FS21 STT 180 Homework 2

Kaitlyn Watson

Oct 17-Oct 30, 2021

Setting up:

Load `tidyverse`, which includes `dplyr`, `ggplot2`, `tidyr`, and other packages, and the load ‘knitr.

```
library(tidyverse)
library(knitr)
```

Homework 2 has two sections. In Section 1 you will use data visualization and write function to analyze a dataset.For Section 2 you will read an article, explore the data, validate the claims and come to own conclusions.

General Instructions:

- This is an individual assignment. You may consult with others as you work on the assignment, but each student should write up a separate set of solutions.
- Rather than creating a new Rmd file, just add your solutions to the supplied Rmd file. **Submit both the Rmd file and the resulting HTML file to D2L**. Please note if you are compiling to a pdf, change the option in the YAML and also delete the .css lines in the file.
- Except for questions, or parts of questions, that ask for your commentary, use R in a code chunk to answer the questions.
- The code chunk option `echo = TRUE` is specified in the setup code chunk at the beginning of the document. Please do not override this in your code chunks.
- A solution will lose points if the Rmd file does not compile. If one of your code chunks is causing your Rmd file to not compile, you can use the `eval = FALSE` option. Another possibility is to use the `error = TRUE` option in the code chunk.
- This Homework is due on **Saturday, October 30th, 2021 on or before 11 pm.**

Section 1

For the first section of this homework will use the same birth dataset you used for Homework 1. Please use the `BirthDataWithRegionColors.csv` file for this HW. The dataset contains information about births in the United States. The full data set is from the Centers for Disease Control. The data for this homework assignment is a “small” sample (chosen at random) of slightly over one million records from the full data set. The data for this homework assignment also only contain a subset of the variables in the full data set.

Introduction

Read in the data and convert the data frame to a tibble.

```
birth_data <- read.csv("BirthDataWithRegionColors.csv", header = TRUE)
birth_data <- as_tibble(birth_data)
```

A glimpse of the data:

```
glimpse(birth_data)
```

```
Rows: 1,048,575
Columns: 10
$ year          <int> 1969, 1969, 1969, 1969, 1969, 1969, 1969, 1969, 1969, 19~
$ month         <int> 2, 3, 5, 5, 5, 6, 8, 8, 11, 11, 11, 3, 3, 6, 7, 10, 3, 8~
$ state         <chr> "CA", "CA", "CA", "CA", "CA", "CA", "CA", "CA", "CA", "C~
$ is_male       <lgl> TRUE, FALSE, TRUE, FALSE, TRUE, TRUE, FALSE, FALSE, FALS~
$ weight_pounds <dbl> 6.999677, 6.876218, 7.187070, 7.749249, 7.374463, 9.6253~
$ mother_age    <int> 20, 25, 30, 17, 22, 26, 26, 19, 25, 26, 26, 31, 28, 24, ~
$ child_race    <int> 1, 2, 1, 1, 4, 2, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 2, 2,~
$ plurality     <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ~
$ region        <chr> "West", "West", "West", "West", "West", "West", "West", ~
$ state_color   <chr> "blue", "blue", "blue", "blue", "blue", "blue", "blue", ~
```

The variables in the data set are:

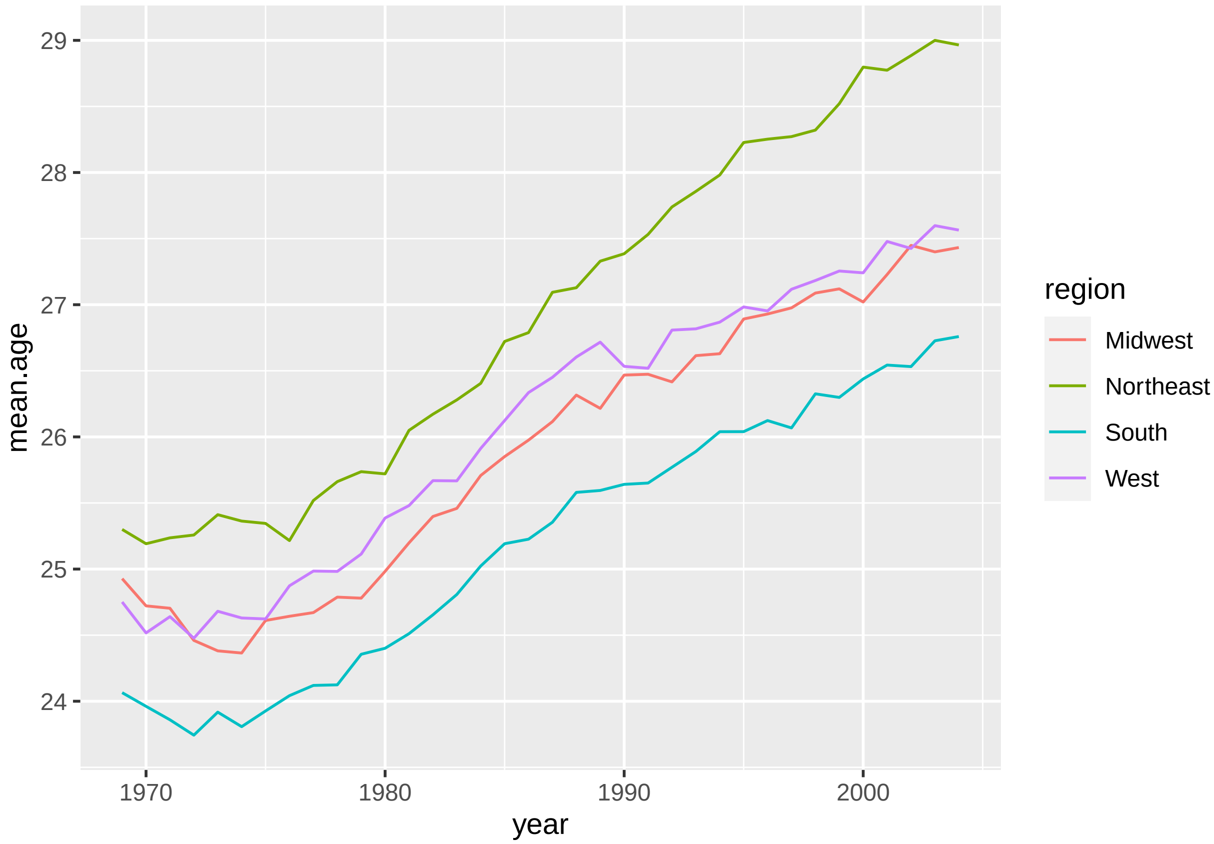
Variable	Description
`year`	the year of the birth
`month`	the month of the birth
`state`	the state where the birth occurred, including “DC” for Washington D.C.
`is_male`	which is `TRUE` if the child is male, `FALSE` otherwise
`weight_pounds`	the child’s birth weight in pounds
`mother_age`	the age of the mother
`child_race`	race of the child.
`plurality`	the number of children born as a result of the pregnancy, with 1 representing a single birth, 2 representing twins, etc.

Combine `dplyr`with `ggplot2` to create graphical displays of the data. Use `filter`, `group_by`, and `summarize` build the required data frame.

