BACS - HW14

109006241, helped by 109006217

Let's reconsider the security questionnaire from last week, where consumers were asked security related questions about one of the e-commerce websites they had recently used.

```
data = read_excel("security_questions.xlsx", sheet="data")
```

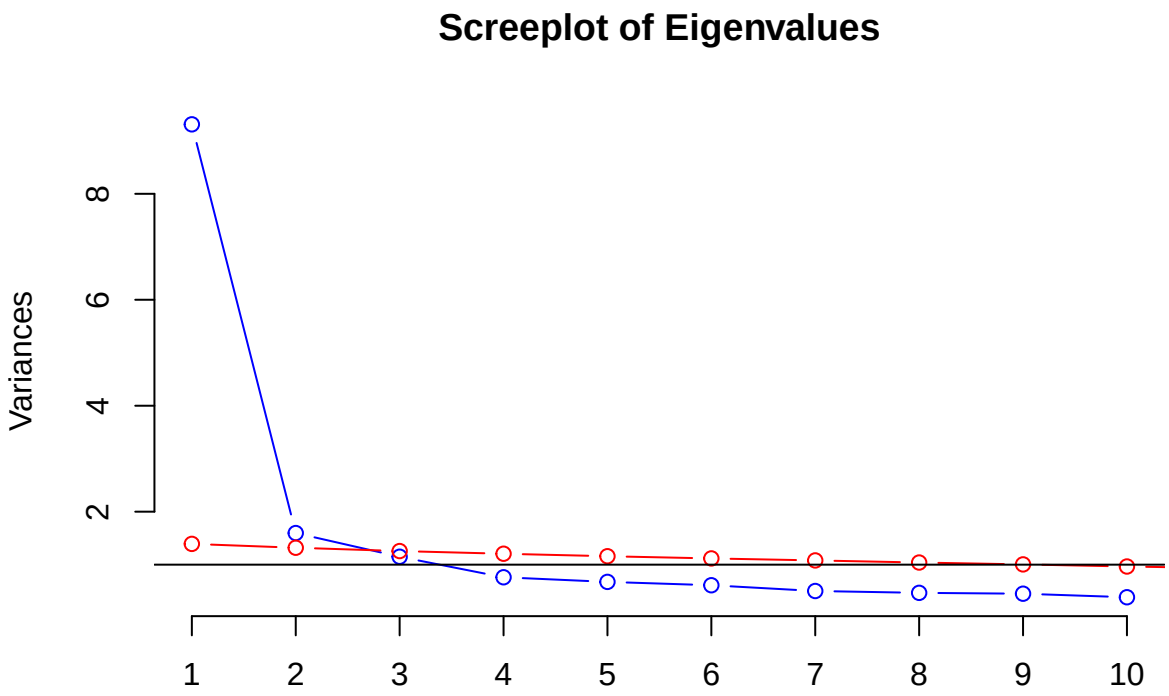
Question 1) Earlier, we examined a dataset from a security survey sent to customers of e-commerce websites. However, we only used the eigenvalue > 1 criteria and the screeplot "elbow" rule to find a suitable number of components. Let's perform a parallel analysis as well this week:

a. Show a single visualization with scree plot of data, scree plot of simulated noise (use average eigenvalues of >= 100 noise samples), and a horizontal line showing the eigenvalue = 1 cutoff.

```
data_pca = prcomp(data, scale.=T)

sim_noise_ev = function(n, p) {
  noise = data.frame(replicate(p, rnorm(n)))
  eigen(cor(noise))$values
}
evalues_noise = replicate(100, sim_noise_ev(nrow(data), ncol(data)))
evalues_mean = apply(evalues_noise, 1, mean)

screeplot(data_pca, type="lines", col="blue", main="Screeplot of Eigenvalues")
lines(evalues_mean, type="b", col="red")
abline(h=1)
```



b. How many dimensions would you retain if we used Parallel Analysis?

3 dimensions.

Question 2) Earlier, we treated the underlying dimensions of the security dataset as composites and examined their eigenvectors (weights). Now, let's treat them as factors and examine factor loadings (use the principal() method from the psych package)

