

2024 SISBID Dimension Reduction Demo

Genevera I. Allen & Yufeng Liu

PCA Demo Using Digits Data

Load Packages

```
library(ggplot2)
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg      ggplot2
```

Load Digits Data

```
#code for digits - ALL
rm(list=ls())
load("UnsupL_SISBID_2024.Rdata")
```

Create Subset of just 3's and 8's

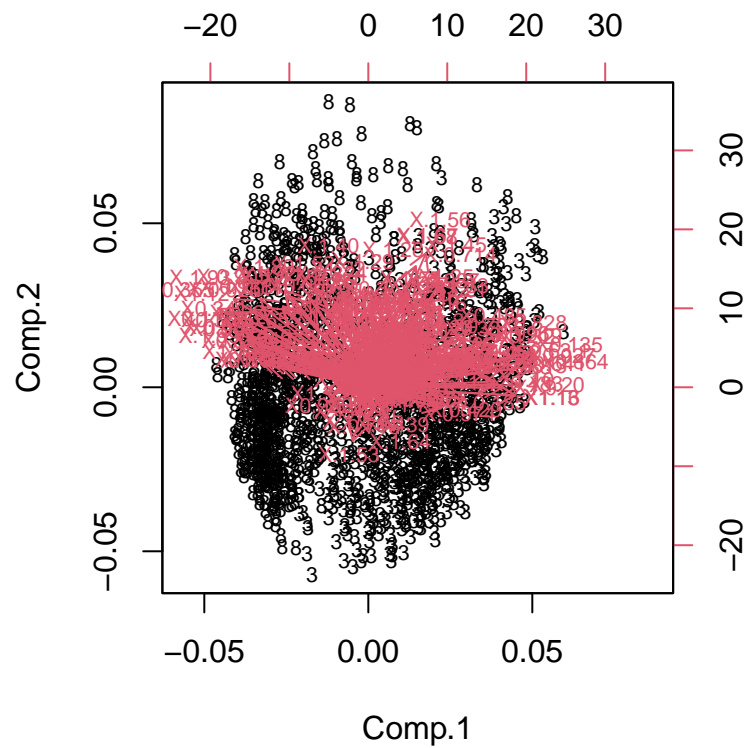
```
dat38 = rbind(digits[which(rownames(digits)==3),],digits[which(rownames(digits)==8),])
```

Try Princomp

With Centering & Scaling

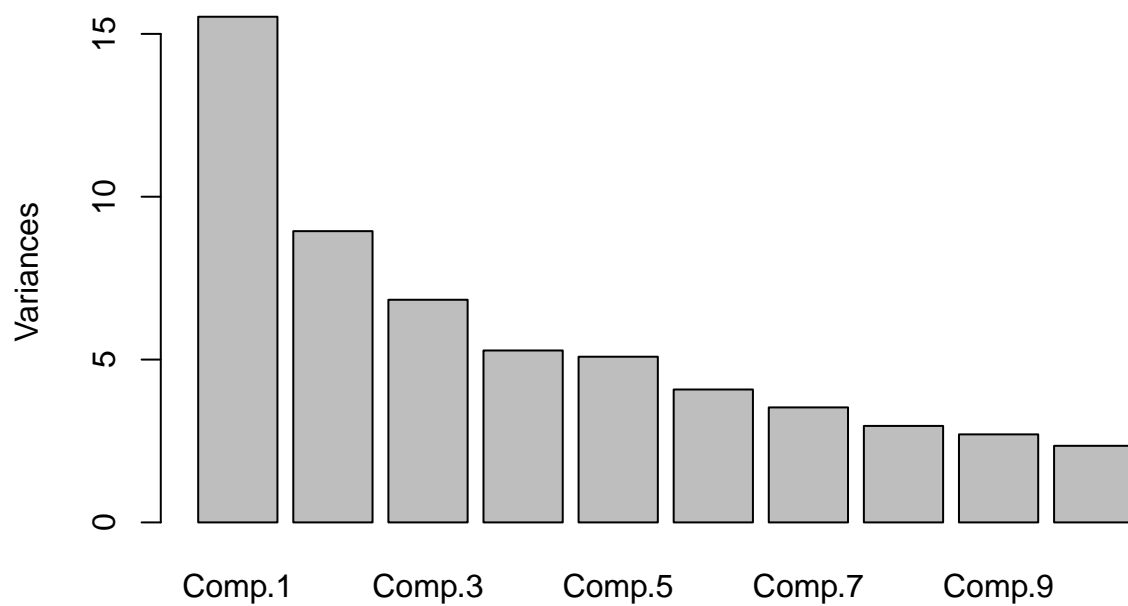
```
pc = princomp(dat38) #default - centers and scales
biplot(pc,cex=.7)
```

```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```



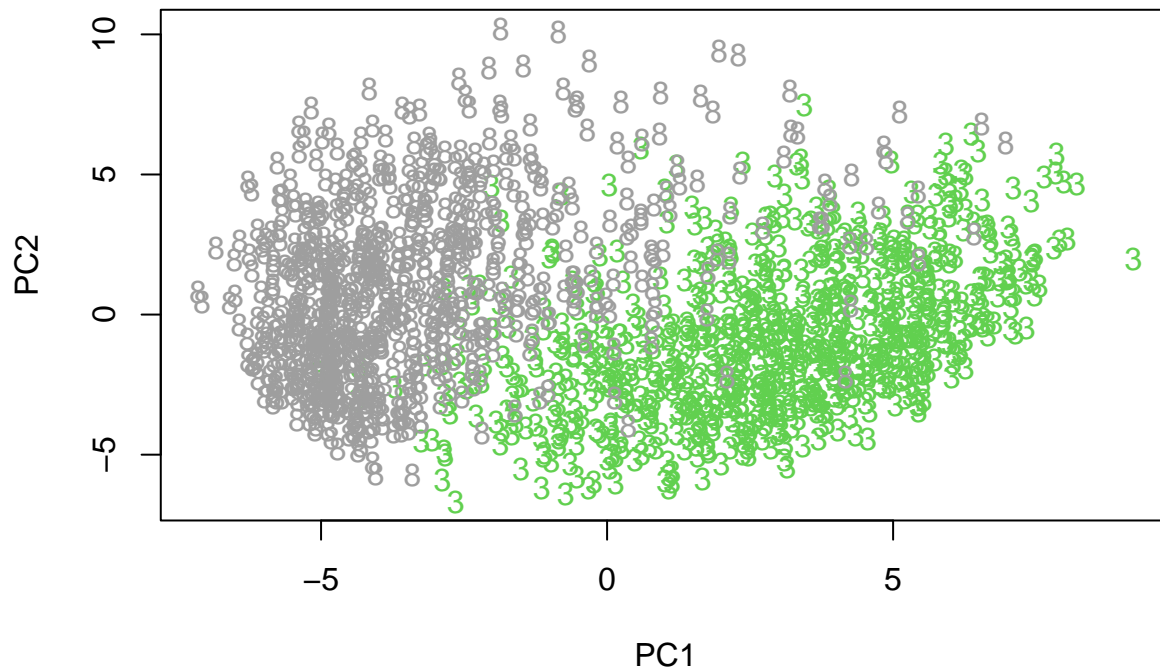
```
screepplot(pc)
```

pc



PC Scatterplot

