UNI
FR

R-Workshop

Caroline Gahrmann, Yves Steiner
(& minor adjustments by Bernhard Piskernik)

Agenda

- Introduction
- Getting started
- Data Wrangling
- Descriptive Statistics
- Inference Statistics
- Resources & Best Practice

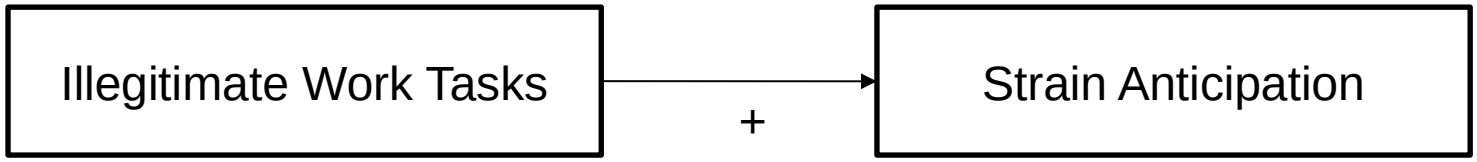
UNI FR

Why R?

- R is Open Source → Free of charge
- R is developed by researchers all over the world and constantly evolves
- R has more statistical functions than any other software
- R (or the R community) react to questions or error messages

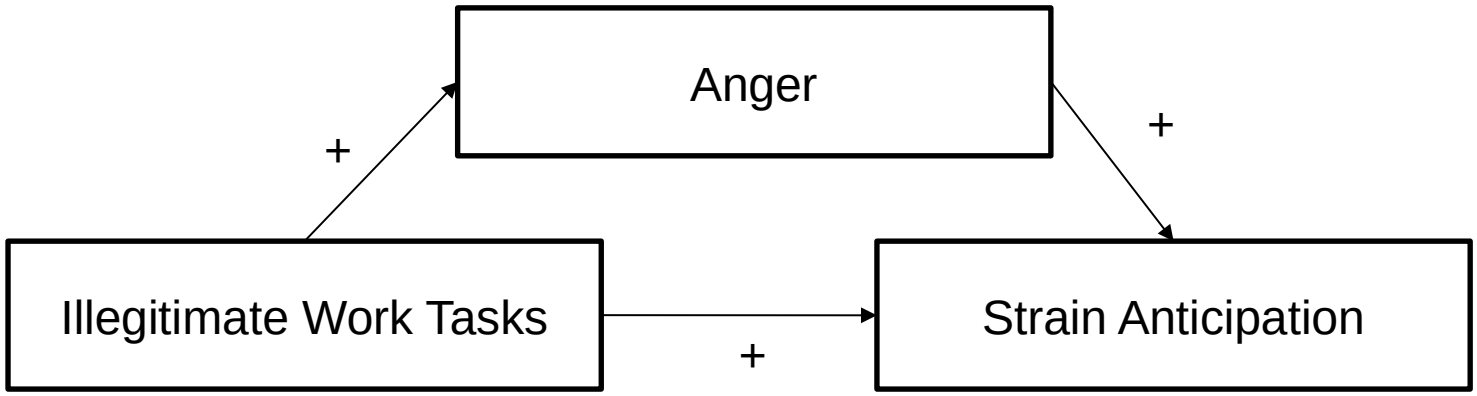
Our Practice Data

For the duration of the workshop, we are interested in exploring the following hypotheses:



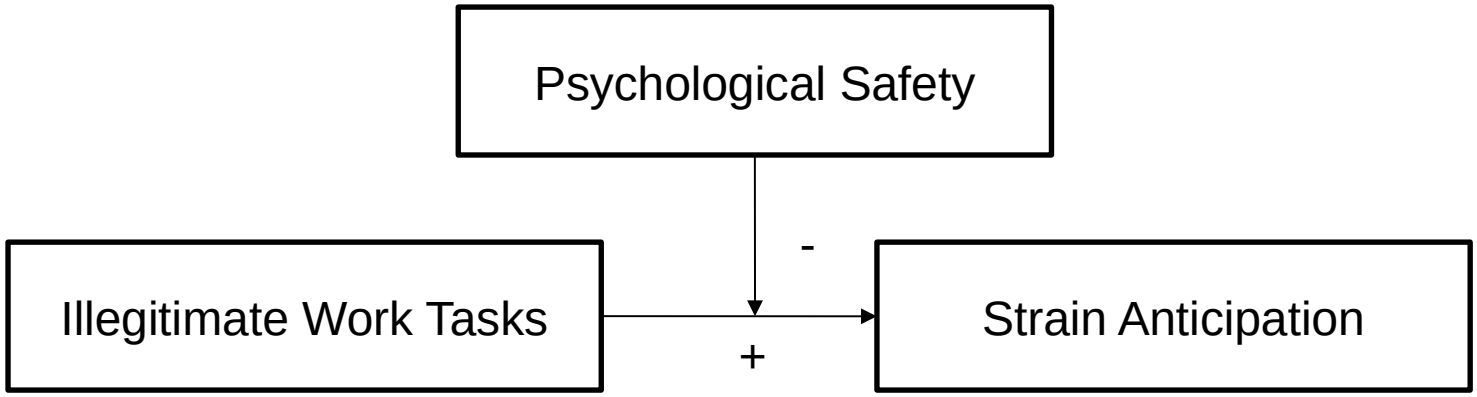
H1: Employees who experience more illegitimate work tasks experience more strain anticipation.

Our Practice Data



H2: The effect of illegitimate work task on strain anticipation is partially mediated by anger.

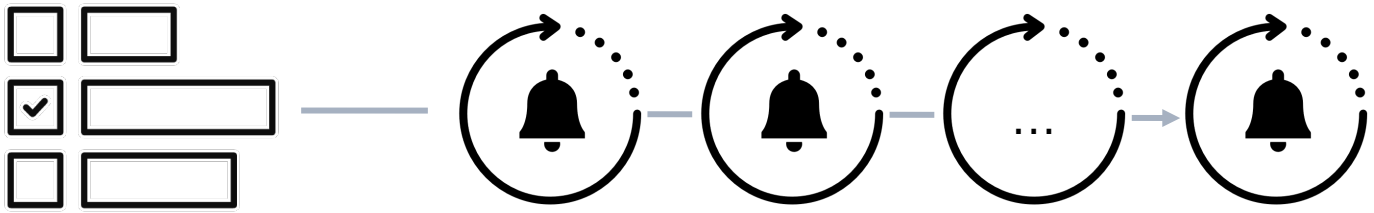
Our Practice Data



H3: The effect of illegitimate work task on strain anticipation is attenuated by psychological safety.

Our Practice Data

To explore these hypotheses, we have implemented the following study design:



BASELINE t0 | DAILY t1 | DAILY t2 | DAILY t3-t9 | DAILY t10

Our Practice Data

We have recruited **N = 94** employees to participate in the study.

To characterize the sample, we assessed the following information:

Variable	Question
ST_GENDER	What gender do you identify with the most?
	1 = female; 2 = male; 3 = not listed
ST_AGE	How old are you?

Our Practice Data

We have collected information about our study variables via these measures:

(1) Illegitimate Work Tasks (Semmer et al., 2010)

Variable	Item
-	Today, I had to take care of work tasks, which …
DA_ILWOTA_01_t1-10	• should have been done by someone else.
DA_ILWOTA_02_t1-10	• went too far and should not have been expected from me.
DA_ILWOTA_03_t1-10	• would not exist, if other people made less mistakes.
DA_ILWOTA_04_t1-10	• made no sense at all.
1 = not at all; 5 = completely	

Our Practice Data

(2) Psychological Safety (Edmondson et al., 1999)

Variable	Item
-	Today, at work …
DA_PSYSAFE_01_t1-10	• it was possible to make mistakes without them being held against you.
DA_PSYSAFE_02_t1-10	• team members could address problems and sensitive issues without hesitation.
DA_PSYSAFE_03_t1-10	• it was possible, to take risks.
DA_PSYSAFE_04_t1-10	• it was easy to ask someone on the team for help.
DA_PSYSAFE_05_t1-10	• nobody would have intentionally undermined my efforts.
DA_PSYSAFE_06_t1-10	• my unique skills were appreciated and applied in the teams.
1 = not at all - 5 = completely; 6 = cannot answer because I was not involved in a team interaction	

Our Practice Data

(3) Anger (Cranford et al., 2006)

Variable	Item
-	Today, I felt…
DA_MOOD_ANG_01_t1-10	• angry.
DA_MOOD_ANG_02_t1-10	• resentful.
DA_MOOD_ANG_03_t1-10	• annoyed.
1 = not at all; 5 = completely	

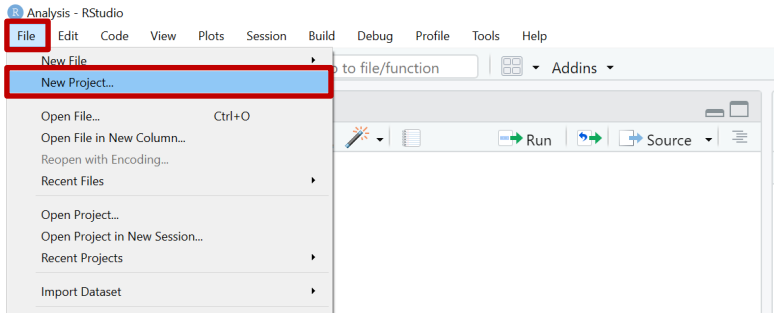
Our Practice Data

(4) Strain Anticipation (Scott et al., 2015)

Variable	Item
DA_ANTC_01_t1-10	• How straining do you think your next workday will be?
DA_ANTC_02_r_t1-10	• How pleasant do you think your next working day will be?
1 = not at all; 100 = very	

Creating a New Project

First things first: We have to create a new project for our data analysis. We should create a folder for this beforehand.

