```
#Method 1
myd=read.table("Boston Housing.txt", header = T)
listvar=c("crime","zn","indus","nox","rm","age","medv")
mydpc=myd[listvar]
library(psych)
describe(mydpc)
pairs(mydpc)
myd$lncrime=log(myd$crime)
listvar=c("lncrime","zn","indus","nox","rm","age","medv")
mydpc=myd[listvar]
#Scatterplot matrix
pairs(mydpc)
M=cor(mydpc)
library(corrplot)
corrplot(M, method="ellipse")
mydpc=myd[listvar]
fa.parallel(mydpc, fa="pc", n.iter = 100, show.legend = FALSE,
main="Scree Plot")
cortest.bartlett(M, n=length(mydpc[,1]))
KMO(M)
principal(mydpc, nfactors = 4, rotate="none")
fit=principal(mydpc, nfactors = 2, rotate="varimax")
fa.sort(fit)
fit$weights
prcomps=as.matrix(myd[listvar])%*%as.matrix(fit$weights)
prcomps

#Method 2
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_Varimax = function(v)
{
  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(v$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 = V1, y = V2, label
= .names, colour = .names, fontface="bold")) +
    coord_fixed(ratio=1) + labs(x = "PC1", y = "PC2")
}

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 = data.frame(pcaData$rot.mat, .names =
row.names(pcaData$loadings))
  print(loadings)
  p + geom_text(data=loadings, mapping=aes(x = X1, y = X1, label
= .names, colour = .names, fontface="bold")) +
    coord_fixed(ratio=1) + labs(x = "PC1", y = "P2")
}
library(ggplot2)
library(GGally)
data = iris[, c(1, 3, 4)]
ggpairs(data)
p = prcomp(data)
p
summary(p)
options(scipen=100, digits=2)
R = p$rotation
rotatedData = as.data.frame(as.matrix(data) %*% R)
round(cov(rotatedData), digits=2)
ggpairs(rotatedData)
irisOrig = iris[, 1:4]
irisMod = irisOrig
```

```
irisMod$Petal.Length = irisMod$Petal.Length * 10
ggpairs(irisMod)
pOrig = prcomp(irisOrig)
pMod = prcomp(irisMod)
print(pMod)
print(pOrig)
summary(pMod)
pModScale = prcomp(irisMod, scale=T)
summary(pModScale)
print(pModScale)
PCA_Plot(pModScale)
plot(pModScale)

#Method 3
library(stats)
data()
head(iris)
plot(iris)
d = iris[1:4]
plot(d)
cor(d)
p = prcomp(d)
print(p)
plot(p)
PCA_Plot(p)
head(d)
p1 = prcomp(d)
p1
summary(p)
round(p1$rotation, 2)
p2 = princomp(d)
p2
summary(p2)
head(p2$scores)
round(p2$loadings, 2)

#Method 4
library(foreign)  # Allows us to read spss files!
library(corrplot)
library(car)
library(QuantPsyc)
library(leaps)
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")
}
hbat = read.spss("HBAT.sav", to.data.frame=T)
head(hbat)
hbatNumeric = hbat[, c(20, 7:19)]
head(hbatNumeric)
plot(hbatNumeric)
cor.hbat = cor(hbatNumeric)
cor.hbat
corrplot(cor.hbat, method="ellipse")
dim(hbatNumeric)
library(psych)
options("scipen"=100, "digits"=5)
round(cor(hbatNumeric[, 2:14]), 2)
MCorrTest = corr.test(hbatNumeric[, 2:14], adjust="none")
MCorrTest
M = MCorrTest$p
M
MTest = ifelse(M < .01, T, F)
MTest
colSums(MTest) - 1
hbatReduced = hbatNumeric[, -c(1, 11, 13)]
head(hbatReduced)
cor(hbatReduced)
p = prcomp(hbatReduced, center=T, scale=T)
plot(p)
abline(1, 0)
summary(p)
print(p)
par(mar=c(2, 2, 2, 2))
plot(p)
```

```
PCA_Plot(p)
PCA_Plot_Secondary(p)
biplot(p)
rawLoadings = p$rotation %*% diag(p$sdev, nrow(p$rotation),
nrow(p$rotation))
print(rawLoadings)
v = varimax(rawLoadings)
ls(v)
v
p2 = psych::principal(hbatReduced, rotate="varimax", nfactors=4,
scores=TRUE)
print(p2$loadings, cutoff=.4, sort=T)
p2$loadings
p2$values
p2$communality
p2$rot.mat
v$loadings
PCA_Plot_Psyc(p2)
PCA_Plot_Psyc_Secondary(p2)
fit = factanal(hbatReduced, 4)
print(fit$loadings, cutoff=.4, sort=T)
summary(fit)
par(mar=c(4, 4, 4, 4))
plot(p)
abline(1, 0)
PCA_Plot(p)

#Method 5
library(corrplot)

# PCA_Plot functions

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")
}
```