Exercise 4

No Fear of Numbers: Introduction to Quantitative Data Analysis in R

Dr. Susanne de Vogel, Data Science Center, University of Bremen

2 December 2025

Research Question: Is there a Life beyond the PhD?

In this workshop, we will work with one coherent example throughout all four parts.We focus on two outcomes of interest:

- `bpsy01`: overall life satisfaction (10 point scale)
- `blcd06`: Children (yes/no)

We will examine how these outcomes are related to different aspects of doctoral researchers' lives and backgrounds:

PhD and Work conditions

- `adbi01`: Status of the doctorate
- `adbi15`: Discipline
- `bdcd17`: Emotional Support during PhD
- `bwdr12`: Perceived scientific Pressure
- `bemp81`: Monthly gross income

Attitudes and Well-Being

- `bldc12`: Satisfaction with work-life balance
- `apar15b`: Relationship with parents
- `bpsy05`: Self-Efficacy

Demographics

- `adem01`: Gender
- `adem02`: Age in years
- `apar10`: Highest vocational degree of parents
- `adem03`: Country of birth
- `blcd01`: Relationship status

Multivariate Analysis: Regression Models

So far, we have looked at our data **one variable at a time** (univariate) and in **pairs** (bivariate).
In this sheet, we take the next step to **multivariate analysis**: we use **regression models** to study how several predictors together are related to an outcome.

In the context of our research story, our main outcomes of interest are still:

- overall life satisfaction (`bpsy01`), and
- having children (yes/no, `blcd06`).

Now we want to answer questions like:

- How is life satisfaction related to **several factors at once** (e.g. work–life compatibility, emotional support, burden to perish, self efficacy, income)?
- Does a predictor still matter for life satisfaction **after controlling for** other variables?
- How do different predictors relate to the **probability of having children**?

In this exercise, you will learn to:

- fit and interpret a **multiple linear regression** with `bpsy01` as the dependent variable,
- build and compare **nested linear models** (adding blocks of predictors and comparing model fit),
- fit and interpret a **logistic regression** with `blcd06` (children yes/no) as the dependent variable,
- report and interpret **regression coefficients**, **confidence intervals**, and (for logistic regression) **odds ratios**,
- summarize the main results in simple sentences (e.g. "Holding other variables constant, higher burden is associated with lower life satisfaction").

2.1. Set-up: Load packages and dataset

1. Install `tidyverse`, `haven`, `tidymodels` if not already installed and load these packages.

Solution:

```r
# Install tidyverse
if (!requireNamespace("tidyverse", quietly = TRUE))
  install.packages("tidyverse")

# Install tidymodels
if (!requireNamespace("tidymodels", quietly = TRUE))
  install.packages("tidymodels")

# Install haven
if (!requireNamespace("haven", quietly = TRUE))
  install.packages("haven")


## load packages
library(tidyverse)
library(haven)
library(tidymodels)
```

2. For this exercise, we will use the file *04_qa_multi_data.sav* in the *exercises* folder. Define the *exercise* folder as your working directory with the function `setwd("path")`.

3. Choose `mydata4` as a name for your data frame and import it with the following structure: `chosen_name <- read_sav("filename")`.

Solution:

```
# set working directory
setwd(
  "C:/Users/Susanne/Nextcloud/share_DSC/003_Trainings/001_Workshops/2025/2025-12-02_SdV_Data_Analysis/Materials/E
  )

# Load the Nacaps data set and name this data frame "mydata3"
mydata4 <- read_sav("04_qa_multi_data.sav")
```

On the right side, in the Environment pane, the data frame `mydata4` should now appear under **Data**.

2.2. Converting categorical to factor variables

In preparation of the multiple linear regression, make sure the categorical variables are factors.

Use `mutate()` and `as_factor()` to transform all categorical variables (`adbi15, adem01, adem03, apar10, bdbi01, blcd01, blcd06`) into a factor variable. Save the result again as `mydata4`.

The code should look like this:

```
mydata4 <- mydata4 %>%
  mutate(
    across(
      c(adbi15, adem01, adem03, apar10, bdbi01, blcd01, blcd06),   # all cat. variables
      ~ haven::as_factor(.x)       # labels -> factor levels
    )
  )

glimpse(mydata4)
```

```
## Rows: 2,354
## Columns: 17
## $ pid       <dbl> 2, 15, 39, 55, 75, 81, 90, 104, 116, 117, 137, 165, 166, 167~
## $ adbi15    <fct> "engineering sciences", "mathematics, natural sciences", "hu~
## $ adem01    <fct> male, male, female, female, female, male, male, male, male, ~
## $ adem02    <dbl> 27, 25, 41, 30, 26, 25, 27, 28, 34, 29, 34, 27, 42, 29, 31, ~
## $ adem03    <fct> "in Germany", "in Germany", "in another country, namely:", "~
## $ apar10    <fct> Other vocational qualification, PhD/doctorate, Bachelor / Ma~
## $ apar15b   <dbl+lbl>  3,  4,  2,  3,  3,  3,  4,  3,  3,  3,  4,  3,  3,  4, ~
## $ bdbi01    <fct> I am doing a PhD/doctorate., I am doing a PhD/doctorate., I ~
## $ bdcd17    <dbl> 3.000000, 4.333333, 4.000000, 4.666667, 1.000000, 3.333333, ~
## $ bemp81    <dbl> 3076, 0, 0, 0, 985, 1762, 4266, 4381, NA, NA, 2592, 660, 428~
## $ bemp81_g3 <dbl+lbl>  3, NA, NA, NA,  1,  2,  3,  3, NA, NA,  3,  1,  3,  3, ~
## $ bdwr12    <dbl> 2.50, 2.50, 2.00, 2.50, 1.00, 3.50, 2.50, 2.50, NA, NA, 4.25~
## $ bpsy01    <dbl+lbl>  5,  7,  7,  8,  5,  5,  7,  7, NA, NA,  8,  5,  8,  9, ~
## $ blcd01    <fct> no, no, yes, yes, yes, no, no, no, NA, NA, yes, no, yes, yes~
## $ blcd06    <fct> no, no, no, no, no, no, no, no, NA, NA, yes, no, yes, no, ye~
## $ blcd12    <dbl+lbl>  8,  3,  8,  8,  5, 10,  3,  5, NA, NA,  3,  7,  9,  9, ~
## $ bpsy05    <dbl> 5.000000, 4.000000, 4.333333, 3.666667, 4.666667, 5.000000, ~
```

