PCA und/oder GLM mit Kollege Dollingers Daten

Kollege Hinrichs

2025-03-07

Inhaltsverzeichnis

## Ausgangspunkt

Dieses Skript enthält den Hinweis auf das Ziel.

> library(ggplot2)  
> library(factoextra)  
> library(FactoMineR)  
  
# Befehl für das angestrebte Modell mit Error  
# model1 <- glm(X8HeardQ5 ~  
# X30Age+X33Gender+X36Region+X37LUI+X39EOI.n,  
# data = df)  
# Error in y - mu : non-numeric argument to binary operator

Der Error deutet an, dass das Problem zunächst mal mit der Formatierung der einen Variable zu tun hat.

Wir wollen das also mal nachbauen. Zuerst die Pakete.

library(pacman)  
p\_load(rio, tidyverse, janitor, gt,  
 factoextra, FactoMineR)

Jetzt laden wir die Daten standardmäßig als df.

df <- import("data/Master3100.csv") %>%   
 clean\_names()

## Data cleaning

Als nächstes schauen wir, was zur Vorbereitung des Datensatzes zu tun ist. Dazu gehört, dass der Typus aller Spalten überprüft wird.

df %>% glimpse()

Rows: 1,048,574  
Columns: 49  
$ x1cool\_q1 <chr> "Strongly agree", "Strongly agree", "Strongly ag…  
$ x2multiling\_q2 <chr> "Definitely yes", "Probably yes", "Probably yes"…  
$ x3u\_multiling\_q58 <chr> "No", "No", "No", "Yes", "Yes", "Yes", "No", "No…  
$ x4comf\_non\_eng\_q13 <chr> "", "", "", "Extremely comfortable", "Moderately…  
$ x5media\_q14 <chr> "Extremely unimportant", "Important", "Extremely…  
$ x6priv\_q62 <chr> "Yes", "No", "That depends (please also explain …  
$ x7priv\_q62b <chr> "", "", "", "Depends on the size of the group & …  
$ x8heard\_q5 <chr> "Yes", "Yes", "No", "No", "Yes", "Not sure", "No…  
$ x9descr\_q63 <chr> "Yes, but I can't describe it well", "Yes, but I…  
$ x10know\_q6 <chr> "both", "zed", "zee", "zee", "both", "zee", "zed…  
$ x11spellimp\_q18 <chr> "Very unimportant", "Important", "Unimportant", …  
$ x12aca\_work\_q19 <chr> "Definitely not", "Probably yes", "Probably not"…  
$ x13distinct\_q4 <chr> "Definitely not", "Probably yes", "Definitely no…  
$ x14w\_esame\_q8 <chr> "Probably not", "Definitely not", "Probably yes"…  
$ x15cdn\_pos\_q12 <chr> "No", "Yes (please specify who):", "No", "Yes (p…  
$ x16most\_prest\_q10 <chr> "Singaporean English", "British English", "Briti…  
$ x17rec\_cdns\_q15 <chr> "Rarely", "Sometimes", "Sometimes", "Often", "So…  
