

R Schnittstellen - der Austausch von Daten

Jan-Philipp Kolb

8 Mai 2017

Einführung und Motivation

Pluspunkte von R

- Als Weg kreativ zu sein ...
- Graphiken, Graphiken, Graphiken
- In Kombination mit anderen Programmen nutzbar
- Zur Verbindung von Datenstrukturen
- Zum Automatisieren
- Um die Intelligenz anderer Leute zu nutzen ;-)
- ...

Gründe

- R ist frei verfügbar. Es kann umsonst runtergeladen werden.
- R ist eine Skriptsprache
- Gute Möglichkeiten für die Visualisierung
- R wird immer populärer
- Popularität von R ist in vielen Bereichen sehr hoch.

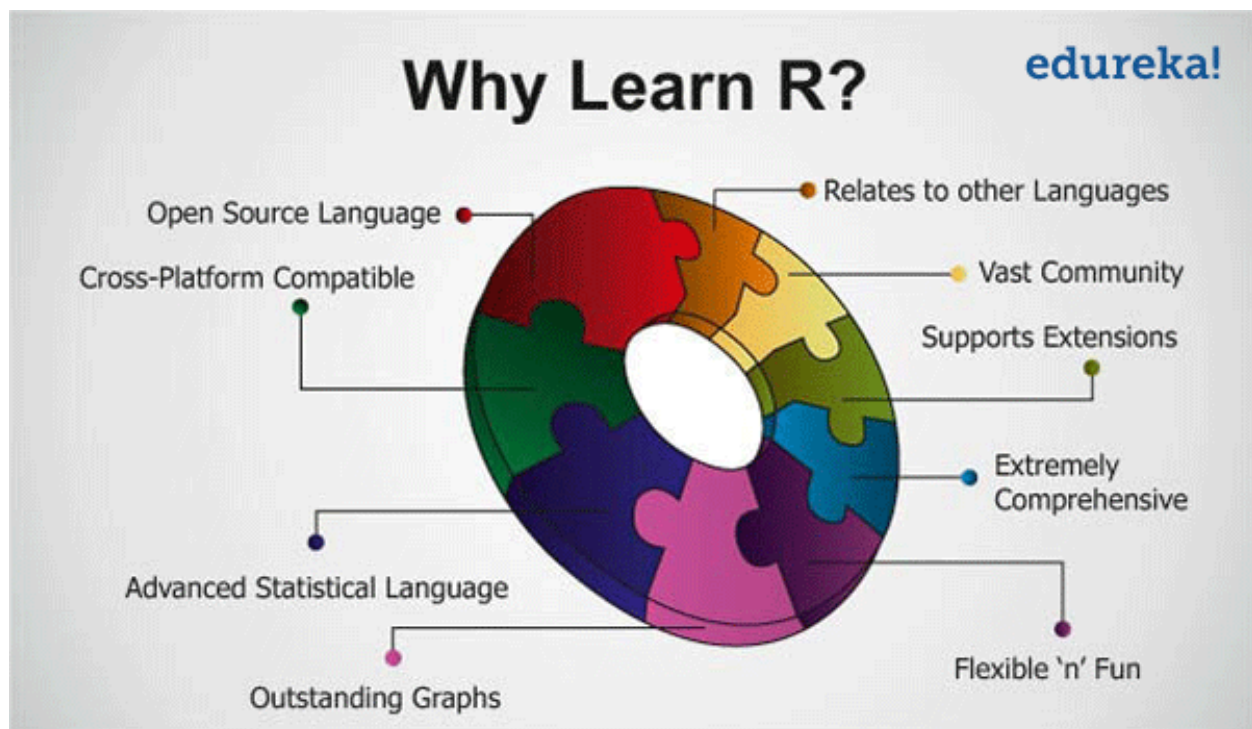


Figure 1:

Warum R?

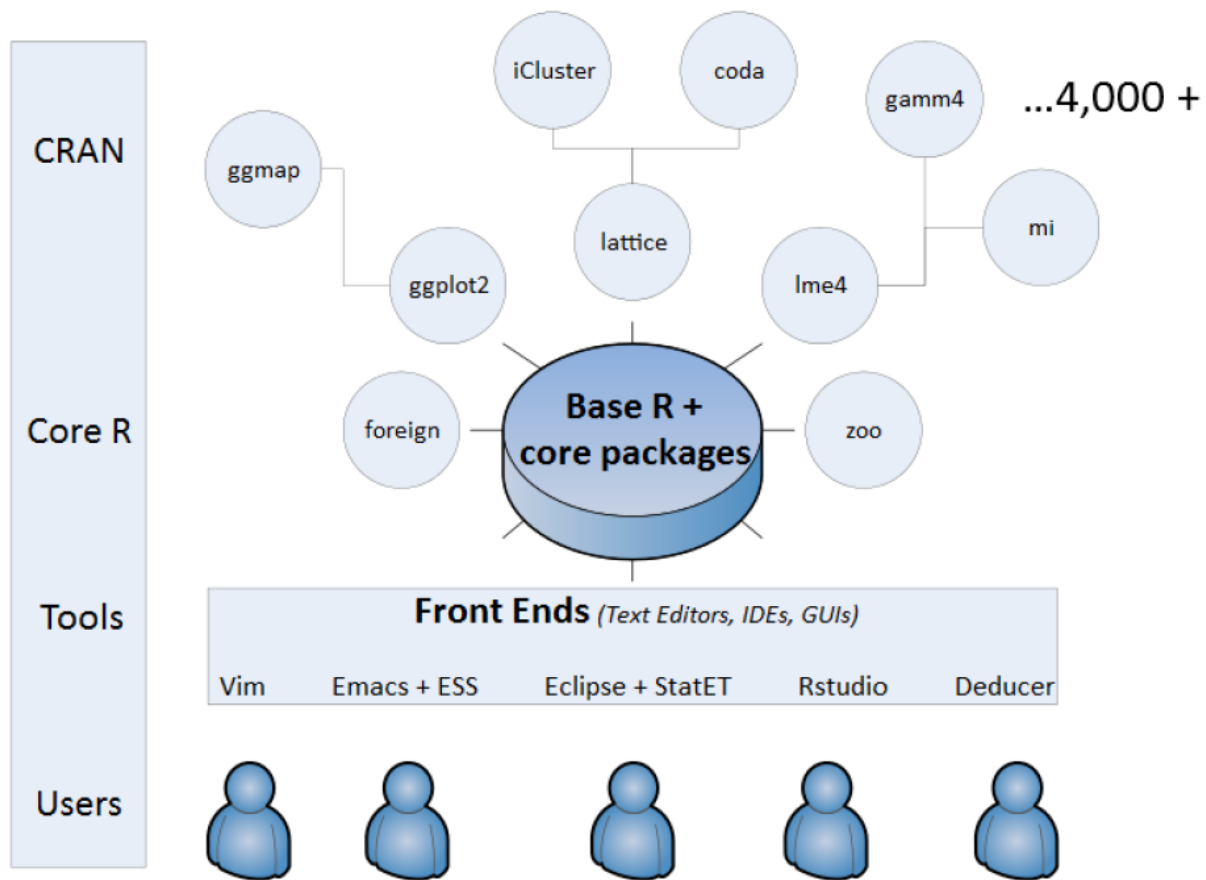


Figure 2: Modularer Aufbau

Die am meisten heruntergeladenen Pakete

CRAN Task Views

Motivation - Nachteile von R

1. Daten werden oft anderswo erfasst/eingegeben (oft Excel, SPSS etc.)
2. Nicht jeder ist bereit mit R zu arbeiten
3. Nicht auf jedem Rechner ist R installiert
4. R ist manchmal zu langsam
5. Schwierigkeiten bei der Arbeit mit großen Datenmengen

