CA2

Chris Eason, Dongook Kim, Rongyu Zhang

The file CA2.csv contains 100 observations on 12 unknown variables. Consider this as some data matrix X. Using Singular Value Decomposition, find lower rank approximations of X for all ranks from 1 – 12

```
X <- read.csv("CA2.csv")
```

For each approximation $\tilde{X}_k$ of rank k, calculate the error, $\Delta_k = X - \tilde{X}_k$

```
s <- svd(X)
svd_approx <- function(rank){
  app <- s$u[,1:rank] %*% as.matrix((diag(s$d))[1:rank, 1:rank]) %*% t(s$v[, 1:rank])
  colnames(app) <- c("V1", "V2", "V3", "V4", "V5", "V6", "V7", "V8", "V9", "V10", "V11", "V1
  return(app)
}

X_k <- list()
for (i in 1:12){
  X_k[[i]] <- svd_approx(i)
}

# For each Approximation of X in Rank K, calculate delta the error
delta <- list()
for(i in 1:12){
  delta[[i]] <- X - X_k[[i]]
}
```

Question 1

Consider the rank 4 approximation. Report the mean vector of the approximation error, i.e. Δ_4

```r
library(knitr)
mean_delta <- colMeans(delta[[4]])
kable(as.matrix(round(mean_delta,3)),
      col.names = "Mean_Error",
      caption = "Mean Vector of the Rank 4 Approximation")
```

Table 1: Mean Vector of the Rank 4 Approximation

	Mean_Error
V1	0.012
V2	-0.011
V3	0.132
V4	0.015
V5	-0.042
V6	0.012
V7	0.010
V8	-0.021
V9	0.061
V10	-0.043
V11	0.138
V12	-0.038

Question 2

Compare the correlation matrix of X with that of $\tilde{X}_2$and briefly interpret.

```r
library(ggcorrplot)
```

```
Loading required package: ggplot2
```

```r
library(patchwork)

x_cor <- round(cor(X), 1)
p1 <- ggcorrplot(x_cor, "square", lab = TRUE,
                 title = "Correlation Coefficient Plot
For The Original X Matrix", lab_size = 3) +
```

```
  labs(caption = "Figure 2") +
  theme(
      axis.text.x = element_text(size = 8),
      axis.text.y = element_text(size = 8)
  )

x_2_cor <- round(cor(X_k[[2]]), 1)
p2 <- ggcorrplot(x_2_cor, "square", lab =TRUE,
                 title = "Correlation Coefficient Plot
For The Approximated X Matrix Using Rank 2", lab_size =2)+
  labs(caption = "Figure 3") +
  theme(
      axis.text.x = element_text(size = 8),
      axis.text.y = element_text(size = 8)
  )

p1
```

