waveley_attempt

Waveley Qiu (wq2162)

2022-04-30

Athletic Identity

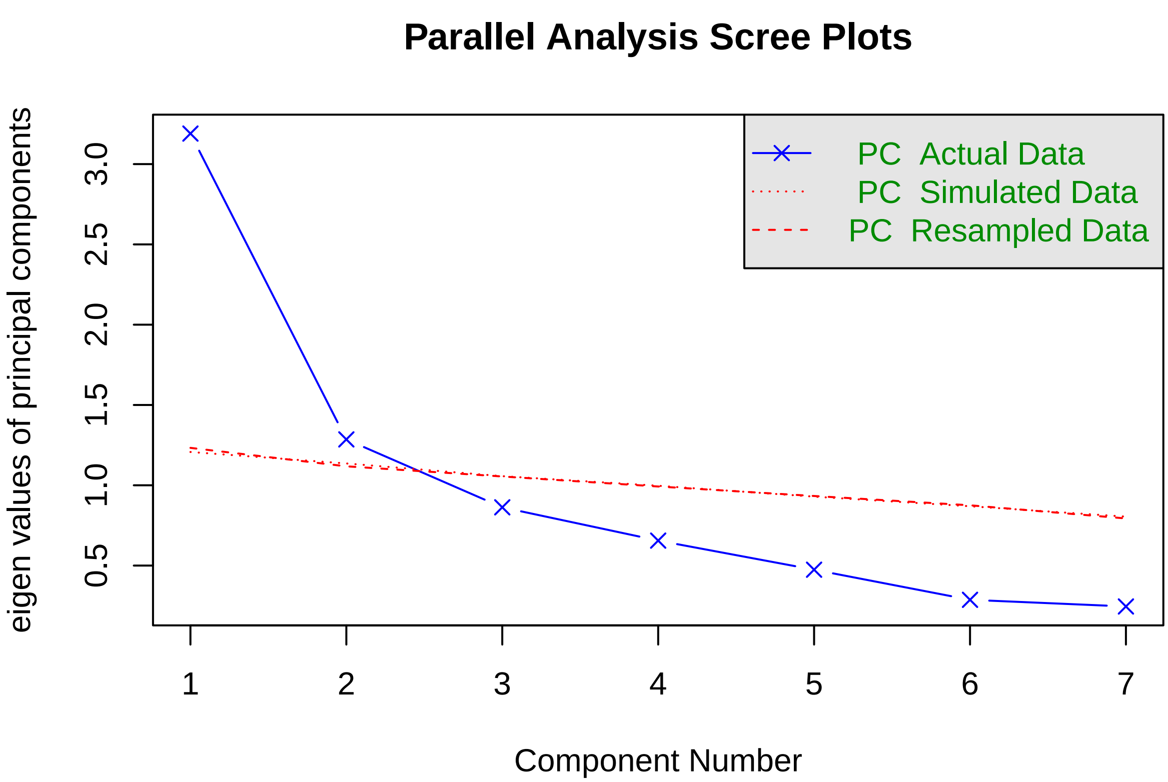
First, let's select the variables we are interested in.

```
athletic_identity <- c("cnsdr_ath", "sprt_goals", "frnds_ath", "sprt_impt", "think_sprt", "bad_sprt", "c

athletic_identity_numeric <- athletes[,athletic_identity] %>% map_df(., as.numeric)

athletic_identity_matrix <- athletic_identity_numeric %>% as.matrix()
```

Polychoric Correlations

Now, let us determine the number of factors that might underlie these variables.

```
athlete_parallel <- fa.parallel(athletic_identity_matrix, cor = "poly", fa = "pc")
```



```
## Parallel analysis suggests that the number of factors =  NA  and the number of components =  2
```

```
athlete_parallel$pc.values
```

```
## [1] 3.1901693 1.2855443 0.8626912 0.6554703 0.4741985 0.2866577 0.2452687
```

