DSC 424
Week 2 HW
Daniel O'Brien

1. a.

   Regularized regressions are regressions that use the practice of regularization to control regression coefficients and reduce the variance and minimize sample error. Regularized regressions do this by applying a penalty parameter. This practice minimizes overfitting. The penalty parameter will shrink the beta values and discourage the model from selecting large beta values.

   Lasso and Ridge regression both use penalty functions, but they use two different penalty functions. Both Ridge and Lasso regressions also modify the regression formula, but Lasso regression applies an optimization technique simultaneously. Lasso regressions perform variable selections as well, while Ridge regressions require the user to select the variables manually. Another difference between these two regression methods is that Ridge regression has a significance computation and Lasso regression does not. Ridge regressions add in a lambda value that serves as the penalty parameter adjustment; the lambda minimizes the sum of squares error. The lambda attempts to minimize overfitting, while simultaneously not changing the beta values too much, adding too much of a bias. Lasso regressions try to minimize the residual error, while at the same time, keep the sum of the absolute values of the betas low.

   b.

   There can be different causes of overfitting. Overfitting can occur when there are too many variables for the number of observations. Multicollinearity, which will be addressed in the next question, but can also be a cause of overfitting. Overfitting can also occur when a dataset has sparse data points, but many variables, as addressed in the class lecture from Wednesday, July 22nd. Performing a principal component analysis (PCA) to reduce the shared variance or multicollinearity by creating subgroups can help remedy the issue of overfitting. Creating subgroups to represent our variables can help overfitting by keeping the same dataset, with the same number of observations, but with a reduced number of variables. Another possible solution to overfitting is applying a regularized regression. Overfitting needs to be addressed because overfitting can lead to inaccuracies in predicting the dependent variable. Additionally, inaccurate or illogical beta coefficients can be a result of overfitting.

   c. Multicollinearity is when there are two or more dependent variables that have a very strong correlation with one another. Multicollinearity can lead to misleading beta coefficient values. Multicollinearity can be diagnosed by looking at correlation matrices, correlation tables or the variance inflation factor (VIF). The VIF value of greater than 10 indicates that more than three variables could be highly correlated with one another. When a single variable or multiple variables have a VIF value greater than 10, it is an indication of multicollinearity, and usually means that one of those variables, typically the variable with the highest VIF value, should be removed. However, considering the context of the study is always important, and a variable that is of important context sometimes may remain in the study even if it has a high VIF value, and removing other variables with significant VIF values may be more important.

2. See discussion forum.

3. The researchers applied exploratory factor analysis to determine coping intentions for those who suffered from cyberbullying. The study found young people intended to cope actively with cyberbullying, however, the more individuals were victims or perpetrators of cyberbullying, the more likely they were to use unproductive strategies as well as have higher rates of depression, anxiety and stress.
   An oblique rotation was used to combat multicollinearity. Due to unobserved variables being categorized as complex, the researchers thought that using an oblique rotation would be considered a more accurate method.
   7 factors were used to categorize the variables. Researchers found that there were 7 distinct sub-groups after running exploratory factor analysis. Creating 7 subgroups accounted for 67.57% of variance, and the scree plot also concluded that 7 subgroups would be appropriate. Additionally, the Eigenvalues for all 7 subgroups were all greater than 1.
   The 7 subgroups are active coping, emotion-focused, humor, religion, denial, substance abuse and distraction. Active coping encompasses coping mechanisms that actively try to make changes in the situation of the victims. Variables included in the active coping category include, ‘concentrate my efforts on doing something about the situation I’m in’, ‘get emotional support from others’, ‘take action to make the situation better’, and ‘try to come up with a strategy about what to do’. Each variable in this category describes an action to resolve the situation. The emotion-focused category mostly includes negative feelings towards self or a feeling of helplessness. Variables in the emotion-focused subgroup include, ‘give up trying to deal with it’, ‘criticize myself’, ‘express my negative feelings’, and ‘blame myself for things that have happened’. The variables in this category are emotional responses, but particularly negative and unproductive emotional responses. The humor subgroup, as it sounds, includes variables that attempt to make light of the situation. Variables in the humor subgroup include, ‘Make jokes about it’, and ‘make fun of the situation’. The religion subgroup includes the variables, ‘try to find comfort in my religion or spiritual beliefs’ and ‘pray or meditate’, this subgroup focuses on variables that incorporate faith, religion or spiritual beliefs. The denial subgroup includes variables that indicate the participant doesn’t accept the fact that cyberbullying is actually occurring. The substance abuse subgroup has variables that indicate drug or alcohol abuse as a coping mechanism. And finally, the distraction subgroup includes variables that divert the attention away from what is actually happening, by focusing the mind elsewhere.
   The components were used in a confirmatory factor analysis after being categorized. The factor analysis led to the conclusion that many students use unproductive coping mechanisms when trying to deal with cyberbullying, and even though students may