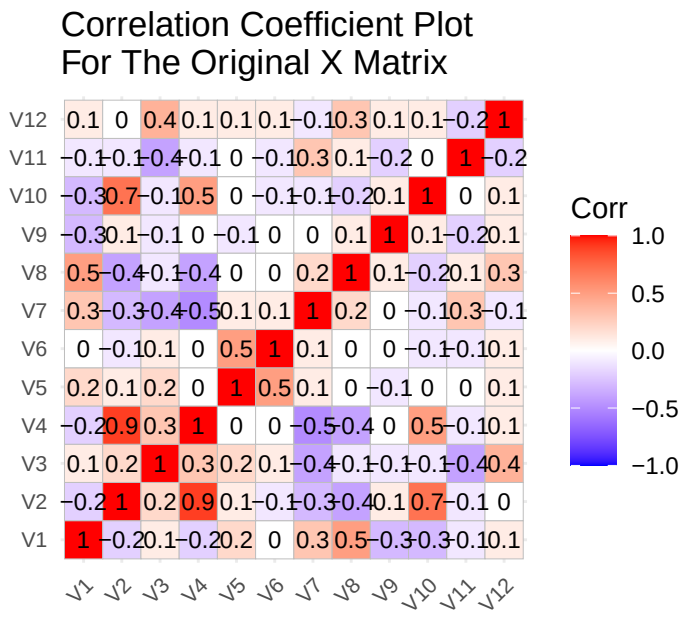

Figure 2

```
p2
```

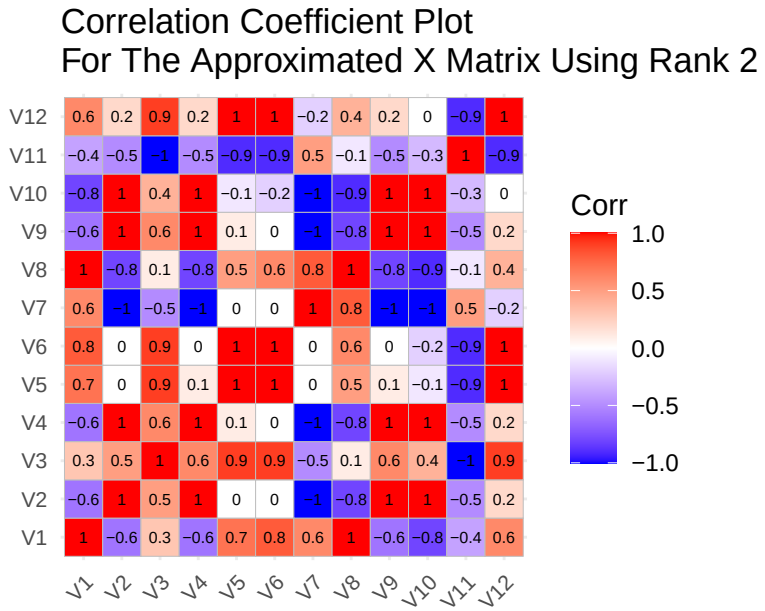

Figure 3

Interpretation:

1. The original data shows that most of the variables are uncorrelated, the rank 2 approximation shows that they are corerlated. For example with V6 and V8 in the original matrix are uncorrelated while the rank 2 approximation correlation coefficient is 0.6.
2. The original data shows slight correlations between variables with each other while the rank 2 approximation of X exagerates these correaltions. For example with V12 and V11 in the original matrix is -0.2 while the rank 2 approximation is -0.9.
3. The rank 2 approximation captures the 2 largest singular values and therefore attempts to capture a majority of the variation in the original X matrix but discards the other variation captured by the other variables . Therefore the rank 2 approximation has altered some of the structures in the data and has affected the relationships between variables, changing their correlation coefficients.

Question 3

Calculate the Frobenius norm, defined as

$$||A||_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{m}|aij|^2}$$

for Δ_k, k =1,2....12. Plot the Frobenius norm as a function of k and briefly describe your findings.

```
FN <- c()
Frobenius_Norm <- function(rank){
   FN[rank]<- sqrt(sum(delta[[rank]]^2))
}

for(i in 1:12){
  FN[i] <- Frobenius_Norm(i)
}

df <- data.frame(x = 1:12, y = FN)
library(ggplot2)
ggplot(df, aes(x=x, y=y))+
  geom_point(color="red") +
  geom_line() +
  labs (
    title = "Frobenius Norm Plot For Delta Error of
Rank k = 1...12 approximations of X",
    x = "Rank k",
    y = "Frobenius Norm",
    caption = "Figure 4"
  )
```

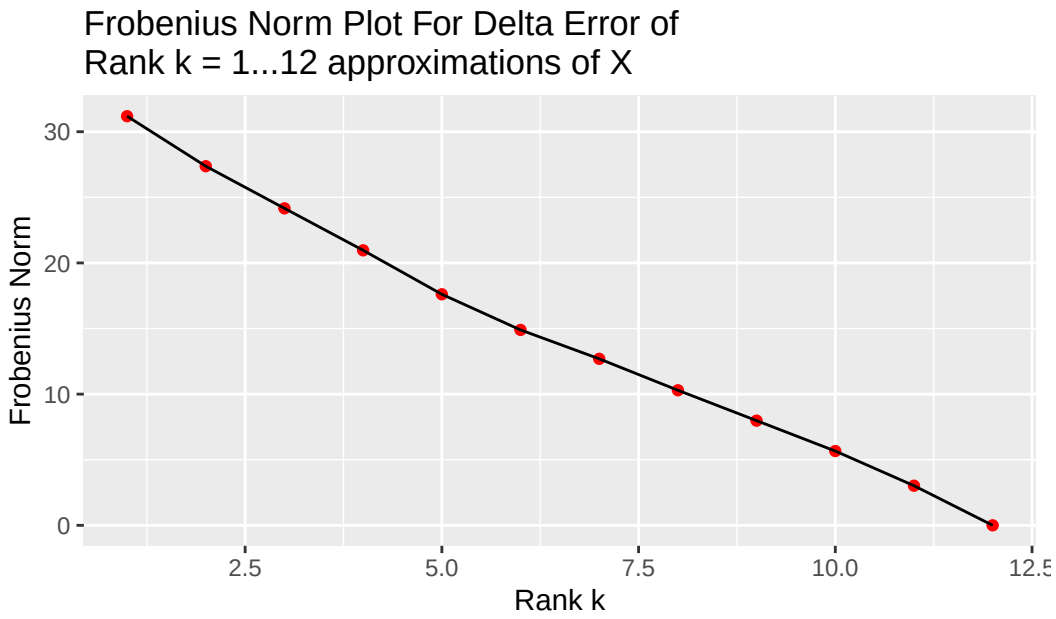

Figure 4

Interpretation:

1. Decreasing (almost linear) trend of the Frobenius Norm as the the approximation of rank K increases.
2. In other words, as the rank of the approximated matrix increases and more eigenvalues are included, the delta error between the original matrix and the rank k approximation decreases, approaching zero at rank 12, where the approximation matches the original dataset exactly.

Question 4

Plot the percentage of the total variation in X retained in $\tilde{X}_k$ for K =1,...12. Again, briefly interpret.

```
total_var <- sum(s$d^2)
retained_var <- c()

for (i in 1:12){
  retained_var[i] <- ((sum(s$d[1:i]^2)) / total_var) * 100
```

```
}

df <- data.frame(x = 1:12, y = retained_var)
library(ggplot2)
ggplot(df, aes(x=x, y=y))+
  geom_point(color="red")+
  geom_line() +
  labs (
    title = "Cumulative Percentage of the total variation in X
that is retained for each rank K approximation of X",
    x = "Rank k",
    y = "Cumulative Percentage of Variance Retained (%)",
    caption = "Figure 5"
  )
```

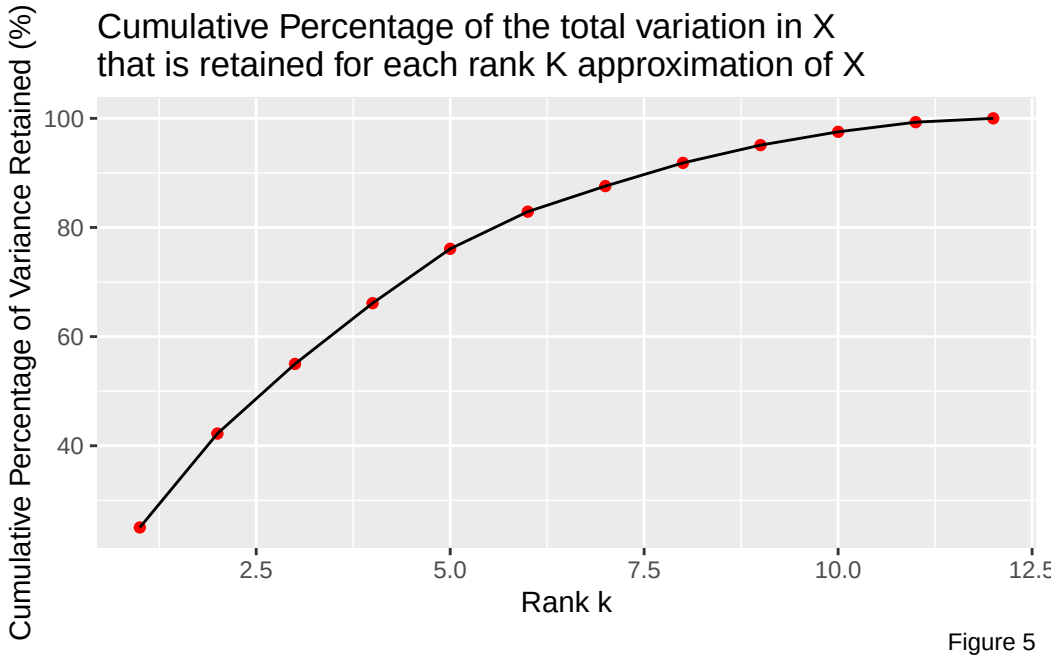

Figure 5

```
print(paste("Percentage of variance retained rank 8:", round(retained_var[8],2)))
```

```
[1] "Percentage of variance retained rank 8: 91.83"
```

Interpretation

1. The amount of variation in the original matrix retained increasing at a decreasing rate by the rank k approximation as the the approximation of rank K increases.
2. This makes sense as we use more eigenvalues to approximate our original matrix, we keep more and more of the variation in the original matrix until the approximation and the original matrix are equal.
3. Shows that by the time the rank k = rank 8, the approximation basically retains 92% of the variation of in the original matrix. Therefore using rank 8 approximation captures most of the data structures in the original matrix while being in lower dimensions and removing some of the linear dependencies between variables.