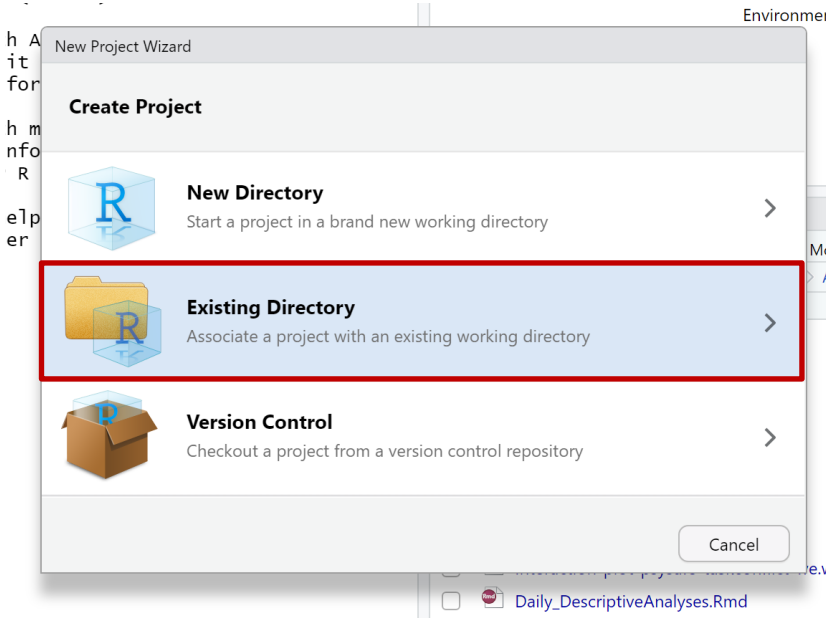


Step 1: Open RStudio.

Step 2: Select "File" on the upper toolbar.

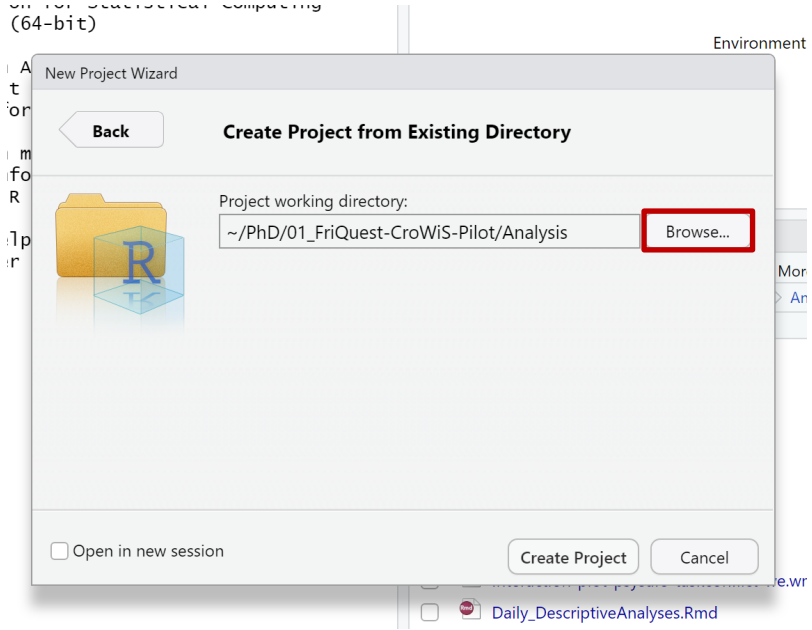
Step 3: Select "New Project".

Creating a New Project



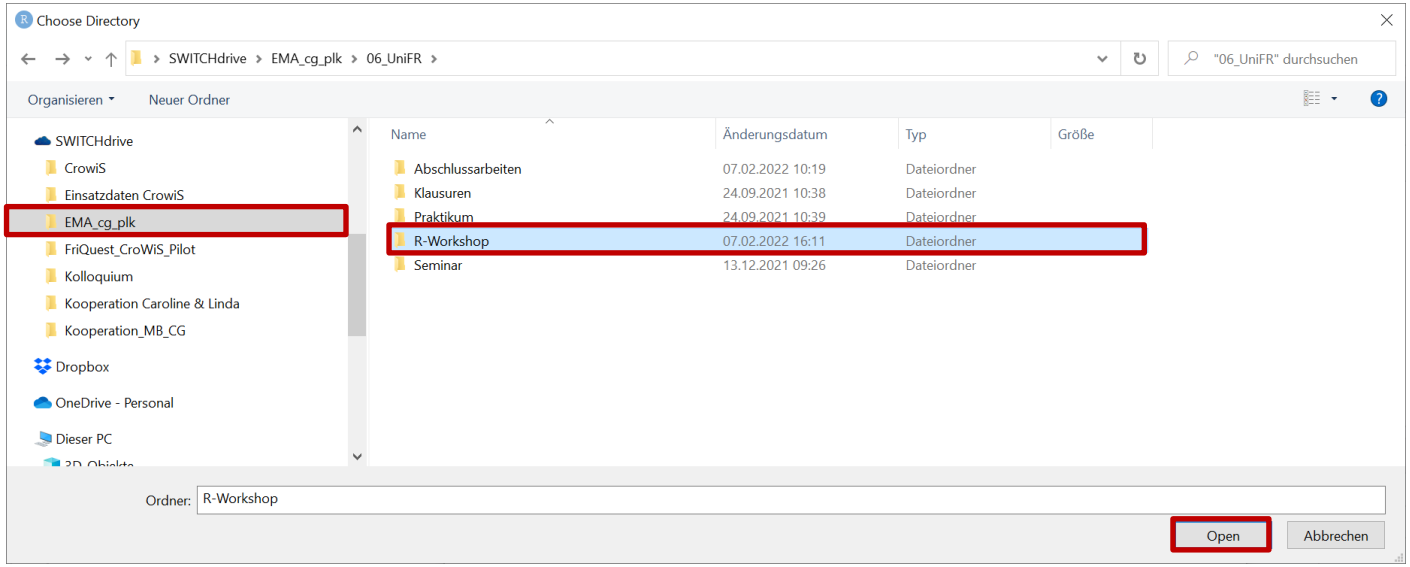
Step 4: Select "Existing Directory".

Creating a New Project



Step 5: Select "Browse".

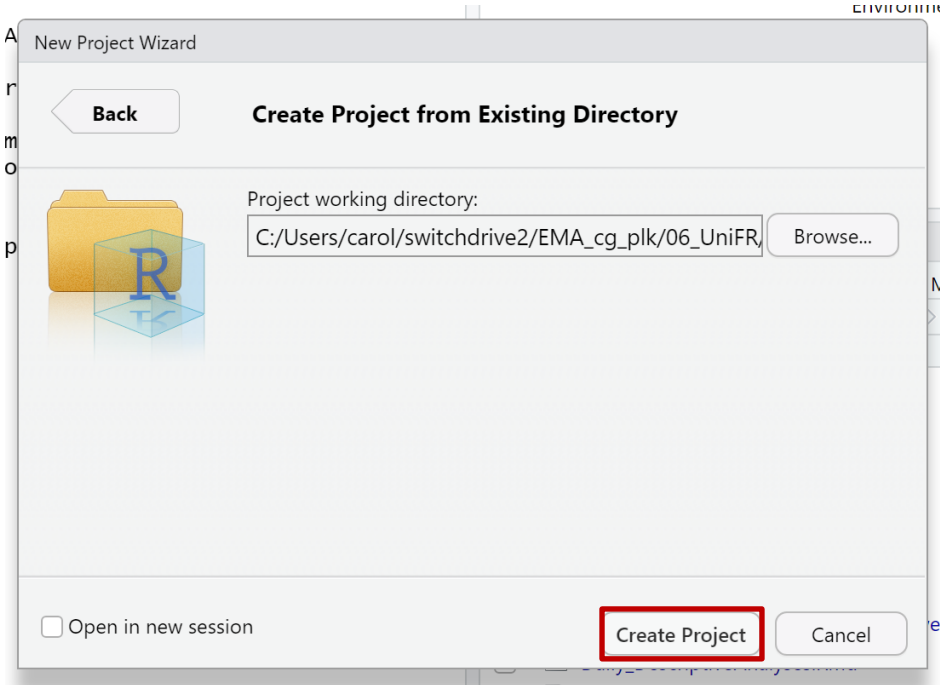
Creating a New Project



Step 6: Select folder of choice.

Step 7: Select "Open".

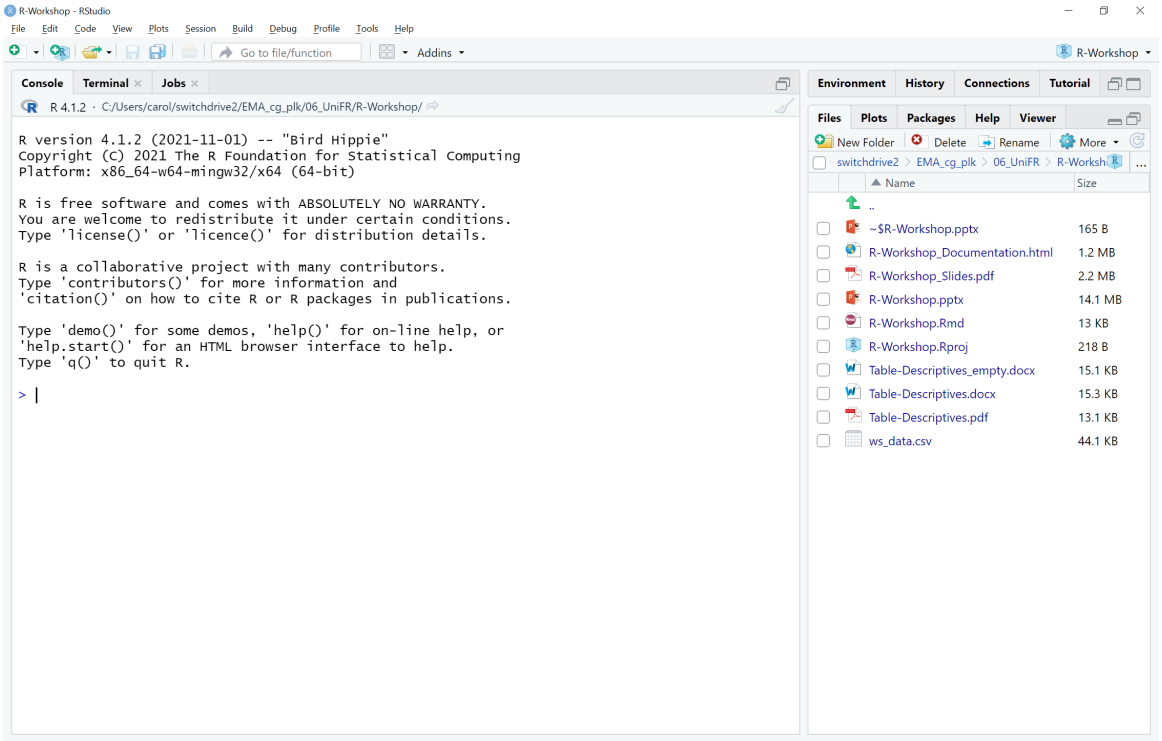
Creating a New Project



Step 8: Select "Create Project".

Creating a New Project

Hurray! We have now created a new project which functions as our working directory.



