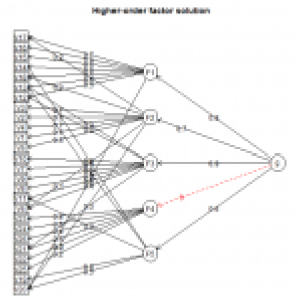
Exploratory Factor Analysis – Exercises



This set of exercises is about exploratory factor analysis. We shall use some basic features of psych package. For quick introduction to exploratory factor analysis and psych package, we recommend this short “how to” guide.

You can download the dataset here. The data is fictitious.

Answers to the exercises are available here.

If you have different solution, feel free to post it.

Exercise 1

Load the data, install the packages psych and GPArotation which we will use in the following exercises, and load it. Describe the data.

Exercise 2

Using the parallel analysis, determine the number of factors.

Exercise 3

Determine the number of factors using Very Simple Structure method.

Exercise 4

Based on normality test, is the Maximum Likelihood factoring method proper, or is OLS/Minres better? *(Tip: Maximum Likelihood method requires normal distribution.)*

Exercise 5

Using oblimin rotation, 5 factors and factoring method from the previous exercise, find the factor solution. Print loadings table with cut off at 0.3.

Exercise 6

Plot factors loadings.

Exercise 7

Plot structure diagram.

Exercise 8

Find the higher-order factor model with five factors plus general factor.

Exercise 9

Find the bifactor solution.

Exercise 10

Reduce the number of dimensions using hierarchical clustering analysis.

Exploratory Factor Analysis –

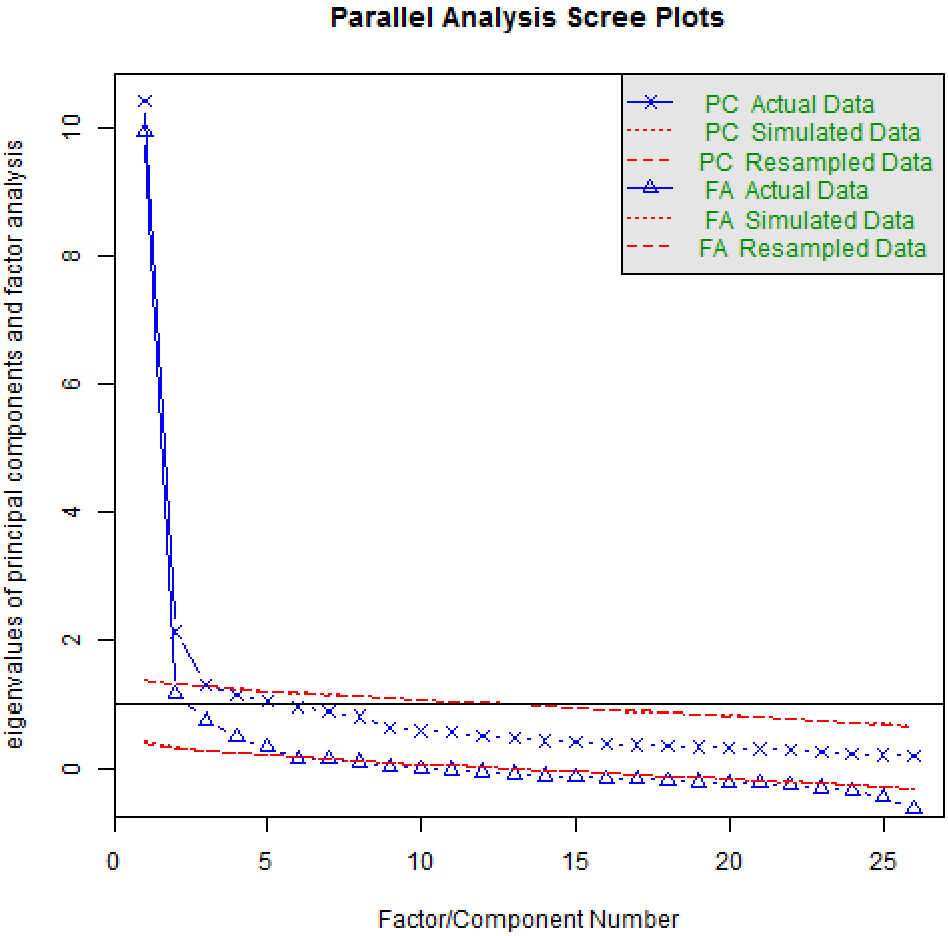
Solutions

Below are the solutions to these exercises on exploratory factor analysis.

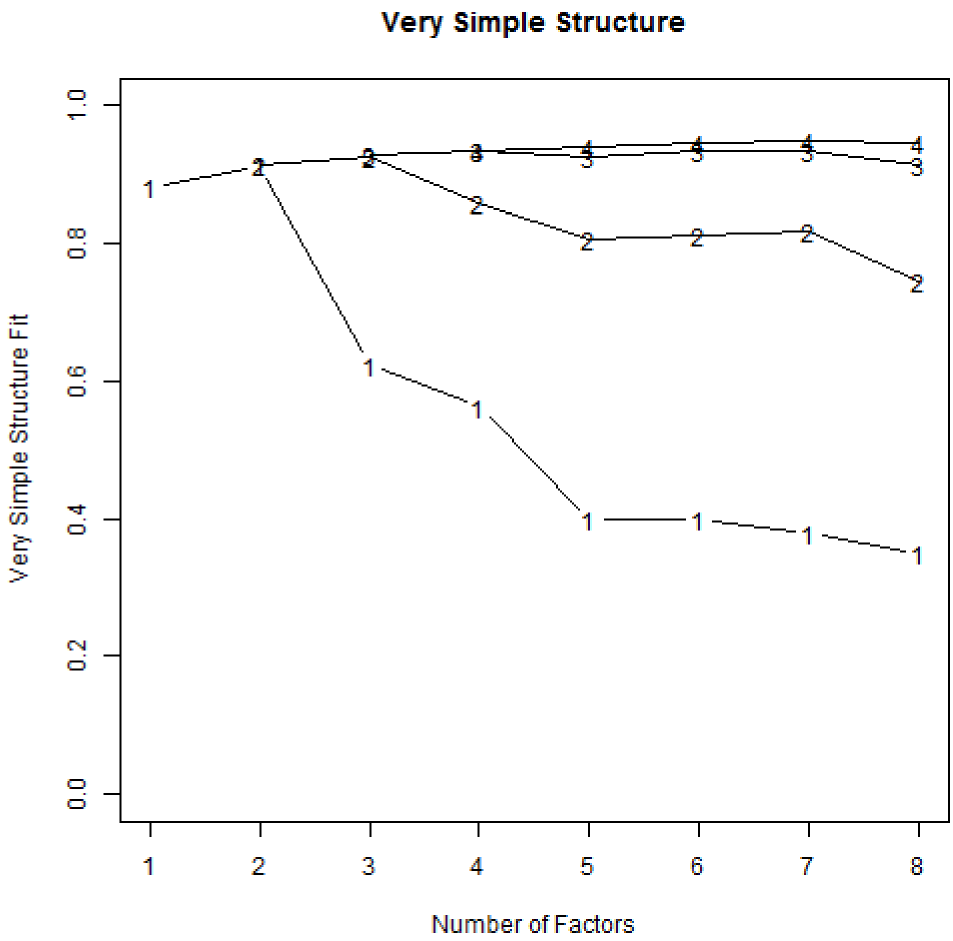
```
####################
#                  #
#    Exercise 1    #
#                  #
####################

install.packages(c("psych", "GPArotation"))
library(psych)
data <- read.file("efa.csv")
describe(data)

##      vars   n mean   sd median trimmed  mad min max range
skew kurtosis
## V1      1 649 4.79 0.47      5    4.89 0.00   3   5     2
-2.16     3.96
## V2      2 649 4.77 0.53      5    4.89 0.00   2   5     3
-2.63     7.59
## V3      3 649 4.62 0.72      5    4.79 0.00   1   5     4
-2.13     4.73
## V4      4 649 4.84 0.45      5    4.96 0.00   2   5     3
-3.13    10.87
## V5      5 649 4.85 0.44      5    4.97 0.00   2   5     3
-3.31    12.40
## V6      6 649 4.83 0.48      5    4.95 0.00   2   5     3
-3.31    12.40
## V7      7 649 4.71 0.61      5    4.85 0.00   2   5     3
-2.20     4.59
## V8      8 649 4.70 0.62      5    4.85 0.00   2   5     3
-2.22     4.70
## V9      9 649 4.50 0.85      5    4.69 0.00   1   5     4
-1.97     3.86
## V10    10 649 4.69 0.72      5    4.86 0.00   1   5     4
-2.81     8.78
## V11    11 649 4.61 0.89      5    4.86 0.00   1   5     4
-2.65     6.63
```

```
## V12   12 649 4.39 1.12      5    4.67 0.00   1   5      4
-1.86     2.39
## V13   13 649 4.68 0.62      5    4.80 0.00   1   5      4
-2.27     5.96
## V14   14 649 4.37 0.80      5    4.50 0.00   1   5      4
-1.17     0.93
## V15   15 649 4.61 0.62      5    4.71 0.00   2   5      3
-1.60     2.54
## V16   16 649 4.71 0.60      5    4.85 0.00   1   5      4
-2.23     5.09
## V17   17 649 4.71 0.62      5    4.85 0.00   1   5      4
-2.42     6.32
## V18   18 649 4.72 0.60      5    4.85 0.00   1   5      4
-2.50     7.29
## V19   19 649 4.56 0.73      5    4.72 0.00   1   5      4
-1.70     2.68
## V20   20 649 4.42 0.90      5    4.60 0.00   1   5      4
-1.73     2.80
## V21   21 649 3.30 0.96      3    3.29 1.48   1   5      4
-0.16    -1.03
## V22   22 649 2.98 1.10      3    2.89 1.48   1   5      4
0.41    -0.98
## V23   23 649 3.27 1.23      3    3.29 1.48   1   5      4
-0.10    -1.11
## V24   24 649 2.69 1.01      3    2.68 1.48   1   5      4
0.33    -0.64
## V25   25 649 2.85 1.07      3    2.85 1.48   1   5      4
0.14    -0.92
## V26   26 649 3.69 0.83      4    3.75 0.00   1   5      4
-0.74     0.53
##       se
## V1  0.02
## V2  0.02
## V3  0.03
## V4  0.02
## V5  0.02
## V6  0.02
## V7  0.02
## V8  0.02
## V9  0.03
## V10 0.03
```

```
## V11 0.03
## V12 0.04
## V13 0.02
## V14 0.03
## V15 0.02
## V16 0.02
## V17 0.02
## V18 0.02
## V19 0.03
## V20 0.04
## V21 0.04
## V22 0.04
## V23 0.05
## V24 0.04
## V25 0.04
## V26 0.03

####################
#                  #
#    Exercise 2    #
#                  #
####################

fa.parallel(data)
```




```
## Parallel analysis suggests that the number of factors =  5
and the number of components =  3

####################
#                  #
#    Exercise 3    #
#                  #
####################

vss(data)
```



