

Basic statistics

A Short Course on Data Analysis Using R Software

Wan Nor Arifin

Unit of Biostatistics and Research Methodology,
Universiti Sains Malaysia.

Email: wnarifin@usm.my
Website: wnarifin.github.io



©Wan Nor Arifin under the Creative Commons Attribution-ShareAlike 4.0 International License.

Contents

1	Comparison of Numerical Data	1
1.1	Two independent samples	1
1.2	Two dependent samples	6
1.3	More than two independent samples	10
2	Comparison of Categorical Data	18
2.1	Two independent samples	18
2.2	Two dependent samples	20
3	Correlations	22
3.1	Pearson's correlation	22
3.2	Spearman's correlation	28
	References	28

1 Comparison of Numerical Data

1.1 Two independent samples

1.1.1 Independent t -test

About the test:

- Parametric test.
- Normally distributed data per group.
- Comparison of means between TWO groups.
- t -statistics.

Analysis:

1. Load `cholest.sav` dataset,

```
library(foreign)
cholest = read.spss("cholest.sav", to.data.frame = TRUE)
str(cholest)

## 'data.frame': 80 obs. of 5 variables:
## $ chol : num 6.5 6.6 6.8 6.8 6.9 7 7 7.2 7.2 7.2 ...
## $ age : num 38 35 39 36 31 38 33 36 40 34 ...
## $ exercise: num 6 5 6 5 4 4 5 5 4 6 ...
## $ sex : Factor w/ 2 levels "female","male": 2 2 2 2 2 2 2 2 2 2 ...
## $ categ : Factor w/ 3 levels "Grp A","Grp B",...: 1 1 1 1 1 1 1 1 1 1 ...
## - attr(*, "variable.labels")= Named chr "cholesterol in mmol/L" "age in year" "duration of exercise" ...
## ..- attr(*, "names")= chr "chol" "age" "exercise" "sex" ...
## - attr(*, "codepage")= int 65001
```

```
head(cholest)
```

```
## chol age exercise sex categ
## 1 6.5 38 6 male Grp A
## 2 6.6 35 5 male Grp A
## 3 6.8 39 6 male Grp A
## 4 6.8 36 5 male Grp A
## 5 6.9 31 4 male Grp A
## 6 7.0 38 4 male Grp A
```

Explore the data. Obtain the basic descriptive statistics.

Mean and SD,

```
by(cholest$chol, cholest$sex, mean)
```

```
## cholest$sex: female
## [1] 8.9275
## -----
## cholest$sex: male
## [1] 7.5325
```

```
by(cholest$chol, cholest$sex, sd)
```

```
## cholest$sex: female
## [1] 0.4551627
## -----
## cholest$sex: male
## [1] 0.4687066
```

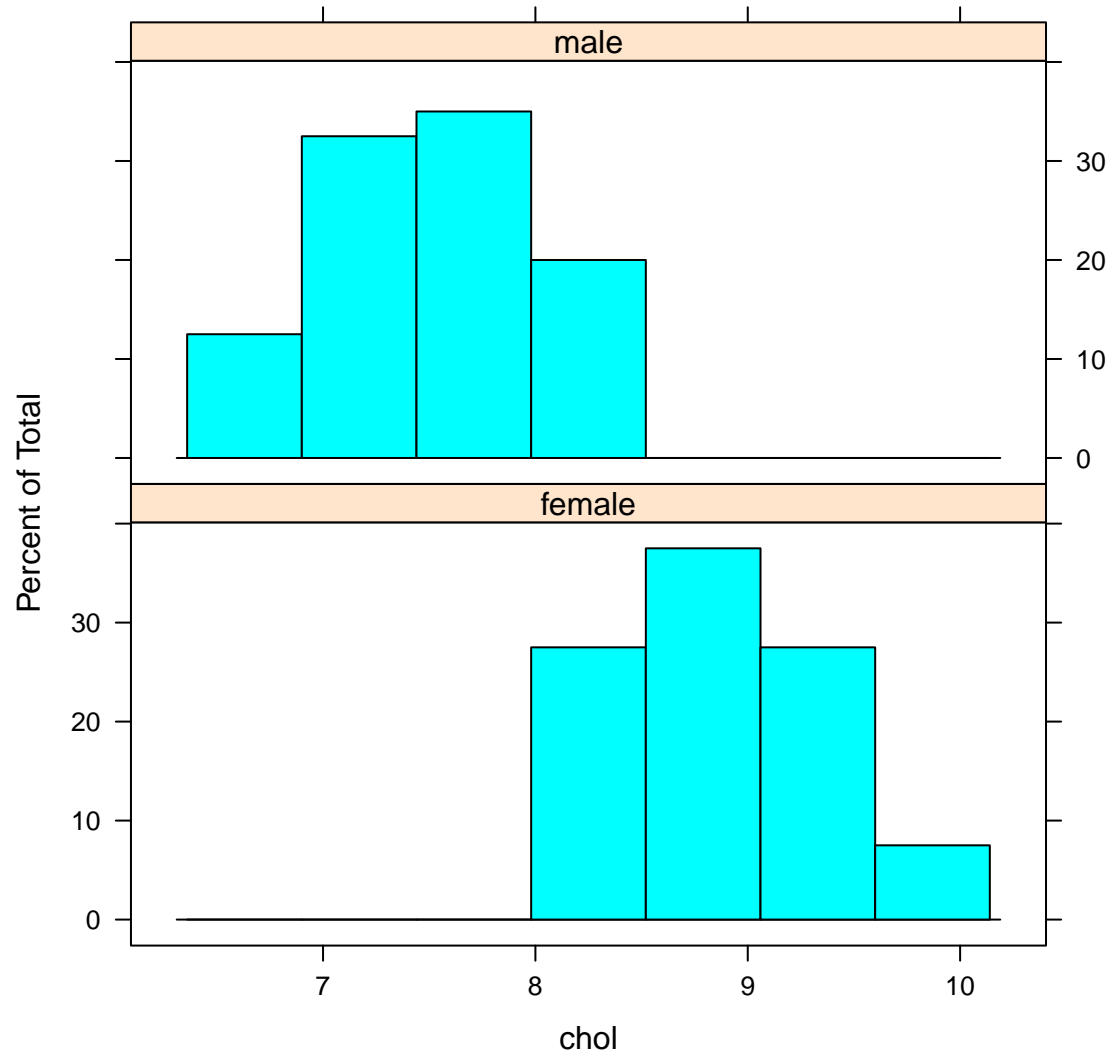
and the number of subjects per group,

```
table(cholest$sex)
```

