

Exploratory factor analysis

Self-test answers



- What is the equation of a straight line?

$$Y_i = b_0 + b_1X_i + \varepsilon_i$$



- Using what you learnt in Chapter 6, or Section 17.6.2, calculate the correlation matrix for the factor scores. Compare this to the correlations of the factors in Output 17.10.

```
cor(pc5$scores)
```

```
          TC1          TC4          TC3          TC2
TC1  1.0000000  0.44299574  0.3620630 -0.18336649
TC4  0.4429957  1.00000000  0.3098527 -0.09732931
TC3  0.3620630  0.30985272  1.0000000 -0.16573305
TC2 -0.1833665 -0.09732931 -0.1657331  1.00000000
```

This matrix is the same as the correlation between factors in the model output:

```
With factor correlations of
          TC1          TC4          TC3          TC2
TC1  1.00  0.44  0.36 -0.18
TC4  0.44  1.00  0.31 -0.10
TC3  0.36  0.31  1.00 -0.17
TC2 -0.18 -0.10 -0.17  1.00
```

The only difference is the number of decimal places displayed, which we could rectify if we wanted to by using the `round()` function and executing:

```
round(cor(pc5$scores), 2)
```



- Can you think of another way of obtaining the structure matrix (the correlations between factors and items) now you've learned about factor scores?

The values in the structure matrix are the correlations between the factor scores, and the responses to each item. For example, let's correlate the factor scores (`pc5$scores`) with the variable `Q01` from the `raqData` dataframe (`raqData$Q01`). We can do this by executing:

```
cor(pc5$scores, raqData$Q01)
```

Let's use `round()` to round the results to 2 decimal places (note we have just put the command above into the `round()` function and specified 2 as the number of decimal places):

```
round(cor(pc5$scores, raqData$Q01), 2)
```

The result is:

```
 [ ,1]
TC1 0.40
TC4 0.50
TC3 0.53
TC2 0.01
```

These correspond to the values in the structure matrix in the book chapter. To make things a little clearer, let's do the same for items 6 and 18 in the questionnaire (which are the top two lines in the structure matrix in the chapter).

```
round(cor(pc5$scores, raqData$Q06), 2)
```

```
 [ ,1]
```

```
TC1 0.78
TC4 0.29
TC3 0.10
TC2 -0.14
```

```
round(cor(pc5$scores, raqData$Q18),2)
```

```
[,1]
TC1 0.76
TC4 0.36
TC3 0.42
TC2 -0.15
```

Again, these correlations are the values in the structure matrix, which shows that the structure matrix is made up of the correlations between the factor scores and the items in the analysis/questionnaire.

Oliver Twisted

Please Sir, can I have some more ... matrix algebra?



Calculation of factor score coefficients

$$B = R^{-1} A$$

$$= \begin{pmatrix} 4.76 & -7.46 & 3.91 & -2.35 & 2.42 & -0.49 \\ -7.46 & 18.49 & -12.42 & 5.45 & -5.54 & 1.22 \\ 3.91 & -12.42 & 10.07 & -3.65 & 3.79 & -0.96 \\ -2.35 & 5.45 & -3.65 & 2.97 & -2.16 & 0.02 \\ 2.42 & -5.54 & 3.79 & -2.16 & 2.98 & -0.56 \\ -0.49 & 1.22 & -0.96 & 0.02 & -0.56 & 1.27 \end{pmatrix} \begin{pmatrix} 0.87 & 0.01 \\ 0.96 & -0.03 \\ 0.92 & 0.04 \\ 0.00 & 0.82 \\ -0.10 & 0.75 \\ 0.09 & 0.70 \end{pmatrix}$$

Column 1 of matrix B

To get the first element of the first column of matrix B , you need to multiply each element in the *first column* of matrix A with the correspondingly placed element in the *first row* of matrix R^{-1} . Add these six products together to get the final value of the first element. To get the second element of the first column of matrix B , you need to multiply each element in the *first column* of matrix A with the correspondingly placed element in the *second row* of matrix R^{-1} . Add these six products together to get the final value ... and so on:

$$\begin{aligned}
B_{11} &= (4.75924 \times 0.87407) + (-7.46190 \times 0.95768) + (3.90949 \times 0.92138) \\
&\quad + (-2.35093 \times -0.00237) + (2.42104 \times -0.09575) + (-0.48607 \times 0.096) \\
&= 0.343
\end{aligned}$$

$$\begin{aligned}
B_{12} &= (-7.4619 \times 0.87407) + (18.48556 \times 0.95768) + (-12.41679 \times 0.92138) \\
&\quad + (5.445 \times -0.00237) + (-5.54427 \times -0.09575) + (1.22155 \times 0.096) \\
&= 0.376
\end{aligned}$$

$$\begin{aligned}
B_{13} &= (3.90949 \times 0.87407) + (-12.41679 \times 0.95768) + (10.07382 \times 0.92138) \\
&\quad + (-3.64853 \times -0.00237) + (3.78869 \times -0.09575) + (-0.95731 \times 0.096) \\
&= 0.362
\end{aligned}$$

$$\begin{aligned}
B_{14} &= (-2.35093 \times 0.87407) + (5.445 \times 0.95768) + (-3.64853 \times 0.92138) \\
&\quad + (2.96922 \times -0.00237) + (-2.16094 \times -0.09575) + (0.02255 \times 0.096) \\
&= 0.000
\end{aligned}$$

$$\begin{aligned}
B_{15} &= (2.42104 \times 0.87407) + (-5.54427 \times 0.95768) + (3.78869 \times 0.92138) \\
&\quad + (-2.16094 \times -0.00237) + (2.97983 \times -0.09575) + (-0.56017 \times 0.096) \\
&= -0.037
\end{aligned}$$

$$\begin{aligned}
B_{16} &= (-0.48607 \times 0.87407) + (1.22155 \times 0.95768) + (-0.95731 \times 0.92138) \\
&\quad + (0.02255 \times -0.00237) + (-0.56017 \times -0.09575) + (1.27072 \times 0.096) \\
&= 0.039
\end{aligned}$$

Column 2 of matrix B

To get the first element of the second column of matrix B , you need to multiply each element in the *second column* of matrix A with the correspondingly placed element in the *first row* of matrix R^{-1} . Add these six products together to get the final value. To get the second element of the second column of matrix B , you need to multiply each element in the *second column* of matrix A with the correspondingly placed element in the *second row* of matrix R^{-1} . Add these six products together to get the final value ... and so on:

$$B_{21} = (4.75924 \times 0.00842) + (-7.46190 \times -0.03653) + (3.90949 \times 0.03178) \\ + (-2.35093 \times 0.81556) + (2.42104 \times 0.75435) + (-0.48607 \times 0.69936) \\ = 0.006$$

$$B_{22} = (-7.4619 \times 0.00842) + (18.48556 \times -0.03653) + (-12.41679 \times 0.03178) \\ + (5.445 \times 0.81556) + (-5.54427 \times 0.75435) + (1.22155 \times 0.69936) \\ = -0.020$$

$$B_{23} = (3.90949 \times 0.00842) + (-12.41679 \times -0.03653) + (10.07382 \times 0.03178) \\ + (-3.64853 \times 0.81556) + (3.78869 \times 0.75435) + (-0.95731 \times 0.69936) \\ = 0.020$$

$$B_{24} = (-2.35093 \times 0.00842) + (5.445 \times -0.03653) + (-3.64853 \times 0.03178) \\ + (2.96922 \times 0.81556) + (-2.16094 \times 0.75435) + (0.02255 \times 0.69936) \\ = 0.473$$

$$B_{25} = (2.42104 \times 0.00842) + (-5.54427 \times -0.03653) + (3.78869 \times 0.03178) \\ + (-2.16094 \times 0.81556) + (2.97983 \times 0.75435) + (-0.56017 \times 0.69936) \\ = 0.437$$

$$B_{26} = (-0.48607 \times 0.00842) + (1.22155 \times -0.03653) + (-0.95731 \times 0.03178) \\ + (0.02255 \times 0.81556) + (-0.56017 \times 0.75435) + (1.27072 \times 0.69936) \\ = 0.405$$

Please Sir, can I have some more ... questionnaires?

What makes a good questionnaire?



As a rule of thumb, never to attempt to design a questionnaire! A questionnaire is very easy to design, but a *good* questionnaire is virtually impossible to design. The point is that it takes a long time to construct a questionnaire, with no guarantees that the end result will be of any use to anyone. A good questionnaire must have three things: discrimination, reliability and validity.

Discrimination

Discrimination is really an issue of item selection. Discrimination simply means that people with different scores on a questionnaire should differ in the construct of interest to you. For example, a questionnaire measuring social phobia should discriminate between people with social phobia and people without it (i.e. people in the different groups should score differently). There are three corollaries to consider:

1. People with the same score should be equal to each other along the measured construct.
2. People with different scores should be different from each other along the measured construct.
3. The degree of difference between people is proportional to the difference in scores.

This is all pretty self-evident really, so what's the fuss about? Well, let's take a really simple example of a three-item questionnaire measuring sociability. Imagine we administered this questionnaire to two people: Jane and Katie. Their responses are shown in Figure 1.

Jane

Katie

	Yes	No
1. I like going to parties	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2. I often go to the pub	<input type="checkbox"/>	<input checked="" type="checkbox"/>
3. I really enjoy meeting people	<input checked="" type="checkbox"/>	<input type="checkbox"/>

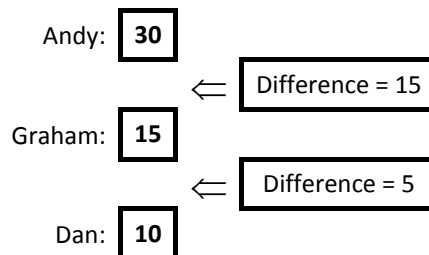
	Yes	No
1. I like going to parties	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2. I often go to the pub	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3. I really enjoy meeting people	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Figure 1

Jane responded yes to items 1 and 3 but no to item 2. If we score a yes with the value 1 and a no with a 0, then we can calculate a total score of 2. Katie on the other hand answers yes to items 1 and 2 but no to item 3. Using the same scoring system her score is also 2. Therefore, numerically you have identical answers (i.e. both Jane and Katie score 2 on this questionnaire); therefore, these two people should be comparable in their sociability – are they?

The answer is: not necessarily. It seems that Katie likes to go to parties and the pub but doesn't enjoy meeting people in general, whereas Jane enjoys parties and meeting people but doesn't enjoy the pub. It seems that Katie likes social situations involving alcohol (e.g. the pub and parties), while Jane likes socializing in general but can't tolerate cigarette smoke. In many ways, therefore, these people are very different because our questions are contaminated by other factors (i.e. attitudes to alcohol or smoky environments). A good questionnaire should be designed such that people with identical numerical scores are identical in the construct being measured — and that's not as easy to achieve as you might think!

A second related point is *score differences*. Imagine you take scores on the Spider Phobia Questionnaire. Imagine you have three participants who do the questionnaire and get the following scores:



Andy scores 30 on the SPQ (very arachnophobic), Graham scores 15 (moderately phobic) and Dan scores 10 (not very phobic at all). Does this mean that Dan and Graham are more similar in their spider phobia than Graham and Andy? In theory this should be the case because Graham's score is more similar to Dan's (difference = 5) than it is to Andy's (difference = 15). In addition, is it the case that Andy is three times more phobic of spiders than Dan is? Is he twice as phobic as Graham? Again, his scores suggest that he should be. The point is that you can't guarantee in advance that differences in score are going to be comparable, yet a questionnaire needs to be constructed such that the difference in score is proportional to the difference between people.

Validity

Items on your questionnaire must measure something, and a good questionnaire measures what you designed it to measure (this is called *validity*). Validity basically means 'measuring what you think you're measuring'. So, an anxiety measure that actually measures assertiveness is not valid; however, a materialism scale that does actually measure materialism is valid. Validity is a difficult thing to assess and it can take several forms:

1. *Content validity*: Items on a questionnaire must relate to the construct being measured. For example, a questionnaire measuring intrusive thoughts is pretty useless if it contains items relating to statistical ability. Content validity is really how representative your questions are — the sampling adequacy of items. This is achieved when items are first selected: don't include items that are blatantly very similar to other items, and ensure that questions cover the full range of the construct.
2. *Criterion validity*: This is basically whether the questionnaire is measuring what it claims to measure. In an ideal world, you could assess this by relating scores on each item to real-world observations (e.g. comparing scores on sociability items with the number of times a person actually goes out to socialize). This is often impractical and so there are other

techniques such as: (a) using the questionnaire in a variety of situations and seeing how predictive it is; (b) seeing how well it correlates with other known measures of your construct (i.e. sociable people might be expected to score highly on extroversion scales); and (c) using statistical techniques such as the Item Validity Index (IVI).

3. **Factorial validity:** This validity basically refers to whether the factor structure of the questionnaire makes intuitive sense. As such, factorial validity is assessed through factor analysis. When you have your final set of items you can conduct a factor analysis on the data (see the book). Factor analysis takes your correlated questions and recodes them into uncorrelated, underlying variables called factors (an example might be recoding the variables height, chest size, shoulder width and weight into an underlying variable called 'build'). As another example, to assess success in a course we might measure attentiveness in seminars, the amount of notes taken in seminars, and the number of questions asked during seminars — all of these variables may relate to an underlying trait such as 'motivation to succeed'. Factor analysis produces a table of items and their correlation, or loading, with each factor. A factor is composed of items that correlate highly with it. Factorial validity can be seen from whether the items tied on to factors make intuitive sense or not. Basically, if your items cluster into meaningful groups then you can infer factorial validity.

