Homework 2

Jinhai,Chen,3170104884

The data set calif_penn_2011.csv contains information about the housing stock of California and Pennsylvania, as of 2011. Information as aggregated into "Census tracts", geographic regions of a few thousand people which are supposed to be fairly homogeneous economically and socially.

1. *Loading and cleaning*

a. Load the data into a dataframe called `ca_pa`.

```
ca_pa <- read.csv("Data/calif_penn_2011.csv",header = T,sep=",")
```

b. How many rows and columns does the dataframe have?

```
row1 <- dim(ca_pa)[1]
column1 <- dim(ca_pa)[2]
```

c. Run this command, and explain, in words, what this does: It acually figures out the number of missing values is every column as follows:

```
colSums(apply(ca_pa,c(1,2),is.na))
```

```
##                            X                     GEO.id2
##                            0                           0
##                      STATEFP                    COUNTYFP
##                            0                           0
##                      TRACTCE                  POPULATION
##                            0                           0
##                     LATITUDE                   LONGITUDE
##                            0                           0
##            GEO.display.label           Median_house_value
##                            0                         599
##                  Total_units                Vacant_units
##                            0                           0
##                 Median_rooms  Mean_household_size_owners
##                          157                         215
## Mean_household_size_renters          Built_2005_or_later
##                          152                          98
##            Built_2000_to_2004                 Built_1990s
##                           98                          98
##                  Built_1980s                 Built_1970s
##                           98                          98
##                  Built_1960s                 Built_1950s
##                           98                          98
##                  Built_1940s       Built_1939_or_earlier
##                           98                          98
##                   Bedrooms_0                  Bedrooms_1
##                           98                          98
##                   Bedrooms_2                  Bedrooms_3
```

```
##                        98                          98
##                Bedrooms_4           Bedrooms_5_or_more
##                        98                          98
##                    Owners                     Renters
##                       100                         100
##   Median_household_income       Mean_household_income
##                       115                         126
```

d. The function `na.omit()` takes a dataframe and returns a new dataframe, omitting any row containing an NA value. Use it to purge the data set of rows with incomplete data.

```r
ca_pa <- na.omit(ca_pa)
```

e. How many rows did this eliminate?

```r
row2 <- dim(ca_pa)[1]
abs(row2-row1)
```

```
## [1] 670
```

f. Are your answers in (c) and (e) compatible? Explain. Positive with following codes.

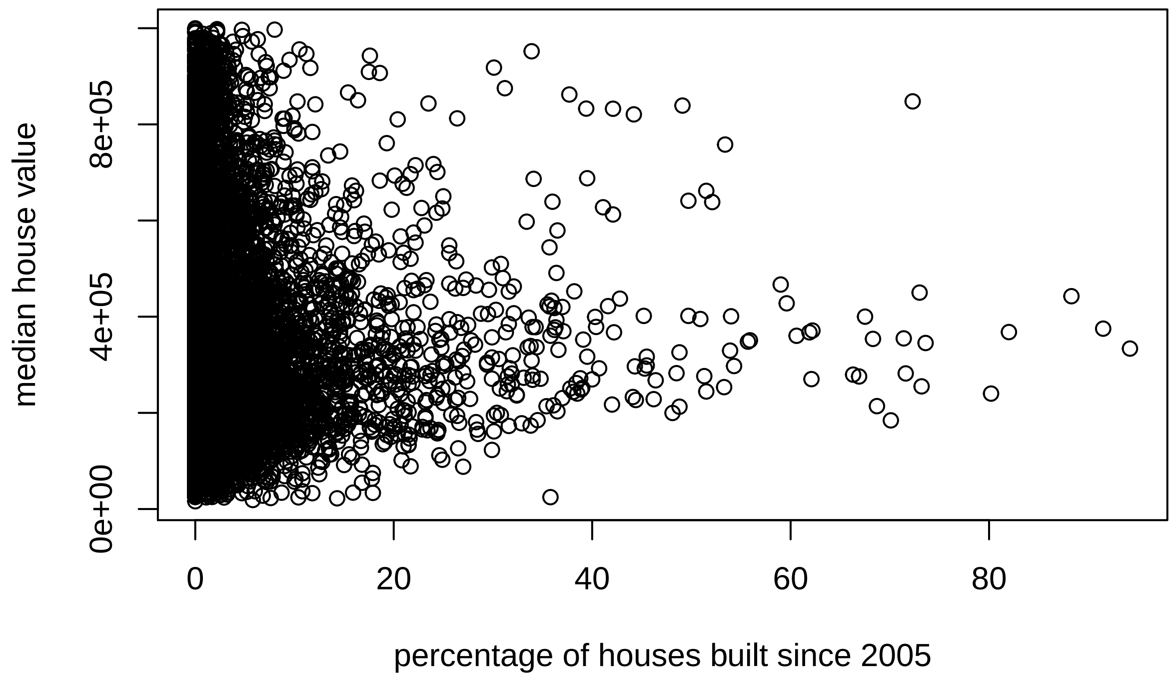
```r
sum(colSums(apply(ca_pa,c(1,2),is.na)))
```

```
## [1] 0
```

