Worksheet 4

Logistic regression in R

Timetable week: 7  
Topic: "Logistic models"

# Learning outcomes

By the end of the session you will know how to:

* Fit and summarise logistic regression models in R
* Interpret results from logistic regression models
* Manipulate the regression output to ease interpretation
* Plot and visualise results from logistic regression models to aid interpretation

# Introduction

This worksheet introduces a new type of regression model: *(binary) logistic regression*.

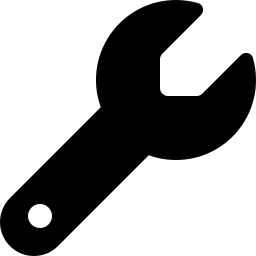
The logistic regression model attempts to **predict the *probability* that an observation falls into one of two categories of a dichotomous dependent (outcome) variable** based on one or more independent (*predictor*, *explanatory*) variables that can be either numeric or categorical (“factors” in R).

In many ways, logistic regression is like linear regression, except for the measurement type of the dependent variable (i.e., linear regression uses a numeric dependent variable rather than a dichotomous one). However, unlike in linear regression, you are not attempting to determine **the predicted value** of the dependent variable, **but the *probability* of being in a particular category** of the dependent variable given the independent variable(s). An observation is assigned to whichever category is predicted as most likely. Logistic modelling is thus often considered a *classification* method rather than a *regression* method, especially in recent *machine learning* parlance.

Just as in linear regression analysis, logistic regression gives us a set of **coefficients** that **show the effect of x on y**. However, because logistic regression is based on other assumptions than linear regression, we cannot interpret these coefficients very easily. They do not represent the expected value of given a one-unit difference in the value of , but instead the ‘**log of the odds**’ of belonging in the outcome category coded as *1* as opposed to *0* (e.g. answering “yes” to a survey question if the variable is coded as “yes” = 1 and “no” = 0).

# Exercise 0: Setup

1. **Open the R Studio interface** by clicking on the ***.Rproj*** file included in the project folder that you created in *Lab2*. The folder should be stored on your Newcastle University **OneDrive** and accessible from any computer.
2. **Create a new blank Quarto document for this lab session and call it *Lab4.qmd***
3. At the top of the page briefly detail what the script is about (e.g. “Lab work for Week x”).
4. **Load R packages** that we will use with the library() function

*  **Tip:** You may need to first install the package with the install.packages() function if it’s not yet installed
* library(tidyverse)  
  library(easystats)  
  library(gtsummary)  
  library(ggformula)  
  library(strengejacke)  
  library(marginaleffects)

# Exercise 1: How does education relate to social trust?

About 30 minutes

To demonstrate logistic regression using R, in this exercise we follow in the footsteps of (Wu 2021) (see reading list). As they note, sociological research has generally concluded that “there is a strong and positive relation between education level and trust”. However, “several studies have shown that education might yield differential impacts on trust in different societies. In Sweden, Sven Oskarsson et al. (2017) show that education has little impact on trust. In China, Cary Wu and Zhilei Shi (2020) suggest that education has a negative impact on people’s trust. Several cross-national studies have also shown that the education and trust association can vary from positive to negative depending on the specific institutional contexts … For example, the effect of education is found to be negative in highly corrupt countries such as Serbia, Turkey, Hungary, Slovakia, Bulgaria, Croatia, Kosovo, and Ukraine” (p. 1167).

The article then investigates how individual-level (micro) factors (such as *education level*) may have different effects on *trust* depending on country-level (macro) factors (such as *sociopolitical risk*). We will focus on the individual (micro) level in this exercise.

Specifically, we will build a model similar to those summarised in Table 1 of the article. We will initially disregard the fact that our dataset contains data from various countries, but in a follow-up exercise we will select one country from the dataset to refit the model on that smaller sample and compare our results to those reported in Wu’s Table 1.

## Step 0: Research question and data

Let’s ask the following research question: what is the association between *education* and *social trust* at a global level?

To answer this question, we will use the **WVS7** dataset. The dataset should already be downloaded and available in your module data folder. Let’s import the data into R, but first, it’s worth highlighting some options that need bearing in mind when importing labelled data of the kind that comes from social surveys:

**It’s important** to note that by default the data\_read() function assumes that numeric variables where all values have a value label are categorical and will convert them into *factors*. That means that all the numeric “values” will be replaced by the “value labels” (e.g. 1=“Yes”/ 2=“No” will become simply “Yes”/“No”). This is usually a reasonable default behaviour, because the majority of functions in R do not handle “value labels” and working with textual (character string) values can be more explicit. However, this may be less appropriate when the dataset contains many long ordered variables (such as 0-10 scale items), as we will most likely want to treat such variables as *numeric* in statistical models.

To cancel this default behaviour, we can add the additional argument convert\_factors = FALSE into the data\_read() function. This is the format of the variables as listed in the Codebook on the module website (https://soc2069.chrismoreh.com/data/data\_main), with both “Values” and “Value labels” kept as in the original survey questionnaire documentation. This is also the format that gets imported when using readRDS(). However, most of the common tabulation and graphing functions will not show the category “labels” in the output either, and for that purpose having variables “converted to factors” (with the original “Values” overwritten by the “Value labels”) may be a better option.

Below we will import the data ***without*** converting to factors.

