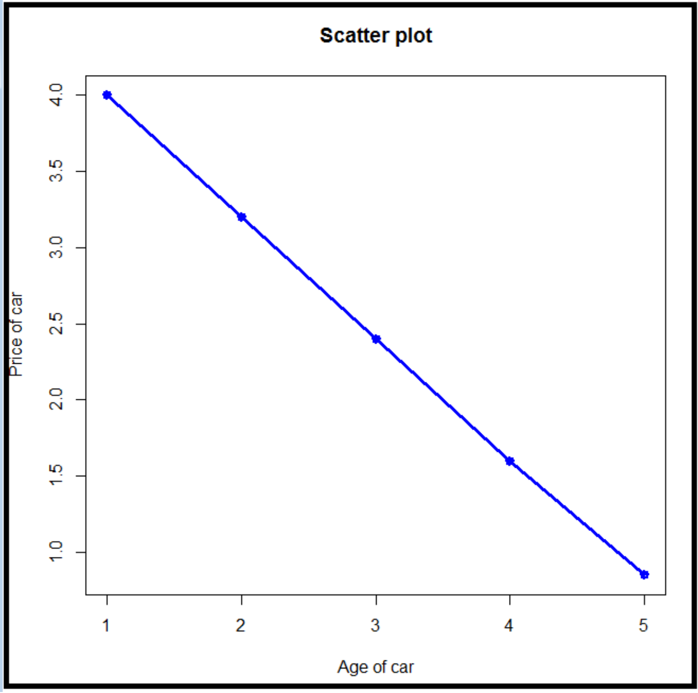
b. **Scatter plot - Negative linear correlation**

The data given below represents a sample seconds sale of 5 cars of a popular brand. The response variable is sale price, and the predictor variable is age of car, measured in years. Using R draw a scatter plot.

Car	y = Sale Price	x = Age of car (years)
1	4.0	1
2	3.2	2
3	2.4	3
4	1.6	4
5	0.85	5



```
> y <- seq(1,5,by=1)
> x <- c(4,3.2,2.4,1.6,0.85)
> plot(x,y,main="Scatter Plot",xlab="Age of car",
+ ylab="Price of car", type="l",col="blue",lwd=3)
> cor(x,y)
[1] -0.9999199
```



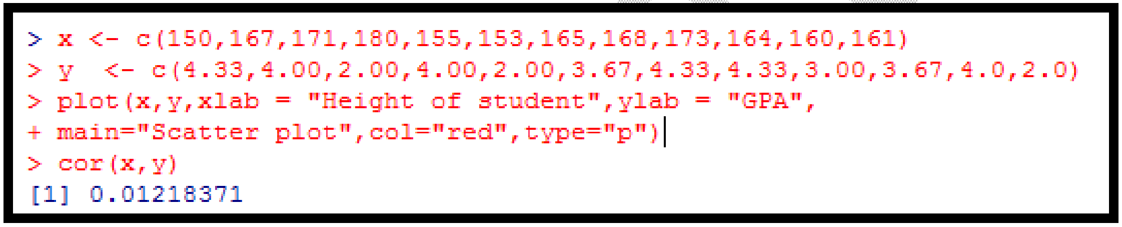
As the correlation coefficient is negative - closer to -1 (value is 0.9999), there is a negative relationship between x and y.

c. **Scatter plot – No correlation**

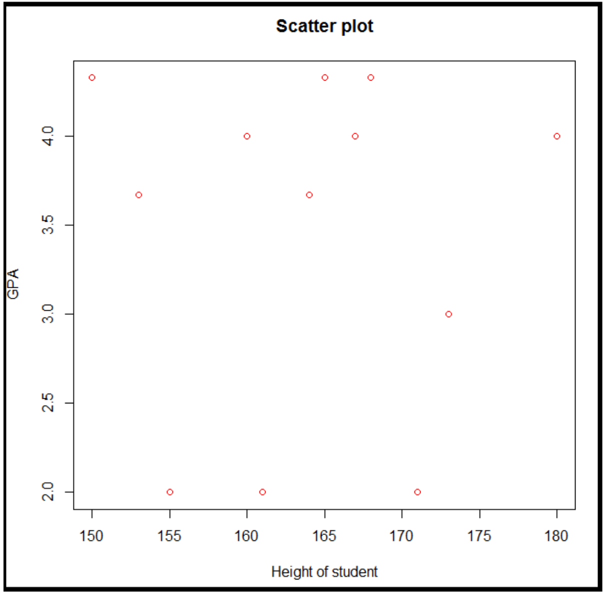
The data given below represents a sample student's GPA scores and their height. The response variable is GPA scores, and the predictor variable is height of the student, measured in cms. Using R draw a scatter plot

.

Car	y = GPA score	x = Height of student (cms)
1	4.33	150
2	4.00	167
3	2.00	171
4	4.00	180
5	2.00	155
6	3.67	153
7	4.33	165
8	4.33	168
9	3.00	173
10	3.67	164
11	4.00	160
12	2.00	161



```
> x <- c(150,167,171,180,155,153,165,168,173,164,160,161)
> y  <- c(4.33,4.00,2.00,4.00,2.00,3.67,4.33,4.33,3.00,3.67,4.0,2.0)
> plot(x,y,xlab = "Height of student",ylab = "GPA",
+ main="Scatter plot",col="red",type="p")
> cor(x,y)
[1] 0.01218371
```

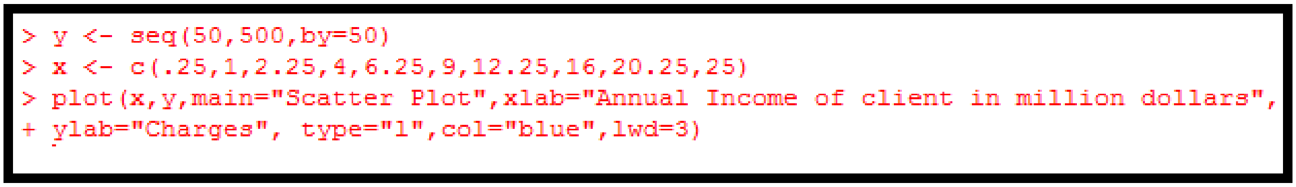


As the correlation coefficient is closer to 0 (value is 0.12), there is no relationship between x and y.

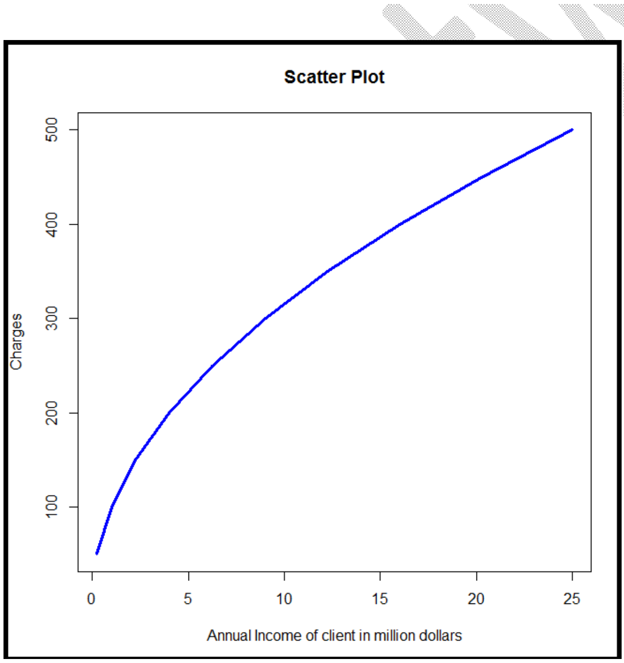
d. **Scatter plot – Curvi-linear correlation**

A tax consultant charges the following rates for his / her clients based on the annual income of the client. Using R draw a scatter plot.

Charges	Annual Income of the client (in million dollars)
50	0.25
100	1.00
150	2.25
200	4.00
250	6.25
300	9.00
350	12.25
400	16.00
450	20.25
500	25.00



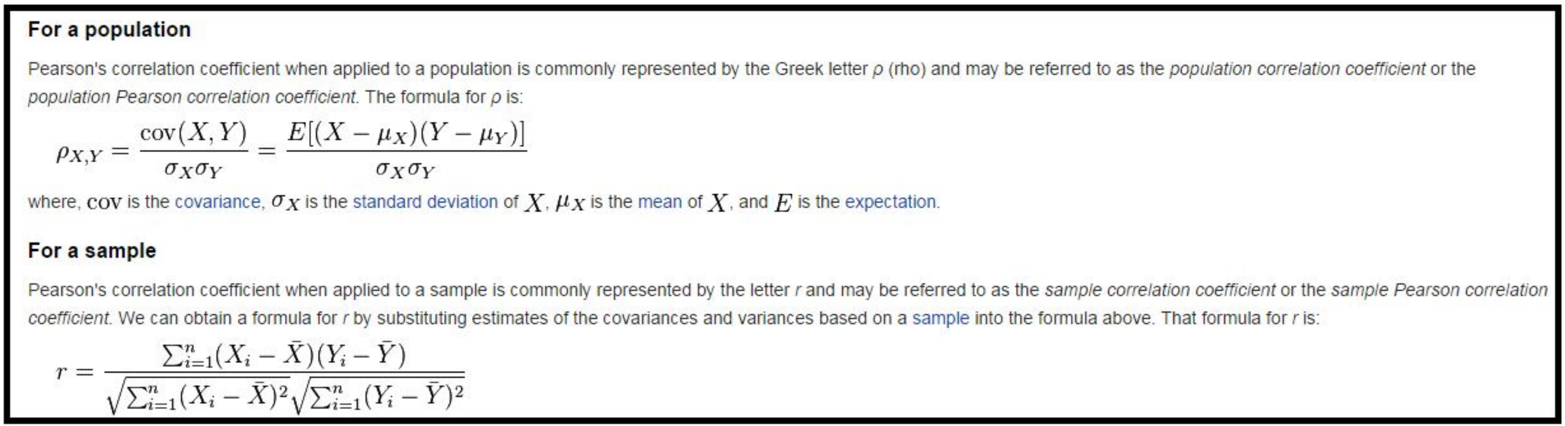
```
> y <- seq(50,500,by=50)
> x <- c(.25,1,2.25,4,6.25,9,12.25,16,20.25,25)
> plot(x,y,main="Scatter Plot",xlab="Annual Income of client in million dollars",
+ ylab="Charges", type="l",col="blue",lwd=3)
```



9.2. Karl Pearson – correlation coefficient

A Scatter diagram gives a rough idea of correlation between two variables. It gives no information about the degree of relationship between the variables.

For quantitative measurement of the degree of relationship between two variables, Karl Pearson has given the formula:



For a population

Pearson's correlation coefficient when applied to a population is commonly represented by the Greek letter ρ (rho) and may be referred to as the *population correlation coefficient* or the *population Pearson correlation coefficient*. The formula for ρ is:

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y}$$

where, cov is the covariance, σ_X is the standard deviation of X, μ_X is the mean of X, and E is the expectation.

For a sample

Pearson's correlation coefficient when applied to a sample is commonly represented by the letter *r* and may be referred to as the *sample correlation coefficient* or the *sample Pearson correlation coefficient*. We can obtain a formula for *r* by substituting estimates of the covariances and variances based on a sample into the formula above. That formula for *r* is:

$$r = \frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i-\bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i-\bar{Y})^2}}$$

It is conventionally taken

$$\mathbf{x} = (X_i - \bar{X}) \ \& \ \mathbf{y} = (Y_i - \bar{Y})$$

Hence, r = ∑xy / (√∑x^2) (√∑y^2). This formula is used when deviations are measured from their mean.

Numerical value of the correlation coefficient

The coefficient of correlation r lies between -1 and +1 inclusive of those values.

i. *When r is positive, the variables x and y increase or decrease together.*

ii. *r = +1 implies that there is a perfect positive correlation between the variables x and y (See Scatter plot)*

iii. *When r is negative, the variables x and y move in the opposite direction (i.e, as one increases, the other decreases).*

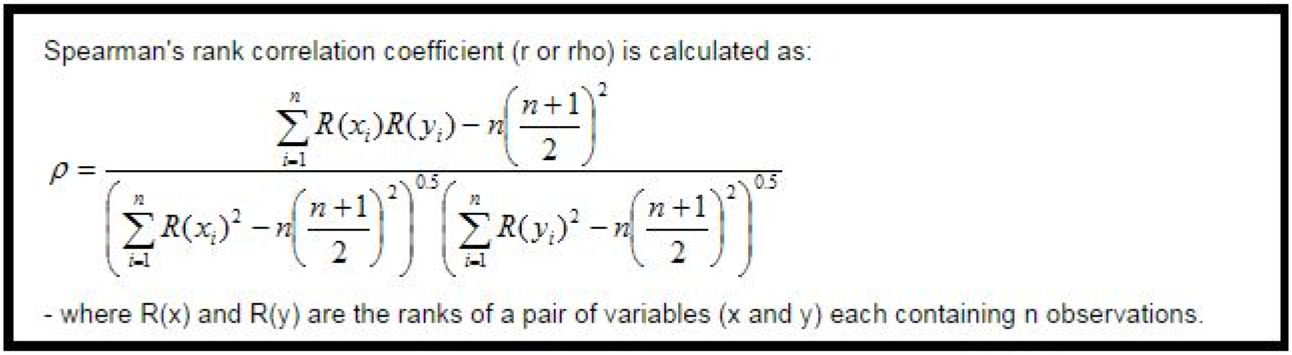
iv. *When r = -1, there is a perfect negative correlation (See scatter plot 2)*

v. *When r = 0, the two variables are uncorrelated.*

9.3. Rank Correlation

The Karl Pearson's formula for calculating r is developed on the assumption that the values of the variables are exactly measurable. In some situations, it may not be possible to give precise values for the variables. In such cases, we can use another measure of correlation coefficient called rank correlation coefficient.

We rank observations in ascending or descending order using the numbers 1,2,3,4,..n and measure the degree of relationship between the ranks instead of actual numerical values. The rank correlation coefficient when there are n ranks in each variable is given by the formula (due to Spearman).

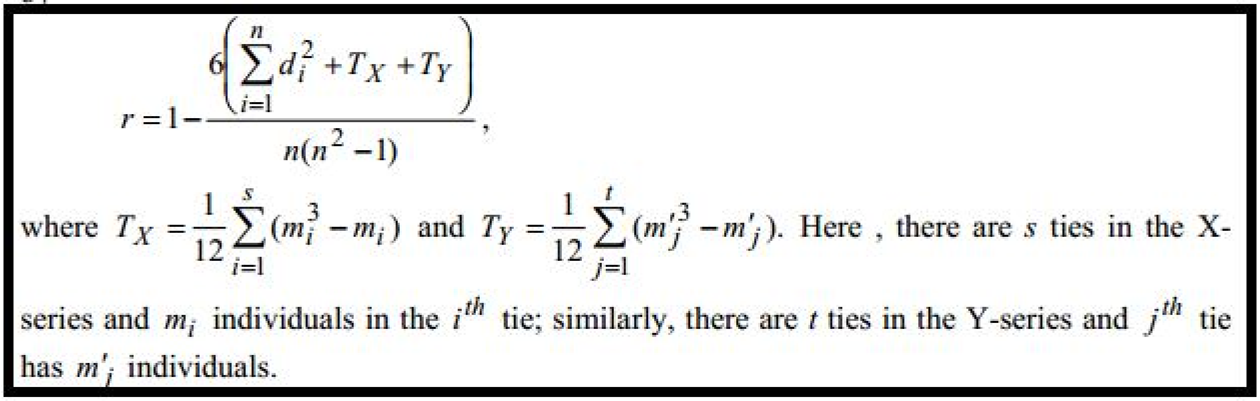


Spearman's rank correlation coefficient (r or rho) is calculated as:

$$\rho = \frac{\sum_{i=1}^{n} R(x_i)R(y_i) - n\left(\frac{n+1}{2}\right)^2}{\left(\sum_{i=1}^{n} R(x_i)^2 - n\left(\frac{n+1}{2}\right)^2\right)^{0.5}\left(\sum_{i=1}^{n} R(y_i)^2 - n\left(\frac{n+1}{2}\right)^2\right)^{0.5}}$$

- where R(x) and R(y) are the ranks of a pair of variables (x and y) each containing n observations.

In case of a tie in the ranks, then the ranks assigned is the average of the ranks assigned to these individuals had there been no tie. In case of ties, the rank correlation coefficient, r is given by the formula:



$$r = 1 - \frac{6\left(\sum_{i=1}^{n} d_i^2 + T_X + T_Y\right)}{n(n^2-1)},$$

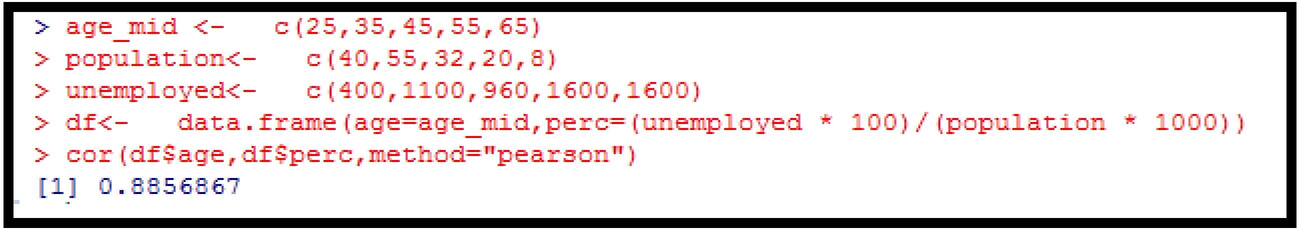
where $T_X = \frac{1}{12}\sum_{i=1}^{s}(m_i^3 - m_i)$ and $T_Y = \frac{1}{12}\sum_{j=1}^{t}(m'^3_j - m'_j)$. Here , there are s ties in the X-series and m_i individuals in the i^{th} tie; similarly, there are t ties in the Y-series and j^{th} tie has m'_j individuals.

9.4. Example – correlation coefficient

The following table gives the age – distribution of the population and the number of unemployed in town. Find the coefficient of correlation between the mid-values of the age- groups and the percentage of unemployed in different age – constituents.

Age	Number of persons in '000	Number of unemployed
20 – 30	40	400
30 – 40	55	1100
40 – 50	32	960
50 - 60	20	1600
60 – 70	8	1600

Solution:



```
> age_mid <-   c(25,35,45,55,65)
> population<-   c(40,55,32,20,8)
> unemployed<-   c(400,1100,960,1600,1600)
> df<-   data.frame(age=age_mid,perc=(unemployed * 100)/(population * 1000))
> cor(df$age,df$perc,method="pearson")
[1] 0.8856867
```

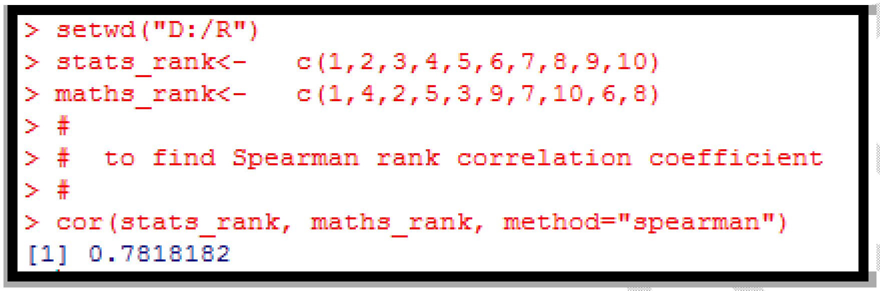
The Pearson coefficient of correlation between the mid-values of the age-groups and the percentage of unemployed in different age – constituents is approximately, 0.886.

9.5. Example – Rank correlation Coefficient

The following are the ranks obtained by 10 students in Statistics and Mathematics. Find the rank correlation coefficient.

Subject	Rank – by student	Rank – by student	Rank – by student	Rank – by student	Rank – by student	Rank – by student	Rank – by student	Rank – by student	Rank – by student	Rank – by student
Stats	1	2	3	4	5	6	7	8	9	10
Maths	1	4	2	5	3	9	7	10	6	8

Solution



```
> setwd("D:/R")
> stats_rank<-   c(1,2,3,4,5,6,7,8,9,10)
> maths_rank<-   c(1,4,2,5,3,9,7,10,6,8)
> #
> #  to find Spearman rank correlation coefficient
> #
> cor(stats_rank, maths_rank, method="spearman")
[1] 0.7818182
```

We find that the rank correlation coefficient is approximately 0.78

Limitations:

1. The formula for correlation coefficient holds only if there is a linear relationship between the variables; that is the relationship between the variables is linear.

2. Correlation theory does not establish casual relationship between the variables. It does not suggest that the variations in y are caused by the variations in x or vice versa.

9.6. Spurious Correlation

Correlations are sometimes observed between variables not conceivably be casually related. This type of correlation is called spurious correlation or chance correlation and they do not provide any casual relationship between the variables involved. For example: number of births and number of murders.

9.7. Probable Error

In a number of situations, we have calculated the correlation coefficient between two variables x, and y, for example, the height of father and height of son. To calculate the correlation coefficient we have taken a pair of 'n' pairs of values from a large population. The correlation coefficient calculated from the sample may not be the same as the correlation coefficient of the population. It is possible to determine the limits between which the coefficient of correlation of the population will lie from the knowledge of the sample correlation coefficient.