Question 1

Create a plot of mean age of mother versus year, which includes separate lines for each of the four regions of the country. (Don’t include data for which the region is missing.) The graphic should look like the following.

```
birth_data%>%
 filter(!is.na(region)) %>%
 group_by(region, year) %>%
summarise(mean.age = mean(mother_age)) %>%
ggplot(mapping=aes(x = year , y=mean.age))+
 geom_line(mapping=aes(color=region))
```

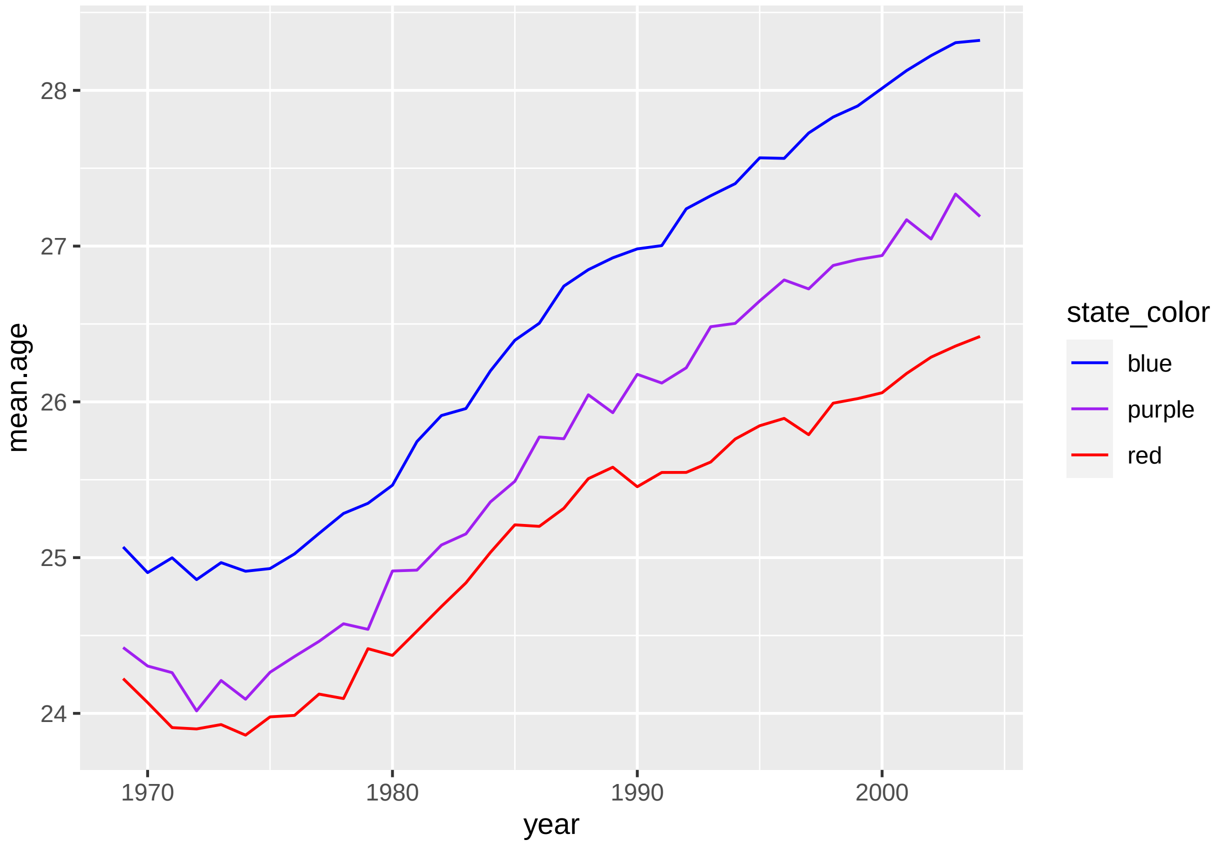


Referenced: Stack Overflow https://stackoverflow.com/questions/17216358/eliminating-nas-from-a-ggplot Accessed on October 27, 2021.

Question 2

Create a graphic of mean age of mother versus year, which includes separate lines for each of the three values of `state_color`. (Don't include data for which `state_color` is missing.) The graphic should look like the following. Notice that the colors are different from the default colors.

```
birth_data%>%
 filter(!is.na(state_color)) %>%
 group_by(state_color, year) %>%
summarise(mean.age = mean(mother_age)) %>%
ggplot(mapping=aes(x = year , y=mean.age))+
 geom_line(mapping=aes(color=state_color))+
 scale_color_manual(values=c("blue", "purple", "red"))
```



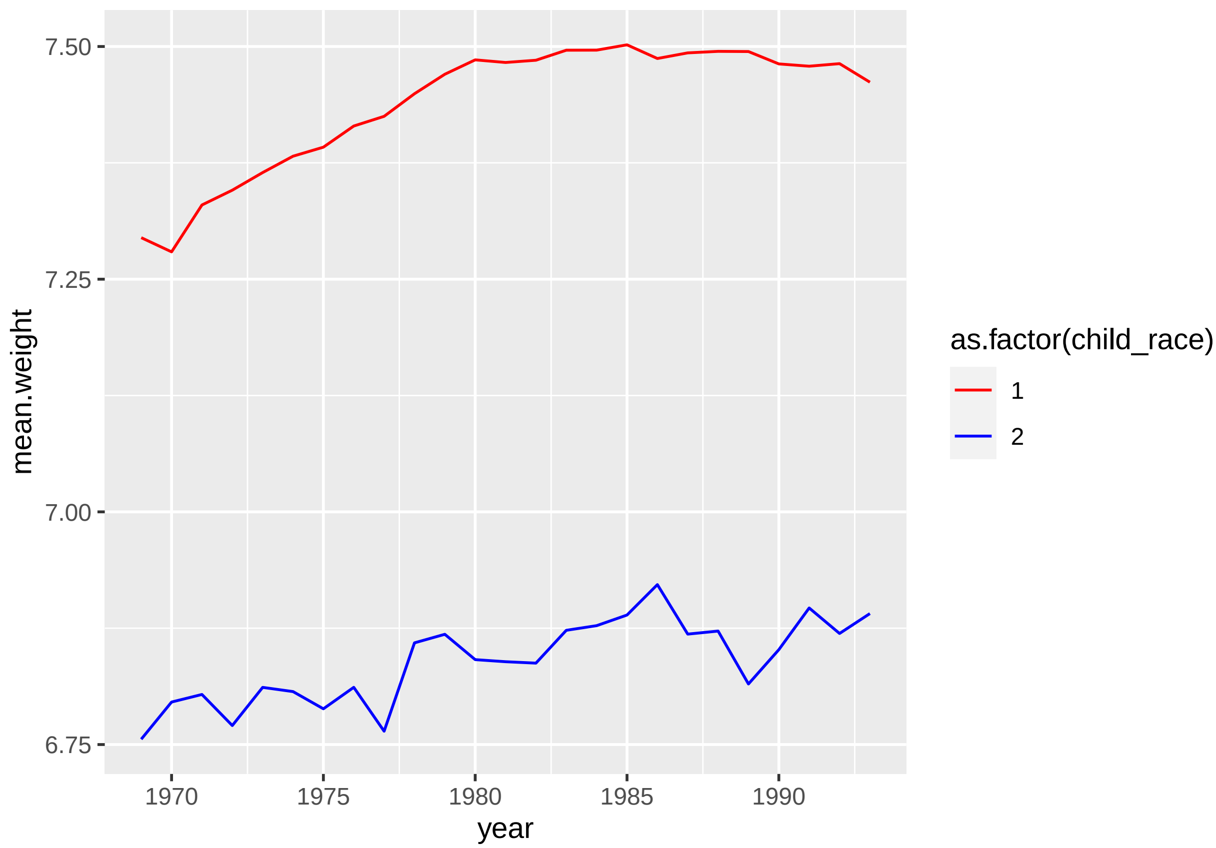
Referenced: Stack Overflow https://stackoverflow.com/questions/5171263/changing-line-colors-with-ggplot
Accessed on October 27, 2021

Write 2-3 sentences comparing Question 1 and Question 2.

