4. Übungsblatt (Faktorielle Umfrageexperimente)

Daria Tisch

1 Organisation

1.1 Arbeitsverzeichnis festsetzen

1.2 Packages installieren und laden

```
# Packages
pkgs <- c(
  "tidyverse",
  "sjPlot",
  "haven",
  "labelled"
)

## Install uninstalled packages
lapply(pkgs[!(pkgs %in% installed.packages())], install.packages)

## Load all packages to library
lapply(pkgs, library, character.only = TRUE)
```

1.3 Einführung

In dieser Übung replizieren wir die Ergebnisse folgender Studie:

Tisch, Daria, and Tamara Gutfleisch. 'Unequal but Just? Experimental Evidence on Distributive Justice Principles in Parental Inter Vivos Transfers'. Socio-Economic Review 21, no. 3 (2023): 1369–90.

Der Datensatz kann hier heruntergeladen werden.

Schenkungen von Eltern an deren Kinder sind von der moralischen Entscheidung geprägt, welches Kind wie viel erhalten soll. So können Eltern Schenkungen zwischen Kindern nach unterschiedlichen Gerechtigkeitsprinzipien aufteilen. Wenden sie das Gleichheitsprinzip an, schenken sie allen Kindern gleich viel. Wenden sie das Bedürfnisprinzip an, schenken sie den Kindern, die größere Bedürfnisse haben (z. B. Arbeitslosigkeit), mehr. Wenden sie das Austauschprinzip an, schenken sie den Kindern mehr, die im Gegenzug mehr für die Eltern machen (z. B. im Haushalt der Eltern helfen). Wenden sie das Anspruchsprinzip an, schenken sie den Kindern mehr, die bestimmte angeborene Statuscharakteristiken haben (z. B. Erstgeborene). Wir wollen untersuchen, inwiefern diese Prinzipien im Kontext von elterlichen Schenkungen von den Befragten befürwortet werden. Unterstützen die Befragten diese Prinzipien im selben Maße für Töchter und Söhne?

1.4 Daten einlesen

```
# Load the data file
df <- read.csv("../daten/just_transfers.csv")
```

1.5 Variablenlabels einlesen

```
# Load the variable labels
variable_labels <- read.csv("../daten/just_transfers_variable_labels.csv", row.names = 1, st

# View variable labels
#print(variable_labels)


# Loop through the labels and assign them to variables in the dataset
for (var in names(variable_labels)) {
  if (var %in% names(df)) {
    var_label(df[[var]]) <- variable_labels[[var]]
  }
}
```

1.6 Value labels einlesen

```r
# Load the value labels
value_labels <- read.csv("../daten/value_labels.csv", stringsAsFactors = FALSE)

# View value labels
head(value_labels)
```

```
   variable               label value
1 firstborn       Son firstborn     1
2 firstborn               Twins     2
3 firstborn  Daughter firstborn     3
4      need      Son unemployed     1
5      need      Equal earnings     2
6      need Daughter unemployed     3
```

1.7 Datenstruktur betrachten

```r
look_for(df)
```

```
 pos variable        label                              col_type missing values
 1   X               -                                  int      0
 2   id_resp         ID of respondent                   int      0
 3   id_within       ID for each observation within in~ int      0
 4   id_vignette     ID of vignette                     int      0
 5   deck            Vignette deck                      int      0
 6   reihevig        Order of vignette                  int      0
 7   firstborn       Vignette: firstborn child          int      0
 8   need            Vignette: income                   int      0
 9   help            Vignette: help                     int      0
 10  daughter        Focal vignette person daughter     int      0
 11  g_need          Child's relative income            int      0
 12  g_help          Child's relative help              int      0
 13  g_firstborn     Child's relative age               int      0
 14  dau_vig         Vignette evaluation for daughter   int      2142
 15  son_vig         Vignette evaluation for son        int      2142
 16  child_vig       Child's amount of inter vivos      int      0
 17  iv_received     inter vivos received               int      18
 18  iv_parents      Inter vivos from parent            int      576
 19  iv_grandparents Inter vivos from granparent        int      576
 20  iv_siblings     Inter vivos from sibling           int      576
```

21	lastiv	Amount last inter vivos	int	672
22	siblings	Having siblings	int	36
23	brothers	Number of brothers	int	726
24	sisters	Number of sisters	int	726
25	r_firstborn	Respond. firstborn	int	726
26	child	Having children	int	42
27	n_sons	Number of sons	int	2166
28	n_daughters	Number of daughters	int	2166
29	gifted	Ever given an inter vivos transfer	int	2166
30	principles	According to which principle gift~	int	3390
31	rel_mo	Relationship to mother	int	708
32	rel_fa	Relationship to father	int	1128
33	rel_ch	Relationship to children	int	2208
34	hhparent	Lives with parent	int	540
35	hhhelp	Helps parent	int	864
36	relstatus	Relationship status	int	54
37	partner	Having a partner	int	54
38	partnerhh	Having a partner in household	int	1116
39	age	Age	int	54
40	deutsch	Born in Germany	int	54
41	mig	Migration background	int	60
42	female	Female	int	72
43	bundl	Federal state	int	60
44	abi	Abitur	int	144
45	employ	Employed	int	54
46	student	Student	int	54
47	unemp	Unemployed, active search	int	54
48	unemp_nosearch	Unemployed, no active search	int	54
49	disabled	Disabled	int	54
50	retired	Retired	int	54
51	house	Housework, care	int	54
52	r_income	Income categories	int	324
53	gideo1	House and children	int	84
54	gideo2	Being housewife	int	90
55	gideo3	Men: work, women: house and family	int	84
56	gideo4	Not good if men stay home	int	90
57	abi2	Abitur and Fachabitur	int	54

2 Fallzahlen

2.1 Wie viele Beobachtungen enthält der Datensatz?

```
nrow(df)
```

```
[1] 4284
```

2.2 Wie viele befragte Personen sind im Datensatz enthalten?

```
length(unique(df$id_resp))
```

```
[1] 714
```

2.3 Wie viele Vignetten hat jede befragte Person bewertet?

```
length(unique(df$id_within))
```

```
[1] 3
```

```
table(df$id_within)
```

```
   1    2    3
1428 1428 1428
```

```
1428/2
```

```
[1] 714
```

2.4 Wie viele Vignetten gibt es?

```r
table(df$id_vignette)
```

```
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20
134 134 134 162 162 162 180 180 180 168 168 168 146 146 146 174 174 174 154 154
 21  22  23  24  25  26  27
154 152 152 152 158 158 158
```

```r
length(unique(df$id_vignette))
```

```
[1] 27
```

2.5 Wie viele Decks gibt es?

```r
table(df$deck)
```

```
  1   2   3   4   5   6   7   8   9
402 486 540 504 438 522 462 456 474
```

```r
length(unique(df$deck))
```

```
[1] 9
```

3 Deskriptive Statistik

3.1 Replikation Tabelle 2

Nun wollen wir Tabelle 2 aus dem Artikel replizieren.

```r
# Filter für die Daten anwenden
filtered_data <- df %>%
  filter(id_within == 1, daughter == 1)

# Deskriptive Statistiken berechnen
```

```r
descriptive_stats <- filtered_data %>%
  summarise(
    Female_Mean = mean(female, na.rm = TRUE),
    Female_SD = sd(female, na.rm = TRUE),
    Female_Min = min(female, na.rm = TRUE),
    Female_Max = max(female, na.rm = TRUE),
    Female_N = sum(!is.na(female)),

    Age_Mean = mean(age, na.rm = TRUE),
    Age_SD = sd(age, na.rm = TRUE),
    Age_Min = min(age, na.rm = TRUE),
    Age_Max = max(age, na.rm = TRUE),
    Age_N = sum(!is.na(age)),

    Child_Mean = mean(child, na.rm = TRUE),
    Child_SD = sd(child, na.rm = TRUE),
    Child_Min = min(child, na.rm = TRUE),
    Child_Max = max(child, na.rm = TRUE),
    Child_N = sum(!is.na(child)),

    Mig_Mean = mean(mig, na.rm = TRUE),
    Mig_SD = sd(mig, na.rm = TRUE),
    Mig_Min = min(mig, na.rm = TRUE),
    Mig_Max = max(mig, na.rm = TRUE),
    Mig_N = sum(!is.na(mig)),

    Gifted_Mean = mean(gifted, na.rm = TRUE),
    Gifted_SD = sd(gifted, na.rm = TRUE),
    Gifted_Min = min(gifted, na.rm = TRUE),
    Gifted_Max = max(gifted, na.rm = TRUE),
    Gifted_N = sum(!is.na(gifted)),

    IV_Received_Mean = mean(iv_received, na.rm = TRUE),
    IV_Received_SD = sd(iv_received, na.rm = TRUE),
    IV_Received_Min = min(iv_received, na.rm = TRUE),
    IV_Received_Max = max(iv_received, na.rm = TRUE),
    IV_Received_N = sum(!is.na(iv_received)),

    ABI_Mean = mean(abi, na.rm = TRUE),
    ABI_SD = sd(abi, na.rm = TRUE),
    ABI_Min = min(abi, na.rm = TRUE),
    ABI_Max = max(abi, na.rm = TRUE),
```

```
    ABI_N = sum(!is.na(abi))
  )

# Umstrukturierung der Tabelle
descriptive_stats_long <- descriptive_stats %>%
  pivot_longer(
    cols = everything(),
    names_to = c("Variable", ".value"),
    names_pattern = "(.*)_(.*)"
  )

# Tabelle mit sjPlot anzeigen und exportieren
tab_df(
  descriptive_stats_long,
  title = "Descriptive Statistics of Respondent Characteristics",
  col.header = c("Variable", "Mean", "SD", "Min", "Max", "N")
)
```

Table 1: Descriptive Statistics of Respondent Characteristics

Variable	Mean	SD	Min	Max	N
Female	0.62	0.49	0	1	702
Age	44.83	15.08	21	73	705
Child	0.50	0.50	0	1	707
Mig	0.12	0.33	0	1	704
Gifted	0.66	0.47	0	1	353
IV_Received	0.87	0.34	0	1	711
ABI	0.86	0.35	0	1	690

4 Render

Wandle dieses Dokument in ein PDF und ein HTML Dokument um.

5 Weiterführende Literatur

- R for Data Science