Introducing R Markdown

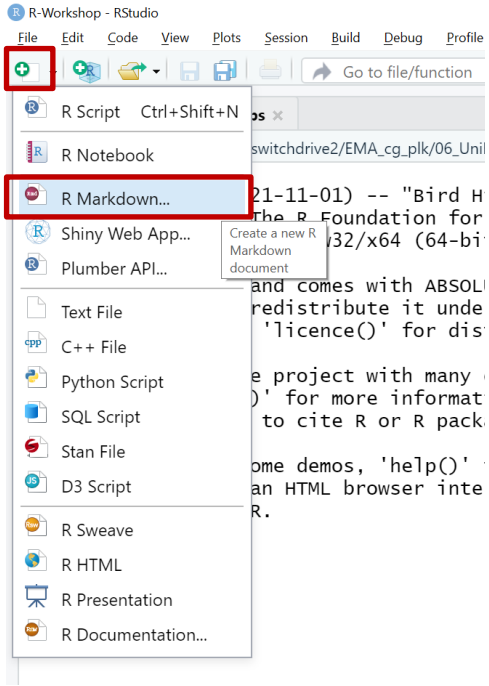
R Markdown can

… save and execute code

… generate static reports in different formats

You should use R Markdown to document and share your analysis with others.

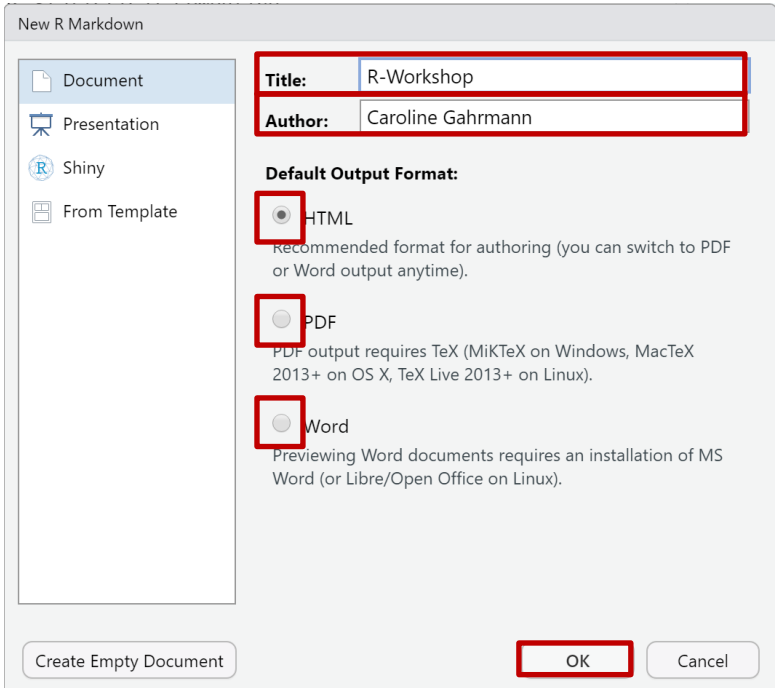
Creating an R Markdown



Step 1: Select the "New File" Icon on the second row of the toolbar.

Step 2: Select "R Markdown".

Creating an R Markdown



Step 3: Specify title of your R Markdown.

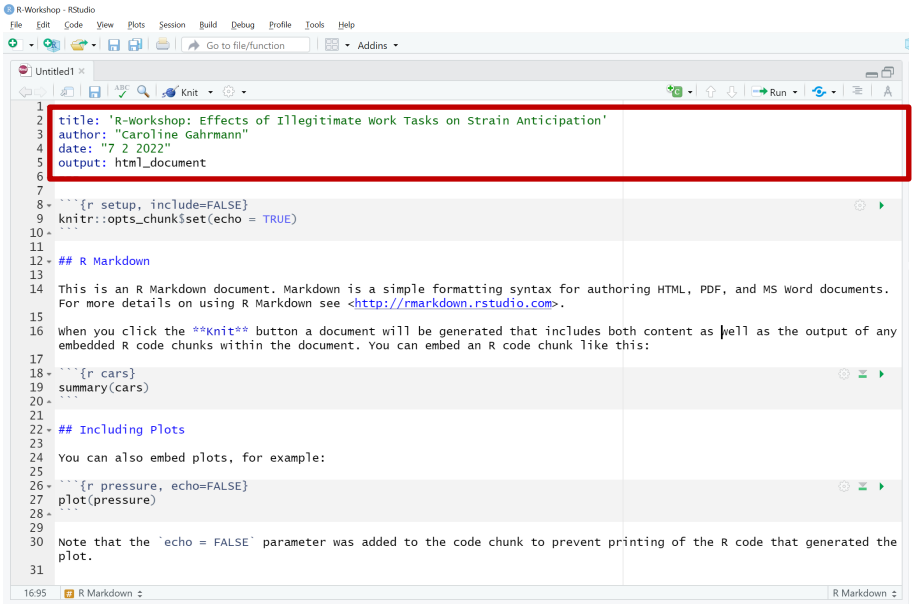
Step 4: Specifiy the author of your R Markdown.

Step 5: Specify which type of document you want to generate.

Step 6: Select 'OK'.

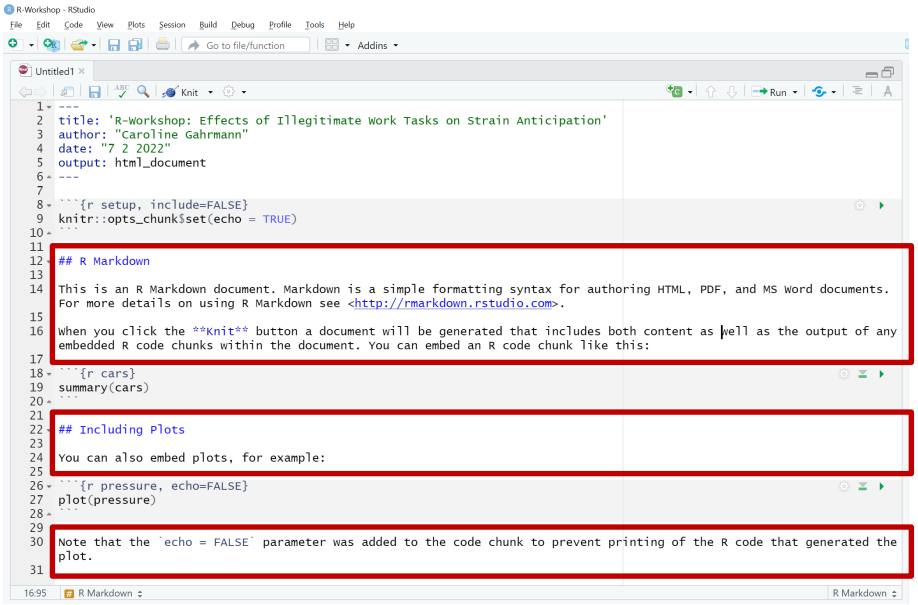
R Markdown explained

R Markdowns consist of three parts:



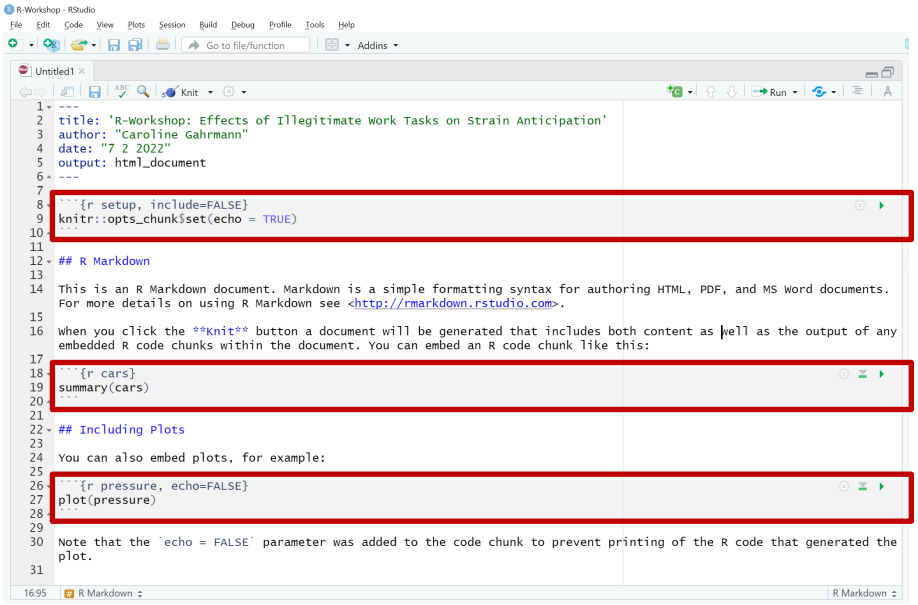
YAML metadata → guides generation of static documents