Was folgt daraus

1. Schnittstelle zu SPSS/Stata/Excel zum Import von Daten
2. Schnittstelle zu Word/LaTeX
3. Möglichkeit HTML Präsentationen zu erzeugen

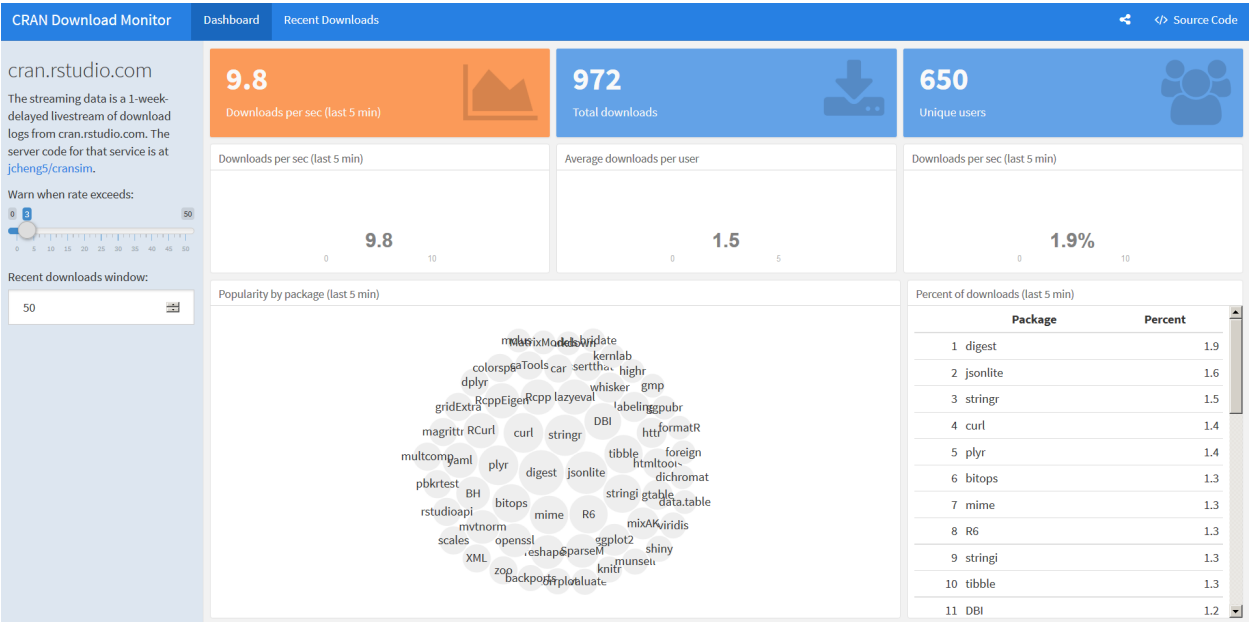


Figure 3:

CRAN Task Views

Bayesian	Bayesian Inference
ChemPhys	Chemometrics and Computational Physics
ClinicalTrials	Clinical Trial Design, Monitoring, and Analysis
Cluster	Cluster Analysis & Finite Mixture Models
DifferentialEquations	Differential Equations
Distributions	Probability Distributions
Econometrics	Econometrics
Environmetrics	Analysis of Ecological and Environmental Data
ExperimentalDesign	Design of Experiments (DoE) & Analysis of Experimental Data
ExtremeValue	Extreme Value Analysis
Finance	Empirical Finance

Figure 4:

4. Nutzung von C++
5. Nutzung von Datenbanken

Die Nutzung von Schnittstellen beim Import/Export

- Interaktion mit Excel, SPSS, Stata, ...



Figure 5: Import

Reproducible Research

Was wird bei Wikipedia unter Reproducibility verstanden?

Darstellung von Ergebnissen

- Mit der Schnittstelle zu Javascript lassen sich interaktive Graphiken erzeugen
- Diese kann man auf Websites, in Präsentationen oder in Dashboards verwenden

Warum die Schnittstelle zu C++?

- Wenn Schnelligkeit wichtig ist, bietet sich C++ an.
- Dies kann bspw. der Fall sein, wenn sich Schleifen nicht vermeiden lassen.
- Man wird bei der Programmierung durch RStudio unterstützt.
- Es gibt eine Rcpp Galerie, wo man sich Anregungen holen kann.
- Allerdings sollte man zunächst versuchen den Rcode so schnell wie möglich zu gestalten.

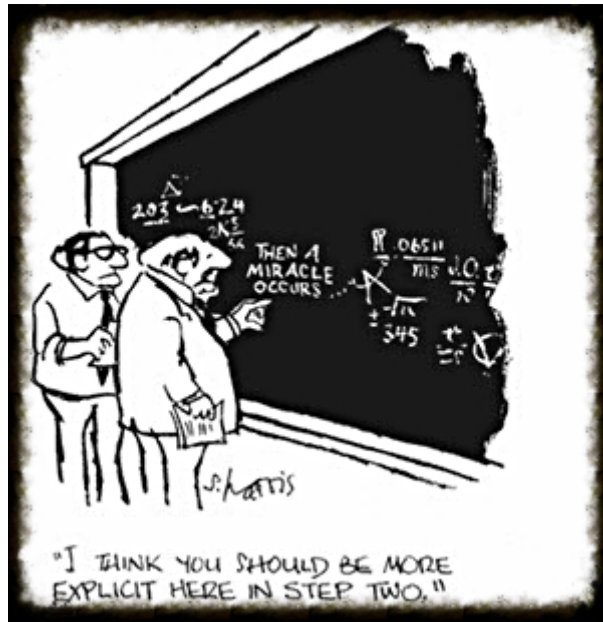


Figure 6:

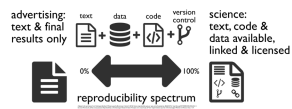


Figure 7:

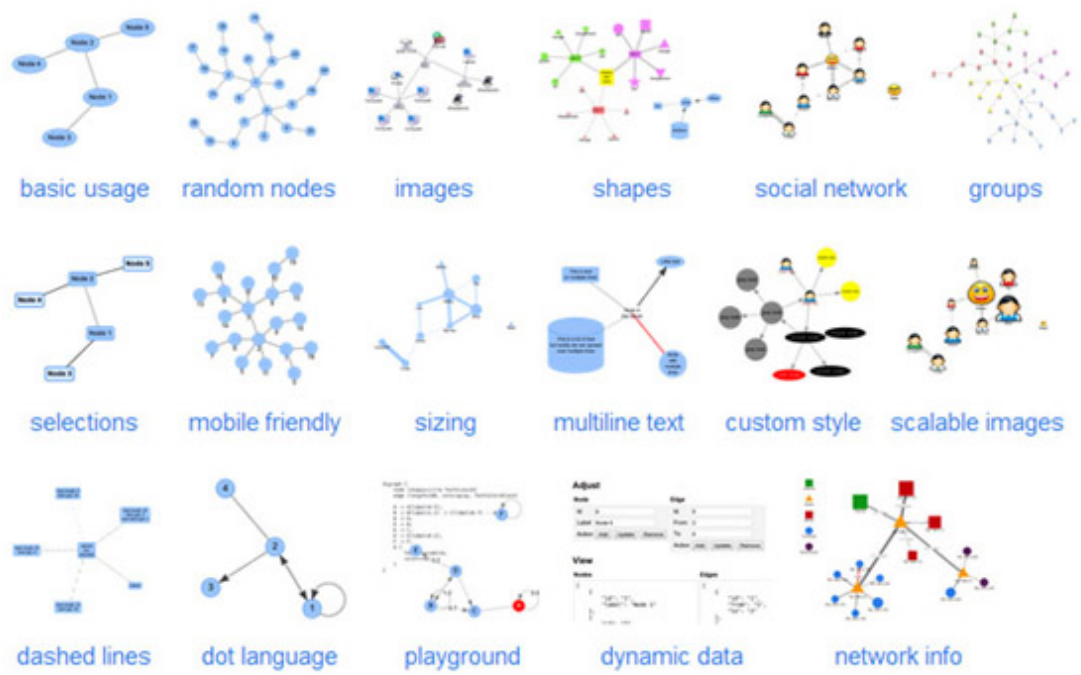


Figure 8:

Die Nutzung von Datenbanken

- Wenn große Datenmengen anfallen, kann die zusätzliche Nutzung von Datenbanken sinnvoll sein.
- In R bestehen Schnittstellen sowohl zu relationalen als auch nicht relationalen Datenbanken.
- Datenbanken sollten allerdings erst genutzt werden, wenn alle Möglichkeiten in R ausgeschöpft sind.

Nutzung der Unterlagen auf GitHub

- Die folgende Seite ist die Startseite für den Kurs:

<https://japhilko.github.io/Interfaces4R/>



Figure 9:

Wo sind die Sourcecodes?

Wie wird das Github Verzeichnis genutzt?

- Auf der folgenden Seite sind alle Sourcecodes enthalten:

<https://github.com/Japhilko/RInterfaces>

- Es lohnt sich immer wieder zu dieser Seite zurückzukehren, weil auch hier alle relevanten Dokumente verlinkt sind.
- Grundsätzlich kann man der Veranstaltung am Besten mit den kompletten File oder der kompletten Browserversion eines Kapitels (sind unter den Kapitelüberschriften verlinkt) folgen. Wenn Teile heruntergeladen werden sollen, bietet es sich an, das entsprechende pdf herunterzuladen.
- Falls Links ins Leere führen - bitte Bescheid sagen.

Informationen ausdrucken

- Zum Ausdrucken eignen sich die pdf-Dateien besser.
- Diese können mit dem Raw Button heruntergeladen werden.

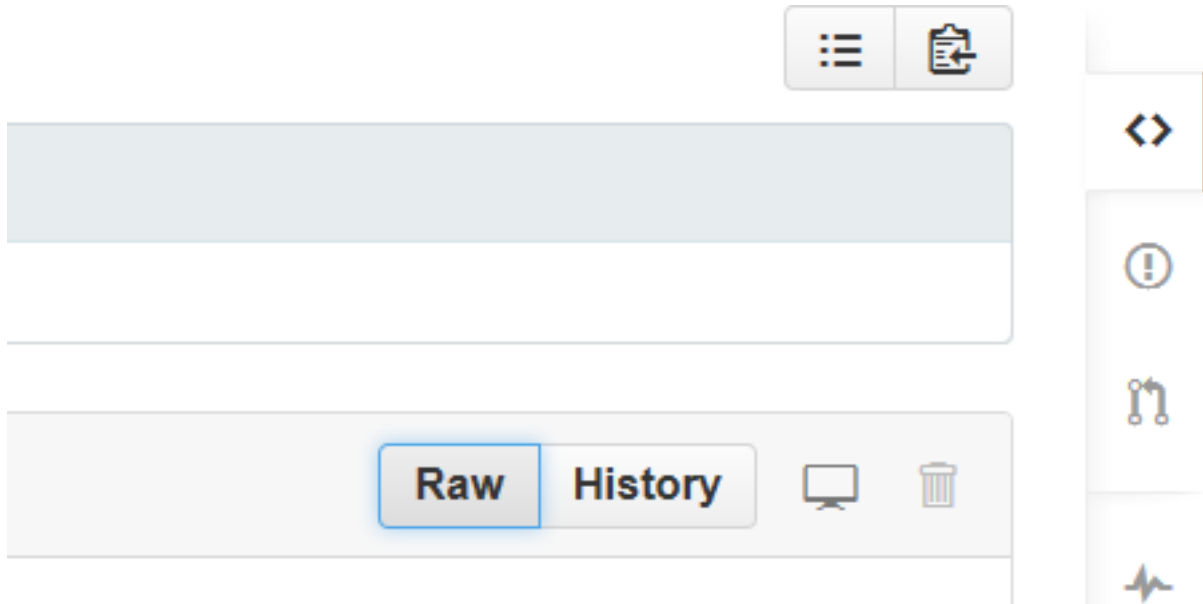


Figure 10: Raw Button zum Download

Weitere Dateien herunterladen

- Begleitend zu den Folien wird meistens auch ein R-File angeboten.
- Hier können Sie entweder das gesamte R-File herunterladen und in R ausführen oder einzelne Befehle per Copy/Paste übernehmen.
- Vereinzelt sind auch Datensätze vorhanden.
- .csv Dateien können direkt von R eingelesen werden (wie das geht werde ich noch zeigen).
- Wenn die .csv Dateien heruntergeladen werden sollen auch den Raw Button verwenden.
- Alle anderen Dateien (bspw. .RData) auch mittels Raw Button herunterladen.