Probable error is a method of obtaining the correlation coefficient of the population. It is defined as:

P.E. = $0.674 * (1 - r^2) / \sqrt{N}$, where r is the correlation coefficient of the sample of N pairs of observations.

This can be expressed as P.E. = 2/3 S.E where S.E is the standard error of the correlation coefficient and is given by
S.E. = $(1 - r^2) / \sqrt{N}$

The limits of the population correlation coefficient are given by

$$\rho = r \pm P.E.$$

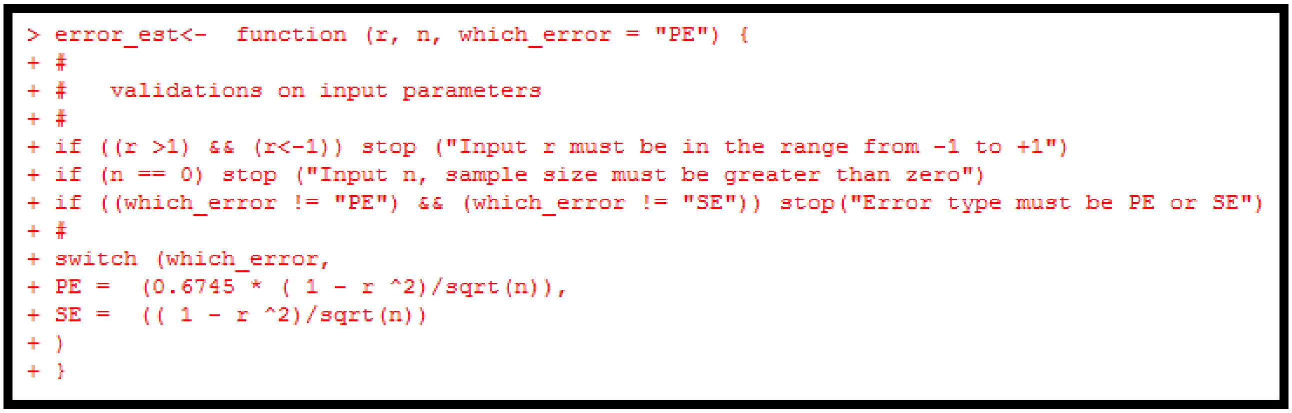
where ρ denotes the correlation coefficient in the population.

Note:

i) If the value of r is less than the probable error, then there is no evidence of correlation.
ii) If the value of r is more than six times, the probable error, then the presence of correlation coefficient is certain.
iii) Since r lies between -1 and +1, the probable error is never negative.
iv) The formula for P.E. is valued only if
   a. the sample chosen to find r is a simple random variable
   b. the population is normal.

9.8. Exercise Probable Error

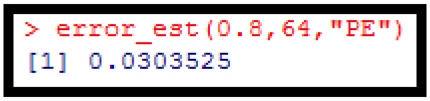
Write a function in R to calculate the Probable Error (by default) or Standard Error based on the input r (the correlation coefficient) and n the sample size. Validate the input – r to accept values from -1 to +1 and n sample size > 0.



```
> error_est<-  function (r, n, which_error = "PE") {
+ #
+ #   validations on input parameters
+ #
+ if ((r >1) && (r<-1)) stop ("Input r must be in the range from -1 to +1")
+ if (n == 0) stop ("Input n, sample size must be greater than zero")
+ if ((which_error != "PE") && (which_error != "SE")) stop("Error type must be PE or SE")
+ #
+ switch (which_error,
+ PE =   (0.6745 * ( 1 - r ^2)/sqrt(n)),
+ SE =   (( 1 - r ^2)/sqrt(n))
+ )
+ }
```

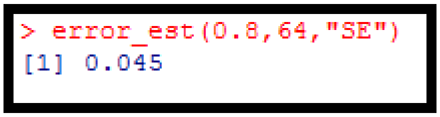
Use the above function, **error_est** to calculate the probable error and also the standard error of the coefficient of correlation as per the following details:

r = 0.8 ; n = 64 -> Probable Error



```
> error_est(0.8,64,"PE")
[1] 0.0303525
```

r = 0.8 ; n = 64 -> Standard Error



```
> error_est(0.8,64,"SE")
[1] 0.045
```

10. Regression

Correlation denotes the association between two quantitative variables. Regression involves estimating the best equation to summarize the association.

The degree of association is measured by a correlation coefficient, denoted by r. It is sometimes called Pearson's correlation coefficient after its originator and is a measure of linear association. If a curved line is needed to express the relationship, other and more complicated measures of correlation must be used.

Correlation describes the strength of an association between two variables, and is completely symmetrical, the correlation between A and B is the same as the correlation between B and A. However, if the two variables are related, it means that when one changes by a certain amount the other changes on an average by a certain amount. If y represents the dependent variable and x the independent variable, this relationship is described as the regression of y on x.

The regression equation representing how much y changes with any given change of x can be used to construct a regression line on a scatter diagram, and in the simplest case this is assumed to be a straight line. The direction in which the line slopes depends on whether the correlation is positive or negative. When the two sets of observations increase or decrease together (positive) the line slopes upwards them left to right; when one set decreases as the other increases the line slopes downwards from left to right. As the line must be straight, it will probably pass through few, if any, of the dots. Given that the association is well described by a straight line we have to define two features of the line if we are to place it correctly on the diagram. First one is its distance from the baseline and the other being its slope. This is explained in the following regression equation.

$y = a + bx$

Regression equation enables us to predict y from x and gives us a better summary of the relationship between the two variables.

11. Various Regression models

a. Linear Regression

The two basic types of regression are simple linear regression and multiple linear regression.

- Simple linear regression uses one independent variable to explain and / or predict the value of a dependent variable, Y
- Multiple linear regression uses two or more independent variables are used to predict the value of a dependent variable.

The difference between the two is the number of independent variables. In both cases, there is only a single dependent variable.

Linear Regression: $Y = \alpha + \beta X + \varepsilon$

Multiple Regression: $Y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \ldots\ldots + \beta_n X_n + \varepsilon$

Where Y = the variable that we are trying to predict
X = the variable that we are using to predict Y
α = the intercept
β = the slope
ε = the regression residual

- This is the oldest type of regression, designed 250 years ago
- Computations (On small data) could be easily be carried out by a human being or computer.
- This can be used for interpolation, but not suitable for predictive analytics
- This has many drawbacks when applied to modern data, e.g., sensitivity to both outliers and cross-correlations (both in variable and observation domains) and subject to over-fitting. A better solution is piecewise-linear regression, in particular for time series.

b. Polynomial Regression

Polynomial regression uses one independent variable x and a dependent variable y.

$Y = \alpha + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \ldots\ldots + \beta_n X^n + \varepsilon$

- In principle this is no different from fitting multiple regression model except that the powers of x play the role of different independent variables. But polynomial regression has special features.
- We fit a polynomial to smooth out fluctuations in the data caused by random or uncontrolled errors, not because it is thought to represent the relationship.
- While fitting the polynomial regression the form of the null hypothesis takes is that polynomial regression being fitted represents certain relationship and secondly, whether terms of higher degree contributes significantly to the relationship.

c. Logistic Regression

We use the logistic regression equation to predict the probability of a dependent variable taking the dichotomy values 0 or 1. Suppose $x_1, x_2, x_3, ..x_p$ are the independent variables, α and β_k ($k = 1,2..p$) are the parameters, and E(y) is the expected value of the dependent variable y, then the logistic regression equation is

$$E(y) = 1/\left(1 + e^{-(\alpha + \Sigma\beta_k x_k)}\right)$$

- This is used extensively in clinical trials, scoring and fraud detection, when the response is binary (chance of succeeding or failing, e.g. for a new tested drug or credit card transaction).
- This also suffers same drawbacks as linear regression (not robust, model-dependent), and computing regression coefficients involves using complex iterative, numerically unstable algorithm.
- This can be well approximated by linear regression after transforming the response (ligit transform).
- Some versions (Poisson or Cox regression) have been designed for a non-binary response, for categorical data (classification), ordered integer response (age groups), and even continuous response (regression trees)

12. Statistical Models in R

R includes a variety of tools for complex modeling, among them:

- glm() for generalized linear models
- gam() for generalized additive models
- lme() and lmer() for linear mixed-effects models
- nls() and nlme() for nonlinear models

R functions such as aov(), lm() use a formula interface to specify the variables to be included in the analysis. The formula determines the model that will be built and tested by the R procedure. The basic format of such a formula is

Response variable ~ explanatory variables

The tilde should be read “is modeled by” or “is modeled as a function of”.

A basic regression analysis would be formulated this way:

y ~ x

… where “x” is the explanatory variable or IV and “y” is the response variable or DV. Additional variables would be added in as follows:

y ~ x + z which would make this z multiple regression with two predictors. This raises a critical issue that must be understood to get model formulae correct.

Symbols used as mathematical operators in other contexts do not have their usual mathematical meaning inside model formulae.

The following table lists the meaning of these symbols when used in a formula.

Symbol	Example	Meaning
+	+x	Include this variable
-	-x	Delete this variable
:	x:z	Include interaction between these variables
*	x*z	Include these variables and the interactions between them
/	x/z	Nesting: include z nested within x
|	x | z	Conditioning: include x given z
^	(u + v +w)^3	Include these variables and all interactions up to three way
poly	poly(x,3)	Polynomial regression: Orthogonal polynomials
Error	Error(a/b)	Specify the error term
I	I(x*z)	As is: include a new variable consisting of these variables multiplied
I	-I	Intercept: Delete the intercept (regress through the origin)

- Some formula structures can be specified in more than one way..
  - y ~ u + v + w + u : v + u :w + u: v: w
  - y ~ u * v * w – u :v :w
  - y ~ (u + v + w) ^ 2
- The nature of variables – binary, categorical (factors), numerical – will determine the nature of the analysis.
- For example. If u and v are factors
  - y ~ u + v dictates an analysis of variance (without the interaction term).
- If u and v are numerical, the same formula would dictate a multiple regression.
- If u is numerical and v is a factor, then an analysis of covariance is dictated.

13. Least square method – Line of best fit

A line of best fit is a straight line that is the best possible approximation of the given set of data.

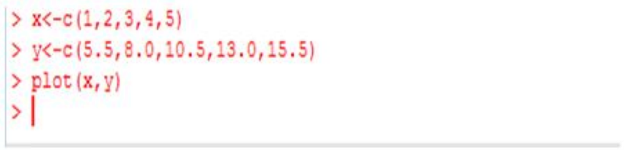
It is used to study the nature of relation between two variables.

Let us assume y = 2.5 x + 3

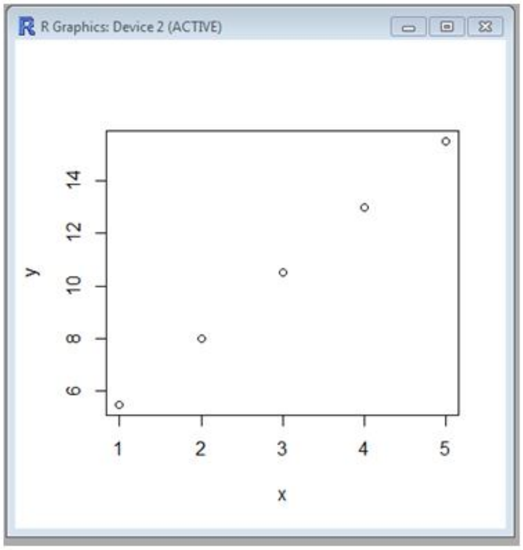
x	y
1	5.5
2	8
3	10.5
4	13
5	15.5

A line of best fit can be roughly determined using an eyeball method by drawing a straight line on a scatter plot so that the number of points above the line and below the line is about equal (Note that the line passes through as many points as possible).

Plot the points on a coordinate plane.



```
> x<-c(1,2,3,4,5)
> y<-c(5.5,8.0,10.5,13.0,15.5)
> plot(x,y)
>
```



A more accurate way of finding the line of best fit is the least square method.

Use the following steps to find the equation of line of best fit for a set of ordered pairs.
Step 1: Calculate the mean of the x-values and the mean of y-values.
Step 2: Compute the sum of squares of the x-values.

Step 3: Compute the sum of each x-value multiplied by its corresponding y-value.

Step 4: Calculate the slope of the line using the formula:

$$m = \frac{\sum xy - \frac{(\sum x)(\sum y)}{n}}{\sum x^2 - \frac{(\sum x)^2}{n}}$$

where n is the total number of data points.

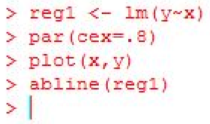
Step 5: Compute the y-intercept of the line using the formula:

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

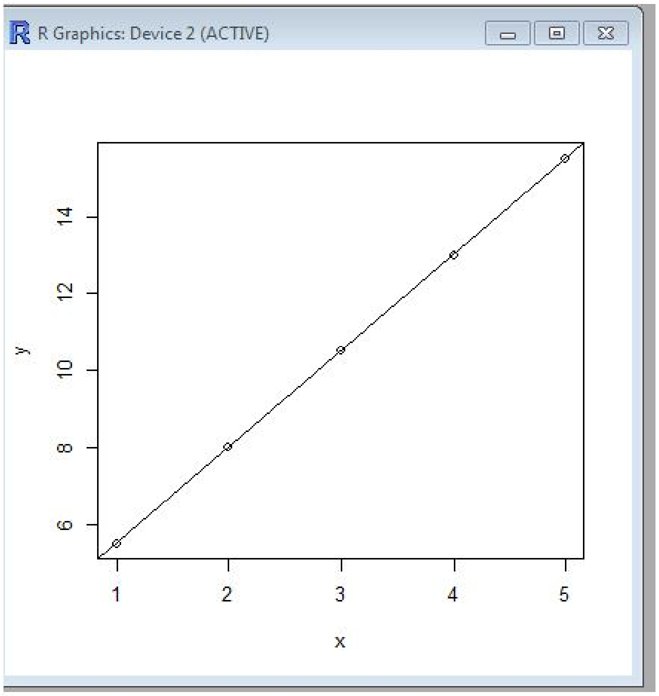
where $\bar{y}$ and $\bar{x}$ are the mean of the x- and y-coordinates of the data points respectively.

Step 6: Use the slope and the y-intercept to form the equation of the line.

x	1	2	3	4	5
y	5.5	8.0	10.5	13.0	15.5



```
> reg1 <- lm(y~x)
> par(cex=.8)
> plot(x,y)
> abline(reg1)
> |
```



Calculate the slope of the line:

						Σ
x	1	2	3	4	5	15
y	5.5	8.0	10.5	13.0	15.5	52.5
xy	5.5	16.0	31.5	52.0	77.5	182.5
x^2	1	4	9	16	25	55

$$m = \frac{\sum xy - \frac{(\sum x)(\sum y)}{n}}{\sum x^2 - \frac{(\sum x)^2}{n}}$$

where n is the total number of data points.

m = (182.5 – (15 X 52.5)/5) / (55 – (15^2 / 5) = 25 / 10 = 2.5

Compute the y-intercept of the line using the formula:

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

where $\bar{y}$ and $\bar{x}$ are the mean of the x- and y-coordinates of the data points respectively.

b = (52.5 / 5) - 2.5 *(15/5) = 10.5 – 7.5 = 3

The regression equation is y = 2.5 x + 3.0.

14. Simple Linear Regression

A simple linear regression model that describes the relationship between two variables x and y can be expressed by the following equation. The numbers **α** and **β** are called parameters, and **ε** is the error term.

y = α + βx + ε

Using R, we can compute the slope and intercept as follows:

lm is used to fit linear models. It can be used to carry out regression, single stratum analysis of variance and analysis of covariance (although aov may provide a more convenient interface for these).

lm (formula, data, subset, weights, na.action, method = "qr", model = TRUE, x = FALSE, y = FALSE, qr = TRUE, singular.ok = TRUE, contrasts = NULL, offset, ...)

Arguments

formula an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are given under 'Details'.

data — an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula), typically the environment from which lm is called.

subset — an optional vector specifying a subset of observations to be used in the fitting process.

weights — an optional vector of weights to be used in the fitting process. Should be NULL or a numeric vector. If non-NULL, weighted least squares is used with weights weights (that is, minimizing sum(w*e^2)); otherwise ordinary least squares is used. See also 'Details',

na.action — a function which indicates what should happen when the data contain NAs. The default is set by the na.action setting of options, and is na.fail if that is unset. The 'factory-fresh' default is na.omit. Another possible value is NULL, no action. Value na.exclude can be useful.

method — the method to be used; for fitting, currently only method = "qr" is supported; method = "model.frame" returns the model frame (the same as with model = TRUE, see below).

model, x, y, qr — logicals. If TRUE the corresponding components of the fit (the model frame, the model matrix, the response, the QR decomposition) are returned.

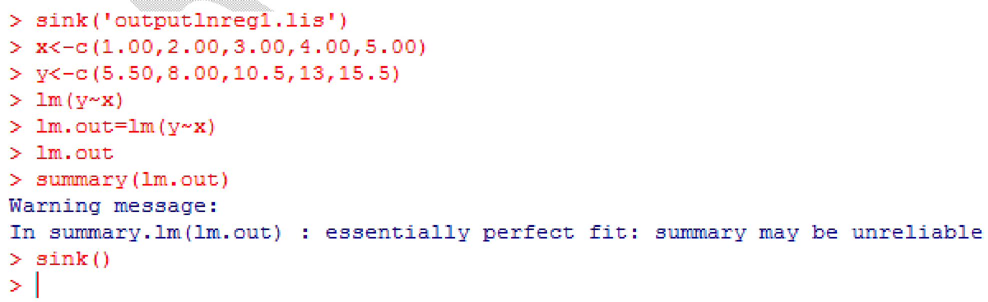
singular.ok — logical. If FALSE (the default in S but not in **R**) a singular fit is an error.

contrasts — an optional list. See the contrasts.arg of model.matrix.default.

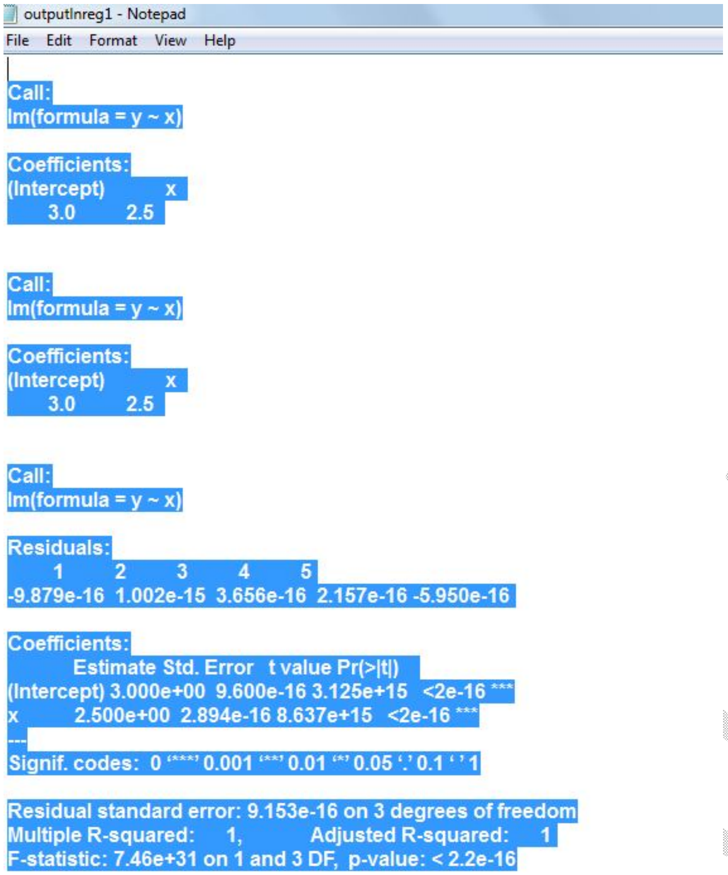
offset — this can be used to specify an *a priori* known component to be included in the linear predictor during fitting. This should be NULL or a numeric vector of length equal to the number of cases. One or more offset terms can be included in the formula instead or as well, and if more than one are specified their sum is used. See model.offset.

... — additional arguments to be passed to the low level regression fitting functions (see below).

