Homework #2, DSA 5103 Fall 2016

Sanjiv Shah (OU ID 113180542)

September 26, 2016

## Libraries:

library(reshape2)  
library(ggplot2)  
library(robustbase)  
library(outliers)  
library(fitdistrplus)

library(Amelia)

library(mice)

library(plyr)  
library(HSAUR2)

library(VIM)

library(devtools)  
library(ggbiplot)

## Problem 1: Concordance and Discordance

x = c(3, 4, 2, 1, 7, 6, 5)  
y = c(4, 3, 7, 6, 5, 2, 1)

### C = sum(pmin(max(x)-x, max(y)-y))

C = 0; D = 0

### Calculate the Concordance and Discordance based on the definition

for(i in 1:length(x))  
 for(j in 1:length(y))  
 {  
 if(x[i] < x[j]){  
 if(y[i] < y[j])   
 C = C + 1   
 if(y[i] > y[j])   
 D = D + 1  
 }  
 }

### No. of Concordent Pairs:

C

## [1] 6

### No. of Discordent Pairs:

D

## [1] 15

## Problem 2: Outliersexample.r

### After running the entire script, I got African Elephant as the final answer.

## Problem 3: Advanced Density Plots

### Step 1: Randomly generate 4 variables (500 datapoints each) using 4 different distributions

### Normal distribution

a<-rnorm(500)

### Poisson distribution

b <- rpois(500,1)

### Uniform Distribution

c<-runif(500, min=-5, max =5)

### Exponential distribution

d <- rexp(500)

### Create Dataframe

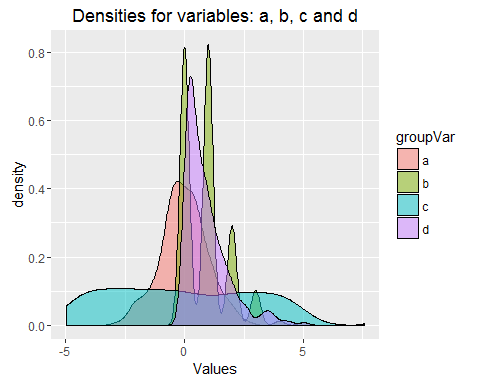
df <- data.frame(a,b,c,d)

### Reshape the dataframe

df2 <- melt(df, measure.vars=c("a","b","c","d"),variable.name="groupVar",value.name="value")

### Generate Overlapping Density Plots

ggplot(df2, aes(x=value, fill=groupVar)) + geom\_density(alpha=0.5) +   
 labs(x = "Values",title = "Densities for variables: a, b, c and d")



**Problem 4: Shark Attacks**

GSAF <- read.csv("ISE 5103 GSAF.csv", header = TRUE)

### 4(a): Comments on Timeliness of Data:

#### The data is spread over a 100 years. Therefore more than a few aspects of the data are likely

#### to be affected by changes in populations, reporting, availibility of medical services over the time, etc.

#### Also, 'Provoked Incidents' is a very subjective variable that may have varied in interpretation

#### over the years.

### 4(b): Ceate a data frame that limits the data to year 2000 and later

GSAFdata <- GSAF[GSAF$Year>1999,]

### 4(c) Convert Date to R date type

newDate <-as.Date(GSAFdata$Date, "%d-%B-%y")

#### Append new date field to GSAFdata

GSAFdata <- data.frame(newDate,GSAFdata)

#### 4(d): Calculate percentage of missing data (as a % of original dataset )

missing <- GSAFdata[is.na(GSAF$newDate),]

## Warning in is.na(GSAF$newDate): is.na() applied to non-(list or vector) of  
## type 'NULL'

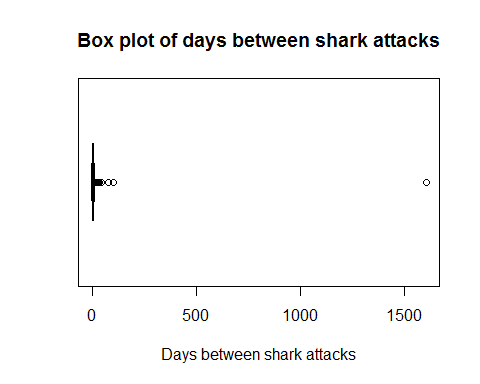
GSAFdata <-GSAFdata[!is.na(GSAFdata$newDate), ]

