Homework 2: Housing Price

1. *Loading and cleaning*

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

```r
ca_pa<-read.csv("data/calif_penn_2011.csv")
```

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

```r
ncol(ca_pa)
```

```
## [1] 34
```

```r
nrow(ca_pa)
```

```
## [1] 11275
```

```
c. Run this command, and explain, in words, what this does:
```

```r
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
```

```
#it returns the number of cell with "NA" in the data frame.
```

```
d. The function ‘na.omit()‘ takes a dataframe and returns a new dataframe, omitting any row containing
```

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

```
e. How many rows did this eliminate?
```

```
nrow(ca_pa)-nrow(ca_pa.omit)
```

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

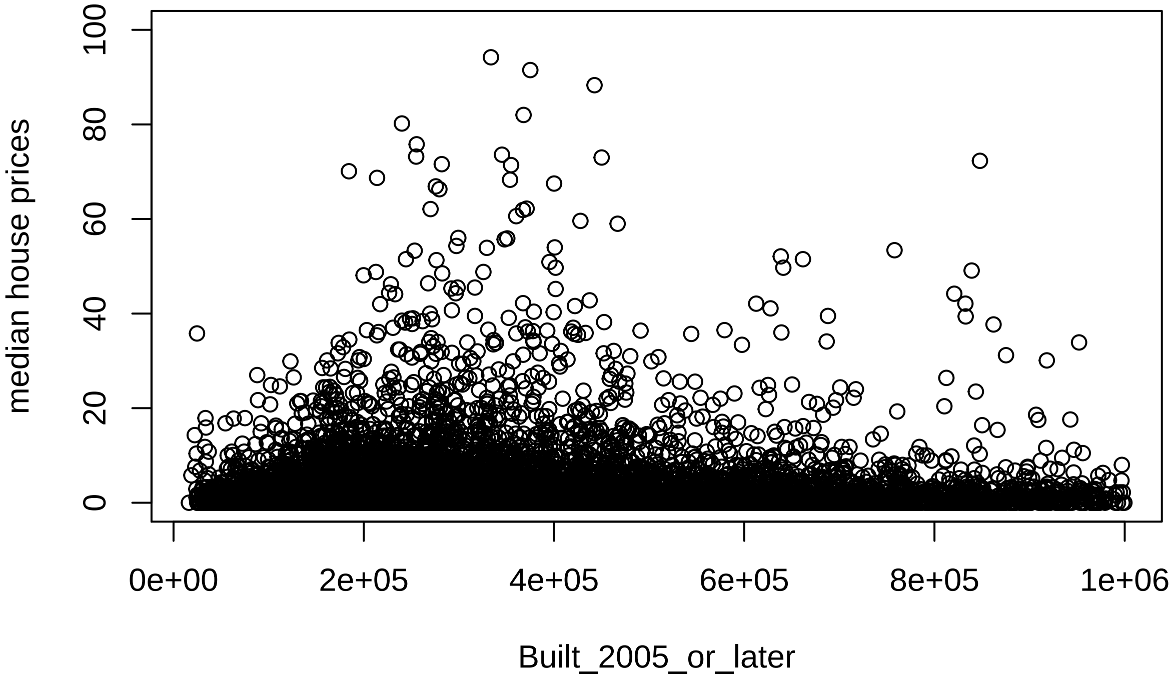
```
#670 rows.
```

```
f. Are your answers in (c) and (e) compatible? Explain.
No, because there are more than one NA in some rows.
```

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.