2. *This Very New House*

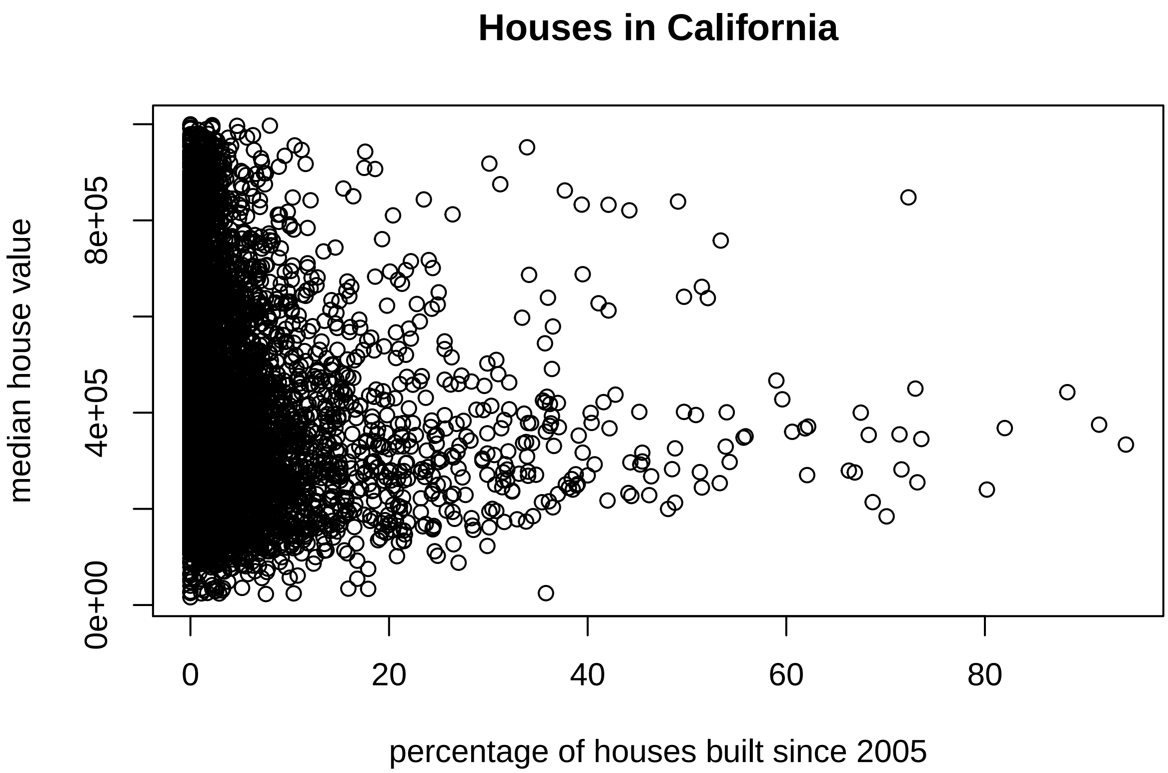
a. The variable `Built_2005_or_later` indicates the percentage of houses in each Census tract built since 2005. Plot median house prices against this variable.

```r
plot(ca_pa$Median_house_value ~ ca_pa$Built_2005_or_later,
     xlab="percentage of houses built since 2005",ylab="median house value")
```