Organisatorisches

- Zusätzlich gibt es in jedem Kapitel eine oder mehrere Aufgabe(n), da man nur durch eigenes Trainieren auf der Lernkurve vorankommt.
- Die Quellen für die Punkte auf den Folien sind als Link meist in der Überschrift hinterlegt.
- Die Links sind nur im HTML Dokument zu sehen aber auch in der pdf vorhanden.

Links und Quellen

Wen Github näher interessiert:

- Hello World

- Understanding the GitHub flow

Basis R . . .

- Wenn man nur R herunterlädt und installiert, sieht das so aus:

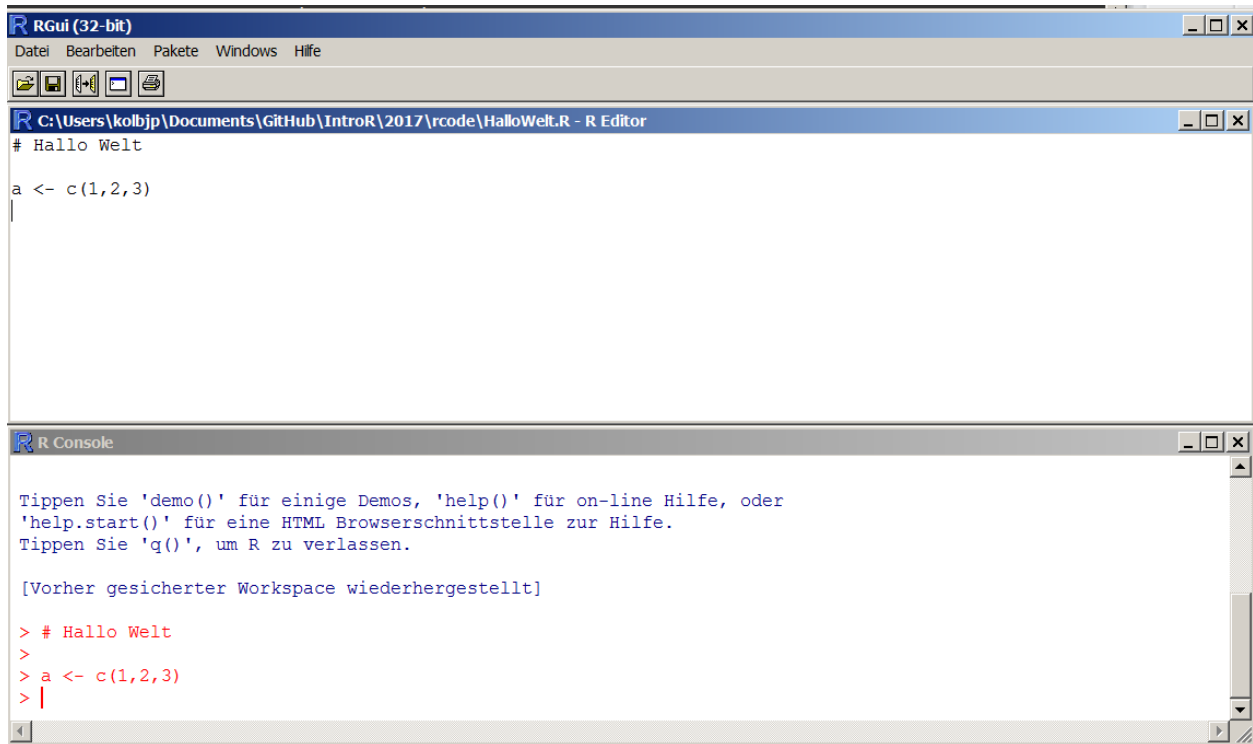


Figure 11:

- So habe ich bis 2012 mit R gearbeitet.

. . . und Rstudio

- Rstudio bietet Heute sehr viel Unterstützung:
- und macht einige Themen dieses Workshops erst möglich

Aufgabe - Zusatzpakete

Gehen Sie auf <https://cran.r-project.org/> und suchen Sie in dem Bereich, wo die Pakete vorgestellt werden, nach Paketen, . . .

- für Reproducible Research
- für interaktive Darstellungen
- für High-Performance Computing
- um mit großen Datenmengen umzugehen

