National Insurance Case: Factor Analysis

Mitchell J. Lovett

11/1/2020

Preample

In this case we are going to use the National Insurance Case data that is used in class to explore ideas related to dimensionality reduction. We load the data and relevant libraries here. You may need to install the libraries. Some commented out statements might help with the installation of some libraries.

```
dir = "~/Dropbox/Analytics Design/Cases/National Ins"
setwd(dir)
#library(devtools)
#install_github("vqv/ggbiplot")
#install.packages('psych')
library(foreign)
library(ggplot2)
library(ggbiplot)
```

```
## Loading required package: plyr

## Loading required package: scales

## Loading required package: grid
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(psych)
```

```
##
## Attaching package: 'psych'

## The following objects are masked from 'package:scales':
##
##     alpha, rescale
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```r
library(stargazer)
```

```
##
## Please cite as:
```

```
##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.
```

```
##  R package version 5.2.2. https://CRAN.R-project.org/package=stargazer
```

```r
filnm = "national"; #this is the name of the file
natLabData <- read.spss(paste(filnm,".sav",sep=""),to.data.frame=TRUE,use.value.labels=TRUE,trim_values=
```

```
## Warning in read.spss(paste(filnm, ".sav", sep = ""), to.data.frame = TRUE, :
## Undeclared level(s) 2, 3, 4, 5, 6, 7, 8, 9 added in variable: oq
```

```r
natData <- read.spss(paste(filnm,".sav",sep=""),to.data.frame=TRUE,use.value.labels=FALSE);
```

We are going to focus on the first 22 dimensions. These dimensions relate to the perceptions of the firms along various service quality aspects. These dimensions are expected to be highly related. To assist with later developments, we create a variable itemNames with the names of all of the dimensions to be used for the factor analysis.

```r
#obtaining names for the items, constructing subset of data
itemNames = names(natData)[1:22]
summary(natData[,itemNames])
```

```
##        p1              p2             p3            p4              p5
##  Min.   :1.000   Min.   :1.00   Min.   :1.0   Min.   :1.000   Min.   :1.000
##  1st Qu.:4.000   1st Qu.:5.00   1st Qu.:4.0   1st Qu.:5.000   1st Qu.:5.000
##  Median :6.000   Median :6.00   Median :6.0   Median :6.000   Median :6.000
##  Mean   :5.329   Mean   :5.59   Mean   :5.2   Mean   :5.458   Mean   :5.451
##  3rd Qu.:7.000   3rd Qu.:7.00   3rd Qu.:7.0   3rd Qu.:7.000   3rd Qu.:7.000
##  Max.   :7.000   Max.   :7.00   Max.   :7.0   Max.   :7.000   Max.   :7.000
##  NA's   :8       NA's   :7      NA's   :10    NA's   :10      NA's   :19
##        p6              p7              p8              p9             p10
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.00
##  1st Qu.:5.000   1st Qu.:5.000   1st Qu.:5.000   1st Qu.:4.000   1st Qu.:5.00
##  Median :6.000   Median :6.000   Median :6.500   Median :6.000   Median :6.00
##  Mean   :5.828   Mean   :5.423   Mean   :5.884   Mean   :5.142   Mean   :5.42
##  3rd Qu.:7.000   3rd Qu.:7.000   3rd Qu.:7.000   3rd Qu.:7.000   3rd Qu.:7.00
##  Max.   :7.000   Max.   :7.000   Max.   :7.000   Max.   :7.000   Max.   :7.00
##  NA's   :6       NA's   :11      NA's   :9       NA's   :11      NA's   :16
##       p11             p12             p13             p14
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000
##  1st Qu.:5.000   1st Qu.:4.000   1st Qu.:6.000   1st Qu.:5.000
##  Median :5.000   Median :5.000   Median :6.000   Median :6.000
##  Mean   :5.367   Mean   :5.306   Mean   :6.152   Mean   :5.599
##  3rd Qu.:6.000   3rd Qu.:6.000   3rd Qu.:7.000   3rd Qu.:7.000
##  Max.   :7.000   Max.   :7.000   Max.   :7.000   Max.   :7.000
##  NA's   :48      NA's   :53      NA's   :22      NA's   :28
##       p15             p16             p17             p18
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000
##  1st Qu.:5.000   1st Qu.:5.000   1st Qu.:5.000   1st Qu.:5.000
##  Median :6.000   Median :6.000   Median :6.000   Median :6.000
##  Mean   :5.523   Mean   :5.637   Mean   :5.784   Mean   :5.494  
##  3rd Qu.:7.000   3rd Qu.:7.000   3rd Qu.:7.000   3rd Qu.:7.000  
##  Max.   :7.000   Max.   :7.000   Max.   :7.000   Max.   :7.000  
##  NA's   :21      NA's   :18      NA's   :17      NA's   :20     
##       p19             p20            p21             p22       
##  Min.   :1.000   Min.   :1.00   Min.   :1.000   Min.   :1.000  
##  1st Qu.:5.000   1st Qu.:5.00   1st Qu.:6.000   1st Qu.:5.000  
##  Median :6.000   Median :6.00   Median :7.000   Median :6.000  
##  Mean   :5.554   Mean   :5.58   Mean   :6.072   Mean   :5.783  
##  3rd Qu.:7.000   3rd Qu.:7.00   3rd Qu.:7.000   3rd Qu.:7.000  
##  Max.   :7.000   Max.   :7.00   Max.   :7.000   Max.   :7.000  
##  NA's   :18      NA's   :21     NA's   :20      NA's   :22
```

The scales are all from 1 to 7 (as expected), but there are missing data (NA's). We will return to the missing data later.