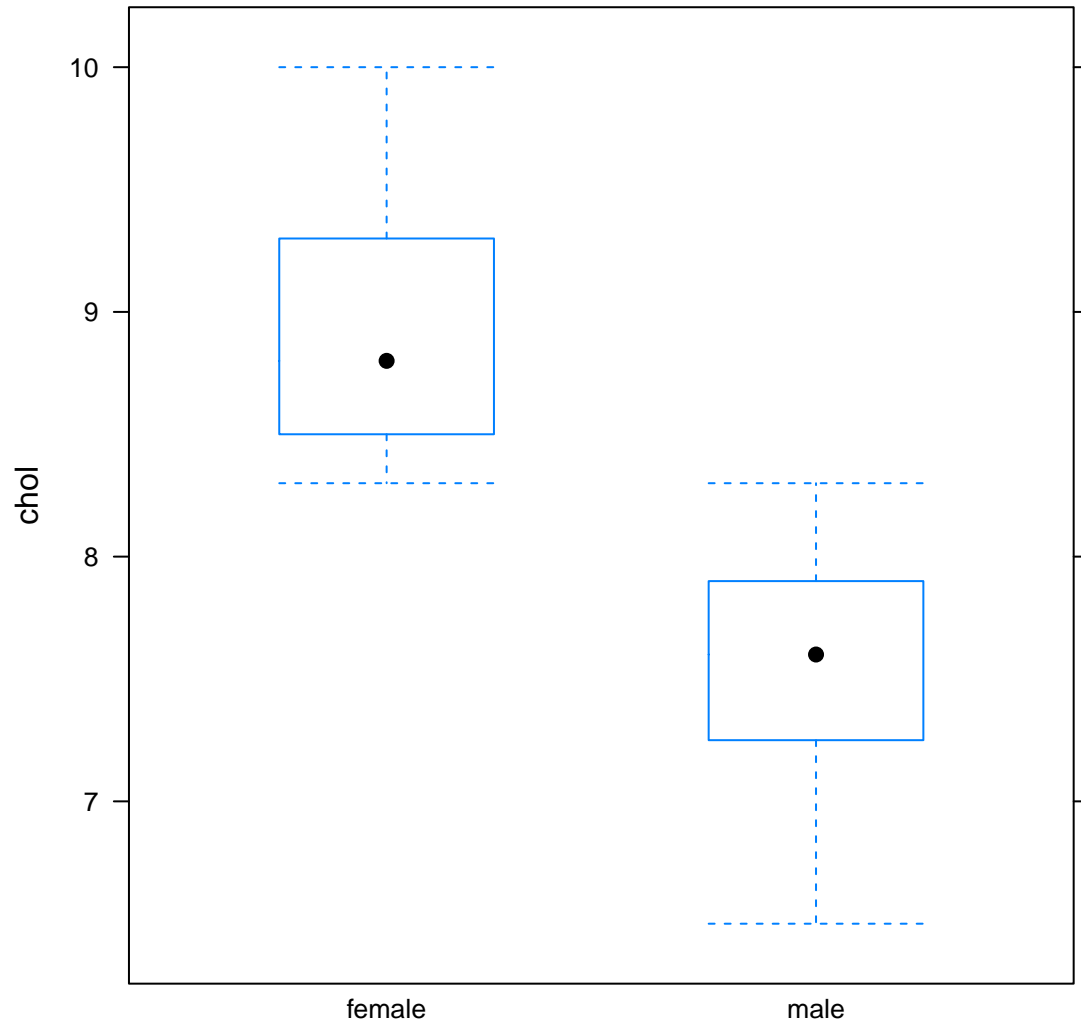
```
##
## female male
## 40 40
```

2. Check the **normality assumption** of the data by group,

```
library(lattice)
histogram(~ chol | sex, data = cholest, layout = c(1, 2))
```



```
bwplot(chol ~ sex, data = cholest)
```



3. Check the equality of variance assumption,

```
var.test(chol ~ sex, data = cholest) # equal*
```

```
##
## F test to compare two variances
##
## data: chol by sex
## F = 0.94304, num df = 39, denom df = 39, p-value = 0.8556
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
## 0.4987744 1.7830278
## sample estimates:
## ratio of variances
## 0.9430422
```

*Choose:

- Equal variance = Standard Two Sample *t*-test.

- Unequal variance = Welch Two Sample t -test.

4. Perform independent t -test,

```
t.test(chol ~ sex, data = cholest) # significant difference
```

```
##
## Welch Two Sample t-test
##
## data: chol by sex
## t = 13.504, df = 77.933, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  1.189337 1.600663
## sample estimates:
## mean in group female    mean in group male
##           8.9275           7.5325
```

The function default is **Welch Two Sample t -test** (takes care of the unequal variance).

You can also obtain the standard t -test (equal variance assumed),

```
t.test(chol ~ sex, data = cholest, var.equal = TRUE)
```

```
##
## Two Sample t-test
##
## data: chol by sex
## t = 13.504, df = 78, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  1.18934 1.60066
## sample estimates:
## mean in group female    mean in group male
##           8.9275           7.5325
```

1.1.2 Mann-Whitney U test (Wilcoxon rank-sum test)

About the test:

- Non-parametric test.
- Data are not normally distributed.
- Suitable for ordinal data.
- Involves ranking all observations (regardless of groups) and obtaining the sums per group.
- W -statistics.

Analysis:

1. Obtain descriptive statistics for non-normal data, median and IQR,

```
by(cholest$chol, cholest$sex, median)
```

```
## cholest$sex: female
## [1] 8.8
## -----
## cholest$sex: male
```

```
## [1] 7.6
by(cholest$chol, cholest$sex, IQR)

## cholest$sex: female
## [1] 0.8
## -----
## cholest$sex: male
## [1] 0.625
```

2. Perform Mann-Whitney U test,

```
wilcox.test(chol ~ sex, data = cholest, exact = FALSE)

##
## Wilcoxon rank sum test with continuity correction
##
## data: chol by sex
## W = 1598, p-value = 1.568e-14
## alternative hypothesis: true location shift is not equal to 0
```

1.2 Two dependent samples

1.2.1 Paired *t*-test

About the test:

- Parametric test.
- Normally distributed DIFFERENCES between TWO paired observations (e.g. *SBP after* – *SBP before*).
- Compares the observations between TWO time points for any changes (e.g. any changes after treatment?).
- If there are changes, the differences between the time points $\neq 0$.
- *t*-statistics.

Analysis:

1. Load `sbp.csv` dataset,

```
sbp = read.csv("sbp.csv")
str(sbp)

## 'data.frame': 11 obs. of 2 variables:
## $ S1: int 110 120 120 130 100 120 135 100 140 130 ...
## $ S2: int 100 120 130 130 100 130 140 100 140 130 ...

sbp

##      S1  S2
## 1  110 100
## 2  120 120
## 3  120 130
## 4  130 130
## 5  100 100
## 6  120 130
## 7  135 140
## 8  100 100
```

```
## 9 140 140
## 10 130 130
## 11 130 130
```

Explore the data. Obtain the basic descriptive statistics.

Mean and SD,

```
mean(sbp$S1); sd(sbp$S1)
```

```
## [1] 121.3636
```

```
## [1] 13.43334
```

```
mean(sbp$S2); sd(sbp$S2)
```

```
## [1] 122.7273
```

```
## [1] 15.5505
```

```
mean(sbp$S2 - sbp$S1); sd(sbp$S2 - sbp$S1)
```

```
## [1] 1.363636
```

```
## [1] 5.518564
```