Question 3

Create a graphic of mean weight of the child versus year, which includes separate lines for the two top race categories, white and black. The graphic should look like the following. Notice that the legend is different from the default legend. You'll want to investigate `scale_color_discrete` to change the legend.

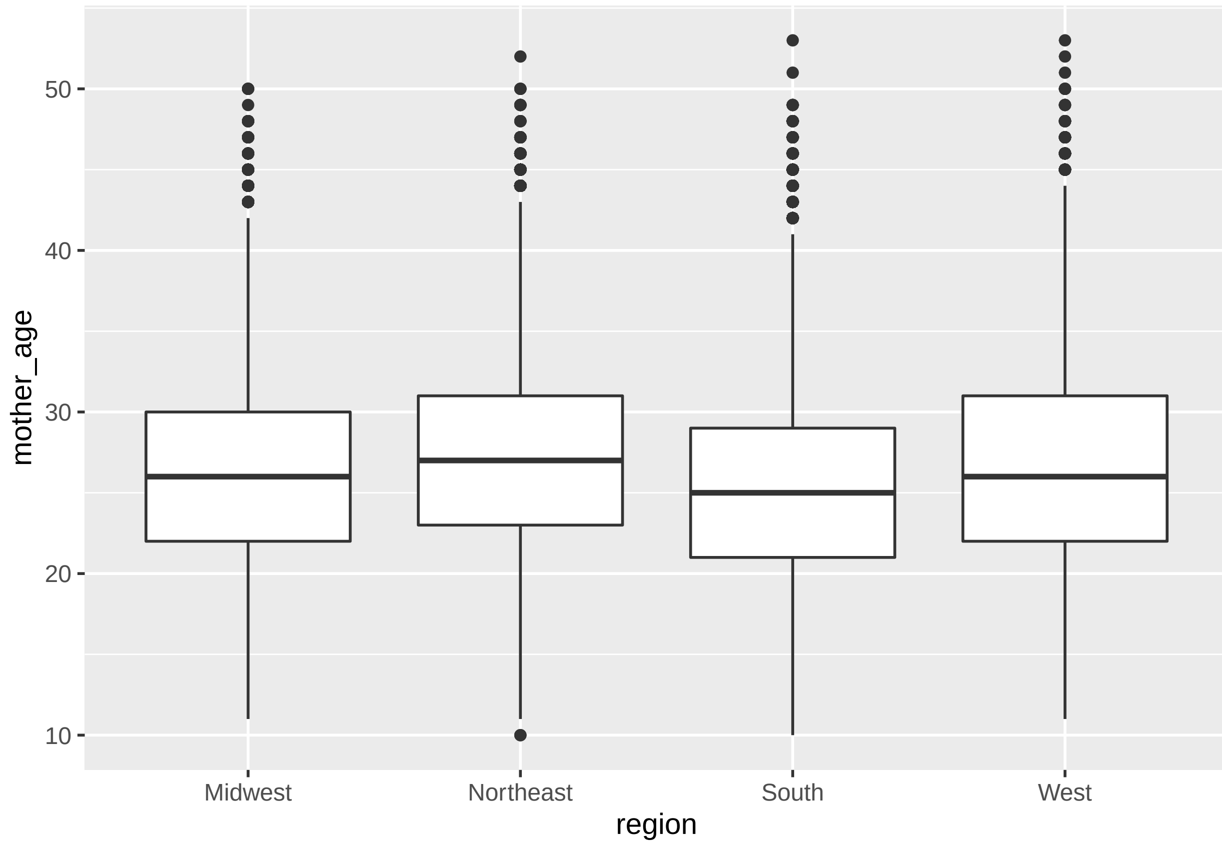
```
birth_data %>%
filter(child_race <= 2) %>%
group_by(child_race, year) %>%
summarise(mean.weight = mean(weight_pounds, na.rm=TRUE)) %>%
ggplot(mapping=aes(x = year , y=mean.weight))+
 geom_line(mapping=aes(color=as.factor(child_race))) +
scale_color_manual(values = c("red", "blue"))
```



Question 4

Create a graphic showing side-by-side boxplots of the age of the mother for the four regions. (Don't include data for which `region` is missing.) The graphic should look like the following.

```
birth_data%>%
 filter(!is.na(region)) %>%
 group_by(region, year) %>%
ggplot(mapping=aes(x = region , y=mother_age))+
 geom_boxplot()
```



Question 5

Write a function called `quantitative_summary` which takes two inputs:
`x`: A numeric vector
`group`: A factor vector of the same length as x

and produces a **list** as output which contains the following elements:

`missing`: The number of missing values in x
`means`: The means of x for each level of groups.
`sds`: The standard deviations of x for each level of groups
`is.binary`: Set to FALSE for for this function

Here is an example of the function in action.

```
quantitative_summary <- function(x,group) {
  list(missing = sum(is.na(x)),
       means = tapply(x, group,mean,na.rm = TRUE),
       sds = tapply(x, group, sd,na.rm = TRUE),
       is.binary = FALSE)
    }
quantitative_summary(birth_data$weight_pounds, birth_data$is_male)
```

```
$missing
[1] 1583
```

```
$means
   FALSE     TRUE
7.178759 7.438649

$sds
   FALSE     TRUE
1.283197 1.353845

$is.binary
[1] FALSE
```

Hint:

- When computing the means and standard deviations, you need to exclude missing values using `na.rm`.

Reference Code: Stack Overflow https://stackoverflow.com/questions/58618101/how-to-create-a-function-that-checks-if-a-variable-is-binary-in-r#comment103558697_58618101 Accessed On October 27,2021.

Question 6

Write a function called `binary_summary` which takes two inputs:

`x`: A vector containing the values 0 and 1 (possibly NA)
`group`: A factor vector of the same length as x

and produces a **list** as output which contains the following elements:

`missing`: The number of missing values in x
`prop`: The proportion of 1s in x for each level of groups
`is.binary`: Set to TRUE for for this function.

```r
binary_summary <- function(x,group) {
  list(missing = sum(is.na(x)),
       prop = prop.table(table(x,group)),
      is.binary = TRUE)

}
```

Here is an example of the function in action using **plurality defined as a binary variable (single vs multiple births)**:

```r
#First define a new binary variable using plurality (single vs multiple).
birth_data <- birth_data %>%
  mutate(plurality1 = ifelse(plurality==1, 1, 2))

binary_summary(birth_data$plurality1, birth_data$is_male)
```

```
$missing
[1] 27869

$prop
   group
```

```
x        FALSE       TRUE
  1 0.47408950 0.50015186
  2 0.01280192 0.01295672

$is.binary
[1] TRUE
```

Referenced: Yash Manne, TA Accessed on October 29, 2021.

Section 2

Flint is the second poorest city of its size in the United States and has spent six of the past 15 years in a state of financial emergency. One of the cost-cutting measures taken by emergency managers was to stop buying water, sourced from Lake Huron, from the Detroit Water and Sewerage Department. Instead, Flint would use the Flint River for its water supply while waiting for a new pipeline to Lake Huron to be opened. The move was expected to save roughly $5 million over a period of two years.

