Business Analytics using Statistical Modeling
Assignment 14

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.

```
library(openxlsx)
```

```
## Warning: package 'openxlsx' was built under R version 3.2.5
```

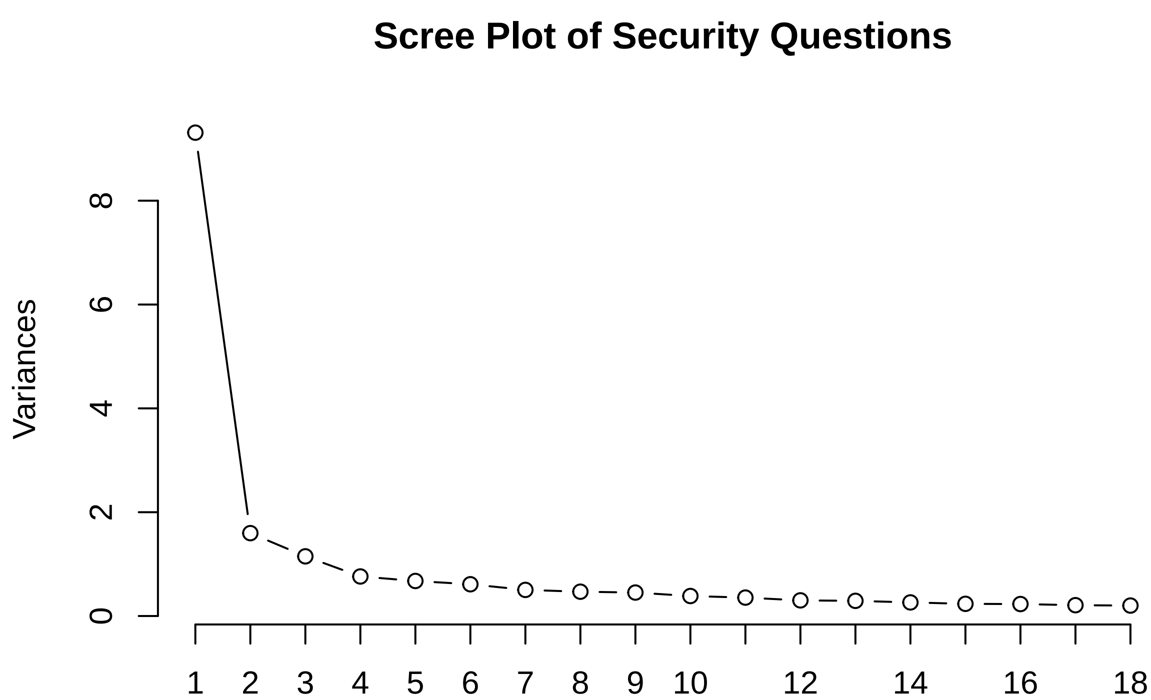
```
sec_qs <- read.xlsx('../13-security_questions.xlsx', sheet = 'data')
```

Question 1

We saw that identifying the number of principal components to keep can be challenging.

a. Report your earlier findings from applying the "eigenvalue > 1" and screeplot criteria to the security dataset.

```
sec_qs_pca <- prcomp(sec_qs, scale. = TRUE)
screeplot(sec_qs_pca, type = 'l', npcs = 18, main = 'Scree Plot of Security Questions')
```



```r
eigen(cor(sec_qs), 2)$values
```

```
##  [1] 9.3109533 1.5963320 1.1495582 0.7619759 0.6751412 0.6116636 0.5029855
##  [8] 0.4682788 0.4519711 0.3851964 0.3548816 0.3013071 0.2922773 0.2621437
## [15] 0.2345788 0.2304642 0.2087471 0.2015441
```

Based on the scree plot criteria, we should retain 1 factor in our analysis.

Based on the eigenvalues criteria (eigenvalue > 1), we should retain 3 factors in our analysis.

