homework 6

Sam Helms, Charles Wong, Woods Connell, and Hannah Knight

Problem 1

(nothing to turn in here)

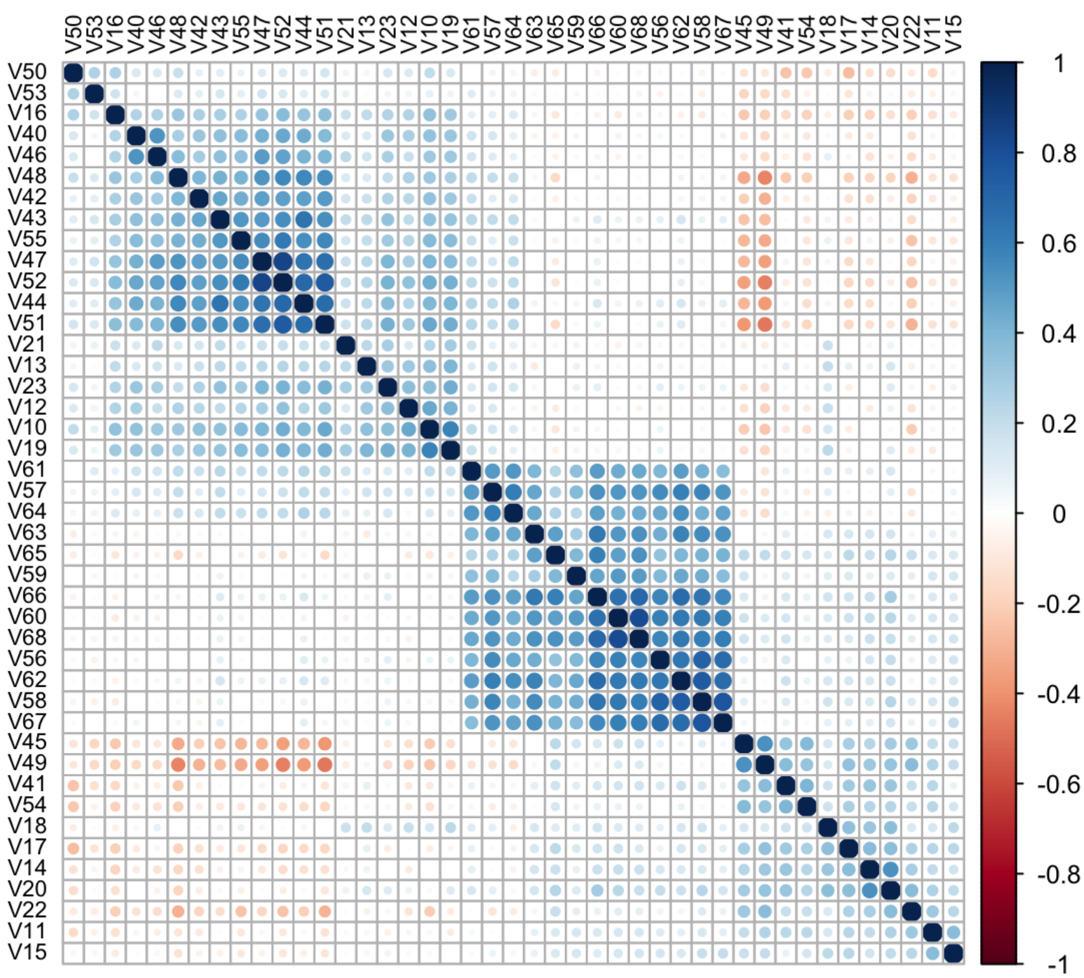
Problem 2

```r
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```r
df <- read.csv('ohiocrime.csv')
df <- sapply( df, as.numeric )

c <- cor(df, use = "pairwise.complete.obs")
corrplot(c, tl.col = "black", number.cex=.7,
         order = "hclust", tl.pos = "lt", tl.cex=.7)
```



Problem 3

Compute KMO or other measure (i.e. just look at matrix produced above) to comment on suitability of data for factor analysis.

Almost all of the KMO values are above 0.9, suggesting that the data is very suitable for factor analysis.