and the number of subjects,

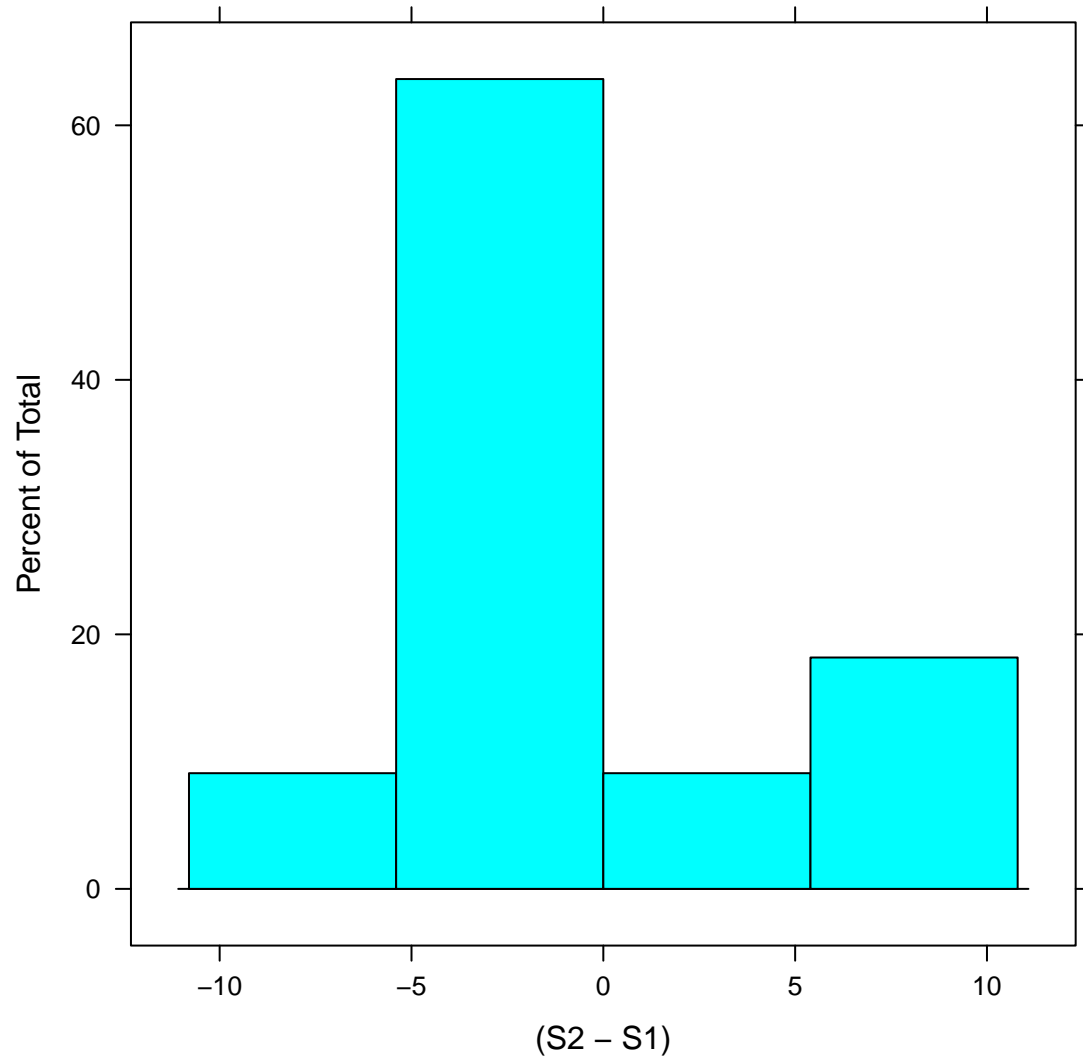
```
lengths(sbp)
```

```
## S1 S2
```

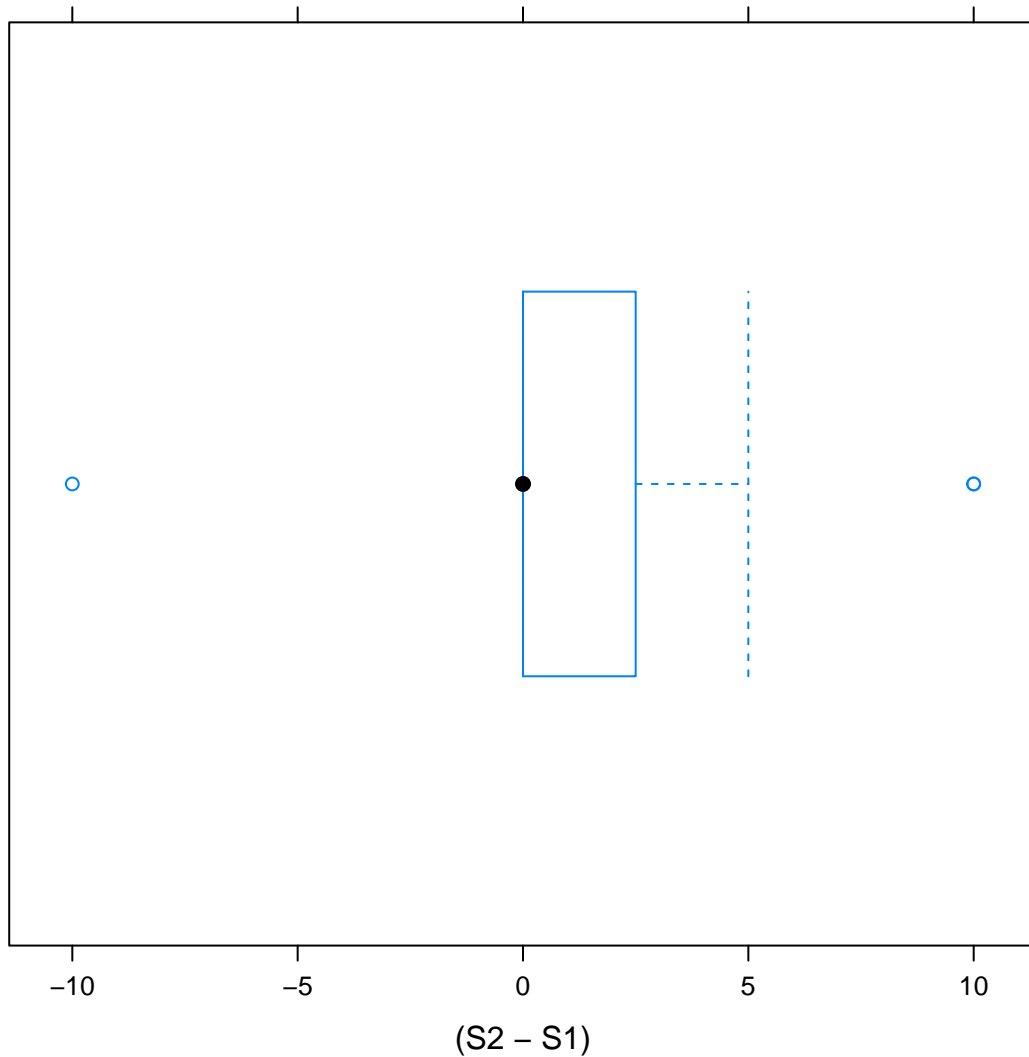
```
## 11 11
```

2. Check the **normality assumption** of the differences ($S2 - S1$),

```
histogram(~ (S2 - S1), data = sbp) # not perfectly normal
```



```
bwplot(~ (S2 - S1), data = sbp)
```

3. Perform paired t -test,

```
t.test(sbp$S1, sbp$S2, paired = TRUE) # no significant difference
```

```
##
## Paired t-test
##
## data: sbp$S1 and sbp$S2
## t = -0.81954, df = 10, p-value = 0.4316
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -5.071058 2.343785
## sample estimates:
## mean of the differences
## -1.363636
```

1.2.2 Wilcoxon signed-rank test

About the test:

- Non-parametric alternative.
- The differences are not normally distributed.
- Suitable for ordinal data.
- Involves signing (+/-) and ranking the differences (hence *signed-rank* test).
- V -statistics.

Analysis:

1. Obtain descriptive statistics for non-normal data: median and IQR,

```
median(sbp$S1); IQR(sbp$S1)
```

```
## [1] 120
```

```
## [1] 15
```

```
median(sbp$S2); IQR(sbp$S2)
```

```
## [1] 130
```

```
## [1] 20
```

2. Perform Wilcoxon signed-rank test,

```
wilcox.test(sbp$S2, sbp$S1, paired = TRUE, exact = FALSE)
```

```
##
```

```
## Wilcoxon signed rank test with continuity correction
```

```
##
```

```
## data: sbp$S2 and sbp$S1
```

```
## V = 7, p-value = 0.5708
```

```
## alternative hypothesis: true location shift is not equal to 0
```

1.3 More than two independent samples

1.3.1 One-way ANOVA

About the test:

- Parametric test.
- Comparison of means for THREE/MORE groups.
- F -statistics.

Analysis:

1. Explore the data. Obtain basic descriptive statistics,

```
by(cholest$chol, cholest$categ, mean)
```

```
## cholest$categ: Grp A
```

```
## [1] 7.248
```

```
## -----
```

```
## cholest$categ: Grp B
## [1] 8.293939
## -----
## cholest$categ: Grp C
## [1] 9.25
```

```
by(cholest$chol, cholest$categ, sd)
```

```
## cholest$categ: Grp A
## [1] 0.3355592
## -----
## cholest$categ: Grp B
## [1] 0.3091717
## -----
## cholest$categ: Grp C
## [1] 0.3569047
```

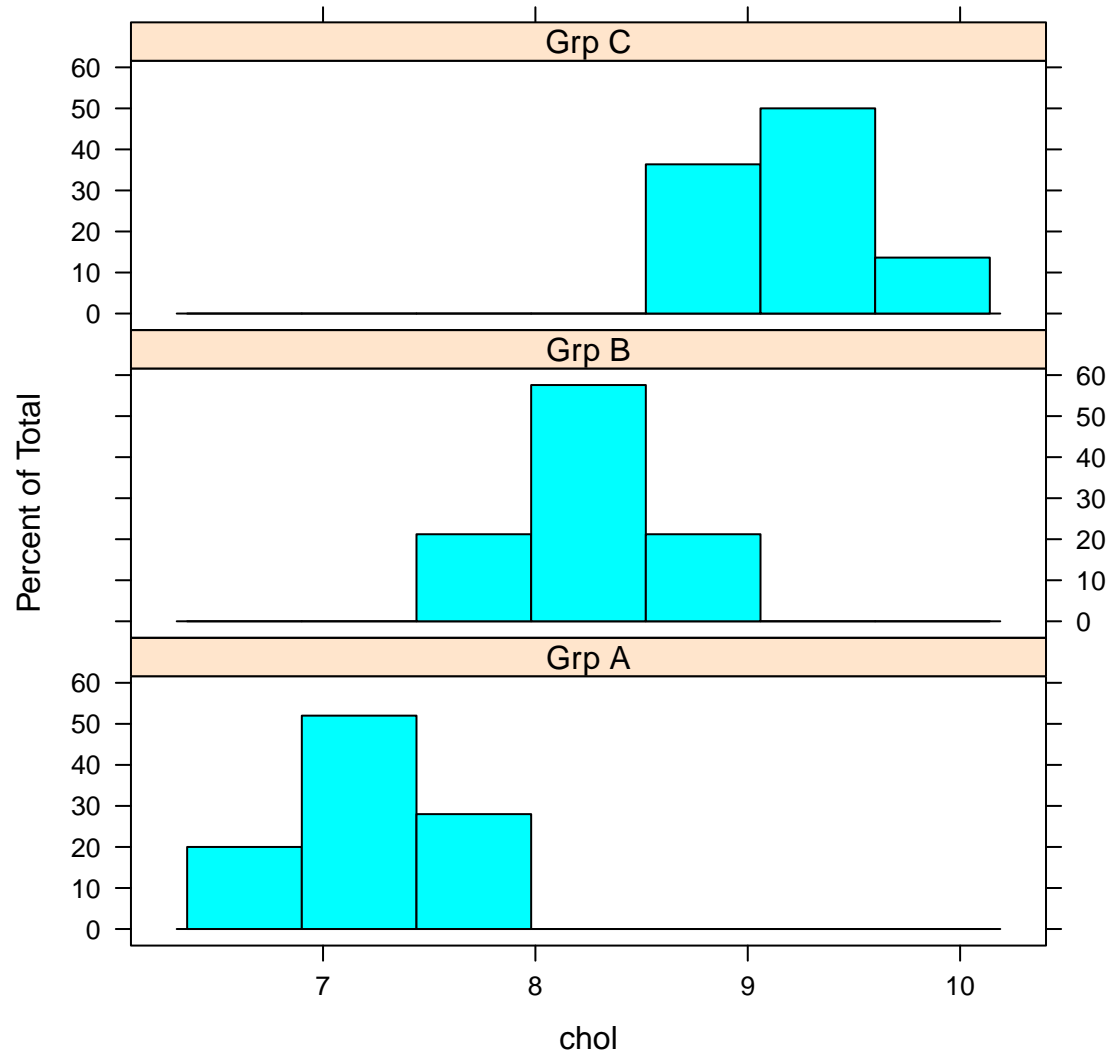
and the number of subjects per group,

```
table(cholest$categ)
```