Models for lm are specified symbolically. A typical model has the form response ~ terms where response is the (numeric) response vector and terms is a series of terms which specifies a linear predictor for response.



```
> sink('outputlnreg1.lis')
> x<-c(1.00,2.00,3.00,4.00,5.00)
> y<-c(5.50,8.00,10.5,13,15.5)
> lm(y~x)
> lm.out=lm(y~x)
> lm.out
> summary(lm.out)
Warning message:
In summary.lm(lm.out) : essentially perfect fit: summary may be unreliable
> sink()
> |
```



```
Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)          x
       3.0         2.5


Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)          x
       3.0         2.5


Call:
lm(formula = y ~ x)

Residuals:
         1          2          3          4          5
-9.879e-16  1.002e-15  3.656e-16  2.157e-16 -5.950e-16

Coefficients:
             Estimate Std. Error   t value Pr(>|t|)
(Intercept) 3.000e+00  9.600e-16 3.125e+15   <2e-16 ***
x           2.500e+00  2.894e-16 8.637e+15   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 9.153e-16 on 3 degrees of freedom
Multiple R-squared:     1,	Adjusted R-squared:     1
F-statistic: 7.46e+31 on 1 and 3 DF,  p-value: < 2.2e-16
```

If, for x_i the regression equation predicts a value of y_{est}, the regression error is $(y_i - y_{est})$. It can easily be shown that any straight line passing through the mean values x and y will give a total prediction error $\sum(y_i - y_{est})$ of zero because the positive and negative terms exactly cancel. To remove the negative signs we square the differences and the regression equation chosen to minimize the sum of squares of the prediction errors, $S^2 = \sum(y_i - y_{est})^2$

It can be shown that the one straight line that minimizes S^2, the least squares estimate gives:

$$b = \frac{\Sigma xy - n\bar{x}\bar{y}}{(n-1)SD(x)^2}$$

$$a = \bar{y} - b\bar{x}$$

The standard error of the slope SE_b is given by:

$$SE_b = \frac{S_{res}}{\sqrt{\Sigma(x-\bar{x})^2}}$$

where S_{res} is the residual standard deviation, is given by:

$$S_{res} = \sqrt{\frac{\sum(y_i - y_{est})^2}{n-2}}$$

14.1. Estimated Simple Regression Equation

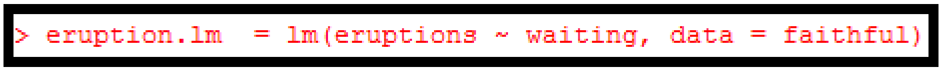
If we choose the parameters α and β in the simple linear regression model so as to minimize the sum of squares of the error term ε, we will have the so called estimated simple regression equation. It allows us to compute the fitted values of y based on values of x.

ŷ = a + b x

Apply the simple linear regression model for the data set faithful, and estimate the next eruption duration if the waiting time since the last eruption has been 90 minutes.

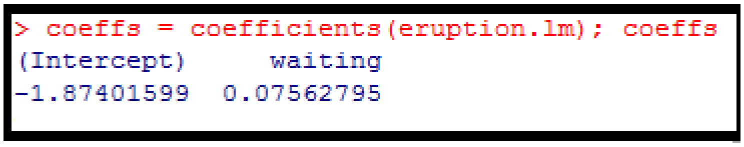
Solution

We apply the lm function to a formula that describes the variable eruptions by the variable waiting, and save the linear regression model in a new variable eruption.lm.



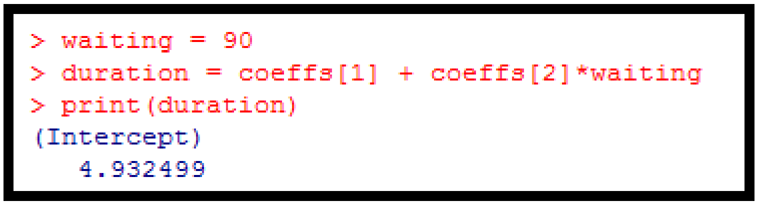
```
> eruption.lm  = lm(eruptions ~ waiting, data = faithful)
```

Then we extract the parameters of the estimated regression equation with the coefficients function.



```
> coeffs = coefficients(eruption.lm); coeffs
(Intercept)     waiting 
-1.87401599  0.07562795 
```

Now, we fit the eruption duration using the estimated regression equation



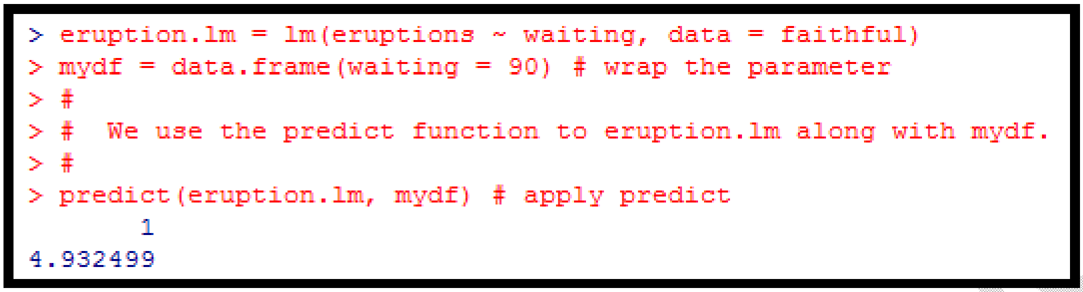
```
> waiting = 90
> duration = coeffs[1] + coeffs[2]*waiting
> print(duration)
(Intercept) 
   4.932499 
```

Conclusion

Based on the simple linear regression model, if the waiting time since the last eruption has been 90 minutes, we expect the next one to last 4.932499 minutes.

Alternative Solution

We include the waiting parameter inside a new data frame named as mydf.



```
> eruption.lm = lm(eruptions ~ waiting, data = faithful)
> mydf = data.frame(waiting = 90) # wrap the parameter
> #
> #  We use the predict function to eruption.lm along with mydf.
> #
> predict(eruption.lm, mydf) # apply predict
       1
4.932499
```

The coefficient of determination of a linear regression model is the quotient of the variances of the fitted values and observed values of the dependent variable.

We denote y, as the observed value of the dependent variable, $\bar{y}$ as its mean, and $\hat{y}_i$- as the fitted value, then the coefficient of determination is:

$$r^2 = \frac{\sum(\hat{y_i} - \bar{y})^2}{\sum(y_i - \bar{y})^2}$$

14.2. Coefficient of Determination, r^2 or R^2

The coefficient of determination, r^2 is useful because it gives the proportion of the variance (fluctuation) of one variable that is predictable from the other variable. It is a measure that allows us to determine how certain one can be in making predictions from a certain model / graph.

The coefficient of determination is the ratio of the explained variation to the total variation.

The coefficient of determination is such that $0 \leq r^2 \leq 1$, and denotes the strength of the linear association between x and y.

The coefficient of determination represents the percent of the data that is the closest to the line of best fit. For example, if r = 0.922, then r^2 = 0.850, which means that 85% of the total variation in y can be explained by the linear relationship between x and y (as described by the regression equation). The other 15% if the total variation in y remains unexplained.

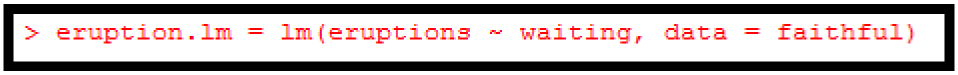
The coefficient of determination is a measure of how well the regression line represents the data. If the regression line passes exactly through every point on the scatter plot, it would be able to explain all of the variation, the further the line is away from the points, the less is it able to explain.

Problem

Find the coefficient of determination for the simple linear regression model of the data set faithful.

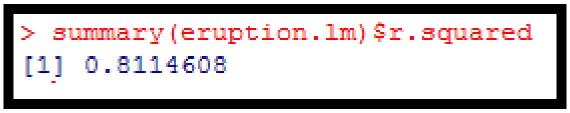
Solution

We apply the lm function to a formula that describes the variable eruptions by the variable waiting, and save the linear regression model in a new variable eruption.lm.



```
> eruption.lm = lm(eruptions ~ waiting, data = faithful)
```

Then we extract the coefficient of determination from the ***r.squared*** attribute of its summary.



```
> summary(eruption.lm)$r.squared
[1] 0.8114608
```

Answer

The coefficient of determination of the simple linear regression for the data faithful is 0.8114608. That means 81% of the total variation in eruptions can be explained by the linear relationship between waiting and eruptions (as described by the regression equation).

14.3. Significance Test for Linear Regression

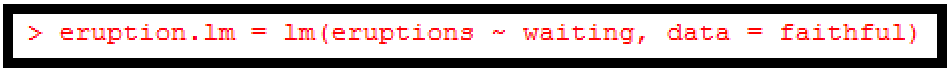
Assume that the error ε in the linear regression model is independent of x and is normally distributed with zero mean and constant variance. We can decide whether there is any significant relationship between x and y by testing the null hypothesis that β = 0.

Problem

Decide whether there is a significant relationship between the variables in the linear regression model of the data set faithful at 0.05 significance level.

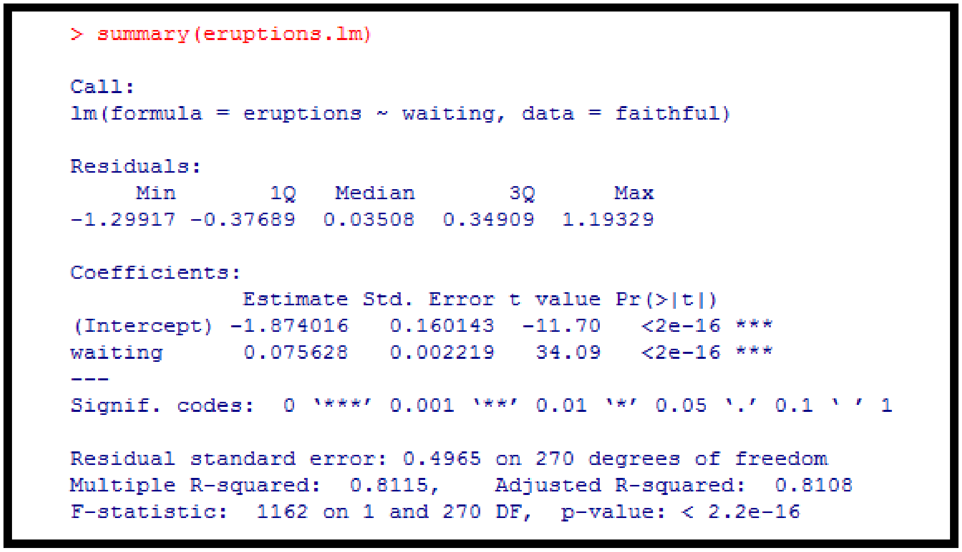
Solution

We apply the lm function to a formula that describes the variable eruptions by the variable waiting, and save the linear regression model in a new variable eruptions.lm.



```
> eruption.lm = lm(eruptions ~ waiting, data = faithful)
```

Then we print out the F-statistics of the significance test with the summary function.



```
> summary(eruptions.lm)

Call:
lm(formula = eruptions ~ waiting, data = faithful)

Residuals:
     Min       1Q   Median       3Q      Max
-1.29917 -0.37689  0.03508  0.34909  1.19329

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.874016   0.160143  -11.70   <2e-16 ***
waiting      0.075628   0.002219   34.09   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4965 on 270 degrees of freedom
Multiple R-squared:  0.8115,    Adjusted R-squared:  0.8108
F-statistic:  1162 on 1 and 270 DF,  p-value: < 2.2e-16
```

Answer

As the p-value is much less than 0.05, we reject the null hypothesis that β = 0. Hence there is a significant relationship between the variables in the linear regression model of the data set faithful.

14.4. Confidence Interval for Linear Regression

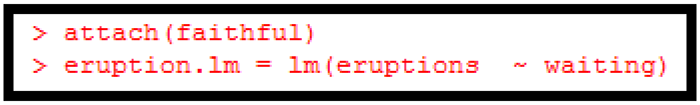
Let us assume that the error term **ε** in the linear regression model is independent of x, and is normally distributed, with mean zero and constant variance. For a given value of x, the interval estimate for the mean of the independent variable, y is called the confidence interval.

Problem

In the dataset faithful, develop a 95 % confidence interval of the mean eruption for the waiting time of 80 minutes.

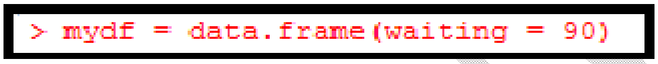
Solution

We apply the ***lm*** function to a formula that describes the variable eruptions by the variable waiting, and save the linear regression model in a new variable **eruption.lm**.



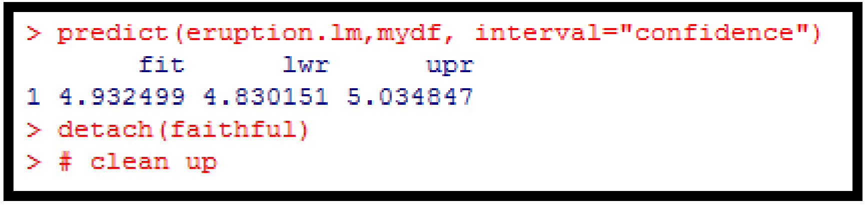
```
> attach(faithful)
> eruption.lm = lm(eruptions  ~ waiting)
```

Then we create a new data frame that set the waiting time value.



```
> mydf = data.frame(waiting = 90)
```

We now apply the predict function and set the predictor variable in the mydf argument. We also set the interval type as "confidence", and use the default 0.95 confidence level.



```
> predict(eruption.lm,mydf, interval="confidence")
       fit      lwr      upr
1 4.932499 4.830151 5.034847
> detach(faithful)
> # clean up
```

Conclusion

The 95 % confidence level of the mean eruption duration for the waiting time of 90 minutes is between 4.830151 and 5.034847.

14.5. Prediction Interval for Linear Regression

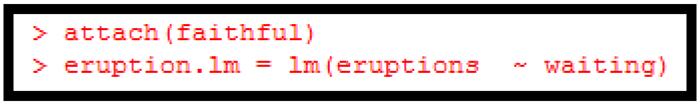
Let us assume that the error term **ε** in the simple linear regression model is independent of x, and is normally distributed, with mean zero and constant variance. For a given value of x, the interval estimate for the mean of the independent variable, y is called the prediction interval.

Problem

In the data set faithful, develop a 95 % prediction interval of the eruption duration for the waiting time of 90 minutes.

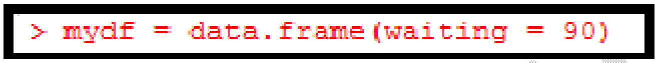
Solution

We apply the ***lm*** function to a formula that describes the variable eruptions by the variable waiting, and save the linear regression model in a new variable **eruption.lm**.



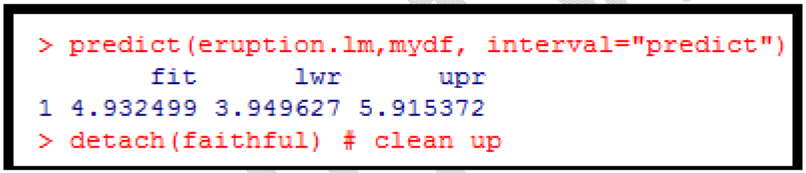
```
> attach(faithful)
> eruption.lm = lm(eruptions  ~ waiting)
```

Then we create a new data frame that set the waiting time value.



```
> mydf = data.frame(waiting = 90)
```

We now apply the predict function and set the predictor variable in the mydf argument. We also set the interval type as “predict”, and use the default 0.95 confidence level.



```
> predict(eruption.lm,mydf, interval="predict")
       fit      lwr      upr
1 4.932499 3.949627 5.915372
> detach(faithful) # clean up
```

Conclusion

The 95% prediction level of the mean eruption duration for the waiting time of 90 minutes is between 3.949627 and 5915372.

14.6. Residual Plot

The residual data of the simple linear regression model is the difference between the observed data of the dependent variable y and the fitted values ŷ.

Residual = y – ŷ

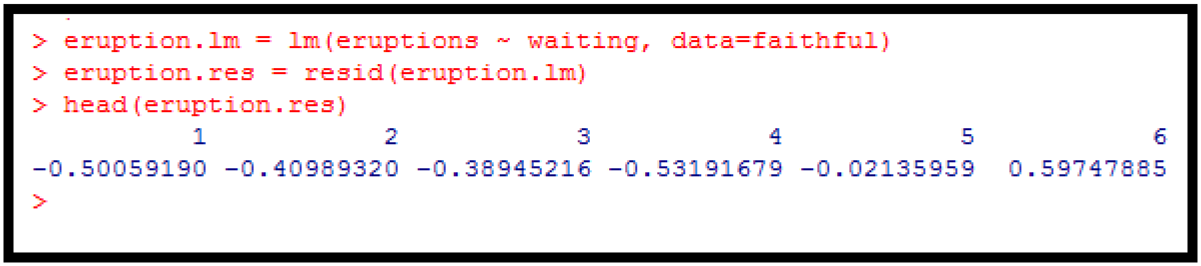
Problem

Plot the residual of the simple linear regression model of the data set **faithful** against the independent variable **waiting**.

Solution

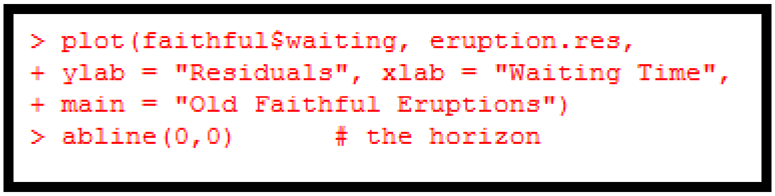
We apply the **lm** function to a formula that describes the variable eruptions by the variable **waiting**, and save the linear regression model in a new variable **eruption.lm**.

Then we compute the residual with the **resid** function.

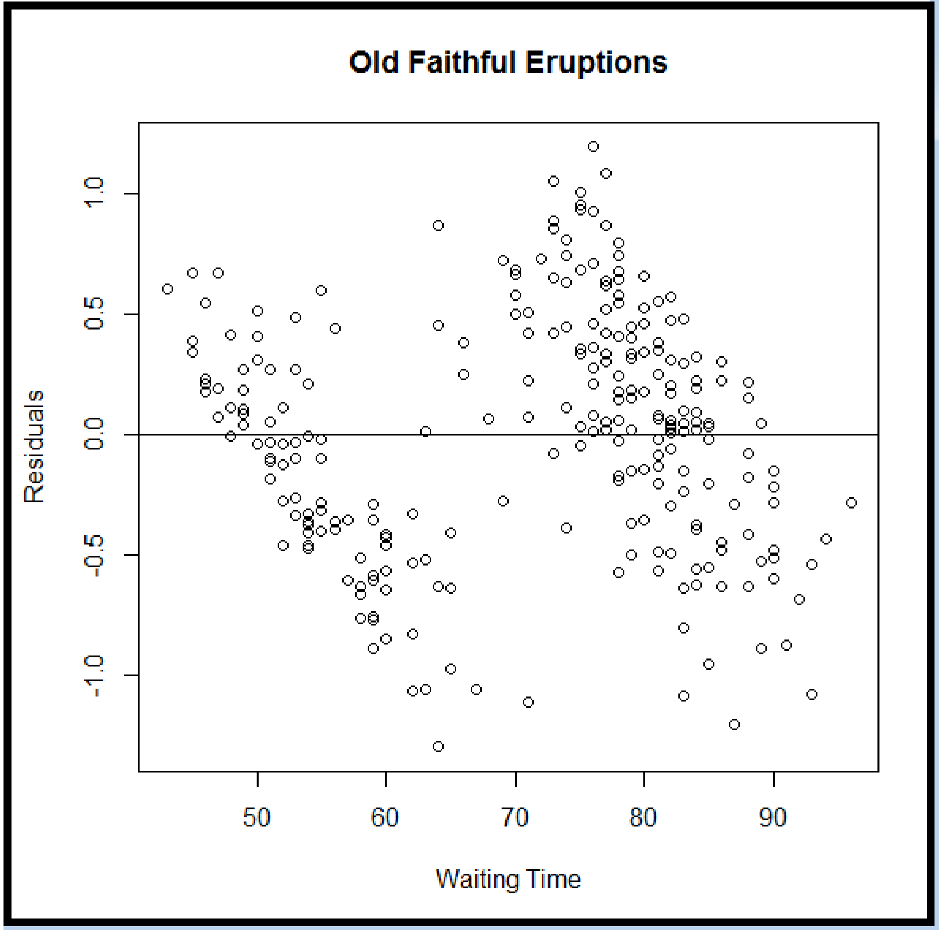


```
> eruption.lm = lm(eruptions ~ waiting, data=faithful)
> eruption.res = resid(eruption.lm)
> head(eruption.res)
          1           2           3           4           5           6
-0.50059190 -0.40989320 -0.38945216 -0.53191679 -0.02135959  0.59747885
>
```

We now plot the residual against the observed values of the variable **waiting**.



```
> plot(faithful$waiting, eruption.res,
+ ylab = "Residuals", xlab = "Waiting Time",
+ main = "Old Faithful Eruptions")
> abline(0,0)       # the horizon
```



14.7. Standard Residual

The standard residual is the residual divided by its standard deviation.

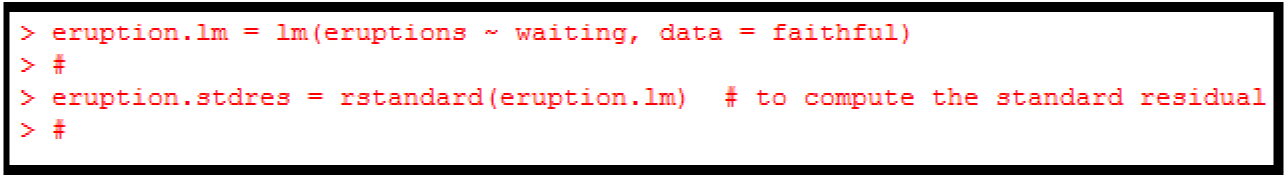
Standard Residual i = Residual i / Standard Deviation of Residual i

Problem

Plot the standardized residual of the simple linear regression model of the data set **faithful** against the independent variable **waiting**.

