Final assessment part 2: Variation between the orginal English and the newly translated Dutch ASA Questionnaire

Johan Hensman

21 June, 2023

Contents

1 Introduction

This document presents statistical analyses of variation between English and Dutch ASA questionnaires for item level, construct/dimension level, and short versions of the ASA questionnaire as reported in the paper. Code based on Fengxiang Li with adaptations made to Dutch.

We use the following packages:

```r
library(foreign) # Open various data files
library(nlme)    # Run multilevel linear models
library(car)     # Package linear regression
library(haven) # Use read_sav fuction
library(BayesianFirstAid)
library(dplyr) # Use select function
library(knitr) # Get markdown file
library(tinytex) # Use TeX environment
library(pander) # For pandering tables
panderOptions("table.alignment.default","left")
```

2 Data files

The input data for this analysis is based on 3 files. These aren't the fully raw data collected on Qualtrics as it pre-processed to anonymize and delete columns that are not needed/allowed to publish.

2.1 Final__ASA__Dutch__Round__1__First__Half__anonym.sav"

The first survey was split in two to prevent fatigue. In the first group first 12 constructs were asked and the second group the last 12 constructs. The participants rated 44 and 46 questionnaire items each and 14 attention checks that all had to be correct. The participants had the option to not recommend their data, but this was of importance when creating the questionnaire. In this final assessment all data will be used. this file contains data of the first 12 constructs

2.2 Final__ASA__Dutch__Round__1__Second__Half__anonym.sav

this file contains data from the last 12 constructs

2.3 Final__ASA__Dutch__Round__2__anonym.sav

After the first round was performed 37 items needed to be retranslated as they had a low correlation value. We received 37 new translation as well as 27 alternative translations that needed to be evaluated. The results of that survey is saved in this file.

```r
dataset <- read_sav("Final_ASA_Dutch_Round_1_First_Half_anonym.sav")
dataset2 <- read_sav("Final_ASA_Dutch_Round_1_Second_Half_anonym.sav")
dataset3 <- read_sav("Final_ASA_Dutch_Round_2_anonym.sav")
#Importing data
```

```
#"dataset" contains ASA question items from the first 12 constructs
#"dataset2" contains ASA question items from the last 12 constructs
```

The data has some columns that are not necessary for this analysis, so we only select the attention checks and the questionnaire items of the survey.

```
dataset <- data.frame(select(dataset, AC_English_1:Q_ITEMS_438.0))
dataset2 <- data.frame(select(dataset2, AC_English_1:Q42_442))
dataset3 <- data.frame(select(dataset3, AC_Dutch_1:English_UAI3))
#Selecting only relevant question items, which include attention checks
#and ASA questionnaire items
```

Here, we change the column names of such that the ASA questionnaire items are renamed according to which language they are in, and which construct they belong to. In the Legend Q2.txt you can find the questions that they belonged to.

```
#Here, all the column names are changed so that ASA questionnaire items are
#named according to their construct.
colnames(dataset)[59:102] <- c("D_HLA1", "D_HLA2", "D_HLA3", "D_HLA4", "D_HLB1",
                               "D_HLB2", "D_HLB3", "D_HLB4", "D_HLB5", "D_NA1",
                               "D_NA2", "D_NA3", "D_NA4", "D_NA5", "D_NB1",
                               "D_NB2", "D_NB3", "D_AAS1", "D_AAS2", "D_AAS3",
                               "D_AU1", "D_AU2", "D_AU3", "D_PF1", "D_PF2",
                               "D_PF3", "D_AL1", "D_AL2", "D_R_AL3", "D_AL4",
                               "D_AL5", "D_AS1", "D_AS2", "D_AS3", "D_APP1",
                               "D_R_APP2", "D_APP3", "D_UAA1", "D_UAA2",
                               "D_R_UAA3", "D_R_AE1", "D_AE2", "D_AE3",
                               "D_R_AE4")
#Changing columns of Dutch ASA question items in dataset

colnames(dataset2)[61:106] <- c("D_UE1", "D_UE2", "D_UE3", "D_UT1", "D_UT2",
                                "D_UT3", "D_UAL1", "D_UAL2", "D_UAL3", "D_UAL4",
                                "D_UAL5", "D_UAL6", "D_AA1", "D_AA2", "D_AA3",
                                "D_R_AC1", "D_R_AC2", "D_R_AC3", "D_R_AC4",
                                "D_AI1", "D_AI2", "D_R_AI3", "D_AI4", "D_AT1",
                                "D_AT2", "D_R_AT3", "D_SP1", "D_SP2", "D_SP3",
                                "D_IIS1", "D_IIS2", "D_IIS3", "D_IIS4",
                                "D_AEI1", "D_AEI2", "D_R_AEI3", "D_AEI4",
                                "D_R_AEI5", "D_UEP1", "D_UEP2", "D_UEP3",
                                "D_UEP4", "D_UAI1", "D_UAI2", "D_UAI3",
```

```
                                  "D_UAI4")
#Changing columns of Dutch ASA question items in dataset2

colnames(dataset)[8:51] <- c("HLA1", "HLA2", "HLA3", "HLA4", "HLB1", "HLB2",
                             "HLB3", "HLB4", "HLB5", "NA1", "NA2", "NA3", "NA4",
                             "NA5", "NB1", "NB2", "NB3", "AAS1", "AAS2", "AAS3",
                             "AU1", "AU2", "AU3", "PF1", "PF2", "PF3", "AL1",
                             "AL2", "R_AL3", "AL4", "AL5", "AS1", "AS2", "AS3",
                             "APP1", "R_APP2", "APP3", "UAA1", "UAA2", "R_UAA3",
                             "R_AE1", "AE2", "AE3", "R_AE4")
#Changing columns of English ASA question items in dataset

colnames(dataset2)[8:53] <- c("UE1", "UE2", "UE3", "UT1", "UT2", "UT3", "UAL1",
                              "UAL2", "UAL3", "UAL4", "UAL5", "UAL6", "AA1",
                              "AA2", "AA3", "R_AC1", "R_AC2", "R_AC3", "R_AC4",
                              "AI1", "AI2", "R_AI3", "AI4", "AT1", "AT2",
                              "R_AT3", "SP1", "SP2", "SP3", "IIS1", "IIS2",
                              "IIS3", "IIS4", "AEI1", "AEI2", "R_AEI3", "AEI4",
                              "R_AEI5", "UEP1", "UEP2", "UEP3", "UEP4", "UAI1",
                              "UAI2", "UAI3", "UAI4")
#Changing columns of English ASA question items in dataset2
```