2.3. Multiple linear regression: life satisfaction <- PhD % work condition

We want to model **overall life satisfaction** (`bpsy01`) as a function of PhD characteristics and work conditions:

- `bdbi01`: PhD status (factor)
- `adbi15`: doctoral discipline (factor)
- `bdcd17`: – emotional support during PhD (5-point Likert scale, treated as metric)
- `bdwr12`: – burden index (5-point Likert scale, treated as metric)
- `bemp81`: – monthly gross income in €

We use **multiple linear regression** with `lm()` and **tidy the output** with *tidy*.

1. Create a cleaned data frame `mydata_lm_ex1` that only contains rows with non-missing values for the dependent variable and all predictors in this model.

The code should look like this:

```
mydata_lm_ex1 <- mydata4 %>%
  filter(
    !is.na(bpsy01),   # dependent variable: life satisfaction
    !is.na(bdbi01),
    !is.na(adbi15),
    !is.na(bdcd17),
    !is.na(bdwr12),
    !is.na(bemp81)
  )
```

The new data frame `mydata_lm_ex1` should now appear in your environment pane.

2. Fit a multiple linear regression with `bpsy01` as the dependent variable and `bdbi01`, `adbi15`, `bdcd17`, `bdwr12`, and `bemp81` as predictors.

The code should look like this:

```
lm_phd <- lm(  # save results in data frame lm_phd
  bpsy01 ~ bdbi01 + adbi15 + bdcd17 + bdwr12 + bemp81, # bpsy01 as dependent var, others as
 ↪  predictors
  data = mydata_lm_ex1 # use data frame mydata_lm_phd
)
```

Tipp: By default, R uses the first factor level (often the lowest numeric code) as the reference category.You can change the reference directly inside the regression call using `relevel()` in the formula. For example like this:

```
lm_phd <- lm(
  bpsy01 ~
    bdbi01 +
    relevel(adbi15, ref = "mathematics, natural sciences") +  # change reference category here
    bdcd17 +
    bdwr12 +
    bemp81,
  data = mydata_lm_ex1
)
```

3. Create a tidy table of coefficients and check the results.

Do it like this:

```r
# use tidy data frame lm_phd, save results in data frame lm_phd_results and display 95% confidence
↪  intervals
lm_phd_results <- tidy(lm_phd, conf.int = TRUE) %>%
  mutate(across(where(is.numeric), ~ round(.x, 3)) # optional: round output to three decimals
  )

lm_phd_results
```

```
## # A tibble: 14 x 7
##    term                  estimate std.error statistic p.value conf.low conf.high
##    <chr>                    <dbl>     <dbl>     <dbl>   <dbl>    <dbl>     <dbl>
##  1 (Intercept)              6.31      0.21     30.1     0        5.90      6.72
##  2 bdbi01I have complet~    0.064     0.119     0.538   0.591   -0.17      0.298
##  3 bdbi01I have interru~    0.094     0.272     0.345   0.73    -0.439     0.627
##  4 bdbi01I have quit my~    0.452     0.388     1.17    0.244   -0.309     1.21
##  5 adbi15sports             0.31      0.451     0.688   0.491   -0.574     1.20
##  6 adbi15law, economics~   -0.002     0.146    -0.011   0.991   -0.287     0.284
##  7 adbi15mathematics, n~   -0.026     0.137    -0.192   0.848   -0.296     0.243
##  8 adbi15human medicine~    0.312     0.162     1.93    0.054   -0.005     0.63
##  9 adbi15agricultural, ~    0.026     0.293     0.089   0.929   -0.549     0.6
## 10 adbi15engineering sc~   -0.155     0.158    -0.984   0.325   -0.464     0.154
## 11 adbi15art, art theory    0.303     0.316     0.96    0.337   -0.316     0.922
## 12 bdcd17                   0.342     0.035     9.86    0        0.274     0.41
## 13 bdwr12                  -0.342     0.051    -6.77    0       -0.441    -0.243
## 14 bemp81                   0         0         5.02    0        0         0
```

You get the following information:

- *term*: variable name (intercept and predictors)
- *estimate*: regression coefficient (change in bpsy01 if the predictor increases by 1, holding - other variables constant; for factors: difference to the reference category, this is always the first)
- *std.error, statistic, p.value* = test statistics
- *conf.low, conf.high* = lower and upper bound of the 95% CI

How do you interpret the results?