Validity is a necessary but not sufficient condition of a questionnaire.

Reliability

A questionnaire must be not only valid, but also reliable. Reliability is basically the ability of the questionnaire to produce the same results under the same conditions. To be reliable the questionnaire must first be valid. Clearly the easiest way to assess reliability is to test the same group of people twice: if the questionnaire is reliable you'd expect each person's scores to be the same at both points in time. So, scores on the questionnaire should correlate perfectly (or very nearly!). However, in reality, if we did test the same people twice then we'd expect some practice effects and confounding effects (people might remember their responses from last time). Also this method is not very useful for questionnaires purporting to measure something that we would expect to change (such as depressed mood or anxiety). These problems can be overcome using the *alternate form* method in which two comparable questionnaires are devised and compared. Needless to say, this is a rather time-consuming way to ensure reliability, and fortunately there are statistical methods to make life much easier.

The simplest statistical technique is the *split-half method*. This method randomly splits the questionnaire items into two groups. A score for each subject is then calculated based on each half of the scale. If a scale is very reliable we'd expect a person's score to be the same on one half of the scale as on the other, and so the two halves should correlate perfectly. The correlation between the two halves is the statistic computed in the split-half method, large correlations being a sign of reliability.¹ The problem with this method is that there are a number of ways in which a set of data can be split into two and so the results might be a result of the way in which the data were split. To overcome this problem, Cronbach suggested splitting the data in two in every conceivable way and computing the correlation coefficient for each split. The average of these values is known as Cronbach's alpha, which is the most common measure of scale reliability. As a rough guide, a value of .8 is seen as an acceptable value for Cronbach's alpha; values substantially lower indicate an unreliable scale (see the book for more detail).

How to design your questionnaire

Step 1: Choose a construct

First you need to decide on what you would like to measure. Once you have done this use PsychLit and the Web of Knowledge to do a basic search for some information on this topic. I don't expect you to search through reams of material, but just get some basic background on the construct you're testing and how it might relate to psychologically important things. For example, if you looked at empathy, this is seen as an important component of Carl Roger's client-centred therapy; therefore,

¹ In fact the correlation coefficient is adjusted to account for the smaller sample on which scores from the scale are based (remember that these scores are based on *half* of the items on the scale).

having the personality trait of empathy might be useful if you were to become a Rogerian therapist. It follows then that having a questionnaire to measure this trait might be useful for selection purposes on Rogerian therapy training courses. So, basically you need to set some kind of context to why the construct is important — this information will form the basis of your introduction.

Step 2: Decide on a response scale

A fundamental issue is how you want respondents to answer questions. You could choose to have:

- *Yes/no or yes/no/don't know scales*: This forces people to give one answer or another even though they might feel that they are neither a *yes* nor *no*. Also, imagine you were measuring intrusive thoughts and you had an item 'I think about killing children'. Chances are everyone would give a no response to that statement (even if they did have those thoughts) because it is a very undesirable thing to admit. Therefore, all this item is doing is subtracting a value from everybody's score – it tells you nothing meaningful, it is just noise in the data. This scenario can also occur when you have a rating scale with a don't know response (because people just cannot make up their minds and opt for the neutral response). This is why it is sometimes nice to have questionnaires with a neutral point to help you identify which things people really have no feelings about. Without this midpoint you are simply making people go one way or the other which is comparable to balancing a coin on its edge and seeing which side up it lands when it falls. Basically, when forced 50% will choose one option while 50% will choose the opposite – this is just noise in your data.
- *Likert scale*: This is the standard agree–disagree ordinal categories response. It comes in many forms:
 - *3-point*: Agree⇒Neither Agree nor Disagree⇒Disagree
 - *5-point*: Agree⇒Midpoint⇒Neither Agree nor Disagree⇒Midpoint⇒Disagree
 - *7-Point*: Agree⇒2 Midpoints⇒Neither Agree nor Disagree⇒2 Midpoints⇒Disagree

Questions should encourage respondents to use all points of the scale. Ideally, the statistical distribution of responses to a single item should be normal with a mean that lies at the centre of the scale (so on a 5-point Likert scale the mean on a given question should be 3). The range of scores should also cover all possible responses.

Step 3: Generate your items

Once you've found a construct to measure and decided on the type of response scale you're going to use, the next task is to generate items. I want you to restrict your questionnaire to around 30 items (20 minimum). The best way to generate items is to 'brainstorm' a small sample of people. This involves getting people to list as many facets of your construct as possible. For example, if you devised a questionnaire on exam anxiety, you might ask a number of students (20 or so) from a variety of courses (arts and science), years (first, second and final) and even institutions (friends at other universities) to list (on a piece of paper) as many things about exams as possible that make them anxious. It is good if you can include people within this sample that you think might be at the extremes of your construct (e.g. select a few people who get very anxious about exams and some who are very calm). This enables you to get items that span the entire spectrum of the construct that you want to measure.

This will give you a pool of items to inspire questions. Rephrase your sample's suggestions in a way that fits the rating scale you've chosen and then eliminate any questions that are basically the same. You should hopefully begin with a pool of say 50–60 questions that you can reduce to about 30 by eliminating obviously similar questions.

Things to consider:

1. *Wording of questions*: The way in which questions are phrased can bias the answers that people give; For example, Gaskell, Wright, and O'Muirheartaigh (1993) report several studies in which subtle changes in the wording of survey questions can radically affect people's responses. Gaskell et al.'s article is a very readable and useful summary of this work and their conclusions might be useful to you when thinking about how to phrase your questions.
2. *Response bias*: This is the tendency of respondents to give the same answer to every question. Try to reverse-phrase a few items to avoid response bias (and remember to score these items in reverse when you enter the data into **R**).

Step 4: Collect the data

Once you've written your questions, randomize their order and produce your questionnaire. This is the questionnaire that you're going to test. Photocopy the questionnaire and administer it to as many people as possible (one benefit of making these questionnaires short is that it minimizes the time taken to complete them!). You should aim for 50–100 respondents, but the more you get, the better your analysis (which is why I suggest working in slightly bigger groups to make data collection easier).

Step 5: Analysis

Enter the data into **R** by having each question represented by a column in **R**. Translate your response scale into numbers (i.e. a 5-point Likert might be 1 = completely disagree, 2 = disagree, 3 = neither agree nor disagree, 4 = agree, 5 = completely agree). Reverse-phrased items should be scored in reverse too!

What we're trying to do with this analysis is to first eliminate any items on the questionnaire that aren't useful. So, we're trying to reduce our 30 items further before we run our factor analysis. We can do this by looking at descriptive statistics and also correlations between questions.

Descriptive statistics: The first thing to look at is the statistical distribution of item scores. This alone will enable you to throw out many redundant items. Therefore, the first thing to do when piloting a questionnaire is look at descriptive statistics on the questionnaire items. This is easily done in **R** (see the book chapter). We're on the lookout for:

1. *Range:* Any item that has a limited range (all the points of the scale have not been used).
2. *Skew:* I mentioned above that ideally each question should elicit a normally distributed set of responses across subjects (each item's mean should be at the centre of the scale and there should be no skew). To check for items that produce skewed data, look for the *skewness* and its standard error in your **R** output. We have also discovered in this book that you can divide the skewness by its standard error to form a z-score (see Chapter 5).
3. *Standard deviation:* Related to the range and skew of the distribution, items with high or low standard deviations may cause problems, so be wary of high and low values for the SD.

These are your first steps. Basically if any of these rules are violated then your items become non-comparable (in terms of the factor analysis) which makes the questionnaire pretty meaningless!

Correlations: All of your items should intercorrelate at a significant level if they are measuring aspects of the same thing. If any items do not correlate at a 5% or 1% level of significance then exclude them. You can get a table of intercorrelations in **R**. The book gives more detail on screening correlation coefficients for items that correlate with few others or correlate too highly with other items (multicollinearity and singularity).

Factor analysis: When you've eliminated any items that have distributional problems or do not correlate with each other, then run your factor analysis on the remaining items and try to interpret the resulting factor structure. The book chapter details the process of factor analysis. What you should do is examine the factor structure and decide:

1. Which factors to retain.
2. Which items load on to those factors.
3. What your factors represent.
4. If there are any items that don't load highly on to any factors, they should be eliminated from future versions of the questionnaire (for our purposes you need only state that they are not useful items as you won't have time to revise and retest your questionnaires!).

Step 6: Assess the questionnaire

Having looked at the factor structure, you need to check the reliability of your items and the questionnaire as a whole. You should run a reliability analysis on the questionnaire. This is explained in Section 17.8 of the book. There are two things to look at: (1) the Item Reliability Index (IRI), which is the correlation between the score on the item and the score on the test as a whole multiplied by the standard deviation of that item (called the corrected item–total correlation in SPSS). SPSS will do this corrected item–total correlation and we'd hope that these values would be significant for all items. Although we don't get significance values as such, we can look for correlations greater than about .3 (although the exact value depends on the sample size, this is a good cut-off for the size of sample you'll probably have). Any items having correlations less than .3 should be excluded from the

questionnaire. (2) Cronbach's alpha, as we've seen, should be .8 or more and the deletion of an item should not affect this value too much (see the reliability analysis handout for more detail).

The end?

You should conclude by describing your factor structure and the reliability of the scale. Also say whether there are items that you would drop in a future questionnaire. In an ideal world we'd then generate new items to add to the retained items and start the whole process again!

Please Sir, can I have some more ... kmo?

To use the KMO test you first need to execute the following function written by G. Jay Kerns, Youngstown State University (see <http://tolstoy.newcastle.edu.au/R/e2/help/07/08/22816.html>):

```
kmo = function( data ){
  library(MASS)
  X <- cor(as.matrix(data))
  iX <- ginv(X)
  S2 <- diag(diag((iX^-1)))
  AIS <- S2**%iX**%S2
  IS <- X+AIS-2*S2
  Dai <- sqrt(diag(diag(AIS)))
  IR <- ginv(Dai)**%IS**%ginv(Dai)
  AIR <- ginv(Dai)**%AIS**%ginv(Dai)
  a <- apply((AIR - diag(diag(AIR)))^2, 2, sum)
  AA <- sum(a)
  b <- apply((X - diag(nrow(X)))^2, 2, sum)
  BB <- sum(b)
  MSA <- b/(b+a)
  AIR <- AIR-diag(nrow(AIR))+diag(MSA)
  kmo <- BB/(AA+BB)
  if (kmo >= 0.00 && kmo < 0.50){test <- 'The KMO test yields a degree of common
variance unacceptable for FA.'}
  else if (kmo >= 0.50 && kmo < 0.60){test <- 'The KMO test yields a degree of
common variance miserable.'}
  else if (kmo >= 0.60 && kmo < 0.70){test <- 'The KMO test yields a degree of
common variance mediocre.'}
  else if (kmo >= 0.70 && kmo < 0.80){test <- 'The KMO test yields a degree of
common variance middling.' }
  else if (kmo >= 0.80 && kmo < 0.90){test <- 'The KMO test yields a degree of
common variance meritorious.' }
  else { test <- 'The KMO test yields a degree of common variance marvelous.' }
  ans <- list( overall = kmo,
              report = test,
              individual = MSA,
              AIS = AIS,
              AIR = AIR )
  return(ans)
}
```

Labcoat Leni's real research

World wide addiction?

Problem

Nichols, L. A., & Nicki, R. (2004). *Psychology of Addictive Behaviors*, 18(4), 381–384



The Internet is now a household tool. In 2007 it was estimated that around 179 million people worldwide used the Internet (over 100 million of those were in the USA and Canada). From the increasing popularity (and usefulness) of the Internet has emerged a new phenomenon: Internet addiction. This is now a serious and recognized problem, but until very recently it was very difficult to research this topic because there was not a psychometrically sound measure of Internet addiction. That is, until Laura Nichols and Richard Nicki developed the Internet Addiction Scale, IAS (Nichols & Nicki, 2004). (Incidentally, while doing some research on this topic I encountered an Internet addiction recovery website that I won't name but that offered a whole host of resources that would keep you online for ages, such as questionnaires, an online support group, videos, articles, a recovery blog and podcasts. It struck me that that this was a bit like having a recovery centre for heroin addiction where the addict arrives to be greeted by a nice-looking counsellor who says 'there's a huge pile of heroin in the corner over there, just help yourself'.)

Anyway, Nichols and Nicki developed a 36-item questionnaire to measure internet addiction. It contained items such as 'I have stayed on the Internet longer than I intended to' and 'My grades/work have suffered because of my Internet use' which could be responded to on a 5-point scale (Never, Rarely, Sometimes, Frequently, Always). They collected data from 207 people to validate this measure.

The data from this study are in the file **Nichols & Nicki (2004).dat**. The authors dropped two items because they had low means and variances, and dropped three others because of relatively low correlations with other items. They performed a principal components analysis on the remaining 31 items. Labcoat Leni wants you to run some descriptive statistics to work out which two items were dropped for having low means/variances, then inspect a correlation matrix to find the three items that were dropped for having low correlations. Finally, he wants you to run a principal components analysis on the data.