As the data got imported from Qualtrics not all cells had numeric values. So, we remove all text from the cells and make the text numerical.

```
dataset <- as.data.frame(lapply(dataset[1:102],
                              function(y) as.numeric(gsub('[a-zA-Z]', '', y))))
dataset2 <- as.data.frame(lapply(dataset2[1:106],
                              function(y) as.numeric(gsub('[a-zA-Z]', '', y))))
dataset3 <- as.data.frame(lapply(dataset3[1:115],
                              function(y) as.numeric(gsub('[a-zA-Z]', '', y))))
#Transform the data to numeric representation
```

In this section we can omit the attention checks and only keep the questionnaire items.

```
d1 <- as.data.frame(select(dataset, HLA1:R_AE4, D_HLA1:D_R_AE4))
# Select scores of 44 English items and corresponding Dutch translations
d2 <- as.data.frame(select(dataset2, UE1:UAI4, D_UE1:D_UAI4))
# Select scores of 46 English items and corresponding Dutch translations
```

```r
d3 <- as.data.frame(select(dataset3, English_HLA1:English_UAI3, Dutch_HLA:Dutch_UAI3))
# Select scores of 37 English items and corresponding Dutch translations
```

We manually checked which translation had the highest ICC value and picked them to create the variable with the best values. The chosen translations and their corresponding ICC value can be found in the same folder of this codebase under Best_ICC_and_translations.xlsx. To confirm these values we calculate the ICC values again from the datafiles. So we combine the necessary columns from each dataset.

```r
d_total1 <- cbind(select(d1,HLA1), select(d3,English_HLA1),
                  select(d1,HLA3:HLB3), select(d3,English_HLB2),
                  select(d1,HLB5:NA1), select(d3,English_NA1:English_NB1),
                  select(d1, NB2),  select(d3, English_NB3), select(d1, AAS1),
                  select(d3, English_AAS1:English_AAS2), select(d1, AU1:AU2),
                  select(d3, English_AU1), select(d1, PF1:PF3),
                  select(d3, English_AL1), select(d1, AL2:AL5),
                  select(d3, English_AS1), select(d1, AS2),
                  select(d3, English_AS2), select(d1,APP1:R_UAA3),
                  select(d3,English_AE1:English_AE2), select(d1, AE3:R_AE4),
                  select(d2, UE1), select(d3,English_UE1), select(d2, UE3:UT1),
                  select(d3, English_UT1), select(d2, UT3:UAL1),
                  select(d3, English_UAL1), select(d2, UAL3:AA2),
                  select(d3, English_AA2), select(d2, R_AC1:AI4),
                  select(d3, English_AT1), select(d2, AT2:R_AT3),
                  select(d3, English_SP1), select(d2, SP2:SP3),
                  select(d3, English_IIS1: English_AEI1),
                  select(d2, AEI2:R_AEI5),
                  select(d3, English_UEP), select(d2, UEP2:UAI1),
                  select(d3, English_UAI1: English_UAI3))
#select all English items from their correspoding datasets.


d_total2 <- cbind(select(d1,D_HLA1), select(d3,Dutch_HLA),
                  select(d1,D_HLA3:D_HLB3), select(d3,Dutch_HLB2),
                  select(d1,D_HLB5:D_NA1), select(d3,Dutch_NA1),
                  select(d3,Dutch_NA2_Alt1), select(d3,Dutch_NA3),
                  select(d3,Dutch_NA4_Alt2:Dutch_NB1),select(d1, D_NB2),
                  select(d3,Dutch_NB3_Alt2) , select(d1, D_AAS1),
                  select(d3, Dutch_AAS1), select(d3, Dutch_AAS2_Alt4),
                  select(d1, D_AU1:D_AU2), select(d3, Dutch_AU1_Alt1),
```

```
                 select(d1, D_PF1:D_PF3), select(d3, Dutch_AL1_Alt1),
                 select(d1, D_AL2: D_AL5), select(d3, Dutch_AS1_Alt1),
                 select(d1, D_AS2), select(d3, Dutch_AS2_Alt3),
                 select(d1,D_APP1:D_R_UAA3), select(d3, Dutch_AE1_Alt1),
                 select(d3, Dutch_AE2), select(d1, D_AE3:D_R_AE4),
                 select(d2, D_UE1), select(d3, Dutch_UE1_Alt2),
                 select(d2, D_UE3:D_UT1), select(d3, Dutch_UT1_Alt1),
                 select(d2, D_UT3:D_UAL1), select(d3, Dutch_UAL1_Alt1),
                 select(d2, D_UAL3:D_AA2), select(d3, Dutch_AA2),
                 select(d2, D_R_AC1:D_AI4), select(d3, Dutch_AT1_Alt1),
                 select(d2, D_AT2:D_R_AT3), select(d3, Dutch_SP),
                 select(d2, D_SP2:D_SP3), select(d3, Dutch_IIS1_Alt2),
                 select(d3, Dutch_IIS2_Alt2), select(d3, Dutch_IIS3_Alt2),
                 select(d3, Dutch_IIS4: Dutch_AEI1_Alt1),
                 select(d2, D_AEI2:D_R_AEI5), select(d3, Dutch_UEP1_Alt1),
                 select(d2, D_UEP2:D_UAI1),
                 select(d3, Dutch_UAI1: Dutch_UAI3))
#select all Dutch items from their correspoding datasets.

d_total_Item <- cbind(d_total1, d_total2)
# Combine evaluation scores of the best items across the 3 data files.
```