# Change the file path to the one on your computer!  
  
wvs <- data\_read("D:/GitHub/SOC2069/Data/for\_analysis/wvs7.rds", convert\_factors = FALSE)  
  
# or #  
  
wvs <- readRDS("D:/GitHub/SOC2069/Data/for\_analysis/wvs7.rds")

## Step 1: Find and describe individual variables

The next step is to identify the variables that are most useful for answering the research question we posed. We can do this by browsing through the WVS7 codebook (https://soc2069.chrismoreh.com/data/data\_main#wvs7-codebook). If our aim is to stay as close as possible to (Wu 2021)’s analysis, we can also read the sections of the paper where they describe their choice of variables:

“For the WVS, I use the standard survey item asking, “Generally speaking, would you say that most people can be trusted or that you can’t be too careful in dealing with people?” (Rosenberg 1956). The variable is coded on a 0 to 1 scale, with 1 corresponding to high levels of trust.” (p. 1170)

“For the WVS, I measure educational attainment using respondents’ highest education level attained with eight categories, namely, 1 = no formal education or inadequately completed elementary education, 2 = completed (compulsory) elementary education, 3 = incomplete secondary school: technical/vocational, 4 = complete secondary school: technical/vocational, 5 = incomplete secondary: university, 6 = complete secondary: university, 7 = some university without degree, and 8 = university with degree/higher education.

In some analyses, I treat education as a categorical variable. To reduce the number of categories, I recode respondents’ education into Primary, Secondary, Post-secondary, and Tertiary. This is also consistent with the most recent wave of the WVS coding” (p. 1171)

“I also control for relevant demographic covariates such as gender, age, income, marital status, and occupational status at the individual level” (p. 1172)

Based on the above description, the most appropriate variables available to us in the WVS7 dataset are:

Table : WVS7, selected variables (94278 rows and 7 variables, 7 shown)

| ID | Name | Label | Type | Missings | Values | Value Labels | N |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | Q57 | Most people can be trusted | categorical | 1273 (1.4%) | 1 | Most people can be trusted | 22552 (24.2%) |
|  |  |  |  |  | 2 | Need to be very careful | 70453 (75.8%) |
| 2 | Q275 | Highest educational level: Respondent [ISCED 2011] | categorical | 1012 (1.1%) | 0 | Early childhood education (ISCED 0) / no education | 4708 (5.0%) |
|  |  |  |  |  | 1 | Primary education (ISCED 1) | 11101 (11.9%) |
|  |  |  |  |  | 2 | Lower secondary education (ISCED 2) | 14082 (15.1%) |
|  |  |  |  |  | 3 | Upper secondary education (ISCED 3) | 23880 (25.6%) |
|  |  |  |  |  | 4 | Post-secondary non-tertiary education (ISCED 4) | 8367 (9.0%) |
|  |  |  |  |  | 5 | Short-cycle tertiary education (ISCED 5) | 7818 (8.4%) |
|  |  |  |  |  | 6 | Bachelor or equivalent (ISCED 6) | 16133 (17.3%) |
|  |  |  |  |  | 7 | Master or equivalent (ISCED 7) | 6076 (6.5%) |
|  |  |  |  |  | 8 | Doctoral or equivalent (ISCED 8) | 1101 (1.2%) |
| 3 | Q260 | Sex | categorical | 95 (0.1%) | 1 | Male | 44403 (47.1%) |
|  |  |  |  |  | 2 | Female | 49780 (52.9%) |
| 4 | Q262 | Age | numeric | 510 (0.5%) | [16, 103] |  | 93768 |
| 5 | Q288 | Scale of incomes | numeric | 2928 (3.1%) | 1 | Lower step | 7078 (7.7%) |
|  |  |  |  |  | 2 | Second step | 5344 (5.9%) |
|  |  |  |  |  | 3 | Third step | 10339 (11.3%) |
|  |  |  |  |  | 4 | Fourth step | 12847 (14.1%) |
|  |  |  |  |  | 5 | Fifth step | 21949 (24.0%) |
|  |  |  |  |  | 6 | Sixth step | 14108 (15.4%) |
|  |  |  |  |  | 7 | Seventh step | 10643 (11.7%) |
|  |  |  |  |  | 8 | Eight step | 5504 (6.0%) |
|  |  |  |  |  | 9 | Nineth step | 1706 (1.9%) |
|  |  |  |  |  | 10 | Tenth step | 1832 (2.0%) |
| 6 | Q273 | Marital status | categorical | 577 (0.6%) | 1 | Married | 52028 (55.5%) |
|  |  |  |  |  | 2 | Living together as married | 7489 (8.0%) |
|  |  |  |  |  | 3 | Divorced | 4256 (4.5%) |
|  |  |  |  |  | 4 | Separated | 2082 (2.2%) |
|  |  |  |  |  | 5 | Widowed | 5427 (5.8%) |
|  |  |  |  |  | 6 | Single | 22419 (23.9%) |
| 7 | Q279 | Employment status | categorical | 1186 (1.3%) | 1 | Full time (30 hours a week or more) | 34071 (36.6%) |
|  |  |  |  |  | 2 | Part time (less than 30 hours a week) | 7972 (8.6%) |
|  |  |  |  |  | 3 | Self employed | 13309 (14.3%) |
|  |  |  |  |  | 4 | Retired/pensioned | 11961 (12.8%) |
|  |  |  |  |  | 5 | Homemaker not otherwise employed | 12388 (13.3%) |
|  |  |  |  |  | 6 | Student | 5219 (5.6%) |
|  |  |  |  |  | 7 | Unemployed | 7073 (7.6%) |
|  |  |  |  |  | 8 | Other | 1099 (1.2%) |
|  | | | | | | | |