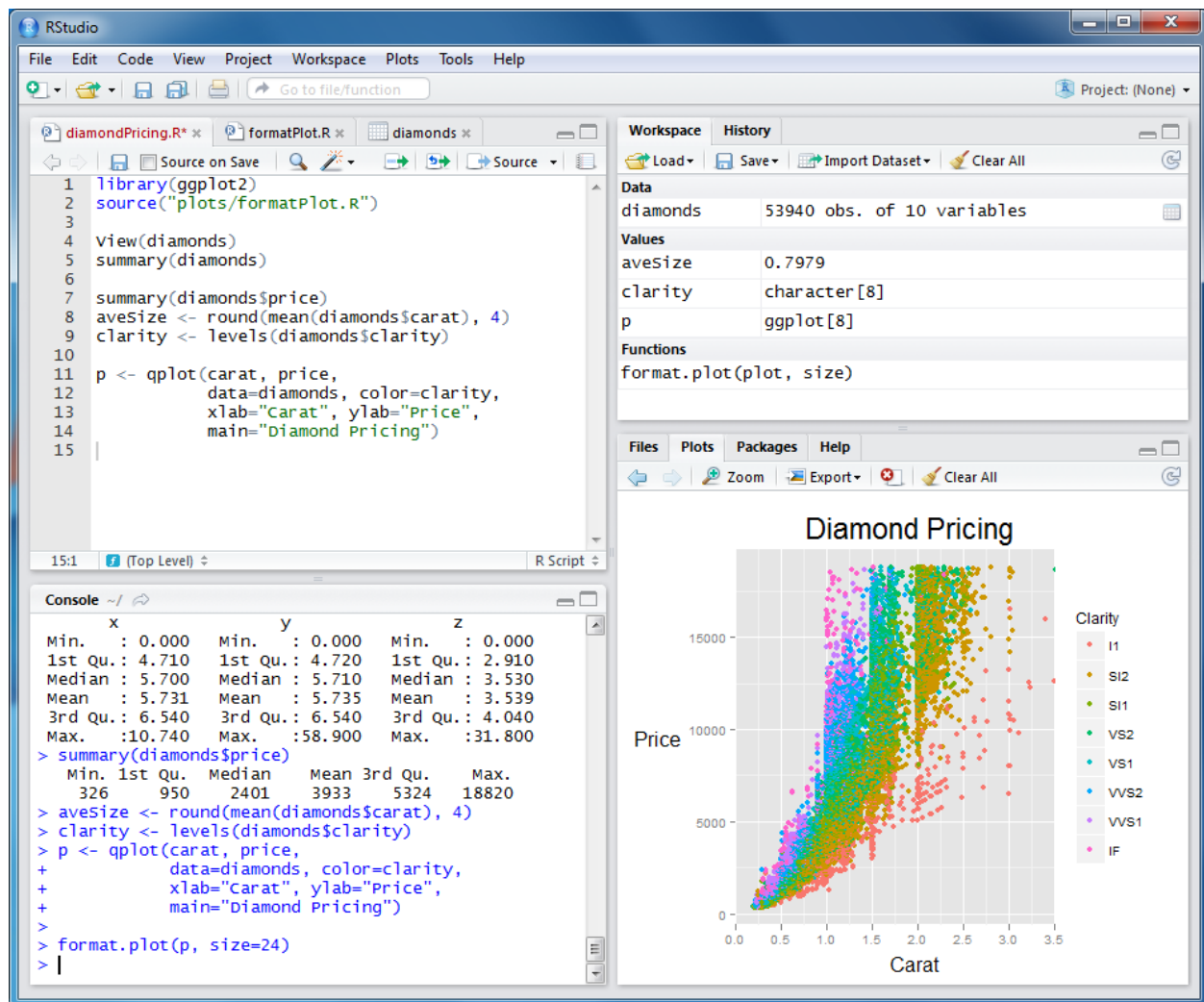


Figure 12:

Datenimport

Dateiformate in R

- Von R werden quelloffene, nicht-proprietäre Formate bevorzugt
- Es können aber auch Formate von anderen Statistik Software Paketen eingelesen werden
- R-user speichern Objekte gerne in sog. Workspaces ab
- Auch hier jedoch gilt: (fast) alles andere ist möglich

Formate - base package

R unterstützt von Haus aus schon einige wichtige Formate:

- CSV (Comma Separated Values): `read.csv()`
- FWF (Fixed With Format): `read.fwf()`
- Tab-getrennte Werte: `read.delim()`

Datenimport leicht gemacht mit Rstudio

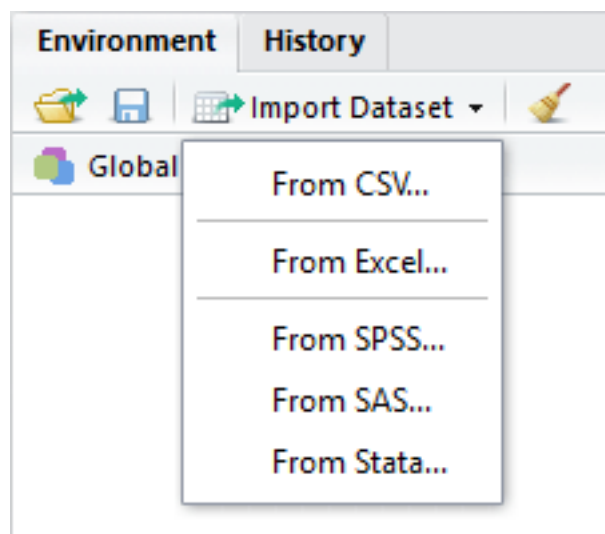


Figure 13: Import Button

CSV aus dem Web einladen

- Datensatz:

<https://data.montgomerycountymd.gov/api/views/6rqk-pdub/rows.csv?accessType=DOWNLOAD>

- Datenimport mit Rstudio

Der Arbeitsspeicher

So findet man heraus, in welchem Verzeichnis man sich gerade befindet

Import Text Data

File/Url:

Data Preview:

Full Name (character) *	Gender (character) *	Current Annual Salary (character) *	2015 Gross Pay Recessed (character) *	2015 Overtime Pay (character) *	Department (character) *	Department Name (character) *	Division (character) *	Assignment Category (character) *	Position Title (character) *	Underfilled Job Title (character)
Aarhus, Pam J.	F	\$68878.16	\$72336.79	N/A	POL	Department of Police	MSB Information Management and Technology Divisi...	Fulltime-Regular	Office Services Coordinator	N/A
Aaron, David J.	M	\$96908.09	\$101857.00	\$4640.99	POL	Department of Police	ISB Major Crimes Division Fugitive Section	Fulltime-Regular	Master Police Officer	N/A
Aaron, Marsha M.	F	\$104196.06	\$103019.73	N/A	HHS	Department of Health and Human Services	Adult Protective and Case Management Services	Fulltime-Regular	Social Worker IV	N/A
Abadio, Codfred A.	M	\$50697.79	\$54181.46	\$4445.15	COR	Correction and Rehabilitation	PRRS Facility and Security	Fulltime-Regular	Resident Supervisor II	N/A
Abadi, Essayas	M	\$92931.00	\$93468.35	N/A	HCA	Department of Housing and Community Affairs	Single Family Housing Program	Fulltime-Regular	Planning Specialist III	N/A
Abbamoto, Drew B.	M	\$67715.00	\$81392.40	\$10027.11	POL	Department of Police	PSB 6th District Special Assignment Team	Fulltime-Regular	Police Officer III	N/A
Abdelmoniem, Marwan M.	M	\$62286.30	\$59663.27	N/A	HHS	Department of Health and Human Services	Head Start	Fulltime-Regular	Administrative Specialist II	N/A
AbdulChani, Hasimah J.	F	\$45828.92	\$46783.23	\$6.38	POL	Department of Police	PSB Traffic Division Automated Traffic Enforcement S...	Fulltime-Regular	Police Aide	N/A
Abduljabar, Saeed	M	\$61040.57	\$66861.98	\$6569.81	DCS	Department of General Services	Facilities Maintenance	Fulltime-Regular	Electrician I	N/A
Abdur-Raheem, Mikael A.	M	\$56404.96	\$71943.08	\$15342.84	DOT	Department of Transportation	Transit Silver Spring Ride On	Fulltime-Regular	Bus Operator	N/A
Abeku, Hiruth	F	\$151585.60	\$164945.06	N/A	HHS	Department of Health and Human Services	STD and HIV Services	Parttime-Regular	Medical Doctor III - Physician	N/A
Abeku, Zekarias S.	M	\$44825.99	\$51693.47	\$5240.75	DOT	Department of Transportation	Transit Nicholson Ride On	Fulltime-Regular	Bus Operator	N/A
Abelina, Amiraza	M	\$39062.00	\$450.00	N/A	DOT	Department of Transportation	Transportation Management	Fulltime-Regular	Traffic Management Technician II	Traffic Management Techni...
Abelova, Sherry R.	F	\$93436.50	\$90833.10	N/A	HHS	Department of Health and Human Services	Adult Protective and Case Management Services	Fulltime-Regular	Social Worker III	N/A
Abera, Yoseph M.	M	\$117811.00	\$115786.22	N/A	DTIS	Department of Technology Services	EASD - ERP Applications Support	Fulltime-Regular	Senior Information Technology Specialist	N/A
Abi Jomaa, Rania F.	F	\$53009.99	\$46850.36	N/A	LIB	Department of Public Libraries	Olivey Library	Fulltime-Regular	Library Assistant I	N/A
Abijomas, Ryan Z.	M	\$17268.00	\$14665.53	N/A	LIB	Department of Public Libraries	Silver Spring Library	Parttime-Regular	Library Desk Assistant	N/A
Abiru, Lydia B.	F	\$40429.58	\$41746.34	\$5500.53	DOT	Department of Transportation	Transit Catonsville Ride On	Fulltime-Regular	Bus Operator	N/A
Abkarian, Maral	F	\$20925.51	\$9976.52	\$45.28	POL	Department of Police	PSB Traffic Division School Safety Section	Parttime-Regular	Crossing Guard	N/A
Abouraya, Nadia L.	F	\$16602.01	\$15392.92	N/A	HHS	Department of Health and Human Services	Community Support Network for People with Disabilities	Parttime-Regular	Office Clerk	N/A

