waveley_attempt

Waveley Qiu (wq2162)

2022-04-30

Was there a difference in emotional well being between an athlete and non-athlete during the COVID-19 lockdown?

Latent Variable Construction

Latent Variable 1: Athletic Identity

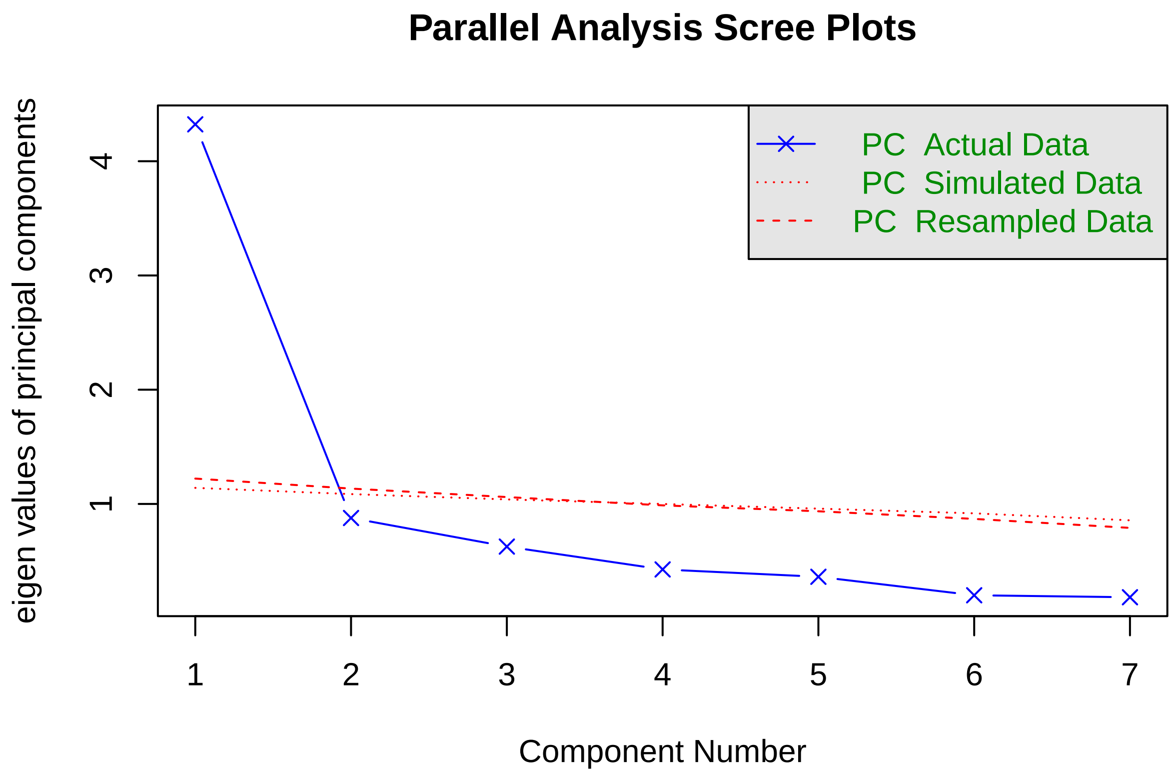
First, let's select the variables we are interested in.

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

athletic_identity_numeric <- dataset[,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 =  1
```