a. Looking at the loadings of the first 3 principal components, to which components does each item seem to best belong?

```
data_principal = principal(data, nfactor=3, rotate="none", scores=TRUE)
data_principal$loadings
```

```
##
## Loadings:
##     PC1    PC2    PC3
## Q1   0.817 -0.139
```

```
## Q2   0.673
## Q3   0.766
## Q4   0.623  0.643  0.108
## Q5   0.690        -0.542
## Q6   0.683 -0.105  0.207
## Q7   0.657 -0.318  0.324
## Q8   0.786        -0.343
## Q9   0.723 -0.232  0.204
## Q10  0.686        -0.533
## Q11  0.753 -0.261  0.173
## Q12  0.630  0.638  0.122
## Q13  0.712
## Q14  0.811         0.157
## Q15  0.704        -0.333
## Q16  0.758 -0.203  0.183
## Q17  0.618  0.664  0.110
## Q18  0.807 -0.114
##
##                  PC1   PC2   PC3
## SS loadings    9.311 1.596 1.150
## Proportion Var 0.517 0.089 0.064
## Cumulative Var 0.517 0.606 0.670
```

```
first3_loadings = data_principal$loadings[, 1:3]
first3_loadings = round(first3_loadings, 3)
first3_loadings
```

```
##       PC1    PC2    PC3
## Q1  0.817 -0.139 -0.002
## Q2  0.673 -0.014  0.089
## Q3  0.766 -0.033  0.090
## Q4  0.623  0.643  0.108
## Q5  0.690 -0.031 -0.542
## Q6  0.683 -0.105  0.207
## Q7  0.657 -0.318  0.324
## Q8  0.786  0.042 -0.343
## Q9  0.723 -0.232  0.204
## Q10 0.686 -0.099 -0.533
## Q11 0.753 -0.261  0.173
## Q12 0.630  0.638  0.122
## Q13 0.712 -0.065  0.084
## Q14 0.811 -0.100  0.157
## Q15 0.704  0.011 -0.333
## Q16 0.758 -0.203  0.183
## Q17 0.618  0.664  0.110
## Q18 0.807 -0.114 -0.065
```

```
PC1 = c(); PC2 = c(); PC3 = c()
for(i in 1:nrow(first3_loadings)) {
  row = first3_loadings[i,]
  rowname = rownames(first3_loadings)[i]
  if(row[1] == max(row)) {
    PC1 = append(PC1, rowname)
```

```
  } else if(row[2] == max(row)) {
    PC2 = append(PC2, rowname)
  } else if(row[3] == max(row)) {
    PC3 = append(PC3, rowname)
  }
}
cat(" PC1:", paste(PC1, collapse=", "), "\n",
     "PC2:", paste(PC2, collapse=", "), "\n",
     "PC3:", paste(PC3, collapse=", "))
```

```
##  PC1: Q1, Q2, Q3, Q5, Q6, Q7, Q8, Q9, Q10, Q11, Q13, Q14, Q15, Q16, Q18
##  PC2: Q4, Q12, Q17
##  PC3:
```

b. How much of the total variance of the security dataset do the first 3 PCs capture?

```
# Total Variance Explained
sum(data_principal$values[1:3])
```

```
## [1] 12.05684
```

```
# Proportions
summary(data_pca)$importance[2, 1:3]
```

```
##     PC1     PC2     PC3
## 0.51728 0.08869 0.06386
```

c. Looking at commonality and uniqueness, which items are less than adequately explained by the first 3 principal components?

```
data_principal$communality
```

```
##        Q1        Q2        Q3        Q4        Q5        Q6        Q7        Q8
## 0.6869041 0.4605433 0.5951359 0.8138147 0.7713420 0.5201104 0.6371369 0.7375512
##        Q9       Q10       Q11       Q12       Q13       Q14       Q15       Q16
## 0.6178667 0.7642903 0.6648554 0.8185557 0.5181043 0.6930021 0.6063756 0.6485852
##       Q17       Q18
## 0.8347032 0.6679663
```