Previewing first 50 entries.

Import Options:

Name: ☒ First Row as Names Delimiter: Escape:
Skip: ☒ Trim Spaces Quotes: Comment:
☒ Open Data Viewer Local: NA:

Code Preview:

```
library(readr)
rows <- read_csv("https://data.montgomerycountymd.gov/api/views/6rqk-pdub/rows.csv?accessType=DOWNLOAD")
view(rows)
```

Figure 14:

`getwd()`

So kann man das Arbeitsverzeichnis ändern:

Man erzeugt ein Objekt in dem man den Pfad abspeichert:

```
main.path <- "C:/\" # Beispiel für Windows
main.path <- "/users/Name/" # Beispiel für Mac
main.path <- "/home/user/" # Beispiel für Linux
```

Und ändert dann den Pfad mit `setwd()`

`setwd(main.path)`

Bei Windows ist es wichtig Slashes anstelle von Backslashes zu verwenden.

Alternative - Arbeitsspeicher

Das Paket `readr`

`install.packages("readr")`

`library(readr)`

- `readr` auf dem Rstudio Blogg

Import von Excel-Daten

- `library(readr)` ist für den Import von fremden Datenformaten hilfreich
- Wenn Excel-Daten vorliegen - als `.csv` abspeichern

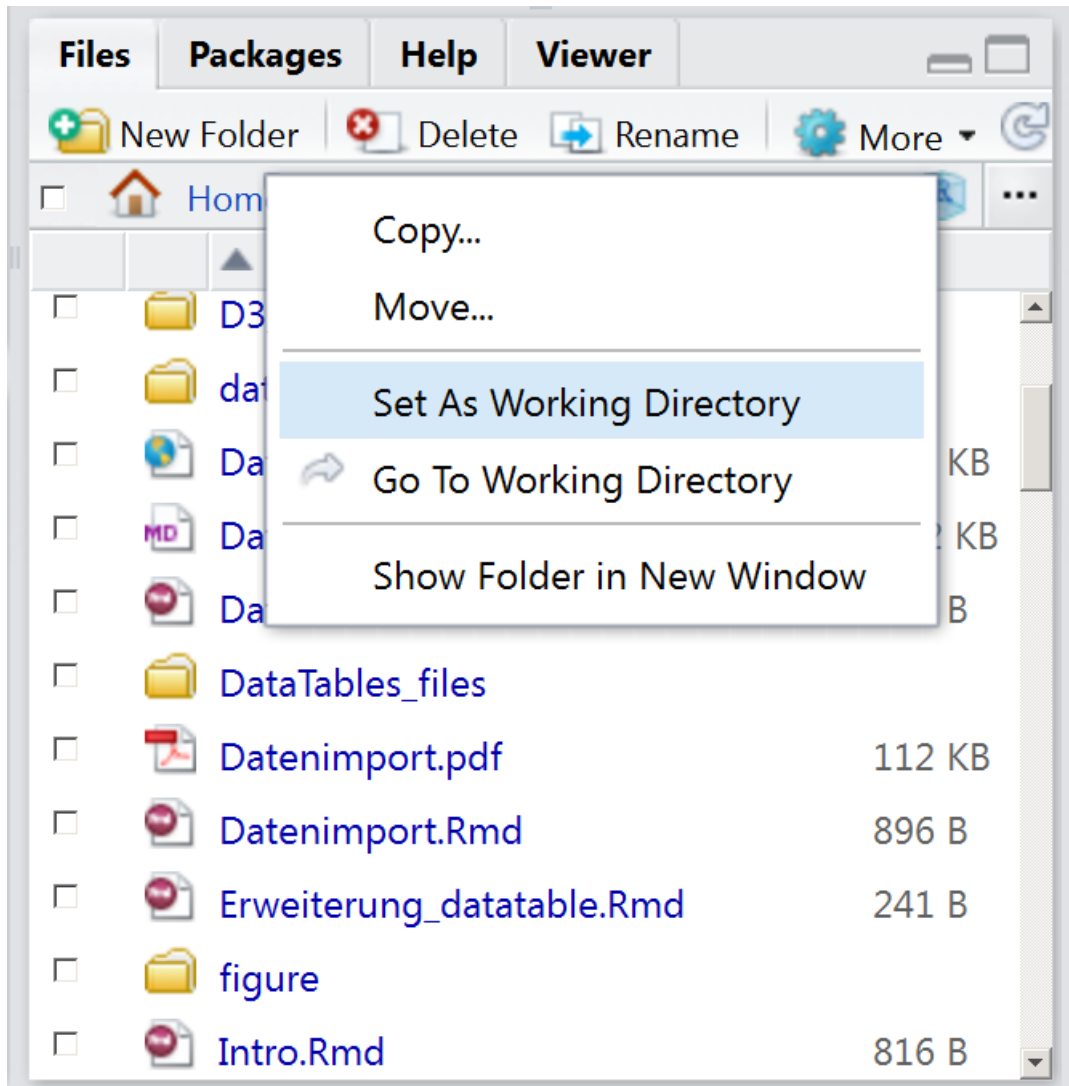


Figure 15:

readr 0.2.0

October 28, 2015 in [Packages](#), [tidyverse](#)

readr 0.2.0 is now available on CRAN. readr makes it easy to read many types of tabular data, including csv, tsv and fixed width. Compared to base equivalents like `read.csv()`, readr is much faster and gives more convenient output: it never converts strings to factors, can parse date/times, and it doesn't munge the column names.