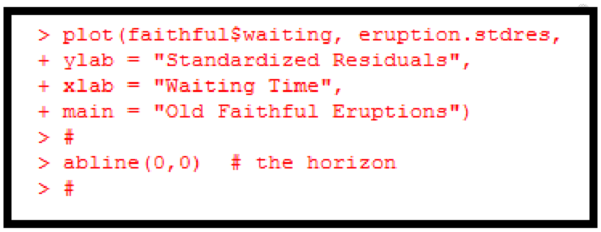
Solution

We apply the **lm** function to a formula that describes the variable eruptions by the variable **waiting**, and save the linear regression model in a new variable **eruption.lm**. Then we compute the standardized residual with the ***rstandard*** function.

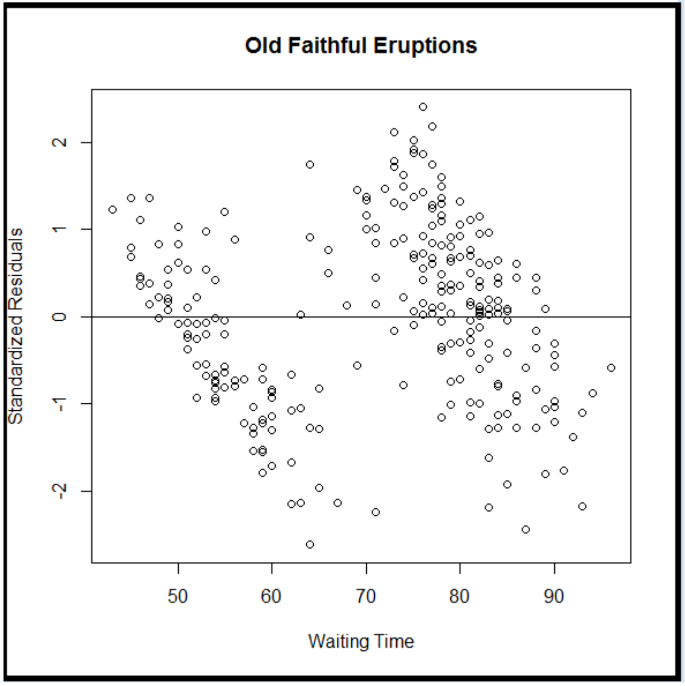


```
> eruption.lm = lm(eruptions ~ waiting, data = faithful)
> #
> eruption.stdres = rstandard(eruption.lm)   # to compute the standard residual
> #
```

We now plot the standardized residual against the observed values of the variable **waiting**.



```
> plot(faithful$waiting, eruption.stdres,
+ ylab = "Standardized Residuals",
+ xlab = "Waiting Time",
+ main = "Old Faithful Eruptions")
> #
> abline(0,0)   # the horizon
> #
```



14.8. Normal Probability Plot of Residuals

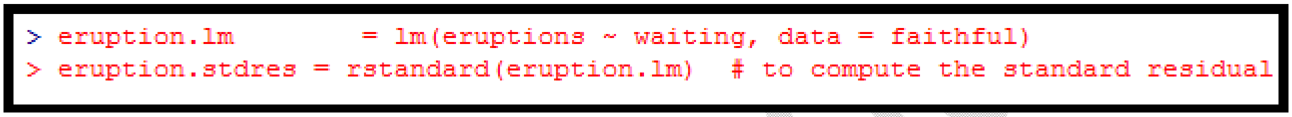
The normal probability plot is a graphical tool for comparing a data set with the normal distribution. We can use it with the standard residual of the linear regression model and see if the error term e is actually normally distributed.

Problem

Create a normal probability plot for the standardized residual of the data set faithful.

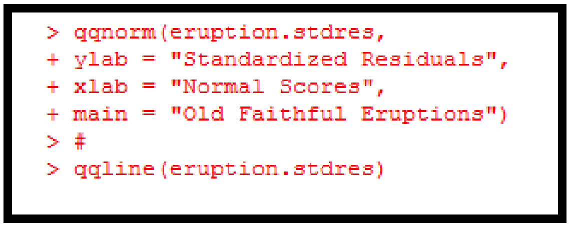
Solution

We apply the **lm** function to a formula that describes the variable eruptions by the variable waiting, and save the linear regression model in a new variable **eruption.lm**. Then we compute the standardized residual with the ***rstandard*** function.

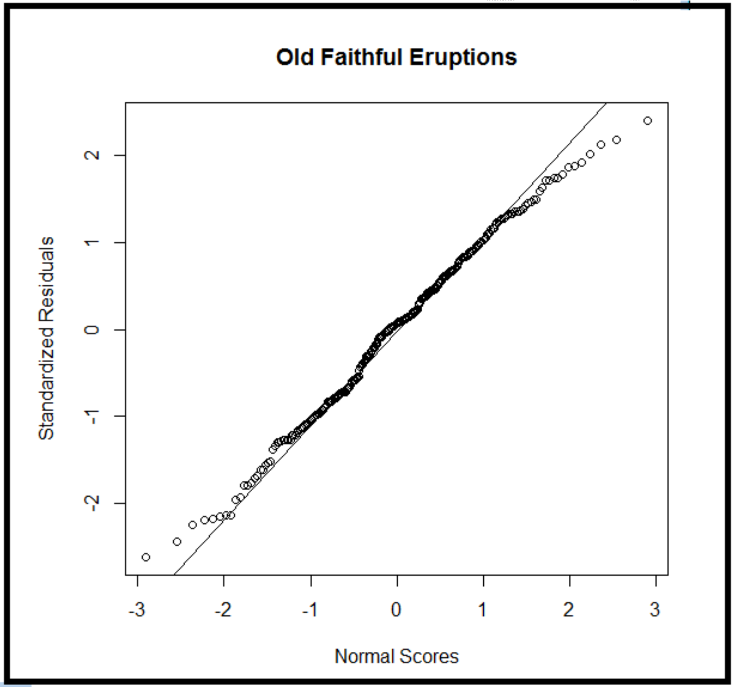


```
> eruption.lm        = lm(eruptions ~ waiting, data = faithful)
> eruption.stdres = rstandard(eruption.lm)  # to compute the standard residual
```

We now create the normal probability plot with the **qqnorm** function, and add the **qqline** for further comparison.



```
> qqnorm(eruption.stdres,
+ ylab = "Standardized Residuals",
+ xlab = "Normal Scores",
+ main = "Old Faithful Eruptions")
> #
> qqline(eruption.stdres)
```



About Quantile-Quantile Plots

qqnorm is a generic function the default method of which produces a normal QQ plot of the values in y. qqline adds a line to a “theoretical”, by default normal, quantile-quantile plot which passes through the probs quantiles, by default the first and third quartiles.

qqplot produces a QQ plot of two datasets.

Graphical parameters may be given as arguments to qqnorm, qqplot and qqline.

Usage

```
qqnorm(y, ...)
#
##        Default S3 method:
#
qqnorm(y, ylim, main = "Normal Q-Q Plot", xlab = "Theoretical Quantiles", ylab = "Sample Quantiles", plot.it
= TRUE, datax = FALSE,   ...)
qqline(y, datax = FALSE, ...)
qqplot(x, y, plot.it = TRUE, xlab = deparse(substitute(x)), ylab = deparse(substitute(y)), ...)
```

Arguments

x	The first sample for qqplot.
y	The second or only data sample.
xlab, ylab, main	plot labels.
plot.it logical.	Should the result be plotted?
datax logical.	Should data values be on the x-axis?
ylim, ...	graphical parameters.

Value

For qqnorm and qqplot, a list with components

x	The x coordinates of the points that were/would be plotted
y	The original y vector, i.e., the corresponding y coordinates including NAs.

References

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) *The New S Language*. Wadsworth & Brooks/Cole.

15. Multiple Linear Regression

A multiple linear regression (MLR) model that describes a dependent variable y by independent variables $x_1, x_2, x_3, \ldots, x_p$ (p > 1) is expressed by the equation as follows:

$y = \alpha + \sum \beta_k x_k + \varepsilon$ where the α, β_k (k =1,2,..p) are the parameters and ε is the error term.

15.1. Estimated Multiple Regression Equation

If we choose the parameters α and β_k (k =1,2,..p) in the multiple linear regression model so as to minimize the sum of squares of the error term ε, we will have the so called estimated multiple regression equation. It allows us to compute fitted values of y based on a set of values of x_k (k =1,2,..p).

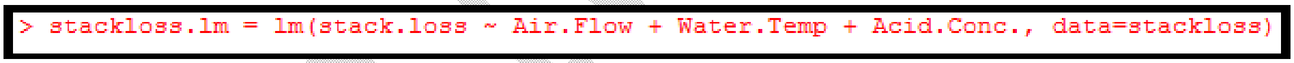
$\hat{y} = a + \sum b_k x_k$

Problem

Apply the multiple linear regression model for the built-in data set stackloss, and predict the stackloss if the air flow is 72, water temperature is 20 and acid concentration is 85.

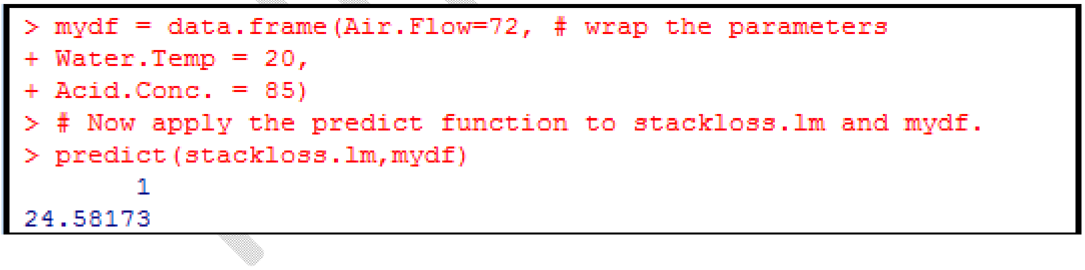
Solution

We apply the **lm** function to a formula that describes the variable **stack.loss** by the variables **Air.flow**, **Water.Temp** and **Acid.Conc**. Save the linear regression model in a new variable **stackloss.lm**.



```
> stackloss.lm = lm(stack.loss ~ Air.Flow + Water.Temp + Acid.Conc., data=stackloss)
```

We also wrap the parameters inside a new data frame named mydf.



```
> mydf = data.frame(Air.Flow=72, # wrap the parameters
+ Water.Temp = 20,
+ Acid.Conc. = 85)
> # Now apply the predict function to stackloss.lm and mydf.
> predict(stackloss.lm,mydf)
       1
24.58173
```

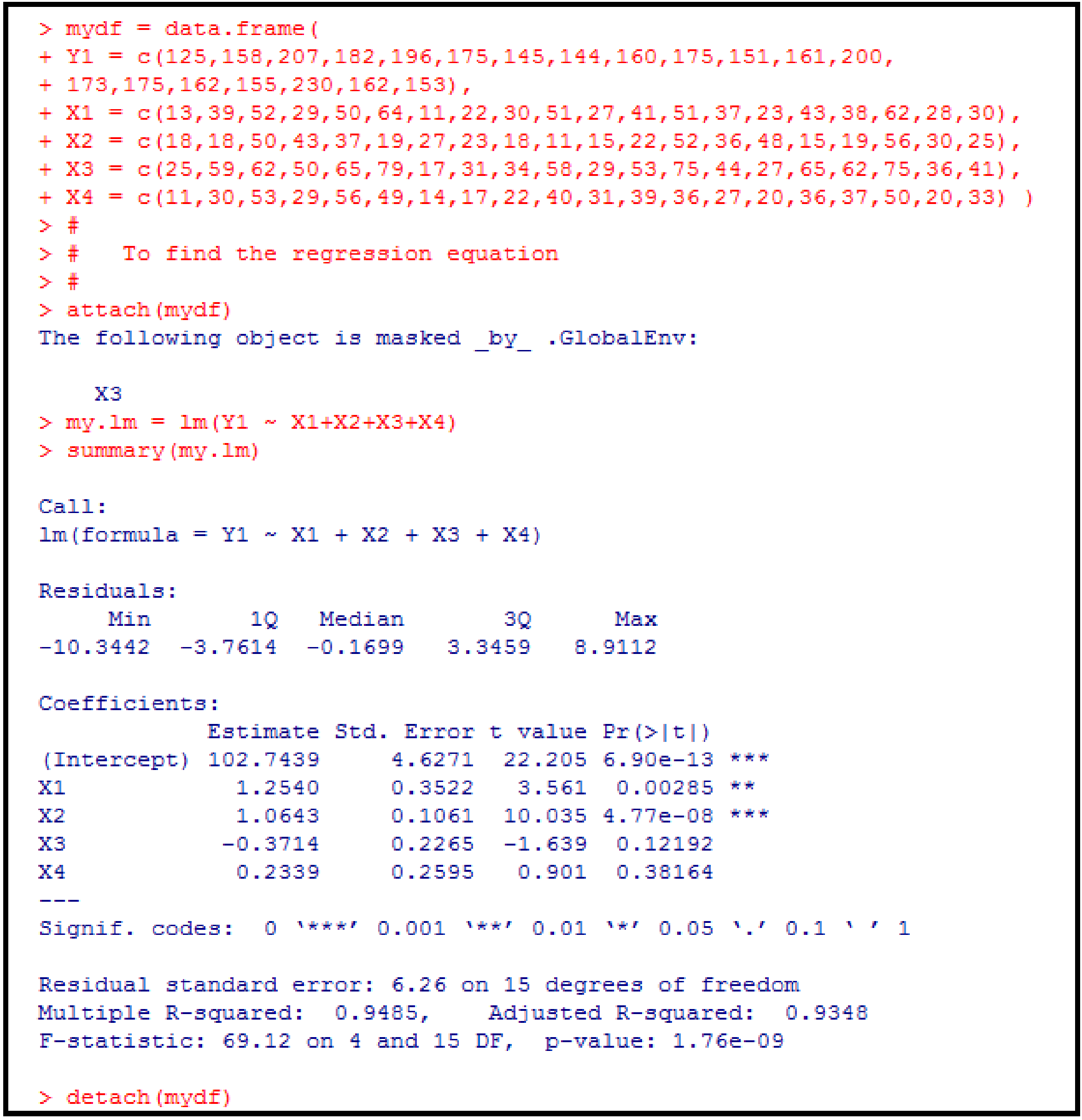
Conclusion

Based on the multiple linear regression model and the given parameters, the predicted stackloss is 24.58173.

Example - Find the regression equation between Y1 and X1,X2,X3,X4

- Y1 is a measure of success in graduate school
- X1 is a measure of intellectual ability
- X2 is a measure of work ethic
- X3 is a second measure of intellectual ability
- X4 is a measure of spatial ability

Solution



```
> mydf = data.frame(
+ Y1 = c(125,158,207,182,196,175,145,144,160,175,151,161,200,
+ 173,175,162,155,230,162,153),
+ X1 = c(13,39,52,29,50,64,11,22,30,51,27,41,51,37,23,43,38,62,28,30),
+ X2 = c(18,18,50,43,37,19,27,23,18,11,15,22,52,36,48,15,19,56,30,25),
+ X3 = c(25,59,62,50,65,79,17,31,34,58,29,53,75,44,27,65,62,75,36,41),
+ X4 = c(11,30,53,29,56,49,14,17,22,40,31,39,36,27,20,36,37,50,20,33) )
> #
> #   To find the regression equation
> #
> attach(mydf)
The following object is masked _by_ .GlobalEnv:

    X3
> my.lm = lm(Y1 ~ X1+X2+X3+X4)
> summary(my.lm)

Call:
lm(formula = Y1 ~ X1 + X2 + X3 + X4)

Residuals:
     Min       1Q   Median       3Q      Max 
-10.3442  -3.7614  -0.1699   3.3459   8.9112 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 102.7439     4.6271  22.205 6.90e-13 ***
X1            1.2540     0.3522   3.561  0.00285 ** 
X2            1.0643     0.1061  10.035 4.77e-08 ***
X3           -0.3714     0.2265  -1.639  0.12192    
X4            0.2339     0.2595   0.901  0.38164    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 

Residual standard error: 6.26 on 15 degrees of freedom
Multiple R-squared:  0.9485,    Adjusted R-squared:  0.9348 
F-statistic: 69.12 on 4 and 15 DF,  p-value: 1.76e-09 

> detach(mydf)
```

The regression equation is

Y1 = 102.7439 + 1.254* X1 + 1.0643 * X2 – 0.3714 * X3 + 0.2339 * X4

15.2. Multiple Coefficient of Determinatiion

The coefficinet of determination of a multiple linear regression model is the quotient of the variances of the fitted values and observed values of the depedent variable. If we denote y_i as the observed values of the dependent variable, $\overline{Y}$ as its mean, and $\hat{Y}_i$ as the fitted value, then the coefficient of determination is:

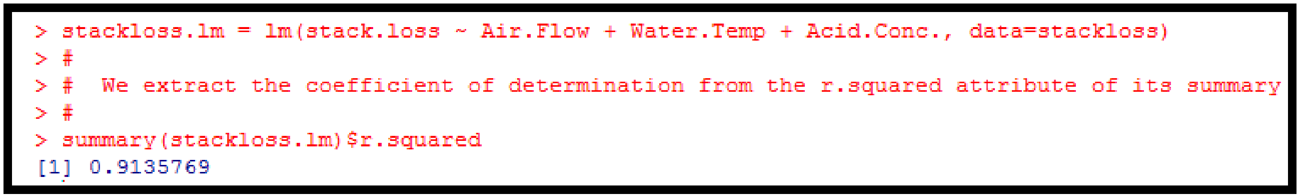
$$R^2 = \frac{\sum(\hat{y}_i - \bar{y})^2}{\sum(y_i - \bar{y})^2}$$

Problem 1

Find the coefficient of determination for the multiple linear regression model of the data set stackloss.

Solution

We apply the **lm** function to a formula that describes the variable **stack.loss** by the variables **Air.flow**, **Water.Temp** and **Acid.Conc**. Save the linear regression model in a new variable **stackloss.lm**.



```
> stackloss.lm = lm(stack.loss ~ Air.Flow + Water.Temp + Acid.Conc., data=stackloss)
> #
> #  We extract the coefficient of determination from the r.squared attribute of its summary
> #
> summary(stackloss.lm)$r.squared
[1] 0.9135769
```

Conclusion

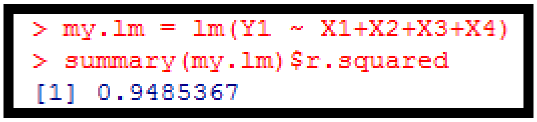
The coefficient of determination of the multiple linear regression model for the data set stackloss is 0.9135769. This means that over 91% of the total variation in **stack.loss** can be explained by the linear relationship between **Air.flow**, **Water.Temp** and **Acid.Conc** and **stack.loss** (as described by the regression equation). The other 9% of the total variation in **stack.loss** remains unexplained.

Problem 2

Find the coefficient of determination for the multiple linear regression model of the data Y_1 and (X_1,X_2,X_3 and X_4)

Solution

We apply the **lm** function to a formula that describes the variable $Y_1$by the variables and (X_1,X_2,X_3 and X_4) . Save the linear regression model in a new variable **my.lm.**



```
> my.lm = lm(Y1 ~ X1+X2+X3+X4)
> summary(my.lm)$r.squared
[1] 0.9485367
```

Conclusion

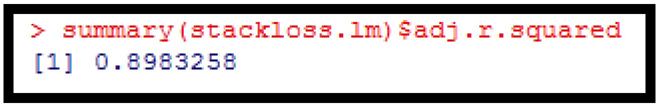
The coefficient of determination of the multiple linear regression model for the above data is 0.9485347. This means that approximately 95% of the total variation in **Y1** can be explained by the linear relationship between (X_1,X_2,X_3 and X_4) and Y_1 (as described by the regression equation). The other 5% of the total variation in $\mathbf{Y_1}$ remains unexplained.

15.3. Adjusted Coefficient of Determination

The adjusted coefficient of determination of a multiple linear regression model is defined in terms of the coefficient of determination as follows, where n is the number of observations in the data set, and p is the number of independent variables.

$R^2{}_{adj} = 1 – (1 – R^2) ((n -1) /(n- p -1))$

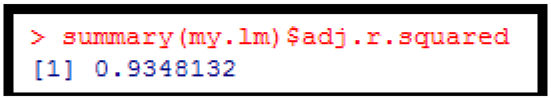
We extract the adjusted coefficient of determination from the adj.r.squared attribute of the summary of stackloss.lm as determined in the previous page.



```
> summary(stackloss.lm)$adj.r.squared
[1] 0.8983258
```

The ajusted coefficient of determination of the multiple linear regression model for the data stackloss is 0.8983258.

Similarly, when we extract adjusted coefficient of determination from the adj.r.squared attribute of the summary of my.lm as determined in the earlier pages.



```
> summary(my.lm)$adj.r.squared
[1] 0.9348132
```

The ajusted coefficient of determination of the multiple linear regression model for the linear relationship between (X_1,X_2,X_3 and X_4) and Y_1 is 0.9348132.

Note:

- The use of an adjusted R2 is an attempt to take account of the phenomenon of the R2 automatically and spuriously increasing when extra explanatory variables are added to the model.
- It is a modification due to Theil of R2 that adjusts for the number of explanatory terms in a model relative to the number of data points.
- The adjusted R2 can be negative, and its value will always be less than or equal to that of R2.
- Unlike R2, the adjusted R2 increases when a new explanator is included only if the new explanator improves the R2 more than would be expected by chance.
- If a set of explanatory variables with a predetermined hierarchy of importance are introduced into a regression one at a time, with the adjusted R2 computed each time, the level at which adjusted R2 reaches a maximum, and decreases afterward, would be the regression with the ideal combination of having the best fit without excess/unnecessary terms.