```
PC1 <- as.matrix(x=pc$scores[,1])
PC2 <- as.matrix(x=pc$scores[,2])
plot(PC1,PC2,type="n",xlab="PC1",ylab="PC2")
text(PC1,PC2,rownames(dat38),col=rownames(dat38))
```



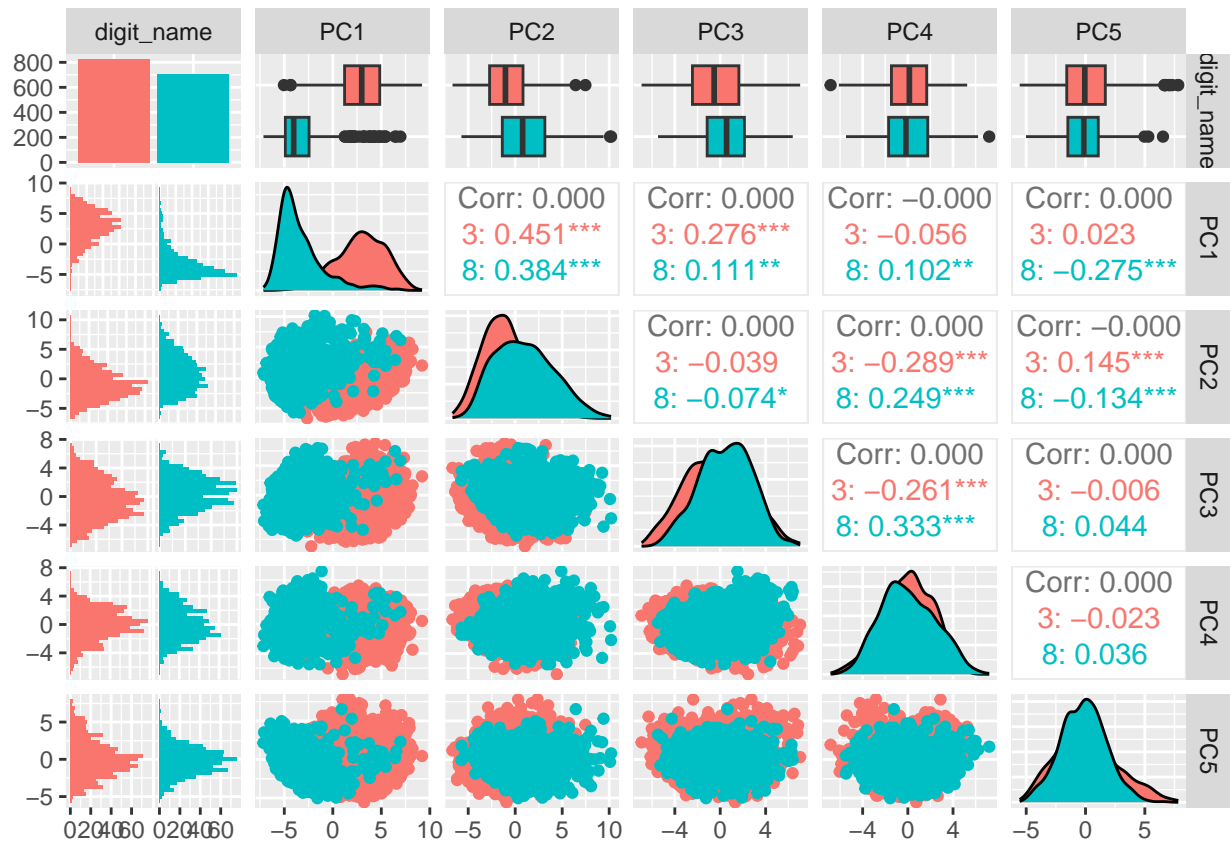
Pairs Plot Using ggpairs

```
PC1 <- as.matrix(x=pc$scores[,1])
PC2 <- as.matrix(pc$scores[,2])
PC3 <- as.matrix(pc$scores[,3])
PC4 <- as.matrix(pc$scores[,4])
PC5<-as.matrix(pc$scores[,5])

pc.df.digits <- data.frame(digit_name = row.names(dat38), PC1, PC2,PC3, PC4, PC5)