Core factor analysis idea

```
cor1 = cor(natData[,itemNames],use="complete.obs")
#stargazer(cor1,type = "text")
stargazer(cor1[c(1:4,11:14),c(1:4,11:14)],type = "text") #subset is easier to read!
```

```
## 
## ===================================================
##      p1    p2    p3    p4    p11   p12   p13   p14 
## ---------------------------------------------------
## p1    1    0.785 0.748 0.837 0.354 0.368 0.403 0.363
## p2  0.785    1   0.722 0.759 0.378 0.380 0.530 0.450
## p3  0.748 0.722    1   0.780 0.334 0.336 0.465 0.473
## p4  0.837 0.759 0.780    1   0.394 0.386 0.477 0.449
## p11 0.354 0.378 0.334 0.394    1   0.868 0.451 0.531
## p12 0.368 0.380 0.336 0.386 0.868    1   0.449 0.501
## p13 0.403 0.530 0.465 0.477 0.451 0.449    1   0.629
## p14 0.363 0.450 0.473 0.449 0.531 0.501 0.629   1  
## ---------------------------------------------------
```

Here we select a subset of the correlation matrix to illustrate the idea of how the factors are identified. In the subset of the correlation matrix, we see that p1-p4 have high correlations with each other. However, p11 and p12 are less correlated with p1-p4, but highly correlated with one another. This leads to two relatively distinct groups of variables. Ultimately, these end up being separated in the principle components.

Need for Complete Cases

We noted earlier that there were missing cases in the variables we plan to study. We need to drop these for later analyses. We examine here the impact of using only complete.obs (what we need for these methods).

```
complete.obs = apply(!is.na(natData[,itemNames]),1,all)
table(complete.obs)
```

```
## complete.obs
## FALSE  TRUE 
##    77   208
```

By considering only complete cases, we lose 77 cases. This number is somewhat larger than any one variable, but not too bad given total survey dataset size. We could do a deep dive to see how dropping these cases affects the representativeness of the sample. For our analysis approach here, the main input is correlations. We show below that the correlations do not differ much.

```
itemsComplete = natData[complete.obs,itemNames] #are all items available
cor2 = cor(itemsComplete)
sum(cor1 == cor2)/length(cor1)
```

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

```
cor3 = cor(natData[,itemNames],use="pairwise.complete.obs")
sum(abs(cor2 - cor3)<.05)/length(cor1)
```

```
## [1] 0.9545455
```

```
sum(abs(cor2 - cor3)<.1)/length(cor1)
```

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

Notice cor1 and cor2 give the same answer. But the pairwise complete version (cor3) gives slightly different numbers. Though different, the missing data doesn't affect the overall naure of the relationships, so we move forward with the complete observations.

Identify and set up the interpretation variables

Next, we do some data set up. We identify three variables that we might use to interpret the themes or narrowed set of factors –

- their overall service quality perception (oq)
- whether they recommend the company (rec)
- whether they encountered a problem (prob)

These variables can help interpret the meaning of the dimensions that arise from the analysis. Typically, we will plot summaries of such variables on the principle component plots. In perceptual mapping, these plots are usually segment brand preferences. In these plots aimed to extract themes or narrow the dimensions, we use overall quality perceptions, whether they'd recommend national, and whether they experienced a problem.

First, we create variables that identify the complete set of observations for each of these variables, since we will need complete observations for the later analyses.

```
#Set up variables for interpretation dimensions
complete.obsRec = complete.obs & !is.na(natData$rec)
complete.obsProb = complete.obs & !is.na(natData$prob)
complete.obsOQ = complete.obs & !is.na(natData$oq)
complete.obsAll = complete.obsOQ & complete.obsProb & complete.obsRec
```

Run Principle Component Analysis (PCA)

