Survey Data: Coding & Validation

Miriam Sorace

18/11/2022

Initial SetUp

1. Global Settings

2. Set Working Directory

3. Install Required Packages & Call Relevant Libraries:

```
if (!("tidyverse" %in% installed.packages())) {
    install.packages("tidyverse")
}
library(tidyverse)

if (!("dplyr" %in% installed.packages())) {
    install.packages("dplyr")
}
library(dplyr)

if (!("janitor" %in% installed.packages())) {
    install.packages("janitor")
}
library(janitor)

if (!("ggplot2" %in% installed.packages())) {
    install.packages("ggplot2")
}
library(ggplot2)


if (!("scales" %in% installed.packages())) {
    install.packages("scales")
}
library(scales)

if (!("broom" %in% installed.packages())) {
    install.packages("broom")
}
library(broom)
```

```
if (!("stargazer" %in% installed.packages())) {
    install.packages("stargazer")
}
library(stargazer)
```

Cognitive Interviewing

4. Load .csv of completed responses from Google Forms

Something to note: when downloading from Google Forms the variable name is the entire survey item, which can get unwieldy. Just easier to rename directly on the excel file and then upload on to R. Or you can always change the variable names with the line of code provided below, but the excel workaround will usually save you time (unless it is a long survey you are working with).

```
MyCognInt <- read.csv("CognInterview.csv")
```

5. Coding/Data Cleaning

```
# generate ID variable
MyCognInt <- tibble::rowid_to_column(MyCognInt, "Identification No.")

# renaming variables, new name first, old name in-between quotes.
MyCognInt <- dplyr::rename(MyCognInt, ID = "Identification No.")

# dropping variables that are not needed
MyCognInt <- dplyr::select(MyCognInt, -Timestamp)


# re-coding empty cells and/or NAs **in the entire dataset**

MyCognInt[MyCognInt == "" | MyCognInt == " "] <- NA
MyCognInt[MyCognInt == "N/A" | MyCognInt == "n/a" | MyCognInt == "N/a" |
    MyCognInt == "-"] <- NA


# use class() and table() on the variables to see how they are stored
# in R

class(MyCognInt$QA)
```

```
## [1] "character"
```

```
table(MyCognInt$QA)
```

```
##
##             Agree          Disagree        Don't Know           Neutral
##                 9                 1                 1                 6
##    Strongly Agree Strongly Disagree
##                 1                 2
```

```
class(MyCognInt$QB)
```

```
## [1] "character"
```

```
table(MyCognInt$QB)
```

```
## 
##          Agree        Neutral Strongly Agree 
##             12              6             2 
```

```
# Google Forms converts multiple choice to character: recode
# string/text/character variables, transform them to factor or
# numeric

simplify.likert <- function(x) {
    case_when(x == "Strongly Agree" ~ 4, x == "Agree" ~ 3, x == "Neutral" ~
        2, x == "Disagree" ~ 1, x == "Strongly Disagree" ~ 0, TRUE ~ NA_real_  #sets the rest (Don't Kn
) %>%
        factor(levels = c(0, 1, 2, 3, 4), labels = c("Strongly Disagree",
            "Disagree", "Neutral", "Agree", "Strongly Agree"))
}


# apply recoding function to the relevant variable.  create a new
# variable name, good to retain the old variable to check if the
# re-code functioned correctly!
MyCognInt <- MyCognInt %>%
    mutate(QA_rec = simplify.likert(QA))

# check whether the recode has worked
table(MyCognInt$QA, MyCognInt$QA_rec)
```

```
## 
##                      Strongly Disagree Disagree Neutral Agree Strongly Agree
##   Agree                              0        0       0     9              0
##   Disagree                           0        1       0     0              0
##   Don’t Know                         0        0       0     0              0
##   Neutral                            0        0       6     0              0
##   Strongly Agree                     0        0       0     0              1
##   Strongly Disagree                  2        0       0     0              0
```

```
# do the same for QB (also an agree-disagree Likert scale in my case)

MyCognInt <- MyCognInt %>%
    mutate(QB_rec = simplify.likert(QB))

# check whether the recode has worked
table(MyCognInt$QB, MyCognInt$QB_rec)
```

```
## 
##                   Strongly Disagree Disagree Neutral Agree Strongly Agree
##   Agree                          0        0       0    12              0
##   Neutral                        0        0       6     0              0
##   Strongly Agree                 0        0       0     0              2
```

```
# Difficulty questions are numeric but have no labels. To add value
# labels:

MyCognInt$QADiff <- ordered(MyCognInt$QA_CIprompt3, levels = c(1, 2, 3,
    4, 5), labels = c("Not hard at all", "Somewhat hard", "Quite Hard",
    "Hard", "Very Hard"))

MyCognInt$QBDiff <- ordered(MyCognInt$QB_CIprompt3, levels = c(1, 2, 3,
    4, 5), labels = c("Not hard at all", "Somewhat hard", "Quite Hard",
    "Hard", "Very Hard"))

# check recodes

table(MyCognInt$QA_CIprompt3, MyCognInt$QADiff)
```

```
##
##     Not hard at all Somewhat hard Quite Hard Hard Very Hard
##   1              10             0          0    0         0
##   2               0             6          0    0         0
##   3               0             0          2    0         0
##   4               0             0          0    2         0
```

```
table(MyCognInt$QB_CIprompt3, MyCognInt$QBDiff)
```

```
##
##     Not hard at all Somewhat hard Quite Hard Hard Very Hard
##   1               6             0          0    0         0
##   2               0             7          0    0         0
##   3               0             0          4    0         0
##   4               0             0          0    3         0
```

6. Validation: Answer Behavior in the Cognitive Probes

How often people chose the 'DK' option? How often did they say that the question was too hard to answer (if analysing the cognitive interview probes)? Some simple descriptive tables/plots can help in getting a sense of how people answer the specific survey question. Providing some descriptive graphs is therefore a good first step.

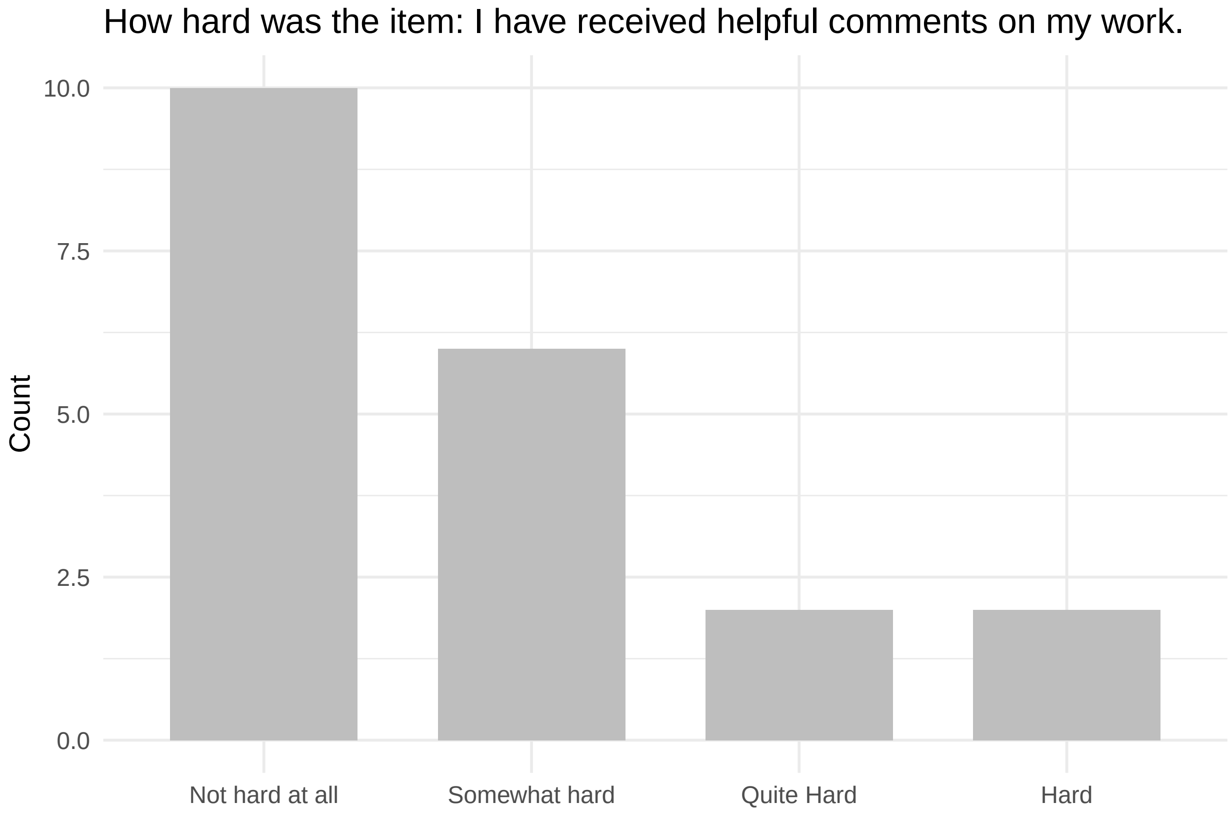
```
# to get number of don't knows/refused in the survey items under
# investigation (QA and QB in this example) run the summary function
# on the recoded variables, which will show how many NAs (if any are
# present)

summary(MyCognInt$QA_rec)
```

```
## Strongly Disagree          Disagree           Neutral             Agree
##                 2                 1                 6                 9
##    Strongly Agree              NA's
##                 1                 1
```

```
summary(MyCognInt$QB_rec)
```

```
## Strongly Disagree          Disagree           Neutral             Agree
##                 0                 0                 6                12
##    Strongly Agree
##                 2
```