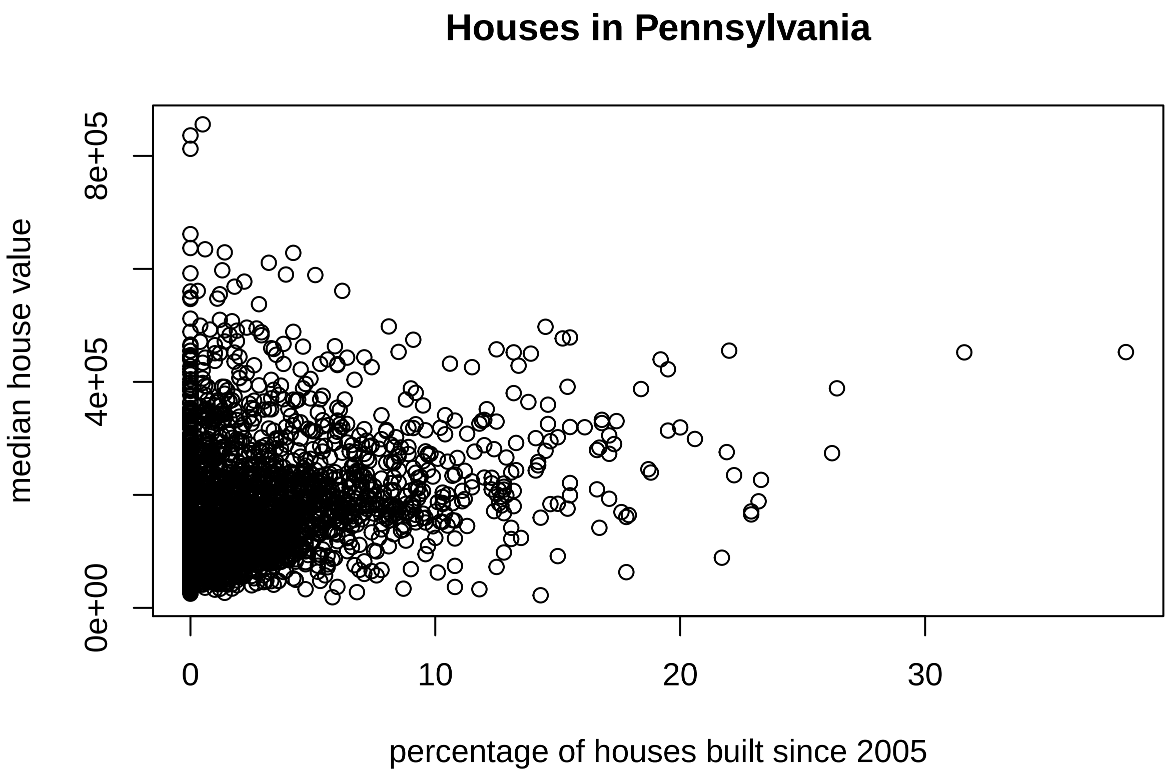


b. Make a new plot, or pair of plots, which breaks this out by state. Note that the state is recorded in the `STATEFP` variable, with California being state 6 and Pennsylvania state 42.

```
plot(ca_pa$Built_2005_or_later[ca_pa$STATEFP==6],ca_pa$Median_house_value[ca_pa$STATEFP==6],
     xlab="percentage of houses built since 2005",
     ylab="median house value",main="Houses in California")
```




```
plot(ca_pa$Built_2005_or_later[ca_pa$STATEFP==42],ca_pa$Median_house_value[ca_pa$STATEFP==42],
     xlab="percentage of houses built since 2005",
     ylab="median house value",main="Houses in Pennsylvania")
```