The Flint River supply was switched on in April 2014. Not long after, problems arose.Flint resident and mother of four LeeAnne Walters noticed that the water coming out of her taps was orange. More worryingly, her family's hair was falling out, her preschool sons had broken out in rashes and one of them had stopped growing.

The orange colour was from iron, but the family's symptoms pointed to a far more dangerous contaminant: lead. (Langkjaer - Bain 2017)

Introduction

The data set consists of 271 homes sampled with three water lead contaminant values at designated time points. The lead content is in parts per billion (ppb). Additionally, some location data is given about each home.

To get started, read in the `flint.csv` file using the function `read.csv`, as was done in the ICA with the cereal data. However, you do not need to use the `attach` function. The data set has five variables:

- **id**: sample id number
- **zip**: zip code in Flint as to the water sample's location
- **ward**: ward in Flint as to the water sample's location
- **draw**: water sample at one of three time points
- **lead**: lead content in parts per billion

Before you get started, read *The murky tale of Flint's deceptive water data* by Langkjaer - Bain (2017).

```r
flint<-read.csv("flint.csv")
glimpse(flint)
```

```
Rows: 813
Columns: 6
$ id                     <int> 1, 2, 4, 5, 6, 7, 8, 9, 12, 13, 15, 16, 17, 18,~
$ zip                    <int> 48504, 48507, 48504, 48507, 48505, 48507, 48507~
$ ward                   <int> 6, 9, 1, 8, 3, 9, 9, 5, 9, 3, 9, 5, 2, 7, 9, 9,~
$ draw                   <chr> "first", "first", "first", "first", "first", "f~
$ lead                   <dbl> 0.344, 8.133, 1.111, 8.007, 1.951, 7.200, 40.63~
$ flushing_time..in.sec. <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
```

Question 1

Select one passage that you found particularly striking (perhaps you strongly agreed or disagreed with it, perhaps it made you question an assumption or seemed unclear) in the article and write a 4-5 sentence paragraph commenting on it.

I found the topic of flushing to be particularly striking to me. After reading the article, it is still unclear why flushing before water testing is permissible if evidence shows that this action can decrease the lead levels after testing. Isn't the purpose of water testing to ensure the quality and safety of drinking water for residents? It made me uncomfortable realizing that officials put money in front of public health and safety. They did not want to pay for new pipes and this was one of several tactics they used to try and make sure the lead count was below the standard. However, thanks to Walters, they were not able to get away with this.

Question 2

How many unique zip codes are in the data set? How many unique wards are in the data set?

```
length(unique(flint$zip))
```

```
[1] 8
```

```
length(unique(flint$ward))
```

```
[1] 10
```

Do the number of wards in the data set match how many wards Flint has? If not, suggest a way to handle this discrepancy.

The number of wards in Flint is 9, while the number of wards in the data set is 10. This is because the zip 48505 is found in Ward 0 and Ward 1. 3 values (from each draw) are found in Ward 0. Therefore, I would remove Ward 0 from the data set and combine the data with Ward 1 in order to have the correct number of Wards in flint.

Question 3

Which ward appears to have the worst water quality? Note that your answer should consider mean, median, and maximum lead levels. Your choice of 'worst ward' should include justification for why some of these statistics are more important to consider than others.

```
flint %>%
  group_by(ward) %>%
  summarise(mean=mean(lead), median=median(lead), maximum=max(lead))
```

```
# A tibble: 10 x 4
    ward  mean median maximum
   <int> <dbl>  <dbl>   <dbl>
 1     0  2.19  0.953    4.88
 2     1  2.58  1.17    23.8
 3     2 24.2   2.49  1051
 4     3  6.71  1.87   118.
```

```
 5      4  3.88  0.896  139.
 6      5  5.97  2.25    66.2
 7      6 12.9   2.11   240.
 8      7  6.55  2.48   105.
 9      8 10.5   3.1    158
10      9  4.83  1.93    51.0
```

Ward 8 appears to have the worst water quality as it has the highest median value. The mean for each ward can be easily skewed by extreme maximum values as seen in Ward 2 and in Ward 6. Therefore, because the maximum is so high, the mean automatically will be higher and this is an inaccurate representation of the ward as a whole. Therefore, instead the median is important to consider in this case as it does not rely on any extreme values. The highest mean is 3.1 seen in Ward 8 meaning that there are several values above 3.1 in Ward 8 which can infer poor water quality. ### Question 4

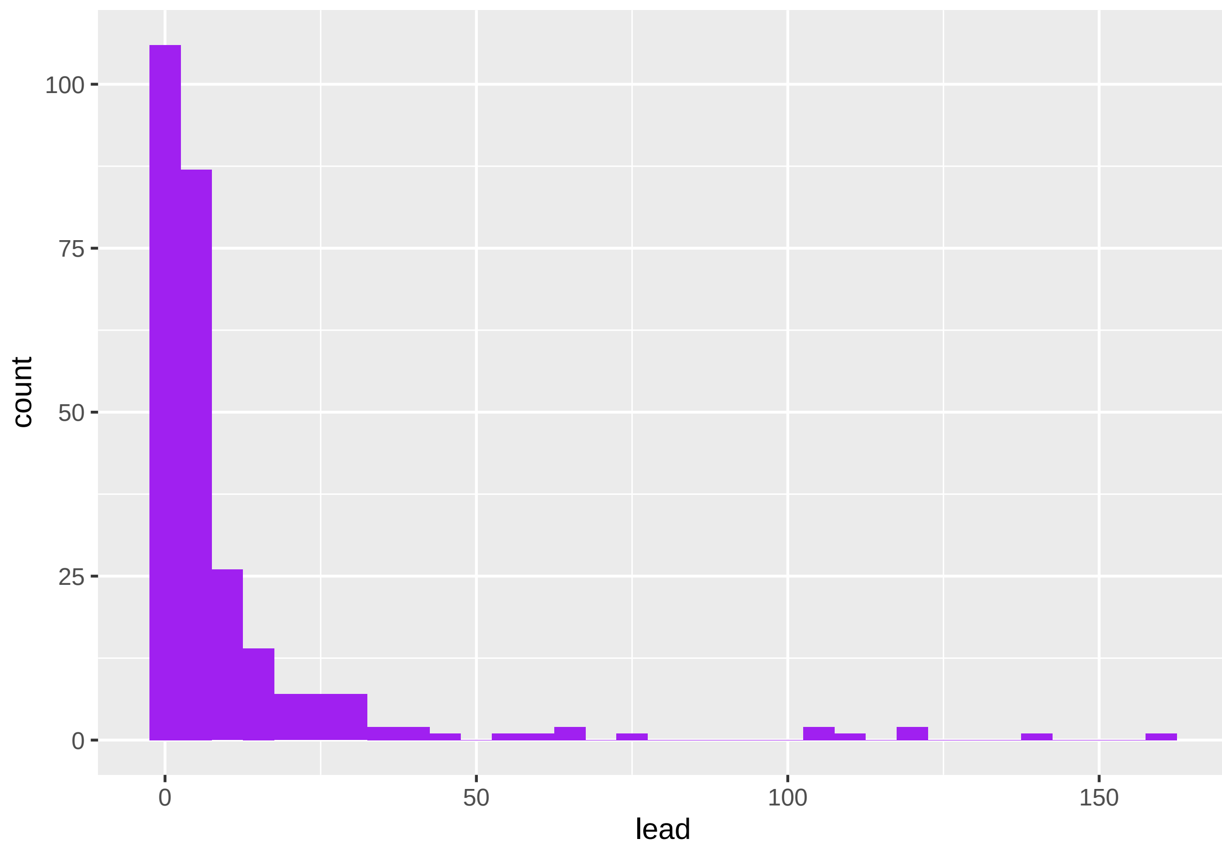
Langkjear-Bain (2017) writes at length about the practice of 'drawing' water before sampling it for lead levels. Compute the median and mean lead values for each draw. How do they compare? Create a histogram of the lead values for just the first draw and comment on the histogram's shape – does it confirm the earlier relationships between mean and median?

```
flint %>%
  group_by(draw) %>%
  summarise(mean=mean(lead), median=median(lead))
```

```
# A tibble: 3 x 3
  draw    mean median
  <chr>  <dbl>  <dbl>
1 first  10.6   3.52
2 second 10.3   1.4
3 third   3.66  0.831
```