believe that the strategies that they use are productive, researchers and parents would consider the coping mechanisms used by most students as unproductive. The research concluded that students need to understand the effectiveness of coping strategies, so that they can productively cope and recover from cyberbullying.

4. a. 50 principle components would be needed to account for 100% of total variation for this dataset. The scree plot identified 8 components with Eigenvalues greater than 1. The ‘knee’ of the scree plot indicates that 6 components may be sufficient for our dataset. I would use 6 components.
b. The second component includes variables N1, N2, N3, N5, N6, N7, N8, N9 and N10. The equation is:

$$RC1 = 0.69E1 - 0.73E2 + 0.66E3 + 0.74E5 - 0.63EN6 + 0.75E7 - 0.63E8 + 0.65E9 - 0.7E10 + \varepsilon$$

The variables in the first component focus on socialization or lack thereof. An appropriate name for the first component would be ‘Social’.
c. The highest value for the first component is 2.895, the lowest value is -2.859. The highest value for the second component is 3.208 and the lowest value is -2.991. The highest value for the third value is 2.776, and the lowest value is -2.75. The highest value for the fourth component is 2.425 and the lowest value is -3.196. The highest value for the fifth component is 2.899 and the lowest value is -3.304. The highest value for the sixth component is 2.71 and the lowest value is -3.344.
d. After running a common factor analysis, there are many similarities and differences. We still have our 6 components. And the variables are divided amongst the 6 components almost the exact same, with only a few variables being placed into difference components. The values for the variables are different in the principle component analysis, than they are for the common factor analysis, however the signs on the values of the variables are the same. I do not think that the common factor analysis changes my ability to interpret the results practically, however it does give more information and another perspective to consider when dividing variables into separate components.

