Homework 1

Junhao Wang(jw3668)

9/14/2018

Note: Grading is based both on your graphs and verbal explanations. Follow all best practices as discussed in class.

Read *Graphical Data Analysis with R*, Ch. 3

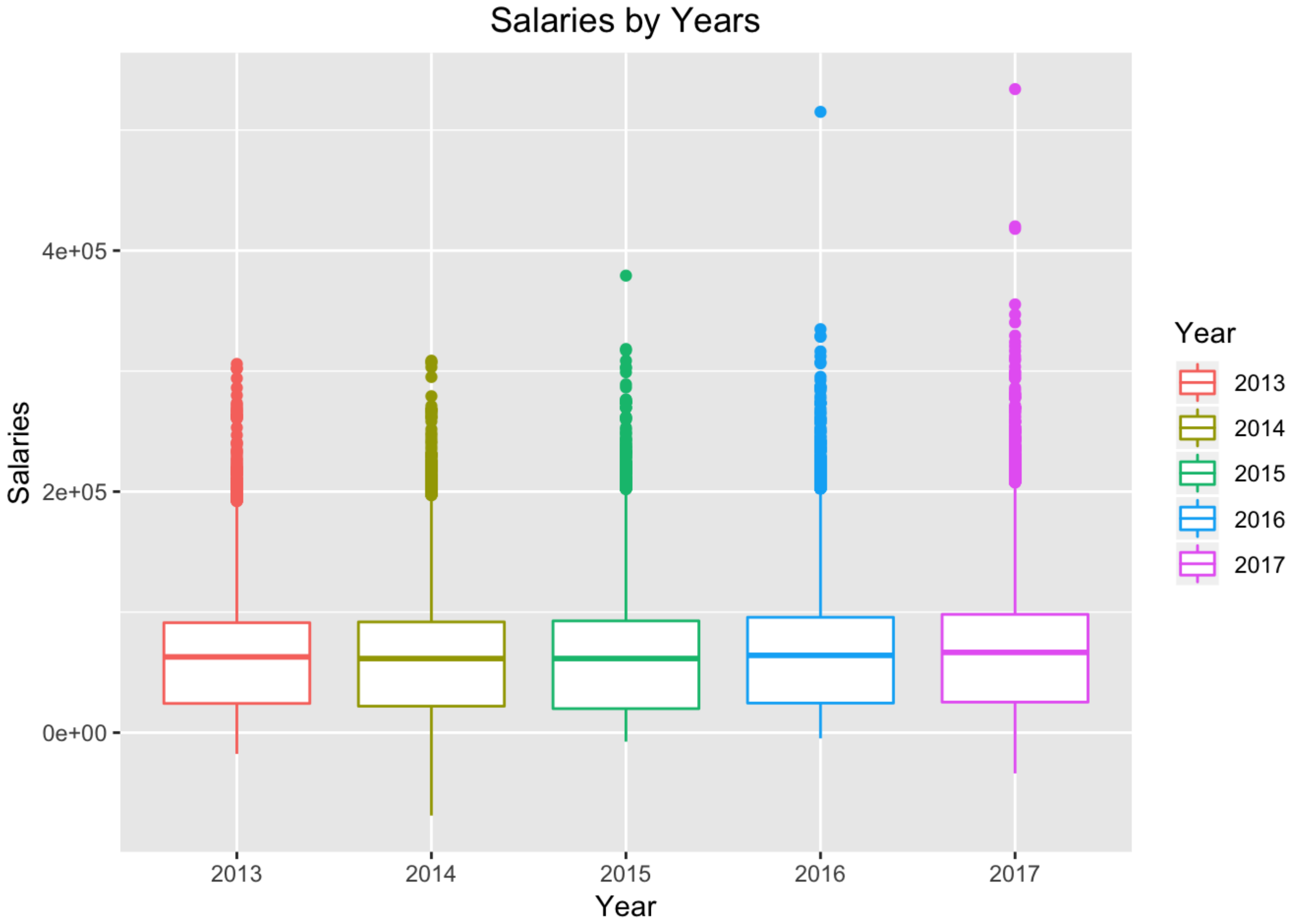
1. Salary

[15 points]

```
library(ggplot2)
library(dplyr)
library(gridExtra)
library(gapminder)
employeeData = read.csv("Employee.csv")
employeeData$Year = as.factor(employeeData$Year)
employeeData$Organization.Group = as.factor(employeeData$Organization.Group)
```

a. Draw multiple boxplots, by year, for the `Salaries` variable in *Employee.csv* (Available in the Data folder in the Files section of CourseWorks, original source: https://catalog.data.gov/dataset/employee-compensation-53987 (https://catalog.data.gov/dataset/employee-compensation-53987)). How do the distributions differ by year?

