



From jamovi to R

Sebastian Jentschke





Overview

- some basic introduction into R
- using R within jamovi (Rj)
- using jamovi in R (`jmv`, `jmvconnect`, `jmvReadWrite`)
- two use cases: using analysis results (regression coefficients) and importing text data





Installing the requirements

- R: <https://cran.uib.no/>
- RStudio [recommended]:
<https://www.rstudio.com/products/rstudio/download/#download>
- packages in R:

```
install.packages(c("jmv", "jmvconnect",  
"jmvReadWrite"), dependencies = TRUE)  
install.packages("haven")
```





Basic introduction into R



Introduction into R-programming

- **variables:**
 - basic: character, complex, numeric, integer, logical
- **structures / containers:**
 - same basic type:
`c()`, `vector()` → `vector[]` – 1D
`matrix()` – 2D
 - can combine types:
list: `list(a = ..., b = ...)` → `list[[]]`
data frame: `data.frame(A, B)` – A, B are vectors / lists
→ `df[1:2,]` – first 2 rows or `df[df$A == "...",]`
`df[, 1:2]` – first 2 columns





Introduction into R-programming

control structures – branching: if-else

```
if (...) {  
    ...  
} else if (...) {  
    ...  
} else {  
    ...  
}
```





Introduction into R-programming

control structures – loops: for, while, apply

```
for (crrSbj in c('subj1', 'subj2', 'subj3')) {
```

```
  ...
```

```
} # you know the number of elements to process
```

```
while (...) {
```

```
  ...
```

```
} # you don't know the number of elements to process
```

```
[break, next]
```

```
apply, sapply, lapply(object, function)
```





Introduction into R-programming

inspect and handle objects:

`ls()` – lists objects available in the current workspace

`rm()` – removes objects from the current workspace

`rm(list = ls())` – removes all objects

`str()` – shows the structure of data frames (and other objects)





Introduction into R-programming

- read text files: `read.csv`, `read.delim`
`read.csv(paste0(dirDta, crrSbj, '_NEO-PI.log'), header = FALSE, sep = "\t")`
- store file (Rdata-files, RDS-files)
`save(list = 'dtaFrm', file = 'FileName.Rdata')`
`saveRDS('object', file = 'FileName.rds')`
- concatenate strings: `paste()`, `paste0()`
`paste0(dirDta, crrSbj, '_NEO-PI.log')`
- arithmetic and statistic:
`mean()` / `colMeans()`, `max()`, `min()`, `range()`, `rank()`,
`scale()`, `sum()` / `colSums()`, `stats::sd()`, `table()`





Using R within jamovi (Rj)



Using R within jamovi (Rj)

some remarks on data / measurement types:

- nominal → R: factor
- ordinal → R: ordered factor
- continuous → R: numeric
- ➔ conversion: `as.numeric / jmvcore::toNumeric`,
`as.factor`, `as.ordered`, `rank`
`sapply(data[2:26], jmvcore::toNumeric)`

need for speed: `package::` vs. `library(package)`





Using R within jamovi (Rj)

accessing you data:

- by column number: `data[2:26]`
- by row and column number: `data[1, 2:26]`
- by column name(s): `data[, "ID"]`, `data[, c("var1", "var2", "var3")]`
- convert to vector (expected by most R commands):
`mean(data[["var1"]])`
`sapply(data[c("var1", "var2", "var3")], mean)`





Using R within jamovi (Rj)

```
sort(rownames(installed.packages()))
```

- `car` (extended regression)
- `cluster`
- `gnm` (generalized nonlinear models)
- `lavaan`
- `lme4`, `nlme`, `lmerTest` (linear / non-linear mixed effects models)
- `psych` (e.g., factor analysis)
- `ROCR` (ROC, sensitivity [TP] / specificity [1 - FP])
- `stats` (part of base R, wealth of different methods, e.g., `lm`, `hclust`)
- `survival` (time for “units” to achieve a given status)
- [`ggplot2` (plotting), `plyr` (data splitting)]





Using R within jamovi (Rj)

typical use cases:

- finding / removing outliers (cf. Mahalanobis)
- analyses that go beyond what is available in jamovi or as modules but is not as “specialized” to require R-packages not included in jamovi
- quick-and-dirty (e.g., trying out code for module development)





Tip for learning (a bit of) R

- install jmvbaseR
+ → find the module and install it
- switch on the Syntax mode
⋮ → tick “Syntax mode”
- run one of the available analyses and see the output and the syntax as you were in R





Code snippets for use in Rj

Modules → jamovi to R → Rj_Examples.R





Using jamovi in R



Using jamovi in R

[1] Accessing your files in R:

- Option 1: `jmvconnect`

`jmvconnect::what()` – *lists the available dataset*

`jmvconnect::read()` – *reads those datasets*

(either by number, e.g., 2, or name / title, e.g., “Tooth Growth”)

- Option 2: `jmvReadWrite`

`jmvReadWrite::read_omv([FILENAME], sveAtt = TRUE, getSyn = TRUE, useFlt = FALSE)`

reads jamovi files into R; sveAtt – extracts the attributes, getSyn – extracts the syntax of the analyses, useFlt – apply filter(s) (remove filtered rows)





Using jamovi in R

[2] Accessing / manipulating data from your files:

- based upon your raw data: e.g., creating plots or running analyses not available from jamovi
- based upon your analyses: run syntax and extract the results; e.g., extract regression coefficients and use them for prediction, or use results in Rmd (R markdown)





Using jamovi in R

[3] Helper functions for file handling tasks:

(see <https://sjentsch.github.io/jmvReadWrite/reference/index.html> → Helper functions for further details, have a look at the Examples)

- `jmvReadWrite::convert_to_omv()`
- `jmvReadWrite::long2wide_omv()`
- `jmvReadWrite::wide2long_omv()` → format for mixed-model-analyses
- `jmvReadWrite::merge_cols_omv()`
- `jmvReadWrite::merge_rows_omv()`
- `jmvReadWrite::sort_omv()`
- **coming soon:** `jmvReadWrite::compute()`,
`jmvReadWrite::recode()`





Two use cases



Use case I

Use regression coefficients for prediction:

Modules → jamovi to R → ExtractCoefs.R

if you had not enough yet, there is another similar example with explanations:

Modules → jamovi to R → jmvExample.R





Use case II

extract data from text files (one per participant):

Modules → jamovi to R → ReadData.R +

NeoCard.zip

(put them in one directory and extract NeoCard.zip)





Summary

Further resources



Summary

- some basic introduction into R
- using R within jamovi (Rj)
- using jamovi in R (`jmv`, `jmvconnect`, `jmvReadWrite`)
- two use cases: using analysis results (regression coefficients) and importing text data





How to get help?

- **generally:** your search engine is your friend
"R [what do you want to do]"
"R [error message]"
- **some history of R:**
https://www.youtube.com/watch?v=iq_biXEIx-U
<https://github.com/revodavid/20-years-of-R>
- **developing modules for jamovi:**
<https://jamovi.readthedocs.io> → developer hub





**Thank you for your
attention!**



UNIVERSITY OF BERGEN