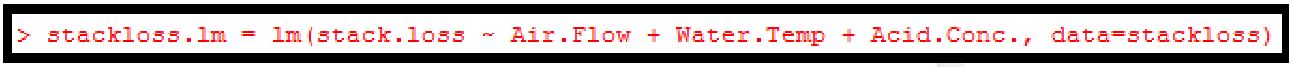
15.4. Significant Test for Multiple Linear Regression

Assume that the error term ε in the mulitple linear regression is independent of x_k (k =1,2,...p), and is normally distributed, with zero mean and constant variance. We can decide whether there is any significant relationship between the dependent variable y and any of the dependent variables of x_k (k =1,2,...p).

Problem 1

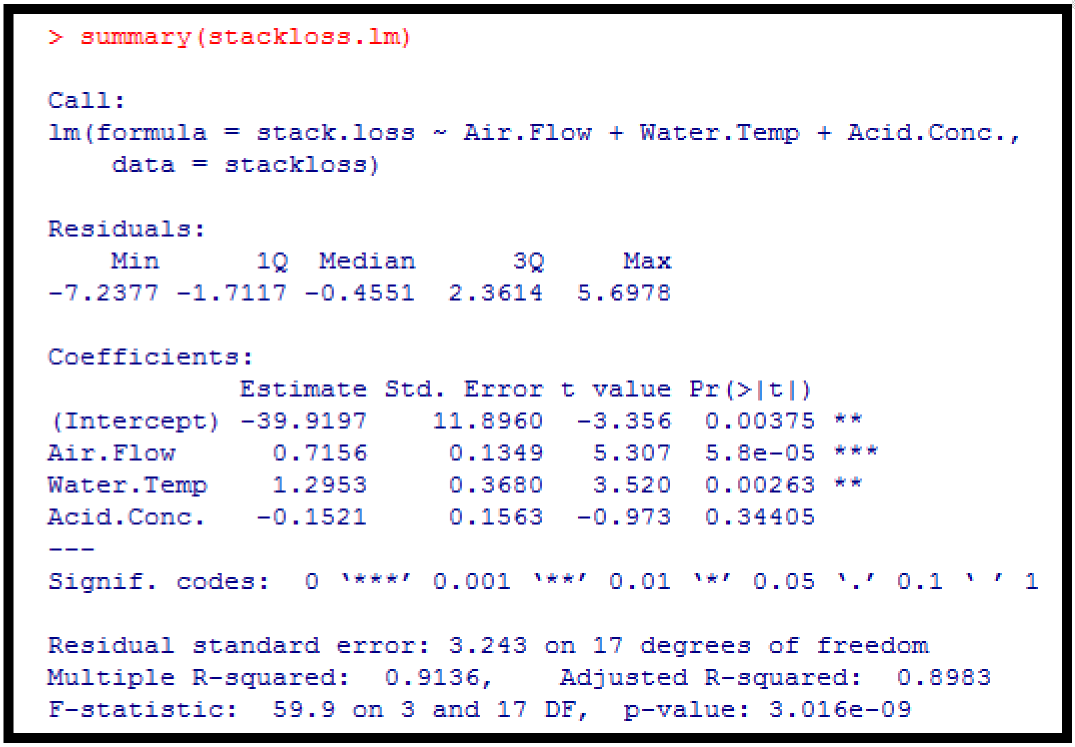
Decide which of the independent variables in the multiple linear regression model of the data set stackloss are statistically significant at .05 significance level.

Solution



```
> stackloss.lm = lm(stack.loss ~ Air.Flow + Water.Temp + Acid.Conc., data=stackloss)
```

The t values of the independent variables can be found with the summary function.



```
> summary(stackloss.lm)

Call:
lm(formula = stack.loss ~ Air.Flow + Water.Temp + Acid.Conc.,
    data = stackloss)

Residuals:
    Min      1Q  Median      3Q     Max 
-7.2377 -1.7117 -0.4551  2.3614  5.6978 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -39.9197    11.8960  -3.356  0.00375 ** 
Air.Flow      0.7156     0.1349   5.307  5.8e-05 ***
Water.Temp    1.2953     0.3680   3.520  0.00263 ** 
Acid.Conc.   -0.1521     0.1563  -0.973  0.34405    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.243 on 17 degrees of freedom
Multiple R-squared:  0.9136,    Adjusted R-squared:  0.8983 
F-statistic:  59.9 on 3 and 17 DF,  p-value: 3.016e-09
```

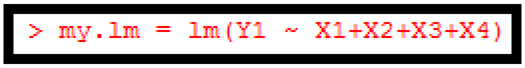
Conclusion

As the p-value of Aire Flow and Water.Temp are less than 0.05, they are both statistically significant in the multiple linear regression model of stackloss.

Problem 2

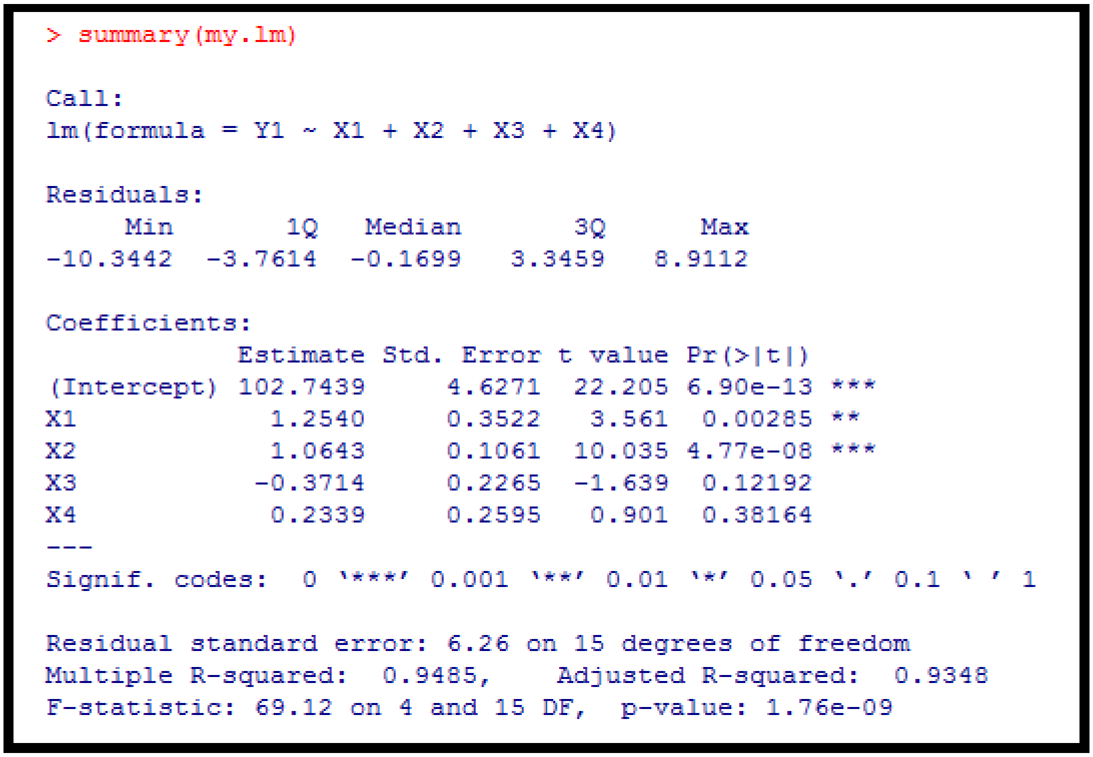
Decide which of the independent variables in the multiple linear regression model between (X_1,X_2,X_3 and X_4) and Y_1 are statistically significant at .05 significance level.

Solution



```
> my.lm = lm(Y1 ~ X1+X2+X3+X4)
```

The t values of the independent variables can be found with the summary function.



```
> summary(my.lm)

Call:
lm(formula = Y1 ~ X1 + X2 + X3 + X4)

Residuals:
     Min       1Q   Median       3Q      Max 
-10.3442  -3.7614  -0.1699   3.3459   8.9112 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 102.7439     4.6271  22.205 6.90e-13 ***
X1            1.2540     0.3522   3.561  0.00285 ** 
X2            1.0643     0.1061  10.035 4.77e-08 ***
X3           -0.3714     0.2265  -1.639  0.12192    
X4            0.2339     0.2595   0.901  0.38164    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.26 on 15 degrees of freedom
Multiple R-squared:  0.9485,    Adjusted R-squared:  0.9348 
F-statistic: 69.12 on 4 and 15 DF,  p-value: 1.76e-09
```

Conclusion

As the p-value of X_1 and X_2 are less than 0.05, they are both statistically significant in the multiple linear regression model.

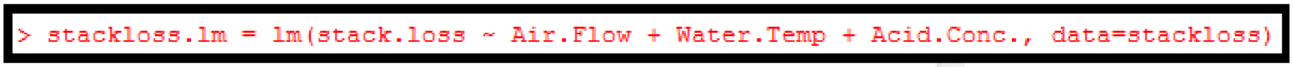
15.5. Confidence Interval for Multiple Linear Regression

Assume that the error term ε in the mulitple linear regression is independent of x_k (k =1,2,...p), and is normally distributed, with zero mean and constant variance. For a given set of values of x_k (k =1,2,...p), and the interval estimate for the mean of the dependent variable, is called the confidence interval.

Problem

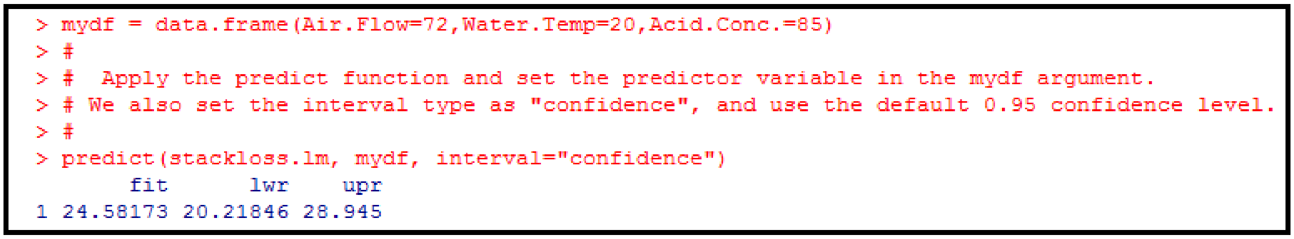
In the data set stackloss, develop a 95% confidence interval of the stack loss if the air flow is 72, water temperature is 20 and acid concentration is 85.

Solution



```
> stackloss.lm = lm(stack.loss ~ Air.Flow + Water.Temp + Acid.Conc., data=stackloss)
```

We wrap the parameters inside a new data frame, mydf.



```
> mydf = data.frame(Air.Flow=72,Water.Temp=20,Acid.Conc.=85)
> #
> #  Apply the predict function and set the predictor variable in the mydf argument.
> # We also set the interval type as "confidence", and use the default 0.95 confidence level.
> #
> predict(stackloss.lm, mydf, interval="confidence")
       fit      lwr    upr
1 24.58173 20.21846 28.945
```

Conclusion

The 95% confidence interval of the stack loss with the given parameters is between 20.21846 and 28.945.

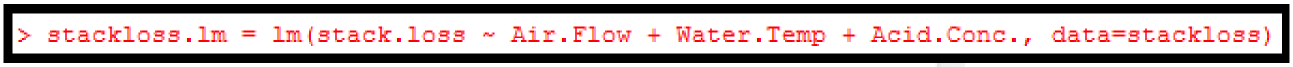
15.6. Prediction Interval for Multiple Linear Regression

Assume that the error term ε in the mulitple linear regression is independent of x_k (k =1,2,...p), and is normally distributed, with zero mean and constant variance. For a given set of values of x_k (k =1,2,...p), and the interval estimate for the dependent variable y is called the prediction interval.

Problem

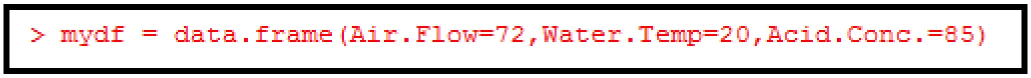
In the data set stackloss, develop a 95% prediction interval of the stack loss if the air flow is 72, water temperature is 20 and acid concentration is 85.

Solution



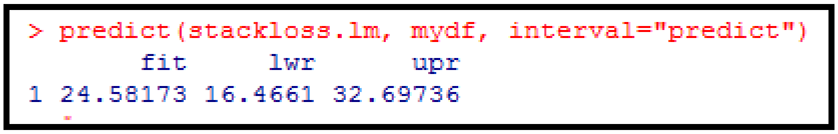
```
> stackloss.lm = lm(stack.loss ~ Air.Flow + Water.Temp + Acid.Conc., data=stackloss)
```

We wrap the parameters inside a new data frame, mydf.



```
> mydf = data.frame(Air.Flow=72,Water.Temp=20,Acid.Conc.=85)
```

Let us apply the predict function and set the predictor variable in the predictor variable in the mydf argument. We also set the interval type as :predict", and use the default 0.95 confidence level.



```
> predict(stackloss.lm, mydf, interval="predict")
       fit     lwr      upr
1 24.58173 16.4661 32.69736
```

Conclusion

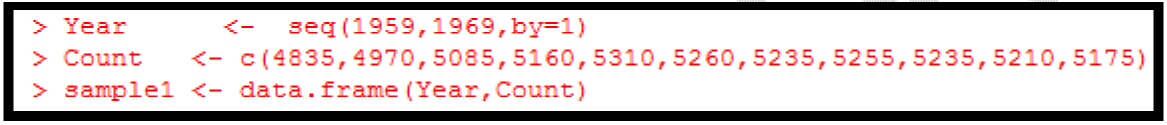
The 95% confidence interval of the stack loss with the given parameters is between 16.466 and 32.69736.

16. Polynomial Regression

Polynomial regression uses one independent variable x and a dependent variable y.

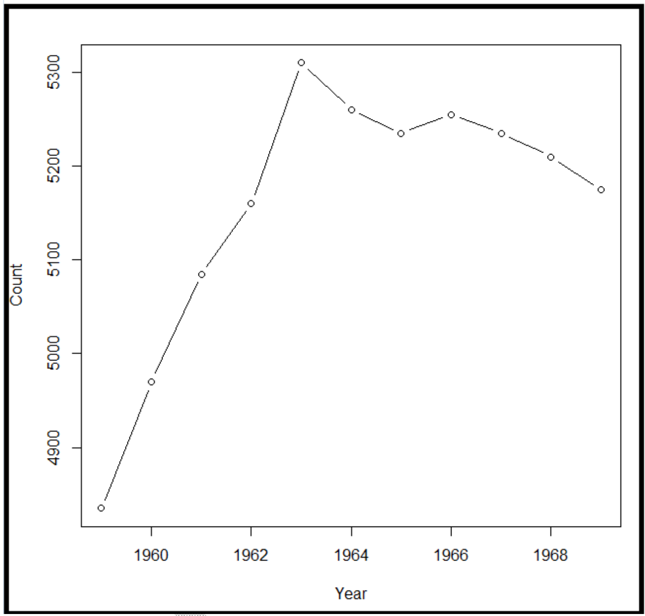
$Y = \alpha + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \ldots\ldots + \beta_n X^n + \varepsilon$

Assume we want to create a ploynomial that can approximate better the following dataset on the customers vising a popular restaurant in an Indian City in January over 10 years.



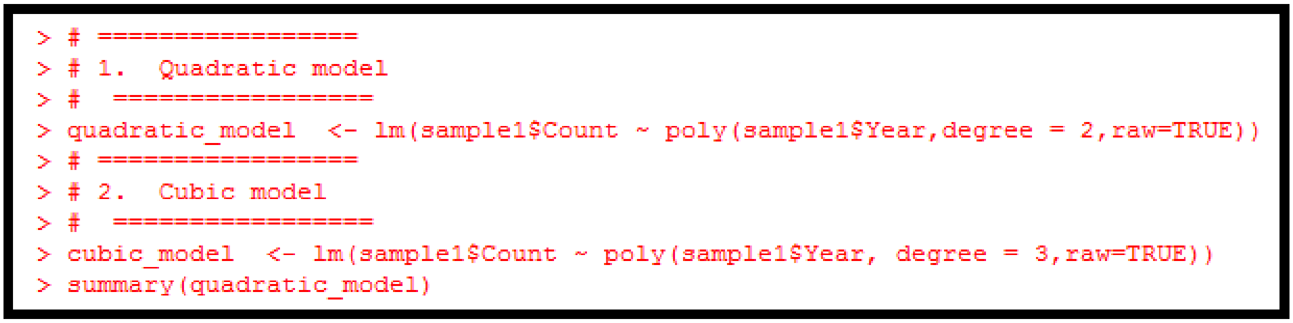
```
> Year     <-  seq(1959,1969,by=1)
> Count   <- c(4835,4970,5085,5160,5310,5260,5235,5255,5235,5210,5175)
> sample1 <- data.frame(Year,Count)
```

Use the scatter plot to check the nature of data.



From the above scatter plot we find that the data may not be linear.
We will explore the quadratic and cubic model.

a. Creating the models

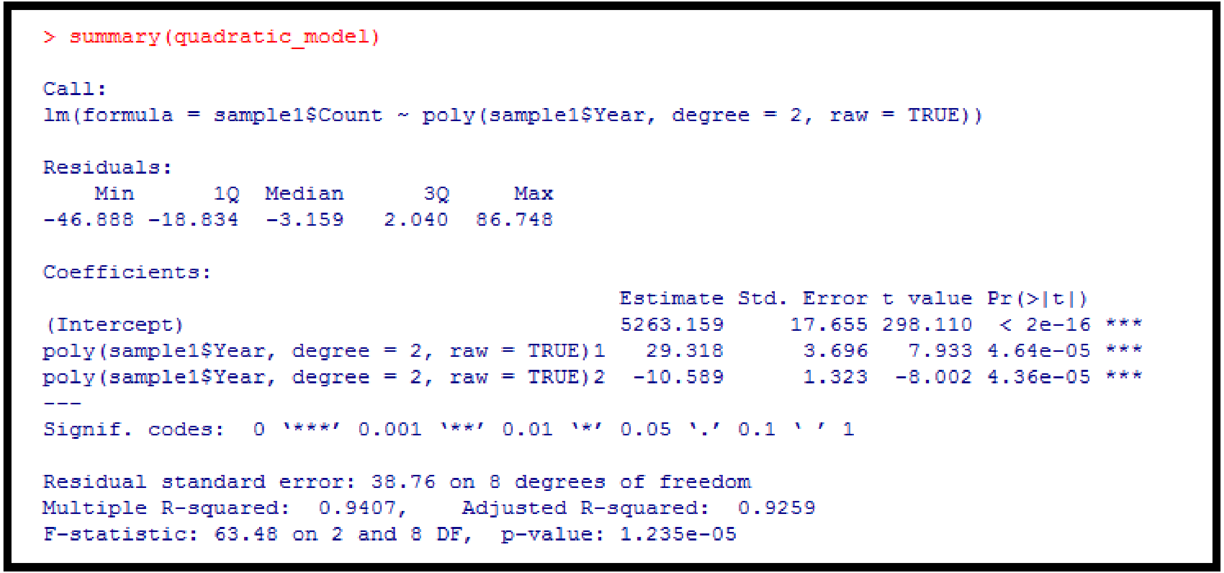


```
> # =================
> # 1.  Quadratic model
> #  =================
> quadratic_model  <- lm(sample1$Count ~ poly(sample1$Year,degree = 2,raw=TRUE))
> # =================
> # 2.  Cubic model
> #  =================
> cubic_model  <- lm(sample1$Count ~ poly(sample1$Year, degree = 3,raw=TRUE))
> summary(quadratic_model)
```

b. Evaluating the models

It is helpful to summarize and compare our potential models using the summary(MODEL) and anova(MODEL1,MODEL2,MODEL3) functions.

i. Quadratic Model



```
> summary(quadratic_model)

Call:
lm(formula = sample1$Count ~ poly(sample1$Year, degree = 2, raw = TRUE))

Residuals:
    Min      1Q  Median      3Q     Max 
-46.888 -18.834  -3.159   2.040  86.748 

Coefficients:
                                           Estimate Std. Error t value Pr(>|t|)    
(Intercept)                                5263.159     17.655 298.110  < 2e-16 ***
poly(sample1$Year, degree = 2, raw = TRUE)1   29.318      3.696   7.933 4.64e-05 ***
poly(sample1$Year, degree = 2, raw = TRUE)2  -10.589      1.323  -8.002 4.36e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 38.76 on 8 degrees of freedom
Multiple R-squared:  0.9407,	Adjusted R-squared:  0.9259 
F-statistic: 63.48 on 2 and 8 DF,  p-value: 1.235e-05
```

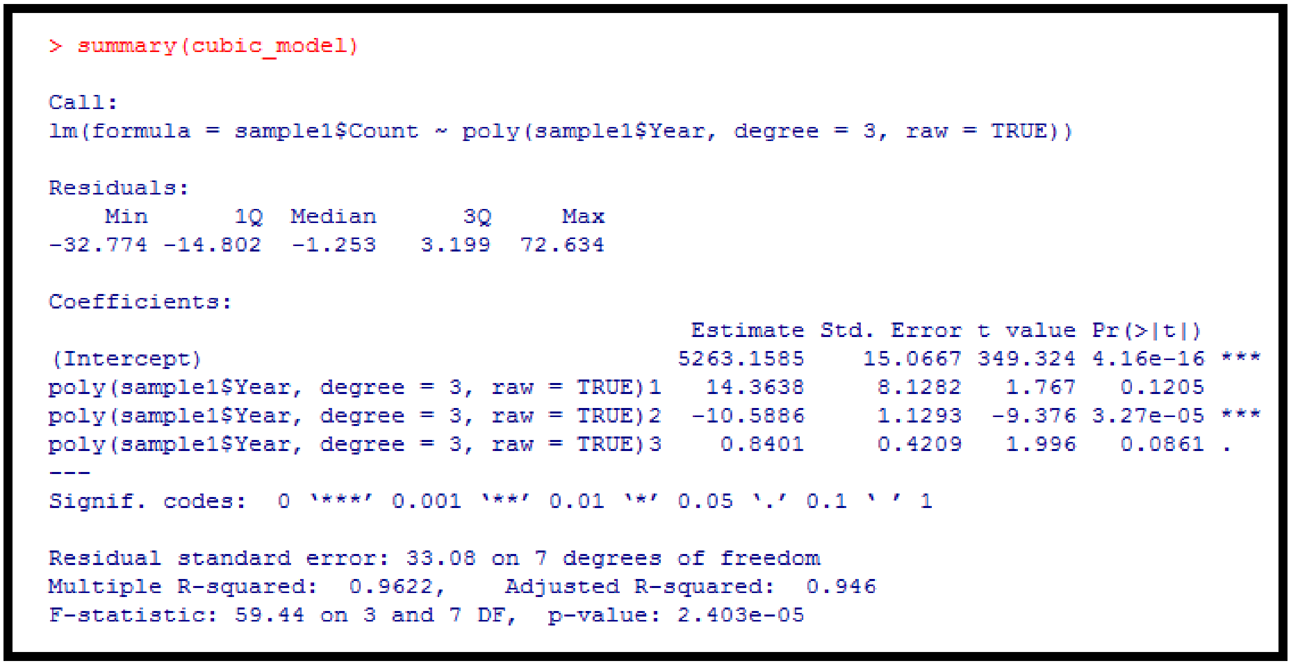
The output of summary(quadratic_model) is given above.