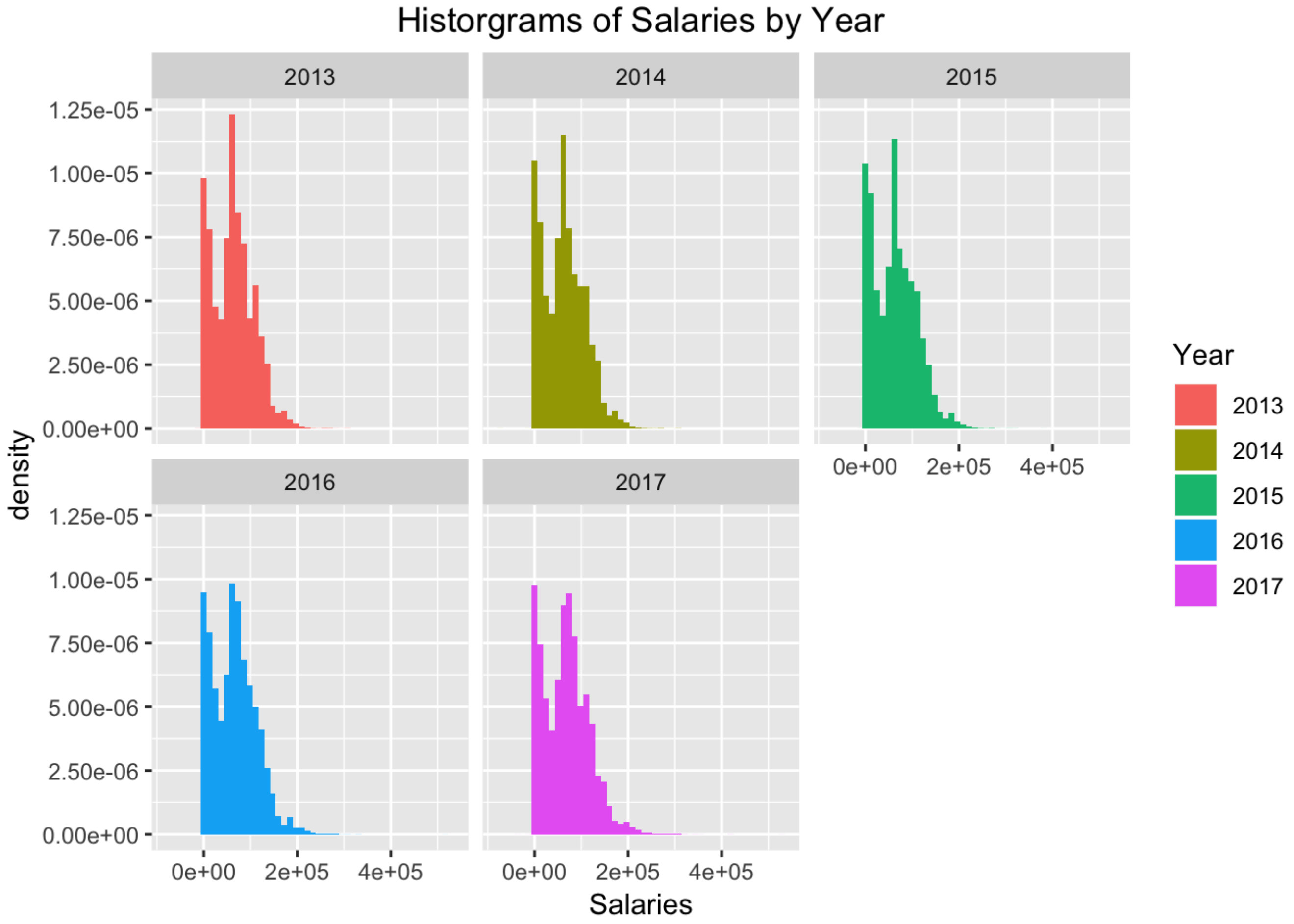
```
ggplot(employeeData,aes(x = Year, y = Salaries, color =Year)) +
  geom_boxplot() +
  ggtitle("Salaries by Years") +
  scale_x_discrete(name = "Year") +
  theme(plot.title = element_text(hjust = 0.5))
```



From the plot above, what we can discover is that 25th percentile,50th percentile, 75 percentile of salaries doesn't really change from 2013 to 2017. However, we can notice the highest salaries for 2017 is higher than all of the following year.

b. Draw histograms, faceted by year, for the same data. What additional information do the histograms provide?

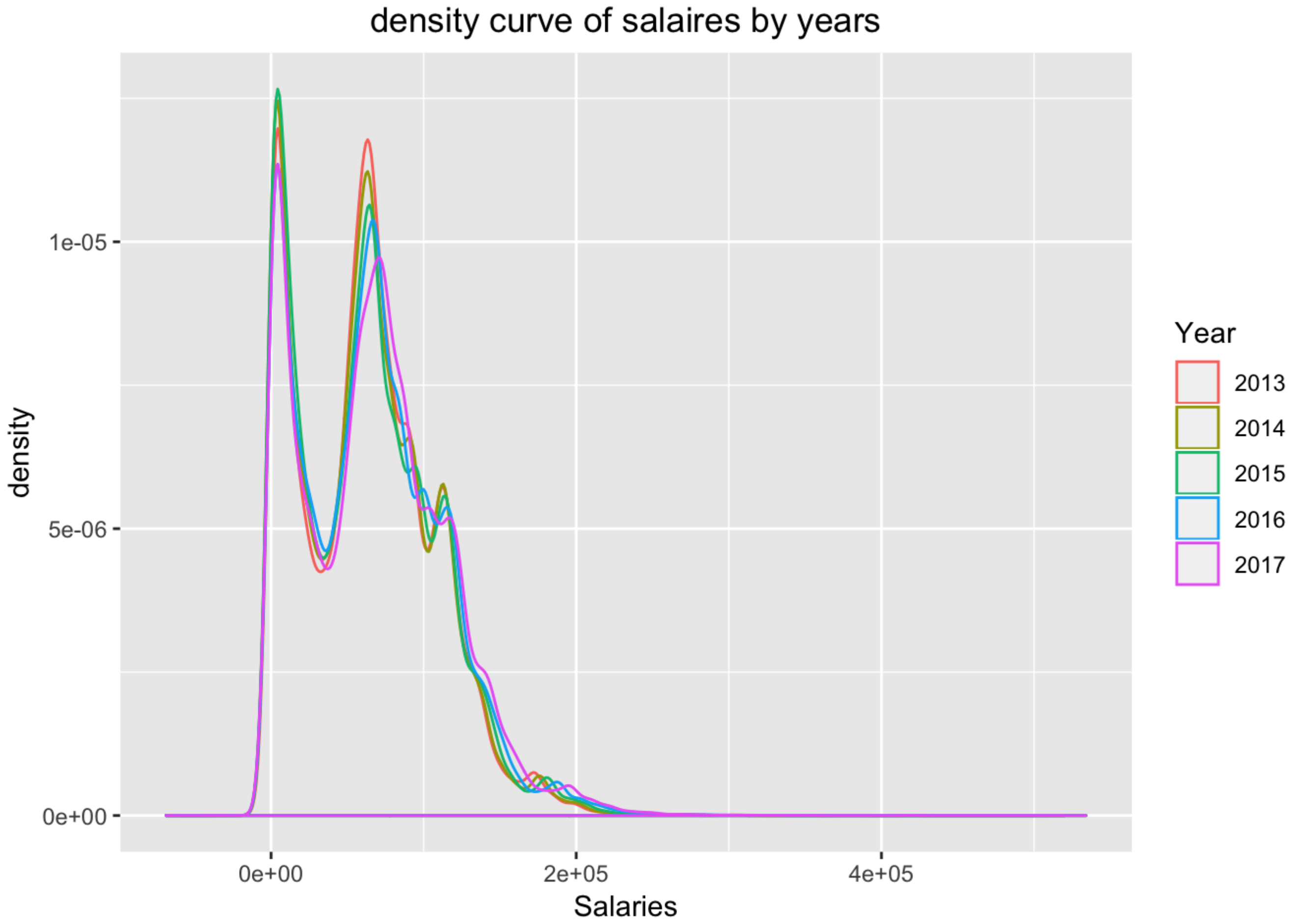
```
ggplot(employeeData,aes(x = Salaries, y =..density.., fill = Year)) +
  geom_histogram(bins = 50) +
  facet_wrap(~Year) +
  ggtitle("Historgrams of Salaries by Year") +
  theme(plot.title = element_text(hjust = 0.5))
```



The histograms acutally tells me a relatively more acurate distribution for the salaries across the years. From the plot, it’s not hard to discover that the distribution for salaries doesn’t vary much accross years, yet there are slightly more outilers, people with mich higher salaires, in 2016 and 2017.

c. Plot overlapping density curves of the same data, one curve per year, on a single set of axes. Each curve should be a different color. What additional information do you learn?

```
ggplot(employeeData,aes(x = Salaries, color = Year)) +
  geom_density(alpha = 0.5) +
  ggtitle("density curve of salaires by years") +
  theme(plot.title = element_text(hjust = 0.5))
```



The density curve give us a very accurate infomation about the distribution of the salaries of each year. It confirms the idea the salaires distribution from year 2013 to year 2017 doesnt change that much. However, what I can also notice is that the proportition of people in the middle class is dropping a bit as years pass by ,and move more to the higher salaries range.

d. Sum up the results of a), b) and c): what kinds of questions, specific to this dataset, would be best answered about the data by each of the three graphical forms?