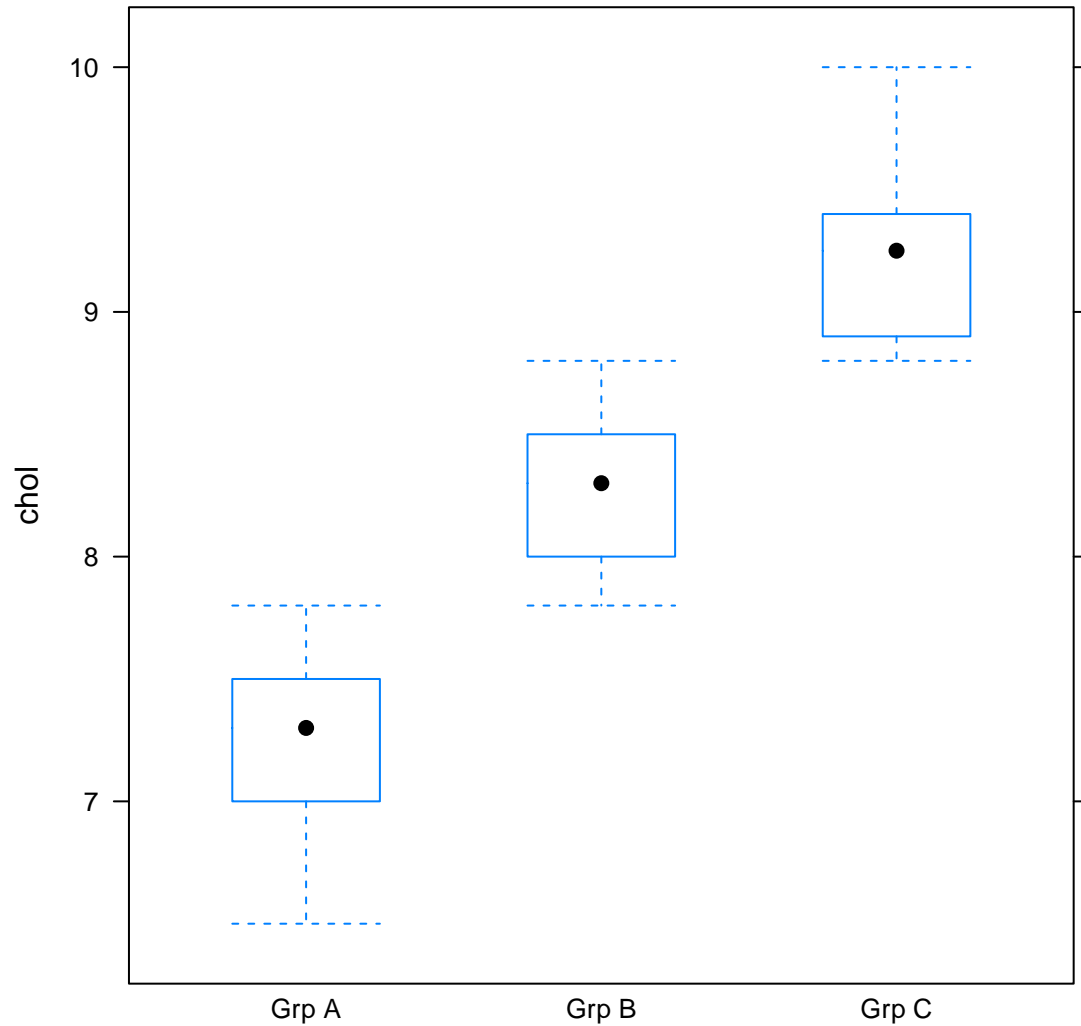
```
##
## Grp A Grp B Grp C
##    25    33    22
```

2. Check the **normality assumption** of the data per group,

```
histogram(~ chol | categ, data = cholest, layout = c(1, 3))
```



```
bwplot(chol ~ categ, data = cholest)
```



However, we will mainly rely on **residuals** for the normality assessment.

3. Check the **equality of variance assumption**,

```
bartlett.test(chol ~ categ, data = cholest)
```

```
##
## Bartlett test of homogeneity of variances
##
## data: chol by categ
## Bartlett's K-squared = 0.53515, df = 2, p-value = 0.7652
```

4. Perform one-way ANOVA test,

```
aov_chol = aov(chol ~ categ, data = cholest)
summary(aov_chol) # significant difference between three groups
```

```
##           Df Sum Sq Mean Sq F value Pr(>F)
## categ      2  47.13   23.57   215.1 <2e-16 ***
## Residuals 77   8.44    0.11
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Notice here we save the output of `aov()` into `aov_chol` first. This allows further extraction of full output from `aov_chol` ANOVA object.

Alternatively, for unequal variance, we can use Welch's version of ANOVA

```
oneway.test(chol ~ categ, data = cholest)
```

```
##
## One-way analysis of means (not assuming equal variances)
##
## data: chol and categ
## F = 194.55, num df = 2.000, denom df = 46.546, p-value < 2.2e-16
```

5. Post-hoc test, to look for significant group pairs,