Solution

First, it is very important to remember to load in the data:

```
internetData<-read.delim("Nichols & Nicki (2004).dat", header = TRUE)
```

We can then calculate the correlation matrix, using the `cor()` function and executing:
`internetMatrix<-cor(internetData)`

Executing this command creates a matrix of correlation coefficients called `internetMatrix`. We can use this matrix in the analysis (although we don't have to). To make our eyes hurt a little less, let's use the `round()` function to display only 2 decimal places of the correlation matrix that we have just created:

```
round(internetMatrix, 2)
```

We can then use the resulting correlation matrix below to check the pattern of relationships. First, scan the matrix for correlations greater than .3, then look for variables that only have a small number of correlations greater than this value. Then scan the correlation coefficients themselves and look for

any greater than .9. If any are found then you should be aware that a problem could arise because of multicollinearity in the data.

	ias1	ias2	ias3	ias4	ias5	ias6	ias7	ias8	ias9	ias10	ias11	ias12
ias1	1.00	0.43	0.46	0.35	0.52	0.56	0.48	0.48	0.51	0.43	0.42	0.43
ias2	0.43	1.00	0.33	0.54	0.38	0.24	0.39	0.32	0.29	0.30	0.26	0.32
ias3	0.46	0.33	1.00	0.52	0.47	0.41	0.49	0.62	0.50	0.40	0.43	0.46
ias4	0.35	0.54	0.52	1.00	0.46	0.27	0.45	0.44	0.37	0.37	0.27	0.44
ias5	0.52	0.38	0.47	0.46	1.00	0.48	0.43	0.59	0.51	0.52	0.34	0.44
ias6	0.56	0.24	0.41	0.27	0.48	1.00	0.50	0.43	0.50	0.59	0.42	0.50
ias7	0.48	0.39	0.49	0.45	0.43	0.50	1.00	0.47	0.54	0.60	0.41	0.60
ias8	0.48	0.32	0.62	0.44	0.59	0.43	0.47	1.00	0.63	0.48	0.43	0.54
ias9	0.51	0.29	0.50	0.37	0.51	0.50	0.54	0.63	1.00	0.56	0.44	0.49
ias10	0.43	0.30	0.40	0.37	0.52	0.59	0.60	0.48	0.56	1.00	0.51	0.64
ias11	0.42	0.26	0.43	0.27	0.34	0.42	0.41	0.43	0.44	0.51	1.00	0.51
ias12	0.43	0.32	0.46	0.44	0.44	0.50	0.60	0.54	0.49	0.64	0.51	1.00
ias13	0.12	0.37	0.19	0.31	0.24	0.10	0.22	0.24	0.21	0.21	0.23	0.26
ias14	0.49	0.38	0.40	0.36	0.40	0.44	0.37	0.42	0.45	0.49	0.42	0.43
ias15	0.51	0.35	0.42	0.27	0.37	0.39	0.36	0.42	0.45	0.44	0.55	0.38
ias16	0.52	0.30	0.40	0.31	0.36	0.39	0.27	0.42	0.34	0.25	0.40	0.27
ias17	0.35	0.25	0.39	0.29	0.40	0.50	0.50	0.43	0.50	0.57	0.46	0.60
ias18	0.47	0.28	0.36	0.34	0.47	0.39	0.55	0.46	0.54	0.58	0.46	0.50
ias19	0.46	0.28	0.65	0.42	0.51	0.49	0.44	0.63	0.48	0.49	0.49	0.54
ias20	0.48	0.29	0.44	0.42	0.47	0.45	0.53	0.53	0.52	0.56	0.46	0.55
ias21	0.47	0.29	0.45	0.36	0.52	0.49	0.55	0.57	0.63	0.58	0.54	0.61
ias22	0.16	0.15	0.18	0.15	0.15	0.16	0.15	0.29	0.10	0.22	0.33	0.22
ias23	0.28	0.19	0.26	0.25	0.34	0.29	0.21	0.30	0.32	0.30	0.31	0.25
ias24	0.42	0.31	0.60	0.41	0.49	0.38	0.49	0.54	0.51	0.45	0.53	0.49
ias25	0.45	0.26	0.35	0.37	0.47	0.36	0.39	0.51	0.41	0.40	0.42	0.43
ias26	0.52	0.28	0.44	0.27	0.43	0.65	0.44	0.47	0.46	0.51	0.57	0.53
ias27	0.40	0.29	0.39	0.26	0.35	0.33	0.37	0.40	0.34	0.28	0.36	0.27
ias28	0.49	0.20	0.37	0.22	0.39	0.57	0.42	0.49	0.50	0.48	0.58	0.47
ias29	0.54	0.32	0.40	0.43	0.55	0.46	0.48	0.45	0.51	0.58	0.45	0.53
ias30	0.47	0.30	0.42	0.39	0.47	0.42	0.41	0.43	0.41	0.38	0.36	0.43
ias31	0.33	0.24	0.51	0.28	0.33	0.28	0.33	0.39	0.30	0.30	0.30	0.34
ias32	0.22	0.15	0.26	0.17	0.25	0.36	0.14	0.23	0.13	0.27	0.19	0.22
ias33	0.50	0.36	0.45	0.47	0.60	0.35	0.50	0.49	0.53	0.47	0.44	0.46
ias34	0.44	0.20	0.29	0.22	0.42	0.47	0.43	0.41	0.37	0.52	0.34	0.37
ias35	0.38	0.27	0.43	0.25	0.42	0.26	0.25	0.47	0.25	0.27	0.31	0.25
ias36	0.49	0.32	0.46	0.35	0.47	0.51	0.66	0.56	0.55	0.64	0.49	0.56

	ias13	ias14	ias15	ias16	ias17	ias18	ias19	ias20	ias21	ias22	ias23	ias24
ias1	0.12	0.49	0.51	0.52	0.35	0.47	0.46	0.48	0.47	0.16	0.28	0.42
ias2	0.37	0.38	0.35	0.30	0.25	0.28	0.28	0.29	0.29	0.15	0.19	0.31
ias3	0.19	0.40	0.42	0.40	0.39	0.36	0.65	0.44	0.45	0.18	0.26	0.60
ias4	0.31	0.36	0.27	0.31	0.29	0.34	0.42	0.42	0.36	0.15	0.25	0.41
ias5	0.24	0.40	0.37	0.36	0.40	0.47	0.51	0.47	0.52	0.15	0.34	0.49
ias6	0.10	0.44	0.39	0.39	0.50	0.39	0.49	0.45	0.49	0.16	0.29	0.38
ias7	0.22	0.37	0.36	0.27	0.50	0.55	0.44	0.53	0.55	0.15	0.21	0.49
ias8	0.24	0.42	0.42	0.42	0.43	0.46	0.63	0.53	0.57	0.29	0.30	0.54
ias9	0.21	0.45	0.45	0.34	0.50	0.54	0.48	0.52	0.63	0.10	0.32	0.51
ias10	0.21	0.49	0.44	0.25	0.57	0.58	0.49	0.56	0.58	0.22	0.30	0.45
ias11	0.23	0.42	0.55	0.40	0.46	0.46	0.49	0.46	0.54	0.33	0.31	0.53
ias12	0.26	0.43	0.38	0.27	0.60	0.50	0.54	0.55	0.61	0.22	0.25	0.49
ias13	1.00	0.19	0.11	0.10	0.12	0.16	0.16	0.19	0.27	0.31	0.20	0.33
ias14	0.19	1.00	0.47	0.34	0.41	0.47	0.43	0.57	0.43	0.30	0.35	0.46
ias15	0.11	0.47	1.00	0.41	0.37	0.52	0.51	0.52	0.43	0.20	0.44	0.40
ias16	0.10	0.34	0.41	1.00	0.25	0.34	0.43	0.38	0.26	0.27	0.26	0.32
ias17	0.12	0.41	0.37	0.25	1.00	0.51	0.43	0.54	0.52	0.18	0.26	0.43
ias18	0.16	0.47	0.52	0.34	0.51	1.00	0.45	0.64	0.55	0.15	0.42	0.45
ias19	0.16	0.43	0.51	0.43	0.43	0.45	1.00	0.52	0.53	0.19	0.32	0.61
ias20	0.19	0.57	0.52	0.38	0.54	0.64	0.52	1.00	0.57	0.26	0.41	0.51
ias21	0.27	0.43	0.43	0.26	0.52	0.55	0.53	0.57	1.00	0.27	0.28	0.56
ias22	0.31	0.30	0.20	0.27	0.18	0.15	0.19	0.26	0.27	1.00	0.39	0.21
ias23	0.20	0.35	0.44	0.26	0.26	0.42	0.32	0.41	0.28	0.39	1.00	0.35
ias24	0.33	0.46	0.40	0.32	0.43	0.45	0.61	0.51	0.56	0.21	0.35	1.00
ias25	0.20	0.35	0.40	0.35	0.36	0.52	0.48	0.58	0.41	0.28	0.40	0.49
ias26	0.14	0.53	0.65	0.49	0.54	0.51	0.60	0.62	0.55	0.28	0.47	0.47
ias27	0.18	0.40	0.41	0.36	0.17	0.24	0.38	0.34	0.26	0.18	0.20	0.44
ias28	0.19	0.51	0.71	0.44	0.43	0.51	0.53	0.61	0.54	0.32	0.52	0.46
ias29	0.19	0.47	0.47	0.44	0.48	0.57	0.46	0.71	0.54	0.18	0.44	0.49
ias30	0.21	0.45	0.39	0.34	0.42	0.43	0.52	0.46	0.50	0.28	0.28	0.55
ias31	0.17	0.31	0.28	0.22	0.29	0.18	0.42	0.35	0.36	0.16	0.18	0.55
ias32	0.26	0.30	0.22	0.27	0.26	0.13	0.24	0.30	0.26	0.33	0.24	0.21
ias33	0.23	0.43	0.47	0.35	0.36	0.55	0.44	0.62	0.54	0.15	0.29	0.46
ias34	0.10	0.42	0.48	0.37	0.39	0.54	0.39	0.52	0.41	0.20	0.32	0.27
ias35	0.16	0.45	0.43	0.35	0.26	0.35	0.46	0.45	0.24	0.26	0.48	0.42
ias36	0.20	0.54	0.57	0.39	0.58	0.65	0.57	0.69	0.59	0.27	0.47	0.49

	ias25	ias26	ias27	ias28	ias29	ias30	ias31	ias32	ias33	ias34	ias35	ias36
ias1	0.45	0.52	0.40	0.49	0.54	0.47	0.33	0.22	0.50	0.44	0.38	0.49
ias2	0.26	0.28	0.29	0.20	0.32	0.30	0.24	0.15	0.36	0.20	0.27	0.32
ias3	0.35	0.44	0.39	0.37	0.40	0.42	0.51	0.26	0.45	0.29	0.43	0.46
ias4	0.37	0.27	0.26	0.22	0.43	0.39	0.28	0.17	0.47	0.22	0.25	0.35

```

ias5  0.47  0.43  0.35  0.39  0.55  0.47  0.33  0.25  0.60  0.42  0.42  0.47
ias6  0.36  0.65  0.33  0.57  0.46  0.42  0.28  0.36  0.35  0.47  0.26  0.51
ias7  0.39  0.44  0.37  0.42  0.48  0.41  0.33  0.14  0.50  0.43  0.25  0.66
ias8  0.51  0.47  0.40  0.49  0.45  0.43  0.39  0.23  0.49  0.41  0.47  0.56
ias9  0.41  0.46  0.34  0.50  0.51  0.41  0.30  0.13  0.53  0.37  0.25  0.55
ias10 0.40  0.51  0.28  0.48  0.58  0.38  0.30  0.27  0.47  0.52  0.27  0.64
ias11 0.42  0.57  0.36  0.58  0.45  0.36  0.30  0.19  0.44  0.34  0.31  0.49
ias12 0.43  0.53  0.27  0.47  0.53  0.43  0.34  0.22  0.46  0.37  0.25  0.56
ias13 0.20  0.14  0.18  0.19  0.19  0.21  0.17  0.26  0.23  0.10  0.16  0.20
ias14 0.35  0.53  0.40  0.51  0.47  0.45  0.31  0.30  0.43  0.42  0.45  0.54
ias15 0.40  0.65  0.41  0.71  0.47  0.39  0.28  0.22  0.47  0.48  0.43  0.57
ias16 0.35  0.49  0.36  0.44  0.44  0.34  0.22  0.27  0.35  0.37  0.35  0.39
ias17 0.36  0.54  0.17  0.43  0.48  0.42  0.29  0.26  0.36  0.39  0.26  0.58
ias18 0.52  0.51  0.24  0.51  0.57  0.43  0.18  0.13  0.55  0.54  0.35  0.65
ias19 0.48  0.60  0.38  0.53  0.46  0.52  0.42  0.24  0.44  0.39  0.46  0.57
ias20 0.58  0.62  0.34  0.61  0.71  0.46  0.35  0.30  0.62  0.52  0.45  0.69
ias21 0.41  0.55  0.26  0.54  0.54  0.50  0.36  0.26  0.54  0.41  0.24  0.59
ias22 0.28  0.28  0.18  0.32  0.18  0.28  0.16  0.33  0.15  0.20  0.26  0.27
ias23 0.40  0.47  0.20  0.52  0.44  0.28  0.18  0.24  0.29  0.32  0.48  0.47
ias24 0.49  0.47  0.44  0.46  0.49  0.55  0.55  0.21  0.46  0.27  0.42  0.49
ias25 1.00  0.48  0.39  0.52  0.53  0.43  0.27  0.21  0.53  0.41  0.48  0.51
ias26 0.48  1.00  0.41  0.76  0.56  0.47  0.25  0.28  0.49  0.63  0.52  0.64
ias27 0.39  0.41  1.00  0.46  0.32  0.25  0.39  0.15  0.37  0.27  0.41  0.35
ias28 0.52  0.76  0.46  1.00  0.56  0.39  0.22  0.30  0.41  0.56  0.45  0.65
ias29 0.53  0.56  0.32  0.56  1.00  0.45  0.28  0.26  0.68  0.59  0.38  0.64
ias30 0.43  0.47  0.25  0.39  0.45  1.00  0.43  0.33  0.43  0.30  0.31  0.49
ias31 0.27  0.25  0.39  0.22  0.28  0.43  1.00  0.20  0.33  0.11  0.33  0.35
ias32 0.21  0.28  0.15  0.30  0.26  0.33  0.20  1.00  0.26  0.25  0.26  0.19
ias33 0.53  0.49  0.37  0.41  0.68  0.43  0.33  0.26  1.00  0.47  0.37  0.52
ias34 0.41  0.63  0.27  0.56  0.59  0.30  0.11  0.25  0.47  1.00  0.49  0.58
ias35 0.48  0.52  0.41  0.45  0.38  0.31  0.33  0.26  0.37  0.49  1.00  0.43
ias36 0.51  0.64  0.35  0.65  0.64  0.49  0.35  0.19  0.52  0.58  0.43  1.00

```

We know that the authors eliminated three items for having low correlations. If we scan the correlation matrix, we can see that the lowest correlations are for items IAS-13 ('I have felt a persistent desire to cut down or control my use of the Internet'), IAS-22 ('I have neglected things which are important and need doing'), and IAS-32 ('I find myself thinking/longing about when I will go on the Internet again'). As such these variables will also be excluded from the factor analysis.