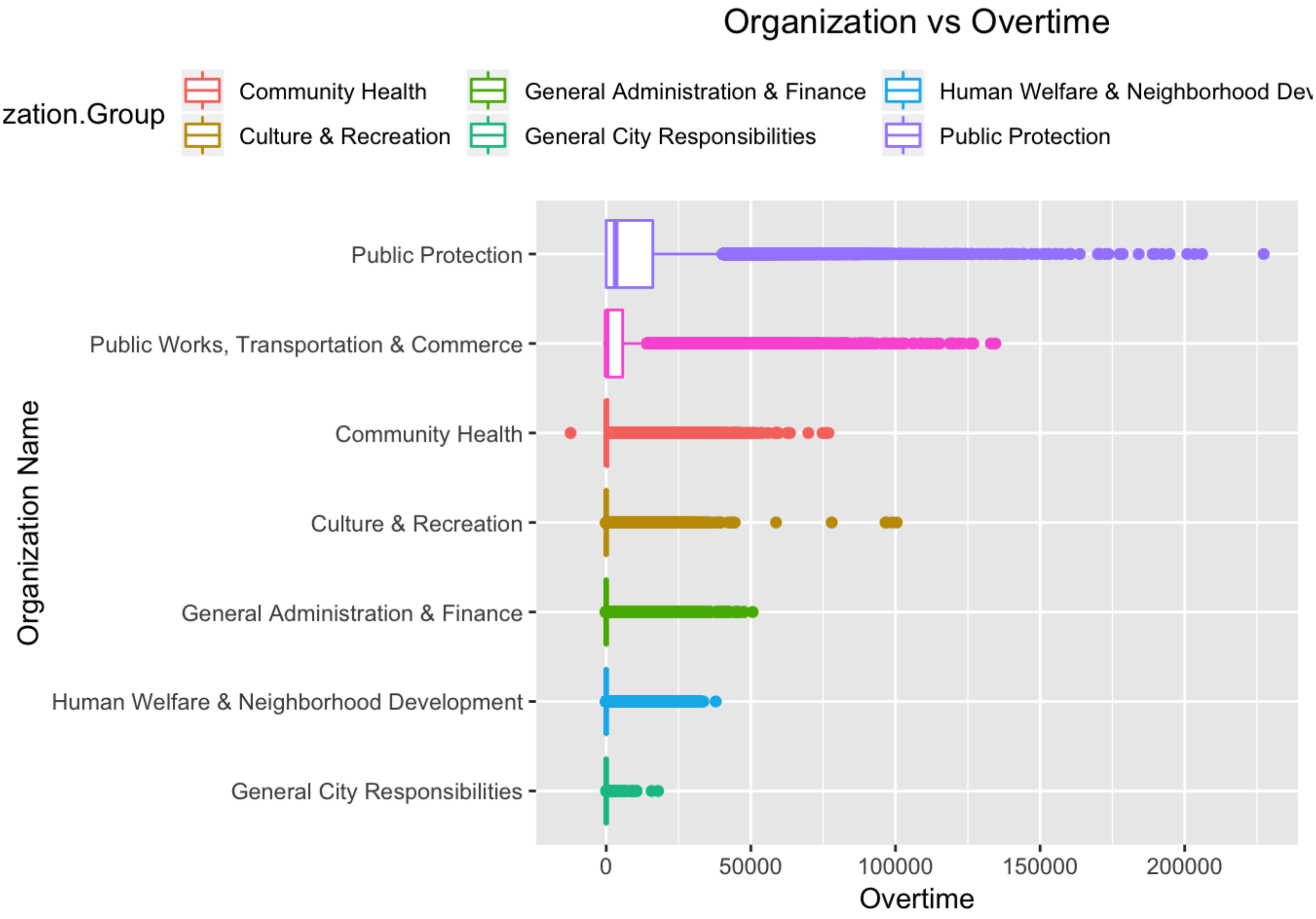
For this particular dataset, the density curve is the best choice. The distribution of salaries by years are very close to each other. Boxplot cannot really capture the nuance from years to years since the the quantiles doesn't move by that much. The histograms cannot make useful inference due to the high similarities the salaires distribution share. Under this circumstance, a density curve convey the most information as it capture very detailed shape of the distribution

2. Overtime

[10 points]

a. Draw multiple horizontal boxplots, grouped by `Organization Group` for the `Overtime` variable in *Employee.csv* The boxplots should be sorted by group median.Why aren't the boxplots particularly useful?