3. *Nobody Home*
   The vacancy rate is the fraction of housing units which are not occupied. The dataframe contains columns giving the total number of housing units for each Census tract, and the number of vacant housing units.

a. Add a new column to the dataframe which contains the vacancy rate. What are the minimum, maximum, mean, and median vacancy rates?

```
vac_rate <- ca_pa$Vacant_units / ca_pa$Total_units
ca_pa <- data.frame(ca_pa,vac_rate)
min(vac_rate)
```

```
## [1] 0
```

```
max(vac_rate)
```

```
## [1] 0.965311
```

```
mean(vac_rate)
```

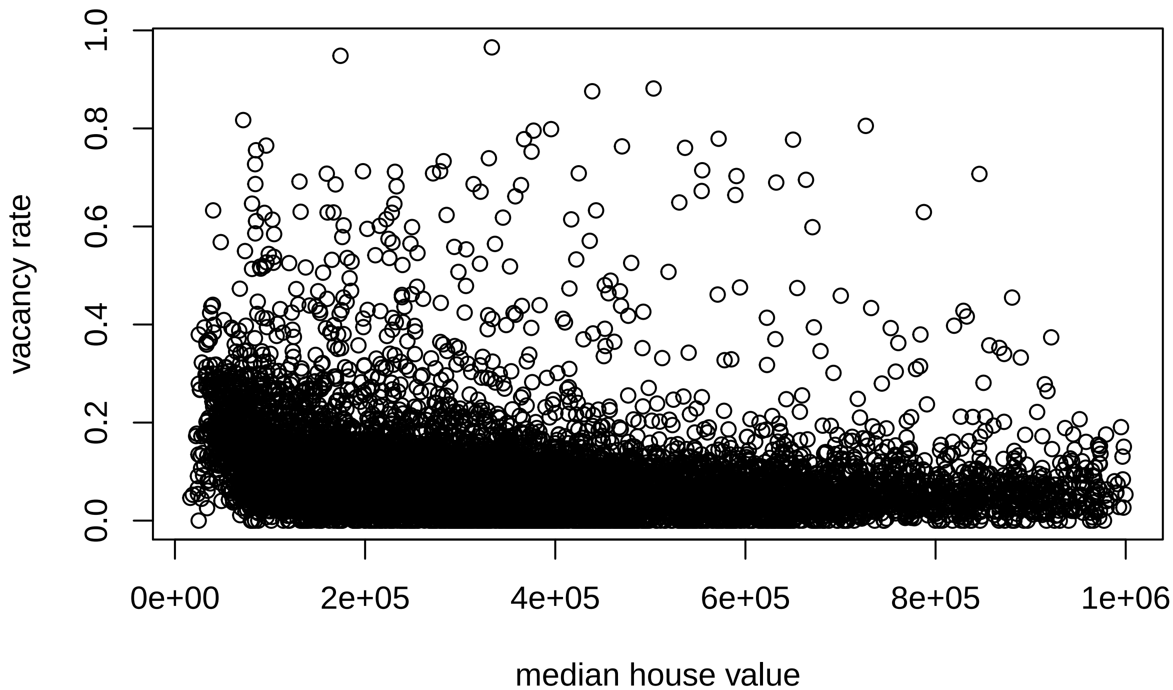
```
## [1] 0.08888789
```

```
median(vac_rate)
```

```
## [1] 0.06767283
```