To see more clearly which items had the lowest overall correlations, we could look at which items had the lowest average correlation. We can calculate the average correlation for each item using the `stat.desc()` function in the `pastecs` package.

First install the '`pastecs`' package, if you haven't already installed it, by executing:

```
install.packages("pastecs")
```

and then load the package:

```
library(pastecs)
```

We can then get the mean correlation (along with some other descriptives) rounded to 2 decimal places by executing:

```
round(stat.desc(internetMatrix),2)
```

mean	ias1	ias2	ias3	ias4	ias5	ias6	ias7	ias8	ias9	ias10	ias11	ias12
	0.45	0.32	0.43	0.36	0.44	0.43	0.44	0.47	0.45	0.46	0.43	0.45
mean	ias13	ias14	ias15	ias16	ias17	ias18	ias19	ias20	ias21	ias22	ias23	ias24
	0.22	0.43	0.44	0.36	0.41	0.45	0.47	0.50	0.47	0.25	0.34	0.46
mean	ias25	ias26	ias27	ias28	ias29	ias30	ias31	ias32	ias33	ias34	ias35	ias36
	0.43	0.50	0.34	0.48	0.48	0.42	0.32	0.26	0.46	0.40	0.38	0.51

Looking at the (edited) output above, we can see more clearly that items 13, 22 and 32 had the lowest average correlations and should therefore be excluded from the factor analysis.

Next we want to have a look at the means and variance. To get the descriptives, we again can use the `stat.desc()` function by executing:

```
internetDescriptives<-stat.desc(internetData)
```

We can then round the descriptive to 2 decimal places by executing:

```
round(internetDescriptives,2)
```

	ias1	ias2	ias3	ias4	ias5	ias6	ias7	ias8	ias9	ias10	ias11	ias12
mean	1.49	1.59	2.68	2.01	1.51	1.22	1.41	2.09	1.66	1.36	1.48	1.71
var	0.68	0.86	1.15	1.15	0.72	0.32	0.64	1.27	0.91	0.48	0.57	0.74
	ias13	ias14	ias15	ias16	ias17	ias18	ias19	ias20	ias21	ias22	ias23	ias24
mean	2.03	1.33	1.23	1.30	1.31	1.33	2.03	1.32	1.58	1.25	1.14	1.89
var	1.26	0.42	0.27	0.48	0.46	0.47	0.90	0.41	0.92	0.43	0.18	0.92
	ias25	ias26	ias27	ias28	ias29	ias30	ias31	ias32	ias33	ias34	ias35	ias36
mean	1.39	1.25	1.91	1.24	1.23	1.51	2.27	1.54	1.35	1.11	1.50	1.27
var	0.48	0.32	0.97	0.32	0.32	0.64	1.07	0.80	0.56	0.12	0.71	0.34

The resulting (edited) output is above (NB: I have deleted everything except the means and variances from the table as these are the only descriptives that we are interested in here) and shows us that the items with the lowest values are IAS-23 ('I see my friends less often because of the time that I spend on the Internet') and IAS-34 ('When I use the Internet, I experience a buzz or a high'). Therefore, these items will also be excluded from the factor analysis.

Before we do the principal components analysis, we need to remove **ias13**, **ias22**, **ias32**, **ias23** and **ias34** from the *internetData* dataframe. To do this we need to use the *remove.vars()* function from the *gdata* package. Therefore, we first need to install and load the *gdata* package by executing:

```
install.packages("gdata")
library(gdata)
```

We can then remove the variables by executing:

```
internetData.2<-remove.vars(internetData, c("ias13", "ias22", "ias32", "ias23",
"ias34"))
```

Executing the above command will create a new dataframe called *internetData.2* which does not contain the variables **ias13**, **ias22**, **ias 32**, **ias 23** and **ias34**. R helpfully tells you that it has done what you asked it to do by printing in the output the following:

```
Removing variable 'ias13'
Removing variable 'ias22'
Removing variable 'ias32'
Removing variable 'ias23'
Removing variable 'ias34'
```

We should now run Bartlett's test and the KMO on the *internetData.2* dataframe. We can run this test either on the raw data or on the correlation matrix. To run it from the raw data simply input the dataframe (in this case *internetData.2*) into the function:

```
cortest.bartlett(internetData.2)
```

For factor analysis to work we need some relationships between variables and if the *R*-matrix were an identity matrix then all correlation coefficients would be zero. Therefore, we want this test to be *significant* (i.e., have a significance value less than .05). A significant test tells us that the *R*-matrix is not an identity matrix; therefore, there are some relationships between the variables we hope to include in the analysis. For these data, Bartlett's test is highly significant, $\chi^2(465) = 4238.98$, $p < .001$, and therefore factor analysis is appropriate.

```
R was not square, finding R from data
$chisq
[1] 4238.976

$p.value
[1] 0

$df
[1] 465
```

Next we'd also like the KMO. Once you have executed the code of the function itself (see the book chapter), you can use it by simply entering the name of your dataframe into it and executing:

```
kmo(internetData.2)
```

The results of the KMO test are shown below. Kaiser (1974) recommends a bare minimum of .5, and values between .5 and .7 are mediocre, values between .7 and .8 are good, values between .8 and .9 are great and values above .9 are superb (Hutcheson & Sofroniou, 1999). For these data the overall value is .94, which falls into the range of being superb (or 'marvellous' as the report puts it), so we should be confident that the sample size and the data are adequate for factor analysis.

```
$overall
[1] 0.9421769

$report
[1] "The KMO test yields a degree of common variance marvelous."

$individual
ias1      ias2      ias3      ias4      ias5      ias6      ias7      ias8
0.9516499 0.8951047 0.9425295 0.9230381 0.9515424 0.9241382 0.9195672 0.9471428

ias9      ias10     ias11     ias12     ias14     ias15     ias16     ias17
0.9556244 0.9437068 0.9540661 0.9564852 0.9386538 0.9169296 0.9406837 0.9567060

ias18     ias19     ias20     ias21     ias24     ias25     ias26     ias27
0.9634669 0.9659927 0.9646791 0.9595863 0.9463071 0.9465699 0.9388335 0.9331421

ias28     ias29     ias30     ias31     ias33     ias35     ias36
0.9182153 0.9373334 0.9501855 0.8939196 0.9288033 0.9209140 0.9531873
```

Finally, we'd like the determinant of the correlation matrix. To find the determinant, we use the `det()` function, into which we place the name of a correlation matrix. As we have not yet computed this matrix for the `internetData.2` dataframe, we can get the determinant by putting the `cor()` function for the raw data into the `det()` function:

```
det(cor(internetData.2))
[1] 3.556999e-10
```

This value is greater than the necessary value of 0.00001 and, as such, our determinant does not seem problematic and we do not need to remove any more variables from the dataframe at this stage.

Next we can do the principal components analysis. As I mentioned in the book chapter, when conducting principal components analysis we begin by establishing the linear variates within the data and then decide how many of these variates to retain (or 'extract'). Therefore, our starting point is to create a principal components model that has the same number of factors as there are variables in the data: by doing this we are just reducing the data set down to its underlying factors. By extracting as many factors as there are variables we can inspect their eigenvalues and make decisions about which factors to extract.

To create this model from the raw data (NB: you can also create this model from the correlation matrix – if you have created one – but both methods will give you identical results; see the book chapter) we execute (remember that we now have 31 variables in the dataframe rather than 36):

```
pc1 <- principal(internetData.2, nfactors = 31, rotate = "none")
```

This command creates a model called `pc1`, which extracts 31 factors – the same as the number of variables. We have set the rotation method to "none", which means that we won't carry out factor rotation because we don't need to at this stage.

We can look at the results of the principal components analysis by executing its name:

```
pc1
```

The (edited) output below shows the results of the first principal components model. The first part of the output is the unrotated loadings; currently these are not interesting, and so to save space I have not included them in the output below.

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10	PC11	PC12	PC13	PC14	PC15
SS loadings	14.43	1.65	1.56	1.21	1.01	0.87	0.81	0.77	0.74	0.66	0.62	0.59	0.54	0.53	0.47
Proportion Var	0.47	0.05	0.05	0.04	0.03	0.03	0.03	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02
Cumulative Var	0.47	0.52	0.57	0.61	0.64	0.67	0.69	0.72	0.74	0.77	0.79	0.80	0.82	0.84	0.85

	PC16	PC17	PC18	PC19	PC20	PC21	PC22	PC23	PC24	PC25	PC26	PC27	PC28	PC29
SS loadings	0.45	0.42	0.41	0.38	0.35	0.32	0.31	0.29	0.27	0.24	0.23	0.22	0.19	0.17
Proportion Var	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Cumulative Var	0.87	0.88	0.90	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.97	0.98	0.99	0.99

PC30 PC31

```
SS loadings    0.15 0.13
Proportion Var 0.00 0.00
Cumulative Var 1.00 1.00
```

The thing to look at is the eigenvalues. The eigenvalues associated with each factor represent the variance explained by that particular linear component. **R** calls these *SS loadings* (sums of squared loadings), because they are the sum of the squared loadings. (You can also find them in a variable associated with the model called *values*, so in our case we could access this variable using `pc1$values`.)

R also displays the eigenvalues in terms of the proportion of variance explained. Factor 1 explains 14.43 units of variance out of a possible 31 (the number of factors), so as a proportion this is $14.43/31 = 0.47$; this is the value that **R** reports. We can convert these proportions to percentages by multiplying by 100; so, factor 1 explains 47% of the total variance. It should be clear that the first few factors explain relatively large amounts of variance (especially factor 1) whereas subsequent factors explain only small amounts of variance. Based on Kaiser's criterion of retaining factors with eigenvalues greater than 1, we would retain five factors.

Let's rerun the analysis, specifying that we want to retain five factors. To do this, we use an identical command to the previous model but we change `nfactors = 31` to `nfactors = 5` because we now want only five factors. (We should also change the name of the resulting model so that we don't overwrite the previous one.)

```
pc2 <- principal(internetData.2, nfactors = 5, rotate = "none")
```

We can look at this model by executing its name:

```
pc2
```

The output below shows the second principal components model. Again, the output contains the unrotated factor loadings, but only for the first five factors. Notice that these are unchanged from the previous factor loading matrix. Also notice that the eigenvalues (SS loadings), proportions of variance explained and cumulative proportion of variance explained are also unchanged (except now there are only five of them, because we only have five components). However, the communalities (the *h2* column) and uniquenesses (the *u2* column) are changed. Remember that the communality is the proportion of common variance within a variable. Principal component analysis works on the initial assumption that all variance is common; therefore, before extraction the communalities are all 1. In effect, all of the variance associated with a variable is assumed to be common variance. Once factors have been extracted, we have a better idea of how much variance is, in reality, common. The communalities in the output reflect this common variance. So, for example, we can say that 63% of the variance associated with question 1 is common, or shared, variance. Another way to look at these communalities is in terms of the proportion of variance explained by the underlying factors. Before extraction, there were as many factors as there are variables, so all variance is explained by the factors and communalities are all 1. However, after extraction some of the factors are discarded and so some information is lost. The retained factors cannot explain all of the variance present in the data, but they can explain some. The amount of variance in each variable that can be explained by the retained factors is represented by the communalities after extraction.

