German and Dutch Translations of the Artificial-Social-Agent Questionnaire Instrument for Evaluating Human-Agent Interactions

Correlation and Variation between English and Dutch ASA Questionnaire

Nele Albers, Boleslav A. Khodakov

18 June, 2024

Contents

Introduction

This document presents a statistical analyses of the correlation and variation between the English and Dutch ASA questionnaires for the item level, construct/dimension level, and short version of the ASA questionnaire. The code is based on the one by Li et al. (2023).

Required files: summative_first_half_transformed_dutch.sav, summative_second_half_transformed_dutch.sav.

Authored by Nele Albers, Andrea Bönsch, Jonathan Ehret, Boleslav A. Khodakov, and Willem-Paul Brinkman.

Load packages

Let's load the packages that we need.

```
library(BayesianFirstAid)  # Run Bayesian t-test
library(car)     # Package linear regression
library(dplyr) # Use select function
library(foreign) # Open various data files
library(formatR)  # For formatting
library(haven) # Use read_sav fuction
library(knitr) # Get markdown file
library(nlme)    # Run multilevel linear models
library(pander) # For pandering tables
library(rethinking) # Run ulam
panderOptions("table.alignment.default","left")
panderOptions('round',2)
```

Define constants

And let's define a few constants that we use throughout.

```
NUM_ITEMS_SHORT_ASAQ = 24
NUM_ITEMS_FULL_ASAQ = 90
NUM_PRECISION = 4  # precision for determining ICC classification
```

Load preprocessed data

The input data used in the analysis are the preprocessed data files 'summative_first_half_transformed_dutch.sav' and 'summative_second_half_transformed_dutch.sav'.

Let's first load the data from the first half.

```
# Load data for first half
d1 <- data.frame(read_sav("summative_first_half_transformed_dutch.sav"))
```

And we also load the data from the second half of the questionnaire.

```
# Load data for second half
d2 <- data.frame(read_sav("summative_second_half_transformed_dutch.sav"))
```

Analyses results

Correlation between English and Dutch ASA Questionnaire

ICC values for 90 items

We combined the scores of 44 items and 46 items as well as their corresponding translations in data frames 'd1' and 'd2'. Then we calculated ICC values for the 90 items. The multilevel model that we fit on the data set is a random intercept model. This model includes a fixed intercept (~1) and participant as a random intercept, indicated by random = ~1|id. Here, 'id' indicates the participant code for the bilingual participants whose scores were used to calculate ICC values. We calculated ICC as: $\rho_I = \frac{\tau^2}{\tau^2+\sigma^2}$ whereby τ^2 is the variance between participants, and σ^2 is the variance within the score of individual (Finch, Bolin, and Kelley 2019). For the ICC calculation we defined the *getICC* function.

```
getICC <-function(model)
# Function for ICC value calculation using multilevel linear model
{
  vc.model <- VarCorr(model)
```

```
  # Estimated variances and correlations between the random-effects terms
  sigma_var <-as.numeric(vc.model[2,1])
  # Variance within the groups
  tau_var <- as.numeric(vc.model[1,1])
  # Variance between the groups
  icc <- tau_var/(tau_var + sigma_var)
  # Calculate ICC value
  return(icc)
}
```

Data frames ‘d1’ and ‘d2’ both have 120 data points, which we combine in single data frame.

```
# Combine evaluation scores of 44 items and 46 items for all participants
d_total <- cbind(select(d1,Q_E_HLA1:Q_E_R_AE4), select(d2,Q_E_UE1:Q_E_UAI4),
           select(d1,Q_DU_HLA1:Q_DU_R_AE4),select(d2,Q_DU_UE1:Q_DU_UAI4))
```

Next, we defined a function to run a multilevel model and obtain the associated ICC value for that model. As input, this function accepts the scores in both languages and the participant ID number. Before the model can be fitted this input data is transformed into a long format. The function returns ICC in value.

```
getLME <-function(s_1,s_2)
# Function for a linear mixed-effects model
{
  # Row names that represent the ID number of each participant
  id<-rownames(s_2)
  # Transform Dutch scores from wide format to long format and label as 1
  score_Dutch<- data.frame(id, s_1, language= 1)
  # Transform English scores from wide format to long format and label as 2
  Score_English<- data.frame(id, s_2, language= 2)
  # Combine Dutch and English scores in the long format
  Score_total <- rbind(score_Dutch, Score_English)
  # Linear mixed-effects model with a fixed intercept and
  # a random intercept of participant's ID number
  m0 <- lme(score ~ 1, data = Score_total, random = ~1|id, method = "ML")
  return(getICC(m0))
}
```

With the $getLME$ function defined, the next step is to use this function to calculate the ICC value for each of the 90 ASA questionnaire items, and in addition, calculate the grand mean of these 90 ICC values. When going through the list of ASAQ items, we use the fact that in the data frame the first 90 columns present the results of the English ASAQ version and the last 90 columns present the results of the Dutch ASAQ version.

First we define a function to calculate the ICC values.

```
calculate_item_ICC_values <- function(data, n=NUM_ITEMS_FULL_ASAQ){

  l_ICC <- data.frame(ItemID = double(), Item = character(), icc = double())

  # Numbers of columns in d_total
  Dutch_column_offset <- ncol(data) /2

  # The value of n is equal to the number of columns divided by 2.
  for (i in 1:n)
  # Go step by step to 90 items of the ASA questionnaire,
  # whereby i is the ASA questionnaire item number
```

```
  {

    # Select scores of Dutch version of ASAQ item i
    score_Dutch <- data.frame(score=data[,i + Dutch_column_offset])

    # Select scores of English version of ASAQ items i
    score_English <- data.frame(score=data[,i])

    # Calculate ICC and add it to the list of ICC values,
    # with ID number of the ASA questionnaire item
    l_ICC <- rbind(l_ICC, data.frame(i, icc = getLME(score_Dutch, score_English)))

  }
  return(l_ICC)
}
```

And then we use this function to compute the ICC values.

```
l_ICC <- calculate_item_ICC_values(d_total)

l_ICC$Item = colnames(select(d_total,Q_E_HLA1:Q_E_UAI4)) # Add name code for each item
pander(l_ICC, caption = "All participants - ICC values for 90 items")
```

Table 1: All participants - ICC values for 90 items

i	icc	Item
1	0.78	Q_E_HLA1
2	0.39	Q_E_HLA2
3	0.77	Q_E_HLA3
4	0.79	Q_E_HLA4
5	0.69	Q_E_HLB1
6	0.71	Q_E_HLB2
7	0.74	Q_E_HLB3
8	0.66	Q_E_HLB4
9	0.65	Q_E_HLB5
10	0.84	Q_E_NA1
11	0.75	Q_E_NA2
12	0.7	Q_E_NA3
13	0.71	Q_E_NA4
14	0.58	Q_E_NA5
15	0.86	Q_E_NB1
16	0.65	Q_E_NB2
17	0.69	Q_E_NB3
18	0.58	Q_E_AAS1
19	0.56	Q_E_AAS2
20	0.66	Q_E_AAS3
21	0.65	Q_E_AU1
22	0.62	Q_E_AU2
23	0.48	Q_E_AU3
24	0.71	Q_E_PF1
25	0.68	Q_E_PF2
26	0.62	Q_E_PF3
27	0.72	Q_E_AL1
28	0.41	Q_E_AL2