As the draws continue, the mean and median both drastically increase, especially seen in Sample 3. Additionally, the average of the water sample is much higher then the median die to the fact that the average takes into account extreme values.

```
flint %>%
  subset(draw=="first") %>%
  ggplot(mapping=aes(lead)) +
    geom_histogram(binwidth=5, fill="purple", position="dodge")
```



The histogram absolutely matches the mean and median values listed above. This is becuase the majority of the data is depicted lower than 25 (representing a relative low median), but also depicts the extreme values (representing th higher average). ### Question 5

Compute the sample quantile for the *85th percentile* of lead values for each draw. Comment on what you observe. Is any draw above the EPA action threshold level?

```
lead.level<-flint %>%
  group_by(draw) %>%
  summarise(quantile(lead, probs = .85))
lead.level
```

```
# A tibble: 3 x 2
  draw   `quantile(lead, probs = 0.85)`
  <chr>                           <dbl>
1 first                           16.5
2 second                           8.84
3 third                            4.52
```

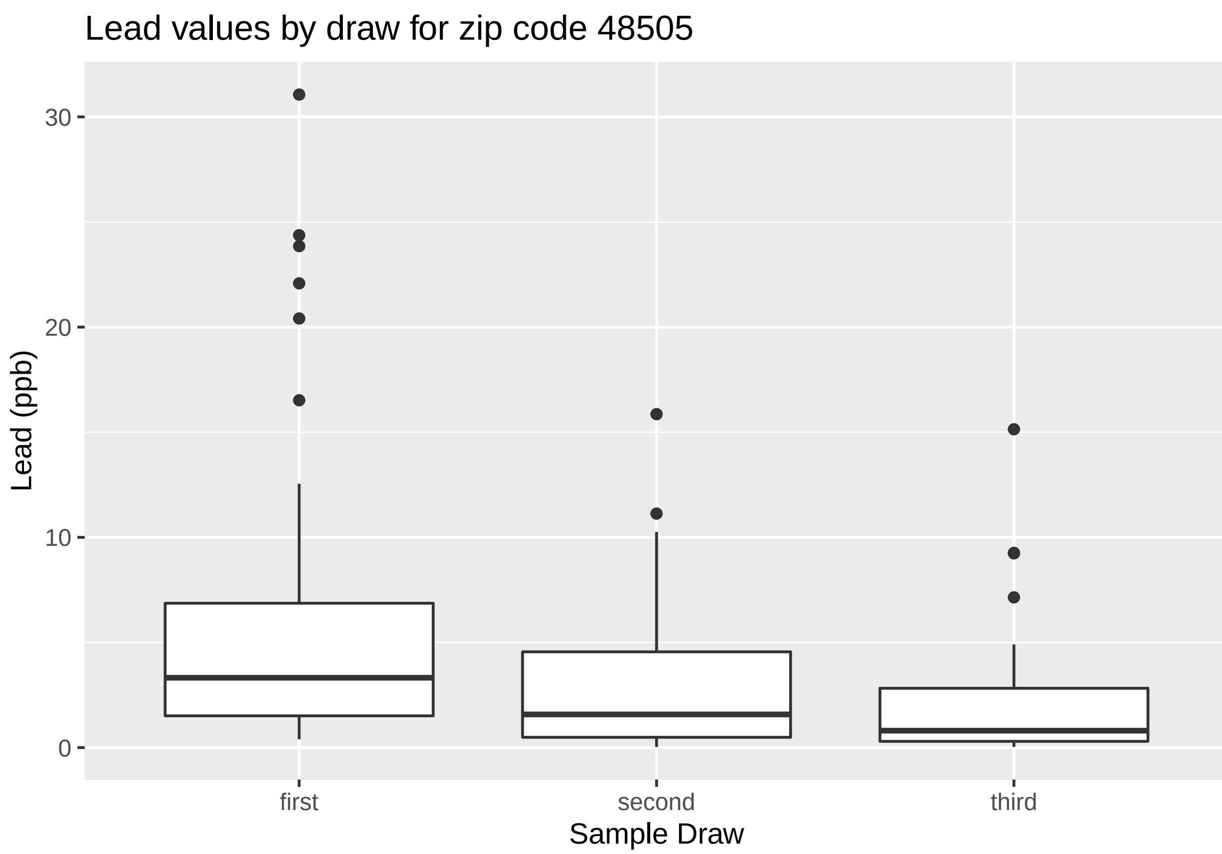
The sample at the 85th percentile appears to be dividing in half after each draw is taken. The EPA action threshold level is 15 ppb. The first draw is clearly over this being at 16.5 ppb. ### Question 6

Using the data from zip code **48505**, create a plot similar to that shown in the reference file for zipcode 48503.

In 1-2 sentences, comment on whether the plot confirms or contradicts the statement below, pulled from Langkjear-Bain (2017)

"Pre-stagnation flushing" – as it is known – "may potentially lower" lead levels as flushing "removes water that may have been in contact with the lead service line for extended periods"

```
flint%>%
  subset(zip == 48505) %>%
  ggplot(mapping=aes(x=as.factor(draw), y=lead), color="black")+
    geom_boxplot()+
    labs(x="Sample Draw", y="Lead (ppb)",
    title="Lead values by draw for zip code 48505")
```



This plot supports the idea that pre-stagnant flushing can lead to lower levels of lead in the water. The samples collected from this specific zip code show the majority of lead levels to be below the 15 ppb mark that is needed action. Additionally, the median continues to decrease after each draw and flush indicatinf that flushing lowers lead levels. ### Question 7

What is the largest lead value? What draw and zip code does it belong to? Comment on how we should handle this value if further statistical analysis were to be performed.

```
max<-flint %>%
  arrange(desc(lead))
glimpse(max)
```

```
Rows: 813
Columns: 6
$ id                         <int> 97, 258, 24, 205, 95, 24, 196, 35, 67, 83, 35, ~
$ zip                        <int> 48504, 48504, 48504, 48507, 48506, 48504, 48506~
```

```
$ ward                      <int> 2, 2, 6, 8, 4, 6, 3, 6, 7, 6, 6, 8, 8, 6, 2, 6,~
$ draw                      <chr> "second", "second", "second", "first", "first",~
$ lead                      <dbl> 1051.00, 259.80, 239.70, 158.00, 138.80, 120.00~
$ flushing_time..in.sec. <int> 45, 45, 45, 0, 0, 0, 0, 0, 0, 0, 120, 120, 45, ~
```

The largest lead value is 1051.000 from the second draw at zip code 48504. This value is significantly larger than all of the other lead levels which therefore determines it as an outlier. As stated in the article, this outlier is important and should not be omitted. Rather, it should be investigated further as to why this particular area has such a high lead level. Standardization of the data would be the best way to handle this maximum value.

Referenced: Yash Manne, TA Accessed on October 29, 2021.

What is the smallest lead value? What draw and zip code does it belong to?

```r
min<-flint %>%
  arrange(lead)
glimpse(min)
```

```
Rows: 813
Columns: 6
$ id                        <int> 141, 141, 6, 6, 235, 132, 161, 198, 213, 9, 104~
$ zip                       <int> 48505, 48505, 48505, 48505, 48503, 48506, 48507~
$ ward                      <int> 1, 1, 3, 3, 7, 4, 9, 7, 9, 5, 1, 7, 9, 3, 1, 9,~
$ draw                      <chr> "third", "second", "third", "second", "third", ~
$ lead                      <dbl> 0.031, 0.032, 0.035, 0.048, 0.058, 0.094, 0.096~
$ flushing_time..in.sec. <int> 120, 45, 120, 45, 120, 120, 45, 45, 45, 120, 12~
```

The smallest lead value is 0.031 from the first draw at 48505. ### Question 8

One way to standarize the data is to use z-scores. Based on each draw, compute z-scores for the lead values. How many z-scores exceed three in absolute value for each draw?

```r
#Extract lead values by draws
lead.draw1<-flint %>%
  filter(draw=="first") %>%
  pull(lead)
lead.draw2<-flint %>%
  filter(draw=="second") %>%
  pull(lead)
lead.draw3<-flint %>%
  filter(draw=="third") %>%
  pull(lead)


#find the z scores for each draw using: {lead.value.draw# - mean(lead.value.draw#)} /sd(lead.value.#dra
print("z-scores of draw 1")
```

```
[1] "z-scores of draw 1"
```

```r
zscore1<-(lead.draw1- mean(lead.draw1)) /sd(lead.draw1)
zscore1
```

```
  [1] -0.477811739 -0.116553896 -0.442237882 -0.122397841 -0.403278249
  [6] -0.159826918  1.390673749 -0.442748068 -0.002133161 -0.206207434
 [11] -0.291640345  0.636526551 -0.187237803 -0.305368978 -0.378557433
 [16] -0.473451970 -0.433935770 -0.468350113 -0.348549239  5.071895343
 [21] -0.358752953  0.272439496 -0.401747692 -0.244842404 -0.238673796
 [26] -0.464129486 -0.211448432 -0.015119706  2.608162307  0.452859705
 [31]  4.589537971 -0.259081223 -0.365107084 -0.287234196 -0.475214430
 [36] -0.448592013 -0.344050329  0.088308846 -0.470529998 -0.380227132
 [41] -0.238302751 -0.433796628 -0.275824589 -0.211170149 -0.456662223
 [46] -0.418351916 -0.374290426 -0.384215856 -0.420253518 -0.351935017
 [51] -0.381386645  0.939855129 -0.286445728 -0.393028154 -0.410281707
 [56] -0.449009438 -0.139605012 -0.347296965  4.390101750 -0.286167444
 [61] -0.362602536 -0.378696575  0.040536914 -0.318262762 -0.431431222
 [66]  0.059552925  0.014563824 -0.338809331 -0.436764982 -0.201569382
 [71] -0.254767835 -0.212978989  0.950522647  4.269512408 -0.429761524
 [76] -0.441263892 -0.343864807 -0.470251714 -0.096703035 -0.030100613
 [81] -0.301194732  3.022804124  5.943849053 -0.456662223 -0.157786175
 [86] -0.418583819 -0.445901943 -0.453601109 -0.376655832 -0.057418737
 [91] -0.459491435 -0.452024171 -0.428694772 -0.231484816  0.946812206
 [96] -0.425633658  0.612408682 -0.040814512 -0.276242014 -0.305183456
[101] -0.454528719 -0.398501055 -0.320117983  0.104542026 -0.468582016
[106] -0.174807824 -0.468164591  2.242683837 -0.477579836 -0.272809856
[111] -0.455178047 -0.436996884  0.243683576 -0.230928249 -0.328466476
[116] -0.471735891 -0.247949899 -0.439686954 -0.462042363 -0.199389498
[121] -0.003988382 -0.388343722 -0.420902845 -0.243219086  1.509407871
[126] -0.424566906  2.578478776 -0.410328087 -0.407452495 -0.369837896
[131] -0.320257124 -0.404066717  0.760826335 -0.439315910 -0.385931935
[136] -0.402582541 -0.402907204 -0.455595471  0.025695148 -0.230881869
[141] -0.203378223 -0.271279299 -0.464083106 -0.089653196 -0.375774602
[146] -0.474750625  1.029833331  0.103150611  1.269620601 -0.380830079
[151] -0.354764228  0.016419045 -0.205326204 -0.085618091 -0.119429488
[156] -0.342473391 -0.388622005 -0.451003800  0.759434919 -0.451745888
[161] -0.361582164 -0.407869920 -0.245630873  0.675486185  0.227450395
[166] -0.456986887 -0.258710179  0.741810323 -0.407174212  0.394884060
[171]  0.837354187  0.444047407 -0.359263138  0.380969905 -0.363159102
[176]  4.997686517  0.779378541 -0.438110017 -0.276659439  0.042855940
[181] -0.189649590 -0.336072880 -0.461300275  6.834354970 -0.448452872
[186] -0.100877281  0.038217888 -0.251289296 -0.449426863 -0.416264793
[191] -0.440336281 -0.470576378  0.004360111 -0.312882622 -0.394094906
[196] -0.442748068 -0.165067916 -0.432126930 -0.206207434 -0.456662223
[201] -0.062427833 -0.396367552 -0.282132340 -0.270769113 -0.458842107
[206] -0.407406115 -0.438898485 -0.113446401 -0.323132716 -0.383844812
[211] -0.455363569 -0.460419045 -0.362880819 -0.360793696 -0.385607272
[216] -0.289645983 -0.368168198  1.089200392 -0.251753101  0.235798888
[221] -0.352537964 -0.442748068 -0.429483240 -0.451328464 -0.394280428
[226] -0.305647261 -0.416403935 -0.142434224 -0.234499549 -0.460929231
[231] -0.414873378 -0.425726419 -0.374012143  0.530315168  0.271975691
[236]  2.115601222 -0.380922840 -0.470576378  0.857297809  0.076713717
[241] -0.333985757  0.271048081 -0.337696198 -0.440243520  0.133761752
[246] -0.390384465  0.847094096 -0.187608847 -0.449797907 -0.332176917
[251] -0.406200221  0.153241569 -0.202960798  0.393028839 -0.418027253
[256] -0.442098741 -0.312882622 -0.330460838 -0.316036497 -0.343354621
[261] -0.447849925 -0.437970875 -0.388900289  0.294238339 -0.339690561
[266]  0.170866165  0.346184517  0.100367780 -0.443536537  0.878632847
```

```
[271] -0.294933362
```

```r
print("z-scores of draw 2")
```

```
[1] "z-scores of draw 2"
```

```r
zscore2<-(lead.draw2- mean(lead.draw2)) /sd(lead.draw2)
zscore2
```

```
  [1] -0.1491923184  0.0069428018 -0.1509100417 -0.0422788541 -0.1518281352
  [6] -0.1318077745 -0.0085167075 -0.1155190196 -0.1371682557 -0.0903454892
 [11] -0.1403667749 -0.0222881094 -0.0673635367 -0.1362649702 -0.1418771867
 [16] -0.1370497920 -0.1489405831 -0.1433283667 -0.1425583528  3.3969288667
 [21] -0.1465268858 -0.0006092574 -0.1358059235 -0.1159040265 -0.0281520612
 [26] -0.1492367423  0.2749668605  0.0469242912 -0.1131201302 -0.1000743183
 [31]  1.0390575474 -0.1021030087 -0.1494292458 -0.0980604359 -0.1480965295
 [36] -0.1504361870 -0.1150155490 -0.0913524304 -0.1502288756 -0.1470155485
 [41] -0.0761742724 -0.1460086073 -0.1318818143 -0.1128239710 -0.1406925500
 [46] -0.1328147157 -0.1103066180 -0.1456236003 -0.1452237854 -0.1257661272
 [51] -0.1467193893  0.4779839791 -0.0985046746 -0.1264769093 -0.1451349377
 [56] -0.1500807960 -0.0754486824 -0.1236930130  0.0375952770 -0.1472820917
 [61] -0.0534144393 -0.0953801953 -0.1482742250 -0.1482149932 -0.1417439150
 [66] -0.0097161522 -0.0378660823 -0.1443204999 -0.1348878300 -0.1362501622
 [71] -0.1468378530 -0.1388119392 -0.0824972709 -0.0060289704 -0.0972459981
 [76] -0.1204056461 -0.1082779276 -0.1182584920 -0.0764111997 -0.1379826934
 [81] -0.1413144842  0.0199738057 -0.1118910696 -0.1495773254 15.4106260428
 [86] -0.1480965295 -0.1471636281 -0.1482149932 -0.1438022214 -0.0867323472
 [91] -0.0802316532 -0.1495773254 -0.1504361870 -0.0654385021 -0.0255162445
 [96] -0.1482298011 -0.1184658034  0.0122736670 -0.1384269322 -0.0652608066
[101] -0.1458012958 -0.0936328561 -0.0697328102 -0.1199317914 -0.1498290607