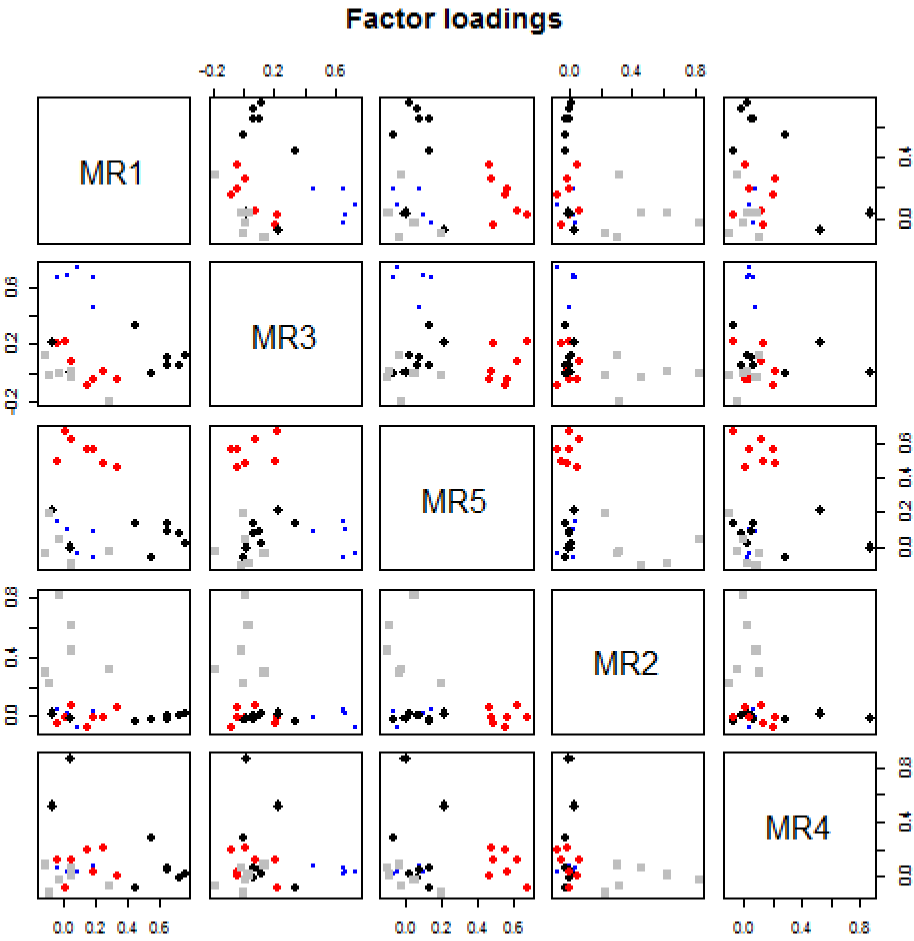

```
##
## Very Simple Structure
## Call: vss(x = data)
## VSS complexity 1 achieves a maximimum of 0.91  with  2
factors
## VSS complexity 2 achieves a maximimum of 0.93  with  3
factors
##
## The Velicer MAP achieves a minimum of 0.01  with  3
factors
## BIC achieves a minimum of  -742.22  with  5  factors
## Sample Size adjusted BIC achieves a minimum of  -160.17
with  8  factors
##
## Statistics by number of factors
##   vss1 vss2   map dof chisq     prob sqresid  fit RMSEA
BIC SABIC complex
## 1 0.88 0.00 0.014 299  2003 1.9e-249    14.6 0.88 0.095
67  1016     1.0
```

```
## 2 0.91 0.91 0.014 274  1546 3.3e-176    10.5 0.91 0.086
-228   642     1.0
## 3 0.62 0.93 0.013 250  1095 4.9e-106     9.0 0.93 0.073
-524   270     1.4
## 4 0.56 0.86 0.015 227   783  2.9e-62     8.1 0.93 0.062
-687    34     1.7
## 5 0.40 0.81 0.019 205   585  2.8e-38     7.3 0.94 0.054
-742   -91     1.9
## 6 0.40 0.81 0.022 184   502  2.4e-31     6.4 0.95 0.053
-689  -105     2.0
## 7 0.38 0.82 0.024 164   425  4.8e-25     5.7 0.95 0.050
-637  -116     2.1
## 8 0.35 0.75 0.027 145   318  5.0e-15     5.3 0.96 0.044
-621  -160     2.3
##   eChisq  SRMR eCRMS  eBIC
## 1   2201 0.072 0.075   265
## 2   1093 0.051 0.055  -682
## 3    665 0.040 0.045  -953
## 4    461 0.033 0.040 -1009
## 5    320 0.028 0.035 -1007
## 6    237 0.024 0.031  -955
## 7    162 0.020 0.028  -900
## 8    115 0.017 0.025  -824

####################
#                  #
#    Exercise 4    #
#                  #
####################

sapply(data, shapiro.test)

##           V1                            V2
## statistic 0.4880158                     0.4818245
## p.value   1.790874e-39                  1.218162e-39
## method    "Shapiro-Wilk normality test" "Shapiro-Wilk
normality test"
## data.name "X[[i]]"                      "X[[i]]"
##           V3                            V4
## statistic 0.5837358                     0.4039763
## p.value   1.193739e-36                  1.287584e-41
```

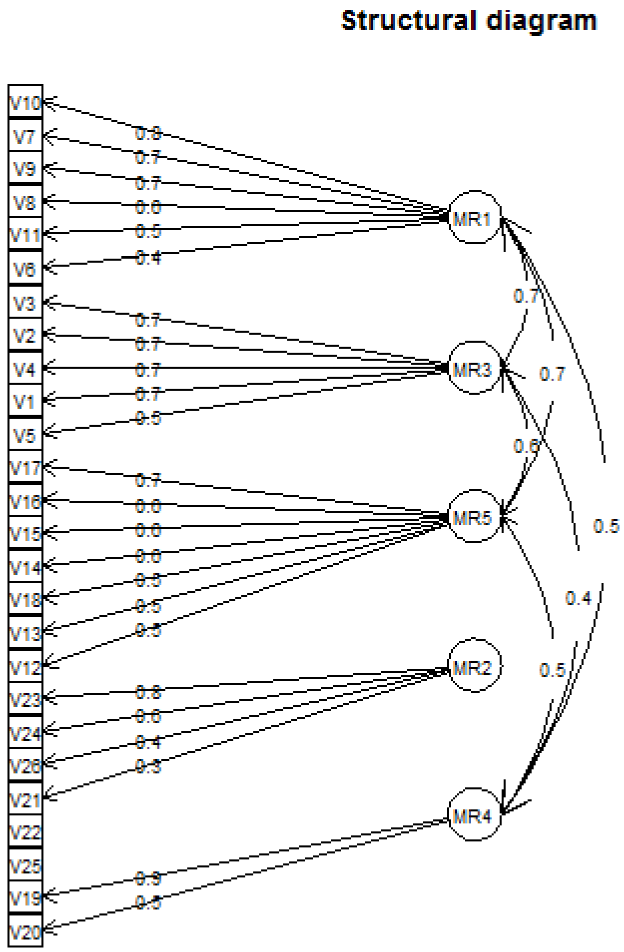
```
## method     "Shapiro-Wilk normality test" "Shapiro-Wilk
normality test"
## data.name "X[[i]]"                      "X[[i]]"
##           V5                            V6
## statistic 0.386337                      0.4010764
## p.value   4.917495e-42                  1.097388e-41
## method     "Shapiro-Wilk normality test" "Shapiro-Wilk
normality test"
## data.name "X[[i]]"                      "X[[i]]"
##           V7                            V8
## statistic 0.5352881                     0.5343438
## p.value   3.876222e-38                  3.636428e-38
## method     "Shapiro-Wilk normality test" "Shapiro-Wilk
normality test"
## data.name "X[[i]]"                      "X[[i]]"
##           V9                            V10
## statistic 0.631653                      0.4956688
## p.value   4.943564e-35                  2.898898e-39
## method     "Shapiro-Wilk normality test" "Shapiro-Wilk
normality test"
## data.name "X[[i]]"                      "X[[i]]"
##           V11                           V12
## statistic 0.4952979                     0.6048789
## p.value   2.83163e-39                   5.900295e-36
## method     "Shapiro-Wilk normality test" "Shapiro-Wilk
normality test"
## data.name "X[[i]]"                      "X[[i]]"
##           V13                           V14
## statistic 0.5630808                     0.7476871
## p.value   2.666746e-37                  2.854301e-30
## method     "Shapiro-Wilk normality test" "Shapiro-Wilk
normality test"
## data.name "X[[i]]"                      "X[[i]]"
##           V15                           V16
## statistic 0.6405669                     0.5328645
## p.value   1.031309e-34                  3.290914e-38
## method     "Shapiro-Wilk normality test" "Shapiro-Wilk
normality test"
## data.name "X[[i]]"                      "X[[i]]"
##           V17                           V18
## statistic 0.5253764                     0.5207042
```

```
## p.value   1.993172e-38                    1.462449e-38
## method     "Shapiro-Wilk normality test" "Shapiro-Wilk
normality test"
## data.name "X[[i]]"                        "X[[i]]"
##           V19                             V20
## statistic 0.6448004                       0.6737761
## p.value   1.469911e-34                    1.828638e-33
## method     "Shapiro-Wilk normality test" "Shapiro-Wilk
normality test"
## data.name "X[[i]]"                        "X[[i]]"
##           V21                             V22
## statistic 0.8599259                       0.8613114
## p.value   1.369292e-23                    1.744914e-23
## method     "Shapiro-Wilk normality test" "Shapiro-Wilk
normality test"
## data.name "X[[i]]"                        "X[[i]]"
##           V23                             V24
## statistic 0.8992078                       0.8899322
## p.value   3.080095e-20                    4.157814e-21
## method     "Shapiro-Wilk normality test" "Shapiro-Wilk
normality test"
## data.name "X[[i]]"                        "X[[i]]"
##           V25                             V26
## statistic 0.8948388                       0.8292828
## p.value   1.17986e-20                     9.748275e-26
## method     "Shapiro-Wilk normality test" "Shapiro-Wilk
normality test"
## data.name "X[[i]]"                        "X[[i]]"

####################
#                  #
#    Exercise 5    #
#                  #
####################

f.solution <- fa(data, nfactors=5, rotate="oblimin",
fm="minres")
print(f.solution$loadings, cutoff=0.3)

##
## Loadings:
```