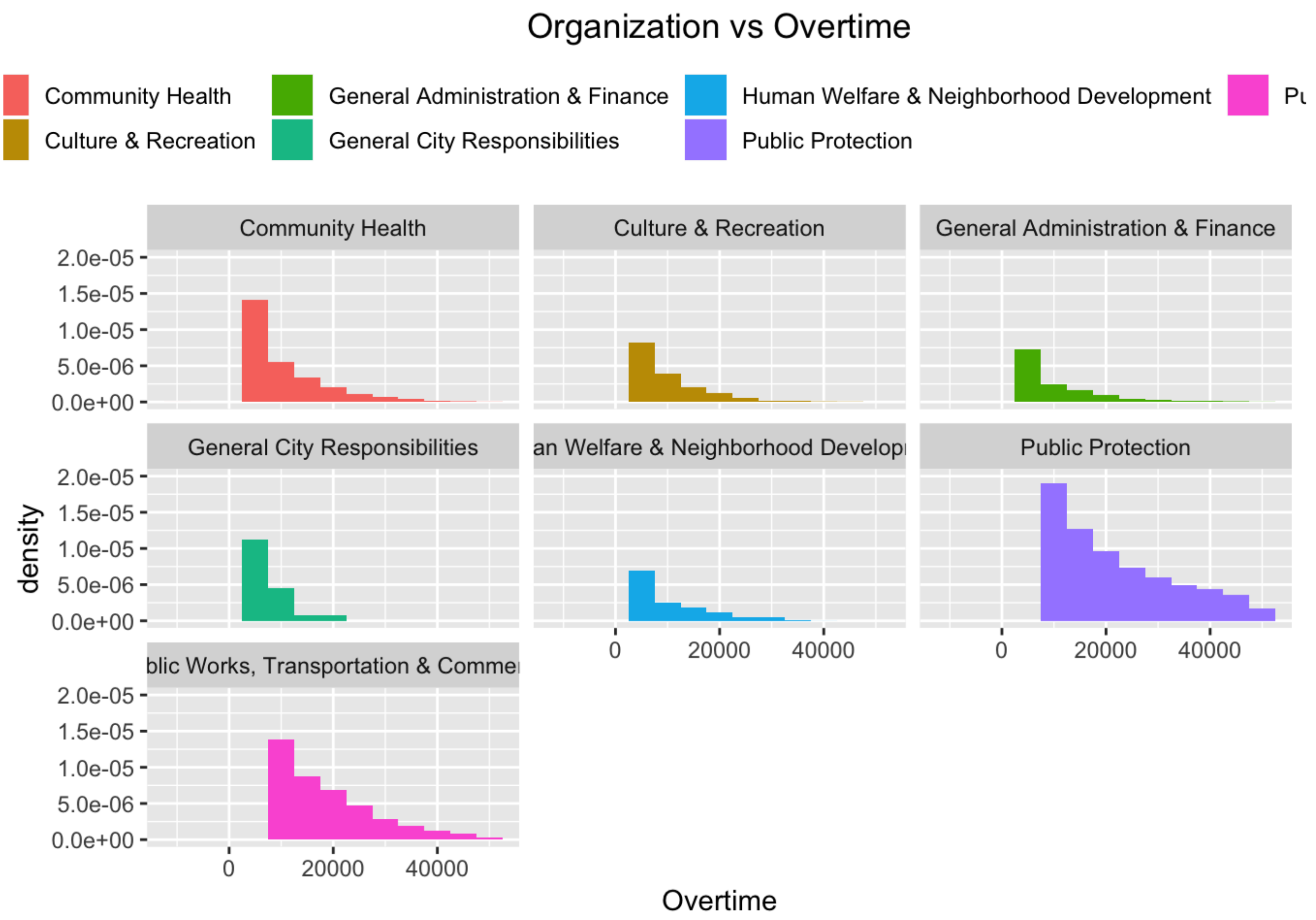
```
ggplot(employeeData, aes(x = reorder(Organization.Group,Overtime, Fun = median()) , y =
Overtime, color = Organization.Group)) +
  geom_boxplot() +
  coord_flip() +
  xlab("Organization Name") +
  theme(legend.position="top") +
  ggtitle("Organization vs Overtime") +
  theme(plot.title =element_text(hjust = 0.5)) +
  guides(fill=guide_legend(title="Organization"))
```



The boxplots aren't very useful in this scenario because there are too many outliers. The majority of the people doesn't over time or not by much which make the boxplots looks like a line for the bottom five sectors The only information we can get from this graph is basically that the public protection and pulbic works have more overtime while the public protection have the highest overtime. We cannot infer much about the other sectors.

b. Either subset the data or choose another graphical form (or both) to display the distributions of `Overtime` by `Organization Group` in a more meaningful way. Explain how this form improves on the plots in part a).

```
employeeData = employeeData[employeeData$Overtime < 50000,]
ggplot(employeeData, aes(x = Overtime, y = ..density.., fill = Organization.Group )) +
  geom_histogram(binwidth = 5000) +
  facet_wrap(~Organization.Group) +
  ggtitle("Organization vs Overtime") +
  ylim(0,0.00002) +
  theme(plot.title = element_text(hjust= 0.5)) +
  theme(legend.position="top")
```



I first subset out some outliers in the data so that the data won't by very skewed. Then I choose boxplots to analyze the distribution of overtime given by different secotrs. What can be clearly seen is that public protection has high overtime when comparing with the rest which have similar distributions

3. Boundaries

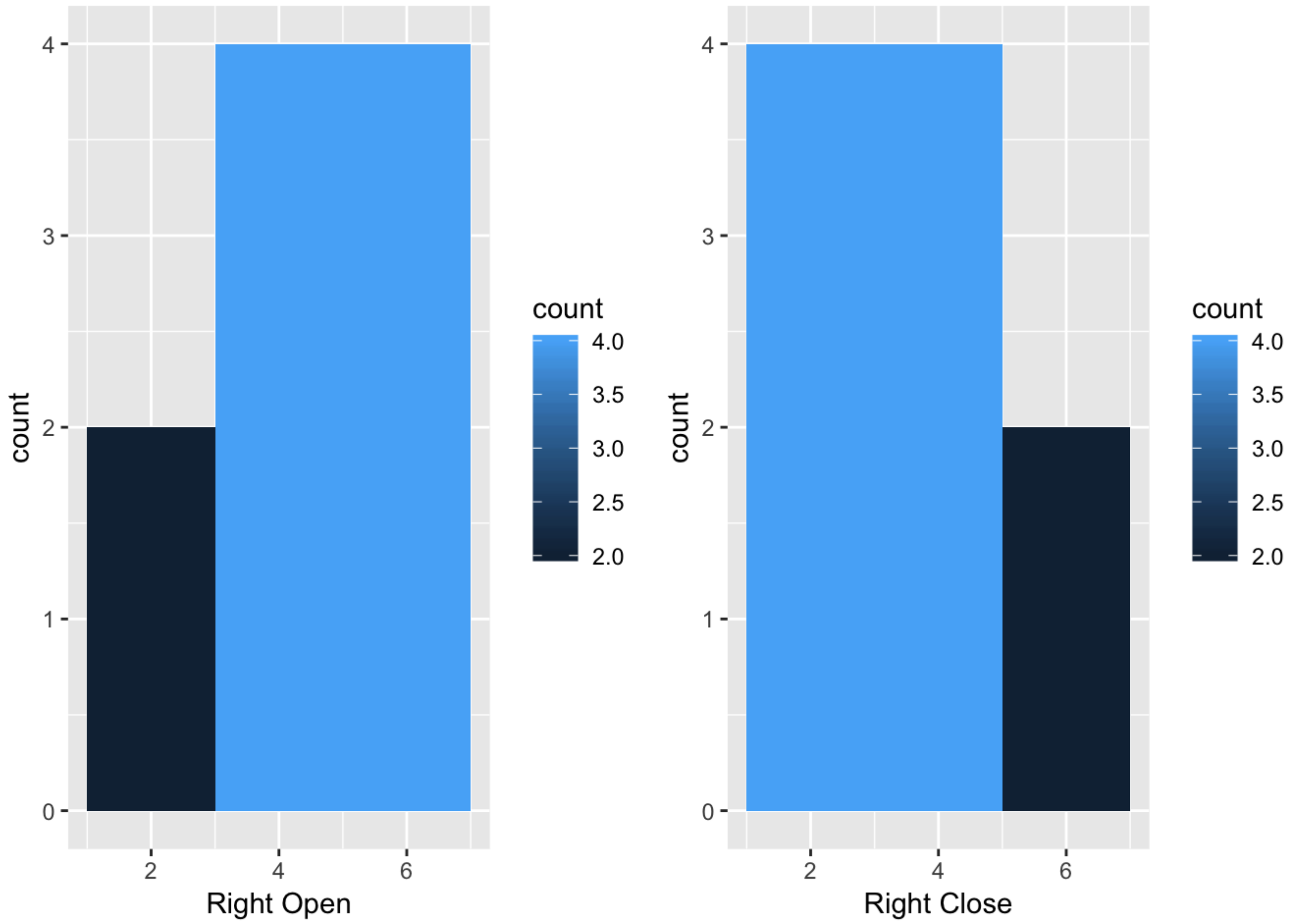
[10 points]

