Exam-02

Pablo Vivas

2023-12-19

Table of contents

# Question 1

data = read\_csv("./data/personality-1.csv")  
  
data\_sub = data |>   
 select(str\_c("pers", c("03", "07", 11, 13, 16, 26, 36)))

## a)

The single-factor model using seven indicators is statistically identified. This because the model contains parameters and the correlation matrix contains unique pieces of information, also known as degrees of freedom. Hence, we have more degrees of freedom than model parameters which results in the model being statistically identified. (We have degrees of freedom to spare 🥳)

## b)

mod\_01 = '  
 f1 =~ pers03 + pers07 + pers11 + pers13 + pers16 + pers26 + pers36  
'  
  
cfa\_01 = cfa(mod\_01, data = data\_sub)  
  
fitMeasures(cfa\_01) |>   
 tidy() |>  
 mutate(  
 measure = names,  
 value = round(x, 2),  
 .keep = "unused") |>   
 filter(measure %in% c("chisq", "pvalue", "cfi", "tli", "rmsea", "srmr")) |>   
 gt() |>   
 cols\_width(  
 value ~ px(100)  
 )

| measure | value |
| --- | --- |
| chisq | 335.48 |
| pvalue | 0.00 |
| cfi | 0.46 |
| tli | 0.19 |
| rmsea | 0.23 |
| srmr | 0.17 |

**?(caption)**

Given the results of **?@tbl-fit-one-factor-cfa**, the model doesn’t seem to fit the data. Hence, no interpretation of the parameter estimates is needed.

## c)

residuals(cfa\_01, type = "standardized")

$type  
[1] "standardized"  
  
$cov  
 pers03 pers07 pers11 pers13 pers16 pers26 pers36  
pers03 0.000   
pers07 -1.273 0.000   
pers11 -1.573 -1.031 0.000   
pers13 4.594 1.889 -3.704 0.000   
pers16 -1.317 0.380 7.996 -5.858 0.000   
pers26 -0.538 -2.414 6.839 -4.201 8.375 0.000   
pers36 -3.965 0.214 5.930 -1.999 7.634 7.289 0.000

We see huge standardized residuals, specifically in the correlation estimates of pers36 and the other items. This is probably due to the wording of this question and its relationship to the other items (they may be correlated). Also, the biggest misfit is happening in the estimate for the correlation between per26 and per16, more than 8 units off. However, it is unclear why is this happening with this two items that seem somewhat different (per16:generates enthusiasm in others, per26: assertive).

modindices(cfa\_01, sort = T)

# A tibble: 21 × 8  
 lhs op rhs mi epc sepc.lv sepc.all sepc.nox  
 <chr> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
 1 pers16 ~~ pers26 86.3 0.505 0.505 0.449 0.449  
 2 pers11 ~~ pers16 76.9 0.485 0.485 0.423 0.423  
 3 pers16 ~~ pers36 69.9 0.452 0.452 0.406 0.406  
 4 pers26 ~~ pers36 62.4 0.498 0.498 0.383 0.383  
 5 pers11 ~~ pers26 53.4 0.471 0.471 0.352 0.352  
 6 pers03 ~~ pers13 49.4 0.833 0.833 1.71 1.71   
 7 pers11 ~~ pers36 39.1 0.401 0.401 0.302 0.302  
 8 pers13 ~~ pers16 28.2 -0.247 -0.247 -0.358 -0.358  
 9 pers13 ~~ pers26 15.7 -0.212 -0.212 -0.263 -0.263  
10 pers03 ~~ pers36 13.4 -0.182 -0.182 -0.231 -0.231  
# ℹ 11 more rows

As suspected, adding some correlation terms between the error component of the items will improve the fit in the model. This makes sense as some of the items are very similar and probably is easy to understand within the personality theories. Given that we have only degrees of freedom left, I will add correlation terms as specified in the table above. This because I am interested in maintaining an over-identified model while still be able to asses the fit of it.