R Markdown explained



Text → details statistical analysis

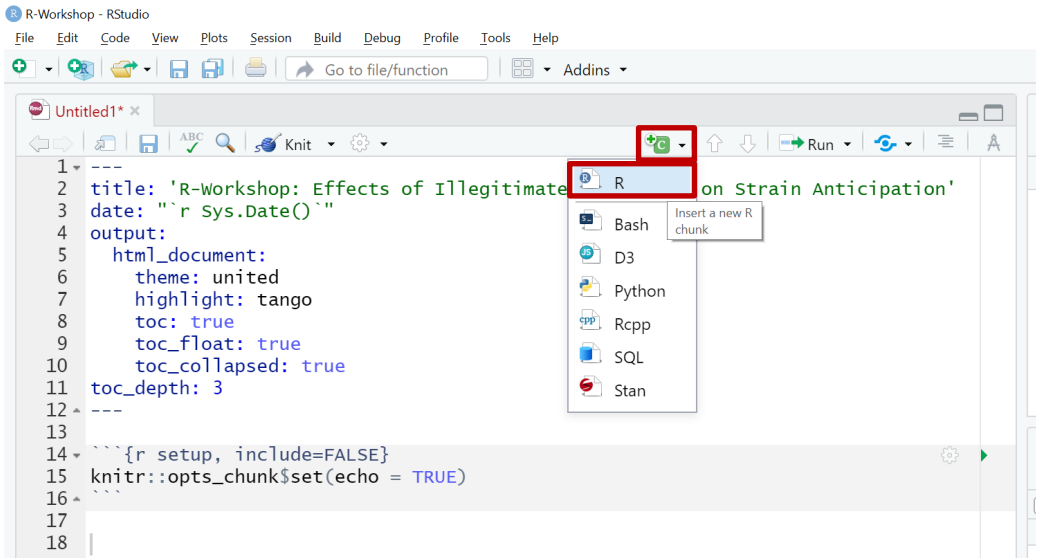
R Markdown explained



Code → R functions

R Markdown How To

To insert new code chunks …

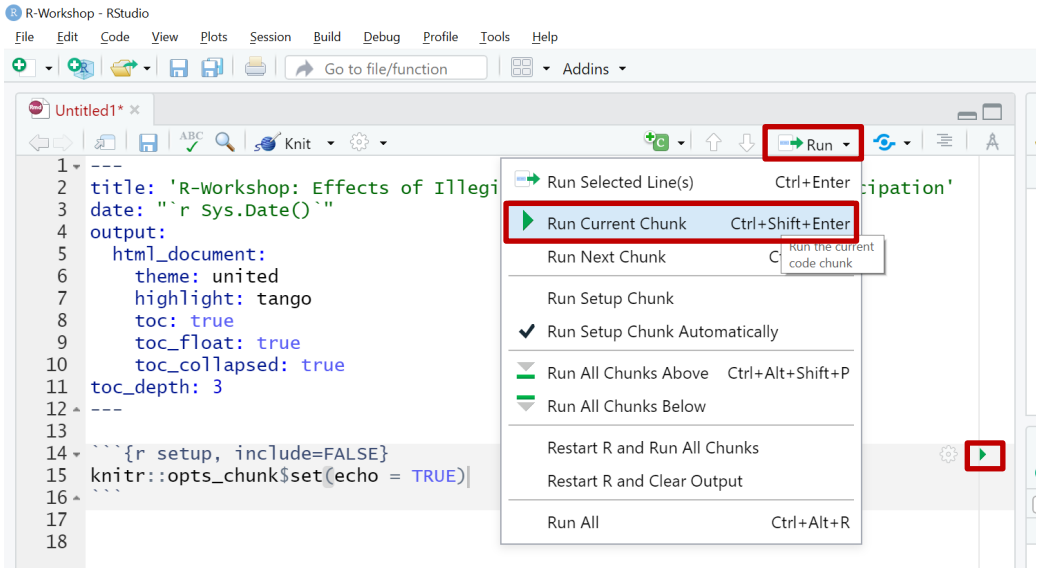


Step 1: Select the "Insert a new code chunk" Icon in the toolbar of the Editor.

Step 2: Select the "Insert a new R chunk" Icon.

R Markdown How To

To run a code chunk …



Step 1: Click into the code chunk you want to run.

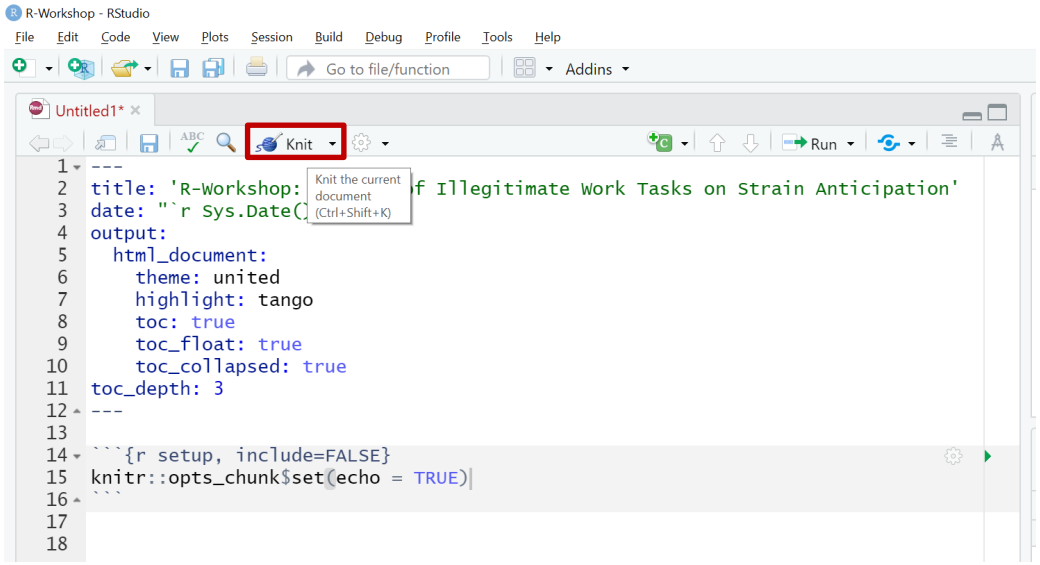
Step 2: Select the "Run" Icon in the toolbar of the Editor.

Step 3: Select "Run Current Chunk".

Alternative: Select the green triangle in the code chunk.

R Markdown How To

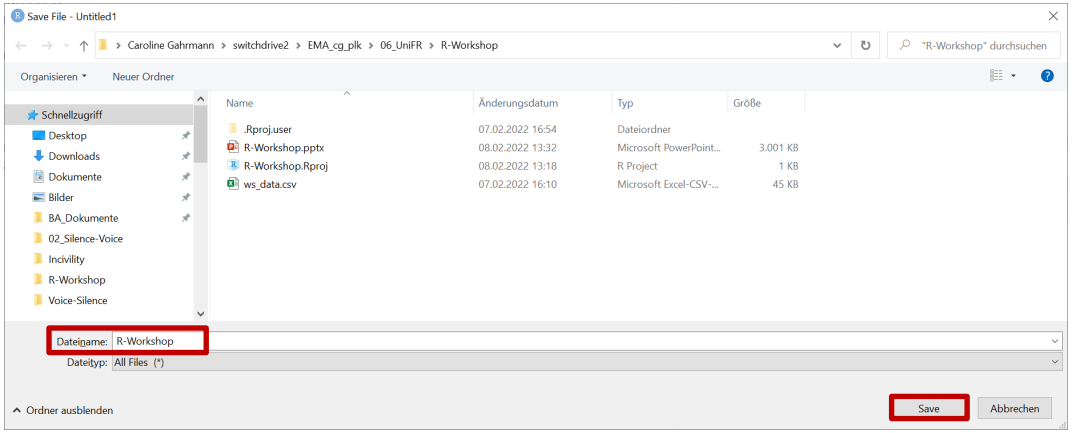
To generate a static report …



Step 1: Select the "Knit" Icon in the toolbar of the Editor.

R Markdown How To

To generate static report …



Step 2: Type in document title.

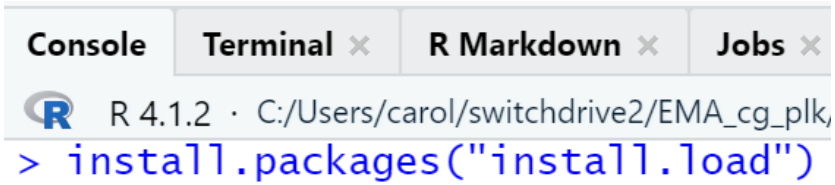
Step 3: Select "Save".

You only have to do Step 2 and Step 3 once per R Markdown.

Prepare Packages

We want to first install all relevant packages.

In R, installing and loading packages are separate processes. We can join these procedures by installing the 'install.load' package:



Note that we only have to install the package once. We do this in the Console and not in the RMarkdown / R-script.

Prepare Packages

To install packages, if necessary, and load them we can now use **library(install.load) and install_load()**:

```r
```{r}
library("install.load")

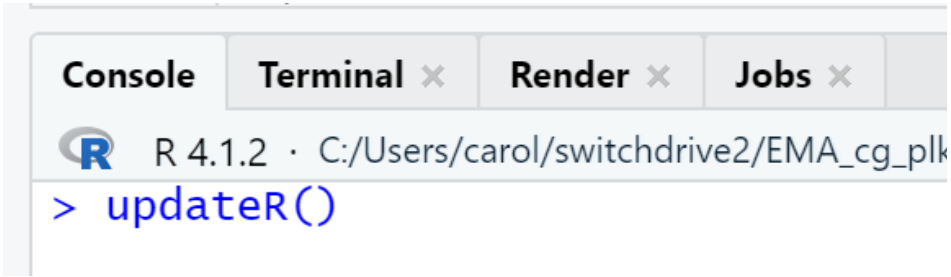
install_load("installr",
             "tidyverse",
             "summarytools",
             "skimr",
             "psych",
             "interactions")
```
```

library("install.load")
install_load("package1", "package2", "…")

Please load 'install.load' and prepare 'installr', 'tidyverse', 'summarytools', 'skimr', 'psych' and 'interactions' for the workshop.

Check and Update R Version

To check if we have the newest R version installed, we can use updateR() in the Console:

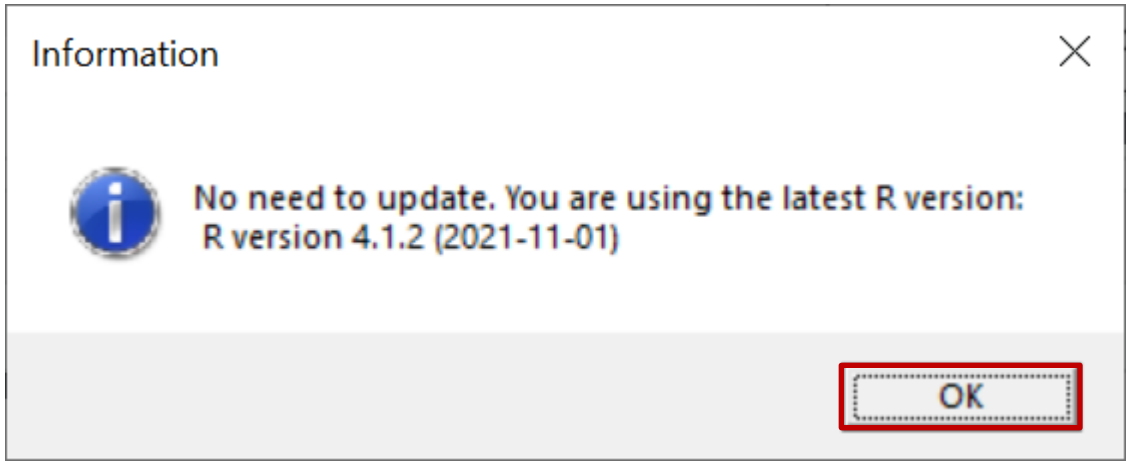


updateR()

Please execute this code in the Console and do not save it in the RMarkdown. The RMarkdown won't be able to compile otherwise.

