BACS Homework 1

109062710

February 25, 2020

I. Importing Dataset

```
customers <- read.table("./customers.txt", header=TRUE)
```

After importing, we would like to see the detailed information of our data set.

```
summary(customers)
```

```
##       age
##  Min.   :18.00
##  1st Qu.:34.00
##  Median :47.00
##  Mean   :46.81
##  3rd Qu.:52.50
##  Max.   :85.00
```

II. Questions

1. The fifth element in the dataset

```
customers$age[c(1:20)]
```

```
##  [1] 49 69 41 73 45 71 50 43 70 32 47 77 64 50 50 45 49 47 62 50
```

First, we should print out the first 20 numbers in the data set to see what numbers there are. As we can see, we know the fifth entry in the data set is `45`.

The code below is how we acquire the fifth entry of the data.

```
customers$age[5]
```

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

Below is the explanation how the code above works:

First, we call our data set called `customer`. Then, we use `$` to access a user specified column. Since there is only one column, `$age` it is.

What `customers$age[5]` tells us is we want the fifth entry `[5]` in `age` column in a data set called `customers`

2. What is the fifth lowest age?

First, we have to order the `age` column in an ascending manner. Then we print out the first ten entries of the ascending ordered data set.

labs(title = "Age of customers", y = "Age")

```
age_ascending <- customers[order(customers$age),]
age_ascending[c(0:10)]
```

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

As it's can be seen here, the fifth lowest age is `19`. In order to extract the fifth lowest age in the ascending ordered data set, we can do it like below

```
age_ascending[c(5)]
```

```
## [1] 19
```

3. Extract the five lowest ages together.

There are two ways to do it

First, the hard-coded one

```
age_ascending[c(1,2,3,4,5)]
```

```
## [1] 18 19 19 19 19
```

Second, it's much simpler than the previous one

```
age_ascending[c(1:5)]
```

```
## [1] 18 19 19 19 19
```

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

```
customers[order(customers$age, decreasing=TRUE),][c(1:5)]
```

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

5. What is the average (mean) age?

```
colMeans(customers)
```

```
##      age
## 46.80702
```

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

Slower way,

```
mean <- colMeans(customers)
sum <- sum((customers$age - mean)^2)
std <- sqrt(sum / length(customers$age))
std
```

```
## [1] 16.34927
```

```
sd(customers$age)
```

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

As we can see here, the standard deviation value from both ways differ by `0.02` points which is considered to be really small.

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

```
mean <- colMeans(customers)
age_diff <- abs(customers$age - mean)
age_diff[c(1:10)]
```

```
##  [1]  2.192982 22.192982  5.807018 26.192982  1.807018 24.192982  3.192982
##  [8]  3.807018 23.192982 14.807018
```

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

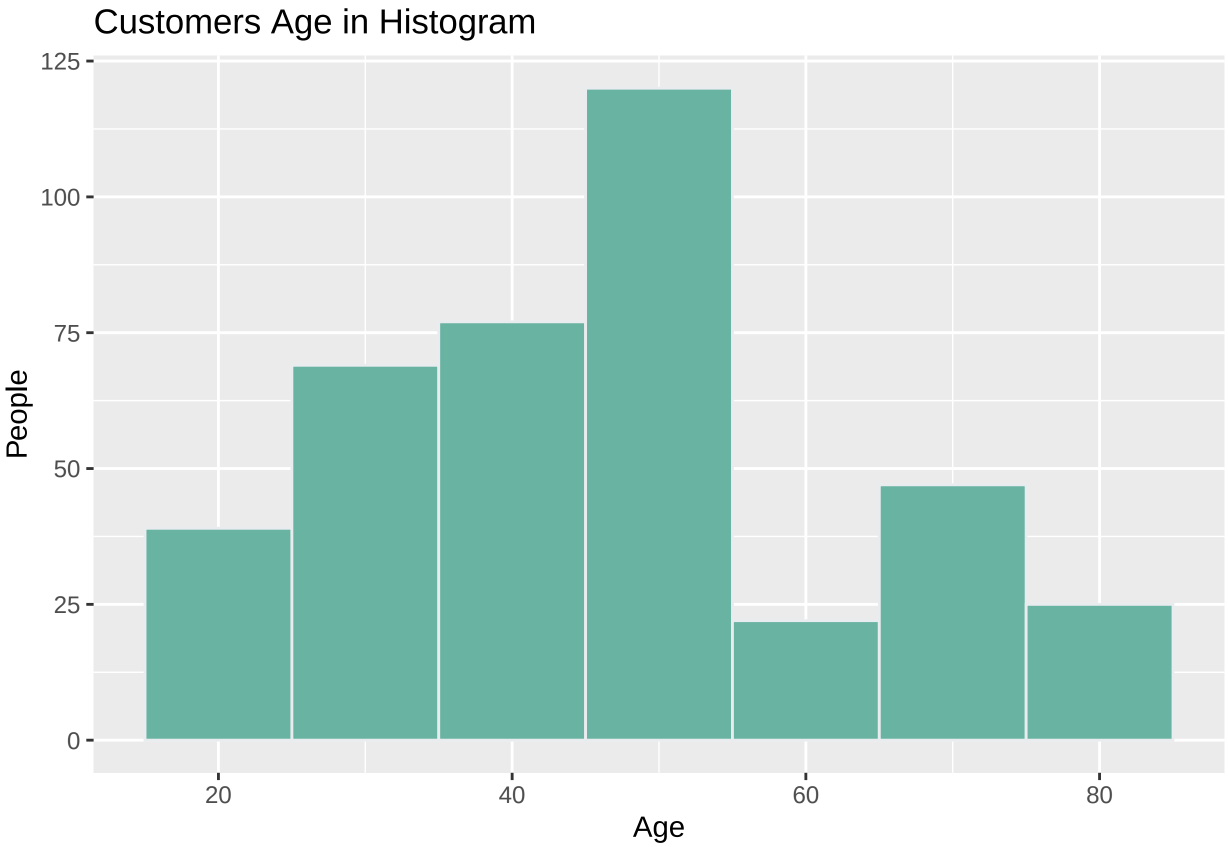
```
avg_age_diff <- mean(age_diff)
avg_age_diff
```

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

9. Visualize the raw data as we did in class: (a) Histogram, (b) Density Plot, (c) Box Plot (d) Strip Chart

(a) Histogram

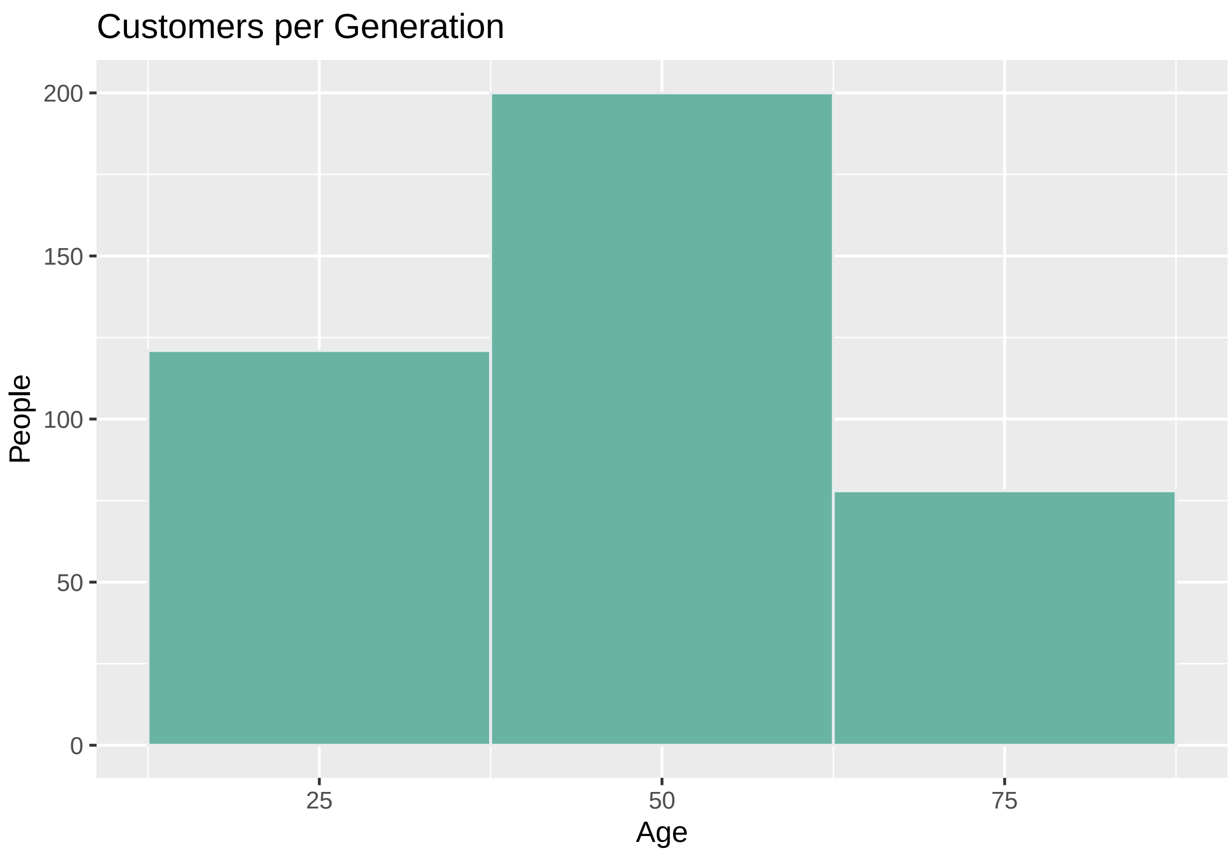
```
ggplot(customers, aes(x=age)) +
  geom_histogram(
    binwidth=10,
    fill="#69b3a2",
    color="#e9ecef"
  ) +
  ggtitle("Customers Age in Histogram") +
  labs(x="Age", y="People")
```



From the table above, most customers are in their fifties.

On Google, one generation is around 20 to 30 years. Let’s take a number in between and set the `binwidth=25`.

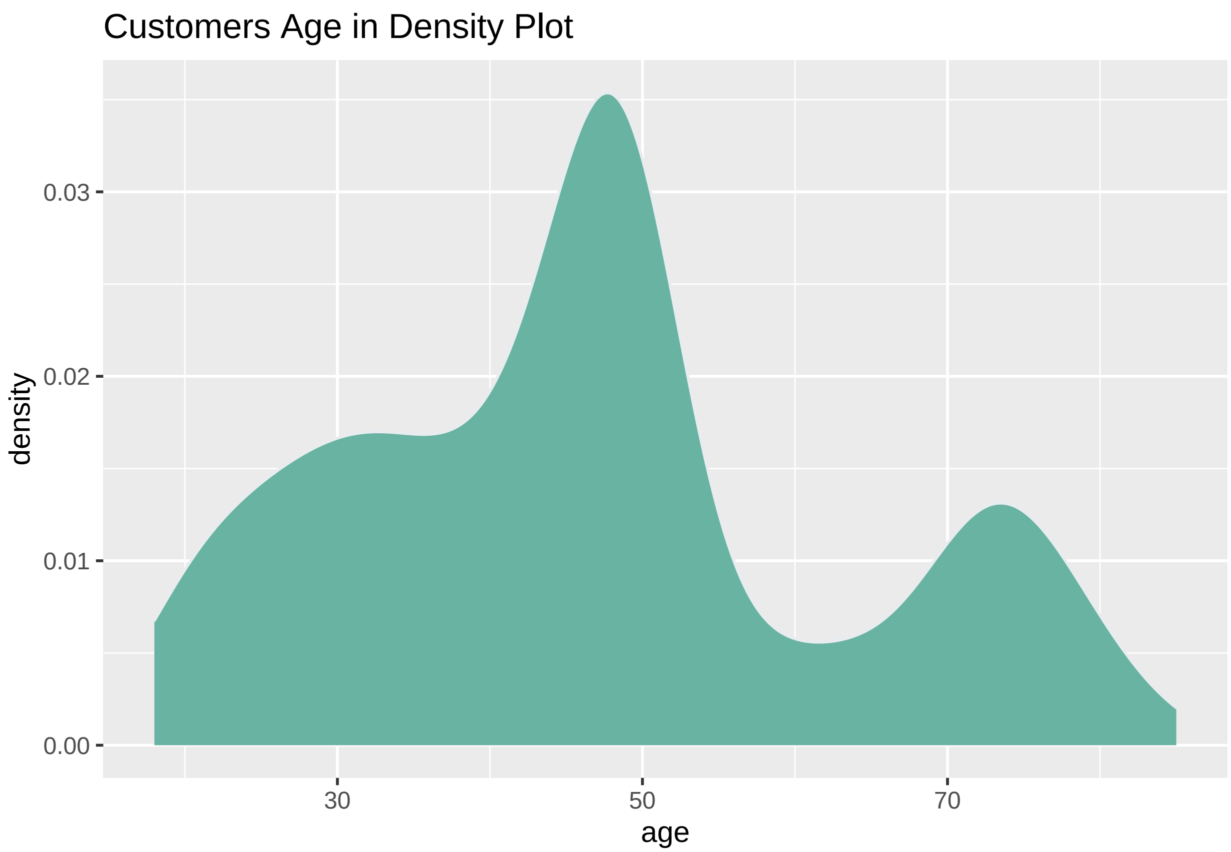
```
ggplot(customers, aes(x=age)) +
  geom_histogram(
    binwidth=25,
    fill="#69b3a2",
    color="#e9ecef"
  ) +
  ggtitle("Customers per Generation") +
  labs(x = "Age", y = "People")
```



As we can see here, most customers are around 26 to 50.

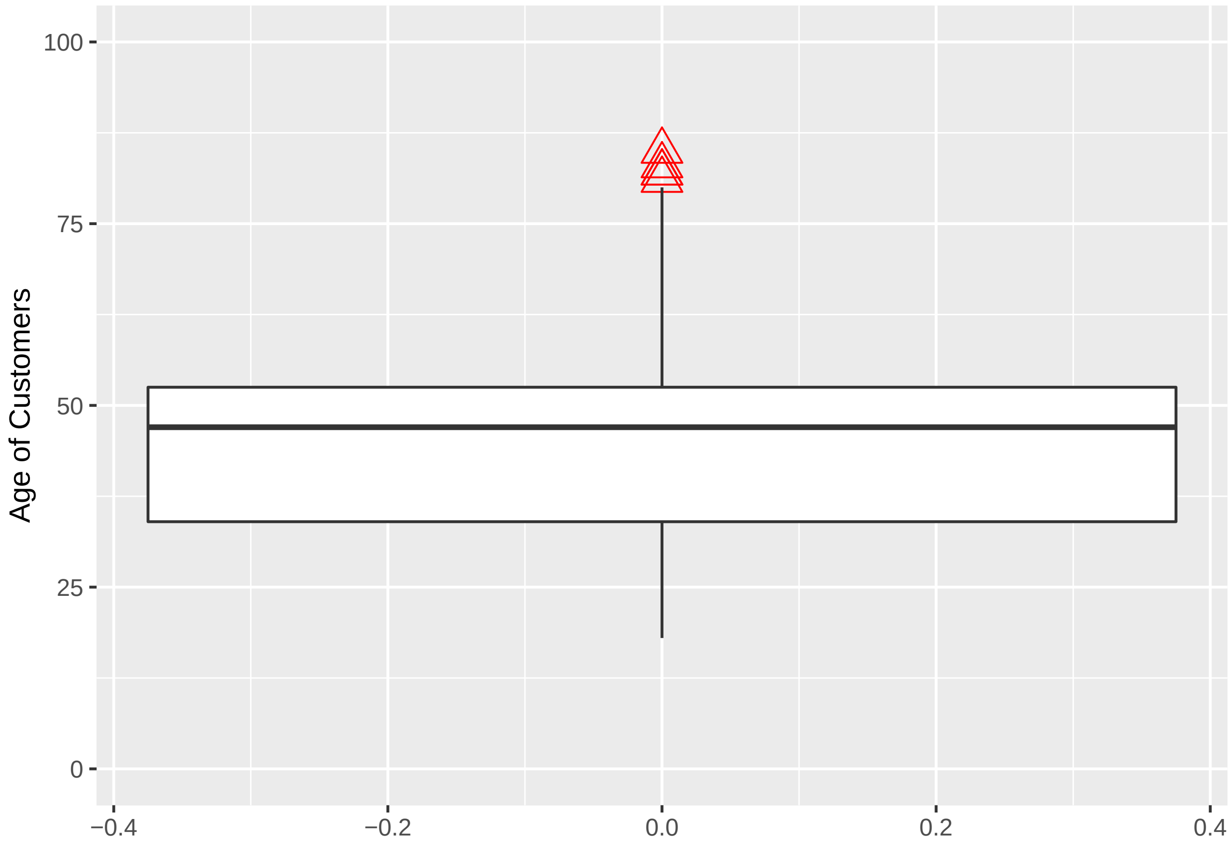
(b) Density Plot

```
ggplot(customers, aes(x=age)) +
  geom_density(
    fill="#69b3a2",
    color="#e9ecef"
  ) +
  ggtitle("Customers Age in Density Plot")
```