As the columns of the 2nd round didn't had the proper naming, we renamed all columns again. (most values contained xx_xx_Alt 1 as there were multiple translation, but we want only the proper tags in the end.)

```
colnames(d_total_Item)[1:180] <- c("HLA1", "HLA2", "HLA3", "HLA4", "HLB1", "HLB2",
                             "HLB3", "HLB4", "HLB5", "NA1", "NA2", "NA3", "NA4",
                             "NA5", "NB1", "NB2", "NB3", "AAS1", "AAS2", "AAS3",
                             "AU1", "AU2", "AU3", "PF1", "PF2", "PF3", "AL1",
                             "AL2", "R_AL3", "AL4", "AL5", "AS1", "AS2", "AS3",
                             "APP1", "R_APP2", "APP3", "UAA1", "UAA2", "R_UAA3",
                             "R_AE1", "AE2", "AE3", "R_AE4","UE1", "UE2", "UE3",
                             "UT1", "UT2", "UT3", "UAL1", "UAL2", "UAL3",
                             "UAL4", "UAL5", "UAL6", "AA1", "AA2", "AA3",
                             "R_AC1", "R_AC2", "R_AC3", "R_AC4", "AI1", "AI2",
                             "R_AI3", "AI4", "AT1", "AT2", "R_AT3", "SP1",
                             "SP2", "SP3", "IIS1", "IIS2", "IIS3", "IIS4",
                             "AEI1", "AEI2", "R_AEI3", "AEI4", "R_AEI5", "UEP1",
                             "UEP2", "UEP3", "UEP4", "UAI1", "UAI2", "UAI3",
```

```
                              "UAI4", "D_HLA1", "D_HLA2", "D_HLA3", "D_HLA4",
                              "D_HLB1", "D_HLB2", "D_HLB3", "D_HLB4", "D_HLB5",
                              "D_NA1", "D_NA2", "D_NA3", "D_NA4", "D_NA5",
                              "D_NB1", "D_NB2", "D_NB3", "D_AAS1", "D_AAS2",
                              "D_AAS3", "D_AU1", "D_AU2", "D_AU3", "D_PF1",
                              "D_PF2","D_PF3", "D_AL1", "D_AL2", "D_R_AL3",
                              "D_AL4", "D_AL5", "D_AS1", "D_AS2", "D_AS3",
                              "D_APP1", "D_R_APP2", "D_APP3", "D_UAA1",
                              "D_UAA2", "D_R_UAA3", "D_R_AE1", "D_AE2", "D_AE3",
                              "D_R_AE4", "D_UE1", "D_UE2", "D_UE3", "D_UT1",
                              "D_UT2", "D_UT3", "D_UAL1", "D_UAL2", "D_UAL3",
                              "D_UAL4", "D_UAL5", "D_UAL6", "D_AA1", "D_AA2",
                              "D_AA3", "D_R_AC1", "D_R_AC2", "D_R_AC3",
                              "D_R_AC4", "D_AI1", "D_AI2", "D_R_AI3", "D_AI4",
                              "D_AT1", "D_AT2", "D_R_AT3", "D_SP1", "D_SP2",
                              "D_SP3", "D_IIS1", "D_IIS2", "D_IIS3", "D_IIS4",
                              "D_AEI1", "D_AEI2", "D_R_AEI3", "D_AEI4",
                              "D_R_AEI5", "D_UEP1", "D_UEP2", "D_UEP3",
                              "D_UEP4", "D_UAI1", "D_UAI2", "D_UAI3", "D_UAI4")
```

we already prepare also the data we need for the construct level. As construct level icc is only valid if all questions from 1 construct are answered by the same person, we had to take almost all values from the First dataset(which was split in two). There was one construct where all questions were asked in round 2 and resulted in a higher ICC value. So the construct level data contains 1 construct (with the tag IIS) from Round 2 and the rest originate from round 1. The necessary columns were renamed to show the correct tags.

```
d_total1 <- cbind(select(d1,HLA1:R_AE4), select(d2,UE1:SP3),
                  select(d3, English_IIS1: English_IIS4), select(d2, AEI1:UAI4))
#select all English items from their correspoding datasets.

d_total2 <- cbind(select(d1,D_HLA1:D_R_AE4), select(d2,D_UE1:D_SP3),
                  select(d3, Dutch_IIS1_Alt2), select(d3, Dutch_IIS2_Alt2),
                  select(d3, Dutch_IIS3_Alt2), select(d3, Dutch_IIS4),
                  select(d2, D_AEI1:D_UAI4))
#select all Dutch items from their correspoding datasets.

d_total_Construct <- cbind(d_total1, d_total2)
#combine the items into 1.
```

```r
colnames(d_total_Construct)[74:77] <- c("IIS1", "IIS2", "IIS3", "IIS4")
colnames(d_total_Construct)[164:167] <-c("D_IIS1", "D_IIS2", "D_IIS3", "D_IIS4")
# rename the necessary columns.
```

All scores that have an R_ in front have to be reversed. This was necessary for items like 'I don't like [the agent]' and 'I like [the agent]' as it will have opposite values so questions like 'i don't like [the agent]' are reversed.

```r
for (i in grep("R_",colnames(d_total_Construct))){
# Find column number of reversing-scoring items and translations
  d_total_Construct[[i]][] <- d_total_Construct[[i]][]*(-1)
  # Reverse scores of reverse-scoring items and translations
}

for (i in grep("R_",colnames(d_total_Item))){
# Find column number of reversing-scoring items and translations
  d_total_Item[[i]][] <- d_total_Item[[i]][]*(-1)
  # Reverse scores of reverse-scoring items and translations
}
```

3 Analyses results reported in Section Results

The results were reported in the results section of the paper. The mean score differences between the English and Dutch questionnaires are estimates for how well the translated questionnaire is compared to the original one. 95% credible interval of mean paired difference was calculated by Bayesian paired *t*-test, for item level, construct/dimension level, and the short version of the ASA questionnaire.

3.1 Model/function for variation calculation

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 the function to return the necessary outputs from the Bayesian paired *t*-test.

```r
getBAYES <-function(ID, ss_1, ss_2, B_output)
# 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
```

```r
# ID is the identification number added in the return data
# frame row to identify an item or construct
{ 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
                  zero_excl = ifelse((as.numeric(B_output[["stats"]][1,5])>0)   # HDIlo
                                     | (as.numeric(B_output[["stats"]][1,6])<0),# HDIup
                                     '*','')
#add “*” marker if the low bound of HDI is large than zero,
# or the upper bound is smaller than zero
  )
return(l)  # Line 1 in the bayes.t.test output of mu_diff
}
```

3.2 Mean score differences for 90 items

This chunk shows how we calculated the Variation of the 90 items of the English ASA questionnaire with the Dutch variant.

```r
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())
# Initialize output of Items with credible bias indication

set.seed(1) # Make sure that estimations of Bayesian analyses remain the same
n <- ncol(d_total_Item)
# Numbers of all columns in d_total, i.e. English and Dutch scores combined
Dutch_column_offset <- n /2
```

```r
# Offset for the column position of the first Dutch ASAQ items

for (i in 1:90)
# Go step by step to 90 ASA questionnaire items
{
  score_Dutch <- d_total_Item[,i+ Dutch_column_offset] # Dutch scores
  score_English <- d_total_Item[,i] # English item scores
  fit <- bayes.t.test(score_Dutch, score_English, paired = TRUE)
  # conduct a Bayesian paired t-test on the Dutch and English score of ASAQ item
  item_list <- rbind(item_list, getBAYES(i, score_Dutch, score_English, fit))
  # store results from Bayesian analysis in a list to print later
}

# Print results
item_list$Item = colnames(select(d_total_Item,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 1: Items with credible bias indication (Part 1)

ID	mean_Dutch	sd_Dutch	mean_English	sd_English	Item
1	-1.7	1.343	-1.667	1.446	HLA1
2	-1.2	1.584	-1.667	1.348	HLA2
3	-1.633	1.217	-1.833	1.315	HLA3
4	-1.367	1.497	-1.433	1.654	HLA4
5	-2.1	1.125	-1.9	1.094	HLB1
6	-0.7333	1.68	-0.6	1.632	HLB2
7	-1.133	1.592	-0.9	1.709	HLB3
8	-0.7	1.803	-1.367	1.189	HLB4
9	-0.3667	1.65	-0.7667	1.695	HLB5
10	-2.733	0.5208	-2.6	0.8137	NA1
11	-0.8333	1.51	-1.133	1.502	NA2
12	-0.7333	1.596	-1.033	1.81	NA3
13	-1.267	1.53	-1.467	1.592	NA4
14	-0.1	1.296	-0.4	1.589	NA5
15	-1.733	1.596	-2.2	1.297	NB1
16	-1.6	1.133	-1.167	1.577	NB2
17	-1.3	1.725	-1.433	1.278	NB3

ID	mean_Dutch	sd_Dutch	mean_English	sd_English	Item
18	1.1	1.447	1.1	1.539	AAS1
19	1.667	0.9942	1.233	1.591	AAS2
20	1.033	1.299	1.167	1.262	AAS3
21	0.5333	1.717	0.6333	1.564	AU1
22	0.5	1.57	0.5333	1.456	AU2
23	0.7667	1.357	0.8	1.297	AU3
24	0.3667	1.608	0.3333	1.583	PF1
25	1.333	1.647	0.9333	1.53	PF2
26	0.3667	1.351	0.5333	1.479	PF3
27	0.9	1.155	1.067	1.461	AL1
28	0.5333	1.502	0.4333	1.633	AL2
29	0.7	1.878	0.8333	1.859	R_AL3
30	0.7	1.393	0.7333	1.484	AL4
31	-0.4667	1.943	-0.6	1.886	AL5
32	-0.4333	1.569	-1.133	1.613	AS1
33	-0.7	1.557	-0.5667	1.695	AS2
34	1	1.313	0.3667	1.497	AS3
35	-0.6667	1.561	-0.8	1.518	APP1
36	-0.9333	1.552	-0.7	1.664	R_APP2
37	-1.567	1.357	-1.433	1.382	APP3
38	0.06667	1.68	0.3333	1.493	UAA1
39	0.3333	1.322	0.4	1.38	UAA2
40	1.6	1.673	1.533	1.717	R_UAA3
41	0.2667	1.741	0.3333	1.749	R_AE1
42	1.333	1.241	1.1	1.494	AE2
43	1.033	1.847	1.167	1.51	AE3
44	0.8	1.789	1.267	1.574	R_AE4
45	2.1	1.094	2.1	0.8449	UE1
46	1.8	0.8469	1.667	0.8442	UE2
47	2.233	0.8976	2.333	0.6609	UE3
48	-0.5333	1.306	-0.7333	1.172	UT1
49	0.6667	1.061	0.4333	1.633	UT2
50	-0.06667	1.66	0.06667	1.413	UT3
51	-0.3667	1.81	-0.7	1.466	UAL1
52	0.2	1.243	-0.03333	1.45	UAL2
53	0.1667	1.367	-0.03333	1.586	UAL3
54	0.4	1.453	0.4667	1.548	UAL4
55	-0.4667	1.756	-1.1	1.626	UAL5

ID	mean_Dutch	sd_Dutch	mean_English	sd_English	Item
56	-0.5	1.676	-0.3667	1.608	UAL6
57	0.9333	1.929	0.6333	2.059	AA1
58	0.4	1.453	0.5333	1.717	AA2
59	1.133	1.196	0.4	1.453	AA3
60	0.3	1.968	0.5333	1.833	R_AC1
61	0.9667	1.884	0.8333	1.895	R_AC2
62	0.6667	1.668	0.6667	1.807	R_AC3
63	-0.4	1.94	-0.1	1.971	R_AC4
64	0.7	1.489	0.4667	1.57	AI1
65	-0.2667	1.721	-0.1333	1.717	AI2
66	-0.1667	2.036	-0.06667	1.78	R_AI3
67	0.1	1.768	-0.03333	1.732	AI4
68	1.133	1.252	1.033	1.351	AT1
69	1.1	1.296	1.133	1.279	AT2
70	1.567	1.501	1.467	1.279	R_AT3
71	0.6	1.248	0.2	1.243	SP1
72	-0.5333	1.871	-0.2667	2.05	SP2
73	-1.067	1.946	-0.9	2.023	SP3
74	-0.2	1.472	-0.4	1.329	IIS1
75	0.1	1.583	0.3333	1.269	IIS2
76	0.4	1.38	0.5333	1.137	IIS3
77	0.4	1.248	0.3667	1.326	IIS4
78	-1.9	1.094	-1.933	1.202	AEI1
79	-1.833	1.416	-1.8	1.472	AEI2
80	-1.367	1.771	-1.6	1.589	R_AEI3
81	-0.7667	1.633	-0.8	1.71	AEI4
82	-1.6	1.694	-1.7	1.489	R_AEI5
83	1	1.05	0.9667	1.189	UEP1
84	1	1.531	0.9	1.447	UEP2
85	1.533	1.252	1.467	1.224	UEP3
86	0.7667	1.382	1.067	1.701	UEP4
87	0.8	1.669	0.5	1.57	UAI1
88	0	1.93	-0.4	1.632	UAI2
89	-0.3667	1.691	0	1.682	UAI3
90	-0.1	1.539	0.3333	1.322	UAI4

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

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

ID	mean_diff	sd_diff	HDIlo	HDIup	Item
1	0.002753	0.196	-0.3932	0.3905	HLA1
2	0.2842	0.2772	-0.1267	0.8378	HLA2
3	0.1916	0.1888	-0.1827	0.5587	HLA3
4	0.09786	0.2095	-0.3092	0.5159	HLA4
5	-0.1945	0.186	-0.5547	0.177	HLB1
6	-0.1361	0.333	-0.8151	0.5004	HLB2
7	-0.1907	0.2017	-0.5835	0.204	HLB3
8	0.6724	0.2588	0.1532	1.171	HLB4
9	0.3178	0.2461	-0.1686	0.8067	HLB5
10	1.218e-07	9.414e-05	-0.0001867	0.0001838	NA1
11	0.3598	0.3173	-0.2435	1.001	NA2
12	0.1462	0.1988	-0.1228	0.5949	NA3
13	0.2003	0.244	-0.2645	0.6979	NA4
14	0.2839	0.2798	-0.257	0.838	NA5
15	-1.298e-06	0.000315	-0.000625	0.0005848	NB1
16	-0.4072	0.2562	-0.9141	0.09451	NB2
17	0.1372	0.3113	-0.4621	0.7592	NB3
18	0.02549	0.2198	-0.4091	0.4625	AAS1
19	0.3788	0.231	-0.07554	0.8317	AAS2
20	-0.1368	0.1527	-0.4377	0.1611	AAS3
21	-0.09231	0.2205	-0.5305	0.3433	AU1
22	0.01882	0.1997	-0.3777	0.4122	AU2
23	-0.03981	0.1731	-0.3976	0.3052	AU3
24	0.05014	0.1662	-0.2862	0.3729	PF1
25	0.4062	0.2255	-0.03233	0.8576	PF2
26	-0.15	0.225	-0.5964	0.2899	PF3
27	-0.1512	0.1987	-0.5524	0.2381	AL1
28	0.1255	0.1569	-0.1693	0.4478	AL2
29	-0.13	0.2218	-0.6351	0.2672	R_AL3
30	1.045e-05	0.0005586	-0.0009258	0.0009474	AL4
31	0.007682	0.04665	-0.01947	0.0529	AL5
32	0.7064	0.2561	0.2161	1.223	AS1
33	-0.1272	0.2512	-0.6233	0.3619	AS2

ID	mean_diff	sd_diff	HDIlo	HDIup	Item
34	0.5656	0.2109	0.1472	0.9748	AS3
35	0.1384	0.2124	-0.2715	0.5632	APP1
36	-0.2088	0.1865	-0.5811	0.1553	R_APP2
37	-0.1292	0.2045	-0.5429	0.2724	APP3
38	-0.2334	0.2514	-0.7302	0.2564	UAA1
39	-0.04757	0.1881	-0.4225	0.3189	UAA2
40	-0.002504	0.09386	-0.2073	0.2102	R_UAA3
41	-0.1494	0.2378	-0.6192	0.3245	R_AE1
42	1.736e-06	0.0002776	-0.0005216	0.0005113	AE2
43	1.933e-06	0.0003194	-0.0006121	0.0006481	AE3
44	-0.3904	0.2186	-0.8182	0.03274	R_AE4
45	6.422e-07	0.0001822	-0.0003657	0.0003548	UE1
46	-8.247e-07	8.692e-05	-0.0001669	0.0001753	UE2
47	-6.132e-07	0.0001057	-0.0002065	0.0002084	UE3
48	-9.4e-07	0.0006444	-0.0009699	0.0009986	UT1
49	0.1878	0.2132	-0.2363	0.606	UT2
50	1.583e-06	0.0002237	-0.0004006	0.0004045	UT3
51	0.3357	0.2263	-0.09423	0.7893	UAL1
52	0.2512	0.2338	-0.212	0.7046	UAL2
53	0.1685	0.251	-0.3248	0.6614	UAL3
54	-0.06959	0.2146	-0.4868	0.3522	UAL4
55	0.6028	0.273	0.05703	1.128	UAL5
56	-0.1138	0.2763	-0.6524	0.4328	UAL6
57	0.2507	0.2597	-0.257	0.7649	AA1
58	-0.1038	0.3037	-0.6997	0.4902	AA2
59	0.691	0.1842	0.3377	1.063	AA3
60	-0.1305	0.243	-0.6053	0.3507	R_AC1
61	3.407e-09	0.0001702	-0.0003349	0.000336	R_AC2
62	-0.0563	0.2188	-0.487	0.3781	R_AC3
63	7.211e-06	0.0006924	-0.001023	0.001049	R_AC4
64	0.2249	0.1765	-0.1282	0.5668	AI1
65	-0.09174	0.1924	-0.4795	0.2798	AI2
66	-0.05925	0.2221	-0.4969	0.3779	R_AI3
67	0.03278	0.1185	-0.1929	0.3247	AI4
68	0.1109	0.1906	-0.2744	0.4869	AT1
69	-0.009683	0.1774	-0.3619	0.3377	AT2
70	0.1176	0.1731	-0.1951	0.4918	R_AT3
71	0.4111	0.1938	0.02	0.7886	SP1

ID	mean_diff	sd_diff	HDIlo	HDIup	Item
72	-0.2155	0.2505	-0.7235	0.2692	SP2
73	-0.1891	0.262	-0.7014	0.3304	SP3
74	8.141e-07	0.0001763	-0.0003429	0.0003466	IIS1
75	-0.2317	0.2081	-0.6297	0.1823	IIS2
76	-0.1252	0.2003	-0.5292	0.2606	IIS3
77	0.01585	0.2308	-0.4447	0.465	IIS4
78	0.02837	0.1892	-0.3539	0.419	AEI1
79	2.669e-06	0.0002276	-0.0004284	0.0004596	AEI2
80	8.327e-09	0.0001544	-0.0003049	0.000307	R_AEI3
81	-0.07684	0.1862	-0.432	0.3063	AEI4
82	9.913e-07	0.0001488	-0.0002825	0.0003033	R_AEI5
83	-0.007975	0.2103	-0.423	0.4093	UEP1
84	0.06041	0.1657	-0.2433	0.464	UEP2
85	5.382e-06	0.0006134	-0.0008324	0.0008574	UEP3
86	-0.3226	0.2425	-0.817	0.1407	UEP4
87	0.2407	0.2269	-0.2215	0.6751	UAI1
88	0.3316	0.2771	-0.1964	0.8894	UAI2
89	-0.3634	0.1618	-0.6906	-0.0532	UAI3
90	-0.3781	0.2814	-0.9334	0.1813	UAI4

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

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

ID	n_eff	Rhat	P_posterior	zero_excl	Item
1	16692	1	0.5154		HLA1
2	1203	1.009	0.8457		HLA2
3	17661	1	0.8531		HLA3
4	18439	1	0.687		HLA4
5	18067	1	0.8546		HLB1
6	16982	1.001	0.6605		HLB2
7	16280	1	0.8301		HLB3
8	18384	1	0.9953	*	HLB4
9	15350	1	0.9103		HLB5
10	19328	1	0.5016		NA1
11	18564	1	0.875		NA2

ID	n_eff	Rhat	P_posterior	zero_excl	Item
12	1265	1.01	0.7702		NA3
13	15841	1.001	0.8015		NA4
14	17571	1	0.8478		NA5
15	21225	1.001	0.5015		NB1
16	15656	1	0.9449		NB2
17	19006	1	0.6744		NB3
18	15465	1.001	0.5576		AAS1
19	16946	1	0.9516		AAS2
20	17646	1	0.8227		AAS3
21	17818	1	0.662		AU1
22	16671	1	0.5419		AU2
23	15614	1.001	0.598		AU3
24	18053	1	0.6199		PF1
25	18807	1	0.9628		PF2
26	18238	0.9999	0.7512		PF3
27	18789	1	0.7819		AL1
28	10578	1	0.792		AL2
29	4964	1.001	0.7271		R_AL3
30	4438	1.005	0.5029		AL4
31	20067	1	0.5228		AL5
32	18370	1	0.9966	*	AS1
33	18202	1	0.6973		AS2
34	16304	1	0.9971	*	AS3
35	17404	1.001	0.7487		APP1
36	16560	1	0.8734		R_APP2
37	15692	1	0.743		APP3
38	14541	1.001	0.8243		UAA1
39	17827	1	0.6007		UAA2
40	3577	1.002	0.5		R_UAA3
41	16934	0.9999	0.7472		R_AE1
42	13991	1	0.5009		AE2
43	17593	1.001	0.5011		AE3
44	13603	1	0.9673		R_AE4
45	22034	1	0.5006		UE1
46	18890	1	0.5018		UE2
47	20079	1.001	0.5031		UE3
48	7579	1.019	0.5022		UT1
49	17383	1	0.8126		UT2

ID	n_eff	Rhat	P_posterior	zero_excl	Item
50	16554	1	0.5013		UT3
51	18634	1	0.9335		UAL1
52	17486	1.001	0.8639		UAL2
53	16959	1.001	0.7529		UAL3
54	18020	1	0.6304		UAL4
55	18075	1	0.9864	*	UAL5
56	18664	1	0.6633		UAL6
57	15307	1	0.8376		AA1
58	17787	1	0.6344		AA2
59	16543	1	0.9997	*	AA3
60	15410	0.9999	0.7016		R_AC1
61	19357	1	0.5021		R_AC2
62	16413	1	0.6096		R_AC3
63	15689	1.042	0.5011		R_AC4
64	16993	1	0.9		AI1
65	16378	1	0.6817		AI2
66	18641	0.9999	0.6068		R_AI3
67	4488	1.022	0.59		AI4
68	17738	1	0.7273		AT1
69	17782	1	0.5183		AT2
70	4067	1.006	0.7503		R_AT3
71	18824	1	0.9812	*	SP1
72	16840	1.001	0.8113		SP2
73	18414	1	0.7696		SP3
74	21561	1.001	0.5		IIS1
75	18968	1	0.8696		IIS2
76	18040	1	0.737		IIS3
77	17638	1	0.5253		IIS4
78	12221	1.002	0.5507		AEI1
79	21182	1	0.5025		AEI2
80	25899	1.001	0.5003		R_AEI3
81	12979	1	0.678		AEI4
82	18917	1	0.5014		R_AEI5
83	17481	1	0.522		UEP1
84	3605	1.005	0.627		UEP2
85	27399	1.028	0.5022		UEP3
86	17865	1	0.9083		UEP4
87	17222	1	0.8594		UAI1

ID	n_eff	Rhat	P_posterior	zero_excl	Item
88	16029	1	0.8888		UAI2
89	16997	1	0.9867	*	UAI3
90	16616	1	0.9175		UAI4

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

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)
# 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

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

Table 4: Grand mean of 90 items

Variable	Grand_mean
mean_Dutch	0.0177777777777778
sd_Dutch	1.50149642604075
mean_English	-0.0403703703703704
sd_English	1.50386206752884
mean_diff	0.162679487126916
sd_diff	0.176586754319658
minimum_diff	-0.407158882575437
maximum_diff	0.70641549380519
n_zero_excl	7
percent_zero_excl	7.78

3.3 Mean score differences for 24 constructs and related dimensions

We first calculate the means of each construct/dimension and the we use in the same way as the item level comparison the getBayes function to obtain the mean differences for each construct. Furthermore, we again calculate the grand mean, standard deviation and range for the mean differences.

```
Dutch_column_offset <- length(d_total_Construct)/2

i <- which(names(d_total_Construct)%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"))
#vector with the indexes of the first item of all constructs.

k <- c(ncol(select(d_total_Construct, HLA1:HLA4)),
       ncol(select(d_total_Construct, HLB1:HLB5)),
       ncol(select(d_total_Construct, NA1:NA5)),
       ncol(select(d_total_Construct, NB1:NB3)),
       ncol(select(d_total_Construct, AAS1:AAS3)),
       ncol(select(d_total_Construct, AU1:AU3)),
       ncol(select(d_total_Construct, PF1:PF3)),
       ncol(select(d_total_Construct, AL1:AL5)),
       ncol(select(d_total_Construct, AS1:AS3)),
       ncol(select(d_total_Construct, APP1:APP3)),
       ncol(select(d_total_Construct, UAA1:R_UAA3)),
       ncol(select(d_total_Construct, R_AE1:R_AE4)),
       ncol(select(d_total_Construct, UE1:UE3)),
       ncol(select(d_total_Construct, UT1:UT3)),
       ncol(select(d_total_Construct, UAL1:UAL6)),
       ncol(select(d_total_Construct, AA1:AA3)),
       ncol(select(d_total_Construct, R_AC1:R_AC4)),
       ncol(select(d_total_Construct, AI1:AI4)),
       ncol(select(d_total_Construct, AT1:R_AT3)),
       ncol(select(d_total_Construct, SP1:SP3)),
       ncol(select(d_total_Construct, IIS1:IIS4)),
       ncol(select(d_total_Construct, AEI1:R_AEI5)),
       ncol(select(d_total_Construct, UEP1:UEP4)),
       ncol(select(d_total_Construct, UAI1:UAI4)))
# 'k' is a vector with the number of questionnaire items of each
# construct/dimension
```

```
h <- cbind.data.frame(i,k)
# 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

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())
# Initialize output of Constructs/dimensions with credible bias indication

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

for(p in 1:24)
# Go step by step to 24 constructs/dimensions
{
  i = h[p,1]
  # The column with the first English ASAQ item of the construct/dimension
  j = i+ Dutch_column_offset
  # The column with the first Dutch ASAQ item of the construct/dimension
  k = h[p,2] # The number of columns/items of the construct/dimension
  s_Dutch <- data.frame(d_total_Construct[,j:(j+k-1)]) # Select Dutch scores
  s_English <- data.frame(d_total_Construct[,i:(i+k-1)]) # Select English scores
  average_s_Dutch <- data.frame(rowMeans(s_Dutch))
  # Dutch score means for each construct/dimension per participant
  average_s_English <- data.frame(rowMeans(s_English))
  # English score means for each construct/dimension per participant
  colnames(average_s_Dutch) <- c("score")
  # Rename Dutch mean column
  colnames(average_s_English) <- c("score")
  # Rename English mean column
  score <- data.frame(cbind(average_s_Dutch,average_s_English))
  # Combine averaged scores of Dutch and English constructs/dimensions
  score_Dutch <- score[,1]
  # Select averaged scores of each Dutch construct/dimension,
  # make sure data format is suitable for Bayesian paired t-test
```

```
  score_English <- score[,2]
  # Select averaged scores of each English construct/dimension,
  # make sure data format is suitable for Bayesian paired t-test
  fit <- bayes.t.test(score_Dutch,score_English, paired = TRUE)
  # Conduct Bayesian t-test
  con_list <- rbind(con_list,getBAYES(p,score_Dutch,score_English,fit))
  # Call function 'getBAYES' to obtain relevant information
  # from Bayesian t-test output and add result to output list
}

# Print results
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')
# Add construct/dimension name code
pander(select(con_list,ID,mean_Dutch,sd_Dutch,mean_English,sd_English,Construct),
       caption = "Constructs/dimensions with credible bias indication (Part 1)")
```

Table 5: Constructs/dimensions with credible bias indication (Part 1)

ID	mean_Dutch	sd_Dutch	mean_English	sd_English	Construct
1	-1.667	1.173	-1.55	1.367	HLA
2	-1.087	1.06	-1.087	1.25	HLB
3	-1.747	0.786	-1.44	0.9547	NA
4	-1.9	0.7886	-1.789	0.8905	NB
5	1.044	1.298	1.122	1.246	AAS
6	0.3556	1.561	0.6778	1.317	AU
7	0.6889	1.235	0.6	1.27	PF
8	0.4	1.335	0.38	1.381	AL
9	-0.4778	1.485	-0.3111	1.271	AS
10	-1.056	1.244	-0.9778	1.171	APP
11	0.6667	1.395	0.7556	1.21	UAA
12	0.5833	1.471	0.95	1.362	AE
13	2.233	0.7487	2.211	0.7138	UE
14	0.1667	1.05	0.06667	1.102	UT
15	-0.1611	1.149	-0.3389	1.121	UAL
16	0.8778	1.463	0.6889	1.652	AA
17	0.3833	1.557	0.4833	1.611	AC
18	0.09167	1.558	0.05833	1.529	AI

ID	mean_Dutch	sd_Dutch	mean_English	sd_English	Construct
19	1.356	1.151	1.278	1.155	AT
20	-0.4889	1.606	-0.2556	1.679	SP
21	0.175	1.151	0.2083	0.9217	IIS
22	-1.433	1.423	-1.56	1.345	AEI
23	1.208	0.8537	1.2	1.084	UEP
24	0.1417	1.313	0.2583	1.304	UAI

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

Table 6: Constructs/dimensions with credible bias indication (Part 2)

ID	mean_diff	sd_diff	HDIlo	HDIup	Construct
1	-0.07852	0.1373	-0.3565	0.182	HLA
2	-0.001905	0.1017	-0.2017	0.1979	HLB
3	-0.293	0.1568	-0.5918	0.02542	NA
4	-0.0835	0.1428	-0.3666	0.1986	NB
5	-0.1664	0.183	-0.508	0.2175	AAS
6	-0.2493	0.1646	-0.5768	0.06579	AU
7	0.0773	0.1132	-0.1466	0.2971	PF
8	-0.01977	0.09935	-0.216	0.1763	AL
9	-0.1368	0.158	-0.4463	0.1737	AS
10	-0.08977	0.1229	-0.3364	0.1509	APP
11	-0.08065	0.1411	-0.3617	0.1965	UAA
12	-0.2292	0.1104	-0.4512	-0.01489	AE
13	0.02788	0.07792	-0.122	0.1829	UE
14	0.05178	0.1375	-0.2144	0.3273	UT
15	0.1866	0.1239	-0.05559	0.4302	UAL
16	0.2096	0.1644	-0.1101	0.5374	AA
17	-0.09191	0.112	-0.3125	0.1272	AC
18	0.03209	0.1394	-0.2459	0.3024	AI
19	0.1618	0.1276	-0.08717	0.418	AT
20	-0.2484	0.1594	-0.5534	0.06896	SP
21	-0.02539	0.1295	-0.2834	0.2295	IIS
22	0.1248	0.1101	-0.0869	0.3484	AEI
23	-0.02473	0.1121	-0.2378	0.2023	UEP

ID	mean_diff	sd_diff	HDIlo	HDIup	Construct
24	-0.1303	0.1647	-0.4693	0.1858	UAI

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

Table 7: Constructs/dimensions with credible bias indication (Part 3)

ID	n_eff	Rhat	P_posterior	zero_excl	Construct
1	16344	1	0.7147		HLA
2	17926	1	0.5062		HLB
3	19413	1.001	0.9692		NA
4	15633	1	0.7248		NB
5	12116	1.001	0.8282		AAS
6	11543	1	0.9399		AU
7	18176	1	0.7554		PF
8	16966	1	0.589		AL
9	16421	1	0.8116		AS
10	17009	1	0.7744		APP
11	18604	1	0.7209		UAA
12	10481	1.003	0.9838	*	AE
13	18491	1	0.6401		UE
14	13465	1	0.6381		UT
15	18536	1	0.9343		UAL
16	18313	1	0.9011		AA
17	17500	1	0.798		AC
18	17721	1	0.5945		AI
19	17288	1.001	0.8975		AT
20	19195	1	0.9409		SP
21	18703	1.001	0.5837		IIS
22	17826	1	0.8782		AEI
23	14137	1	0.6007		UEP
24	17933	1	0.7928		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")
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)
# 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
pander(GrandMean, caption = "Grand mean of 24 constructs/dimensions")
```

Table 8: Grand mean of 24 constructs/dimensions

Variable	Grand_mean
mean_Dutch	0.0148148148148148
sd_Dutch	1.24401203647195
mean_English	0.0678935185185185
sd_English	1.2460823000598
mean_diff	0.117555303793596
sd_diff	0.132894474015286
minimum_diff	-0.292999741864869
maximum_diff	0.20957144084273
n_zero_excl	1
percent_zero_excl	4.17

3.4 Mean score differences between English and Chinese short version of ASA questionnaire

In the same manner as the item and the construct/dimension level, we also calculate the mean score differences for the short representative questionnaire in the same manner as the previous sections.

```
s_Dutch <- select(d_total_Item,D_HLA2,D_HLB5,D_NA4,D_NB3,D_AAS1,D_AU1,D_PF1,
                  D_AL2,D_AS1,D_APP1,D_UAA1, D_R_AE1,D_UE2,D_UT3,D_UAL1,D_AA2,
                  D_R_AC1,D_R_AI3,D_AT1,D_SP2,D_IIS2,D_R_AEI3,D_UEP3,D_UAI4)
# Select Dutch versions of the 24 representative ASAQ items
```

```r
s_English <- select(d_total_Item,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)
# Select English versions of the 24 representative ASAQ items
d_total_Short <- cbind(s_Dutch,s_English)
# Combine Dutch and English scores
n <- ncol(d_total_Short) # Numbers of all columns in d_total_Short
Dutch_column_offset <- n /2

short_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())
# Initialize output of Representative items with credible bias indication


for (i in 1:24)
# Go step by step to 24 representative items of the ASA questionnaire
{
  score_Dutch <- as.numeric(d_total_Short[,i]) # Select Dutch scores
  score_English <- as.numeric(d_total_Short[,i+ Dutch_column_offset]) # Select English scores
  fit<- bayes.t.test(score_Dutch, score_English, paired = TRUE)
  # conduct a Bayesian paired t-test on the Chinese and English score of ASAQ item
  short_list <- rbind(short_list, getBAYES(i, score_Dutch, score_English, fit))
  # store results from Bayesian analysis in a list to print later
}

# Print results
short_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')
# Add item name code
pander(select(short_list,ID,mean_Dutch,sd_Dutch,mean_English,sd_English,Item),
       caption = "Representative items with credible bias indication (Part 1)")
```

Table 9: Representative items with credible bias indication (Part 1)

ID	mean_Dutch	sd_Dutch	mean_English	sd_English	Item
1	-1.2	1.584	-1.667	1.348	HLA2
2	-0.3667	1.65	-0.7667	1.695	HLB5
3	-1.267	1.53	-1.467	1.592	NA4
4	-1.3	1.725	-1.433	1.278	NB3
5	1.1	1.447	1.1	1.539	AAS1
6	0.5333	1.717	0.6333	1.564	AU1
7	0.3667	1.608	0.3333	1.583	PF1
8	0.5333	1.502	0.4333	1.633	AL2
9	-0.4333	1.569	-1.133	1.613	AS1
10	-0.6667	1.561	-0.8	1.518	APP1
11	0.06667	1.68	0.3333	1.493	UAA1
12	0.2667	1.741	0.3333	1.749	R_AE1
13	1.8	0.8469	1.667	0.8442	UE2
14	-0.06667	1.66	0.06667	1.413	UT3
15	-0.3667	1.81	-0.7	1.466	UAL1
16	0.4	1.453	0.5333	1.717	AA2
17	0.3	1.968	0.5333	1.833	R_AC1
18	-0.1667	2.036	-0.06667	1.78	R_AI3
19	1.133	1.252	1.033	1.351	AT1
20	-0.5333	1.871	-0.2667	2.05	SP2
21	0.1	1.583	0.3333	1.269	IIS2
22	-1.367	1.771	-1.6	1.589	R_AEI3
23	1.533	1.252	1.467	1.224	UEP3
24	-0.1	1.539	0.3333	1.322	UAI4

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

Table 10: Representative items with credible bias indication (Part 2)

ID	mean_diff	sd_diff	HDIlo	HDIup	Item
1	0.2865	0.2815	-0.1103	0.8738	HLA2
2	0.3153	0.2455	-0.1759	0.7969	HLB5

ID	mean_diff	sd_diff	HDIlo	HDIup	Item
3	0.1987	0.2457	-0.285	0.6859	NA4
4	0.1332	0.3116	-0.5095	0.7182	NB3
5	0.02205	0.2184	-0.4268	0.4382	AAS1
6	-0.08978	0.2209	-0.519	0.3505	AU1
7	0.04919	0.1641	-0.2742	0.3774	PF1
8	0.1259	0.156	-0.1774	0.4368	AL2
9	0.7042	0.2545	0.1893	1.205	AS1
10	0.1397	0.2103	-0.2668	0.563	APP1
11	-0.2365	0.251	-0.7305	0.2503	UAA1
12	-0.1491	0.2394	-0.6229	0.3297	R_AE1
13	5.916e-08	8.743e-05	-0.0001728	0.0001687	UE2
14	-6.611e-07	0.0002032	-0.000397	0.0003985	UT3
15	0.3345	0.2286	-0.1166	0.7782	UAL1
16	-0.1003	0.3045	-0.7094	0.4871	AA2
17	-0.1336	0.2413	-0.6136	0.3399	R_AC1
18	-0.06092	0.2215	-0.5076	0.3628	R_AI3
19	0.1095	0.1932	-0.2726	0.4964	AT1
20	-0.216	0.2494	-0.7102	0.275	SP2
21	-0.2317	0.2075	-0.6487	0.1803	IIS2
22	-2.187e-06	0.0001521	-0.0002969	0.0003024	R_AEI3
23	9.476e-06	0.0006899	-0.0008578	0.0008502	UEP3
24	-0.3741	0.2835	-0.9156	0.209	UAI4

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

Table 11: Representative items with credible bias indication (Part 3)

ID	n_eff	Rhat	P_posterior	zero_excl	Item
1	1278	1.003	0.8421		HLA2
2	16519	1	0.9061		HLB5
3	16318	1	0.7971		NA4
4	18344	1	0.6727		NB3
5	15735	1	0.5507		AAS1
6	17784	1	0.6584		AU1
7	17115	1	0.6235		PF1

ID	n_eff	Rhat	P_posterior	zero_excl	Item
8	10531	1	0.7935		AL2
9	18833	1.001	0.9961	*	AS1
10	18168	1.001	0.752		APP1
11	16084	1.001	0.8278		UAA1
12	15449	1.001	0.7448		R_AE1
13	20184	1.001	0.5004		UE2
14	20554	1.001	0.5012		UT3
15	17801	1	0.9305		UAL1
16	17448	1	0.6277		AA2
17	15358	1	0.7122		R_AC1
18	17008	1	0.6082		R_AI3
19	17002	1	0.722		AT1
20	17068	1.001	0.8141		SP2
21	17643	1.001	0.8736		IIS2
22	27127	1	0.5087		R_AEI3
23	7547	1.003	0.5012		UEP3
24	15246	1	0.9102		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")
Grand_mean <- c(mean(short_list$mean_Dutch),mean(short_list$sd_Dutch),
                mean(short_list$mean_English),mean(short_list$sd_English),
                mean(abs(short_list$mean_diff)),mean(short_list$sd_diff),
                min(short_list$mean_diff),max(short_list$mean_diff),
                sum(short_list$zero_excl=="*"),round(sum(short_list$zero_excl=="*")
                /length(short_list$ID),digits=4)*100)
GrandMean <- cbind(Variable, Grand_mean)
# 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
pander(GrandMean, caption = "Grand mean of 24 representative items")
```

Table 12: Grand mean of 24 representative items

Variable	Grand_mean
mean_Dutch	0.0125
sd_Dutch	1.59809522503529
mean_English	-0.0319444444444444
sd_English	1.51926465965196
mean_diff	0.167114265020475
sd_diff	0.197067443758612
minimum_diff	-0.374125692747397
maximum_diff	0.704201755249558
n_zero_excl	1
percent_zero_excl	4.17