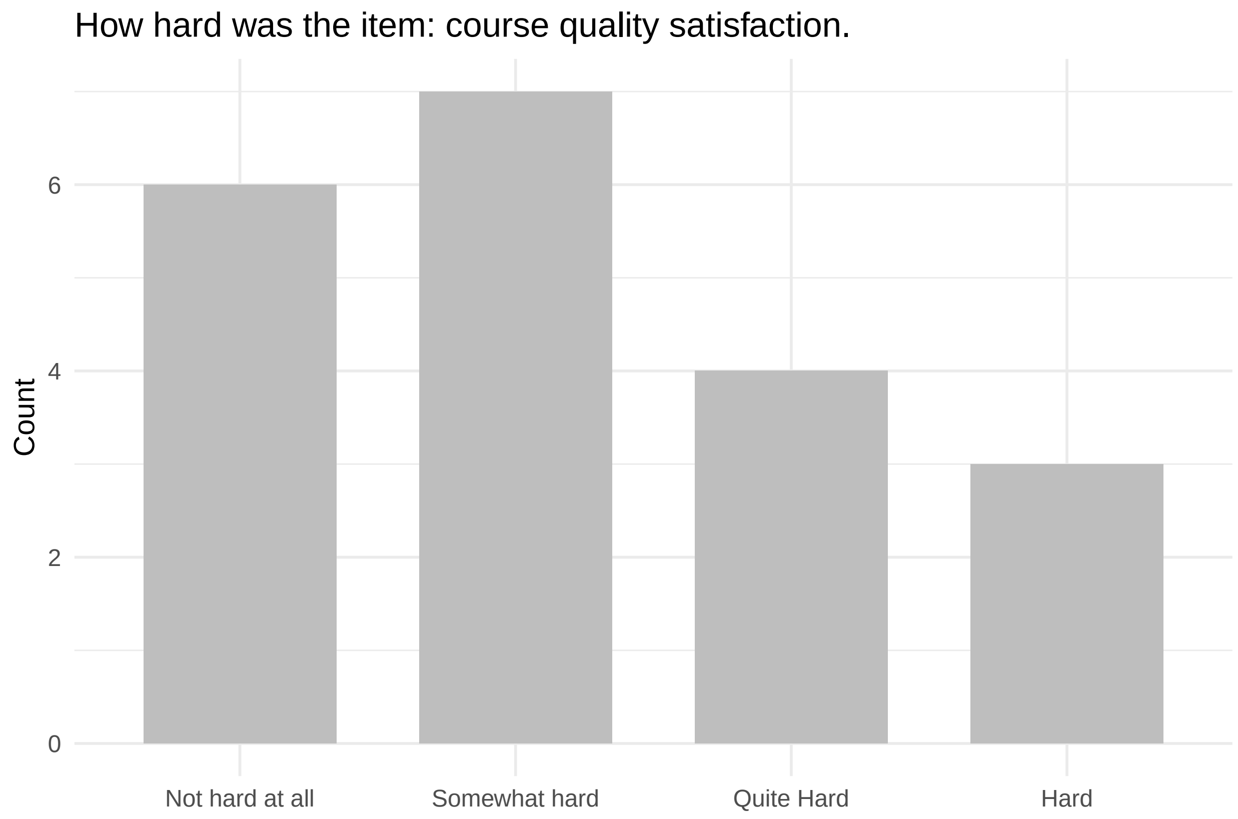
```
# to assess how hard each question was perceived (cognitive interview
# prompt no.3), it might be nice to create a histogram or a bar chart
# of the answers

graphQADiff <- ggplot(data = MyCognInt, aes(x = QADiff)) + geom_bar(stat = "count",
    width = 0.7, fill = "gray") + theme_minimal() + labs(title = "How hard was the item: I have receivec
    x = " ", y = "Count")

graphQADiff
```



```
summary(as.numeric(MyCognInt$QADiff))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     1.0     1.0     1.5     1.8     2.0     4.0
```

```
graphQBDiff <- ggplot(data = MyCognInt, aes(x = QBDiff)) + geom_bar(stat = "count",
    width = 0.7, fill = "gray") + theme_minimal() + labs(title = "How hard was the item: course quality
    x = " ", y = "Count")

graphQBDiff
```



```
summary(as.numeric(MyCognInt$QBDiff))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##     1.0     1.0     2.0     2.2     3.0     4.0 
```

