hw2

张尉航

2024-07-02

1.

## [1] 1

ca\_pa<-read.csv("data/calif\_penn\_2011.csv")  
dim(ca\_pa)

## [1] 11275 34

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

print("先生成一个与原数据框相同的矩阵，表明对应位置的值然后累积总和，也就是计算ca\_pa中每一列NA的值得数量，并且返回向量")

## [1] "先生成一个与原数据框相同的矩阵，表明对应位置的值然后累积总和，也就是计算ca\_pa中每一列NA的值得数量，并且返回向量"

ca\_pa\_clean<-na.omit(ca\_pa)  
r\_eliminated <- nrow(ca\_pa) - nrow(ca\_pa\_clean)  
r\_eliminated

## [1] 670

print("一个计算每列的NA数量，另一个通过计算删除的行数，如果NA大于0且与删除的行数匹配，则一致")

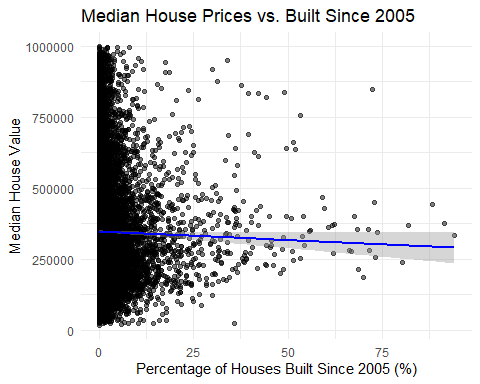
## [1] "一个计算每列的NA数量，另一个通过计算删除的行数，如果NA大于0且与删除的行数匹配，则一致"

2.

## [1] 2

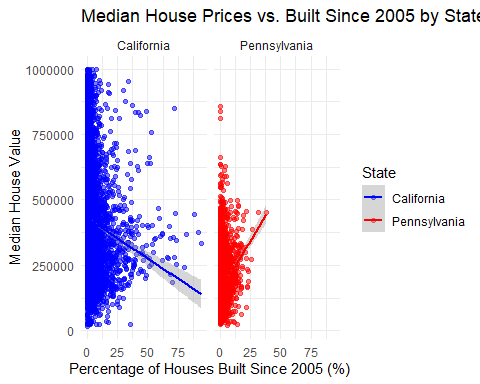
library(ggplot2)  
ggplot(ca\_pa\_clean, aes(x = Built\_2005\_or\_later, y = Median\_house\_value)) +   
 geom\_point(alpha = 0.5) +   
 geom\_smooth(method = "lm", color = "blue") +   
 labs(title = "Median House Prices vs. Built Since 2005",   
 x = "Percentage of Houses Built Since 2005 (%)",   
 y = "Median House Value") +  
 theme\_minimal()

## `geom\_smooth()` using formula = 'y ~ x'



ggplot(ca\_pa\_clean, aes(x = Built\_2005\_or\_later, y = Median\_house\_value, color = as.factor(STATEFP))) +   
 geom\_point(alpha = 0.5) +   
 geom\_smooth(method = "lm") +   
 labs(title = "Median House Prices vs. Built Since 2005 by State",   
 x = "Percentage of Houses Built Since 2005 (%)",   
 y = "Median House Value",   
 color = "State") +  
 facet\_wrap(~ STATEFP, labeller = as\_labeller(c(`6` = "California", `42` = "Pennsylvania"))) +  
 scale\_color\_manual(values = c("6" = "blue", "42" = "red"),   
 labels = c("California", "Pennsylvania")) +  
 theme\_minimal()

## `geom\_smooth()` using formula = 'y ~ x'



3.

## [1] 3

ca\_pa\_clean<-na.omit(ca\_pa)  
nrow(ca\_pa\_clean)

## [1] 10605

ca\_pa$Vacancy\_Rate<-ca\_pa$Vacant\_units/ca\_pa$Total\_units  
summary(ca\_pa$Vacancy\_Rate)

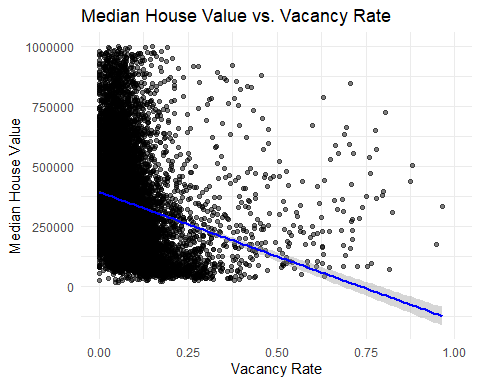
## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 0.00000 0.03808 0.06766 0.08918 0.10992 1.00000 98

ggplot(ca\_pa,aes(x=Vacancy\_Rate,y=Median\_house\_value))+  
 geom\_point(alpha = 0.5) +   
 geom\_smooth(method = "lm", color = "blue") +   
 labs(title = "Median House Value vs. Vacancy Rate",   
 x = "Vacancy Rate",   
 y = "Median House Value") +  
 theme\_minimal()

## `geom\_smooth()` using formula = 'y ~ x'

## Warning: Removed 599 rows containing non-finite outside the scale range  
## (`stat\_smooth()`).

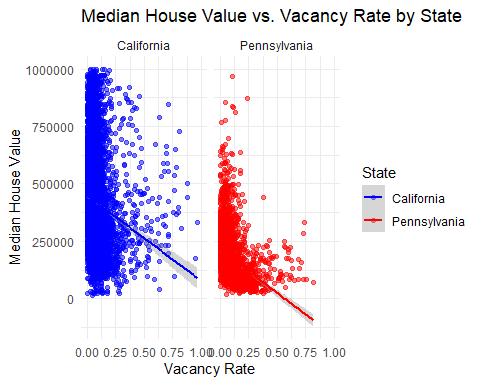
## Warning: Removed 599 rows containing missing values or values outside the scale range  
## (`geom\_point()`).



ggplot(ca\_pa, aes(x = Vacancy\_Rate, y = Median\_house\_value, color = as.factor(STATEFP))) +   
 geom\_point(alpha = 0.5) +   
 geom\_smooth(method = "lm") +   
 labs(title = "Median House Value vs. Vacancy Rate by State",   
 x = "Vacancy Rate",   
 y = "Median House Value",   
 color = "State") +  
 facet\_wrap(~ STATEFP, labeller = as\_labeller(c(`6` = "California", `42` = "Pennsylvania"))) +  
 scale\_color\_manual(values = c("blue", "red"),   
 labels = c("California", "Pennsylvania")) +  
 theme\_minimal()

## `geom\_smooth()` using formula = 'y ~ x'

## Warning: Removed 599 rows containing non-finite outside the scale range  
## (`stat\_smooth()`).  
## Removed 599 rows containing missing values or values outside the scale range  
## (`geom\_point()`).



4.

## [1] 4

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

print("这段代码的作用是找出加利福尼亚州的accmhv，然后计算普查区中房屋建造比例的中位数，首先创造一个空向量acca，然后用for循环遍历ca\_pa的每一行，检查是否属于，如果属于，就添加，第二个for循环普查每个普查区，最后，计算accmhv中的中位数")

## [1] "这段代码的作用是找出加利福尼亚州的accmhv，然后计算普查区中房屋建造比例的中位数，首先创造一个空向量acca，然后用for循环遍历ca\_pa的每一行，检查是否属于，如果属于，就添加，第二个for循环普查每个普查区，最后，计算accmhv中的中位数"

median(ca\_pa$Built\_2005\_or\_later[ca\_pa$STATEFP == 6 & ca\_pa$COUNTYFP == 1])

## [1] NA

mean(ca\_pa$Built\_2005\_or\_later[ca\_pa$STATEFP == 6 & ca\_pa$COUNTYFP == 1])

## [1] NA

mean(ca\_pa$Built\_2005\_or\_later[ca\_pa$STATEFP == 6 & ca\_pa$COUNTYFP == 85])

## [1] 3.160215

mean(ca\_pa$Built\_2005\_or\_later[ca\_pa$STATEFP == 42 & ca\_pa$COUNTYFP == 3])

## [1] NA

5.

## [1] 5

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"))  
table(gender)

## gender  
## Male female   
## 0 91

table(gender, exclude=NULL)

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

print("第一步中的gender是一个因子向量，使用table计算各因子水平的频数，为91个female和92个male，第二步通过levels参数重新排序了gender因子的水平顺序，第三步将male写成了Male，所以Male未输出，最后一步用exclude=NULL，显示了所有可能的因子水平")

## [1] "第一步中的gender是一个因子向量，使用table计算各因子水平的频数，为91个female和92个male，第二步通过levels参数重新排序了gender因子的水平顺序，第三步将male写成了Male，所以Male未输出，最后一步用exclude=NULL，显示了所有可能的因子水平"

6.

## [1] 6

p\_r<-function(x,cutoff){  
 mean(x>cutoff)  
}  
x<-1:100  
cutoff<-50  
p\_r(x,cutoff)

## [1] 0.5

library(Devore7)

## 载入需要的程序包：MASS

## 载入需要的程序包：lattice

data(ex01.36)  
head(ex01.36)

## C1  
## 1 389  
## 2 356  
## 3 359  
## 4 363  
## 5 375  
## 6 424

summary(ex01.36)

## C1   
## Min. :325.0   
## 1st Qu.:359.0   
## Median :369.5   
## Mean :370.7   
## 3rd Qu.:391.2   
## Max. :424.0

library(lattice)  
p\_e\_7 <- mean(ex01.36 > 7)  
p\_e\_7

## [1] 1

7.

## [1] 7

library(MASS)  
data(Rabbit)  
str(Rabbit)

## 'data.frame': 60 obs. of 5 variables:  
## $ BPchange : num 0.5 4.5 10 26 37 32 1 1.25 4 12 ...  
## $ Dose : num 6.25 12.5 25 50 100 200 6.25 12.5 25 50 ...  
## $ Run : Factor w/ 10 levels "C1","C2","C3",..: 1 1 1 1 1 1 2 2 2 2 ...  
## $ Treatment: Factor w/ 2 levels "Control","MDL": 1 1 1 1 1 1 1 1 1 1 ...  
## $ Animal : Factor w/ 5 levels "R1","R2","R3",..: 1 1 1 1 1 1 2 2 2 2 ...

Rabbit\_unstacked <- unstack(Rabbit, BPchange ~ Treatment + Dose + Rabbit)

## Warning in Ops.factor(Treatment, Dose): '+' 对因子没有意义

## Warning in Ops.factor(left, right): '+' 对因子没有意义  
## Warning in Ops.factor(left, right): '+' 对因子没有意义  
## Warning in Ops.factor(left, right): '+' 对因子没有意义