# W1_HW

110078509

2022/2/17

```
rm(list=ls()) #remove the random variable to fresh the working environment
ls() #suppose to be nothing 'character(0)'
```

```
## character(0)
```

```
# import raw data as data.df

#set the working directory path first
getwd() #current wd"C:/Users/LeoShr/p_space/NTHU/BACS/W1"
```

```
## [1] "C:/Users/LeoShr/p_space/NTHU/BACS/W1"
```

```
txt_wd <- paste(getwd(), '/customers.txt',sep = "" );txt_wd
```

```
## [1] "C:/Users/LeoShr/p_space/NTHU/BACS/W1/customers.txt"
```

```
data.df <- read.table(txt_wd,
                          header = TRUE,
                          row.names = NULL,
                         )
```

```
#peap the dataframe we got
str(data.df)
```

```
## 'data.frame':    399 obs. of  1 variable:
##  $ age: int  49 69 41 73 45 71 50 43 70 32 ...
```

```
attach(data.df)
```

I attached the data.df for Q1-Q6 cuz I am lazy to specify the dataframe I used. Kindly run my code from top to bottom with order.

#Q1. What is the 5th element in the original list of ages?

```
ans1 <-age[5]; ans1 #45
```

```
## [1] 45
```

#2. What is the fifth lowest age?

```
unique(sort(age))[5] #22
```

```
## [1] 22
```

*Explain:* sort the vector then remove the duplicate data via unique

Q3. Extract the five lowest ages together

```
head(unique(sort(age)),5)
```

```
## [1] 18 19 20 21 22
```

```
#ans: [1] 18 19 20 21 22
```

*Explain:* use sort to get the ascending data then get the first 5 via 'head'

Q4. Get the five highest ages by first sorting them in decreasing order first.

```
head(unique(sort(age, decreasing = TRUE)), 5)
```

```
## [1] 85 83 82 81 80
```

```
# ans : [1] 85 83 82 81 80
```

*Explain:*same concept with the Q3 but with parameter setting as decreasing. That how we got the five highest ages.

Q5. What is the average (mean) age?

```
mean(age)
```

```
## [1] 46.80702
```

```
# ans : [1] 46.80702
```

*Explain:* Get average value of numeric data via mean()

Q6. What is the standard deviation of ages? (guess or google the standard deviation function in R)

```
sd(age)
```

```
## [1] 16.3698
```

```
detach(data.df)#I detach the data frame here
```

*Explain:* sd means Standard Deviation.

Q7. Make a new variable called age_diff, with the difference between each age and the mean age

```
require("dplyr")
```

```
## 載入需要的套件：dplyr
```

```
##
## 載入套件：'dplyr'
```

```
## 下列物件被遮斷自 'package:stats':
##
##     filter, lag
```

```
## 下列物件被遮斷自 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
data_2factor<- data.df;#to maintain the raw data unchanged
data_2factor  <- mutate(data_2factor, age_diff = age - mean(age) )
str(data_2factor)#show  the result here
```

```
## 'data.frame':    399 obs. of  2 variables:
##  $ age     : int  49 69 41 73 45 71 50 43 70 32 ...
##  $ age_diff: num  2.19 22.19 -5.81 26.19 -1.81 ...
```

*Explain:* 1. i detached the raw df first because we're gonna to mutate the variable with additional column called age_diff 2. Using mutate function 3. show the result here via str()

Q8. What is the average "difference between each age and the mean age"?

*Explain:*

HINT: think carefully why someone would want to know this, and what it implies about how to do #6

```
mean(abs(data_2factor$age_diff))
```

```
## [1] 12.66948
```

```
mean(data_2factor$age_diff)
```

```
## [1] -1.623275e-15
```

This question required to follow the mind set of Q6, which is Standard Deviation. The Standard Deviation is the average distance form the mean, therefore, we should use absolute to represent distance in math.

Q9. Visualize the raw data as we did in class: (a) histogram, (b) density plot, (c) boxplot+stripchart

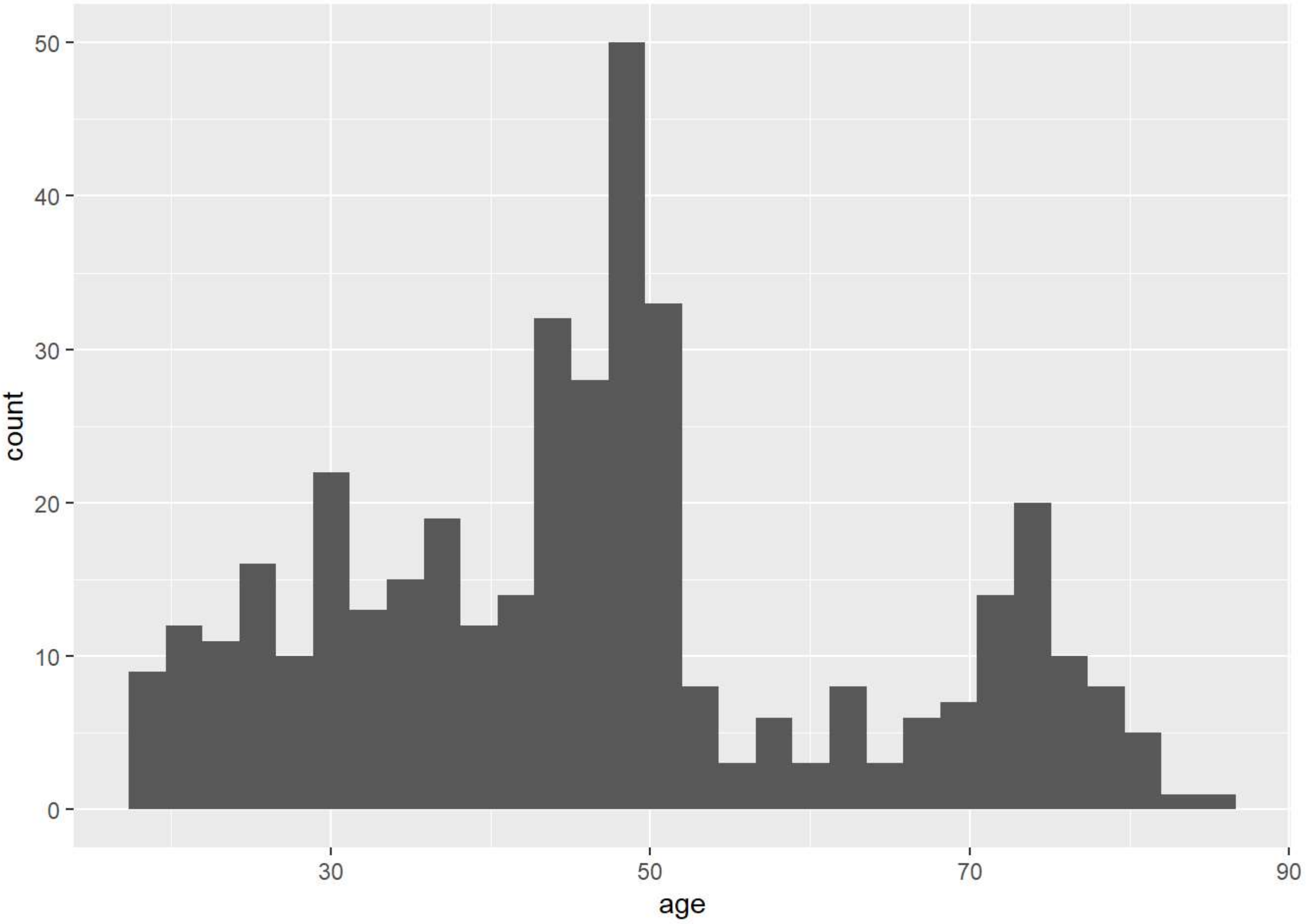
a. histogram

```
require('ggplot2')
```

```
## 載入需要的套件：ggplot2
## Warning: 套件 'ggplot2' 是用 R 版本 4.1.2 來建造的
```

```
data_2factor %>% #pipeline
    ggplot( aes(x = age)) +
    geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

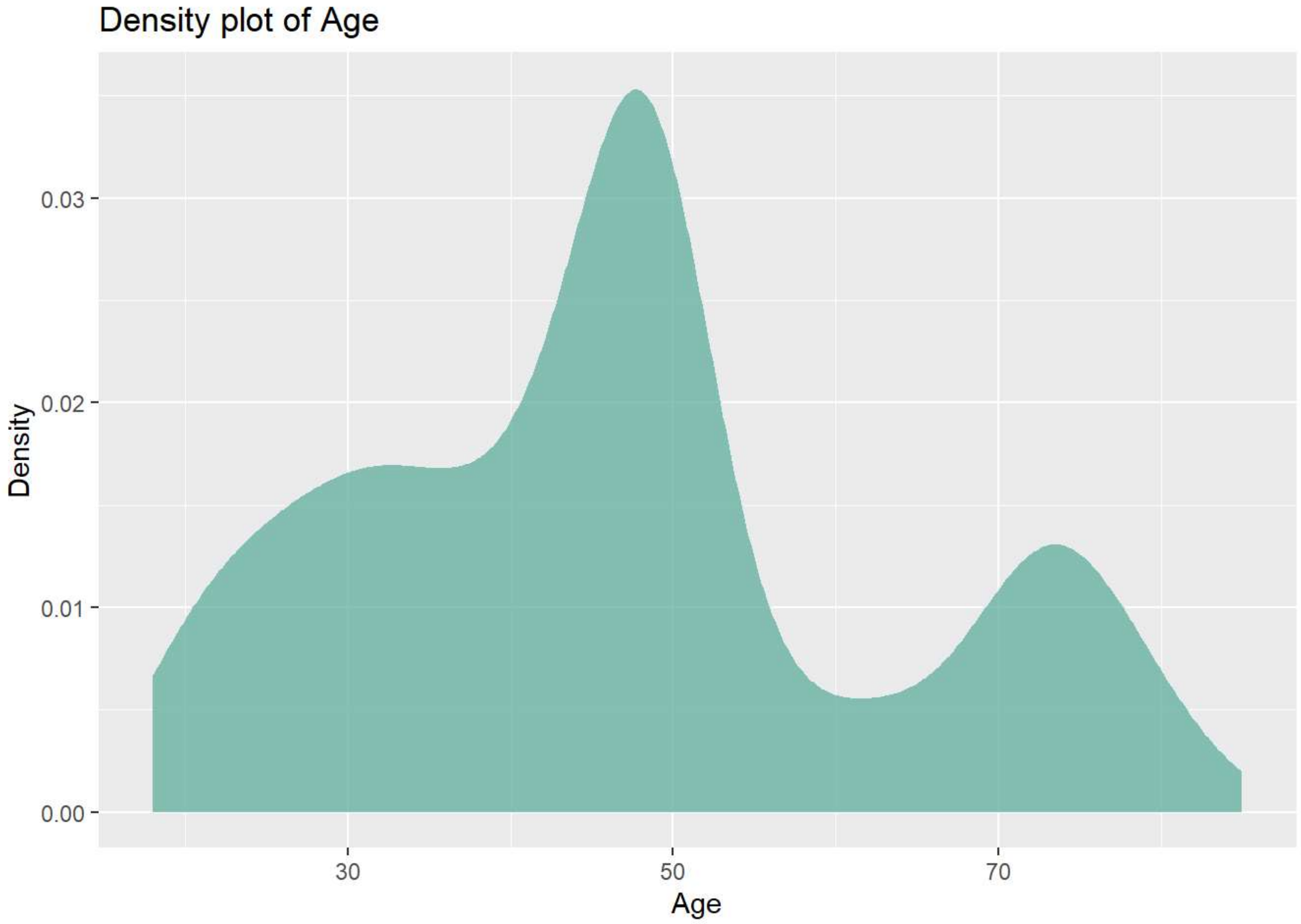


*Explain:*

Using ggplot with geom_histogram()

b. density plot

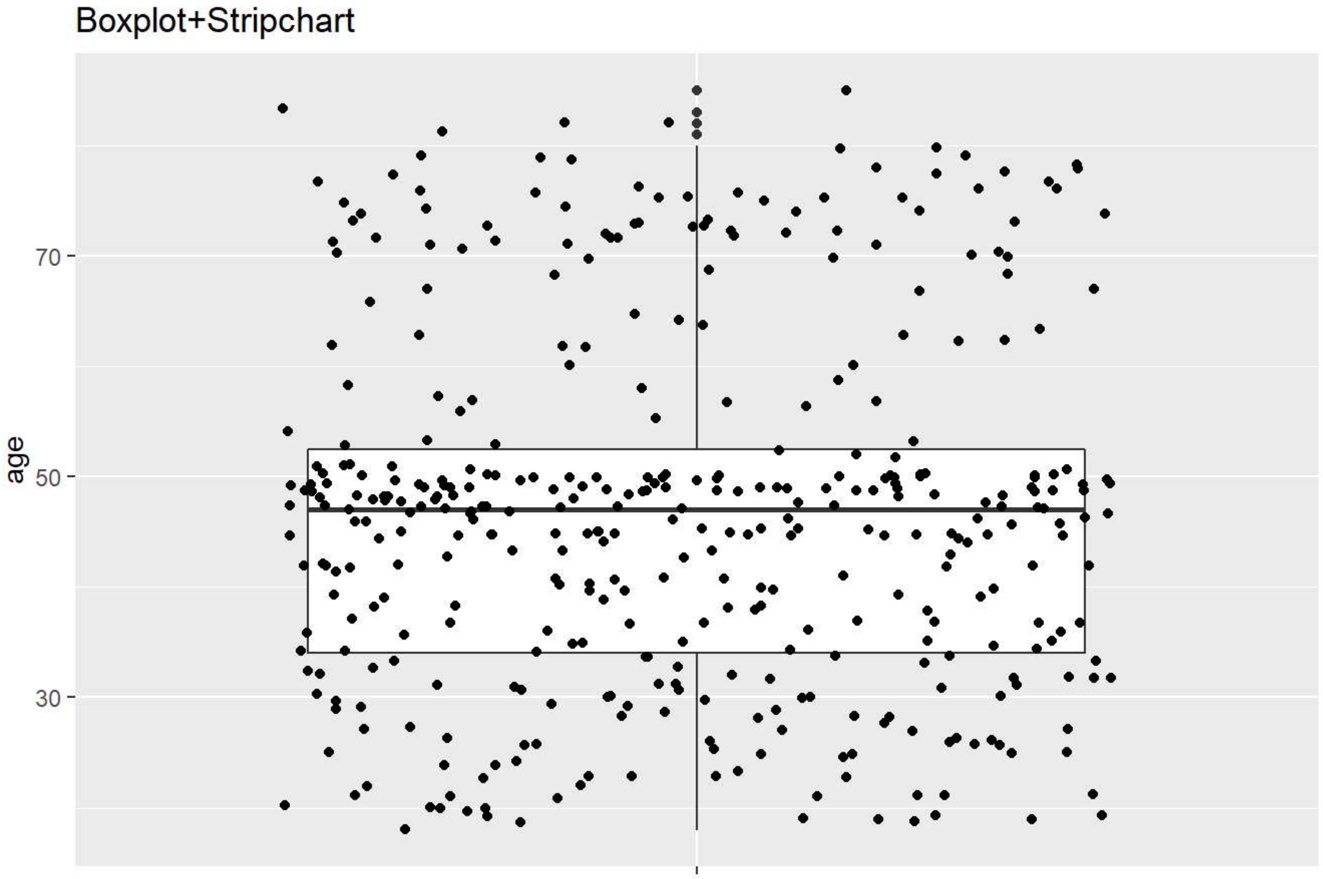
```
data_2factor  %>%
ggplot( aes(x=age)) +
    geom_density(fill="#69b3a2", color="#e9ecef", alpha=0.8)+
    #fill the distribution region with color
    # Note: alpha for transparent level
    ggtitle("Density plot of Age") + #add the title of chart
    xlab('Age')+ #x axis label
    ylab('Density') #y  label
```



*Explain:* Using ggplot with geom_density().

c. boxplot+stripchart

```
data_2factor %>%
    ggplot(aes(x = "", y = age)) +
    geom_boxplot() +
    geom_jitter()+
    #The reason of adoption of geom_jitter is for better      #visualization performance. I m
ade them overlapped.
    ggtitle("Boxplot+Stripchart")+
    xlab('') #remove the x lab
```



*Explain:* As the graph shows, 50% of the data point located in the range if IQR (Q3-Q1)