If we decide that these are all the variables that we will need for our current analysis, we can shrink the wvs dataframe to these selected few columns to make it cleaner to work with. The select() function comes handy for this task. We can either overwrite the dataset that we have in the Environment, or we can save the reduced dataset under a new name (maybe ex1, for “Exercise 1”):

ex1 <- wvs |>   
 select(Q57, Q275, Q260, Q262, Q288, Q273, Q279)  
  
# or #  
  
ex1 <- select(wvs,   
 Q57, Q275, Q260, Q262, Q288, Q273, Q279)

We can make a codebook table like the one above with the data\_codebook() function:

data\_codebook(ex1)

To get a better sense of how the two numeric variables in this dataset are distributed, we can use the describe\_distribution() function that we have used before:

describe\_distribution(ex1)

Variable | Mean | SD | IQR | Range | Skewness | Kurtosis | n | n\_Missing  
------------------------------------------------------------------------------------------  
Q262 | 43.41 | 16.58 | 26 | [16.00, 103.00] | 0.40 | -0.71 | 93768 | 510  
Q288 | 4.89 | 2.08 | 2 | [1.00, 10.00] | 0.03 | -0.28 | 91350 | 2928

We can also look more closely to our main variable on *social trust*:

ex1 |> data\_tabulate(Q57)

Most people can be trusted (Q57) <categorical>  
# total N=94278 valid N=93005  
  
Value | N | Raw % | Valid % | Cumulative %  
------+-------+-------+---------+-------------  
1 | 22552 | 23.92 | 24.25 | 24.25  
2 | 70453 | 74.73 | 75.75 | 100.00  
<NA> | 1273 | 1.35 | <NA> | <NA>

## Step 2: Recoding and renaming variables

Before we proceed with the analysis, we will need to make some alterations to our variables. Our data is generally tidy, but some adjustments are needed:

* Variable Q57 will serve as our response (dependent) variable - the one we which to explain/model. We can see from the frequency table that it has two valid categories (“1” and “2”), which we can read off from the codebook table that they refer to “Most people can be trusted” and “Need to be very careful”, respectively. Given that it is a binary (dichotomous) variable, we will use *binary logistic regression* to model it. When we enter a variable into a logistic regression model as dependent variable in R, the first category/level will be automatically recoded internally to the value “0” and be used as the reference category; the latter category/level will be recoded to “1”, and it will be the indicator category that the model predicts in contrast to the reference category. In our case, that will mean that the category referring to “Need to be very careful” will be treated as the *indicator* variable by default, so effectively the coefficients of the explanatory (predictor, independent) variables will be estimating the propensity towards “distrust”. If we first reversed the order of the categories in the dependent variable, the model would be estimating “trust”, which is conceptually cleaner and more straightforward to interpret.
* Following (Wu 2021), we will treat our main *explanatory* variable - *Q275*, *Highest educational level* - as numeric. But it in the dataset it is coded as *factor*, so we should change that. Technically, the *education* variable is categorical in nature, but given that it has 9 levels/categories (0 to 8) and there is a clear ordering in the levels running from low education to high education, we can conceptually treat it as numeric, which makes it easier to use a single scale (with a *mean*, *standard deviation*, etc.)
* (Wu 2021) also noted that in some models they used a recoded categorical version of the education, reduced to 4 categories: “Primary”, “Secondary”, “Post-secondary”, and “Tertiary” education. We can also create this variable to potentially include it in our models
* Finally, while we’re at it, we can also rename our variables to have more human-readable names than the original ones.

We can achieve all of the above in a single call to the {datawizard} function data\_modify() like this:

ex1 <- ex1 |> # (1)  
 data\_modify(trusting = reverse(Q57), # (2)  
 educ\_num = as\_numeric(Q275), # (3)  
 educ\_cat = recode\_values(Q275, # (4)  
 recode = list("Primary" = ("0, 1"),   
 "Secondary" = ("2, 3"),   
 "Post-secondary" = ("4, 5"),  
 "Tertiary" = ("6, 7, 8"))), # (4.1)  
 sex = Q260, # (5)  
 age = Q262,  
 income = Q288,  
 marital = Q273,  
 employment = Q279) # (6)   
  
# Code explanation:  
# (1) overwrite the data object with the changes  
# (2) reverse the category order of "Q57" and save it as a new variable called "trusting"  
# (3) change the type of variable "Q275" to numeric and save it as the new variable "educ\_num"  
# (4) recode the "Q275" variable into 4 larger categories and save the new variable as "educ\_cat"  
# (4.1) make sure to close all the brackets under `recode\_values()`  
# (5) rename the remaining variables as "newName" = "oldName"  
# (6) remember to close the brackets under `data\_modify()`

Of course, we could do all the changes one-by-one, but because we can achieve everything we wanted using the data\_modify() option, it’s easier to do it all at once. When you are experimenting with recoding, it is probably a good idea to do it one-by-one and check the results at each step to catch out any errors more easily.