Solution:

For **factors** (`bdbi01`, `adbi15`) the coefficients are differences in the mean compared to the reference category, while holding all other variables constant.

> Example adbi01 (PhD Status) I have completed the PhD/doctorate:
>
> - Estimate: 0.06, p = 0.59, CI: [-0.17; 0.30]
> - Interpretation pattern: On average, people who have completed the PhD report about 0.06 points higher life satisfaction than the reference group of people with ongoing PhDs, controlling for discipline, support, burden and income. The effect is small and not statistically significant (p > .05, CI includes 0).

The same logic applies to all other `bdbi01` and `adbi15` categories.

Describe it in the results section like this:

After controlling for emotional support, burden and income, we do not find clear evidence that life satisfaction differs systematically between PhD status groups or across doctoral disciplines. The estimated differences are small and statistically not significant.

For **metric** predictors (`blcd17`, `bdwr12`, `bemp81`), you see the slope per 1-unit (e.g. 1 Euro income or 1-point on likert scale) increase.

Describe it in the results section like this:

For each one-point increase in perceived emotional support, life satisfaction is on average about 0.34 points higher, holding all other variables constant. The effect is statistically significant and clearly positive ($p < .001$, CI is entirely above 0).

For each one-point increase in burden, life satisfaction decreases on average by about 0.34 points, again controlling for all other predictors. The effect is statistically significant and clearly negative ($p < .001$, CI is entirely below 0).

For each increase of 1 Euro in monthly income, life satisfaction increases by only 0.00012 points. Even if the p-value is small, this effect is substantively tiny. For example, an increase of 1,000 € would correspond to about 0.12 points higher life satisfaction.

In a report, you would summarize the model like this:

In this multiple linear regression, we predicted overall life satisfaction (bpsy01) from PhD status, doctoral discipline, emotional support, scientific pressure, and income.

After controlling for all other variables, emotional support is positively associated with life satisfaction, and scientific pressure is negatively associated with life satisfaction; both effects are statistically significant and of moderate size. Income shows a statistically significant but very small positive effect. Differences between PhD status groups and doctoral disciplines are small and not statistically significant once support, burden and income are taken into account.

**4. Use the `glance()` function to get an overview of the model fit:

```
lm_phd_fit <- glance(lm_phd)
lm_phd_fit
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC
##       <dbl>         <dbl> <dbl>     <dbl>    <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1     0.103        0.0966  1.79      16.1 2.52e-35    13 -3671. 7372. 7455.
## # i 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

Important values:

- *r.squared*: proportion of variance in life satisfaction explained by the model
- *adj.r.squared*: adjusted R^2 (corrected for number of predictors)
- *p.value* (for the overall F-test): tests if the model explains more variance than a model with no predictors*

How would you interpret the results?

Solution:

You got $R^2 = 0.103$, adjusted $R^2 = 0.097$, F-statistic = 16.1, $p < .001$ (2.52e-35)

The regression model with PhD status, discipline, emotional support, burden and income explains **about 10% of the variance** in overall life satisfaction. The overall model is statistically significant.

2.4. Multiple linear regression: life satisfaction, nested models

In the next step, we build a **set of nested regression models**. This means that each new model contains all predictors from the previous model and then adds one block of additional variables. By comparing these models, we can see **how the explained variance changes** and whether certain predictors remain important **after controlling for** other factors.

Model 1: PhD and work conditions

In Model 1, we predict life satisfaction (`bpsy01`) from PhD-related characteristics and work conditions. The predictors are PhD status (`bdbi01`), doctoral discipline (`adbi15`), emotional support during the PhD (`bdcd17`), scientific pressure (`bdwr12`), and monthly income (`bemp81`). This model answers the question: *How are life satisfaction and PhD/work conditions related, without controlling for attitudes, well-being or demographics?*

Model 2: + attitudes and well-being

Model 2 extends Model 1 by adding attitudes and well-being variables: satisfaction with work–life balance (`blcd12`), relationship with parents (`apar15b`), and self-efficacy (`bpsy05`). All predictors from Model 1 remain in the model. This model asks: *Do PhD/work conditions still matter for life satisfaction after we also account for attitudes and well-being?*

Model 3: Full model with demographics

Model 3 is the full model. It includes all predictors from Model 2 and adds demographic controls: gender (`adem01`), age in years (`adem02`), parents' highest vocational degree (`apar10`), country of birth (`adem03`), and relationship status (`blcd01`). This model answers: *Which predictors are associated with life satisfaction when we simultaneously control for PhD/work conditions, attitudes and well-being, and basic demographic characteristics?*

1. Create a cleaned data frame `mydata_lm_ex2` that that has no missing values on any variable used in the three models. This way all models are based on the same set of cases.

Solution:

```
mydata_lm_ex2 <- mydata4 %>%
  filter(
    !is.na(bpsy01),
    !is.na(bdbi01),
    !is.na(adbi15),
    !is.na(bdcd17),
    !is.na(bdwr12),
    !is.na(bemp81),
    !is.na(blcd12),
    !is.na(apar15b),
    !is.na(bpsy05),
    !is.na(adem01),
    !is.na(adem02),
    !is.na(apar10),
    !is.na(adem03),
    !is.na(blcd01),
    !is.na(blcd06)
  )
```

2. First, fit a simple model with *only PhD and work condition predictors* like in exercise 2.3.

Solution:

```r
lm_ex2_m1 <- lm( # save results in new data frame for model 1
  bpsy01 ~ bdbi01 + adbi15 + bdcd17 + bdwr12 + bemp81,
  data = mydata_lm_ex2
)

lm_ex2_m1_results <- tidy(lm_ex2_m1, conf.int = TRUE) %>%    # coefficients + CI
  mutate(across(where(is.numeric), ~ round(.x, 3)))      # round cols to 3 decimals

lm_ex2_m1_results
```

```
## # A tibble: 14 x 7
##    term                   estimate std.error statistic p.value conf.low conf.high
##    <chr>                     <dbl>     <dbl>     <dbl>   <dbl>    <dbl>     <dbl>
##  1 (Intercept)               6.42      0.22     29.2     0         5.99      6.85
##  2 bdbi01I have complet~     0.095     0.14      0.678   0.498    -0.179     0.368
##  3 bdbi01I have interru~     0.121     0.279     0.432   0.666    -0.427     0.669
##  4 bdbi01I have quit my~    -0.098     0.487    -0.201   0.84     -1.05      0.857
##  5 adbi15sports              0.306     0.468     0.655   0.513    -0.611     1.22
##  6 adbi15law, economics~     0.04      0.153     0.261   0.794    -0.261     0.341
##  7 adbi15mathematics, n~    -0.023     0.144    -0.159   0.874    -0.306     0.26
##  8 adbi15human medicine~     0.286     0.171     1.68    0.094    -0.049     0.622
##  9 adbi15agricultural, ~     0.018     0.299     0.06    0.952    -0.569     0.605
## 10 adbi15engineering sc~    -0.097     0.166    -0.583   0.56     -0.423     0.229
## 11 adbi15art, art theory     0.274     0.327     0.838   0.402    -0.367     0.915
## 12 bdcd17                    0.338     0.036     9.30    0         0.267     0.409
## 13 bdwr12                   -0.386     0.054    -7.13    0        -0.493    -0.28
## 14 bemp81                    0         0         4.87    0         0         0
```

```r
lm_ex2_m1_fit <- glance(lm_ex2_m1) %>%             # model fit stats (R², etc.)
  mutate(across(where(is.numeric), ~ round(.x, 3)))# round numeric cols

lm_ex2_m1_fit
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squared sigma statistic p.value    df logLik   AIC   BIC
##       <dbl>         <dbl> <dbl>     <dbl>   <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1     0.103         0.096  1.80      14.7       0    13 -3369. 6768. 6850.
## # i 3 more variables: deviance <dbl>, df.residual <dbl>, nobs <dbl>
```

3. Next, extend the model by adding *attitudes and well-being* variables. This model contains all predictors from Model 1 *plus* the new ones.

What changes compared to Model 1? What do the new predictors in Model 2 tell us?

Solution:

```r
lm_ex2_m2 <- lm(
  bpsy01 ~ bdbi01 + adbi15 + bdcd17 + bdwr12 + bemp81 +
            blcd12 + apar15b + bpsy05,
  data = mydata_lm_ex2
)

lm_ex2_m2_results <- tidy(lm_ex2_m2, conf.int = TRUE) %>%
  mutate(across(where(is.numeric), ~ round(.x, 3)))

lm_ex2_m2_results
```

```
## # A tibble: 17 x 7
##    term                  estimate std.error statistic p.value conf.low conf.high
##    <chr>                    <dbl>     <dbl>     <dbl>   <dbl>    <dbl>     <dbl>
##  1 (Intercept)              0.96      0.363     2.64    0.008    0.247     1.67
##  2 bdbi01I have complet~   -0.125     0.122    -1.02    0.305   -0.364     0.114
##  3 bdbi01I have interru~    0.249     0.243     1.03    0.304   -0.227     0.726
##  4 bdbi01I have quit my~   -0.473     0.423    -1.12    0.264   -1.30      0.358
##  5 adbi15sports            -0.047     0.407    -0.114   0.909   -0.844     0.751
##  6 adbi15law, economics~   -0.018     0.133    -0.137   0.891   -0.28      0.243
##  7 adbi15mathematics, n~    0.059     0.125     0.472   0.637   -0.187     0.305
##  8 adbi15human medicine~    0.376     0.149     2.53    0.012    0.084     0.668
##  9 adbi15agricultural, ~    0.018     0.26      0.068   0.946   -0.492     0.527
## 10 adbi15engineering sc~   -0.248     0.144    -1.72    0.086   -0.531     0.035
## 11 adbi15art, art theory    0.383     0.284     1.35    0.178   -0.174     0.94
## 12 bdcd17                   0.213     0.032     6.66    0         0.151     0.276
## 13 bdwr12                  -0.084     0.049    -1.73    0.084   -0.18      0.011
## 14 bemp81                   0         0         4.66    0         0         0
## 15 blcd12                   0.33      0.017    19.1     0         0.296     0.364
## 16 apar15b                  0.121     0.054     2.24    0.025     0.015     0.226
## 17 bpsy05                   0.655     0.061    10.6     0         0.534     0.775
```