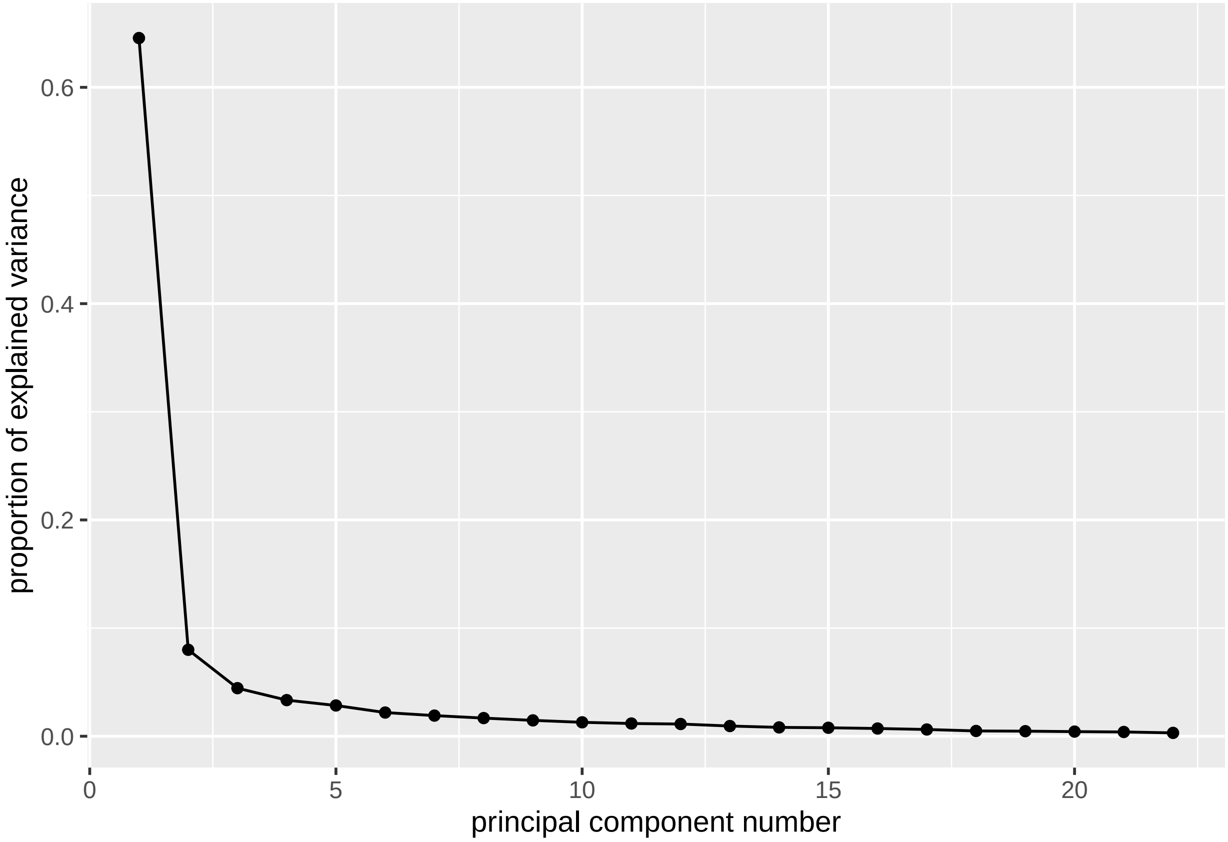
Now we run the analysis via principle components analysis (PCA) via prcomp(). Notice we are subsetting to include only the complete observations for all the variables.

```
resultPCA = prcomp(~.,data=natData[complete.obsAll,itemNames],scale. = TRUE,na.action=na.omit)
```

ScreePlot for Selecting Number of Factors

With PCA, we first plot the screeplot to evaluate how many components to use. To interpret the screeplot, we consider the "elbow rule." The elbow rule suggests to pick the number of factors, principle components or segments based on when the line bends sharply (where the elbow is). Usually we look at the elbow and just after the elbow to see if these are reasonable options. There are other approaches to selecting the number, but for these methods, we will focus on this approach.

```
#resultPCA$group = items[,'BrandPref']?
ggscreeplot(resultPCA) #scree plot looks like there are 2 or 3 factors by the elbow rule
```



Based on the screeplot and the elbow rule, it appears that 2 or 3 factors are options to consider.

Printing the rotations

PCA produces several objects of interest. The first we will discuss is the "rotation," which can be thought of as the new variables (factors or principle components) we are creating.

We print these rotations using the stargazer function from the stargazer library. This function is useful for printing pretty tables easily. Here we print a "text" table.

```
stargazer(resultPCA$rotation[,1:3],type="text")
```

```
##
## ========================
##       PC1    PC2    PC3
## ------------------------
## p1  -0.211 0.130  -0.394
## p2  -0.234 0.112  -0.133
## p3  -0.215 0.074  -0.221
## p4  -0.227 0.086  -0.258
## p5  -0.214 0.010  -0.022
## p6  -0.231 0.060  0.077
## p7  -0.167 -0.124 0.358
## p8  -0.218 0.003  0.247
## p9  -0.229 0.023  -0.119
## p10 -0.226 0.032  0.049
## p11 -0.126 -0.585 -0.266
## p12 -0.126 -0.578 -0.309
## p13 -0.178 -0.236 0.414
## p14 -0.167 -0.368 0.265
## p15 -0.238 0.004  -0.020
## p16 -0.239 0.063  0.008
## p17 -0.239 0.120  -0.003
## p18 -0.235 0.110  0.008
## p19 -0.240 0.117  0.015
## p20 -0.235 0.142  -0.092
## p21 -0.215 0.034  0.256
## p22 -0.221 0.062  0.132
## ------------------------
```

```
#can see the actual rotations here.
```

