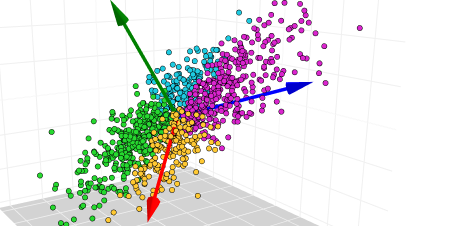
Week 9 - AYUPod - Principal Component Analysis

Table of Contents



(Source: kaggle.com)

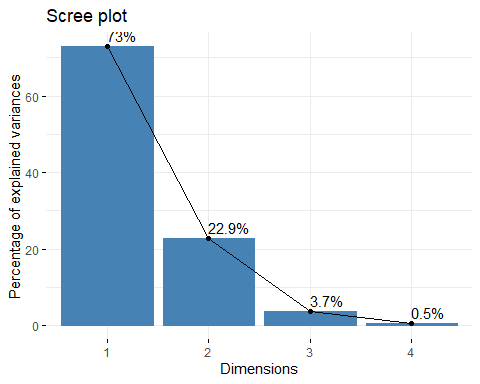
library(factoextra)  
library(tidyverse) # data manipulation and visualization  
library(gridExtra)  
data(iris)  
head(iris)

## Sepal.Length Sepal.Width Petal.Length Petal.Width Species  
## 1 5.1 3.5 1.4 0.2 setosa  
## 2 4.9 3.0 1.4 0.2 setosa  
## 3 4.7 3.2 1.3 0.2 setosa  
## 4 4.6 3.1 1.5 0.2 setosa  
## 5 5.0 3.6 1.4 0.2 setosa  
## 6 5.4 3.9 1.7 0.4 setosa

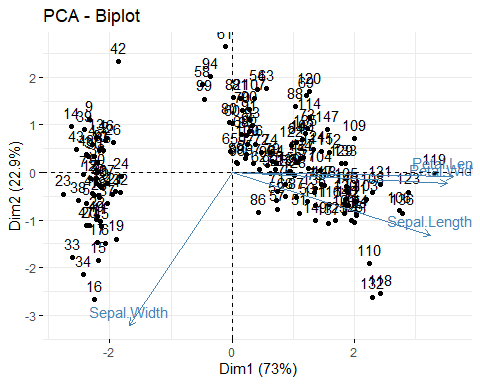
# The variable Species (index = 5) is removed  
# before the PCA analysis  
res.pca <- prcomp(iris[, -5], scale = TRUE)  
# Extract the eigenvalues/variances  
get\_eig(res.pca)

## eigenvalue variance.percent cumulative.variance.percent  
## Dim.1 2.91849782 72.9624454 72.96245  
## Dim.2 0.91403047 22.8507618 95.81321  
## Dim.3 0.14675688 3.6689219 99.48213  
## Dim.4 0.02071484 0.5178709 100.00000

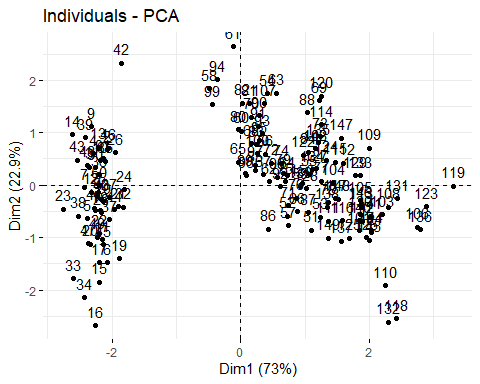
# Default plot  
fviz\_eig(res.pca, addlabels = TRUE)



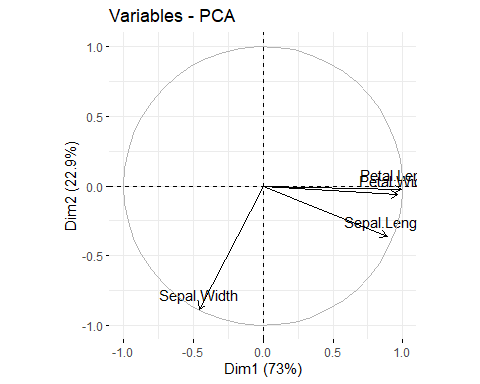
fviz\_pca(res.pca)