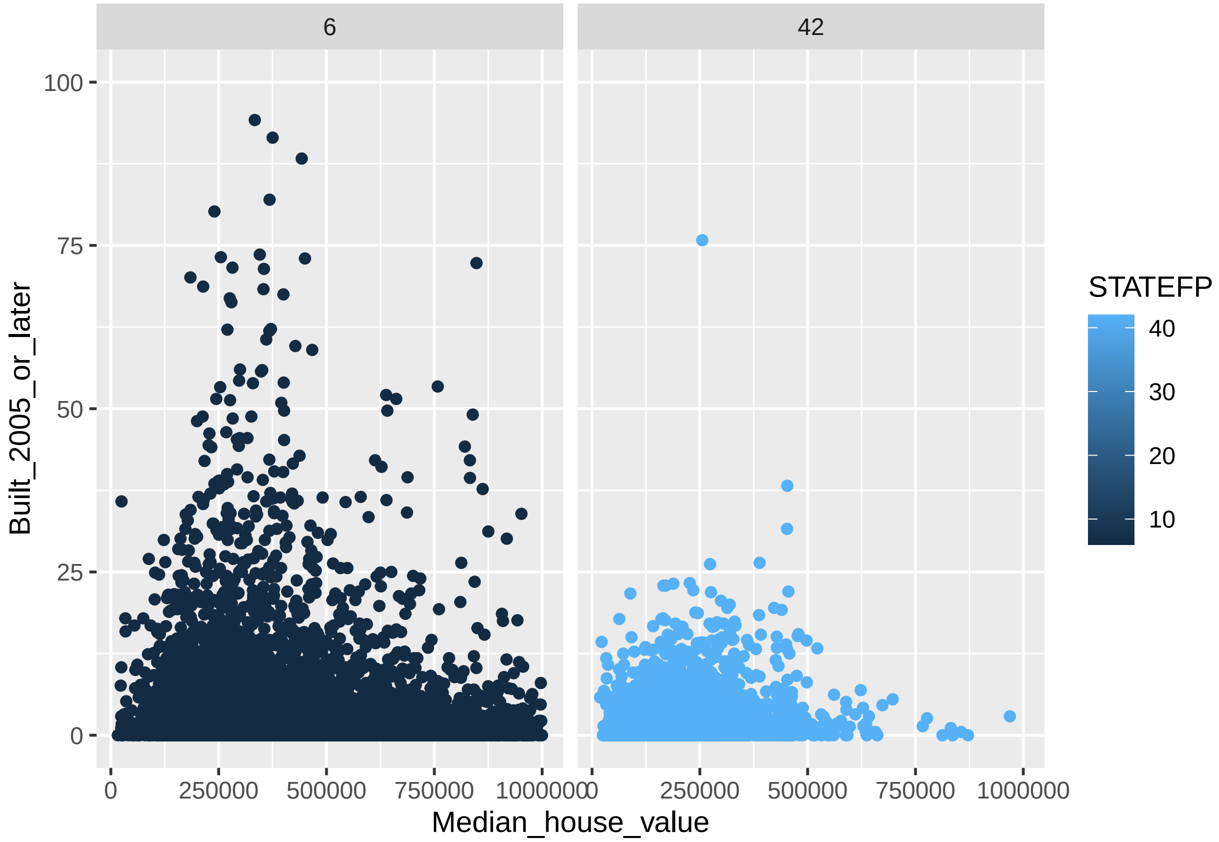
```
ca_pa.05<-ca_pa$Built_2005_or_later
ca_pa.me<-ca_pa$ Median_house_value
plot(ca_pa.me,ca_pa.05,xlab = "Built_2005_or_later",ylab = "median house prices")
```



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.

```
ggplot(data = ca_pa) +
    geom_point(aes(x =Median_house_value, y = Built_2005_or_later, color = STATEFP))+
    facet_wrap(~ STATEFP)
```

```
## Warning: Removed 599 rows containing missing values (geom_point).
```



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.

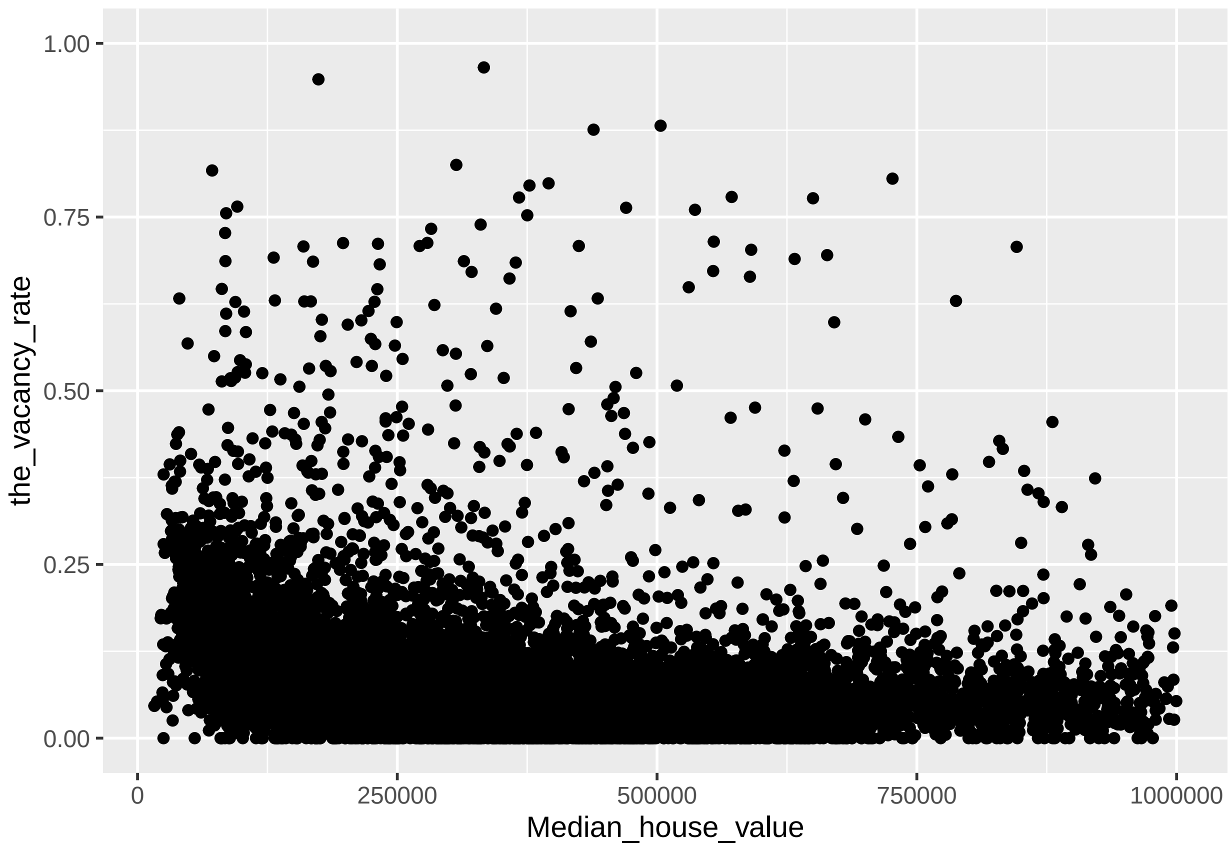
   a. Add a new column to the dataframe which contains the vacancy rate. What are the minimum, maximum, mean, and median vacancy rates?

```
ca_pa.new <- mutate(ca_pa,the_vacancy_rate = Vacant_units / Total_units)
```

```
b. Plot the vacancy rate against median house value.
```

```
ggplot(data = ca_pa.new) +
    geom_point(aes(x =Median_house_value, y = the_vacancy_rate))
```

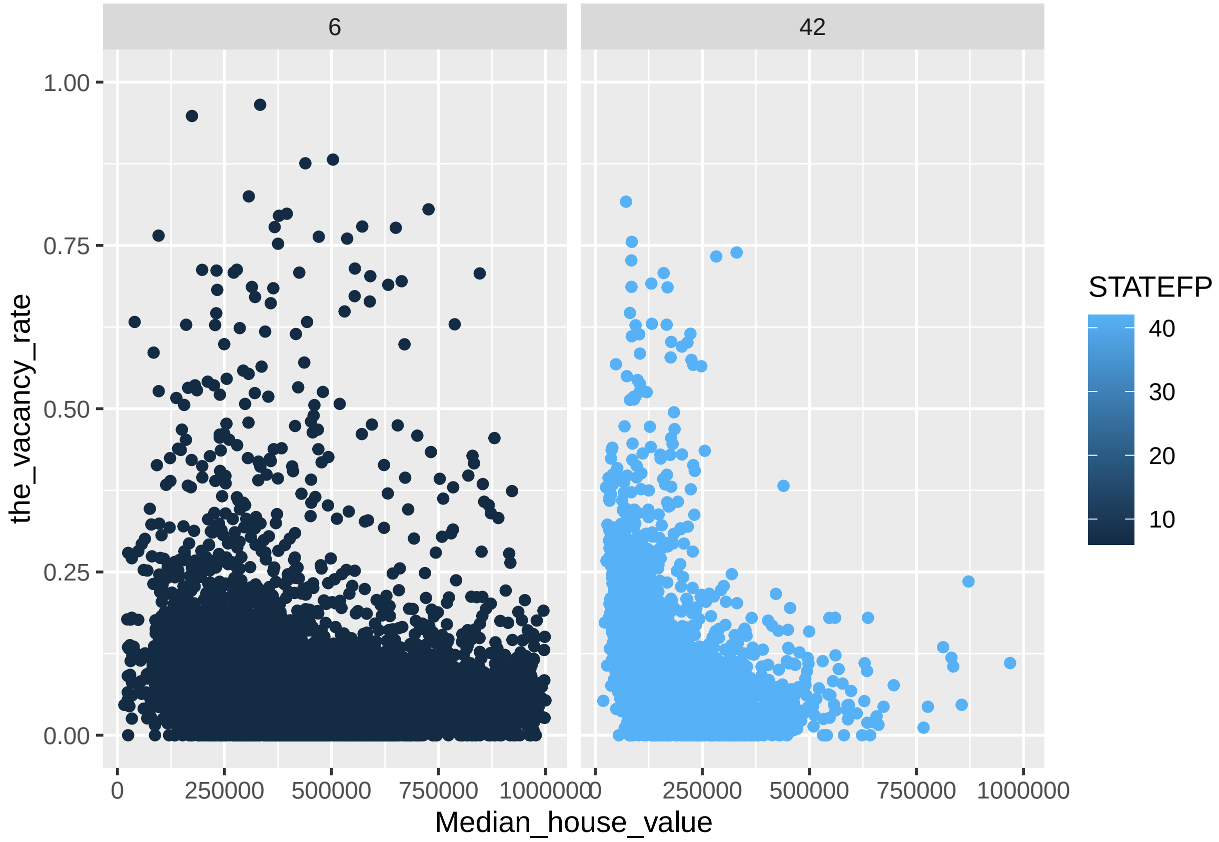
```
## Warning: Removed 599 rows containing missing values (geom_point).
```



c. Plot vacancy rate against median house value separately for California and for Pennsylvania. Is ther

```
ggplot(data = ca_pa.new)+
  geom_point(aes(x = Median_house_value, y = the_vacancy_rate,color = STATEFP)) +
  facet_wrap(~ STATEFP)
```

```
## Warning: Removed 599 rows containing missing values (geom_point).
```



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.

```r
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)
```

```
## [1] NA
```

```r
#this code sifts all the stata of house value related to Alameda County, and then calculate the median
```

```
b. Give a single line of R which gives the same final answer as the block of code.  Note: there are at
```

```r
median((na.omit(ca_pa)%>%filter(COUNTYFP==1,STATEFP==6))$Median_house_value)
```

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

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

```r
#Alameda
ca_pa.ala1<-filter(ca_pa.omit,COUNTYFP==1,STATEFP==6)$Built_2005_or_later
ca_pa.ala2<-filter(ca_pa.omit,COUNTYFP==1,STATEFP==6)$Total_units
ca_pa.ala<-ca_pa.ala1/ca_pa.ala2
sum(ca_pa.ala1/ca_pa.ala2)*100
```

```
## [1] 88.34551
```

```r
#Santa Clara
ca_pa.sc1<-filter(ca_pa.omit,COUNTYFP==85,STATEFP==6)$Built_2005_or_later
ca_pa.sc2<-filter(ca_pa.omit,COUNTYFP==85,STATEFP==6)$Total_units
sum(ca_pa.sc1/ca_pa.sc2)*100
```

```
## [1] 60.84691
```

```r
#Allegheny Counties
ca_pa.all1<-filter(ca_pa.omit,COUNTYFP==3,STATEFP==42)$Built_2005_or_later
ca_pa.all2<-filter(ca_pa.omit,COUNTYFP==3,STATEFP==42)$Total_units
ca_pa.all<-ca_pa.all1/ca_pa.all2
sum(ca_pa.all1/ca_pa.all2)*100
```

```
## [1] 45.54192
```

d. The ‘cor‘ function calculates the correlation coefficient between two variables. What is the correla

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

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

```r
##Alameda
ca_pa.ala1<-filter(ca_pa.omit,COUNTYFP==1,STATEFP==6)$Built_2005_or_later
ca_pa.ala2<-filter(ca_pa.omit,COUNTYFP==1,STATEFP==6)$Median_house_value
cor(ca_pa.ala1,ca_pa.ala2)
```

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

```r
#Santa Clara
ca_pa.sc1<-filter(ca_pa.omit,COUNTYFP==85,STATEFP==6)$Built_2005_or_later
ca_pa.sc2<-filter(ca_pa.omit,COUNTYFP==85,STATEFP==6)$Median_house_value
cor(ca_pa.sc1,ca_pa.sc2)
```

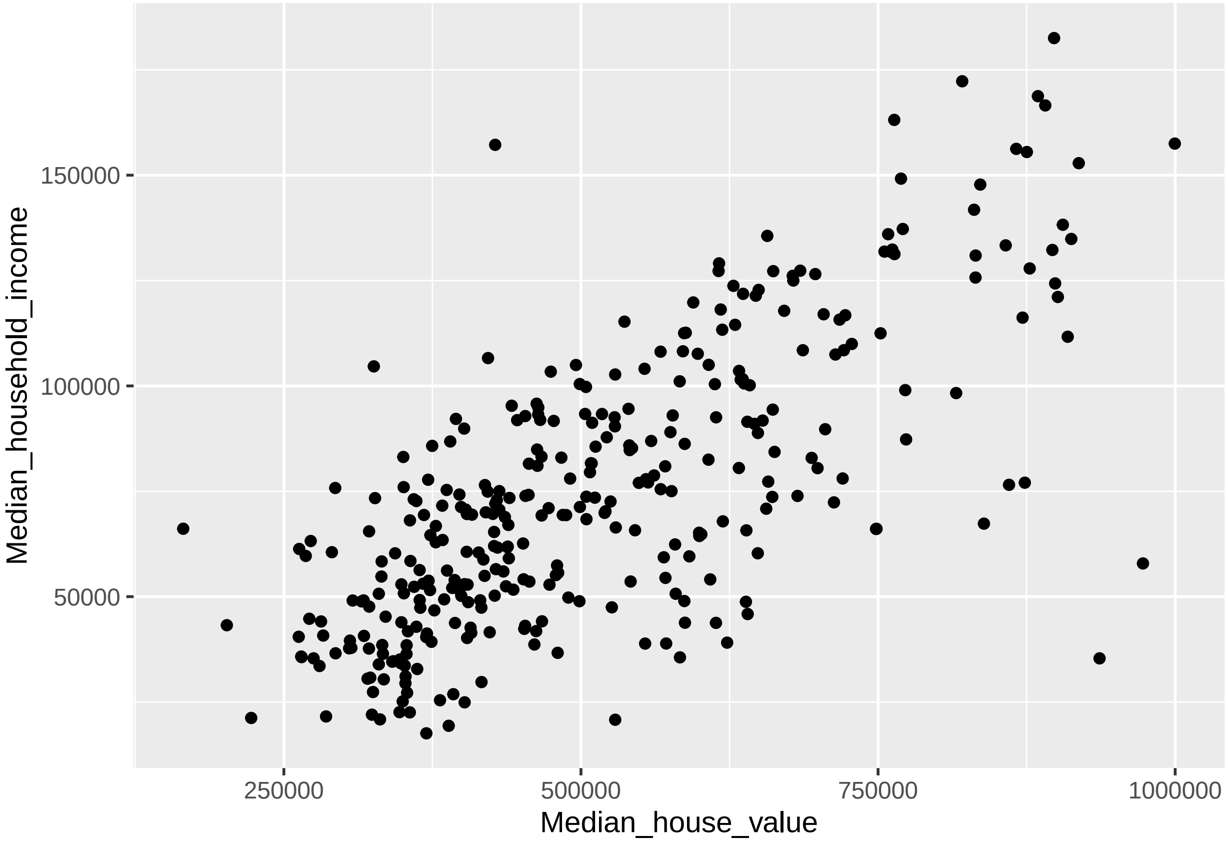
```
## [1] -0.1726203
```

```
#Allegheny Counties
ca_pa.all1<-filter(ca_pa.omit,COUNTYFP==3,STATEFP==42)$Built_2005_or_later
ca_pa.all2<-filter(ca_pa.omit,COUNTYFP==3,STATEFP==42)$Median_house_value
cor(ca_pa.all1,ca_pa.all2)
```

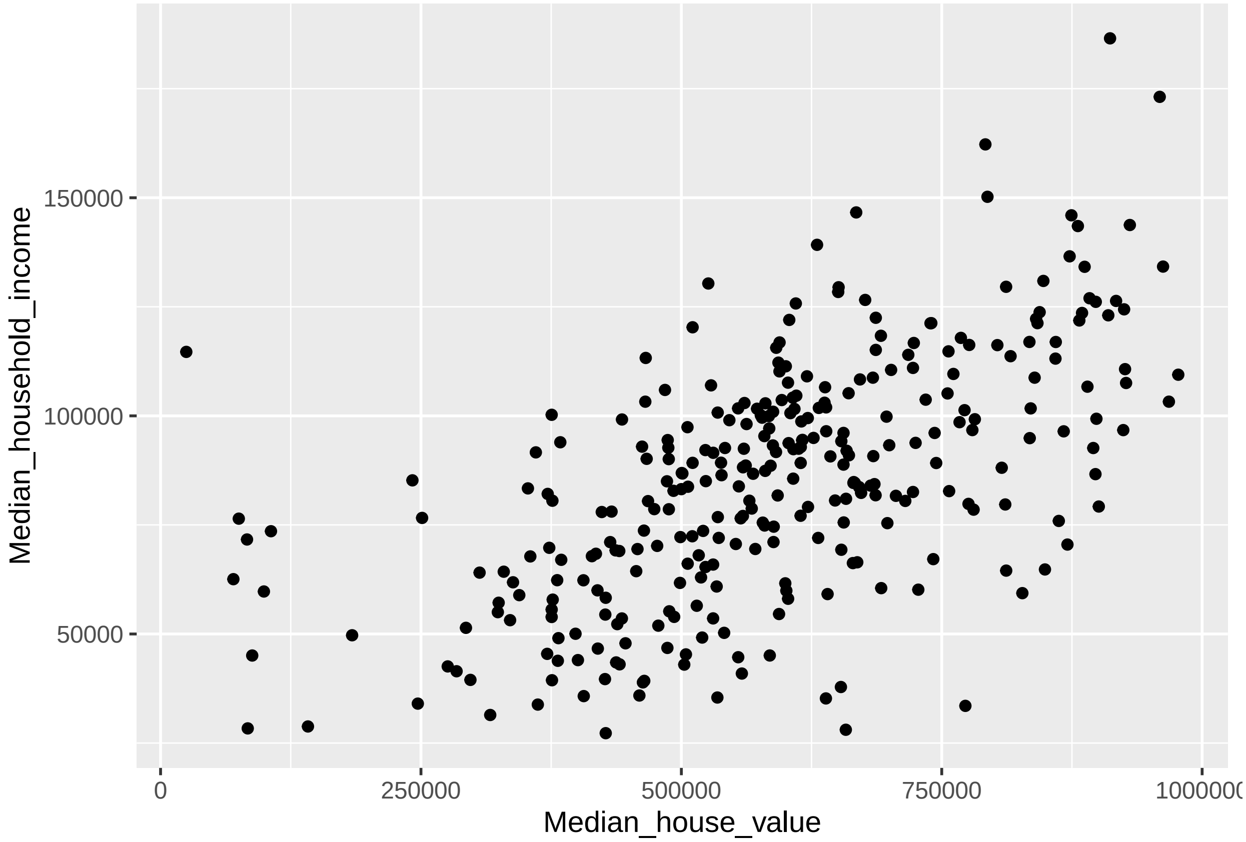
```
## [1] 0.1939652
```

```
e. Make three plots, showing median house values against median income, for Alameda, Santa Clara, and Al
```

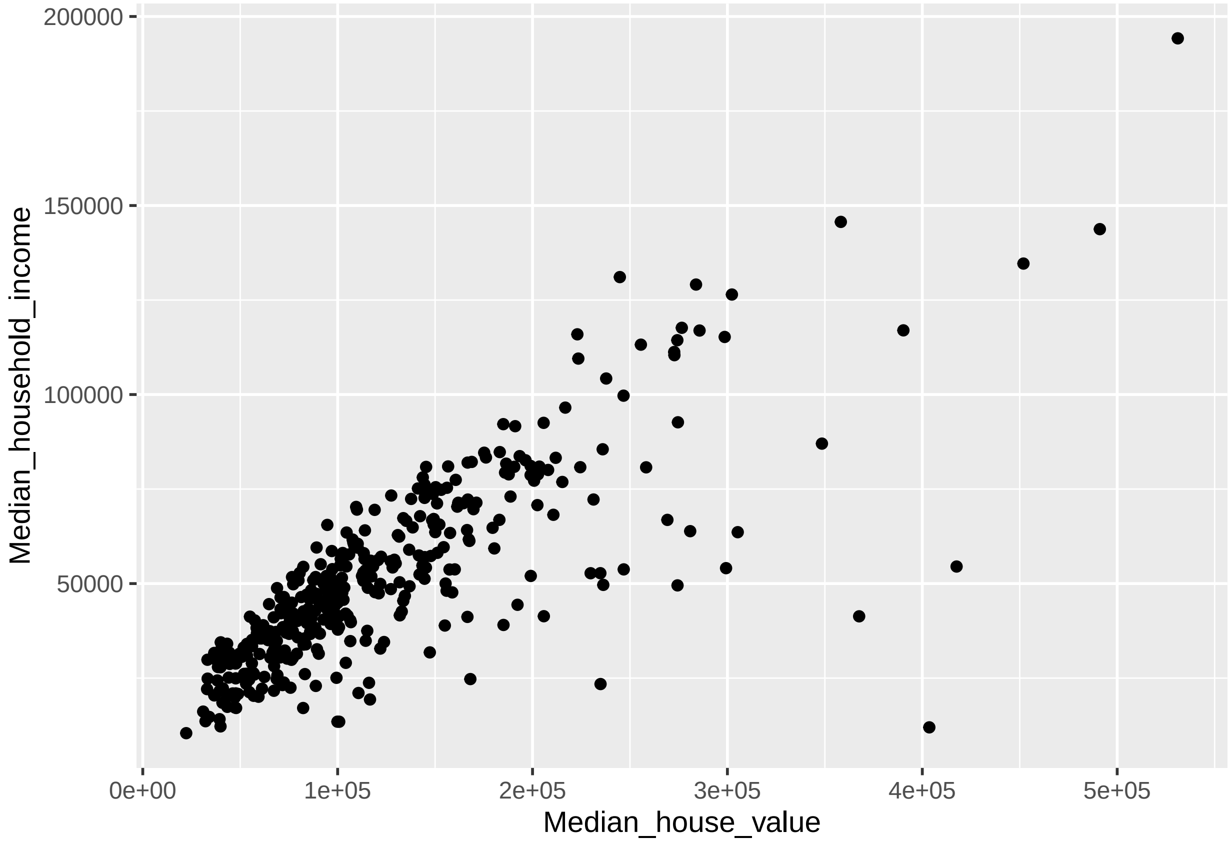
```
ggplot(data = (ca_pa.omit %>%
filter(COUNTYFP==1,STATEFP==6) %>%      select(COUNTYFP,Median_house_value,Median_household_income))) +
  geom_point(aes(x = Median_house_value, y = Median_household_income))
```



```
ggplot(data = na.omit(ca_pa %>% filter(COUNTYFP==85,STATEFP==6) %>%      select(Median_house_value,Mediar
  geom_point(aes(x = Median_house_value, y = Median_household_income))
```



```
ggplot(data = na.omit(ca_pa %>% filter(COUNTYFP==3,STATEFP==42) %>%      select(Median_house_value,Median
  geom_point(aes(x = Median_house_value, y = Median_household_income))
```



MB.Ch1.11. Run the following code:

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

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

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

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

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

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

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

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

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

Explain the output from the successive uses of table(). #for t“he first time, it created”gender" with 92 “female” and 92 “male”, and then in the second time, with the use of level(), it changed the location of two columns, after that, in the third time, it replace “male” with “Male”. “Male“ has not been assigned, so it’s the defaulting number 0, however, the number of male did not eleminate, so when include”null“, wen can still see a number”92".

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

. . . .

```
Rabbit.1<-unstack(Rabbit,BPchange~Animal)
Dose<-unstack(Rabbit, Dose~Animal)[,1]
Treatment<-unstack(Rabbit, Treatment~Animal)[,1]
Rabbit.new<-data.frame(Treatment,Dose,Rabbit.1)
```