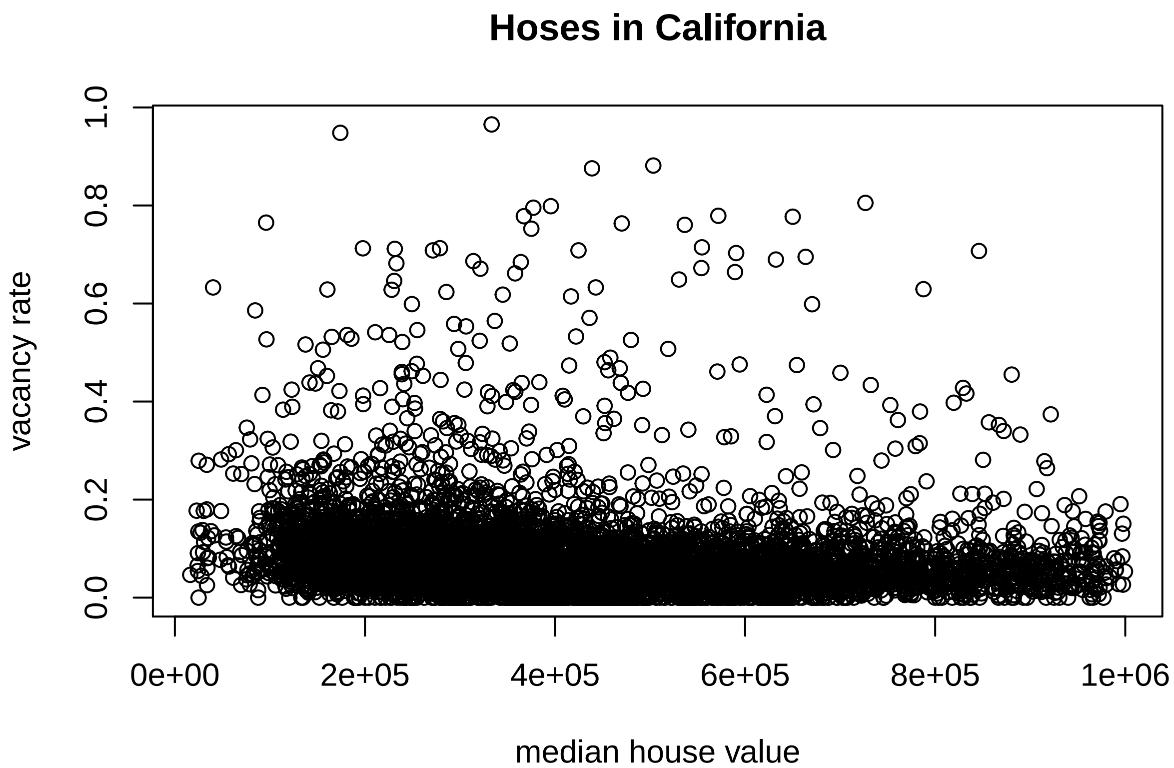
b. Plot the vacancy rate against median house value.

```
plot(ca_pa$Median_house_value,ca_pa$vac_rate,
     xlab="median house value",ylab="vacancy rate")
```