Now that we have the communalities, we can go back to Kaiser's criterion to see whether we still think that five factors should have been extracted. Kaiser's criterion is accurate when there are less than 30 variables and the communalities after extraction are greater than .7, or when the sample size exceeds 250 and the average communality is greater than .6. For these data the sample size is 207, there are 31 variables and the mean communality is .64, so extracting five factors is probably not warranted.

```
Principal Components Analysis
Call: principal(r = internetData.2, nfactors = 5, rotate = "none")
Standardized loadings based upon correlation matrix
      PC1  PC2  PC3  PC4  PC5  h2  u2
ias1 0.70 0.10 0.12 0.19 0.28 0.63 0.37
ias2 0.48 0.33 -0.16 0.43 0.37 0.69 0.31
ias3 0.68 0.40 -0.11 -0.24 0.06 0.69 0.31
ias4 0.56 0.37 -0.36 0.31 0.09 0.68 0.32
ias5 0.69 0.18 -0.16 0.17 -0.08 0.57 0.43
ias6 0.67 -0.20 0.03 -0.15 0.38 0.66 0.34
```

```

ias7  0.70 -0.09 -0.33  0.02  0.12  0.62  0.38
ias8  0.74  0.20 -0.07 -0.12 -0.09  0.61  0.39
ias9  0.72 -0.09 -0.18 -0.05  0.06  0.57  0.43
ias10 0.73 -0.30 -0.24 -0.04  0.08  0.68  0.32
ias11 0.67 -0.14  0.14 -0.19  0.10  0.54  0.46
ias12 0.72 -0.18 -0.30 -0.12  0.06  0.66  0.34
ias14 0.67  0.00  0.14  0.09  0.10  0.48  0.52
ias15 0.68 -0.09  0.39  0.04  0.10  0.64  0.36
ias16 0.55  0.18  0.35  0.16  0.28  0.56  0.44
ias17 0.65 -0.33 -0.22 -0.16  0.05  0.61  0.39
ias18 0.71 -0.29 -0.05  0.21 -0.20  0.68  0.32
ias19 0.75  0.17  0.05 -0.26 -0.03  0.66  0.34
ias20 0.79 -0.16  0.03  0.14 -0.26  0.73  0.27
ias21 0.74 -0.19 -0.24 -0.14 -0.03  0.66  0.34
ias24 0.72  0.25 -0.09 -0.29 -0.17  0.70  0.30
ias25 0.66  0.05  0.13  0.16 -0.40  0.64  0.36
ias26 0.77 -0.21  0.35 -0.06  0.11  0.78  0.22
ias27 0.53  0.33  0.32 -0.07  0.08  0.51  0.49
ias28 0.74 -0.26  0.43 -0.08  0.03  0.80  0.20
ias29 0.76 -0.14 -0.02  0.28 -0.16  0.69  0.31
ias30 0.64  0.13 -0.10 -0.08 -0.03  0.45  0.55
ias31 0.50  0.44 -0.12 -0.42 -0.08  0.64  0.36
ias33 0.71  0.07 -0.10  0.32 -0.24  0.68  0.32
ias35 0.56  0.28  0.42  0.03 -0.28  0.64  0.36
ias36 0.80 -0.23  0.01  0.01 -0.07  0.70  0.30

```

```

                PC1 PC2 PC3 PC4 PC5
SS loadings    14.43 1.65 1.56 1.21 1.01
Proportion Var  0.47 0.05 0.05 0.04 0.03
Cumulative Var  0.47 0.52 0.57 0.61 0.64

```

Test of the hypothesis that 5 factors are sufficient.

The degrees of freedom for the null model are 465 and the objective function was 21.76

The degrees of freedom for the model are 320 and the objective function was 3.53

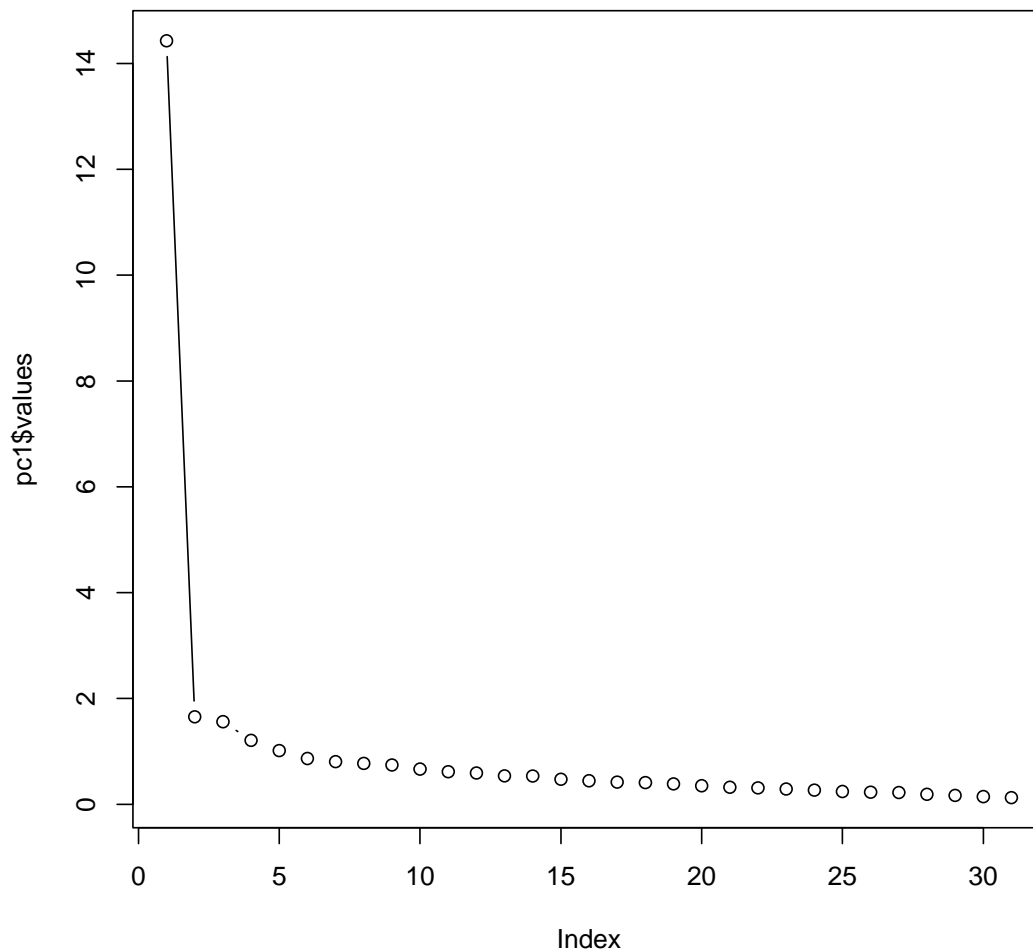
The number of observations was 207 with Chi Square = 676.34 with prob < 1.2e-27

Fit based upon off diagonal values = 0.99

We should also consider the scree plot. As mentioned above, the eigenvalues are stored in a variable called *pc1\$values*, and we can draw a quick scree plot using the *plot()* function, by executing:

```
plot(pc1$values, type = "b")
```

This command simply plots the eigenvalues (*y*) against the factor number (*x*). By default, the *plot()* function will plot points (*type="p"*). We want to see a line so that we can look at the trend (we could ask for this by specifying *type="l"*), but ideally we want to look at both a line and points on the same graph, which is why we specify *type="b"*.



The resulting scree plot shows a clear one-factor solution. This is the solution that the authors adopted. Because we are retaining only one factor we can ignore the rotated factor solution and just look at the unrotated factor loading matrix. This shows that all items have a high loading on factor 1.

Interpreting the factor loading matrix is a little complex, and we can make it easier by using the `print.psych()` function. This does two things: first, it removes loadings that are below a certain value that we specify (by using the `cut` option); and second, it reorders the items to try to put them into their factors, which we request using the `sort` option. Generally you should be very careful with the cut-off value – if you think that a loading of .4 will be interesting, you should use a lower cut-off (say, .3), because you don't want to miss a loading that was .39. Execute this command:

```
print.psych(pc2, cut = 0.3, sort = TRUE)
```

```
Principal Components Analysis
Call: principal(r = internetData.2, nfactors = 5, rotate = "none")
Standardized loadings based upon correlation matrix
  item PC1  PC2  PC3  PC4  PC5  h2  u2
ias36 31 0.80                0.70 0.30
ias20 19 0.79                0.73 0.27
ias26 23 0.77      0.35    0.78 0.22
ias29 26 0.76                0.69 0.31
ias19 18 0.75                0.66 0.34
ias21 20 0.74                0.66 0.34
ias8   8 0.74                0.61 0.39
ias28 25 0.74      0.43    0.80 0.20
ias10 10 0.73                0.68 0.32
ias9   9 0.72                0.57 0.43
```

```

ias24 21 0.72                0.70 0.30
ias12 12 0.72                0.66 0.34
ias18 17 0.71                0.68 0.32
ias33 29 0.71                0.68 0.32
ias1  1 0.70                0.63 0.37
ias7  7 0.70       -0.33    0.62 0.38
ias5  5 0.69                0.57 0.43
ias15 14 0.68       0.39    0.64 0.36
ias3  3 0.68   0.40    0.69 0.31
ias6  6 0.67                0.38 0.66 0.34
ias11 11 0.67                0.54 0.46
ias14 13 0.67                0.48 0.52
ias25 22 0.66                -0.40 0.64 0.36
ias17 16 0.65  -0.33    0.61 0.39
ias30 27 0.64                0.45 0.55
ias35 30 0.56       0.42    0.64 0.36
ias4  4 0.56   0.37  -0.36  0.31    0.68 0.32
ias16 15 0.55                0.35    0.56 0.44
ias27 24 0.53   0.33   0.32    0.51 0.49
ias31 28 0.50   0.44    -0.42    0.64 0.36
ias2  2 0.48   0.33    0.43  0.37  0.69 0.31

```

```

          PC1 PC2 PC3 PC4 PC5
SS loadings 14.43 1.65 1.56 1.21 1.01
Proportion Var 0.47 0.05 0.05 0.04 0.03
Cumulative Var 0.47 0.52 0.57 0.61 0.64

```

Test of the hypothesis that 5 factors are sufficient.

The degrees of freedom for the null model are 465 and the objective function was 21.76

The degrees of freedom for the model are 320 and the objective function was 3.53
The number of observations was 207 with Chi Square = 676.34 with prob < 1.2e-27

Fit based upon off diagonal values = 0.99

The authors reported their analysis as follows: ‘We conducted principal-components analyses on the log transformed scores of the IAS (see above). On the basis of the scree test (Cattell, 1978) and the percentage of variance accounted for by each factor, we judged a one-factor solution to be most appropriate. This component accounted for a total of 47% of the variance. A value for loadings of .30 (Floyd & Widaman, 1995) was used as a cut-off for items that did not relate to a component.

‘All 31 items loaded on this component, which was interpreted to represent aspects of a general factor relating to Internet addiction reflecting the negative consequences of excessive Internet use’ (p. 382)

Smart Alex’s solutions

Task 1

- The University of Sussex is constantly seeking to employ the best people possible as lecturers (no, really, it is). Anyway, they wanted to revise a questionnaire based on Bland’s theory of research methods lecturers. This theory predicts that good research methods lecturers should have four characteristics: (1) a profound love of statistics; (2) an enthusiasm for experimental design; (3) a love of teaching; and (4) a complete absence of normal interpersonal skills. These characteristics should be related (i.e. correlated). The ‘Teaching of Statistics for Scientific Experiments’ (TOSSE) already existed, but the university revised this questionnaire and it became the ‘Teaching of Statistics for Scientific Experiments – Revised’ (TOSSE–R). They gave this questionnaire to 239 research methods lecturers around the world to see if it supported Bland’s theory. The questionnaire is in Figure 17.9 (reproduced below), and the data are in **TOSSE.R.dat**. Conduct a factor analysis (with appropriate rotation) to see the factor structure of the data.

SD = Strongly Disagree, D = Disagree, N = Neither, A = Agree, SA = Strongly Agree						
		SD	D	N	A	SA
1	I once woke up in the middle of a vegetable patch hugging a turnip that I'd mistakenly dug up thinking it was Roy's largest root	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

2	If I had a big gun I'd shoot all the students I have to teach	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3	I memorize probability values for the <i>F</i> -distribution	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4	I worship at the shrine of Pearson	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5	I still live with my mother and have little personal hygiene	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
6	Teaching others makes me want to swallow a large bottle of bleach because the pain of my burning oesophagus would be light relief in comparison	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7	Helping others to understand sums of squares is a great feeling	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
8	I like control conditions	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
9	I calculate three ANOVAs in my head before getting out of bed every morning	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
10	I could spend all day explaining statistics to people	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11	I like it when people tell me I've helped them to understand factor rotation	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
12	People fall asleep as soon as I open my mouth to speak	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
13	Designing experiments is fun	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
14	I'd rather think about appropriate dependent variables than go to the pub	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
15	I soil my pants with excitement at the mere mention of factor analysis	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
16	Thinking about whether to use repeated or independent measures thrills me	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
17	I enjoy sitting in the park contemplating whether to use participant observation in my next experiment	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
18	Standing in front of 300 people in no way makes me lose control of my bowels	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
19	I like to help students	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
20	Passing on knowledge is the greatest gift you can bestow on an individual	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
21	Thinking about Bonferroni corrections gives me a tingly feeling in my groin	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
22	I quiver with excitement when thinking about designing my next experiment	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
23	I often spend my spare time talking to the pigeons ... and even they die of boredom	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
24	I tried to build myself a time machine so that I could go back to the 1930s and follow Fisher around on my hands and knees licking the floor on which he'd just trodden	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
25	I love teaching	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
26	I spend lots of time helping students	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
27	I love teaching because students have to pretend to like me or they'll get bad marks	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
28	My cat is my only friend	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

It goes without saying that first you need to set your working directory to where the **TOSSE.R.dat** file is saved and then load in the data by executing:

```
tossData<-read.delim("TOSSE.R.dat", header = TRUE)
```

There are some data missing in our *tossData* dataframe. To create a new data set (I am going to call the new dataset *tossData.2*) without missing data we can use the *na.omit()* function and execute:

```
tossData.2<-na.omit(tossData)
```

We can run Bartlett's test by executing:

```
cortest.bartlett(tossData.2)
```

For factor analysis to work we need some relationships between variables and if the R -matrix were an identity matrix then all correlation coefficients would be zero. Therefore, we want this test to be *significant* (i.e., have a significance value less than .05). A significant test tells us that the R -matrix is not an identity matrix; therefore, there are some relationships between the variables we hope to include in the analysis. For these data, Bartlett's test is highly significant, $\chi^2(378) = 2989.77, p < .001$, and therefore factor analysis is appropriate.

```
R was not square, finding R from data
$chisq
[1] 2989.769

$p.value
[1] 0

$df
[1] 378
```

Next we'd also like the KMO. To do this we can use `kmo()`, written by G. Jay Kerns, which calculates the KMO and a variety of other things. The function itself is easy to use, but because it is not part of a package you will have to execute the function manually before you can use it (see *Oliver Twisted*). Once you have executed the code of the function itself (you only need to do this once per session and so if you have already executed this function you do not need to execute it again), you can use it by simply entering the name of your dataframe into it and executing:

```
kmo(tossData.2)

$overall
[1] 0.8940547

$report
[1] "The KMO test yields a degree of common variance meritorious."
```

The results of the KMO test are shown above. For these data the overall value is .89, which is great, so we should be confident that the sample size and the data are adequate for factor analysis.