```
pairwise.t.test(cholest$chol, cholest$categ, p.adj = "bonferroni")
```

```
##
## Pairwise comparisons using t tests with pooled SD
##
## data: cholest$chol and cholest$categ
##
##      Grp A  Grp B
## Grp B <2e-16 -
## Grp C <2e-16 5e-16
##
## P value adjustment method: bonferroni
# all pairs significant difference
```

Here, it works as if we do multiple independent *t*-tests. We adjust for multiple comparison by Bonferroni correction.

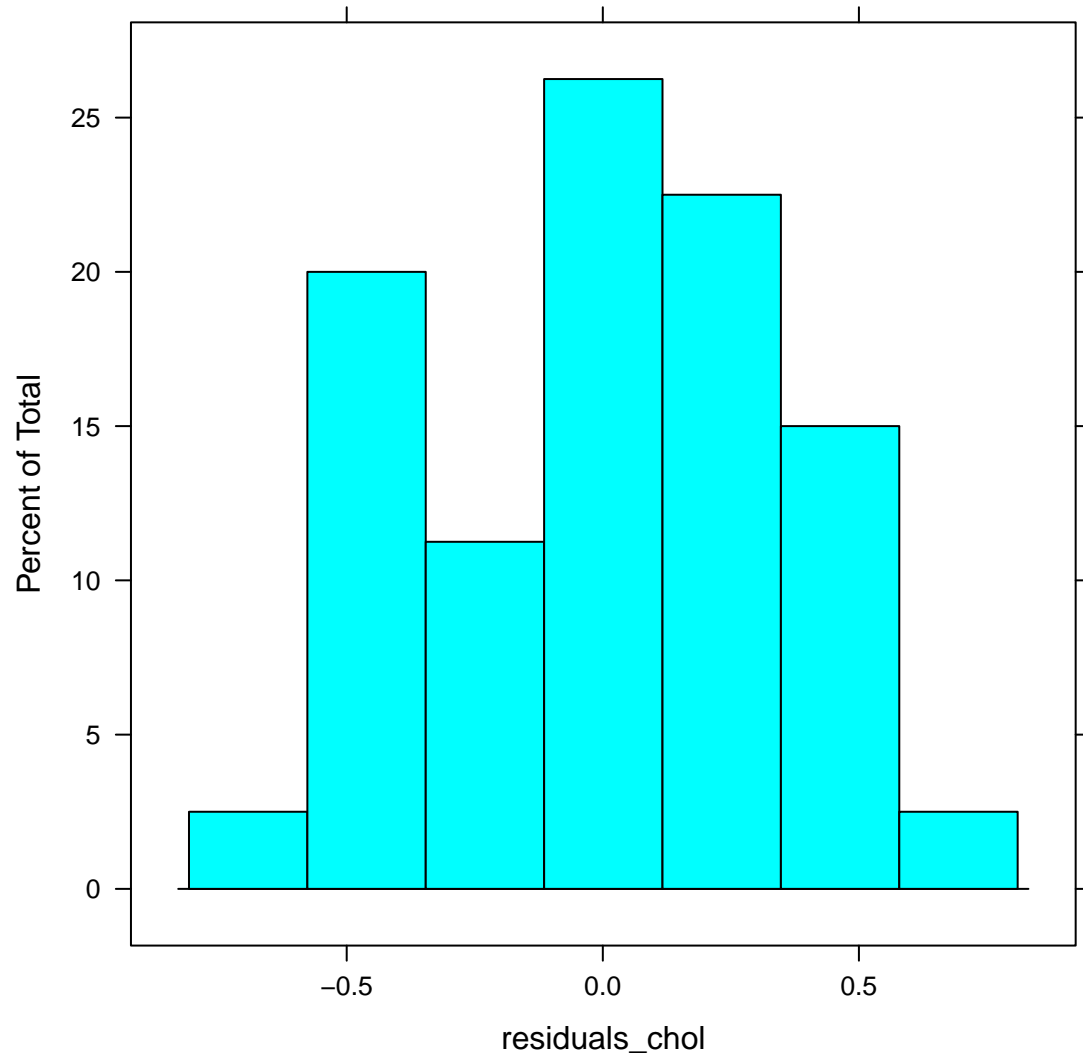
6. Check the **normality of the residuals**,

Save the residuals as `residual_chol`. We also need to use `as.numeric()` to extract proper numerical data from `aov_chol` ANOVA object, and save it again to `residuals_chol`

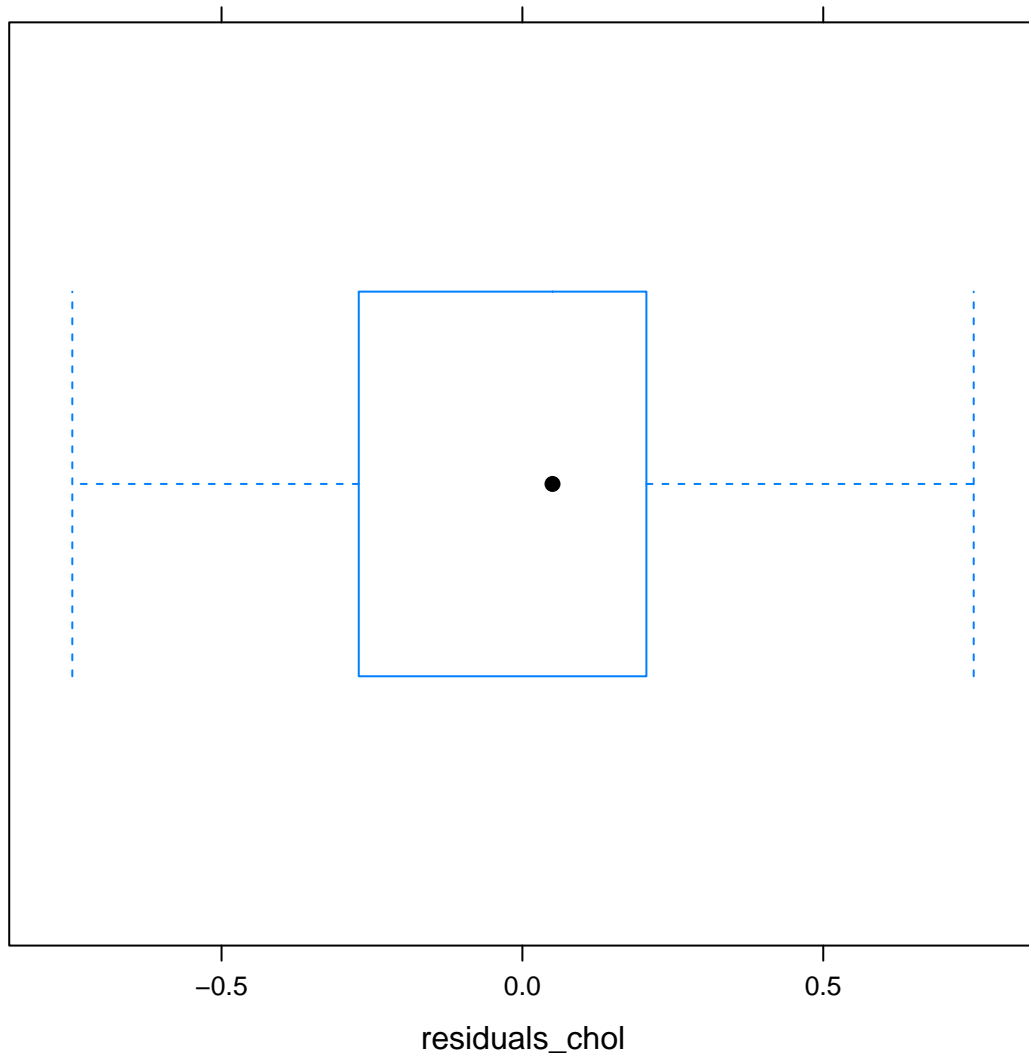
```
residuals_chol = residuals(aov_chol)
residuals_chol = as.numeric(residuals_chol)
```

Then, check the normality,

```
histogram(~ residuals_chol) # normal
```



```
bwplot(~ residuals_chol)
```



1.3.2 Kruskal-Wallis test

About the test:

- Non-parametric alternative of ANOVA.
- Involves ranking all observations (regardless of groups) and obtaining the average of ranks per group.
- H -statistics.

Analysis:

1. Obtain descriptive statistics for non-normal data, median and IQR,

```
by(cholest$chol, cholest$categ, median)
```

```
## cholest$categ: Grp A
## [1] 7.3
```



```

## -----
## cholest$categ: Grp B
## [1] 8.3
## -----
## cholest$categ: Grp C
## [1] 9.25
by(cholest$chol, cholest$categ, IQR)

## cholest$categ: Grp A
## [1] 0.5
## -----
## cholest$categ: Grp B
## [1] 0.5
## -----
## cholest$categ: Grp C
## [1] 0.475

2. Perform Kruskal-Wallis test,
kruskal.test(chol ~ categ, data = cholest)

##
## Kruskal-Wallis rank sum test
##
## data: chol by categ
## Kruskal-Wallis chi-squared = 69.188, df = 2, p-value = 9.464e-16

3. Post-hoc test, to look for significant group pairs,
pairwise.wilcox.test(cholest$chol, cholest$categ, p.adj = "bonferroni")

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): cannot compute exact p-value
## with ties

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): cannot compute exact p-value
## with ties

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): cannot compute exact p-value
## with ties

##
## Pairwise comparisons using Wilcoxon rank sum test
##
## data: cholest$chol and cholest$categ
##
##      Grp A    Grp B
## Grp B 3.3e-10 -
## Grp C 1.4e-08 1.5e-09
##
## P value adjustment method: bonferroni

```

Here, it works as if we do multiple Mann-Whitney U tests (remember the test is also known as Wilcoxon rank-sum test). We adjust for multiple comparison by Bonferroni correction.

2 Comparison of Categorical Data

2.1 Two independent samples

2.1.1 Chi-squared test for association

About the test:

- Non-parametric test.
- Association between TWO categorical variables.
- Cross-tabulation between the variables, usually 2 x 2, but can be any levels.
- The association between the variables are made by comparing the **observed** cell counts with the **expected** cell counts if the variables are not associated to each other.
- Requirement – < 25% **expected** cell counts < 5.
- χ^2 statistics.

Analysis:

1. The data.

	cancer	
smoking	lung cancer	no lung cancer
smoking	20	12
no smoking	55	113

Now, load lung.csv,

```
lung = read.csv("lung.csv")
str(lung)
```

```
## 'data.frame': 200 obs. of 2 variables:
## $ Smoking: Factor w/ 2 levels "no smoking","smoking": 2 2 2 2 2 2 2 2 2 2 ...
## $ Cancer : Factor w/ 2 levels "cancer","no cancer": 1 1 1 1 1 1 1 1 1 1 ...
```