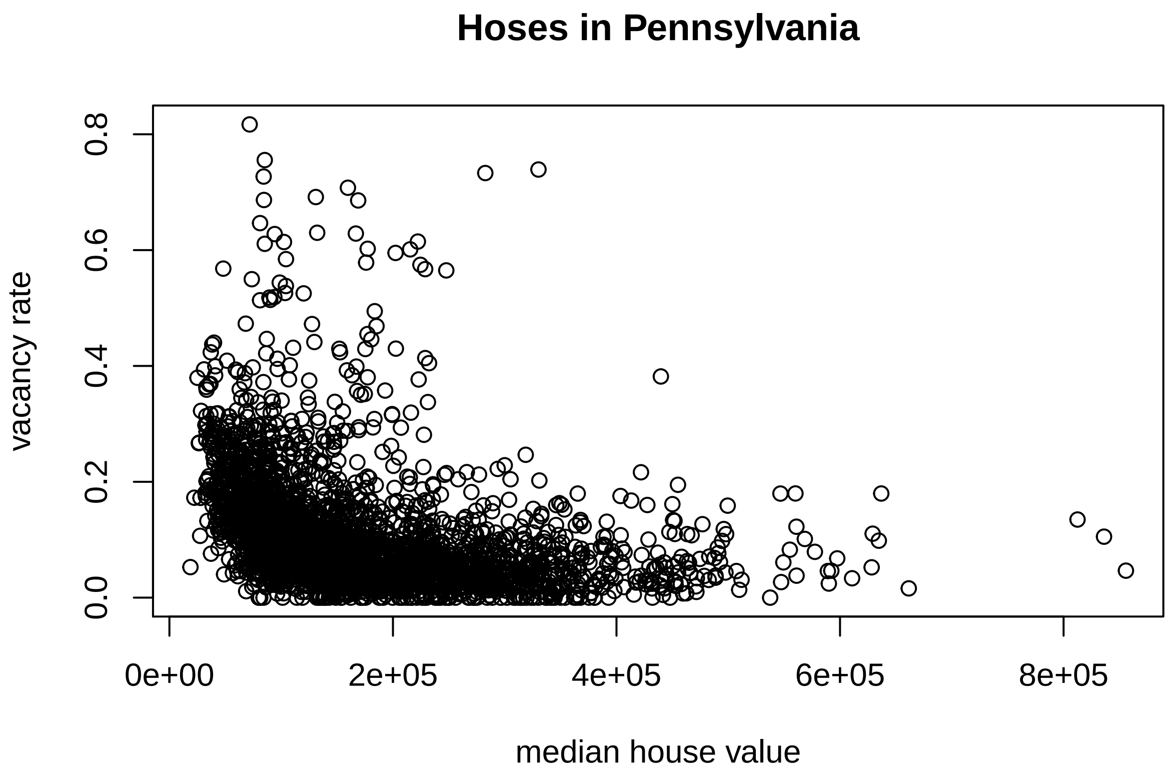


c. Plot vacancy rate against median house value separately for California and for Pennsylvania. Is there a difference?

```
plot(ca_pa$Median_house_value[ca_pa$STATEFP==6],ca_pa$vac_rate[ca_pa$STATEFP==6],
     xlab="median house value",ylab="vacancy rate",main="Hoses in California")
```




```
plot(ca_pa$Median_house_value[ca_pa$STATEFP==42],ca_pa$vac_rate[ca_pa$STATEFP==42],
     xlab="median house value",ylab="vacancy rate",main="Hoses in Pennsylvania")
```



There are of course differences. The median house value in California is much higher than that in Pennsylvania. In Canlifornia people shares all kinds of houses, as a result of which almost any house of any median value has some samples of vacancy; while in Pennsylvania, houses of low midian value has higher vacancy rate than those of high median value.

4. The column `COUNTYFP` contains a numerical code for counties within each state. We are interested in Alameda County (county 1 in California), Santa Clara (county 85 in California), and Allegheny County (county 3 in Pennsylvania).

a. Explain what the block of code at the end of this question is supposed to accomplish, and how it does it. It is supposed to pick up the tracts in different county and compute median of the median house value of the tracts.