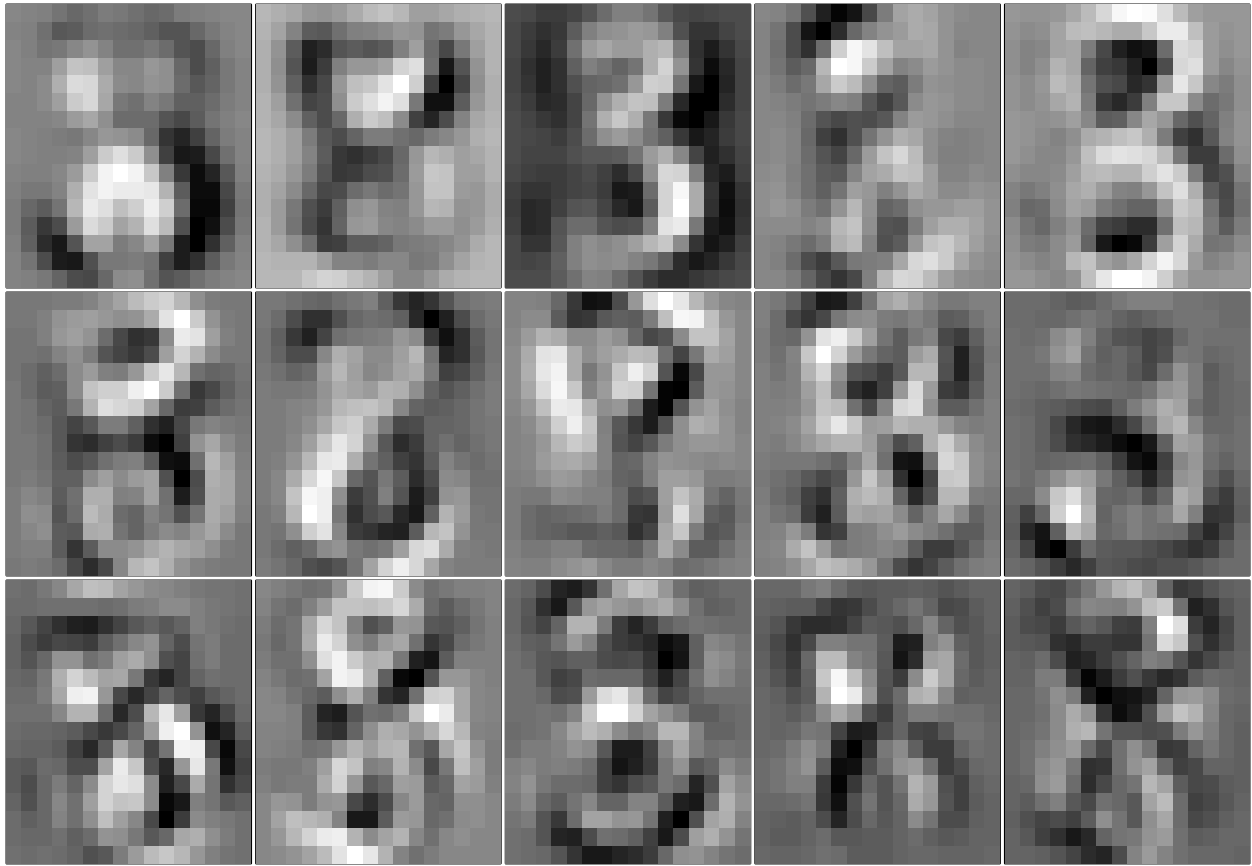
ggpairs(pc.df.digits, mapping = aes(color = digit_name))

## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```



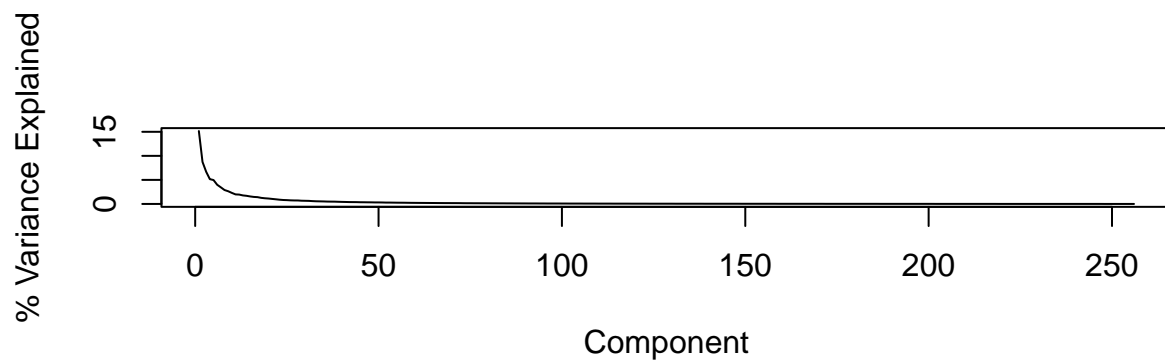
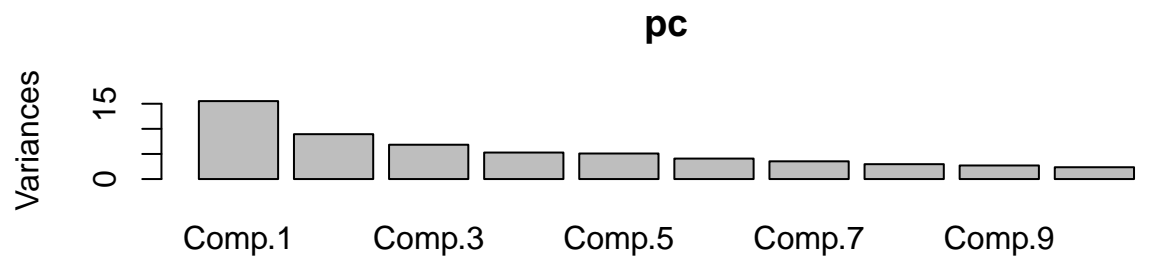
PC Loadings

```
par(mfrow=c(3,5),mar=c(.1,.1,.1,.1))
for(i in 1:15){
  imagedigit(pc$loadings[,i])
}
```



