A19

Bryana Benson
Conner Bryan

A1p

Libraries

```
#install.packages(c("psych", "GPArotation"))
library(readr)
library(psych)
library(GPArotation)
library(dplyr)
library(ggplot2)
```

Reading the Data

```
url<-"https://www.richardtwatson.com/data/jury.csv"
P_data<-read_csv(url)
head(P_data)
```

```
## # A tibble: 6 x 18
##   Crime Phys_attr_manip Gender_defendent Gender_subject Sentence Serious
##   <dbl>           <dbl>            <dbl>          <dbl>    <dbl>   <dbl>
## 1     1               2                2              1        4       4
## 2     1               2                2              1        3       4
## 3     1               2                2              1        4       3
## 4     1               2                2              1        8       5
## 5     1               2                2              1        2       6
## 6     1               2                2              1        7       3
## # ... with 12 more variables: Exciting <dbl>, Calm <dbl>,
## #   Independent <dbl>, Sincere <dbl>, Warm <dbl>, Phys_attr <dbl>,
## #   Kind <dbl>, Intelligent <dbl>, Strong <dbl>, Sophist <dbl>,
## #   Happy <dbl>, Sociable <dbl>
```

```
colnames(P_data)
```

```
##  [1] "Crime"           "Phys_attr_manip"  "Gender_defendent"
##  [4] "Gender_subject"  "Sentence"         "Serious"
##  [7] "Exciting"        "Calm"             "Independent"
## [10] "Sincere"         "Warm"             "Phys_attr"
## [13] "Kind"            "Intelligent"      "Strong"
## [16] "Sophist"         "Happy"            "Sociable"
```

```
P_data=P_data[complete.cases(P_data),]
P_factor <- P_data %>% select(-Crime,-Phys_attr_manip,
                              -Gender_defendent,-Sentence,
                              -Gender_subject)
colnames(P_factor)
```

```
##  [1] "Serious"     "Exciting"    "Calm"        "Independent" "Sincere"
##  [6] "Warm"        "Phys_attr"   "Kind"        "Intelligent" "Strong"
## [11] "Sophist"     "Happy"       "Sociable"
```

Running Diagnostics

```
P_cor <- cor(P_factor)
cortest.bartlett(P_factor, n=nrow(P_data)) # Null: correlations don't matter
```

```
## $chisq
## [1] 1077.908
##
## $p.value
## [1] 1.115361e-175
##
## $df
## [1] 78
```

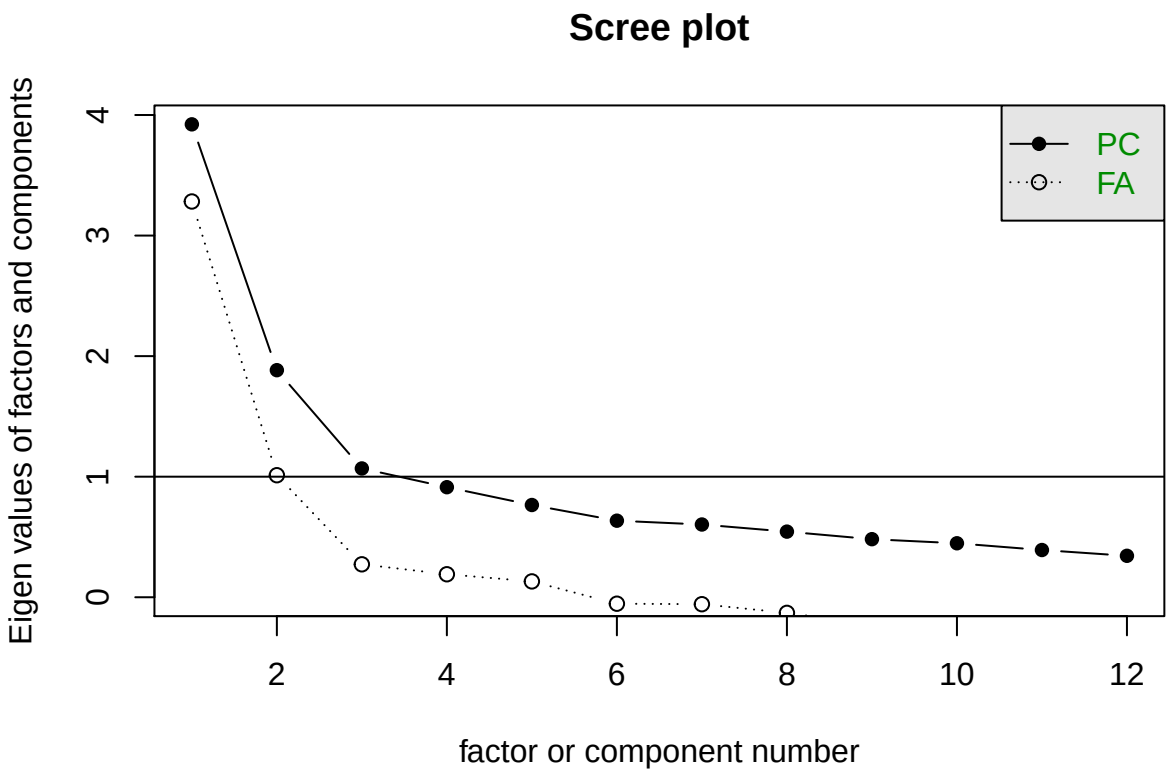
```
                                          # Also: is matrix factorable?
KMO(P_factor) # measure the quality of data for factor analysis
```

```
## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = P_factor)
## Overall MSA =  0.81
## MSA for each item =
##     Serious    Exciting        Calm Independent     Sincere        Warm
##        0.44        0.79        0.82        0.79        0.70        0.68
##   Phys_attr        Kind Intelligent      Strong     Sophist       Happy
##        0.79        0.76        0.89        0.87        0.85        0.88
##    Sociable
##        0.87
```

```
              # desired score between .8 and 1

##### Removing those variables whose KMO index is <.7
P_factor_red <- P_factor %>% select(-Serious)
P_cor_1<-cor(P_factor_red)

##### How many factors?
scree(P_factor_red)
```