```
athlete_parallel$pc.values
```

```
## [1] 4.3226192 0.8770825 0.6268442 0.4268273 0.3625864 0.2006307 0.1834097
```

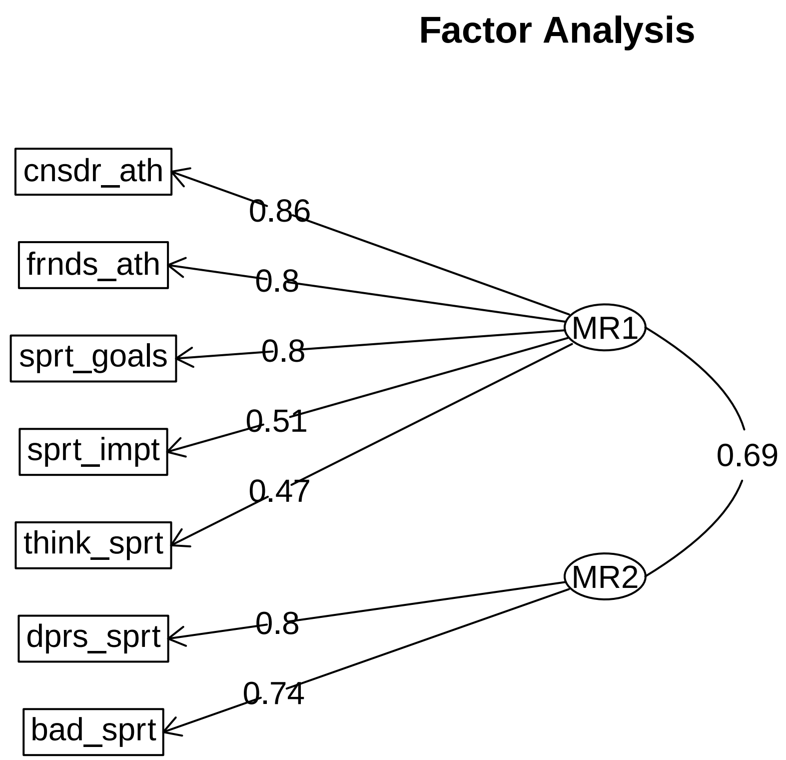
PCA indicates that two factors underlie these variables.

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.

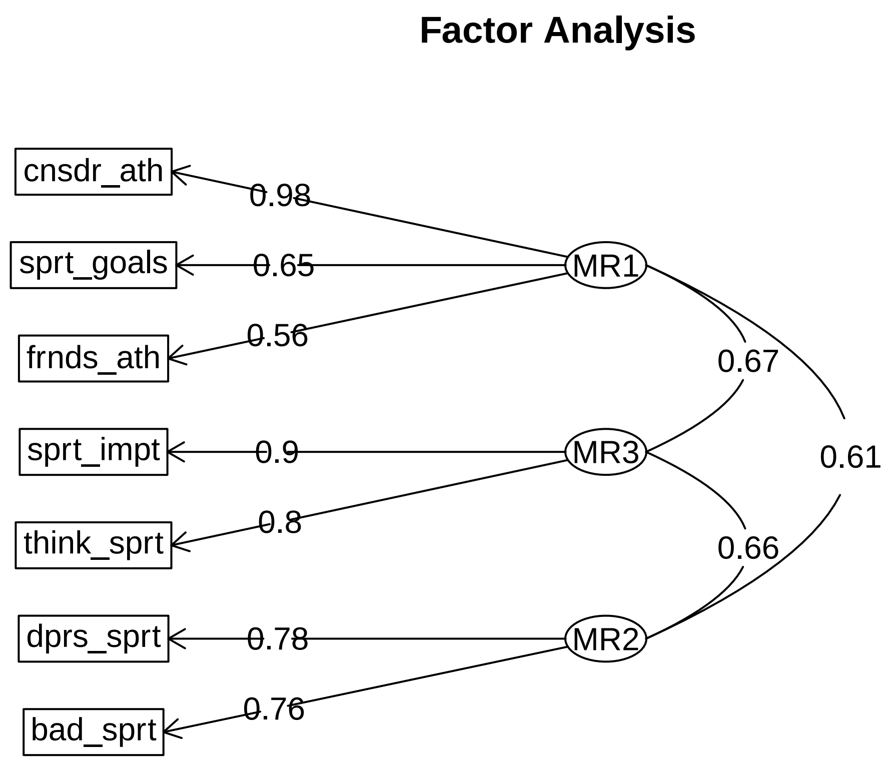
First, the 2-factor model:

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



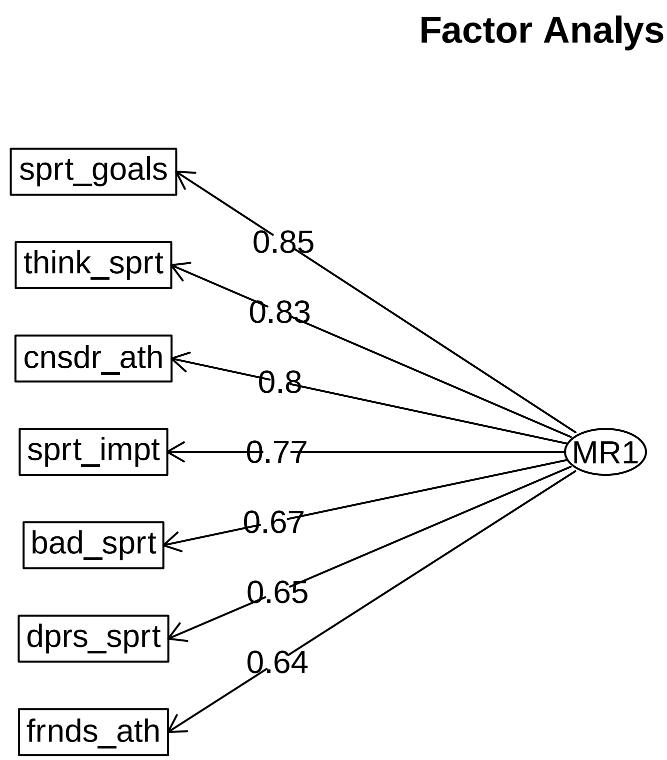
Now, the 3-factor model:

```
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)
```



The 3-factor model seems to fit the data the best, as it has the smallest BIC. We will proceed by using the 3-factor model for athletic_identity.

Reliability

LV 1: External Identity

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

external_identity_numeric <- dataset[,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
##       0.84      0.85     0.8      0.65 5.5 0.0098  5.1 1.5     0.63
##
##  lower alpha upper     95% confidence boundaries
## 0.82 0.84 0.86
##
##  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      0.72    0.56      0.56 2.5    0.021    NA  0.56
```

```
## sprt_goals      0.78      0.78    0.63      0.63 3.5    0.016    NA  0.63
## frnds_ath       0.85      0.86    0.75      0.75 5.9    0.011    NA  0.75
##
##  Item statistics
##              n raw.r std.r r.cor r.drop mean  sd
## cnsdr_ath  518  0.91  0.91  0.86   0.78  5.1 1.8
## sprt_goals 518  0.87  0.88  0.80   0.73  5.5 1.5
## frnds_ath  518  0.84  0.84  0.69   0.64  4.6 1.7
##
## Non missing response frequency for each item
##               1    2    3    4    5    6    7 miss
## cnsdr_ath  0.09 0.04 0.03 0.08 0.25 0.27 0.24 0.31
## sprt_goals 0.04 0.03 0.03 0.05 0.26 0.30 0.29 0.31
## frnds_ath  0.07 0.09 0.08 0.14 0.28 0.23 0.12 0.31
```

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()
```

```r
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 for athlete_identity is as follows:

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

CFA

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

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

```
## lavaan 0.6-10 ended normally after 23 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        17
##
##                                                   Used       Total
##   Number of observations                           518         753
##
## Model Test User Model:
##
```

```
##   Test statistic                                42.386
##   Degrees of freedom                                11
##   P-value (Chi-square)                           0.000
##
## Model Test Baseline Model:
##
##   Test statistic                              2011.633
##   Degrees of freedom                                21
##   P-value                                        0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    0.984
##   Tucker-Lewis Index (TLI)                       0.970
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -6060.200
##   Loglikelihood unrestricted model (H1)      -6039.008
##
##   Akaike (AIC)                               12154.401
##   Bayesian (BIC)                             12226.650
##   Sample-size adjusted Bayesian (BIC)        12172.689
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.074
##   90 Percent confidence interval - lower         0.051
##   90 Percent confidence interval - upper         0.098
##   P-value RMSEA <= 0.05                          0.041
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.025
##
## 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            1.295    0.055   23.488    0.000
##     cnsdr_ath             1.557    0.066   23.609    0.000
##     frnds_ath             1.168    0.068   17.084    0.000
##   internal_value =~
##     sprt_impt             1.433    0.065   22.061    0.000
##     think_sprt            1.581    0.066   24.079    0.000
##   negative_events =~
##     dprs_sprt             1.315    0.069   19.168    0.000
##     bad_sprt              1.378    0.073   18.999    0.000
##
```

```
## Covariances:
##                         Estimate  Std.Err  z-value  P(>|z|)
##   external_identity ~~
##     internal_value         0.751    0.027   28.071    0.000
##     negative_evnts         0.729    0.032   22.965    0.000
##   internal_value ~~
##     negative_evnts         0.696    0.034   20.512    0.000
##
## Variances:
##                     Estimate  Std.Err  z-value  P(>|z|)
##    .sprt_goals         0.569    0.058    9.742    0.000
##    .cnsdr_ath          0.800    0.083    9.584    0.000
##    .frnds_ath          1.512    0.106   14.278    0.000
##    .sprt_impt          0.837    0.088    9.482    0.000
##    .think_sprt         0.592    0.094    6.277    0.000
##    .dprs_sprt          0.989    0.108    9.187    0.000
##    .bad_sprt           1.134    0.120    9.454    0.000
##     external_dntty     1.000
##     internal_value     1.000
##     negative_evnts     1.000
```