Finally, we'd like the determinant of the correlation matrix. To find the determinant, we use the `det()` function, into which we place the name of a correlation matrix. We haven't computed this matrix for the current data yet (`tossData`). This is not a problem, though, because we can just get the determinant by putting the `cor()` function for the raw data into the `det()` function:

```
det(cor(tossData.2))

[1] 1.240294e-06
```

The determinant of the correlation matrix is .00000124, which is smaller than .00001 and, therefore, indicates that multicollinearity could be a problem in these data (although, strictly speaking, because we're using principal components analysis we don't need to worry).

Next we can do the principal components analysis. As I mentioned in the book chapter, when conducting principal components analysis we begin by establishing the linear variates within the data and then decide how many of these variates to retain (or 'extract'). Therefore, our starting point is to create a principal components model that has the same number of factors as there are variables in the data: by doing this we are just reducing the dataset down to its underlying factors. By extracting as many factors as there are variables we can inspect their eigenvalues and make decisions about which factors to extract.

To create this model from the raw data (you can also create this model from the correlation matrix if you have created one, but both methods will give you identical results; see the book chapter) we execute:

```
pc1 <- principal(tossData.2, nfactors = 28, rotate = "none")
```

This command creates a model called `pc1`, which extracts 28 factors – the same as the number of variables. We have set the rotation method to *"none"*, which means that we won't carry out factor rotation because we don't need to at this stage.

We can look at the results of the principal components analysis by executing its name:

```
pc1
```

The (edited) output below shows the results of the first principal components model. The first part of the output is the unrotated loadings; currently these are not interesting, and so to save space I have not included them in the output below.

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10
SS loadings	9.06	2.79	1.66	1.51	1.18	0.99	0.93	0.82	0.79	0.74
Proportion Var	0.32	0.10	0.06	0.05	0.04	0.04	0.03	0.03	0.03	0.03
Cumulative Var	0.32	0.42	0.48	0.54	0.58	0.61	0.65	0.68	0.70	0.73

	PC11	PC12	PC13	PC14	PC15	PC16	PC17	PC18	PC19	PC20	PC21
SS loadings	0.71	0.65	0.62	0.57	0.54	0.52	0.49	0.45	0.42	0.38	0.34
Proportion Var	0.03	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.01	0.01
Cumulative Var	0.76	0.78	0.80	0.82	0.84	0.86	0.88	0.89	0.91	0.92	0.94

	PC22	PC23	PC24	PC25	PC26	PC27	PC28
SS loadings	0.33	0.31	0.29	0.26	0.25	0.21	0.16
Proportion Var	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Cumulative Var	0.95	0.96	0.97	0.98	0.99	0.99	1.00

Test of the hypothesis that 28 factors are sufficient.

The degrees of freedom for the null model are 378 and the objective function was 13.6

The degrees of freedom for the model are -28 and the objective function was 0
The number of observations was 231 with Chi Square = 0 with prob < NA

Fit based upon off diagonal values = 1

The thing to look at is the eigenvalues (SS loadings). The eigenvalues associated with each factor represent the variance explained by that particular linear component. (You can also find them in a variable associated with the model called *values*, so in our case we could access this variable using `pc1$values`).

R also displays the eigenvalues in terms of the proportion of variance explained. Factor 1 explains 9.06 units of variance out of a possible 28 (the number of factors) so as a proportion this is $9.06/28 = 0.32$; this is the value that **R** reports. We can convert these proportions to percentages by multiplying by 100; so, factor 1 explains 32% of the total variance). It should be clear that the first few factors explain relatively large amounts of variance (especially factor 1) whereas subsequent factors explain only small amounts of variance. Based on Kaiser's criterion of retaining factors with eigenvalues greater than 1, we would retain five factors. Is this warranted?

Let's rerun the analysis, specifying that we want to retain five factors. To do this, we use an identical command to the previous model but we change `nfactors = 28` to `nfactors = 5`. (We should also change the name of the resulting model so that we don't overwrite the previous one.)

```
pc2 <- principal(tossData.2, nfactors = 5, rotate = "none")
```

We can look at this model by executing its name:

```
pc2
```

The output below shows the second principal components model. Again, the output contains the unrotated factor loadings, but only for the first five factors. Notice that these are unchanged from the previous factor loading matrix. Also notice that the eigenvalues (SS loadings), proportions of variance explained and cumulative proportion of variance explained are also unchanged (except now there are only five of them, because we only have five components). However, the communalities (the *h2* column) and uniquenesses (the *u2* column) are changed. Remember that the communality is the proportion of common variance within a variable. Principal component analysis works on the initial assumption that all variance is common; therefore, before extraction the communalities are all 1. In effect, all of the variance associated with a variable is assumed to be common variance. Once factors have been extracted, we have a better idea of how much variance is, in reality, common. The communalities in the output reflect this common variance. So, for example, we can say that 65% of the variance associated with question 1 is common, or shared, variance. Another way to look at these communalities is in terms of the proportion of variance explained by the underlying factors. Before extraction, there were as many factors as there are variables, so all variance is explained by the factors and communalities are all 1. However, after extraction some of the factors are discarded and so some information is lost. The retained factors cannot explain all of the variance present in the

data, but they can explain some. The amount of variance in each variable that can be explained by the retained factors is represented by the communalities after extraction.

Now that we have the communalities, we can go back to Kaiser's criterion to see whether we still think that five factors should have been extracted. Kaiser's criterion is accurate when there are less than 30 variables and the communalities after extraction are greater than .7, or when the sample size exceeds 250 and the average communality is greater than .6. For these data the sample size is 239, there are 28 variables, and the mean communality is .579, so extracting five factors is not really warranted.

```
Principal Components Analysis
Call: principal(r = tossData.2, nfactors = 5, rotate = "none")
Standardized loadings based upon correlation matrix
      PC1  PC2  PC3  PC4  PC5  h2  u2
q1  0.68 -0.34 -0.09  0.10 -0.21 0.65 0.35
q2  0.37 -0.54 -0.01  0.30  0.31 0.62 0.38
q3  0.58  0.26 -0.22  0.31 -0.19 0.59 0.41
q4  0.60  0.03 -0.15  0.45 -0.05 0.59 0.41
q5  0.45 -0.24  0.52 -0.05 -0.10 0.54 0.46
q6  0.29 -0.50 -0.11  0.26  0.45 0.62 0.38
q7  0.53  0.39 -0.15  0.16  0.05 0.49 0.51
q8  0.80  0.06 -0.15 -0.07  0.09 0.68 0.32
q9  0.72 -0.30 -0.11  0.11 -0.03 0.64 0.36
q10 0.50 -0.27 -0.04 -0.10  0.29 0.42 0.58
q11 0.68  0.26  0.12 -0.02 -0.05 0.54 0.46
q12 0.13  0.07  0.52 -0.07 -0.01 0.30 0.70
q13 0.67 -0.02 -0.21 -0.17 -0.06 0.53 0.47
q14 0.61 -0.15  0.16 -0.52  0.14 0.71 0.29
q15 0.56 -0.17 -0.11 -0.09 -0.39 0.51 0.49
q16 0.65  0.09 -0.03 -0.50  0.04 0.68 0.32
q17 0.77 -0.13 -0.19 -0.24  0.08 0.71 0.29
q18 0.42 -0.52  0.17  0.11  0.15 0.51 0.49
q19 0.19  0.62 -0.02  0.10  0.32 0.54 0.46
q20 0.46  0.38  0.05  0.19  0.29 0.48 0.52
q21 0.67  0.10 -0.25 -0.06 -0.19 0.57 0.43
q22 0.79 -0.20 -0.19 -0.25  0.00 0.77 0.23
q23 0.43 -0.25  0.52  0.24 -0.11 0.59 0.41
q24 0.61  0.16 -0.13  0.34 -0.33 0.65 0.35
q25 0.50  0.44  0.16  0.10  0.26 0.55 0.45
q26 0.53  0.50  0.14 -0.21  0.04 0.60 0.40
q27 0.58  0.38  0.33  0.13  0.09 0.62 0.38
q28 0.46 -0.02  0.52  0.08 -0.22 0.54 0.46

      SS loadings      PC1  PC2  PC3  PC4  PC5
Proportion Var  9.06 2.79 1.66 1.51 1.18
Cumulative Var  0.32 0.10 0.06 0.05 0.04
Cumulative Var  0.32 0.42 0.48 0.54 0.58
```

Test of the hypothesis that 5 factors are sufficient.

The degrees of freedom for the null model are 378 and the objective function was 13.6
The degrees of freedom for the model are 248 and the objective function was 2.49
The number of observations was 231 with Chi Square = 539.34 with prob < 1.1e-23

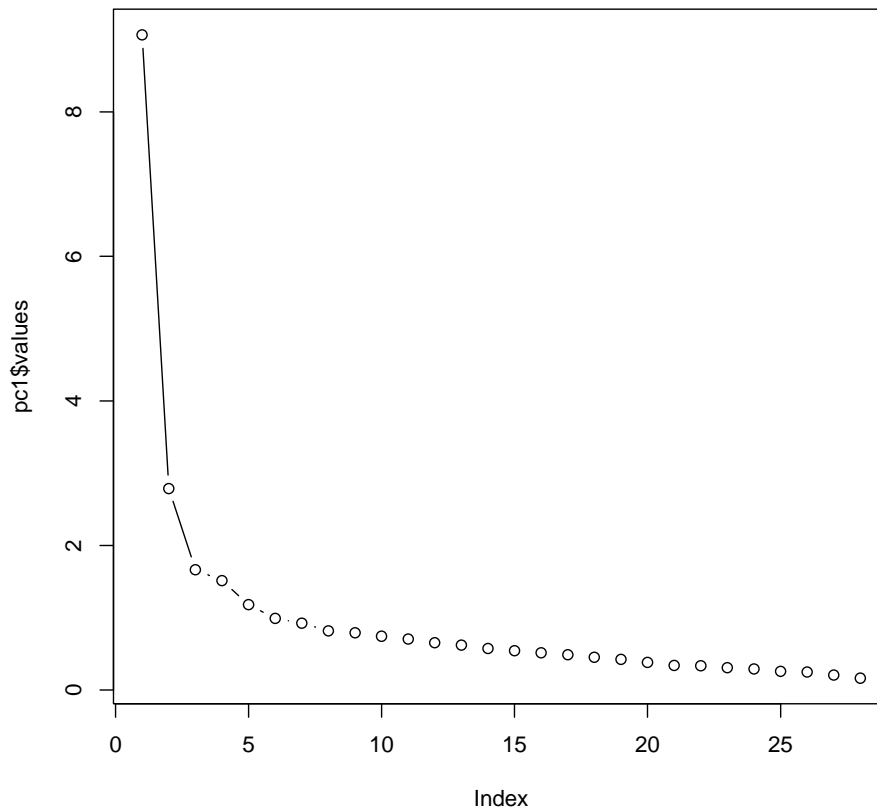
Fit based upon off diagonal values = 0.97

Sample size: MacCallum et al. (1999) have demonstrated that when communalities after extraction are above .5, a sample size between 100 and 200 can be adequate, and even when communalities are below .5 a sample size of 500 should be sufficient. We have a sample size of 239 with some communalities below .5, and so the sample size may not be adequate. However, the KMO measure of sampling adequacy is .894, which is above Kaiser's (1974) recommendation of .5. This value is 'meritorious' according to Hutcheson and Sofroniou (1999). As such, the evidence suggests that the sample size is adequate to yield distinct and reliable factors.

