Factor Analysis

Huey Kwik & Richard Zhang

May 17, 2016

Solutions adapted from work by Huey Kwik & Richard Zhang (Signal Cohort #2).

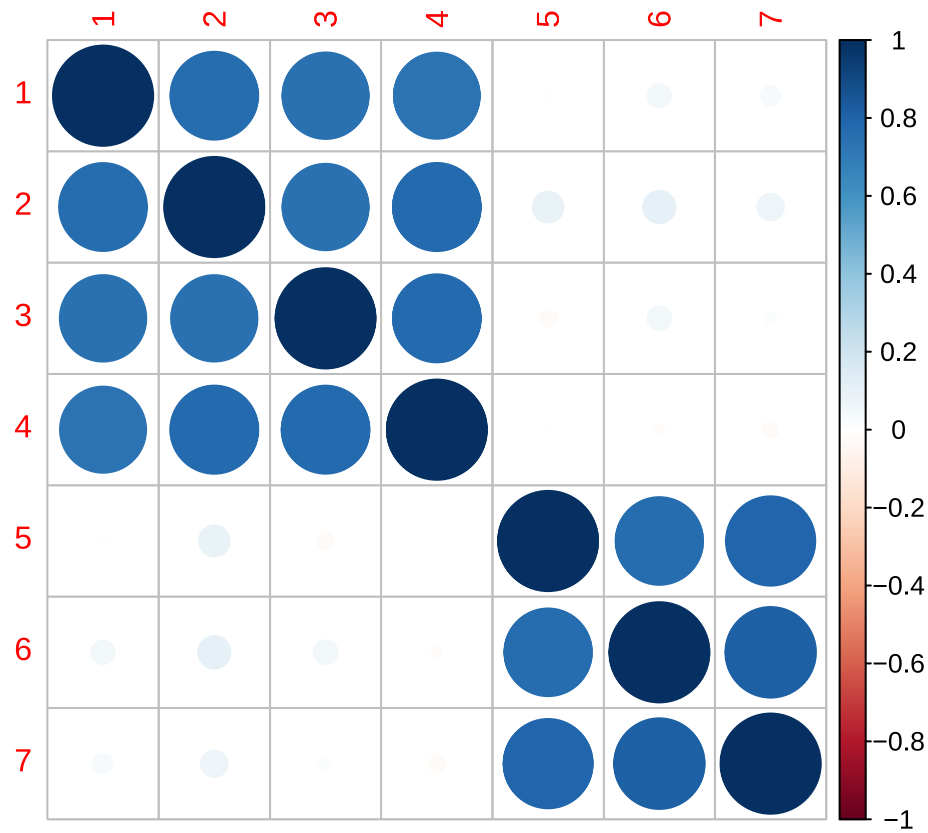
Factor Analysis

Let's create a factors data frame!

```
X = rnorm(100)
Y = rnorm(100)
Z = rnorm(100)

factors = data.frame(X,Y,Z)
colnames(factors) = c("X","Y","Z")

noisyIndicators = function(feature, k, correlation) {
  noisies = lapply(1:k, function(x) {
    error = rnorm(length(feature))
    d = sqrt(1-correlation^2)
    correlation * feature + d * error
  })
  return(noisies)
}

xProxies = noisyIndicators(X, 4, 0.9)
yProxies = noisyIndicators(Y, 3, 0.9)
noisies = data.frame(xProxies, yProxies)
colnames(noisies) = as.character(1:7)
```

Check out the correlations!

```
corrplot(cor(noisies), is.corr = TRUE)
```

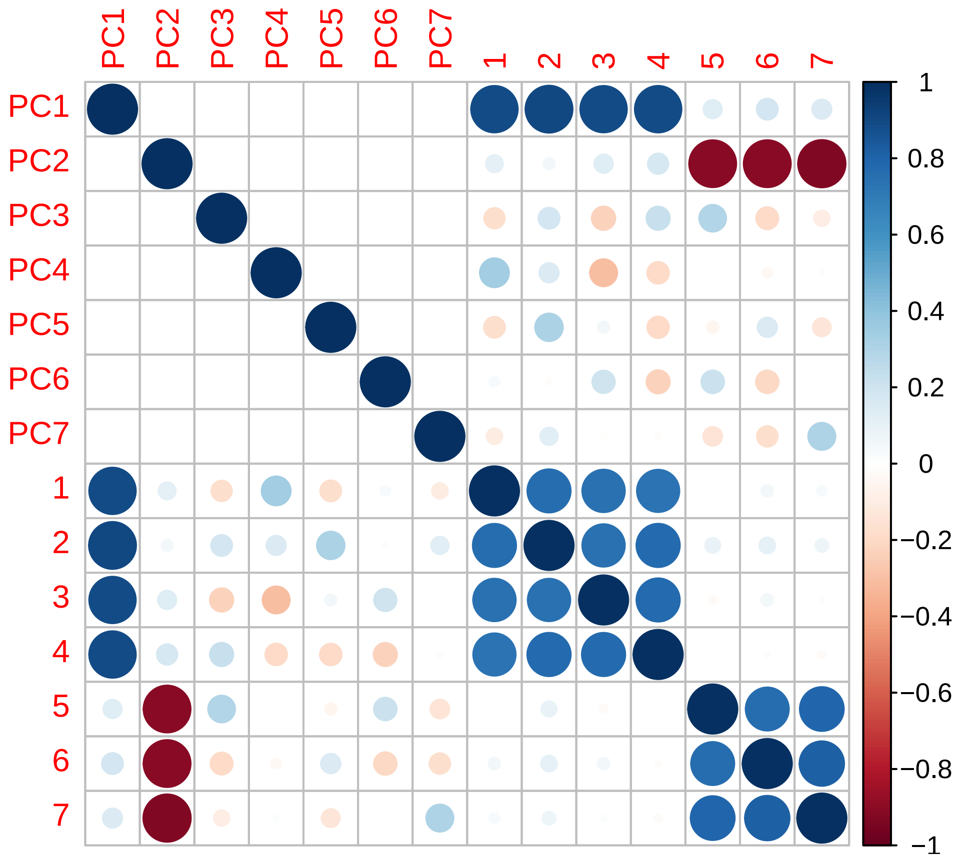



```r
cor(cbind(X, noisies))
```

```
##              X            1          2           3            4            5
## X 1.00000000  0.876642218 0.86845112  0.87648115  0.875168165  0.016068987
## 1 0.87664222  1.000000000 0.76888047  0.74443197  0.738252032 -0.004630102
## 2 0.86845112  0.768880474 1.00000000  0.74307901  0.772564203  0.094249628
## 3 0.87648115  0.744431974 0.74307901  1.00000000  0.771106341 -0.028559233
## 4 0.87516817  0.738252032 0.77256420  0.77110634  1.000000000 -0.000877307
## 5 0.01606899 -0.004630102 0.09424963 -0.02855923 -0.000877307  1.000000000
## 6 0.03183138  0.054670452 0.10448636  0.05675003 -0.010894037  0.765101282
## 7 0.04074964  0.038917144 0.07046893  0.01459206 -0.023761752  0.793888309
##             6           7
## X  0.03183138  0.04074964
## 1  0.05467045  0.03891714
## 2  0.10448636  0.07046893
## 3  0.05675003  0.01459206
## 4 -0.01089404 -0.02376175
## 5  0.76510128  0.79388831
## 6  1.00000000  0.81700686
## 7  0.81700686  1.00000000
```

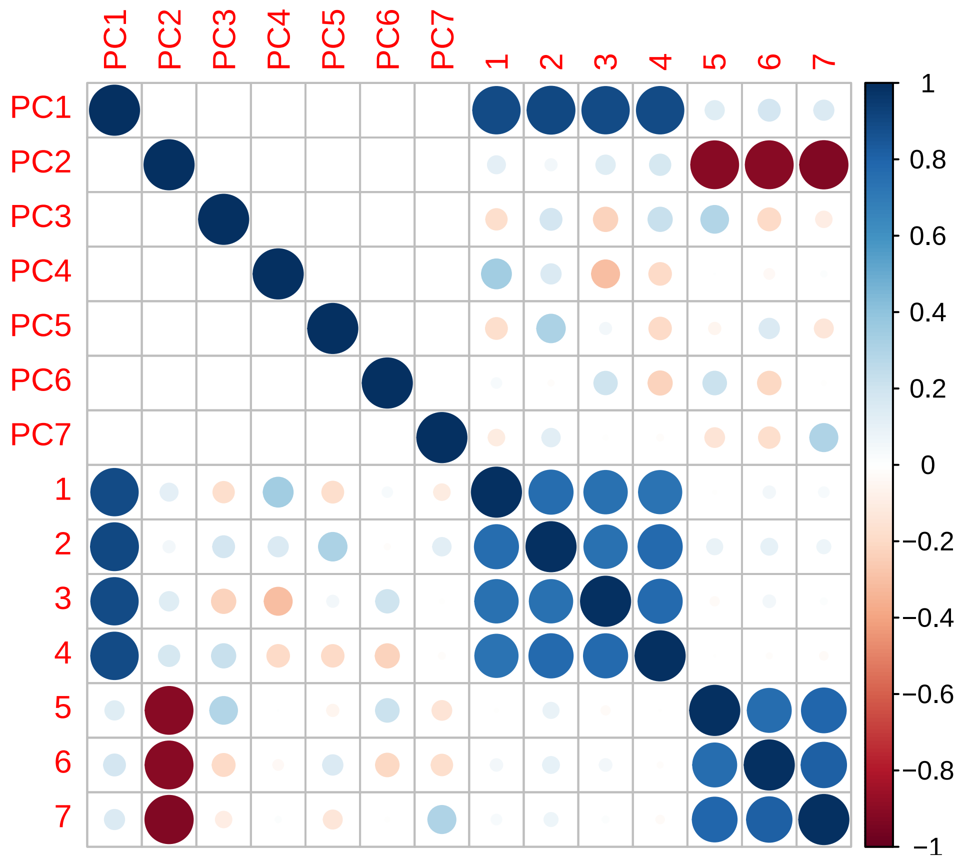
Run PCA on the noisy dataframe

```r
pca = prcomp(noisies, scale=TRUE)
cor_pca = cor(cbind(pca$x,noisies))
corrplot(cor_pca, is.corr=TRUE)
```