b. Give a single line of R which gives the same final answer as the block of code. Note: there are at least two ways to do this; you just have to find one.

```
median(ca_pa$Median_house_value[ca_pa$STATEFP==6 & ca_pa$COUNTYFP==1])
```

```
## [1] 474050
```

c. For Alameda, Santa Clara and Allegheny Counties, what were the average percentages of housing built since 2005?

```
#Alameda
mean(ca_pa$Built_2005_or_later[ca_pa$STATEFP==6 & ca_pa$COUNTYFP==1])
```

```
## [1] 2.820468
```

```
#Santa Clara
mean(ca_pa$Built_2005_or_later[ca_pa$STATEFP==6 & ca_pa$COUNTYFP==85])
```

```
## [1] 3.200319
```

```r
#Alleghency
mean(ca_pa$Built_2005_or_later[ca_pa$STATEFP==42 & ca_pa$COUNTYFP==3])
```

```
## [1] 1.474219
```

d. The `cor` function calculates the correlation coefficient between two variables. What is the correlation between median house value and the percent of housing built since 2005 in (i) the whole data, (ii) all of California, (iii) all of Pennsylvania, (iv) Alameda County, (v) Santa Clara County and (vi) Allegheny County?

```r
#whole data
cor(ca_pa$Built_2005_or_later,ca_pa$Median_house_value)
```

```
## [1] -0.01893186
```

```r
#California
cor(ca_pa$Built_2005_or_later[ca_pa$STATEFP==6],
    ca_pa$Median_house_value[ca_pa$STATEFP==6])
```

```
## [1] -0.1153604
```

```r
#Pennsylvania
cor(ca_pa$Built_2005_or_later[ca_pa$STATEFP==42],
    ca_pa$Median_house_value[ca_pa$STATEFP==42])
```

```
## [1] 0.2681654
```

```r
#Alameda
cor(ca_pa$Built_2005_or_later[ca_pa$STATEFP==6 & ca_pa$COUNTYFP==1],
    ca_pa$Median_house_value[ca_pa$STATEFP==6 & ca_pa$COUNTYFP==1])
```

```
## [1] 0.01303543
```

```r
#Santa Clara
cor(ca_pa$Built_2005_or_later[ca_pa$STATEFP==6 & ca_pa$COUNTYFP==85],
    ca_pa$Median_house_value[ca_pa$STATEFP==6 & ca_pa$COUNTYFP==85])
```

```
## [1] -0.1726203
```

```r
#Allegheny
cor(ca_pa$Built_2005_or_later[ca_pa$STATEFP==42 & ca_pa$COUNTYFP==3],
    ca_pa$Median_house_value[ca_pa$STATEFP==42 & ca_pa$COUNTYFP==3])
```

```
## [1] 0.1939652
```

e. Make three plots, showing median house values against median income, for Alameda, Santa Clara, and Allegheny Counties. (If you can fit the information into one plot, clearly distinguishing the three counties, that's OK too.)

```r
#Alameda
plot(ca_pa$Median_household_income[ca_pa$STATEFP==6 & ca_pa$COUNTYFP==1],
     ca_pa$Median_house_value[ca_pa$STATEFP==6 & ca_pa$COUNTYFP==1],
     xlab="Median house value",ylab="Median income",main="Houses in Alameda")

#Santa Clara
plot(ca_pa$Median_household_income[ca_pa$STATEFP==6 & ca_pa$COUNTYFP==85],
     ca_pa$Median_house_value[ca_pa$STATEFP==6 & ca_pa$COUNTYFP==85],
     xlab="Median house value",ylab="Median income",main="Houses in Santa Clara")
```

```r
#Allegheny
plot(ca_pa$Median_household_income[ca_pa$STATEFP==42 & ca_pa$COUNTYFP==3],
     ca_pa$Median_house_value[ca_pa$STATEFP==42 & ca_pa$COUNTYFP==3],
     xlab="Median house value",ylab="Median income",main="Houses in Allegheny")

acca <- c()
for (tract in 1:nrow(ca_pa)) {
  if (ca_pa$STATEFP[tract] == 6) {
    if (ca_pa$COUNTYFP[tract] == 1) {
      acca <- c(acca, tract)
    }
  }
}
accamhv <- c()
for (tract in acca) {
  accamhv <- c(accamhv, ca_pa[tract,10])
}
median(accamhv)
```