We should also consider the scree plot. As mentioned above, the eigenvalues are stored in a variable called *pc1\$values*, and we can draw a quick scree plot using the *plot()* function, by executing:

```
plot(pc1$values, type = "b")
```

This command simply plots the eigenvalues (y) against the factor number (x). By default, the *plot()* function will plot points (*type="p"*). We want to see a line so that we can look at the trend (we could ask for this by specifying *type="l"*), but ideally we want to look at both a line and points on the same graph, which is why we specify *type="b"*.



The scree plot shows clear inflexions at 3 and 5 factors and so using the scree plot you could justify extracting 3 or 5 factors.

Rotation: You should perform an oblique rotation because the question says that the constructs we're measuring are related. To perform an oblique rotation, we could execute:

```
pc3 <- principal(tossData.2, nfactors = 5, rotate = "oblimin")
```

We can look at the factor loadings from this model in a nice easy-to-digest format by executing:

```
print.psych(pc3, cut = 0.3, sort = TRUE)
```

```
Principal Components Analysis
Call: principal(r = tossData.2, nfactors = 5, rotate = "oblimin")
Standardized loadings based upon correlation matrix
```

item	TC1	TC4	TC2	TC3	TC5	h2	u2
q16	16	0.83				0.68	0.32
q14	14	0.81				0.71	0.29
q22	22	0.74				0.77	0.23
q17	17	0.71				0.71	0.29
q13	13	0.58				0.53	0.47
q8	8	0.54				0.68	0.32
q10	10	0.43			0.41	0.42	0.58
q24	24		0.76			0.65	0.35
q3	3		0.68			0.59	0.41
q4	4		0.60			0.59	0.41
q21	21	0.46	0.48			0.57	0.43
q1	1	0.30	0.46			0.65	0.35
q15	15	0.39	0.46			0.51	0.49
q9	9	0.33	0.37		0.35	0.64	0.36
q19	19			0.73		0.54	0.46
q25	25			0.67		0.55	0.45
q20	20			0.64		0.48	0.52
q27	27			0.57	0.39	0.62	0.38
q26	26	0.39		0.48		0.60	0.40
q7	7		0.38	0.45		0.49	0.51
q11	11	0.30		0.33		0.54	0.46
q23	23				0.72	0.59	0.41

q28	28			0.70		0.54	0.46
q5	5			0.68		0.54	0.46
q12	12			0.53		0.30	0.70
q6	6					0.81	0.62
q2	2					0.75	0.62
q18	18			0.31		0.52	0.51

		TC1	TC4	TC2	TC3	TC5
SS loadings		4.83	3.45	2.91	2.57	2.45
Proportion Var		0.17	0.12	0.10	0.09	0.09
Cumulative Var		0.17	0.30	0.40	0.49	0.58

With factor correlations of

	TC1	TC4	TC2	TC3	TC5
TC1	1.00	0.38	0.23	0.34	0.23
TC4	0.38	1.00	0.26	0.22	0.22
TC2	0.23	0.26	1.00	0.13	-0.09
TC3	0.34	0.22	0.13	1.00	0.20
TC5	0.23	0.22	-0.09	0.20	1.00

Test of the hypothesis that 5 factors are sufficient.

The degrees of freedom for the null model are 378 and the objective function was 13.6

The degrees of freedom for the model are 248 and the objective function was 2.49

The number of observations was 231 with Chi Square = 539.34 with prob < 1.1e-23

Fit based upon off diagonal values = 0.97

Looking at the pattern matrix above (and using loadings greater than .4 as recommended by Stevens) we see the following pattern:

Factor 1:

- Q 16. Thinking about whether to use repeated or independent measures thrills me
- Q 14. I'd rather think about appropriate dependent variables than go to the pub
- Q 22. I quiver with excitement when thinking about designing my next experiment
- Q 17. I enjoy sitting in the park contemplating whether to use participant observation in my next experiment
- Q 13. Designing experiments is fun
- Q 8. I like control conditions
- Q 10. I could spend all day explaining statistics to people

Factor 2:

- Q 19. I like to help students
- Q 25. I love teaching
- Q 20. Passing on knowledge is the greatest gift you can bestow on an individual
- Q 27. I love teaching because students have to pretend to like me or they'll get bad marks
- Q 26. I spend lots of time helping students
- Q 7. Helping others to understand sums of squares is a great feeling

Factor 3:

- Q 23. I often spend my spare time talking to the pigeons ... and even they die of boredom
- Q 28. My cat is my only friend
- Q 5. I still live with my mother and have little personal hygiene
- Q 12. People fall asleep as soon as I open my mouth to speak

Factor 4:

- Q 24. I tried to build myself a time machine so that I could go back to the 1930s and follow Fisher around on my hands and knees licking the floor on which he'd just trodden
- Q 3. I memorize probability values for the F-distribution
- Q 4. I worship at the shrine of Pearson
- Q 21. Thinking about Bonferroni corrections gives me a tingly feeling in my groin
- Q 1. I once woke up in the middle of a vegetable patch hugging a turnip that I'd mistakenly dug up thinking it was Roy's largest root
- Q 15. I soil my pants with excitement at the mere mention of factor analysis

Factor 5:

- Q 6. Teaching others makes me want to swallow a large bottle of bleach because the pain of my burning oesophagus would be light relief in comparison
- Q 2. If I had a big gun I'd shoot all the students I have to teach
- Q 18. Standing in front of 300 people in no way makes me lose control of my bowels

No factor:

- Q 9. I calculate three ANOVAs in my head before getting out of bed every morning
- Q 11. I like it when people tell me I've helped them to understand factor rotation

Factor 1 seems to relate to research methods, factor 2 to teaching, factor 3 to general social skills, factor 4 to statistics and factor 5 to, well, er, teaching again. All in all, this isn't particularly satisfying and doesn't really support the four-factor model. We saw earlier that the extraction of five factors probably wasn't justified. In fact the scree plot seems to indicate three. Let's rerun the analysis but asking **R** for three factors. Let's see how this changes the pattern matrix. We can do this by executing:

```
pc4 <- principal(tossData.2, nfactors = 3, rotate = "oblimin")
print.psych(pc4, cut = 0.3, sort = TRUE)
```

```
Principal Components Analysis
Call: principal(r = tossData.2, nfactors = 3, rotate = "oblimin")
Standardized loadings based upon correlation matrix
```

item	TC1	TC2	TC3	h2	u2
q22	22	0.84		0.70	0.30
q17	17	0.79		0.64	0.36
q9	9	0.76		0.63	0.37
q8	8	0.73		0.67	0.33
q1	1	0.72		0.59	0.41
q13	13	0.70		0.50	0.50
q21	21	0.69		0.52	0.48
q15	15	0.58		0.35	0.65
q4	4	0.58		0.38	0.62
q3	3	0.55	0.35	0.46	0.54
q24	24	0.53		0.42	0.58
q10	10	0.51		0.32	0.68
q16	16	0.51		0.43	0.57
q2	2	0.48	-0.44	0.43	0.57
q6	6	0.48	-0.44	0.35	0.65
q14	14	0.43	0.33	0.42	0.58
q19	19		0.65	0.42	0.58
q26	26		0.65	0.55	0.45
q25	25		0.59	0.47	0.53
q27	27		0.58	0.59	0.41
q20	20		0.49	0.36	0.64
q7	7	0.41	0.48	0.46	0.54
q11	11	0.38	0.44	0.54	0.46
q5	5			0.70	0.53
q23	23			0.70	0.52
q28	28			0.66	0.48
q12	12			0.56	0.29
q18	18	0.38	-0.38	0.39	0.48

```
SS loadings      TC1 TC2 TC3
Proportion Var  7.38 3.44 2.69
Cumulative Var  0.26 0.39 0.48
```

```
With factor correlations of
      TC1 TC2 TC3
TC1  1.00 0.22 0.41
TC2  0.22 1.00 0.05
TC3  0.41 0.05 1.00
```

Test of the hypothesis that 3 factors are sufficient.

The degrees of freedom for the null model are 378 and the objective function was 13.6

The degrees of freedom for the model are 297 and the objective function was 3.3

The number of observations was 231 with Chi Square = 718.84 with prob < 5.8e-37

Fit based upon off diagonal values = 0.96

Looking at the pattern matrix (and using loadings greater than .4 as recommended by Stevens) we see the following pattern:

Factor 1:

- Q 22. I quiver with excitement when thinking about designing my next experiment
- Q 8. I like control conditions
- Q 17. I enjoy sitting in the park contemplating whether to use participant observation in my next experiment
- Q 21. Thinking about Bonferroni corrections gives me a tingly feeling in my groin
- Q 13. Designing experiments is fun
- Q 9. I calculate three ANOVAs in my head before getting out of bed every morning
- Q 3. I memorize probability values for the F -distribution
- Q 1. I once woke up in the middle of a vegetable patch hugging a turnip that I'd mistakenly dug up thinking it was Roy's largest root
- Q 24. I tried to build myself a time machine so that I could go back to the 1930s and follow Fisher around on my hands and knees licking the floor on which he'd just trodden
- Q 4. I worship at the shrine of Pearson
- Q 16. Thinking about whether to use repeated or independent measures thrills me
- Q 15. I soil my pants with excitement at the mere mention of factor analysis
- Q 10. I could spend all day explaining statistics to people
- Q 14. I'd rather think about appropriate dependent variables than go to the pub

Factor 2:


- Q 19. I like to help students
- Q 7. Helping others to understand sums of squares is a great feeling
- Q 11. I like it when people tell me I've helped them to understand factor rotation
- Q 2. If I had a big gun I'd shoot all the students I have to teach (note negative weight)
- Q 6. Teaching others makes me want to swallow a large bottle of bleach because the pain of my burning oesophagus would be light relief in comparison (note negative weight)
- Q 26. I spend lots of time helping students
- Q 25. I love teaching
- Q 20. Passing on knowledge is the greatest gift you can bestow on an individual
- Q 27. I love teaching because students have to pretend to like me or they'll get bad marks

Factor 3:

- Q 5. I still live with my mother and have little personal hygiene
- Q 23. I often spend my spare time talking to the pigeons ... and even they die of boredom
- Q 28. My cat is my only friend
- Q 12. People fall asleep as soon as I open my mouth to speak
- Q 27. I love teaching because students have to pretend to like me or they'll get bad marks
- Q 18. Standing in front of 300 people in no way makes me lose control of my bowels

This analysis is a lot clearer-cut: factor 1 relates to a love of methods and statistics, factor 2 to a love of teaching, and factor 3 to an absence of normal social skills. This doesn't support the original four-factor model suggested because the data indicate that love of methods and statistics can't be separated (if you love one you love the other).

Task 2

- Dr Sian Williams (University of Brighton) devised a questionnaire to measure organizational ability. She predicted five factors to do with organizational ability: (1) preference for organization; (2) goal achievement; (3) planning approach; (4) acceptance of delays; and (5) preference for routine. These dimensions are *theoretically independent*. Williams's questionnaire (Figure 17.10, and reproduced below) contains 28 items using a 7-point Likert scale (1 = strongly disagree, 4 = neither, 7 = strongly agree). She gave it to 239 people. Run a principal components analysis on the data in **Williams.dat**. 

- 1 I like to have a plan to work to in everyday life
- 2 I feel frustrated when things don't go to plan
- 3 I get most things done in a day that I want to
- 4 I stick to a plan once I have made it
- 5 I enjoy spontaneity and uncertainty
- 6 I feel frustrated if I can't find something I need
- 7 I find it difficult to follow a plan through
- 8 I am an organized person
- 9 I like to know what I have to do in a day
- 10 Disorganized people annoy me
- 11 I leave things to the last minute
- 12 I have many different plans relating to the same goal
- 13 I like to have my documents filed and in order
- 14 I find it easy to work in a disorganized environment
- 15 I make 'to do' lists and achieve most of the things on it
- 16 My workspace is messy and disorganized
- 17 I like to be organized
- 18 Interruptions to my daily routine annoy me
- 19 I feel that I am wasting my time
- 20 I forget the plans I have made
- 21 I prioritize the things I have to do
- 22 I like to work in an organized environment
- 23 I feel relaxed when I don't have a routine
- 24 I set deadlines for myself and achieve them
- 25 I change rather aimlessly from one activity to another during the day
- 26 I have trouble organizing the things I have to do
- 27 I put tasks off to another day
- 28 I feel restricted by schedules and plans

First set your working directory to where the **Williams.dat** file is saved and then load in the data by executing:

```
williamsData<-read.delim("Williams.dat", header = TRUE)
```

We have some missing data in our *williamsData* dataframe. To create a new data set (I am going to call the new dataset *williamsData.2*) without missing data we can use the *na.omit()* function and execute:

```
williamsData.2<-na.omit(williamsData)
```

We can run the Bartlett's test by executing:
`cortest.bartlett(williamsData.2)`

For factor analysis to work we need some relationships between variables and if the *R*-matrix were an identity matrix then all correlation coefficients would be zero. Therefore, we want this test to be *significant* (i.e., have a significance value less than .05). A significant test tells us that the *R*-matrix is not an identity matrix; therefore, there are some relationships between the variables we hope to include in the analysis. For these data, Bartlett's test is highly significant, $\chi^2(378) = 2989.77, p < .001$, and therefore factor analysis is appropriate.

```
R was not square, finding R from data
$chisq
[1] 2989.769

$p.value
[1] 0

$df
[1] 378
```

Next we'd also like the KMO. Once you have executed the code of the function itself (you only need to do this once per session and so if you have already executed this function you do not need to execute it again), you can use it by simply entering the name of your dataframe into it and executing:

```
kmo(williamsData.2)

$overall
[1] 0.8940547

$report
[1] "The KMO test yields a degree of common variance meritorious."
```

The results of the KMO test are shown above. For these data the overall value is 0.89, which is great, so we should be confident that the sample size and the data are adequate for factor analysis.