i	icc	Item
29	0.47	Q_E_R_AL3
30	0.53	Q_E_AL4
31	0.22	Q_E_AL5
32	0.77	Q_E_AS1
33	0.43	Q_E_AS2
34	0.69	Q_E_AS3
35	0.68	Q_E_APP1
36	0.79	Q_E_R_APP2
37	0.72	Q_E_APP3
38	0.7	Q_E_UAA1
39	0.72	Q_E_UAA2
40	0.29	Q_E_R_UAA3
41	0.74	Q_E_R_AE1
42	0.69	Q_E_AE2
43	0.74	Q_E_AE3
44	0.61	Q_E_R_AE4
45	0.72	Q_E_UE1
46	0.51	Q_E_UE2
47	0.62	Q_E_UE3
48	0.75	Q_E_UT1
49	0.61	Q_E_UT2
50	0.74	Q_E_UT3
51	0.57	Q_E_UAL1
52	0.79	Q_E_UAL2
53	0.64	Q_E_UAL3
54	0.65	Q_E_UAL4
55	0.48	Q_E_UAL5
56	0.65	Q_E_UAL6
57	0.62	Q_E_AA1
58	0.53	Q_E_AA2
59	0.77	Q_E_AA3
60	0.52	Q_E_R_AC1
61	0.67	Q_E_R_AC2
62	0.76	Q_E_R_AC3
63	0.7	Q_E_R_AC4
64	0.51	Q_E_AI1
65	0.78	Q_E_AI2
66	0.75	Q_E_R_AI3
67	0.71	Q_E_AI4
68	0.74	Q_E_AT1
69	0.69	Q_E_AT2
70	0.72	Q_E_R_AT3
71	0.62	Q_E_SP1
72	0.82	Q_E_SP2
73	0.74	Q_E_SP3
74	0.62	Q_E_IIS1
75	0.73	Q_E_IIS2
76	0.46	Q_E_IIS3
77	0.71	Q_E_IIS4
78	0.71	Q_E_AEI1
79	0.84	Q_E_AEI2
80	0.82	Q_E_R_AEI3

i	icc	Item
81	0.77	Q_E_AEI4
82	0.86	Q_E_R_AEI5
83	0.62	Q_E_UEP1
84	0.56	Q_E_UEP2
85	0.61	Q_E_UEP3
86	0.54	Q_E_UEP4
87	0.57	Q_E_UAI1
88	0.26	Q_E_UAI2
89	0.66	Q_E_UAI3
90	0.64	Q_E_UAI4

```r
Variable <- c("Grand_mean","SD","Minimum","Maximum")
# Define the names of the statistics
Value <- c(round(mean(l_ICC$icc),digits=2),round(sd(l_ICC$icc),digits=2),
           round(min(l_ICC$icc),digits=2),round(max(l_ICC$icc),digits=2))
# Calculate the grand mean, standard deviation,
# minimum and maximum values of ICC values of 90 items
description <- cbind(Variable, Value) # Descriptive statistics of ICC values of 90 items

# Print results
pander(description, caption = paste("All participants - Descriptive",
                                    "statistics of ICC values of 90 items"))
```

Table 2: All participants - Descriptive statistics of ICC values of 90 items

Variable	Value
Grand_mean	0.65
SD	0.13
Minimum	0.22
Maximum	0.86

For the assessment of the correlation between the English and Dutch ASA Questionnaire, we followed Cicchetti's classification of ICC categories (Cicchetti 1994). Then we get the categories of ICC classifications and number of ICC values in each classification category.

```r
Classification <- c("Excellent", "Good", "Fair", "Poor")
ICC_Range <- c("0.75-1.00", "0.60-0.74", "0.40-0.59", "0-0.39")
# Categories of ICC classifications by Cicchetti (1994)
n_item <- length(l_ICC$icc)  # Number of ICC values
round_ICC <- round(l_ICC$icc, digits = NUM_PRECISION)  # Round ICC values
Number <- c(length(l_ICC[which(round_ICC >= 0.75 & round_ICC <=
    1), ]$icc), length(l_ICC[which(round_ICC >= 0.6 & round_ICC <=
    0.7499), ]$icc), length(l_ICC[which(round_ICC >= 0.4 & round_ICC <=
    0.5999), ]$icc), length(l_ICC[which(round_ICC >= 0 & round_ICC <=
    0.3999), ]$icc))
# Calculate number of ICC values in classification category
Percentage <- c(round(Number[1]/n_item, digits = 4) * 100, round(Number[2]/n_item,
    digits = 4) * 100, round(Number[3]/n_item, digits = 4) *
    100, round(Number[4]/n_item, digits = 4) * 100)
# Calculate percentage of ICC values in classification
```

```
# category
ICC_category <- cbind(Classification, ICC_Range, Number, Percentage)

# Print results
pander(ICC_category, caption = "Categories of ICC classifications and
       number of ICC values in classification category for 90 items")
```

Table 3: Categories of ICC classifications and number of ICC values in classification category for 90 items

Classification	ICC_Range	Number	Percentage
Excellent	0.75-1.00	18	20
Good	0.60-0.74	50	55.56
Fair	0.40-0.59	18	20
Poor	0-0.39	4	4.44

Removing English Prefix 'Q_E_'

For easier legibility of the code below, and for better compatibility with the legacy codebase (from the Chinese translation creation/validation), the Prefix 'Q_E_' is removed from English items (e.g. 'HLA1' instead of 'Q_E_HLA1'). The Dutch item-prefixes ('Q_DU_') remain.

```
for ( col in 1:NUM_ITEMS_FULL_ASAQ){
    colnames(d_total)[col] <-  sub("Q_E_", "", colnames(d_total)[col])
}
```

ICC values for 24 constructs and related dimensions

We combined the scores of Construct 1-8 (first half) and Construct 9-19 (second half), as the input data for the correlation analysis for 24 constructs/dimensions. Then we called the function $getLME$ to calculate ICC values for each construct/dimension.

```
Dutch_column_offset = ncol(d_total)/2
# 'i' is a vector with the column number of the first
# English version of item of the construct/dimension
i <- which(names(d_total) %in% c("HLA1", "HLB1", "NA1", "NB1",
    "AAS1", "AU1", "PF1", "AL1", "AS1", "APP1", "UAA1", "R_AE1",
    "UE1", "UT1", "UAL1", "AA1", "R_AC1", "AI1", "AT1", "SP1",
    "IIS1", "AEI1", "UEP1", "UAI1"))

# 'k1' is a vector with the number of questionnaire items
# of each construct/dimension for Construct 1-8 Note that
# we assume here that construct/dimension items are
# adjacent columns in the data frame
k1 <- c(ncol(select(d_total, HLA1:HLA4)), ncol(select(d_total,
    HLB1:HLB5)), ncol(select(d_total, NA1:NA5)), ncol(select(d_total,
    NB1:NB3)), ncol(select(d_total, AAS1:AAS3)), ncol(select(d_total,
    AU1:AU3)), ncol(select(d_total, PF1:PF3)), ncol(select(d_total,
    AL1:AL5)), ncol(select(d_total, AS1:AS3)), ncol(select(d_total,
    APP1:APP3)), ncol(select(d_total, UAA1:R_UAA3)), ncol(select(d_total,
    R_AE1:R_AE4)))

# 'k2' is a vector with the number of questionnaire items
# of each construct/dimension for Construct 9-19
```

```
k2 <- c(ncol(select(d_total, UE1:UE3)), ncol(select(d_total,
    UT1:UT3)), ncol(select(d_total, UAL1:UAL6)), ncol(select(d_total,
    AA1:AA3)), ncol(select(d_total, R_AC1:R_AC4)), ncol(select(d_total,
    AI1:AI4)), ncol(select(d_total, AT1:R_AT3)), ncol(select(d_total,
    SP1:SP3)), ncol(select(d_total, IIS1:IIS4)), ncol(select(d_total,
    AEI1:R_AEI5)), ncol(select(d_total, UEP1:UEP4)), ncol(select(d_total,
    UAI1:UAI4)))

# Combine k1 and k2 into a single vector with the number of
# questionnaire items of each construct/dimension of the
# entire ASAQ
k = c(k1, k2)
# Combine i and k into a data frame, whereby i indicates
# the column number of the first English item of a
# construct and k the total number of adjacent
# questionnaire items associated with the construct
h <- cbind.data.frame(i, k)

# Initialize output of ICC values of 24
# constructs/dimensions
l_ICC <- data.frame(ConstructID = double(), Construct = character(),
    icc = double())

# Go step by step to 24 constructs/dimensions of the ASA
# questionnaire
for (p in 1:NUM_ITEMS_SHORT_ASAQ) {
    # Column number of the first ASAQ item in English of
    # the construct/dimension
    i <- h[p, 1]
    # The column number of the first ASAQ item in the Dutch
    # version of the construct/dimension
    j <- i + Dutch_column_offset
    # The number of ASAQ items associate to the
    # construct/dimension
    k <- h[p, 2]
    # Select the scores of all the ASAQ items in Dutch
    # associated with the construct/dimension
    s_Dutch <- data.frame(d_total[, j:(j + k - 1)])
    # Select the score of all the ASAQ items in English
    # associated with the construct/dimension
    s_English <- data.frame(d_total[, i:(i + k - 1)])
    # Calculate the mean score of ASAQ items in Dutch
    # associated with the construct/dimension per
    # participant
    average_s_Dutch <- data.frame(rowMeans(s_Dutch))
    # Doing the same but now for English version of the
    # items
    average_s_English <- data.frame(rowMeans(s_English))
    colnames(average_s_Dutch) <- c("score")  # Rename Dutch mean column
    colnames(average_s_English) <- c("score")  # Rename English mean column
    # Call function 'getLME' for ICC value calculation
    l_ICC <- rbind(l_ICC, data.frame(p, icc = getLME(average_s_Dutch,
        average_s_English)))
```

```
}
# Add construct/dimension name code
l_ICC$Construct = c("HLA", "HLB", "NA", "NB", "AAS", "AU", "PF",
    "AL", "AS", "APP", "UAA", "AE", "UE", "UT", "UAL", "AA",
    "AC", "AI", "AT", "SP", "IIS", "AEI", "UEP", "UAI")
pander(l_ICC, caption = "ICC values for 24 constructs/dimensions")
```