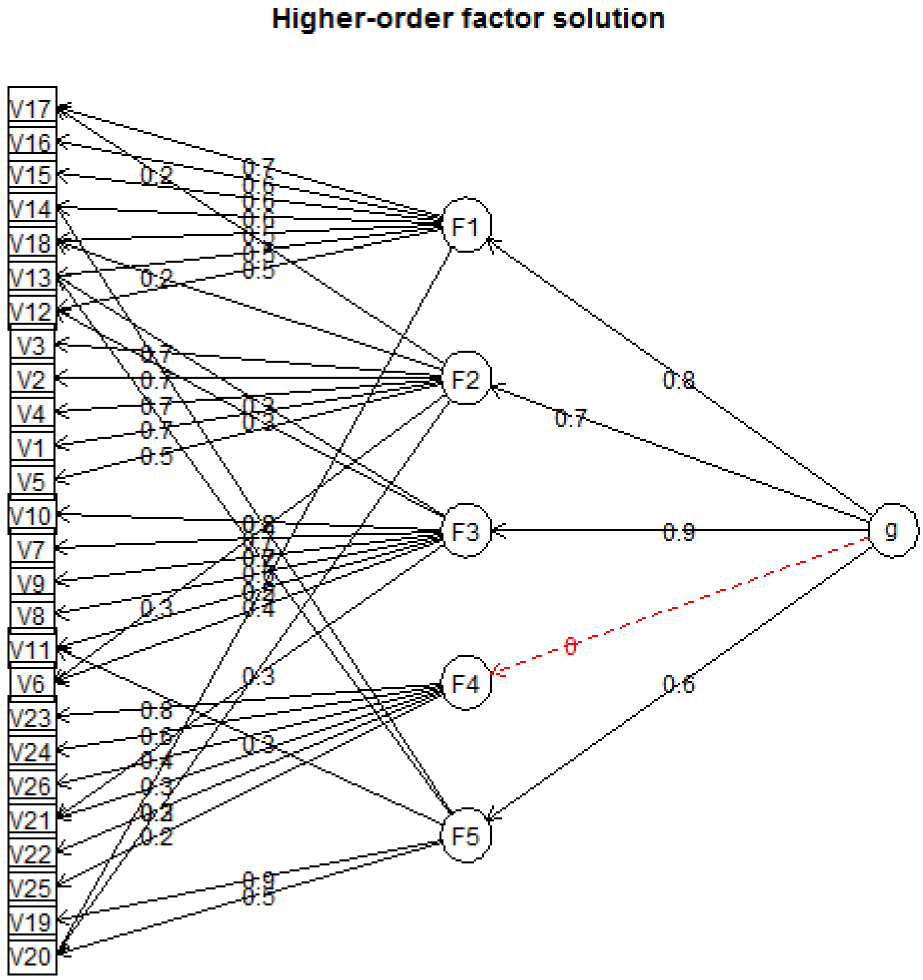
```
##      MR1    MR3    MR5    MR2    MR4
## V1          0.662
## V2          0.675
## V3          0.740
## V4          0.664
## V5          0.460
## V6   0.448  0.336
## V7   0.722
## V8   0.648
## V9   0.652
## V10  0.759
## V11  0.546
## V12  0.344         0.465
## V13                0.483
## V14                0.563
## V15                0.566
## V16                0.621
## V17                0.676
## V18                0.495
## V19                              0.866
## V20                              0.524
## V21                       0.316
## V22
## V23                       0.820
## V24                       0.616
## V25
## V26                       0.448
##
##                  MR1   MR3   MR5   MR2   MR4
## SS loadings    2.887 2.452 2.381 1.522 1.283
## Proportion Var 0.111 0.094 0.092 0.059 0.049
## Cumulative Var 0.111 0.205 0.297 0.355 0.405

####################
#                  #
#    Exercise 6    #
#                  #
####################

plot(f.solution, title="Factor loadings")
```




```
####################
#                  #
#    Exercise 7    #
#                  #
####################

fa.diagram(f.solution, main="Structural diagram")
```




```
####################
#                  #
#    Exercise 8    #
#                  #
####################

omega(data, nfactors = 5, sl=FALSE, title="Higher-order factor
solution")
```

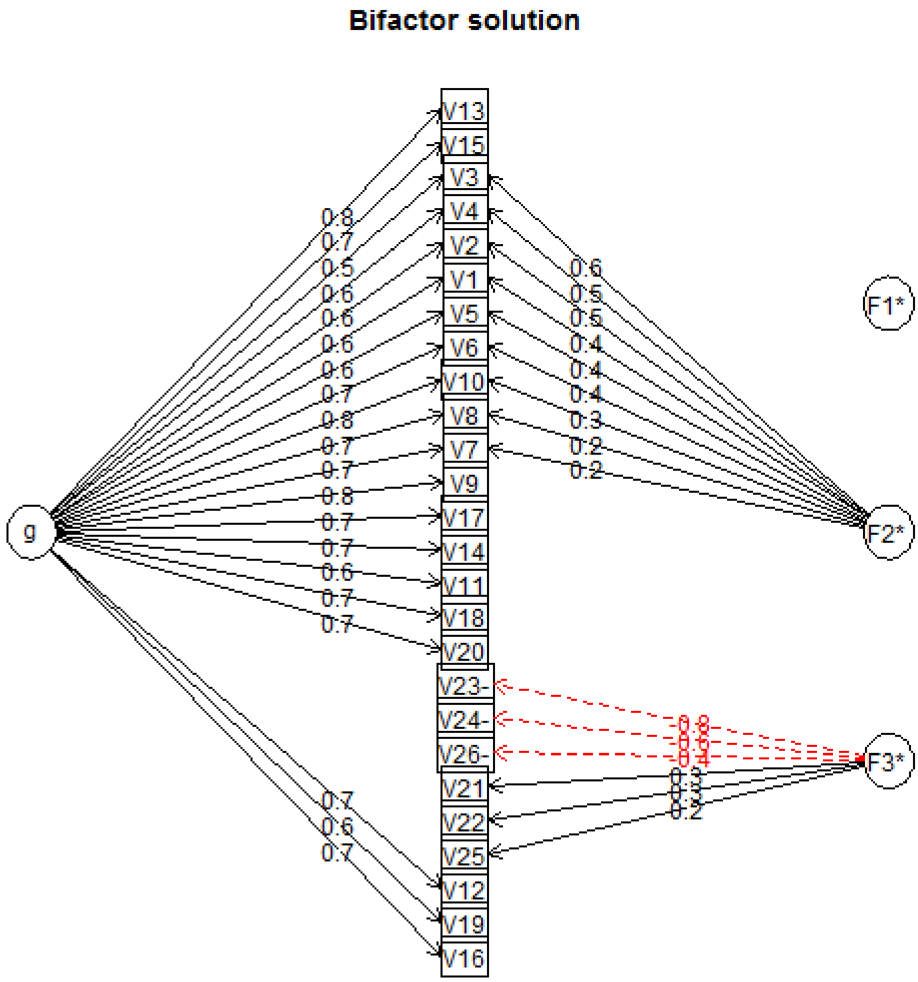



```
## Higher-order factor solution
## Call: omega(m = data, nfactors = 5, title = "Higher-order
factor solution",
##     sl = FALSE)
## Alpha:                 0.91
## G.6:                   0.94
## Omega Hierarchical:    0.81
## Omega H asymptotic:    0.86
## Omega Total            0.94
##
## Schmid Leiman Factor loadings greater than  0.2
##        g   F1*   F2*   F3*   F4*   F5*   h2   u2   p2
## V1   0.62        0.45                   0.58 0.42 0.65
## V2   0.62        0.46                   0.59 0.41 0.65
## V3   0.61        0.50                   0.63 0.37 0.58
## V4   0.62        0.45                   0.60 0.40 0.63
## V5   0.62        0.31                   0.50 0.50 0.77
## V6   0.70        0.23  0.23             0.62 0.38 0.80
## V7   0.72              0.37             0.65 0.35 0.79
```

```
## V8    0.74              0.33             0.66 0.34 0.82
## V9    0.76              0.33             0.69 0.31 0.83
## V10   0.77              0.38             0.76 0.24 0.79
## V11   0.59              0.28        0.22 0.46 0.54 0.74
## V12   0.64  0.27                         0.52 0.48 0.79
## V13   0.75  0.28                         0.69 0.31 0.82
## V14   0.65  0.33                         0.57 0.43 0.73
## V15   0.62  0.33                         0.50 0.50 0.76
## V16   0.67  0.36                         0.60 0.40 0.75
## V17   0.68  0.39                         0.63 0.37 0.72
## V18   0.60  0.29                         0.48 0.52 0.75
## V19   0.57                          0.69 0.79 0.21 0.41
## V20   0.60                          0.41 0.57 0.43 0.63
## V21                           0.32       0.15 0.85 0.01
## V22                           0.30       0.10 0.90 0.00
## V23-                         -0.82       0.68 0.32 0.00
## V24-                         -0.62       0.38 0.62 0.00
## V25                           0.23       0.08 0.92 0.00
## V26-                         -0.45       0.20 0.80 0.00
##
## With eigenvalues of:
##    g  F1*  F2*  F3*  F4*  F5*
## 8.68 0.81 1.12 0.74 1.52 0.80
##
## general/max  5.71   max/min =   2.06
## mean percent general =  0.55    with sd =  0.32 and cv of
0.58
## Explained Common Variance of the general factor =  0.64
##
## The degrees of freedom are 205  and the fit is  0.92
## The number of observations was  649  with Chi Square =
585.24  with prob <  2.8e-38
## The root mean square of the residuals is  0.03
## The df corrected root mean square of the residuals is  0.03
## RMSEA index =  0.054  and the 10 % confidence intervals are
0.048 0.059
## BIC =  -742.22
##
## Compare this with the adequacy of just a general factor and
no group factors
## The degrees of freedom for just the general factor are 299
```

```
and the fit is  3.3
## The number of observations was  649  with Chi Square =
2103.59  with prob <  3.5e-268
## The root mean square of the residuals is  0.09
## The df corrected root mean square of the residuals is  0.09
##
## RMSEA index =  0.097  and the 10 % confidence intervals are
0.093 0.1
## BIC =  167.43
##
## Measures of factor score adequacy
##                                                  g   F1*
F2*   F3*  F4*
## Correlation of scores with factors           0.92  0.68
0.77  0.64 0.88
## Multiple R square of scores with factors      0.84  0.46
0.59  0.41 0.77
## Minimum correlation of factor score estimates 0.67 -0.08
0.19 -0.19 0.55
##                                               F5*
## Correlation of scores with factors           0.82
## Multiple R square of scores with factors      0.68
## Minimum correlation of factor score estimates 0.35
##
##  Total, General and Subset omega for each subset
##                                                  g  F1*
F2*  F3*  F4*
## Omega total for total scores and subscales    0.94 0.88
0.87 0.88 0.21
## Omega general for total scores and subscales  0.81 0.71
0.62 0.72 0.00
## Omega group for total scores and subscales    0.07 0.17
0.25 0.16 0.21
##                                               F5*
## Omega total for total scores and subscales    0.78
## Omega general for total scores and subscales  0.42
## Omega group for total scores and subscales    0.37

####################
#                  #
#    Exercise 9    #
```