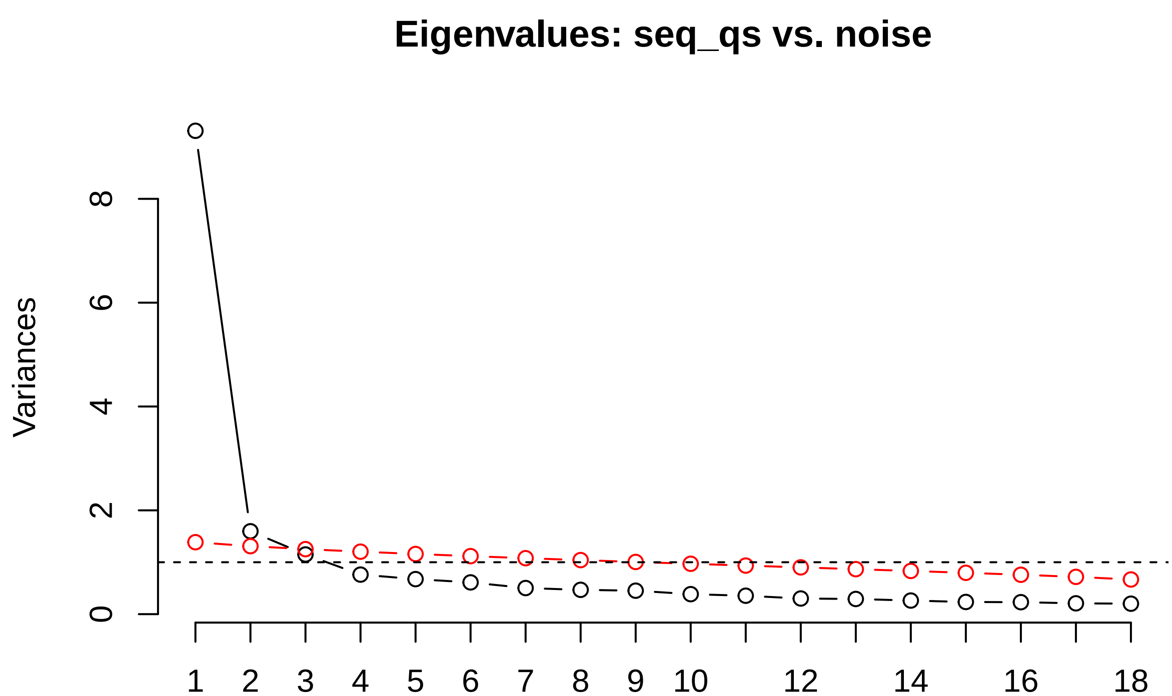
b. Perform a parallel analysis to find out how many principal components have higher eigenvalues than their counterparts in random datasets of the same dimensions as the security dataset.

```
sim_noise <- function(n, p) {
  noise <- data.frame(replicate(p, rnorm(n)))
  return(eigen(cor(noise))$values)
}

set.seed(0)
evalues_noise <- replicate(10000, sim_noise(nrow(sec_qs), ncol(sec_qs)))

evalues_mean <- apply(evalues_noise, 1, mean)

screeplot(sec_qs_pca, type = 'l', npcs = 18, main = 'Eigenvalues: seq_qs vs. noise')
lines(evalues_mean, type = 'b', col = 'red')
abline(h = 1, lty = 2)
```



There are 2 principal components which have higher eigenvalues than their counterparts in random datasets of the same dimensions as the security dataset.

Question 2

Earlier, we examined the eigenvectors of the security dataset. This time, let's examine loadings of our principal components (use the `principal()` method from the `psych` package).

```
library(psych)
sec_qs_principal <- principal(sec_qs, nfactor = 18, rotate = 'none', scores = TRUE)
```

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

```
loadings <- sec_qs_principal$loadings
loadings[, 1][abs(loadings[, 1]) > 0.7]
```

```
##        Q1        Q3        Q8        Q9       Q11       Q13       Q14
## 0.8169846 0.7655215 0.7861054 0.7230295 0.7529735 0.7119085 0.8114677
##       Q15       Q16       Q18
## 0.7040428 0.7575616 0.8067284
```

```
loadings[, 2][abs(loadings[, 2]) > 0.7]
```

```
## named numeric(0)
```

```
loadings[, 3][abs(loadings[, 3]) > 0.7]
```

```
## named numeric(0)
```

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

```
paste(round(sec_qs_principal$Vaccounted[3, 3] * 100, 2), '%', sep = '')
```

```
## [1] "66.98%"
```

c. Looking at commonality and uniqueness, which item's variance is least explained by the first 3 principal components?

```
sec_qs_pca_ori <- principal(sec_qs, nfactors = 3, rotate = 'none', scores = TRUE)
names(which.min(sec_qs_pca_ori$communality))
```

```
## [1] "Q2"
```

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

```
loadings_ori <- loadings[, 1:3]
sim_loads <- function(loadings, x) {
  return((abs(loadings[x, 1] - loadings[x, 2]) < 0.1 |
           abs(loadings[x, 2] - loadings[x, 3]) < 0.1 |
           abs(loadings[x, 1] - loadings[x, 3]) < 0.1) &
           (loadings[x, 1] < 0.7 & loadings[x, 2] < 0.7 & loadings[x, 3] < 0.7)
         )
}
sim_loads(loadings_ori, 1:18)
```

```
##    Q1    Q2    Q3    Q4    Q5    Q6    Q7    Q8    Q9   Q10   Q11   Q12
## FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
##   Q13   Q14   Q15   Q16   Q17   Q18
## FALSE FALSE FALSE FALSE  TRUE FALSE
```

There are 3 items (Q4, Q12, and Q17) that share similar loadings.