Check and Update R Version

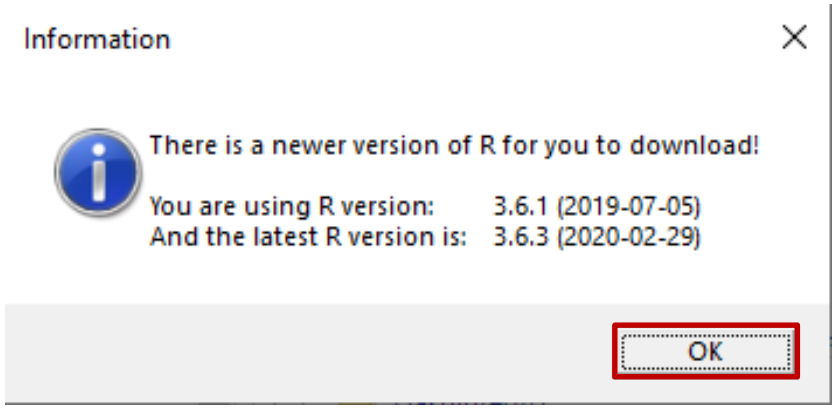
If we have the newest R version already, we get the following message.



We only have to select 'OK' an move on.

Check and Update R Version

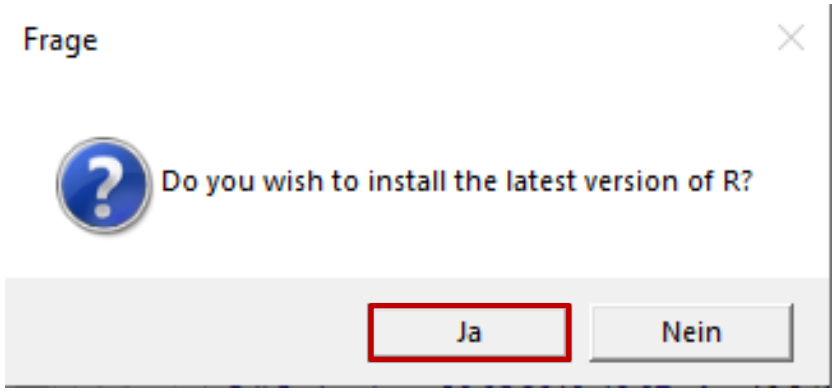
If we don't have the newest R version, we get the following message:



We can now download the newest version:

Step 1: Select 'OK'.

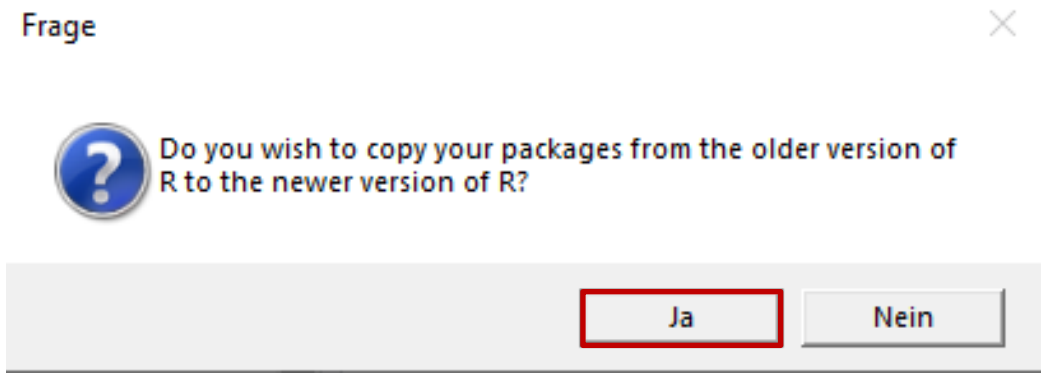
Check and Update R Version



Step 2: Select 'Yes'.

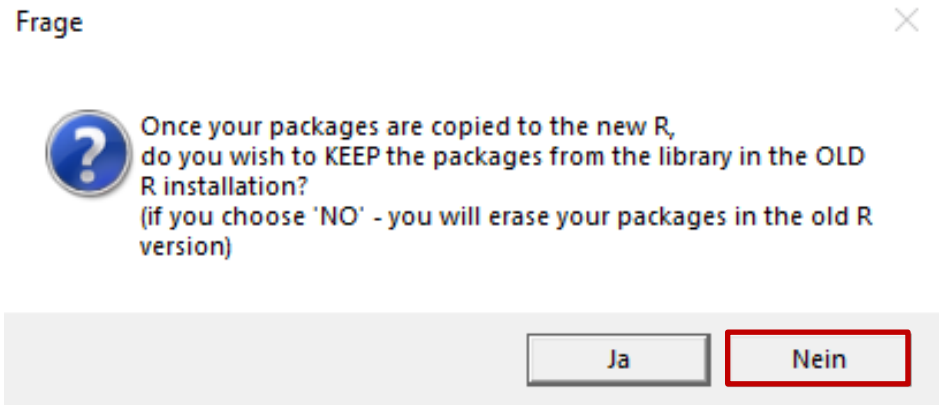
Check and Update R Version

Hurray! We get to keep our packages!



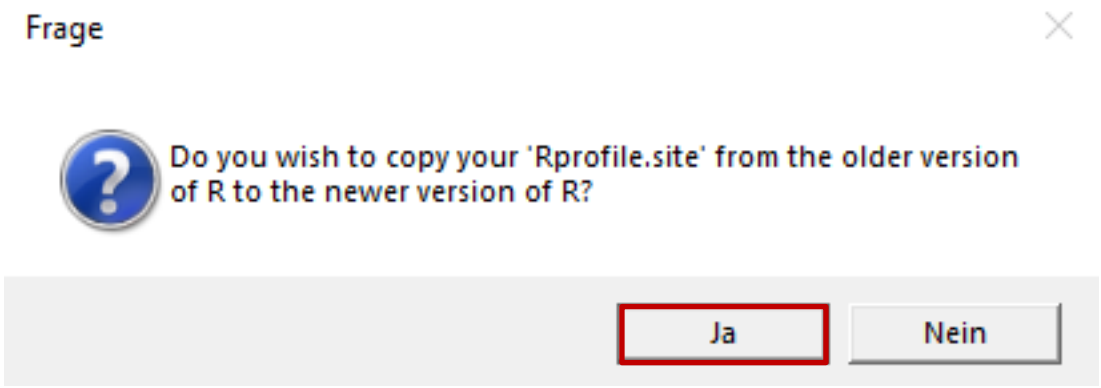
Step 3: Select 'Yes': You want to keep all your packages in place.

Check and Update R Version



Step 4: Select 'No': You don't want to use the old version of R and therefore don't need the packages for that version any longer.

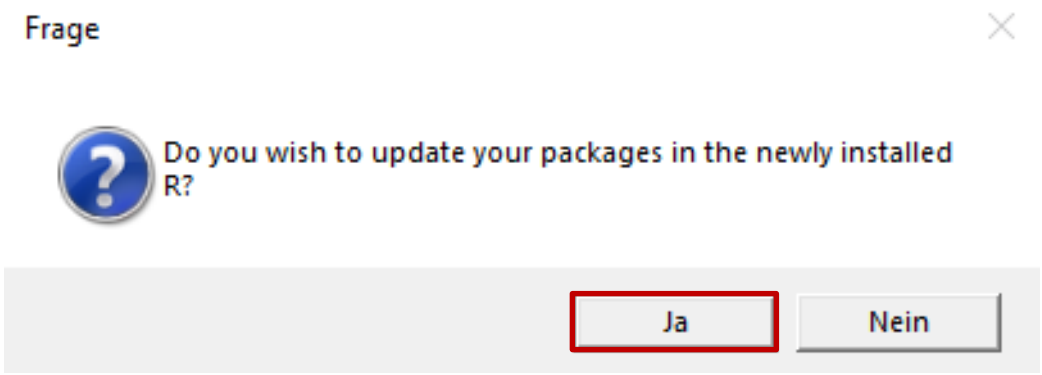
Check and Update R Version



Step 5: Select 'Yes'.

Check and Update R Version

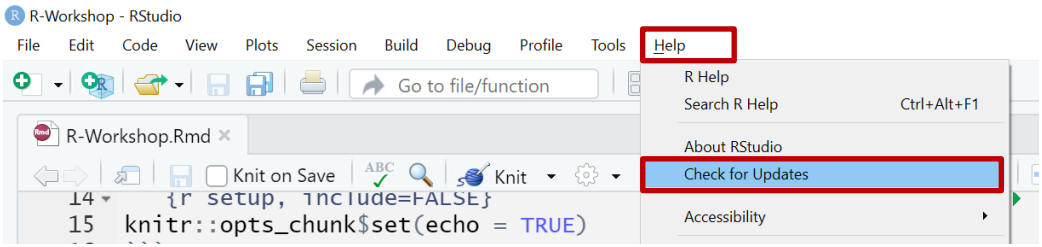
We even have the chance to update all our packages!



Step 6: Select 'Yes'.

RStudio Updates

We can also check if we have the newest version of RStudio:



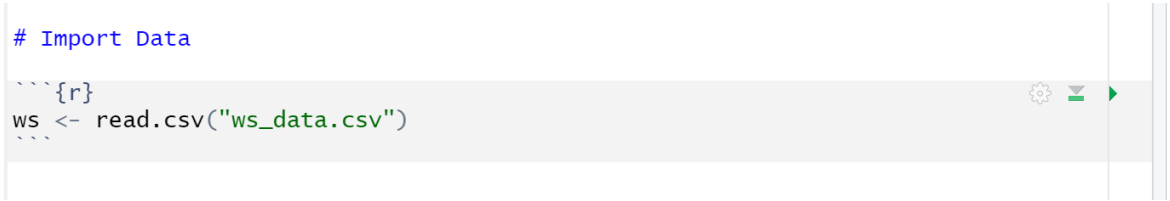
Step 1: Select 'Help' on the upper toolbar.

Step 2: Select 'Check for Updates'.

If a new version is available, the RStudio page opens in the browser. We can install it there.

Import Data

Our practice data is a .csv file. To import it, we can use **read.csv()**:



Step 1: internal-name-for-dataset <- read.csv("dataset.csv")

Step 2: run code chunk.

You can also import data in other formats using the "read" function. You find more information here: https://intro2r.com/importing-data.html

What now?

We have imported our data. It looks like this:

session	ST_GENDER	ST_AGE	DA_ILWOTA_01_t1	DA_ILWOTA_01_t2	DA_ILWOTA_01_t3
GNzseWN54oLXACHpgu76FSaVPWr-0RoPU0FLwzVNVJ2F4o...	1	27	3	5	1
En6I3O-6uskCuUt3PbKgoF8SrEbpiZEHkjcQtUttftkSx3YiHSF9...	2	44	1	1	1
gV_qMoDBuP3sFBKD0hlGctOK5A7m9_6q8pWHepcrgfIZ9Zb...	1	32	1	1	1
52uu82_bvBNo_KPUY8gwVHjqZHU8GyOCZhRPTetexFLa769...	2	65	1	1	1
JaACkwHXlc8HB5c6nYWkILQWz9hhyLwuvFMyxgZJ3P1iq_wq...	1	34	2	1	1
5UqUs7skgodf-KWAEGDIJo28GTEiQy8fE2txL4BUv8IodvDSb...	1	31	3	1	2
lENipEbGFFhnQKfDJ_mAluvWHDt0tuRliaPeF9iNXJq0E1dXmt...	1	32	1	1	1

In the end, we want to probe our hypotheses via regression analyses.