If you study these numbers closely, you can see the following points about the three principle components we printed:

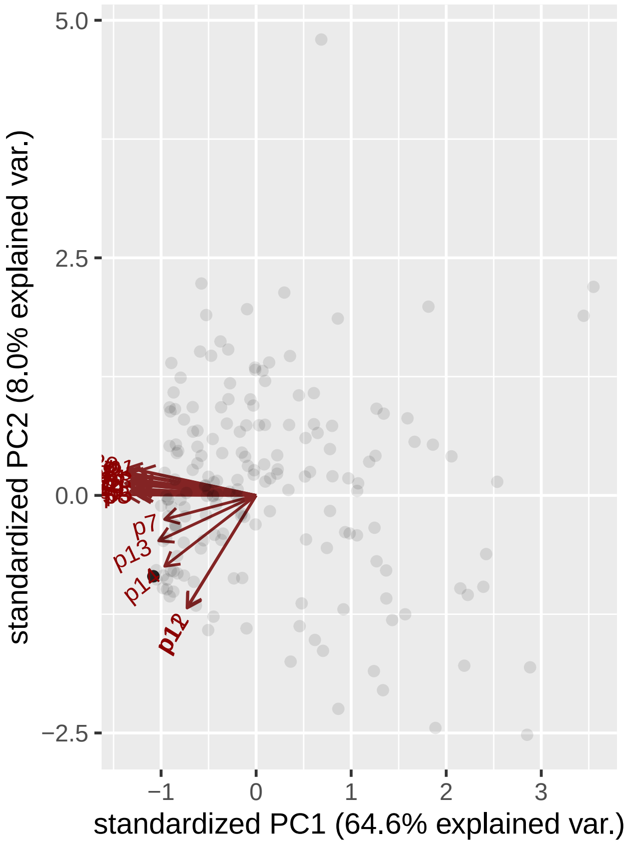
- The first principle component (column) has all variables as negative values. This implies that all variables have the same basic relationship to the first principle component. However, if we look closely, we can see that some variables have a much smaller magnitude (p11-14 and p7).
- The second principle component has most variables close to zero, except p11-p14 and to a much lesser extent p7.
- The third principle component is much more of a mix.

Overall, this suggests that the first two factors largely separate on p11-p14. In looking at the survey questions, p11-p14 relate to the tangibles aspects of the company (modern-looing equipment, visually appealing physical facilities, neat-appearing employees, etc.) and P7 has to do with convenient operating hours.

Plotting pairs of principle components

We now plot these first two factors.

```
ggbiplot(resultPCA,alpha=.1,ellipse=TRUE)
```



In the plot, the horizontal dimension means corresponds to the first principle component (PC1) and the vertical to the second principle component (PC2). These values are standardized so that a value of 1 means 1 standard deviation away and 0 is the mean value.

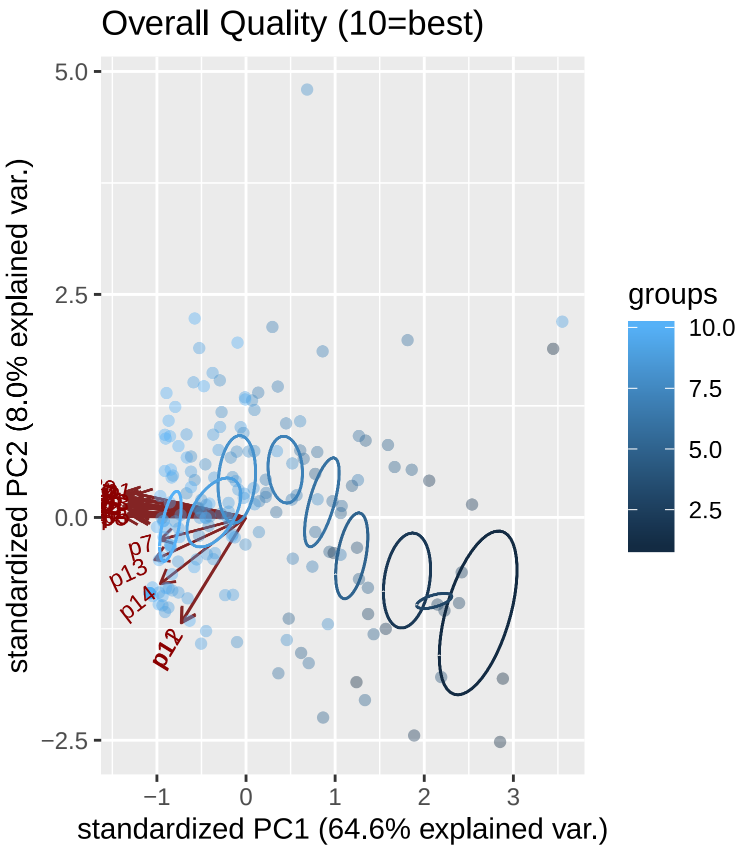
One arrow is present for each of the 22 statements. The horizontal dimension has all of the arrows pointing about the same distance except for the ones pointing down. These arrows correspond to the P11-P14 statements discussed above.

Along the axis, we also see that PC1 explains 65% and PC2 explains 8% of the variation. This indicates that most of the variation is captured in the first component.

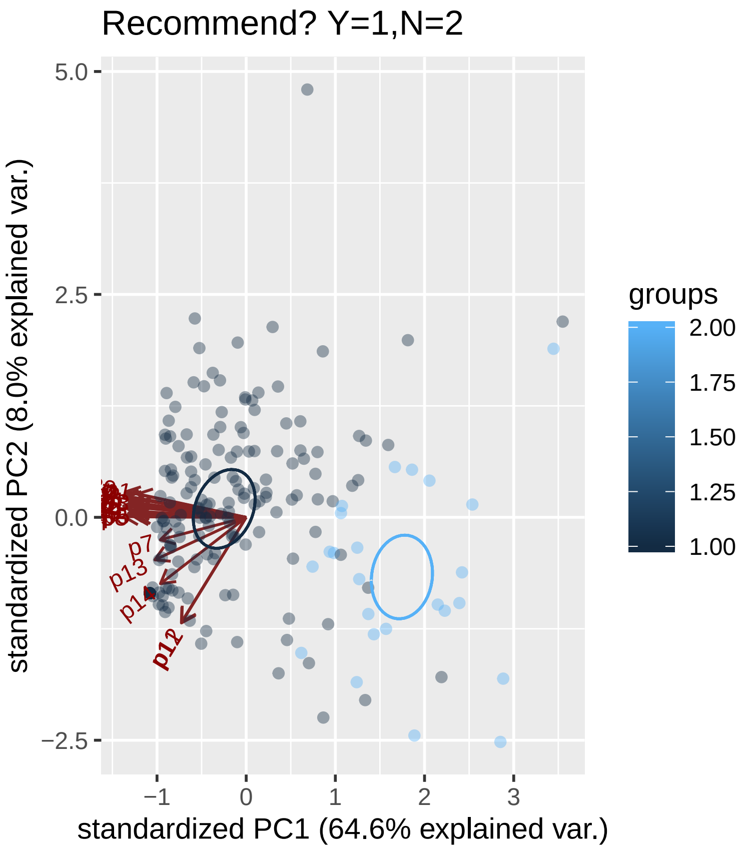
Plotting with interpretation variables added

We now plot these based along with another variable to illustrate how these components relate to other variables. We make one plot for each of the overall sevice quality (oq), whether they would recommend the service (rec), and whether they experienced a problem (prob). These variables are added using the *group* option. Along with this option, we add the option to include ellipses and to make the ellipse probability be .1. You can try adjusting these to see how it affects the plot.

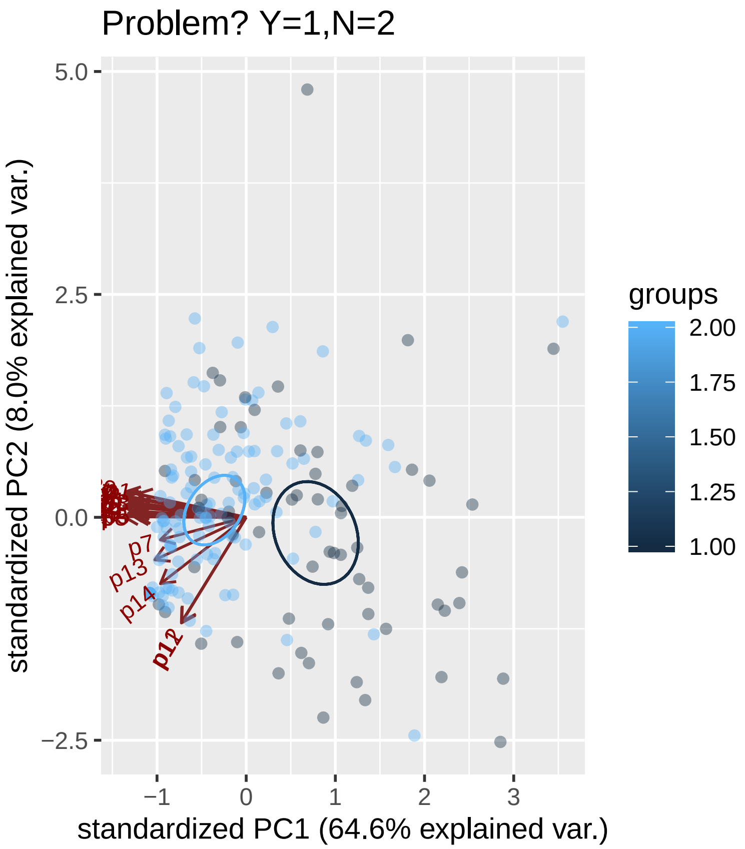
```
gg1 = ggbiplot(resultPCA,alpha=.4,group=natData[complete.obsAll,'oq'],choices=c(1,2),ellipse=TRUE,ellips
gg1 + ggtitle("Overall Quality (10=best)")
```



```
gg2 = ggbiplot(resultPCA,alpha=.4,group=natData[complete.obsAll,'rec'],choices=c(1,2),ellipse=TRUE,elli
gg2 + ggtitle("Recommend? Y=1,N=2")
```



```
gg3 = ggbiplot(resultPCA,alpha=.4,group=natData[complete.obsAll,'prob'],choices=c(1,2),ellipse=TRUE,elli
gg3 + ggtitle("Problem? Y=1,N=2")
```