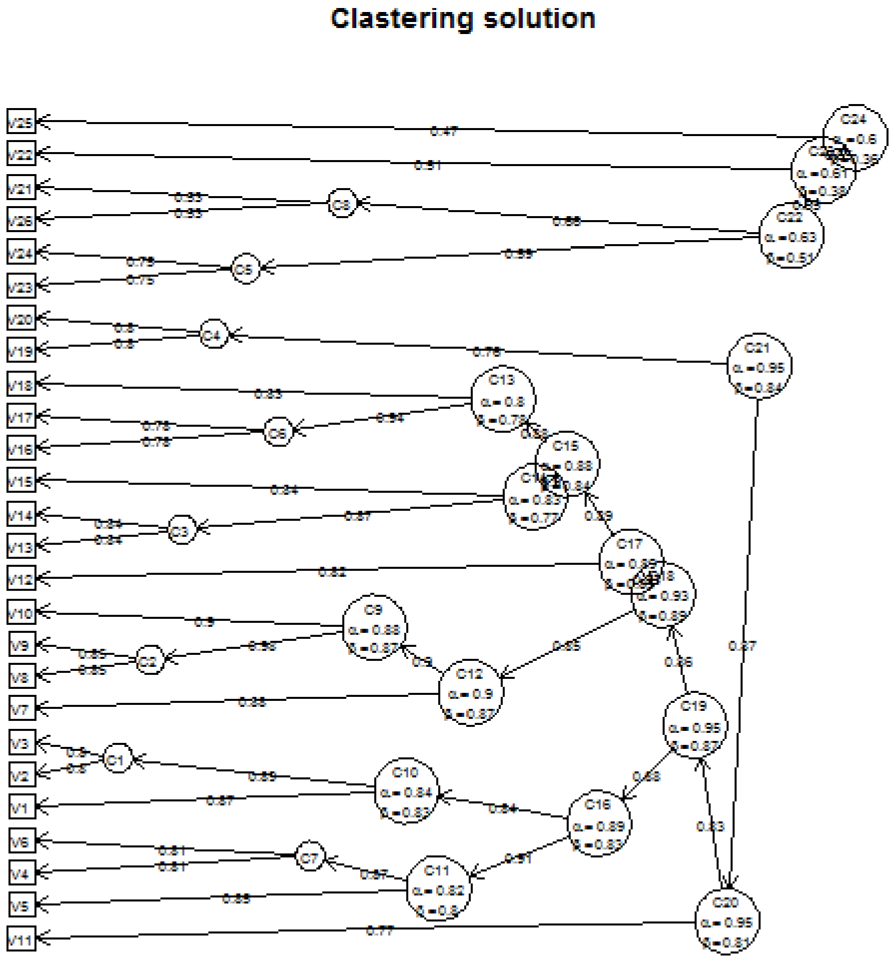
```
#                   #
####################

omega(data, title="Bifactor solution")
```




```
## Bifactor solution
## Call: omega(m = data, title = "Bifactor solution")
## Alpha:                 0.91
## G.6:                   0.94
## Omega Hierarchical:    0.86
## Omega H asymptotic:    0.92
## Omega Total            0.94
##
## Schmid Leiman Factor loadings greater than  0.2
##           g   F1*   F2*   F3*   h2   u2   p2
## V1     0.59        0.44        0.54 0.46 0.64
## V2     0.58        0.48        0.56 0.44 0.59
## V3     0.54        0.56        0.63 0.37 0.47
## V4     0.56        0.54        0.60 0.40 0.52
## V5     0.61        0.37        0.50 0.50 0.73
```

```
## V6    0.68        0.35       0.59 0.41 0.79
## V7    0.72        0.21       0.56 0.44 0.91
## V8    0.74        0.22       0.61 0.39 0.92
## V9    0.77                   0.64 0.36 0.94
## V10   0.77        0.26       0.66 0.34 0.89
## V11   0.61                   0.38 0.62 0.97
## V12   0.70                   0.50 0.50 0.98
## V13   0.83                   0.70 0.30 1.00
## V14   0.75                   0.58 0.42 0.97
## V15   0.69                   0.48 0.52 0.99
## V16   0.74                   0.56 0.44 0.99
## V17   0.71                   0.53 0.47 0.97
## V18   0.65                   0.44 0.56 0.97
## V19   0.63                   0.41 0.59 0.98
## V20   0.66                   0.44 0.56 0.98
## V21                     0.34 0.13 0.87 0.02
## V22                     0.27 0.07 0.93 0.00
## V23-                   -0.81 0.66 0.34 0.00
## V24-                   -0.61 0.37 0.63 0.00
## V25                     0.23 0.06 0.94 0.00
## V26-                   -0.44 0.19 0.81 0.00
##
## With eigenvalues of:
##    g  F1*  F2*  F3*
## 9.28 0.03 1.55 1.53
##
## general/max  5.98   max/min =   57.78
## mean percent general =  0.66    with sd =  0.4 and cv of
0.6
## Explained Common Variance of the general factor =  0.75
##
## The degrees of freedom are 250  and the fit is  1.72
## The number of observations was  649  with Chi Square =
1095.18  with prob <  4.9e-106
## The root mean square of the residuals is  0.04
## The df corrected root mean square of the residuals is  0.05
## RMSEA index =  0.073  and the 10 % confidence intervals are
0.068 0.077
## BIC =  -523.68
##
## Compare this with the adequacy of just a general factor and
```

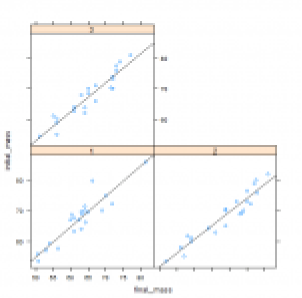
```
no group factors
## The degrees of freedom for just the general factor are 299
and the fit is  3.37
## The number of observations was  649  with Chi Square =
2150.99  with prob <  4.8e-277
## The root mean square of the residuals is  0.09
## The df corrected root mean square of the residuals is  0.09
##
## RMSEA index =  0.099  and the 10 % confidence intervals are
0.094 0.102
## BIC =  214.83
##
## Measures of factor score adequacy
##                                                  g   F1*
F2*  F3*
## Correlation of scores with factors            0.97  0.08
0.83 0.87
## Multiple R square of scores with factors      0.93  0.01
0.69 0.76
## Minimum correlation of factor score estimates 0.86 -0.99
0.38 0.53
##
##  Total, General and Subset omega for each subset
##                                                  g  F1*
F2*  F3*
## Omega total for total scores and subscales    0.94 0.74
0.94 0.51
## Omega general for total scores and subscales  0.86 0.74
0.82 0.43
## Omega group for total scores and subscales    0.07 0.00
0.13 0.08

####################
#                  #
#    Exercise 10   #
#                  #
####################

iclust(data, title="Clastering solution")
```



```
## ICLUST (Item Cluster Analysis)
## Call: iclust(r.mat = data, title = "Clastering solution")
##
## Purified Alpha:
##  C21  C24
## 0.95 0.60
##
## G6* reliability:
## C21 C24
##   1   1
##
## Original Beta:
##  C21  C24
## 0.84 0.36
##
## Cluster size:
## C21 C24
##  20   6
##
```

```
## Item by Cluster Structure matrix:
##       O   P   C21   C24
## V1  C21 C21  0.69 -0.01
## V2  C21 C21  0.69 -0.01
## V3  C21 C21  0.67 -0.13
## V4  C21 C21  0.68  0.02
## V5  C21 C21  0.68 -0.02
## V6  C21 C21  0.75 -0.01
## V7  C21 C21  0.75  0.03
## V8  C21 C21  0.78  0.03
## V9  C21 C21  0.80  0.01
## V10 C21 C21  0.81  0.05
## V11 C21 C21  0.62 -0.03
## V12 C21 C21  0.67  0.08
## V13 C21 C21  0.80  0.01
## V14 C21 C21  0.69 -0.07
## V15 C21 C21  0.65  0.04
## V16 C21 C21  0.72  0.08
## V17 C21 C21  0.72  0.01
## V18 C21 C21  0.65 -0.05
## V19 C21 C21  0.61 -0.05
## V20 C21 C21  0.66 -0.03
## V21 C24 C24  0.03  0.36
## V22 C24 C24  0.02  0.31
## V23 C24 C24 -0.02  0.72
## V24 C24 C24 -0.02  0.57
## V25 C24 C24  0.00  0.27
## V26 C24 C24 -0.03  0.51
##
## With eigenvalues of:
##  C21  C24
## 10.0  1.4
##
## Purified scale intercorrelations
##  reliabilities on diagonal
##  correlations corrected for attenuation above diagonal:
##      C21   C24
## C21 0.95 -0.01
## C24 0.00  0.60
##
## Cluster fit =  0.91   Pattern fit =  0.99  RMSR =  0.05
```