Finally, we'd like the determinant of the correlation matrix. To find the determinant, we use the *det()* function, into which we place the name of a correlation matrix. We haven't computed this matrix for the current data yet (*williamsData*). This is not a problem, though, because we can just get the determinant by putting the *cor()* function for the raw data into the *det()* function:

```
det(cor(williamsData.2))

[1] 1.240294e-06
```

The determinant of the correlation matrix is .00000124, which is smaller than .00001 and, therefore, indicates that multicollinearity could be a problem in these data (although, strictly speaking, because we're using principal components analysis we don't need to worry).

Next we can do the principal component analysis. We are predicting that there will be five factors and that these factors will be unrelated to each other. Therefore, we will conduct principal components analysis with varimax rotation, specifying that **R** extracts five factors. We can do this by executing (I am using the raw data but you can also create this model from the correlation matrix if you have created one, both methods will give you identical results; see the book chapter):

```
pc1 <- principal(williamsData.2, nfactors = 5, rotate = "varimax")
```

This command creates a model called *pc1*, which extracts 5 factors – the same as the number of variables. We have set the rotation method to “*varimax*”, which means that we will carry out varimax factor rotation on these data.

We can look at the results of the principal components analysis by executing its name:

```
pc1
```

The output below shows the results of the first principal components model. The first part of the output is the unrotated loadings; currently these are not interesting, and so to save space I have not included them in the output below.

```
Principal Components Analysis
Call: principal(r = williamsData.2, nfactors = 5, rotate = "varimax")
Standardized loadings based upon correlation matrix
      RC1  RC4  RC2  RC5  RC3  h2  u2
org1  0.41  0.54 -0.09  0.35  0.23  0.65  0.35
```

```

org2  0.09  0.14 -0.04  0.76  0.11  0.62  0.38
org3  0.15  0.67  0.35  0.03  0.00  0.59  0.41
org4  0.07  0.62  0.31  0.32  0.07  0.59  0.41
org6  0.25  0.07 -0.01  0.18  0.67  0.54  0.46
org7  0.08  0.03  0.01  0.78 -0.05  0.62  0.38
org9  0.20  0.40  0.54  0.00 -0.04  0.49  0.51
org10 0.59  0.38  0.36  0.23  0.08  0.68  0.32
org11 0.43  0.47  0.05  0.45  0.17  0.64  0.36
org12 0.44  0.04  0.12  0.45  0.08  0.42  0.58
org13 0.39  0.34  0.43 -0.01  0.29  0.54  0.46
org14 0.03 -0.13  0.14 -0.07  0.51  0.30  0.70
org16 0.59  0.37  0.16  0.13  0.02  0.53  0.47
org17 0.76 -0.09  0.11  0.15  0.29  0.71  0.29
org18 0.45  0.51 -0.13  0.02  0.19  0.51  0.49
org19 0.78  0.08  0.24 -0.03  0.13  0.68  0.32
org20 0.71  0.29  0.17  0.28  0.05  0.71  0.29
org21 0.22  0.10 -0.11  0.59  0.32  0.51  0.49
org22 -0.01 -0.01  0.71 -0.12 -0.12  0.54  0.46
org23 0.10  0.16  0.65  0.13  0.07  0.48  0.52
org24 0.51  0.52  0.19  0.02  0.01  0.57  0.43
org25 0.75  0.34  0.09  0.28  0.08  0.77  0.23
org26 0.02  0.22  0.03  0.29  0.67  0.59  0.41
org27 0.13  0.74  0.24  0.03  0.14  0.65  0.35
org28 0.17  0.12  0.69  0.04  0.18  0.55  0.45
org29 0.41  0.13  0.57 -0.23  0.20  0.60  0.40
org30 0.17  0.21  0.61  0.01  0.41  0.62  0.38
org31 0.13  0.23  0.12  0.03  0.67  0.54  0.46

```

```

                RC1 RC4 RC2 RC5 RC3
SS loadings    4.57 3.45 3.24 2.63 2.32
Proportion Var 0.16 0.12 0.12 0.09 0.08
Cumulative Var 0.16 0.29 0.40 0.50 0.58

```

Test of the hypothesis that 5 factors are sufficient.

The degrees of freedom for the null model are 378 and the objective function was 13.6

The degrees of freedom for the model are 248 and the objective function was 2.49
The number of observations was 231 with Chi Square = 539.34 with prob < 1.1e-23

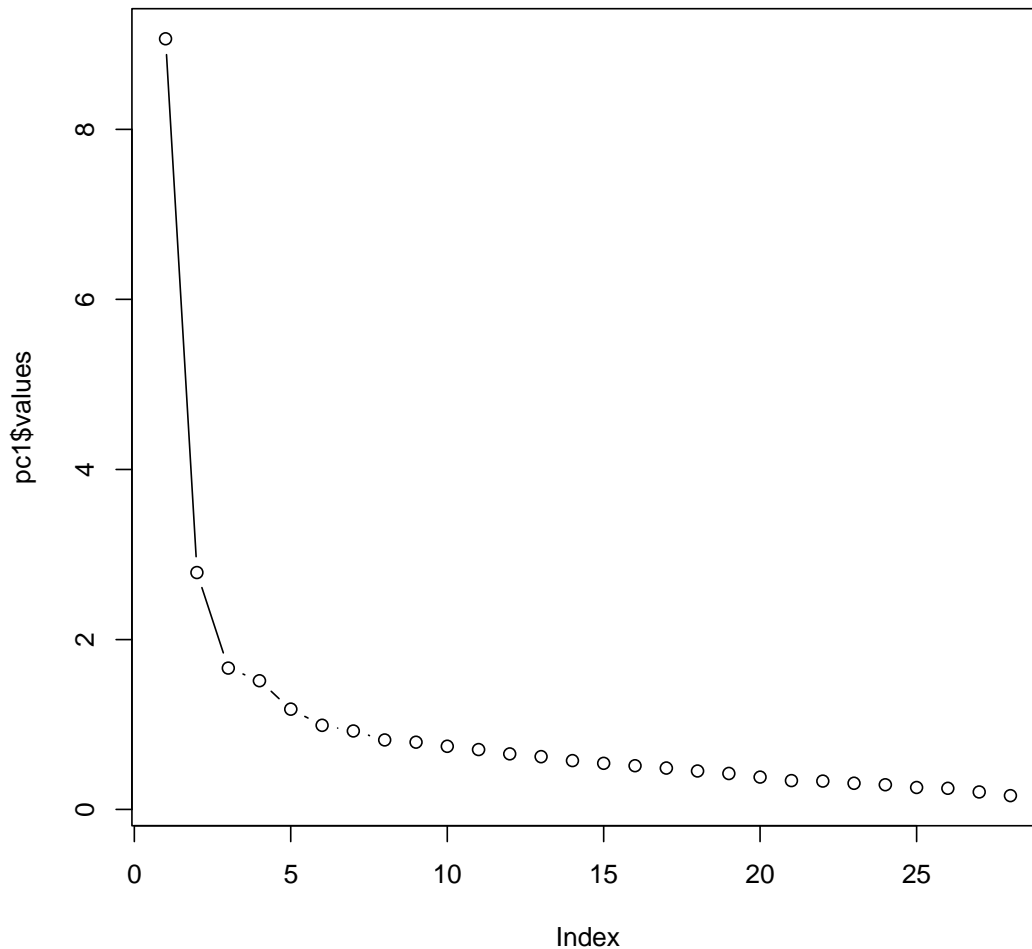
Fit based upon off diagonal values = 0.97

The thing to look at is the eigenvalues. The eigenvalues (SS loadings) associated with each factor represent the variance explained by that particular linear component. (You can also find them in a variable associated with the model called *values*, so in our case we could access this variable using *pc1\$values*). Based on Kaiser's criterion of retaining factors with eigenvalues greater than 1, we would retain five factors. Is this warranted? Kaiser's criterion is accurate when there are less than 30 variables and the communalities after extraction are greater than .7, or when the sample size exceeds 250 and the average communality is greater than .6. For these data the sample size is 239 and the mean communality is .579, so extracting five factors is not really warranted.

We should also consider the scree plot. As mentioned above, the eigenvalues are stored in a variable called *pc1\$values*, and we can draw a quick scree plot using the *plot()* function, by executing:

```
plot(pc1$values, type = "b")
```

This command simply plots the eigenvalues (*y*), against the factor number (*x*). By default, the *plot()* function will plot points (*type="p"*). We want to see a line so that we can look at the trend (we could ask for this by specifying *type="l"*), but ideally we want to look at both a line and points on the same graph, which is why we specify *type="b"*.



The scree plot shows clear inflexions at three and five factors and so using the scree plot you could justify extracting three or five factors.

We can look at the factor loadings from this model in a nice easy-to-digest format by executing:

```
print.psych(pc1, cut = 0.3, sort = TRUE)
```

```
Principal Components Analysis
Call: principal(r = williamsData.2, nfactors = 5, rotate = "varimax")
Standardized loadings based upon correlation matrix
  item  RC1  RC4  RC2  RC5  RC3  h2  u2
org19  16  0.78
org17  14  0.76
org25  22  0.75  0.34
org20  17  0.71
org16  13  0.59  0.37
org10  8  0.59  0.38  0.36
org27  24  0.74
org3   3  0.67  0.35
org4   4  0.62  0.31  0.32
org1   1  0.41  0.54  0.35
org24  21  0.51  0.52
org18  15  0.45  0.51
org11  9  0.43  0.47  0.45
org22  19  0.71
org28  25  0.69
org23  20  0.65
org30  27  0.61  0.41
org29  26  0.41  0.57
org9   7  0.40  0.54
org13  11  0.39  0.34  0.43
org7   6  0.78
org2   2  0.76  0.62  0.38
      0.68 0.32
      0.71 0.29
      0.77 0.23
      0.71 0.29
      0.53 0.47
      0.68 0.32
      0.65 0.35
      0.59 0.41
      0.59 0.41
      0.65 0.35
      0.57 0.43
      0.51 0.49
      0.64 0.36
      0.54 0.46
      0.55 0.45
      0.48 0.52
      0.62 0.38
      0.60 0.40
      0.49 0.51
      0.54 0.46
      0.62 0.38
      0.62 0.38
```

org21	18			0.59	0.32	0.51	0.49
org12	10	0.44		0.45		0.42	0.58
org31	28				0.67	0.54	0.46
org26	23				0.67	0.59	0.41
org6	5				0.67	0.54	0.46
org14	12				0.51	0.30	0.70

		RC1	RC4	RC2	RC5	RC3
SS loadings	4.57	3.45	3.24	2.63	2.32	
Proportion Var	0.16	0.12	0.12	0.09	0.08	
Cumulative Var	0.16	0.29	0.40	0.50	0.58	

Test of the hypothesis that 5 factors are sufficient.

The degrees of freedom for the null model are 378 and the objective function was 13.6

The degrees of freedom for the model are 248 and the objective function was 2.49
The number of observations was 231 with Chi Square = 539.34 with prob < 1.1e-23

Fit based upon off diagonal values = 0.97

Looking at the rotated component matrix (and using loadings greater than .4 as recommended by Stevens) we see the following pattern:

Factor 1 (RC1): preference for organization

- org10: I am an organized person
- org16: I like to have my documents filed and in order
- org17: I find it easy to work in a disorganized environment
- org19: My workspace is messy and disorganized
- org20: I like to be organized
- org25: I like to work in an organized environment
- org12: Disorganised people annoy me

Factor 2 (RC2): goal achievement

- org9: I find it difficult to follow a plan through
- org13: I leave things to the last minute
- org22: I feel that I am wasting my time
- org23: I forget the plans I have made
- org28: I change rather aimlessly from one activity to another during the day
- org29: I have trouble organizing the things I have to do
- org30: I put tasks off to another day

Factor 3 (RC3): preference for routine

- org6: I enjoy spontaneity and uncertainty
- org14: I have many different plans relating to the same goal
- org26: I feel relaxed when I don't have a routine
- org31: I feel restricted by schedules and plans

Note: It's odd that none of these have reverse loadings.

Factor 4 (RC4): plan approach

- org1: I like to have a plan to work to in everyday life
- org3: I get most things done in a day that I want to
- org4: I stick to a plan once I have made it
- org11: I like to know what I have to do in a day
- org18: I make 'to do' lists and achieve most of the things on it
- org24: I prioritize the things I have to do
- org27: I set deadlines for myself and achieve them

Factor 5 (RC5): acceptance of delays

- org2: I feel frustrated when things don't go to plan
- org7: I feel frustrated if I can't find something I need
- org21: Interruptions to my daily routine annoy me

Therefore, it seems as though there is some factorial validity to the structure.

References

- Cattell, R. B. (1978). *The scientific use of factor analysis in behavioral and life sciences*. New York: Plenum.
- Floyd, F. J., & Widaman, K. F. (1995). Factor analysis in the development and refinement of clinical assessment instruments. *Psychological Assessment, 7*, 286–299.
- Gaskell, G. D., Wright, D. B., & O'Muircheartaigh, C. A. (1993). Measuring scientific interest: The effect of knowledge questions on interest ratings. *Journal for the Public Understanding of Science, 2*, 39–57.