$ x18software\_q17 <chr> "Yes", "Yes", "Yes", "No", "Yes", "Yes", "", "",…  
$ x19on\_dict\_q20 <chr> "Neither yes or no", "Yes", "No", "Yes", "Yes", …  
$ x20recog\_eh\_q21 <chr> "Definitely yes", "Probably yes", "Definitely ye…  
$ x21cdn\_sorry\_more\_q23 <chr> "Definitely yes", "Definitely yes", "Definitely …  
$ x22engl\_yes\_q67 <chr> "Disagree", "Disagree", "Agree", "Disagree", "Di…  
$ x23fr\_yes\_q67 <chr> "Disagree", "Disagree", "Neither disagree nor ag…  
$ x24indg\_yes\_q67 <chr> "Strongly agree", "Agree", "Agree", "Strongly ag…  
$ x25more\_e\_fq67 <chr> "Neither disagree nor agree", "Agree", "Disagree…  
$ x26cdn\_way\_q9 <chr> "Probably not", "Probably yes", "Probably yes", …  
$ x30age <chr> "15-19", "20-24", "15-19", "20-24", "20-24", "20…  
$ x31live\_prov <chr> "Canada", "Canada", "United States", "Canada", "…  
$ x32live\_land <chr> "British Columbia", "British Columbia", "Califor…  
$ x33gender <chr> "female", "female", "male", "non-binary", "non-b…  
$ x34edu\_comp <chr> "high school", "high school", "high school", "hi…  
$ x35degree\_year <chr> "Second year", "Fourth year+", "I am not current…  
$ x36region <chr> "BC", "BC", "BC", "US", "BC", "Singapore", "BC",…  
$ x37lui <chr> "0", "0", "0", "4", "0", "3", "0", "0", "0", "0"…  
$ x38dist\_cult <chr> "no", "yes (please specify):", "no", "yes (pleas…  
$ x39eoi\_n <chr> "", "12", "", "4", "", "8", "4", "", "", "", "3"…  
$ x40eoi\_calc <chr> "", "12", "", "4", "", "8", "4", "", "", "", "3"…  
$ eoi\_check <chr> "1", "1", "1", "1", "1", "1", "1", "1", "1", "1"…  
$ hood <chr> "", "somewhat", "", "only a little", "", "somewh…  
$ hood\_number <chr> "0", "2", "0", "1", "0", "2", "2", "0", "0", "0"…  
$ rel <chr> "", "somewhat", "", "not at all", "", "", "not a…  
$ rel\_number <chr> "0", "2", "0", "0", "0", "0", "0", "0", "0", "0"…  
$ comm <chr> "", "somewhat", "", "only a little", "", "somewh…  
$ comm\_number <chr> "0", "2", "0", "1", "0", "2", "1", "0", "0", "0"…  
$ eoi\_work <chr> "", "very much", "", "not at all", "", "somewhat…  
$ eoi\_work\_number <chr> "0", "3", "0", "0", "0", "2", "1", "0", "0", "0"…  
$ eoi\_online <chr> "", "very much", "", "somewhat", "", "somewhat",…  
$ eoi\_online\_number <chr> "0", "3", "0", "2", "0", "2", "0", "0", "0", "0"…  
$ x41soc\_media <chr> "4-6 hours", "1-2 hours", "4-6 hours", "2-4 hour…