Variance explained

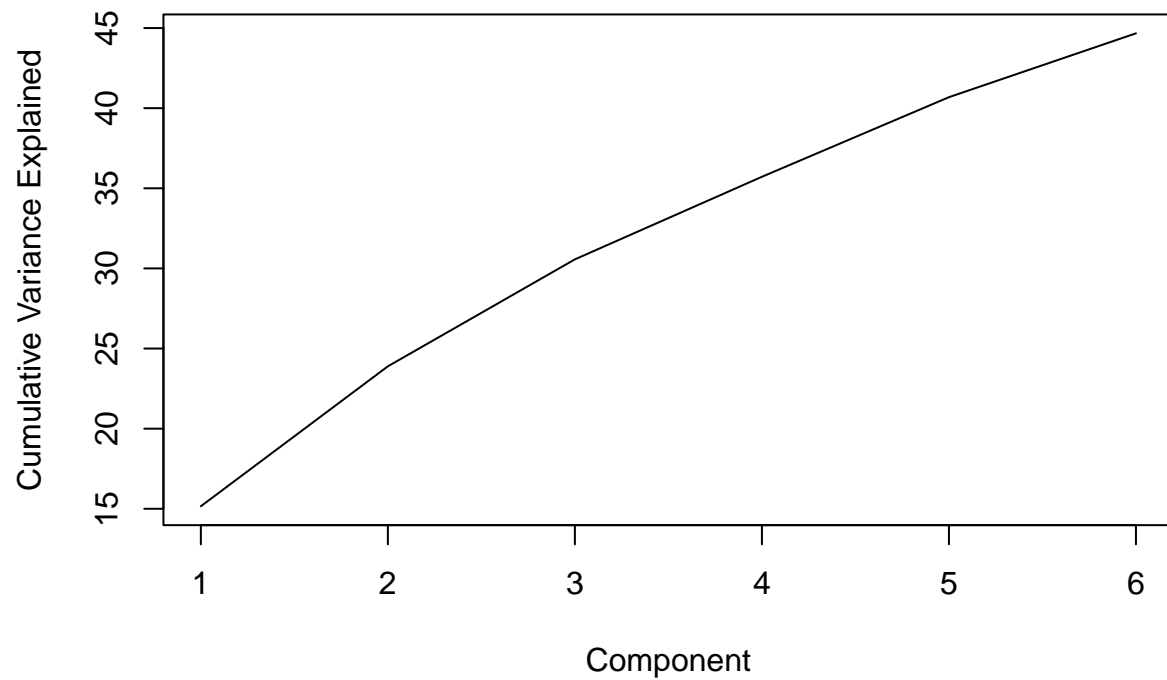
```
varex = 100*pc$sdev^2/sum(pc$sdev^2)
par(mfrow=c(2,1))
screeplot(pc)
plot(varex,type="l",ylab="% Variance Explained",xlab="Component")
```



Cumulative variance explained

```
#cumulative variance explained
cvarex = NULL
for(i in 1:ncol(cdat)){
  cvarex[i] = sum(varex[1:i])
}
plot(cvarex,type="l",ylab="Cumulative Variance Explained",xlab="Component", main = "Principal Component
```

Principal Component vs. Variance Explained



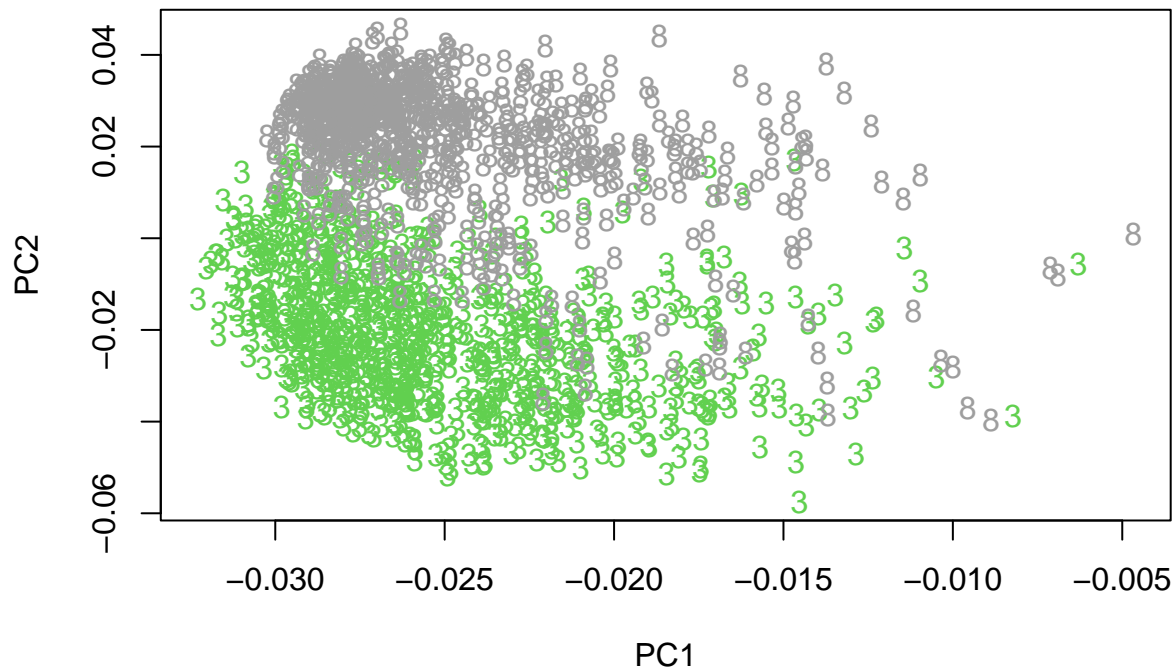
Compare to SVD

Without Centering & Scaling

```
svdd = svd(dat38)
U = svdd$u
V = svdd$v #PC loadings
D = svdd$d
Z = dat38%*%V #PCs
```

PC Scatterplots

```
PC1 <- U[,1]
PC2 <- U[,2]
plot(PC1,PC2,type="n",xlab="PC1",ylab="PC2")
text(PC1,PC2,rownames(dat38),col=rownames(dat38))
```



