EALTA, Dublin 2019

IRT Analyses Using the Statistical Software R

Steffen Brandt

OPENCAMPUS.sh

Goals

(I) Calculating a score for the PISA 2015 reading assessment, trying to get as close as possible to the original ones.

(II) Using the same approach to calculate scores for your personal data sets.

Structure

(1) Getting Familiar with R and RStudio

(2) Data Preparation and Exploration

(3) Estimation of a Unidimensional IRT Model

(4) Revision of Item and Test Characteristics

(5) Estimation of Individual Ability Scores

(6) Getting More Reliable Scores Using an IRT Model with Regression

(7) Estimation of a Multidimensional Model

What is R?

- An “interpreter” language
- Developed by statisticians
  (Python, e.g., was developed by computer scientists)
- Together with Python the most popular data science language

Why should I learn R?

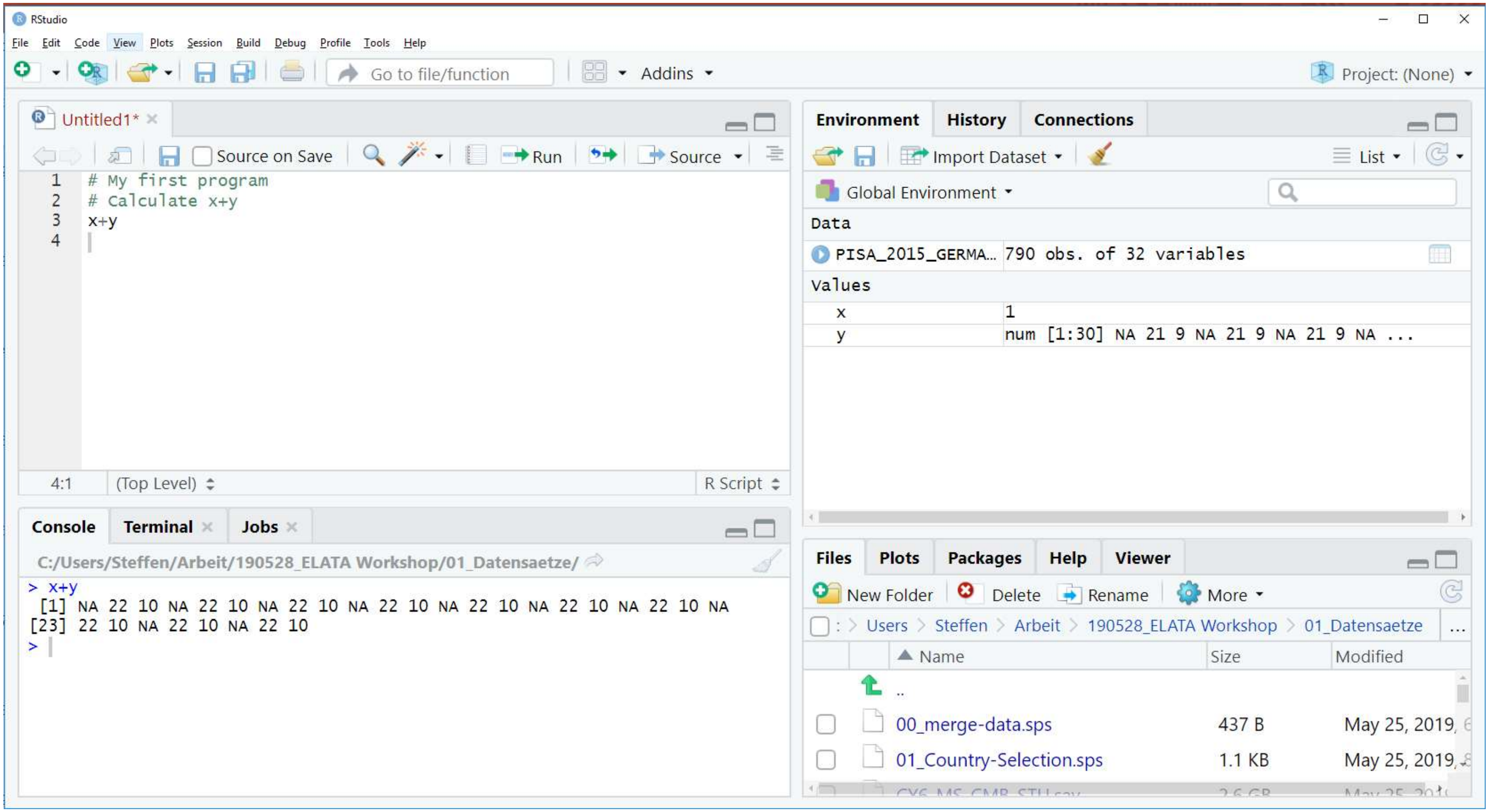
- It’s Open Source!
- Today, data preparation and analysis skills are critical success factors in all mid to large size companies
- In academia R is getting more and more the standard language for statistical analyses
- It has a large community that probably already had every problem you will have.

Why do I need RStudio?

It is an integrated developer environment (IDE) for R that provides you:

- quick access to help files, data files, and saved program code,
- information about data objects currently in the working environment (data tables, variables, functions, etc.),
- easy access to graphical outputs,
- installation of additional function libraries (packages),
- integration of version control,
- and much more…

Using RStudio



Where do I find help?

- Google!
  - Stackoverflow
  - Cheatsheets
  - R function documentation

What is an R program?

A sequence of definitions

- `x <- 1`
- `y <- c(2, 15, 39, 24)`

Change the workspace environment

or expressions

- `x + y`
- `3^2/2`
- `mean(y)`
- `x = 1`

Generate an output

All names (for variables, functions, …) are case sensitive!

Tasks

- ❑ Calculate the standard deviation of the vector including the numbers from 1 to 10.

```
sd(c(1,2,3,4,5,6,7,8,9,10))
```

Functions and Packages

- Are the engines of any R program
- Function arguments (input data) are provided in parentheses, for example:
  `mean(x=c(1,3,4,NA), na.rm=TRUE)`
- Basic functions are included in the “base” package, the function library coming with the standard installation
- Additional functions can be used after installing a corresponding package from the Internet

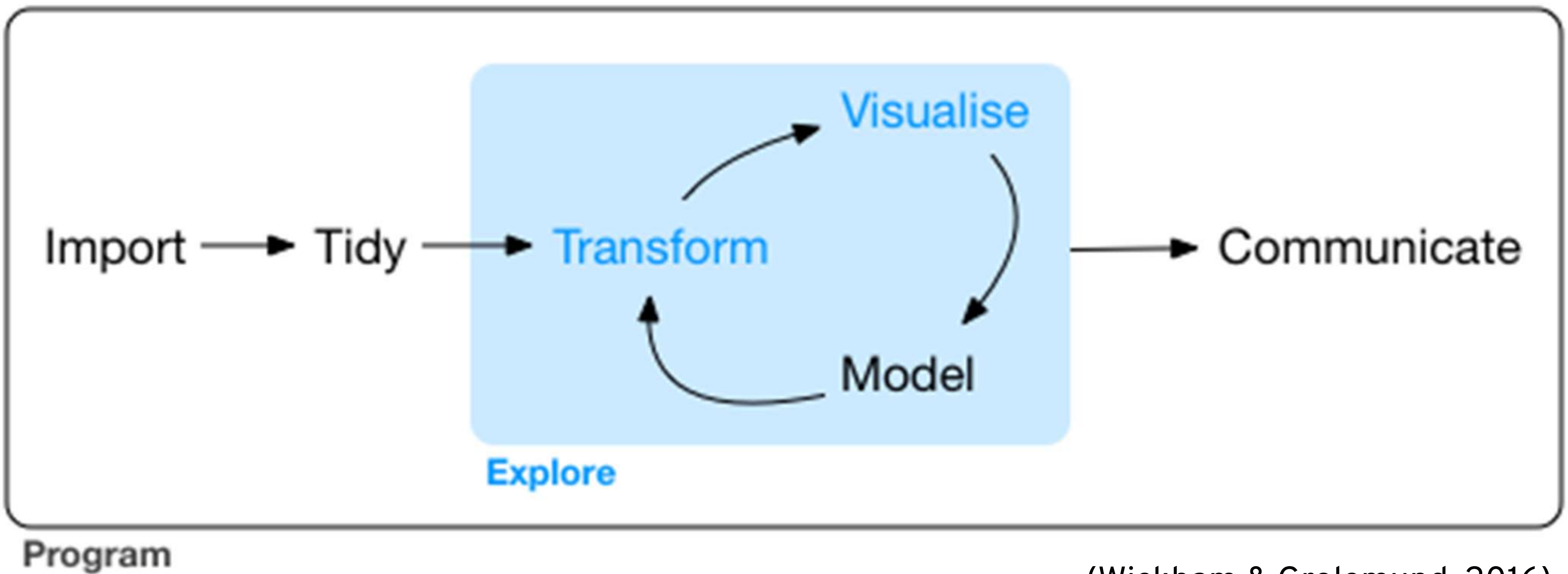
Tasks

- ❑ Install the following packages:
  haven, dplyr, ggplot, tam, psych

- ❑ Load the functions of each of these packages into the environment using the function `library()`.