In these plots the ellipses represent the distribution of values for the variable plotted on the perceptual map. In the first plot this is the oq variable. Each ellipse corresponds to different values of the variables. With dark blue representing low values and light blue representing high values. The center of the ellipse is the mean and the ellipse represent 10% of the distribution. These ellipse plots are visual ways of depicting the relationship the factors have with other variables, which helps in interpretation.

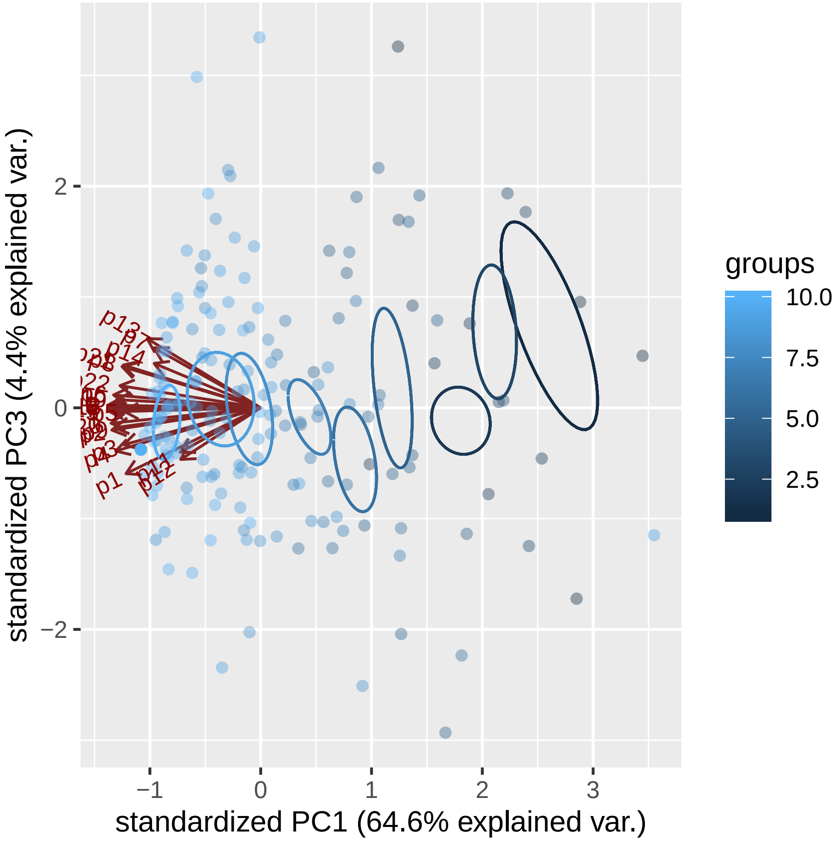
These plots show clearly that overall quality, recommending, and having a problem are really helpful for interpretation. The first factor presents big differences in all three variables. Higher service quality is related to lower values of the first factor (note 10 is best quality). Recommending the service (Yes = 1, No =2) is related to lower values and having a problem (Yes = 1, No = 2) is related to higher values. Thus, the direction of these relationships implies the first factor is related to negative perceptions of the company.

The plots also reveal an interesting feature of the second component. For overall service quality, we see an inverted U shape where the highest overall quality perceptions are lower than the moderately high levels, but the lowest levels are the highest. Those willing to recommend national have higher values than those not willing. Those having a problem have lower values than those not having a problem. Thus, these suggest that higher values the second factor (PC2) are generally related to better overall quality, recommendations, and less problems. Interestingly, given the negative signs on the variables, higher quality here corresponds to lower perceptions of tangibles.

