Homework 1

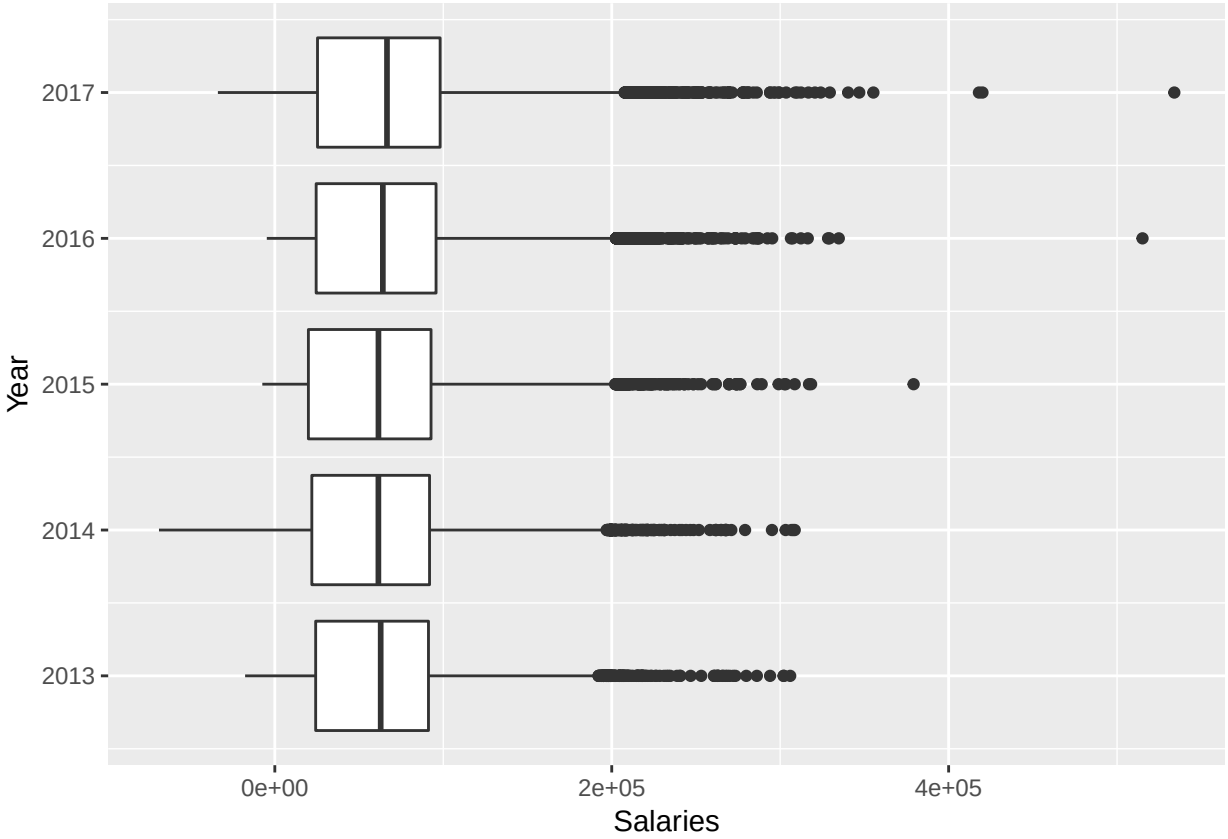
Eric Boxer UNI ecb2198

Sept 22, 2018

1. Salary

a) How do the distributions differ by year?

```
Employee_Compensation <- read_csv("resources/Employee_Compensation.csv")
ggplot(Employee_Compensation,
       mapping = aes(x = Year, y = Salaries, group = Year)) +
  geom_boxplot() +
  coord_flip()
```



From 2013 to 2017, the maximum employee salaries seem to increase. Mean salary and hinge spread look like they may increase slightly. There may be a greater number of high earners, by which I mean those with salaries greater than $200,000, but in a boxplot I cannot be certain about that.

```
Employee_Compensation %>%
  group_by(Year) %>%
  summarise(mean = mean(Salaries),
            iqr = IQR(Salaries),
            range = max(Salaries) - min(Salaries),
            high_earners = sum(Salaries > 200000),
            perc_high_earners = high_earners / n())
```

```
## # A tibble: 5 x 6
##    Year   mean    iqr   range high_earners perc_high_earners
##   <int>  <dbl>  <dbl>   <dbl>        <int>             <dbl>
## 1  2013 62876. 66972. 323577.          105           0.00266
## 2  2014 61914. 69886. 377404.          132           0.00323
## 3  2015 61969. 72799. 386625.          201           0.00467
## 4  2016 64874. 71178. 519872           338           0.00767
## 5  2017 67062. 72792. 567794.          413           0.00904
```

Summary statistics support some of the visual findings. An upward trend in mean salary could be the result of automatic wage increases tied to inflation, but the reason behind an increasing number of high earners is non-obvious. Perhaps the total number of employees is going down or staying constant as employees are promoted / earning greater incentives? Is a certain President really "draining the swamp"?

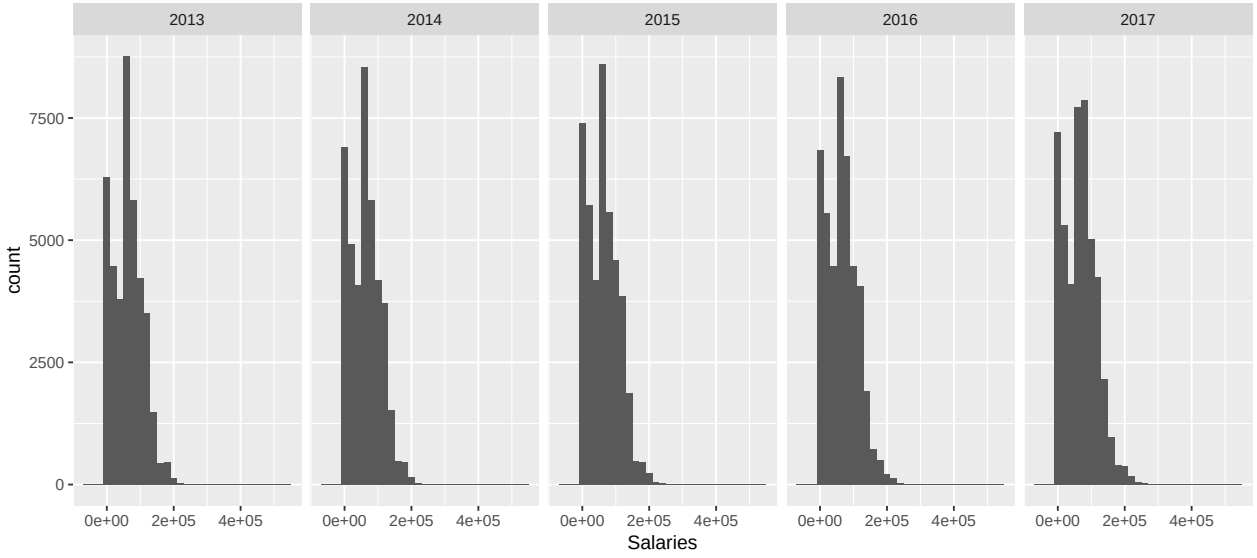
```
Employee_Compensation %>%
  group_by(Year) %>%
  summarise(n = n()) %>%
  mutate(perc_change = n / lag(n))
```

```
## # A tibble: 5 x 3
##    Year     n perc_change
##   <int> <int>       <dbl>
## 1  2013 39476       NA
## 2  2014 40868        1.04
## 3  2015 43078        1.05
## 4  2016 44087        1.02
## 5  2017 45693        1.04
```

Nope, the total number of employees has increased, go figure.