Wir sehen in der Vorschau hier zwei wichtige Dinge:

* Der Datensatz ist urs lang, über 1 Million Zeilen, das ist schön.
* Es werden alle Zeilen als Typ “character” gelesen, das ist wahrscheinlich nicht das, was wir wollen.

### Probleme mit Spaltenklassifizierung beheben

Ich weiß nicht genau, woran das letztere Problem liegt, aber wir müssen jetzt einfach mal per Hand die falsch erkannten Spalten umkategorisieren. Wir machen eine Liste der Spalten, die als ganze Zahlen (integers) gelesen werden sollen:

read\_as\_num <- c('x37lui', "x39eoi\_n", "x40eoi\_calc",  
 "eoi\_check", "hood\_number", "rel\_number",  
 "comm\_number", "eoi\_work\_number",   
 "eoi\_online\_number")

Jetzt werden alle diese Spalten *par force* in numerische umgewandelt.

df <- df %>%  
 mutate(across(all\_of(read\_as\_num), as.integer))  
  
df %>% glimpse()

Rows: 1,048,574  
Columns: 49  
$ x1cool\_q1 <chr> "Strongly agree", "Strongly agree", "Strongly ag…  
$ x2multiling\_q2 <chr> "Definitely yes", "Probably yes", "Probably yes"…  
$ x3u\_multiling\_q58 <chr> "No", "No", "No", "Yes", "Yes", "Yes", "No", "No…  
$ x4comf\_non\_eng\_q13 <chr> "", "", "", "Extremely comfortable", "Moderately…  
$ x5media\_q14 <chr> "Extremely unimportant", "Important", "Extremely…  
$ x6priv\_q62 <chr> "Yes", "No", "That depends (please also explain …  
$ x7priv\_q62b <chr> "", "", "", "Depends on the size of the group & …  
$ x8heard\_q5 <chr> "Yes", "Yes", "No", "No", "Yes", "Not sure", "No…  
$ x9descr\_q63 <chr> "Yes, but I can't describe it well", "Yes, but I…  
$ x10know\_q6 <chr> "both", "zed", "zee", "zee", "both", "zee", "zed…  
$ x11spellimp\_q18 <chr> "Very unimportant", "Important", "Unimportant", …  
$ x12aca\_work\_q19 <chr> "Definitely not", "Probably yes", "Probably not"…  
$ x13distinct\_q4 <chr> "Definitely not", "Probably yes", "Definitely no…  
$ x14w\_esame\_q8 <chr> "Probably not", "Definitely not", "Probably yes"…  
$ x15cdn\_pos\_q12 <chr> "No", "Yes (please specify who):", "No", "Yes (p…  
$ x16most\_prest\_q10 <chr> "Singaporean English", "British English", "Briti…  
$ x17rec\_cdns\_q15 <chr> "Rarely", "Sometimes", "Sometimes", "Often", "So…  
$ x18software\_q17 <chr> "Yes", "Yes", "Yes", "No", "Yes", "Yes", "", "",…  
$ x19on\_dict\_q20 <chr> "Neither yes or no", "Yes", "No", "Yes", "Yes", …  
$ x20recog\_eh\_q21 <chr> "Definitely yes", "Probably yes", "Definitely ye…  
$ x21cdn\_sorry\_more\_q23 <chr> "Definitely yes", "Definitely yes", "Definitely …  
$ x22engl\_yes\_q67 <chr> "Disagree", "Disagree", "Agree", "Disagree", "Di…  
$ x23fr\_yes\_q67 <chr> "Disagree", "Disagree", "Neither disagree nor ag…  
$ x24indg\_yes\_q67 <chr> "Strongly agree", "Agree", "Agree", "Strongly ag…  
$ x25more\_e\_fq67 <chr> "Neither disagree nor agree", "Agree", "Disagree…  
$ x26cdn\_way\_q9 <chr> "Probably not", "Probably yes", "Probably yes", …  
$ x30age <chr> "15-19", "20-24", "15-19", "20-24", "20-24", "20…  
$ x31live\_prov <chr> "Canada", "Canada", "United States", "Canada", "…  
$ x32live\_land <chr> "British Columbia", "British Columbia", "Califor…  
$ x33gender <chr> "female", "female", "male", "non-binary", "non-b…  
$ x34edu\_comp <chr> "high school", "high school", "high school", "hi…  
$ x35degree\_year <chr> "Second year", "Fourth year+", "I am not current…  
$ x36region <chr> "BC", "BC", "BC", "US", "BC", "Singapore", "BC",…  
$ x37lui <int> 0, 0, 0, 4, 0, 3, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, …  
$ x38dist\_cult <chr> "no", "yes (please specify):", "no", "yes (pleas…  
$ x39eoi\_n <int> NA, 12, NA, 4, NA, 8, 4, NA, NA, NA, 3, NA, NA, …  
$ x40eoi\_calc <int> NA, 12, NA, 4, NA, 8, 4, NA, NA, NA, 3, NA, NA, …  
$ eoi\_check <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, …  
$ hood <chr> "", "somewhat", "", "only a little", "", "somewh…  
$ hood\_number <int> 0, 2, 0, 1, 0, 2, 2, 0, 0, 0, 1, 0, 0, 0, 0, 0, …  
$ rel <chr> "", "somewhat", "", "not at all", "", "", "not a…  
$ rel\_number <int> 0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …  
$ comm <chr> "", "somewhat", "", "only a little", "", "somewh…  
$ comm\_number <int> 0, 2, 0, 1, 0, 2, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, …  
$ eoi\_work <chr> "", "very much", "", "not at all", "", "somewhat…  
$ eoi\_work\_number <int> 0, 3, 0, 0, 0, 2, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, …  
$ eoi\_online <chr> "", "very much", "", "somewhat", "", "somewhat",…  
$ eoi\_online\_number <int> 0, 3, 0, 2, 0, 2, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, …  
$ x41soc\_media <chr> "4-6 hours", "1-2 hours", "4-6 hours", "2-4 hour…

Das sieht besser aus.