```
lm_ex2_m2_fit <- glance(lm_ex2_m2) %>%
  mutate(across(where(is.numeric), ~ round(.x, 3)))

lm_ex2_m2_fit
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squared sigma statistic p.value    df logLik   AIC   BIC
##       <dbl>         <dbl> <dbl>     <dbl>   <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1     0.325         0.318  1.56      50.0       0    16 -3130. 6297. 6394.
## # i 3 more variables: deviance <dbl>, df.residual <dbl>, nobs <dbl>
```

In Model 2 (now including attitudes & well-being) **you now see**

- bdcd17 (support) +0.21, still highly significant
- bdwr12 (pressure) –0.08, turns insignificant
- bemp81 (income) no longer significant (p .08, CI crosses 0)
- adbi15 (health sciences) +0.38, highly significant on 1% level

In Model 2, where we additionally control for attitudes and well-being, the effect for emotional support (bdcd17) decreases but remains significant. This suggests that part of their association with life satisfaction is shared with attitudes/well-being, but they still have their own independent contribution.

In contrast, the effects of scientific pressure (bdwr12) and income (bemp81) are no longer significant (the confidence interval includes 0), suggesting that income differences do not explain additional variance in life satisfaction beyond the other predictors in this model.

Finally, for doctoral discipline (adbi15), students in human medicine / health sciences report on average about 0.38 points higher life satisfaction than students in the reference discipline, and this difference becomes statistically significant at the 1% level, even after controlling for support, pressure, income, and attitudes.

What do the **new predictors** in Model 2 tell us?

After adding attitudes and well-being, work–life balance and self-efficacy emerge as strong positive predictors of life satisfaction, while the relationship with parents shows a smaller but still significant positive effect.

Overall, the extended model explains about one third of the variance in life satisfaction, which is clearly more than the PhD/work-conditions-only model.

4. Finally, build a *full model* that also includes *demographic controls*. Again, we keep all predictors from Model 2 and add the demographic variables.

What **changes compared** to Model 2?

Solution:

```
lm_ex2_m3 <- lm(
  bpsy01 ~ bdbi01 + adbi15 + bdcd17 + bdwr12 + bemp81 +
            blcd12 + apar15b + bpsy05 +
            adem01 + adem02 + apar10 + adem03 + blcd01 + blcd06,
  data = mydata_lm_ex2
)

lm_ex2_m3_results <- tidy(lm_ex2_m3, conf.int = TRUE) %>%
  mutate(across(where(is.numeric), ~ round(.x, 3)))

lm_ex2_m3_results
```

```
## # A tibble: 26 x 7
##    term                  estimate std.error statistic p.value conf.low conf.high
##    <chr>                    <dbl>     <dbl>     <dbl>   <dbl>    <dbl>     <dbl>
##  1 (Intercept)              3.01      0.479     6.27    0         2.07      3.95
##  2 bdbi01I have complet~   -0.168     0.12     -1.41    0.16     -0.403     0.066
##  3 bdbi01I have interru~    0.132     0.238     0.556   0.578    -0.335     0.599
##  4 bdbi01I have quit my~   -0.423     0.414    -1.02    0.307    -1.24      0.389
##  5 adbi15sports            -0.175     0.398    -0.44    0.66     -0.955     0.605
##  6 adbi15law, economics~   -0.096     0.131    -0.734   0.463    -0.354     0.161
##  7 adbi15mathematics, n~   -0.03      0.127    -0.237   0.812    -0.28      0.219
##  8 adbi15human medicine~    0.269     0.151     1.79    0.074    -0.026     0.564
##  9 adbi15agricultural, ~   -0.043     0.255    -0.166   0.868    -0.543     0.458
## 10 adbi15engineering sc~   -0.344     0.145    -2.37    0.018    -0.628    -0.059
## # i 16 more rows
```

```
lm_ex2_m3_fit <- glance(lm_ex2_m3) %>%
  mutate(across(where(is.numeric), ~ round(.x, 3)))

lm_ex2_m3_fit
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squared sigma statistic p.value    df logLik   AIC   BIC
##       <dbl>         <dbl> <dbl>     <dbl>   <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1     0.362         0.352  1.52      37.6       0    25 -3083. 6220. 6366.
## # i 3 more variables: deviance <dbl>, df.residual <dbl>, nobs <dbl>
```

```
print(lm_ex2_m3_results, n = Inf)  # show all rows
```

```
## # A tibble: 26 x 7
##    term                  estimate std.error statistic p.value conf.low conf.high
##    <chr>                    <dbl>     <dbl>     <dbl>   <dbl>    <dbl>     <dbl>
##  1 (Intercept)              3.01      0.479     6.27    0         2.07      3.95
```