Table 4: ICC values for 24 constructs/dimensions

p	icc	Construct
1	0.89	HLA
2	0.89	HLB
3	0.89	NA
4	0.83	NB
5	0.78	AAS
6	0.83	AU
7	0.78	PF
8	0.58	AL
9	0.83	AS
10	0.87	APP
11	0.71	UAA
12	0.86	AE
13	0.77	UE
14	0.81	UT
15	0.84	UAL
16	0.79	AA
17	0.84	AC
18	0.87	AI
19	0.85	AT
20	0.85	SP
21	0.8	IIS
22	0.93	AEI
23	0.79	UEP
24	0.72	UAI

```
# Define the names of the statistics
Variable <- c("Grand_mean", "SD", "Minimum", "Maximum")
# Calculate the grand mean, standard deviation, minimum and
# maximum values of ICC values of 24 constructs/dimensions
Value <- c(round(mean(l_ICC$icc), digits = 2), round(sd(l_ICC$icc),
    digits = 2), round(min(l_ICC$icc), digits = 2), round(max(l_ICC$icc),
    digits = 2))
# Descriptive statistics of ICC values of 24
# constructs/dimensions
description <- cbind(Variable, Value)

# Print results
pander(description, caption = "Descriptive statistics of ICC values
of 24 constructs/dimensions")
```

Table 5: Descriptive statistics of ICC values of 24 constructs/dimensions

Variable	Value
Grand_mean	0.82
SD	0.07
Minimum	0.58
Maximum	0.93

And we classify the resulting ICC values again based on (Cicchetti 1994).

```
Classification <- c("Excellent", "Good", "Fair", "Poor")
ICC_Range <- c("0.75-1.00", "0.60-0.74", "0.40-0.59", "0-0.39")
# Categories of ICC classifications by Cicchetti (1994)
n_item <- length(l_ICC$icc)  # Number of ICC values
round_ICC <- round(l_ICC$icc, digits = NUM_PRECISION)  # Round ICC values
# Calculate number of ICC values in classification category
Number <- c(length(l_ICC[which(round_ICC >= 0.75 & round_ICC <=
    1), ]$icc), length(l_ICC[which(round_ICC >= 0.6 & round_ICC <=
    0.7499), ]$icc), length(l_ICC[which(round_ICC >= 0.4 & round_ICC <=
    0.5999), ]$icc), length(l_ICC[which(round_ICC >= 0 & round_ICC <=
    0.3999), ]$icc))
# Calculate percentage of ICC values in classification
# category
Percentage <- c(round(Number[1]/n_item, digits = 4) * 100, round(Number[2]/n_item,
    digits = 4) * 100, round(Number[3]/n_item, digits = 4) *
    100, round(Number[4]/n_item, digits = 4) * 100)
ICC_category <- cbind(Classification, ICC_Range, Number, Percentage)

# Print results
pander(ICC_category, caption = "Categories of ICC classifications and number
       of ICC values in classification category for 24 constructs/dimensions")
```

Table 6: Categories of ICC classifications and number of ICC values in classification category for 24 constructs/dimensions

Classification	ICC_Range	Number	Percentage
Excellent	0.75-1.00	21	87.5
Good	0.60-0.74	2	8.33
Fair	0.40-0.59	1	4.17
Poor	0-0.39	0	0

ICC values between English and Dutch scores for the short version of ASA questionnaire

The last ICC calculation is for the ASAQ items of the short version of the ASAQ. The procedure is similar to ICC calculation of the 90 items, only this time, we select only the relevant 24 items first.

```
# Select Dutch versions of the 24 representative ASAQ items
s_Dutch <- select(d_total, Q_DU_HLA2, Q_DU_HLB5, Q_DU_NA4, Q_DU_NB3,
    Q_DU_AAS1, Q_DU_AU1, Q_DU_PF1, Q_DU_AL2, Q_DU_AS1, Q_DU_APP1,
    Q_DU_UAA1, Q_DU_R_AE1, Q_DU_UE2, Q_DU_UT3, Q_DU_UAL1, Q_DU_AA2,
    Q_DU_R_AC1, Q_DU_R_AI3, Q_DU_AT1, Q_DU_SP2, Q_DU_IIS2, Q_DU_R_AEI3,
    Q_DU_UEP3, Q_DU_UAI4)
```

```r
# Select English versions of the 24 representative ASAQ
# items
s_English <- select(d_total, HLA2, HLB5, NA4, NB3, AAS1, AU1,
    PF1, AL2, AS1, APP1, UAA1, R_AE1, UE2, UT3, UAL1, AA2, R_AC1,
    R_AI3, AT1, SP2, IIS2, R_AEI3, UEP3, UAI4)
# Combine Dutch and English scores
ss <- cbind(s_Dutch, s_English)

n <- ncol(ss)  # Numbers of all columns in ss
English_column_offset <- n/2

# Initialize output of ICC values of 24 representative
# items
l_ICC <- data.frame(ID = double(), Item = character(), icc = double())
# Go step by step to 24 representative items of the ASA
# questionnaire
for (i in 1:NUM_ITEMS_SHORT_ASAQ) {
    # Select Dutch scores of the ASAQ item
    score_Dutch <- data.frame(score = ss[, i])
    # Select English scores of the ASAQ item
    score_English <- data.frame(score = ss[, i + English_column_offset])
    # Call function 'getLME' for ICC value calculation
    l_ICC <- rbind(l_ICC, data.frame(i, icc = getLME(score_Dutch,
        score_English)))
}
l_ICC$Item <- colnames(s_English)  # Add item name code
pander(l_ICC, caption = "ICC values for 24 representative items")
```

Table 7: ICC values for 24 representative items

i	icc	Item
1	0.39	HLA2
2	0.65	HLB5
3	0.71	NA4
4	0.69	NB3
5	0.58	AAS1
6	0.65	AU1
7	0.71	PF1
8	0.41	AL2
9	0.77	AS1
10	0.68	APP1
11	0.7	UAA1
12	0.74	R_AE1
13	0.51	UE2
14	0.74	UT3
15	0.57	UAL1
16	0.53	AA2
17	0.52	R_AC1
18	0.75	R_AI3
19	0.74	AT1
20	0.82	SP2
21	0.73	IIS2
22	0.82	R_AEI3

i	icc	Item
23	0.61	UEP3
24	0.64	UAI4

```
# Define the names of the statistics
Variable <- c("Grand_mean", "SD", "Minimum", "Maximum")
# Calculate the grand mean, standard deviation, minimum and
# maximum values of ICC values of 24 representative items
Value <- c(round(mean(l_ICC$icc), digits = 2), round(sd(l_ICC$icc),
    digits = 2), round(min(l_ICC$icc), digits = 2), round(max(l_ICC$icc),
    digits = 2))
# Descriptive statistics of ICC values of 24 representative
# items
description <- cbind(Variable, Value)

# Print results
pander(description, caption = "Descriptive statistics of ICC values
of 24 representative items")
```