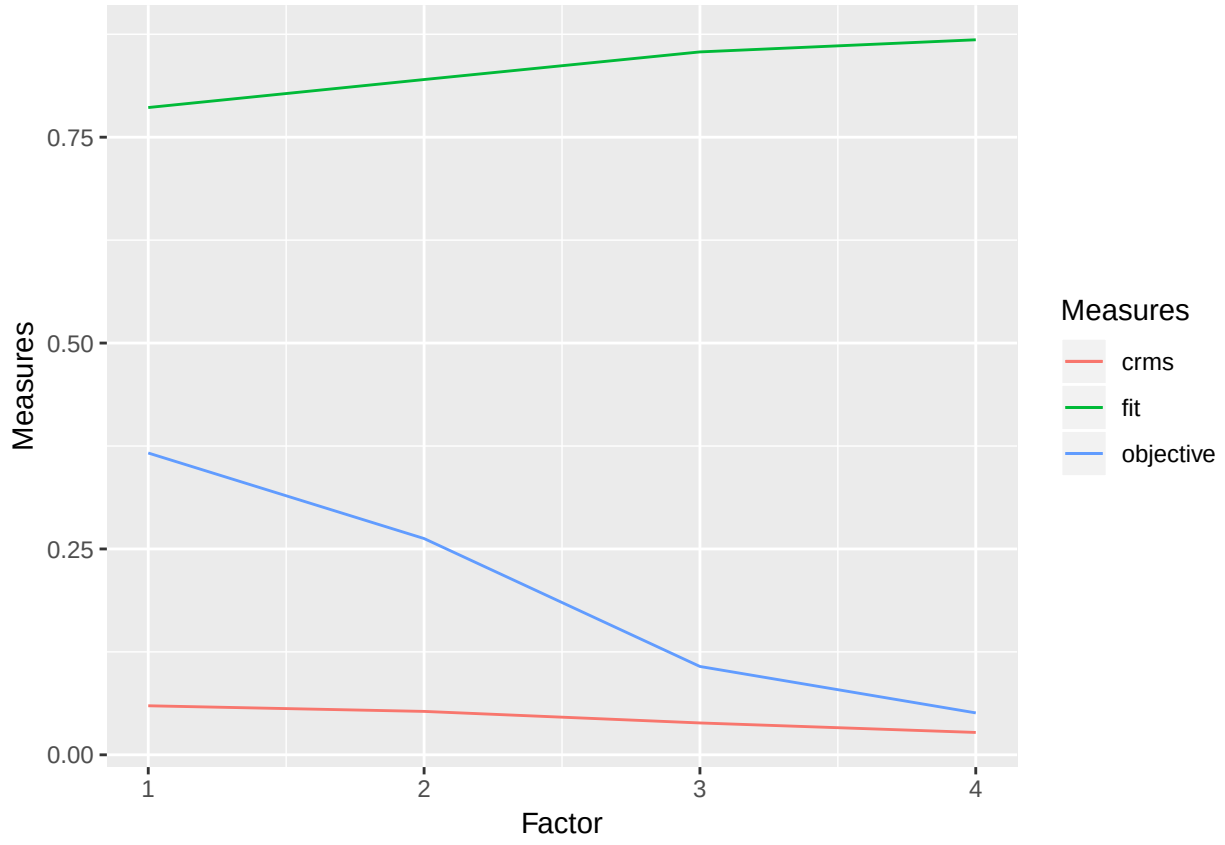
Testing Different Numbers of Factors

```
results <- tibble(factors = integer())
start <- 2
end <- 5
for(i in start:end) {
  fit <- fa(r = P_cor_1, nfactors = i, rotate = "oblimin", fm="minres")
  # Record fit measures
  results[i-start+1,1] <- i-1
  # rms adjusted for degrees of freedom
  # the sum of the squared off diagonal residuals divided by the degrees of freedom
  results[i-start+1,2] <- fit$rms
  results[i-start+1,3] <- fit$crms
  # How well the factor model reproduce the correlation matrix
  results[i-start+1,4] <- fit$fit
  # Value of the function that is minimized by a maximum likelihood procedures
  results[i-start+1,5] <- fit$objective
}

colnames(results)[2]<-"rms"
colnames(results)[3]<-"crms"
colnames(results)[4]<-"fit"
colnames(results)[5]<-"objective"

#plotting some measures
```

```
ggplot(data=results) +
  geom_line(mapping = aes(x=factors, y=objective, color = 'objective')) +
  geom_line(mapping = aes(x=factors, y=crms, color='crms')) +
  geom_line(mapping = aes(x=factors, y=fit, color='fit')) +
  scale_color_hue() +
  labs(color = 'Measures') +
  xlab('Factor') +
  ylab('Measures')
```