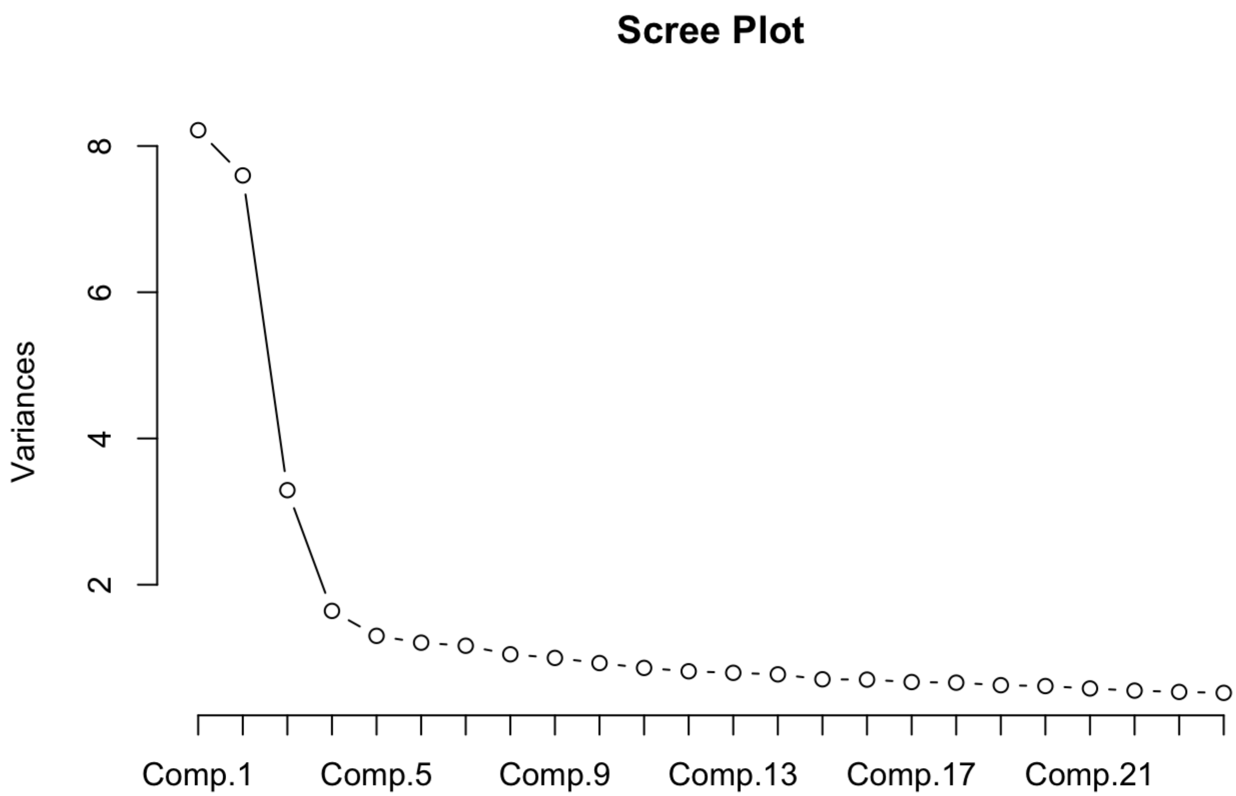
```
## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = c)
## Overall MSA =  0.91
## MSA for each item = 
##  V10  V11  V12  V13  V14  V15  V16  V17  V18  V19  V20  V21  V22  V23  V40 
## 0.91 0.82 0.89 0.88 0.84 0.81 0.95 0.90 0.83 0.91 0.85 0.83 0.89 0.92 0.88 
##  V41  V42  V43  V44  V45  V46  V47  V48  V49  V50  V51  V52  V53  V54  V55 
## 0.86 0.95 0.94 0.94 0.91 0.89 0.89 0.93 0.92 0.74 0.96 0.89 0.67 0.85 0.95 
##  V56  V57  V58  V59  V60  V61  V62  V63  V64  V65  V66  V67  V68 
## 0.95 0.93 0.91 0.93 0.92 0.94 0.94 0.93 0.90 0.91 0.94 0.91 0.90
```

Problem 4

Use Principle Components (or appropriate option in Factor Analysis) to decide on a number of latent factors. You can use Scree Plot, eigenvalue>1, or parallel analysis.

It looks like there is an elbow around component 5. We will use 5 latent factors.

```
fit <- princomp(covmat = c)
screeplot(fit, main='Scree Plot', npcs = 24, type = "lines")
```



Problem 5

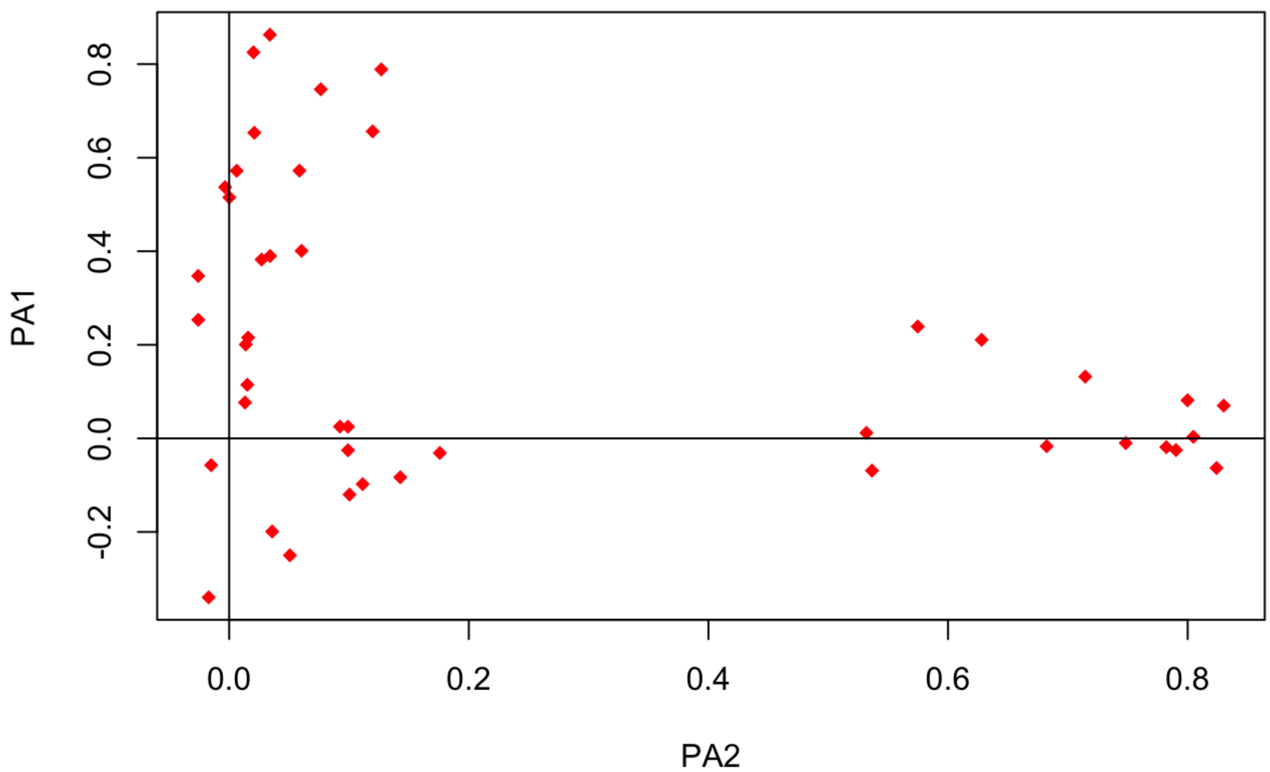
Perform a series of factor analyses using orthogonal models. First, try at least two extraction methods (choose from Principle Components, Principle Axis Factoring, Iterative Principle Components, Maximum Likelihood). Use some method for comparing extraction methods to choose a 'best' method (i.e. RMSR or # residuals greater than .05).

According to the RMSR and the # residuals greater than .05, the iterative PCA does a better job extracting latent factors (both are lower).

Iterative PCA

```
library(psych)
NFACTS <- 5
clean_data <- df[complete.cases(df)]
fact3=fa(df, nfactors=NFACTS, rotate="varimax", SMC=FALSE, fm="pa")

#get loading plot for first two factors
plot(fact3$loadings, pch=18, col='red')
abline(h=0)
abline(v=0)
text(fact3$loadings, labels=names(df))
```




```
#get reproduced correlation matrix
repro3=fact3$loadings%*%t(fact3$loadings)
#residual correlation matrix
resid3=c-repro3

#get root-mean squared residuals
len=length(resid3[upper.tri(resid3)])
RMSR3=sqrt(sum(resid3[upper.tri(resid3)]^2)/len)
RMSR3
```

```
## [1] 0.03241084
```

```
#get proportion of residuals greater than 0.05 in absolute value
sum(rep(1,len)[abs(resid3[upper.tri(resid3)])>0.05])/len
```