### Leere Zellen als NA darstellen

Es gibt in den character-Spalten ein paar leere Zellen, die sollten besser als NA gespeichert werden.

df <- df %>%   
 mutate(across(where(is.character), ~ na\_if(.x, "")))

## Versuchen, das Modell laufen zu lassen

Versuchen wir es mal mit dem ursprünglichen Skript.

model1 <- glm(x8heard\_q5 ~  
 x30age + x33gender + x36region + x37lui +  
 x39eoi\_n,  
 data = df)  
# Error in y - mu : non-numeric argument to binary operator

|  |
| --- |
| Des Pudels Kern |
| Das Kernproblem ist, dass die glm()-Funktion nicht mit nichtnumerischen, nichtfaktoriellen Variablen als Prädiktoren klarkommt. Also nicht nur die Spalten, die eh schon Zahlen enthielten, sondern auch Spalten vom Typ “character” müssen entweder numerisch oder Faktoren sein, damit glm() damit klar kommt. |

Die Fehlermeldung ist noch die selbe wie bei Kollege Dollinger. Wir überprüfen also mal, ob all die Variablen im Modell tatsächlich numerisch sind, denn das wird ja hier moniert.

## Weitere Problembehebung

Eine Überprüfung ergibt, dass von den 5 Variablen plus der Antwortvariable nur 2 numerisch sind, nämlich x37lui und x39eoi\_n.

Es müssen also die folgenden Variablen auf numerisch übersetzt werden:

* x30age
* x33gender
* x36region
* x8heard\_q5

|  |
| --- |
| Numerisch oder faktoriell? |
| Eine von diesen vier ist **inhärent numerisch und skalar**, das ist x30age. Die wollen wir in *numerische Werte* umwandeln. Die anderen drei, x33gender, x36region und x8heard\_q5, sind nicht inhärent numerisch geordnet, deshalb wandeln wir sie in *Faktoren* um. |

Schauen wir mal, wie viele verschiedene Werte die erste hat.

df %>%   
 count(x30age)

x30age n  
1 14 and under 12  
2 15-19 165  
3 20-24 472  
4 25-29 348  
5 30-34 423  
6 35-39 343  
7 40-44 331  
8 45-49 228  
9 50-54 174  
10 55-59 157  
11 60-64 117  
12 65-69 112  
13 70-74 45  
14 75-79 24  
15 80-84 11  
16 85-89 2  
17 x 2  
18 <NA> 1045608

Das sind 18 verschiedene Werte. Ganz automatisch kann man das nicht in numerische Werte umwandeln, also werden wir das re-coden mit der Funktion case\_match(). Genauer gesagt, wir erstellen eine neue Spalte age\_num, also die numerische Version von x30age.

df <- df %>%   
 mutate(  
 age\_num = case\_match(  
 x30age,  
 "14 and under" ~ 14,  
 "15-19" ~ 19,  
 "20-24" ~ 24,  
 "25-29" ~ 29,  
 "30-34" ~ 34,  
 "35-39" ~ 39,  
 "40-44" ~ 44,  
 "45-49" ~ 49,  
 "50-54" ~ 54,  
 "55-59" ~ 59,  
 "60-64" ~ 64,  
 "65-69" ~ 69,  
 "70-74" ~ 74,  
 "75-79" ~ 79,  
 "80-84" ~ 84,  
 "85-89" ~ 89,  
 "x" ~ NA  
 )  
 )

Und nun wandeln wir die beiden anderen in Faktoren um.

# Einen Vektor mit den Namen der vars definieren, die umgewandelt werden sollen  
vars\_to\_factor <- c("x33gender", "x36region", "x8heard\_q5")  
  
df <- df %>%  
 mutate(across(all\_of(vars\_to\_factor), as.factor))

Jetzt noch einmal kucken, ob die Umwandlung geklappt hat.

df %>%   
 select(  
 age\_num, x33gender, x36region, x8heard\_q5  
 ) %>%   
 glimpse()

Rows: 1,048,574  
Columns: 4  
$ age\_num <dbl> 19, 24, 19, 24, 24, 24, 24, 19, 19, 54, 24, 24, 24, 24, 24,…  
$ x33gender <fct> female, female, male, non-binary, non-binary, non-binary, m…  
$ x36region <fct> BC, BC, BC, US, BC, Singapore, BC, ON, BC, X, MB, AB, ON, B…  
$ x8heard\_q5 <fct> Yes, Yes, No, No, Yes, Not sure, No, Not sure, No, No, Yes,…

|  |
| --- |
| Hinweis |
| Der Variablentyp <dbl> ist einer von mehreren numerischen Typen. |

Ja, hat geklappt.