This is a big release, so below I describe the new features divided into four main categories:

Figure 16:

```
library(readr)
rows <- read_csv("https://data.montgomerycountymd.gov/api/views/6rqk-pdub/rows.csv?accessType=DOWNLOAD")
```

.csv-Daten aus dem Web importieren - zweites Beispiel

```
url <- "https://raw.githubusercontent.com/Japhilko/GeoData/master/2015/data/whcSites.csv"
```

```
whcSites <- read_csv(url)
```

```
head(data.frame(whcSites$name_en, whcSites$category))
```

```
##                               whcSites.name_en
## 1 Cultural Landscape and Archaeological Remains of the Bamiyan Valley
## 2                               Minaret and Archaeological Remains of Jam
## 3                               Historic Centres of Berat and Gjirokastra
## 4                               Butrint
## 5                               Al Qal'a of Beni Hammad
## 6                               M'Zab Valley
## whcSites.category
## 1 Cultural
## 2 Cultural
## 3 Cultural
## 4 Cultural
## 5 Cultural
## 6 Cultural
```

Das Paket haven

```
install.packages("haven")
```

```
library(haven)
```

- Das R-Paket `haven` auf dem Rstudio Blogg

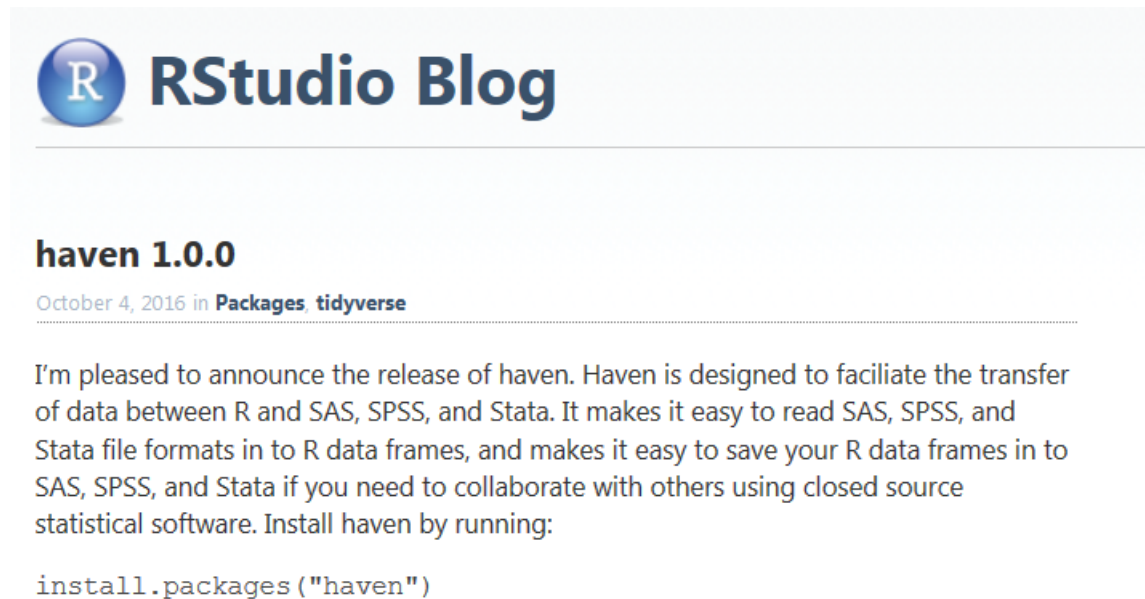


Figure 17:

SPSS Dateien einlesen

- Zunächst muss wieder der Pfad zum Arbeitsverzeichnis angegeben werden.
- SPSS-Dateien können auch direkt aus dem Internet geladen werden:

```
install.packages("haven")
```

```
library(haven)
```

```
mtcars <- read_sav("https://github.com/Japhilko/RInterfaces/raw/master/data/mtcars.sav")
```

stata Dateien einlesen

```
library(haven)
```

```
oecd <- read_dta("https://github.com/Japhilko/IntroR/raw/master/2017/data/oecd.dta")
```

Links

- Quick-R - Import Data
- Datenimport bei R-bloggers
- Importing Data into R

- Mapping von Arbeitslosendaten in den USA
- Das Paket readr

Aufgabe - Datenimport

- Gehen Sie auf meine Github Seite und laden Sie den Datensatz zu den Weltkulturerbestätten (whcsites) herunter
- Laden Sie den Datensatz mit einer geeigneten Funktion in Ihre Console.
- Finden Sie heraus, wieviele Beobachtungen und Variablen der Datensatz umfasst.
- Wieviele kulturelle/natürliche Weltkulturerbestätten gibt es im Datensatz?

Datenexport

Die Exportformate von R

- In R werden offene Dateiformate bevorzugt
- Genauso wie `read.X()` Funktionen stehen viele `write.X()` Funktionen zur Verfügung
- Das eigene Format von R sind sog. Workspaces (`.RData`)

Beispieldatensatz erzeugen

```
A <- c(1,2,3,4)
B <- c("A","B","C","D")

mydata <- data.frame(A,B)
```

Überblick Daten Import/Export

- wenn mit R weitergearbeitet wird, eignet sich das `.RData` Format am Besten:

```
save(mydata, file="mydata.RData")
```

Daten in .csv Format abspeichern

```
write.csv(mydata,file="mydata.csv")
```

- Wenn mit Deutschem Excel weitergearbeitet werden soll, eignet sich `write.csv2` besser

```
write.csv2(mydata,file="mydata.csv")
```


- Sonst sieht das Ergebnis so aus:


Das Paket xlsx


```
library(xlsx)
write.xlsx(mydata,file="mydata.xlsx")
```


	A	
1	,"A","B"	
2	1,1,"A"	
3	2,2,"B"	
4	3,3,"C"	
5	4,4,"D"	
6		

Figure 18:

 **R xlsx package : A quick start guide to manipulate Excel files in R**

 AdChoices
 [Microsoft Excel](#)
[Download for Java](#)
[Excel Tutorial](#)



- Install and load xlsx package
- Read an Excel file
- Write data to an Excel file

Figure 19:

Das Paket foreign

Reading/Writing Stata (.dta) files with Foreign

December 4, 2012

By is.R()

Figure 20:

- Funktionen im Paket foreign

Daten in stata Format abspeichern

```
library(foreign)
write.dta(mydata,file="data/mydata.dta")
```

Das Paket rio

```
install.packages("rio")
```

Daten als .sav abspeichern (SPSS)

R topics documented:

lookup.xport	2
read.arff	3
read.dbf	4
read.dta	5
read.epiinfo	7
read.mtp	8
read.octave	9
read.spss	10
read.ssd	12
read.systat	14
read.xport	15
S3 read functions	16
write.arff	17
write.dbf	18
write.dta	19
write.foreign	21

Figure 21:

Import, Export, and Convert Data Files

The idea behind `rio` is to simplify the process of importing data into R and exporting data from R. This process is, probably unnecessarily, extremely complex for beginning R users. Indeed, R supplies [an entire manual](#) describing the process of data import/export. And, despite all of that text, most of the packages described are (to varying degrees) out-of-date. Faster, simpler, packages with fewer dependencies have been created for many of the file types described in that document. `rio` aims to unify data I/O (importing and exporting) into two simple functions: `import()` and `export()` so that beginners (and experienced R users) never have to think twice (or even once) about the best way to read and write R data.