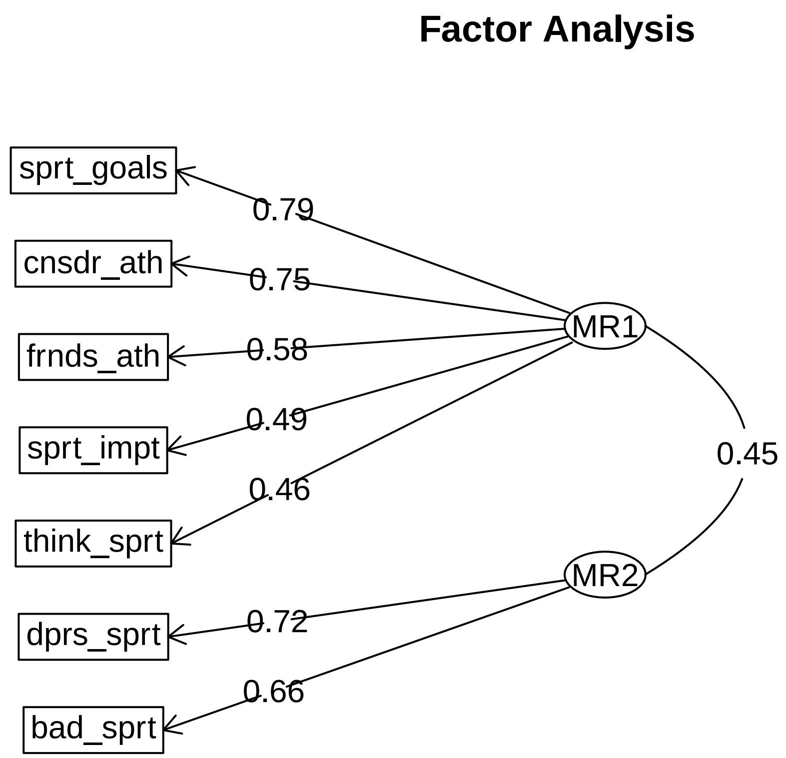
PCA indicates that two factors underlie these variables.

EFA

We now want to examine which variables might load on which factors. To do so, we will perform EFA on a 2-factor model, and also on 1- and 3- factor models.

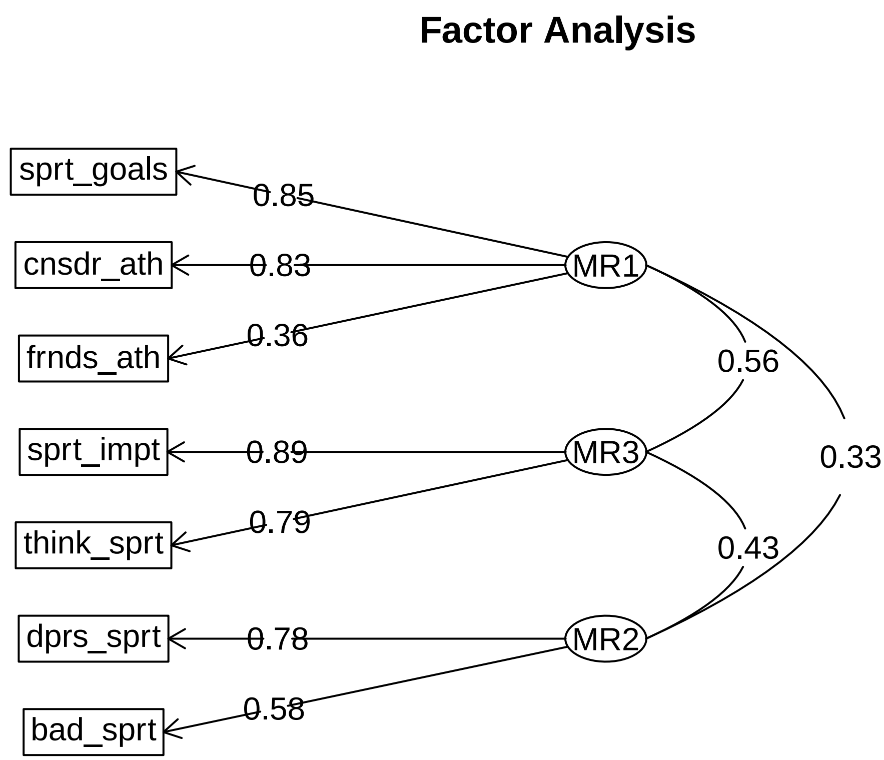
First, the 2-factor model:

```
athletic_efa2 <- fa(r = athletic_identity_matrix, nfactors = 2, cor = "poly")
fa.diagram(athletic_efa2, digits = 2, simple = TRUE)
```