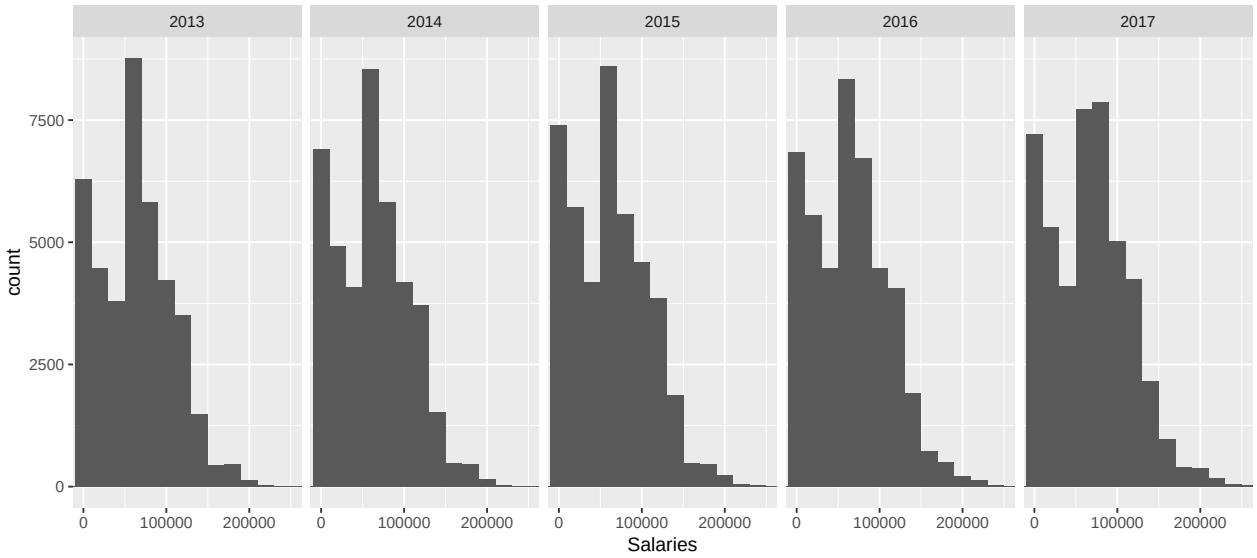
b) What additional information do the histograms provide?

```
ggplot(Employee_Compensation, mapping = aes(x = Salaries)) +
  geom_histogram(binwidth = 20000) +
  facet_wrap(~ Year, nrow = 1)
```




```
#Cleaned-up version with modified binwidth and xlim cutting off nonvisible outliers
ggplot(Employee_Compensation, mapping = aes(x = Salaries)) +
  geom_histogram(binwidth = 20000) +
```

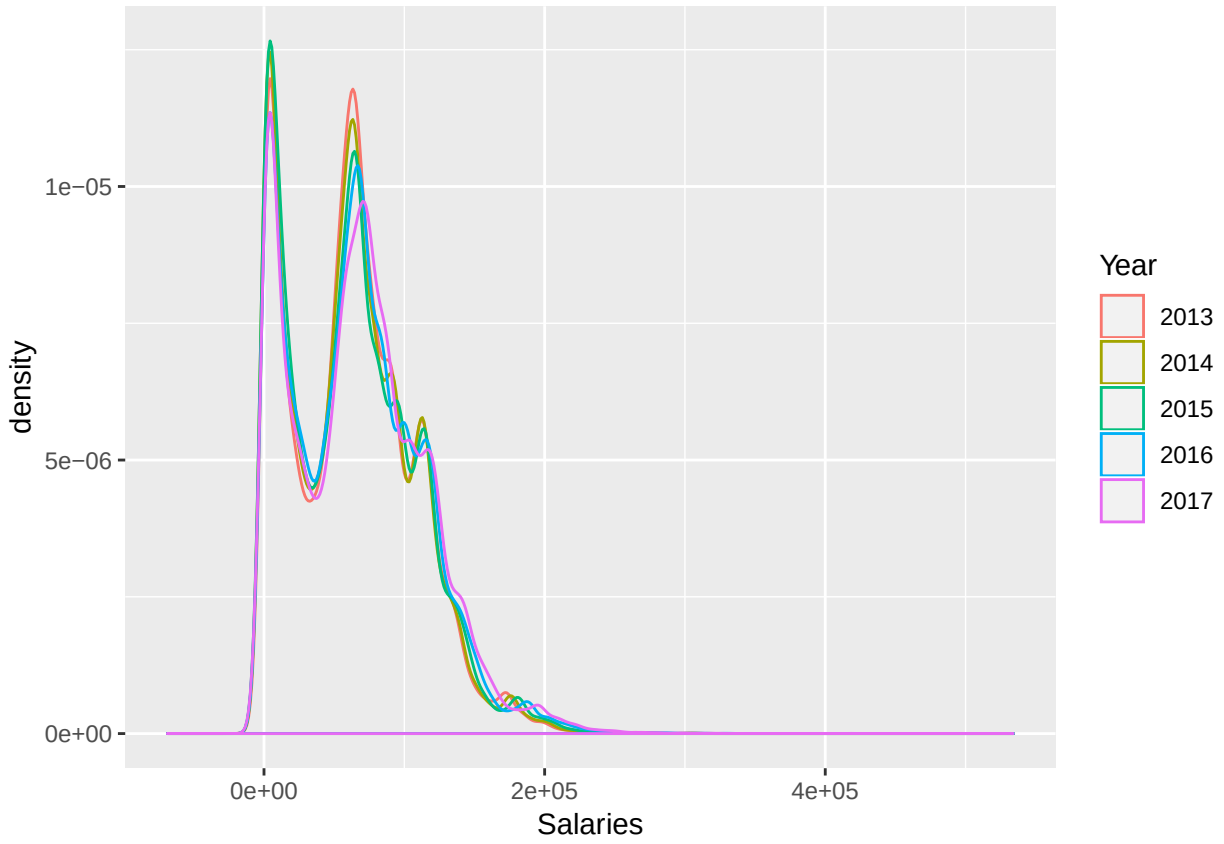
```
  coord_cartesian(xlim = c(0, 250000)) +
  facet_wrap(~ Year, nrow = 1) +
  scale_x_continuous(breaks = c(0, 100000, 200000),
                     labels = c('0', '100000', '200000'))
```



Every year exhibits a bimodal distribution of salaries. From 2013 to 2017, there was an increasing trend in the number of employees earning at the lower peak, less than $20,000, and at the higher peak, between $60,000 and $80,000. Also, there looks to be an increasing number of total employees, because as both peaks increase so too does the count of salaries in the trough between them, at about $40,000.

c) What additional information do you learn?

```
df_Emp_Comp <- data.frame(Employee_Compensation$Salaries)
df_Emp_Comp$Year <- factor(x = Employee_Compensation$Year,
                           levels = c(2013, 2014, 2015, 2016, 2017))
ggplot(df_Emp_Comp, mapping = aes(Employee_Compensation.Salaries, group = Year, color = Year)) +
  geom_density() +
  xlab('Salaries')
```



From the histogram, I could observe that the number of employees earning salaries at each of the peaks was slightly increasing year on year. The density plot shows that the *proportion* of employees earning at both peaks has decreased. We can see that this has happened as the distribution of salaries has become more right-skewed. This reflects that the number of high earners has been increasing, as was first suggested by the boxplot in part a).

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?

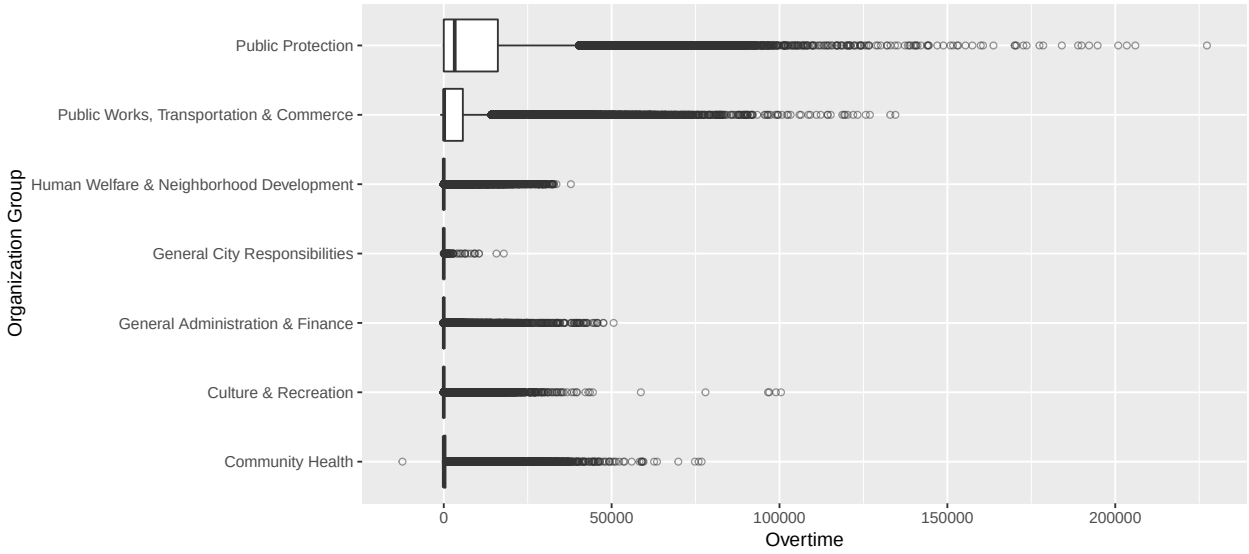
In summary, from 2013 to 2017, from the graphs we have seen that the proportion of employees earning large salaries (defined as greater than $200,000) has increased. As a result, although the number of employees earning in the two most popular salary ranges ([0, 20,000] and [60,000, 80,000]) has increased, the proportion of federal employees earning in those ranges has decreased. The data has developed a greater right-skew.
The boxplot best answers precise questions about the placement of the median and hinges, which have stayed very close over all years. It is also the most explicit in displaying the outlier salaries, but without giving a good sense of the proportion of total salaries that are considered outliers.
The histogram is better at answering questions about the shapes of distributions. From boxplots I had no idea that there was a bimodal distribution.
The density graph also depicts the distribution shape, but in a more approximate, smoothed form. With the density graph it is feasible to overlay each year onto one figure and from there to make a different kind of direct comparison.