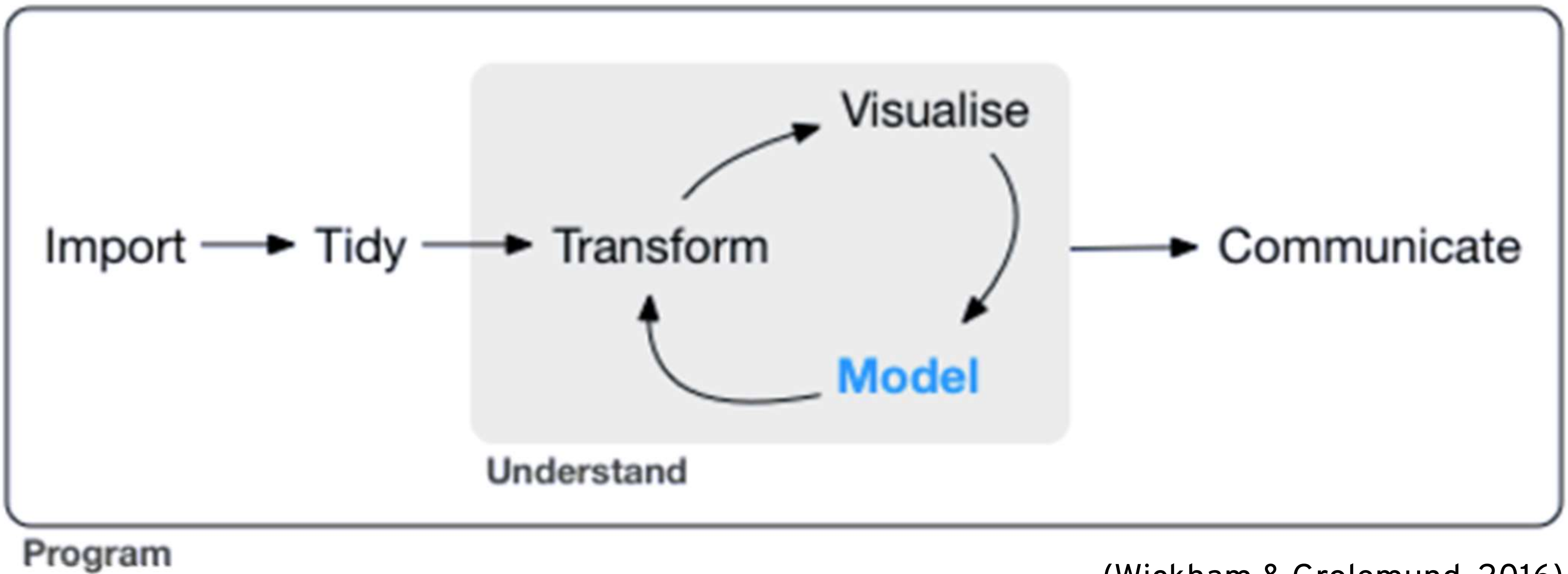
- ❑ Try to get an idea of the type of functions that each of these packages provides.

Data Preparation



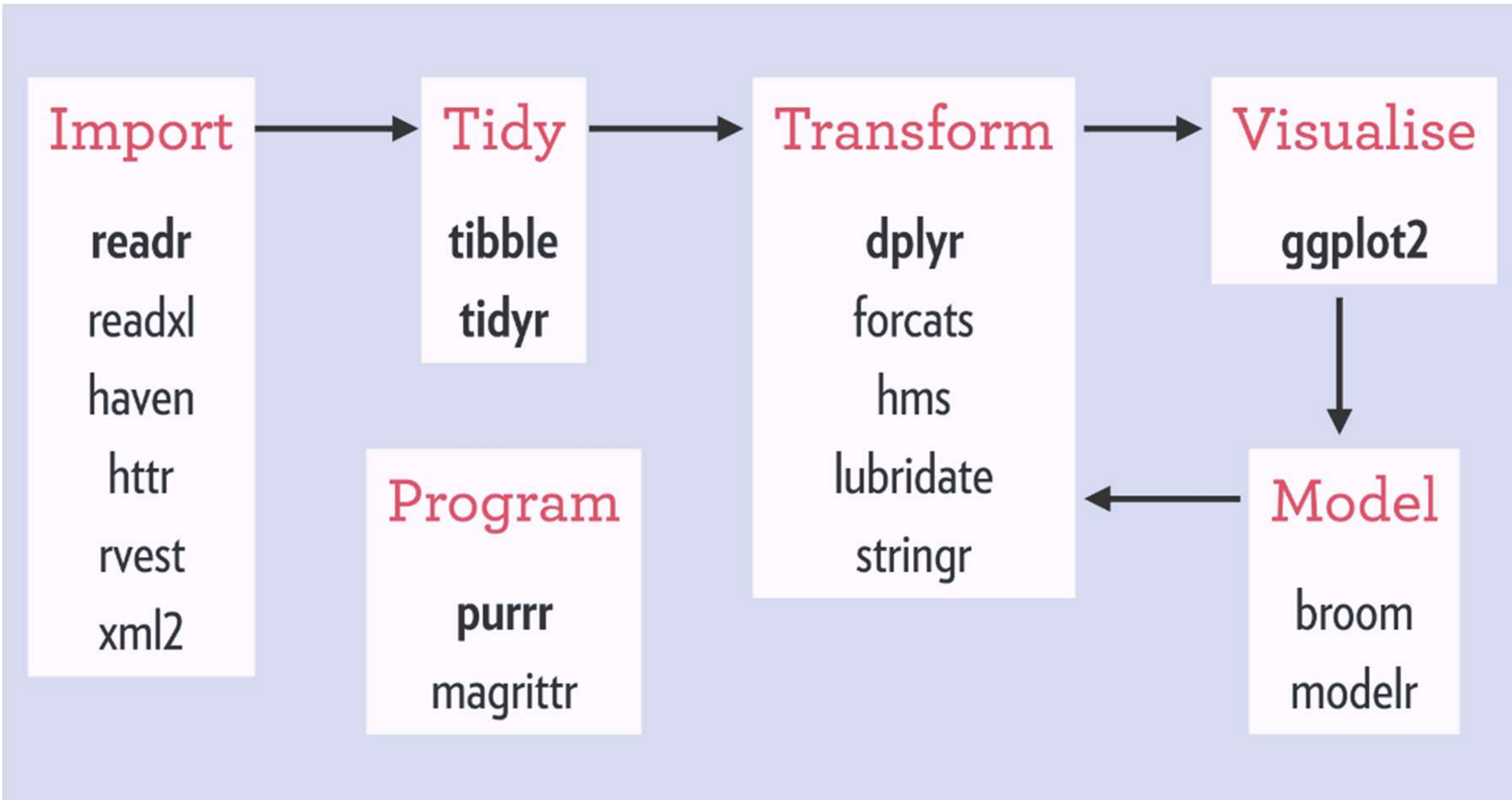
(Wickham & Grolemund, 2016)

Data Modelling



(Wickham & Grolemund, 2016)

Tidyverse



(cf. https://rviews.rstudio.com/2017/06/08/what-is-the-tidyverse/)

R Objects

- An R object can be:
  a data table, variable, vector, function, list, graphical output, and many more

- All R objects can be exported and imported to the environment using, the following functions:
  `save(`*`object_name`*`, file="filename.Rda")`
  `load("filename.Rda")`

Tasks

- ❑ Create a working directory for this workshop on your computer.
- ❑ Download the R object starting with the name “PISA_2015” provided here on GitHub: https://github.com/steffen74/EALTA-2019-Workshop
- ❑ Import the R object into your environment.
- ❑ Try to get an idea of the content of the R object.