Table 8: Descriptive statistics of ICC values of 24 representative items

Variable	Value
Grand_mean	0.65
SD	0.12
Minimum	0.39
Maximum	0.82

And we classify the resulting ICC values again based on (Cicchetti 1994).

```
# Categories of ICC classifications by Cicchetti (1994)
Classification <- c("Excellent", "Good", "Fair", "Poor")
ICC_Range <- c("0.75-1.00", "0.60-0.74", "0.40-0.59", "0-0.39")
n_item <- length(l_ICC$icc)  # Number of ICC values
round_ICC <- round(l_ICC$icc, digits = NUM_PRECISION)  # Round ICC values
# Calculate number of ICC values in classification category
Number <- c(length(l_ICC[which(round_ICC >= 0.75 & round_ICC <=
    1), ]$icc), length(l_ICC[which(round_ICC >= 0.6 & round_ICC <=
    0.7499), ]$icc), length(l_ICC[which(round_ICC >= 0.4 & round_ICC <=
    0.5999), ]$icc), length(l_ICC[which(round_ICC >= 0 & round_ICC <=
    0.3999), ]$icc))
# Calculate percentage of ICC values in classification
# category
Percentage <- c(round(Number[1]/n_item, digits = 4) * 100, round(Number[2]/n_item,
    digits = 4) * 100, round(Number[3]/n_item, digits = 4) *
    100, round(Number[4]/n_item, digits = 4) * 100)
ICC_category <- cbind(Classification, ICC_Range, Number, Percentage)

# Print results
pander(ICC_category, caption = "Categories of ICC classifications and number
       of ICC values in classification category for 24 representative items")
```

Table 9: Categories of ICC classifications and number of ICC values in classification category for 24 representative items

Classification	ICC_Range	Number	Percentage
Excellent	0.75-1.00	4	16.67
Good	0.60-0.74	13	54.17
Fair	0.40-0.59	6	25
Poor	0-0.39	1	4.17

Variation Between English and Dutch ASA Questionnaire

The mean score differences between the English and Dutch questionnaires are estimates for absolute accuracy in score equivalence between the two languages. 95% credible intervals of mean paired differences were calculated using a Bayesian paired *t*-test, for item level, construct and dimension level, and the short version of the ASA questionnaire. We used the combined input data of both halves.

Mean score differences for 90 items

We used the Bayesian pairwise *t*-test to estimate the difference in ASAQ items score between the English and the Dutch version. First we define function establish sample means and standard deviation, next relevant information is extracted from output date produced by Bayesian *t*-test.

```
# Function to obtain mean, and sd values of ss_1 (Dutch)
# and ss_2 (English), and relevant information from
# Bayesian t-test output stored in B_output,
# this is take from the 1 line for Bayes output
# which relates to the estimation of the means and mean difference
# ID is the identification number added in the return data
# frame row to identify an item or construct
getBAYES <-function(ID, ss_1, ss_2, B_output)
{ l <- data.frame(ID,
                  mean_Dutch = mean(ss_1), # Mean of Dutch translation
                  sd_Dutch = sd(ss_1), # Standard deviation of Dutch translation
                  mean_English = mean(ss_2), # Mean of English item
                  sd_English = sd(ss_2), # Standard deviation of English item
                  mean_diff = as.numeric(B_output[["stats"]][1,1]), # Mean of mu difference
                  sd_diff = as.numeric(B_output[["stats"]][1,2]), # Standard deviation
                  HDIlo = as.numeric(B_output[["stats"]][1,5]), # HDIlo
                  HDIup = as.numeric(B_output[["stats"]][1,6]), # HDIup
                  n_eff = as.numeric(B_output[["stats"]][1,16]),# n_eff
                  Rhat = as.numeric(B_output[["stats"]][1,15]), # Rhat
                  P_posterior = max(B_output[["stats"]][1,8],   # %<comp
                                    B_output[["stats"]][1,7]),  # %>comp
                  # Add "*" marker if the low bound of HDI is large than zero,
                  # or the upper bound is smaller than zero
                  zero_excl = ifelse((as.numeric(B_output[["stats"]][1,5])>0)   # HDIlo
                                     | (as.numeric(B_output[["stats"]][1,6])<0),# HDIup
                                     '*','')
  )
return(l)
}
```

With the function $getBAYES$ defined, we now go examine for each ASAQ item the difference between Dutch and English scores.

```r
# Initialize output of Items with credible bias indication
item_list <- data.frame(Item = character(), ID = double(), mean_Dutch = double(),
    sd_Dutch = double(), mean_English = double(), sd_English = double(),
    mean_diff = double(), sd_diff = double(), HDIlo = double(),
    HDIup = double(), zero_excl = character())

# Numbers of all columns in d_total, i.e. English and Dutch
# scores combined
n <- ncol(d_total)
# Offset for the column position of the first Dutch ASAQ
# items
Dutch_column_offset <- n/2

# Go step by step to 90 ASA questionnaire items
for (i in 1:NUM_ITEMS_FULL_ASAQ) {
    score_Dutch <- d_total[, i + Dutch_column_offset]  # Dutch scores
    score_English <- d_total[, i]  # English item scores
    set.seed(1)  # Make sure that estimations of Bayesian analyses remain the same
    # Conduct a Bayesian paired t-test on the Dutch and
    # English score of ASAQ item
    fit <- bayes.t.test(score_Dutch, score_English, paired = TRUE)

    # Store results from Bayesian analysis in a list to
    # print later
    item_list <- rbind(item_list, getBAYES(i, score_Dutch, score_English,
        fit))
}

# Print results
item_list$Item = colnames(select(d_total, HLA1:UAI4))
# Add item name code
pander(select(item_list, ID, mean_Dutch, sd_Dutch, mean_English,
    sd_English, Item), caption = "Items with credible bias indication (Part 1)")
```

Table 10: Items with credible bias indication (Part 1)

ID	mean_Dutch	sd_Dutch	mean_English	sd_English	Item
1	-1.45	1.88	-1.57	1.9	HLA1
2	-1.32	1.83	-1.44	1.96	HLA2
3	-1.31	1.95	-1.34	1.95	HLA3
4	-1.28	1.97	-1.41	1.84	HLA4
5	-1.35	1.74	-1.16	1.73	HLB1
6	-0.86	1.76	-0.75	1.76	HLB2
7	-0.7	1.89	-0.82	1.78	HLB3
8	-0.97	1.77	-1.1	1.73	HLB4
9	-0.28	1.73	-0.69	1.77	HLB5
10	-1.77	1.86	-1.69	1.88	NA1
11	-1.11	1.98	-1.21	1.93	NA2
12	-0.91	1.95	-0.96	1.98	NA3
13	-1.32	1.86	-1.02	1.91	NA4
14	-0.17	1.92	-0.18	1.86	NA5
15	-1.51	1.95	-1.71	1.84	NB1
16	-0.5	1.77	-0.58	1.82	NB2

ID	mean_Dutch	sd_Dutch	mean_English	sd_English	Item
17	-0.37	1.79	-0.45	1.9	NB3
18	1.04	1.67	1.36	1.33	AAS1
19	1.28	1.4	1.27	1.44	AAS2
20	1.27	1.46	1.09	1.58	AAS3
21	1.26	1.33	1.18	1.48	AU1
22	1.21	1.3	1.38	1.34	AU2
23	1.46	1.22	1.32	1.36	AU3
24	1.26	1.36	1.13	1.43	PF1
25	1.49	1.5	1.39	1.51	PF2
26	1	1.35	1.13	1.44	PF3
27	0.52	1.69	0.06	1.77	AL1
28	0.93	1.59	0.22	1.8	AL2
29	1.43	1.53	0.69	1.88	R_AL3
30	1.42	1.38	1.5	1.24	AL4
31	0.58	1.56	-1.02	1.9	AL5
32	-0.55	1.72	-0.58	1.72	AS1
33	-0.17	1.72	0	1.6	AS2
34	0.49	1.89	0.47	1.76	AS3
35	-0.36	1.72	-0.28	1.67	APP1
36	-0.14	1.91	-0.07	1.98	R_APP2
37	-0.73	1.95	-0.8	2.02	APP3
38	0.93	1.33	1.03	1.45	UAA1
39	0.62	1.51	0.76	1.57	UAA2
40	1.89	1.51	1.06	1.85	R_UAA3
41	0.13	1.74	0.35	1.83	R_AE1
42	0.91	1.56	0.87	1.7	AE2
43	0.58	1.74	0.96	1.58	AE3
44	1.12	1.64	1.07	1.58	R_AE4
45	1.97	1.12	1.88	1.1	UE1
46	1.84	1.17	1.92	0.98	UE2
47	2.13	1.01	2.07	0.99	UE3
48	-0.42	1.52	-0.56	1.41	UT1
49	0.88	1.31	0.98	1.39	UT2
50	0.48	1.6	0.36	1.47	UT3
51	0.23	1.62	-0.16	1.69	UAL1
52	-0.03	1.53	-0.06	1.62	UAL2
53	0.15	1.69	-0.16	1.78	UAL3
54	1.04	1.35	1.07	1.39	UAL4
55	0.43	1.51	0.52	1.54	UAL5
56	0.34	1.73	0.53	1.71	UAL6
57	1.73	1.35	1.85	1.31	AA1
58	0.94	1.46	1.18	1.47	AA2
59	1.92	1.29	2.02	1.26	AA3
60	1.79	1.48	1.61	1.5	R_AC1
61	1.62	1.38	1.69	1.44	R_AC2
62	1.53	1.41	1.55	1.31	R_AC3
63	1.54	1.63	1.59	1.43	R_AC4
64	1.34	1.51	1.15	1.49	AI1
65	0.39	1.83	0.51	1.75	AI2
66	0.78	1.91	0.88	1.86	R_AI3
67	-0.01	1.86	0.05	1.88	AI4
68	1.26	1.53	1.24	1.41	AT1

ID	mean_Dutch	sd_Dutch	mean_English	sd_English	Item
69	1.15	1.41	1.12	1.56	AT2
70	1.46	1.53	1.48	1.51	R_AT3
71	0.07	1.7	0.17	1.68	SP1
72	-0.63	1.96	-0.51	1.82	SP2
73	-1.02	1.72	-1.07	1.68	SP3
74	0.21	1.37	0.17	1.34	IIS1
75	0.25	1.49	0.28	1.33	IIS2
76	0.04	1.53	0.17	1.34	IIS3
77	0.14	1.55	0.13	1.43	IIS4
78	-1.01	2.04	-1.27	1.84	AEI1
79	-1.14	1.95	-1.09	1.93	AEI2
80	-0.82	1.96	-0.77	2.02	R_AEI3
81	-0.84	1.91	-0.6	1.92	AEI4
82	-1.09	2	-1.1	1.98	R_AEI5
83	1.16	1.5	1.1	1.41	UEP1
84	0.76	1.62	0.38	1.71	UEP2
85	1.02	1.55	0.84	1.61	UEP3
86	0.83	1.59	1.14	1.57	UEP4
87	0.86	1.75	0.88	1.73	UAI1
88	1.87	1.21	1.04	1.48	UAI2
89	1.53	1.32	1.52	1.38	UAI3
90	0.58	1.79	0.52	1.72	UAI4

```
pander(select(item_list, ID, mean_diff, sd_diff, HDIlo, HDIup,
    Item), caption = "Items with credible bias indication (Part 2)")
```

Table 11: Items with credible bias indication (Part 2)

ID	mean_diff	sd_diff	HDIlo	HDIup	Item
1	0	0	0	0	HLA1
2	0.06	0.07	-0.07	0.22	HLA2
3	0	0	0	0	HLA3
4	0	0	0	0	HLA4
5	-0.22	0.1	-0.42	-0.03	HLB1
6	-0.11	0.11	-0.32	0.11	HLB2
7	0.14	0.1	-0.06	0.34	HLB3
8	0.18	0.12	-0.05	0.41	HLB4
9	0.26	0.12	0.02	0.49	HLB5
10	0	0	0	0	NA1
11	0	0	0	0	NA2
12	0	0.02	-0.04	0.05	NA3
13	-0.06	0.07	-0.22	0.04	NA4
14	-0.02	0.14	-0.29	0.25	NA5
15	0	0	0	0	NB1
16	0.1	0.13	-0.16	0.37	NB2
17	0.09	0.13	-0.17	0.35	NB3
18	-0.26	0.12	-0.49	-0.02	AAS1
19	0.08	0.1	-0.11	0.28	AAS2
20	0.11	0.09	-0.06	0.28	AAS3
21	0	0	0	0	AU1
22	0	0	0	0	AU2

ID	mean_diff	sd_diff	HDIlo	HDIup	Item
23	0.15	0.11	-0.07	0.37	AU3
24	0.1	0.09	-0.08	0.28	PF1
25	0.16	0.1	-0.05	0.35	PF2
26	-0.09	0.1	-0.29	0.11	PF3
27	0.41	0.12	0.19	0.64	AL1
28	0.69	0.16	0.38	0.99	AL2
29	0.71	0.15	0.41	1.02	R_AL3
30	0	0.01	-0.02	0.02	AL4
31	1.59	0.16	1.28	1.92	AL5
32	0.01	0.04	-0.05	0.09	AS1
33	-0.18	0.16	-0.49	0.12	AS2
34	0.02	0.13	-0.24	0.27	AS3
35	-0.1	0.11	-0.32	0.13	APP1
36	0	0.03	-0.05	0.05	R_APP2
37	0	0	0	0	APP3
38	0	0.01	-0.02	0.02	UAA1
39	0	0.01	-0.03	0.01	UAA2
40	0.35	0.22	0	0.78	R_UAA3
41	-0.23	0.11	-0.45	-0.03	R_AE1
42	-0.01	0.09	-0.2	0.18	AE2
43	0	0	0	0	AE3
44	0.03	0.12	-0.21	0.28	R_AE4
45	0	0	0	0	UE1
46	0	0.01	-0.02	0.02	UE2
47	0	0	0	0	UE3
48	0	0.01	0	0	UT1
49	-0.13	0.1	-0.31	0.06	UT2
50	0.09	0.08	-0.07	0.26	UT3
51	0	0.01	-0.02	0.03	UAL1
52	0	0	0	0	UAL2
53	0.27	0.11	0.05	0.49	UAL3
54	0	0	0	0	UAL4
55	-0.08	0.13	-0.34	0.19	UAL5
56	-0.19	0.13	-0.45	0.05	UAL6
57	0	0	0	0	AA1
58	-0.22	0.11	-0.44	0	AA2
59	0	0	0	0	AA3
60	0.09	0.1	-0.12	0.29	R_AC1
61	0	0	0	0	R_AC2
62	0	0	0	0	R_AC3
63	0.01	0.03	-0.05	0.06	R_AC4
64	0.21	0.11	-0.01	0.44	AI1
65	-0.06	0.1	-0.27	0.14	AI2
66	0	0	0	0	R_AI3
67	-0.03	0.06	-0.17	0.07	AI4
68	0	0	0	0	AT1
69	-0.03	0.09	-0.22	0.15	AT2
70	-0.02	0.05	-0.15	0.08	R_AT3
71	-0.13	0.09	-0.32	0.04	SP1
72	-0.12	0.09	-0.3	0.05	SP2
73	0.01	0.04	-0.06	0.09	SP3
74	0	0	0	0	IIS1