Let’s have a look at the recoded dataset; because we did not overwrite any of the original variables, we will have both the old and the newly recoded/renamed variable in the dataset, but we will be referring to the new variables from now on:

## Save the codebook table to an object:  
  
ex1\_codebook <- ex1 |> data\_codebook()  
  
## Open in a Viewer window:  
  
View(ex1\_codebook)

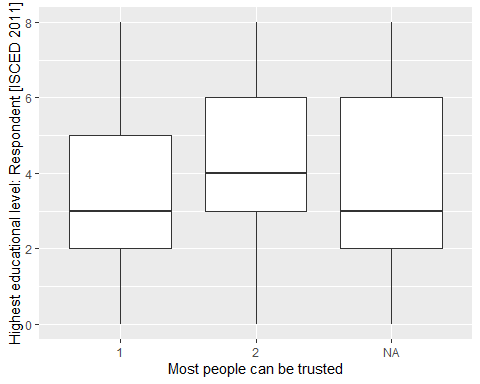
Table : ex1 (94278 rows and 15 variables, 15 shown)

| ID | Name | Label | Type | Missings | Values | Value Labels | N |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | Q57 | Most people can be trusted | categorical | 1273 (1.4%) | 1 | Most people can be trusted | 22552 (24.2%) |
|  |  |  |  |  | 2 | Need to be very careful | 70453 (75.8%) |
| 2 | Q275 | Highest educational level: Respondent [ISCED 2011] | categorical | 1012 (1.1%) | 0 | Early childhood education (ISCED 0) / no education | 4708 (5.0%) |
|  |  |  |  |  | 1 | Primary education (ISCED 1) | 11101 (11.9%) |
|  |  |  |  |  | 2 | Lower secondary education (ISCED 2) | 14082 (15.1%) |
|  |  |  |  |  | 3 | Upper secondary education (ISCED 3) | 23880 (25.6%) |
|  |  |  |  |  | 4 | Post-secondary non-tertiary education (ISCED 4) | 8367 (9.0%) |
|  |  |  |  |  | 5 | Short-cycle tertiary education (ISCED 5) | 7818 (8.4%) |
|  |  |  |  |  | 6 | Bachelor or equivalent (ISCED 6) | 16133 (17.3%) |
|  |  |  |  |  | 7 | Master or equivalent (ISCED 7) | 6076 (6.5%) |
|  |  |  |  |  | 8 | Doctoral or equivalent (ISCED 8) | 1101 (1.2%) |
| 3 | Q260 | Sex | categorical | 95 (0.1%) | 1 | Male | 44403 (47.1%) |
|  |  |  |  |  | 2 | Female | 49780 (52.9%) |
| 4 | Q262 | Age | numeric | 510 (0.5%) | [16, 103] |  | 93768 |
| 5 | Q288 | Scale of incomes | numeric | 2928 (3.1%) | 1 | Lower step | 7078 (7.7%) |
|  |  |  |  |  | 2 | Second step | 5344 (5.9%) |
|  |  |  |  |  | 3 | Third step | 10339 (11.3%) |
|  |  |  |  |  | 4 | Fourth step | 12847 (14.1%) |
|  |  |  |  |  | 5 | Fifth step | 21949 (24.0%) |
|  |  |  |  |  | 6 | Sixth step | 14108 (15.4%) |
|  |  |  |  |  | 7 | Seventh step | 10643 (11.7%) |
|  |  |  |  |  | 8 | Eight step | 5504 (6.0%) |
|  |  |  |  |  | 9 | Nineth step | 1706 (1.9%) |
|  |  |  |  |  | 10 | Tenth step | 1832 (2.0%) |
| 6 | Q273 | Marital status | categorical | 577 (0.6%) | 1 | Married | 52028 (55.5%) |
|  |  |  |  |  | 2 | Living together as married | 7489 (8.0%) |
|  |  |  |  |  | 3 | Divorced | 4256 (4.5%) |
|  |  |  |  |  | 4 | Separated | 2082 (2.2%) |
|  |  |  |  |  | 5 | Widowed | 5427 (5.8%) |
|  |  |  |  |  | 6 | Single | 22419 (23.9%) |
| 7 | Q279 | Employment status | categorical | 1186 (1.3%) | 1 | Full time (30 hours a week or more) | 34071 (36.6%) |
|  |  |  |  |  | 2 | Part time (less than 30 hours a week) | 7972 (8.6%) |
|  |  |  |  |  | 3 | Self employed | 13309 (14.3%) |
|  |  |  |  |  | 4 | Retired/pensioned | 11961 (12.8%) |
|  |  |  |  |  | 5 | Homemaker not otherwise employed | 12388 (13.3%) |
|  |  |  |  |  | 6 | Student | 5219 (5.6%) |
|  |  |  |  |  | 7 | Unemployed | 7073 (7.6%) |
|  |  |  |  |  | 8 | Other | 1099 (1.2%) |
| 8 | trusting | Most people can be trusted | categorical | 1273 (1.4%) | 1 | Need to be very careful | 70453 (75.8%) |
|  |  |  |  |  | 2 | Most people can be trusted | 22552 (24.2%) |
| 9 | educ\_num | Highest educational level: Respondent [ISCED 2011] | numeric | 1012 (1.1%) | 0 | Early childhood education (ISCED 0) / no education | 4708 (5.0%) |
|  |  |  |  |  | 1 | Primary education (ISCED 1) | 11101 (11.9%) |
|  |  |  |  |  | 2 | Lower secondary education (ISCED 2) | 14082 (15.1%) |
|  |  |  |  |  | 3 | Upper secondary education (ISCED 3) | 23880 (25.6%) |
|  |  |  |  |  | 4 | Post-secondary non-tertiary education (ISCED 4) | 8367 (9.0%) |
|  |  |  |  |  | 5 | Short-cycle tertiary education (ISCED 5) | 7818 (8.4%) |
|  |  |  |  |  | 6 | Bachelor or equivalent (ISCED 6) | 16133 (17.3%) |
|  |  |  |  |  | 7 | Master or equivalent (ISCED 7) | 6076 (6.5%) |
|  |  |  |  |  | 8 | Doctoral or equivalent (ISCED 8) | 1101 (1.2%) |
| 10 | educ\_cat | Highest educational level: Respondent [ISCED 2011] | categorical | 1012 (1.1%) | Post-secondary |  | 16185 (17.4%) |
|  |  |  |  |  | Primary |  | 15809 (17.0%) |
|  |  |  |  |  | Secondary |  | 37962 (40.7%) |
|  |  |  |  |  | Tertiary |  | 23310 (25.0%) |
| 11 | sex | Sex | categorical | 95 (0.1%) | 1 | Male | 44403 (47.1%) |
|  |  |  |  |  | 2 | Female | 49780 (52.9%) |
| 12 | age | Age | numeric | 510 (0.5%) | [16, 103] |  | 93768 |
| 13 | income | Scale of incomes | numeric | 2928 (3.1%) | 1 | Lower step | 7078 (7.7%) |
|  |  |  |  |  | 2 | Second step | 5344 (5.9%) |
|  |  |  |  |  | 3 | Third step | 10339 (11.3%) |
|  |  |  |  |  | 4 | Fourth step | 12847 (14.1%) |
|  |  |  |  |  | 5 | Fifth step | 21949 (24.0%) |
|  |  |  |  |  | 6 | Sixth step | 14108 (15.4%) |
|  |  |  |  |  | 7 | Seventh step | 10643 (11.7%) |
|  |  |  |  |  | 8 | Eight step | 5504 (6.0%) |
|  |  |  |  |  | 9 | Nineth step | 1706 (1.9%) |
|  |  |  |  |  | 10 | Tenth step | 1832 (2.0%) |
| 14 | marital | Marital status | categorical | 577 (0.6%) | 1 | Married | 52028 (55.5%) |
|  |  |  |  |  | 2 | Living together as married | 7489 (8.0%) |
|  |  |  |  |  | 3 | Divorced | 4256 (4.5%) |
|  |  |  |  |  | 4 | Separated | 2082 (2.2%) |
|  |  |  |  |  | 5 | Widowed | 5427 (5.8%) |
|  |  |  |  |  | 6 | Single | 22419 (23.9%) |
| 15 | employment | Employment status | categorical | 1186 (1.3%) | 1 | Full time (30 hours a week or more) | 34071 (36.6%) |
|  |  |  |  |  | 2 | Part time (less than 30 hours a week) | 7972 (8.6%) |
|  |  |  |  |  | 3 | Self employed | 13309 (14.3%) |
|  |  |  |  |  | 4 | Retired/pensioned | 11961 (12.8%) |
|  |  |  |  |  | 5 | Homemaker not otherwise employed | 12388 (13.3%) |
|  |  |  |  |  | 6 | Student | 5219 (5.6%) |
|  |  |  |  |  | 7 | Unemployed | 7073 (7.6%) |
|  |  |  |  |  | 8 | Other | 1099 (1.2%) |
|  | | | | | | | |