## d)

mod\_02 = '  
 f1 =~ pers03 + pers07 + pers11 + pers13 + pers16 + pers26 + pers36  
 pers16 ~~ pers26  
 pers11 ~~ pers16  
 pers16 ~~ pers36  
 pers26 ~~ pers36  
 pers11 ~~ pers26  
 pers03 ~~ pers13  
 pers11 ~~ pers36  
 pers13 ~~ pers16  
 pers13 ~~ pers26  
 pers03 ~~ pers36  
'  
  
cfa\_02 = cfa(mod\_02, data = data\_sub)  
  
fitMeasures(cfa\_02) |>   
 tidy() |>  
 mutate(  
 measure = names,  
 value = round(x, 2),  
 .keep = "unused") |>   
 filter(measure %in% c("chisq", "pvalue", "cfi", "tli", "rmsea", "srmr")) |>   
 gt() |>   
 cols\_width(  
 value ~ px(100)  
 )

| measure | value |
| --- | --- |
| chisq | 5.92 |
| pvalue | 0.21 |
| cfi | 1.00 |
| tli | 0.98 |
| rmsea | 0.03 |
| srmr | 0.02 |

**?(caption)**

Given the results of **?@tbl-fit-one-factor-cfa-corr**, we can claim that the data does fit the specified one-factor model with some correlated errors. Comparing this table with the results of **?@tbl-fit-one-factor-cfa**, it is clear the improvement of the original model fitted originally. Also, it is worth mentioning that these two models are nested (the model with no correlation terms is nested into the model with correlation error terms). Because of this fact, we can compare the fit of this two nested models through a test.

anova(cfa\_01, cfa\_02)

# A tibble: 2 × 8  
 Df AIC BIC Chisq `Chisq diff` RMSEA `Df diff` `Pr(>Chisq)`  
 <int> <dbl> <dbl> <dbl> <dbl> <dbl> <int> <dbl>  
1 4 8608. 8706. 5.92 NA NA NA NA   
2 14 8917. 8975. 335. 330. 0.269 10 8.62e-65

The results on this test point out what we found with the goodness of fit indexes: the second model does a better job in describing the relationship between the 7 items we analyzed. From a information-based criteria, the second model also has smaller values of AIC and BIC.

## e)

summary(cfa\_02, standardized=T)

lavaan 0.6.16 ended normally after 46 iterations  
  
 Estimator ML  
 Optimization method NLMINB  
 Number of model parameters 24  
  
 Used Total  
 Number of observations 440 452  
  
Model Test User Model:  
   
 Test statistic 5.922  
 Degrees of freedom 4  
 P-value (Chi-square) 0.205  
  
Parameter Estimates:  
  
 Standard errors Standard  
 Information Expected  
 Information saturated (h1) model Structured  
  
Latent Variables:  
 Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
 f1 =~   
 pers03 1.000 0.505 0.515  
 pers07 1.296 0.458 2.831 0.005 0.654 0.666  
 pers11 0.160 0.147 1.089 0.276 0.081 0.068  
 pers13 1.274 0.153 8.313 0.000 0.643 0.613  
 pers16 0.374 0.142 2.627 0.009 0.189 0.189  
 pers26 0.150 0.152 0.990 0.322 0.076 0.065  
 pers36 0.469 0.170 2.762 0.006 0.237 0.203  
  
Covariances:  
 Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
 .pers16 ~~   
 .pers26 0.518 0.061 8.495 0.000 0.518 0.453  
 .pers11 ~~   
 .pers16 0.491 0.061 8.029 0.000 0.491 0.424  
 .pers16 ~~   
 .pers36 0.446 0.061 7.321 0.000 0.446 0.397  
 .pers26 ~~   
 .pers36 0.528 0.070 7.550 0.000 0.528 0.397  
 .pers11 ~~   
 .pers26 0.492 0.070 7.054 0.000 0.492 0.359  
 .pers03 ~~   
 .pers13 0.247 0.120 2.058 0.040 0.247 0.355  
 .pers11 ~~   
 .pers36 0.423 0.069 6.159 0.000 0.423 0.313  
 .pers13 ~~   
 .pers16 -0.081 0.038 -2.112 0.035 -0.081 -0.099  
 .pers26 -0.019 0.044 -0.438 0.661 -0.019 -0.020  
 .pers03 ~~   
 .pers36 -0.072 0.042 -1.727 0.084 -0.072 -0.074  
  