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

Q1	**I am convinced that this site respects the confidentiality of the transactions received from me**
Q2	All communications with this site are restricted to the site and me
Q3	**This site checks the information communicated with me for accuracy**
Q4	This site provides me with some evidence to protect against its denial of having received a transaction from r
Q5	The transactions I send are transmitted to the real site to which I want to transmit
Q6	This site checks all communications between the site and me for protection from wiretapping or eavesdroppir
Q7	This site never sells my personal information in their computer databases to other companies
Q8	**This site ascertains my identity before processing the transactions received from me**
Q9	**I can remove my personal information from this site when I want to**
Q10	The messages I receive are transmitted from the real site from which I want to receive them
Q11	**This site devotes time and effort to preventing unauthorized access to my personal information**
Q12	This site takes steps to make sure that the information in transit is not deleted
Q13	**This site provides me with some evidence to protect against its denial of having sent a message**
Q14	**This site devotes time and effort to verify the accuracy of the information in transit**
Q15	**This site ascertains my identity before sending any messages to me**
Q16	**Databases that contain my personal information are protected from unauthorized access**
Q17	This site provides me with some evidence to protect against its denial of having participated in a transaction
Q18	**This site uses some security controls for the confidentiality of the transactions received from me**

Figure 1: Items that seem to belong to first principal component

Looking at the wording of the questions, it seems like they are related to information confidentiality.

Question 3

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

```
sec_qs_pca_rot <- principal(sec_qs, nfactors = 3, rotate = 'varimax', scores = TRUE)
```

a. Individually, does each rotated component explain the same, or different, amount of variance than the three principal components?

```
sec_qs_pca_ori$Vaccounted[4, ]
```

```
##        PC1        PC2        PC3
## 0.77225464 0.13240049 0.09534487
```

```
sec_qs_pca_rot$Vaccounted[4, ]
```

```
##       RC1       RC3       RC2
## 0.4655570 0.2894737 0.2449692
```

Each rotated component explains the different amount of variance than the three principal components.

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

```r
paste(round(sec_qs_pca_ori$Vaccounted[3, 3] * 100, 2), '%', sep = '')
```

```
## [1] "66.98%"
```

```r
paste(round(sec_qs_pca_rot$Vaccounted[3, 3] * 100, 2), '%', sep = '')
```

```
## [1] "66.98%"
```

Together, the three rotated components explain the same cumulative variance as the three principal components combined.

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

```r
loadings_rot <- sec_qs_pca_rot$loadings[, 1:3]
loadings_ori[c(4, 12, 17), ]
```

```
##           PC1       PC2       PC3
## Q4  0.6233733 0.6430783 0.1080319
## Q12 0.6303505 0.6375312 0.1215228
## Q17 0.6175336 0.6642605 0.1100612
```

```r
loadings_rot[c(4, 12, 17), ]
```

```
##           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. those items have 1 loading that is more than 0.7, so they have more clearly differentiated loadings among rotated components.

d. Can you now 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)

```r
loadings_rot[, 1][abs(loadings_rot[, 1]) > 0.7]
```

```
##        Q7        Q9       Q11       Q14       Q16
## 0.7895344 0.7378148 0.7573493 0.7187578 0.7396241
```

Items 7, 9, 11, 14, and 16 load best upon RC1.

The questions are related to the protection of users' personal information.

```r
loadings_rot[, 2][abs(loadings_rot[, 2]) > 0.7]
```

```
##        Q5        Q8       Q10
## 0.8279850 0.7062018 0.8229206
```

Items 5, 8, and 10 load best upon RC3.

The questions are related to the transaction sending process.

```r
loadings_rot[, 3][abs(loadings_rot[, 3]) > 0.7]
```

```
##        Q4       Q12       Q17
## 0.8536838 0.8542346 0.8703910
```

Items 4, 12, and 17 load best upon RC2.

The questions are related to the transaction authorization and confirmation.

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

```r
sec_qs_pca_rot2 <- principal(sec_qs, nfactors = 2, rotate = 'varimax', scores = TRUE)
loadings_rot2 <- sec_qs_pca_rot2$loadings[, 1:2]
loadings_rot2[, 1][abs(loadings_rot2[, 1]) > 0.7]
```

```
##        Q1        Q7        Q9       Q11       Q14       Q16       Q18
## 0.7830951 0.7284256 0.7451939 0.7855784 0.7591295 0.7615661 0.7616746
```

```r
loadings_rot2[, 2][abs(loadings_rot2[, 2]) > 0.7]
```

```
##        Q4       Q12       Q17
## 0.8638430 0.8623433 0.8795921
```

The items that load best upon RC1 and RC2 when we used 3 rotated components, are all still load best upon RC1 and RC2 if we used 2 rotated components. It means that the meaning of our rotated components does not change.