(c) Box Plot

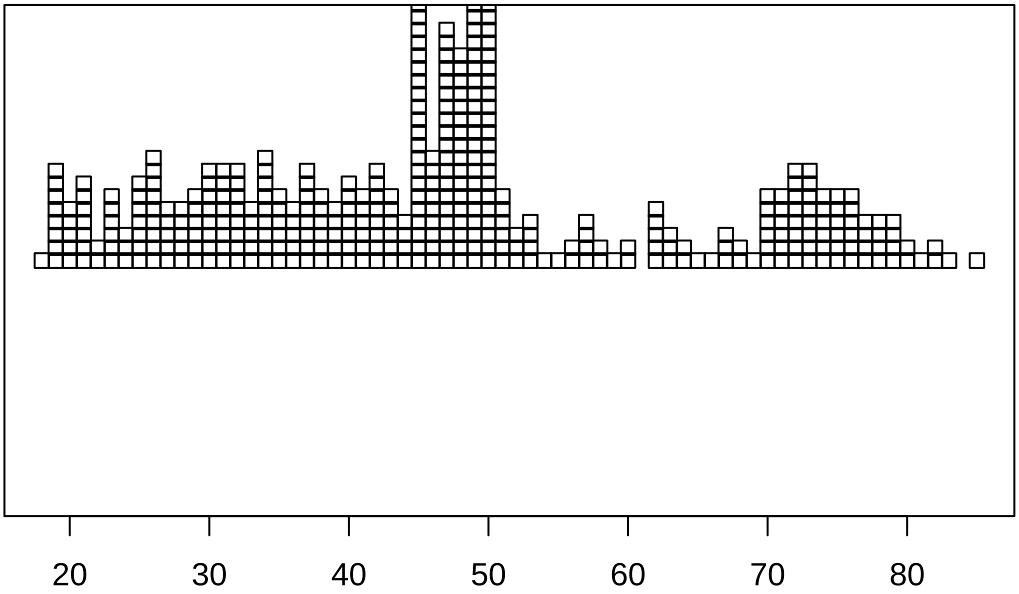
```
customers_age = customers$age
ggplot() +
  geom_boxplot(
    aes(y=customers_age),
    outlier.color = "red",
    outlier.shape=2,
    outlier.size = 5
  ) +
  ylim(c(0, 100)) +
  labs(y = "Age of Customers")
```



We can see that there are few outliers in red triangles.

(d) Strip Chart

```
stripchart(x = customers$age, method="stack")
```


20
30
40
50
60
70
80