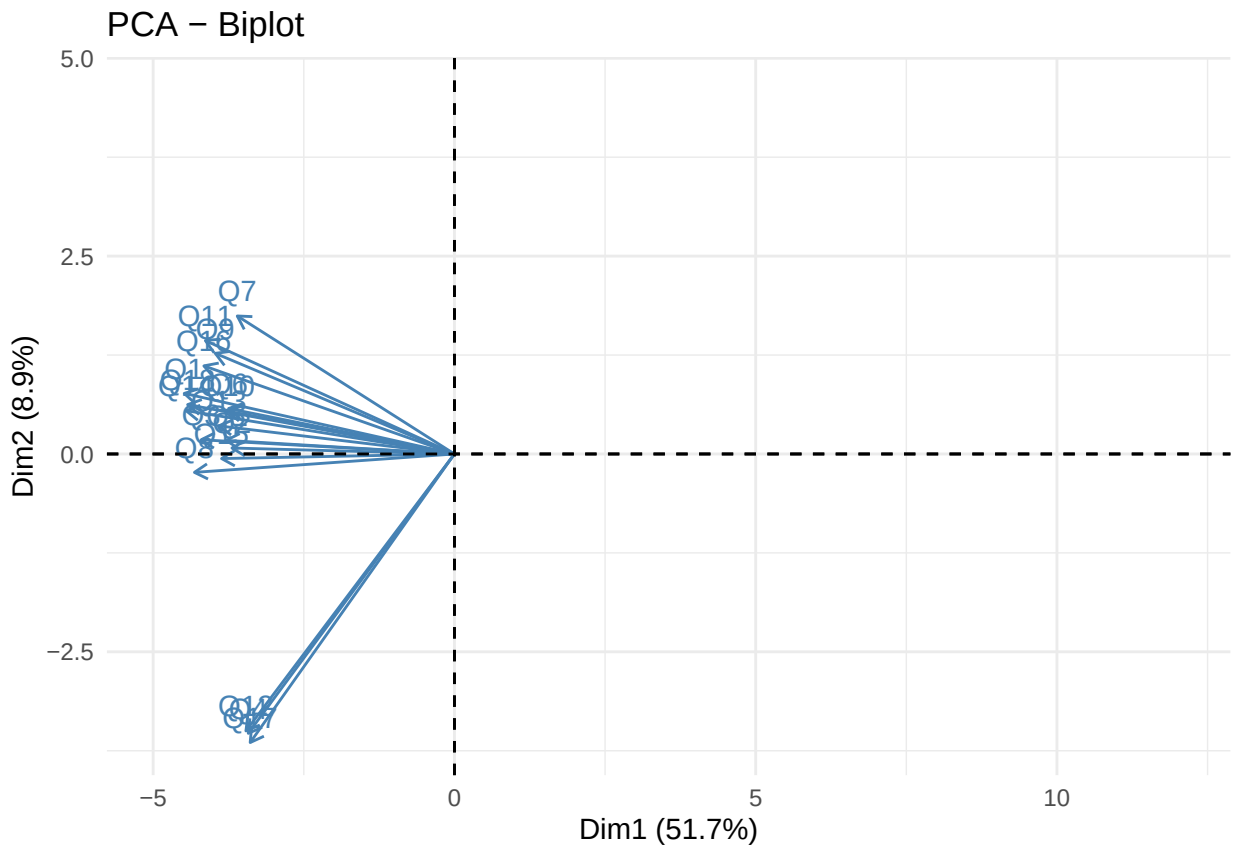
```
data_principal$uniquenesses
```

```
##        Q1        Q2        Q3        Q4        Q5        Q6        Q7        Q8
## 0.3130959 0.5394567 0.4048641 0.1861853 0.2286580 0.4798896 0.3628631 0.2624488
##        Q9       Q10       Q11       Q12       Q13       Q14       Q15       Q16
## 0.3821333 0.2357097 0.3351446 0.1814443 0.4818957 0.3069979 0.3936244 0.3514148
##       Q17       Q18
## 0.1652968 0.3320337
```

I used a threshold of uniqueness > 0.5 to determine whether items are less than adequately explained by the first three principal components or not. Using this threshold, we consider **Q2** as an item that is less than adequately explained by the first 3 principal components, because its uniqueness $= 0.54$, which is bigger than 0.5.

d. How many measurement items share similar loadings between 2 or more components?

```
fviz_pca_biplot(data_pca, invisible ="ind") + theme_minimal()
```



3: Q4, Q12, and Q17

e. Can you interpret a 'meaning' behind the first principal component from the items that load best upon it? (see the wording of the questions of those items)

It's **difficult** to interpret the meanings by using principle components (according to the lecture handouts).

Question 3) To improve interpretability of loadings, let's rotate our principal component axes using the varimax technique to get rotated components (extract and rotate only three principal components)

```
data_principal2 = principal(data, nfactor=3, rotate="varimax", scores=TRUE)
data_principal$loadings
```

```
##
## Loadings:
##     PC1    PC2    PC3
## Q1   0.817 -0.139
## Q2   0.673
## Q3   0.766
## Q4   0.623  0.643  0.108
## Q5   0.690        -0.542
## Q6   0.683 -0.105  0.207
## Q7   0.657 -0.318  0.324
## Q8   0.786        -0.343
## Q9   0.723 -0.232  0.204
## Q10  0.686        -0.533
## Q11  0.753 -0.261  0.173
## Q12  0.630  0.638  0.122
## Q13  0.712
## Q14  0.811         0.157
## Q15  0.704        -0.333
## Q16  0.758 -0.203  0.183
## Q17  0.618  0.664  0.110
## Q18  0.807 -0.114
##
##                  PC1   PC2   PC3
## SS loadings    9.311 1.596 1.150
## Proportion Var 0.517 0.089 0.064
## Cumulative Var 0.517 0.606 0.670
```

```
data_principal2$loadings
```

```
##
## Loadings:
##     RC1   RC3   RC2
## Q1  0.660 0.450 0.221
## Q2  0.544 0.286 0.288
## Q3  0.621 0.337 0.311
## Q4  0.218 0.193 0.854
## Q5  0.244 0.828 0.162
## Q6  0.652 0.199 0.234
## Q7  0.790 0.103
## Q8  0.382 0.706 0.305
## Q9  0.738 0.234 0.138
## Q10 0.277 0.823 0.102
```

```
## Q11 0.757 0.278 0.118
## Q12 0.233 0.186 0.854
## Q13 0.593 0.315 0.259
## Q14 0.719 0.310 0.283
## Q15 0.342 0.656 0.244
## Q16 0.740 0.267 0.174
## Q17 0.205 0.187 0.870
## Q18 0.609 0.495 0.227
##
##                   RC1   RC3   RC2
## SS loadings     5.613 3.490 2.954
## Proportion Var  0.312 0.194 0.164
## Cumulative Var  0.312 0.506 0.670
```