7. Cognitive Interviewing: Probing Comprehension and Recall

Cognitive interviewing by its nature has to be open-ended. To analyse answers to the open probes, content analysis of the answers' texts is required. We're going to see automated text analysis methods next term, at this stage and for the first report, you are supposed to do this more qualitatively.

Go through each text from your files and try to systematize the answers into categories. When I have done this for the cognitive interview you filled out in class, the following patterns emerged.

Helpful Feedback NSS Question

Comprehension The main themes that emerged when students were asked to report what 'receiving helpful comments' means to them are *usable, practical advice on how to do the work better* (mentioned in

18 out of 20 answers), *praising comments* (mentioned in 5 out of 20 answers), and *clearly highlighting the mistakes* (mentioned in 5 out of 20 answers). Respondents seemed to understand helpful as re-usable, and many mentioned future-orientation in their answers.

One respondent made this point clearly when complaining about 'comments that are too specific to the individual piece of work and don't have much value going forward'. Another wrote: 'In all honesty, the majority of this feedback has been pretty useless to me as I am receiving it AFTER I have finished a module, where those particular skills were useful. Yes, there are some aspects which spread across different modules and assessments, but specific feedback rarely addressed this. Usual feedback was concerned with the content of a particular essay. I'm not going to rewrite the essay. I'm likely not going to look into the topic again in much detail during my degree. Therefore, it is hard to call most of the feedback useful.'

From this initial qualitative overview, 'helpful comments' means: general/multi-purpose/'future-proof' applied pieces of advice that simply signposts what was done well together with mistakes and that can be used in future work to achieve better grades in other modules/assessments.

Retrieval Most respondents clearly outlined processes of recollection of past experiences (comparison of previous helpful and unhelpful feedback, verbal conversations with lecturers and academic advisors). Some mentioned that the word 'work' drove them to also include work experiences, rather than simply University assignments. This highlights a potential wording issue in the NSS.

Recommendations

On the basis of the results from the cognitive interviewing on the survey item on helpfulness of the feedback, it appears that the NSS item works pretty well. It is not perceived as difficult to answer, and the memories recollected are in line with what would be expected. Also, the near consensus on the meaning of 'helpful feedback' is a suggestion that the item is generally understood in the same way by different respondents - hence having good reliability.