ID	mean_diff	sd_diff	HDIlo	HDIup	Item
75	0	0	0	0	IIS2
76	-0.04	0.12	-0.28	0.19	IIS3
77	0.03	0.09	-0.14	0.2	IIS4
78	0	0	0	0	AEI1
79	0	0	0	0	AEI2
80	0	0	0	0	R_AEI3
81	-0.02	0.04	-0.1	0.04	AEI4
82	0	0	0	0	R_AEI5
83	0.01	0.03	-0.04	0.06	UEP1
84	0.36	0.14	0.09	0.64	UEP2
85	0.16	0.13	-0.1	0.4	UEP3
86	-0.3	0.11	-0.51	-0.09	UEP4
87	0	0.11	-0.22	0.21	UAI1
88	0.64	0.12	0.41	0.88	UAI2
89	0	0.1	-0.18	0.19	UAI3
90	0.01	0.09	-0.18	0.19	UAI4

```
pander(select(item_list, ID, n_eff, Rhat, P_posterior, zero_excl,
    Item), caption = "Items with credible bias indication (Part 3)")
```

Table 12: Items with credible bias indication (Part 3)

ID	n_eff	Rhat	P_posterior	zero_excl	Item
1	19451	1	0.5		HLA1
2	10751	1	0.83		HLA2
3	20690	1	0.5		HLA3
4	20117	1	0.5		HLA4
5	16040	1	0.99	*	HLB1
6	18659	1	0.85		HLB2
7	18773	1	0.92		HLB3
8	16671	1	0.93		HLB4
9	3383	1	0.99	*	HLB5
10	19622	1	0.5		NA1
11	19902	1	0.5		NA2
12	14888	1	0.52		NA3
13	1120	1	0.85		NA4
14	17849	1	0.57		NA5
15	20423	1	0.5		NB1
16	17889	1	0.77		NB2
17	18606	1	0.76		NB3
18	13196	1	0.99	*	AAS1
19	17058	1	0.8		AAS2
20	17973	1	0.9		AAS3
21	21459	1	0.5		AU1
22	9620	1.02	0.5		AU2
23	19538	1	0.91		AU3
24	17473	1	0.87		PF1
25	14050	1	0.94		PF2
26	15518	1	0.83		PF3
27	13699	1	1	*	AL1
28	16548	1	1	*	AL2

ID	n_eff	Rhat	P_posterior	zero_excl	Item
29	17995	1	1	*	R_AL3
30	4973	1	0.5		AL4
31	17977	1	1	*	AL5
32	2497	1.12	0.6		AS1
33	18769	1	0.88		AS2
34	18328	1	0.57		AS3
35	18286	1	0.81		APP1
36	10163	1.01	0.5		R_APP2
37	20206	1	0.51		APP3
38	4637	1	0.54		UAA1
39	1979	1.01	0.57		UAA2
40	950	1	0.99	*	R_UAA3
41	17513	1	0.98	*	R_AE1
42	16197	1	0.55		AE2
43	22410	1	0.5		AE3
44	17952	1	0.58		R_AE4
45	28874	1	0.5		UE1
46	10996	1.01	0.51		UE2
47	20630	1	0.51		UE3
48	16949	1	0.5		UT1
49	19835	1	0.91		UT2
50	11647	1	0.86		UT3
51	3365	1.03	0.56		UAL1
52	20285	1	0.5		UAL2
53	13429	1	0.99	*	UAL3
54	19748	1	0.5		UAL4
55	18557	1	0.71		UAL5
56	19206	1	0.94		UAL6
57	18452	1	0.51		AA1
58	15711	1	0.98	*	AA2
59	24183	1	0.5		AA3
60	13989	1	0.8		R_AC1
61	20779	1	0.5		R_AC2
62	27312	1	0.5		R_AC3
63	9691	1.01	0.57		R_AC4
64	18741	1	0.97		AI1
65	9410	1	0.72		AI2
66	18329	1	0.5		R_AI3
67	2922	1	0.71		AI4
68	25241	1	0.5		AT1
69	12282	1	0.64		AT2
70	2812	1.01	0.61		R_AT3
71	16380	1	0.93		SP1
72	10593	1	0.92		SP2
73	2818	1.09	0.55		SP3
74	21138	1	0.5		IIS1
75	24291	1	0.5		IIS2
76	11336	1	0.63		IIS3
77	17459	1	0.64		IIS4
78	20256	1	0.5		AEI1
79	21255	1	0.51		AEI2
80	20715	1	0.5		R_AEI3

ID	n_eff	Rhat	P_posterior	zero_excl	Item
81	1155	1.03	0.7		AEI4
82	20599	1	0.51		R_AEI5
83	2567	1	0.62		UEP1
84	17957	1	0.99	*	UEP2
85	17882	1	0.89		UEP3
86	16945	1	1	*	UEP4
87	17741	1	0.52		UAI1
88	10828	1	1	*	UAI2
89	17453	1	0.51		UAI3
90	17288	1	0.55		UAI4

```r
# Calculate Grand mean information across the statistics
# obtained from 90 items Define the names of the statistics
Variable <- c("mean_Dutch", "sd_Dutch", "mean_English", "sd_English",
    "mean_diff", "sd_diff", "minimum_diff", "maximum_diff", "n_zero_excl",
    "percent_zero_excl")

# Calculate the grand means of mean_Dutch, sd_Dutch,
# mean_English, sd_English, sd_diff, grand mean of the
# absolute value of mean differences, number of items with
# credible bias indication, and percentage of these items
Grand_mean <- c(mean(item_list$mean_Dutch), mean(item_list$sd_Dutch),
    mean(item_list$mean_English), mean(item_list$sd_English),
    mean(abs(item_list$mean_diff)), mean(item_list$sd_diff),
    min(item_list$mean_diff), max(item_list$mean_diff), sum(item_list$zero_excl ==
        "*"), round(sum(item_list$zero_excl == "*")/length(item_list$ID),
        digits = 4) * 100)

# Print results
GrandMean <- cbind(Variable, Grand_mean)
pander(GrandMean, caption = "Grand mean of 90 items")
```

Table 13: Grand mean of 90 items

Variable	Grand_mean
mean_Dutch	0.366203703703704
sd_Dutch	1.62376815473755
mean_English	0.307962962962963
sd_English	1.62832116037787
mean_diff	0.108819554176695
sd_diff	0.0664392043462865
minimum_diff	-0.297249372166589
maximum_diff	1.59412632875887
n_zero_excl	14
percent_zero_excl	15.56

Mean score differences for 24 constructs and related dimensions

Next, step is to repeat the Bayesian *t*-test analysis but this time on a construct level. 95% credible interval of mean pairwise difference by Bayesian paired *t*-test was calculated for 24 constructs and related dimensions. It would reveal the variation between 24 English ASA constructs/dimensions and corresponding Dutch translations. Before the t-test can be performed, we first have to calculate the construct score for each participant by taking the average score of the related ASAQ score. We have to do this both for the English and the Dutch version of the ASAQ.

```
# Initialize output of Constructs/dimensions with credible
# bias indication
con_list <- data.frame(Construct = character(), ID = double(),
    mean_Dutch = double(), sd_Dutch = double(), mean_English = double(),
    sd_English = double(), mean_diff = double(), sd_diff = double(),
    mean_diff = double(), HDIlo = double(), HDIup = double(),
    zero_excl = character())

# Numbers of all columns in d_total, i.e. English and Dutch
# scores combined
n <- ncol(d_total)
# Offset for the column position of the first Dutch ASAQ
# items
Dutch_column_offset <- n/2

# Go step by step to 24 constructs/dimensions
for (p in 1:NUM_ITEMS_SHORT_ASAQ) {
    # The column with the first English ASAQ item of the
    # construct/dimension
    i = h[p, 1]
    # The column with the first Dutch ASAQ item of the
    # construct/dimension
    j = i + Dutch_column_offset
    k = h[p, 2]  # The number of columns/items of the construct/dimension
    s_Dutch <- data.frame(d_total[, j:(j + k - 1)])  # Select Dutch scores
    s_English <- data.frame(d_total[, i:(i + k - 1)])  # Select English scores
    # Dutch score means for each construct/dimension per
    # participant
    average_s_Dutch <- data.frame(rowMeans(s_Dutch))
    # English score means for each construct/dimension per
    # participant
    average_s_English <- data.frame(rowMeans(s_English))
    colnames(average_s_Dutch) <- c("score")  # Rename Dutch mean column
    colnames(average_s_English) <- c("score")  # Rename English mean column
    # Combine averaged scores of Dutch and English
    # constructs/dimensions
    score <- data.frame(cbind(average_s_Dutch, average_s_English))
    # Select averaged scores of each Dutch
    # construct/dimension, make sure data format is
    # suitable for Bayesian paired t-test
    score_Dutch <- score[, 1]
    # Select averaged scores of each English
    # construct/dimension, make sure data format is
    # suitable for Bayesian paired t-test
    score_English <- score[, 2]
    set.seed(1)  # Make sure that estimations of Bayesian analyses remain the same
    # Conduct Bayesian t-test
    fit <- bayes.t.test(score_Dutch, score_English, paired = TRUE)
    # Call function 'getBAYES' to obtain relevant
    # information from Bayesian t-test output and add
```

```
    # result to output list
    con_list <- rbind(con_list, getBAYES(p, score_Dutch, score_English,
        fit))
}

# Print results Add construct/dimension name code
con_list$Construct = c("HLA", "HLB", "NA", "NB", "AAS", "AU",
    "PF", "AL", "AS", "APP", "UAA", "AE", "UE", "UT", "UAL",
    "AA", "AC", "AI", "AT", "SP", "IIS", "AEI", "UEP", "UAI")
pander(select(con_list, ID, mean_Dutch, sd_Dutch, mean_English,
    sd_English, Construct), caption = "Constructs/dimensions with credible bias indication (Part 1)")
```