a. Individually, does each rotated component (RC) explain the same, or different, amount of variance than the corresponding principal components (PCs)?

   Individually, each rotated component (RC) explain **different** amount of variance than the corresponding principal components (PCs).

b. Together, do the three rotated components explain the same, more, or less cumulative variance as the three principal components combined?

   Together, the three rotated components explain the **same** cumulative variance as the three principal components combined, which is 0.67.

c. Looking back at the items that shared similar loadings with multiple principal components (#2d), do those items have more clearly differentiated loadings among rotated components?

```
data_principal2$loadings[c(4,12,17), 1:3]
```

```
##            RC1       RC3       RC2
## Q4  0.2182880 0.1933627 0.8536838
## Q12 0.2327616 0.1861745 0.8542346
## Q17 0.2054021 0.1869028 0.8703910
```

   Yes, Q4, Q12, and Q17 have more clearly differentiated loadings among the rotated components.

d. Can you now more easily interpret the "meaning" of the 3 rotated components from the items that load best upon each of them? (see the wording of the questions of those items)

```
first3_loadings2 = data_principal2$loadings[, 1:3]
first3_loadings2 = round(first3_loadings2, 3)

RC1 = c(); RC2 = c(); RC3 = c()
for(i in 1:nrow(first3_loadings2)) {
  row = first3_loadings2[i,]
  rowname = rownames(first3_loadings2)[i]
  if(row[1] == max(row)) {
    RC1 = append(RC1, rowname)
  } else if(row[2] == max(row)) {
```

```
    RC3 = append(RC3, rowname)
  } else if(row[3] == max(row)) {
    RC2 = append(RC2, rowname)
  }
}
cat(" RC1:", paste(RC1, collapse=", "), "\n",
     "RC3:", paste(RC3, collapse=", "), "\n",
     "RC2:", paste(RC2, collapse=", "))
```

```
##  RC1: Q1, Q2, Q3, Q6, Q7, Q9, Q11, Q13, Q14, Q16, Q18
##  RC3: Q5, Q8, Q10, Q15
##  RC2: Q4, Q12, Q17
```

Yes, we can.

RC1 is generally about data protection.

RC3 is generally about transaction processes.

RC2 is generally about evidences to protect against denials.

e. If we reduced the number of extracted and rotated components to 2, does the meaning of our rotated components change?

```
data_principal3 = principal(data, nfactor=2, rotate="varimax", scores=TRUE)

first3_loadings3 = data_principal3$loadings[, 1:2]
first3_loadings3 = round(first3_loadings3, 3)

RC1 = c(); RC2 = c()
for(i in 1:nrow(first3_loadings3)) {
  row = first3_loadings3[i,]
  rowname = rownames(first3_loadings3)[i]
  if(row[1] == max(row)) {
    RC1 = append(RC1, rowname)
  } else if(row[2] == max(row)) {
    RC2 = append(RC2, rowname)
  }
}
cat(" RC1:", paste(RC1, collapse=", "), "\n",
     "RC2:", paste(RC2, collapse=", "))
```

```
##  RC1: Q1, Q2, Q3, Q5, Q6, Q7, Q8, Q9, Q10, Q11, Q13, Q14, Q15, Q16, Q18
##  RC2: Q4, Q12, Q17
```

Yes, it will change, except for RC2, because it still got the same items as before, which are Q4, Q12, Q17.

(ungraded) Looking back at all our results and analyses of this dataset (from this week and previous), how many components (1-3) do you believe we should extract and analyze to understand the security dataset? Feel free to suggest different answers for different purposes.

I think we should extract and analyze 3 components, because it brings the best result.