Behavioral Data Science Week 4 - Factor Analysis

Jonathan Gragg

9/19/2021

Summary

The initial goal was to determine whether or not 17 variables in the employee survey could be averaged into one variable. I first glimpsed the data and turned all of the factors into a numerical data value in order to evaluate. My initial choice was to run the factors through a scree plot and see what that tells me. It led to the conclustion that there needed to be a least 3 factors. That would contradict the initial hypothesis that these variables could become 1 factor so my initial conclusion was that these variables could not be averaged into one variable.

After that I fit a model to both an Orthogonal analysis and a Oblique analysis. In an orthogonal you are testing whether or not the factors should remain independent while in an oblique you are testing whether or not the factors should become 1 factor. In both findings I was able to see that there was some variables that would point to factor 2 and factor 1. This confirmed my conclusion that these variables could not be averaged into 1 score.

Analysis

loading in relevant Libraries

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.1     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
library(stringr)
library(psych)
```

```
## Warning: package 'psych' was built under R version 4.0.5

##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

loading in the data

```
data <- read.csv("survey_results_public.csv")

#selecting the columns that we will be working with
df <- data %>% select(ProblemSolving:ChangeWorld) %>% drop_na()

#converting all the of the column values into a numerical rating scale
df[df=='Strongly disagree'] <- 1
df[df=='Disagree'] <- 2
df[df=='Somewhat agree'] <- 3
df[df=='Agree'] <- 4
df[df=='Strongly agree'] <- 5

columns = 1:17

for (i in columns){
  df[,i] = as.integer(df[,i])
}

glimpse(df)
```

```
## Rows: 28,892
## Columns: 17
## $ ProblemSolving      <int> 5, 5, 5, 5, 5, 5, 4, 5, 5, 5, 5, 4, 4, 5, 5, 5,...
## $ BuildingThings      <int> 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 4, 4, 5, 5,...
## $ LearningNewTech     <int> 4, 5, 5, 5, 4, 5, 4, 5, 5, 4, 4, 5, 5, 4, 5, 5,...
## $ BoringDetails       <int> 2, 3, 2, 1, 2, 2, 1, 1, 2, 5, 4, 1, 3, 3, 3, 2,...
## $ JobSecurity         <int> 5, 4, 3, 4, 3, 3, 4, 4, 5, 3, 4, 4, 1, 4, 4, 4,...
## $ DiversityImportant  <int> 4, 5, 4, 5, 3, 1, 5, 4, 4, 4, 3, 5, 2, 4, 1, 3,...
## $ AnnoyingUI          <int> 4, 4, 4, 5, 5, 4, 5, 4, 3, 4, 4, 5, 5, 4, 5, 5,...
## $ FriendsDevelopers   <int> 2, 3, 4, 3, 3, 2, 1, 2, 4, 2, 4, 5, 3, 2, 2, 2,...
## $ RightWrongWay       <int> 3, 2, 3, 3, 3, 4, 2, 2, 2, 2, 3, 3, 2, 3, 4, 1,...
## $ UnderstandComputers <int> 2, 2, 1, 1, 2, 2, 5, 5, 2, 3, 1, 4, 3, 4, 2, 3,...
## $ SeriousWork         <int> 5, 4, 5, 5, 4, 4, 5, 4, 4, 3, 4, 5, 3, 5, 4, 4,...
## $ InvestTimeTools     <int> 5, 3, 4, 2, 4, 4, 4, 4, 4, 4, 4, 3, 3, 4, 3, 5,...
## $ WorkPayCare         <int> 1, 2, 2, 1, 2, 3, 1, 1, 3, 1, 2, 2, 2, 2, 5, 1,...
## $ KinshipDevelopers   <int> 4, 3, 5, 4, 3, 3, 5, 4, 5, 3, 4, 4, 3, 4, 2, 4,...
## $ ChallengeMyself     <int> 4, 4, 5, 5, 4, 4, 4, 4, 5, 4, 4, 4, 5, 5, 4, 4,...
## $ CompetePeers        <int> 2, 2, 3, 5, 3, 3, 1, 1, 2, 1, 4, 3, 2, 1, 2, 4,...
## $ ChangeWorld         <int> 4, 4, 4, 4, 3, 4, 5, 5, 5, 3, 3, 3, 5, 3, 3, 4,...
```

```
#Is a FA analysis appropriate?

#if the p value is small then FA analysis is appropriate
cortest.bartlett(df)
```

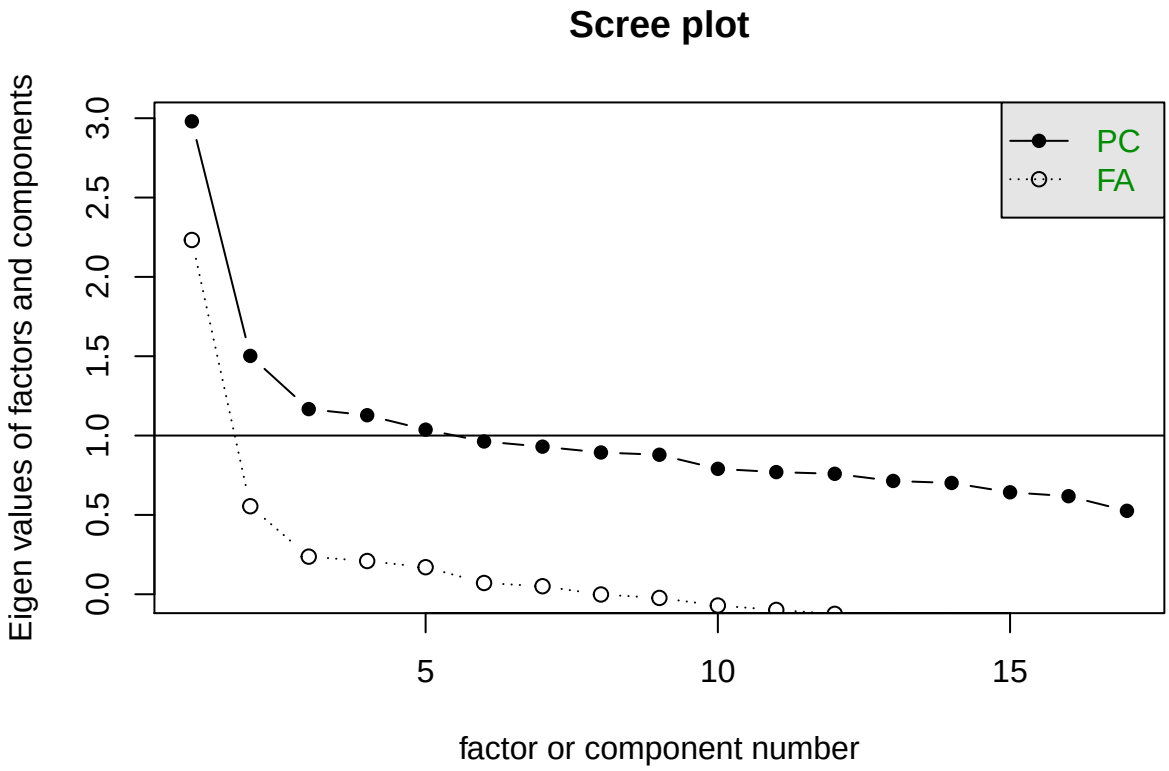
```
## R was not square, finding R from data
```

```
## $chisq
## [1] 45444.6
##
## $p.value
## [1] 0
##
## $df
## [1] 136
```

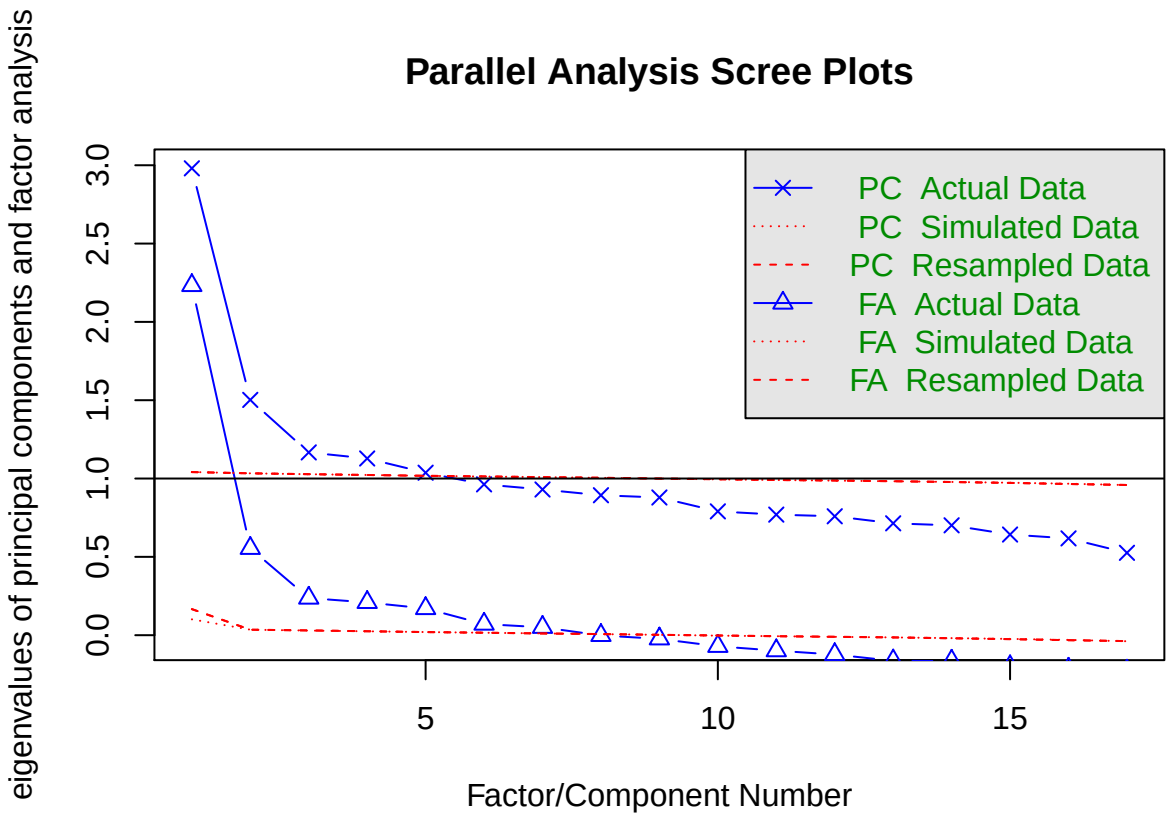
```
#if overall MSA is over a certain threshold then yes
KMO(df)
```

```
## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = df)
## Overall MSA =  0.8
## MSA for each item =
##      ProblemSolving      BuildingThings     LearningNewTech       BoringDetails
##                0.80                0.82                0.85                0.64
##         JobSecurity  DiversityImportant          AnnoyingUI   FriendsDevelopers
##                0.61                0.75                0.80                0.70
##       RightWrongWay UnderstandComputers         SeriousWork     InvestTimeTools
##                0.65                0.61                0.83                0.85
##         WorkPayCare   KinshipDevelopers     ChallengeMyself        CompetePeers
##                0.68                0.85                0.82                0.67
##         ChangeWorld
##                0.82
```

```
scree(df)
```




```
#Number of factors?
fa.parallel(df)
```




```
## Parallel analysis suggests that the number of factors =  7  and the number of components =  5
```

Based on the parallel analysis scree plot I would say that there should be 3 factors and not 1. That would lead me to believe that these metrics cannot be averaged into one.

```
#orthogonal rotate
fa2 <- fa(df, nfactors = 3, rotate = "varimax", covar = FALSE)
print(fa2$loadings, cutoff = 0.001, digits = 3)
```

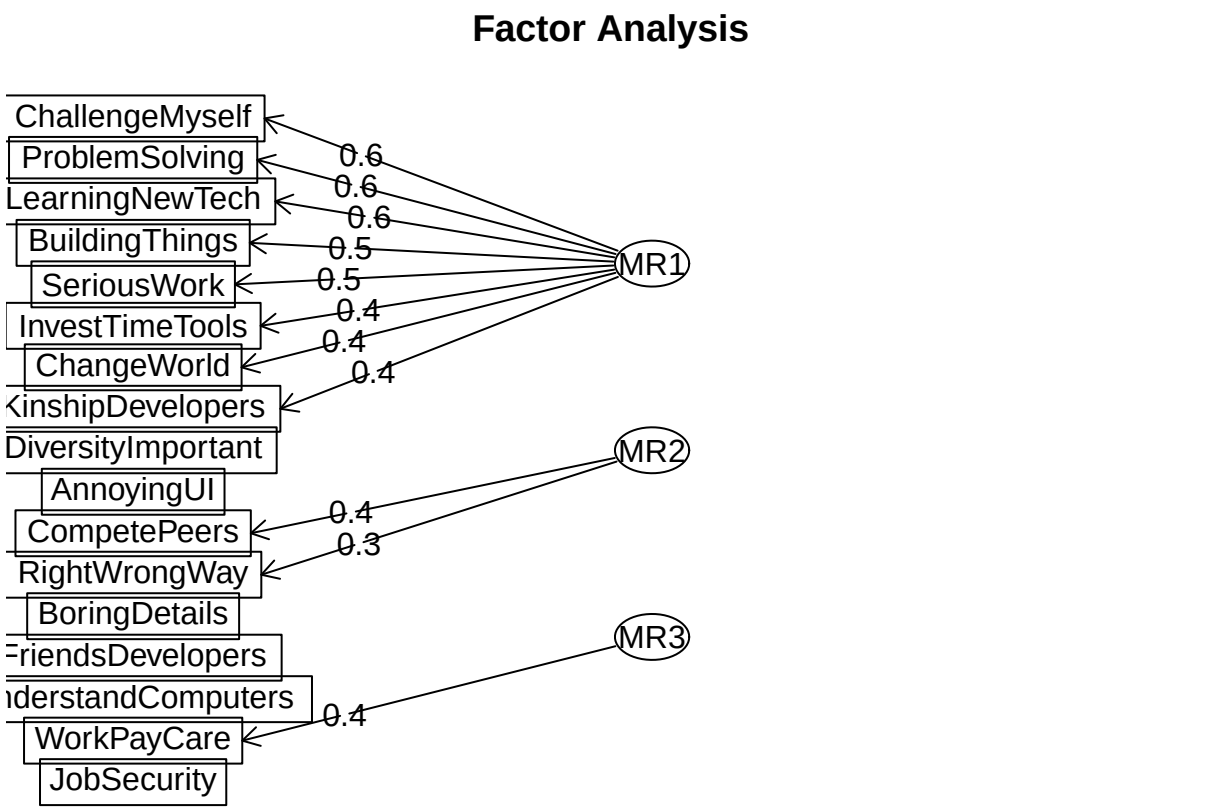
```
##
## Loadings:
##                      MR1    MR2    MR3
## ProblemSolving       0.605 -0.205  0.042
## BuildingThings       0.549 -0.185  0.067
## LearningNewTech      0.560 -0.035  0.021
## BoringDetails       -0.109  0.222  0.046
## JobSecurity          0.135  0.076  0.293
## DiversityImportant   0.243  0.025 -0.140
## AnnoyingUI           0.204  0.047  0.029
## FriendsDevelopers    0.113  0.205  0.031
## RightWrongWay        0.113  0.343  0.139
## UnderstandComputers -0.064  0.160 -0.035
## SeriousWork          0.455  0.050  0.041
## InvestTimeTools      0.432  0.121  0.013
## WorkPayCare         -0.168  0.190  0.394
```

```
## KinshipDevelopers    0.365  0.104 -0.012
## ChallengeMyself      0.620  0.035 -0.097
## CompetePeers         0.164  0.429  0.125
## ChangeWorld          0.392  0.276 -0.286
##
##                  MR1   MR2   MR3
## SS loadings    2.263 0.646 0.400
## Proportion Var 0.133 0.038 0.024
## Cumulative Var 0.133 0.171 0.195
```

```
fa2$communalities
```

```
##      ProblemSolving      BuildingThings     LearningNewTech       BoringDetails
##           0.41017872          0.34033932          0.31571210          0.06331603
##         JobSecurity  DiversityImportant          AnnoyingUI   FriendsDevelopers
##           0.10957666          0.07915641          0.04469198          0.05558629
##       RightWrongWay UnderstandComputers         SeriousWork     InvestTimeTools
##           0.15000278          0.03100849          0.21102298          0.20171476
##         WorkPayCare   KinshipDevelopers     ChallengeMyself        CompetePeers
##           0.21922111          0.14413285          0.39529187          0.22658740
##         ChangeWorld
##           0.31144135
```

```
fa.diagram(fa2)
```



```r
#oblique rotate
fa3 <- fa(df, nfactors = 3, rotate = "promax")
```

```
## Loading required namespace: GPArotation
```

```r
print(fa3$loadings, cutoff = 0.001, digits = 3)
```

```
##
## Loadings:
##                      MR1    MR2    MR3
## ProblemSolving       0.626 -0.245  0.060
## BuildingThings       0.569 -0.223  0.083
## LearningNewTech      0.567 -0.069  0.032
## BoringDetails       -0.125  0.227  0.037
## JobSecurity          0.141  0.050  0.294
## DiversityImportant   0.238  0.020 -0.137
## AnnoyingUI           0.203  0.033  0.031
## FriendsDevelopers    0.100  0.198  0.027
## RightWrongWay        0.094  0.331  0.131
## UnderstandComputers -0.078  0.168 -0.041
## SeriousWork          0.456  0.021  0.048
## InvestTimeTools      0.427  0.096  0.017
## WorkPayCare         -0.169  0.176  0.386
## KinshipDevelopers    0.359  0.084 -0.009
## ChallengeMyself      0.618  0.005 -0.087
## CompetePeers         0.138  0.415  0.114
## ChangeWorld          0.364  0.274 -0.289
##
##                  MR1   MR2   MR3
## SS loadings    2.276 0.642 0.395
## Proportion Var 0.134 0.038 0.023
## Cumulative Var 0.134 0.172 0.195
```

because there are variables that are higher in MR2 than in MR1 we can conclude that there needs to be more than 1 factor.

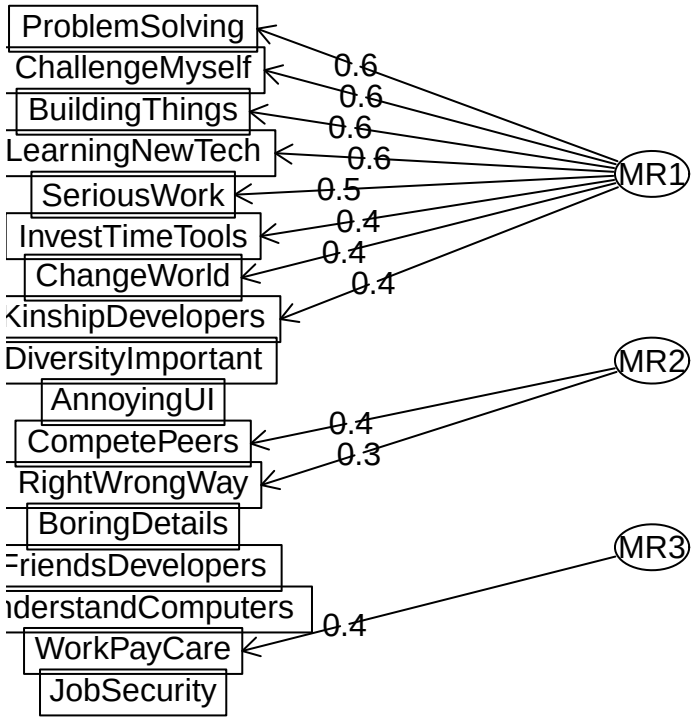
```r
#look at correlation between latent factors
fa3$Phi
```

```
##             MR1        MR2         MR3
## MR1  1.00000000 0.12668629 -0.04350051
## MR2  0.12668629 1.00000000  0.09041632
## MR3 -0.04350051 0.09041632  1.00000000
```

There is very little correlation between the factors which would lead me to believe that my original conclusion was correct that these variables should not be averaged into 1.

```r
fa.diagram(fa3)
```

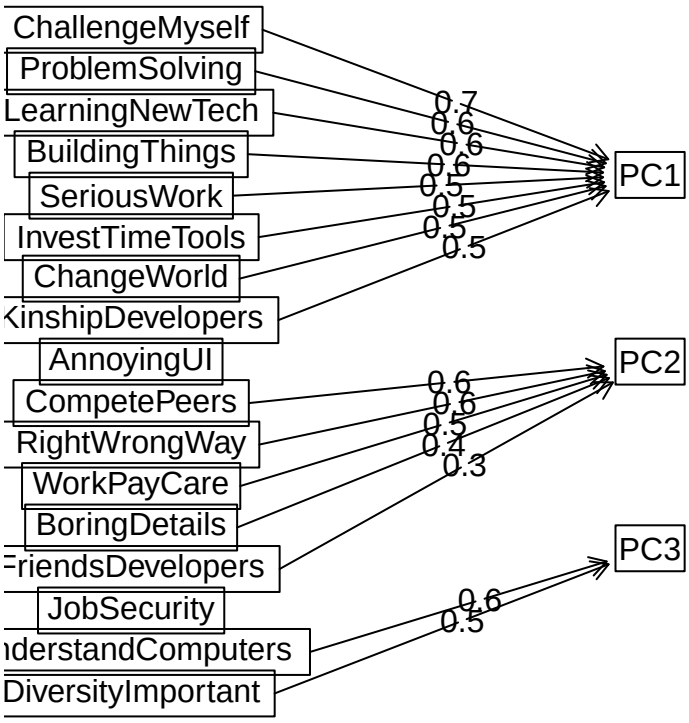
Factor Analysis




```
##PC vs. FA
pc1 <- principal(df, nfactors = 3, rotate = "none", covar = FALSE)
class(pc1)
```

```
## [1] "psych"     "principal"
```

```
fa.diagram(pc1)
```




```
alpha(df)
```

```
## Warning in alpha(df): Some items were negatively correlated with the total scale and probably
## should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option

## Some items ( BoringDetails UnderstandComputers WorkPayCare ) were negatively correlated with the tota
## probably should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option

##
## Reliability analysis
## Call: alpha(x = df)
##
##   raw_alpha std.alpha G6(smc) average_r S/N    ase mean   sd median_r
##       0.56       0.6    0.63     0.082 1.5 0.0038  3.6 0.34    0.077
##
##  lower alpha upper     95% confidence boundaries
## 0.55 0.56 0.57
##
##  Reliability if an item is dropped:
##                     raw_alpha std.alpha G6(smc) average_r S/N alpha se  var.r
## ProblemSolving           0.53      0.57    0.59     0.076 1.3   0.0040 0.0099
## BuildingThings           0.53      0.57    0.59     0.076 1.3   0.0040 0.0106
## LearningNewTech          0.52      0.56    0.58     0.074 1.3   0.0041 0.0107
```

```
## BoringDetails            0.57      0.62    0.64     0.093 1.6   0.0037 0.0120
## JobSecurity              0.55      0.60    0.62     0.085 1.5   0.0039 0.0134
## DiversityImportant       0.55      0.59    0.62     0.083 1.5   0.0039 0.0129
## AnnoyingUI               0.55      0.59    0.62     0.084 1.5   0.0039 0.0134
## FriendsDevelopers        0.55      0.60    0.62     0.086 1.5   0.0038 0.0133
## RightWrongWay            0.54      0.59    0.62     0.084 1.5   0.0039 0.0133
## UnderstandComputers      0.58      0.62    0.64     0.092 1.6   0.0036 0.0125
## SeriousWork              0.53      0.57    0.59     0.077 1.3   0.0040 0.0116
## InvestTimeTools          0.52      0.57    0.59     0.076 1.3   0.0041 0.0120
## WorkPayCare              0.58      0.63    0.64     0.096 1.7   0.0036 0.0110
## KinshipDevelopers        0.53      0.57    0.60     0.078 1.4   0.0040 0.0126
## ChallengeMyself          0.52      0.56    0.58     0.073 1.3   0.0041 0.0097
## CompetePeers             0.53      0.59    0.61     0.081 1.4   0.0040 0.0133
## ChangeWorld              0.52      0.57    0.60     0.077 1.3   0.0041 0.0123
##                       med.r
## ProblemSolving        0.076
## BuildingThings        0.075
## LearningNewTech       0.074
## BoringDetails         0.082
## JobSecurity           0.081
## DiversityImportant    0.076
## AnnoyingUI            0.076
## FriendsDevelopers     0.080
## RightWrongWay         0.078
## UnderstandComputers   0.084
## SeriousWork           0.073
## InvestTimeTools       0.075
## WorkPayCare           0.082
## KinshipDevelopers     0.074
## ChallengeMyself       0.075
## CompetePeers          0.075
## ChangeWorld           0.073
##
##  Item statistics
##                          n raw.r std.r   r.cor r.drop mean   sd
## ProblemSolving       28892  0.39 0.483  0.4677  0.292  4.5 0.65
## BuildingThings       28892  0.39 0.472  0.4441  0.286  4.5 0.66
## LearningNewTech      28892  0.45 0.517  0.5005  0.342  4.4 0.73
## BoringDetails        28892  0.20 0.157  0.0074  0.024  2.7 1.00
## JobSecurity          28892  0.32 0.314  0.2082  0.159  3.8 0.95
## DiversityImportant   28892  0.35 0.342  0.2479  0.178  3.8 1.04
## AnnoyingUI           28892  0.33 0.336  0.2323  0.174  4.1 0.91
## FriendsDevelopers    28892  0.34 0.292  0.1756  0.145  2.9 1.15
## RightWrongWay        28892  0.38 0.329  0.2297  0.192  3.1 1.13
## UnderstandComputers  28892  0.24 0.172  0.0211  0.030  3.0 1.22
## SeriousWork          28892  0.42 0.465  0.4226  0.298  4.1 0.79
## InvestTimeTools      28892  0.44 0.471  0.4258  0.304  3.6 0.90
## WorkPayCare          28892  0.13 0.099 -0.0593 -0.037  2.2 0.97
## KinshipDevelopers    28892  0.43 0.445  0.3850  0.286  3.5 0.89
## ChallengeMyself      28892  0.48 0.544  0.5477  0.371  4.2 0.74
## CompetePeers         28892  0.42 0.376  0.2946  0.242  2.8 1.11
## ChangeWorld          28892  0.47 0.454  0.4028  0.298  3.6 1.09
##
## Non missing response frequency for each item
```

```
##                         1    2    3    4    5 miss
## ProblemSolving       0.00 0.01 0.06 0.35 0.58    0
## BuildingThings       0.00 0.01 0.06 0.36 0.57    0
## LearningNewTech      0.00 0.01 0.10 0.40 0.49    0
## BoringDetails        0.09 0.39 0.31 0.16 0.05    0
## JobSecurity          0.02 0.08 0.25 0.42 0.23    0
## DiversityImportant   0.04 0.08 0.24 0.39 0.26    0
## AnnoyingUI           0.01 0.05 0.16 0.38 0.40    0
## FriendsDevelopers    0.11 0.31 0.26 0.24 0.09    0
## RightWrongWay        0.07 0.25 0.32 0.23 0.13    0
## UnderstandComputers  0.11 0.26 0.26 0.23 0.14    0
## SeriousWork          0.00 0.03 0.16 0.48 0.33    0
## InvestTimeTools      0.01 0.10 0.32 0.41 0.16    0
## WorkPayCare          0.25 0.45 0.20 0.07 0.03    0
## KinshipDevelopers    0.02 0.10 0.33 0.43 0.12    0
## ChallengeMyself      0.00 0.02 0.12 0.47 0.38    0
## CompetePeers         0.11 0.32 0.29 0.21 0.07    0
## ChangeWorld          0.03 0.14 0.29 0.30 0.23    0
```