Table 14: Constructs/dimensions with credible bias indication (Part 1)

ID	mean_Dutch	sd_Dutch	mean_English	sd_English	Construct
1	-1.34	1.62	-1.44	1.76	HLA
2	-0.83	1.55	-0.9	1.52	HLB
3	-1.05	1.49	-1.01	1.51	NA
4	-0.79	1.53	-0.91	1.55	NB
5	1.2	1.3	1.24	1.27	AAS
6	1.31	1.01	1.29	1.21	AU
7	1.25	1.07	1.22	1.18	PF
8	0.97	1.14	0.29	1.34	AL
9	-0.08	1.57	-0.04	1.43	AS
10	-0.41	1.51	-0.38	1.55	APP
11	1.15	1.14	0.95	1.3	UAA
12	0.69	1.22	0.81	1.32	AE
13	1.98	0.95	1.95	0.82	UE
14	0.31	1.15	0.26	1.15	UT
15	0.36	1.13	0.29	1.14	UAL
16	1.53	1.05	1.68	1.14	AA
17	1.62	1.15	1.61	1.08	AC
18	0.63	1.42	0.65	1.38	AI
19	1.29	1.34	1.28	1.36	AT
20	-0.53	1.54	-0.47	1.45	SP
21	0.16	1.25	0.19	1.08	IIS
22	-0.98	1.77	-0.96	1.78	AEI
23	0.94	1.28	0.87	1.25	UEP
24	1.21	1.07	0.99	1.07	UAI

```
pander(select(con_list, ID, mean_diff, sd_diff, HDIlo, HDIup,
    Construct), caption = "Constructs/dimensions with credible bias indication (Part 2)")
```

Table 15: Constructs/dimensions with credible bias indication (Part 2)

ID	mean_diff	sd_diff	HDIlo	HDIup	Construct
1	0.08	0.06	-0.03	0.21	HLA
2	0.08	0.07	-0.05	0.21	HLB
3	-0.04	0.06	-0.16	0.09	NA

ID	mean_diff	sd_diff	HDIlo	HDIup	Construct
4	0.12	0.08	-0.04	0.28	NB
5	-0.06	0.07	-0.2	0.07	AAS
6	0.01	0.06	-0.11	0.13	AU
7	0.03	0.07	-0.1	0.16	PF
8	0.68	0.09	0.49	0.85	AL
9	-0.03	0.08	-0.2	0.13	AS
10	-0.03	0.07	-0.17	0.11	APP
11	0.13	0.09	-0.04	0.3	UAA
12	-0.14	0.06	-0.25	-0.02	AE
13	0.04	0.05	-0.06	0.13	UE
14	0.04	0.06	-0.07	0.16	UT
15	0.05	0.05	-0.06	0.15	UAL
16	-0.16	0.06	-0.28	-0.03	AA
17	0.02	0.06	-0.1	0.13	AC
18	0	0.06	-0.12	0.13	AI
19	0	0.06	-0.11	0.12	AT
20	-0.06	0.07	-0.2	0.08	SP
21	-0.03	0.06	-0.15	0.09	IIS
22	0	0.06	-0.12	0.12	AEI
23	0.08	0.07	-0.05	0.22	UEP
24	0.2	0.07	0.07	0.34	UAI

```
pander(select(con_list, ID, n_eff, Rhat, P_posterior, zero_excl,
    Construct), caption = "Constructs/dimensions with credible bias indication (Part 3)")
```

Table 16: Constructs/dimensions with credible bias indication (Part 3)

ID	n_eff	Rhat	P_posterior	zero_excl	Construct
1	17865	1	0.92		HLA
2	18563	1	0.9		HLB
3	17831	1	0.73		NA
4	18721	1	0.93		NB
5	18345	1	0.82		AAS
6	18078	1	0.56		AU
7	18090	1	0.7		PF
8	18701	1	1	*	AL
9	19247	1	0.66		AS
10	18123	1	0.68		APP
11	8905	1	0.94		UAA
12	19237	1	0.99	*	AE
13	18120	1	0.77		UE
14	18509	1	0.77		UT
15	18400	1	0.82		UAL
16	19414	1	0.99	*	AA
17	19645	1	0.61		AC
18	16285	1	0.51		AI
19	18678	1	0.53		AT
20	18571	1	0.8		SP
21	18107	1	0.68		IIS
22	16549	1	0.5		AEI

ID	n_eff	Rhat	P_posterior	zero_excl	Construct
23	18740	1	0.89		UEP
24	18754	1	1	*	UAI

```
# Determine grand (abs) means
Variable <- c("mean_Dutch", "sd_Dutch", "mean_English", "sd_English",
    "mean_diff", "sd_diff", "minimum_diff", "maximum_diff", "n_zero_excl",
    "percent_zero_excl")
# Calculate grand mean of mean_Dutch, sd_Dutch,
# mean_English, sd_English, sd_diff, grand mean of the
# absolute value of mean differences, number of
# constructs/dimensions with credible bias indication, and
# percentage of these constructs/dimensions
Grand_mean <- c(mean(con_list$mean_Dutch), mean(con_list$sd_Dutch),
    mean(con_list$mean_English), mean(con_list$sd_English), mean(abs(con_list$mean_diff)),
    mean(con_list$sd_diff), min(con_list$mean_diff), max(con_list$mean_diff),
    sum(con_list$zero_excl == "*"), round(sum(con_list$zero_excl ==
        "*")/length(con_list$ID), digits = 4) * 100)
GrandMean <- cbind(Variable, Grand_mean)
pander(GrandMean, caption = "Grand mean of 24 constructs/dimensions")
```

Table 17: Grand mean of 24 constructs/dimensions

Variable	Grand_mean
mean_Dutch	0.440989583333333
sd_Dutch	1.3023367846407
mean_English	0.393732638888889
sd_English	1.31774947186162
mean_diff	0.0886763912399634
sd_diff	0.0663805916592571
minimum_diff	-0.157661832497502
maximum_diff	0.675833528060088
n_zero_excl	4
percent_zero_excl	16.67

Mean score differences between English and Dutch short version of ASA questionnaire