## Erneutes model fitting

Jetzt probieren wir es nochmal mit dem Modell, mit der neuen var age\_num und den jetzt in Faktoren umgewandelten vars x33gender, x36region und x8heard\_q5.

model2 <- glm(x8heard\_q5 ~  
 age\_num + x33gender + x36region + x37lui +  
 x39eoi\_n,  
 data = df)  
#Error in glm.fit(x = c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, : NA/NaN/Inf in 'y'

Geht immer noch nicht. Ich glaube, das liegt jetzt daran, dass die Antwortvariable x8heard\_q5 außer Yes und No auch noch andere Werte enthält.

## Probleme mit der Antwortvariable lösen

Insgesamt hat sie die folgenden Werte:

df %>%  
 count(x8heard\_q5)

x8heard\_q5 n  
1 No 1020  
2 Not sure 444  
3 x 2  
4 Yes 1494  
5 <NA> 1045614

Ich schlage vor, wir trennen uns für den Zweck dieses Modells von allen Zeilen, in der die response etwas anderes ist als Yes oder No.

Zuvor lassen wir die Werte in der Spalte kategorisch klein schreiben, also yes oder no anstatt Yes oder No.

df <- df %>%   
 mutate(x8heard\_q5 = tolower(x8heard\_q5))  
  
df\_filtered <- df %>%   
 filter(  
 x8heard\_q5 %in% c("yes", "no")  
 )

Der Datensatz ist so erheblich kleiner, nur ein fünfhundertstel des Ausgangs-Datensatzes.

## Model fitting mit reduziertem Datensatz

model3 <- glm(x8heard\_q5 ~  
 age\_num + x33gender + x36region + x37lui +  
 x39eoi\_n,  
 data = df\_filtered) # jetzt mit df\_filtered statt df  
# Error in y - mu : non-numeric argument to binary operator

Der Fehler besteht weiterhin.

|  |
| --- |
| Auch Du, y-Variable?! |
| Die Antwortvariable **muss**, nach weiterer Recherche, numerisch sein. Also werden wir sie ebenfalls re-coden. |

## Reparatur der Antwortvariable

df\_filtered <- df\_filtered %>%   
 mutate(  
 x8heard\_q5 = case\_match(  
 x8heard\_q5,  
 "yes" ~ 1,  
 "no" ~ 0  
 )  
 )  
  
  
df\_filtered %>% count(x8heard\_q5)

x8heard\_q5 n  
1 0 1020  
2 1 1494

## Vierter Versuch

model4 <- glm(x8heard\_q5 ~  
 age\_num + x33gender + x36region + x37lui +  
 x39eoi\_n,  
 data = df\_filtered)   
model4 %>% summary()