a. Find or create a small dataset (< 100 observations) for which right open and right closed histograms for the same parameters are not identical. Display the full dataset (that is, show the numbers) and the plots of the two forms.

```
c <- rpois(10,3)
c <- data.frame(c)
colnames(c) = ("Generated_Data")
print(c)
```

```
##    Generated_Data
## 1               5
## 2               3
## 3               4
## 4               6
## 5               5
## 6               6
## 7               1
## 8               3
## 9               1
## 10              4
```

```
g1 = ggplot(c,aes(x = Generated_Data, fill = ..count..)) +
  stat_bin(binwidth = 2, right = FALSE) +
  xlab("Right Open")
g2 = ggplot(c,aes(x = Generated_Data, fill = ..count..)) +
  stat_bin(binwidth = 2, right = TRUE) +
  xlab("Right Close")
grid.arrange(g1,g2,ncol = 2)
```

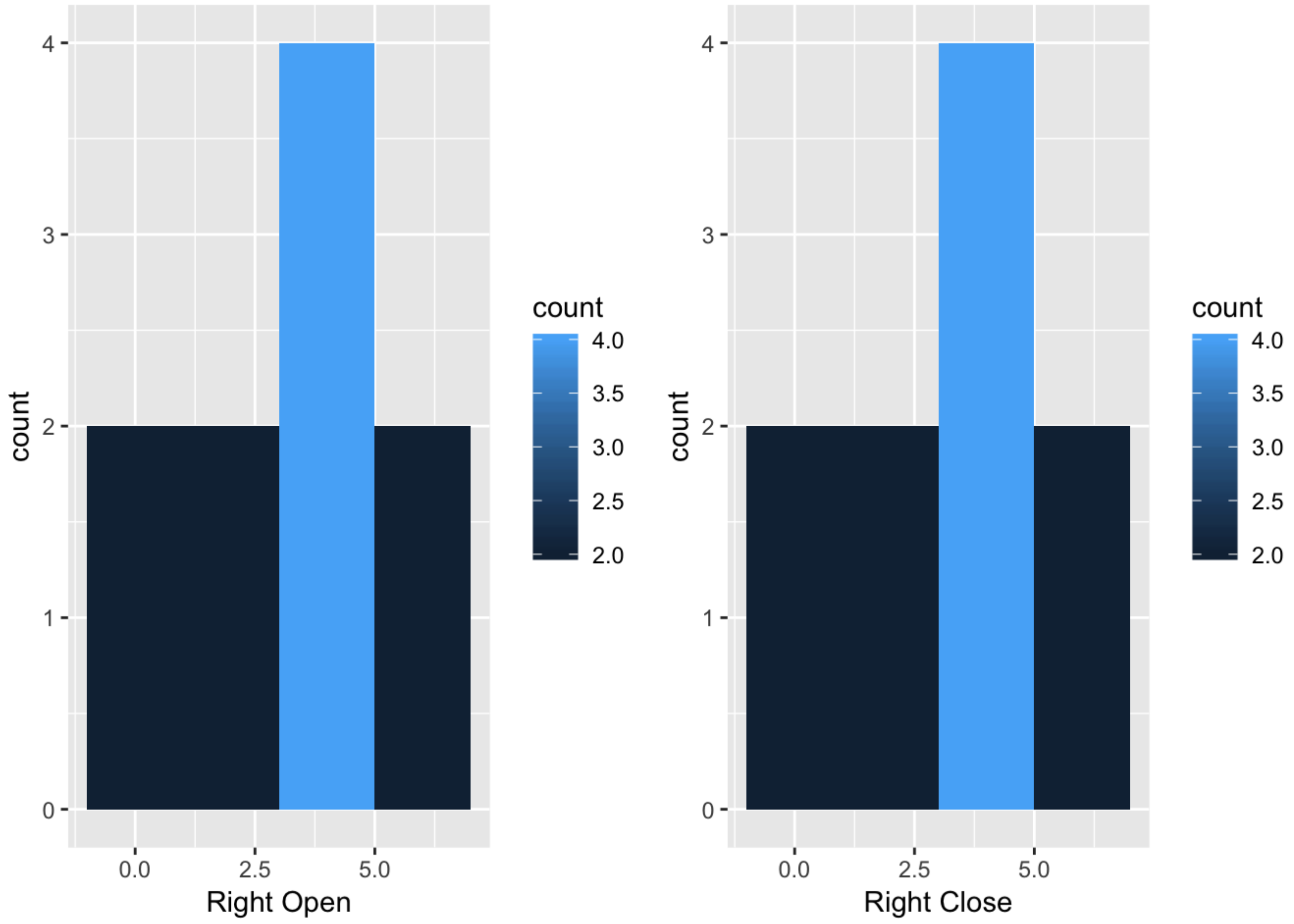


b. Adjust parameters–the same for both–so that the right open and right closed versions become identical. Explain your strategy.

```
c = c - 0.01
print(c)
```

```
##    Generated_Data
## 1            4.99
## 2            2.99
## 3            3.99
## 4            5.99
## 5            4.99
## 6            5.99
## 7            0.99
## 8            2.99
## 9            0.99
## 10           3.99
```

```
g1 = ggplot(c,aes(x = Generated_Data,fill = ..count..)) +
  stat_bin(binwidth = 2, right = FALSE) +
  xlab("Right Open")
g2 = ggplot(c,aes(x = Generated_Data,fill = ..count..)) +
  stat_bin(binwidth = 2, right = TRUE) +
  xlab("Right Close")
grid.arrange(g1,g2,ncol = 2)
```