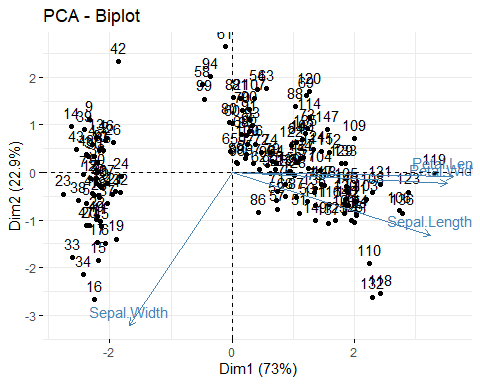
fviz\_pca\_ind(res.pca)



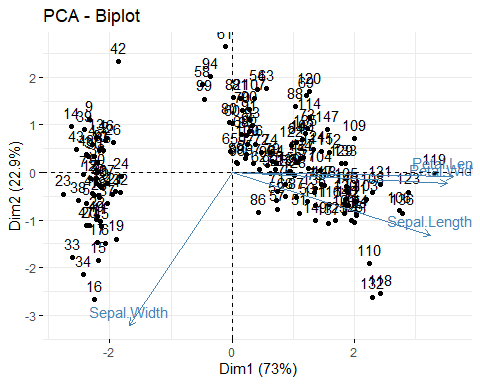
fviz\_pca\_var(res.pca)



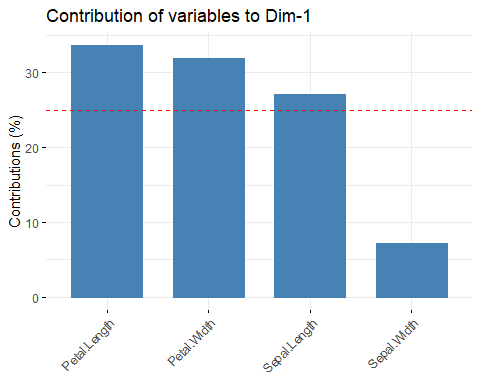
fviz\_pca\_biplot(res.pca)



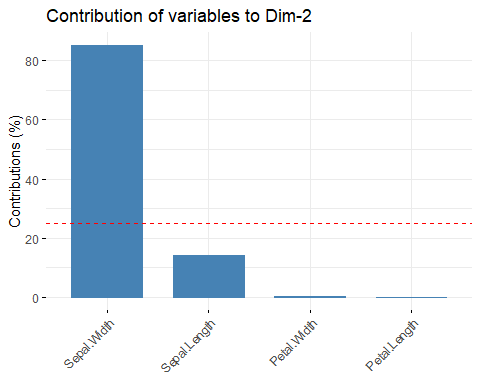
fviz\_pca(res.pca)



# Contributions of variables to PC1  
fviz\_contrib(res.pca, choice = "var", axes = 1, top = 10)

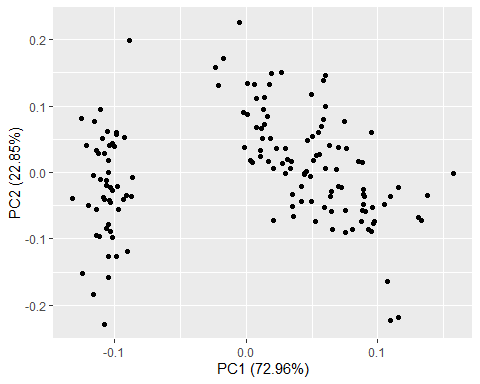


# Contributions of variables to PC2  
fviz\_contrib(res.pca, choice = "var", axes = 2, top = 10)

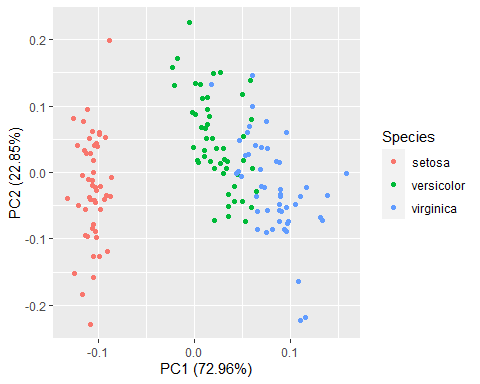


# Variable contribution to all components

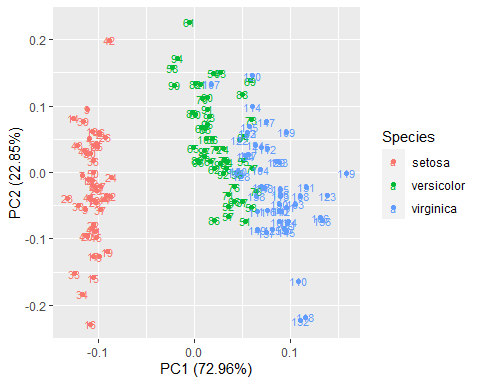
library(ggfortify)  
  
autoplot(res.pca)



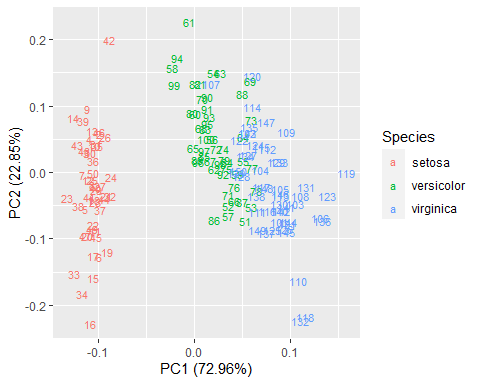
autoplot(res.pca, data = iris, colour = 'Species')



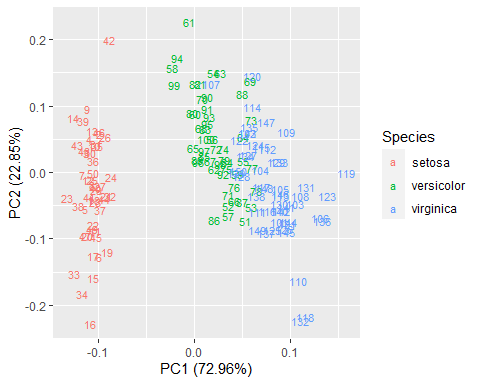
autoplot(res.pca, data = iris, colour = 'Species', label = TRUE, label.size = 3)



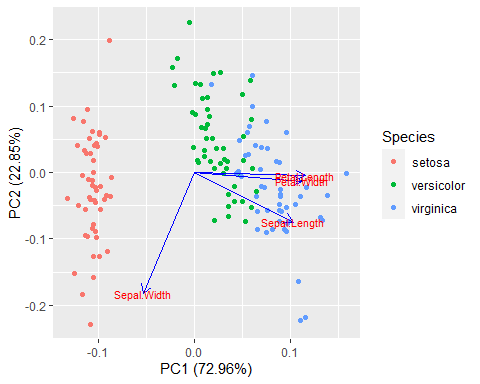
autoplot(res.pca, data = iris, colour = 'Species', shape = FALSE, label.size = 3)



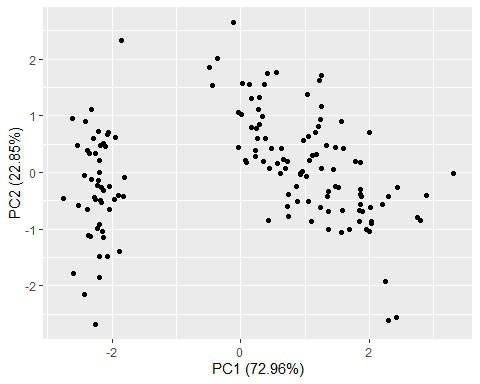
autoplot(res.pca, data = iris, colour = 'Species', shape = FALSE, label.size = 3)



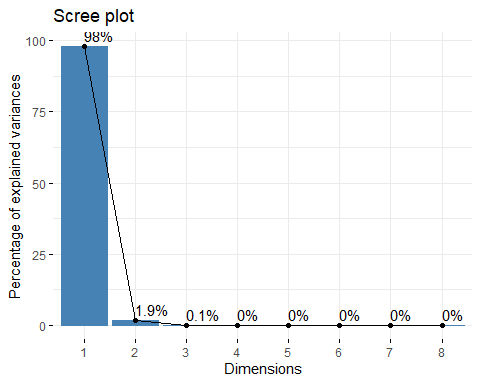
autoplot(res.pca, data = iris, colour = 'Species',  
 loadings = TRUE, loadings.colour = 'blue',  
 loadings.label = TRUE, loadings.label.size = 3)



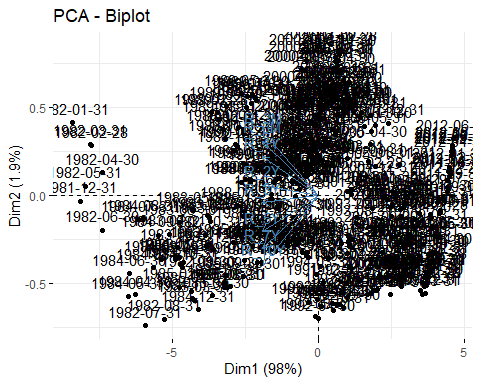
autoplot(res.pca, scale = 0)



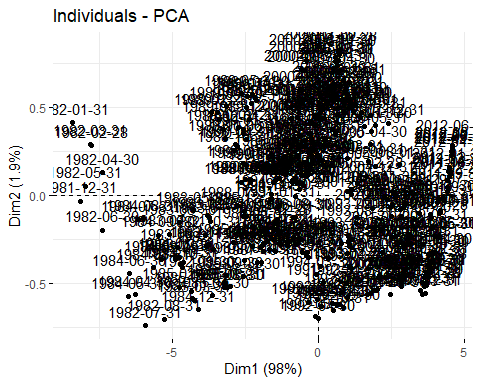
library(YieldCurve)  
data(FedYieldCurve)  
  
M <- as.matrix(FedYieldCurve)  
  
  
res.pca = prcomp(M, scale = TRUE)  
  
fviz\_eig(res.pca, addlabels = TRUE)



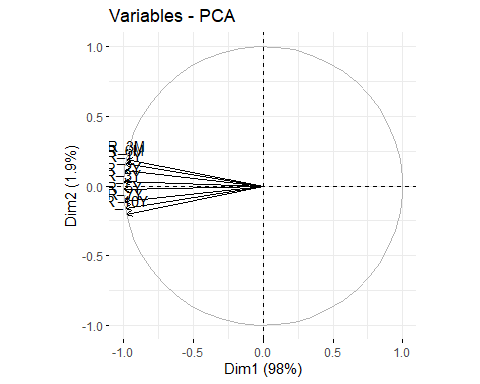
fviz\_pca(res.pca)



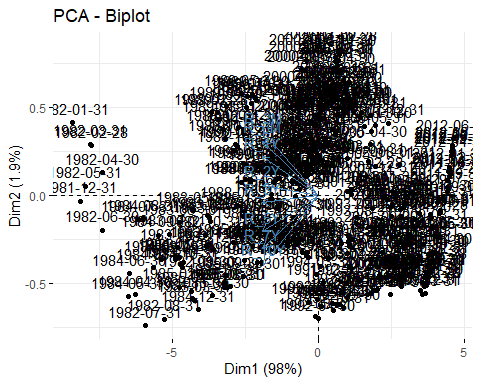
fviz\_pca\_ind(res.pca)



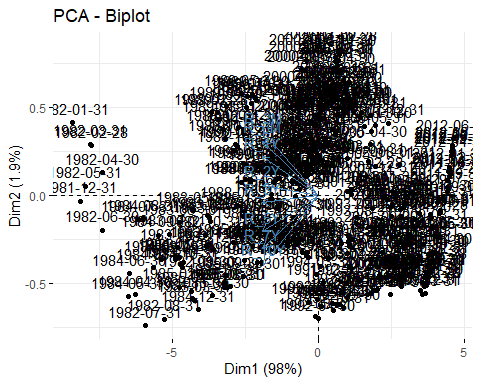
fviz\_pca\_var(res.pca)



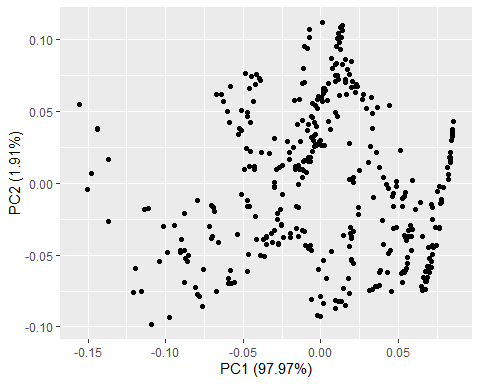
fviz\_pca\_biplot(res.pca)



fviz\_pca(res.pca)



library(ggfortify)  
  
autoplot(res.pca)



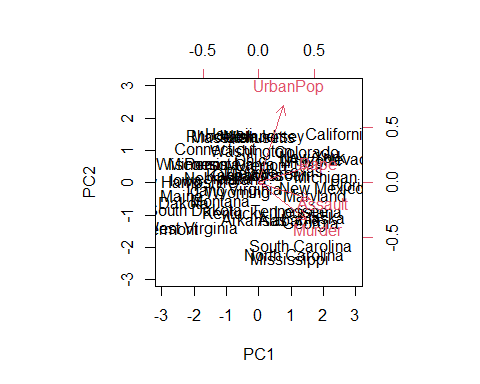
# plot arrangement  
data("USArrests")  
results <- prcomp(USArrests, scale = TRUE)  
results$rotation <- -results$rotation  
results$rotation

## PC1 PC2 PC3 PC4  
## Murder 0.5358995 -0.4181809 0.3412327 -0.64922780  
## Assault 0.5831836 -0.1879856 0.2681484 0.74340748  
## UrbanPop 0.2781909 0.8728062 0.3780158 -0.13387773  
## Rape 0.5434321 0.1673186 -0.8177779 -0.08902432

#reverse the signs of the scores  
results$x <- -1\*results$x  
  
#display the first six scores  
head(results$x)

## PC1 PC2 PC3 PC4  
## Alabama 0.9756604 -1.1220012 0.43980366 -0.154696581  
## Alaska 1.9305379 -1.0624269 -2.01950027 0.434175454  
## Arizona 1.7454429 0.7384595 -0.05423025 0.826264240  
## Arkansas -0.1399989 -1.1085423 -0.11342217 0.180973554  
## California 2.4986128 1.5274267 -0.59254100 0.338559240  
## Colorado 1.4993407 0.9776297 -1.08400162 -0.001450164

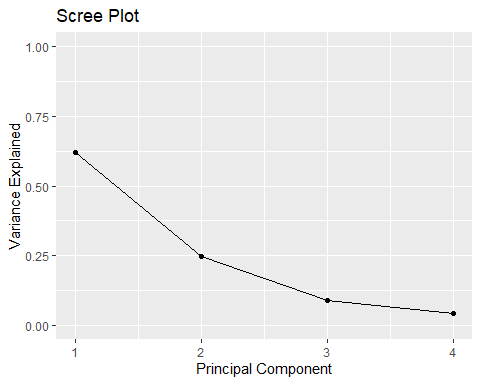
biplot(results, scale = 0)



#calculate total variance explained by each principal component  
results$sdev^2 / sum(results$sdev^2)

## [1] 0.62006039 0.24744129 0.08914080 0.04335752

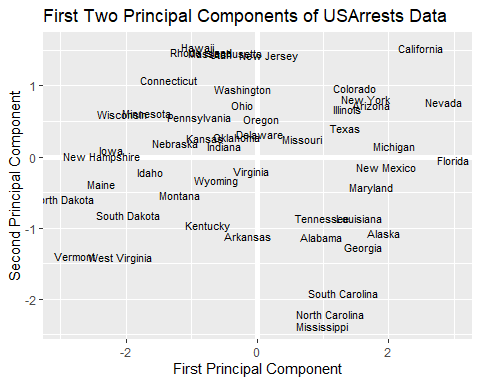
#calculate total variance explained by each principal component  
var\_explained = results$sdev^2 / sum(results$sdev^2)  
  
#create scree plot  
qplot(c(1:4), var\_explained) +   
 geom\_line() +   
 xlab("Principal Component") +   
 ylab("Variance Explained") +  
 ggtitle("Scree Plot") +  
 ylim(0, 1)



library(tidyverse) # data manipulation and visualization  
library(gridExtra) # plot arrangement  
data("USArrests")  
apply(USArrests, 2, var)

## Murder Assault UrbanPop Rape   
## 18.97047 6945.16571 209.51878 87.72916

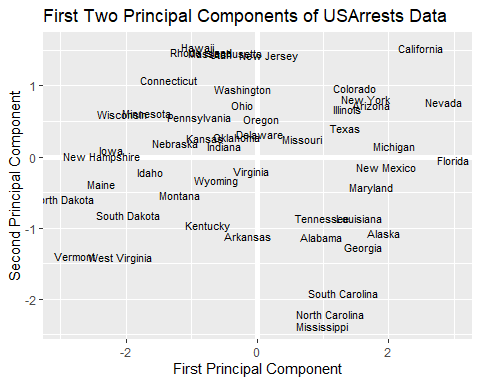
scaled\_df <- apply(USArrests, 2, scale)  
arrests.cov <- cov(scaled\_df)  
arrests.eigen <- eigen(arrests.cov)  
  
phi <- arrests.eigen$vectors[,1:2]  
  
phi <- -phi  
row.names(phi) <- c("Murder", "Assault", "UrbanPop", "Rape")  
colnames(phi) <- c("PC1", "PC2")  
PC1 <- as.matrix(scaled\_df) %\*% phi[,1]  
PC2 <- as.matrix(scaled\_df) %\*% phi[,2]  
  
# Create data frame with Principal Components scores  
PC <- data.frame(State = row.names(USArrests), PC1, PC2)  
ggplot(PC, aes(PC1, PC2)) +   
 modelr::geom\_ref\_line(h = 0) +  
 modelr::geom\_ref\_line(v = 0) +  
 geom\_text(aes(label = State), size = 3) +  
 xlab("First Principal Component") +   
 ylab("Second Principal Component") +   
 ggtitle("First Two Principal Components of USArrests Data")



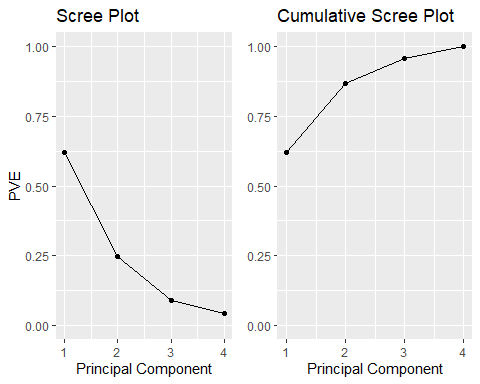
PVE <- arrests.eigen$values / sum(arrests.eigen$values)  
round(PVE, 2)

## [1] 0.62 0.25 0.09 0.04

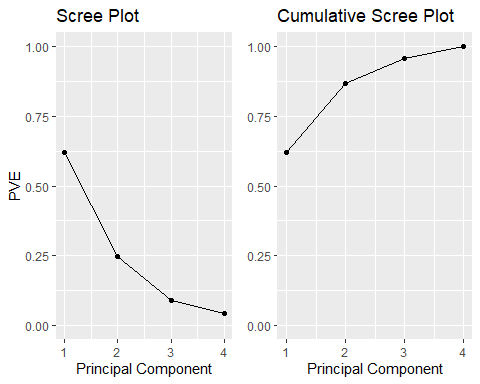
# Calculate Principal Components scores  
PC1 <- as.matrix(scaled\_df) %\*% phi[,1]  
PC2 <- as.matrix(scaled\_df) %\*% phi[,2]  
PC <- data.frame(State = row.names(USArrests), PC1, PC2)  
# Plot Principal Components for each State  
ggplot(PC, aes(PC1, PC2)) +   
 modelr::geom\_ref\_line(h = 0) +  
 modelr::geom\_ref\_line(v = 0) +  
 geom\_text(aes(label = State), size = 3) +  
 xlab("First Principal Component") +   
 ylab("Second Principal Component") +   
 ggtitle("First Two Principal Components of USArrests Data")



PVE <- arrests.eigen$values / sum(arrests.eigen$values)  
# PVE (aka scree) plot  
PVEplot <- qplot(c(1:4), PVE) +   
 geom\_line() +   
 xlab("Principal Component") +   
 ylab("PVE") +  
 ggtitle("Scree Plot") +  
 ylim(0, 1)  
  
# Cumulative PVE plot  
cumPVE <- qplot(c(1:4), cumsum(PVE)) +   
 geom\_line() +   
 xlab("Principal Component") +   
 ylab(NULL) +   
 ggtitle("Cumulative Scree Plot") +  
 ylim(0,1)  
  
grid.arrange(PVEplot, cumPVE, ncol = 2)



# PVE (aka scree) plot  
PVEplot <- qplot(c(1:4), PVE) +   
 geom\_line() +   
 xlab("Principal Component") +   
 ylab("PVE") +  
 ggtitle("Scree Plot") +  
 ylim(0, 1)  
  
# Cumulative PVE plot  
cumPVE <- qplot(c(1:4), cumsum(PVE)) +   
 geom\_line() +   
 xlab("Principal Component") +   
 ylab(NULL) +   
 ggtitle("Cumulative Scree Plot") +  
 ylim(0,1)  
  
grid.arrange(PVEplot, cumPVE, ncol = 2)



## For Modeling (PC Regression)

library(tidyverse)  
d <- read\_csv("data/TermLife.csv")  
d1 <- d[d$FACE>0, ]