Variances:  
 Estimate Std.Err z-value P(>|z|) Std.lv Std.all  
 .pers03 0.708 0.105 6.728 0.000 0.708 0.735  
 .pers07 0.536 0.149 3.597 0.000 0.536 0.556  
 .pers11 1.390 0.094 14.787 0.000 1.390 0.995  
 .pers13 0.687 0.152 4.525 0.000 0.687 0.624  
 .pers16 0.966 0.067 14.335 0.000 0.966 0.964  
 .pers26 1.350 0.091 14.778 0.000 1.350 0.996  
 .pers36 1.309 0.091 14.381 0.000 1.309 0.959  
 f1 0.255 0.103 2.481 0.013 1.000 1.000

With the above information we can claim that these items are measuring a latent factor that I’ve decided to name active worker. It is worth mentioning that some of the items are measuring the same aspect of this latent variable, that is why some correlated errors are introduced in the model.

# Question 2

## a)

cor\_mat = data |>   
 drop\_na() |>   
 cor()  
  
cortest.bartlett(cor\_mat, n = nrow(data |> drop\_na()))

$chisq  
[1] 6730.363  
  
$p.value  
[1] 0  
  
$df  
[1] 946

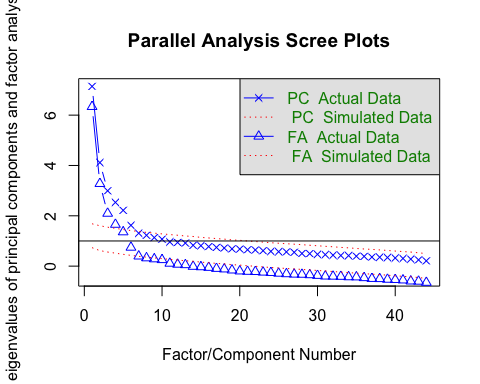
KMO(data |> drop\_na())

Kaiser-Meyer-Olkin factor adequacy  
Call: KMO(r = drop\_na(data))  
Overall MSA = 0.84  
MSA for each item =   
pers01 pers02 pers03 pers04 pers05 pers06 pers07 pers08 pers09 pers10 pers11   
 0.83 0.76 0.91 0.87 0.85 0.85 0.89 0.90 0.79 0.80 0.79   
pers12 pers13 pers14 pers15 pers16 pers17 pers18 pers19 pers20 pers21 pers22   
 0.76 0.87 0.87 0.76 0.85 0.84 0.83 0.81 0.82 0.86 0.87   
pers23 pers24 pers25 pers26 pers27 pers28 pers29 pers30 pers31 pers32 pers33   
 0.90 0.87 0.75 0.84 0.84 0.90 0.79 0.68 0.84 0.85 0.80   
pers34 pers35 pers36 pers37 pers38 pers39 pers40 pers41 pers42 pers43 pers44   
 0.79 0.71 0.89 0.84 0.88 0.88 0.79 0.74 0.88 0.89 0.66

Given the results of this two test, we can conclude that the data is suitable for factor analysis. First, the Bartlett’s test of sphericity provides evidence that our correlation matrix is different from the identity matrix and that there is an intercorrelation between variables that can be explained by common factors. Then, the KMO provides evidence that our data suitable for this analysis, with an overall measure of sampling adequacy of , which is great

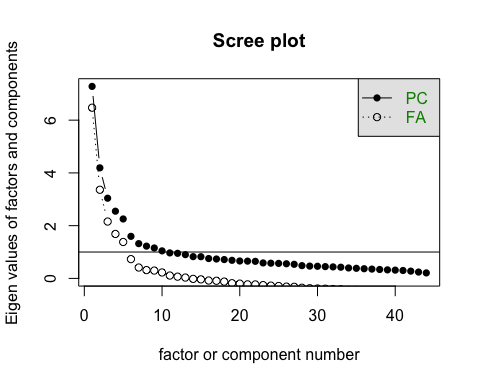
## b)

fa.parallel(data, fm = "ml", n.obs = nrow(data |> drop\_na()))



