# HW5_ZHENGZHI_LIN

*zhengzhi lin*

*2019.10.7*

```
#p2
#a
pro_suc <- function(x,m){
  p <- apply(x, m, sum)/apply(x,m,length)
  return(p)
}
#b
set.seed(12345)
P4b_data <- matrix(rbinom(10, 1, prob = (30:40)/100), nrow = 10, ncol = 10, byrow = FALSE)
#c
pro_suc(P4b_data,m=1)
```

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

```
pro_suc(P4b_data,m=2)
```

```
##  [1] 0.6 0.6 0.6 0.6 0.6 0.6 0.6 0.6 0.6 0.6
```

```
#row is 1 and coloum is .6

#d

P4b_data <- matrix(0, nrow = 10, ncol = 10, byrow = FALSE)
for (i in 1:10) {
  P4b_data[i,] <- rbinom(10, 1, prob = .5)
}
pro_suc(P4b_data,m=1)
```

```
##  [1] 0.2 0.5 0.5 0.4 0.4 0.7 0.6 0.5 0.7 0.7
```

```
pro_suc(P4b_data,m=2)
```

```
##  [1] 0.7 0.5 0.5 0.6 0.5 0.3 0.8 0.6 0.4 0.3
```

```
#p4
#1
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag
```

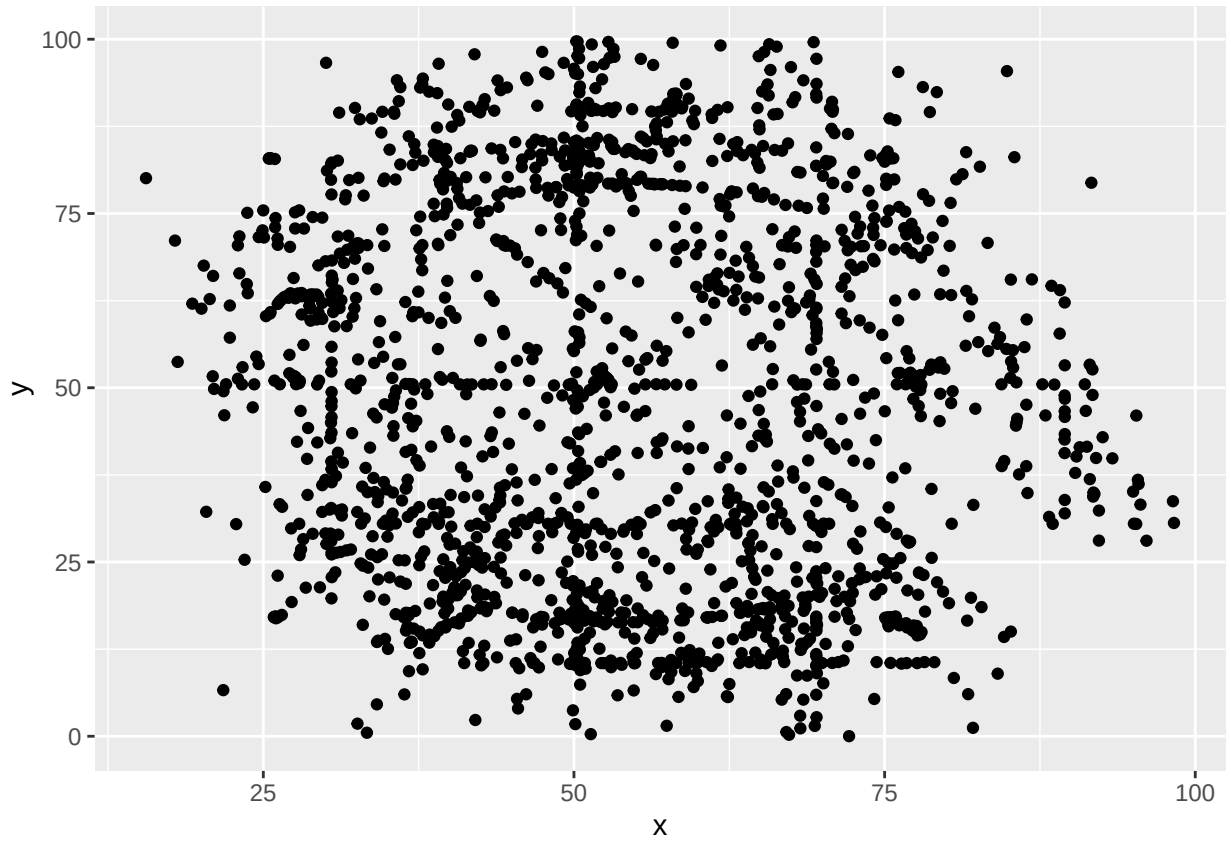
```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

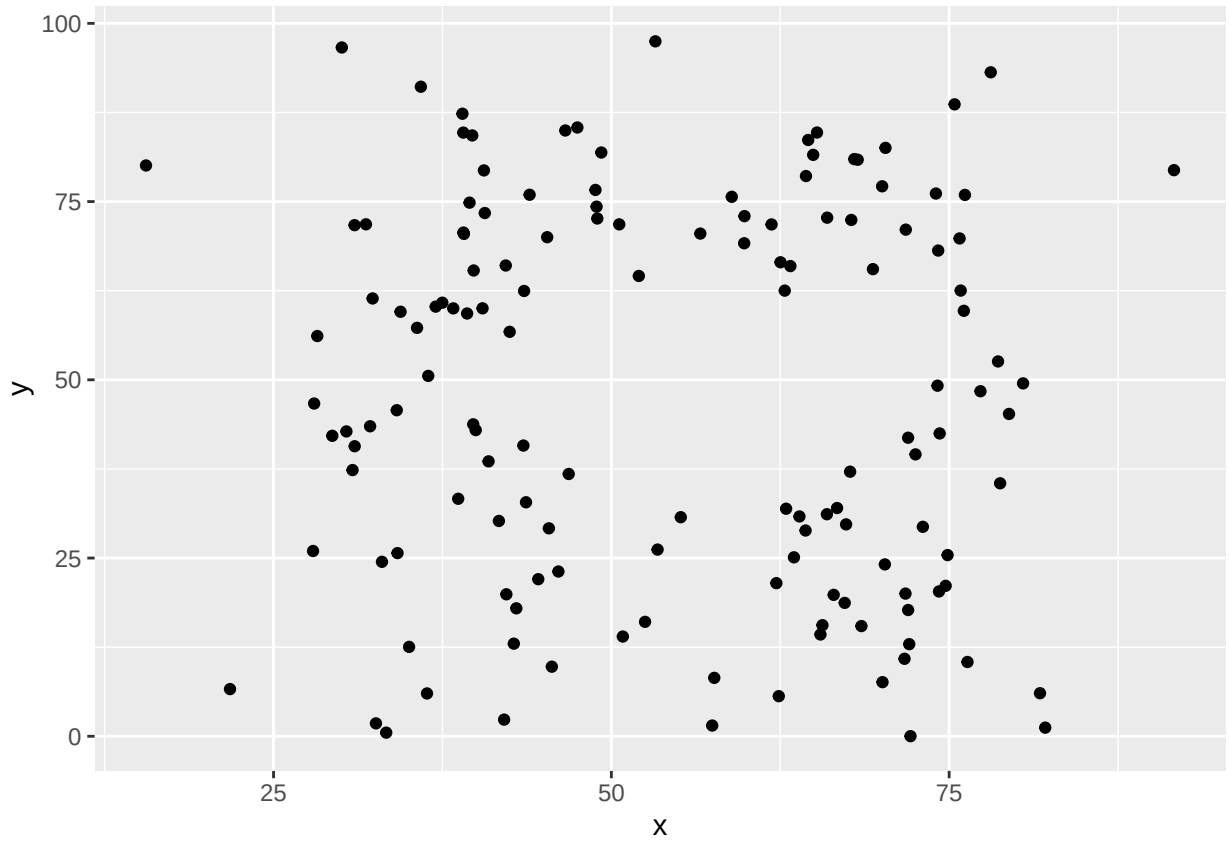
```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
dat <- readRDS("HW4_data.rds", refhook = NULL)
dat <- dat %>% rename(x = dev1, y = dev2)


f_plot <- function(i){
  if(i==0)
    return(ggplot(dat, aes(x=x,y=y)) +
    geom_point())
  else
    ggplot(dat[which(dat$Observer==i),], aes(x=x,y=y)) +
      geom_point()
}

f_plot(0)
```

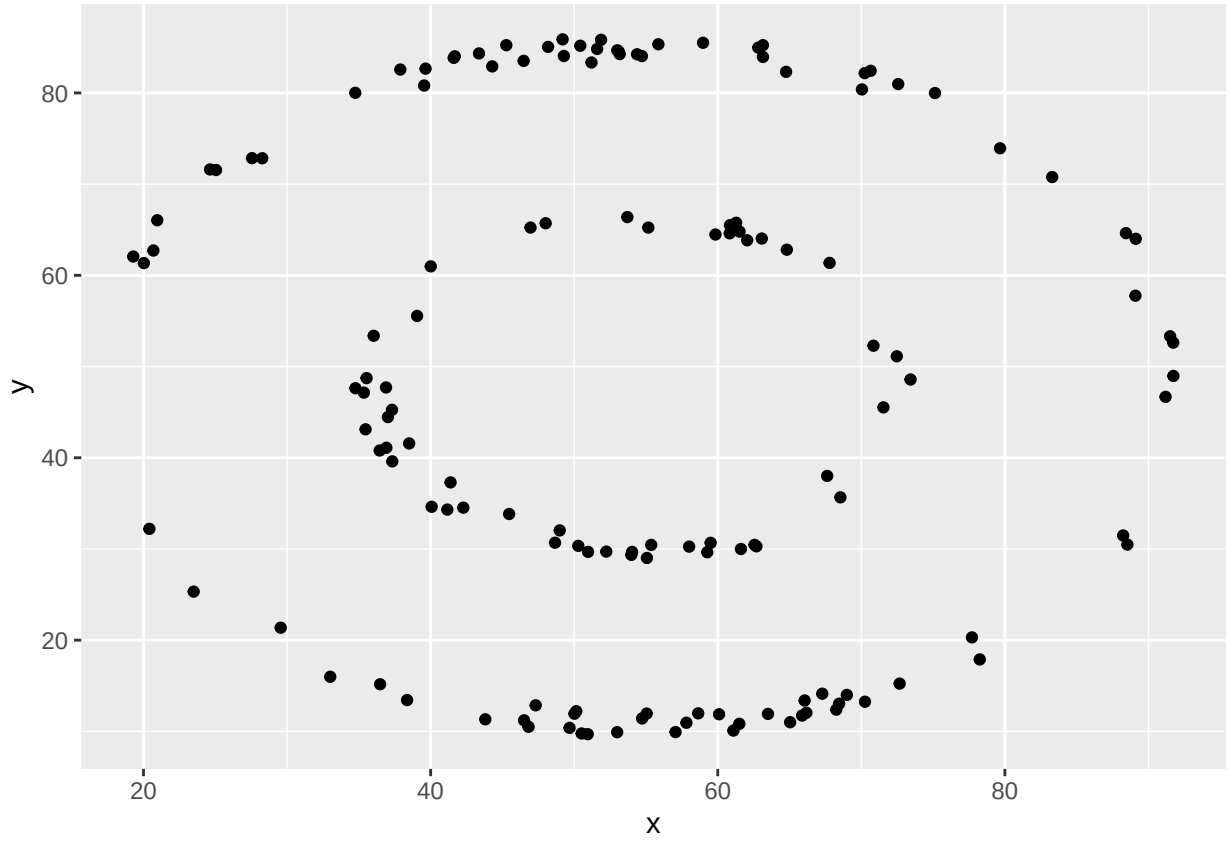



```
lapply(1:13, f_plot)
```

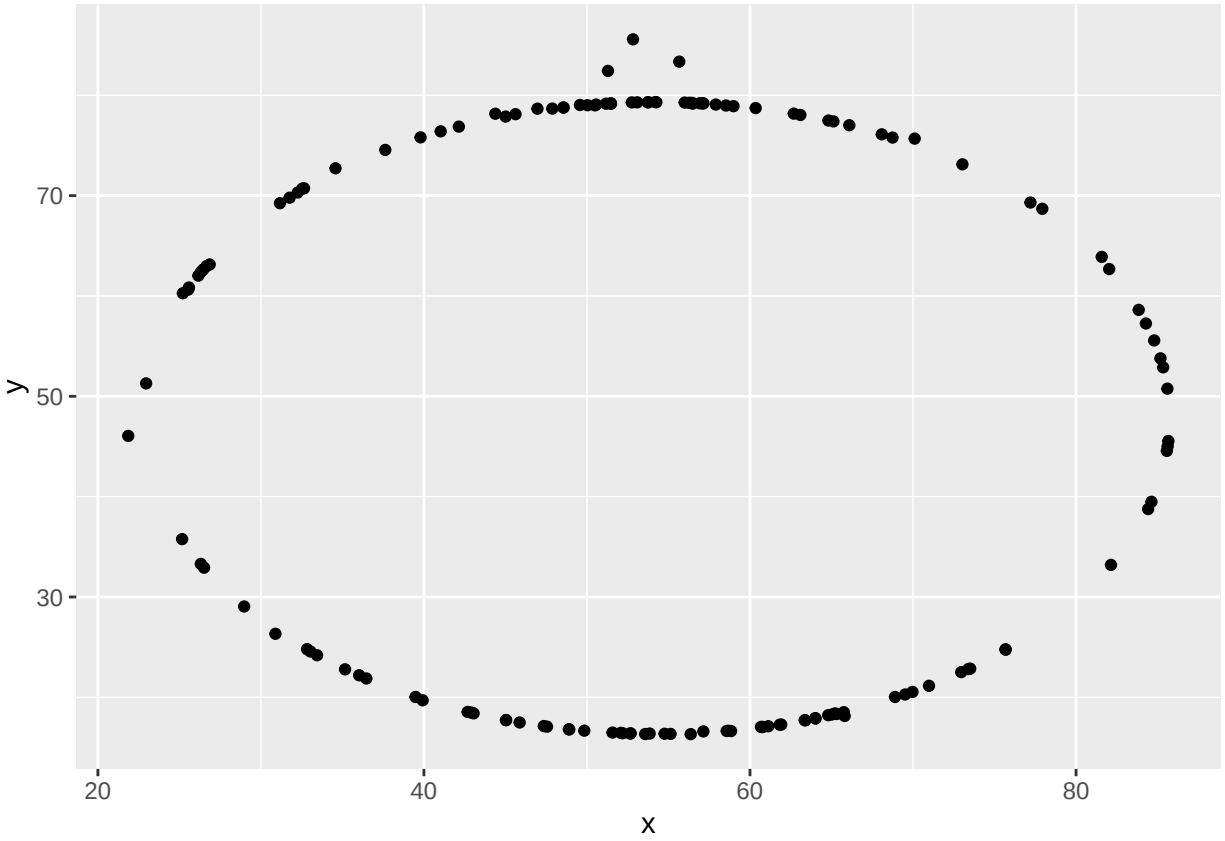
```
## [[1]]
```



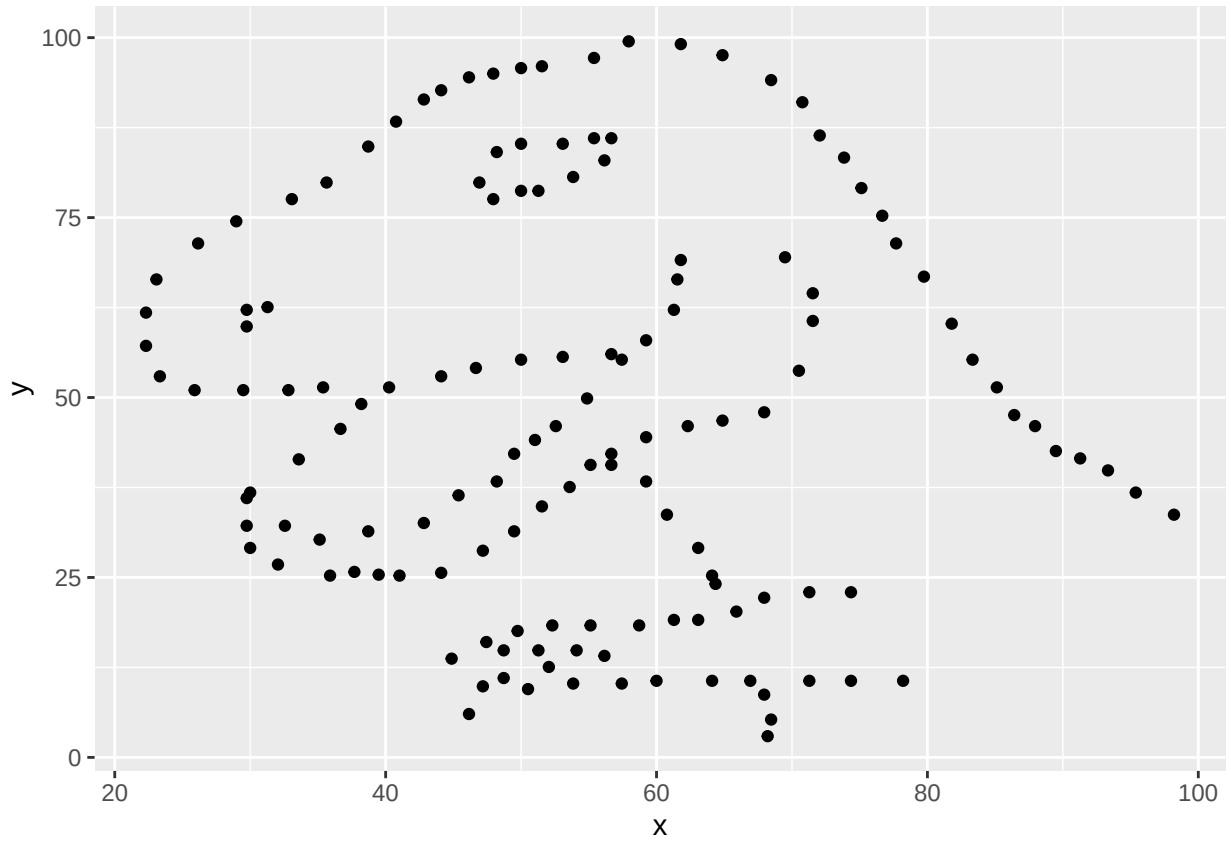

```
##
## [[2]]
```



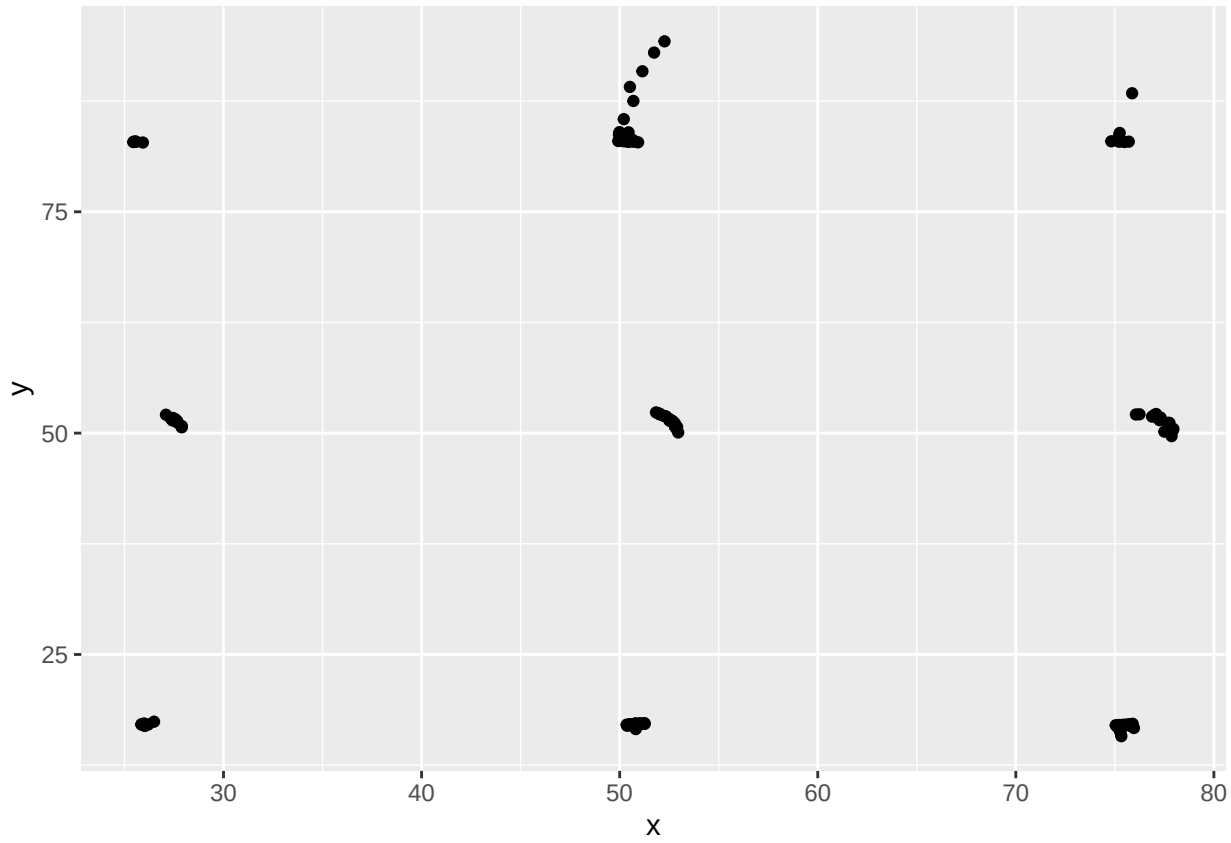

```
##
## [[3]]
```



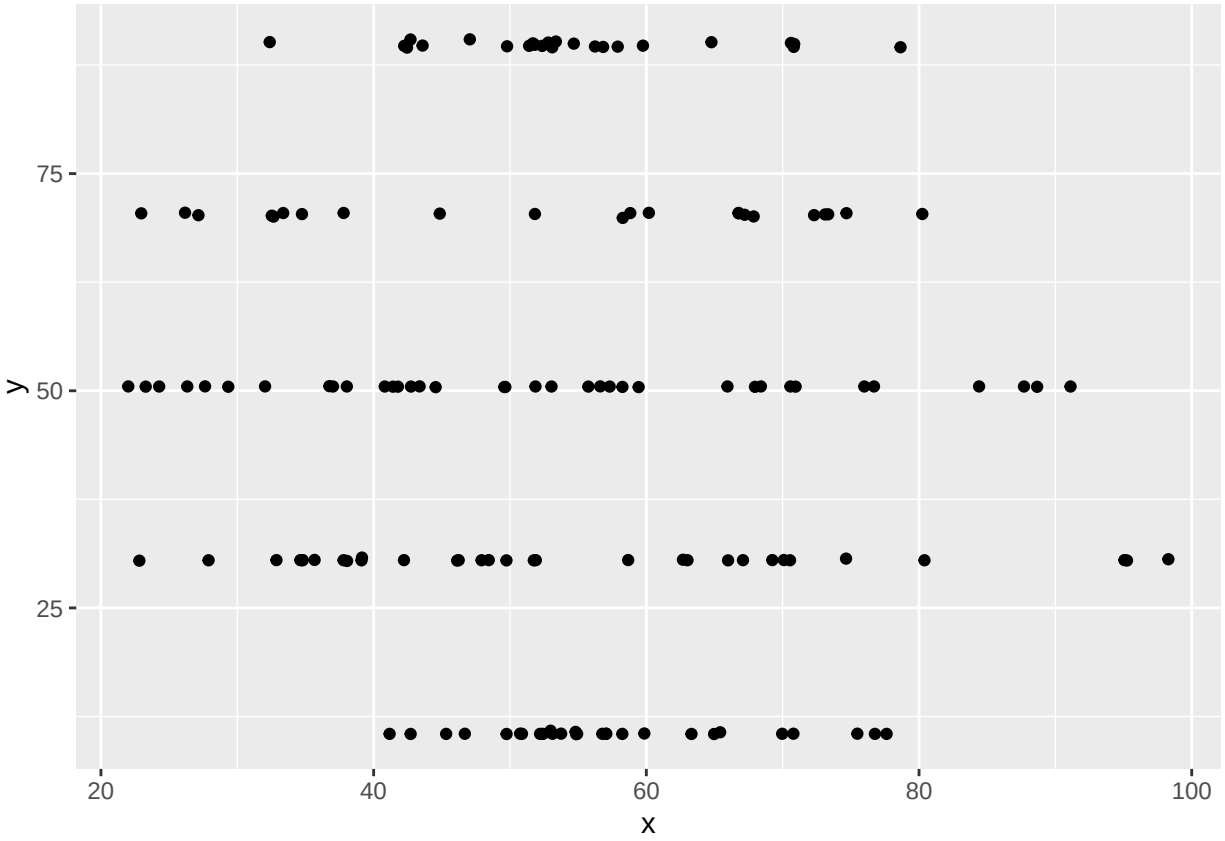

```
##
## [[4]]
```



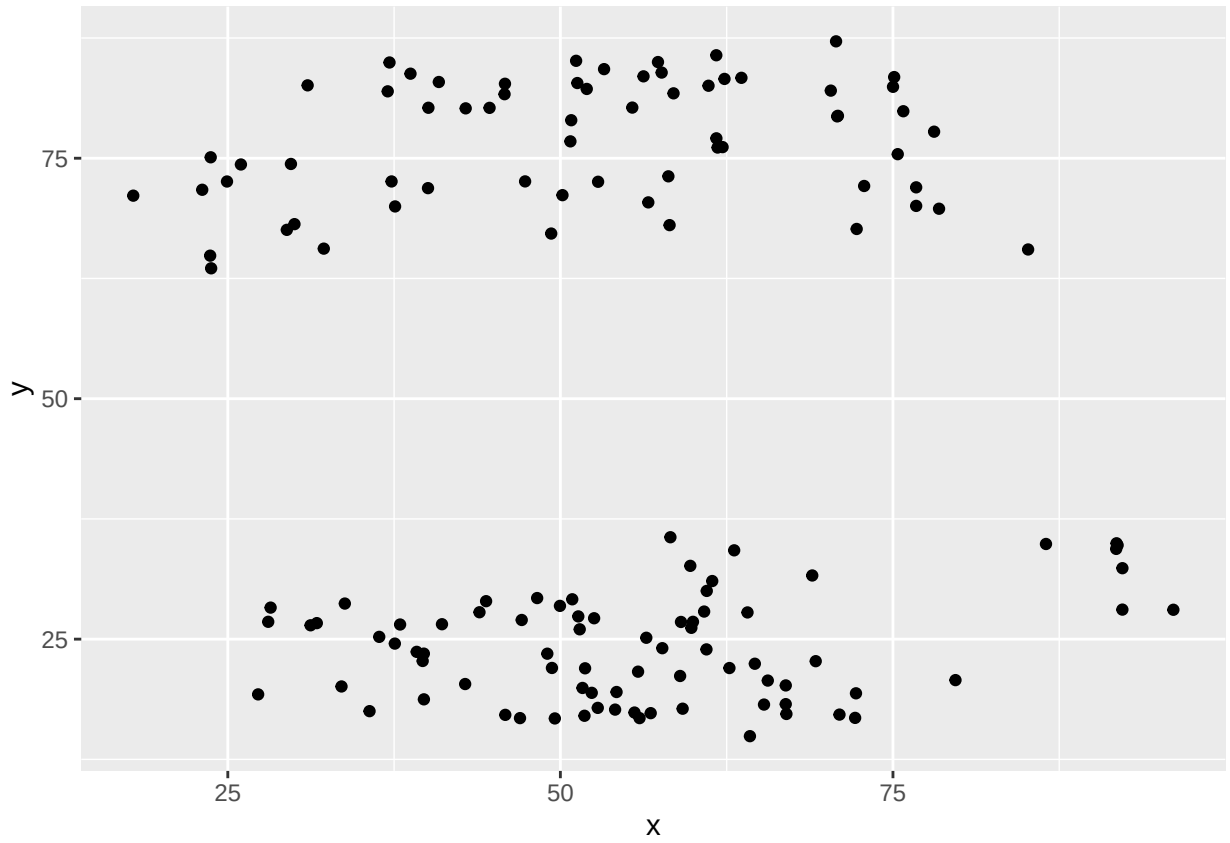

```
##
## [[5]]
```



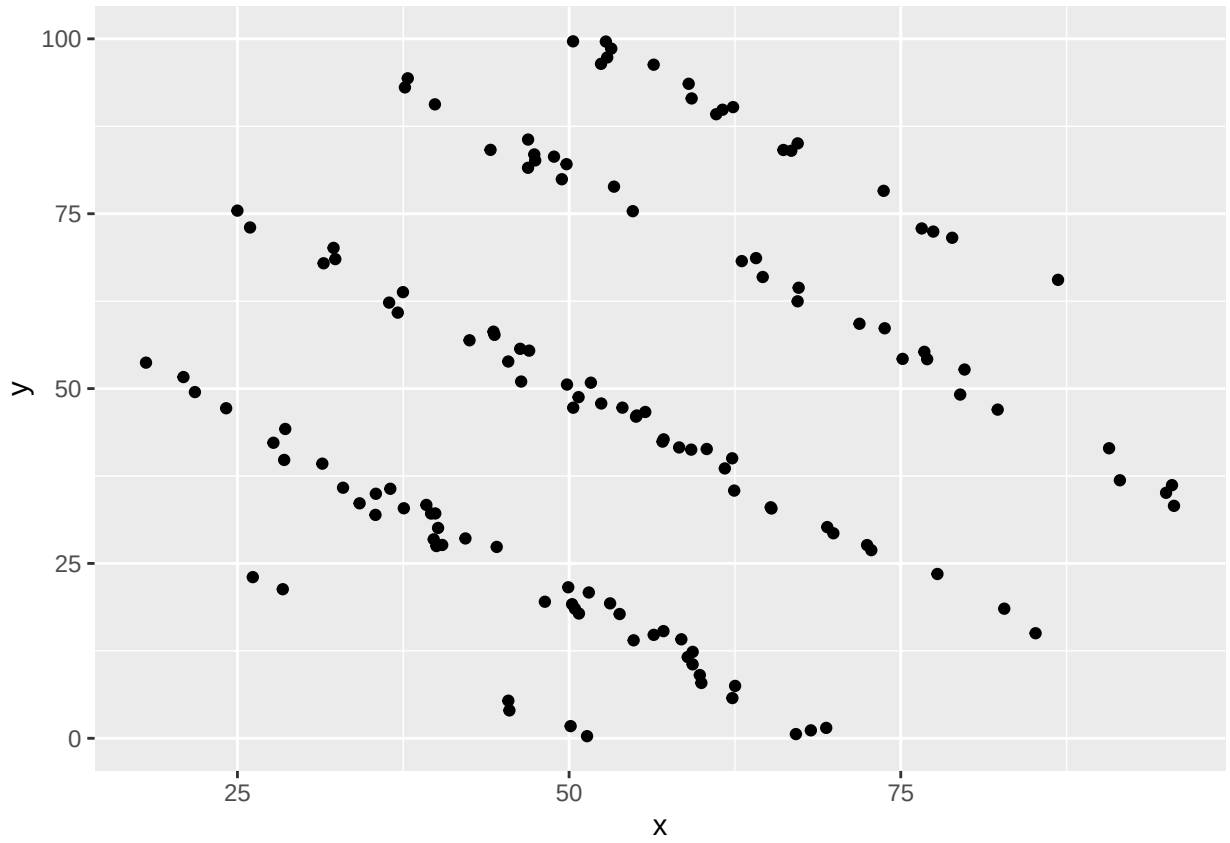

```
##
## [[6]]
```



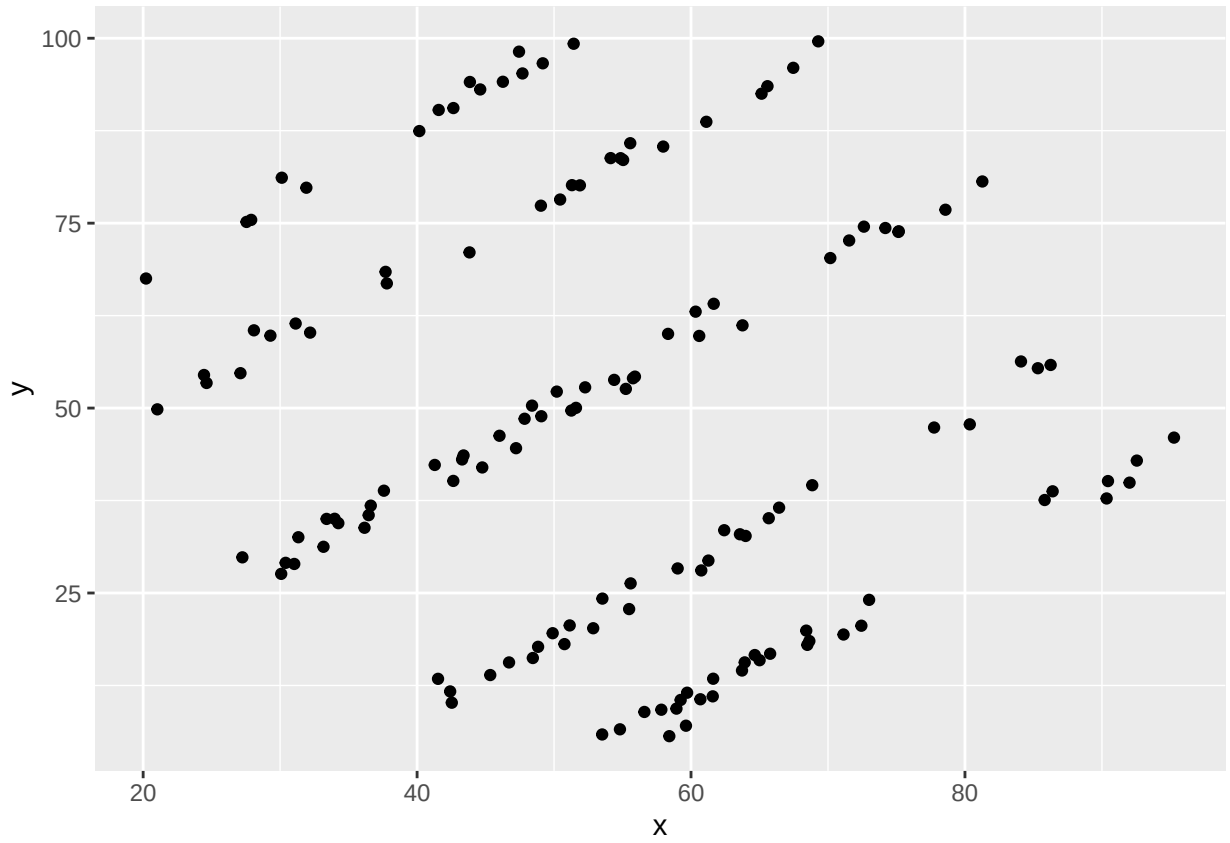

```
##
## [[7]]
```



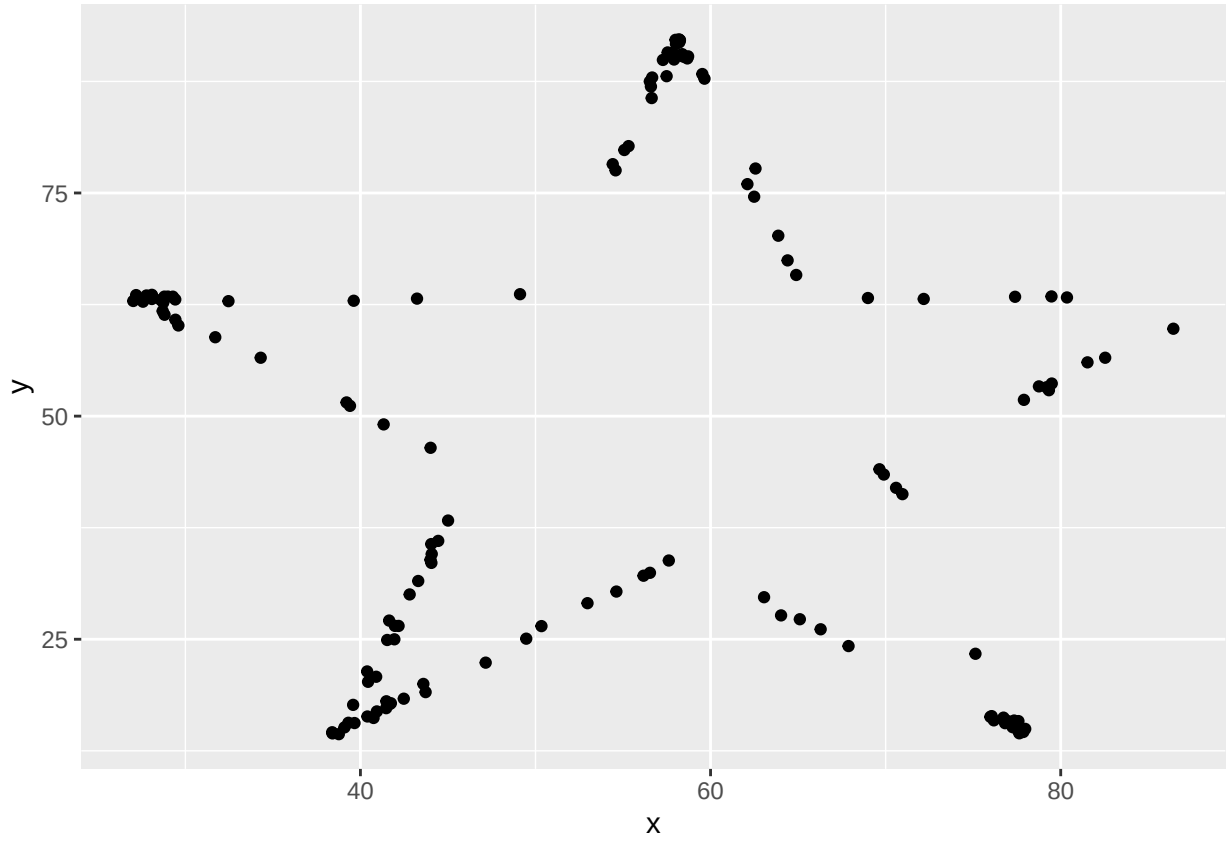

```
##
## [[8]]
```



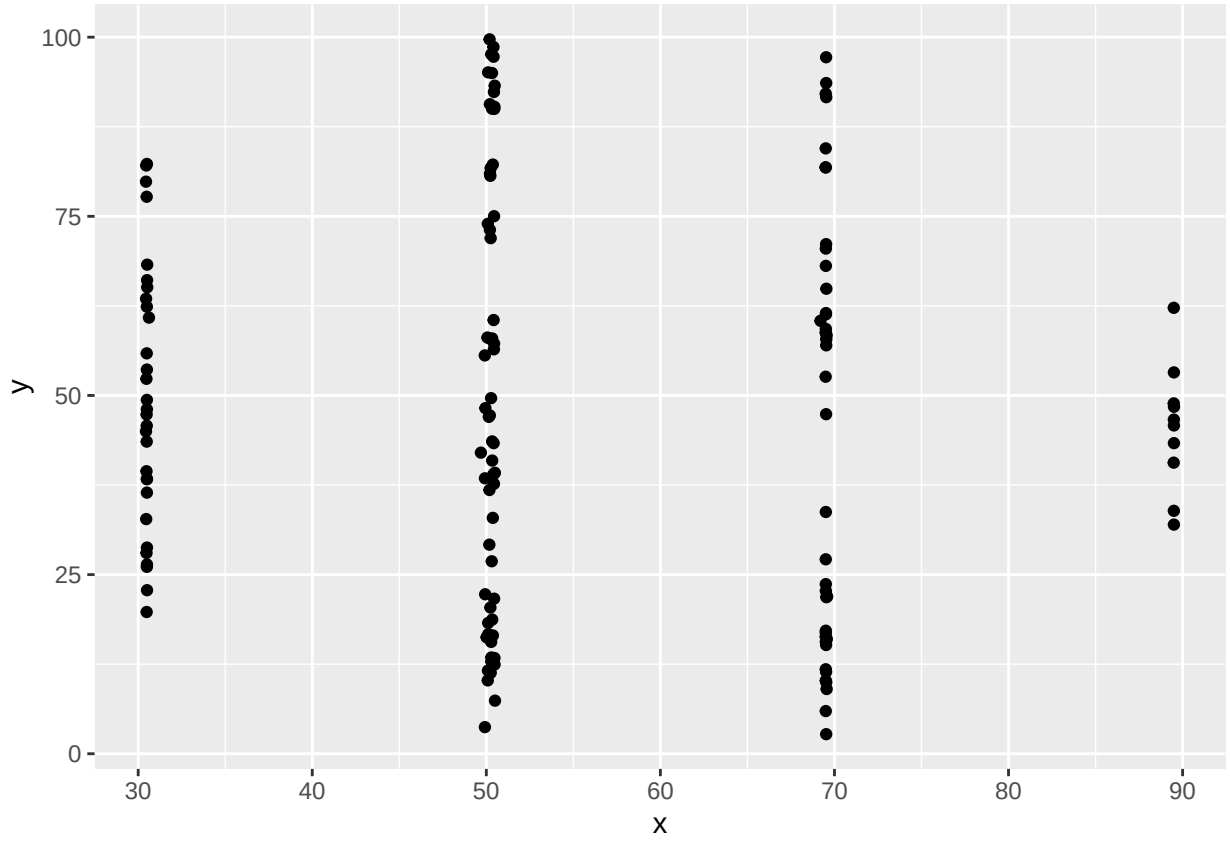

```
##
## [[9]]
```



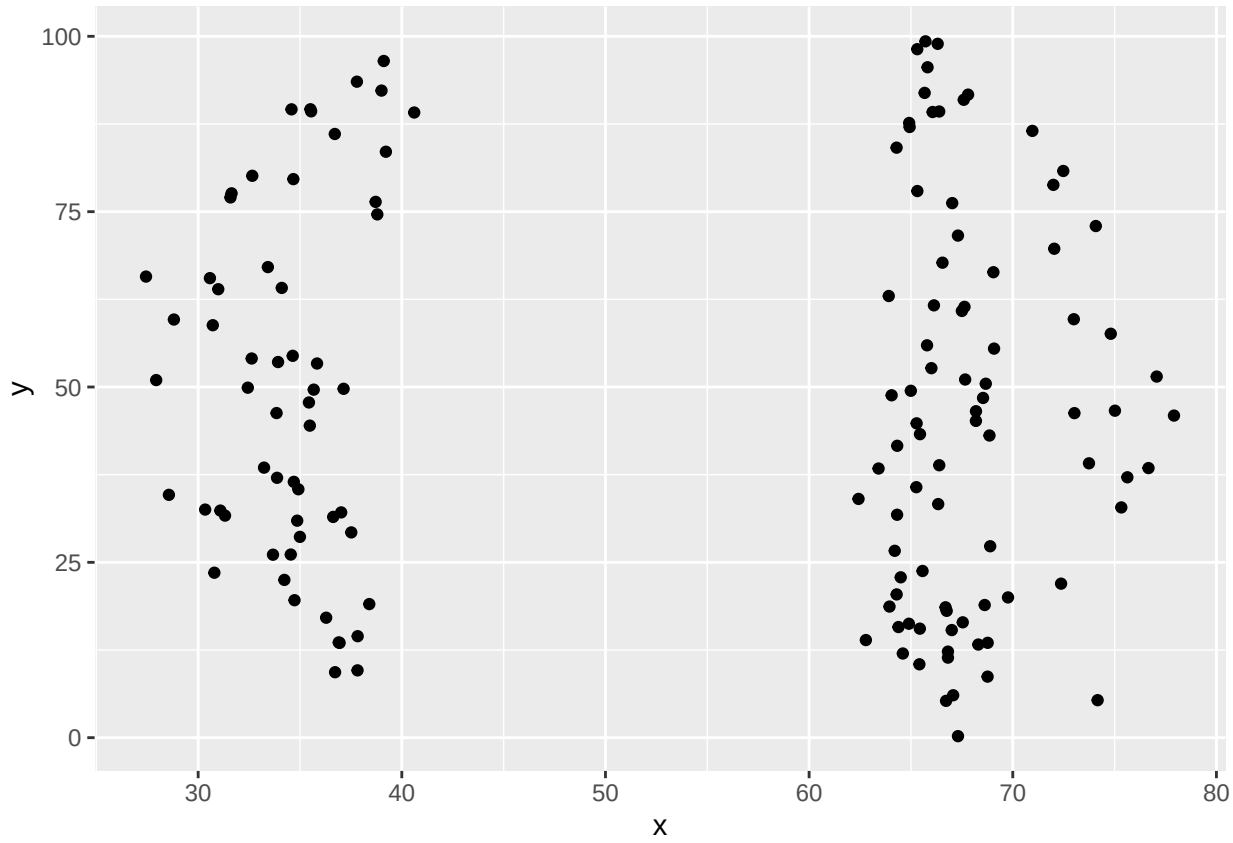

```
##
## [[10]]
```



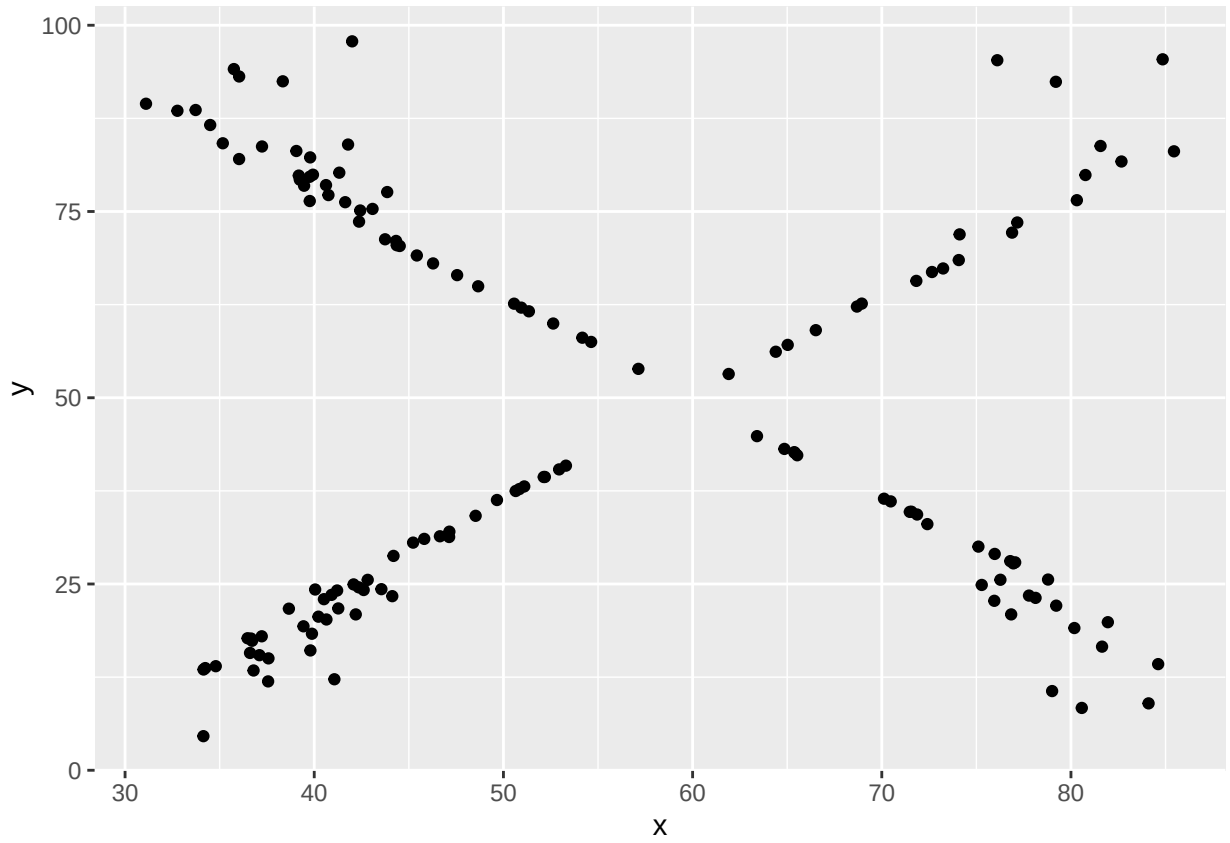

```
##
## [[11]]
```




```
##
## [[12]]
```




```
##
## [[13]]
```




## 5

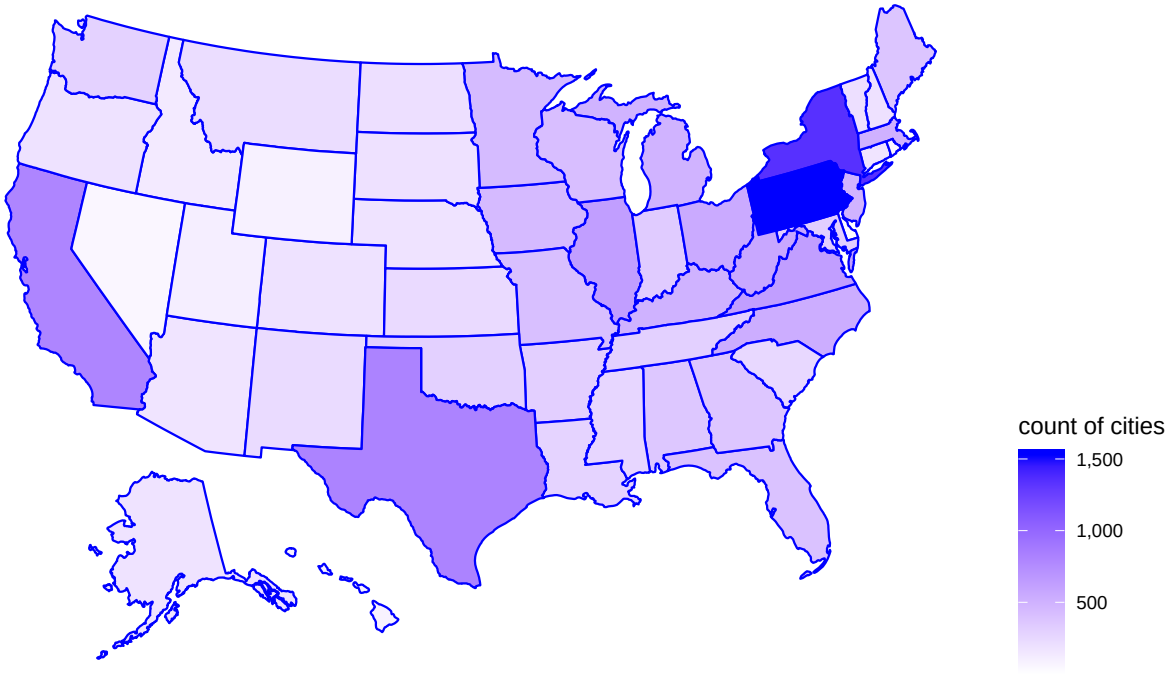
```
#a
library(dplyr)
library(downloader)
download("http://www.farinspace.com/wp-content/uploads/us_cities_and_states.zip","us_cities_states.zip"
unzip("us_cities_states.zip", exdir=".")
library(data.table)
```

```
##
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

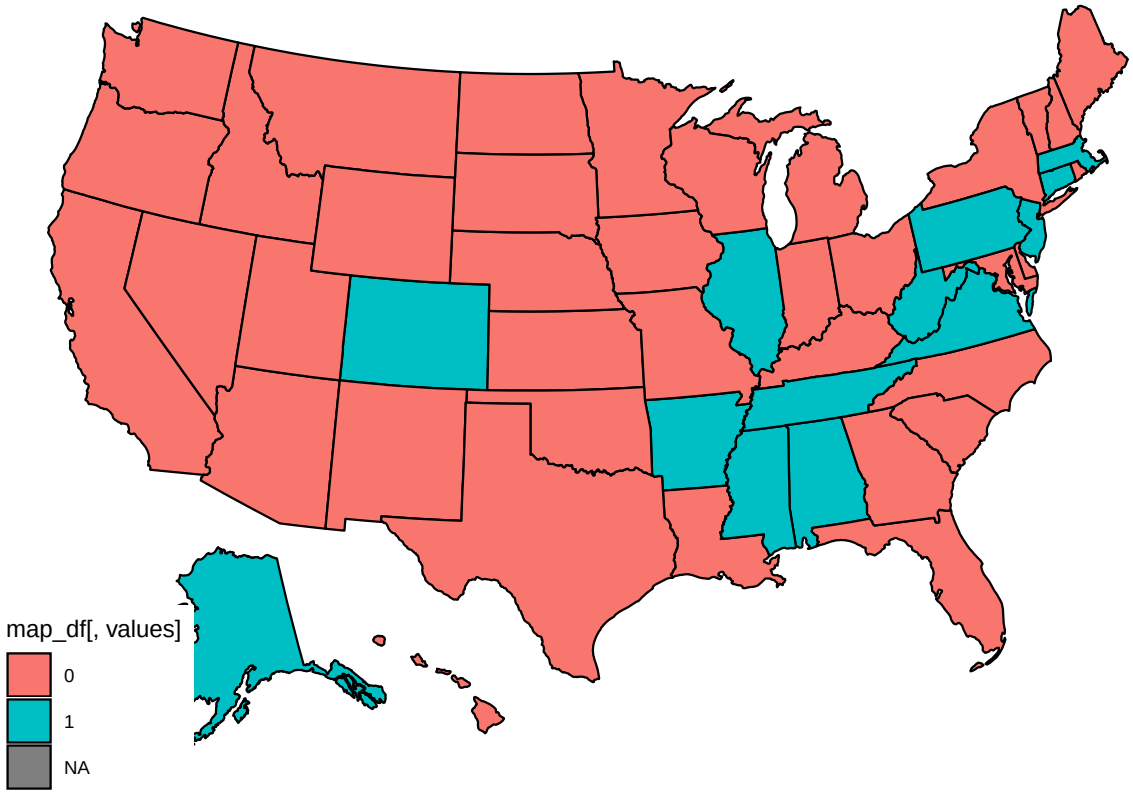
```
states <- fread(input = "./us_cities_and_states/states.sql",skip = 23,sep = "'", sep2 = ",", header = F
cities <- fread(input = "./us_cities_and_states/cities_extended.sql",skip = 23,sep = "'", sep2 = ",", he

#b
dat_cities <- distinct(cities,V2,.keep_all=TRUE)
dat_cities %>% group_by(V4) %>% count()
```

```
## # A tibble: 52 x 2
## # Groups:   V4 [52]
##    V4        n
##    <chr> <int>
##  1 AK      182
##  2 AL      367
##  3 AR      312
##  4 AZ      174
##  5 CA      807
##  6 CO      193
##  7 CT      176
##  8 DC        2
##  9 DE       27
## 10 FL      403
## # ... with 42 more rows
```

```r
count_cities <- dat_cities %>% group_by(V4) %>% count()
names(count_cities) <- c("state", "count")

#c
load("fifty_states.rda")
crimes <- data.frame(state = tolower(rownames(USArrests)), USArrests)
letter_count <- data.frame(matrix(NA,nrow=50, ncol=26))
state_name <- paste(crimes$state,sep = "")
letter <- paste(letters,sep = "")
rownames(letter_count) <- state_name
colnames(letter_count) <- letter

getCount <- function(state_name,letter){
  s2 <- gsub(letter,"",state_name)
  count <- nchar(state_name) - nchar(s2)
  if(count < 0)
    count <- 0

  return(count)
}
for(i in 1:50){
  for (j in 1:26) {
    letter_count[i,j] <- getCount(state_name[i],letter[j])
  }
}

#d
library(ggplot2)
library(usmap)

plot_usmap(data = count_cities, values = "count", color = "blue") +
  scale_fill_continuous(
    low = "white", high = "blue", name = "count of cities", label = scales::comma
  ) + theme(legend.position = "right")
```




```
letter_count <- cbind(letter_count,c(rep(0,50)))
colnames(letter_count)[27] <- "3 or not"
for (i in 1:50){
  for(j in 1:26){
    if(letter_count[i,j] >= 3)
      letter_count[i,27] <- 1
  }
}

dat5 <- as.data.frame(cbind(rownames(letter_count),letter_count[,27]))
colnames(dat5) <- c("state","3 or not")

plot_usmap(data = dat5, values = "3 or not",labels = FALSE)
```


map_df[, values]
0
1
NA