We observe the values of beta(5263.159), beta1(29.318) and beta2(-10.589), which appear to be significant and all are less than 0.05 significance level.

The equation of polynomial of degree 2 of our model is:

Count = 5263.1597 + 29.318 x – 10.589x^2

j. Cubic Model



```
> summary(cubic_model)

Call:
lm(formula = sample1$Count ~ poly(sample1$Year, degree = 3, raw = TRUE))

Residuals:
    Min      1Q  Median      3Q     Max 
-32.774 -14.802  -1.253   3.199  72.634 

Coefficients:
                                           Estimate Std. Error t value Pr(>|t|)    
(Intercept)                               5263.1585    15.0667 349.324 4.16e-16 ***
poly(sample1$Year, degree = 3, raw = TRUE)1   14.3638     8.1282   1.767   0.1205    
poly(sample1$Year, degree = 3, raw = TRUE)2  -10.5886     1.1293  -9.376 3.27e-05 ***
poly(sample1$Year, degree = 3, raw = TRUE)3    0.8401     0.4209   1.996   0.0861 .  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 33.08 on 7 degrees of freedom
Multiple R-squared:  0.9622,    Adjusted R-squared:  0.946 
F-statistic: 59.44 on 3 and 7 DF,  p-value: 2.403e-05
```

The output of summary(cubic_model) is given above.

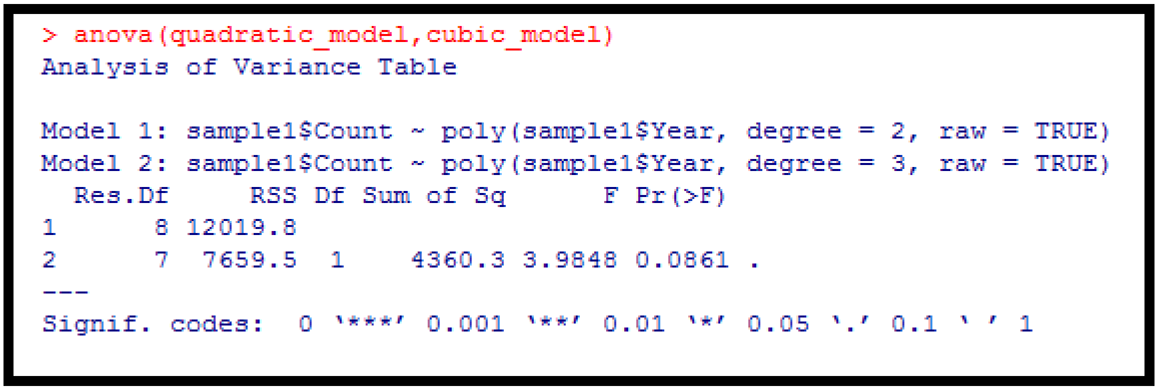
The equation of polynomial of degree 3 of our model is:

Count = 5263.1585 + 14.3638 x – 10.5886x^2 + 0.8401 x^3

We observe the values of beta(5263.1585), beta1(14.3638), beta2(-10.5886) and beta3(0.8401).

We observe that the coefficients beta1 and beta3 are not significant.

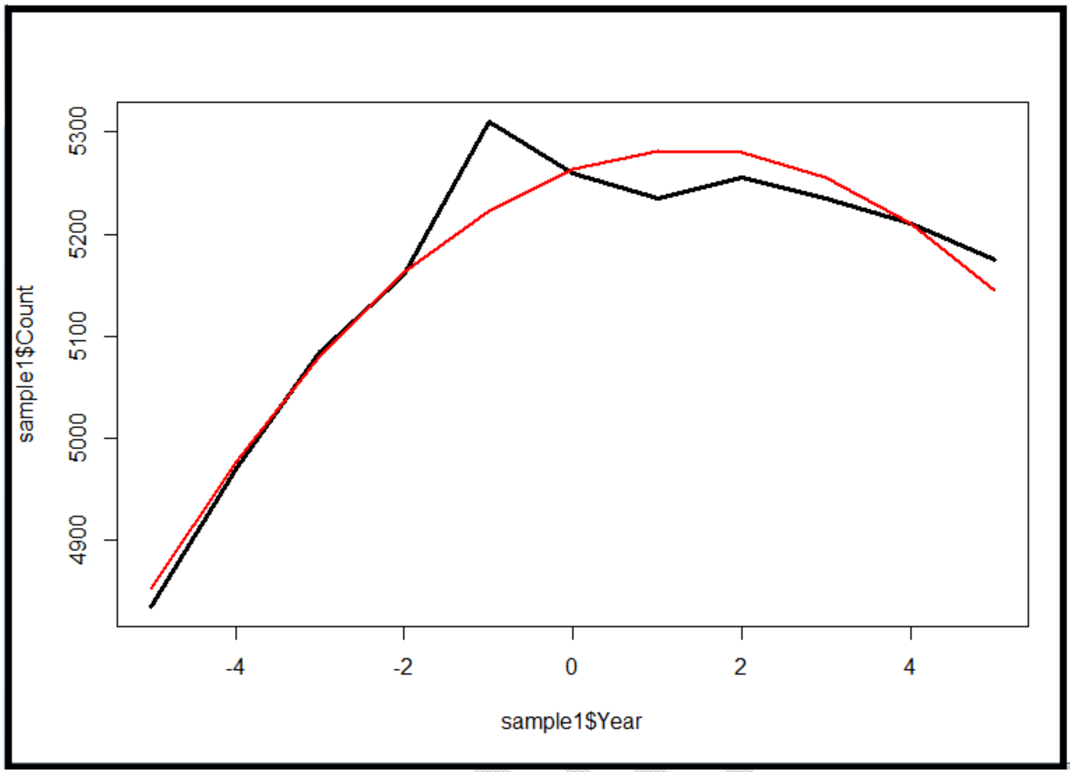
Multiple R –squared in the 2nd degree model is 94.07 % while in the 3rd degree model it is 96.22%. It seems that there has been an increase in accuracy of the model, but it is a significant increase? We can compare the two model with an ANOVA table.



```
> anova(quadratic_model,cubic_model)
Analysis of Variance Table

Model 1: sample1$Count ~ poly(sample1$Year, degree = 2, raw = TRUE)
Model 2: sample1$Count ~ poly(sample1$Year, degree = 3, raw = TRUE)
  Res.Df     RSS Df Sum of Sq      F Pr(>F)  
1      8 12019.8                             
2      7  7659.5  1    4360.3 3.9848 0.0861 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

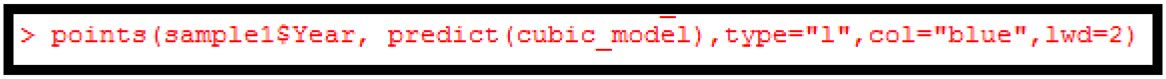
Since the p-value is greater than 0.05, we accept the null hypothesis: there wasn't a significant improvement of the model.

Now add to the scatter plot -we have already drawn the progress of 2nd degree polynomial.

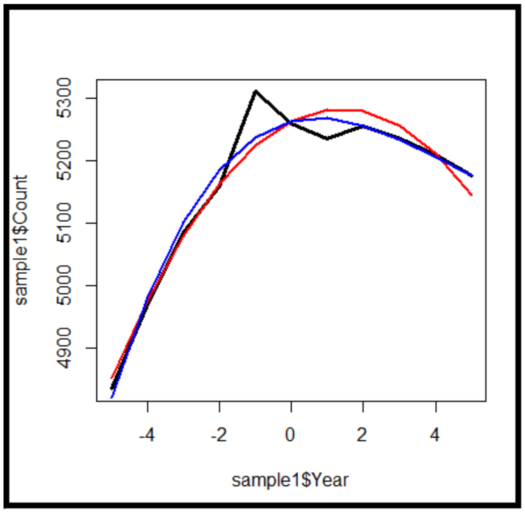


The function predict() compute the Y values given the X values. The coordinates are linked with the lines. With a few values this method is highly debilitating.

Let us add the graph of the polynomial of 3rd degree.



```
> points(sample1$Year, predict(cubic_model),type="l",col="blue",lwd=2)
```



The two models have very similar trends.

17. Logistic Regression

We use the logistic regression equation to predict the probability of a dependent variable taking the dichotomy values 0 or 1. Suppose $x_1, x_2, ... x_p$ are independent variables, α and β_k (k=1,2,...p) are parameters, and E(y) is the expected value of the dependent variable y, then the logistic regression equation is:

$E(y) = 1 / (1 + e^{-(\alpha + \Sigma\beta_k x_k)})$

For example, in the built-in data set mtcars, the data column am represents the transmission type of the automobile model (0 = automatic, 1 = manual). With the logistic regression equation, we can model the probability of a manual transmission in a vehicle based on its engine horsepower and weight data.

$P(\text{Manual Transmission}) = 1 / (1 + e^{-(\alpha + \beta_1 * \text{horse power} + \beta_2 * \text{weight})})$

17.1. Estimated Logistic Regression Equation

Using the generalized linear model, an estimated logistic regression equation can be formulated as below:

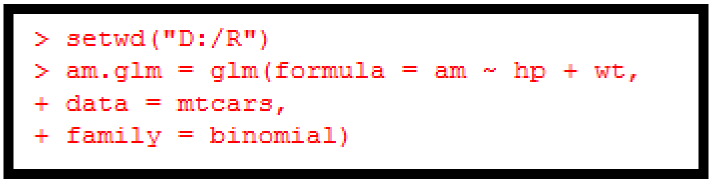
The coefficients a and b_k (k=1,2...,p) are determined according to a maximum likelihood approach, and it allows us to estimate the probability of the dependent variable y taking on the value 1 for the given values of x_k(k=1,2....p).

Problem

By use of the logistic regression equation of vehicle transmission in the data set mtcars, estimate the probability of a vehicle being fitted with a manual transmission if it has a 120 hp engine and weights 2800 lbs.

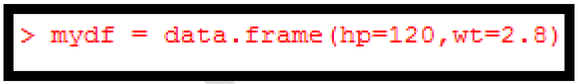
Solution

We apply the function glm to a formula that describes the transmission type (am) by the horsepower (hp) and weight (wt). This creates a generalized linear model (GLM) in the binomial family.



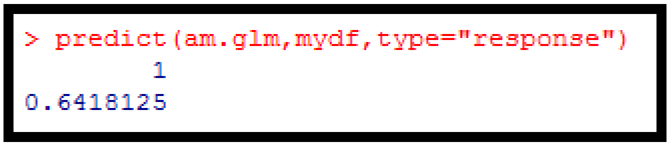
```
> setwd("D:/R")
> am.glm = glm(formula = am ~ hp + wt,
+ data = mtcars,
+ family = binomial)
```

We then wrap the test parameters inside a data frame mydf.



```
> mydf = data.frame(hp=120,wt=2.8)
```

Now we apply the function predict to the generalized linear model **am.glm** along with **mydf**. We will have to select *response* prediction type in order to obtain the predicted probability.



```
> predict(am.glm,mydf,type="response")
        1
0.6418125
```

Conclusion

For an automobile with 120 hp engine and 2800 lbs weight, the probability of it being fitted with a manual transmission is about 64%.

17.2. Significance Test for Logistic Regression Equation

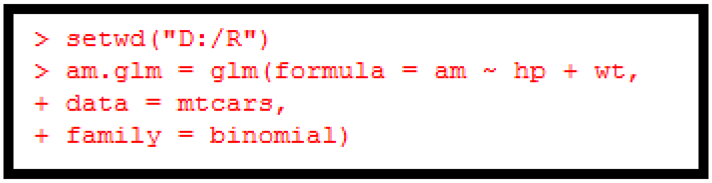
We can decide whether there is any significant relationship between the dependent variable y and the independent variables x_k (k = 1,2,...p) in the logistic regression equation. In particular, if any of the null hypothesis that β_k (k = 1,2,...p) is valid, then x_k is statistically insignificant in the logitic regression model.

Problem

At .05 significance level, decide if any of the independent variables in the logistic regression model of vechicle transmission in data set mtcars is statistically insignificant.

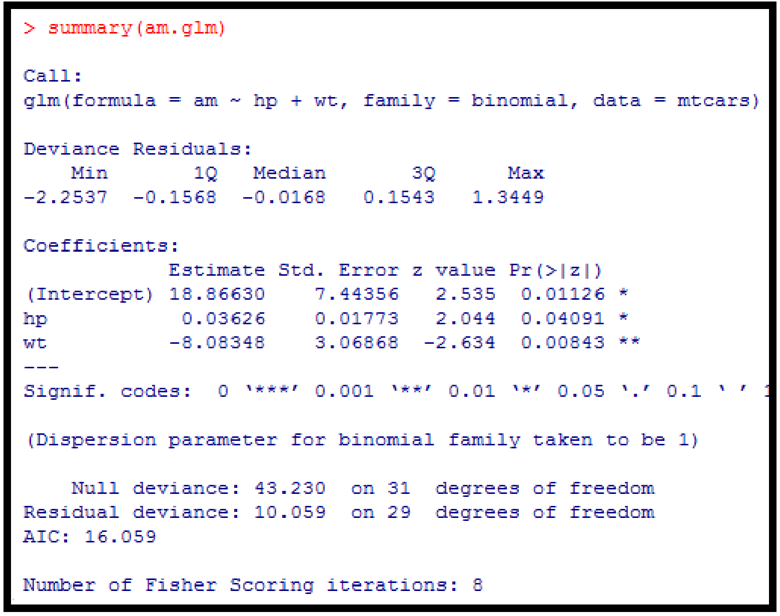
Solution

We apply the function glm to a formula that describes the transmission type (am) by the horsepower (hp) and weight (wt). This creates a generalized linear model (GLM) in the binomial family.



```
> setwd("D:/R")
> am.glm = glm(formula = am ~ hp + wt,
+ data = mtcars,
+ family = binomial)
```

We then print out the summary of the generalized linear model and check for the p-values of the hp and wt variables.



```
> summary(am.glm)

Call:
glm(formula = am ~ hp + wt, family = binomial, data = mtcars)

Deviance Residuals:
    Min       1Q   Median       3Q      Max  
-2.2537  -0.1568  -0.0168   0.1543   1.3449  

Coefficients:
            Estimate Std. Error z value Pr(>|z|)  
(Intercept) 18.86630    7.44356   2.535  0.01126 * 
hp           0.03626    0.01773   2.044  0.04091 * 
wt          -8.08348    3.06868  -2.634  0.00843 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 43.230  on 31  degrees of freedom
Residual deviance: 10.059  on 29  degrees of freedom
AIC: 16.059

Number of Fisher Scoring iterations: 8
```

Conclusion

As the p-values of the hp and wt variables are both less than 0.05, neither hp nor wt is insignificant in the logistic regression model.

17.3. Another example for Logistic Regression Equation

Ref: http://www.tatvic.com/blog/logistic-regression-with-r/

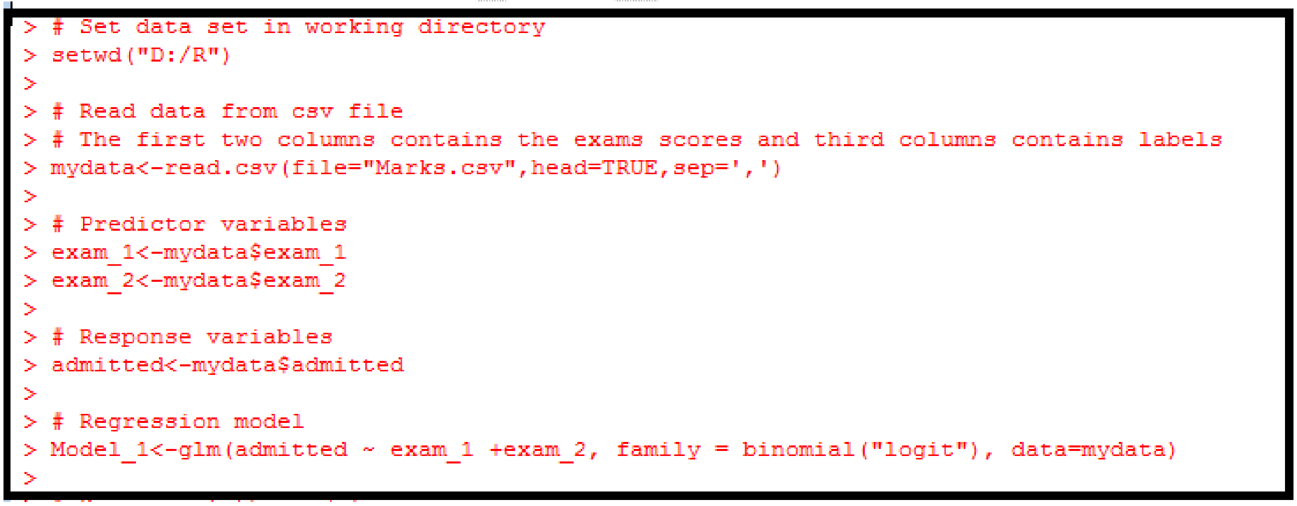
- Logistic regression is one of the type of regression and it is used to predict outcome of the categorical dependent variable (i.e. categorical variable has limited number of categorical values) based on the one or more independent variables.
- In binomial or binary logistic regression, the outcome can have only two possible types of values (e.g. "Yes","No","Success","Failure").
- Multinomial logistic refers to cases where the outcome can have three ot more possible types of values (e.g."Good" Vs "Very Good" Vs "Best"). Generally outcome is coded as "0" or "1" in binary logistic regression.

Suppose you want to predict whether a student will get admission based on his two exam scores. For this problem, we have a historical data from previous applicants which can be used as the dataset to build a model.

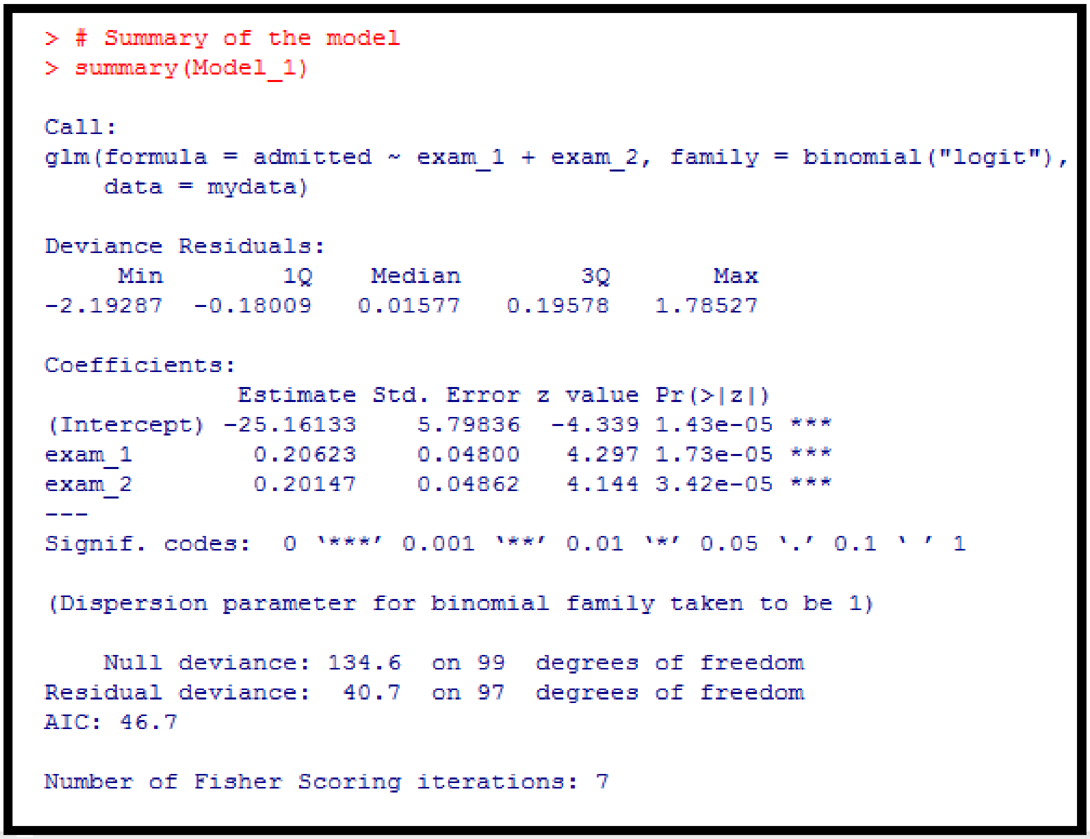
The data set contains:
1. exam_1 - Exam_1 score
2. exam_2- Exam_2 score
3. admitted - 1 if admitted or 0 if not admitted

In the above parametes, parameter admitted has value 1 or 0 for each observation. Now, we will generate a model that can predict, will student get admission based on two exam scores?

Here, admitted is considered as dependent variable, exam_1 and exam_2 are considered as independent variables.



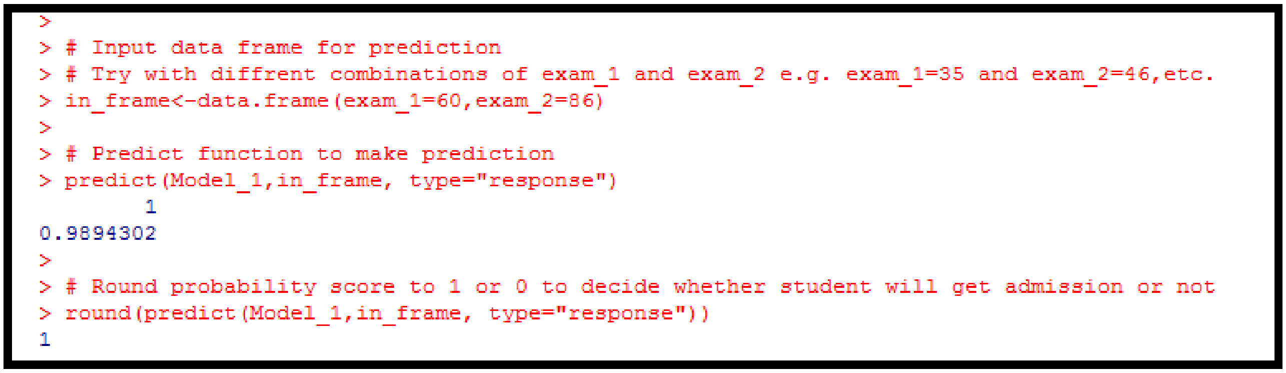
```
> # Set data set in working directory
> setwd("D:/R")
>
> # Read data from csv file
> # The first two columns contains the exams scores and third columns contains labels
> mydata<-read.csv(file="Marks.csv",head=TRUE,sep=',')
>
> # Predictor variables
> exam_1<-mydata$exam_1
> exam_2<-mydata$exam_2
>
> # Response variables
> admitted<-mydata$admitted
>
> # Regression model
> Model_1<-glm(admitted ~ exam_1 +exam_2, family = binomial("logit"), data=mydata)
>
```