What do have to do to get there?

Rescale Items

We have inverted items in our data:

Variable	Item
DA_ANTC_01_t1-10	• How straining do you think your next workday will be?
DA_ANTC_02_r_t1-10	• How pleasant do you think your next working day will be?
1 = not at all; 100 = very	

We have to reverse 'DA_ANTC_02_r_t1' – 'DA _ANTC_02_r_t10' so it reflects employee *strain* anticipation.

Rescale Items

To revese items, we can use **mutate ()**:

```
# Data Preparation

## Reverse Items

```{r}
ws <- ws %>%
  mutate(DA_ANTC_02_t1 = 101 - DA_ANTC_02_r_t1,
         DA_ANTC_02_t2 = 101 - DA_ANTC_02_r_t2,
         DA_ANTC_02_t3 = 101 - DA_ANTC_02_r_t3,
         DA_ANTC_02_t4 = 101 - DA_ANTC_02_r_t4,
         DA_ANTC_02_t5 = 101 - DA_ANTC_02_r_t5,
         DA_ANTC_02_t6 = 101 - DA_ANTC_02_r_t6,
         DA_ANTC_02_t7 = 101 - DA_ANTC_02_r_t7,
         DA_ANTC_02_t8 = 101 - DA_ANTC_02_r_t8,
         DA_ANTC_02_t9 = 101 - DA_ANTC_02_r_t9,
         DA_ANTC_02_t10 = 101 - DA_ANTC_02_r_t10)
```
```

internal-name-for-dataset <- internal-name-for-dataset %>%
mutate(variable_new = scale-max-plus-scale-min – variable_old)

The function is sensitive to the maximum and minimum of the scale.

Code NAs

We have values that should be NAs in our data:

DA_PSYSAFE_06_t1-10	• my unique skills were appreciated and applied in the teams.
1 = not at all - 5 = completely; 6 = cannot answer because I was not involved in a team interaction	

We have to ensure that '6' is replaced by 'NA' for all items measuring Psychological Safety.

Code NAs

To convert values to NA, we can use **na_if()**:

```
## Include NA option

```{r}
ws <- ws %>%
  mutate(across(starts_with("DA_PSYSAFE_"), ~ na_if(., 6)))
```
```

internal-name-for-dataset <- internal-name-for-dataset %>%
mutate(across(starts_with("shared-beginning-of-itemnames"),
~ na_if(., scale-value-should-be-NA)

Instead of 'starts_with', you may sometimes want to use other specifiers.

If you want to convert a given value (e.g., "-99") to NA …

1. … across all numeric variables in the dataset where(is.numeric)
2. … for specific variables c()

Aggregate Data

Each of our study variables is currently measured by multiple items at multiple times.

But, to estimate our regression model we have to comprise all of this data into one score.

How do we go about this?

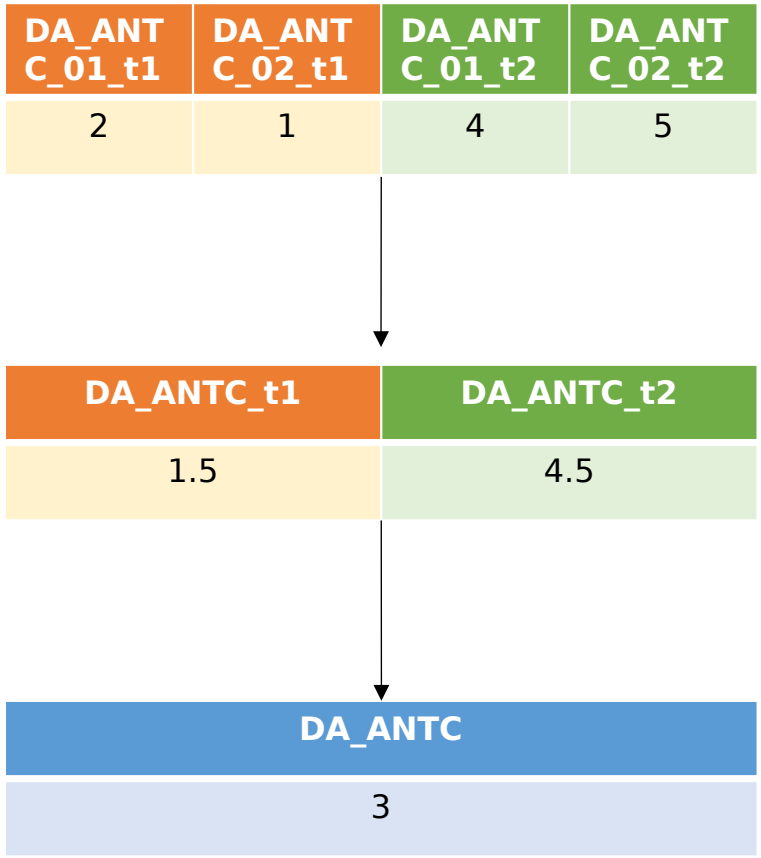
Aggregate Data

All this data …

Step 1: Aggregate mean score within occasion

Step 2: Aggregate mean score over occasions

Voilà!



Aggregate Data

To aggregate data, we can use **rowMeans()**:

````
# Aggregate Data

## Illegitimate Work Tasks

### Within occasions

#### T1

```{r}
ws <- ws %>%
  mutate(ILWOTA_t1 = rowMeans(select(ws,
            c(DA_ILWOTA_01_t1,
              DA_ILWOTA_02_t1,
              DA_ILWOTA_03_t1,
              DA_ILWOTA_04_t1)),
            na.rm = TRUE))
```
````

```
internal-name-for-dataset <- internal-name-for-dataset %>%
mutate(scale-name-aggregated =
          rowMeans(select(internal-name-for-dataset,
          c(scale-item-1,
            scale-item-X)), na.rm = TRUE))
```

Aggregate Data

You can do it!

1. Aggregate the occasion specific mean scores for "Illegitimate Work Tasks".

2. Now let's try something a bit trickier: Aggregate the occasion specific mean scores into a scale specific mean score for "Illegitimate Work Task".

Aggregate Data

Aggregating the occasion specific mean scores into a scale specific mean score.

```
### Between occasions

```{r}
ws <- ws %>%
  mutate(ILWOTA = rowMeans(select(ws,
            c(ILWOTA_t1,
              ILWOTA_t2,
              ILWOTA_t3,
              ILWOTA_t4,
              ILWOTA_t5,
              ILWOTA_t6,
              ILWOTA_t7,
              ILWOTA_t8,
              ILWOTA_t9,
              ILWOTA_t10)),
            na.rm = TRUE))
```
```

Aggregate Data

OK, OK – aggregating becomes repetitive.

You have now learned how to do it and can hopefully apply it to your own thesis.

For the workshop you can simply copy and paste the code from the documentation on Moodle.

Characterizing the sample

We have information on gender and age of participants to characterize the sample.

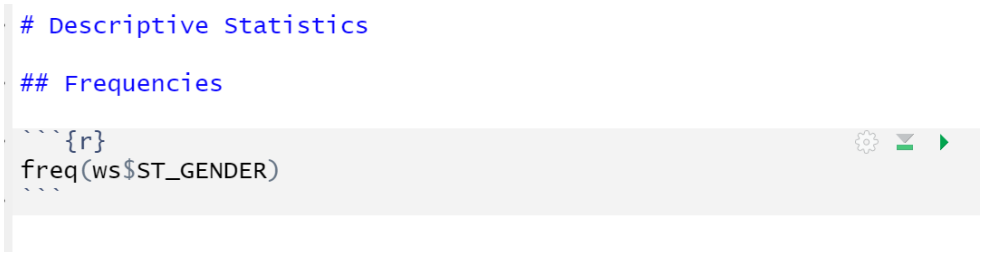
What can we do with this?

Gender → determine relative proportion of the sample per gender affiliation

Age → mean age and standard deviation

Characterizing the sample

To examine proportions, we can use **freq()**:



freq(internal-name-for-dataset$variable-of-interest)

Please note that we use the freq funtion from the 'summarytools' package.

Characterizing the sample

How we interpret the output:

Frequencies
ws$ST_GENDER
Type: Integer

	Freq	% Valid	% Valid Cum.	% Total	% Total Cum.
1	51	59.30	59.30	54.26	54.26
2	35	40.70	100.00	37.23	91.49
<NA>	8			8.51	100.00
Total	94	100.00	100.00	100.00	100.00

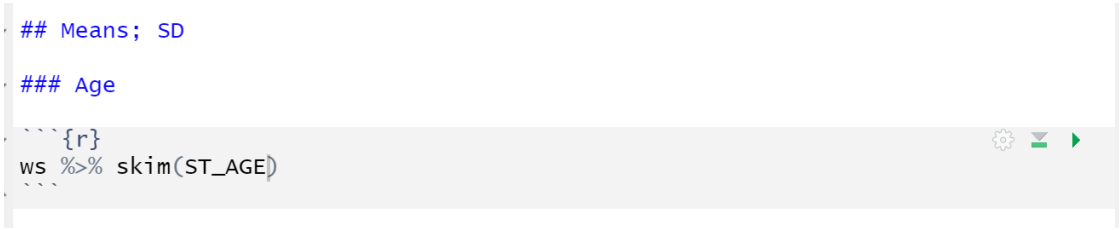
54.26 % of the sample identifies as female.

37.23 % of the sample identifies as male.

8.51 % of the sample did not respond.

Characterizing the sample

To examine means and standard deviations, we can use **skim()**:



internal-name-for-dataset %>% skim(variable-of-interest)

Characterizing the sample

How we interpret the output:

```
-- Data Summary ------------------------
                           Values
Name                       Piped data
Number of rows             94
Number of columns          175
_____________________
Column type frequency:
  numeric                  1
_____________________
Group variables            None

-- Variable type: numeric ---------------------------------------------------------------
# A tibble: 1 x 11
  skim_variable n_missing complete_rate  mean    sd    p0   p25   p50   p75  p100 hist
* <chr>             <int>         <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <chr>
1 ST_AGE                8         0.915  36.6  13.9    18    25    30  47.8    65 ▇▆▂▂▂
```

The average age of participants is 36.6 years (*SD* = 13.9).