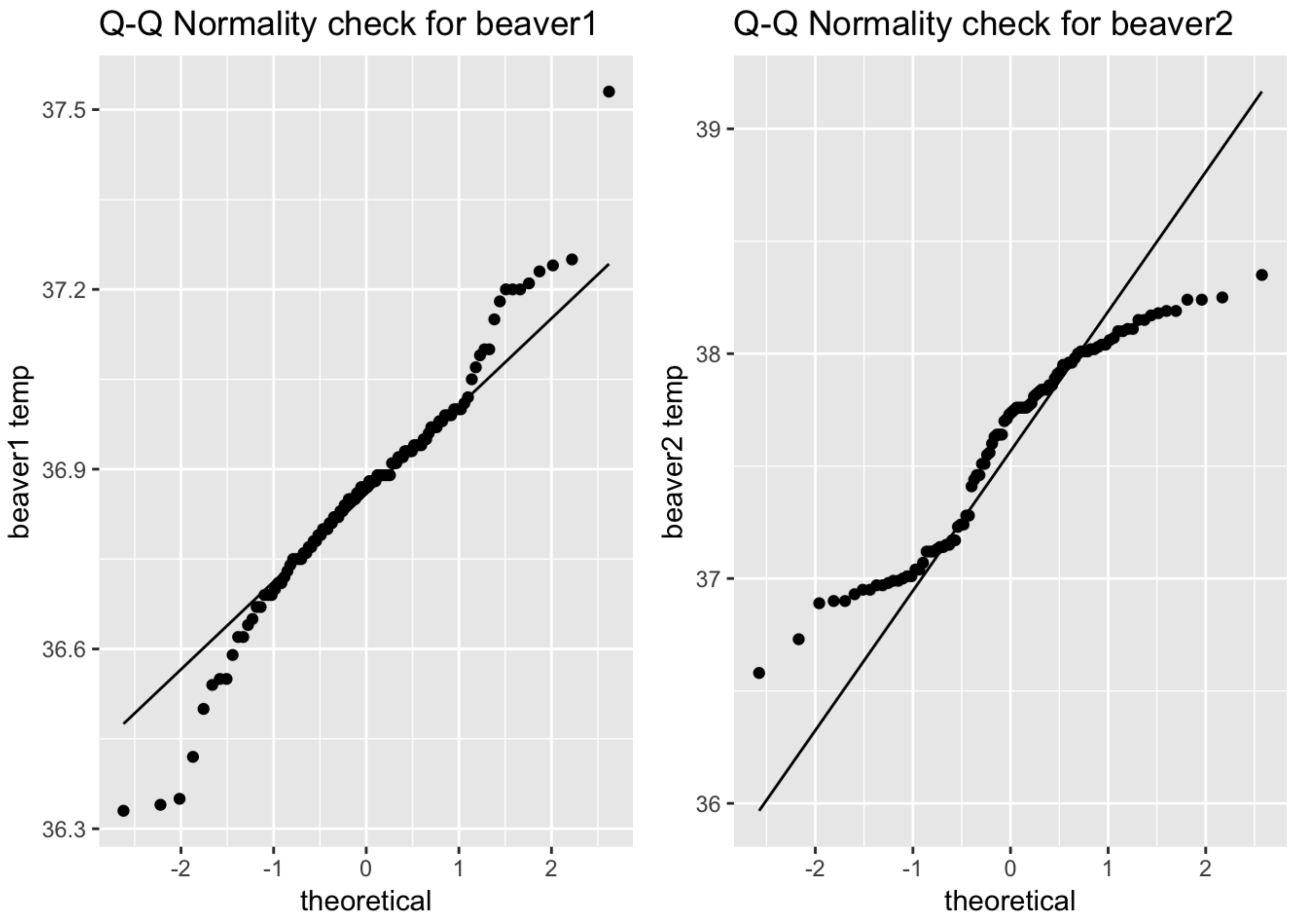
The strategy I take is to subtract the data by a very tiny number so that right-closed, right-opened will not count differently when on the boundary.

4. Beavers

[10 points]

a. Use QQ (quantile-quantile) plots with theoretical normal lines to compare `temp` for the built-in *beaver1* and *beaver2* datasets. Which appears to be more normally distributed?

```
g4a1= ggplot(beaver1, aes(sample = temp)) +
  stat_qq() +
  stat_qq_line() +
  ggtitle("Q-Q Normality check for beaver1") +
  ylab("beaver1 temp")
g4a2 = ggplot(beaver2, aes(sample = temp)) +
  stat_qq() +
  stat_qq_line() +
  ggtitle("Q-Q Normality check for beaver2") +
  ylab("beaver2 temp")
grid.arrange(g4a1,g4a2,ncol = 2)
```