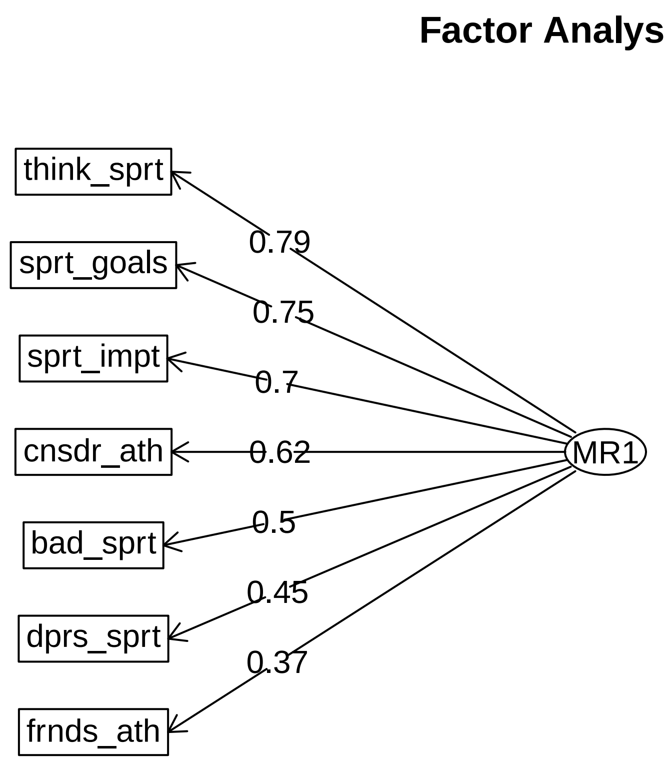
Now, the 3-factor model:

```
athletic_efa3 <- fa(r = athletic_identity_matrix, nfactors = 3, cor = "poly")
fa.diagram(athletic_efa3, digits = 2, simple = TRUE)
```



Now, the 1-factor model:

```
athletic_efa1 <- fa(r = athletic_identity_matrix, nfactors = 1, cor = "poly")
fa.diagram(athletic_efa1, digits = 2, simple = TRUE)
```



Reliability

LV 1: External Identity

```
external_identity <- c("cnsdr_ath", "sprt_goals", "frnds_ath")

external_identity_numeric <- athletes[,external_identity] %>% map_df(., as.numeric)

external_identity_matrix <- external_identity_numeric %>% as.matrix()
psych::alpha(external_identity_matrix)
```

```
##
## Reliability analysis
## Call: psych::alpha(x = external_identity_matrix)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.65      0.68    0.62      0.42 2.1 0.032  5.7 0.89     0.33
##
##  lower alpha upper     95% confidence boundaries
## 0.59 0.65 0.72
##
##  Reliability if an item is dropped:
##            raw_alpha std.alpha G6(smc) average_r  S/N alpha se var.r med.r
## cnsdr_ath       0.47      0.49    0.33      0.33 0.97    0.052    NA  0.33
## sprt_goals      0.46      0.47    0.31      0.31 0.89    0.055    NA  0.31
## frnds_ath       0.75      0.76    0.61      0.61 3.15    0.025    NA  0.61
##
```

```
##  Item statistics
##              n raw.r std.r r.cor r.drop mean   sd
## cnsdr_ath  356  0.79  0.82  0.71   0.53  5.9 1.11
## sprt_goals 356  0.78  0.83  0.72   0.56  6.1 0.96
## frnds_ath  356  0.76  0.70  0.41   0.35  5.2 1.38
##
## Non missing response frequency for each item
##               1    2    3    4    5    6    7 miss
## cnsdr_ath  0.01 0.01 0.03 0.05 0.24 0.33 0.35 0.02
## sprt_goals 0.00 0.00 0.02 0.03 0.20 0.35 0.40 0.02
## frnds_ath  0.01 0.04 0.06 0.10 0.31 0.29 0.17 0.02
```

Chronbach's alpha suggests we should drop `frnds_ath`.

```
external_identity <- c("cnsdr_ath", "sprt_goals")

external_identity_numeric <- athletes[,external_identity] %>% map_df(., as.numeric)

external_identity_matrix <- external_identity_numeric %>% as.matrix()
psych::alpha(external_identity_matrix)
```

```
##
## Reliability analysis
## Call: psych::alpha(x = external_identity_matrix)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.75      0.76    0.61      0.61 3.2 0.025    6 0.93     0.61
##
##  lower alpha upper     95% confidence boundaries
## 0.7 0.75 0.8
##
##  Reliability if an item is dropped:
##            raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
## cnsdr_ath       0.71      0.61    0.37      0.61 1.6       NA     0  0.61
## sprt_goals      0.53      0.61    0.37      0.61 1.6       NA     0  0.61
##
##  Item statistics
##              n raw.r std.r r.cor r.drop mean   sd
## cnsdr_ath  356  0.91   0.9   0.7   0.61  5.9 1.11
## sprt_goals 356  0.88   0.9   0.7   0.61  6.1 0.96
##
## Non missing response frequency for each item
##               1    2    3    4    5    6    7 miss
## cnsdr_ath  0.01 0.01 0.03 0.05 0.24 0.33 0.35 0.02
## sprt_goals 0.00 0.00 0.02 0.03 0.20 0.35 0.40 0.02
```

LV 2: Internal Value

```
internal_value <- c("sprt_impt", "think_sprt")

internal_value_numeric <- athletes[,internal_value] %>% map_df(., as.numeric)

internal_value_matrix <- internal_value_numeric %>% as.matrix()
psych::alpha(internal_value_matrix)
```

```
##
```

```
## Reliability analysis
## Call: psych::alpha(x = internal_value_matrix)
##
##   raw_alpha std.alpha G6(smc) average_r S/N  ase mean  sd median_r
##       0.81      0.81    0.67      0.67 4.1 0.02    5 1.4     0.67
##
##  lower alpha upper     95% confidence boundaries
## 0.77 0.81 0.85
##
##  Reliability if an item is dropped:
##             raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
## sprt_impt       0.67      0.67    0.45      0.67 2.1       NA     0  0.67
## think_sprt      0.68      0.67    0.45      0.67 2.1       NA     0  0.67
##
##  Item statistics
##               n raw.r std.r r.cor r.drop mean  sd
## sprt_impt  356  0.91  0.91  0.75   0.67  5.1 1.5
## think_sprt 356  0.92  0.91  0.75   0.67  4.9 1.5
##
## Non missing response frequency for each item
##                1    2    3    4    5    6    7 miss
## sprt_impt  0.03 0.04 0.10 0.10 0.28 0.27 0.18 0.02
## think_sprt 0.02 0.05 0.14 0.12 0.29 0.22 0.16 0.02
```

LV 3: Negative Events

```
negative_events <- c("dprs_sprt", "bad_sprt")

negative_events_numeric <- athletes[,negative_events] %>% map_df(., as.numeric)

negative_events_matrix <- negative_events_numeric %>% as.matrix()
psych::alpha(negative_events_matrix)
```

```
##
## Reliability analysis
## Call: psych::alpha(x = negative_events_matrix)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean  sd median_r
##       0.63      0.63    0.46      0.46 1.7 0.039  5.6 1.2     0.46
##
##  lower alpha upper     95% confidence boundaries
## 0.55 0.63 0.7
##
##  Reliability if an item is dropped:
##            raw_alpha std.alpha G6(smc) average_r  S/N alpha se var.r med.r
## dprs_sprt      0.40      0.46    0.21      0.46 0.85       NA     0  0.46
## bad_sprt       0.53      0.46    0.21      0.46 0.85       NA     0  0.46
##
##  Item statistics
##              n raw.r std.r r.cor r.drop mean  sd
## dprs_sprt 356  0.83  0.85  0.58   0.46  5.7 1.3
## bad_sprt  356  0.88  0.85  0.58   0.46  5.5 1.4
##
## Non missing response frequency for each item
##              1    2    3    4    5    6    7 miss
```

```
## dprs_sprt 0.01 0.02 0.02 0.06 0.30 0.29 0.29 0.02
## bad_sprt  0.02 0.04 0.04 0.08 0.24 0.29 0.28 0.02
```

Our final model is as follows:

external_identity = sprt_goals + cnsdr_ath internal_value = sprt_impt + think_sprt negative_events = dprs_sprt + bad_sprt

CFA

```r
athlete_model <-
'external_identity =~ sprt_goals + cnsdr_ath
 internal_value =~ sprt_impt + think_sprt
 negative_events =~ dprs_sprt + bad_sprt'

athlete_2CFA = cfa(athlete_model, data = athletic_identity_matrix,
                   ordered = names(athletic_identity_matrix),
                   std.lv = TRUE)
summary(athlete_2CFA, fit.measures = TRUE)
```

```
## lavaan 0.6-10 ended normally after 27 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        15
##
##                                                   Used       Total
##   Number of observations                           356         363
##
## Model Test User Model:
##
##   Test statistic                                 5.234
##   Degrees of freedom                                 6
##   P-value (Chi-square)                           0.514
##
## Model Test Baseline Model:
##
##   Test statistic                               617.425
##   Degrees of freedom                                15
##   P-value                                        0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    1.000
##   Tucker-Lewis Index (TLI)                       1.003
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -3255.389
##   Loglikelihood unrestricted model (H1)      -3252.772
##
##   Akaike (AIC)                                6540.778
##   Bayesian (BIC)                              6598.902
##   Sample-size adjusted Bayesian (BIC)         6551.316
##
```

```
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.000
##   90 Percent confidence interval - lower         0.000
##   90 Percent confidence interval - upper         0.064
##   P-value RMSEA <= 0.05                          0.872
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.017
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Latent Variables:
##                        Estimate  Std.Err  z-value  P(>|z|)
##   external_identity =~
##     sprt_goals            0.871    0.062   14.013    0.000
##     cnsdr_ath             0.751    0.066   11.326    0.000
##   internal_value =~
##     sprt_impt             1.076    0.083   12.962    0.000
##     think_sprt            1.443    0.086   16.700    0.000
##   negative_events =~
##     dprs_sprt             0.805    0.084    9.537    0.000
##     bad_sprt              1.030    0.101   10.171    0.000
##
## Covariances:
##                        Estimate  Std.Err  z-value  P(>|z|)
##   external_identity ~~
##     internal_value        0.511    0.054    9.498    0.000
##     negative_evnts        0.397    0.067    5.912    0.000
##   internal_value ~~
##     negative_evnts        0.513    0.061    8.432    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .sprt_goals        0.169    0.085    1.984    0.047
##    .cnsdr_ath         0.668    0.080    8.347    0.000
##    .sprt_impt         1.114    0.130    8.585    0.000
##    .think_sprt        0.251    0.180    1.394    0.163
##    .dprs_sprt         0.923    0.119    7.751    0.000
##    .bad_sprt          1.009    0.176    5.743    0.000
##     external_dntty    1.000
##     internal_value    1.000
##     negative_evnts    1.000
```