Experimental Design Exercises



In this set of exercises we shall follow the practice of conducting an experimental study. Researcher wants to see if there is any influence of working-out on body mass. Three groups of subjects with similar food and sport habits were included in the experiment. Each group was subjected to a different set of exercises. Body mass was measured before and after workout. The focus of the research is the difference in body mass between groups, measured after working-out. In order to examine these effects, we shall use paired t test, t test for independent samples, one-way and two-ways analysis of variance and analysis of covariance.

You can download the dataset here. The data is fictious.

Answers to the exercises are available here.

If you have different solution, feel free to post it.

Exercise 1

Load the data. Calculate descriptive statistics and test for the normality of both initial and final measurements for whole sample and for each group.

Exercise 2

Is there effect of exercises and what is the size of that effect for each group? (Tip: *You should use paired t test.*)

Exercise 3

Is the variance of the body mass on final measurement the same for each of the three groups? (Tip: *Use Levene's test for homogeneity of variances*)

Exercise 4

Is there a difference between groups on final measurement and what is the effect size? (Tip: *Use one-way ANOVA*)

Learn more about statistics for your experimental design in the online course [Learn By Example: Statistics and Data Science in R](). In this course you will learn how to:

- Work thru regression problems
- use different statistical tests and interpret them
- And much more

Exercise 5

Between which groups does the difference of body mass appear after the working-out? (Tip: *Conduct post-hoc test.*)

Exercise 6

What is the impact of age and working-out program on body mass on final measurement? (Tip: *Use two-way between groups ANOVA.*)

Exercise 7

What is the origin of effect of working-out program between subjects of different age? (Tip: *You should conduct post-hoc test.*)

Exercise 8

Is there a linear relationship between initial and final measurement of body mass for each group?

Exercise 9

Is there a significant difference in body mass on final measurement between groups, while controlling for initial measurement?

Exercise 10

How much of the variance is explained by independent variable? How much of the variance is explained by covariate?

Experimental Design Solutions

Below are the solutions to these exercises on Experimental design exercises

```
####################
#                  #
#    Exercise 1    #
#                  #
####################

data <- read.csv("experimental-design.csv")
as.factor(data$group) -> data$group
as.factor(data$age) -> data$age
summary(data$initial_mass)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   53.50   62.18   68.90   67.70   72.27   86.00

summary(data$final_mass)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   50.60   60.08   64.45   65.44   72.00   81.30
```

```
shapiro.test(data$initial_mass)
```

```
##
##      Shapiro-Wilk normality test
##
## data:  data$initial_mass
## W = 0.98306, p-value = 0.5053
```

```
shapiro.test(data$final_mass)
```

```
##
##      Shapiro-Wilk normality test
##
## data:  data$final_mass
## W = 0.97517, p-value = 0.2073
```

```
sapply(split(data$initial_mass, data$group), summary)
```

```
##             1     2     3
## Min.    53.50 53.50 54.50
## 1st Qu. 63.40 62.00 62.18
## Median  68.20 69.65 68.25
## Mean    67.24 68.26 67.58
## 3rd Qu. 70.00 73.00 72.72
## Max.    86.00 82.00 81.00
```

```
sapply(split(data$final_mass, data$group), summary)
```

```
##             1     2     3
## Min.    50.60 52.00 51.00
## 1st Qu. 60.08 60.25 61.00
## Median  62.95 70.50 65.00
## Mean    62.43 68.64 65.25
## 3rd Qu. 64.62 75.00 72.00
## Max.    81.30 81.00 77.00
```

```
sapply(split(data$initial_mass, data$group), shapiro.test)

##           1                             2
## statistic 0.9561618                     0.9735745
## p.value   0.415793                      0.7920937
## method    "Shapiro-Wilk normality test" "Shapiro-Wilk
normality test"
## data.name "X[[i]]"                      "X[[i]]"
##           3
## statistic 0.9777148
## p.value   0.8768215
## method    "Shapiro-Wilk normality test"
## data.name "X[[i]]"

sapply(split(data$final_mass, data$group), shapiro.test)

##           1                             2
## statistic 0.9447748                     0.9231407
## p.value   0.2479696                     0.08832153
## method    "Shapiro-Wilk normality test" "Shapiro-Wilk
normality test"
## data.name "X[[i]]"                      "X[[i]]"
##           3
## statistic 0.9453135
## p.value   0.2543017
## method    "Shapiro-Wilk normality test"
## data.name "X[[i]]"

####################
#                  #
#    Exercise 2    #
#                  #
####################


invisible(sapply(split(data, data$group), function(x)
    {
      t.test(x$initial_mass, x$final_mass, paired=TRUE) -> t
                                             cat(sprintf("Group
%d\r\nstatistic=%.3f\r\ndf=%d\r\np=%.3f\r\neta^2=%.3f\r\n\r\n"
```

```
,
                unique(x$group), t$statistic, t$parameter,
t$p.value,
              t$statistic^2/(t$statistic^2+t$parameter))

    }))

## Group 1

## statistic=7.474

## df=21

## p=0.000

## eta^2=0.727

##

## Group 2

## statistic=-0.687

## df=21

## p=0.500

## eta^2=0.022

##

## Group 3

## statistic=4.372

## df=21

## p=0.000

## eta^2=0.477
```

```
##


####################
#                  #
#    Exercise 3    #
#                  #
####################

library("car")
leveneTest(data$final_mass, data$group, center=mean)

## Levene's Test for Homogeneity of Variance (center = mean)
##       Df F value Pr(>F)
## group  2   1.232 0.2986
##       63

####################
#                  #
#    Exercise 4    #
#                  #
####################

print(summary(aov(final_mass ~ group, data)) -> f)

##             Df Sum Sq Mean Sq F value Pr(>F)
## group        2    425  212.64   3.626 0.0323 *
## Residuals   63   3694   58.64
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '
' 1

ss = f[[1]]$'Sum Sq'
paste("eta squared=", round(ss[1] / (ss[1]+ss[2]), 3))

## [1] "eta squared= 0.103"

####################
#                  #
#    Exercise 5    #
```

```
#                  #
####################

summary(f <- aov(final_mass ~ group, data))

##             Df Sum Sq Mean Sq F value Pr(>F)
## group        2    425  212.64   3.626 0.0323 *
## Residuals   63   3694   58.64
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '
' 1

TukeyHSD(f, "group")

##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = final_mass ~ group, data = data)
##
## $group
##          diff        lwr       upr     p adj
## 2-1  6.209091  0.6669343 11.751248 0.0245055
## 3-1  2.818182 -2.7239748  8.360338 0.4455950
## 3-2 -3.390909 -8.9330657  2.151248 0.3128050

# significant difference appears between 1st and 2nd group
(p<0.05)

####################
#                  #
#    Exercise 6    #
#                  #
####################

options(contrasts = c("contr.helmert", "contr.poly"))
m.lm <- lm(final_mass ~ age + group + age*group, data=data)
print(m.anova <- Anova(m.lm, type=3))

## Anova Table (Type III tests)
##
```

```
## Response: final_mass
##              Sum Sq Df   F value    Pr(>F)
## (Intercept) 267773  1 8536.0541 < 2.2e-16 ***
## age           1388  2   22.1282 7.725e-08 ***
## group          186  2    2.9678  0.059415 .
## age:group      564  4    4.4981  0.003152 **
## Residuals     1788 57
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '
' 1
```

```
m.anova[[1]][2:4] / (m.anova[[1]][2:4]+m.anova[[1]][5])
```