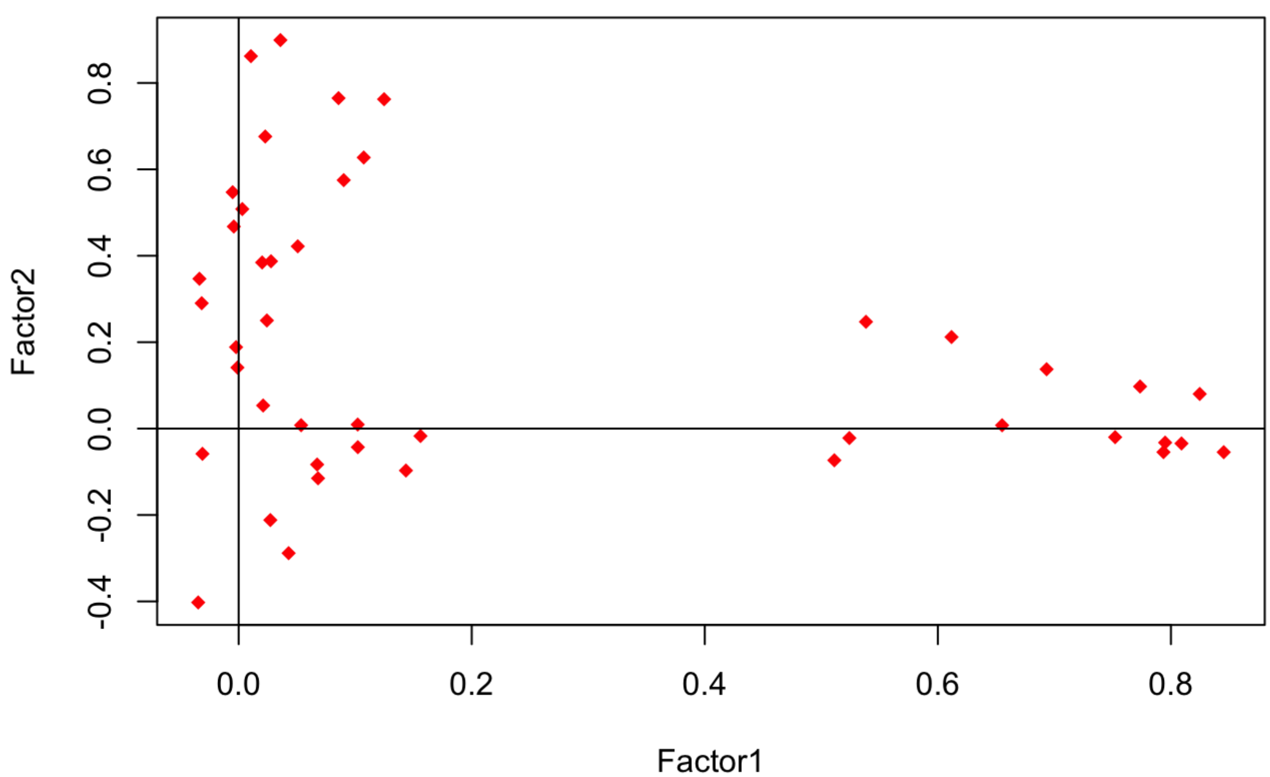
```
## [1] 0.09745293
```

Principal axis factoring

```
############################################################
##  Perform Factor Analysis using Maximum Likelihood (only option in factanal)
##  with Varimax Rotation
############################################################

fact1=factanal(df[complete.cases(df), ],factors=NFACTS,rotation="varimax")

#get loading plot for first two factors
plot(fact1$loadings, pch=18, col='red')
abline(h=0)
abline(v=0)
text(fact1$loadings, labels=names(df))
```




```
#get reproduced correlation matrix
repro1=fact1$loadings%*%t(fact1$loadings)
#residual correlation matrix
resid1=fact1$cor-repro1
#get root-mean squared residuals
len=length(resid1[upper.tri(resid1)])
RMSR1=sqrt(sum(resid1[upper.tri(resid1)]^2)/len)
RMSR1
```

```
## [1] 0.03427399
```

```
#get proportion of residuals greater than 0.05 in absolute value
sum(rep(1,len)[abs(resid1[upper.tri(resid1)])>0.05])/len
```

```
## [1] 0.1162791
```

Problem 6

The below plot suggests that the first principal axis is widely varying in how much influence it has on the variables, while the second principal axis has a very strong influence on a small subset (about a third) of the variables.

```
fit <- fa(df, fm="pa", nfactors=5, rotate="varimax")
load <- fit$loadings[,1:2]
plot(load, type="p", pch=16, col="purple")
abline(h=0)
abline(v=0)
text(load, labels = names(df))
```