```
head(lung)
```

```
## Smoking Cancer
## 1 smoking cancer
## 2 smoking cancer
## 3 smoking cancer
## 4 smoking cancer
## 5 smoking cancer
## 6 smoking cancer
```

Now, we create cross-tabulation of the categorical variables,

```
tab_lung = table(Smoking = lung$Smoking, Cancer = lung$Cancer)
str(tab_lung)
```

```
## 'table' int [1:2, 1:2] 55 20 113 12
## - attr(*, "dimnames")=List of 2
## ..$ Smoking: chr [1:2] "no smoking" "smoking"
## ..$ Cancer : chr [1:2] "cancer" "no cancer"
```

and view the table,

```
tab_lung
```

```
##           Cancer
## Smoking      cancer no cancer
## no smoking    55      113
## smoking       20       12
```

```
addmargins(tab_lung)
```

```
##           Cancer
## Smoking      cancer no cancer Sum
## no smoking    55      113 168
## smoking       20       12  32
## Sum           75      125 200
```

2. Perform chi-squared test for association. Two ways to do,

by using the table,

```
chisq.test(tab_lung)
```

```
##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab_lung
## X-squared = 8.9286, df = 1, p-value = 0.002807
```

or by using the variables directly,

```
chisq.test(lung$Smoking, lung$Cancer)
```

```
##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data:  lung$Smoking and lung$Cancer
## X-squared = 8.9286, df = 1, p-value = 0.002807
```

But remember, for chi-squared test, you must review the table to get an idea about the association.

3. Check assumption – $< 25\%$ **expected** cell counts < 5 .

The expected cell counts,

```
chisq.test(tab_lung)$expected
```

```
##           Cancer
## Smoking      cancer no cancer
## no smoking    63      105
## smoking       12       20
```

No count < 5 , thus we can rely on chi-squared test.

2.1.2 Fisher's exact test

About the test:

- Alternative of chi-squared test.
- Usually small cell counts, i.e. chi-squared test requirement is not fulfilled.
- Gives exact P -value, no statistical distribution involved.

Analysis:

1. Perform Fisher's exact test,

```
fisher.test(tab_lung)
```

```
##
## Fisher's Exact Test for Count Data
##
## data:  tab_lung
## p-value = 0.002414
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.1215695 0.6836086
## sample estimates:
## odds ratio
##  0.2940024
```

2.2 Two dependent samples

2.2.1 McNemar's test

About the test:

- Non-parametric test.
- Association between TWO repeated categorical outcomes.
- Cross-tabulation is limited to 2 x 2 only.
- The concern is whether the subjects still have the same outcomes (concordant) or different outcomes (discordant) upon repetition (pre-post).
- The association is determined by looking at the discordant cells.
- χ^2 statistics.

Analysis:

1. The data.

	second	
first	approve	disapprove
approve	794	150
disapprove	86	570

*Data from Agresti (2003), Table 10.1 Rating of Performance of Prime Minister

Now, we are going to enter the data in form of counts directly. This is done as follows,

```
tab_pm = read.table(header = FALSE, text = "
794 150
86 570
")
tab_pm
```

```
##      V1  V2
## 1 794 150
## 2  86 570
```

```
str(tab_pm)
```

```
## 'data.frame':    2 obs. of  2 variables:
## $ V1: int  794 86
## $ V2: int  150 570
```

which is a data frame.

To properly format the data into a table, do as follows in two steps,

```
tab_pm = as.matrix(tab_pm) # first convert to a matrix
tab_pm = as.table(tab_pm)  # then convert to a table
str(tab_pm)
```

```
## 'table' int [1:2, 1:2] 794 86 150 570
## - attr(*, "dimnames")=List of 2
## ..$ : chr [1:2] "A" "B"
## ..$ : chr [1:2] "V1" "V2"
```

Now it is a proper table from `str()`.

The table needs proper headers. Now we give them proper names,

```
dimnames(tab_pm) = list(first = c("approve", "disapprove"), second = c("approve", "disapprove"))
str(tab_pm)
```

```
## 'table' int [1:2, 1:2] 794 86 150 570
## - attr(*, "dimnames")=List of 2
## ..$ first : chr [1:2] "approve" "disapprove"
## ..$ second: chr [1:2] "approve" "disapprove"
```

Now we view the table,

```
tab_pm
```

```
##           second
## first      approve disapprove
## approve      794      150
## disapprove   86      570
```

```
addmargins(tab_pm)
```

```
##           second
## first      approve disapprove Sum
## approve      794      150  944
## disapprove   86      570  656
## Sum          880      720 1600
```

2. Perform McNemar's test,

```
mcnemar.test(tab_pm)
```

```
##
## McNemar's Chi-squared test with continuity correction
##
## data:  tab_pm
## McNemar's chi-squared = 16.818, df = 1, p-value = 4.115e-05
```

3 Correlations

3.1 Pearson's correlation

About the analysis:

- Determine the strength of relationship between TWO numerical variables.
- Normally distributed data.
- Correlation coefficient, r .
- Range: negative correlation -1 to +1 positive correlation.
- Interpretation (modified from Cohen (1992)):

Correlation	r
None or very small	< 0.1
Small	0.1 - 0.3
Medium	0.3 - 0.5
Large	> 0.5

Analysis:

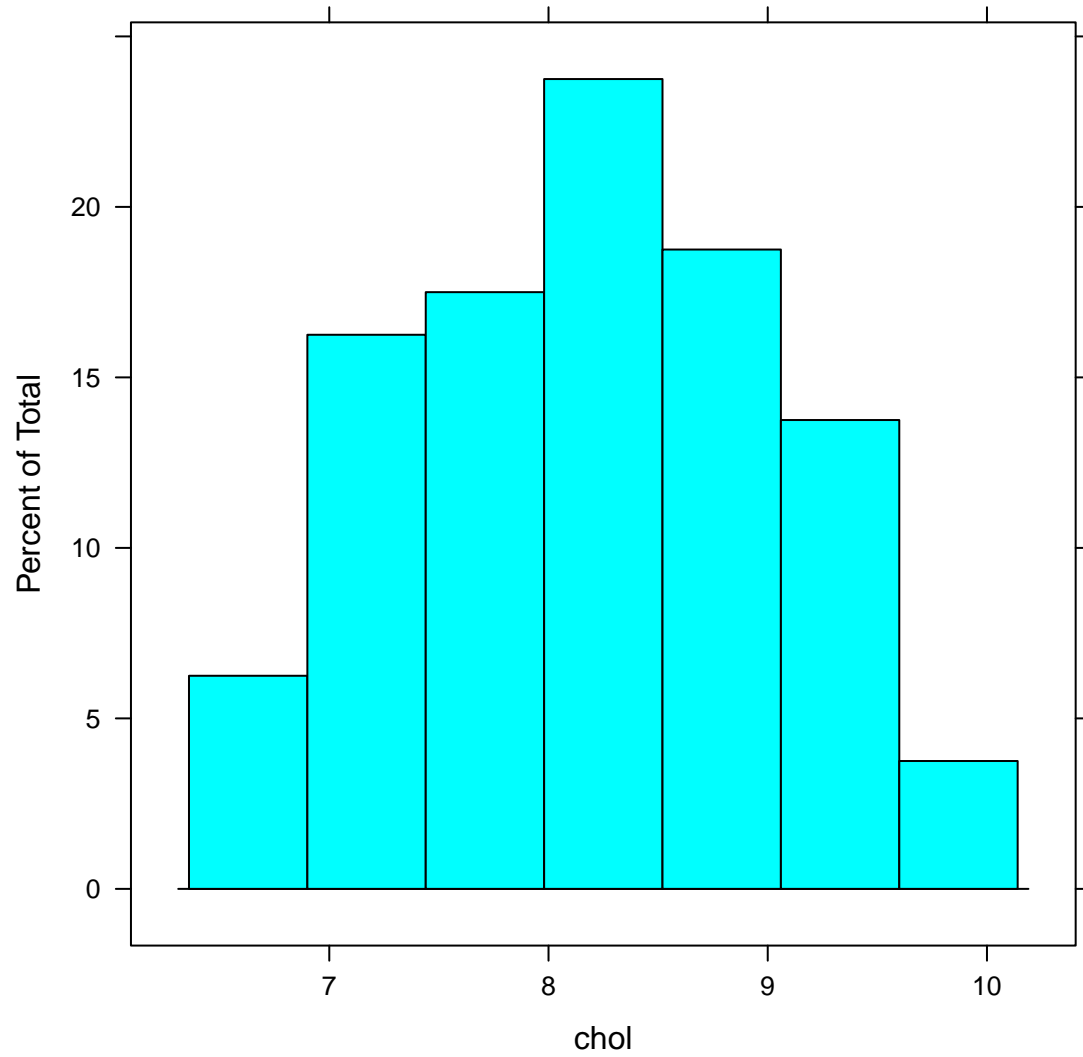
1. Load the data, `cholest.sav`. Explore the data,