Parallel analysis suggests that the number of factors = 6 and the number of components = 6

scree(data |> drop\_na())



Both the parallel analysis and the scree-plot suggest that the number of factors to be extracted is .

efa\_01 = fa(cor\_mat,   
 nfactors = 6,   
 n.obs = nrow(data |> drop\_na()),   
 fm = "ml",   
 rotate = "none")  
  
summary(efa\_01)

Factor analysis with Call: fa(r = cor\_mat, nfactors = 6, n.obs = nrow(drop\_na(data)), rotate = "none",   
 fm = "ml")  
  
Test of the hypothesis that 6 factors are sufficient.  
The degrees of freedom for the model is 697 and the objective function was 3.26   
The number of observations was 433 with Chi Square = 1344.09 with prob < 1.7e-43   
  
The root mean square of the residuals (RMSA) is 0.04   
The df corrected root mean square of the residuals is 0.04   
  
Tucker Lewis Index of factoring reliability = 0.846  
RMSEA index = 0.046 and the 10 % confidence intervals are 0.043 0.05  
BIC = -2887.21

Based on the goodness-of-fit indices, this model seems to be doing a decent job in explaining the intercorrelation between the items. Even thought the test is significant, the other measures to evaluate how good the data fits the model are withing acceptable ranges.

## c)

Because these items are similar, I would prefer an oblique rotation to allow correlation between the latent factors.

efa\_02 = fa(cor\_mat,   
 nfactors = 6,   
 n.obs = nrow(data |> drop\_na()),   
 fm="ml",   
 rotate = "promax")  
  
efa\_02$loadings

Loadings:  
 ML2 ML1 ML3 ML5 ML4 ML6   
pers01 -0.663 0.210 0.333 -0.106 0.313  
pers02 0.187 -0.129 0.193 -0.451 0.364  
pers03 0.664 0.122 0.129  
pers04 0.122 0.512 -0.147 -0.120  
pers05 0.120 -0.199 0.535   
pers06 0.128 0.728 0.119   
pers07 0.190 0.502   
pers08 -0.620 0.165  
pers09 0.166 -0.716 0.162  
pers10 0.362 0.275  
pers11 -0.267 0.485  
pers12 -0.321 0.196  
pers13 0.528 0.104 0.253   
pers14 0.676   
pers15 0.118 0.138 -0.140 0.530   
pers16 -0.225 0.130 0.139 0.528  
pers17 -0.138 -0.107 0.405 0.139  
pers18 -0.725 0.149   
pers19 0.631 0.126 0.130 -0.145  
pers20 -0.114 0.156 0.471 0.178  
pers21 0.892 0.108 -0.162  
pers22 0.671   
pers23 -0.616 -0.123   
pers24 -0.626 0.201  
pers25 -0.135 0.515 0.201  
pers26 0.131 -0.319 -0.181 -0.204 0.512  
pers27 0.269 -0.316 0.166  
pers28 0.597 0.161   
pers29 0.452 0.127  
pers30 0.150 0.530   
pers31 0.700 0.188 -0.117  
pers32 0.156 0.664   
pers33 0.655 0.133 -0.108 0.249  
pers34 0.143 -0.638 0.207  
pers35 0.224 -0.205   
pers36 -0.512 0.225 -0.101 0.427  
pers37 -0.228 -0.107 -0.400 0.234  
pers38 0.500 0.195  
pers39 0.384 0.437 0.135   
pers40 0.139 0.621   
pers41 -0.387 0.105  
pers42 0.120 0.103 0.595   
pers43 -0.525 0.164 0.236  
pers44 0.118 0.491   
  
 ML2 ML1 ML3 ML5 ML4 ML6  
SS loadings 3.636 3.222 3.193 2.780 2.441 1.936  
Proportion Var 0.083 0.073 0.073 0.063 0.055 0.044  
Cumulative Var 0.083 0.156 0.228 0.292 0.347 0.391

We see some of the largest factor loading to be negative. I attribute this to the nature of the items in this test. Given that these items are trying to measure personality traits, it is tenable to have items with negative loads on some of the traits and still have a meaningful interpretation. Take, for example, item 8 careless. A negative load of this item on some of the latent variable would indicate that the person is not careless, which means that this worker is careful. In other words, these items may be in reverse coding.