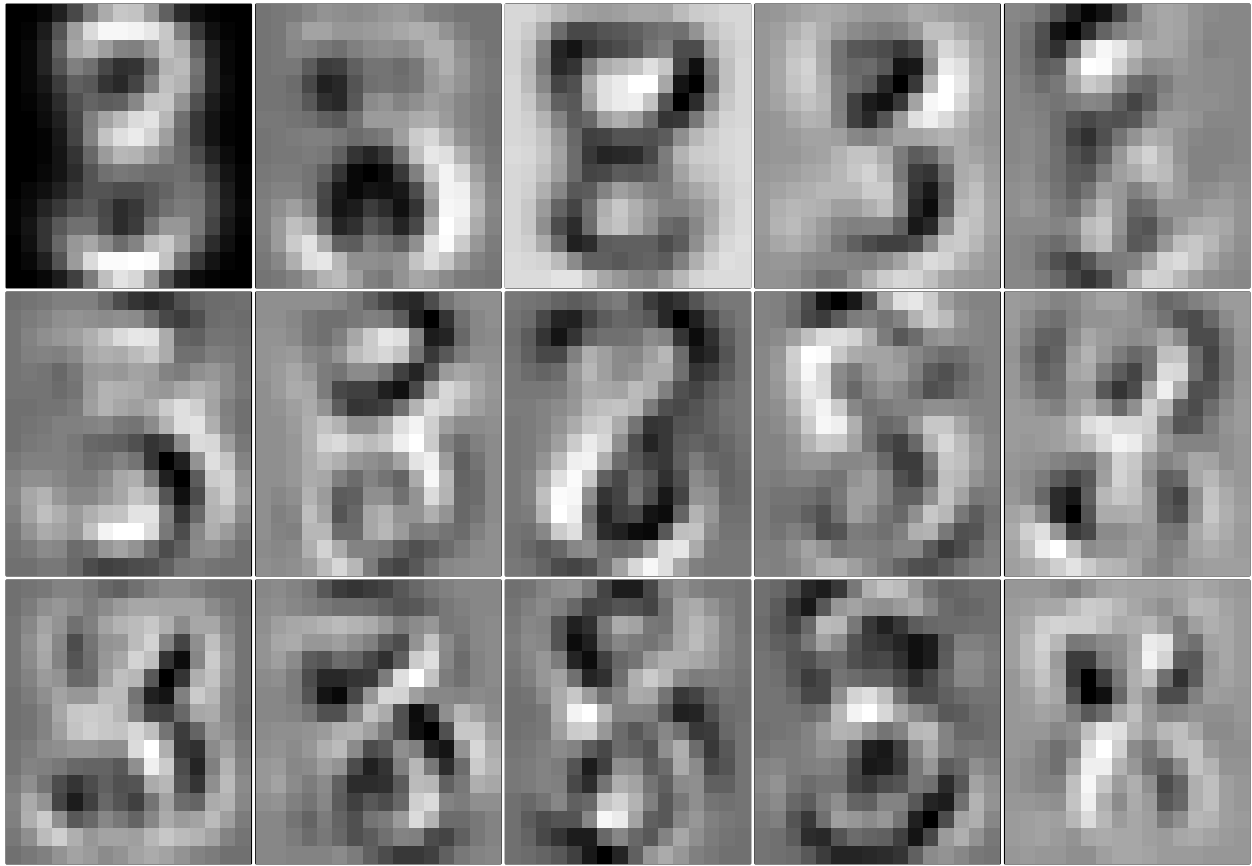
Pairs Plot Using ggpairs

```
PC1 <- U[,1]
PC2 <- U[,2]
PC3 <- U[,3]
PC4 <- U[,4]
PC5 <- U[,5]

pc.df.digits <- data.frame(digit_name = row.names(dat38), PC1, PC2, PC3, PC4, PC5)

ggpairs(pc.df.digits, mapping = aes(color = digit_name))

## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```

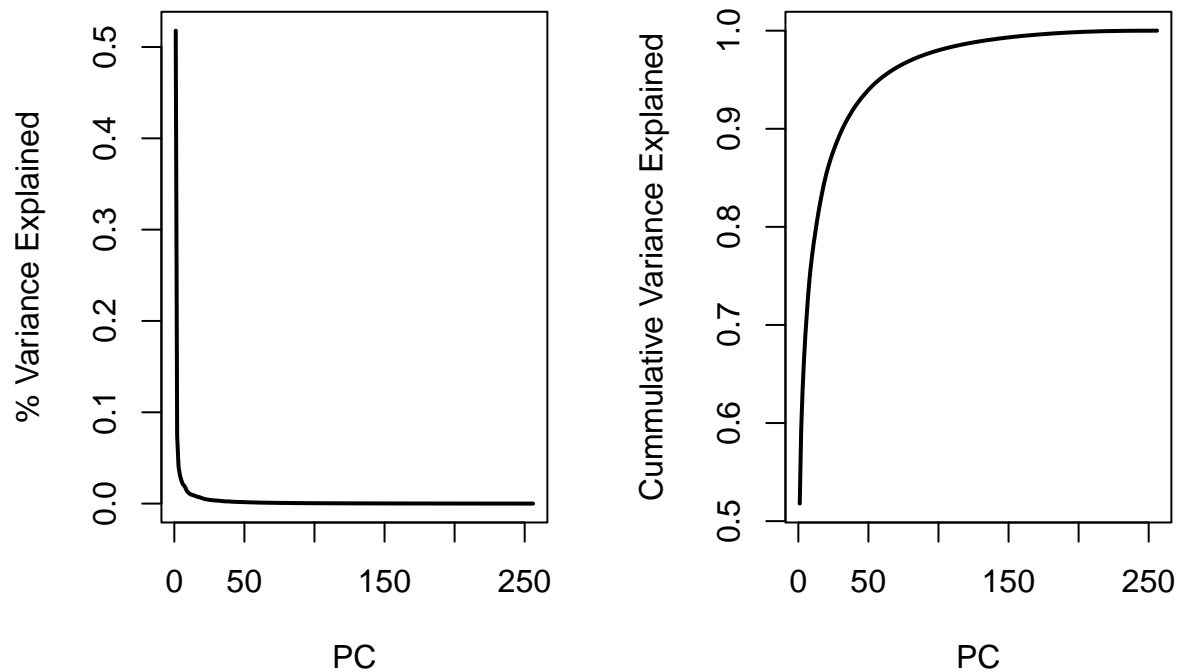



Variance Explained

```
#Variance Explained
varex = 0
cumvar = 0
denom = sum(D^2)
for(i in 1:256){
  varex[i] = D[i]^2/denom
  cumvar[i] = sum(D[1:i]^2)/denom
}
```

Screeplot

```
par(mfrow=c(1,2))
plot(1:256,varex,type="l",lwd=2,xlab="PC",ylab="% Variance Explained")
plot(1:256,cumvar,type="l",lwd=2,xlab="PC",ylab="Cumulative Variance Explained")
```



PCA Demo Using College Data

Load in Packages

```
library(ISLR)
```

```
data(College)
cdat = College[,2:18]
dim(cdat)
```

```
## [1] 777 17
```

```
names(cdat)
```

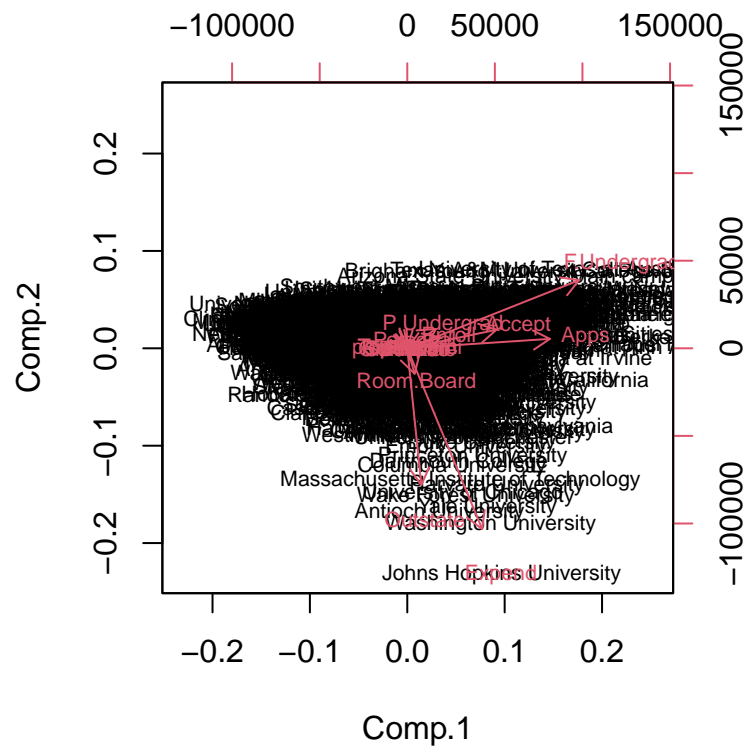
```
## [1] "Apps"      "Accept"    "Enroll"    "Top10perc" "Top25perc"
## [6] "F.Undergrad" "P.Undergrad" "Outstate"  "Room.Board" "Books"
## [11] "Personal"   "PhD"       "Terminal"  "S.F.Ratio"  "perc.alumni"
## [16] "Expend"    "Grad.Rate"
```

```
pc = princomp(cdat) #default - centers and scales
```

```
#Go back and display these plots side by side
```

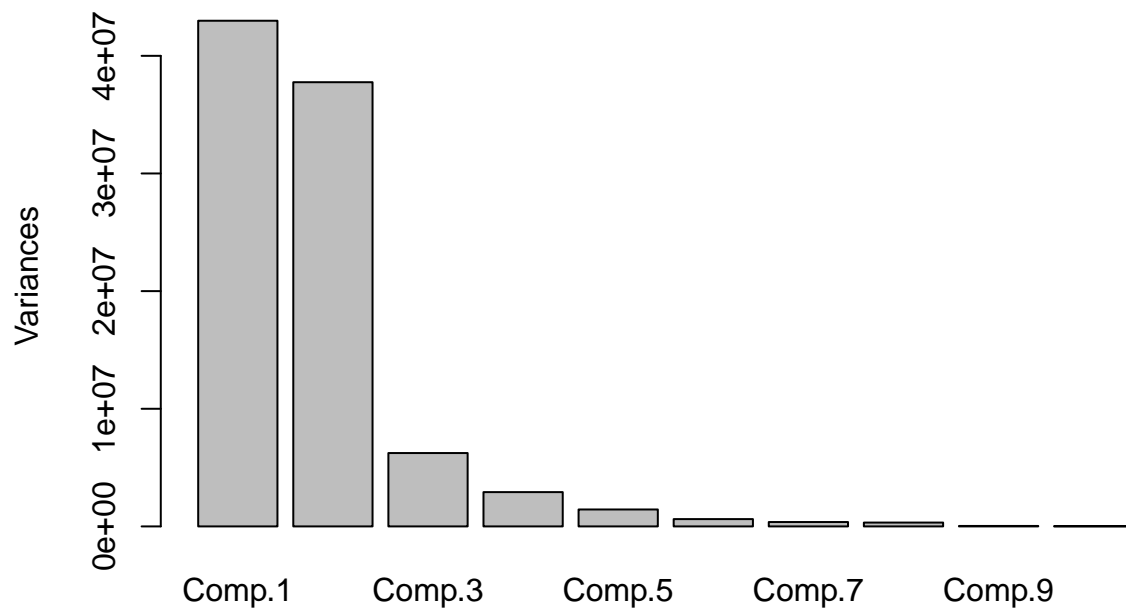
```
biplot(pc, cex=.7)
```

```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```



```
screepplot(pc)
```

pc

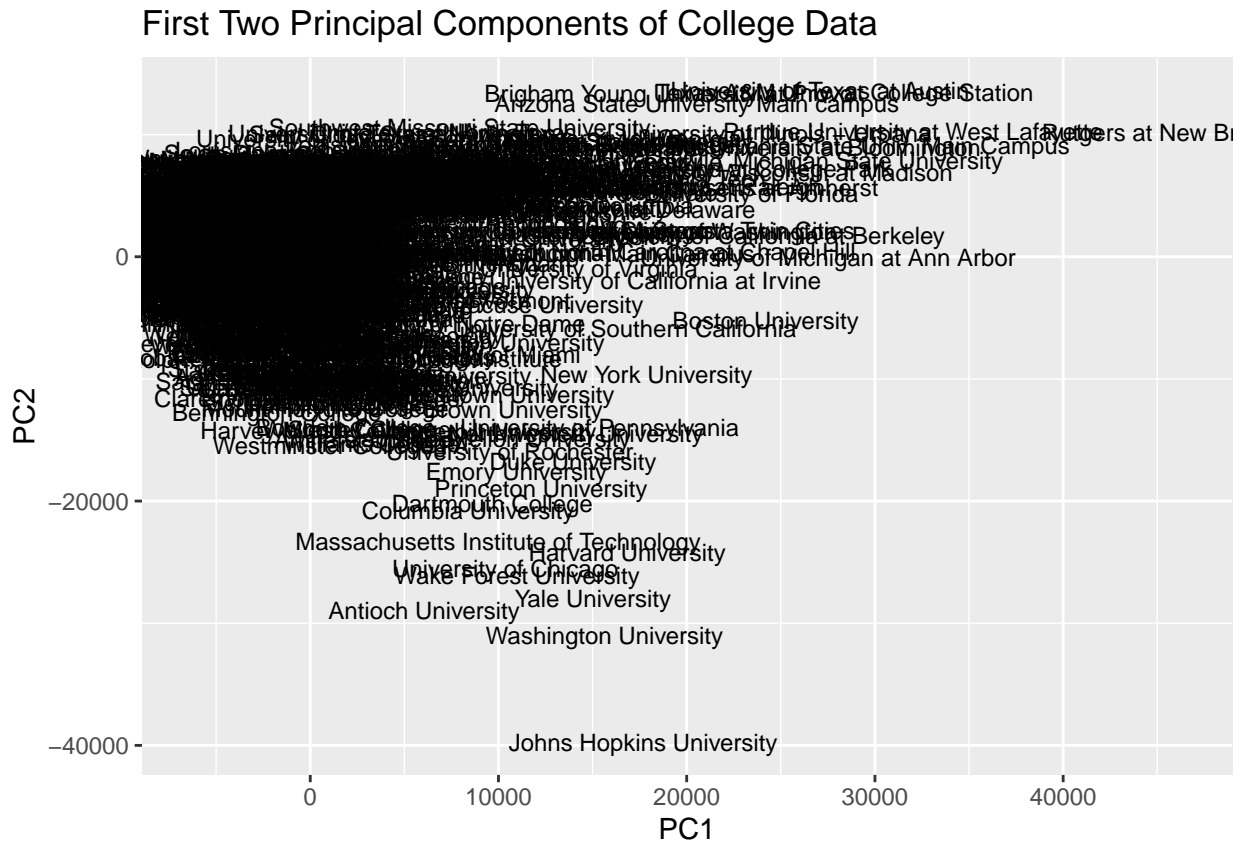


scatter plots - patterns among observations

```
PC1 <- as.matrix(x=pc$scores[,1])
PC2 <- as.matrix(x=pc$scores[,2])

PC <- data.frame(State = row.names(cdat), PC1, PC2)
```

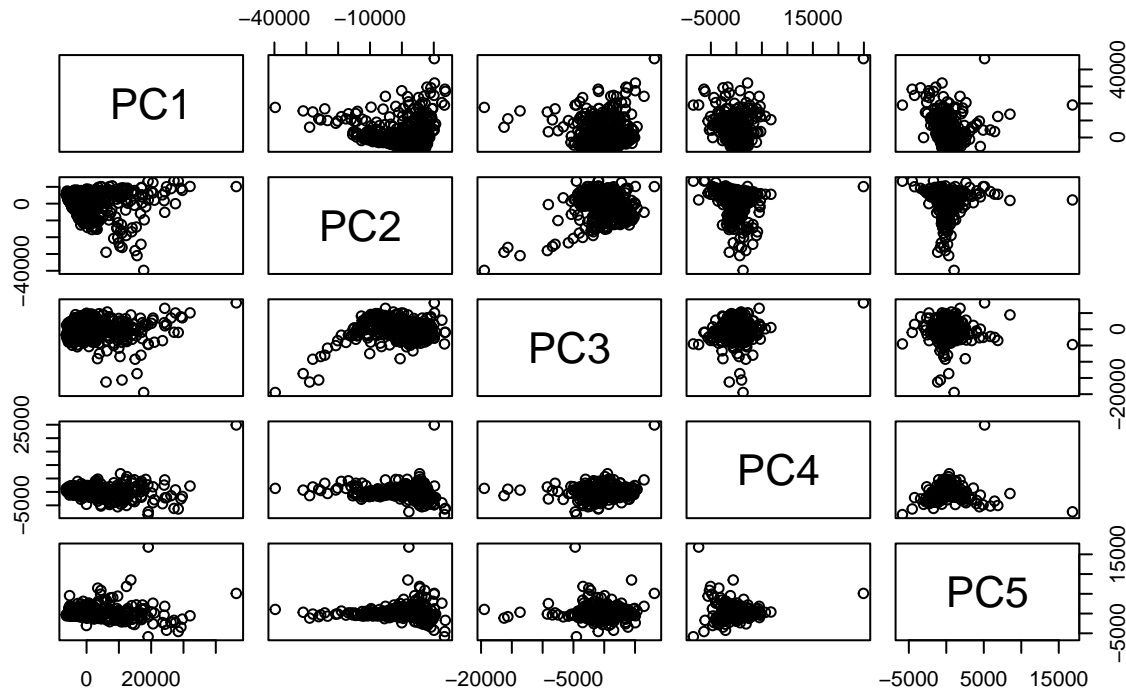
```
ggplot(PC, aes(PC1, PC2)) +
  geom_text(aes(label = State), size = 3) +
  xlab("PC1") +
  ylab("PC2") +
  ggtitle("First Two Principal Components of College Data")
```



Pairs Plot

```
comp_labels<-c("PC1", "PC2", "PC3", "PC4", "PC5")
pairs(pc$scores[,1:5], labels = comp_labels, main = "Pairs of PC's for College Data")
```

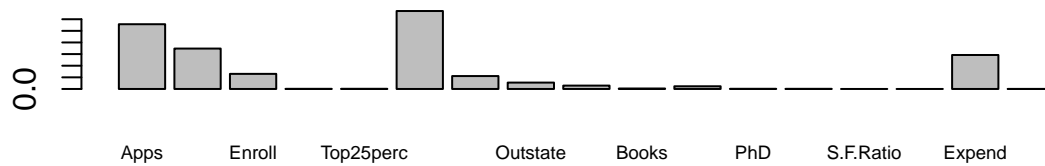
Pairs of PC's for College Data



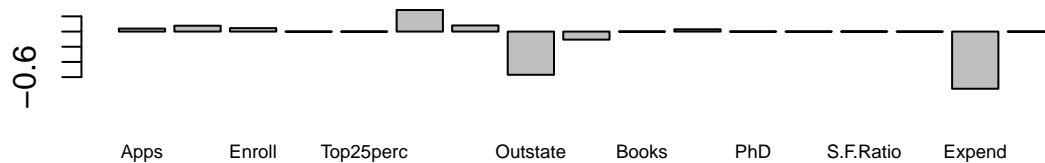
Loadings - variables that contribute to these patterns

```
par(mfrow=c(2,1))
barplot(pc$loadings[,1],cex.names=.6,main="PC 1 Loadings")
barplot(pc$loadings[,2],cex.names=.6,main="PC 2 Loadings")
```

PC 1 Loadings

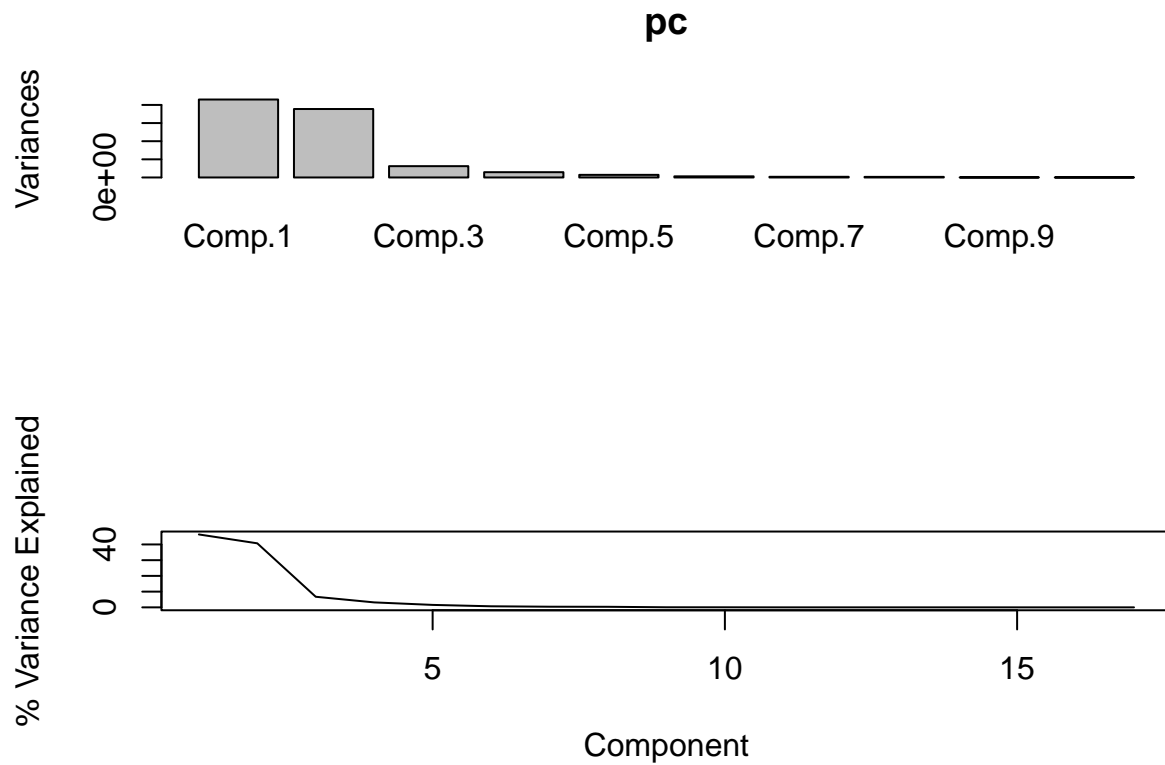


PC 2 Loadings



Variance explained

```
varex = 100*pc$sdev^2/sum(pc$sdev^2)
par(mfrow=c(2,1))
screeplot(pc)
plot(varex,type="l",ylab="% Variance Explained",xlab="Component")
```



Cumulative variance explained

```
#cumulative variance explained
cvarex = NULL
for(i in 1:ncol(cdat)){
  cvarex[i] = sum(varex[1:i])
}
plot(cvarex,type="l",ylab="Cumulative Variance Explained",xlab="Component", main = "Principal Component
```

Principal Component V. Variance Explained