```
##  2 bdbi01I have complet~   -0.168     0.12     -1.41    0.16    -0.403     0.066
##  3 bdbi01I have interru~    0.132     0.238     0.556   0.578   -0.335     0.599
##  4 bdbi01I have quit my~   -0.423     0.414    -1.02    0.307   -1.24      0.389
##  5 adbi15sports            -0.175     0.398    -0.44    0.66    -0.955     0.605
##  6 adbi15law, economics~   -0.096     0.131    -0.734   0.463   -0.354     0.161
##  7 adbi15mathematics, n~   -0.03      0.127    -0.237   0.812   -0.28      0.219
##  8 adbi15human medicine~    0.269     0.151     1.79    0.074   -0.026     0.564
##  9 adbi15agricultural, ~   -0.043     0.255    -0.166   0.868   -0.543     0.458
## 10 adbi15engineering sc~   -0.344     0.145    -2.37    0.018   -0.628    -0.059
## 11 adbi15art, art theory    0.283     0.278     1.02    0.309   -0.262     0.828
## 12 bdcd17                   0.188     0.032     5.93    0        0.126     0.25
## 13 bdwr12                  -0.102     0.048    -2.12    0.034   -0.196    -0.008
## 14 bemp81                   0         0         4.82    0        0         0
## 15 blcd12                   0.332     0.017    19.6     0        0.298     0.365
## 16 apar15b                  0.086     0.053     1.63    0.103   -0.017     0.19
## 17 bpsy05                   0.645     0.06     10.7     0        0.527     0.763
## 18 adem01male              -0.08      0.081    -0.989   0.323   -0.24      0.079
## 19 adem01other              0.076     0.172     0.443   0.658   -0.262     0.414
## 20 adem02                  -0.036     0.008    -4.76    0       -0.051    -0.021
## 21 apar10Bachelor / Mas~    0.174     0.112     1.55    0.121   -0.046     0.394
## 22 apar10Other vocation~    0.116     0.114     1.02    0.309   -0.107     0.339
## 23 apar10Unclassifiable     0.126     0.26      0.484   0.629   -0.385     0.636
## 24 adem03in another cou~   -0.093     0.107    -0.866   0.387   -0.304     0.118
## 25 blcd01no                -0.527     0.091    -5.78    0       -0.705    -0.348
## 26 blcd06no                -0.647     0.111    -5.84    0       -0.864    -0.43
```

In Model 3 (now including demographics) **you now see**

- bdwr12 (pressure) –0.10, turns significant again
- bemp81 (income) tiny effect, but significant again
- adbi15 (health sciences) +0.30, highly significant on 1% level
- adbi15 (engineering sciences) –0.32, p .03, significant negative effect
- apar15b (relationship with parents) turns insignificant

In the full model, pressure (bdwr12) shows a small but now statistically significant negative effect (about –0.10): higher pressure is still associated with slightly lower life satisfaction, even after adding demographic controls. The effect of income (bemp81) remains tiny in size but becomes statistically significant again, which means that income differences are detectable in the model, although they are not practically very important.

For doctoral discipline (adbi15), the positive effect for human medicine / health sciences stays clearly visible (about +0.30, significant at the 1% level), while engineering sciences show now a moderate negative effect (about –0.32, p .03), indicating lower life satisfaction compared to the reference discipline.

Finally, the effect of relationship with parents (apar15b) becomes statistically non-significant in the full model, suggesting that once demographics are included, this variable no longer explains additional variance in life satisfaction beyond the other predictors.

What does the **full model including demographics** in Model 3 tell us?

In the full model, most demographic variables (gender, parental education, country of birth) show no clear independent effect on life satisfaction. Age has only a very small positive association.

One demographic variable with a substantial and significant effect is relationship status: being not in a relationship is associated with clearly lower life satisfaction, even after controlling for PhD/work conditions, attitudes, well-being and discipline. A similar significant negative association can be found for parenthood.

Adding demographics increases the explained variance from about 33% to 36%.

5. Compute which model fits best.

Do it like this:

```
model_summaries <- bind_rows(
  glance(lm_ex2_m1) %>% mutate(model = "Model 1"),
  glance(lm_ex2_m2) %>% mutate(model = "Model 2"),
  glance(lm_ex2_m3) %>% mutate(model = "Model 3")
) %>%
  mutate(across(where(is.numeric), ~ round(.x, 3)))

model_summaries
```

```
## # A tibble: 3 x 13
##   r.squared adj.r.squared sigma statistic p.value    df logLik   AIC   BIC
##       <dbl>         <dbl> <dbl>     <dbl>   <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1     0.103         0.096  1.80      14.7       0    13 -3369. 6768. 6850.
## 2     0.325         0.318  1.56      50.0       0    16 -3130. 6297. 6394.
## 3     0.362         0.352  1.52      37.6       0    25 -3083. 6220. 6366.
## # i 4 more variables: deviance <dbl>, df.residual <dbl>, nobs <dbl>,
## #   model <chr>
```

Solution:

Model 1 → Model 2 (big improvement)

- Adjusted R^2 increases from 0.10 to 0.32: Attitudes and well-being explain a lot of additional variance in life satisfaction.
- AIC and BIC both decrease strongly (6772 → 6300 / 6853 → 6397) → Model 2 fits the data better than Model 1.

Model 2 → Model 3 (small extra improvement)

- Adjusted R^2 increases further from 0.318 to 0.352: Adding demographics gives some additional explanatory power, but much less than the step from Model 1 to Model 2.
- AIC and BIC go down again (6300 → 6255; 6397 → 6396) → the full model is statistically preferred, but the gain in fit is modest.

Thus, the **full model fits best**, but most of the explanatory power already comes from work conditions and especially attitudes and well-being.

2.5. Multiple logistic regression: probability of having children <- full model

We want to predict the probability of **having children (yes/no)** (`blcd06`) as a function of PhD and work conditions, attitudes and well-being and demographics.

We estimate the model with the `glm()` function and use `tidy()` to get odds ratios.

1. Create a cleaned data frame `mydata_lm_ex3` that only contains rows with non-missing values for the dependent variable and all predictors in this model. Solution:

```r
mydata_glm_ex3 <- mydata4 %>%
  filter(
    !is.na(bpsy01),
    !is.na(bdbi01),
    !is.na(adbi15),
    !is.na(bdcd17),
    !is.na(bdwr12),
    !is.na(bemp81),
    !is.na(blcd12),
    !is.na(apar15b),
    !is.na(bpsy05),
    !is.na(adem01),
    !is.na(adem02),
    !is.na(apar10),
    !is.na(adem03),
    !is.na(blcd01),
    !is.na(blcd06)
  )
```

2. Fit the full logistic regression model on having children (yes/no) `blcd06`.

To model the log-odds of `blcd06` for "yes" with reference category "no", we have to relevel our dependent variable with `relevel()`.

The code goes like this:

```r
glm_full <- glm(
  relevel(blcd06, ref = "no") ~                  # change reference category
    bdbi01 + adbi15 + bdcd17 + bdwr12 + bemp81 + # predictors
    blcd12 + apar15b + bpsy05 + bpsy01 +
    adem01 + adem02 + apar10 + adem03 + blcd01,
  data   = mydata_glm_ex3,                       # take dataset without missings
  family = binomial(link = "logit")              # logistic regression
)
```

3. Tidy the table and display odds ratios.

The code should look like this:

```r
glm_ex3_results <- tidy(
  glm_full,
  exponentiate = TRUE, # exp(coef) = odds ratios instead of log-odds
  conf.int     = TRUE  # 95% confidence intervals
) %>%
  mutate(across(where(is.numeric), ~ round(.x, 3))) # round numeric values to three decimals for
   ↪ readability


glm_ex3_results
```

```
## # A tibble: 26 x 7
##    term                   estimate std.error statistic p.value conf.low conf.high
##    <chr>                     <dbl>     <dbl>     <dbl>   <dbl>    <dbl>     <dbl>
##  1 (Intercept)               0         0.955    -8.25    0        0         0.002
##  2 bdbi01I have complet~     1.40      0.222     1.52    0.128    0.902     2.15
##  3 bdbi01I have interru~     2.36      0.38      2.26    0.024    1.09      4.90
##  4 bdbi01I have quit my~     2.67      0.742     1.32    0.186    0.569    10.9
```

```
##  5 adbi15sports             1.74      0.724     0.761   0.446    0.353     6.54
##  6 adbi15law, economics~    1.04      0.249     0.149   0.881    0.638     1.70
##  7 adbi15mathematics, n~    0.829     0.252    -0.744   0.457    0.507     1.36
##  8 adbi15human medicine~    1.38      0.295     1.10    0.272    0.774     2.47
##  9 adbi15agricultural, ~    0.396     0.629    -1.47    0.141    0.1       1.23
## 10 adbi15engineering sc~    1.46      0.277     1.37    0.17     0.851     2.52
## # i 16 more rows
```

```
print(glm_ex3_results, n = Inf)  # show all rows
```

```
## # A tibble: 26 x 7
##    term                  estimate std.error statistic p.value conf.low conf.high
##    <chr>                    <dbl>     <dbl>     <dbl>   <dbl>    <dbl>     <dbl>
##  1 (Intercept)              0         0.955    -8.25    0        0         0.002
##  2 bdbi01I have complet~    1.40      0.222     1.52    0.128    0.902     2.15
##  3 bdbi01I have interru~    2.36      0.38      2.26    0.024    1.09      4.90
##  4 bdbi01I have quit my~    2.67      0.742     1.32    0.186    0.569    10.9
##  5 adbi15sports             1.74      0.724     0.761   0.446    0.353     6.54
##  6 adbi15law, economics~    1.04      0.249     0.149   0.881    0.638     1.70
##  7 adbi15mathematics, n~    0.829     0.252    -0.744   0.457    0.507     1.36
##  8 adbi15human medicine~    1.38      0.295     1.10    0.272    0.774     2.47
##  9 adbi15agricultural, ~    0.396     0.629    -1.47    0.141    0.1       1.23
## 10 adbi15engineering sc~    1.46      0.277     1.37    0.17     0.851     2.52
## 11 adbi15art, art theory    1.58      0.45      1.02    0.306    0.644     3.79
## 12 bdcd17                   1.12      0.064     1.71    0.087    0.985     1.27
## 13 bdwr12                   0.982     0.093    -0.191   0.848    0.817     1.18
## 14 bemp81                   1         0        -2.41    0.016    1         1
## 15 blcd12                   0.857     0.037    -4.15    0        0.796     0.922
## 16 apar15b                  1.05      0.108     0.441   0.659    0.85      1.30
## 17 bpsy05                   0.831     0.129    -1.44    0.15     0.646     1.07
## 18 bpsy01                   1.38      0.056     5.80    0        1.24      1.54
## 19 adem01male               1.05      0.165     0.308   0.758    0.762     1.45
## 20 adem01other              1.06      0.335     0.186   0.852    0.536     2.01
## 21 adem02                   1.21      0.016    11.7     0        1.17      1.25
## 22 apar10Bachelor / Mas~    0.677     0.217    -1.8     0.072    0.445     1.04
## 23 apar10Other vocation~    0.505     0.227    -3.01    0.003    0.325     0.791
## 24 apar10Unclassifiable     0.702     0.497    -0.711   0.477    0.252     1.80
## 25 adem03in another cou~    0.999     0.213    -0.004   0.997    0.652     1.51
## 26 blcd01no                 0.148     0.292    -6.55    0        0.08      0.254
```

In the output table, estimate now shows **odds ratios** (because we used exponentiate = TRUE).

Odds ratio $>$ 1: the predictor is associated with higher odds of having children (blcd06 = "yes").

Odds ratio $<$ 1: the predictor is associated with lower odds of having children.

The 95% confidence interval shows the uncertainty around each odds ratio; if it includes 1, the effect is not statistically significant at the 5% level.

How do you interpret the results?

Solution:

** Effects and effect sizes **

Interrupting the PhD (`bdbi01` I have interrupted my PhD/doctoral project): OR 2.36, p = .024. People who interrupted their PhD have about 2.4 times higher odds of having children than the reference group (ongoing

PhD), holding all other variables constant. Completing or quitting the PhD (other bdbi01 categories) are not clearly different from the reference group (their CIs include 1).

For discipline (`adbi15`) none of the odds ratios is statistically significant in this model (all $p > .05$).

For emotional support (bdcd17) and scientific pressure (bdwr12), there is no meaningful association with having children .

Income (`bemp81`): $p = .016$ but OR is essentially 1.00 (because income is measured in €, the per-Euro effect is tiny). The effect is statistically detectable, but substantively negligible: income does not strongly change the odds of having children in this model.

Satisfaction with work–life balance (`blcd12`): OR 0.86, $p < .001$ Each one-point increase in work–life balance satisfaction is associated with about a 14% reduction in the odds of having children. In other words: respondents who are more satisfied with their work–life balance are less likely to have children – which fits the idea that balancing work and family is harder when children are present.

No significant effect for relationship with parents (`apar15b`) and Self-efficacy (`bpsy05`)

Life satisfaction (`bpsy01`): OR 1.38, $p < .001$ For each additional point on the 0–10 life satisfaction scale, the odds of having children are about 38% higher, controlling for all other variables.

Looking and the demographical characteristics, gender (`adem01`), parents' vocational degree `apar10` and country of birth (`adem03`) don't have significant effects. Strong predictors are age and relationship status.

Age in years (`adem02`): OR 1.21, $p < .001$: Each additional year of age is associated with roughly 21% higher odds of having children.

Relationship status (`blcd01 no`): OR 0.15, $p < .001$: Respondents not in a relationship have only about 15% of the odds of having children compared with those in the reference category (in a relationship). T

Summary to write in a report:

In the full logistic regression, the odds of having children are mainly related to age, relationship status, life satisfaction, work–life balance, and to some extent having interrupted the PhD. Most other predictors – including gender, discipline, country of birth and several attitude variables – do not show clear independent effects once everything is considered simultaneously.

4. Finally, calculate the overall model fit including McFadden R^2.

The code should look like this:

```
glm_ex3_fit <- glance(glm_full)

glm_ex3_fit %>%
  mutate(
    mcfadden_r2 = 1 - deviance / null.deviance
  )
```

```
## # A tibble: 1 x 9
##   null.deviance df.null logLik   AIC   BIC deviance df.residual  nobs
##           <dbl>   <int>  <dbl> <dbl> <dbl>    <dbl>       <int> <int>
## 1         1518.    1681  -586. 1224. 1365.    1172.        1656  1682
## # i 1 more variable: mcfadden_r2 <dbl>
```

How do you interpret the model fit?

Solution:

The full logistic regression model has a McFadden pseudo-R^2 of about 0.23. Values between 0.2 and 0.4 are often interpreted as a good model fit in logistic regression. This means that our model substantially improves the prediction of who has children compared to an intercept-only model, even though there may be still a lot of unexplained variation.

Take Home checklist: Multivariate regression in R

#	Question / task	Useful functions / tools
1	What is your **outcome variable** and which type of regression do you need?	Check scale & distribution with `str()`, `summary()`, `frq()`, `skimr::skim()`, histograms / bar charts via `ggplot()`
2	Are predictors coded correctly (metric vs. factor, reference categories, NAs)?	Recode with `mutate()`, `across()`, `case_when()`, set NAs; convert labelled vars with `haven::as_factor()`, set factor order / reference with `forcats::fct_relevel()` or `relevel()`
3	Fit a **baseline model (Model 1)** with a small set of predictors	Linear: `lm(outcome ~ x1 + x2, data = ...)`; Logistic: `glm(I(outcome == "yes") ~ ..., family = binomial)`; get coefficients with `broom::tidy(model, conf.int = TRUE)`; model fit with `broom::glance(model)`
4	Build **extended / nested models** (Model 2, Model 3 …) by adding predictor blocks	Specify new formulas in `lm()` / `glm()`; summarize and compare fit with `bind_rows(glance(m1), glance(m2), glance(m3)) %>% mutate(across(where(is.numeric), round, 3))`
5	How do you **interpret coefficients**?	Linear: change in outcome per 1-unit change in predictor, "holding other variables constant"; Logistic: use `tidy(glm_model, exponentiate = TRUE, conf.int = TRUE)` for **odds ratios** and 95% CIs, check if CI includes 1
6	How well does the **model fit** the data?	For linear: use `glance()` → `r.squared`, `adj.r.squared`; for logistic: `glance()` → McFadden pseudo-R^2: `mutate(mcfadden_r2 = 1 - deviance / null.deviance)`
7	Can you **communicate the results** in simple sentences?	From `tidy()` and `glance()`: report direction and size of effects, p-values and confidence intervals, plus overall fit (R^2 / pseudo-R^2); use phrases like "holding other variables constant…" and focus on **substantive** importance, not only significance