Data Types

There are 3 basic types of variables:

- ❑ `boolean` (`TRUE, FALSE`)
- ❑ `numeric` (`1.1392`)
- ❑ `character` (`"text"`)

And various additional more specific types:

- ❑ `integer` (`12;` subtype of `numeric`)
- ❑ `date` (`"2019-04-11";` subtype of `numeric`)
- ❑ `factor` (`"lower secondary", "upper secondary", …;` subtype of `integer`)

For each data type missing values are defined as NA.

Vectors

- Via the function `c()` you can concatenate single data values.
- All elements in a vector must be of the same type.
  → A vector has a unique type.
- Examples:

```
signed_up <- c(FALSE, TRUE, FALSE)
city <- c("kiel", "hamburg", "berlin")
distance <- c(112, 343, 235)
sampleSize <- 12
```

Data Tables

Data tables consist of vectors with equal length.

There are different definitions of data tables:

- `dataframe` (base package)
  Standard data table, which is still most commonly used.
- `tibble` (tidyverse package)
  Optimized version of data frame with focus on data preparation and cleaning.
- `data.table` (data.table Package)
  Optimized for very large data tables.

Example Data

- Packages often include example data.

- `ggplot2` includes a tibble named `mpg`.
  (Fuel economy data from 1999 and 2008 for 38 popular models of car)

- The dataset is invisibly imported into the environment when calling `library(ggplot2)`.

Working with Data Tables

Selecting a vector/variable/column from a data table

- ❑ `PISA$HISCED` (vector of the data table column with the name class)
- ❑ `PISA[[1]]` (vector of the first column in the data table)
- ❑ ~~`PISA[,1]` (*data table* consisting of only the first column)~~

Adding or changing a vector in a data table

- ❑ `PISA$AVRG_ISCED <- .5*PISA$MISCED + .5*PISA$FISCED`
- ❑ `PISA$CR055Q01S <- PISA$CR055Q01S*2`

Working with Data Tables and dplyr

Beispiel:

```
PISA %>%
  select (EAPREAD, MISCED, FISCED) %>%
  filter (EAPREAD<500) %>%
  mutate (AVRG_ISCED = .5*MISCED + .5*FISCED)
```

Pipe Operator: `%>%`

- ❑ Concatenates the application of several functions into a single expression.
- ❑ The function after the operator uses the result of the function executed before as its first function argument.
- ❑ Allows in a more readable and very often more effective data preparation.

Important dplyr Functions

Select variables from a data table: `select()`

- Reduces the data table to the variables with the given provided.

Filter cases from a data table: `filter()`

- Reduces the data table to cases for which the given logical condition ís true.

Add variables: `mutate()`

- Adds a new variable with name provide before the equal sign and with the value provided via the expression after the equal sign.

Equal and Assignment Operators

Definition of objects: `a <- x`

Setting function arguments: `mean(x, na.rm = TRUE)`

Defining new variable names when they are part of the argument of a function:

```
tibble(var1=c(1,2,3), var2=c(1,2,3))
```

Comparison of objects: `a == x`

Do not use the equal sign (=) for the definition of objects!

Logical Operators

- Equal: ==
- Or: |
- And: &
- Greater or equal as: >=
- Smaller or equal as: <=

The result of a logical operation is always a logical vector (or value).

Tasks

- Select the variable HISCED from the data table and look at the output
- Add a new variable HISCED_father to the data table which indicates for each student whether the ISCED level of the father is higher than the ISCED level of the mother.
- Use the function `table()` to check on the number of respective values in HISCED_father
- Use the function describe() to get an impression of the values in the data table.

Visualizing Data

- The package ggplot currently provides the most powerful functions for graphical outputs.

- Example of a histogram:
```
ggplot(PISA) +
   geom_histogram(aes(x=EAPREAD))
```

- Great source for examples including the program code: https://www.r-graph-gallery.com/

Online Learning Ressources

- Introduction to R Online Course | DataCamp
  https://www.datacamp.com/courses/free-introduction-to-r
- Data Science: R Basics | Edx
  https://www.edx.org/course/r-basics2
- A Concepual Introduction to Item Response Theory | YouTube
  https://www.youtube.com/playlist?list=PLJNUIJnElUzDmrIPunMyF3tTvIHb65wNb

Why bother with IRT?

In comparison to CTT it

- ❑ assumes a more realistic response pattern including probability,
- ❑ provides a theory about the relationship of item difficulty and measures trait level,
- ❑ provides trait level dependent reliability measure,
- ❑ can “natively” deal with incomplete answers, and
- ❑ provides various straightforward methods to check for item and test quality.

IRT Software

- Facets
- PARSCALE
- ConQuest
- mdltm
- R package mirt
- R package TAM

Why the R Package TAM?

Why R?

- ❑ Open source!
- ❑ All data preparation and analyses in a single software
- ❑ Large community

Why TAM?

- ❑ Very well maintained
- ❑ Very flexible model definition
- ❑ Large set of support functions
- ❑ (If you used ConQuest before: proximity of the result output)

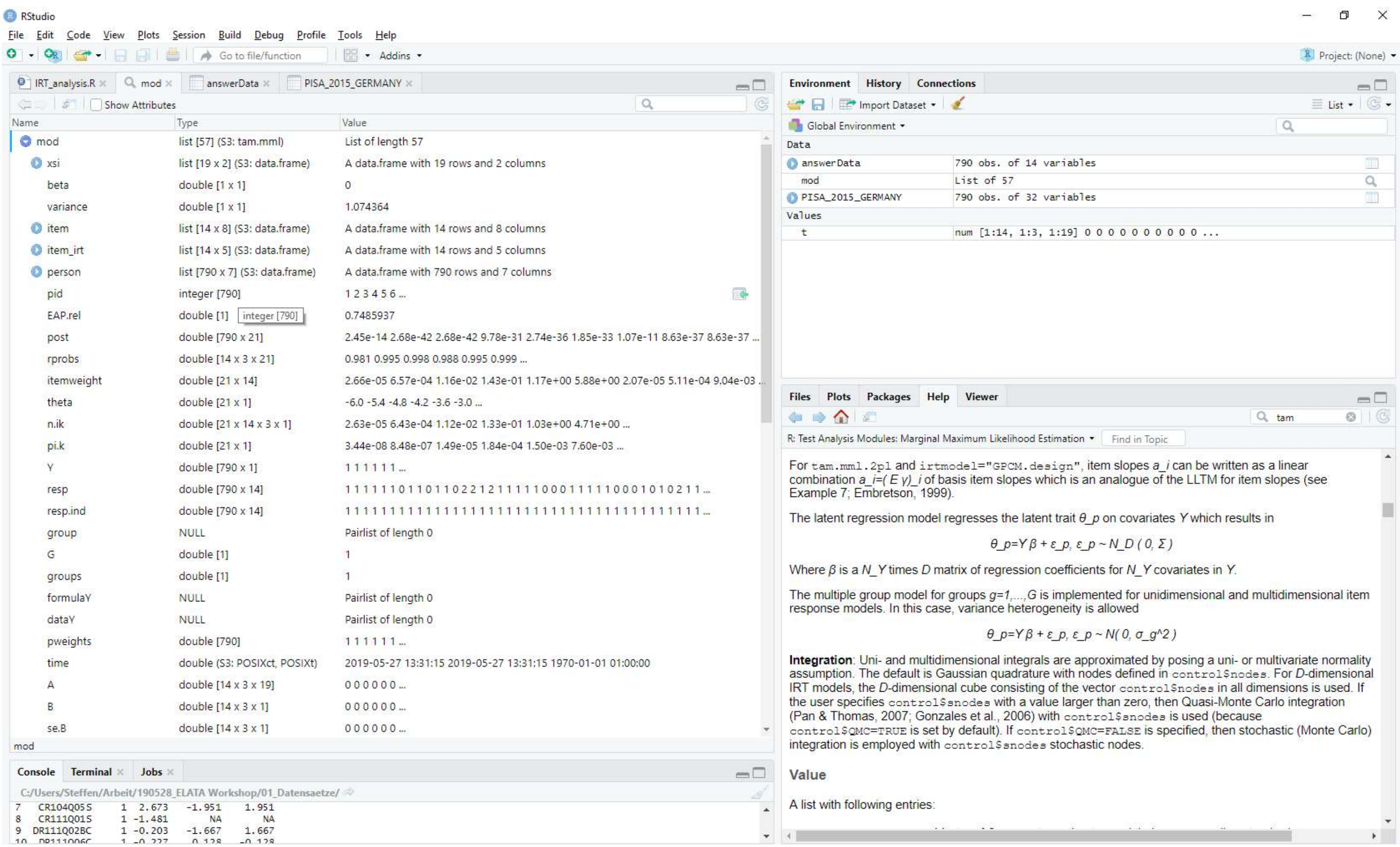
Estimation of a Unidimensional IRT Model

```
# Prepare a data set ‘resp’ including only the
# variables with the item response data

# import functions
library(tam)

# estimate model
mod <- tam(resp)
```