```
> # Summary of the model
> summary(Model_1)

Call:
glm(formula = admitted ~ exam_1 + exam_2, family = binomial("logit"),
    data = mydata)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.19287  -0.18009   0.01577   0.19578   1.78527

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -25.16133    5.79836  -4.339 1.43e-05 ***
exam_1        0.20623    0.04800   4.297 1.73e-05 ***
exam_2        0.20147    0.04862   4.144 3.42e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 134.6  on 99  degrees of freedom
Residual deviance:  40.7  on 97  degrees of freedom
AIC: 46.7

Number of Fisher Scoring iterations: 7
```

After generating the model. let us try to predict using this model. Suppose we have two exam marks of a student, 60 of exam_1 and 85 of exam_2. We will predict that will student get admission?

Following the R code for predicting probability of student get admission.



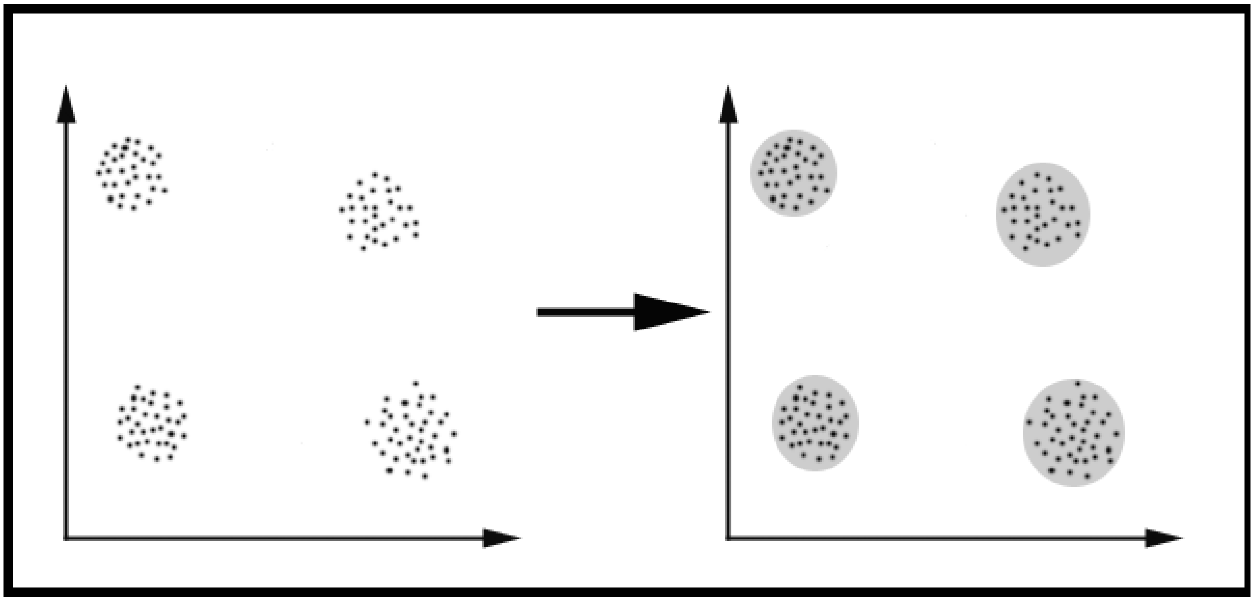
```
>
> # Input data frame for prediction
> # Try with diffrent combinations of exam_1 and exam_2 e.g. exam_1=35 and exam_2=46,etc.
> in_frame<-data.frame(exam_1=60,exam_2=86)
>
> # Predict function to make prediction
> predict(Model_1,in_frame, type="response")
        1
0.9894302
>
> # Round probability score to 1 or 0 to decide whether student will get admission or not
> round(predict(Model_1,in_frame, type="response"))
1
```

Conclusion:

Since the output is 1, the student will get admission.

18. Clustering in R

Clustering is the task of assigning a set of objects into groups (called clusters) so that the objects in the same cluster are more similar (in some sense or another) to each other than to those in other clusters.



A cluster is therefore a collection of objects which are “similar” between them and are “dissimilar” to the objects belonging to other clusters.

In the diagram, we easily identify the four clusters into which the data can be divided; the similarity criterion is distance; two or more objects belong to the same cluster if they are “close” according to a given distance (in this case geometrical distance). This is called distance-based clustering.

Another kind of clustering is conceptual clustering: two or more objects belong to the same cluster if this one defines a concept common to all that objects. In other words, objects are grouped according to their fit to descriptive concepts, not according to simple similarity measures.

The goals of clustering

To determine the intrinsic grouping in a set of unlabeled data

Some of them are:

1. Finding representatives for homogeneous groups (data reduction)
2. Finding natural clusters and describe their unknown properties (natural data types)
3. Finding useful and suitable groupings (useful data classes)
4. Finding unusual data objects (outlier detection)

19. Possible applications

Clustering algorithms can be applied in many fields, for instance:

- ***Marketing:*** finding groups of customers with similar behaviour given a large database of customer data containing their properties and past buying records
- ***Biology:*** Classification of plants and animals given their features
- ***Libraries:*** Book Ordering
- ***Insurance:*** Identifying groups of motor insurance policy holders with a high average claim cost; identifying frauds
- ***City-Planning:*** Identifying groups of houses according to their house type, value and geographical location.
- ***Earthquake studies:*** Clustering observed earthquake epicentres to identify dangerous zones

20. Clustering algorithms

1. Exclusive clustering
2. Overlapping clustering
3. Hierarchical clustering
4. Probabilistic clustering

The four most used clustering algorithms:

1 K-means
2 Fuzzy C-means
3 Hierarchical clustering
4 Mixture of Gaussians

Each of these algorithms belongs to one of the clustering types listed above. K-means is an exclusive clustering algorithm. Fuzzy C-means is an overlapping clustering algorithm. Hierarchical is obvious and lastly Mixture of Gaussian is a probabilistic clustering algorithm.

20.1. K-Means clustering

This is a prototype-based, partitional clustering technique that attempts find a number of specified clusters (k), which are presented by their centroids (mean).

K-means algorithm proceeds in such a way that the elements are assigned randomly to k clusters and the centroid (mean) is calculated for each cluster. In the next step, the elements are reassigned in such a manner that it belongs to the cluster with closest centroid. This process is iterated until two consecutive steps end up in the same assignment of elements. In R package, k-means clustering is done using the function kmeans().

Syntax:

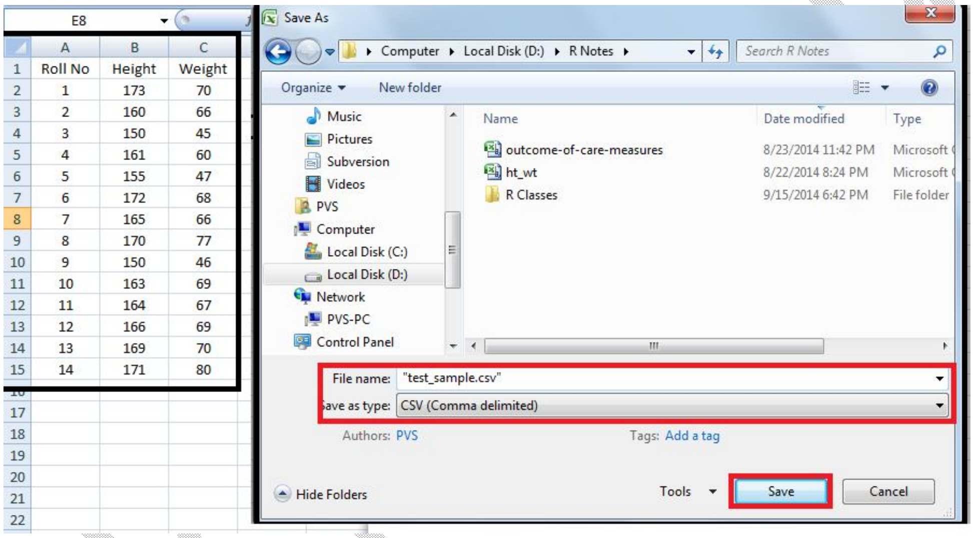
kmeans(x, centers, iter.max = 10, nstart = 1, algorithm = c(“Hartigan-Wong”,”Lloyd”,”Forgy”,”MacQueen”), trace= FALSE)

Arguments

x	Numeric matrix of data, or an object that can be coerced to such a matrix
centers	Either the number of clusters, say k or a set of initial (distinct) cluster centers. If a number, a random set of (distinct) rows in x is chosen as the initial centers.
iter.max	The maximum number of iterations allowed
nstart	If centers is a number, how many random set should be chosen
Algothithm	Determines the algorithm to be used
Trace	A logical value which product the tracing information on the progress of the algorithm, if TRUE

20.2. Example 1

Let us illustrate this with the help of an example. We shall calculate the k-means of the height and weight of fourteen students in the file test_sample.csv.



	A	B	C
1	Roll No	Height	Weight
2	1	173	70
3	2	160	66
4	3	150	45
5	4	161	60
6	5	155	47
7	6	172	68
8	7	165	66
9	8	170	77
10	9	150	46
11	10	163	69
12	11	164	67
13	12	166	69
14	13	169	70
15	14	171	80

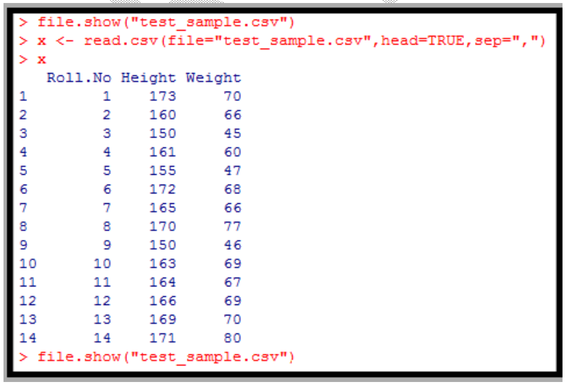
Name	Date modified	Type
outcome-of-care-measures	8/23/2014 11:42 PM	Microsoft
ht_wt	8/22/2014 8:24 PM	Microsoft
R Classes	9/15/2014 6:42 PM	File folder

File name: "test_sample.csv"

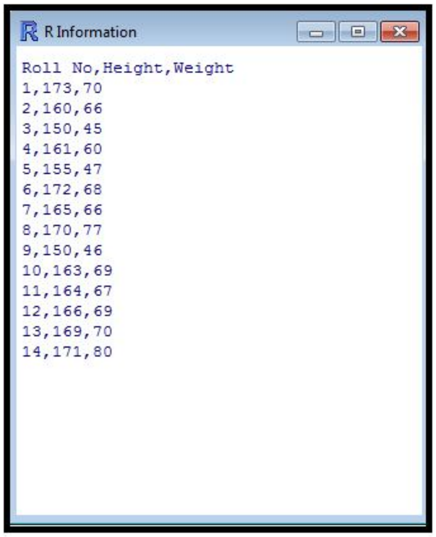
Save as type: CSV (Comma delimited)

Authors: PVS

Tags: Add a tag



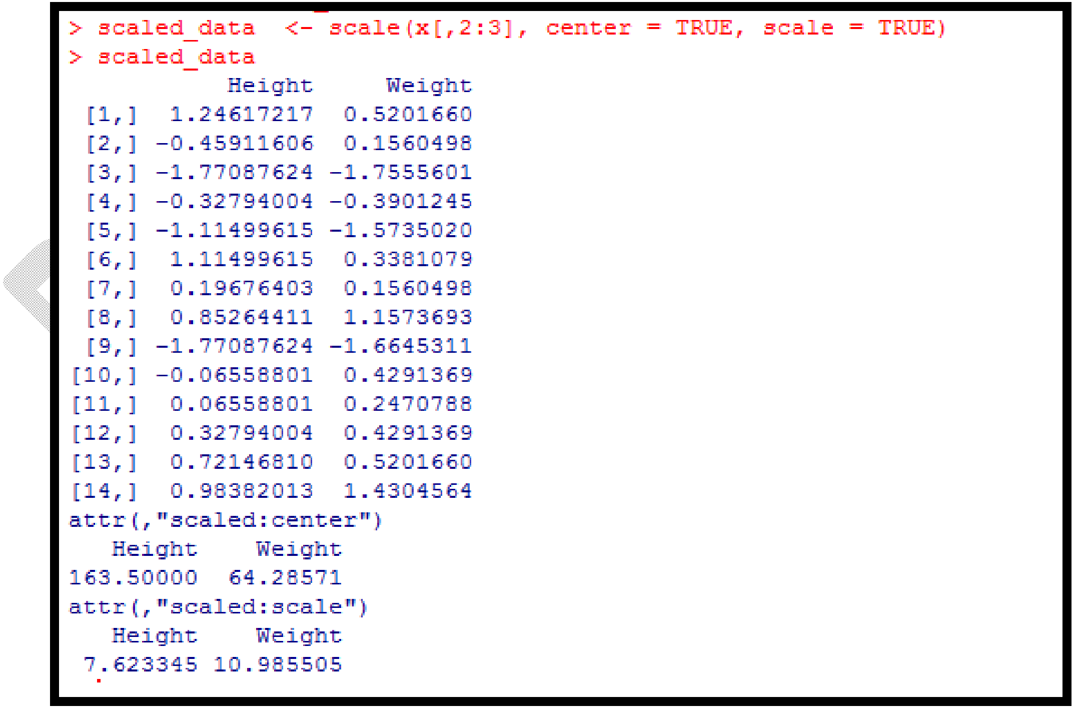
```
> file.show("test_sample.csv")
> x <- read.csv(file="test_sample.csv",head=TRUE,sep=",")
> x
   Roll.No Height Weight
1        1    173     70
2        2    160     66
3        3    150     45
4        4    161     60
5        5    155     47
6        6    172     68
7        7    165     66
8        8    170     77
9        9    150     46
10      10    163     69
11      11    164     67
12      12    166     69
13      13    169     70
14      14    171     80
> file.show("test_sample.csv")
```



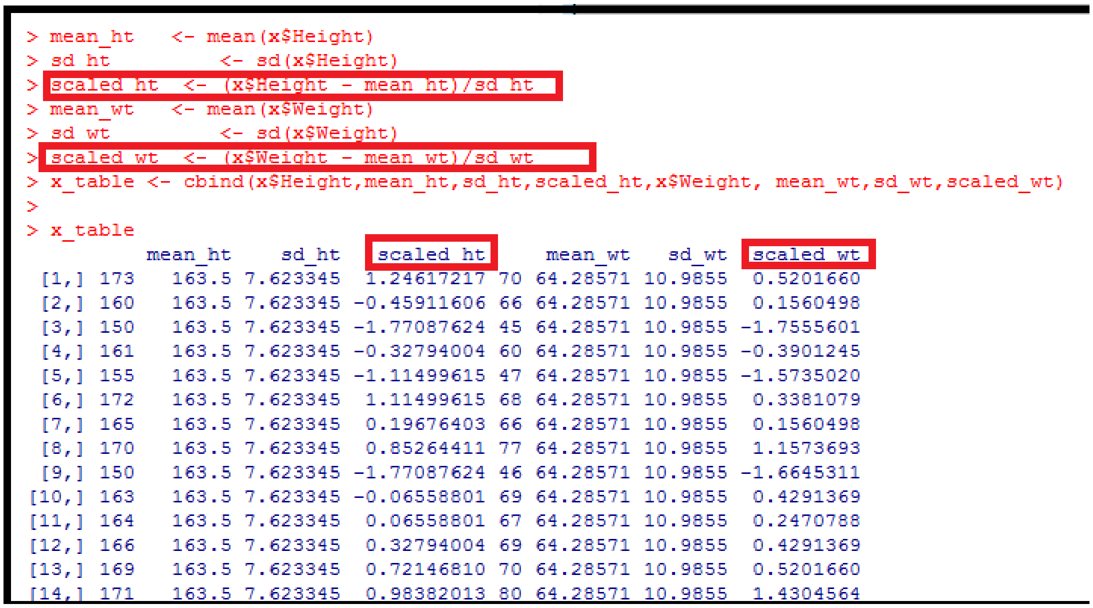
```
Roll No,Height,Weight
1,173,70
2,160,66
3,150,45
4,161,60
5,155,47
6,172,68
7,165,66
8,170,77
9,150,46
10,163,69
11,164,67
12,166,69
13,169,70
14,171,80
```

Since the data lies in different range, we normalize it using the function scale().

The function scale() centers and / or scales the columns of a numeric matrix. The attribute center determines whether centering should be done for the data. If center is TRUE, then centering is done. Similarly, the attribute scale determines whether scaling should be done for the centered data. Please note that the scaled height is arrived at by using the formula:
Height – mean(Height) / standard deviation (Height). Similarly, we do it for weight.