2. Overtime

a) Why aren't the boxplots particularly useful?

```
colnames(Employee_Compensation)[4] <- 'Organization_Group'
Employee_Compensation %>%
```

```
  ggplot(mapping = aes(x = reorder(Organization_Group, Overtime, FUN = median), y = Overtime)) +
  geom_boxplot(outlier.shape = 1, outlier.alpha = .5) +
  xlab('Organization Group') +
  coord_flip()
```



The boxplots are not particularly useful because data is concentrated at low values of overtime. The hinge-spreads of the groups with the five lowest median overtime values are all very close to zero. As a result, many of the higher values are treated as outliers and have become indistinguishable, even with a modified marker and alpha.
These boxplots do show explicitly that somebody earned a negative overtime value in Community Health. Also, the Public Protection and Public Works, Transportation & Commerce groups display much greater hinge-spread and have more overtime values over $100,000 than other groups.

```
#Number of negative overtime values
Employee_Compensation %>%
  filter(Overtime < 0) %>%
  count()
```

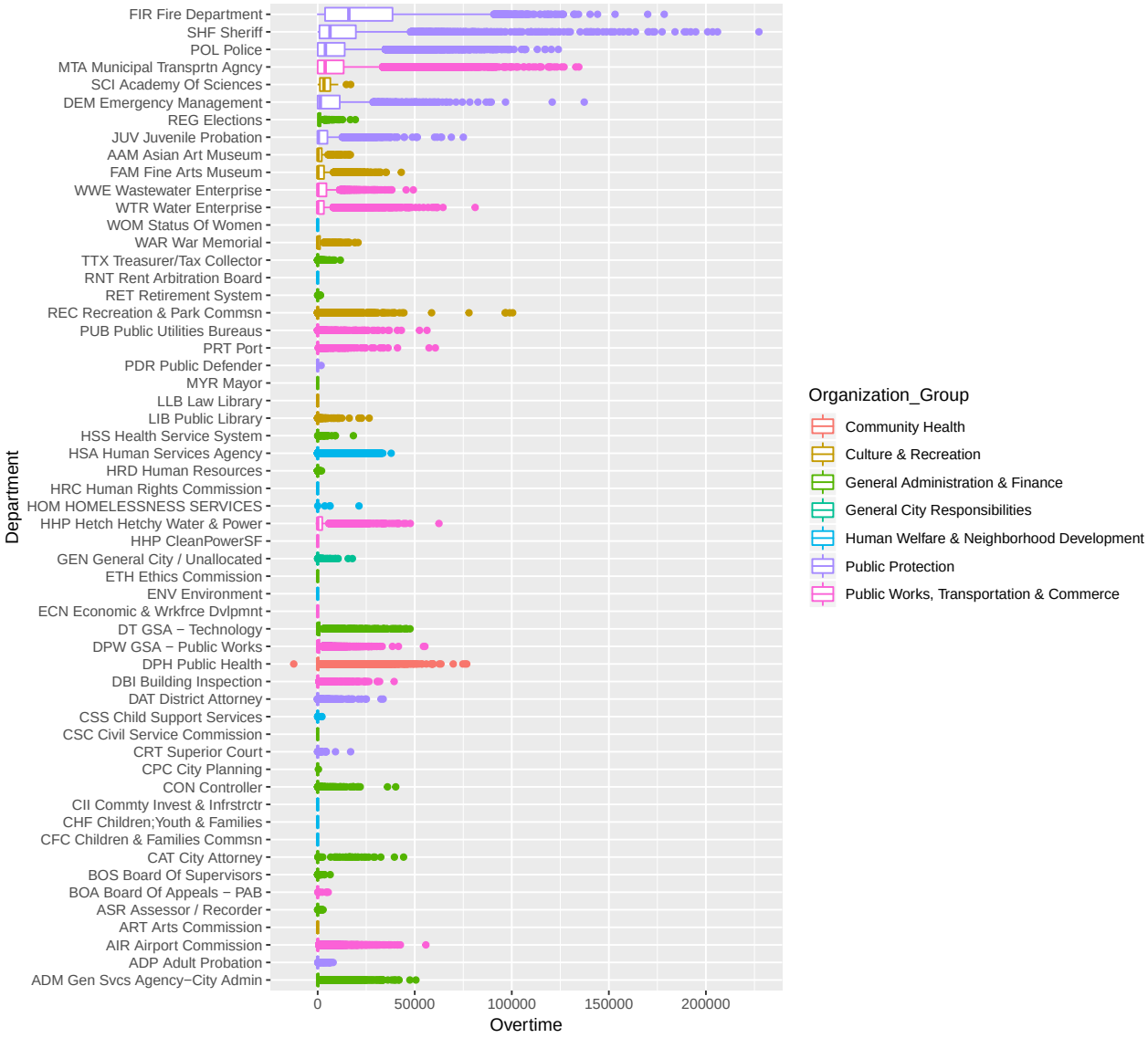
```
## # A tibble: 1 x 1
##       n
##   <int>
## 1    32
```

```
#Which employees earned the most negative overtime value?
Employee_Compensation %>%
  filter(Overtime < -500) %>%
  select(Organization_Group, Department, Overtime)
```

```
## # A tibble: 3 x 3
##   Organization_Group                  Department                Overtime
##   <chr>                               <chr>                        <dbl>
## 1 Community Health                    DPH Public Health            -612.
## 2 Community Health                    DPH Public Health          -12309.
## 3 Public Works, Transportation & Comm~ HHP Hetch Hetchy Water & ~   -1073.
```

b) Explain how this form improves on the plots in part a).

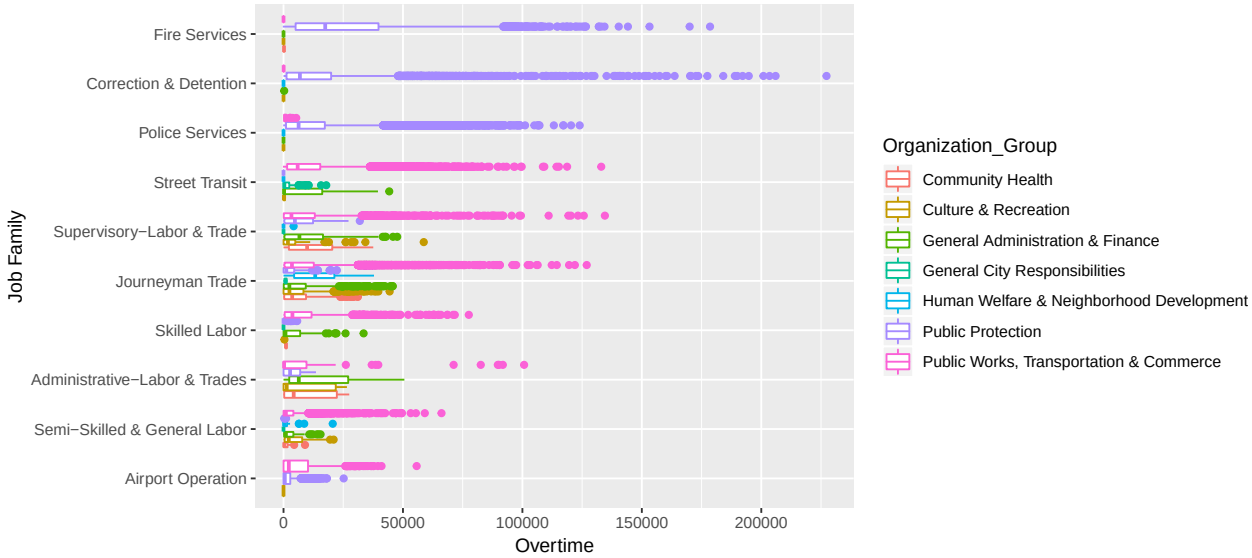
```
Employee_Compensation %>%
  group_by(Organization_Group, Department) %>%
  ggplot(mapping = aes(x = reorder(Department, Overtime, FUN = median),
                       y = Overtime,
                       color = Organization_Group)) +
  geom_boxplot() +
  xlab('Department') +
  coord_flip()
```



Now we can see that most of the high Overtime earners are in emergency response departments (Fire, Sheriff, Police) and in utilities department (MTA, Water, Public Health, Parks). This could be rationalized by considering that these are workers who are more likely to be on call for time-sensitive problems and are earning more overtime as a result.

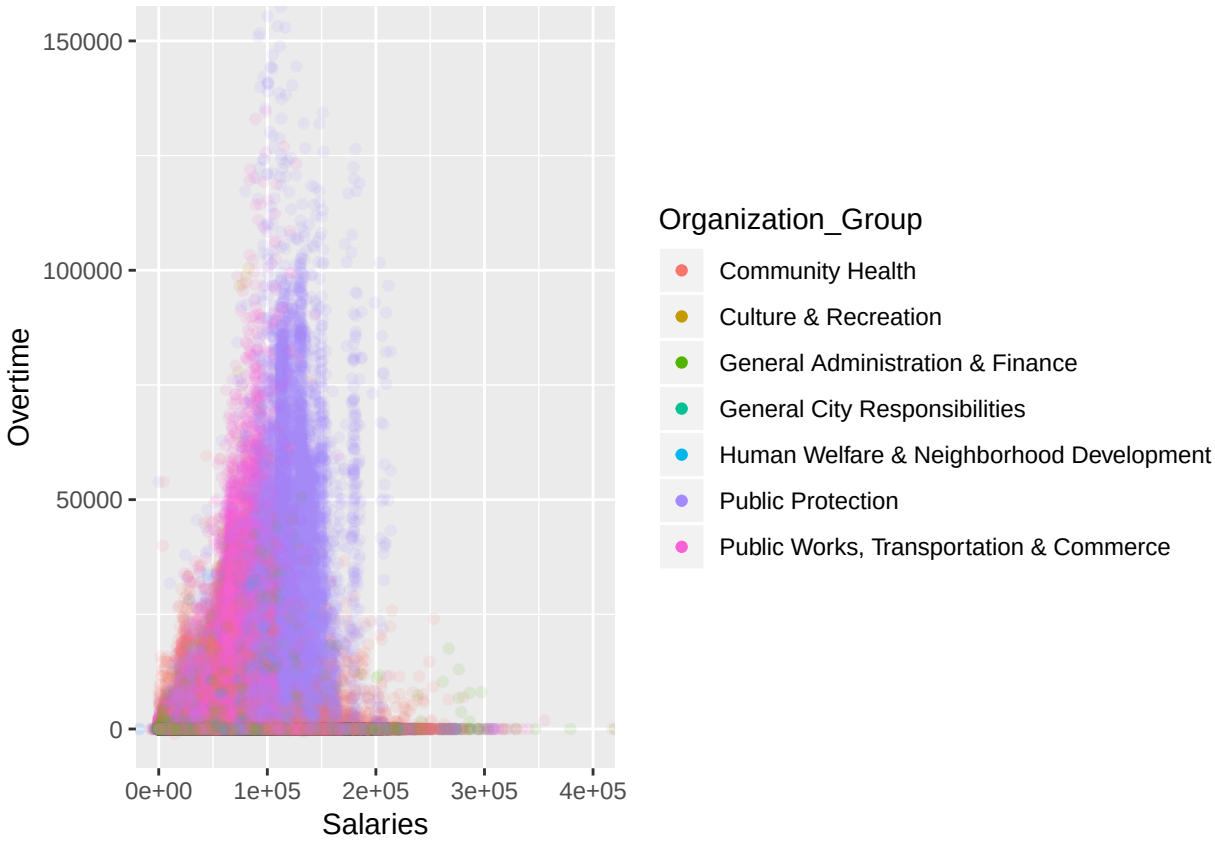
```
colnames(Employee_Compensation)[10] <- 'Job_Family'
df_Overtime <- Employee_Compensation %>%
  group_by(Job_Family) %>%
  summarise(med_Overtime = median(Overtime)) %>%
  arrange(desc(med_Overtime))
```

```
top_Overtime_Jobs <- df_Overtime$Job_Family[1:10]
Employee_Compensation %>%
  group_by(Organization_Group, Job_Family) %>%
  filter(Job_Family %in% top_Overtime_Jobs) %>%
  ggplot(mapping = aes(x = reorder(Job_Family, Overtime, FUN = median), y = Overtime, color = Organizati
  geom_boxplot() +
  xlab('Job Family') +
  coord_flip()
```



Public Protection and Public Works, Transportation & Commerce group employees earn the majority of overtime pay in the top 10 job families. Now we can clearly see that workers in jobs categorized as 'Correction and Detention' are also earning high Overtime pay. Maybe this could motivate an exploration of understaffed prisons. Are corrections officers being compelled to work overtime in lieu of proper staffing?

```
Employee_Compensation %>%
  ggplot(mapping = aes(x = Salaries,
                       y = Overtime,
                       color = Organization_Group)) +
  geom_point(alpha = .1) +
  coord_cartesian(xlim = c(-1000, 400000),
                  ylim = c(-1000, 150000)) +
  guides(color = guide_legend(override.aes = list(alpha = 1)))
```



The employees earning high overtime pay are not earning the most, but many are earning six-figure salaries in addition to overtime pay. This would serve as evidence against an explanation that the high overtime earners are receiving this compensation in exchange for low salaries.

3. Boundaries

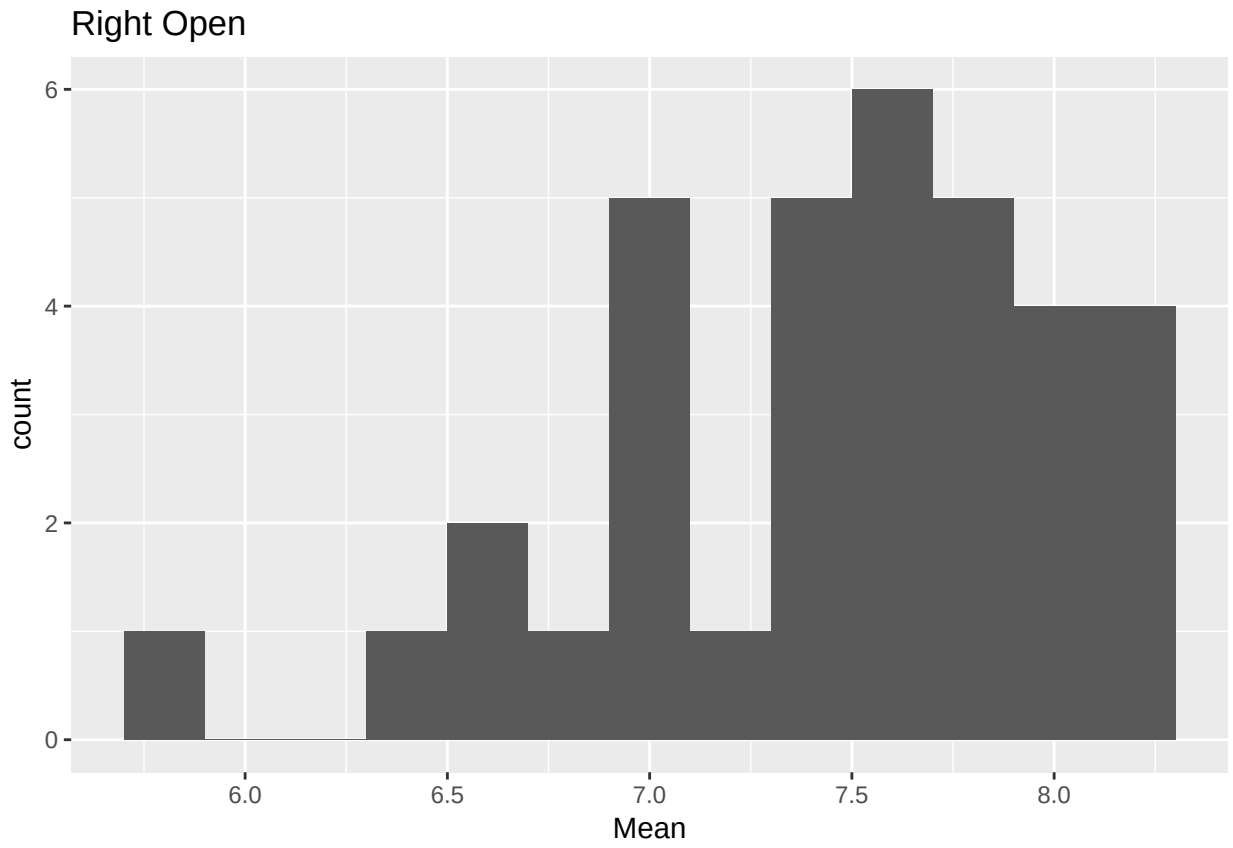
a) Display the full dataset (that is, show the numbers) and the plots of the two forms.

```
happiness <- read_csv('resources/happiness.csv')
happiness$Country <- na.locf(happiness$Country)
happiness %>%
  filter(Gender == 'Both') %>%
  print(n = 35)
```

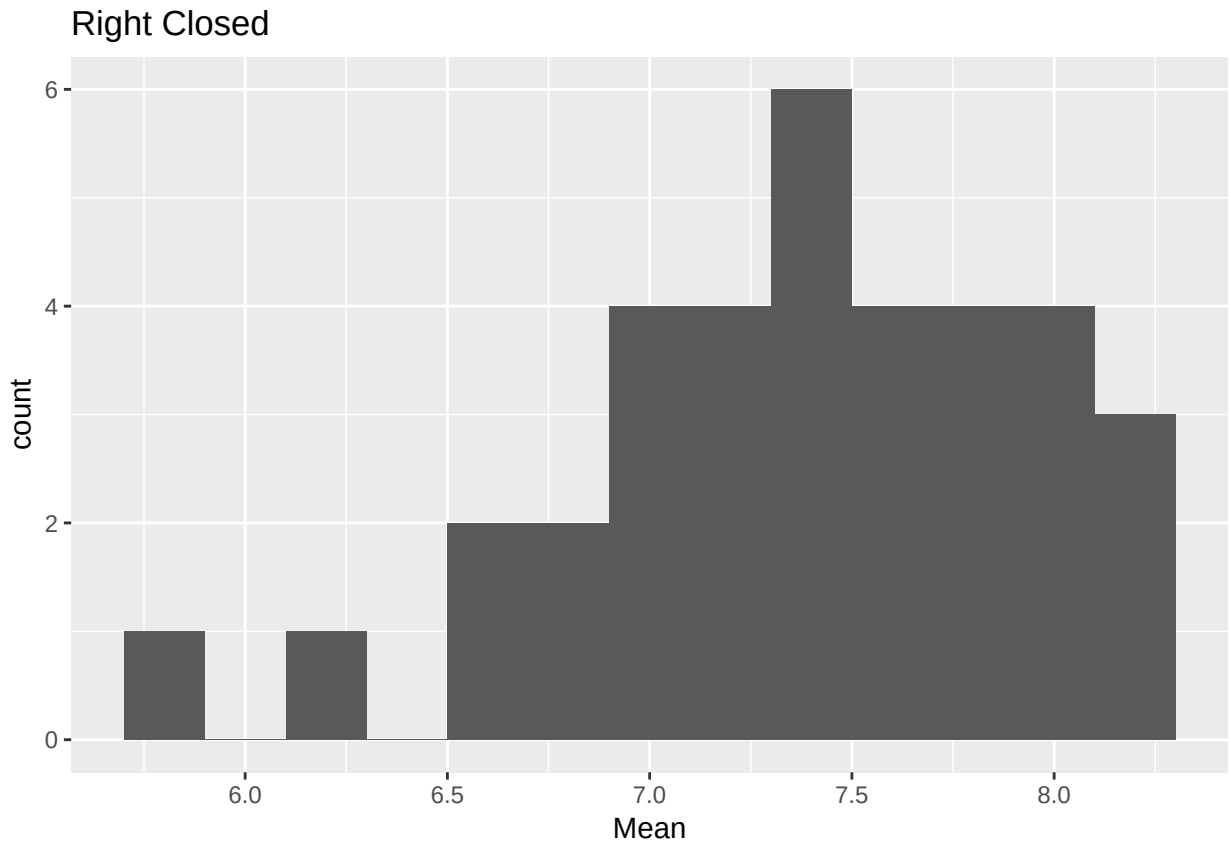
```
## # A tibble: 35 x 4
##    Country Gender  Mean  `N=`
##    <chr>   <chr>  <dbl> <int>
##  1 AT      Both     7.3  1041
##  2 BE      Both     7.8  1010
##  3 BG      Both     5.8   971
##  4 CY      Both     7.7  1002
##  5 CZ      Both     7.5  1221
##  6 DE      Both     7.5  2003
##  7 DK      Both     8.3   997
##  8 EE      Both     7.4  1017
##  9 GR      Both     7.3  1000
## 10 ES      Both     7.6  1013
```

```
## 11 FI      Both     8.3  1001
## 12 FR      Both     7.8  1529
## 13 HR      Both     7     982
## 14 HU      Both     7     998
## 15 IE      Both     8     996
## 16 IT      Both     7    1506
## 17 LT      Both     7.3   997
## 18 LU      Both     8    1001
## 19 LV      Both     6.8   990
## 20 MK      Both     6.3   980
## 21 MT      Both     7.9   991
## 22 NL      Both     8    1010
## 23 NO      Both     8.1   992
## 24 PL      Both     7.4  1488
## 25 PT      Both     6.9   998
## 26 RO      Both     7     997
## 27 SE      Both     8.2  1011
## 28 SI      Both     7.7  1028
## 29 SK      Both     7.5  1119
## 30 TR      Both     6.6  1992
## 31 UK      Both     7.8  1499
## 32 CC3     Both     6.6  3954
## 33 EU 15   Both     7.6 17615
## 34 NMS12   Both     7.2 12819
## 35 EU 27   Both     7.5 30434
```

```
happiness %>%
  filter(Gender == 'Both') %>%
  ggplot(mapping = aes(x = Mean)) +
  geom_histogram(binwidth = .2, closed = 'left') +
  ggtitle('Right Open')
```

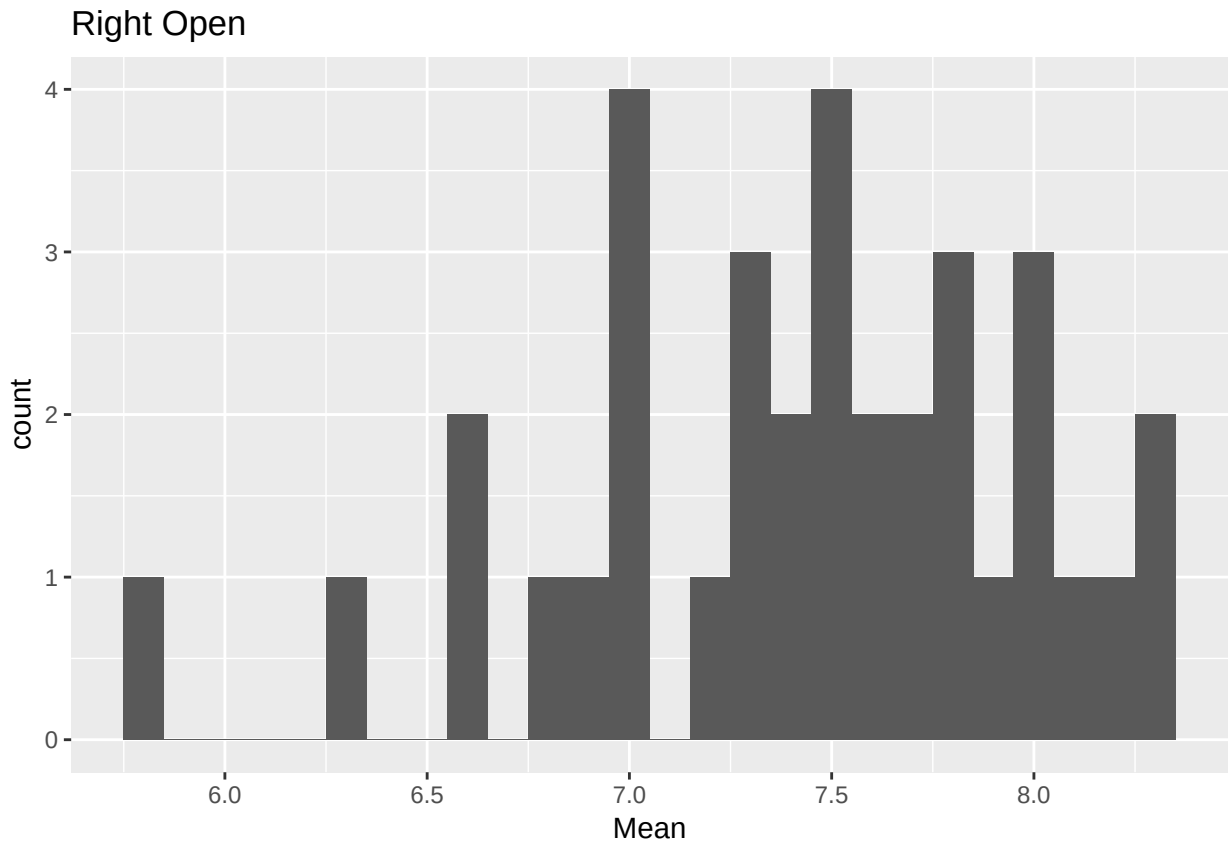



```
happiness %>%
  filter(Gender == 'Both') %>%
  ggplot(mapping = aes(x = Mean)) +
  geom_histogram(binwidth = .2, closed = 'right') +
  ggtitle('Right Closed')
```

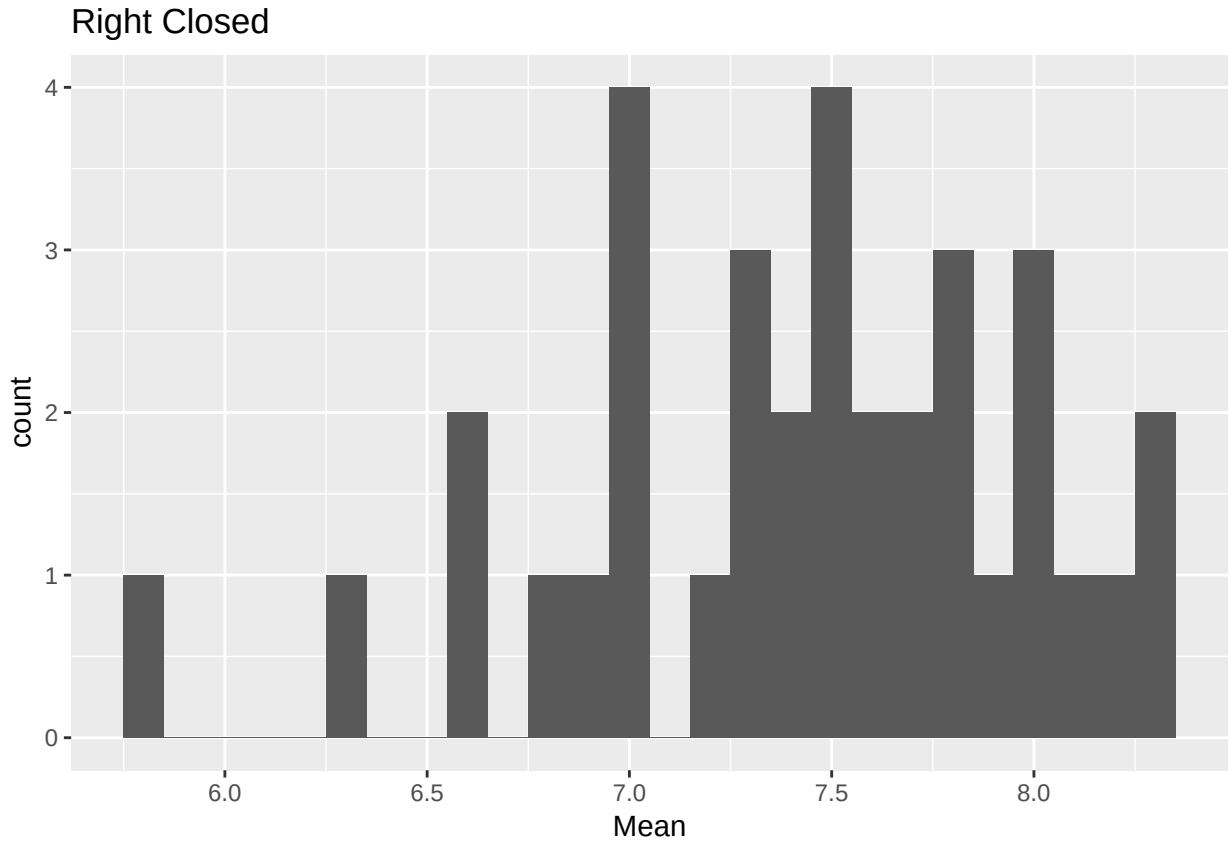


b) Adjust parameters–the same for both–so that the right open and right closed versions become identical. Explain your strategy.

```
happiness %>%
  filter(Gender == 'Both') %>%
  ggplot(mapping = aes(x = Mean)) +
  geom_histogram(binwidth = .1, closed = 'left') +
  ggtitle('Right Open')
```




```
happiness %>%
  filter(Gender == 'Both') %>%
  ggplot(mapping = aes(x = Mean)) +
  geom_histogram(binwidth = .1, closed = 'right') +
  ggtitle('Right Closed')
```

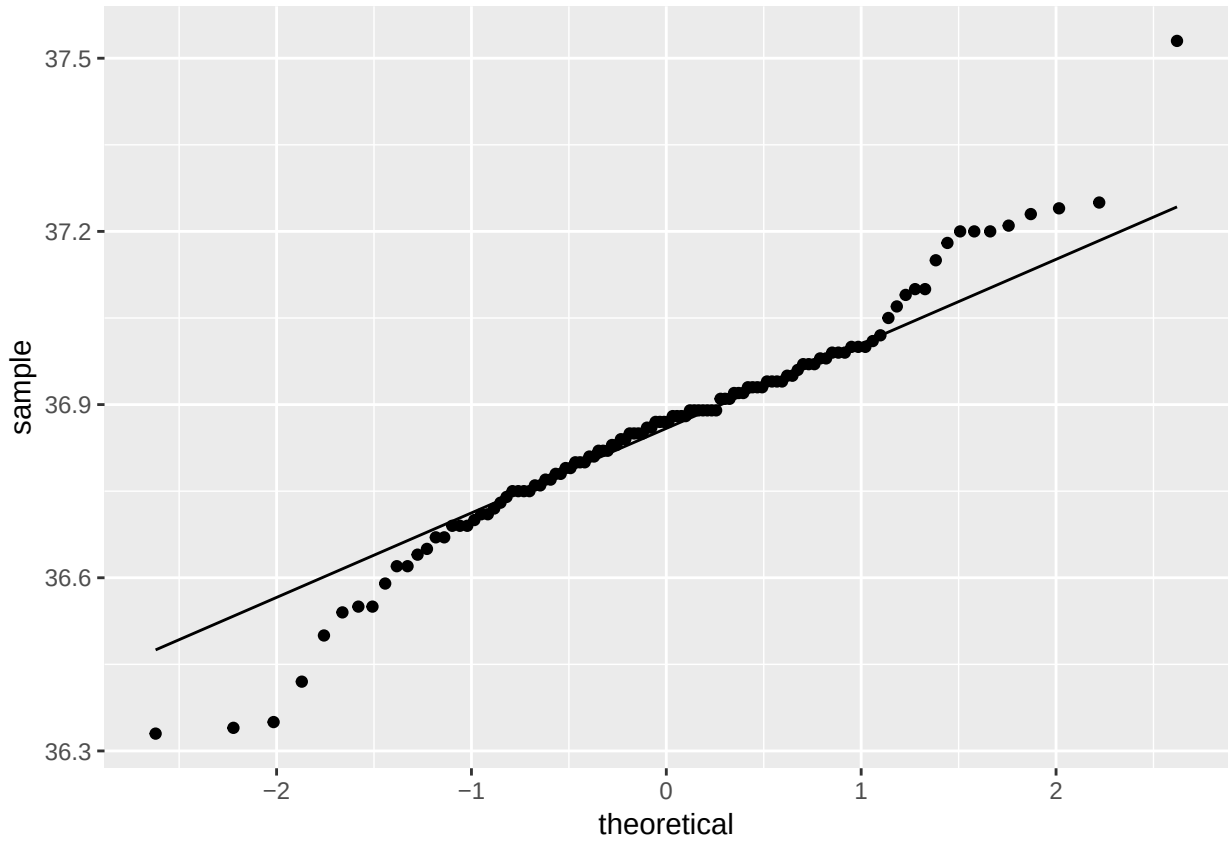


Happiness has been measured in tenths. If the center of each bin is at the default of 0, an example binwidth of .2 will look like (0.1, 0.3] or [0.1, 0.3), depending on boundary conditions. Both will count different numbers of observations if there are an unequal number of observations of 0.1 and 0.3. As a result of the measurement format, if the boundaries of the bins are equal to the cutoffs between values, then the boundaries will matter. For example, geom_histogram(center = 0.5, binwidth = .1) will also display differently depending on the boundary condition used, because the bins will have boundaries in the tenths place and values will be counted or ignored depending on the conditions. To avoid measurements coinciding with bin boundaries we can select a combination of center and binwidth like center = 0, binwidth = .1. This histogram has example bins (0.05, 0.15] or [0.05, 0.15), depending on boundary conditions, and the condition becomes inconsequential because there are no obervations at five-hundredths of a unit precision and ending in .05.

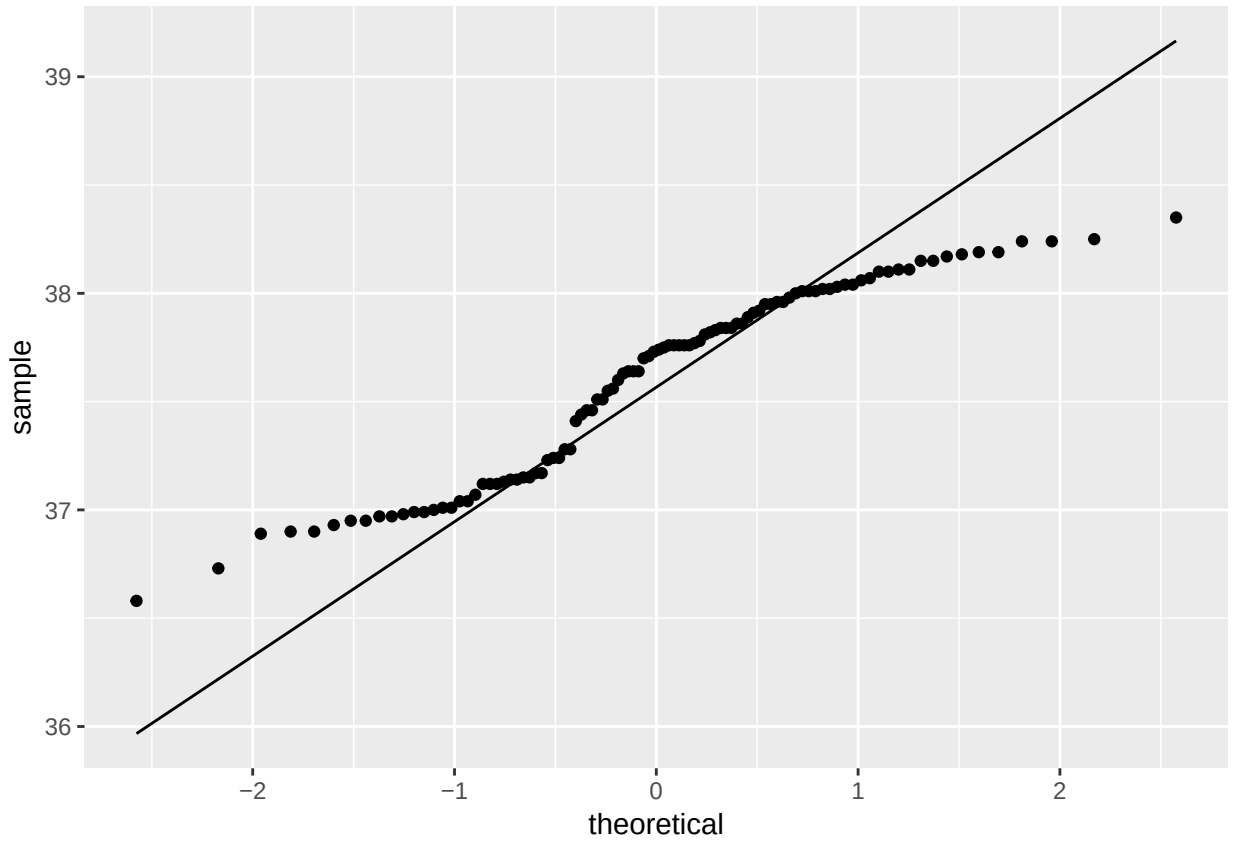
4. Beavers

a) Which appears to be more normally distributed?

```
ggplot(beaver1, aes(sample = temp)) +
  stat_qq() +
  stat_qq_line()
```



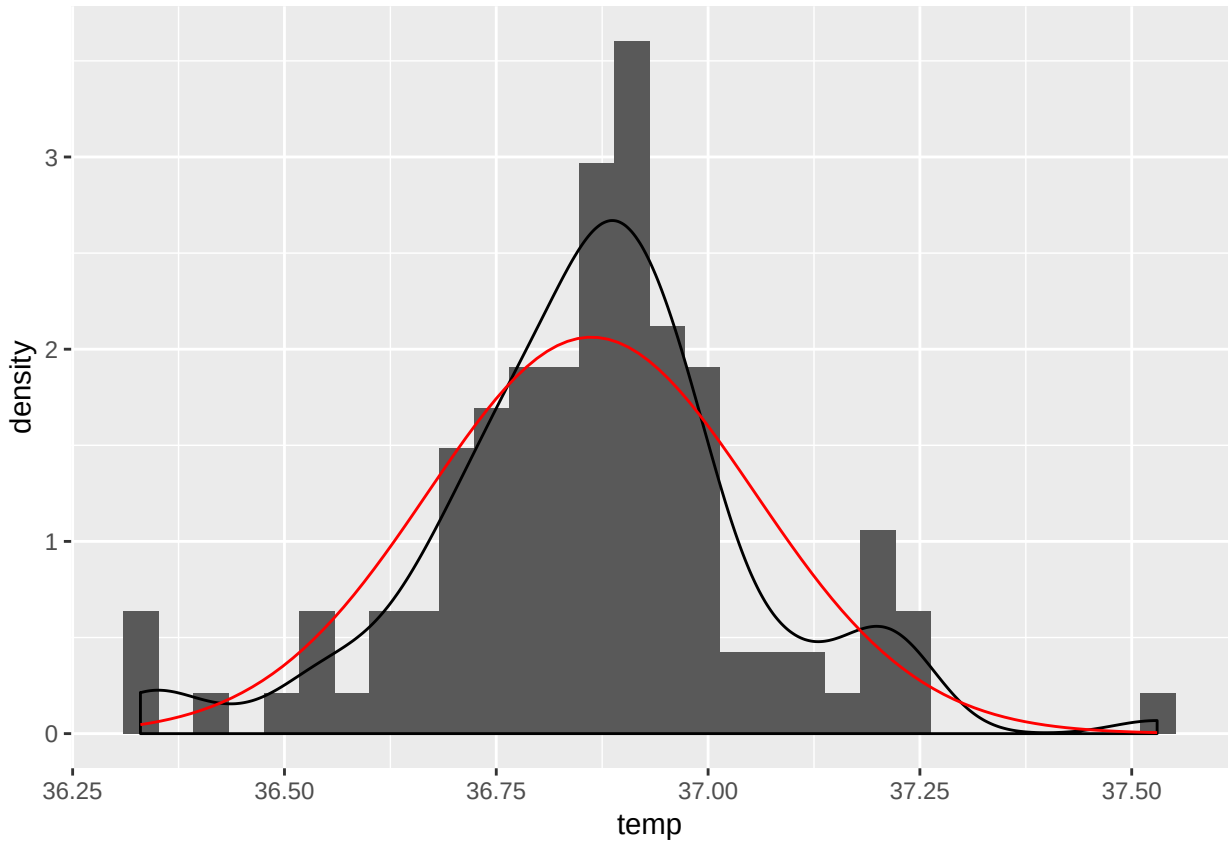

```
ggplot(beaver2, aes(sample = temp)) +
  stat_qq() +
  stat_qq_line()
```



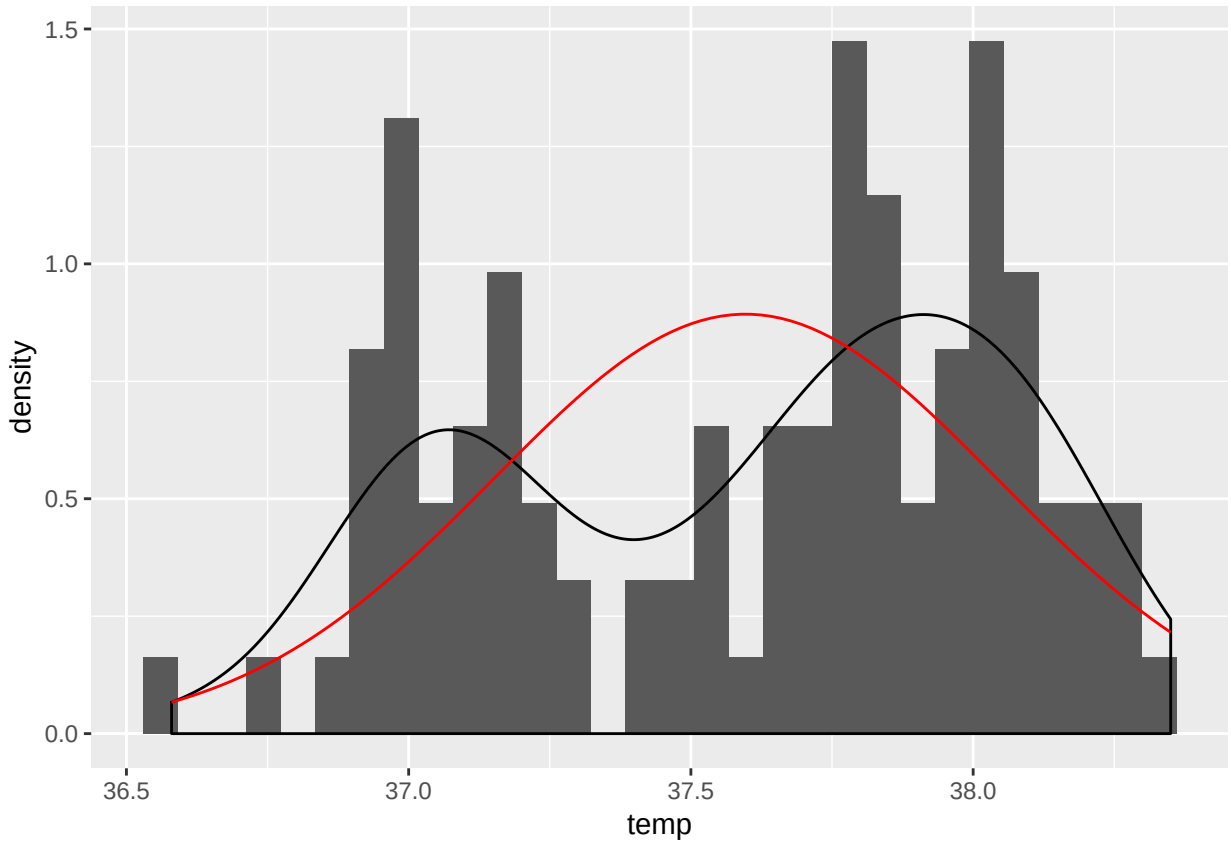
I think that beaver1 is the more normally distributed dataset. Compared to beaver2, beaver1 has middle quantiles which are very close to the theoretical normal line. On the other hand, the quantiles of beaver1 at the left extreme may be further from theoretical normal than that of beaver2.

b) Do you get the same results as in part a)?

```
ggplot(beaver1) +
  geom_histogram(mapping = aes(x = temp, y = ..density..)) +
  geom_density(mapping = aes(temp)) +
  stat_function(fun = dnorm,
                color = 'red',
                args = list(mean = mean(beaver1$temp),
                            sd = sd(beaver1$temp)))
```




```
ggplot(beaver2) +
  geom_histogram(mapping = aes(x = temp, y = ..density..)) +
  geom_density(mapping = aes(temp)) +
  stat_function(fun = dnorm,
                color = 'red',
                args = list(mean = mean(beaver2$temp),
                            sd = sd(beaver2$temp)))
```



The beaver1 data looks to be more normally distributed because it is unimodal about the mean. There are some deviations near the tails, like a bump at about 37.2. The beaver2 distribution is bimodal, with few observations near the mean of the theoretical normal curve. In comparison to beaver1, beaver2 is further from a normal distribution.

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)
```

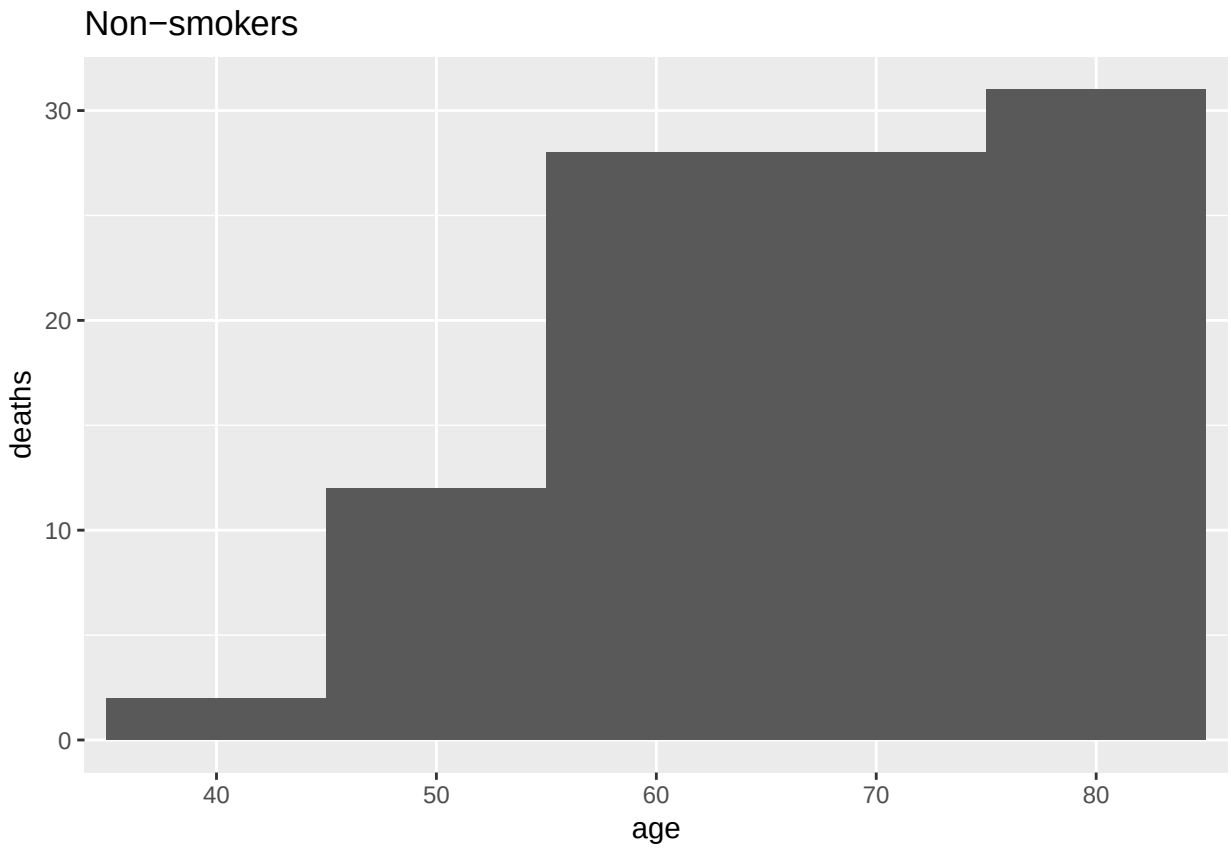
```
##
##  Shapiro-Wilk normality test
##
## data:  beaver2$temp
## W = 0.93336, p-value = 7.764e-05
```

The Shapiro-Wilk test p-value for beaver1 is greater than 0.1 so we cannot reject the null hypothesis that beaver1 is normally distributed. For beaver2, the p-value is approximately 0.00008, less than a reasonable alpha = 0.01 so we can reject the null hypothesis and find that beaver2 does not have a normal distribution.

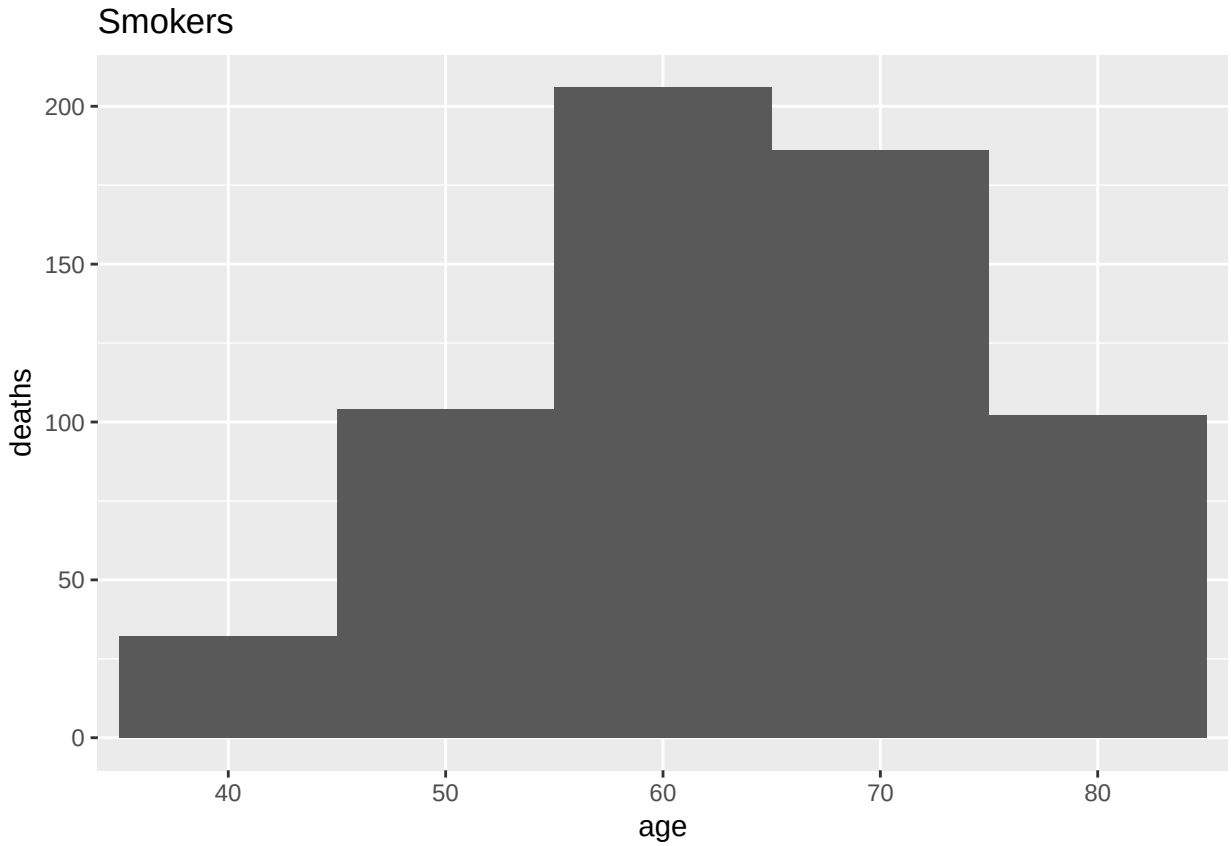
5. Doctors

Draw a histogram of the number of deaths attributed to coronary artery disease among doctors in the *breslow* dataset (**boot** package).

```
breslow %>%
  filter(smoke == 0) %>%
  ggplot(mapping = aes(x = age, y = y)) +
  geom_bar(width = 1, stat = 'identity') +
  ylab('deaths') +
  ggtitle('Non-smokers')
```




```
breslow %>%
  filter(smoke == 1) %>%
  ggplot(mapping = aes(x = age, y = y)) +
  geom_bar(width = 1, stat = 'identity') +
  ylab('deaths') +
  ggtitle('Smokers')
```



The y feature is a count of the number of deaths from coronary artery disease in the age range centered around the age feature. A bar graph of age against y is equivalent to a frequency graph of deaths.