Part 2: Orthogonal Factor Analysis

```r
pca = prcomp(noisies, scale=TRUE)
cor_pca = cor(cbind(pca$x,noisies))
corrplot(cor_pca, is.corr=TRUE)
```



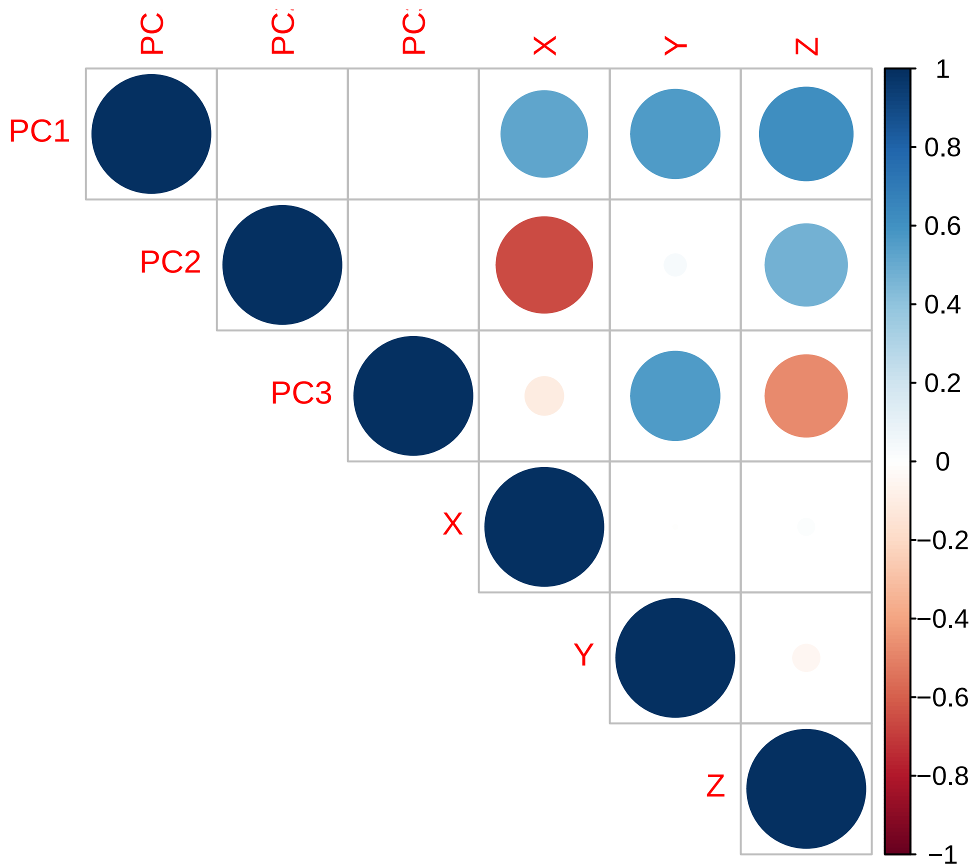
Generate Variables

```
vars = sapply(1:50, function(i) { X*runif(1)+Y*runif(2)+Z*runif(3)+0.5*rnorm(1) })
pca = prcomp(vars, scale=TRUE)
```

Principal component 1 picks up on some of X Y Z Principal component 2 picks up on more of X than Y and Z Principal component 3 picks up on Y and Z

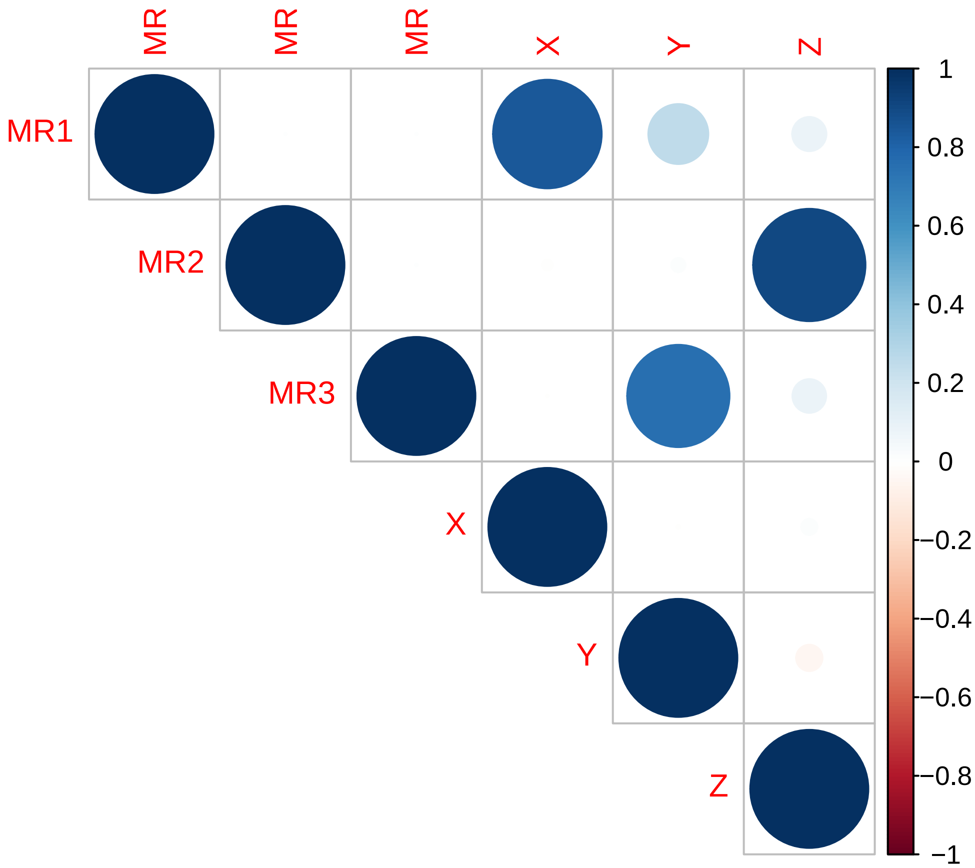
Corrplot of Principal Components with X,Y,Z

```
cor_pca = cor(cbind(pca$x[,1:3], X,Y,Z))
corrplot(cor_pca, is.corr=TRUE, type="upper")
```



MR1 mostly picks up on Y MR2 mostly picks up on X MR3 mostly picks up on Z

```
fac_f = fa(vars, nfactors=3, rotate="varimax")
cor_fac = cor(cbind(fac_f$scores, X,Y,Z))
corrplot(cor_fac, is.corr=TRUE, type="upper")
```



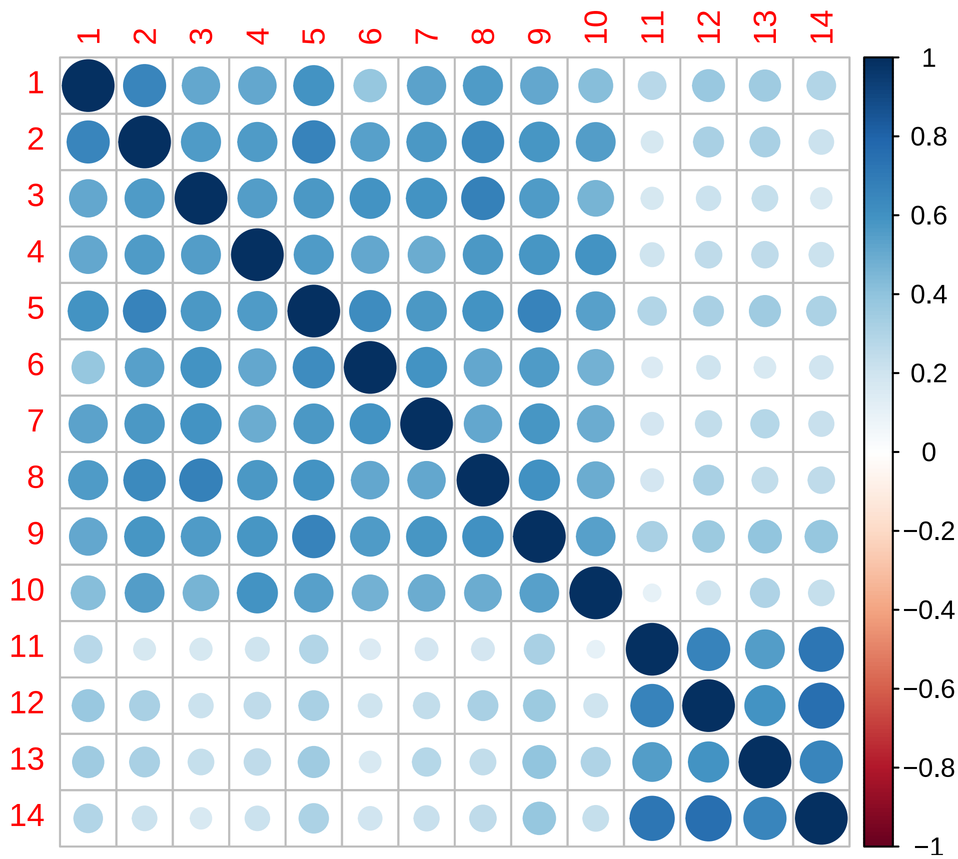
Part 3: Oblique Factor Analysis

Create a noisy dataframe!

```
W = 0.5*X + Y

cor(W,Y)
```

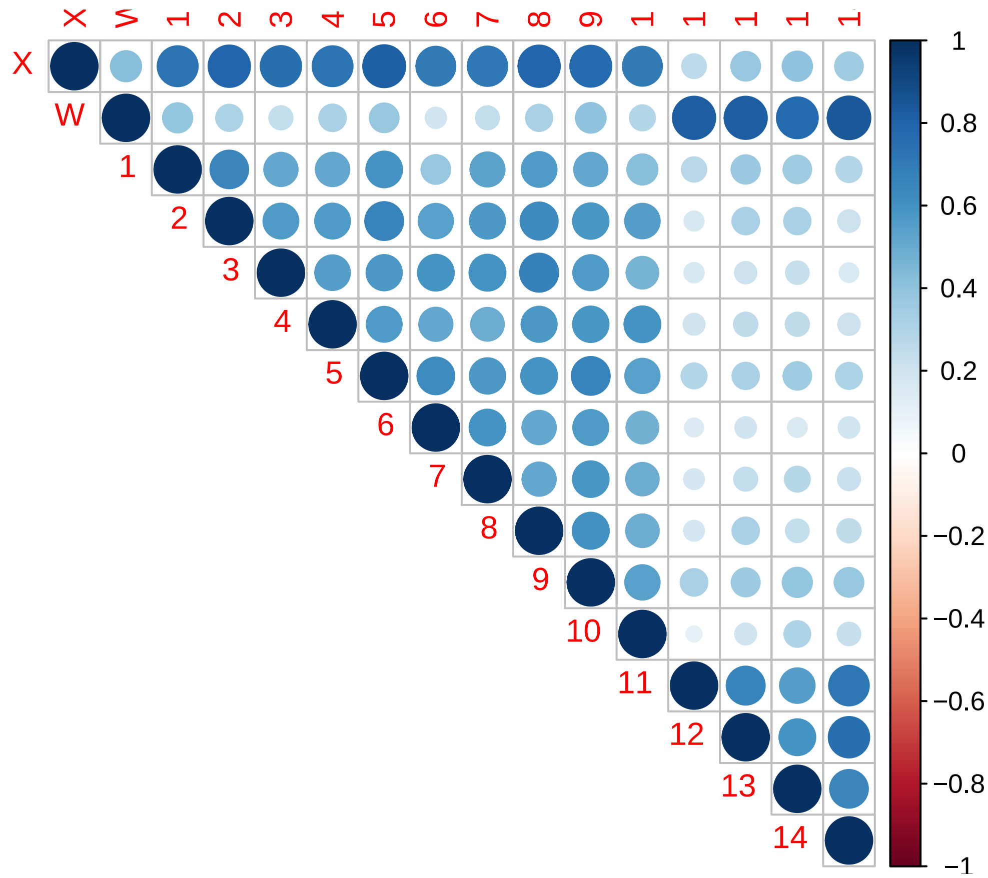
```
## [1] 0.9053488
```

```
xProxies = noisyIndicators(X, 10, 0.8)
wProxies = noisyIndicators(W, 4, 0.8)
noisies = data.frame(xProxies, wProxies)
colnames(noisies) = as.character(1:14)

corrplot(cor(noisies), is.corr = TRUE)
```

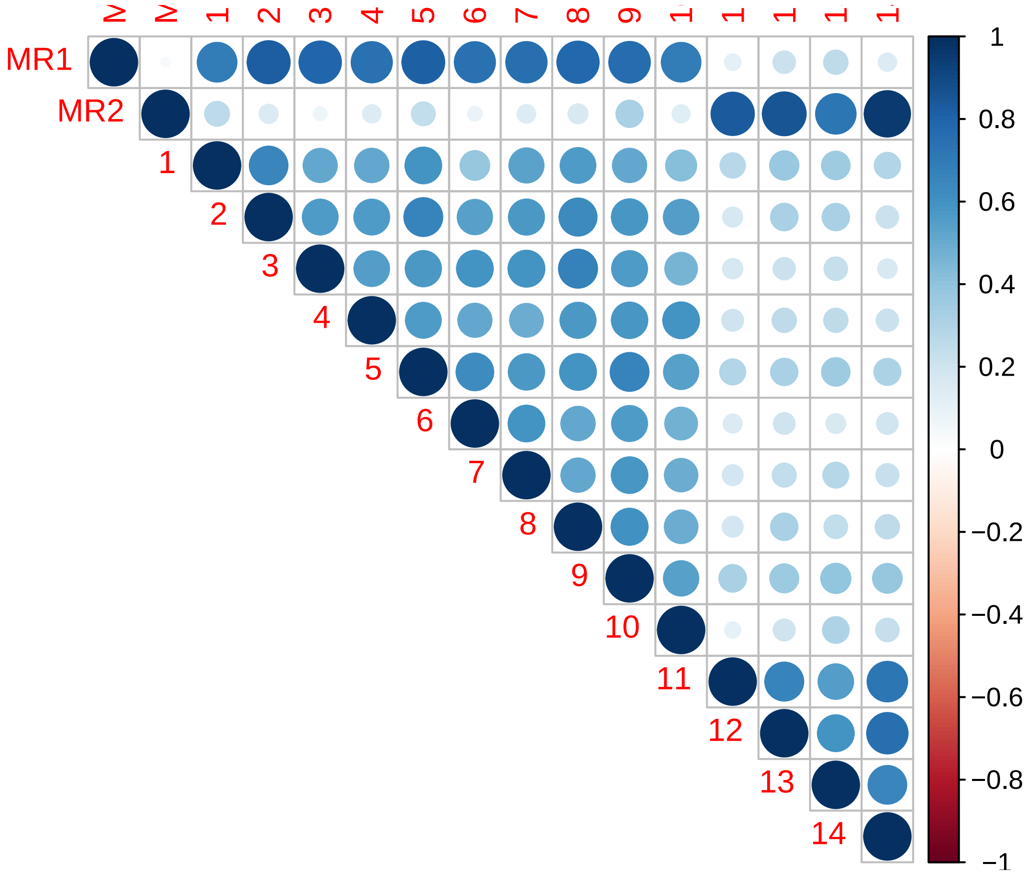


Compare varimax rotation with oblimin rotation

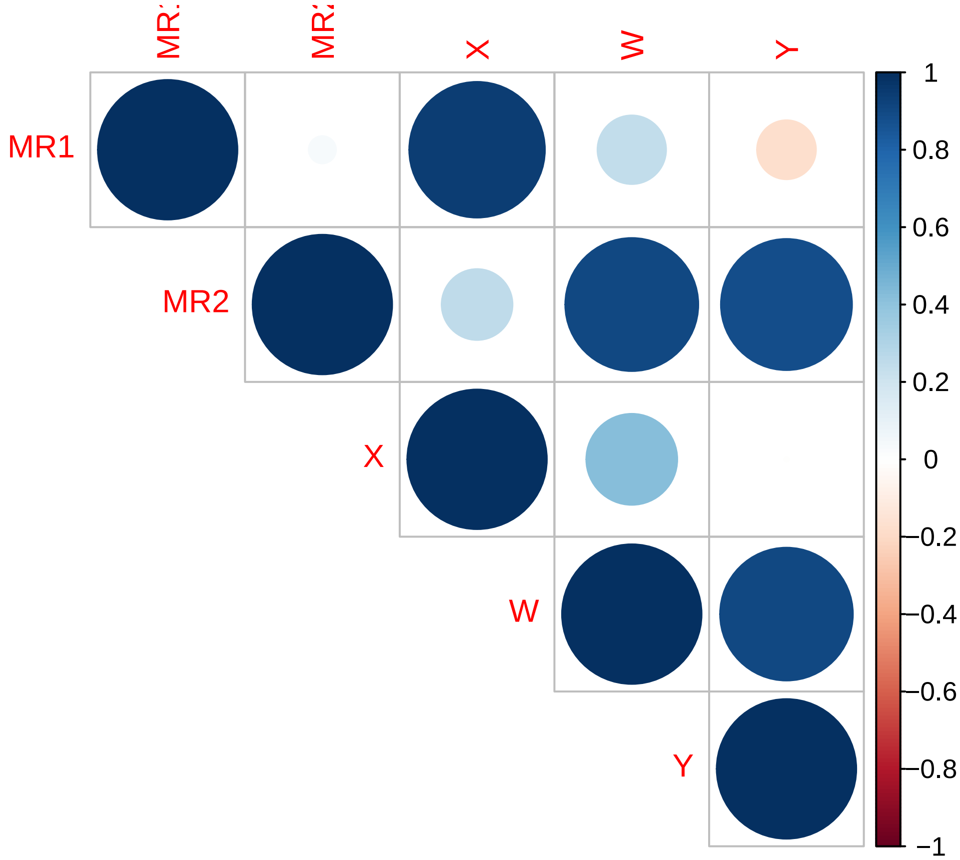
```
corrplot(cor(cbind(X, W, noisies)), is.corr=TRUE, type="upper")
```