Tasks

- ❑ Import all function libraries you need
- ❑ Import the PISA data set
- ❑ Prepare a dataset ‘resp’ including only the variables with item answers
- ❑ Estimate a unidimensional IRT (Rasch) model

The Result Object



What is a List?

- A `list` is an R object that includes a list of other arbitrary R objects

- A data table is a special `list` object where all elements are vectors and of the same length.

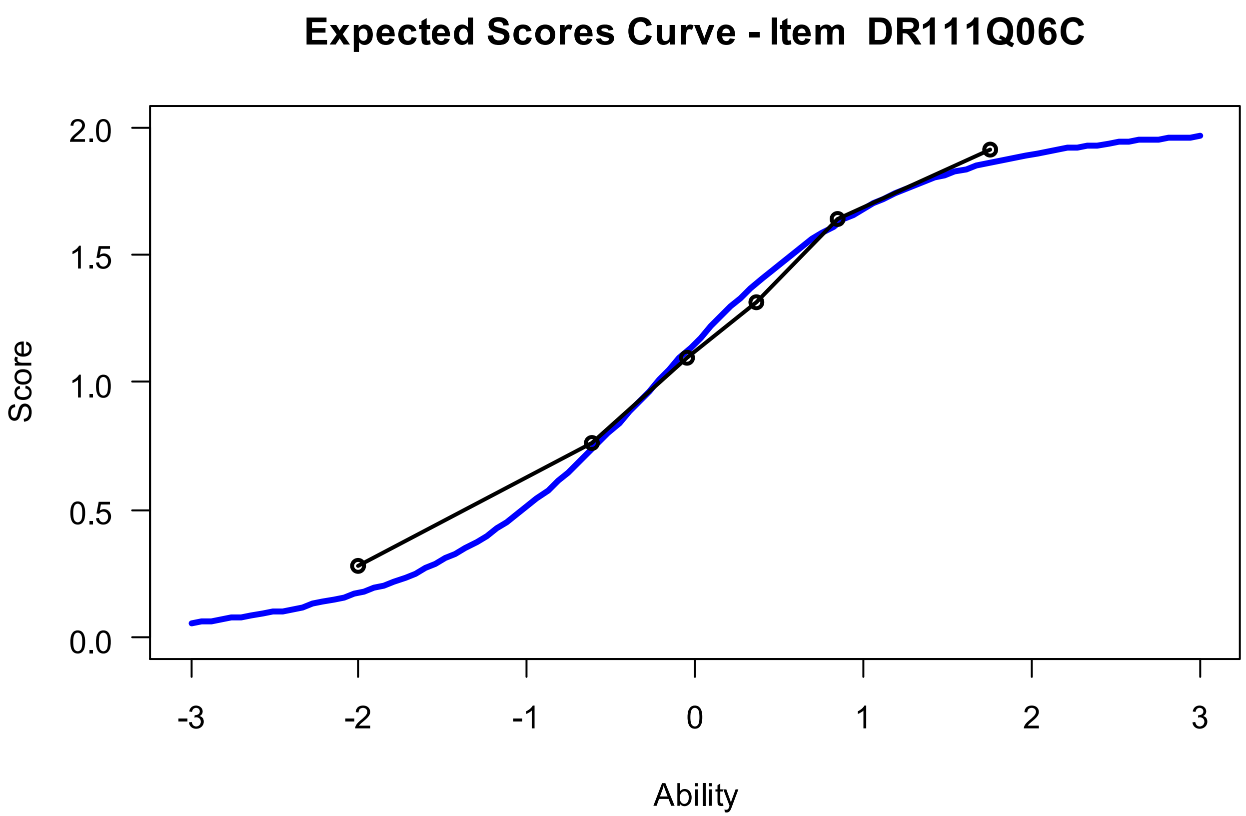
- Selecting elements from a `list` corresponds to selecting elements from a data table:

```
mod$person
mod[[1]]
```

Revision of Item and Test Characteristics

- ❑ Visual check of the item characteristic curves (ICCs)
- ❑ Checking item fit statistics
- ❑ Checking classical test theory measures (point-biserial correlation)
- ❑ Checking on differential item function (DIF)
- ❑ Checking test reliability

Item Characteristic Curve (ICC)



Fit Statistics

$itemfit

	parameter	Outfit	Outfit_t	Outfit_p	Outfit_pholm	Infit	Infit_t	Infit_p	Infit_pholm
1	CR055Q01S_Cat1	0.8977187	-1.52597353	1.270165e-01	7.620987e-01	0.9617561	-0.5351700	0.592532289	1.00000000
2	DR055Q02C_Cat1	0.9722721	-0.73155366	4.644410e-01	1.000000e+00	0.9694420	-0.8020724	0.422511072	1.00000000
3	DR055Q03C_Cat1	1.1489418	3.11587056	1.834028e-03	2.384236e-02	1.0616217	1.3489909	0.177339887	1.00000000
4	DR055Q03C_Cat2	0.9980383	-0.07681248	9.387727e-01	1.000000e+00	1.0231923	0.6282293	0.529853726	1.00000000
5	DR055Q05C_Cat1	0.8126900	-3.38687686	7.069311e-04	1.060397e-02	0.8891780	-1.9343314	0.053072394	0.74301352
6	CR104Q01S_Cat1	0.8937484	-2.91432117	3.564630e-03	4.073553e-02	0.9129786	-2.3626741	0.018143616	0.27215424
7	CR104Q02S_Cat1	1.2679537	5.60698582	2.058805e-08	3.705849e-07	1.1150706	2.5346648	0.011255498	0.19134347
8	CR104Q05S_Cat1	0.9142053	-2.63050752	8.525748e-03	8.525748e-02	0.9691550	-0.9240825	0.355443390	1.00000000
9	CR104Q05S_Cat2	0.6932110	-0.85857955	3.905725e-01	1.000000e+00	1.0122617	0.1574981	0.874852312	1.00000000
10	CR111Q01S_Cat1	0.9033549	-1.92525904	5.419696e-02	4.877726e-01	0.9193399	-1.5915300	0.111490345	1.00000000
11	DR111Q02BC_Cat1	0.8630351	-1.87726338	6.048200e-02	4.877726e-01	0.9256769	-0.9679001	0.333094270	1.00000000
12	DR111Q02BC_Cat2	1.1017983	1.91669203	5.527707e-02	4.877726e-01	1.0790376	1.5165994	0.129367868	1.00000000
13	DR111Q06C_Cat1	1.4677950	8.97853548	2.743947e-19	5.213498e-18	1.1392394	2.9846518	0.002839013	0.05394125
14	DR111Q06C_Cat2	0.9636503	-1.05139728	2.930762e-01	1.000000e+00	1.0037096	0.1027013	0.918200081	1.00000000
15	CR227Q01S_Cat1	1.2383347	5.53996458	3.025328e-08	5.143058e-07	1.1226074	2.9574174	0.003102279	0.05584102
16	CR227Q02S_Cat1	1.0266319	0.31210773	7.549587e-01	1.000000e+00	0.9676074	-0.4632504	0.643184929	1.00000000
17	CR227Q02S_Cat2	1.1394274	4.12508322	3.706007e-05	5.929612e-04	1.0835824	2.5318382	0.011346633	0.19134347
18	DR227Q03C_Cat1	0.8617137	-2.92954128	3.394627e-03	4.073553e-02	0.9116801	-1.8250608	0.067991848	0.88389403
19	DR227Q06C_Cat1	0.7784490	-3.23992894	1.195595e-03	1.673833e-02	0.9188620	-1.1057137	0.268850465	1.00000000