Describing the Study Variables

To summarize descriptive information about our study variables, we have to complete a table like this:

Table 1

Means, Standard Deviations, and Correlations between Study Variables

Variable	*M*	*SD*	1	2	3	4
1. Illegitimate Work Tasks						
2. Psychological Safety						
3. Anger						
4. Strain Anticipation						

Describing the Study Variables

We already know how to examine means and standard deviations via **skim()**:

```
### Study Variables

#### Illegitimate Work Tasks

```{r}
ws %>% skim(ILWOTA)
```
```

- Compute the Mean and SD for all study variables and document in in the Descriptive Statistics Table.

Describing the Study Variables

To compute bivariate correlations, we first have to create a new object that consists only of variables that we wish to include:

```
## Bivariate Correlation Matrix

```{r}
cor_dat <- ws %>%
  select(ILWOTA,
         PSYSAFE,
         ANG,
         SA)
```
```

name-shortened-dataset-cor <- internal-name-for-dataset %>%
select(variable-of-interest1, variable-of-interest-2)

Describing the Study Variables

To compute bivariate correlations, we can now use **corr.test()**:

```r
corr.test(cor_dat)
```

corr.test(name-shortened-dataset-cor)

Describing the Study Variables

Now, we can complete the table summarizing the descriptive statistics:

Table 1

Means, Standard Deviations, and Correlations between Study Variables

Variable	*M*	*SD*	1	2	3	4
1. Illegitimate Work Tasks	1.47	0.42	-			
2. Psychological Safety	3.76	0.87	-.25	-		
3. Anger	1.72	0.69	.38	-.14	-	
4. Strain Anticipation	51.30	11.20	.38	-.13	.17	-

Hypthesis to statistical model (I)

We can address our hypotheses through the regression framework:

H1: Employees who experience more illegitimate work tasks experience more strain anticipation.

→ Simple linear regression analysis

Simple linear regression analysis

To conduct a simple linear regression analyis, we can use **lm()**:

```
## Inference Statistics

## Simple regression

```{r}
summary(lm(SA~ILWOTA, data = ws))
```
```

summary(lm(outcome~predictor, data = internal-name-for-dataset)

Simple linear regression analysis

How we interpret the output:

```
Call:
lm(formula = SA ~ ILWOTA, data = ws)

Residuals:
     Min       1Q   Median       3Q      Max
-29.4083  -5.1855  -0.5037   6.2282  25.8148

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   36.543      4.183   8.736 2.64e-13 ***
ILWOTA        10.032      2.733   3.670 0.000433 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10.47 on 81 degrees of freedom
  (11 Beobachtungen als fehlend gelöscht)
Multiple R-squared:  0.1426,    Adjusted R-squared:  0.132
F-statistic: 13.47 on 1 and 81 DF,  p-value: 0.0004326
```

There was a positive direct effect of illegitimate work tasks on strain anticipation (b=10.03, *SE* = 2.73, $p < .001$).

Simple linear regression analysis

If we want to obtain standardized regression coefficients, we first have to standardize the predictor via **scale ()**:

```
standardize predictor & dependent variable

```{r}
ws <- ws %>%
  mutate(
    ILWOTA_s = scale(ILWOTA, center = T, scale =T)
    , SA_s = scale(SA, center = T, scale =T)
    )
```
```

```
internal-name-for-dataset <- internal-name-for-dataset %>%
        mutate(
                predictor_c = scale(predictor, center =  T, scale = T)
                , outcome_c = scale(outcome, center =  T, scale = T)
        )
```

If we standardize study variables, we substract the mean and deviate by the standard deviation and therefore change their scaling.

Simple linear regression analysis

We can now obtain standardized regression coefficients by entering our standardized predictor into **lm()** :

standardized coefficients

```{r}
summary(lm(SA_s~ILWOTA_s, data = ws))
```

summary(lm(outcome_s~predictor_s, data = internal-name-dataset))

Simple linear regression analysis

Standardization changes the interpretation of our effects but not significance levels:

(I) Non-standardized regression coefficients

```
Call:
lm(formula = SA ~ ILWOTA, data = ws)

Residuals:
     Min       1Q   Median       3Q      Max
-29.4083  -5.1855  -0.5037   6.2282  25.8148

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   36.543      4.183   8.736 2.64e-13 ***
ILWOTA        10.032      2.733   3.670 0.000433 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10.47 on 81 degrees of freedom
  (11 Beobachtungen als fehlend gelöscht)
Multiple R-squared:  0.1426,    Adjusted R-squared:  0.132
F-statistic: 13.47 on 1 and 81 DF,  p-value: 0.0004326
```

When illegitimate work tasks increase by one unit, strain anticipation increases by 10.03 units on average.

(II) Standardized regression coefficients

```
Call:
lm(formula = SA_s ~ ILWOTA_s, data = ws)

Residuals:
     Min       1Q   Median       3Q      Max
-2.61711 -0.46147 -0.04482  0.55426  2.29732

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.375e-16  1.023e-01    0.00 1.000000
ILWOTA_s    3.776e-01  1.029e-01    3.67 0.000433 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9317 on 81 degrees of freedom
  (11 observations deleted due to missingness)
Multiple R-squared:  0.1426,    Adjusted R-squared:  0.132
F-statistic: 13.47 on 1 and 81 DF,  p-value: 0.0004326
```

When illegitimate work tasks increase by one standard deviation, strain anticipation increases by 0.38 standard deviations on average.

- Standardizing regression coefficients allows one to compare the importance of multiple predictors (multiple linear regression).

Hypothesis to statistical model

H2: The effect of illegitimate work task on strain anticipation is partially mediated by anger.

→ Mediation analysis via successive multiple linear regression analyis

Can be accomplished manually using lm() or via helper function from the 'psych' package.

Mediation Analysis

To conduct mediation analysis (in one step), we can use **mediate()**:

```
## Mediation

```{r}
set.seed(42)
print(mediate(SA ~ ILWOTA + (ANG), data=ws))
```
```

print(mediate(outcome ~ predictor + (mediator), data = internal-name-for-dataset))

Note: mediate() uses bootstrapping (random) – due to set.seed() the result is still always the same.

Mediation Analysis

How we interpret the output:

```
Mediation/Moderation Analysis
Call: mediate(y = SA ~ ILWOTA + (ANG), data = ws)

The DV (Y) was  SA . The IV (X) was  ILWOTA . The mediating variable(s) =  ANG .

Total effect(c) of  ILWOTA  on  SA  =  10.03   S.E. =  2.56  t  =  3.91  df=  92   with p =  0.00018
Direct effect (c') of  ILWOTA  on  SA  removing  ANG  =  9.75   S.E. =  2.79  t  =  3.49  df=  91   with p =  0.00074
Indirect effect (ab) of  ILWOTA  on  SA  through  ANG   =  0.29
Mean bootstrapped indirect effect =  0.84  with standard error =  2.1  Lower CI =  -2.66    Upper CI =  5.23
R = 0.38 R2 = 0.14   F = 7.61 on 2 and 91 DF   p-value:  0.000135
```

We did not find evidence for an indirect effect of illegitimate work tasks on strain anticipation via anger (point estimate a*b = 0.84, 95% CI = [-2.66; 5.23]).

Hypothesis to statistical model

H3: The effect of illegitimate work task on strain anticipation is attenuated by psychological safety.

→ Moderation analysis via multiple linear regression analyis

Moderation analysis

We first have to center predictor and moderator via **scale ()**:

```
## Moderation

Center variables

```{r}
ws <- ws %>%
  mutate(ILWOTA_c = scale(ILWOTA, center = TRUE, scale = FALSE),
         PSYSAFE_c = scale(PSYSAFE, center = TRUE, scale = FALSE))
```
```

internal-name-for-dataset <- internal-name-for-dataset %>%
mutate(predictor_c = scale(predictor, center = T, scale = F),
moderator_c = scale(moderator, center = T, scale = F))

- **Centering** predictors and continuous moderators (aka substracting the mean) is **mandatory** to ensure meaningful interpretation of the direct effects.
- **Standardizing** is **voluntary**.

Moderation analysis

To conduct moderation analysis, we can now use **lm ()**:

```
Compute moderation model

```{r}
summary(lm(SA~ILWOTA_c*PSYSAFE_c, data = ws))
```
```

summary(lm(outcome~predictor_c*moderator_c, data = internal-name-for-dataset)

Moderation analysis

How we interpret the output:

```
Call:
lm(formula = SA ~ ILWOTA_c * PSYSAFE_c, data = ws)

Residuals:
     Min       1Q   Median       3Q      Max
-29.5835  -5.1516  -0.6856   6.3027  26.2848

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       51.29651    1.19275  43.007  < 2e-16 ***
ILWOTA_c           9.73199    3.07678   3.163  0.00222 **
PSYSAFE_c         -0.50720    1.41137  -0.359  0.72028
ILWOTA_c:PSYSAFE_c -0.09542   2.86678  -0.033  0.97353
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10.59 on 79 degrees of freedom
  (11 Beobachtungen als fehlend gelöscht)
Multiple R-squared:  0.144,     Adjusted R-squared:  0.1115
F-statistic:  4.43 on 3 and 79 DF,  p-value: 0.006245
```

The data did not support a moderation effect of psychological safety on the relationship between illegitimate work tasks and strain anticipation (b = -0.10, *SE* = 2.87, *p* > .05).