```
> scaled_data  <- scale(x[,2:3], center = TRUE, scale = TRUE)
> scaled_data
           Height     Weight
 [1,]  1.24617217  0.5201660
 [2,] -0.45911606  0.1560498
 [3,] -1.77087624 -1.7555601
 [4,] -0.32794004 -0.3901245
 [5,] -1.11499615 -1.5735020
 [6,]  1.11499615  0.3381079
 [7,]  0.19676403  0.1560498
 [8,]  0.85264411  1.1573693
 [9,] -1.77087624 -1.6645311
[10,] -0.06558801  0.4291369
[11,]  0.06558801  0.2470788
[12,]  0.32794004  0.4291369
[13,]  0.72146810  0.5201660
[14,]  0.98382013  1.4304564
attr(,"scaled:center")
   Height    Weight
163.50000  64.28571
attr(,"scaled:scale")
   Height    Weight
 7.623345 10.985505
```



```
> mean_ht    <- mean(x$Height)
> sd ht           <- sd(x$Height)
> scaled ht   <-  (x$Height - mean ht)/sd ht
> mean_wt    <- mean(x$Weight)
> sd wt           <- sd(x$Weight)
> scaled wt   <-  (x$Weight - mean wt)/sd wt
> x_table <- cbind(x$Height,mean_ht,sd_ht,scaled_ht,x$Weight, mean_wt,sd_wt,scaled_wt)
>
> x_table
          mean_ht    sd_ht   scaled ht     mean_wt    sd_wt  scaled wt
 [1,] 173   163.5 7.623345  1.24617217 70 64.28571 10.9855  0.5201660
 [2,] 160   163.5 7.623345 -0.45911606 66 64.28571 10.9855  0.1560498
 [3,] 150   163.5 7.623345 -1.77087624 45 64.28571 10.9855 -1.7555601
 [4,] 161   163.5 7.623345 -0.32794004 60 64.28571 10.9855 -0.3901245
 [5,] 155   163.5 7.623345 -1.11499615 47 64.28571 10.9855 -1.5735020
 [6,] 172   163.5 7.623345  1.11499615 68 64.28571 10.9855  0.3381079
 [7,] 165   163.5 7.623345  0.19676403 66 64.28571 10.9855  0.1560498
 [8,] 170   163.5 7.623345  0.85264411 77 64.28571 10.9855  1.1573693
 [9,] 150   163.5 7.623345 -1.77087624 46 64.28571 10.9855 -1.6645311
[10,] 163   163.5 7.623345 -0.06558801 69 64.28571 10.9855  0.4291369
[11,] 164   163.5 7.623345  0.06558801 67 64.28571 10.9855  0.2470788
[12,] 166   163.5 7.623345  0.32794004 69 64.28571 10.9855  0.4291369
[13,] 169   163.5 7.623345  0.72146810 70 64.28571 10.9855  0.5201660
[14,] 171   163.5 7.623345  0.98382013 80 64.28571 10.9855  1.4304564
```

The function k_means() will return an object of class kmeans with the following components:

cluster A vector of integers indicating the cluster to which each point is allocated.

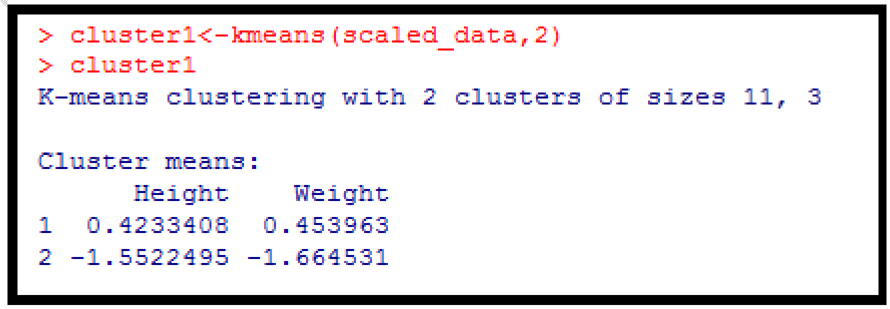
centers A matrix of clusters centers

cluster means Coordinates of the centroid (center) for each cluster

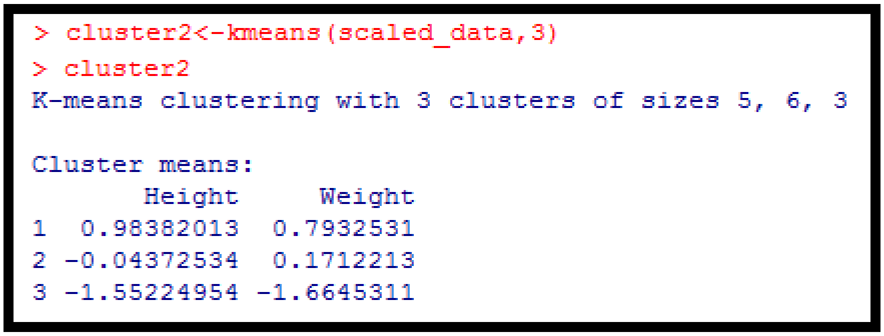
withinss The measure of the total variance in our data set explained by the clustering. This is the difference between the squared distances of each data point to the centroid and the sum of squared distances of each cluster to the centroid.

size The number of points in each cluster

Now, we can apply the k-means() function to our scaled data. The first kmeans() generates two clusters while the second one generates three clusters.

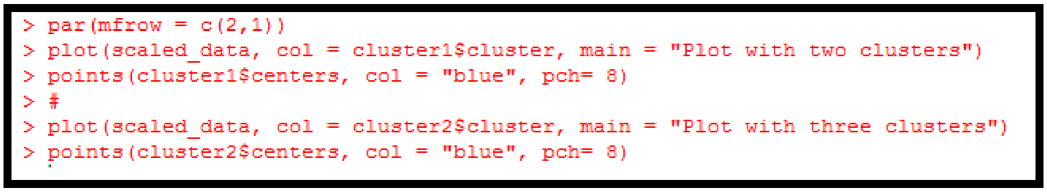


```
> cluster1<-kmeans(scaled_data,2)
> cluster1
K-means clustering with 2 clusters of sizes 11, 3

Cluster means:
      Height    Weight
1  0.4233408  0.453963
2 -1.5522495 -1.664531
```

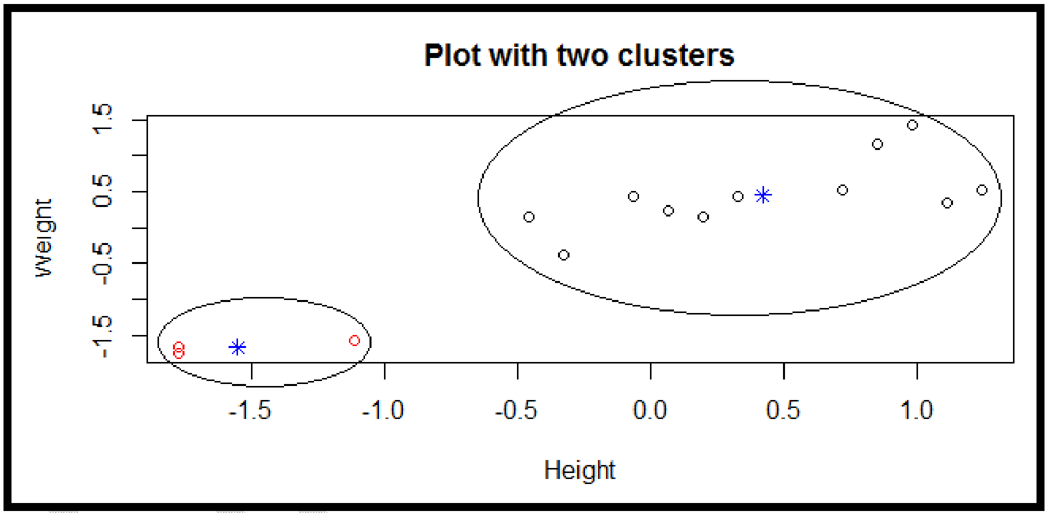


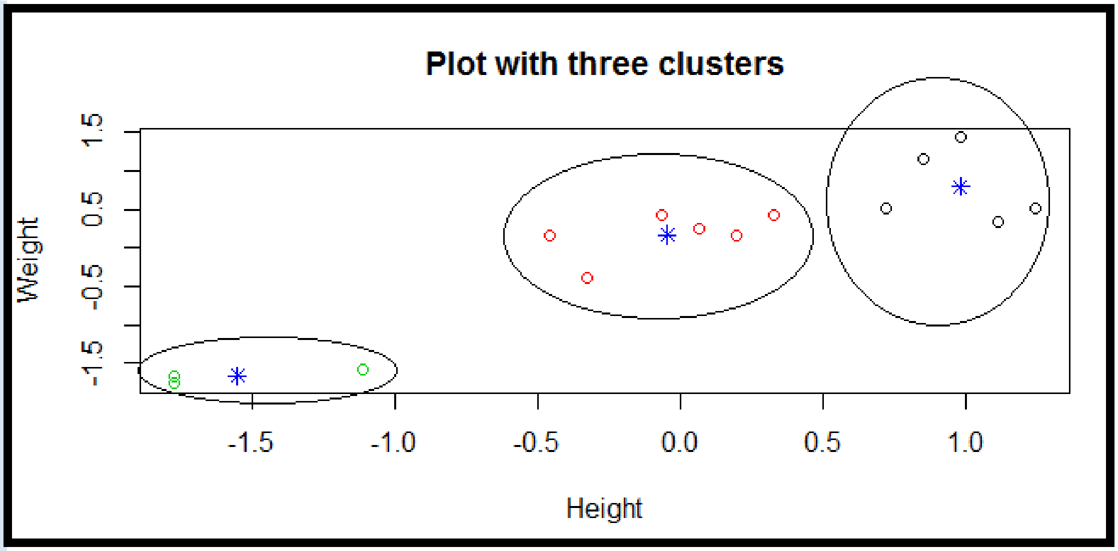
```
> cluster2<-kmeans(scaled_data,3)
> cluster2
K-means clustering with 3 clusters of sizes 5, 6, 3

Cluster means:
       Height     Weight
1  0.98382013  0.7932531
2 -0.04372534  0.1712213
3 -1.55224954 -1.6645311
```

To visualize the results of the k-means, generate plots.



```
> par(mfrow = c(2,1))
> plot(scaled_data, col = cluster1$cluster, main = "Plot with two clusters")
> points(cluster1$centers, col = "blue", pch= 8)
> #
> plot(scaled_data, col = cluster2$cluster, main = "Plot with three clusters")
> points(cluster2$centers, col = "blue", pch= 8)
```





Note:

1 The function par() enables to combine the plots of cluster1 and cluster2 in one single graph. Its attribute mfrow with values 2 and 1 enables to create a matrix of 2 rows and 1 column filled by row.

2 The function points() will indicate the location of the centroid in the plot. The attribute pch of the points() function denotes the plotting character to be used. Its value will be an integer code indicating the symbol.

3 The attribute col is meant for colour of the symbol.

4 We have circled the cluster of students having similar height and weight.

20.3. Example 2

Consider the following data set consisting of the scores of two variables on each of seven individuals:

Subject	A	B
1	1.0	1.0
2	1.5	2.0
3	3.0	4.0
4	5.0	7.0
5	3.5	5.0
6	4.5	5.0
7	3.5	4.5

This data set is to be grouped into two clusters. As a first step in finding an initial partition, let the A & B values of the two individuals furthest apart (using the Euclidean distance measure), define the initial cluster means, giving:

	Individual	Mean Vector (Centroid)
Group 1	1	(1.0, 1.0)
Group 2	4	(5.0, 7.0)

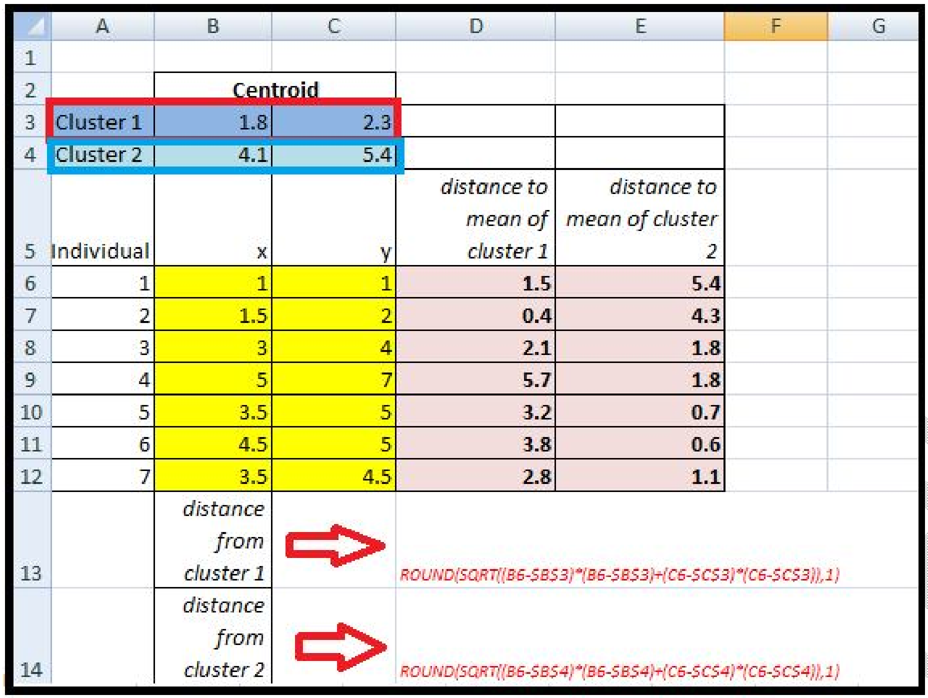
The remaining individuals are now examined in sequence and allocated to the cluster to which they are closest, in terms of Euclidean distance to the cluster mean. The mean vector is recalculated each time a new member is added. Now, we arrive at the following steps:

	Cluster 1		Cluster 2	
Step	Individual	Mean Vector (Centroid)	Individual	Mean Vector (Centroid)
1	1	(1.0, 1.0)	4	(5.0, 7.0)
2	1,2	(1.2,1.5)	4	(5.0, 7.0)
3	1,2,3	(1.8,2.3)	4	(5.0, 7.0)
4	1,2,3	(1.8,2.3)	4,5	(4.2,6.0)
5	1,2,3	(1.8,2.3)	4,5,6	(4.3,5.7)
6	1,2,3	(1.8,2.3)	4,5,6,7	(4.1,5.7)

Now the initial partition has changed, and the two clusters at this stage having the following characteristics:

	Individual	Mean Vector (Centroid)
Cluster 1	1,2,3	(1.8,2.3)
Cluster 2	4,5,6,7	(4.1, 5.4)

But, we cannot yet be sure that each individual has been assigned to the right cluster. So, we compare each individual's distance to its own cluster mean and to that of the opposite cluster.

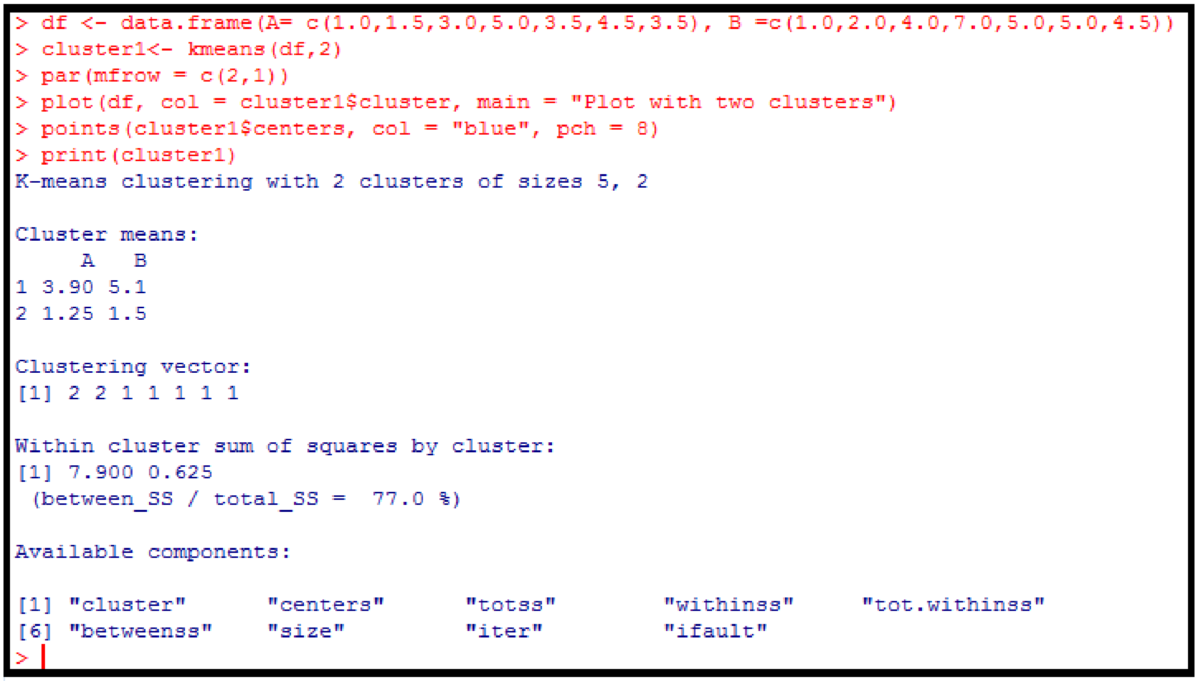


	A	B	C	D	E	F	G
1							
2		Centroid					
3	Cluster 1	1.8	2.3				
4	Cluster 2	4.1	5.4				
5	Individual	x	y	distance to mean of cluster 1	distance to mean of cluster 2		
6	1	1	1	1.5	5.4		
7	2	1.5	2	0.4	4.3		
8	3	3	4	2.1	1.8		
9	4	5	7	5.7	1.8		
10	5	3.5	5	3.2	0.7		
11	6	4.5	5	3.8	0.6		
12	7	3.5	4.5	2.8	1.1		
13		distance from cluster 1	⇨	ROUND(SQRT((B6-B3)*(B6-B3)+(C6-C3)*(C6-C3)),1)			
14		distance from cluster 2	⇨	ROUND(SQRT((B6-B4)*(B6-B4)+(C6-C4)*(C6-C4)),1)			

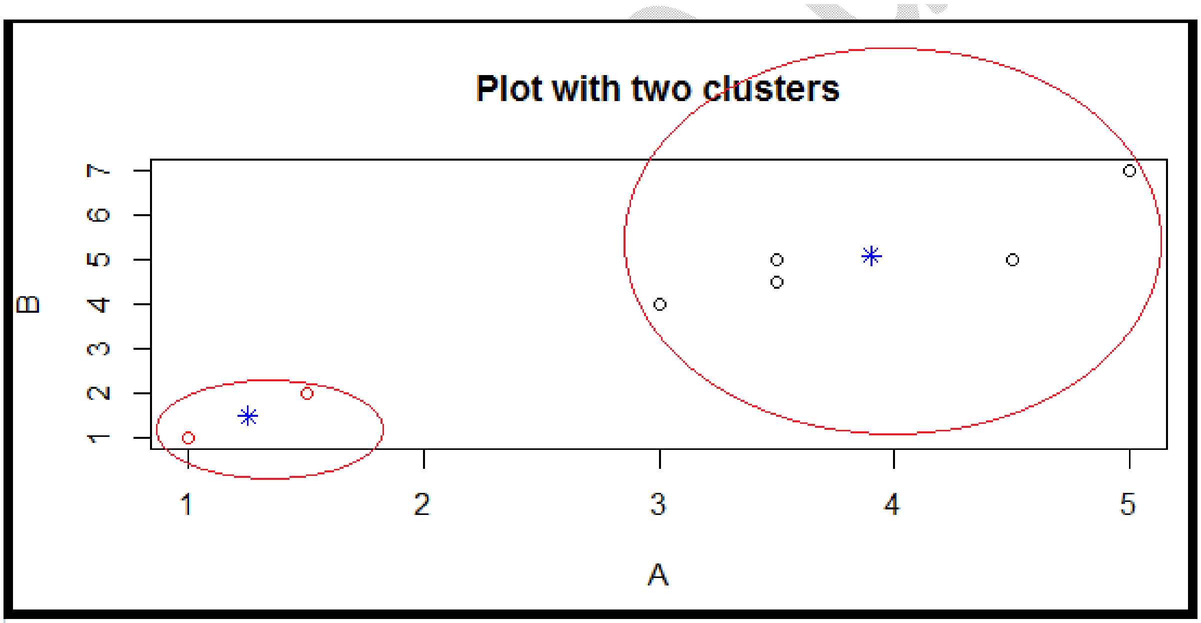
Only individual 3 is nearer to the mean of the opposite cluster (Cluster 2) than its own (Cluster 1). In other words, each individual's distance to its own cluster mean should be smaller that the distance to the other cluster's mean. It is different case for individual 3. Thus, individual 3 is relocated to Cluster 2 resulting in the new partition:

	Individual	Mean Vector (Centroid)
Cluster 1	1,2	(1.3,1.5)
Cluster 2	3,4,5,6,7	(3.9, 5.1)

The iterative relocation would now continue from this new partition until no more relocations occur. However, in this example, each individual is now nearer to its own cluster mean than that of the other cluster and the iteration stops, choosing the latest partitioning as the final cluster solution.



```
> df <- data.frame(A= c(1.0,1.5,3.0,5.0,3.5,4.5,3.5), B =c(1.0,2.0,4.0,7.0,5.0,5.0,4.5))
> cluster1<- kmeans(df,2)
> par(mfrow = c(2,1))
> plot(df, col = cluster1$cluster, main = "Plot with two clusters")
> points(cluster1$centers, col = "blue", pch = 8)
> print(cluster1)
K-means clustering with 2 clusters of sizes 5, 2

Cluster means:
     A   B
1 3.90 5.1
2 1.25 1.5

Clustering vector:
[1] 2 2 1 1 1 1 1

Within cluster sum of squares by cluster:
[1] 7.900 0.625
 (between_SS / total_SS =  77.0 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
> 
```


Plot with two clusters
B
A