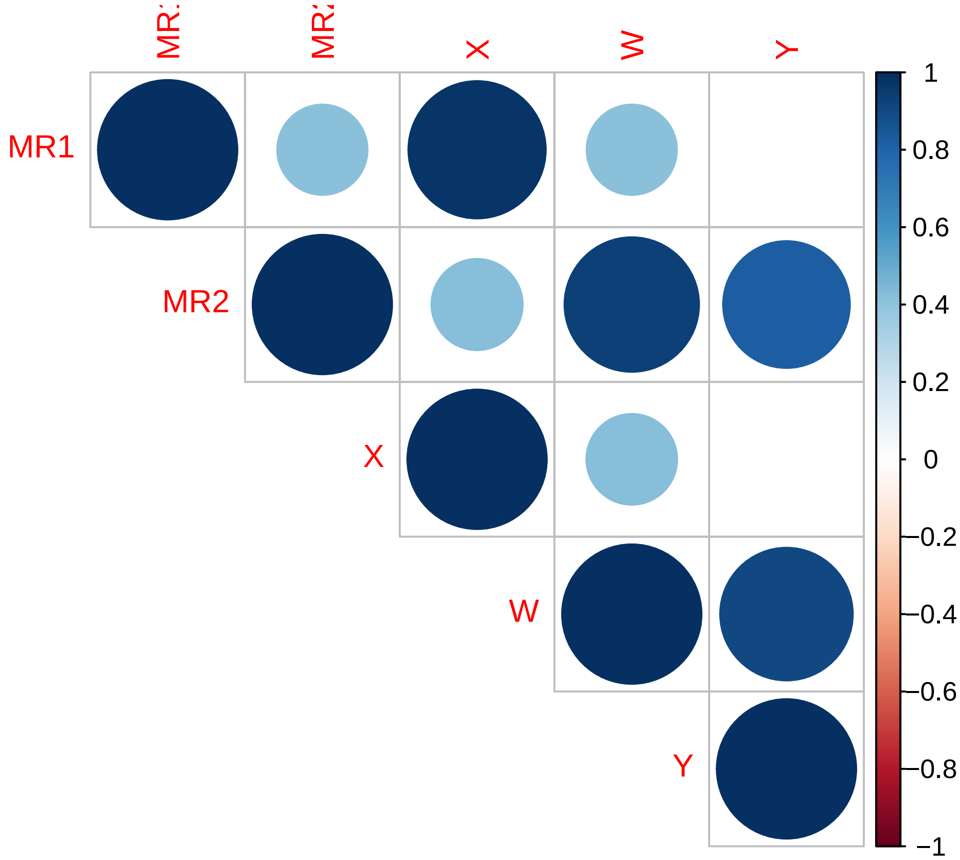

```
library('GPArotation')
fac_f = fa(noisies, nfactors=2, rotate="varimax")
fac_obli = fa(noisies, nfactors=2, rotate="oblimin")

cor_fac_noisies = cor(cbind(fac_f$scores, noisies))
corrplot(cor_fac_noisies, is.corr=TRUE, type="upper")
```




```
cor_fac = cor(cbind(fac_f$scores, X,W,Y))
corrplot(cor_fac, is.corr=TRUE, type="upper")
```




```
cor_fac_obli = cor(cbind(fac_obli$scores, X,W,Y))
corrplot(cor_fac_obli, is.corr=TRUE, type="upper")
```



Compared to varimax:

- MR1 is more highly correlated with MR2
- MR1 and X are more highly correlated
- MR2 and W are more highly correlated

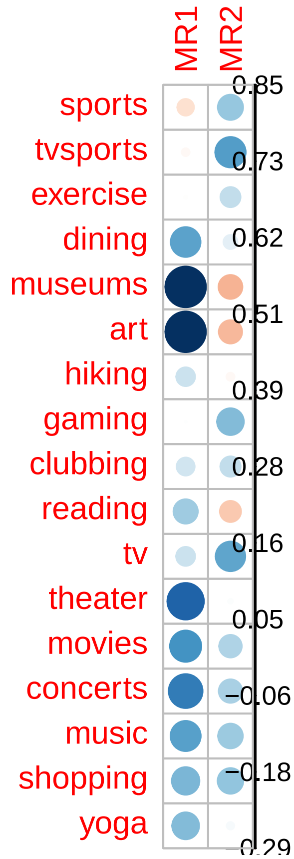
Speed Dating Data

Our goal is to detect factors in the speed dating dataset.

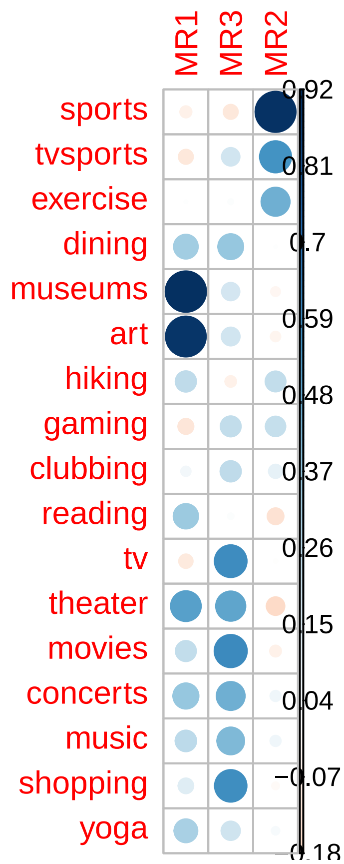
```
df = read.csv("C:/Users/Andrew/Documents/Signal/curriculum/datasets/speed-dating/speeddating-aggregated
activities = select(df, sports:yoga)
fac_activities = fa(activities, nfactor=1, rotate='varimax')
corrplot(fac_activities$loadings, is.corr=FALSE)
```

Corrplots with 1-4 factors, comparing varimax vs oblique rotation.

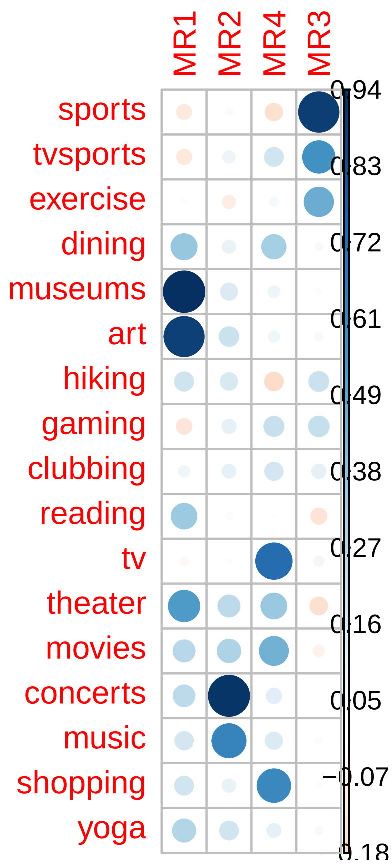
```
fac = function(n, rotation){
  fac_activities = fa(activities, nfactor=n, rotate=rotation)
  corrplot(fac_activities$loadings, is.corr=FALSE)
}
fac(2,"varimax")
```




```
fac(3,"varimax")
```



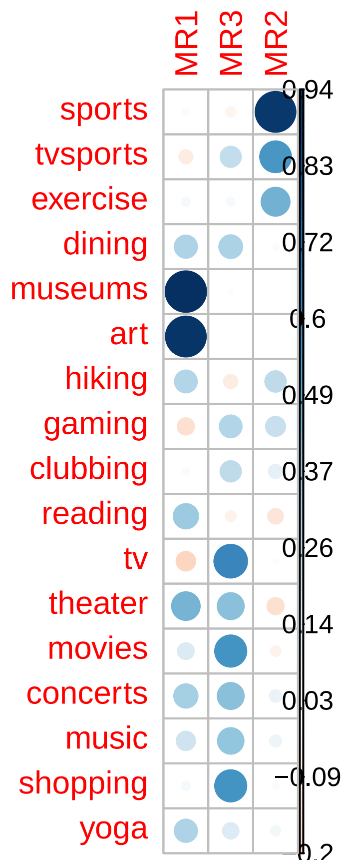

```
fac(4,"varimax")
```



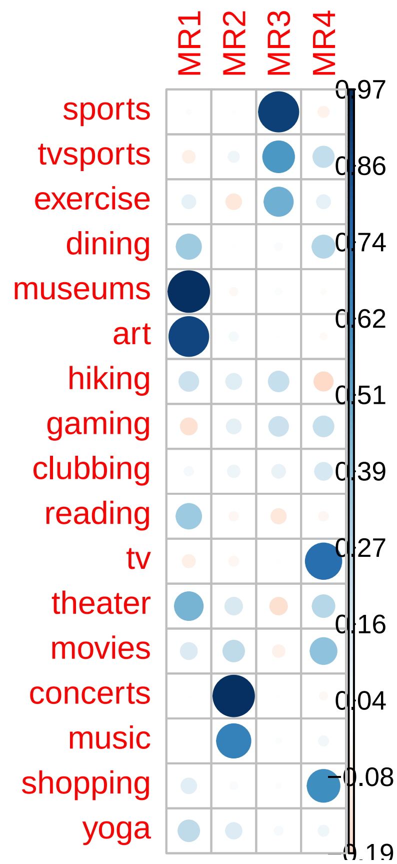

```r
fac(1,"oblimin")
```


```
fac(2,"oblimin")
```


```r
fac(3,"oblimin")
```