[106] -0.1083519674 -0.1497254050 -0.1095958360 -0.1506138825 -0.1319854700
[111] -0.1483334568 -0.1441279965 -0.0977494687 -0.1317337347 -0.1115652945
[116] -0.1495773254 -0.1451349377 -0.1416402593 -0.1480076817 -0.1270099958
[121] -0.0685481735 -0.1109581682 -0.1338068490 -0.0904047210 -0.0386953280
[126] -0.1316745029  0.1103023559 -0.1520650625 -0.1324741327 -0.1229081911
[131] -0.1346064788 -0.1470007405 -0.1391821381 -0.1482298011 -0.1362649702
[136] -0.1487925036 -0.1460530311 -0.1246851462 -0.0406944025 -0.1030655260
[141] -0.1330516431 -0.1289350305 -0.1391377143 -0.1110914398 -0.1362649702
[146] -0.1511173531  0.3769936984  0.0676554339  0.4205290980 -0.0313653883
[151] -0.1248924576  0.0033888916  0.0300432179 -0.1066342441 -0.1385157800
[156] -0.1458309117 -0.1445278113 -0.1395375291  0.3102098031 -0.1383084685
[161] -0.0509267022 -0.1455495605 -0.1328739476 -0.0883167988 -0.0938697834
[166] -0.1446462750 -0.1424250812 -0.0210738568 -0.1500956040  0.0331528893
[171]  0.0355221627 -0.0206296180 -0.1436541418 -0.0777439160 -0.1477411385
[176]  0.4513296528 -0.1386342436 -0.1510581213 -0.1413589081 -0.0625065262
[181] -0.1482594170 -0.0103825103 -0.1495773254  1.1924680032 -0.1455791764
[186] -0.0830599733  0.1902653348 -0.1158596026 -0.1496365572 -0.1465416938
[191] -0.1420548822 -0.1510581213 -0.1180215647 -0.1466157336 -0.1470895883
[196] -0.1466157336 -0.0123519689 -0.1435208701 -0.1421733459 -0.1486888478
[201] -0.0089017144 -0.1347693663 -0.1065453964 -0.0607888029 -0.1151340126
[206] -0.1345768629 -0.1471192042 -0.1051534482 -0.1451645536 -0.0439521535
[211] -0.1330220272 -0.1487776956 -0.1344583992 -0.1418327628 -0.0994079601
[216] -0.1212793156 -0.1193987048 -0.0636615470 -0.1138309122  0.0540321115
```

```
[221] -0.1119058776 -0.1451349377 -0.1482742250 -0.1472968997 -0.1474301713
[226] -0.1444389636 -0.1302085149 -0.1323704770 -0.0887758455 -0.1477115225
[231] -0.0874283213 -0.1503325313  3.6945688436  0.0823153133 -0.1225676081
[236] -0.0830599733 -0.1327850998 -0.1495773254  0.0187891690 -0.1451349377
[241] -0.1482446091  0.0374471974 -0.1162594175 -0.1499327164  0.1719034656
[246] -0.1292459976 -0.0715245732 -0.1239743642 -0.1485407683 -0.1461122630
[251] -0.1472228599  0.0281181832 -0.0927739945  0.1735323411 -0.1456532162
[256] -0.1435801020 -0.1442760760 -0.1458753356 -0.1407813977 -0.1416106434
[261] -0.1488369274  0.1326623741 -0.1510285054 -0.0589526160 -0.1146453500
[266] -0.1335254978  0.1467299352 -0.0359410476 -0.1477707544 -0.1042945866
[271] -0.0881983351
```

```
print("z-scores of draw 3")
```

```
[1] "z-scores of draw 3"
```

```
zscore3<-(lead.draw3- mean(lead.draw3)) /sd(lead.draw3)
zscore3
```

```
  [1] -0.333605807 -0.085373136 -0.335693391 -0.026256564 -0.344043726
  [6] -0.328386848  0.234501609 -0.337875865 -0.224577008 -0.129117502
 [11] -0.333416027  0.065122665 -0.211387275 -0.244409053 -0.293751938
 [16] -0.298876007 -0.326014594 -0.313109532 -0.282554899  2.471821914
 [21] -0.324875912 -0.085278246 -0.279803084 -0.194022375 -0.119628486
 [26] -0.328956189 -0.145248830  1.378687216 -0.149803558 -0.124942335
 [31]  8.621653492  0.040546112 -0.322313878 -0.013066831 -0.280941766
 [36] -0.336167842 -0.136518935 -0.335978062  1.089272213 -0.320036514
 [41] -0.098657759 -0.320700745 -0.316146017 -0.209015021 -0.299919799
 [46] -0.316335797 -0.206927437 -0.315197115 -0.230460198 -0.236438279
 [51] -0.311970850  3.884736467 -0.252474717 -0.210723044 -0.328386848
 [56] -0.317094919  0.529325351 -0.081482640  0.082867126 -0.315671566
 [61]  1.644379677  1.990728778 -0.326963496 -0.322883219 -0.231029539
 [66] -0.013920842  0.047378204 -0.260350600 -0.326678825 -0.318897832
 [71] -0.316810248 -0.283219130 -0.044855036 -0.050073995 -0.297642435
 [76] -0.303240955 -0.164511534 -0.325445253  0.100896257 -0.289386991
 [81] -0.306277440  0.026692148 -0.271737420 -0.328386848 -0.221350743
 [86] -0.324781022 -0.326868606 -0.321839427 -0.305802989  0.026027917
 [91] -0.256934554 -0.337875865 -0.335883171  0.664164275 -0.045993718
 [96] -0.323642340 -0.193642814  0.330530456 -0.259211918  0.104312303
[101] -0.312635081 -0.313678873 -0.112416833 -0.189182977 -0.331992675
[106] -0.138511629 -0.327912398 -0.299919799 -0.334934270 -0.246781307
[111] -0.327058386 -0.303240955 -0.138606519 -0.262343294 -0.145533501
[116] -0.321934317 -0.328386848 -0.321839427 -0.338445206 -0.283029350
[121]  0.109341482 -0.058139659 -0.317379589 -0.162803511  0.069202942
[126] -0.311496399  0.489091921 -0.344423286 -0.286635176 -0.243555041
[131] -0.322883219 -0.307131451 -0.289481881  0.037414737 -0.303146065
[136] -0.328291958 -0.304854088 -0.308270133  0.811244030 -0.037263823
[141] -0.299919799 -0.227423713 -0.333036466  4.489186816 -0.300109579
[146] -0.336357622  0.516420288  0.252910301  3.347658134 -0.224387228
[151] -0.203890952  0.477040870  0.051363591  0.079640860 -0.267752033
[156] -0.323452560 -0.310262827 -0.300773810  0.731536292 -0.247540428
[161] -0.257124335 -0.307321232 -0.290905233 -0.250671803 -0.170774285
[166] -0.310262827 -0.300963591  0.280713119 -0.330759102  0.385092300
```

```
[171]  0.468690536  0.286216749 -0.280941766 -0.105584742 -0.313014642
[176]  3.447292807 -0.296788424 -0.337875865 -0.304094966 -0.178270608
[181] -0.312160630 -0.186051601 -0.328386848  8.353114325 -0.311211729
[186] -0.206927437  0.329107103 -0.251146254 -0.332467125  0.100611587
[191] -0.320605855 -0.337875865 -0.192883693 -0.328386848 -0.315766456
[196] -0.328386848 -0.225905470 -0.297357764 -0.290430782 -0.323167889
[201]  0.127085942 -0.299919799 -0.062979058 -0.202847160 -0.295270181
[206] -0.262153513 -0.328291958 -0.100650453 -0.320036514  0.012933074
[211] -0.325255473 -0.341861252 -0.322883219 -0.276766599 -0.231503990
[216] -0.198197542 -0.270598738 -0.185102700 -0.268511154 -0.007278531
[221] -0.318992722 -0.318897832 -0.306372330 -0.318328491  0.007334555
[226] -0.311496399 -0.089263633 -0.243555041 -0.155781639 -0.318613161
[231] -0.225241239 -0.334364929  5.532029727  0.531507825  0.323318803
[236] -0.229416407 -0.216796015 -0.299919799  0.258793491 -0.309408815
[241] -0.331518224  0.508354624 -0.188423855 -0.335978062  0.096246639
[246] -0.267657143 -0.057380538 -0.307795683 -0.327722617 -0.318328491
[251] -0.316810248  0.435573868 -0.235204707  0.408719951 -0.324781022
[256] -0.323167889 -0.299540238 -0.316905138 -0.315197115 -0.321744537
[261] -0.322408768 -0.193073473 -0.308744584 -0.007183641 -0.203890952
[266] -0.316715358  0.057151891 -0.178460388 -0.322693438 -0.172482308
[271]  0.118071377
```

```
#Extract the z.scores which are above 3.
print("length of zscore 1 above 3")
```

```
[1] "length of zscore 1 above 3"
```

```
length(subset(zscore1, zscore1 > 3))
```

```
[1] 8
```

```
print("length of zscore 2 above 3")
```

```
[1] "length of zscore 2 above 3"
```

```
length(subset(zscore2, zscore2 > 3))
```

```
[1] 3
```

```
print("length of zscore 3 above 3")
```

```
[1] "length of zscore 3 above 3"
```

```
length(subset(zscore3, zscore3 > 3))
```

```
[1] 7
```