Satisfaction with Course Quality NSS Question

Comprehension The main themes that emerged when students were asked to report what 'University course quality' means to them are: (1) *content excellence*, including demonstrating research quality, benefiting from extensive expertise and applied knowledge of academic staff, and the amount of information disseminated (mentioned in 11 out of 20 responses); (2) *enjoable and passionate lecturers* (mentioned in 5 out of 20 responses); (3) *level of student support/responsivenes and approachability* (both academic and related to mental health - responses in this area often mention the availability of good student support teams), mentioned in 4 out of 20 responses; (4) *feasible/realistic workloads* mentioned in 3 out of 20 responses; (5) *utilitarian considerations*, including value for money and career-progression (mentioned in 5 out of 20 responses).

Retrieval Most respondents clearly outlined processes of recollection of relevant past experiences (in particular, teaching delivery, teaching contents, clarity of module outlines and learning outcomes, feedback and grades). Some mentioned that retrieval also included estimations of the ease vs. difficulty of the content and how much they feel they have learnt as a result of the degree.

Recommendation There is some agreement on the meaning of this question, with the content expertise and excellence of the academic staff theme being mentioned by more than half respondents. However, it is clear that students understand course quality in different ways and according to very specific priorities. This item was also scored as harder to answer. Clearly this NSS survey item needs some unpacking, and maybe consider specifying what is meant by quality or do a break-down for the various elements that make up a degree course of good quality (research excellence/applied teaching; inspirational teaching and feedback; student support and student services).

Validation

8. Correlational Validity: Simple Visualisations

You will do this step as part of your final survey, not as part of your cognitive interviewing process. Correlational validity is fielding - in addition to your bespoke survey item - either a gold standard question or a survey item that should be a *theoretical correlate* (based on previous studies) of the phenomenon your bespoke survey item is supposed to measure.

Let's load therefore a different dataset - containing the results from a made-up survey on political knowledge.

```r
FinalSurvey <- read.csv("survey.csv")
```

The dataset I have named 'Final Survey' contains 2 variables: PolKnow and PolInt. PolKnow contains a scale trying to capture political knowledge of the respondent, 10 meaning 'most knowledgeable' and 0 meaning 'least knowledgeable'. This is a new survey item that was fielded, and the researcher needs to now validate it against a 'theoretical correlate'.

The theoretical correlate chosen is PolInt which is a scale capturing how interested the respondent is in politics. 10 means they are very interested, 0 that they are not at all interested. Theoretically, people with high political interest should absorb more political information by virtue of superior exposure.