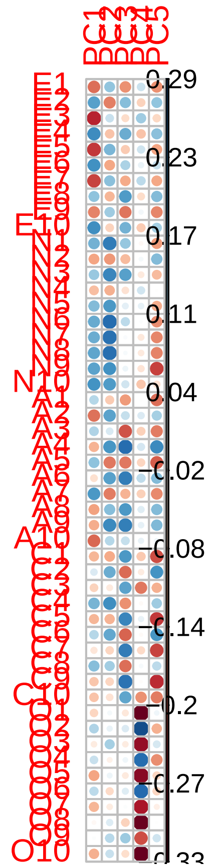

```r
fac(4,"oblimin") # Splits nicely into four factors!
```



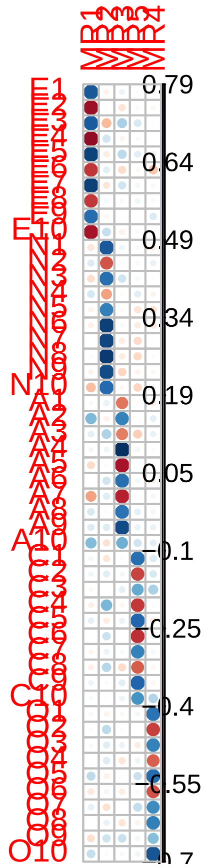
Big Five Personality Data

Compare Principal components and 5 factors analysis

```
df = read.csv("C:/Users/Andrew/Documents/Signal/curriculum/datasets/big-5/data.csv", sep="\t")
questions = select(df, E1:O10)
pca_5 = prcomp(scale(questions))
fac_5 = fa(questions, nfactor=5, rotate="varimax")

corrplot(pca_5$rotation[,1:5], is.corr=FALSE)
```



```r
corrplot(fac_5$loadings, is.corr=FALSE)
```



The former is very noisy, while the latter cleanly indicates 5 factors.

Regression Analysis: Predicting Gender using Questions

```r
gender_questions = cbind(df$gender, questions)
colnames(gender_questions)[1] = "gender"
gender_questions = dplyr::filter(gender_questions, gender != 3)
gender_questions$gender[gender_questions$gender == 2] = 0

fit = glm(gender ~ ., data = gender_questions, family = "binomial")
summary(fit) # Many many questions seem to be correlated with gender
```

```
##
## Call:
## glm(formula = gender ~ ., family = "binomial", data = gender_questions)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3908  -0.9209  -0.6161   1.0934   2.5505
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.359413   0.240968  -5.641 1.69e-08 ***
## E1           0.042750   0.017621   2.426 0.015262 *
## E2           0.156390   0.017039   9.178  < 2e-16 ***
## E3           0.068770   0.019596   3.509 0.000449 ***
```

```
## E4           0.128780   0.018606   6.921 4.48e-12 ***
## E5          -0.064293   0.019063  -3.373 0.000744 ***
## E6           0.136239   0.017432   7.815 5.49e-15 ***
## E7          -0.016877   0.017094  -0.987 0.323489
## E8           0.014910   0.015842   0.941 0.346600
## E9           0.203596   0.015220  13.377  < 2e-16 ***
## E10         -0.067985   0.016624  -4.090 4.32e-05 ***
## N1          -0.212953   0.016885 -12.612  < 2e-16 ***
## N2           0.118626   0.016809   7.057 1.70e-12 ***
## N3          -0.039166   0.018162  -2.156 0.031048 *
## N4          -0.012176   0.014725  -0.827 0.408317
## N5           0.070668   0.015298   4.619 3.85e-06 ***
## N6          -0.049504   0.017746  -2.790 0.005278 **
## N7          -0.053248   0.019872  -2.680 0.007371 **
## N8          -0.100914   0.020307  -4.969 6.72e-07 ***
## N9          -0.084193   0.017656  -4.768 1.86e-06 ***
## N10          0.074949   0.016948   4.422 9.76e-06 ***
## A1           0.073627   0.013386   5.500 3.79e-08 ***
## A2          -0.053637   0.019601  -2.736 0.006210 **
## A3           0.164755   0.015608  10.556  < 2e-16 ***
## A4          -0.049960   0.022643  -2.206 0.027351 *
## A5           0.060216   0.018490   3.257 0.001127 **
## A6          -0.014683   0.017583  -0.835 0.403691
## A7           0.022570   0.020374   1.108 0.267961
## A8          -0.091281   0.018812  -4.852 1.22e-06 ***
## A9          -0.085147   0.020687  -4.116 3.86e-05 ***
## A10         -0.058216   0.018661  -3.120 0.001810 **
## C1          -0.046409   0.018036  -2.573 0.010078 *
## C2          -0.065940   0.014456  -4.561 5.08e-06 ***
## C3          -0.010923   0.018290  -0.597 0.550354
## C4          -0.039407   0.017077  -2.308 0.021020 *
## C5          -0.017344   0.016255  -1.067 0.285975
## C6          -0.002221   0.014749  -0.151 0.880304
## C7           0.039822   0.016220   2.455 0.014083 *
## C8           0.088492   0.016908   5.234 1.66e-07 ***
## C9          -0.103364   0.015990  -6.464 1.02e-10 ***
## C10          0.017156   0.018519   0.926 0.354232
## O1           0.013403   0.019888   0.674 0.500350
## O2          -0.071681   0.018652  -3.843 0.000121 ***
## O3          -0.058774   0.020216  -2.907 0.003646 **
## O4          -0.044754   0.017896  -2.501 0.012392 *
## O5           0.127658   0.023672   5.393 6.94e-08 ***
## O6           0.055243   0.018868   2.928 0.003414 **
## O7          -0.034630   0.021105  -1.641 0.100835
## O8           0.092574   0.017235   5.371 7.82e-08 ***
## O9           0.057125   0.018139   3.149 0.001637 **
## O10          0.162444   0.023561   6.895 5.40e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 26199  on 19616  degrees of freedom
## Residual deviance: 23053  on 19566  degrees of freedom
```

```
## AIC: 23155
##
## Number of Fisher Scoring iterations: 3
```

Each question predicts gender poorly (low coefficients)

Factor Analysis: Predicting Gender using 5 Factors

```
gender_factors = as.data.frame(cbind(df$gender, fac_5$scores))
colnames(gender_factors)[1] = "gender"
gender_factors = dplyr::filter(gender_factors, gender != 3)
gender_factors$gender[gender_factors$gender == 2] = 0


fit = glm(gender ~ ., data = as.data.frame(gender_factors), family = "binomial")
summary(fit)
```

```
##
## Call:
## glm(formula = gender ~ ., family = "binomial", data = as.data.frame(gender_factors))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2404  -0.9487  -0.7185   1.1945   2.4040
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.49362    0.01550 -31.842  < 2e-16 ***
## MR1         -0.13299    0.01649  -8.064 7.39e-16 ***
## MR2         -0.42860    0.01689 -25.372  < 2e-16 ***
## MR3         -0.50956    0.01719 -29.644  < 2e-16 ***
## MR5         -0.11970    0.01714  -6.984 2.86e-12 ***
## MR4          0.29146    0.01743  16.718  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 26199  on 19616  degrees of freedom
## Residual deviance: 24238  on 19611  degrees of freedom
## AIC: 24250
##
## Number of Fisher Scoring iterations: 4
```

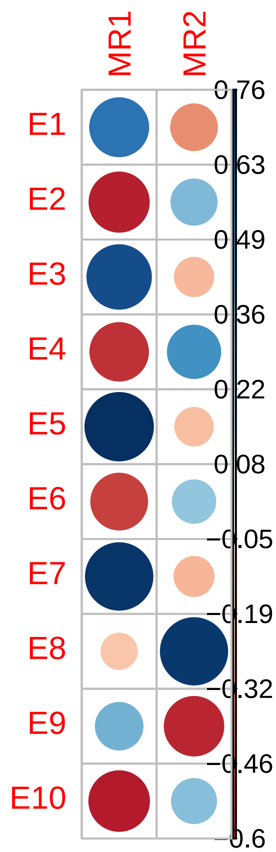
Each factor predicts gender with a comparatively large coefficient.