Moderation analysis

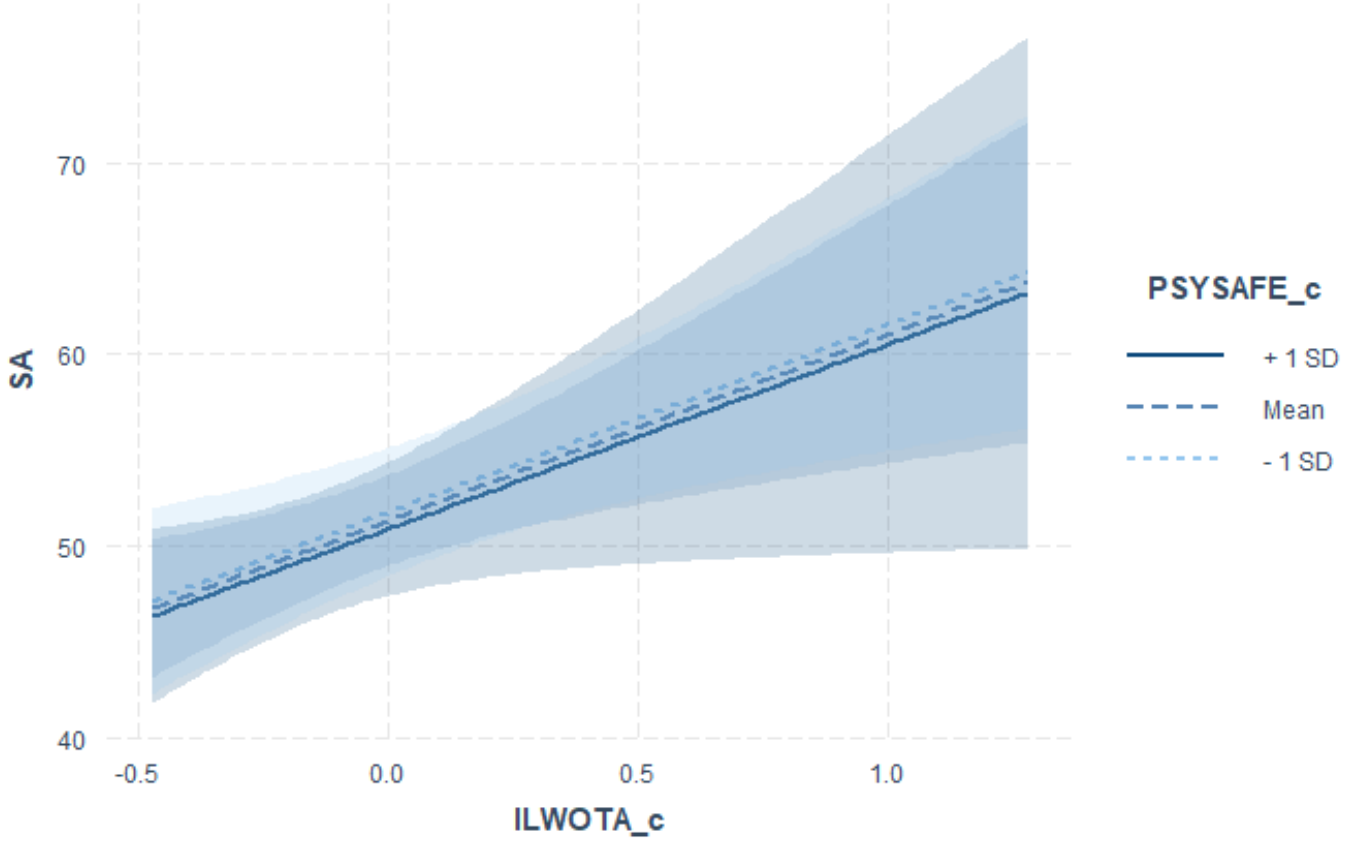
To visualize the moderation effect, we can use **interact_plot()**:

```
Visualize moderation effect

```{r}
interact_plot(
  lm(SA ~ ILWOTA_c * PSYSAFE_c, data = ws),
  pred = ILWOTA_c,
  modx = PSYSAFE_c,
  interval = T,
)
```
```

interact_plot(
lm(outcome~predictor_c*moderator_c, data = internal-name-for-dataset),
pred = predictor_c,
modx = moderator_c,
interval = T)

Moderation analysis



Sidenote: We visualized the effect for demonstartion purposes. If you have a non significant moderation effect it is likely not necessary to visualize it.

Moderation analysis

Moderation can also be understood as a conditional effect. Evaluate with **Johnson-Neyman intervals**:

Johnson-Neyman intervals

```
```{r}
johnson_neyman(
  lm(SA ~ ILWOTA_c * PSYSAFE_c, data = ws)
  , pred = ILWOTA_c
  , modx = PSYSAFE_c
  , control.fdr=T)
```
```

johnson_newman(

lm(outcome~predictor_c*moderator_c, data = internal-name-for-dataset),

pred = predictor_c,

modx = moderator_c,

control.fdr = T)

Moderation analysis


```
JOHNSON-NEYMAN INTERVAL

When PSYSAFE_c is INSIDE the interval [-1.58, 0.75], the slope of ILWOTA_c is p < .05.

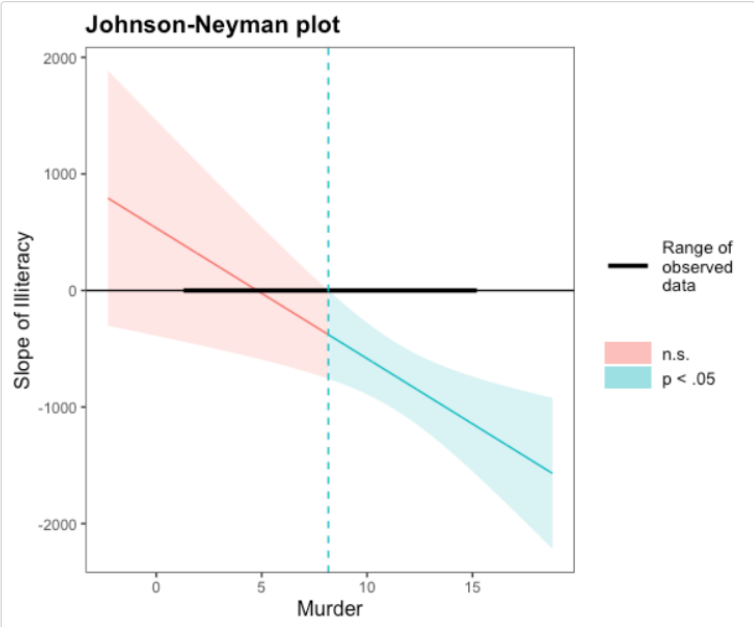
Note: The range of observed values of PSYSAFE_c is [-2.76, 1.24]

Interval calculated using false discovery rate adjusted t = 2.22
```

In our case the slope of ILWOTA is not conditional on PYSAFE.

But it could also look like this example:

```
## JOHNSON-NEYMAN INTERVAL
##
## When Murder is OUTSIDE the interval [-10.61, 8.16], the slope of Illiteracy
## is p < .05.
##
## Note: The range of observed values of Murder is [1.40, 15.10]
```



Resources

Document your environment

```
```{r}
sessionInfo()
```
```

```
R version 4.3.2 (2023-10-31)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: Ubuntu 22.04.3 LTS

Matrix products: default
BLAS:   /usr/lib/x86_64-linux-gnu/blas/libblas.so.3.10.0
LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.10.0

locale:
 [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C               LC_TIME=de_AT.UTF-8        LC_COLLATE=en_US.UTF-8
 [5] LC_MONETARY=de_AT.UTF-8    LC_MESSAGES=en_US.UTF-8    LC_PAPER=de_AT.UTF-8       LC_NAME=C
 [9] LC_ADDRESS=C               LC_TELEPHONE=C             LC_MEASUREMENT=de_AT.UTF-8 LC_IDENTIFICATION=C

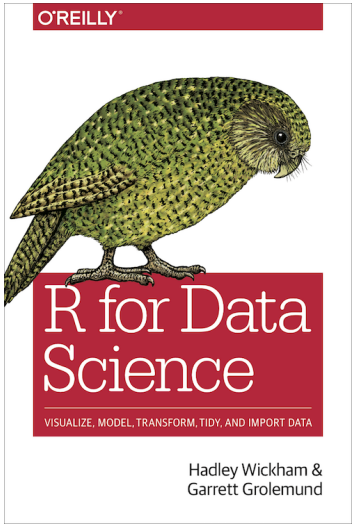
time zone: Europe/Vienna
tzcode source: system (glibc)

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
 [1] summarytools_1.0.1 interactions_1.1.5 psych_2.3.6        skimr_2.1.5        lubridate_1.9.2    forcats_1.0.0      stringr_1.5.0
 [8] dplyr_1.1.2        purrr_1.0.2        readr_2.1.4        tidyr_1.3.0        tibble_3.2.1       ggplot2_3.4.3      tidyverse_2.0.0
[15] installr_0.23.4    install.load_1.2.5

loaded via a namespace (and not attached):
 [1] fastmatch_1.1-4   gtable_0.3.4      xfun_0.40         lattice_0.22-5    tzdb_0.4.0        vctrs_0.6.3       tools_4.3.2
 [8] generics_0.1.3    parallel_4.3.2    fansi_1.0.4       pkgconfig_2.0.3   checkmate_2.2.0   pryr_0.1.6        assertthat_0.2.1
[15] lifecycle_1.0.3   farver_2.1.1      compiler_4.3.2    rapportools_1.1   munsell_0.5.0     mnormt_2.1.1      repr_1.1.6
[22] codetools_0.2-19  htmltools_0.5.6   yaml_2.3.7        pillar_1.9.0      crayon_1.5.2      MASS_7.3-60       magick_2.8.2
[29] nlme_3.1-163      tidyselect_1.2.0  digest_0.6.33     stringi_1.7.12    reshape2_1.4.4    pander_0.6.5      labeling_0.4.3
[36] fastmap_1.1.1     grid_4.3.2        colorspace_2.1-0  cli_3.6.1         magrittr_2.0.3    base64enc_0.1-3   utf8_1.2.3
[43] withr_2.5.0       scales_1.3.0      jtools_2.2.2      backports_1.4.1   timechange_0.2.0  rmarkdown_2.24    matrixStats_1.2.0
[50] hms_1.1.3         evaluate_0.21     knitr_1.43        tcltk_4.3.2       rlang_1.1.1       Rcpp_1.0.11       glue_1.6.2
[57] pkgload_1.3.2.1   rstudioapi_0.15.0 jsonlite_1.8.7    plyr_1.8.8        R6_2.5.1
```

Resources



Programming in R:

https://r4ds.had.co.nz/

Functions and other concerns:

- check documentation of function / package
- ask the community (e.g. via stackoverflow)

Best Practices

Reproducibility:

- use Rprojects, not setwd()
- use RMarkdown
- never load packages or install datasets through the graphical user interface without documenting executed functions in script
- object changes → new object name (we didn't do that)
- use seeds for random parts (e.g. bootstrapping)
- document your environment (be a Pro: use Docker for full reproducability)

Best Practices

Readability:

RMarkdown → use headings sub-headings to clarify structure

- structure: packages, data, tidying, analyses
- only keep code that is necessary and works to ensure specificity
- comment throughout

UNI FR

References

Cranford, J. A., Shrout, P. E., Iida, M., Rafaeli, E., Yip, T., & Bolger, N. (2006). A procedure for evaluating sensitivity to within-person change: Can mood measures in diary studies detect change reliably?. Personality and Social Psychology Bulletin, 32(7), 917-929.

Scott, S.B., Graham-Engeland, J.E., Engeland, C.G. et al. (2015). The Effects of Stress on Cognitive Aging, Physiology and Emotion (ESCAPE) Project. BMC Psychiatry 15(146). doi: 10.1186/s12888-015-0497-7

Semmer, N. K., Tschan, F., Meier, L. L., Facchin, S., & Jacobshagen, N. (2010). Illegitimate tasks and counterproductive work behavior. Applied Psychology: An International Review, 59(1), 70– 96. doi:10.1111/j.1464-0597.2009.00416.x