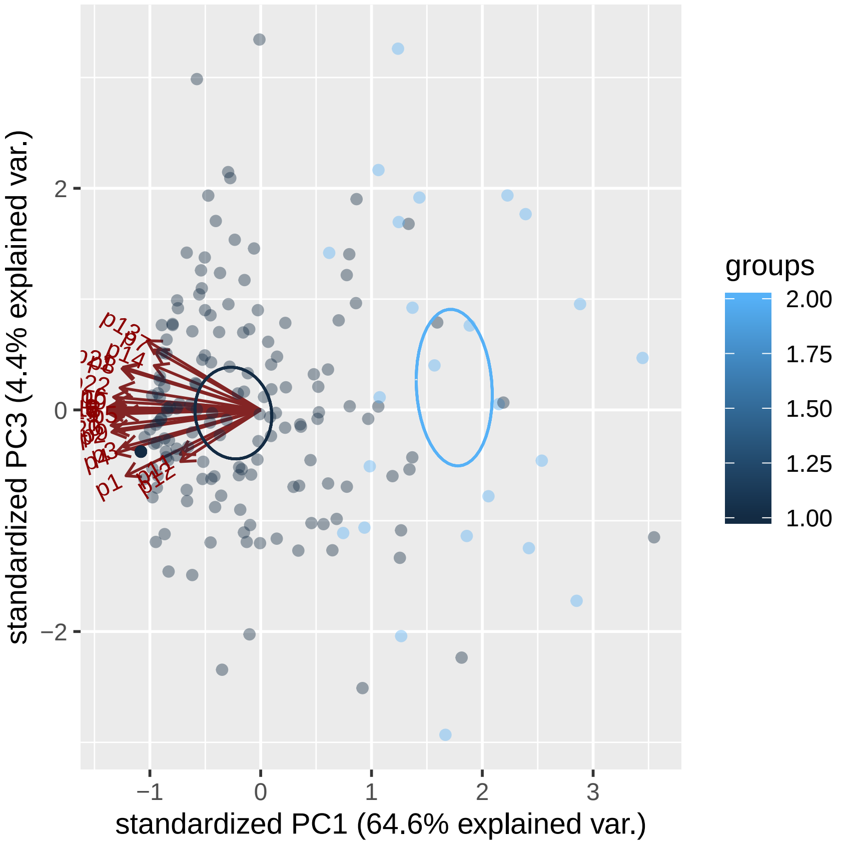
Examining alternative pairs of principle components

We also examined the third factor as depicted in the graphs below. To do so, we change the *choices* option to include PC3 instead of PC2 along with PC1. We find not only that PC3 explains a much smaller portion of the variation, but also that there are not immediately obvious, clear patterns in the items identified for this component.

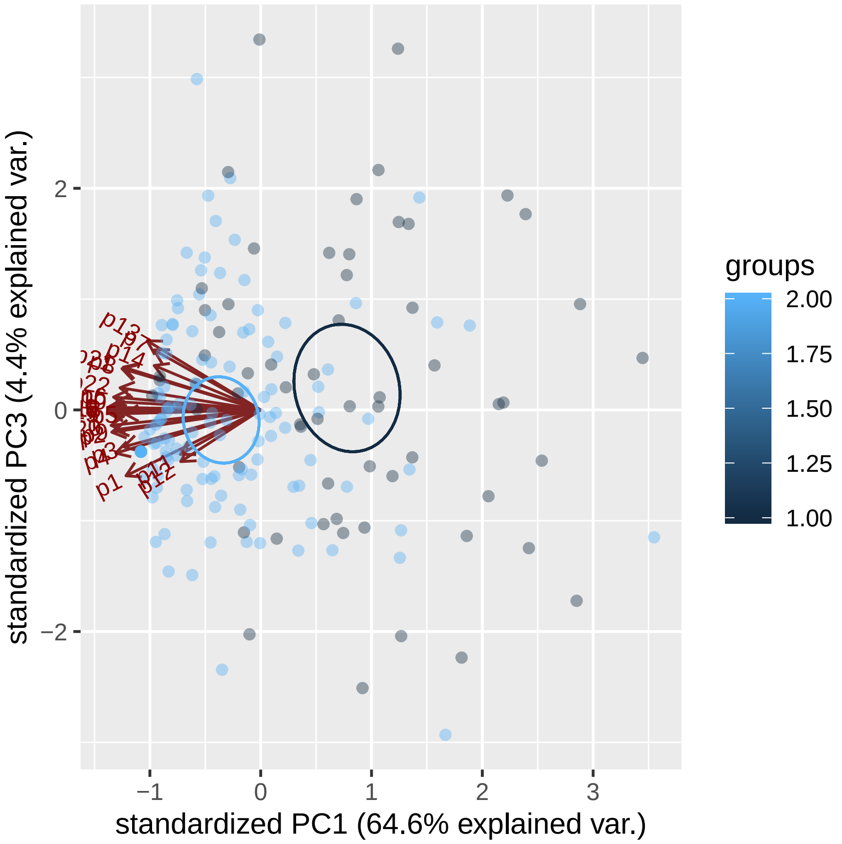
```
##This second set of plots examines what's going on in the third factor
ggbiplot(resultPCA,alpha=.4,group=natData[complete.obsAll,'oq'],choices=c(1,3),ellipse=TRUE,ellipse.prob
```



```
ggbiplot(resultPCA,alpha=.4,group=natData[complete.obsAll,'rec'],choices=c(1,3),ellipse=TRUE,ellipse.pr
```



```
ggbiplot(resultPCA,alpha=.4,group=natData[complete.obsAll,'prob'],choices=c(1,3),ellipse=TRUE,ellipse.p
```



This analysis doesn't reveal much additional insight and the third factor explains only 4% of the variation. Combined with our elbow rule finding, we would likely focus on the first two factors in this case.