Q-Q Normality check for beaver1

Q-Q Normality check for beaver2

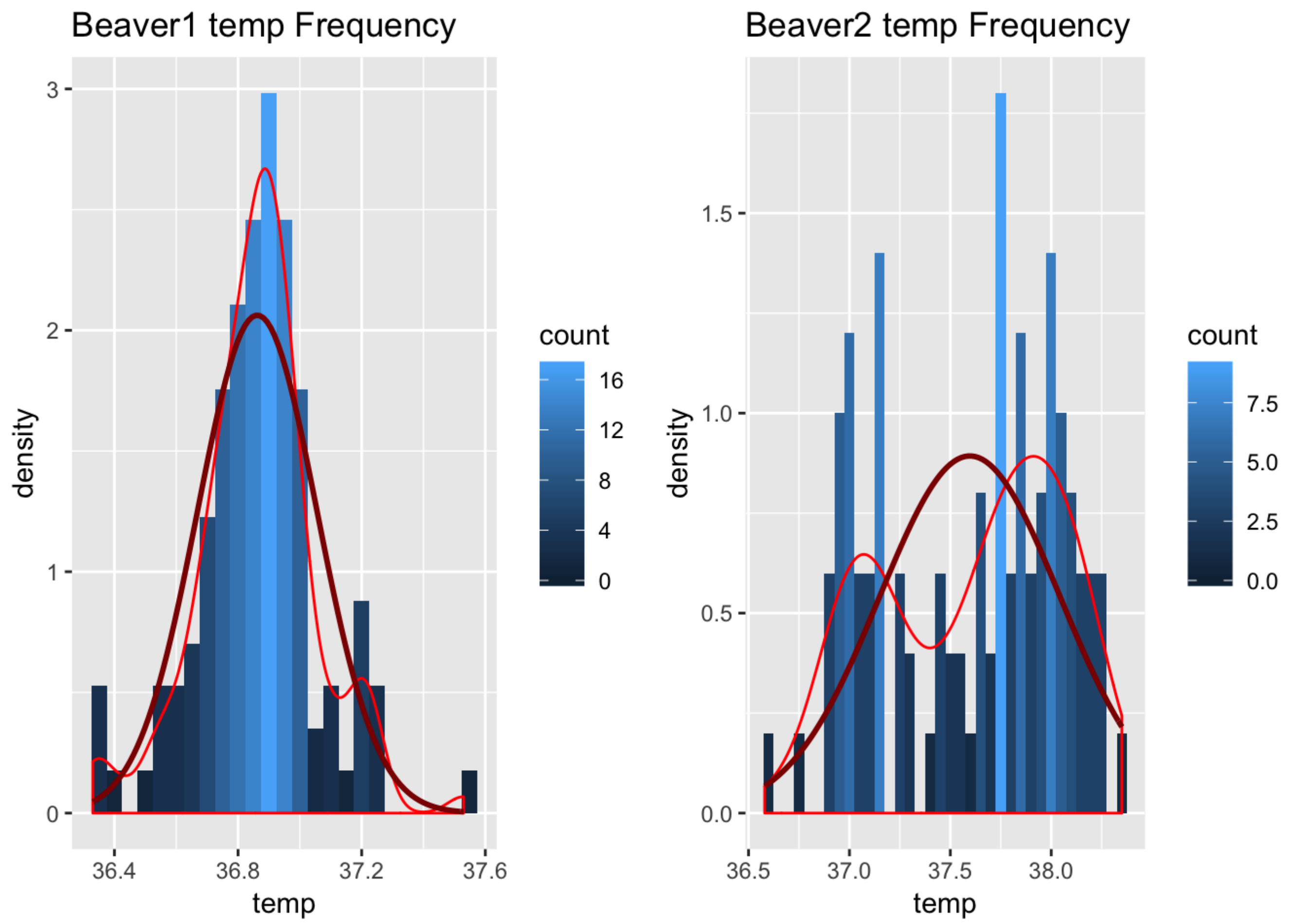
the temp in beaver1 seems more normally distributed becuase there are more points on the normal benmark line. The temp in beaver2 appear to have a curve shape.

b. Draw density histograms with density curves and theoretical normal curves overlaid. Do you get the same results as in part a)?

```
g4b1 = ggplot(beaver1, aes(x = temp)) +
  geom_histogram(binwidth = 0.05,aes(y=..density..,fill=..count..)) +
  geom_density(aes(x = temp), color = "red", alpha = 0.5) +
  stat_function(fun = dnorm, color = "darkred", args = list(mean(beaver1$temp),
  sd(beaver1$temp)),size = 1) +
  ggtitle("Beaver1 temp Frequency")


g4b2 = ggplot(beaver2, aes(x = temp)) +
  geom_histogram(binwidth = 0.05,aes(y=..density..,fill=..count..)) +
  geom_density(aes(x = temp), color = "red", aplha = 0.5) +
  stat_function(fun = dnorm, color = "darkred", args = list(mean(beaver2$temp),
  sd(beaver2$temp)),size = 1) +
  ggtitle("Beaver2 temp Frequency")

grid.arrange(g4b1, g4b2, ncol = 2)
```



The results I get is very similar to part a). We can clearly see from the graph above that the beaver1 temp looks really like a normal distribution, but the beaver2 temp doesn't look like a normal at all

c. Perform the Shapiro-Wilk test for normality using the `shapiro.test()` function. How do the results compare to parts a) and b)?

```
shapiro.test(beaver1$temp)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  beaver1$temp
## W = 0.97031, p-value = 0.01226
```

```
shapiro.test(beaver2$temp)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  beaver2$temp
## W = 0.93336, p-value = 7.764e-05
```

To my surprise, both distribution is not normally distributed. Though, initially I thought the first distribution is normally distributed.

5. Doctors

[5 points]

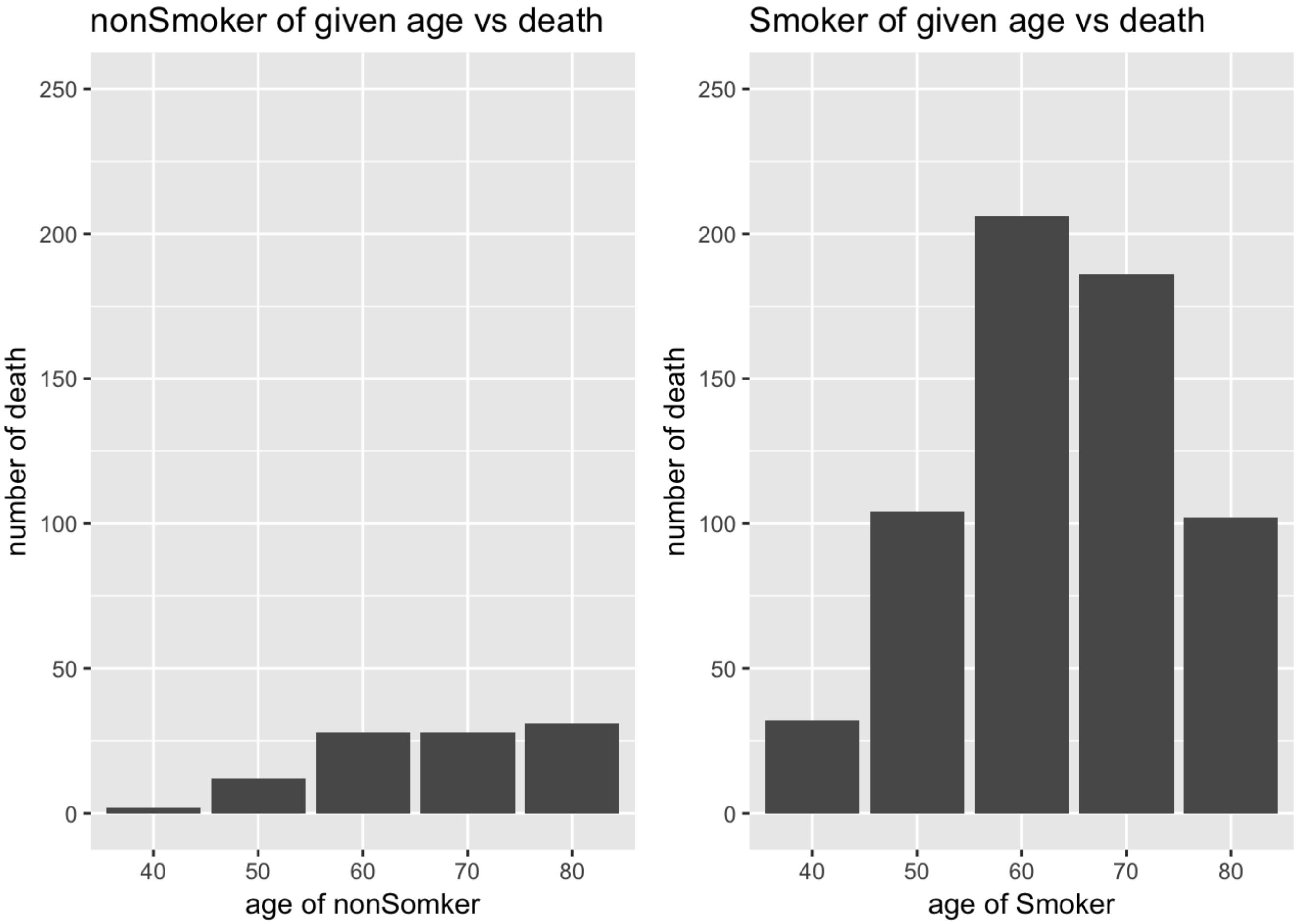
Draw a histogram of the number of deaths attributed to coronary artery disease among doctors in the *breslow* dataset (**boot** package). *(Hint: think carefully about the form in which you are receiving the data.)*

```
require(boot)
breslow0 = breslow[breslow$smoke == 0,]
breslow1 = breslow[breslow$smoke == 1,]

g51 = ggplot(breslow0,aes(breslow0$age,breslow0$y)) +
  geom_col() +
  ylim(0,250) +
  ggtitle("nonSmoker of given age vs death")+
  ylab("number of death") +
  xlab("age of nonSomker")

g52 = ggplot(breslow1,aes(breslow1$age,breslow1$y)) +
  geom_col() +
  ylim(0,250) +
  ggtitle("Smoker of given age vs death") +
  ylab("number of death") +
  xlab("age of Smoker")

grid.arrange(g51, g52, ncol = 2)
```