```
library(foreign)
cholest = read.spss("cholest.sav", to.data.frame = T)
str(cholest)
```

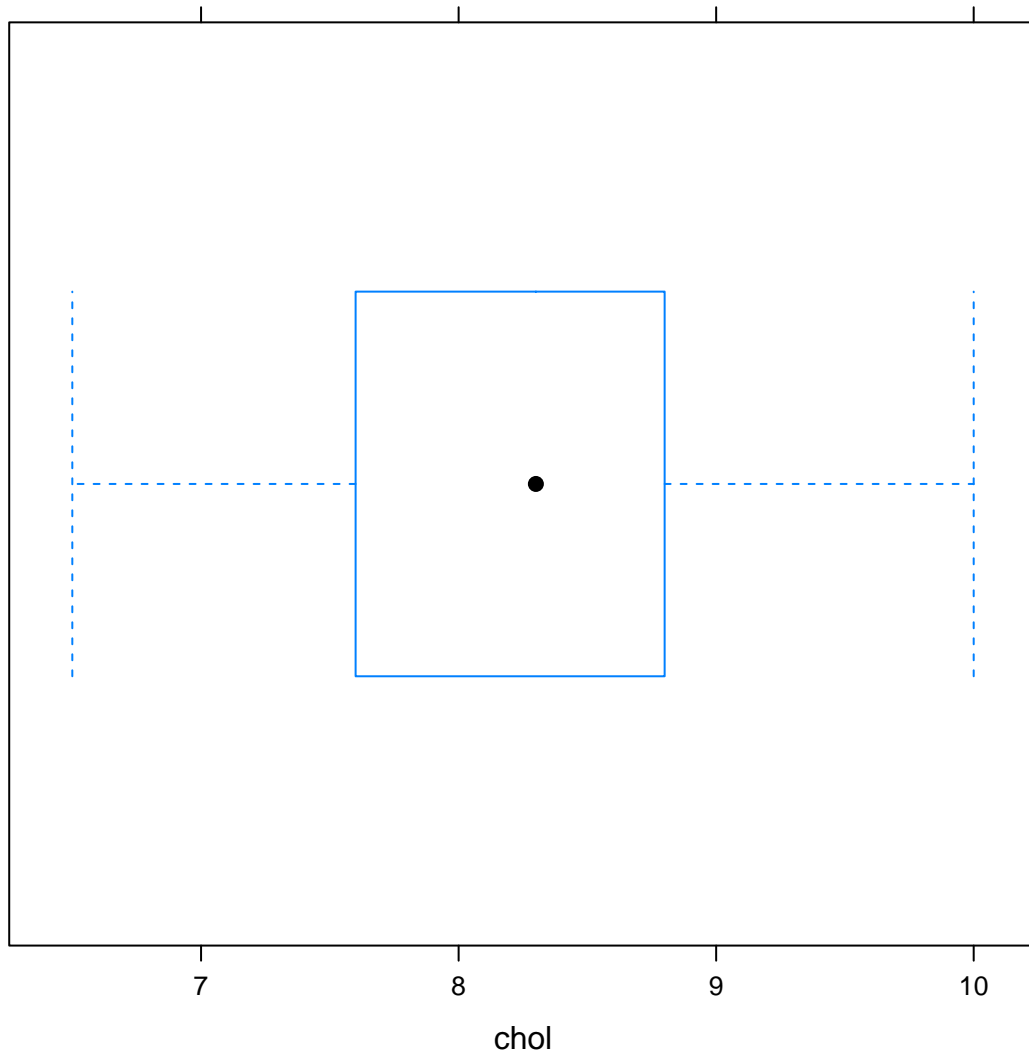
```
## 'data.frame':    80 obs. of  5 variables:
## $ chol      : num  6.5 6.6 6.8 6.8 6.9 7 7 7.2 7.2 7.2 ...
## $ age       : num  38 35 39 36 31 38 33 36 40 34 ...
## $ exercise: num  6 5 6 5 4 4 5 5 4 6 ...
## $ sex       : Factor w/ 2 levels "female","male": 2 2 2 2 2 2 2 2 2 2 ...
## $ categ     : Factor w/ 3 levels "Grp A","Grp B",...: 1 1 1 1 1 1 1 1 1 1 ...
## - attr(*, "variable.labels")= Named chr  "cholesterol in mmol/L" "age in year" "duration of exercise" ...
## ..- attr(*, "names")= chr  "chol" "age" "exercise" "sex" ...
## - attr(*, "codepage")= int 65001
```

and check the normality,

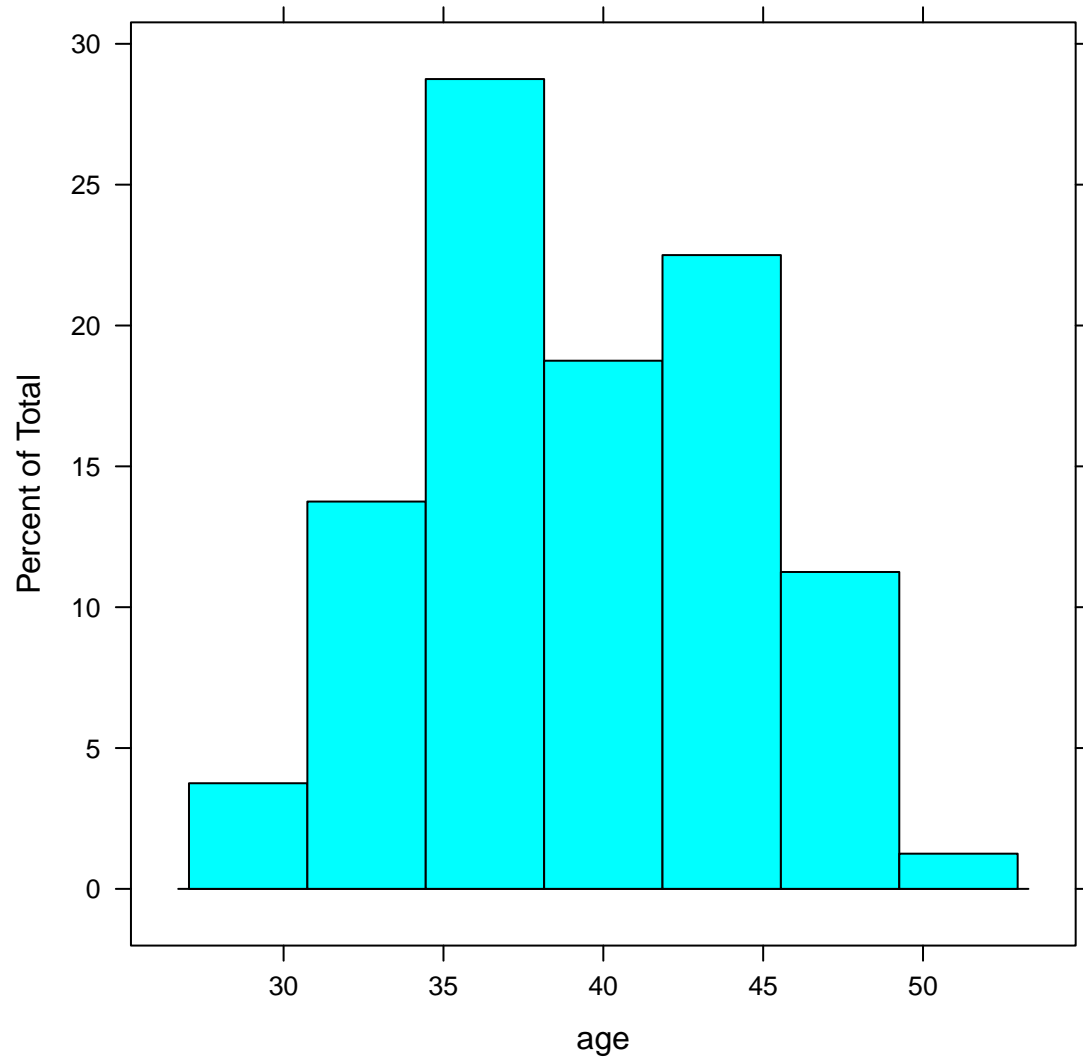
```
library(lattice)
histogram(~ chol, data = cholest)
```



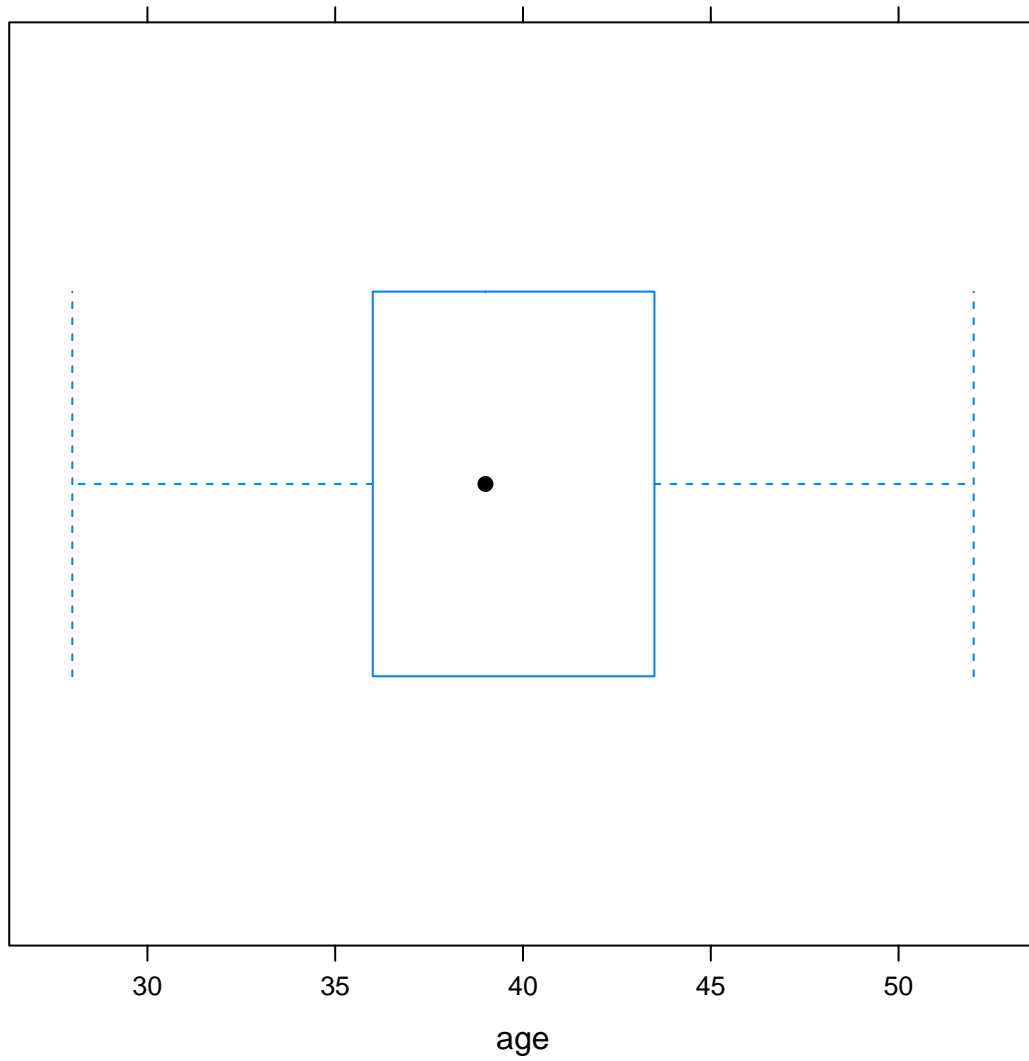
```
bwplot(~ chol, data = cholest)
```



```
histogram(~ age, data = cholest)
```

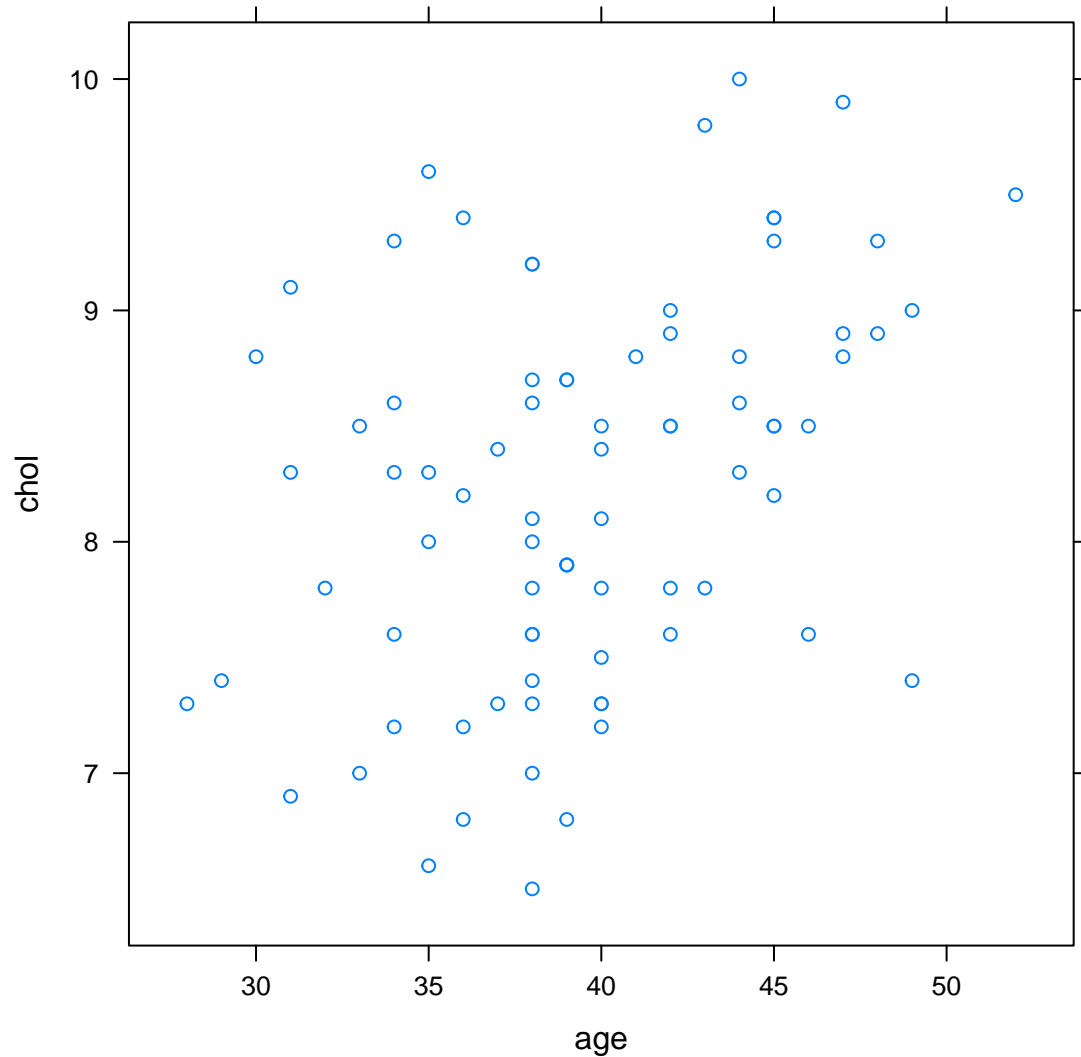



```
bwplot(~ age, data = cholest)
```



2. Plot the scatter plot,

```
xyplot(chol ~ age, data = cholest)
```



3. Perform Pearson's correlation,

```
cor.test(~ chol + age, data = cholest) # r = medium
```

```
##
## Pearson's product-moment correlation
##
## data: chol and age
## t = 3.7119, df = 78, p-value = 0.0003841
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## 0.1833492 0.5595401
## sample estimates:
## cor
## 0.3874574
```

3.2 Spearman's correlation

About the analysis:

- Non-parametric test.
- An alternative to Pearson's correlation.
- Non-normally distributed numerical data and ranked data.
- Spearman's rho correlation coefficient, ρ .

Analysis:

1. Perform Spearman's correlation,

```
cor.test(~ chol + age, data = cholest, method = "spearman")

## Warning in cor.test.default(x = c(6.5, 6.6, 6.8, 6.8, 6.9, 7, 7, 7.2, 7.2, : Cannot
## compute exact p-value with ties

##
## Spearman's rank correlation rho
##
## data: chol and age
## S = 53143, p-value = 0.0005641
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
## rho
## 0.3771289
```

References

- Agresti, A. (2003). *Categorical data analysis*. Wiley. Retrieved from <https://books.google.com.my/books?id=hpEzw4T0sPUC>
- Cohen, J. (1992). A power primer. *Psychological Bulletin*, 112(1), 155.
- R Core Team. (2018). *Foreign: Read data stored by 'minitab', 's', 'sas', 'spss', 'stata', 'systat', 'weka', 'dBase', ...* Retrieved from <https://CRAN.R-project.org/package=foreign>
- Sarkar, D. (2018). *Lattice: Trellis graphics for r*. Retrieved from <https://CRAN.R-project.org/package=lattice>