MB.Ch1.11. Run the following code:

```r
gender <- factor(c(rep("female", 91), rep("male", 92)))
table(gender)
```

```
## gender
## female   male
##     91     92
```

```r
gender <- factor(gender, levels=c("male", "female"))
table(gender)
```

```
## gender
##   male female
##     92     91
```

```r
gender <- factor(gender, levels=c("Male", "female"))
# Note the mistake: "Male" should be "male"
table(gender)
```

```
## gender
##   Male female
##      0     91
```

```r
table(gender, exclude=NULL)
```

```
## gender
##   Male female   <NA>
##      0     91     92
```

```r
rm(gender)  # Remove gender
```

Explain the output from the successive uses of table().

Function `table()` is used for displaying number of labels in a factor. The first part consists 91 females and 92 males, so `table(gender)` displays the number of female and males. For the second code, it changed the levels of gender, leading to the exchange of the males and females in the output. For the third one, the wrong input `Male` leads to the 0 for Male, with male of 92 and female for 91. The last one displays all data including those without levels.

MB.Ch1.12. Write a function that calculates the proportion of values in a vector x that exceed some value cutoff.

```
cutoff_proportion <- function(x,cutoff){
  return(sum(x>cutoff) / length(x))
}
```

(a) Use the sequence of numbers 1, 2, . . . , 100 to check that this function gives the result that is expected.

```
cutoff_proportion(seq(1,100),50)
```

```
## [1] 0.5
```

(b) Obtain the vector ex01.36 from the Devore6 (or Devore7) package. These data give the times required for individuals to escape from an oil platform during a drill. Use dotplot() to show the distribution of times. Calculate the proportion of escape times that exceed 7 minutes.

```
library(Devore7)
```

```
## Loading required package: MASS

##
## Attaching package: 'MASS'

## The following object is masked from 'package:DAAG':
##
##     hills

## The following object is masked from 'package:dplyr':
##
##     select
```

```
ex1 <- Devore7::ex01.36
```

MB.Ch1.18. The Rabbit data frame in the MASS library contains blood pressure change measurements on five rabbits (labeled as R1, R2, . . . ,R5) under various control and treatment conditions. Read the help file for more information. Use the unstack() function (three times) to convert Rabbit to the following form:

Treatment Dose R1 R2 R3 R4 R5

1 Control 6.25 0.50 1.00 0.75 1.25 1.5

2 Control 12.50 4.50 1.25 3.00 1.50 1.5

....

```
library(MASS)
Rabbit <- MASS::Rabbit
treatment <- unstack(Rabbit, Treatment ~ Animal)
dose <- unstack(Rabbit, Dose ~ Animal)
BPc <- unstack(Rabbit, BPchange ~ Animal)
Rabbit <- data.frame(treatment[,1],dose[,1],BPc)
name <- c("Treatment","Dose","R1","R2","R3","R4","R5")
names(Rabbit) <- name
Rabbit
```

```
##    Treatment   Dose    R1    R2    R3    R4   R5
## 1    Control   6.25  0.50  1.00  0.75  1.25  1.5
## 2    Control  12.50  4.50  1.25  3.00  1.50  1.5
## 3    Control  25.00 10.00  4.00  3.00  6.00  5.0
## 4    Control  50.00 26.00 12.00 14.00 19.00 16.0
## 5    Control 100.00 37.00 27.00 22.00 33.00 20.0
```

```
## 6    Control 200.00 32.00 29.00 24.00 33.00 18.0
## 7        MDL   6.25  1.25  1.40  0.75  2.60  2.4
## 8        MDL  12.50  0.75  1.70  2.30  1.20  2.5
## 9        MDL  25.00  4.00  1.00  3.00  2.00  1.5
## 10       MDL  50.00  9.00  2.00  5.00  3.00  2.0
## 11       MDL 100.00 25.00 15.00 26.00 11.00  9.0
## 12       MDL 200.00 37.00 28.00 25.00 22.00 19.0
```