#### 4(f)(i): New variable for days between

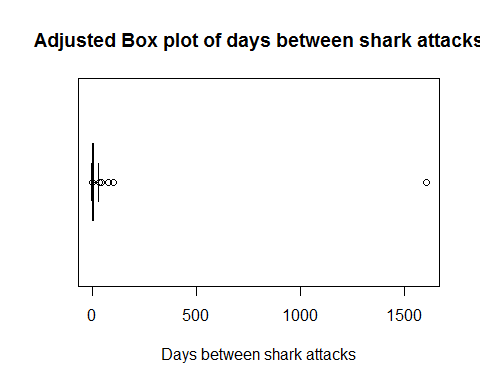
GSAFdata <- GSAFdata[order(GSAFdata$newDate),]  
daysbetwn <- diff(GSAFdata$newDate)  
daysbetwn <- c(0, daysbetwn)  
GSAFdata <- data.frame(daysbetwn, GSAFdata)

#### 4(f)(ii) Boxpolot and adjusted boxtplots

boxplot(GSAFdata$daysbetwn, notch=TRUE,col="beige", horizontal = TRUE, xlab="Days between shark attacks", main="Box plot of days between shark attacks")



adjbox(GSAFdata$daysbetwn,notch=TRUE,col="beige", horizontal = TRUE, xlab="Days between shark attacks",main="Adjusted Box plot of days between shark attacks")



**4(f)(iii) Grubbs test**

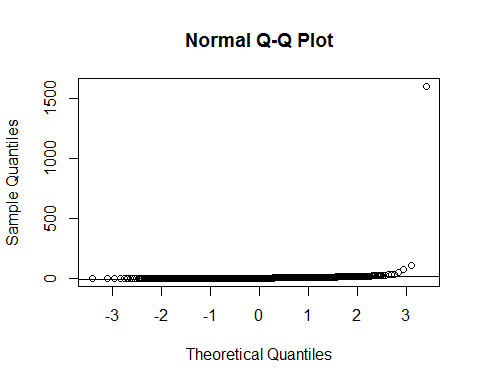
grubbs.test(GSAFdata$daysbetwn, type=10)

## alternative hypothesis: highest value 1605 is an outlier

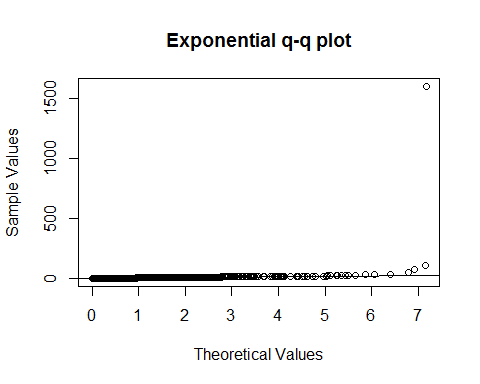
#### Grubbs test is not very helpful since boxplot indicates possible multiple outliers.

### 4(g):

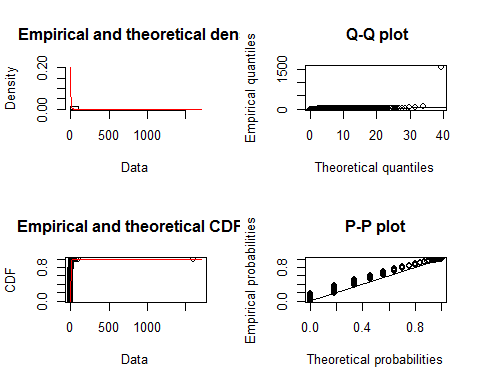
qqnorm(GSAFdata$daysbetwn)  
qqline(GSAFdata$daysbetwn, distribution = qnorm)

 ####exponentially distributed data

x <- rexp(1556)   
qqplot (x, GSAFdata$daysbetwn, main = "Exponential q-q plot", xlab = "Theoretical Values", ylab = "Sample Values")  
qqline(GSAFdata$daysbetwn, distribution = qexp)

 ###4(h):

fitexp <- fitdist(GSAFdata$daysbetwn, "exp")  
plot(fitexp)



gofstat(fitexp)

## Problem 5: Missing Data

### 5(a): misssingness

data(freetrade)

##### Using aggregate function by country

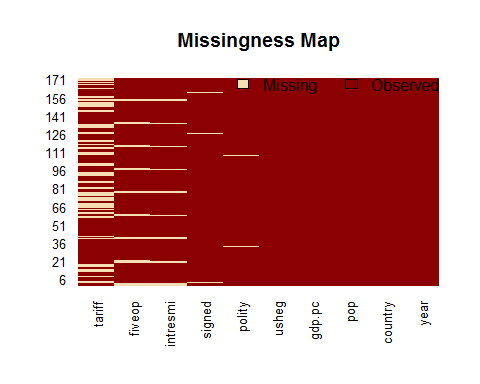
aggregate (freetrade, by=list(freetrade$country), function(x) mean(is.na(x)))

##### Using aggregate function by year

aggregate (freetrade, by=list(freetrade$year), function(x) mean(is.na(x)))

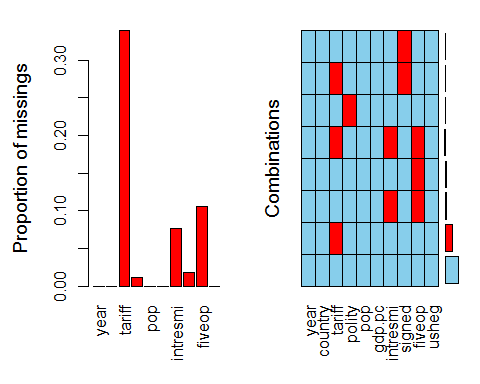
#### Graphical representation: Amelia package

missmap(freetrade, by = list(freetrade$country))



**Graphical representation: VIM package**

a <- aggr(freetrade)



Observations: Variables year, country, pop, gdp, ps and usheg have no missing data. #### Tariff has the most missing data.

### 5(b) Missingness in tariff analysis usin chi-square test

t<-table(freetrade$country,is.na(freetrade$tariff))

# chi-square test for independence

chisq.test(t)

### chi-square test excluding "Nepal"

t<-table(freetrade$country[freetrade$country!="Nepal"],is.na(freetrade$tariff[freetrade$country!="Nepal"]))   
chisq.test(t)

### chi-square test excluding "Philippines"

t<-table(freetrade$country[freetrade$country!="Philippines"],is.na(freetrade$tariff[freetrade$country!="Philippines"]))   
chisq.test(t)

### chi-square test excluding both "Nepal" and Philippines"

L<-freetrade$country!="Philippines" & freetrade$country!="Nepal"   
t<-table(freetrade$country[L], is.na(freetrade$tariff[L]))   
chisq.test(t)

#### There is not sufficient evidence to support null hypothesis (tariff is independent of country)

#### However, when Nepal and Phillipines are removed,p-value is still very high (0.42)

## Problem 6: PCA

### 6(a): Mathematics of PCA

#### 6(a)(i): Create correlation matrix of mtcars

data("mtcars")  
corMat <- cor(mtcars)

#### 6(a)(ii): Eignvalues and eigenvectors

eigen <- eigen(corMat)

#### 6(a)(iii): PC of mtcars attributes

prcomp <- prcomp(mtcars, scale = TRUE)

#### 6(a)(iv): Compre magnitudes of eigen values and PC

compare <- abs(eigen$vectors) - abs(prcomp$rotation)

### PC calculated in (ii) and (iii) the same in magnitude. This is becauase PC are the same as eigenvector with the highest eigen value.

##### 6(a)(v): Orthogonality between PC1 and PC2

PCA <- as.data.frame(prcomp$rotation)  
PCA$PC1%\*%PCA$PC2

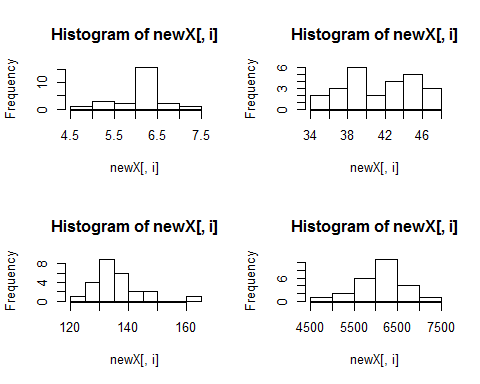
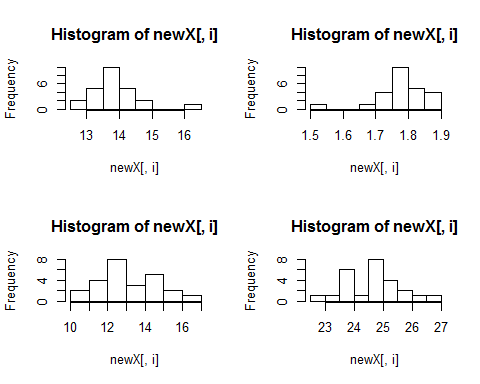
## [,1]  
## [1,] -2.775558e-17

### PC1 and PC2 are orthogonal

### 6(b): Heptathlon data

#### 6(b)(i): Histograms

data("heptathlon")  
par(mfrow=c(2,2))  
a <- apply(heptathlon[,1:8],2,hist)



From the quick review of histograms indicate resaonably normal distribution for all variables ####6(b)(ii): Grubbs test for ONE OUTLIER

grubbs\_dist <- apply(heptathlon[,1:8],2,grubbs.test)

### Launa (PNG) is the outlier competitor for Hurdles, Highjump, longjump, and 800m running (including score)

#### Remove Launa (PNG) from the data

heptathlon <- heptathlon[!rownames(heptathlon) %in% "Launa (PNG)",]

### 6(b)(iii): x <- max(x)-x Transformation

heptathlon[,"hurdles"] <- max(heptathlon$hurdles)-heptathlon[,"hurdles"]  
heptathlon[,"run200m"] <- max(heptathlon$hurdles)-heptathlon[,"run200m"]  
heptathlon[,"run800m"] <- max(heptathlon$hurdles)-heptathlon[,"run800m"]

### 6b(iv): PCA

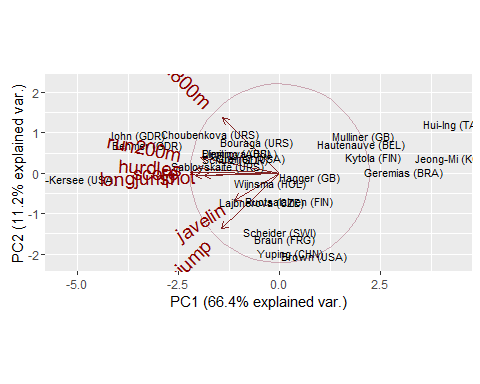
hprcomp <- prcomp(heptathlon, scale = TRUE)

#### Rotations

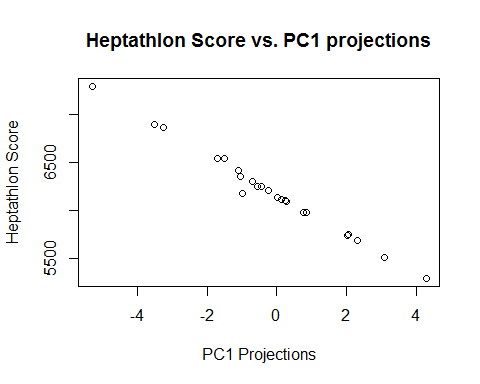
Hpca <- as.data.frame(hprcomp$rotation)

#### ggibiplot for first 2 PC

ggbiplot(hprcomp, circle = T, obs.scale = 1, varname.size = 5, labels = rownames(heptathlon))

 ####Score, Shotput, Hurdles and Longjump are the biggest contributing factors for PC1. #### The angle between these is also very small suggesting strong association. ###6(b)(vi): PCA projections

plot(hprcomp$x[,1], heptathlon$score, main = "Heptathlon Score vs. PC1 projections", xlab = "PC1 Projections", ylab = "Heptathlon Score")

 ####The Heptathlon score and PC1 plot shows very strong correlation. We can say that PC1 is good indicator of Score. ###6(c): Handwriting Analysis #Problem 6(c)(i): Reading three datsets corresponding to digits 1, 4 and 7

digit1<-read.csv("train.1.csv", header=F)   
digit4<-read.csv("train.4.csv", header=F)   
digit7<-read.csv("train.7.csv", header=F)

### Calculate and store PC attributes

prcomp1<-prcomp(digit1)   
prcomp4<-prcomp(digit4)   
prcomp7<-prcomp(digit7)

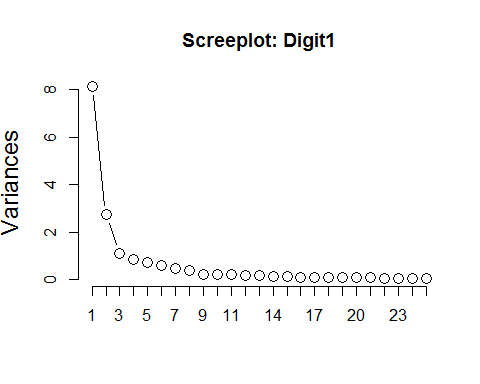
#### Calculate SD, variance and cumulative variance for each PC

sprcomp1<-summary(prcomp1)   
sprcomp4<-summary(prcomp4)   
sprcomp7<-summary(prcomp7)

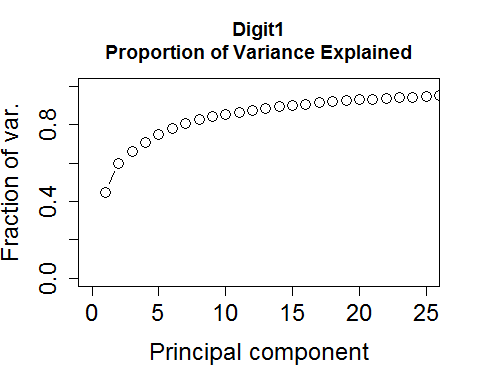
#### Generate 'Screeplots' and 'Proportion of variance explained by PCS' plots

#### Datset: Digit 1

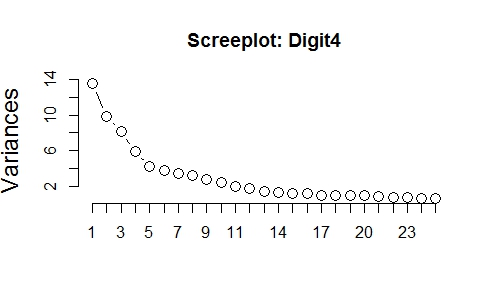
screeplot(prcomp1,npcs=25, type="lines", main="Screeplot: Digit1",cex=1.5, cex.lab=1.5, cex.axis=1.5)



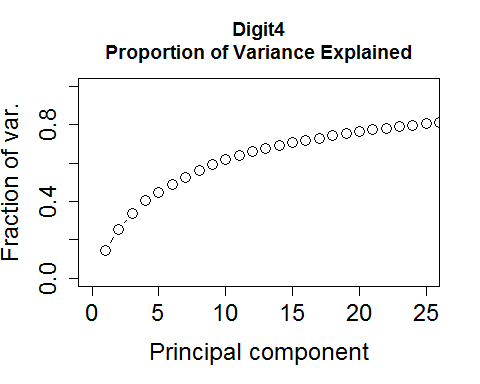
plot(sprcomp1$importance[3,],xlab="Principal component", ylab="Fraction of var.",main="Digit1\nProportion of Variance Explained",ylim=c(0,1), xlim=c(0,25), type='b', cex=1.5, cex.lab=1.5, cex.axis=1.5)

 ####Dataset: Digit 4

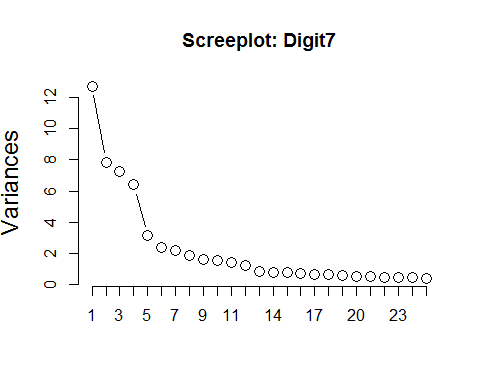
screeplot(prcomp4,npcs=25, type="lines", main="Screeplot: Digit5",cex=1.5, cex.lab=1.5, cex.axis=1.5)



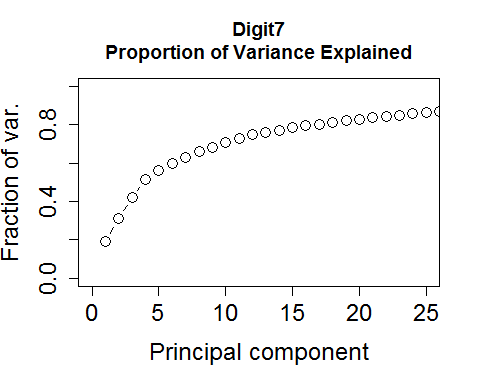
plot(sprcomp4$importance[3,],xlab="Principal component", ylab="Fraction of var.",main="Digit4\nProportion of Variance Explained", ylim=c(0,1), xlim=c(0,25), type='b', cex=1.5, cex.lab=1.5, cex.axis=1.5)

 ####Dataset: Digit 7

screeplot(prcomp7,npcs=25, type="lines", main="Screeplot: Digit7",cex=1.5, cex.lab=1.5, cex.axis=1.5)



plot(sprcomp7$importance[3,],xlab="Principal component", ylab="Fraction of var.",main="Digit7\nProportion of Variance Explained", ylim=c(0,1), xlim=c(0,25), type='b', cex=1.5, cex.lab=1.5, cex.axis=1.5)

 ####Comments: Digits 1, 4 and 7 were selected for the analysis. ####5, 17 and 13 Prinicipal components are required to explain approx. 70% of the variance in each digit respectively. #### Only about 80% of the variance can be explained by including all the PC's for digits 4 and 7 ###6(c)(ii): PCA allows for analyzing images in lower dimensional space. It reduces dimensianlity of ### the dataset by focusing on Principal componenents that account for most variance in the data while ###ignoring the PC's that do not contribute a lot to the variance in images.