Summary

To summarize, our results suggest that most of the dimensions of service quality play a similar role for this company (PC1 explains 64% of variation and loads all negative). Our earlier observations make clear that this first factor represents negative overall perceptions of the company. The second factor is primarily related to tangibles, which have a different relationship. In particular, tangibles has an inverted U relationship with overall quality.

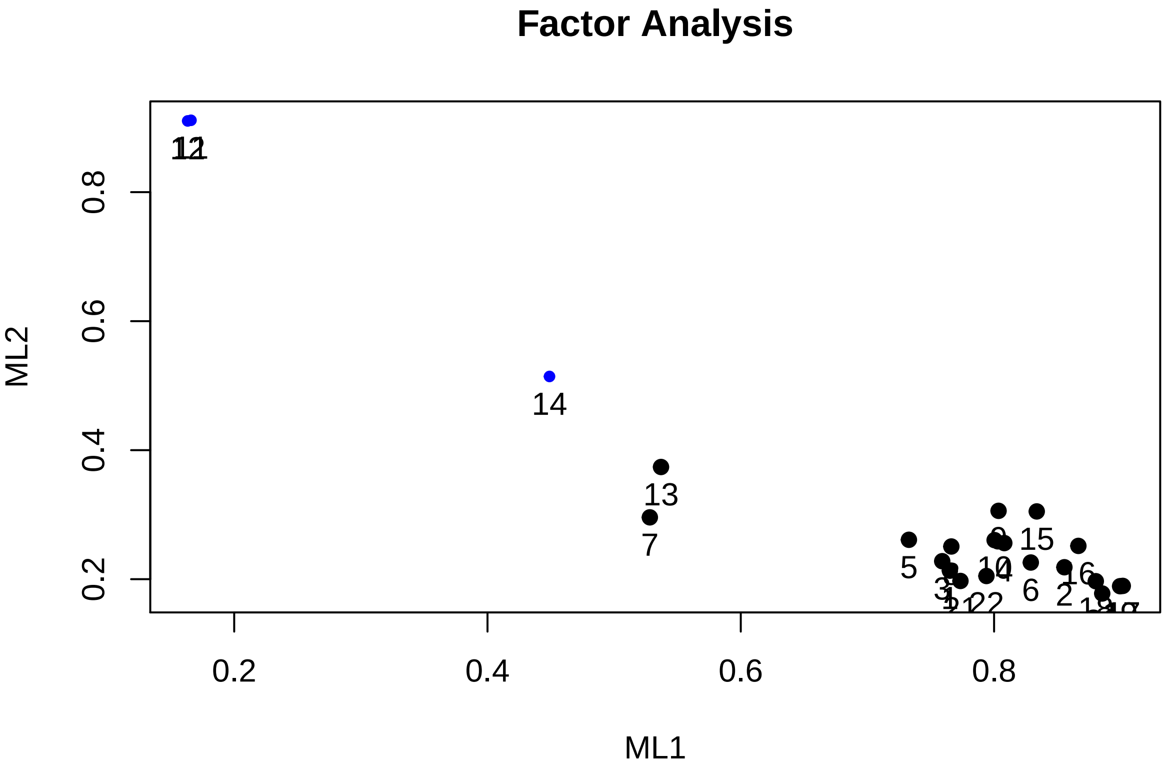
Illustrating Exploratory Factor Analysis

The above analysis focuses on a PCA approach to construct the factor analysis. There are other approaches. We illustrate below with a simple use of one other approach. The below analysis uses a method called maximum likelihood to calibrate the factor analysis. This second approach is optional to learn for class and is more illustrative that factor analysis has a range of methodologies within this class of feature extraction.

For this alternative approach, we also provide the number of factors (nf), the method (fm="ml"), and the rotation approach (rotate="varimax"). Given the number of factors as a constraint, the procedure identifies the factors that explain the data best using a maximum likelihood objective function and then apply the varimax rotation to those factors. This rotation creates a more interpretable picture. Explaining the details of maximum likelihood estimation and this second approach is beyond the scope of the current course and this serves only as an illustration of other possible methods.

```
##Alternative approach is to use maximum likelihood factor analysis
fa2FV = fa(natData[complete.obsAll,itemNames],nfactors=2,fm="ml",rotate="varimax") #common rotation to
```

```
plot(fa2FV)
```



Notice that this two factors solution and the one we found for the PCA solution provide very similar meaning to the two factors. That is, items 11-14 are loaded on to the second factor and the rest on to the first factor.