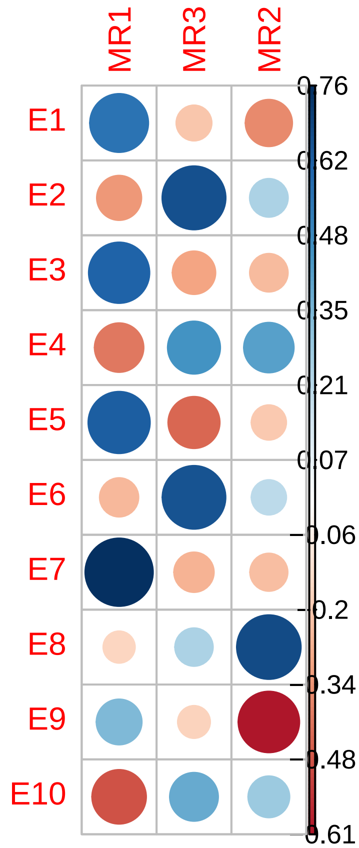
Identifying the 5 factors from the questions:

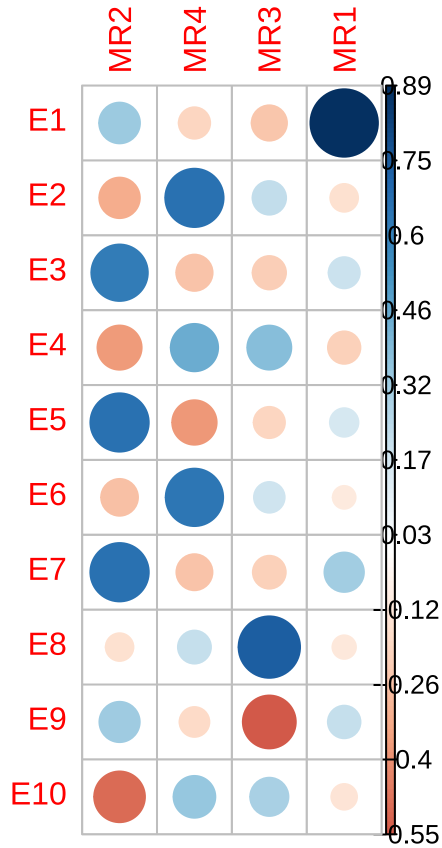
- E1-E10 -> Extraversion
- N1-N10 -> Neuroticism
- A1-A10 -> Agreeableness
- C1-C10 -> Conscientiousness
- O1-O10 -> Openness

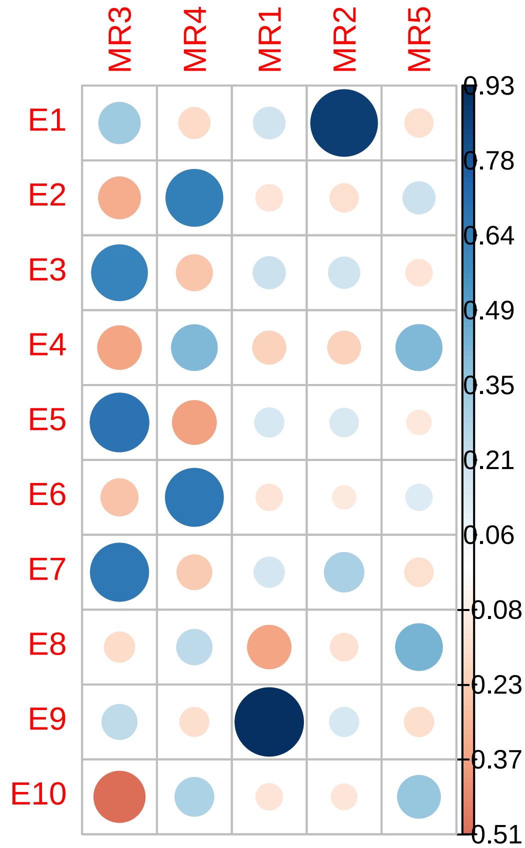
Identifying Subfactors

```
subfactors = function(questions) {
  fac_2 = fa(questions, nfactor=2, rotate="varimax")
  fac_3 = fa(questions, nfactor=3, rotate="varimax")
  fac_4 = fa(questions, nfactor=4, rotate="varimax")
  fac_5 = fa(questions, nfactor=5, rotate="varimax")

  # corrplot
  corrplot(fac_2$loadings, is.corr=FALSE)
  corrplot(fac_3$loadings, is.corr=FALSE)
  corrplot(fac_4$loadings, is.corr=FALSE)
  corrplot(fac_5$loadings, is.corr=FALSE)
}

subfactors(select(df, E1:E10))
```




MR1
MR3
MR2
E1
E2
E3
E4
E5
E6
E7
E8
E9
E10
0.76
0.62
0.48
0.35
0.21
0.07
−0.06
−0.2
−0.34
−0.48
−0.61

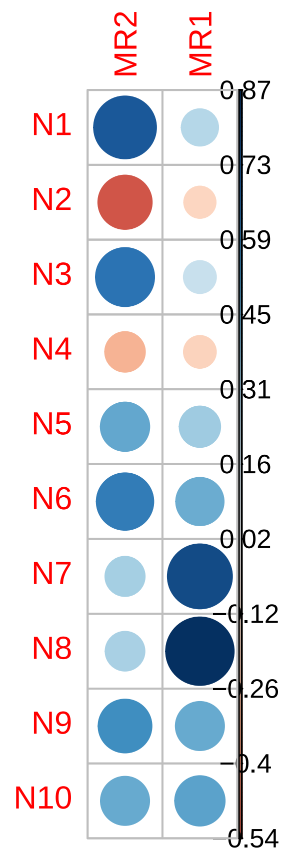

MR2
MR4
MR3
MR1
E1
E2
E3
E4
E5
E6
E7
E8
E9
E10
0.89
0.75
0.6
0.46
0.32
0.17
0.03
-0.12
-0.26
-0.4
-0.55

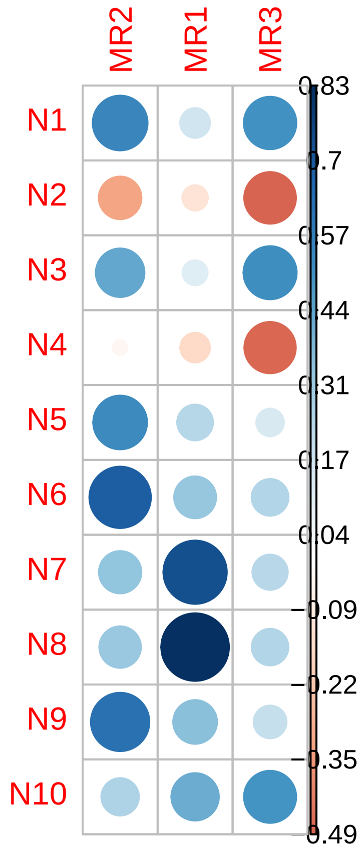


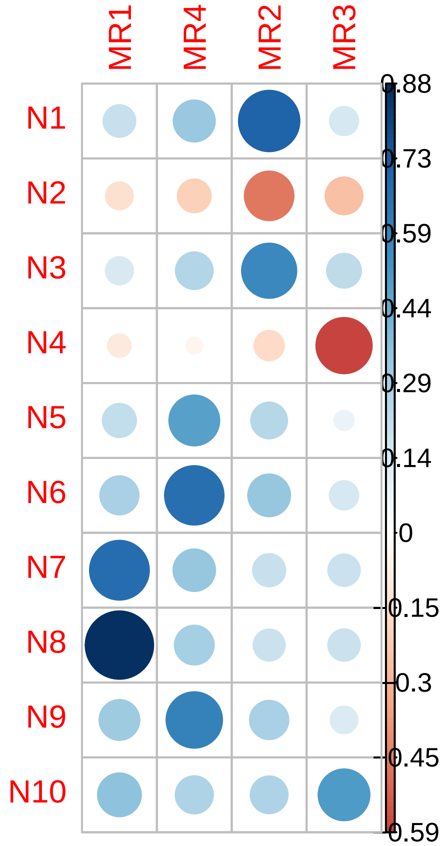
Extraversion:

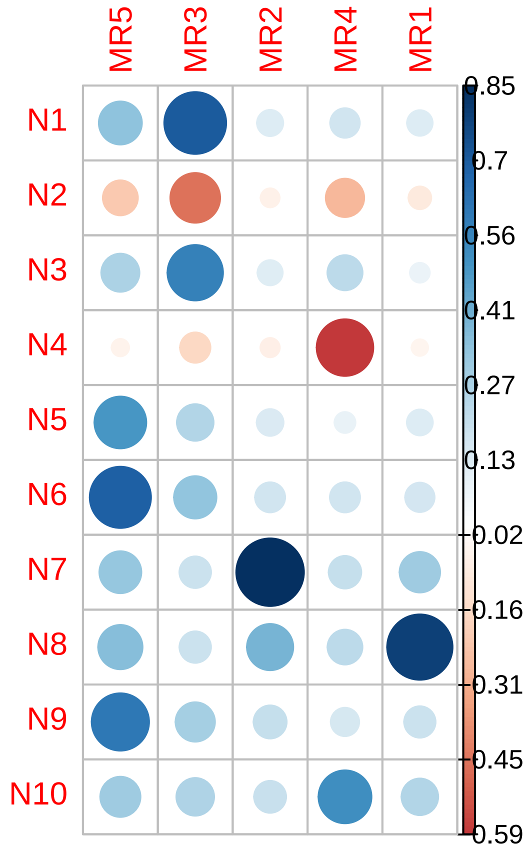
- 2 factor - Separates out E8/E9, but not coherent
- 3 factor - MR1 - talking to people, MR2 - attention, MR3 - talking
- 4 factor MR1 - life of the party, MR2 - talking to people, MR3 - attention, MR4 - talking
- 5 factor - Additional factor doesn't seem to add much value

```
subfactors(select(df, N1:N10))
```