As with the ICC, we also conduct a difference analysis for the representative ASAQ items in the short version of the ASAQ.

```
# Initialize output of Representative items with credible
# bias indication
rep_list <- data.frame(Item = character(), ID = double(), mean_Dutch = double(),
    sd_Dutch = double(), mean_English = double(), sd_English = double(),
    mean_diff = double(), sd_diff = double(), HDIlo = double(),
    HDIup = double(), zero_excl = character())

n <- ncol(ss)  # Numbers of all columns in ss
English_column_offset <- n/2

# Go step by step to 24 representative items of the ASA
# questionnaire
```

```r
for (i in 1:NUM_ITEMS_SHORT_ASAQ) {
    score_Dutch <- as.numeric(ss[, i])  # Select Dutch scores
    score_English <- as.numeric(ss[, i + English_column_offset])  # Select English scores
    set.seed(1)  # Make sure that estimations of Bayesian analyses remain the same
    fit <- bayes.t.test(score_Dutch, score_English, paired = TRUE)
    rep_list <- rbind(rep_list, getBAYES(i, score_Dutch, score_English,
        fit))
}

# Print results Add item name code
rep_list$Item <- c("HLA2", "HLB5", "NA4", "NB3", "AAS1", "AU1",
    "PF1", "AL2", "AS1", "APP1", "UAA1", "R_AE1", "UE2", "UT3",
    "UAL1", "AA2", "R_AC1", "R_AI3", "AT1", "SP2", "IIS2", "R_AEI3",
    "UEP3", "UAI4")
pander(select(rep_list, ID, mean_Dutch, sd_Dutch, mean_English,
    sd_English, Item), caption = "Representative items with credible bias indication (Part 1)")
```

Table 18: Representative items with credible bias indication (Part 1)

ID	mean_Dutch	sd_Dutch	mean_English	sd_English	Item
1	-1.32	1.83	-1.44	1.96	HLA2
2	-0.28	1.73	-0.69	1.77	HLB5
3	-1.32	1.86	-1.02	1.91	NA4
4	-0.37	1.79	-0.45	1.9	NB3
5	1.04	1.67	1.36	1.33	AAS1
6	1.26	1.33	1.18	1.48	AU1
7	1.26	1.36	1.13	1.43	PF1
8	0.93	1.59	0.22	1.8	AL2
9	-0.55	1.72	-0.58	1.72	AS1
10	-0.36	1.72	-0.28	1.67	APP1
11	0.93	1.33	1.03	1.45	UAA1
12	0.13	1.74	0.35	1.83	R_AE1
13	1.84	1.17	1.92	0.98	UE2
14	0.48	1.6	0.36	1.47	UT3
15	0.23	1.62	-0.16	1.69	UAL1
16	0.94	1.46	1.18	1.47	AA2
17	1.79	1.48	1.61	1.5	R_AC1
18	0.78	1.91	0.88	1.86	R_AI3
19	1.26	1.53	1.24	1.41	AT1
20	-0.63	1.96	-0.51	1.82	SP2
21	0.25	1.49	0.28	1.33	IIS2
22	-0.82	1.96	-0.77	2.02	R_AEI3
23	1.02	1.55	0.84	1.61	UEP3
24	0.58	1.79	0.52	1.72	UAI4

```r
pander(select(rep_list, ID, mean_diff, sd_diff, HDIlo, HDIup,
    Item), caption = "Representative items with credible bias indication (Part 2)")
```

Table 19: Representative items with credible bias indication (Part 2)

ID	mean_diff	sd_diff	HDIlo	HDIup	Item
1	0.06	0.07	-0.07	0.22	HLA2
2	0.26	0.12	0.02	0.49	HLB5
3	-0.06	0.07	-0.22	0.04	NA4
4	0.09	0.13	-0.17	0.35	NB3
5	-0.26	0.12	-0.49	-0.02	AAS1
6	0	0	0	0	AU1
7	0.1	0.09	-0.08	0.28	PF1
8	0.69	0.16	0.38	0.99	AL2
9	0.01	0.04	-0.05	0.09	AS1
10	-0.1	0.11	-0.32	0.13	APP1
11	0	0.01	-0.02	0.02	UAA1
12	-0.23	0.11	-0.45	-0.03	R_AE1
13	0	0.01	-0.02	0.02	UE2
14	0.09	0.08	-0.07	0.26	UT3
15	0	0.01	-0.02	0.03	UAL1
16	-0.22	0.11	-0.44	0	AA2
17	0.09	0.1	-0.12	0.29	R_AC1
18	0	0	0	0	R_AI3
19	0	0	0	0	AT1
20	-0.12	0.09	-0.3	0.05	SP2
21	0	0	0	0	IIS2
22	0	0	0	0	R_AEI3
23	0.16	0.13	-0.1	0.4	UEP3
24	0.01	0.09	-0.18	0.19	UAI4

```
pander(select(rep_list, ID, n_eff, Rhat, P_posterior, zero_excl,
    Item), caption = "Representative items with credible bias indication (Part 3)")
```

Table 20: Representative items with credible bias indication (Part 3)

ID	n_eff	Rhat	P_posterior	zero_excl	Item
1	10751	1	0.83		HLA2
2	3383	1	0.99	*	HLB5
3	1120	1	0.85		NA4
4	18606	1	0.76		NB3
5	13196	1	0.99	*	AAS1
6	21459	1	0.5		AU1
7	17473	1	0.87		PF1
8	16548	1	1	*	AL2
9	2497	1.12	0.6		AS1
10	18286	1	0.81		APP1
11	4637	1	0.54		UAA1
12	17513	1	0.98	*	R_AE1
13	10996	1.01	0.51		UE2
14	11647	1	0.86		UT3
15	3365	1.03	0.56		UAL1
16	15711	1	0.98	*	AA2

ID	n_eff	Rhat	P_posterior	zero_excl	Item
17	13989	1	0.8		R_AC1
18	18329	1	0.5		R_AI3
19	25241	1	0.5		AT1
20	10593	1	0.92		SP2
21	24291	1	0.5		IIS2
22	20715	1	0.5		R_AEI3
23	17882	1	0.89		UEP3
24	17288	1	0.55		UAI4

```r
# Calculate grand (abs) mean results
Variable <- c("mean_Dutch", "sd_Dutch", "mean_English", "sd_English",
    "mean_diff", "sd_diff", "minimum_diff", "maximum_diff", "n_zero_excl",
    "percent_zero_excl")
# Calculate grand mean of mean_Dutch, sd_Dutch,
# mean_English, sd_English sd_diff, grand mean of the
# absolute value of mean differences, number of
# representative items with credible bias indication, and
# percentage of these items
Grand_mean <- c(mean(rep_list$mean_Dutch), mean(rep_list$sd_Dutch),
    mean(rep_list$mean_English), mean(rep_list$sd_English), mean(abs(rep_list$mean_diff)),
    mean(rep_list$sd_diff), min(rep_list$mean_diff), max(rep_list$mean_diff),
    sum(rep_list$zero_excl == "*"), round(sum(rep_list$zero_excl ==
        "*")/length(rep_list$ID), digits = 4) * 100)
GrandMean <- cbind(Variable, Grand_mean)

pander(GrandMean, caption = "Grand mean of 24 representative items")
```

Table 21: Grand mean of 24 representative items

Variable	Grand_mean
mean_Dutch	0.379513888888889
sd_Dutch	1.63312118336193
mean_English	0.342361111111111
sd_English	1.63035275661436
mean_diff	0.106713961785693
sd_diff	0.0690792698761533
minimum_diff	-0.264752186368267
maximum_diff	0.688106901421445
n_zero_excl	5
percent_zero_excl	20.83

References

Cicchetti, Domenic V. 1994. "Guidelines, Criteria, and Rules of Thumb for Evaluating Normed and Standardized Assessment Instruments in Psychology." *Psychological Assessment* 6 (4): 284. https://doi.org/10.1037/1040-3590.6.4.284.

Finch, W Holmes, Jocelyn E Bolin, and Ken Kelley. 2019. *Multilevel Modeling Using r*. Crc Press.

Li, Fengxiang, Siska Fitrianie, Merijn Bruijnes, Amal Abdulrahman, Fu Guo, and Willem-Paul Brinkman. 2023. "Mandarin Chinese Translation of the Artificial-Social-Agent Questionnaire Instrument for Evaluating Human-Agent Interaction." *Frontiers in Computer Science* 5.