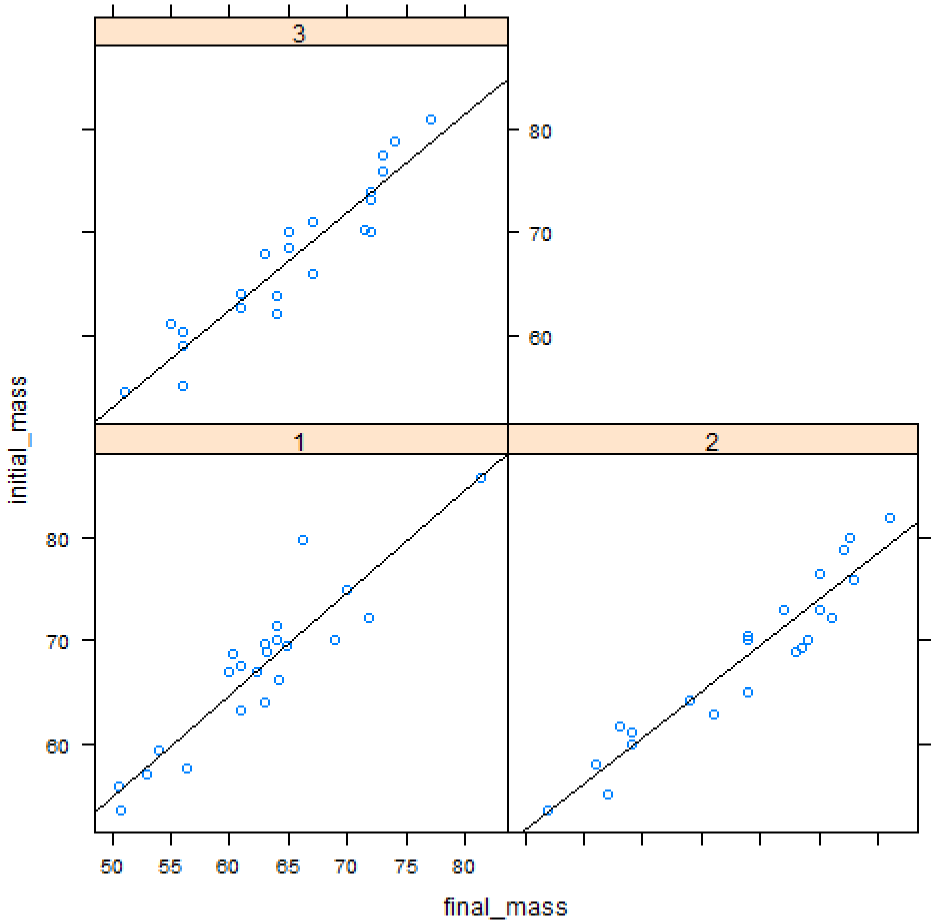
```
## [1] 0.43707242 0.09431139 0.23992294
```

```
####################
#                  #
#    Exercise 7    #
#                  #
####################

m.aov <- aov(final_mass ~ age + group +  age*group, data)
TukeyHSD(x=m.aov, "age")
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = final_mass ~ age + group + age * group,
data = data)
##
## $age
##                       diff        lwr       upr     p adj
## old-middle-age     6.69775   2.382683 11.012817 0.0012494
## young-middle-age  -5.75200  -9.564155 -1.939845 0.0017300
## young-old        -12.44975 -16.764817 -8.134683 0.0000000
```

```
# there is significant difference between all groups

####################
#                  #
```

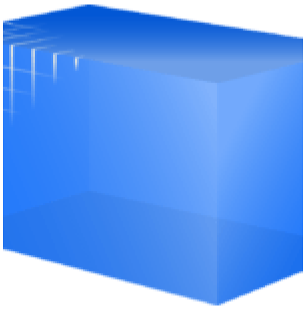
```
#    Exercise 8    #
#                  #
####################

library(lattice)
xyplot(initial_mass ~ final_mass | group, data = data,
panel=function(x, y, ...)
  {
  panel.xyplot(x, y, ...)
  panel.lmline(x, y, ...)
})
```




```
####################
#                  #
#    Exercise 9    #
#                  #
####################

model.1 = lm(final_mass~initial_mass, data=data)
model.2 = lm(final_mass~initial_mass+group, data=data)
```

```
anova(model.1, model.2)

## Analysis of Variance Table
##
## Model 1: final_mass ~ initial_mass
## Model 2: final_mass ~ initial_mass + group
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     64 746.24
## 2     62 442.63  2    303.62 21.264 9.286e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '
' 1

# there is the difference between groups while controlling
initial measurement (p < 0.05)

####################
#                  #
#    Exercise 10   #
#                  #
####################

model.3 = lm(final_mass~initial_mass+group, data=data)
library(heplots)
etasq(model.3, anova=TRUE)

## Anova Table (Type II tests)
##
## Response: final_mass
##              Partial eta^2 Sum Sq Df F value    Pr(>F)
## initial_mass       0.88019 3251.8  1 455.495 < 2.2e-16 ***
## group              0.40686  303.6  2  21.264 9.286e-08 ***
## Residuals                   442.6 62
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '
' 1

# independent variable (group) explains 41%, covariate
(initial_mass) explains 88%
```

Data Structures Exercises

There are 5 important basic data structures in R: vector, matrix, array, list and dataframe. They can be 1-dimensional (vector and list), 2-dimensional (matrix and data frame) or multidimensional (array). They also differ according to homogeneity of elements they can contain: while all elements contained in vector, matrix and array must be of the same type, list and data frame can contain multiple types.

In this set of exercises we shall practice casting between different types of these data structures, together with some basic operations on them. You can find more about data structures on [Advanced R – Data structures] page.

Answers to the exercises are available [here].

If you have different solution, feel free to post it.

Exercise 1

Create a vector named v which contains 10 random integer values between -100 and +100.

Exercise 2

Create a two-dimensional 5×5 array named a comprised of sequence of even integers greater than 25.

Create a list named s containing sequence of 20 capital letters, starting with ‘C’.

Exercise 3

Create a list named l and put all previously created objects in it. Name them a, b and c respectively. How many elements are there in the list? Show the structure of the list. Count all elements recursively.

Exercise 4

Without running commands in R, answer the following questions:

1. what is the result of l[[3]]?
2. How would you access random-th letter in the list element c?
3. If you convert list l to a vector, what will be the type of it's elements?
4. Can this list be converted to an array? What will be the data type of elements in array?

Check the results with R.

Exercise 5

Remove letters from the list l. Convert the list l to a vector and check its class. Compare it with the result from exercise 4, question #3.

Exercise 6

Find the difference between elements in l[["a"]] and l[["b"]]. Find the intersection between them. Is there number 33 in their union?

Exercise 7

Create 5×5 matrix named m and fill it with random numeric values rounded to two decimal places, ranging from 1.00 to 100.00.

Exercise 8

Answer the following question without running R command, then check the result.

What will be the class of data structure if you convert matrix m to:

- vector
- list
- data frame
- array?

Exercise 9

Transpose array l$b and then convert it to matrix.

Exercise 10

Get union of matrix m and all elements in list l and sort it ascending.

Data Structures Solutions

Below are the solutions to these exercises on data structures.

```
####################
#                  #
#    Exercise 1    #
#                  #
####################

v <- sample(-100:100, 10, replace=TRUE)

####################
#                  #
#    Exercise 2    #
#                  #
```

```
####################

a <- array(seq(from = 26, length.out = 25, by = 2), c(5, 5))
s <- LETTERS[match("C", LETTERS):(match("C", LETTERS)+19)]

####################
#                  #
#    Exercise 3    #
#                  #
####################

l <- list(a = v, b = a, c = s)
length(l)

## [1] 3

str(l)

## List of 3
##  $ a: int [1:10] -83 72 -44 71 -54 -17 -40 -76 22 58
##  $ b: num [1:5, 1:5] 26 28 30 32 34 36 38 40 42 44 ...
##  $ c: chr [1:20] "C" "D" "E" "F" ...

length(unlist(l))

## [1] 55

####################
#                  #
#    Exercise 4    #
#                  #
####################

l[[3]]

##  [1] "C" "D" "E" "F" "G" "H" "I" "J" "K" "L" "M" "N" "O"
"P" "Q" "R" "S"
## [18] "T" "U" "V"
```

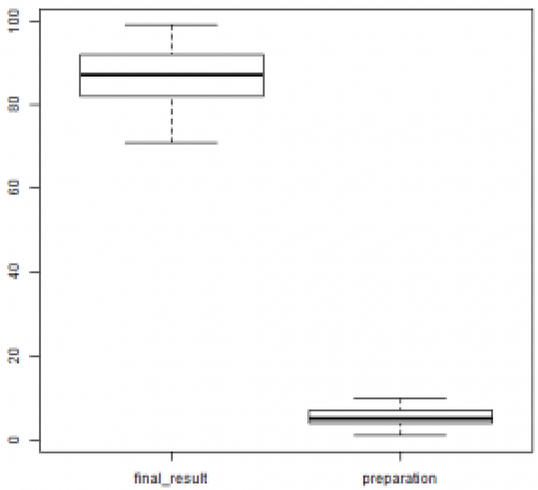
```
l[[3]][sample(1:length(l[[3]]), 1)]

## [1] "O"

class(unlist(l))

## [1] "character"

x <- array(l)
class(x[1])

## [1] "list"

####################
#                  #
#    Exercise 5    #
#                  #
####################

l$c <- NULL
class(unlist(l))

## [1] "numeric"

####################
#                  #
#    Exercise 6    #
#                  #
####################

setdiff(l$a, l$b)

## [1] -83 -44  71 -54 -17 -40 -76  22

intersect(l$a, l$b)

## [1] 72 58
```

```
33 %in% union(l$a, l$b)

## [1] FALSE

####################
#                  #
#    Exercise 7    #
#                  #
####################

m <- matrix(data = round(runif(5*5, 0.99, 100.00), 2), nrow =
5)

####################
#                  #
#    Exercise 8    #
#                  #
####################

class(as.vector(m))

## [1] "numeric"

class(as.list(m))

## [1] "list"

class(as.data.frame(m))

## [1] "data.frame"

class(as.array(m))

## [1] "matrix"

####################
#                  #
#    Exercise 9    #
#                  #
```

```
####################

as.matrix(aperm(l$b))

##      [,1] [,2] [,3] [,4] [,5]
## [1,]   26   28   30   32   34
## [2,]   36   38   40   42   44
## [3,]   46   48   50   52   54
## [4,]   56   58   60   62   64
## [5,]   66   68   70   72   74

####################
#                  #
#    Exercise 10   #
#                  #
####################

sort(union(as.vector(m), unlist(l)))

##  [1] -83.00 -76.00 -54.00 -44.00 -40.00 -17.00   8.02
9.58  10.41  10.46
## [11]  10.51  16.28  20.85  22.00  22.33  25.58  25.66
26.00  27.96  28.00
## [21]  28.07  30.00  32.00  34.00  36.00  37.02  38.00
38.36  40.00  42.00
## [31]  44.00  45.22  46.00  48.00  50.00  52.00  53.18
54.00  56.00  58.00
## [41]  60.00  62.00  64.00  66.00  67.11  68.00  70.00
71.00  72.00  73.88
## [51]  74.00  74.64  83.52  89.62  89.72  91.35  99.19
99.45
```

Student’s Achievement

Research Project – Exercises



In this set of exercises we shall follow standard practice of conducting a research project. The goal of the research is to find the relationship between student's preparations and his achievement on the final exam. Preparations are viewed as the amount of time student spends on preparatory classes and score in mathematics achieved in the final year of school.