With the loading of the rotated solution, I can provide the following interpretation of the latent factors:

* Factor 1 (Calmn): This factor might capture emotional stability and calmness, with items like “emotionally stable,” “calm in tense situations,” and “relaxed” showing high loadings.
* Factor 2 (Reliable): Indicated by high loadings on items like “does a thorough job,” “reliable,” and “perseveres,” this factor likely measures conscientiousness, work ethic, and reliability.
* Factor 3 (Worry): This factor might capture worry, with items like “depressed,” “not relaxed,” and “worries” showing high loadings.
* Factor 4 (Creative): Suggested by high loadings on “original,” “imaginative,” and “values artistic experiences,” this factor seems to represent openness to experience and creativity.
* Factor 5 (Trustworthy): Indicated by high loadings on items like “trusting,” “considerate,” and “co-operative,” this factor likely measures to what extent the worker is trustworthy.
* Factor 6 (Leadership): This factor might be related to assertiveness and leadership qualities, as indicated by items like “assertive,” “generates enthusiasm in others,” and “sophisticated in art & music” having high loadings.

## d)

efa\_02$communalities |>   
 tidy() |>   
 mutate(  
 item = parse\_number(names),  
 communality = x,  
 .keep = "unused"  
 ) |>   
 ggplot(aes(x = item, y = communality)) +  
 geom\_col(fill = "tomato4") +  
 ylim(c(0,1)) +  
 theme\_bw()

|  |
| --- |
| Figure 1: Communalities |

Low communalities in a factor analysis, like the ones observed in [Figure 1](#fig-plot-com) (specifically, pers10, pers12, pers17, pers35), suggest that these items are not well explained by the underlined extracted factors. In this context, this can mean that these specific items do not align well with the underlying constructs we are trying to measure with this instrument. In other words, these items might be measuring aspects of personality that are not captured by the six factors we’ve identified, or they may be less relevant or inconsistent in the context of the other items and factors in this questionnaire.

## e)

Factor Analysis (FA) is more appropriate for this dataset than Principal Component Analysis (PCA) because FA seeks to identify latent variables that explain observed variables, which is suitable for psychological and personality data, like the one we have. FA models the underlying structure that explains correlations between items, focusing on shared variance. In contrast, PCA maximizes total variance, treating all variance as equally important. PCA would provide a different perspective by combining items into components based on total variance, potentially mixing measurement and error variances, which might not be as meaningful for understanding underlying personality constructs.

## f)

data |>   
 drop\_na() |>   
 skimr::skim() |>   
 as\_tibble() |>   
 select(skim\_variable, numeric.sd) |>   
 sample\_n(10) |>   
 gt()

| skim\_variable | numeric.sd |
| --- | --- |
| pers03 | 0.9766104 |
| pers34 | 1.1216669 |
| pers31 | 1.3184107 |
| pers32 | 0.9908917 |
| pers20 | 1.0869224 |
| pers15 | 1.1121947 |
| pers19 | 1.3425434 |
| pers11 | 1.1773949 |
| pers25 | 1.1305018 |
| pers35 | 1.2622846 |

Performing an exploratory factor analysis using the covariance matrix is reasonable for this dataset, as the items (pers01 to pers44) are on the same 5-point scale and have similar standard deviations (and hence variances). This similarity in scaling and variance allows for a meaningful comparison of the covariance among items. The results of the factor analysis using the covariance matrix would focus more directly on the shared variances in their original scale, potentially providing insights that align more closely with the actual variance observed in the data. In conclusion, the result will be very similar.

# Question 3

## a)

By definition of our model, we have the following:

Using linear algebra, we can compute the fitted correlation matrix as follows:

## b)

For each item, the communality can be computed as follows:

* Variable 1:
* Variable 2:
* Variable 3:
* Variable 4:
* Variable 5:
* Variable 6:

The uniqueness can be computed as the complement of the communality, this because the correlation matrix was used to perform the factor analysis:

* Variable 1:
* Variable 2:
* Variable 3:
* Variable 4:
* Variable 5:
* Variable 6: