**CS6220: Data Mining Homework 01**

**Due: Feb. 17th**

**Name: Mingzhe Xu**

**Problem 1:**

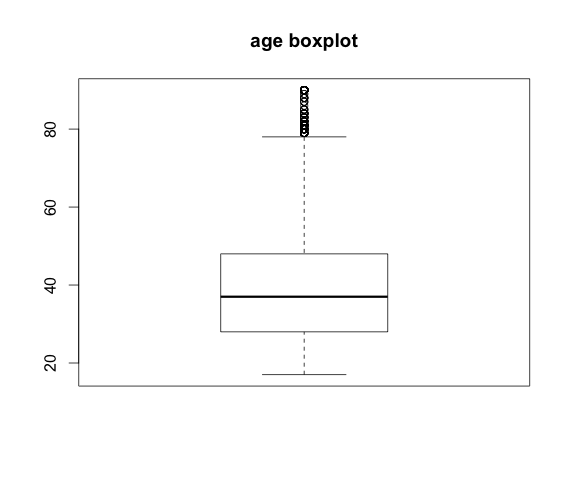
i) Compute mean, min, 1st Quartile, median, 3rd Quartile, Max, mode

(See R code or the snippets at bottom)

ii) Table for summary statistics for continuous variables

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Attribute | Min | Q1 | Median | Q3 | Max | Mean | Mode | Missing |
| Age | 17 | 28 | 37 | 48 | 90 | 38.65 | 36 | 470 |
| Fnlwgt | 12285 | 117549 | 178145 | 237630 | 1490400 | 189664 | 203488 | 0 |
| Edu.num | 1 | 9 | 10 | 12 | 16 | 10 | 9 | 254 |
| Cap.gain | 0 | 0 | 0 | 0 | 99999 | 1079 | 0 | 0 |
| Cap.loss | 0 | 0 | 0 | 0 | 4356 | 87.5 | 0 | 0 |
| Hours | 1 | 40 | 40 | 45 | 99 | 40 | 40 | 737 |

(b) Visualizing Data

ii) Box plot and class-conditional box plot

- Based on the box plot of **age**, we can derive several arguments:

# - the median is 37 so half of the people are below 37, half above.

# - half of the people are from 28 to 48

# - the conditional boxplot:

# -- the median age for income <= 50k is less than 37 and median age for income > 50k is more than 40

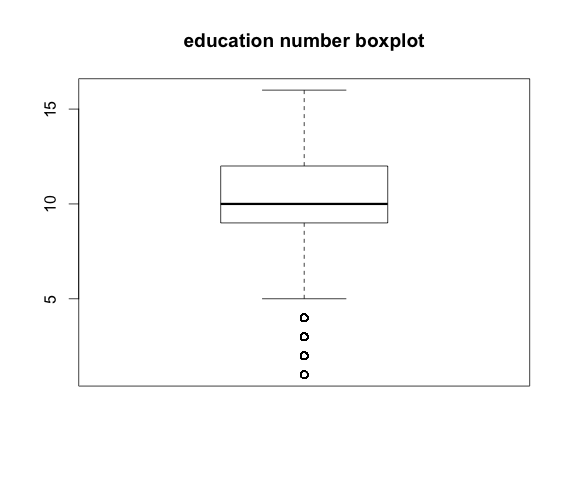


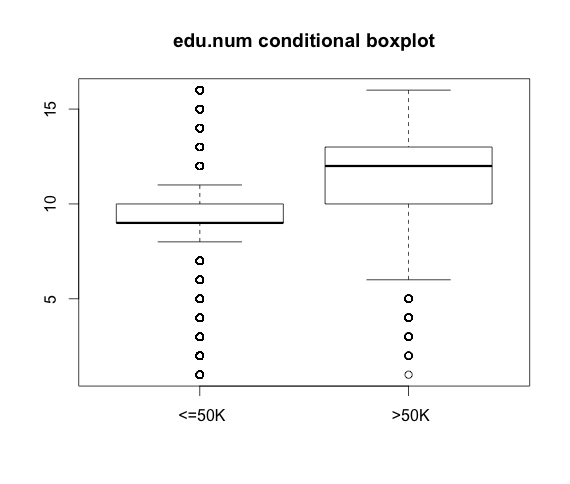
- Based on the box plot of **education number**, we can derive several arguments:

# - the median is 10 and half of the sample is within 9 - 12

# - 25% of the sample is 9 and 25% are 11 or 12

# - median of education number for those income <= 50k is 9, while for income more than 50k, the median education number is obviously more than 10





- Based on the box plot **of hours per week**, we can derive several arguments:

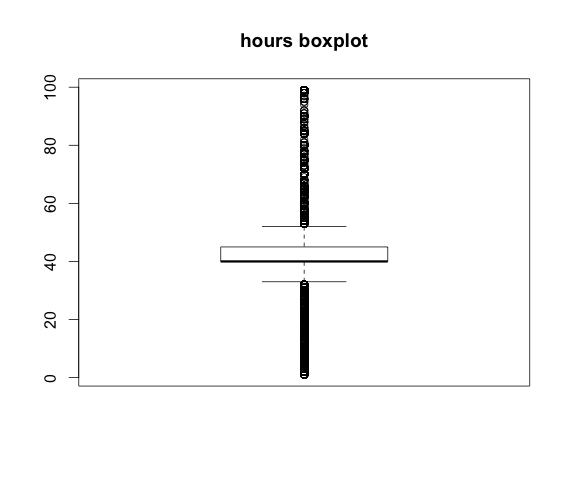
# - the median is 40 and around 25% of the sample is 40

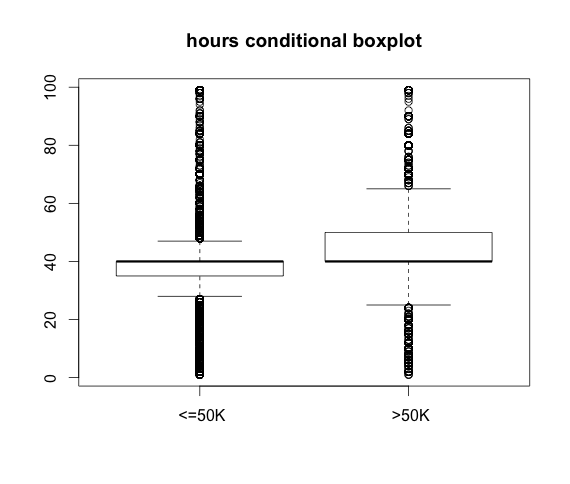
# - 25% of the sample is between 40 and 45

# - the conditional boxplot shows that the median hours from people who has income less or equal to 50k is 40, the same as those who earn more than 50k

# - 75% people whose income is less or equal to 50k work less than or equal to 40 hour

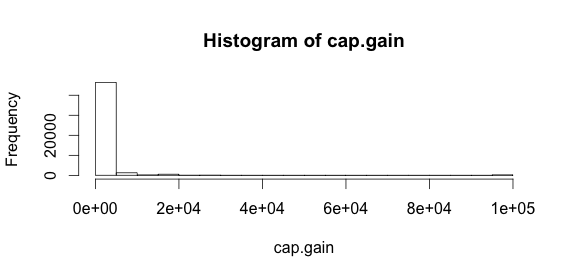
# - 75% people whose income is more than 50k work more than or equal to 40 hours

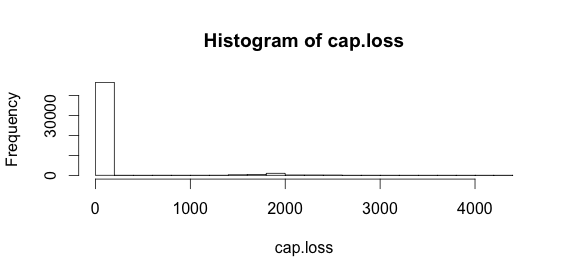




For capital.gain or capital.loss or capital gain, the value is either 0 or a large number and mostly is 0, so we couldn't tell much from the boxplot with median q1 q3 being 0.

iii) Histograms for age, cap.loss, cap.gain





hist(age)

hist(cap.gain)

hist(cap.loss)

# Interpret:

# The histogram tells more than boxplots.

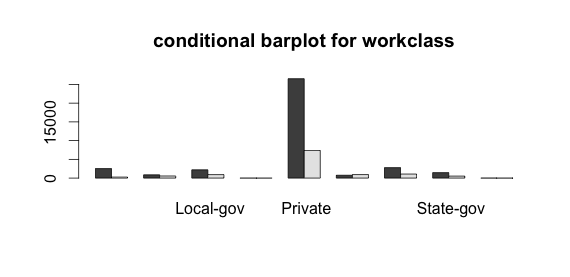
# For age, it is a little bit skewed to the left.

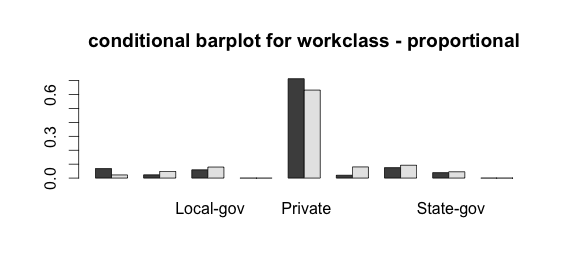
# And there are more people range from age 20 to 45 than other ranges

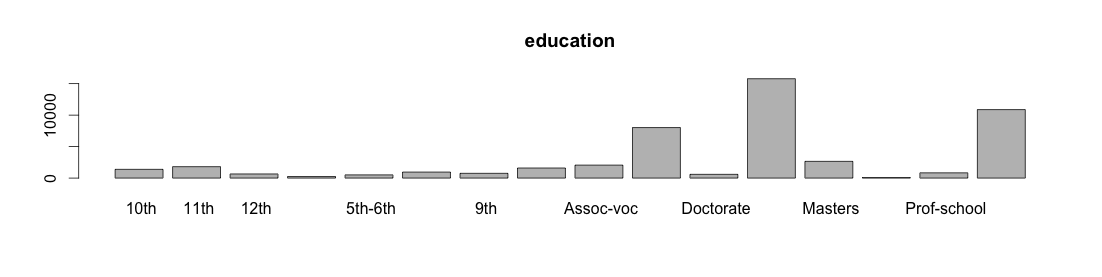
# cap.gain and cap.loss have similar characteristics - both are heavily skewed with most values being 0.

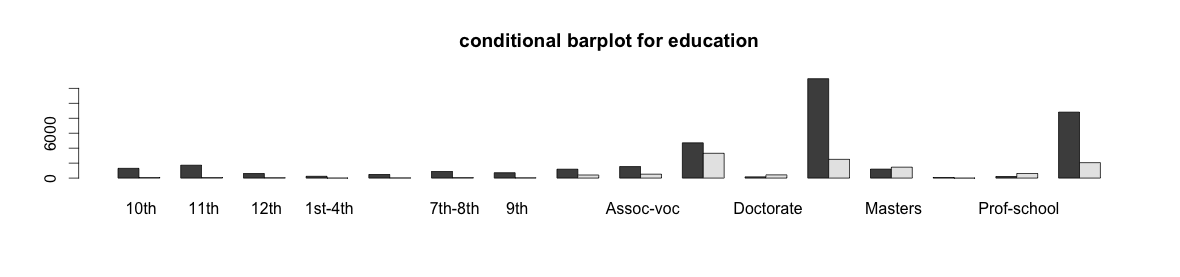
iv) barplots and class conditional bar plots for categorical variables

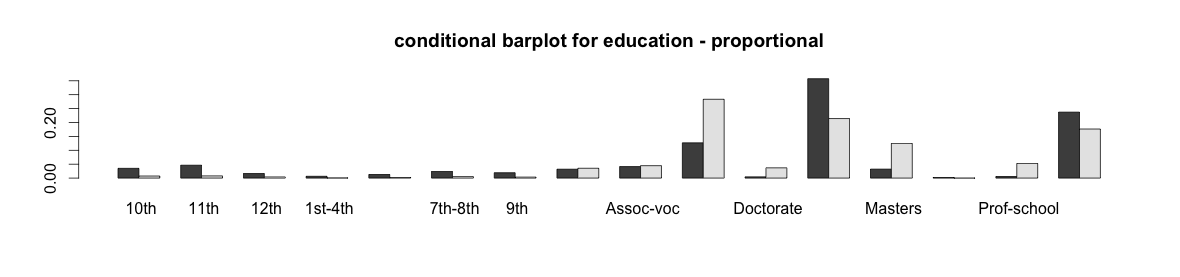


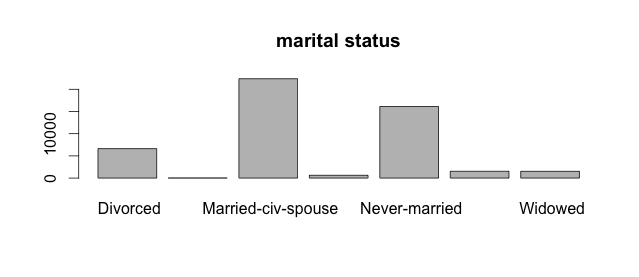


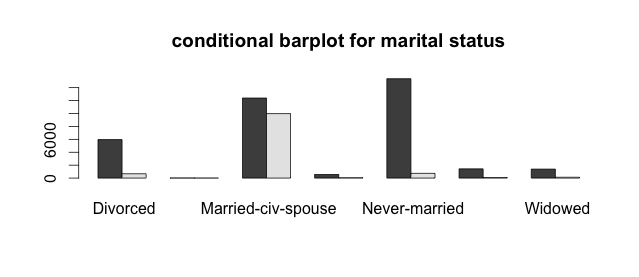


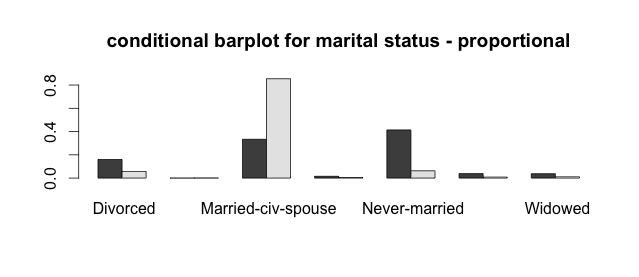




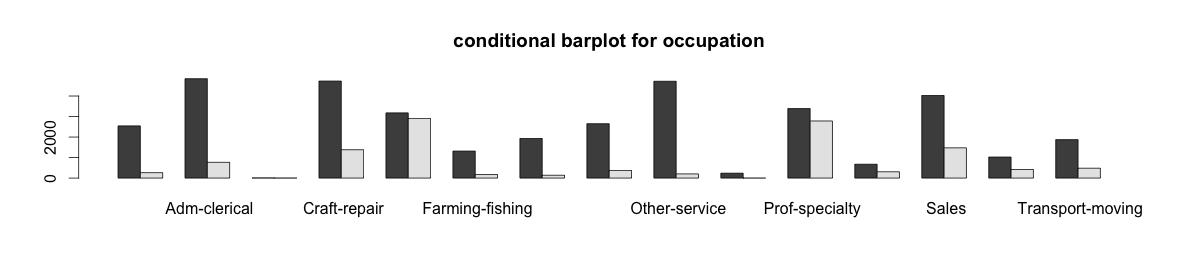


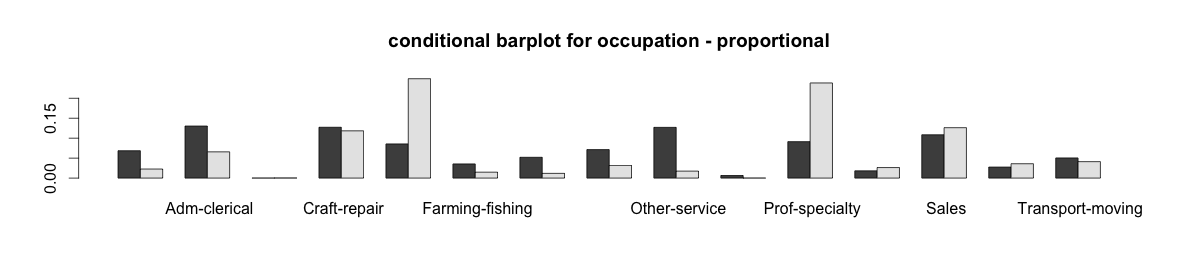


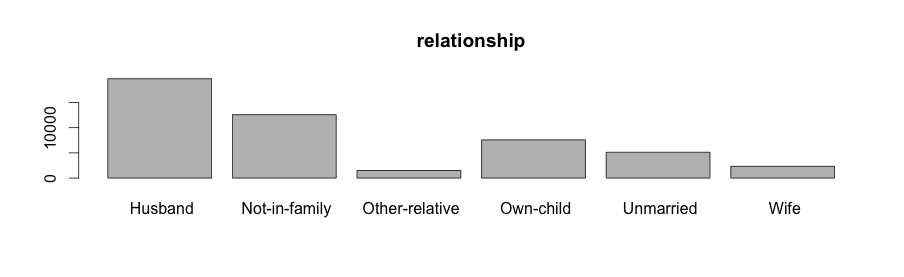


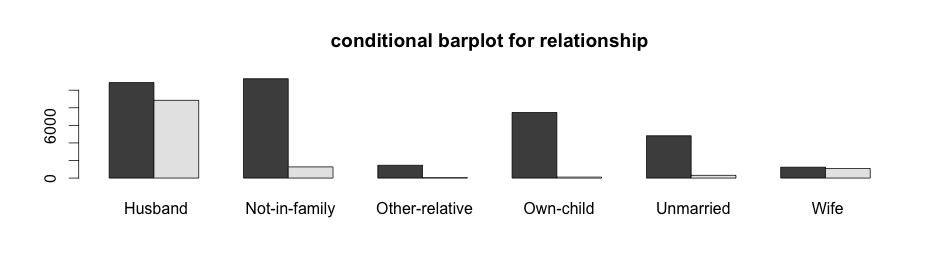


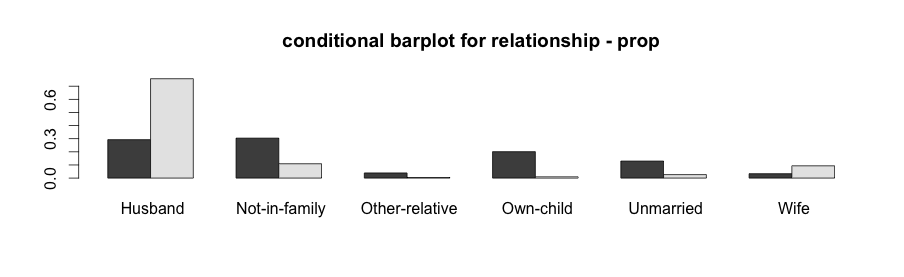


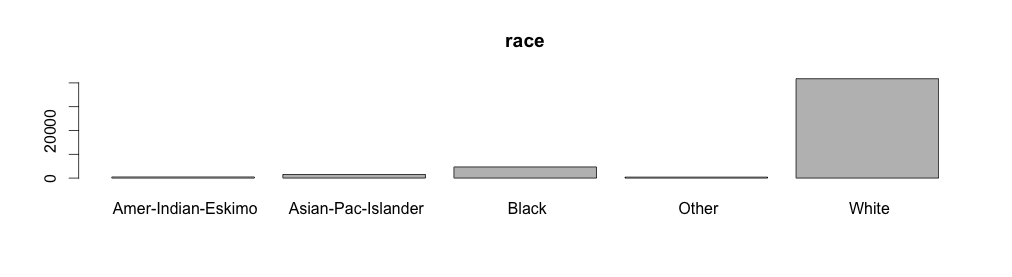


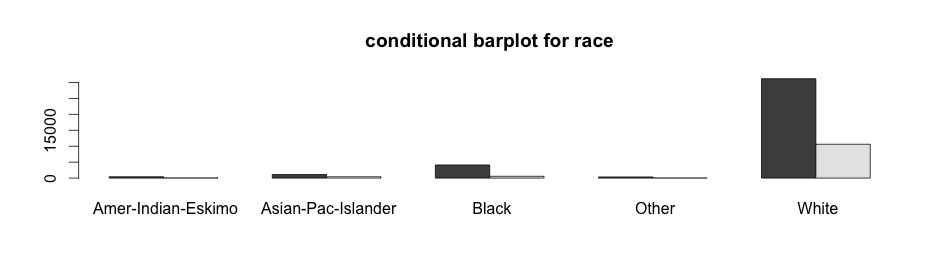


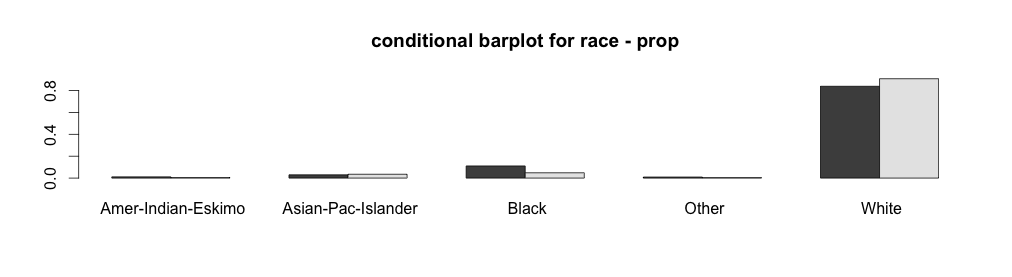


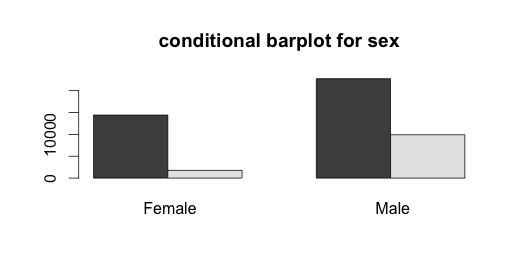


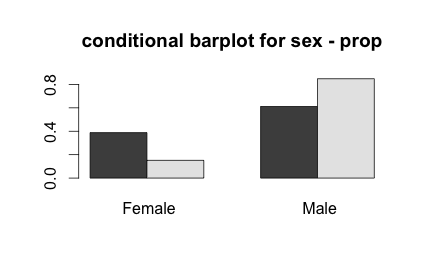


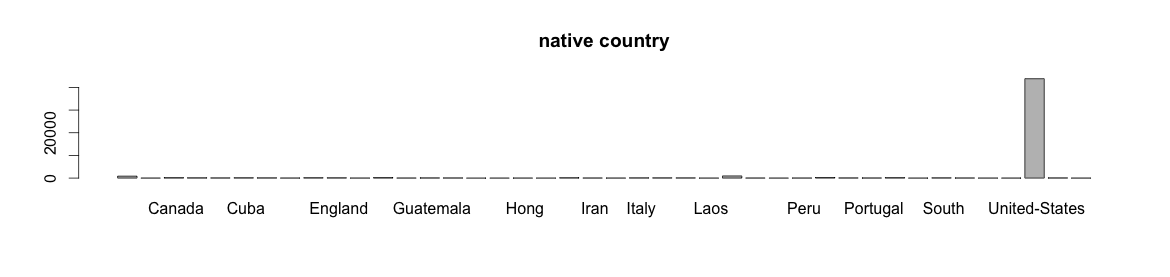


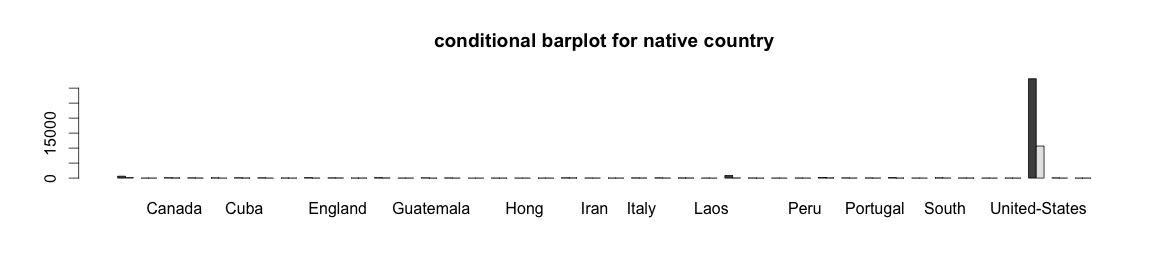


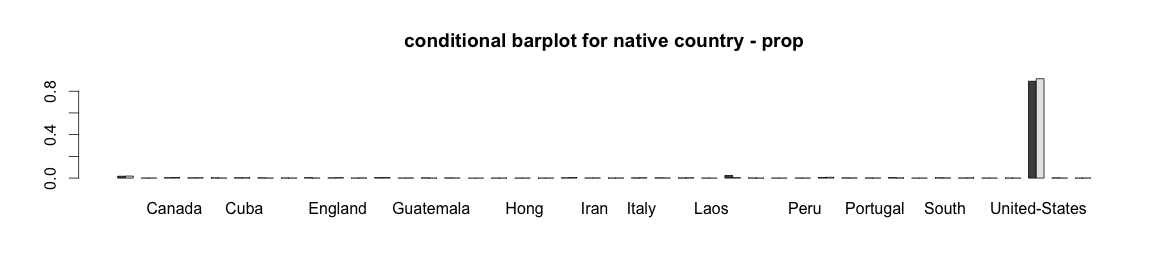












**Problem 2: Item Similarity**

(a) two instances: x1 and x2

i) Euclidean Distance

# sqrt((-1-5)^2 + (6-2)^2 + 3^2 + (-1)^2)

# The result is 7.874008

ii) Manhattan Distance

# abs(-1-5) + abs(6-2)+ abs(3) + abs(-1)

# The result is 14

iii) Minkowski Distance

# h = 0: when h = 0, the distance is (-1-5)^0 + (6-2)^0 + 3^0 + (-1)^0 = 4

# when h = infinity, the result will be the max value of among the 4 dimensions, which is 4

(b) prove that the Euclidean distance is always less than or equal to Manhattan distance

In order to prove it, I need to let the formula to the power of 2 on both side to get rid of the sqrt in Euclidean Distance formula

- Euclidean side: (x1-x2)^2 + (y1-y2)^2 + ...+(d1 - d2)^2

- Manhattan side:(abs(x1-x2) + abs(y1-y2) + ... + abs(d1-d2))^2

Then open the power of 2 for Manhattan. The result will be large than or equal to the value on Euclidean side.

(c) See R code

**Problem 3 finding similar items**

(a)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Element | S1 | S2 | S3 | S4 | H1 | H2 | H3 |
| 0 | 0 | 1 | 0 | 1 | 1 | 2 | 2 |
| 1 | 0 | 1 | 0 | 0 | 3 | 5 | 1 |
| 2 | 1 | 0 | 0 | 1 | 5 | 2 | 0 |
| 3 | 0 | 0 | 1 | 0 | 1 | 5 | 5 |
| 4 | 0 | 0 | 1 | 1 | 3 | 2 | 4 |
| 5 | 1 | 0 | 0 | 0 | 5 | 5 | 3 |

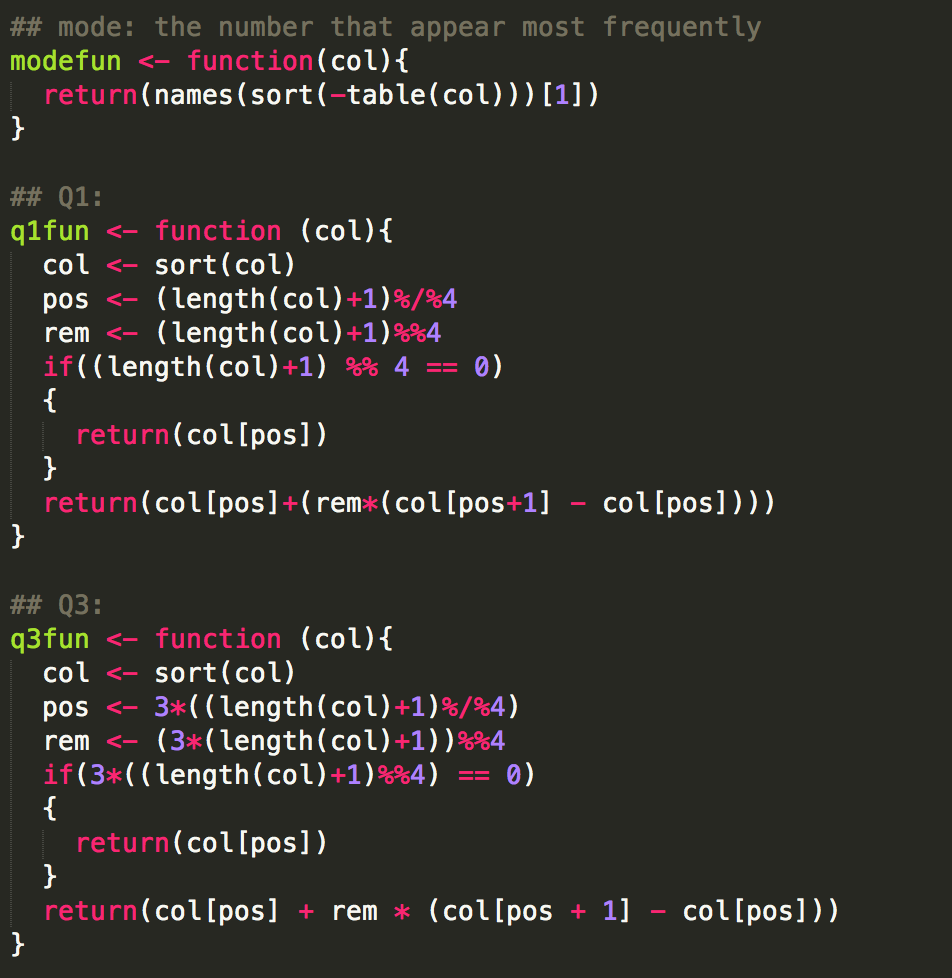
(b) Only h3 is a true permutation

(c)

Some R code snippet for Problem 1:







PROBLEM 2

(C):

meanlessfun <- function(col, adult)

{

sumless = 0

num = 0

less <- adult$income[1]

for(i in 1 : length(col))

{

if(!(is.na(col[i])))

{

if(adult$income[i] == less)

{

sumless = sumless + col[i]

num = num + 1

}

}

}

return(sumless/num)

}

meanmorefun <- function(col, adult)

{

summore = 0

num = 0

less <- adult$income[1]

for(i in 1 : length(col))

{

if(!(is.na(col[i])))

{

if(adult$income[i] != less)

{

summore = summore + col[i]

num = num + 1

}

}

}

return(summore/num)

}

# replace age missing values:

meanlessage = meanlessfun(adult$age, adult)

meanmoreage = meanmorefun(adult$age, adult)

for(i in 1 : length(adult$age))

{

less = adult$income[1]

if(is.na(adult$age[i])){

if(adult$income[i] == less){

adult$age[i] = meanlessage

}else{

adult$age[i] = meanmoreage

}

}

}

# replace education number missing values:

meanlessedu = meanlessfun(adult$education.num, adult)

meanmoreedu = meanmorefun(adult$education.num, adult)

for(i in 1 : length(adult$education.num))

{

less = adult$income[1]

if(is.na(adult$education.num[i])){

if(adult$income[i] == less){

adult$education.num[i] = meanlessedu

}else{

adult$education.num[i] = meanmoreedu

}

}

}

## replace hours missing values:

meanlesshours = meanlessfun(adult$hours.per.week, adult)

meanmorehours = meanmorefun(adult$hours.per.week, adult)

for(i in 1 : length(adult$hours.per.week))

{

less = adult$income[1]

if(is.na(adult$hours.per.week[i])){

if(adult$income[i] == less){

adult$hours.per.week[i] = meanlesshours

}else{

adult$hours.per.week[i] = meanmorehours

}

}

}

## ii) standardize the values

standardfun <- function(col){

mean = meanfun(col)

sd = sd(col)

for(i in 1 : length(col))

{

col[i] = (col[i] - mean) / sd

}

return(col)

}

# to standardize the features:

adult$age = standardfun(adult$age)

adult$education.num = standardfun(adult$education.num)

adult$hours.per.week = standardfun(adult$hours.per.week)

## iii) find neighbor

## the inputs:

# data: the input data set

# mean is a list of mean values for age, education\_num and hours.per.week

# meany is a matrix denote the conditional mean value

# stddev is the standard deviation

# id is the target id

# standardize is a boolean true if the data is standarized, false otherwise

# h is the h in Minkowski's distance

findneighborfun <- function(data, id, standardize, h)

{

if(!standardize){

data$age = standardfun(data$age)

data$education.num = standardfun(data$education.num)

data$hours.per.week = standardfun(data$hours.per.week)

}

mindist = 214748364;

minid = id

targetage <- data[data[, 1] == id,]$age

targetedu <- data[data[, 1] == id,]$education.num

targethours <- data[data[, 1] == id,]$hours.per.week

for(i in 1 : nrow(data))

{

if(!data$id[i] == id)

{

disth <- (data$age[i] - targetage)^h+(data$education.num[i] - targetedu)^h+(data$hours.per.week[i] - targethours)^h

dist <- disth^(1/h)

if(dist < mindist)

{

mindist <- dist

minid <- data$id[i]

}

}

}

return(c(minid, mindist))

}