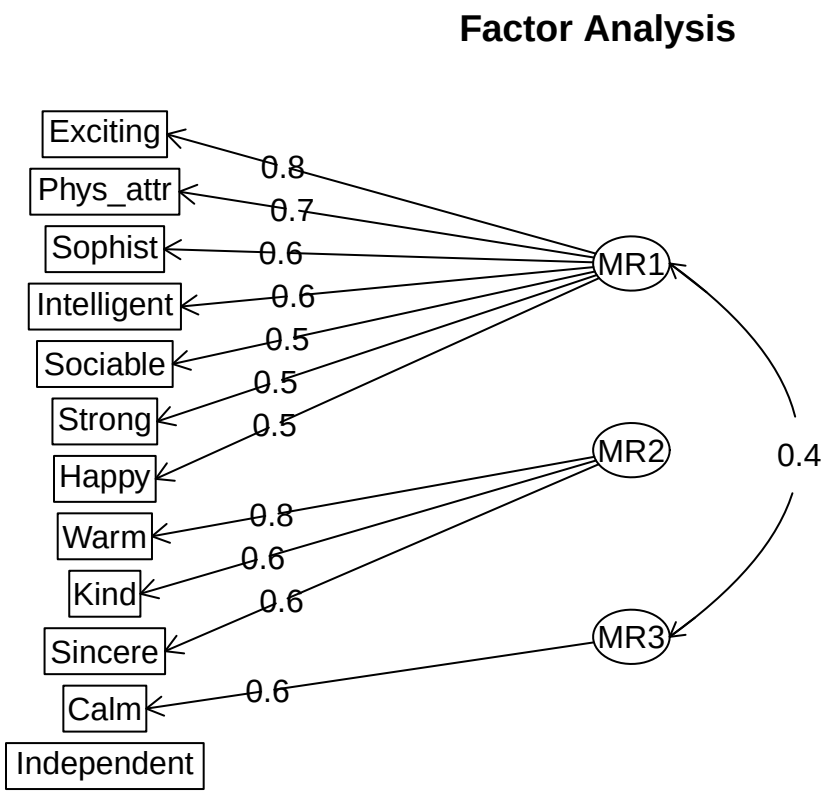
Diagramming Three Factor Solution

```
fit <- fa(r = P_cor_1, nfactors = 3, rotate = "oblimin",fm="minres")
print(fit$loadings,cutoff = 0.3)
```

```
##
## Loadings:
##             MR1    MR2    MR3
## Exciting     0.771
## Calm                       0.638
## Independent
## Sincere             0.634
## Warm                0.839
## Phys_attr    0.671
## Kind                0.637
## Intelligent  0.553
## Strong       0.540
```

```
## Sophist      0.600
## Happy        0.507
## Sociable     0.546
##
##                    MR1   MR2   MR3
## SS loadings    2.589 1.616 0.692
## Proportion Var 0.216 0.135 0.058
## Cumulative Var 0.216 0.350 0.408
```

```
fa.diagram(fit) #diagramming the factors
```




```
##### Renaming factors and redoing diagram with names
colnames(fit$loadings) <- c("Charisma", "Benevolence", "Placidity")
fa.diagram(fit)
```

Factor Analysis