Classical Test Theory Measures

|**********| Group 1
|----------|

	index	group	itemno	item	N	Categ	AbsFreq	RelFreq	rpb.WLE	M.WLE	SD.WLE
1	1	1	1	CR055Q01S	778	0	0	0	-0.46235487	-1.38017901	1.3466267
2	2	1	1	CR055Q01S	778	1	0	0	0.46235487	0.19384856	1.0181713
3	3	1	2	DR055Q02C	682	0	0	0	-0.49515046	-0.69075675	1.1781250
4	4	1	2	DR055Q02C	682	1	0	0	0.49515046	0.49076268	0.8676635
5	5	1	3	DR055Q03C	729	0	0	0	-0.54317108	-0.86871974	1.0563386
6	6	1	3	DR055Q03C	729	1	0	0	-0.06095879	-0.09324403	0.8319815
7	7	1	3	DR055Q03C	729	2	0	0	0.56148709	0.74546806	0.8166986
8	8	1	4	DR055Q05C	701	0	0	0	-0.54183002	-1.17412418	1.1979983
9	9	1	4	DR055Q05C	701	1	0	0	0.54183002	0.37586451	0.8946237
10	10	1	5	CR104Q01S	720	0	0	0	-0.54644107	-0.87832040	1.1362439
11	11	1	5	CR104Q01S	720	1	0	0	0.54644107	0.46994636	0.8886180
12	12	1	6	CR104Q02S	703	0	0	0	-0.28858903	-0.20891336	1.1552008
13	13	1	6	CR104Q02S	703	1	0	0	0.28858903	0.57133297	1.1021710
14	14	1	7	CR104Q05S	693	0	0	0	-0.48989429	-0.41086060	1.0894597
15	15	1	7	CR104Q05S	693	1	0	0	0.46796619	0.72863534	0.8114617
16	16	1	7	CR104Q05S	693	2	0	0	0.12649346	1.56290411	0.4941710
17	17	1	8	CR111Q01S	775	0	0	0	-0.52197597	-1.19730969	1.2719064
18	18	1	8	CR111Q01S	775	1	0	0	0.52197597	0.29993647	0.9356762
19	19	1	9	DR111Q02BC	678	0	0	0	-0.49392087	-1.29905569	1.2502401
20	20	1	9	DR111Q02BC	678	1	0	0	0.05800254	0.13567060	0.8692091
21	21	1	9	DR111Q02BC	678	2	0	0	0.34416199	0.81067227	0.9412875
22	22	1	10	DR111Q06C	770	0	0	0	-0.50033027	-0.88938731	1.2012670
23	23	1	10	DR111Q06C	770	1	0	0	-0.12127781	-0.27180858	0.8097247
24	24	1	10	DR111Q06C	770	2	0	0	0.56984720	0.72800611	0.7968399
25	25	1	11	CR227Q01S	781	0	0	0	-0.34230447	-0.65779558	1.3253033
26	26	1	11	CR227Q01S	781	1	0	0	0.34230447	0.23372783	1.0282066
27	27	1	12	CR227Q02S	784	0	0	0	-0.50374012	-1.49660060	1.3025356
28	28	1	12	CR227Q02S	784	1	0	0	-0.06266370	-0.12687063	0.9055463
29	29	1	12	CR227Q02S	784	2	0	0	0.43197638	0.60659693	0.9427581
30	30	1	13	DR227Q03C	669	0	0	0	-0.53873931	-0.95453403	1.1665866
31	31	1	13	DR227Q03C	669	1	0	0	0.53873931	0.46891935	0.8984994
32	32	1	14	DR227Q06C	762	0	0	0	-0.48672160	-1.38921501	1.1943798
33	33	1	14	DR227Q06C	762	1	0	0	0.48672160	0.24000963	0.9766438

Differential Item Functioning (DIF)

```
Item Facet Parameters Xsi
            parameter       facet    xsi  se.xsi
1           CR055Q01S        item -2.209  0.112
2           DR055Q02C        item -0.709  0.091
3           DR055Q03C        item -0.878  0.089
4           DR055Q05C        item -1.659  0.105
5           CR104Q01S        item -0.796  0.089
6           CR104Q02S        item  1.350  0.095
7           CR104Q05S        item  0.742  0.088
8           CR111Q01S        item -1.559  0.096
9          DR111Q02BC        item -2.201  0.123
10          DR111Q06C        item -0.971  0.088
11          CR227Q01S        item -1.032  0.088
12          CR227Q02S        item -2.196  0.111
13          DR227Q03C        item -1.209  0.099
14          DR227Q06C        item -2.229  0.115
15            gender1      gender -0.231  0.026
16            gender2      gender  0.231  0.026
17  CR055Q01S:gender1 item:gender -0.084  0.080
18  DR055Q02C:gender1 item:gender  0.213  0.071
19  DR055Q03C:gender1 item:gender -0.020  0.071
20  DR055Q05C:gender1 item:gender -0.108  0.078
21  CR104Q01S:gender1 item:gender  0.017  0.070
22  CR104Q02S:gender1 item:gender  0.191  0.073
23  CR104Q05S:gender1 item:gender  0.204  0.070
24  CR111Q01S:gender1 item:gender  0.087  0.074
25 DR111Q02BC:gender1 item:gender -0.357  0.084
26  DR111Q06C:gender1 item:gender -0.091  0.070
27  CR227Q01S:gender1 item:gender  0.059  0.070
28  CR227Q02S:gender1 item:gender -0.073  0.080
29  DR227Q03C:gender1 item:gender -0.065  0.075
30  DR227Q06C:gender1 item:gender  0.026  0.269
31  CR055Q01S:gender2 item:gender  0.084  0.080
32  DR055Q02C:gender2 item:gender -0.213  0.071
33  DR055Q03C:gender2 item:gender  0.020  0.071
```

Item Selection

Possible exclusion criteria:

- Item fit > 1.15 (or 1.20)
- Point-biserial correlation < .2 (or < .3)
- DIF-logit > .25 (or .30)

Literature

Le, L. (2006). *Analysis of differential item functioning*. Presented at the annual conference of the American Educational Research Association (AERA), San Francisco, CA.

OECD. (2005). *PISA 2003 technical report*. Paris: OECD.

Olson, J. F., Martin, M. O., & Mullis, I. V. S. (Eds.). (2008). *TIMSS 2007 technical report*. Chestnut Hill, MA: TIMSS & PIRLS International Study Center, Lynch School of Education, Boston College.

Test Reliability

CTT Reliability

- ❑ Cronbach’s alpha
  (should be above .8, above .7 might be alright…)

IRT Reliabilities

- ❑ WLE reliability
  (does *not* consider multidimensional and regression information, usually close to Cronbach’s Alpha)
- ❑ EAP/PV reliability
  (considers multidimensional and regression information)

Literature

OECD. (2017). *PISA 2015 technical report.* Paris: OECD.

Olson, J. F., Martin, M. O., & Mullis, I. V. S. (Eds.). (2008). *TIMSS 2007 technical report.* Chestnut Hill, MA: TIMSS & PIRLS International Study Center, Lynch School of Education, Boston College.

Estimation of Ability Scores

EAP Scores

- ❑ Expected A Posteriori estimates
- ❑ The mean of each student's distribution of the Plausible Values
- ❑ Are included in the model object

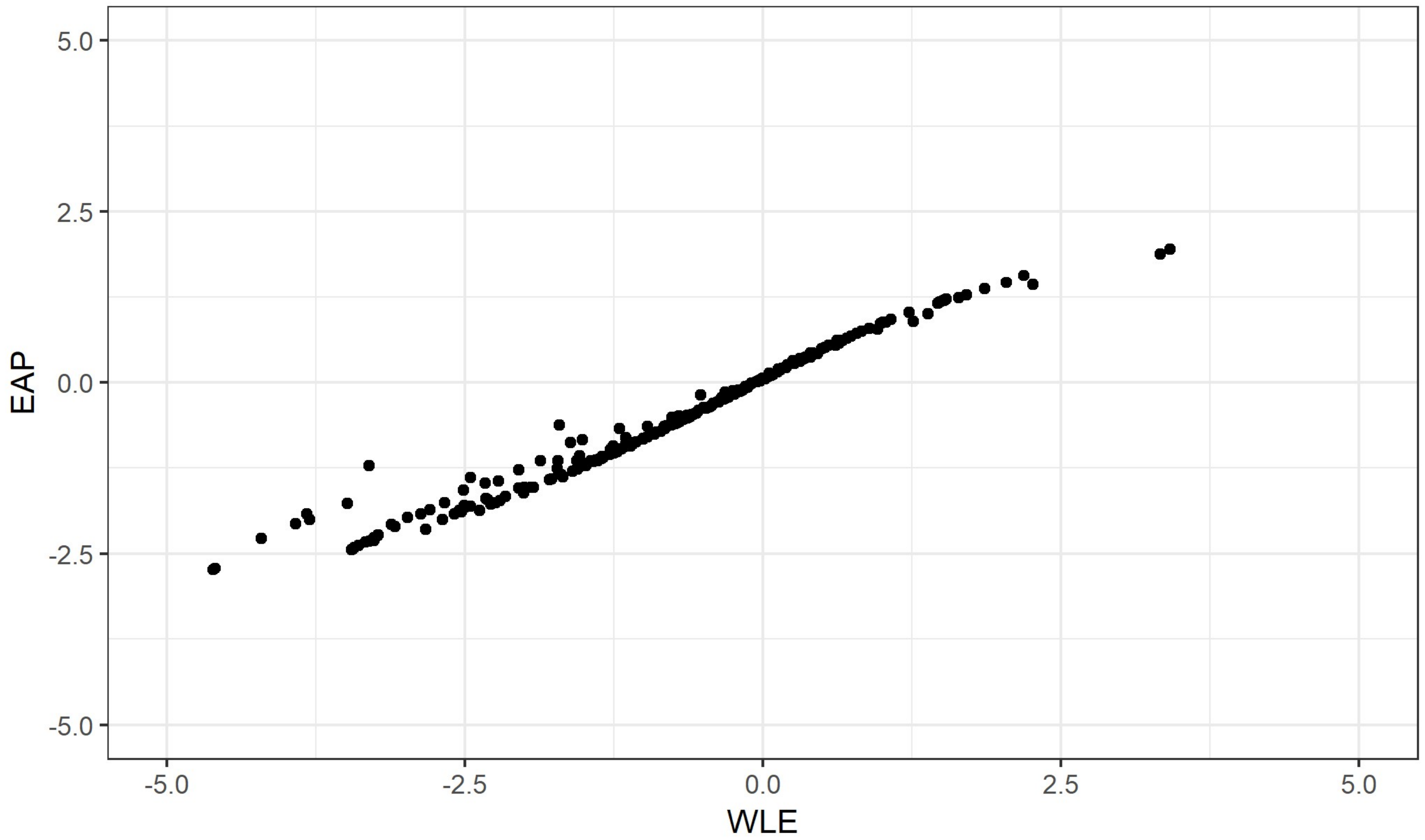
WLE Scores

- ❑ Weighted Likelihood Estimates
- ❑ Best point estimate for a student's ability score
- ❑ Are calculated via the function `tam.wle()`

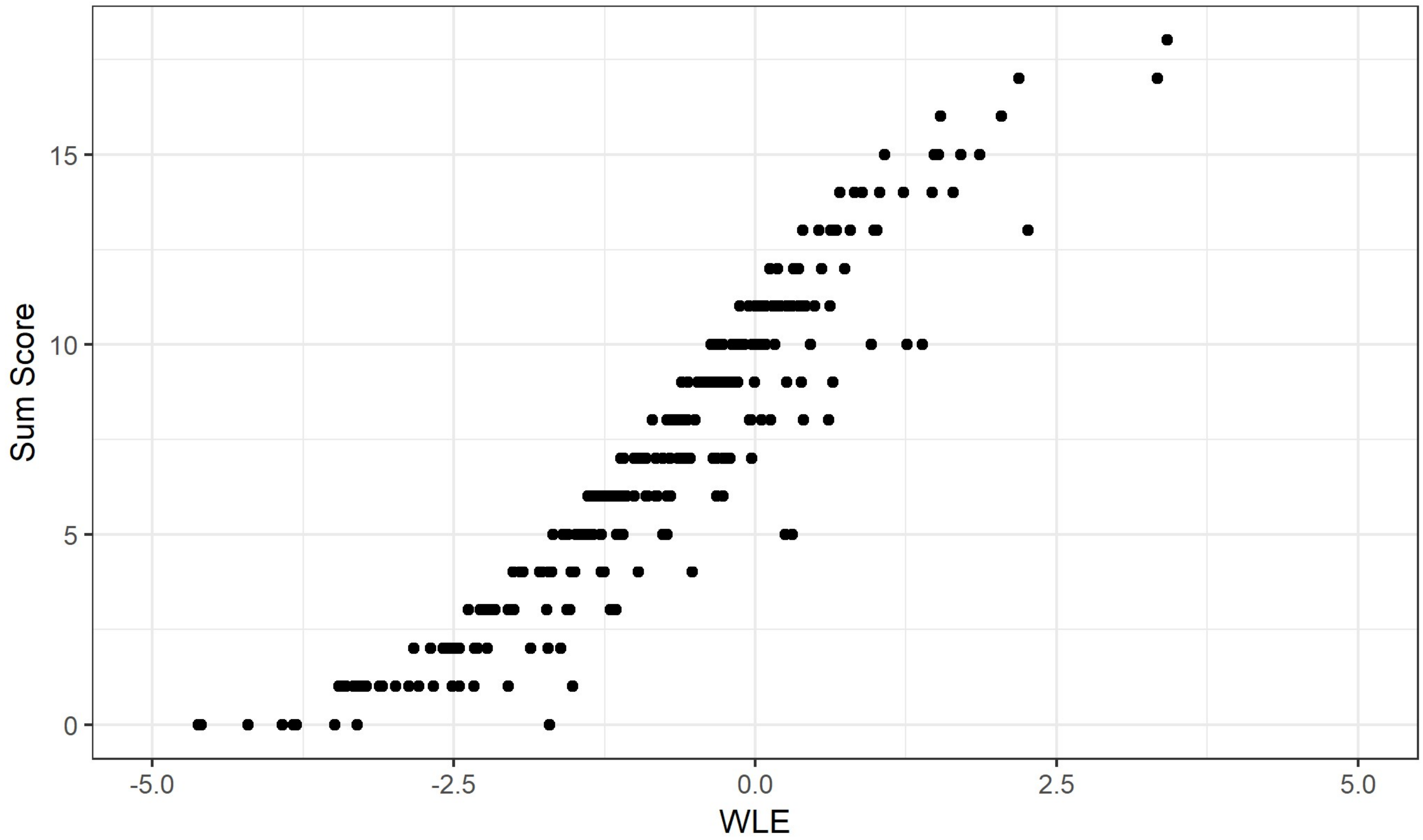
Plausible Values

- ❑ Include regression information for the calculation of group means that are free of estimation errors
- ❑ Are calculated via the function `tam.pv()`