Here is the data set.

Answers to the exercises are available here.

If you have different solution, feel free to post it.

Exercise 1

Load the data and check if the sample size is large enough for conducting multivariate linear regression? *Tip: sample size is "large enough" if it is greater then 50 + 8 * m, where m is the number of predictor variables*

Exercise 2

Calculate descriptive statistics for criterion variable. Was the final test appropriate for the level of knowledge of students? *(Tip: you check it by checking skewness of distribution – we expect the distribution to be symmetric.)*

Exercise 3

Do the students with good score in mathematics in final year differ from those with bed scores regarding the results on the final exam? Did the students with good score in mathematics in final year attend preparatory classes more than those with bed score?

Exercise 4

Calculate correlation matrix for three variables included in a model. Can we expect a multicolinearity problem? Does the correlation between predictor variables justify conducting multiple regression?

Exercise 5

Create multiple linear regression model m to check if number of preparatory classes and score in mathematics in the final year can explain the result on final test.

Exercise 6

Find and eliminate outliers from the data.

Exercise 7

Using the scatter plot, check for the linearity of residual of model m.

Exercise 8

Test the normality of residual of model m.

Exercise 9

1. Is model m statistically significant on the level of 0.05?
2. Which predictor variables significantly contribute to the explanation of criterion variable?

Exercise 10

Does introduction of gender as a predictor variable adds to the explanatory power of the model?

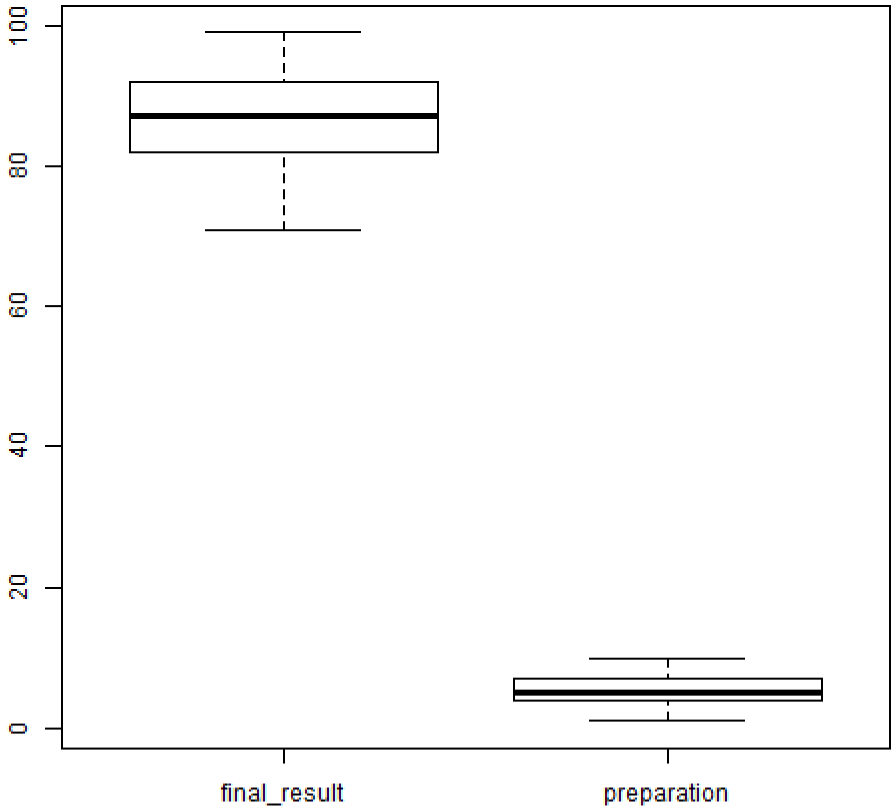
Student's Achievement Research Project – Solutions

Below are the solutions to these exercises on conducting research project in a school.

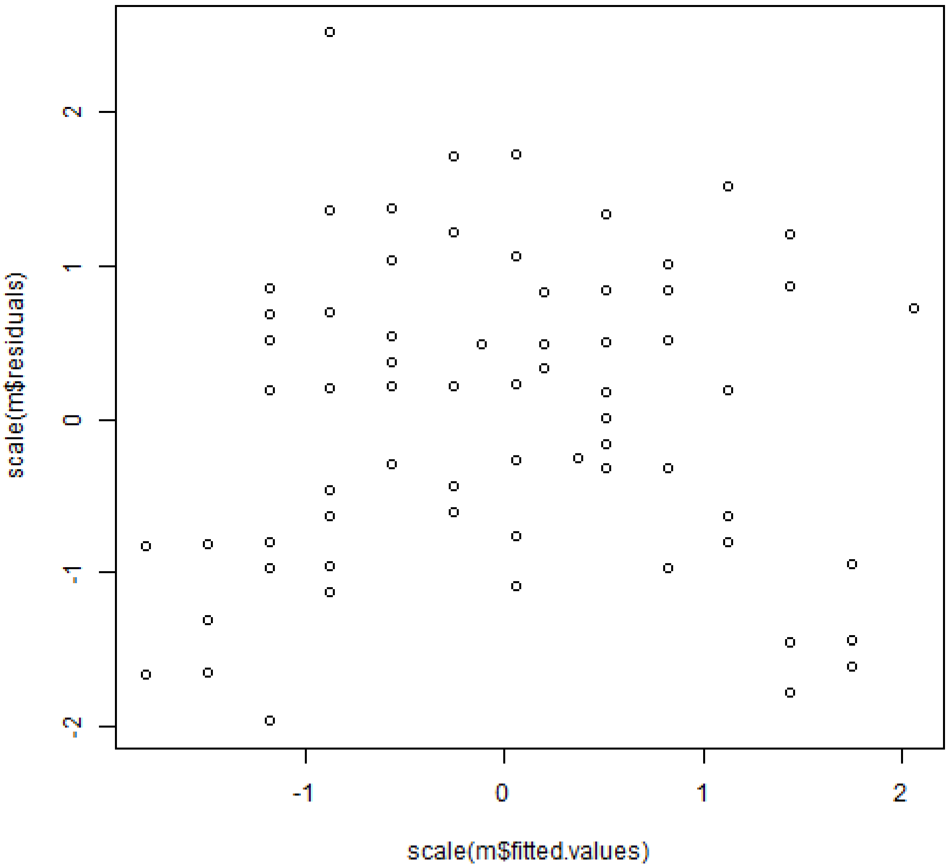
```
####################
#                  #
#    Exercise 1    #
#                  #
####################

data <- read.csv2("school-research.csv")
nrow(data) > 50 + 8 * 2

## [1] TRUE

####################
#                  #
#    Exercise 2    #
#                  #
####################

summary(data$final_result)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   71.00   82.00   87.00   86.42   92.00   99.00

library("moments")
skewness(data$final_result)
```

```
## [1] -0.2864368

# curve is mildly skwewd to the left, which means that the
test was a bit easier

####################
#                  #
#    Exercise 3    #
#                  #
####################

t.test(data$final_result, data$maths, alternative =
"two.sided", paired=FALSE)

##
##      Welch Two Sample t-test
##
## data:  data$final_result and data$maths
## t = 114.43, df = 80.88, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not
equal to 0
## 95 percent confidence interval:
##  84.48041 87.47021
## sample estimates:
##  mean of x  mean of y
## 86.4197531  0.4444444

t.test(data$preparation, data$marths, alternative = "greater",
paired=FALSE)

##
##      One Sample t-test
##
## data:  data$preparation
## t = 22.985, df = 80, p-value < 2.2e-16
## alternative hypothesis: true mean is greater than 0
## 95 percent confidence interval:
##  4.901382      Inf
## sample estimates:
## mean of x
```

```
##  5.283951

####################
#                  #
#    Exercise 4    #
#                  #
####################

cor(data[c(1, 2, 4)], method="pearson")

##              final_result preparation     maths
## final_result    1.0000000   0.3900393 0.3629136
## preparation     0.3900393   1.0000000 0.4323024
## maths           0.3629136   0.4323024 1.0000000

# 1. no
# 2. yes, since the correlation is moderate

####################
#                  #
#    Exercise 5    #
#                  #
####################

m <- lm(data$final_result ~ data$preparation+data$maths)

####################
#                  #
#    Exercise 6    #
#                  #
####################

boxplot(data[c(1, 2)])$outliers
```




```
## NULL

# there are no outliers

####################
#                  #
#    Exercise 7    #
#                  #
####################

plot(scale(m$fitted.values), scale(m$residuals))
```




```
# since there is no pattern, we conclude that relationship is
linear

####################
#                  #
#    Exercise 8    #
#                  #
####################

shapiro.test(scale(m$residuals))$p.value > 0.05

## [1] TRUE

####################
#                  #
#    Exercise 9    #
#                  #
####################
```

```
f <- summary(m)$fstatistic
pf(f[1], f[2], f[3], lower.tail = F) < 0.05

## value
##  TRUE

summary(m)$coefficients[c(2,3), 4] < 0.05

## data$preparation       data$maths
##             TRUE             TRUE

####################
#                  #
#    Exercise 10   #
#                  #
####################

n <- lm(data$final_result ~ data$preparation+data$maths+data$gender)
f <- summary(n)$fstatistic
(summary(m)$adj.r.squared < summary(n)$adj.r.squared) && (pf(f[1], f[2], f[3], lower.tail = F) < 0.05)

## [1] TRUE
```

String Manipulation – Exercises

In this set of exercises we will practice functions that enable us to manipulate strings.You can find more about string manipulation functions in [Handling and Processing Strings in R]() e-book.

Answers to the exercises are available [here]().

If you have different solution, feel free to post it.

Exercise 1

Load text from the [file]() and print it on screen. Text file contains excerpt from novel “Gambler” by Fyodor Dostoyevsky.

Exercise 2

How many paragraphs is there in the excerpt?

Exercise 3

How many characters is there in the excerpt

Exercise 4

Collapse paragraphs into one and display it on the screen (un-list it).

Exercise 5

Convert the text to uppercase and save it to new file “gambler-upper.txt”.

Exercise 6

Change all letters ‘a’ and ‘t’ to ‘A’ and ‘T’.

Exercise 7

Does the text contain word ‘lucky’?

Exercise 8

How many words are there in the excerpt, assuming that words are sub-strings separated by space or new line character?

Exercise 9

How many times is word money repeated in the excerpt?

Exercise 10

Ask the user to input two numbers, divide them and display both numbers and the result on the screen, each of them formatted to 2 decimal places.

String Manipulation – Solutions

Below are the solutions to these exercises on functions that are used to manipulate strings.

```
####################
#                  #
#    Exercise 1    #
#                  #
####################

gambler <-
readLines("http://www.r-exercises.com/wp-content/uploads/2016/11/gambler.txt")
noquote(gambler)

## [1] At length I returned from two weeks leave of absence to find that my patrons had arrived three days ago in Roulettenberg. I received from them a welcome quite different to that which I had expected. The General eyed me coldly, greeted me in rather haughty fashion, and dismissed me to pay my respects to his sister. It was clear that from SOMEWHERE
```

money had been acquired. I thought I could even detect a certain shamefacedness in the General's glance. Maria Philipovna, too, seemed distraught, and conversed with me with an air of detachment. Nevertheless, she took the money which I handed to her, counted it, and listened to what I had to tell. To luncheon there were expected that day a Monsieur Mezentsov, a French lady, and an Englishman; for, whenever money was in hand, a banquet in Muscovite style was always given. Polina Alexandrovna, on seeing me, inquired why I had been so long away. Then, without waiting for an answer, she departed. Evidently this was not mere accident, and I felt that I must throw some light upon matters. It was high time that I did so.

[2] I was assigned a small room on the fourth floor of the hotel (for you must know that I belonged to the General's suite). So far as I could see, the party had already gained some notoriety in the place, which had come to look upon the General as a Russian nobleman of great wealth. Indeed, even before luncheon he charged me, among other things, to get two thousand-franc notes changed for him at the hotel counter, which put us in a position to be thought millionaires at all events for a week! Later, I was about to take Mischa and Nadia for a walk when a summons reached me from the staircase that I must attend the General. He began by deigning to inquire of me where I was going to take the children; and as he did so, I could see that he failed to look me in the eyes. He WANTED to do so, but each time was met by me with such a fixed, disrespectful stare that he desisted in confusion. In pompous language, however, which jumbled one sentence into another, and at length grew disconnected, he gave me to understand that I was to lead the children altogether away from the Casino, and out into the park. Finally his anger exploded, and he added sharply:

[3] "I suppose you would like to take them to the Casino to play roulette? Well, excuse my speaking so plainly, but I know how addicted you are to gambling. Though I am not your mentor, nor wish to be, at least I have a right to require that you shall not actually compromise me."

[4] "I have no money for gambling," I quietly replied.

####################

#   #

```
#    Exercise 2    #
#                  #
####################

length(gambler)

## [1] 4

####################
#                  #
#    Exercise 3    #
#                  #
####################

nchar(gambler)

## [1] 1073 1158  276   50

####################
#                  #
#    Exercise 4    #
#                  #
####################

t <- paste(gambler, collapse="\n")
cat(t)

## At length I returned from two weeks leave of absence to find that my patrons had arrived three days ago in Roulettenberg. I received from them a welcome quite different to that which I had expected. The General eyed me coldly, greeted me in rather haughty fashion, and dismissed me to pay my respects to his sister. It was clear that from SOMEWHERE money had been acquired. I thought I could even detect a certain shamefacedness in the General's glance. Maria Philipovna, too, seemed distraught, and conversed with me with an air of detachment. Nevertheless, she took the money which I handed to her, counted it, and listened to what I had to tell. To luncheon there were expected that day a Monsieur Mezentsov, a French lady, and an Englishman; for, whenever money was in
```

hand, a banquet in Muscovite style was always given. Polina Alexandrovna, on seeing me, inquired why I had been so long away. Then, without waiting for an answer, she departed. Evidently this was not mere accident, and I felt that I must throw some light upon matters. It was high time that I did so.
I was assigned a small room on the fourth floor of the hotel (for you must know that I belonged to the General's suite). So far as I could see, the party had already gained some notoriety in the place, which had come to look upon the General as a Russian nobleman of great wealth. Indeed, even before luncheon he charged me, among other things, to get two thousand-franc notes changed for him at the hotel counter, which put us in a position to be thought millionaires at all events for a week! Later, I was about to take Mischa and Nadia for a walk when a summons reached me from the staircase that I must attend the General. He began by deigning to inquire of me where I was going to take the children; and as he did so, I could see that he failed to look me in the eyes. He WANTED to do so, but each time was met by me with such a fixed, disrespectful stare that he desisted in confusion. In pompous language, however, which jumbled one sentence into another, and at length grew disconnected, he gave me to understand that I was to lead the children altogether away from the Casino, and out into the park. Finally his anger exploded, and he added sharply:
"I suppose you would like to take them to the Casino to play roulette? Well, excuse my speaking so plainly, but I know how addicted you are to gambling. Though I am not your mentor, nor wish to be, at least I have a right to require that you shall not actually compromise me."
"I have no money for gambling," I quietly replied.

```
####################
#                  #
#    Exercise 5    #
#                  #
####################

cat(toupper(gambler), file="gambler-output.txt")

####################
```

```
#                  #
#    Exercise 6    #
#                  #
####################

chartr("at", "AT", gambler)

## [1] "AT lengTh I reTurned from Two weeks leAve of Absence
To find ThAT my pATrons hAd Arrived Three dAys Ago in
RouleTTenberg. I received from Them A welcome quiTe differenT
To ThAT which I hAd expecTed. The GenerAl eyed me coldly,
greeTed me in rAther hAughTy fAshion, And dismissed me To pAy
my respecTs To his sisTer. IT wAs cleAr ThAT from SOMEWHERE
money hAd been Acquired. I ThoughT I could even deTecT A
cerTAin shAmefAcedness in The GenerAl's glAnce. MAriA
PhilipovnA, Too, seemed disTrAughT, And conversed wiTh me wiTh
An Air of deTAchmenT. NeverTheless, she Took The money which I
hAnded To her, counTed iT, And lisTened To whAT I hAd To Tell.
To luncheon There were expecTed ThAT dAy A Monsieur MezenTsov,
A French lAdy, And An EnglishmAn; for, whenever money wAs in
hAnd, A bAnqueT in MuscoviTe sTyle wAs AlwAys given. PolinA
AlexAndrovnA, on seeing me, inquired why I hAd been so long
AwAy. Then, wiThouT wAiTing for An Answer, she depArTed.
EvidenTly This wAs noT mere AccidenT, And I felT ThAT I musT
Throw some lighT upon mATTers. IT wAs high Time ThAT I did
so."
## [2] "I wAs Assigned A smAll room on The fourTh floor of The
hoTel (for you musT know ThAT I belonged To The GenerAl's
suiTe). So fAr As I could see, The pArTy hAd AlreAdy gAined
some noTorieTy in The plAce, which hAd come To look upon The
GenerAl As A RussiAn noblemAn of greAT weAlTh. Indeed, even
before luncheon he chArged me, Among oTher Things, To geT Two
ThousAnd-frAnc noTes chAnged for him AT The hoTel counTer,
which puT us in A posiTion To be ThoughT millionAires AT All
evenTs for A week! LATer, I wAs AbouT To TAke MischA And NAdiA
for A wAlk when A summons reAched me from The sTAircAse ThAT I
musT ATTend The GenerAl. He begAn by deigning To inquire of me
where I wAs going To TAke The children; And As he did so, I
could see ThAT he fAiled To look me in The eyes. He WANTED To
do so, buT eAch Time wAs meT by me wiTh such A fixed,
disrespecTful sTAre ThAT he desisTed in confusion. In pompous
```

```
lAnguAge, however, which jumbled one senTence inTo AnoTher,
And AT lengTh grew disconnecTed, he gAve me To undersTAnd ThAT
I wAs To leAd The children AlTogeTher AwAy from The CAsino,
And ouT inTo The pArk. FinAlly his Anger exploded, And he
Added shArply:"
## [3] "\"I suppose you would like To TAke Them To The CAsino
To plAy rouleTTe? Well, excuse my speAking so plAinly, buT I
know how AddicTed you Are To gAmbling. Though I Am noT your
menTor, nor wish To be, AT leAsT I hAve A righT To require
ThAT you shAll noT AcTuAlly compromise me.\""
## [4] "\"I hAve no money for gAmbling,\" I quieTly replied."

####################
#                  #
#    Exercise 7    #
#                  #
####################

'lucky' %in% gambler

## [1] FALSE

####################
#                  #
#    Exercise 8    #
#                  #
####################

w <- strsplit(t, " ")
length(w[[1]])

## [1] 470

####################
#                  #
#    Exercise 9    #
#                  #
####################

sum(w[[1]][] == 'money')
```

```
## [1] 4
```

```
####################
#                  #
#    Exercise 10   #
#                  #
####################

numbers <- scan(n=2)
sprintf("%.2f / %.2f = %.2f", numbers[1], numbers[2],
numbers[1]/numbers[2])
```

```
## [1] "1.00 / 6.00 = 0.17"
```