## Step 3: Describe the relationship between the main variables of interest

Now that the data has been recoded, we can have a closer look at the relationship between our two main variables: *trusting* and *educ\_num*. Since the *education* variable is now recorded as numeric, we can make use of a *boxplot* for visualisation:

gf\_boxplot(educ\_num ~ trusting, data = ex1)



We can see a slightly higher median education level among those responding that “Most people can be trusted” (category “2”). For a more detailed numerical summary of this association, we can use another function that we know from Lab3, means\_by\_group():

ex1 |> means\_by\_group(educ\_num, trusting)

# Grouped Means for Highest educational level: Respondent [ISCED 2011] by Most people can be trusted  
  
Category | Mean | N | SD | SE | p  
--------------------------------------------------------------------  
Need to be very careful | 3.38 | 69831 | 1.97 | 7.47e-03 | < .001  
Most people can be trusted | 4.09 | 22286 | 2.07 | 0.01 | < .001  
Total | 3.55 | 92117 | 2.02 | 6.66e-03 |   
  
Anova: R2=0.023; adj.R2=0.023; F=2159.726; p=0.000

Here we can see more clearly the difference in average education levels between those with high *trust* levels and those with low *trust* levels. We also know from Lab3 that this table is equivalent to the results from a linear regression model in which the numeric variable educ\_num is the dependent variable and trusting is the explanatory (independent) variable.

However, we are interested here in modelling the trusting variable as a function of *education*, while also accounting for the other demographic covariates. For this, we turn to a *logistic regression model*.

## Step 4: Fit and interpret a simple bivariate logistic regression model

Fitting a logistic regression model is fairly simple. We can use the base R glm() function and would write the following code:

m1 <- glm(trusting ~ educ\_num, data = ex1, family = binomial)

In the code above, instead of the *linear model* function (lm()) we used the *generalised linear model* function (glm()), and because glm() is a larger family of models, we need to also specify which “family” we are fitting; since our dependent variable is binary variable, we are calling the “binomial” family. In all other aspects the formula is the same as the one we already know from *linear regression*.

Let’s look at the model parameters:

model\_parameters(m1)

Parameter | Log-Odds | SE | 95% CI | z | p  
---------------------------------------------------------------------  
(Intercept) | -1.80 | 0.02 | [-1.83, -1.76] | -105.85 | < .001  
educ num | 0.18 | 3.86e-03 | [ 0.17, 0.18] | 45.46 | < .001

It is at the stage of interpretation that logistic regression models become difficult. The coefficients that we obtain in the standard output are similar to those from a linear regression in that they summarise a linear and additive relationship between the independent and dependent variables. The difference is that the dependent variable has been transformed to a *logit* or *logged odds* scale. We could say - following the logic of linear regression - that each unit difference in one’s education level is associated with a *0.18*-point positive difference in the *logged odds*/*logit* value of *trusting*. But this doesn’t make much sense, because the *logit* scale is difficult to comprehend. It would be more intuitive to see the results on a different scale, such as *odds* or *probabilities*.