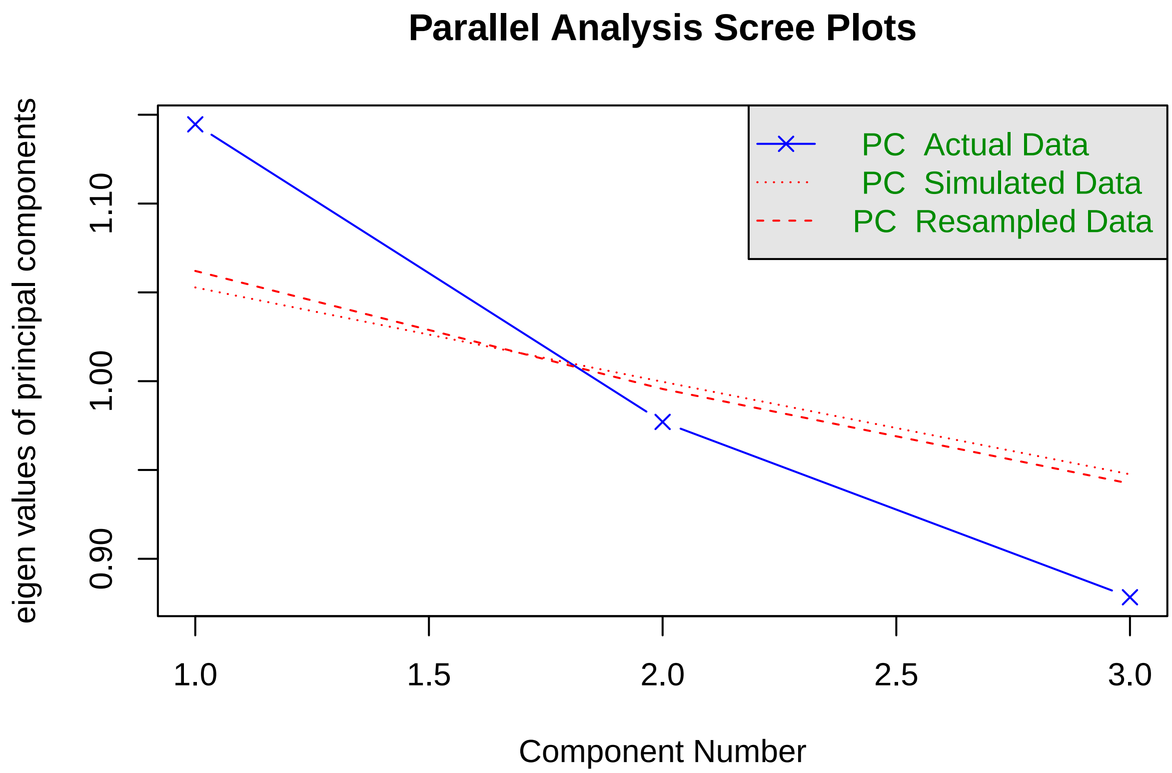
Latent Variable 2: Healthy Lifestyle

```
healthy_lifestyle <- c("hr_sleep", "smoking", "fruit_veg")

healthy_life_numeric <- dataset[,healthy_lifestyle] %>% map_df(., as.numeric)

healthy_life_matrix <- healthy_life_numeric %>% as.matrix()
```

Polychoric Correlations

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

```
health_parallel <- fa.parallel(healthy_life_matrix, fa = "pc")
```




```
## Parallel analysis suggests that the number of factors =  NA  and the number of components =  1
```

```
health_parallel$pc.values
```

```
## [1] 1.1445939 0.9770581 0.8783480
```

1 latent factor appears to underlie these variables. *** need to check if this is the correct way to assess formative LV's ***

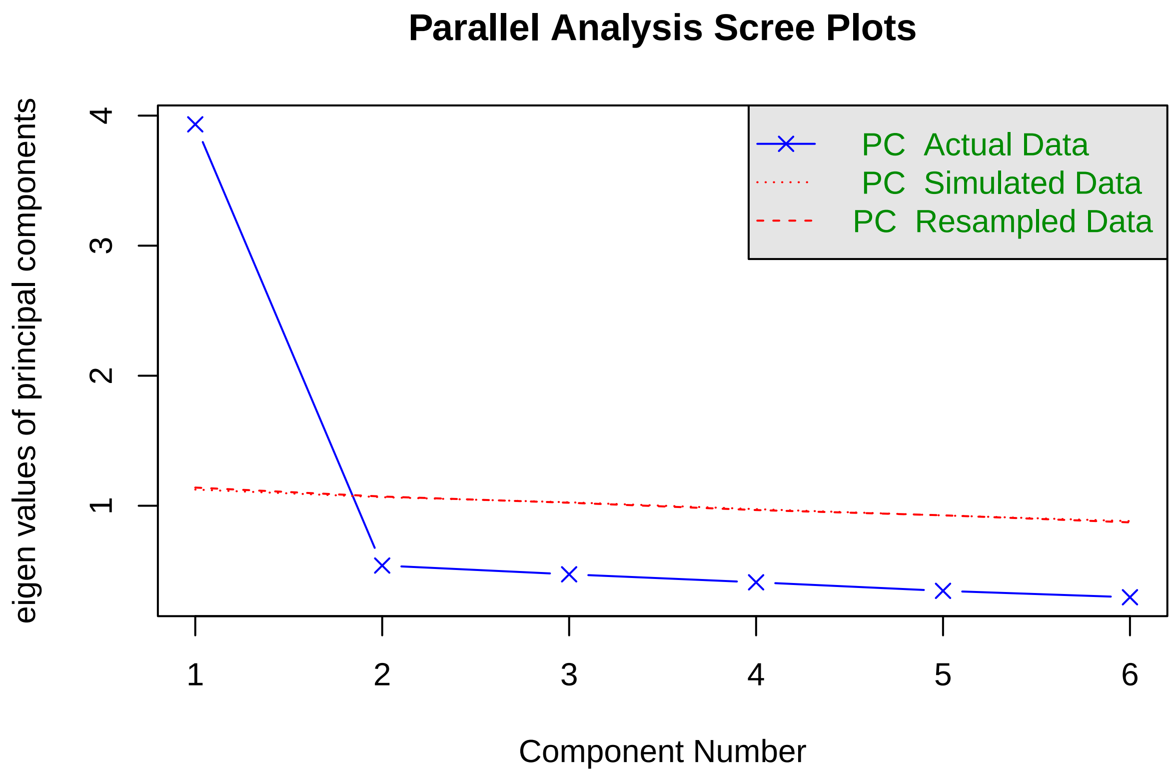
Latent Variable 3: Resilience

Finally, we will look at resilience.

```
resilience <- c("bounce", "strs_evnt", "strs_rcvr", "snap_back", "difficult", "setbacks")

resilience_numeric <- dataset[,resilience] %>% map_df(., as.numeric)

resilience_matrix <- resilience_numeric %>% as.matrix()
```

Polychoric Correlations

```
resilience_parallel <- fa.parallel(resilience_matrix, fa = "pc")
```




```
## Parallel analysis suggests that the number of factors =  NA  and the number of components =  1
```

```
resilience_parallel$pc.values
```

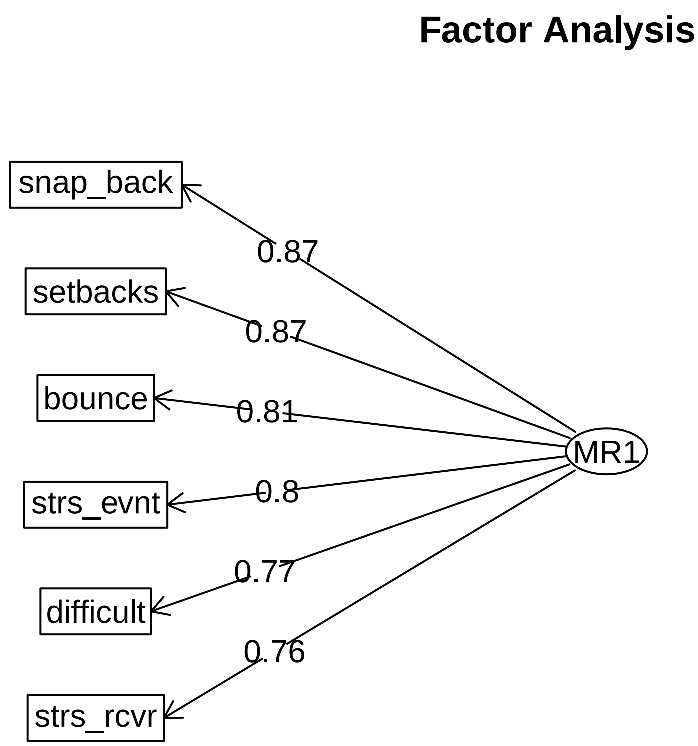
```
## [1] 3.9328582 0.5404267 0.4728952 0.4114713 0.3457324 0.2966162
```

It appears that a 1-factor model will sufficiently explain the variability across these variables.

EFA

First, we can do a 1-factor EFA model:

```
resilience_efa1 <- fa(r = resilience_matrix, nfactors = 1, cor = "poly")
fa.diagram(resilience_efa1, digits = 2, simple = TRUE)
```

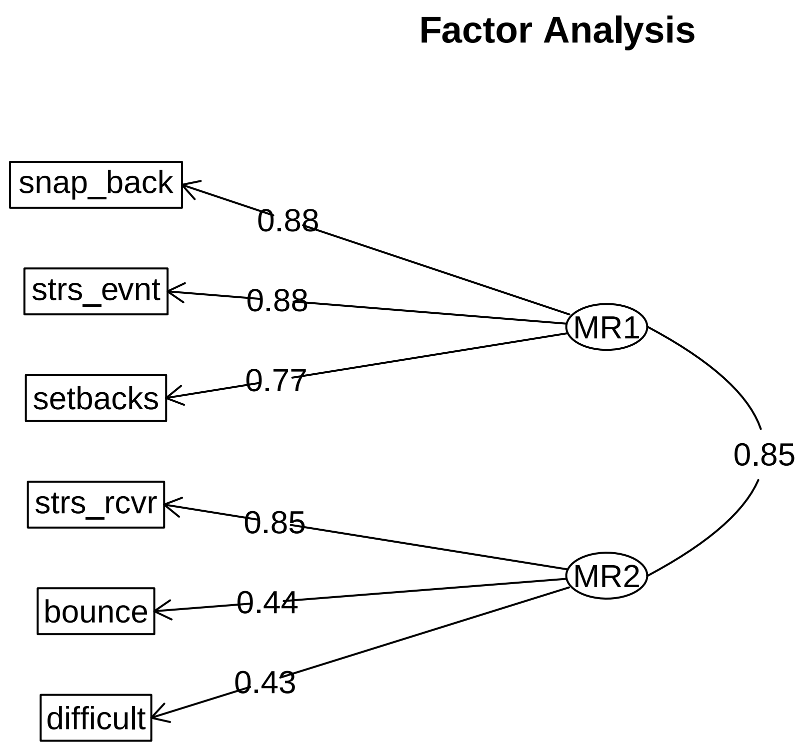



```
summary(resilience_efa1)
```

```
##
## Factor analysis with Call: fa(r = resilience_matrix, nfactors = 1, cor = "poly")
##
## Test of the hypothesis that 1 factor is sufficient.
## The degrees of freedom for the model is 9  and the objective function was  0.08
## The number of observations was  753  with Chi Square =  59.72  with prob <  1.5e-09
##
## The root mean square of the residuals (RMSA) is  0.03
## The df corrected root mean square of the residuals is  0.03
##
## Tucker Lewis Index of factoring reliability =  0.972
## RMSEA index =  0.087  and the 10 % confidence intervals are  0.066 0.108
## BIC =  0.1
```

Now, we can try a 2-factor EFA model:

```
resilience_efa2 <- fa(r = resilience_matrix, nfactors = 2, cor = "poly")
fa.diagram(resilience_efa2, digits = 2, simple = TRUE)
```




```
summary(resilience_efa2)
```

```
##
## Factor analysis with Call: fa(r = resilience_matrix, nfactors = 2, cor = "poly")
##
## Test of the hypothesis that 2 factors are sufficient.
## The degrees of freedom for the model is 4  and the objective function was  0.01
## The number of observations was  753  with Chi Square =  9.94  with prob <  0.041
##
## The root mean square of the residuals (RMSA) is  0.01
## The df corrected root mean square of the residuals is  0.02
##
## Tucker Lewis Index of factoring reliability =  0.993
## RMSEA index =  0.044  and the 10 % confidence intervals are  0.008 0.08
## BIC =  -16.56
##  With factor correlations of
##      MR1  MR2
## MR1 1.00 0.85
## MR2 0.85 1.00
```

Since a 1-factor model is sufficient, we will just proceed with the 1-factor model.

Reliability

```
psych::alpha(resilience_matrix)
```

```
##
```

```
## Reliability analysis
## Call: psych::alpha(x = resilience_matrix)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.89      0.89    0.88      0.59 8.5 0.006  3.5 0.82     0.57
##
##  lower alpha upper     95% confidence boundaries
## 0.88 0.89 0.91
##
##  Reliability if an item is dropped:
##           raw_alpha std.alpha G6(smc) average_r S/N alpha se  var.r med.r
## bounce         0.88      0.88    0.85      0.59 7.1   0.0072 0.0033  0.57
## strs_evnt      0.88      0.88    0.86      0.59 7.2   0.0070 0.0023  0.57
## strs_rcvr      0.88      0.88    0.86      0.60 7.5   0.0068 0.0032  0.60
## snap_back      0.87      0.87    0.84      0.57 6.6   0.0076 0.0017  0.56
## difficult      0.88      0.88    0.86      0.60 7.6   0.0068 0.0031  0.60
## setbacks       0.87      0.87    0.84      0.57 6.6   0.0076 0.0019  0.56
##
##  Item statistics
##             n raw.r std.r r.cor r.drop mean   sd
## bounce    661  0.80  0.81  0.76   0.72  3.9 0.94
## strs_evnt 661  0.80  0.80  0.75   0.70  3.3 1.06
## strs_rcvr 661  0.78  0.78  0.71   0.67  3.5 1.03
## snap_back 661  0.85  0.84  0.81   0.77  3.4 1.03
## difficult 661  0.77  0.77  0.71   0.67  3.3 0.99
## setbacks  661  0.85  0.85  0.82   0.77  3.5 1.01
##
## Non missing response frequency for each item
##              1    2    3    4    5 miss
## bounce    0.02 0.09 0.16 0.48 0.25 0.12
## strs_evnt 0.05 0.20 0.27 0.37 0.11 0.12
## strs_rcvr 0.03 0.18 0.23 0.43 0.14 0.12
## snap_back 0.03 0.20 0.20 0.45 0.12 0.12
## difficult 0.02 0.21 0.28 0.39 0.10 0.12
## setbacks  0.03 0.15 0.22 0.46 0.13 0.12
```

Chronbach's alpha is 0.89 (0.88, 0.91). No items can be dropped to improve this measure, so we will keep all of them in this latent variable.

Final CFA

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

 athlete_identity =~ external_identity + internal_value + negative_events

 healthy_lifestyle =~ hr_sleep + smoking + fruit_veg

 resilience =~ bounce + strs_evnt + strs_rcvr + snap_back + difficult + setbacks
'
fin_df <- dataset %>% map_df(., as.numeric)
```

```r
final_cfa = cfa(final_model,
                data = fin_df,
                std.lv = TRUE)
summary(final_cfa, fit.measures = TRUE)
```

```
## lavaan 0.6-10 ended normally after 50 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        38
##
##                                                   Used       Total
##   Number of observations                           460         753
##
## Model Test User Model:
##
##   Test statistic                               200.588
##   Degrees of freedom                                98
##   P-value (Chi-square)                           0.000
##
## Model Test Baseline Model:
##
##   Test statistic                              3434.549
##   Degrees of freedom                               120
##   P-value                                        0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    0.969
##   Tucker-Lewis Index (TLI)                       0.962
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)             -10292.110
##   Loglikelihood unrestricted model (H1)     -10191.815
##
##   Akaike (AIC)                               20660.219
##   Bayesian (BIC)                             20817.206
##   Sample-size adjusted Bayesian (BIC)        20696.604
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.048
##   90 Percent confidence interval - lower         0.038
##   90 Percent confidence interval - upper         0.057
##   P-value RMSEA <= 0.05                          0.643
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.051
##
## 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            0.481    0.086    5.598    0.000
##     cnsdr_ath             0.580    0.104    5.596    0.000
##     frnds_ath             0.434    0.079    5.473    0.000
##   internal_value =~
##     sprt_impt             0.784    0.066   11.800    0.000
##     think_sprt            0.854    0.074   11.557    0.000
##   negative_events =~
##     dprs_sprt             0.816    0.070   11.601    0.000
##     bad_sprt              0.829    0.071   11.676    0.000
##   athlete_identity =~
##     external_dntty        2.481    0.509    4.872    0.000
##     internal_value        1.510    0.173    8.741    0.000
##     negative_evnts        1.289    0.145    8.917    0.000
##   healthy_lifestyle =~
##     hr_sleep              0.447    0.094    4.777    0.000
##     smoking              -0.374    0.091   -4.116    0.000
##     fruit_veg            -0.092    0.036   -2.539    0.011
##   resilience =~
##     bounce                0.719    0.037   19.237    0.000
##     strs_evnt             0.816    0.043   18.798    0.000
##     strs_rcvr             0.726    0.043   16.890    0.000
##     snap_back             0.861    0.040   21.518    0.000
##     difficult             0.692    0.042   16.545    0.000
##     setbacks              0.834    0.039   21.366    0.000
##
## Covariances:
##                        Estimate  Std.Err  z-value  P(>|z|)
##   athlete_identity ~~
##     healthy_lfstyl        0.440    0.103    4.263    0.000
##     resilience            0.060    0.053    1.123    0.262
##   healthy_lifestyle ~~
##     resilience            0.353    0.098    3.611    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .sprt_goals        0.582    0.062    9.459    0.000
##    .cnsdr_ath         0.825    0.089    9.316    0.000
##    .frnds_ath         1.496    0.111   13.457    0.000
##    .sprt_impt         0.882    0.097    9.084    0.000
##    .think_sprt        0.645    0.102    6.338    0.000
##    .dprs_sprt         0.993    0.121    8.238    0.000
##    .bad_sprt          1.160    0.130    8.959    0.000
##    .hr_sleep          0.864    0.092    9.343    0.000
##    .smoking           1.271    0.100   12.705    0.000
##    .fruit_veg         0.239    0.017   14.415    0.000
##    .bounce            0.337    0.027   12.695    0.000
##    .strs_evnt         0.470    0.036   12.886    0.000
```

```
##    .strs_rcvr         0.522    0.039   13.547    0.000
##    .snap_back         0.313    0.028   11.328    0.000
##    .difficult         0.503    0.037   13.644    0.000
##    .setbacks          0.303    0.026   11.444    0.000
##    .external_dntty    1.000
##    .internal_value    1.000
##    .negative_evnts    1.000
##     athlete_idntty    1.000
##     healthy_lfstyl    1.000
##     resilience        1.000
```

Structural Equation Modeling

```
athlete_sem <- '
  # measurement model
    external_identity =~ sprt_goals + cnsdr_ath + frnds_ath
    internal_value =~ sprt_impt + think_sprt
    negative_events =~ dprs_sprt + bad_sprt

    athlete_identity =~ external_identity + internal_value + negative_events

    healthy_lifestyle =~ hr_sleep + smoking + fruit_veg

    resilience =~ bounce + strs_evnt + strs_rcvr + snap_back + difficult + setbacks

  # structural model - direct effects
    mhc_sf ~ a*athlete_identity + b*healthy_lifestyle + c*resilience
    resilience ~ d*athlete_identity

  # indirect
    indirect_athlete_identity := d*c

  # total
    total_athlete_identity:= d*c + a
'

athlete_sem_fit <- sem(athlete_sem,
                       data = dataset,
                       sample.cov = TRUE,
                       missing = "ML")
summary(athlete_sem_fit)
```

```
## lavaan 0.6-10 ended normally after 130 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        58
##
##   Number of observations                           753
##   Number of missing patterns                         6
##
## Model Test User Model:
##
```

```
##   Test statistic                               273.299
##   Degrees of freedom                               112
##   P-value (Chi-square)                           0.000
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Observed
##   Observed information based on                Hessian
##
## Latent Variables:
##                        Estimate  Std.Err  z-value  P(>|z|)
##   external_identity =~
##     sprt_goals            1.000
##     cnsdr_ath             1.203    0.052   23.046    0.000
##     frnds_ath             0.902    0.054   16.807    0.000
##   internal_value =~
##     sprt_impt             1.000
##     think_sprt            1.099    0.053   20.861    0.000
##   negative_events =~
##     dprs_sprt             1.000
##     bad_sprt              1.045    0.069   15.198    0.000
##   athlete_identity =~
##     external_dntty        1.000
##     internal_value        1.005    0.077   13.015    0.000
##     negative_evnts        0.894    0.072   12.379    0.000
##   healthy_lifestyle =~
##     hr_sleep              1.000
##     smoking              -0.473    0.146   -3.243    0.001
##     fruit_veg            -0.168    0.064   -2.639    0.008
##   resilience =~
##     bounce                1.000
##     strs_evnt             1.112    0.056   19.717    0.000
##     strs_rcvr             1.010    0.054   18.603    0.000
##     snap_back             1.177    0.054   21.788    0.000
##     difficult             0.965    0.053   18.261    0.000
##     setbacks              1.167    0.053   22.074    0.000
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   mhc_sf ~
##     athlt_dntt (a)   -3.163    1.625   -1.947    0.052
##     hlthy_lfst (b)    9.299    4.982    1.866    0.062
##     resilience (c)    8.776    0.762   11.522    0.000
##   resilience ~
##     athlt_dntt (d)    0.050    0.033    1.529    0.126
##
## Covariances:
##                       Estimate  Std.Err  z-value  P(>|z|)
##   athlete_identity ~~
##     healthy_lfstyl       0.431    0.120    3.577    0.000
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
```

```
##    .sprt_goals        5.502    0.066   83.661    0.000
##    .cnsdr_ath         5.092    0.079   64.625    0.000
##    .frnds_ath         4.613    0.074   61.937    0.000
##    .sprt_impt         4.693    0.075   62.892    0.000
##    .think_sprt        4.405    0.077   57.085    0.000
##    .dprs_sprt         5.225    0.072   72.173    0.000
##    .bad_sprt          5.040    0.076   65.923    0.000
##    .hr_sleep         13.207    0.080  165.988    0.000
##    .smoking           1.659    0.043   39.007    0.000
##    .fruit_veg         1.456    0.018   80.199    0.000
##    .bounce            3.856    0.037  105.602    0.000
##    .strs_evnt         3.304    0.041   80.509    0.000
##    .strs_rcvr         3.479    0.040   87.369    0.000
##    .snap_back         3.426    0.040   86.028    0.000
##    .difficult         3.324    0.038   86.493    0.000
##    .setbacks          3.499    0.039   89.051    0.000
##    .mhc_sf           36.091    0.520   69.412    0.000
##    .external_dntty    0.000
##    .internal_value    0.000
##    .negative_evnts    0.000
##     athlete_idntty    0.000
##     healthy_lfstyl    0.000
##    .resilience        0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .sprt_goals        0.570    0.058    9.750    0.000
##    .cnsdr_ath         0.799    0.085    9.448    0.000
##    .frnds_ath         1.512    0.107   14.166    0.000
##    .sprt_impt         0.830    0.089    9.318    0.000
##    .think_sprt        0.601    0.095    6.304    0.000
##    .dprs_sprt         0.984    0.109    8.997    0.000
##    .bad_sprt          1.139    0.122    9.363    0.000
##    .hr_sleep          4.321    0.288   15.004    0.000
##    .smoking           1.262    0.079   15.971    0.000
##    .fruit_veg         0.235    0.014   17.266    0.000
##    .bounce            0.366    0.024   15.331    0.000
##    .strs_evnt         0.476    0.031   15.495    0.000
##    .strs_rcvr         0.523    0.032   16.128    0.000
##    .snap_back         0.335    0.024   13.887    0.000
##    .difficult         0.496    0.031   16.234    0.000
##    .setbacks          0.320    0.023   13.799    0.000
##    .mhc_sf          119.391   18.484    6.459    0.000
##    .external_dntty    0.274    0.085    3.238    0.001
##    .internal_value    0.644    0.104    6.223    0.000
##    .negative_evnts    0.614    0.106    5.806    0.000
##     athlete_idntty    1.403    0.156    9.021    0.000
##     healthy_lfstyl    0.446    0.210    2.124    0.034
##    .resilience        0.516    0.046   11.211    0.000
##
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|)
##     indrct_thlt_dn    0.437    0.289    1.513    0.130
##     ttl_thlt_dntty   -2.726    1.632   -1.670    0.095
```

```
semPaths(athlete_sem_fit,
         whatLabels = "std",
         reorder = FALSE,
         layout = "tree2")
```