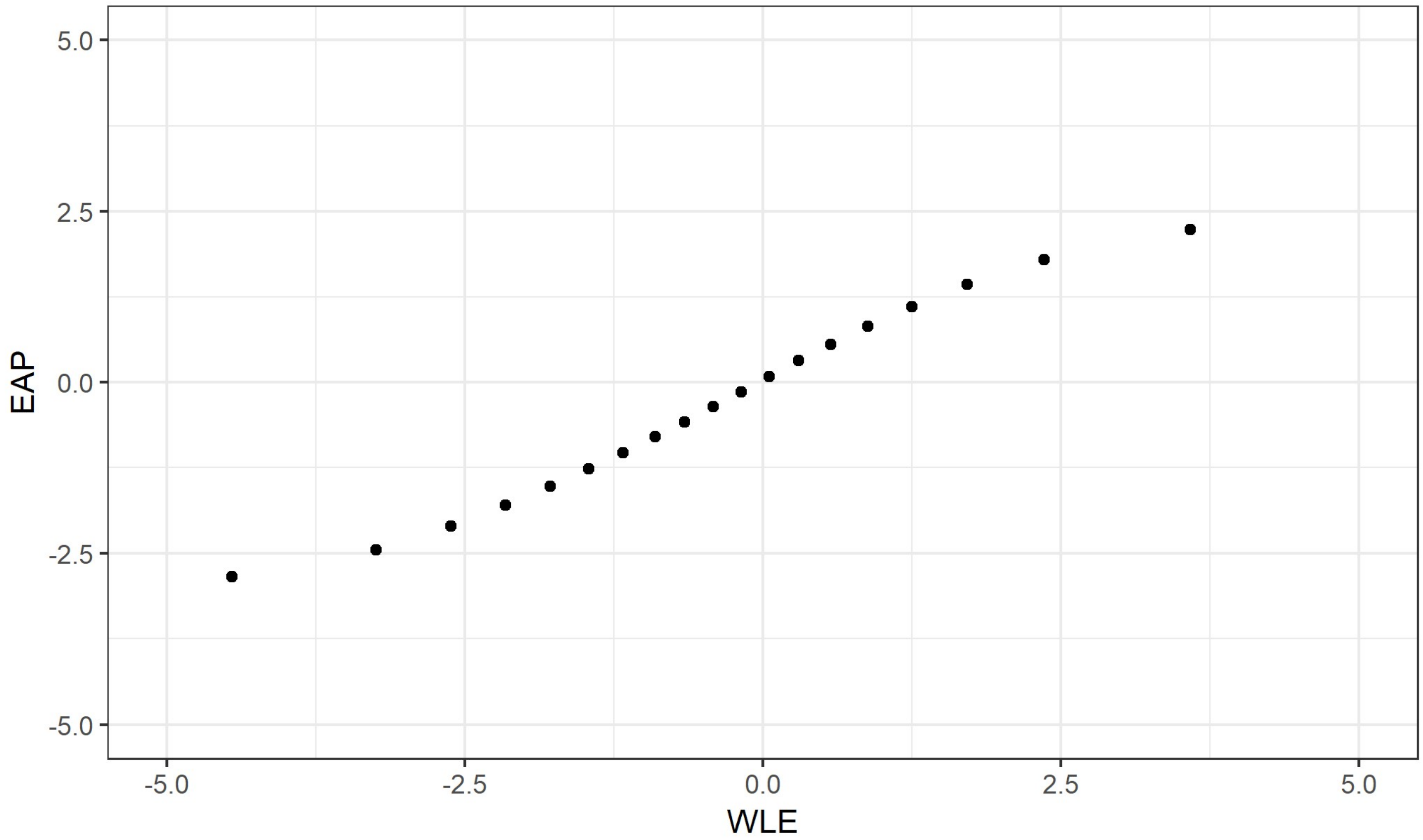
WLE vs. EAP



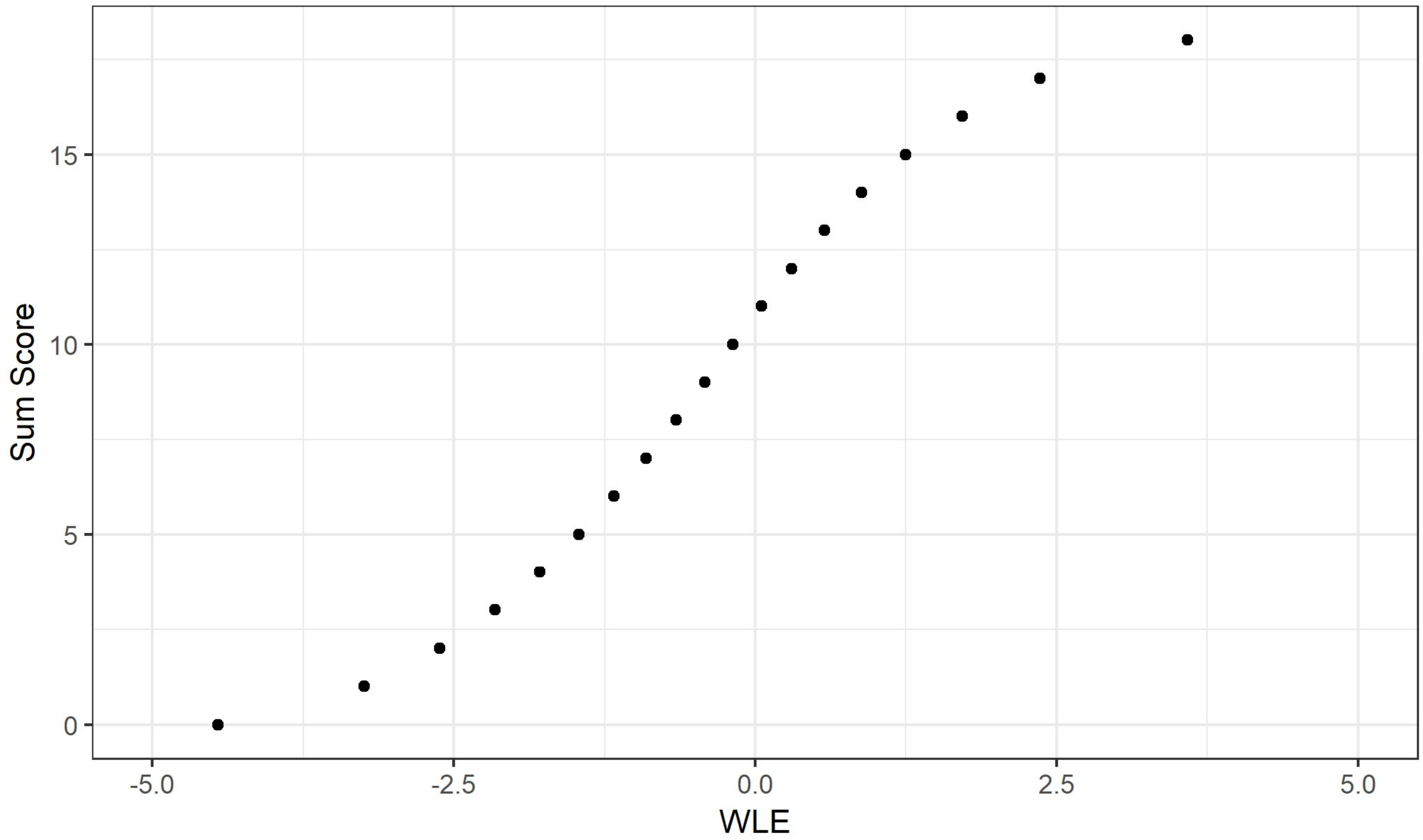
WLE vs. Sum Score



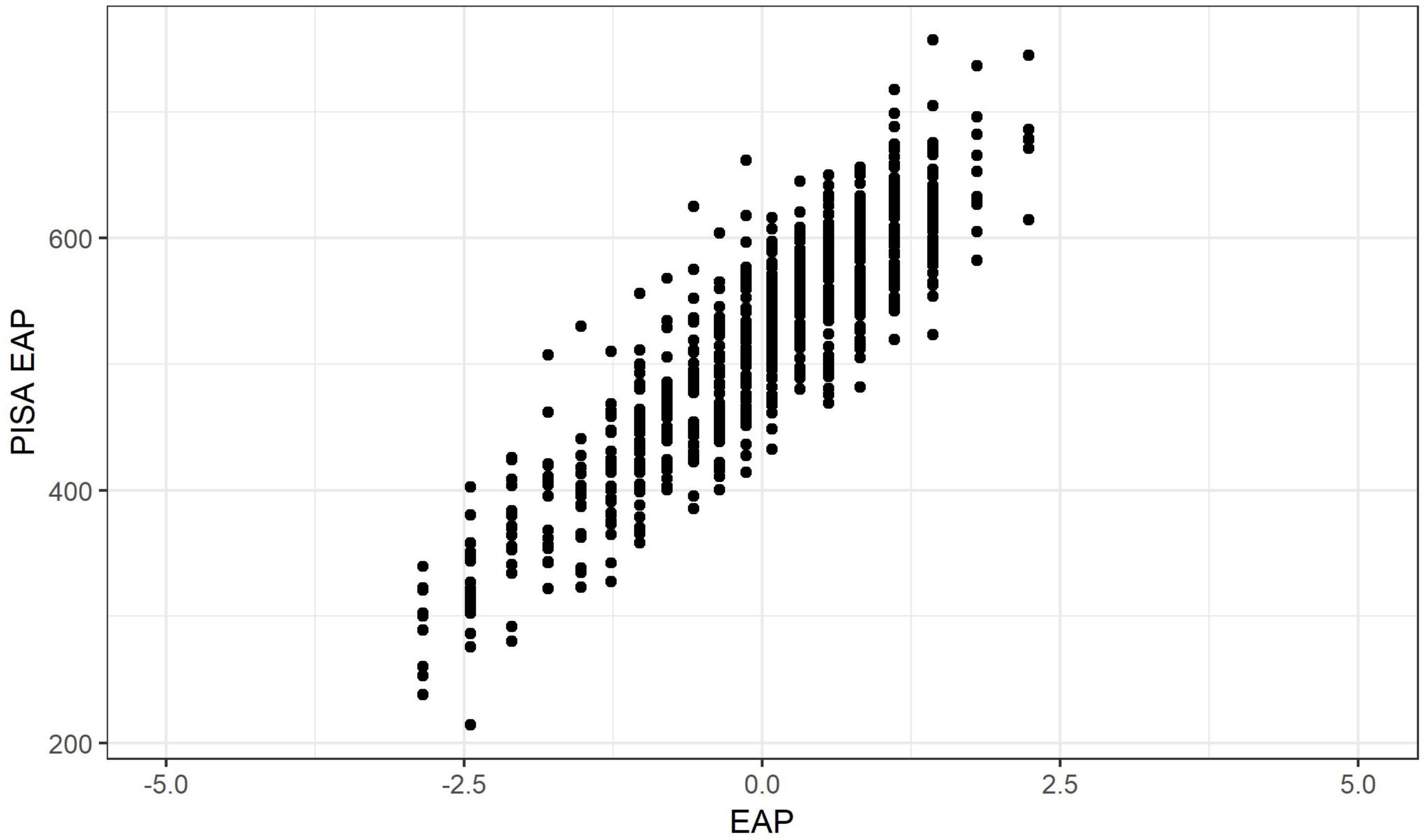
WLE vs. EAP – Without Missings



WLE vs. Sum Score – Without Missings



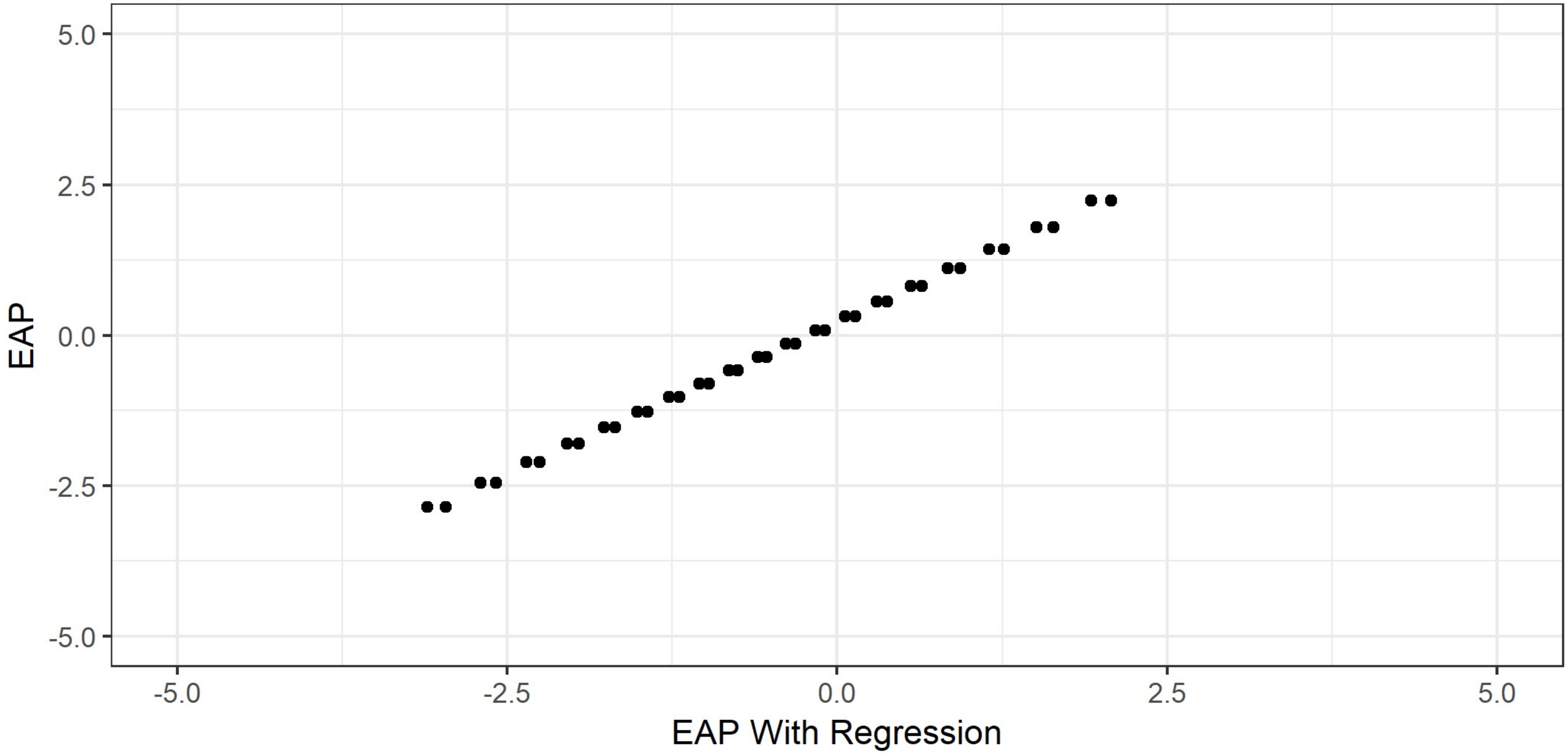
Comparison to the PISA EAP



IRT Model Estimation With Regression

- To estimate correlations and group means without the estimation errors for the individual abilities
- Correlation and group means will only be without estimation errors if the group information is included in the regression ("background") model
- More reliable estimation for data with missings at random due to the information used from the regression model

EAP vs. EAP With Regression



WLE vs. WLE With Regression