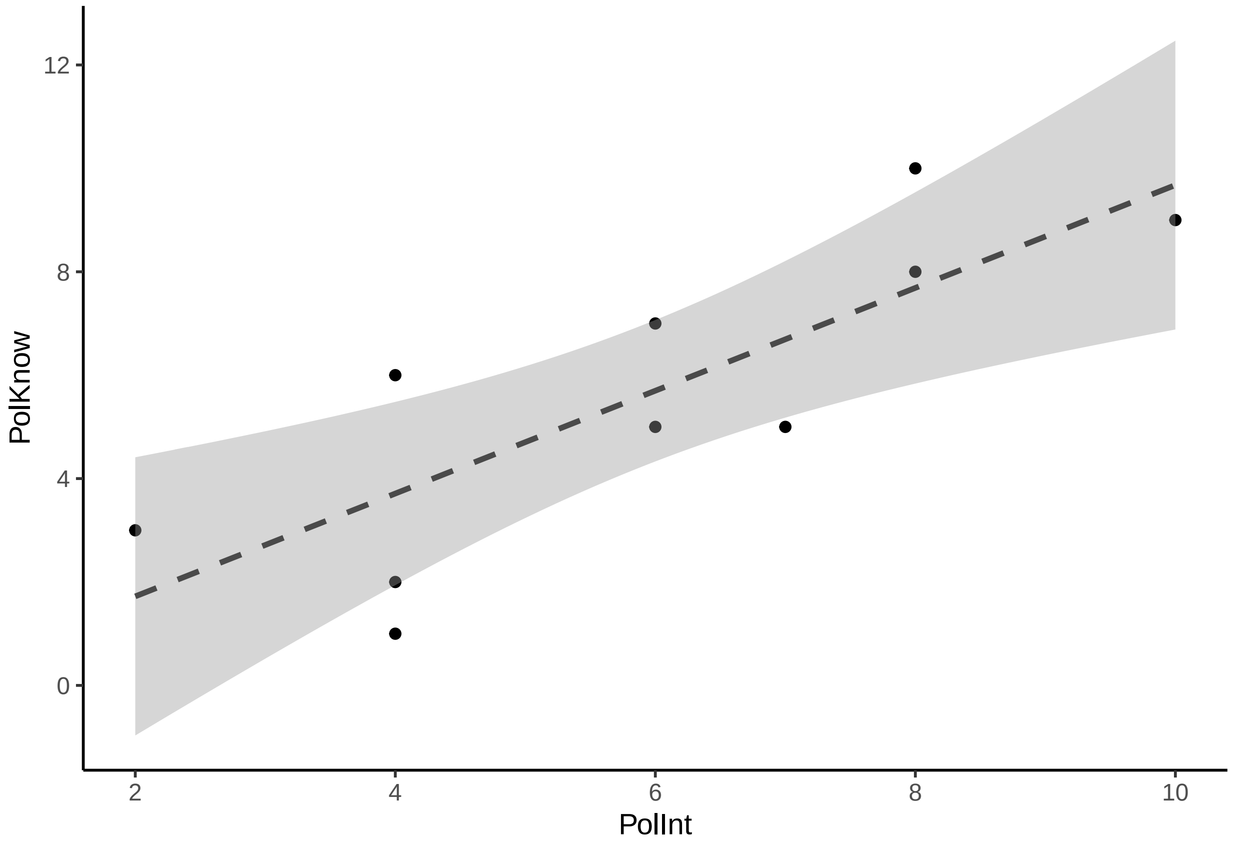
So, if the (made-up) researcher's new (made-up) survey item is validly capturing political knowledge, it should be positively associated with political interest.

We can explore the association between the two variables first descriptively, via cross-tabulations and/or via scatterplots. See relevant lines of code below:

```r
#Cross-Tabulation (better for categorical variables)

table1 <- FinalSurvey %>%
  tabyl(PolKnow, #new survey item in rows,
        PolInt #theoretical predictor in columns
        ) %>%
  adorn_totals("col") %>%
  adorn_percentages("col") %>%
  adorn_pct_formatting(rounding = "half up", digits = 0) %>%
  adorn_ns() %>%
  knitr::kable()

table1
```

PolKnow	10	2	4	6	7	8	Total
1	0% (0)	0% (0)	33% (1)	0% (0)	0% (0)	0% (0)	10% (1)
2	0% (0)	0% (0)	33% (1)	0% (0)	0% (0)	0% (0)	10% (1)
3	0% (0)	100% (1)	0% (0)	0% (0)	0% (0)	0% (0)	10% (1)
5	0% (0)	0% (0)	0% (0)	50% (1)	100% (1)	0% (0)	20% (2)
6	0% (0)	0% (0)	33% (1)	0% (0)	0% (0)	0% (0)	10% (1)
7	0% (0)	0% (0)	0% (0)	50% (1)	0% (0)	0% (0)	10% (1)
8	0% (0)	0% (0)	0% (0)	0% (0)	0% (0)	50% (1)	10% (1)
9	100% (1)	0% (0)	0% (0)	0% (0)	0% (0)	0% (0)	10% (1)
10	0% (0)	0% (0)	0% (0)	0% (0)	0% (0)	50% (1)	10% (1)

```r
#Scatterplot (better for continuous variables
#- can't use ordered/factor variables: use numerical!
#If your variables are not numeric you can just convert them using
#as.numeric()

#place theoretical predictor in x and new survey item in y

ggplot(FinalSurvey, aes(x=PolInt, y=PolKnow)) +
  geom_point(color="black") +
  geom_smooth(method=lm, linetype="dashed",
              color="gray29") +
  theme_classic()
```



If we look at the table, when Political Interest has higher values, Political Knowledge also has higher values. The scatterplot also clearly shows a positive association: the more the respondents report political interest, the more knowledgeable they are.

9. Correlational Validity via Formal Hypothesis Testing

The above was just a way of 'eyeballing' the relationship. For a formal evaluation of the relationship, we can use regression analysis. Regression analysis is a statistical technique that estimates the strength and statistical significance of the association between two (or more) variables. For more information, see: https://stats.idre.ucla.edu/wp-content/uploads/2021/05/R_reg_part1.html#(1)

We won't go into the details here, it will just be sufficient to know that the coefficient next to the second variable is the amount of change in the first variable (the dependent variable - i.e. our new survey item that we are trying to validate) for a 1-unit change in the explanatory (or theoretical predictor) variable. The second thing to know is that if the t-value is above 2, the relationship is statistically significantly different from zero - meaning that an effect of zero (absence of a relationship) is statistically unlikely given the data observed. Whenever the t-value is above 2, the regression table will display stars.

```
# remember: you need to add ,results='asis' for stargazer to output a
# table in your knitted Rmd

model1 <- lm(PolKnow ~ PolInt, data = FinalSurvey)

summary(model1)
```

Call: lm(formula = PolKnow ~ PolInt, data = FinalSurvey)

Residuals: Min 1Q Median 3Q Max -2.7108 -1.4452 -0.1824 1.2949 2.3119

Coefficients: Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.2665 1.6317 -0.163 0.87429
PolInt 0.9943 0.2577 3.859 0.00482 ** — Signif. codes: 0 ‘’ ***0.001*** ’’ *0.01* ” 0.05 ‘.’ 0.1 ’ ’ 1

Residual standard error: 1.874 on 8 degrees of freedom Multiple R-squared: 0.6505, Adjusted R-squared: 0.6068 F-statistic: 14.89 on 1 and 8 DF, p-value: 0.004816

```
# getting a nicer-looking table

stargazer(model1)
```

% Table created by stargazer v.5.2.3 by Marek Hlavac, Social Policy Institute. E-mail: marek.hlavac at gmail.com % Date and time: Wed, Nov 23, 2022 - 13:51:32

Table 2:

	Dependent variable:
	PolKnow
PolInt	0.994^{***}
	(0.258)
Constant	-0.267
	(1.632)
Observations	10
R^2	0.651
Adjusted R^2	0.607
Residual Std. Error	1.874 (df = 8)
F Statistic	14.891^{***} (df = 1; 8)
Note:	*p<0.1; **p<0.05; ***p<0.01

We find that the slope (beta) coefficient for Political Interest is ~ 0.99 and statistically significant at the 0.001 level. This means that the relationship is statistically significantly different from zero. The value of 0.99 means that for a unit increase in political interest there is nearly a unit (0.99 of a unit) increase in political knowledge! Given that the two items are measured using the same scale (disagree-agree in 5-points Likert scales) this is nearly a 1:1 relationship, so quite strong. This is really the best case scenario for validation (and highly unlikely! Just a feature of my fake data :))

REMEMBER: if the two variables are not measured using the same scale, you won't be able to interpret the magnitude of the effect well like in this case. You'll need to standardise your variables first in order to do compare the two variables fully (see below):

```r
# extra code: in case you'll need to correlate two variables measured
# with different scales

modelStd <- lm(scale(PolKnow) ~ scale(PolInt), data = FinalSurvey)

# getting a nicer-looking table

stargazer(modelStd)
```

% Table created by stargazer v.5.2.3 by Marek Hlavac, Social Policy Institute. E-mail: marek.hlavac at gmail.com % Date and time: Wed, Nov 23, 2022 - 13:51:32

Table 3:

	Dependent variable:
	scale(PolKnow)
scale(PolInt)	0.807***
	(0.209)
Constant	0.000
	(0.198)
Observations	10
R^2	0.651
Adjusted R^2	0.607
Residual Std. Error	0.627 (df = 8)
F Statistic	14.891*** (df = 1; 8)
Note:	*p<0.1; **p<0.05; ***p<0.01

This will return the coefficient expressed in standard deviations change: for a standard deviation increase in X, you get [beta coefficient value] standard deviation increase in Y.