If we exponentiate the logistic regression coefficients, we could say that the independent variables relate to the *odds* of the dependent variable, rather than the *logged odds*, which is somewhat easier to relate to. The model\_parameters() function can do that for us in exchange of some additional specification:

model\_parameters(m1, exponentiate = TRUE)

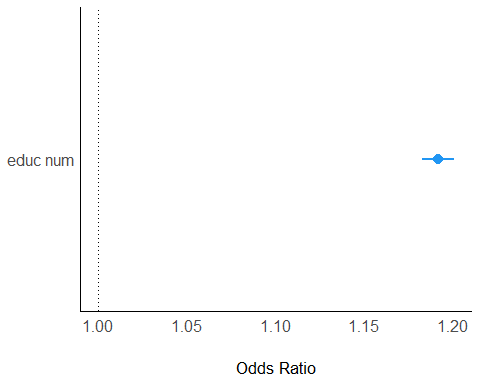
Parameter | Odds Ratio | SE | 95% CI | z | p  
---------------------------------------------------------------------  
(Intercept) | 0.17 | 2.81e-03 | [0.16, 0.17] | -105.85 | < .001  
educ num | 1.19 | 4.60e-03 | [1.18, 1.20] | 45.46 | < .001

The *odds ratios* that we find in this table are somewhat easier to interpret, but we need to understand how *odds* work. By having exponentiated the coefficients, we have moved from an *additive* scale to a *multiplicative* scale. Basically, what this means is that while on an *additive* scale the value “0” means no change (because by *adding* 0 to a number we don not alter the number), on a *multiplicative* scale that role is taken over by “1” (when we multiply a number by “1”, we keep it as it is). Consequently, a coefficient of “1” on the exponentiated scale means *no difference*, while values between 0 and 1 represent a *negative difference* and values greater than 1 a *positive difference*. I our case, the coefficient of **1.19** means that a difference of one degree in education level is associated with a positive difference of **1.19** in the odds of being A *trusting* person. We could further translate this into a percentage: if we subtract 1 from the exponentiated coefficients and multiply it by 100, we get the percentage difference due to a unit change in the independent variable.

For values grater than 1 this calculation is very straightforward. In our case, this translated to a **19%** increase in the *odds* of being a *trusting* person.

It doesn’t add much to our understanding to plot the regression results from a model with a single predictor, but knowing how to plot results will be particularly useful once we expand the model with further explanatory variables. To create a plot of the results, we can simply take the output from the model\_parameters() function and pass it on to the plot() function:

parameters(m1, exponentiate = TRUE) |> plot()



However, the fact that the exponentiated coefficients are measured on a multiplicative scale poses some difficulties and it makes it easy to fall into the trap of interpreting these *odds* as *probabilities*. The issue is that odds can range from 0 to infinity, so the scale is far from standard. *Probabilities*, on the other hand, have the nice feature of being constrained between 0 and 1, so a percentage change has much more meaning.

There are several packages that help us compute various comparative statistics on the *probability* scale from logistic models. Below, we will use the avg\_slopes() function from the {marginaleffects} package to compute the average difference in the probability of being *trusting* based on a one-unit difference in education level:

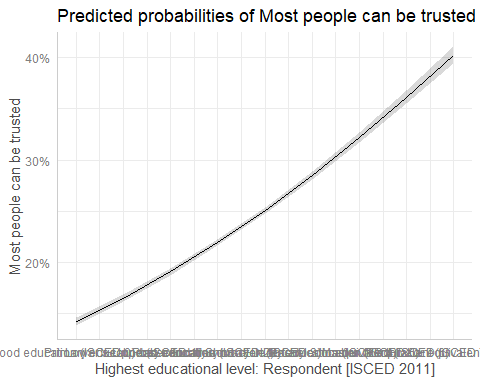
avg\_slopes(m1)

Term Estimate Std. Error z Pr(>|z|) S 2.5 % 97.5 %  
 educ\_num 0.0314 0.00067 46.9 <0.001 Inf 0.0301 0.0327  
  
Columns: term, estimate, std.error, statistic, p.value, s.value, conf.low, conf.high   
Type: response

The result tells us that that probability is just over **3%**.

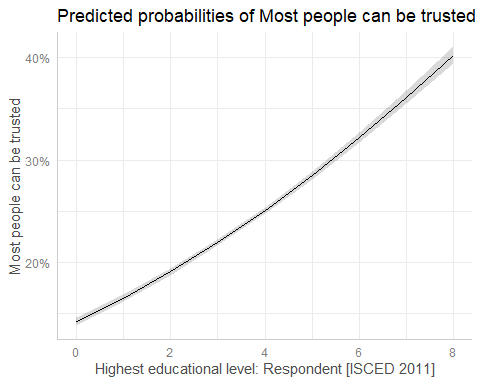
To make these numerical results easier to understand, we can plot the result. There are several options, but probably the most straightforward approach is to estimate the probabilities of being *trusting* at each level of the predictor variable (*education*). The plot\_predictions() function from the {marginaleffects} package or the plot() function of {ggpredict} can be used here. Below, we’ll use the latter.

ggpredict(m1, "educ\_num") |>   
 plot()



This code first calculates predictions of the outcome probability for each level of the explanatory variable (we could save these results as a dataframe), then passes those results on to the plot() function. To remove the overlapping value labels on the horizontal axis,, we could add something like the following to the command:

ggpredict(m1, "educ\_num") |>   
 plot() +  
 scale\_x\_continuous()



This graph makes it a lot clearer how *education* is associated with *trust* in our model, and if *education* is our main variable of interest, we can plot its effect even from a larger model that contains other covariates.