R CODE:

```
library(Hmisc) #Describe Function
library(psych) #Multiple Functions for Statistics and Multivariate Analysis
library(GGally) #ggpairs Function
library(ggplot2) #ggplot2 Functions
library(vioplot) #Violin Plot Function
library(corrplot) #Plot Correlations
library(REdaS) #Bartlett's Test of Sphericity
library(psych) #PCA/FA functions
library(factoextra) #PCA Visualizations
library("FactoMineR") #PCA functions
```

```
}



library(ade4) #PCA Visualizations

#Set Working Directory
setwd("/Users/danielobrien/desktop")


#Read in Datasets
bigfive <- read.csv(file="BIG5.csv", header=TRUE, sep=",")

#Check Sample Size and Number of Variables
dim(bigfive)
#19,719 Sample Size 50 varaibles

#Show for first 6 rows of data
head(bigfive)

names(bigfive)

#Check for Missing Values (i.e. NAs)

#For All Variables
sum(is.na(bigfive))
#0 total missing values

#Show Structure of Dataset
str(bigfive, list.len=ncol(bigfive))

corrplot(cor(M,method="spearman"), method = "number", type = "lower")


PCA_Plot = function(pcaData)
{
  library(ggplot2)

  theta = seq(0,2*pi,length.out = 100)
  circle = data.frame(x = cos(theta), y = sin(theta))
  p = ggplot(circle,aes(x,y)) + geom_path()

  loadings = data.frame(pcaData$rotation, .names = row.names(pcaData$rotation))
  p + geom_text(data=loadings, mapping=aes(x = PC1, y = PC2, label = .names, colour =
.names, fontface="bold")) +
    coord_fixed(ratio=1) + labs(x = "PC1", y = "PC2")
}
```

```
PCA_Plot_Secondary = function(pcaData)
{
  library(ggplot2)

  theta = seq(0,2*pi,length.out = 100)
  circle = data.frame(x = cos(theta), y = sin(theta))
  p = ggplot(circle,aes(x,y)) + geom_path()

  loadings = data.frame(pcaData$rotation, .names = row.names(pcaData$rotation))
  p + geom_text(data=loadings, mapping=aes(x = PC3, y = PC4, label = .names, colour =
.names, fontface="bold")) +
    coord_fixed(ratio=1) + labs(x = "PC3", y = "PC4")
}

PCA_Plot_Psyc = function(pcaData)
{
  library(ggplot2)

  theta = seq(0,2*pi,length.out = 100)
  circle = data.frame(x = cos(theta), y = sin(theta))
  p = ggplot(circle,aes(x,y)) + geom_path()

  loadings = as.data.frame(unclass(pcaData$loadings))
  s = rep(0, ncol(loadings))
  for (i in 1:ncol(loadings))
  {
    s[i] = 0
    for (j in 1:nrow(loadings))
      s[i] = s[i] + loadings[j, i]^2
    s[i] = sqrt(s[i])
  }

  for (i in 1:ncol(loadings))
    loadings[, i] = loadings[, i] / s[i]

  loadings$.names = row.names(loadings)

  p + geom_text(data=loadings, mapping=aes(x = PC1, y = PC2, label = .names, colour =
.names, fontface="bold")) +
    coord_fixed(ratio=1) + labs(x = "PC1", y = "PC2")
}

PCA_Plot_Psyc_Secondary = function(pcaData)
{
  library(ggplot2)

  theta = seq(0,2*pi,length.out = 100)
  circle = data.frame(x = cos(theta), y = sin(theta))
  p = ggplot(circle,aes(x,y)) + geom_path()

  loadings = as.data.frame(unclass(pcaData$loadings))
  s = rep(0, ncol(loadings))
  for (i in 1:ncol(loadings))
  {
    s[i] = 0
    for (j in 1:nrow(loadings))
      s[i] = s[i] + loadings[j, i]^2
    s[i] = sqrt(s[i])
  }

  for (i in 1:ncol(loadings))
    loadings[, i] = loadings[, i] / s[i]

  loadings$.names = row.names(loadings)

  print(loadings)
  p + geom_text(data=loadings, mapping=aes(x = PC3, y = PC4, label = .names, colour =
.names, fontface="bold")) +
    coord_fixed(ratio=1) + labs(x = "PC3", y = "PC4")
}

#Test KMO Sampling Adequacy

library(psych)
KMO(bigfive)
#Overall MSA =  0.91

#Test Bartlett's Test of Sphericity
library(REdaS)
bart_spher(bigfive)
#p-value < 2.22e-16 (Very Small Number)


#Test for Reliability Analysis using Cronbach's Alpha
library(psych)
alpha(bigfive,check.keys=TRUE)
#raw_alpha = 0.88
```

```
#Dependent upon sample size, correlation coefficient, and how items fall on
#components
library(psych)
comp <- fa.parallel(bigfive)
comp

#Create PCA
p = prcomp(bigfive, center=T, scale=T)
p



#Check Scree Plot
plot(p)
abline(1, 0)

#Check PCA Summary Information
summary(p)
print(p)

#Check PCA visualizations
plot(p) #Scree Plot
PCA_Plot(p) #PCA_plot1
PCA_Plot_Secondary(p) #PCA_Plot2
biplot(p) #Biplot

#Calculating the Varimax Rotation Loadings manually
rawLoadings = p$rotation %*% diag(p$sdev, nrow(p$rotation), nrow(p$rotation))
print(rawLoadings)
v = varimax(rawLoadings)

#Options available under varimax function
ls(v)
v

#Best Way to Conduct PCA Analysis

p2 = psych::principal(bigfive, rotate="varimax", nfactors=6, scores=TRUE)
p2
print(p2$loadings, cutoff=.4, sort=T)

#PCAs Other Available Information

ls(p2)
```

```
p2$values
p2$communality
p2$rot.mat

#Calculating scores

scores <- p2$scores
scores_1 <- scores[,1]

min_score <- min(scores_1)
min_score

max_score <- max(scores_1)
max_score

summary(scores_1)
summary(scores_2)
summary(scores_3)
summary(scores_4)
summary(scores_5)
summary(scores_6)

#Factor Analysis
fit = factanal(bigfive, 6)
print(fit$loadings, cutoff=.4, sort=T)
summary(fit)
```