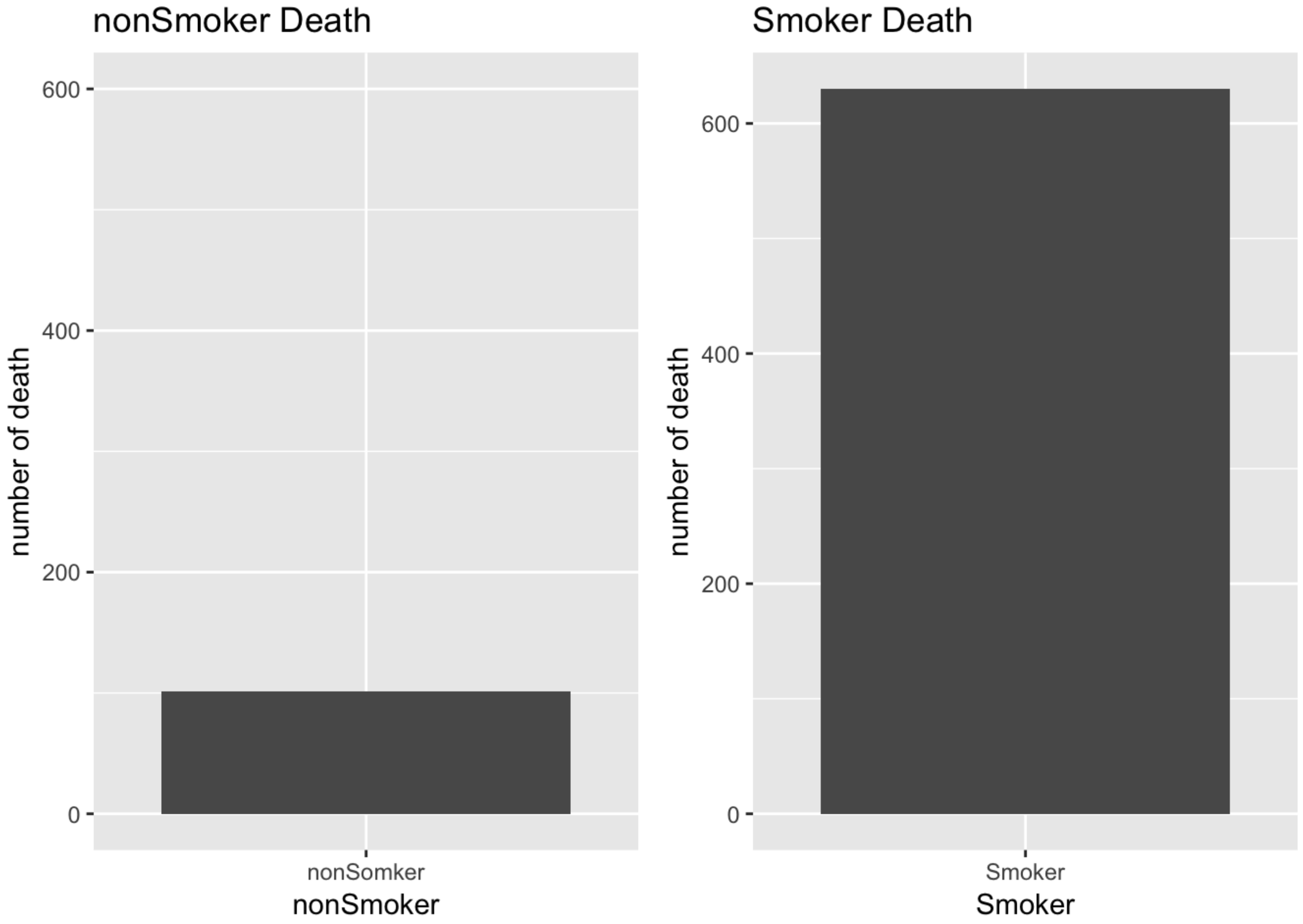


```
a0 = data.frame(category = c("nonSomker"), death = c(sum(breslow0$y)))
a1 = data.frame(category = c("Smoker"), death = c(sum(breslow1$y)))

g53 = ggplot(a0,aes(a0$category,a0$death)) +
  geom_col() +
  ylab("number of death") +
  xlab("nonSmoker") +
  ylim(0,600) +
  ggtitle("nonSmoker Death")


g54 = ggplot(a1,aes(a1$category,a1$death)) +
  geom_col() +
  ylab("number of death") +
  xlab("Smoker") +
  ggtitle("Smoker Death")

grid.arrange(g53,g54, ncol = 2)
```



In case if it is asked for a histogram that is grouped only by smoker vs nonSmoker