MR2
MR1
N1
N2
N3
N4
N5
N6
N7
N8
N9
N10
0.87
0.73
0.59
0.45
0.31
0.16
0.02
−0.12
−0.26
−0.4
−0.54


MR2
MR1
MR3
N1
N2
N3
N4
N5
N6
N7
N8
N9
N10
0.83
0.7
0.57
0.44
0.31
0.17
0.04
−0.09
−0.22
−0.35
−0.49

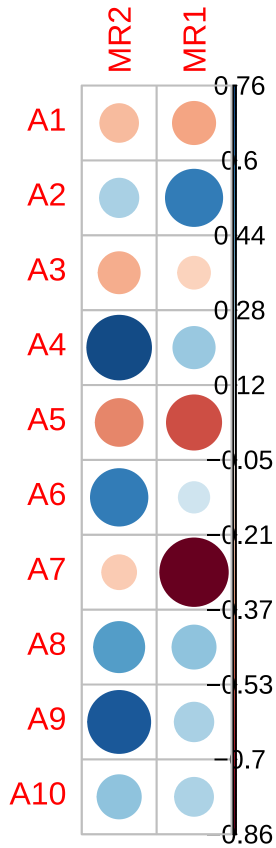

MR1
MR4
MR2
MR3
N1
N2
N3
N4
N5
N6
N7
N8
N9
N10
0.88
0.73
0.59
0.44
0.29
0.14
0
-0.15
-0.3
-0.45
-0.59



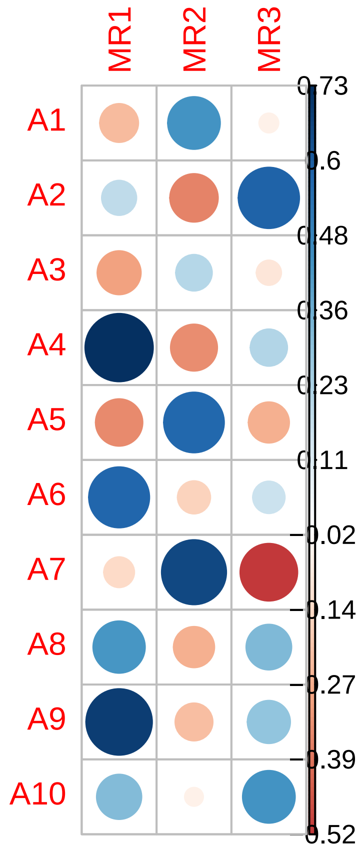
Neuroticism:

- 2 factor - MR1 - mood swings, MR2 - not coherent
- 3 factor - MR1 - mood swings, MR2 - irritability, MR3 - not coherent
- 4 factor - MR1 - mood swings, MR2 - irritability, MR3 - depression, MR4 - irritability
- 5 factor - MR1 - mood swings?, MR2 - mood swings?, MR3 - worry/stress, MR4 - depression, MR5 - upset/irritated

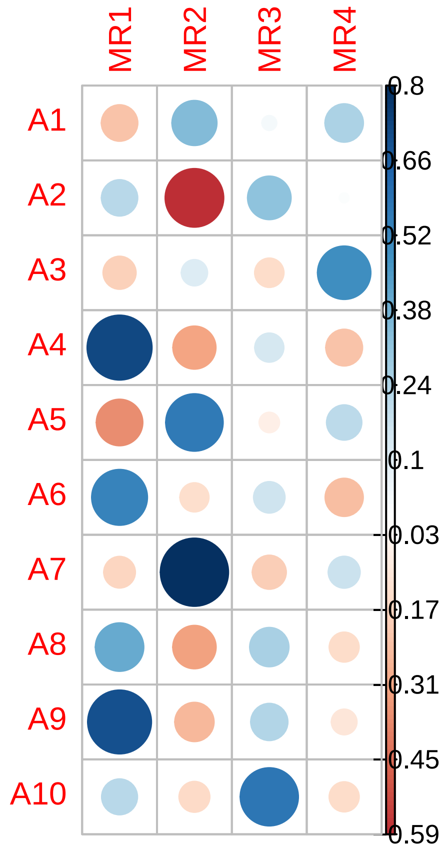
```
subfactors(select(df, A1:A10))
```

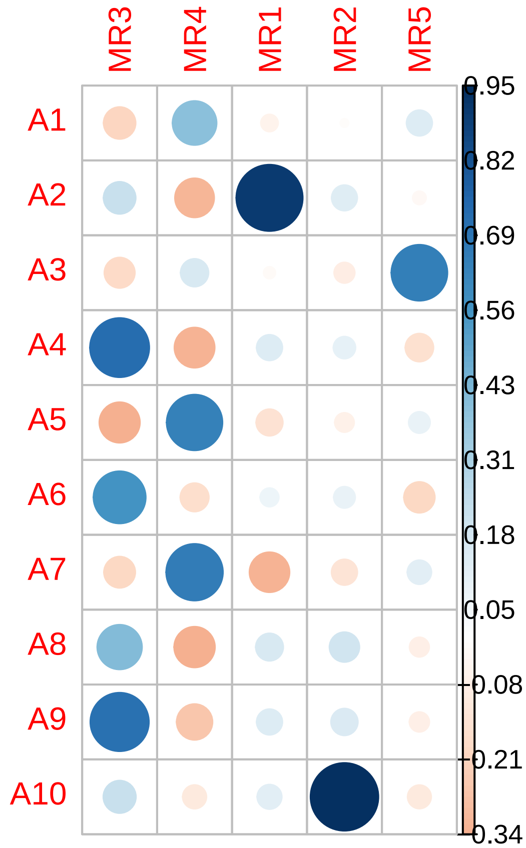


MR2
MR1
A1
A2
A3
A4
A5
A6
A7
A8
A9
A10
0.76
0.6
0.44
0.28
0.12
−0.05
−0.21
−0.37
−0.53
−0.7
−0.86



MR1
MR2
MR3
A1
A2
A3
A4
A5
A6
A7
A8
A9
A10
0.73
0.6
0.48
0.36
0.23
0.11
−0.02
−0.14
−0.27
−0.39
−0.52

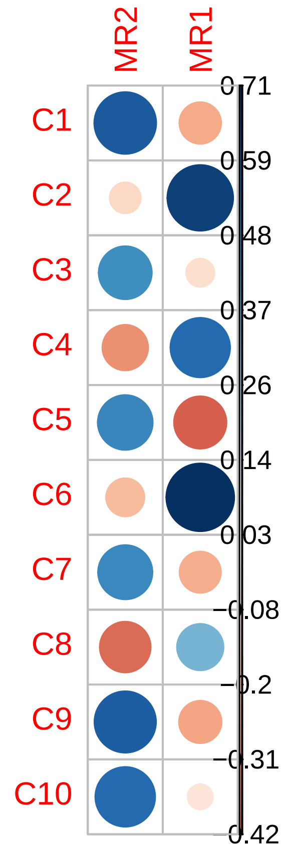

MR1
MR2
MR3
MR4
A1
A2
A3
A4
A5
A6
A7
A8
A9
A10
0.8
0.66
0.52
0.38
0.24
0.1
-0.03
-0.17
-0.31
-0.45
-0.59

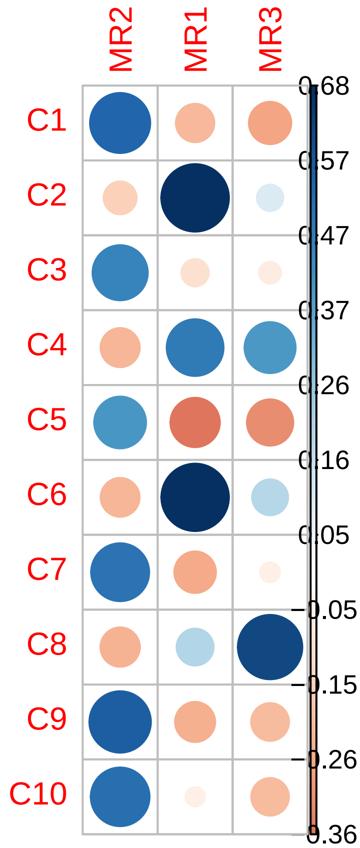


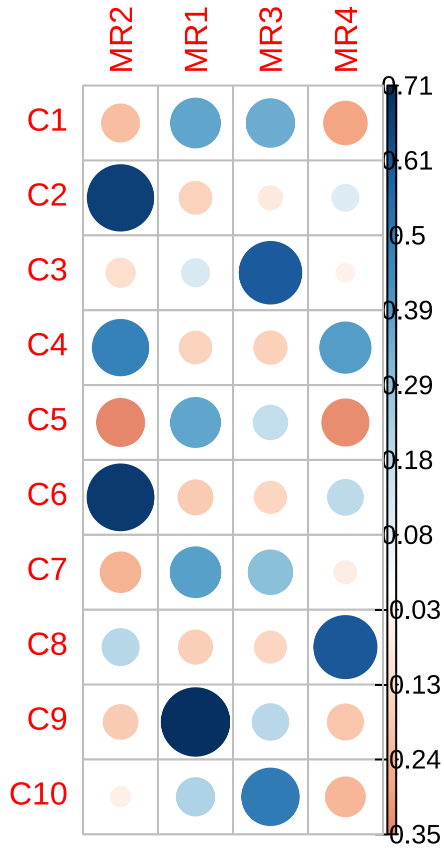
Agreeability:

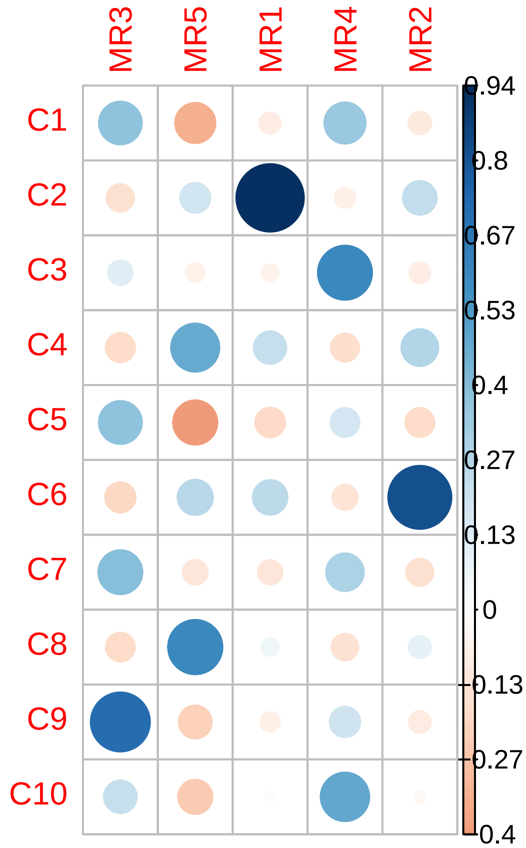
- 2 factor - MR1 - interest in others, MR2 - empathy
- 3 factor - MR1 - empathy, MR2 - interest in others, MR3 - interest in tohers?
- 4 factor - MR1- empathy, MR2 - interest in others, MR3 - make people feel at ease, MR4 - insult people
- 5 factor - MR1 - interest in people, MR2 - make people feel at ease, MR3 - empathy, MR4 - interest in people, MR5 - insult people

```
subfactors(select(df, C1:C10))
```


MR2
MR1
C1
C2
C3
C4
C5
C6
C7
C8
C9
C10
0.71
0.59
0.48
0.37
0.26
0.14
0.03
−0.08
−0.2
−0.31
−0.42


MR2
MR1
MR3
C1
C2
C3
C4
C5
C6
C7
C8
C9
C10
0.68
0.57
0.47
0.37
0.26
0.16
0.05
−0.05
−0.15
−0.26
−0.36

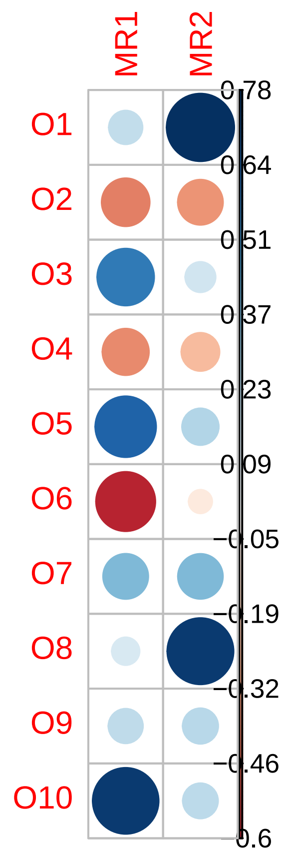

MR2
MR1
MR3
MR4
C1
C2
C3
C4
C5
C6
C7
C8
C9
C10
0.71
0.61
0.5
0.39
0.29
0.18
0.08
-0.03
-0.13
-0.24
-0.35

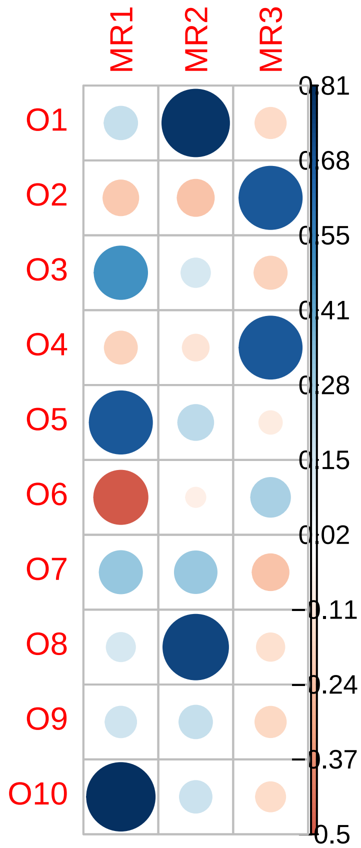


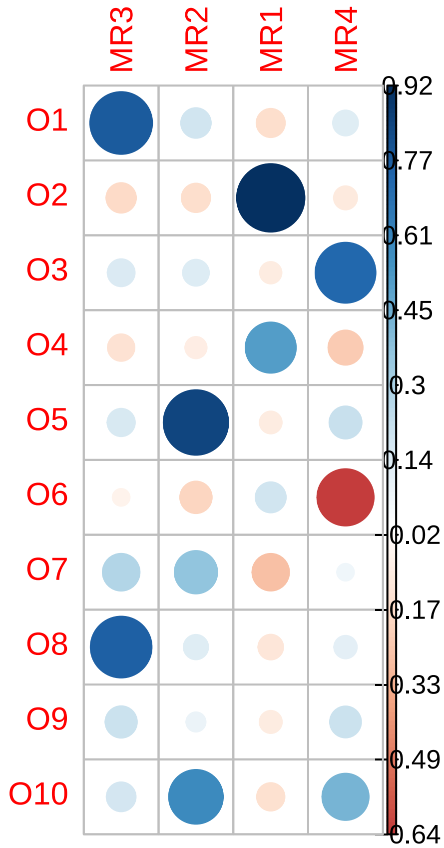
Conscientiousness:

- 2 factor - MR1 - responsibility?, MR2 - tidy/messy
- 3 factor - MR1 - tidy/messy, MR2 - responsibility, MR3 - lack of responsiblity
- 4 factor - MR1 - schedule, MR2 - messy, MR3 - detail-oriented, MR4 - lack of responsibilty
- 5 factor - MR1 - messy, MR2 - forget messy, MR3 - schedule, MR4 - detail oriented, MR5 - lack of responsibility

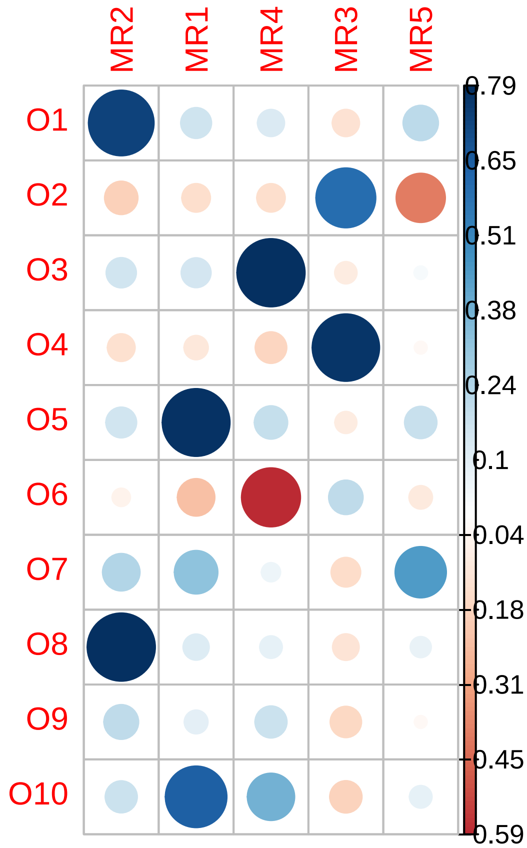
```r
subfactors(select(df, O1:O10))
```


MR1
MR2
O1
O2
O3
O4
O5
O6
O7
O8
O9
O10
0.78
0.64
0.51
0.37
0.23
0.09
−0.05
−0.19
−0.32
−0.46
−0.6


MR1
MR2
MR3
O1
O2
O3
O4
O5
O6
O7
O8
O9
O10
0.81
0.68
0.55
0.41
0.28
0.15
0.02
−0.11
−0.24
−0.37
−0.5



MR3
MR2
MR1
MR4
O1
O2
O3
O4
O5
O6
O7
O8
O9
O10
0.92
0.77
0.61
0.45
0.3
0.14
-0.02
-0.17
-0.33
-0.49
-0.64



Openness:

- 2 factor - MR1 - imagination, MR2 - vocabulary
- 3 factor - MR1 - ideas, MR2 - vocabulary, MR3 - abstract ideas
- 4 factor - MR1 - abstract ideas, MR2 - good ideas?, MR3 - vocabulary, MR4 - imagination
- 5 factor - MR1 - ideas, MR2 - vocabulary, MR3 - abstract ideas, MR4 - imagination, MR5 - learning?