Figure 22:

```
library("rio")  
# create file to convert  
  
export(mtcars, "data/mtcars.sav")
```

Dateiformate konvertieren

```
export(mtcars, "data/mtcars.dta")  
  
# convert Stata to SPSS  
convert("data/mtcars.dta", "data/mtcars.sav")
```

Links Export

- Quick R für das Exportieren von Daten:
- Hilfe zum Export auf dem cran Server
- Daten aus R heraus bekommen

R und Excel

Das Paket `xlsx`

- Eine wichtige Datenquelle - Eurostat

```
library("xlsx")  
dat <- read.xlsx("cult_emp_sex.xls",1)
```

Einige Schritte um R und Excel zu verbinden

- Die Excel-Verbindung

```
install.packages("XLConnect")
```

```
library("XLConnect")
```



Figure 23: Vignette für XLconnect

Eine Excel Datei aus R erzeugen

```
fileXls <- "data/newFile.xlsx"
unlink(fileXls, recursive = FALSE, force = FALSE)
exc <- loadWorkbook(fileXls, create = TRUE)
createSheet(exc, 'Input')
saveWorkbook(exc)
```

Das Arbeitsblatt mit Daten befüllen

```
input <- data.frame('inputType'=c('Day','Month'),'inputValue'=c(2,5))
writeWorksheet(exc, input, sheet = "input", startRow = 1, startCol = 2)
saveWorkbook(exc)
```

BERT - Eine weitere Verbindung zwischen R und Excel

- Schnellstart mit Excel

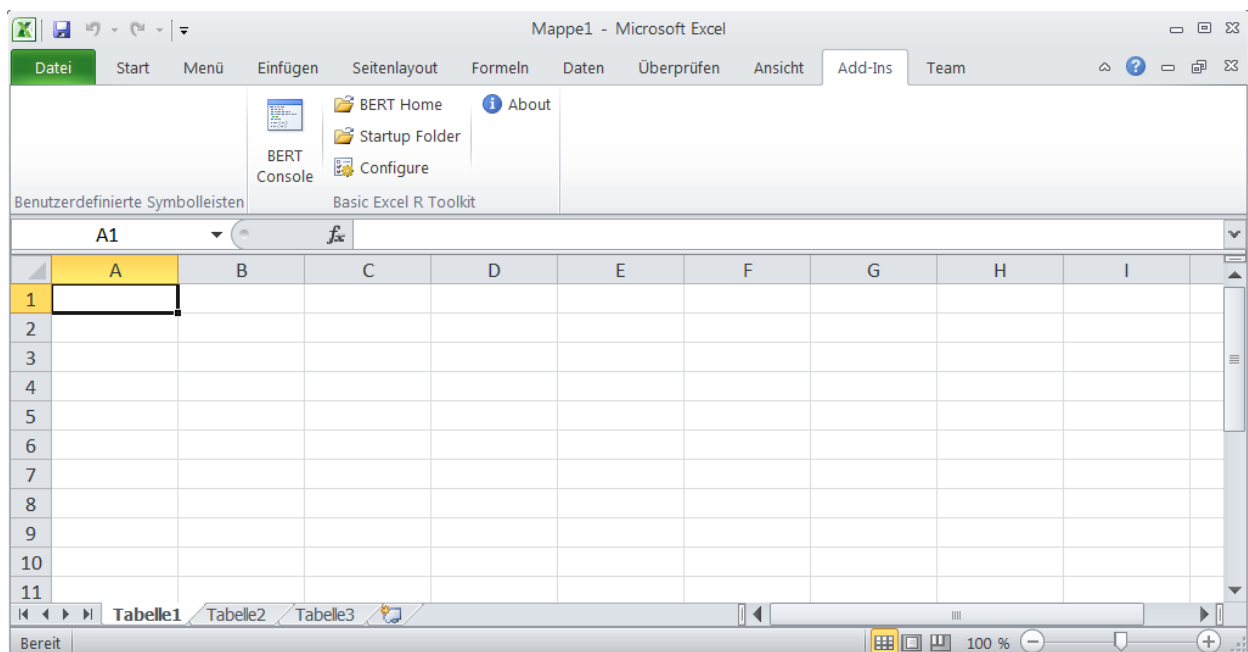


Figure 24:

```
myFunction <- function(){
  aa <- rnorm(200)
  bb <- rnorm(200)
  res <- lm(aa~bb)$res
  return(res)
}
```

Link BERT

- BERT: a newcomer in the R Excel connection

BERT: a newcomer in the R Excel connection

November 30, 2016

By The R Trader



(This article was first published on [R – The R Trader](#), and kindly contributed to [R-bloggers](#))

163
SHARES

f Share

🐦 Tweet

A few months ago a reader point me out this new way of connecting R and Excel. I don't know for how long this has been around, but I never came across it and I've never seen any blog post or article about it. So I decided to write a post as the tool is really worth it and before anyone asks, I'm not related to the company in any way.

Figure 25:

Das Paket readxl

- readxl

```
install.packages("readxl")
```

```
library(readxl)
```

Aufgabe Export nach Excel

- Schränken Sie den Weltkulturerbe Datensatz auf die wichtigsten Spalten ein.
- Erzeugen Sie einen Subdatensatz in dem nur die kulturellen Stätten enthalten sind. Machen Sie dies analog für die natürlichen Stätten.
- Nutzen Sie das Paket `XLconnect` um die Datensätze nach Excel zu übertragen. Erstellen Sie ein Blatt für die kulturellen und eins für die natürlichen Stätten.

Get data out of excel and into R with readxl


April 15, 2015

By hadleywickham

 Like 0  Share  Share 21

(This article was first published on [RStudio Blog](#), and kindly contributed to [R-bloggers](#))

907
SHARES

 Share

 Tweet

I'm pleased to announced that the first version of readxl is now available on CRAN. Readxl makes it easy to get tabular data out of excel. It:

- Supports both the legacy `.xls` format and the modern xml-based `.xlsx` format. `.xls` support is made possible the with [libxls](#) C library, which abstracts away many of the complexities of the underlying binary format. To parse `.xlsx`, we use the insanely fast [RapidXML](#) C++ library.

Figure 26: