Shay Lebovitz

4) *a)*

```r
library(GPArotation)

data <- read.table('/Users/shaylebovitz/R/employment.txt', header = TRUE)
employment <- data[,1:7]

(employment_fa2 <- factanal(employment, 2))
```

```
##
## Call:
## factanal(x = employment, factors = 2)
##
## Uniquenesses:
##   AGR   MAN   CON   SER   FIN   SPS    TC
## 0.005 0.882 0.793 0.364 0.619 0.329 0.192
##
## Loadings:
##     Factor1 Factor2
## AGR -0.738  -0.671
## MAN  0.344
## CON          0.451
## SER          0.791
## FIN -0.268   0.556
## SPS  0.672   0.468
## TC   0.869  -0.230
##
##                Factor1 Factor2
## SS loadings      1.955   1.861
## Proportion Var   0.279   0.266
## Cumulative Var   0.279   0.545
##
## Test of the hypothesis that 2 factors are sufficient.
## The chi square statistic is 15.8 on 8 degrees of freedom.
## The p-value is 0.0453
```

b)

```r
emp_loadings <- matrix(c(-0.738, 0.344, 0.060, 0.099, -0.268, 0.672, 0.869,
                       -0.671, -0.005, 0.451, 0.791, 0.556, 0.468, -0.230),
                       nrow = 7, ncol = 2)
t_emp_loadings <- matrix(c(-0.738, -0.671, 0.344, -0.005, 0.060, 0.451, 0.099,
                           0.791, -0.268, 0.556, 0.672, 0.468, 0.869, -0.230),
                         nrow = 2, ncol = 7)
(employment_fa2_cor <- emp_loadings %*% t_emp_loadings +
  + diag(employment_fa2$unique))
```

```
##           [,1]      [,2]      [,3]       [,4]      [,5]       [,6]       [,7]
```

```
## [1,]  0.999885 -0.250517 -0.346901 -0.6038230 -0.175292 -0.8099640 -0.4869920
## [2,] -0.250517  1.000222  0.018385  0.0301010 -0.094972  0.2288280  0.3000860
## [3,] -0.346901  0.018385  1.000055  0.3626810  0.234676  0.2513880 -0.0515900
## [4,] -0.603823  0.030101  0.362681  0.9995095  0.413264  0.4367160 -0.0958990
## [5,] -0.175292 -0.094972  0.234676  0.4132640  1.000383  0.0801120 -0.3607720
## [6,] -0.809964  0.228828  0.251388  0.4367160  0.080112  0.9991911  0.4763280
## [7,] -0.486992  0.300086 -0.051590 -0.0958990 -0.360772  0.4763280  0.9999779
```

```r
cor(employment)
```

```
##            AGR         MAN         CON         SER         FIN         SPS
## AGR  1.0000000 -0.25438889 -0.34861031 -0.60471243 -0.17575329 -0.81147553
## MAN -0.2543889  1.00000000 -0.03445846 -0.03294004 -0.27374053  0.05028408
## CON -0.3486103 -0.03445846  1.00000000  0.47308319 -0.01802316  0.07200705
## SER -0.6047124 -0.03294004  0.47308319  1.00000000  0.37928368  0.38798122
## FIN -0.1757533 -0.27374053 -0.01802316  0.37928368  1.00000000  0.16601516
## SPS -0.8114755  0.05028408  0.07200705  0.38798122  0.16601516  1.00000000
## TC  -0.4873331  0.24290323 -0.05460530 -0.08489430 -0.39132021  0.47492344
##             TC
## AGR -0.4873331
## MAN  0.2429032
## CON -0.0546053
## SER -0.0848943
## FIN -0.3913202
## SPS  0.4749234
## TC   1.0000000
```

Overall these two correlation matrices are very similar, It appears as though pretty much every element is within 20% of each other. Thus, the factor analysis model appears to capture the observed correlations between the variables.

c)

```r
print (employment_fa2$loadings, cutoff = 0.4)
```

```
##
## Loadings:
##     Factor1 Factor2
## AGR -0.738  -0.671
## MAN
## CON          0.451
## SER          0.791
## FIN          0.556
## SPS  0.672   0.468
## TC   0.869
##
##                Factor1 Factor2
## SS loadings      1.955   1.861
## Proportion Var   0.279   0.266
## Cumulative Var   0.279   0.545
```

```r
print(factanal(employment, 2, rotation = "quartimax")$loadings, cutoff = 0.4)
```

```
##
## Loadings:
##     Factor1 Factor2
## AGR -0.991
```

```
## MAN
## CON
## SER  0.659   0.449
## FIN          0.568
## SPS  0.796
## TC   0.402  -0.804
##
##                Factor1 Factor2
## SS loadings      2.463   1.353
## Proportion Var   0.352   0.193
## Cumulative Var   0.352   0.545
```

```r
print(factanal(employment, 2, rotation = "infomaxT")$loadings, cutoff=  0.4)
```

```
##
## Loadings:
##     Factor1 Factor2
## AGR -0.997
## MAN
## CON
## SER  0.585   0.542
## FIN          0.598
## SPS  0.816
## TC   0.518  -0.735
##
##                Factor1 Factor2
## SS loadings      2.473   1.343
## Proportion Var   0.353   0.192
## Cumulative Var   0.353   0.545
```

All these rotations have two rows with a non-zero loading in each column, which is not ideal. 'quartimax' and 'infomaxT' have essentially identical loading matrices. I believe the original ('varimax') loading matrix is the best, as it has 4 rows with a non-zero loading in just one column, whereas the other two have only 3 with that property.

d)

```r
(employment_fa3 <- factanal(employment, 3))
```

```
##
## Call:
## factanal(x = employment, factors = 3)
##
## Uniquenesses:
##   AGR   MAN   CON   SER   FIN   SPS    TC
## 0.005 0.820 0.414 0.322 0.230 0.255 0.293
##
## Loadings:
##     Factor1 Factor2 Factor3
## AGR -0.902          -0.422
## MAN  0.220  -0.356
## CON                  0.760
## SER  0.381   0.306   0.662
## FIN  0.216   0.847
## SPS  0.859
## TC   0.577  -0.588  -0.165
```

```
##
##                 Factor1 Factor2 Factor3
## SS loadings      2.125   1.297   1.239
## Proportion Var   0.304   0.185   0.177
## Cumulative Var   0.304   0.489   0.666
##
## Test of the hypothesis that 3 factors are sufficient.
## The chi square statistic is 6.29 on 3 degrees of freedom.
## The p-value is 0.0984
```

```
emp_loadings3 <- matrix(c(-0.902, 0.220, 0.024, 0.381, 0.216, 0.859, 0.577,
                          0.062, -0.356, -0.089, 0.306, 0.847, -0.004, -0.588,
                          -0.422, 0.070, 0.760, 0.662, 0.078, 0.087, -0.165),
                        nrow = 7, ncol = 3)
t_emp_loadings3 <- matrix(c(-0.902, 0.062, -0.422, -0.220, -0.356, 0.070,
                            0.024, -0.089, 0.760, 0.381, 0.306, 0.662,
                            0.216, 0.847, 0.078, 0.859, -0.004, 0.087,
                            0.577, -0.588, -0.165), nrow = 3, ncol = 7)
(employment_fa3_cor <- emp_loadings3 %*% t_emp_loadings3 +
    + diag(employment_fa3$unique))
```

```
##           [,1]       [,2]      [,3]       [,4]       [,5]      [,6]       [,7]
## [1,]  1.000532  0.1468280 -0.347886 -0.6040540 -0.1752340 -0.811780 -0.4872800
## [2,] -0.250052  0.9028156  0.090164  0.0212240 -0.2485520  0.196494  0.3247180
## [3,] -0.347886  0.0796040  1.000435  0.4850300 -0.0109190  0.087092 -0.0592200
## [4,] -0.604054 -0.1464160  0.485030  0.9995255  0.3931140  0.383649 -0.0693210
## [5,] -0.175234 -0.3435920 -0.010919  0.3931140  0.9998512  0.188942 -0.3862740
## [6,] -0.811780 -0.1814660  0.087092  0.3836490  0.1889420  1.000679  0.4836400
## [7,] -0.487280  0.0708380 -0.059220 -0.0693210 -0.3862740  0.483640  0.9990704
```

```
cor(employment)
```

```
##            AGR         MAN         CON         SER         FIN         SPS
## AGR  1.0000000 -0.25438889 -0.34861031 -0.60471243 -0.17575329 -0.81147553
## MAN -0.2543889  1.00000000 -0.03445846 -0.03294004 -0.27374053  0.05028408
## CON -0.3486103 -0.03445846  1.00000000  0.47308319 -0.01802316  0.07200705
## SER -0.6047124 -0.03294004  0.47308319  1.00000000  0.37928368  0.38798122
## FIN -0.1757533 -0.27374053 -0.01802316  0.37928368  1.00000000  0.16601516
## SPS -0.8114755  0.05028408  0.07200705  0.38798122  0.16601516  1.00000000
## TC  -0.4873331  0.24290323 -0.05460530 -0.08489430 -0.39132021  0.47492344
##             TC
## AGR -0.4873331
## MAN  0.2429032
## CON -0.0546053
## SER -0.0848943
## FIN -0.3913202
## SPS  0.4749234
## TC   1.0000000
```

Again, we see that these two correlation matrices are extremely similar.

```
print (employment_fa3$loadings, cutoff = 0.4)
```

```
##
## Loadings:
##     Factor1 Factor2 Factor3
## AGR -0.902          -0.422
```

```
## MAN
## CON                   0.760
## SER                   0.662
## FIN           0.847
## SPS   0.859
## TC    0.577  -0.588
##
##                Factor1 Factor2 Factor3
## SS loadings      2.125   1.297   1.239
## Proportion Var   0.304   0.185   0.177
## Cumulative Var   0.304   0.489   0.666
```

```
print(factanal(employment, 3, rotation = "quartimax")$loadings, cutoff = 0.4)
```

```
##
## Loadings:
##     Factor1 Factor2 Factor3
## AGR -0.972
## MAN
## CON                   0.740
## SER  0.481            0.545
## FIN          0.865
## SPS  0.854
## TC   0.583  -0.551
##
##                Factor1 Factor2 Factor3
## SS loadings      2.373   1.318   0.970
## Proportion Var   0.339   0.188   0.139
## Cumulative Var   0.339   0.527   0.666
```

```
print(factanal(employment, 3, rotation = "infomaxT")$loadings, cutoff=  0.4)
```

```
##
## Loadings:
##     Factor1 Factor2 Factor3
## AGR -0.896          -0.436
## MAN
## CON                   0.765
## SER                   0.640
## FIN          0.863
## SPS  0.855
## TC   0.617  -0.560
##
##                Factor1 Factor2 Factor3
## SS loadings      2.122   1.322   1.216
## Proportion Var   0.303   0.189   0.174
## Cumulative Var   0.303   0.492   0.666
```

Here, we see that all three rotations yield loadings with 4 rows with just one non-zero loading. Similarly, all three rotations contain at least m = 3 zeros in each column, and all three have two rows with two non-zero factors. From this perspective, I don't think that one rotation is clearly better than the others, they all seem very even.

e)

```
employment_fa1 <- factanal(employment, 1)
(phi <- 1 - c(sum(employment_fa1$unique), sum(employment_fa2$unique),
```

```r
              sum(employment_fa3$unique))/7)
```

```
## [1] 0.3532472 0.5451621 0.6657872
```

Based on these results, I would chose a model with three factors and use the 'varimax' rotation. Choosing m = 2 yields a Phi value of 0.55, which is too small. Even having m = 3 (Phi = 0.67) is a little small. As discussed earlier, there doesn't seemt to be much of a difference between the loading rotations for a three factor model. For a three factor model with the 'varimax' rotation, we see that Factor 1 greatly predicts AGR, and SPS, as well as TC to a lesser extent. Factor 2 greatly predicts FIN, and TC again to a lesser extent. Factor 3 has a small predictive effect on AGR, and decent effect on SER and FIN. No factor significantly predicts MAN. Off the top of my head, I cannot think of a realistic representation of these factors, as it is unclear what agriculture, social services, and transport/communications would have in common, or what factor would determine finance and transport but is uncorrelated with Factor 1.

5) *a)*

```r
apps <- read.table('/Users/shaylebovitz/R/applicants.txt', header = TRUE)
#m <= 3
apps_fa1 <- factanal(apps, 1)
apps_fa2 <- factanal(apps, 2)
apps_fa3 <- factanal(apps, 3)
(phi <- 1 - c(sum(apps_fa1$unique), sum(apps_fa2$unique), sum(apps_fa3$unique))/7)
```

```
## [1] 0.6432686 0.7813120 0.8333725
```

```r
phi[-1] - phi[-3]
```

```
## [1] 0.13804337 0.05206048
```

```r
apps_fa1$PVAL
```

```
##   objective
## 0.000105917
```

```r
apps_fa2$PVAL
```

```
##  objective
## 0.03630386
```

```r
apps_fa3$PVAL
```

```
## objective
## 0.3521931
```

Based on this analysis, I would say 2 factors is appropriate for this model, as they explain 78% of the variance, and it is also the last statistially significant P-value.

b)

```r
print(apps_fa2$loadings, cutoff = 0.4)
```

```
##
## Loadings:
##     Factor1 Factor2
## Lik         0.667
## Con 0.868
## Luc 0.824
## Sal 0.902
## Dri 0.756   0.409
## Amb 0.856
```

```
## Ent         0.946
##
##                Factor1 Factor2
## SS loadings      3.693   1.776
## Proportion Var   0.528   0.254
## Cumulative Var   0.528   0.781
```

```
apps_fa2$uniquenesses
```

```
##       Lik       Con       Luc       Sal       Dri       Amb       Ent
## 0.5117955 0.2091670 0.2571750 0.1105795 0.2618704 0.1752284 0.0050000
```

```
1-apps_fa2$uniquenesses
```

```
##       Lik       Con       Luc       Sal       Dri       Amb       Ent
## 0.4882045 0.7908330 0.7428250 0.8894205 0.7381296 0.8247716 0.9950000
```

For a two factor model, we see that `Ent` has an extremely high communality, which suggests that its variance is highly explained by the factors. Most others have a relatively high communality, with `Con`, `Luc`, `Sal`, `Dri`, and `Amb` all having communalities > 0.74. `Lik` has a lower communality of 0.49, suggesting the factor model isn't as applicable to it.

c)

```
print (apps_fa2$loadings, cutoff = 0.4)
```

```
##
## Loadings:
##     Factor1 Factor2
## Lik         0.667
## Con 0.868
## Luc 0.824
## Sal 0.902
## Dri 0.756   0.409
## Amb 0.856
## Ent         0.946
##
##                Factor1 Factor2
## SS loadings      3.693   1.776
## Proportion Var   0.528   0.254
## Cumulative Var   0.528   0.781
```

```
print(factanal(apps, 2, rotation = "quartimax")$loadings, cutoff = 0.4)
```

```
##
## Loadings:
##     Factor1 Factor2
## Lik          0.597
## Con  0.889
## Luc  0.861
## Sal  0.942
## Dri  0.832
## Amb  0.904
## Ent  0.536   0.841
##
##                Factor1 Factor2
## SS loadings      4.346   1.123
## Proportion Var   0.621   0.160
```

```
## Cumulative Var   0.621   0.781
```

```
print(factanal(apps, 2, rotation = "infomaxT")$loadings, cutoff=  0.4)
```

```
##
## Loadings:
##     Factor1 Factor2
## Lik         0.634
## Con 0.886
## Luc 0.850
## Sal 0.930
## Dri 0.803
## Amb 0.888
## Ent 0.438   0.896
##
##                Factor1 Factor2
## SS loadings      4.083   1.386
## Proportion Var   0.583   0.198
## Cumulative Var   0.583   0.781
```

Here, we see that 'quartimax' and 'infomaxT' are essentially identical in terms of their loading matrices. They both have one non-zero loading element for every row except `Ent`. The 'varimax' rotation is similar: it has one non-zero loading element for every row except `Dri`. I believe the original 'varimax' rotation is the best because it is the only one with both columns containing at least 2 zeros.

d)

```
print(factanal(apps, 2, rotation = "oblimin")$loadings, cutoff = 0.4)
```

```
##
## Loadings:
##     Factor1 Factor2
## Lik          0.702
## Con  0.939
## Luc  0.865
## Sal  0.949
## Dri  0.728
## Amb  0.884
## Ent          0.989
##
##                Factor1 Factor2
## SS loadings      3.842   1.521
## Proportion Var   0.549   0.217
## Cumulative Var   0.549   0.766
```

```
print(factanal(apps, 2, rotation = "promax")$loadings, cutoff=  0.4)
```

```
##
## Loadings:
##     Factor1 Factor2
## Lik          0.710
## Con  0.942
## Luc  0.867
## Sal  0.951
## Dri  0.726
## Amb  0.885
## Ent          1.000
```

```
##
##                 Factor1 Factor2
## SS loadings       3.854   1.554
## Proportion Var    0.551   0.222
## Cumulative Var    0.551   0.773
```

```
print(factanal(apps, 2, rotation = "infomaxQ")$loadings, cutoff = 0.4)
```

```
##
## Loadings:
##     Factor1 Factor2
## Lik          0.731
## Con  0.930
## Luc  0.852
## Sal  0.934
## Dri  0.702
## Amb  0.867
## Ent          1.029
##
##                 Factor1 Factor2
## SS loadings       3.715   1.654
## Proportion Var    0.531   0.236
## Cumulative Var    0.531   0.767
```

All three of these rotated loading matrices are essentially the same. Factor 1 has non-zero loadings in `Con`, `Luc`, `Sal`, `Dri`, and `Amb`, whereas Factor 2 has non-zero loadings in `Lik` and `Ent`. All of the loading elements are of similar magnitude as well. Thus there is no clear optimal choice.

e) It looks like the oblique rotations give a better loading matrix than the orthogonal rotations. I will randomly choose 'oblimin', as all the oblique rotations are essentialy the same.

```
factor1 <- factanal(apps, 2, rotation = "oblimin")$loadings[,1]
factor2 <- factanal(apps, 2, rotation = "oblimin")$loadings[,2]
cor(factor1, factor2)
```

```
## [1] -0.973436
```

Factor 1 relates to self-confidence, lucidity, salesmanship, drive, and ambition, whereas Factor 2 relates to likeability and enthusiasm. Because it is an oblique rotation, Factor 2 is not necessarily uncorrelated with Factor 1. We see that Factor 1 and Factor 2 are highly negatively correlated, with a correlation of -0.97. Thus, whatever underlying factor increases confidence, lucidity, salesmanship, drive, and ambition yields decreases in likeability and enthusiasm.

6) *a)*

```
anx <- read.csv('/Users/shaylebovitz/R/anxiety.csv', header = TRUE)
anx <- as.matrix(anx[,-1])
(anx_fa2 <- factanal(factors = 2, covmat = list(cov = anx, n.obs = 335)))
```

```
##
## Call:
## factanal(factors = 2, covmat = list(cov = anx, n.obs = 335))
##
## Uniquenesses:
##    x1    x2    x3    x4    x5    x6    x7    x8    x9   x10   x11   x12   x13
## 0.607 0.594 0.644 0.565 0.483 0.721 0.461 0.488 0.520 0.746 0.431 0.564 0.650
##   x14   x15   x16   x17   x18   x19   x20
## 0.559 0.357 0.388 0.471 0.539 0.569 0.562
```

```
##
## Loadings:
##       Factor1 Factor2
##  [1,] 0.542   0.315
##  [2,] 0.592   0.235
##  [3,] 0.288   0.522
##  [4,] 0.471   0.462
##  [5,] 0.115   0.710
##  [6,] 0.276   0.450
##  [7,] 0.375   0.631
##  [8,] 0.665   0.266
##  [9,] 0.657   0.219
## [10,] 0.454   0.219
## [11,] 0.697   0.289
## [12,] 0.584   0.308
## [13,] 0.509   0.302
## [14,] 0.408   0.524
## [15,] 0.686   0.415
## [16,] 0.733   0.275
## [17,] 0.315   0.655
## [18,] 0.617   0.284
## [19,] 0.565   0.335
## [20,] 0.523   0.405
##
##                Factor1 Factor2
## SS loadings      5.592   3.490
## Proportion Var   0.280   0.175
## Cumulative Var   0.280   0.454
##
## Test of the hypothesis that 2 factors are sufficient.
## The chi square statistic is 287.5 on 151 degrees of freedom.
## The p-value is 1.47e-10
```

```r
print(anx_fa2$loadings, cutoff = 0.4)
```

```
##
## Loadings:
##       Factor1 Factor2
##  [1,] 0.542
##  [2,] 0.592
##  [3,]         0.522
##  [4,] 0.471   0.462
##  [5,]         0.710
##  [6,]         0.450
##  [7,]         0.631
##  [8,] 0.665
##  [9,] 0.657
## [10,] 0.454
## [11,] 0.697
## [12,] 0.584
## [13,] 0.509
## [14,] 0.408   0.524
## [15,] 0.686   0.415
## [16,] 0.733
## [17,]         0.655
```

```
## [18,] 0.617
## [19,] 0.565
## [20,] 0.523   0.405
##
##                 Factor1 Factor2
## SS loadings       5.592   3.490
## Proportion Var    0.280   0.175
## Cumulative Var    0.280   0.454
```

Only 4 out of the 20 rows of the loading matrix have a non-zero entry in each column, and thus the vast majority only have one non-zero loading. So it's off to a good start.

```r
anx_fa1 <- factanal(factors = 1, covmat = list(cov = anx, n.obs = 335))
anx_fa3 <- factanal(factors = 3, covmat = list(cov = anx, n.obs = 335))
(phi <- 1 - c(sum(anx_fa1$unique), sum(anx_fa2$unique), sum(anx_fa3$unique))/20)
```

```
## [1] 0.4103534 0.4540977 0.4771117
```

Here, we see that two factors only explain roughly 45% of the correlation matrix, which is not adequate. This is enough information to declare that 2 factors is not appropriate for this model.

b)

```r
anx_fa2_quart <- factanal(factors = 2,
                          covmat = list(cov = anx, n.obs = 335),
                          rotation = "quartimax")
anx_fa2_infoT <- factanal(factors = 2,
                          covmat = list(cov = anx, n.obs = 335),
                          rotation = "infomaxT")
anx_fa2_obl <- factanal(factors = 2,
                          covmat = list(cov = anx, n.obs = 335),
                          rotation = "oblimin")
anx_fa2_pro <- factanal(factors = 2,
                          covmat = list(cov = anx, n.obs = 335),
                          rotation = "promax")
anx_fa2_infoQ <- factanal(factors = 2,
                          covmat = list(cov = anx, n.obs = 335),
                          rotation = "infomaxQ")
print(anx_fa2$loadings, cutoff = 0.4) # overlap, 0 empty
```

```
##
## Loadings:
##       Factor1 Factor2
##  [1,] 0.542
##  [2,] 0.592
##  [3,]         0.522
##  [4,] 0.471   0.462
##  [5,]         0.710
##  [6,]         0.450
##  [7,]         0.631
##  [8,] 0.665
##  [9,] 0.657
## [10,] 0.454
## [11,] 0.697
## [12,] 0.584
## [13,] 0.509
## [14,] 0.408   0.524
```

```
## [15,] 0.686   0.415
## [16,] 0.733
## [17,]         0.655
## [18,] 0.617
## [19,] 0.565
## [20,] 0.523   0.405
##
##                Factor1 Factor2
## SS loadings      5.592   3.490
## Proportion Var   0.280   0.175
## Cumulative Var   0.280   0.454
```

```
print(anx_fa2_quart$loadings, cutoff = 0.4) # 2 overlap, 0 empty
```

```
##
## Loadings:
##       Factor1 Factor2
##  [1,]  0.627
##  [2,]  0.629
##  [3,]  0.515
##  [4,]  0.641
##  [5,]  0.462   0.551
##  [6,]  0.467
##  [7,]  0.645
##  [8,]  0.707
##  [9,]  0.677
## [10,]  0.502
## [11,]  0.746
## [12,]  0.659
## [13,]  0.591
## [14,]  0.619
## [15,]  0.802
## [16,]  0.770
## [17,]  0.606   0.402
## [18,]  0.675
## [19,]  0.657
## [20,]  0.657
##
##                Factor1 Factor2
## SS loadings      8.170   0.912
## Proportion Var   0.409   0.046
## Cumulative Var   0.409   0.454
```

```
print(anx_fa2_infoT$loadings, cutoff = 0.4) # 3 overlap, 1 empty
```

```
##
## Loadings:
##       Factor1 Factor2
##  [1,] 0.596
##  [2,] 0.629
##  [3,]         0.451
##  [4,] 0.557
##  [5,]         0.670
##  [6,]
##  [7,] 0.498   0.539
```

```
##  [8,] 0.706
##  [9,] 0.689
## [10,] 0.490
## [11,] 0.742
## [12,] 0.635
## [13,] 0.560
## [14,] 0.509   0.427
## [15,] 0.758
## [16,] 0.774
## [17,] 0.445   0.575
## [18,] 0.662
## [19,] 0.622
## [20,] 0.596
##
##                Factor1 Factor2
## SS loadings      6.951   2.131
## Proportion Var   0.348   0.107
## Cumulative Var   0.348   0.454
```

```
print(anx_fa2_obl$loadings, cutoff = 0.4) # no overlap, 1 empty
```

```
##
## Loadings:
##       Factor1 Factor2
##  [1,]  0.565
##  [2,]  0.660
##  [3,]          0.478
##  [4,]  0.417
##  [5,]          0.799
##  [6,]
##  [7,]          0.563
##  [8,]  0.740
##  [9,]  0.749
## [10,]  0.491
## [11,]  0.771
## [12,]  0.620
## [13,]  0.527
## [14,]          0.415
## [15,]  0.708
## [16,]  0.823
## [17,]          0.625
## [18,]  0.672
## [19,]  0.586
## [20,]  0.505
##
##                Factor1 Factor2
## SS loadings      6.011   2.104
## Proportion Var   0.301   0.105
## Cumulative Var   0.301   0.406
```

```
print(anx_fa2_pro$loadings, cutoff = 0.4) # no overlap, 1 empty
```

```
##
## Loadings:
##       Factor1 Factor2
```

```
##  [1,]  0.551
##  [2,]  0.671
##  [3,]          0.544
##  [4,]
##  [5,]          0.911
##  [6,]          0.451
##  [7,]          0.639
##  [8,]  0.751
##  [9,]  0.771
## [10,]  0.489
## [11,]  0.781
## [12,]  0.612
## [13,]  0.512
## [14,]          0.470
## [15,]  0.686
## [16,]  0.840
## [17,]          0.711
## [18,]  0.673
## [19,]  0.569
## [20,]  0.467
##
##                Factor1 Factor2
## SS loadings      5.865   2.719
## Proportion Var   0.293   0.136
## Cumulative Var   0.293   0.429
```

```
print(anx_fa2_infoQ$loadings, cutoff = 0.4) # 1 overlap, no empty
```

```
##
## Loadings:
##       Factor1 Factor2
##  [1,]  0.516
##  [2,]  0.653
##  [3,]          0.612
##  [4,]          0.403
##  [5,] -0.421   1.007
##  [6,]          0.509
##  [7,]          0.721
##  [8,]  0.732
##  [9,]  0.760
## [10,]  0.469
## [11,]  0.758
## [12,]  0.580
## [13,]  0.478
## [14,]          0.537
## [15,]  0.638
## [16,]  0.821
## [17,]          0.797
## [18,]  0.647
## [19,]  0.532
## [20,]  0.414
##
##                Factor1 Factor2
## SS loadings      5.415   3.454
## Proportion Var   0.271   0.173
```

```
## Cumulative Var   0.271   0.443
```

Based on the amount of overlap (non-zero element in both Factors) and empty (both factors have zero loading), I would choose either 'oblimin' or 'promax'. I'll go with 'oblimin'.

c)

```r
print(anx_fa2_obl$loadings, cutoff = 0.4)
```

```
##
## Loadings:
##       Factor1 Factor2
##  [1,]  0.565
##  [2,]  0.660
##  [3,]          0.478
##  [4,]  0.417
##  [5,]          0.799
##  [6,]
##  [7,]          0.563
##  [8,]  0.740
##  [9,]  0.749
## [10,]  0.491
## [11,]  0.771
## [12,]  0.620
## [13,]  0.527
## [14,]          0.415
## [15,]  0.708
## [16,]  0.823
## [17,]          0.625
## [18,]  0.672
## [19,]  0.586
## [20,]  0.505
##
##                Factor1 Factor2
## SS loadings      6.011   2.104
## Proportion Var   0.301   0.105
## Cumulative Var   0.301   0.406
```

Firstly, it is important to note that because 'oblimin' is an oblique rotation, the factors are likely correlated. Factor 1 is a decent measure or x1, x2, x4, x8, x9, x10, x11, x12, x13, x15, x18, x19, and x20, while Factor 2 is a measure of x3, x5, x7, x14, and x17. Neither factor explains x6 well. Since all the variables seem very related (all related to anxiety during test-taking), I cannot think of what these underlying factors might be.