## Step 5: Add demographic control variables to the logistic regression model

This step does not contain anything new compared to what we already know from multiple linear regression and what we have learnt in the previous step about the logistic regression model. Below I’ll summarise the code, but will leave the interpretation of the results to you.

Fit the multiple logistic regression model:

m2 <- glm(trusting ~ educ\_num + sex + age + income + marital + employment, data = ex1, family = binomial)

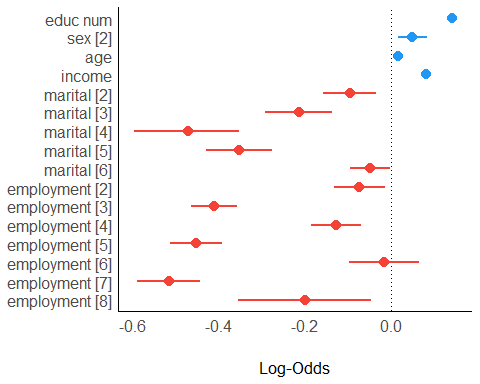
Tabulate the model parameters:

model\_parameters(m2)

Parameter | Log-Odds | SE | 95% CI | z | p  
-----------------------------------------------------------------------  
(Intercept) | -2.69 | 0.05 | [-2.78, -2.60] | -59.04 | < .001  
educ num | 0.14 | 4.33e-03 | [ 0.14, 0.15] | 33.20 | < .001  
sex [2] | 0.05 | 0.02 | [ 0.02, 0.08] | 2.89 | 0.004   
age | 0.02 | 7.07e-04 | [ 0.02, 0.02] | 25.18 | < .001  
income | 0.08 | 4.17e-03 | [ 0.07, 0.09] | 19.79 | < .001  
marital [2] | -0.10 | 0.03 | [-0.16, -0.03] | -3.03 | 0.002   
marital [3] | -0.21 | 0.04 | [-0.29, -0.14] | -5.37 | < .001  
marital [4] | -0.47 | 0.06 | [-0.60, -0.35] | -7.58 | < .001  
marital [5] | -0.35 | 0.04 | [-0.43, -0.28] | -9.09 | < .001  
marital [6] | -0.05 | 0.02 | [-0.09, 0.00] | -2.04 | 0.042   
employment [2] | -0.07 | 0.03 | [-0.13, -0.01] | -2.42 | 0.015   
employment [3] | -0.41 | 0.03 | [-0.46, -0.36] | -15.23 | < .001  
employment [4] | -0.13 | 0.03 | [-0.19, -0.07] | -4.21 | < .001  
employment [5] | -0.45 | 0.03 | [-0.51, -0.39] | -14.84 | < .001  
employment [6] | -0.01 | 0.04 | [-0.10, 0.07] | -0.36 | 0.721   
employment [7] | -0.52 | 0.04 | [-0.59, -0.44] | -13.80 | < .001  
employment [8] | -0.20 | 0.08 | [-0.35, -0.05] | -2.52 | 0.012

Plot the model parameters on the log-odds scale:

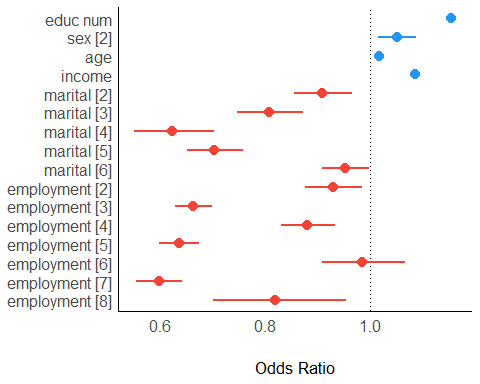
model\_parameters(m2) |> plot()



The plotted results are much more informative in the case of the multiple regression model. We find that higher education, being a female, higher age and higher income brackets all have a positive association with *trust*, while being other than married and in non-full-time employment show a negative association with trust.

We can plot the exponentiated coefficients to interpret odds instead:

model\_parameters(m2, exponentiate = TRUE) |> plot()



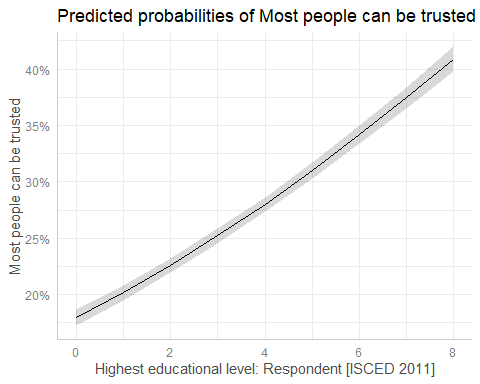
Transform the coefficients to *probability* scale:

avg\_slopes(m2)