Call:  
glm(formula = x8heard\_q5 ~ age\_num + x33gender + x36region +   
 x37lui + x39eoi\_n, data = df\_filtered)  
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 0.5527636 0.0777618 7.108 3.15e-12 \*\*\*  
age\_num 0.0016343 0.0013891 1.176 0.2398   
x33gendermale 0.0187765 0.0397663 0.472 0.6370   
x33gendernon-binary 0.1368296 0.1202679 1.138 0.2557   
x33genderother (please type here) -0.1292640 0.1240198 -1.042 0.2977   
x36regionAUS -0.6044479 0.4905291 -1.232 0.2183   
x36regionAustralia -0.0022077 0.2880382 -0.008 0.9939   
x36regionAUSTRALIA -0.6597897 0.4899585 -1.347 0.1786   
x36regionAustria 0.3614102 0.4916847 0.735 0.4626   
x36regionBangladesh -0.6618892 0.4904417 -1.350 0.1776   
x36regionBC -0.0091264 0.0647564 -0.141 0.8880   
x36regionBotswana -0.6566212 0.4908714 -1.338 0.1815   
x36regionBrazil 0.3520636 0.4947920 0.712 0.4770   
x36regionChile -0.6089406 0.4910580 -1.240 0.2154   
x36regionChina 0.0777496 0.1927256 0.403 0.6868   
x36regionColumbia 0.3550162 0.4904822 0.724 0.4694   
x36regionFrance -0.6390730 0.4899763 -1.304 0.1926   
x36regionGER 0.3809480 0.4901835 0.777 0.4374   
x36regionGermany 0.3198281 0.4908404 0.652 0.5149   
x36regionGERMANY -0.6434056 0.3481677 -1.848 0.0651 .   
x36regionHK -0.6689345 0.4917275 -1.360 0.1742   
x36regionHong Kong -0.3073511 0.2892584 -1.063 0.2884   
x36regionIndia -0.4310770 0.2255494 -1.911 0.0564 .   
x36regionIndonesia 0.3805071 0.4924944 0.773 0.4400   
x36regionIraq -0.6571644 0.4943775 -1.329 0.1842   
x36regionKorea 0.3862055 0.4923899 0.784 0.4331   
x36regionKOREA -0.6752348 0.4920752 -1.372 0.1705   
x36regionMalaysia 0.3683229 0.4897297 0.752 0.4523   
x36regionMB -0.0593198 0.0735368 -0.807 0.4202   
x36regionMeXico -0.3919124 0.2534073 -1.547 0.1225   
x36regionNB 0.2198767 0.1902809 1.156 0.2483   
x36regionNetherlands 0.3156965 0.4912713 0.643 0.5207   
x36regionNL 0.0323445 0.1696628 0.191 0.8489   
x36regionNS 0.0887648 0.2512641 0.353 0.7240   
x36regionON 0.0532586 0.0636081 0.837 0.4027   
x36regionPE 0.3810666 0.4897366 0.778 0.4368   
x36regionPEI 0.0203319 0.2079149 0.098 0.9221   
x36regionPH 0.3839839 0.4903685 0.783 0.4339   
x36regionPHI -0.6593064 0.4902922 -1.345 0.1792   
x36regionPhilippines -0.1973495 0.1913722 -1.031 0.3028   
x36regionQC -0.0228051 0.1081008 -0.211 0.8330   
x36regionSaudia Arabia 0.3507358 0.4901369 0.716 0.4745   
x36regionSCT 0.3511359 0.4897633 0.717 0.4737   
x36regionSingapore -0.1153293 0.3477948 -0.332 0.7403   
x36regionSK 0.0319476 0.1223603 0.261 0.7941   
x36regionSouth Africa 0.3823252 0.4899510 0.780 0.4355   
x36regionTaiwan 0.3578806 0.4920237 0.727 0.4673   
x36regionThailand -0.6465098 0.4908961 -1.317 0.1883   
x36regionUganda -0.6149205 0.4904734 -1.254 0.2104   
x36regionUK -0.0721603 0.2262803 -0.319 0.7499   
x36regionUkraine -0.1375452 0.3526367 -0.390 0.6966   
x36regionUS -0.1419719 0.0971395 -1.462 0.1444   
x36regionVietnam -0.6384409 0.4932316 -1.294 0.1960   
x36regionX -0.0793237 0.0893344 -0.888 0.3749   
x37lui -0.0002815 0.0095759 -0.029 0.9766   
x39eoi\_n 0.0033970 0.0056638 0.600 0.5489   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
(Dispersion parameter for gaussian family taken to be 0.236902)  
  
 Null deviance: 164.28 on 693 degrees of freedom  
Residual deviance: 151.14 on 638 degrees of freedom  
 (1820 observations deleted due to missingness)  
AIC: 1025.7  
  
Number of Fisher Scoring iterations: 2

### Die gute Nachricht

Das Modell läuft jetzt.

### Die schlechte Nachricht

In den Levels mindestens eines Prädiktors zeigt sich dass noch mehr cleaning zu tun ist, es gibt z.B. mehrere Fälle, wo sowohl groß- als auch kleingeschriebene Versionen der Werte mit dabei sind, z.B. AUSTRALIA und australia.

Also, wir schreiben jetzt einfach alle nichtnumerischen Vars komplett klein.

vars\_to\_lower <- c("x33gender", "x36region")  
  
df\_filtered <- df\_filtered %>%   
 mutate(across(all\_of(vars\_to\_lower), tolower)) %>%   
 mutate(across(all\_of(vars\_to\_lower), as.factor))

## Fünfter Versuch

model5 <- glm(x8heard\_q5 ~  
 age\_num + x33gender + x36region + x37lui +  
 x39eoi\_n,  
 data = df\_filtered)   
model5 %>% summary()

Call:  
glm(formula = x8heard\_q5 ~ age\_num + x33gender + x36region +   
 x37lui + x39eoi\_n, data = df\_filtered)  
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 0.5541914 0.0778536 7.118 2.94e-12 \*\*\*  
age\_num 0.0015933 0.0013889 1.147 0.2517   
x33gendermale 0.0187242 0.0397963 0.471 0.6382   
x33gendernon-binary 0.1363370 0.1205713 1.131 0.2586   
x33genderother (please type here) -0.1288871 0.1243349 -1.037 0.3003   
x36regionaus -0.6047934 0.4918011 -1.230 0.2192   
x36regionaustralia -0.1652849 0.2512550 -0.658 0.5109   
x36regionaustria 0.3615577 0.4929610 0.733 0.4636   
x36regionbangladesh -0.6626442 0.4917185 -1.348 0.1783   
x36regionbc -0.0095779 0.0649236 -0.148 0.8828   
x36regionbotswana -0.6572393 0.4921464 -1.335 0.1822   
x36regionbrazil 0.3522853 0.4960694 0.710 0.4779   
x36regionchile -0.6087143 0.4923261 -1.236 0.2168   
x36regionchina 0.0767047 0.1932284 0.397 0.6915   
x36regioncolumbia 0.3542434 0.4917632 0.720 0.4716   
x36regionfrance -0.6384438 0.4912525 -1.300 0.1942   
x36regionger 0.3815345 0.4914566 0.776 0.4378   
x36regiongermany -0.3220741 0.2865264 -1.124 0.2614   
x36regionhk -0.6687011 0.4930129 -1.356 0.1755   
x36regionhong kong -0.3067104 0.2899992 -1.058 0.2906   
x36regionindia -0.4313669 0.2261323 -1.908 0.0569 .   
x36regionindonesia 0.3791507 0.4937758 0.768 0.4429   
x36regioniraq -0.6576424 0.4956669 -1.327 0.1851   
x36regionkorea -0.1444483 0.3517759 -0.411 0.6815   
x36regionmalaysia 0.3680386 0.4910082 0.750 0.4538   
x36regionmb -0.0593196 0.0737279 -0.805 0.4214   
x36regionmexico -0.3915598 0.2540552 -1.541 0.1238   
x36regionnb 0.2197684 0.1907782 1.152 0.2498   
x36regionnetherlands 0.3171479 0.4925453 0.644 0.5199   
x36regionnl 0.0324822 0.1701056 0.191 0.8486   
x36regionns 0.0877968 0.2519016 0.349 0.7276   
x36regionon 0.0531852 0.0637705 0.834 0.4046   
x36regionpe 0.3808792 0.4910142 0.776 0.4382   
x36regionpei 0.0203900 0.2084403 0.098 0.9221   
x36regionph 0.3836721 0.4916436 0.780 0.4355   
x36regionphi -0.6589514 0.4915739 -1.340 0.1806   
x36regionphilippines -0.1973966 0.1918662 -1.029 0.3040   
x36regionqc -0.0225670 0.1083791 -0.208 0.8351   
x36regionsaudia arabia 0.3508517 0.4914141 0.714 0.4755   
x36regionsct 0.3506737 0.4910383 0.714 0.4754   
x36regionsingapore -0.1155260 0.3487033 -0.331 0.7405   
x36regionsk 0.0317886 0.1226737 0.259 0.7956   
x36regionsouth africa 0.3823646 0.4912260 0.778 0.4366   
x36regiontaiwan 0.3584135 0.4933037 0.727 0.4678   
x36regionthailand -0.6474881 0.4921768 -1.316 0.1888   
x36regionuganda -0.6159752 0.4917532 -1.253 0.2108   
x36regionuk -0.0714901 0.2268500 -0.315 0.7528   
x36regionukraine -0.1370992 0.3535412 -0.388 0.6983   
x36regionus -0.1414675 0.0973851 -1.453 0.1468   
x36regionvietnam -0.6375410 0.4945087 -1.289 0.1978   
x36regionx -0.0794206 0.0895665 -0.887 0.3756   
x37lui -0.0004774 0.0095772 -0.050 0.9603   
x39eoi\_n 0.0035682 0.0056674 0.630 0.5292   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
(Dispersion parameter for gaussian family taken to be 0.2381423)  
  
 Null deviance: 164.28 on 693 degrees of freedom  
Residual deviance: 152.65 on 641 degrees of freedom  
 (1820 observations deleted due to missingness)  
AIC: 1026.5  
  
Number of Fisher Scoring iterations: 2

## Hübschere Visualisierung der Modell-Koeffizienten

Dafür gibts ja verschiedene Pakete, zum Bsp. {prettyglm}, {sjplot}, {stargazer} und andere.