Term Contrast Estimate Std. Error z Pr(>|z|) S 2.5 %  
 age dY/dX 0.00309 0.000121 25.433 < 0.001 471.6 0.00285  
 educ\_num dY/dX 0.02494 0.000738 33.807 < 0.001 829.8 0.02349  
 employment 2 - 1 -0.01359 0.005554 -2.448 0.01438 6.1 -0.02448  
 employment 3 - 1 -0.07055 0.004406 -16.010 < 0.001 189.2 -0.07918  
 employment 4 - 1 -0.02329 0.005433 -4.287 < 0.001 15.8 -0.03394  
 employment 5 - 1 -0.07702 0.004872 -15.808 < 0.001 184.6 -0.08657  
 employment 6 - 1 -0.00277 0.007736 -0.359 0.71990 0.5 -0.01794  
 employment 7 - 1 -0.08629 0.005664 -15.234 < 0.001 171.7 -0.09739  
 employment 8 - 1 -0.03589 0.013638 -2.632 0.00849 6.9 -0.06262  
 income dY/dX 0.01431 0.000719 19.917 < 0.001 290.8 0.01290  
 marital 2 - 1 -0.01657 0.005376 -3.083 0.00205 8.9 -0.02711  
 marital 3 - 1 -0.03613 0.006425 -5.624 < 0.001 25.7 -0.04872  
 marital 4 - 1 -0.07486 0.008726 -8.579 < 0.001 56.5 -0.09196  
 marital 5 - 1 -0.05768 0.005850 -9.859 < 0.001 73.8 -0.06915  
 marital 6 - 1 -0.00852 0.004163 -2.047 0.04065 4.6 -0.01668  
 sex 2 - 1 0.00872 0.003010 2.896 0.00378 8.0 0.00282  
 97.5 %  
 0.003325  
 0.026382  
 -0.002709  
 -0.061910  
 -0.012644  
 -0.067473  
 0.012388  
 -0.075190  
 -0.009162  
 0.015721  
 -0.006036  
 -0.023539  
 -0.057754  
 -0.046214  
 -0.000363  
 0.014614  
  
Columns: term, contrast, estimate, std.error, statistic, p.value, s.value, conf.low, conf.high   
Type: response

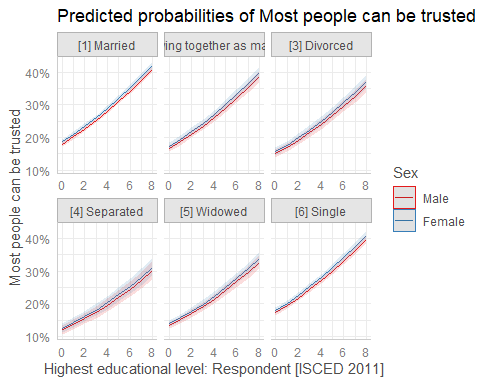
Plot predicted probabilities of *trusting* based on *education*, adjusted for the effect of the other covariates in the model:

ggpredict(m2, c("educ\_num")) |>   
 plot() +  
 scale\_x\_continuous()



Given the multiple explanatory variables included in the model, we can further break down the effects. For example, we can plot the effect of *education* on *trust* **by** *sex* **within** each *marital status*.

ggpredict(m2, c("educ\_num", "sex", "marital")) |>   
 plot() +  
 scale\_x\_continuous()



It looks like the effect of *education* on *trust* is stronger among those married, or single or cohabitating than among the divorced, separated or widowed, for both men and women.

This is a rather complex finding, and we can explore this model in much more depth, as well as thinking about developing it or specifying it further.

For the next exercise, let’s make a few minor tweaks to the data and model and compare the results.

## Exercise 2: Refitting the model for a single country

You will carry out this exercise on your own, but we’ll make two adjustments:

* Instead of treating the entire dataset as undifferentiated, originating from one single population, we will acknowledge the fact that the data originate from various countries and that the local socio-cultural context has an impact om social behaviours and attitudes. To account for this, fit two versions of the multiple logistic regression model we fit in the earlier exercise:
  1. fit the same model as before, but add the *country* variable to the model as a covariate (search the WVS7 for the variable name; you will need to select the variables again from the main dataset, as in Step 1 above);
  2. select only *one* country of your choice, reduce the dataset to that country and fit the model on that data;
* Replace the educ\_num variable with its categorical counterpart educ\_cat and check how it affects the model.

In order to select countries from the data, we need to learn about another function, filter(), which lets us select rows (cases) given some criteria. Let’s make the following changes to the code we used in Step 1 and 2 to select the data we were interested in and transform it:

ex2 <- wvs |>   
 select(B\_COUNTRY, Q57, Q275, Q260, Q262, Q288, Q273, Q279) |> # (1)  
 data\_modify(trusting = reverse(Q57),  
 trusting = as\_label(trusting), # (2)  
 educ\_num = as\_numeric(Q275),   
 educ\_cat = recode\_values(Q275,   
 recode = list("Primary" = ("0, 1"),   
 "Secondary" = ("2, 3"),   
 "Post-secondary" = ("4, 5"),  
 "Tertiary" = ("6, 7, 8"))),  
 country = as\_label(B\_COUNTRY),  
 educ\_cat = as\_label(educ\_cat),  
 sex = as\_factor(Q260),   
 age = Q262,  
 income = Q288,  
 marital = as\_label(Q273),  
 employment = as\_label(Q279))  
  
  
ex2.2 <- ex2 |> filter(country == "Greece") # (3)  
  
# Code explanation:  
# (1) also keep Country variable  
# (2) also transform the categorical variables to factors with character/label values to improve the outputs  
# (3) select one country by character name; here Greece, but you should select another one.

With these datasets you should be able to complete the modelling tasks above. Good luck!

Wu, Cary. 2021. “Education and Social Trust in Global Perspective.” *Sociological Perspectives* 64 (6): 1166–86. <https://doi.org/10.1177/0731121421990045>.