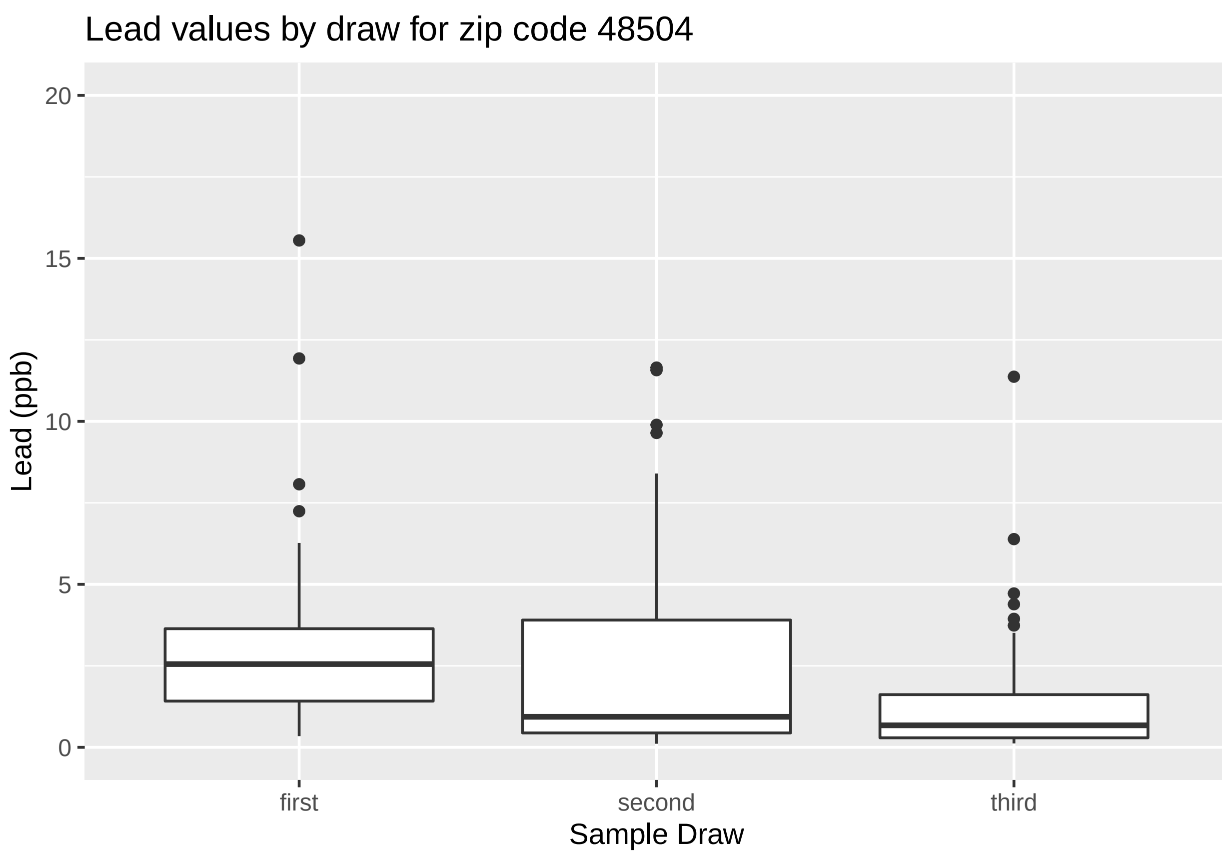
Referenced: Yash Manne, TA Accessed on October 29, 2021.

8 scores in the first draw, 3 scores in the second draw, and 7 scores in the thirs draw exceed 3.

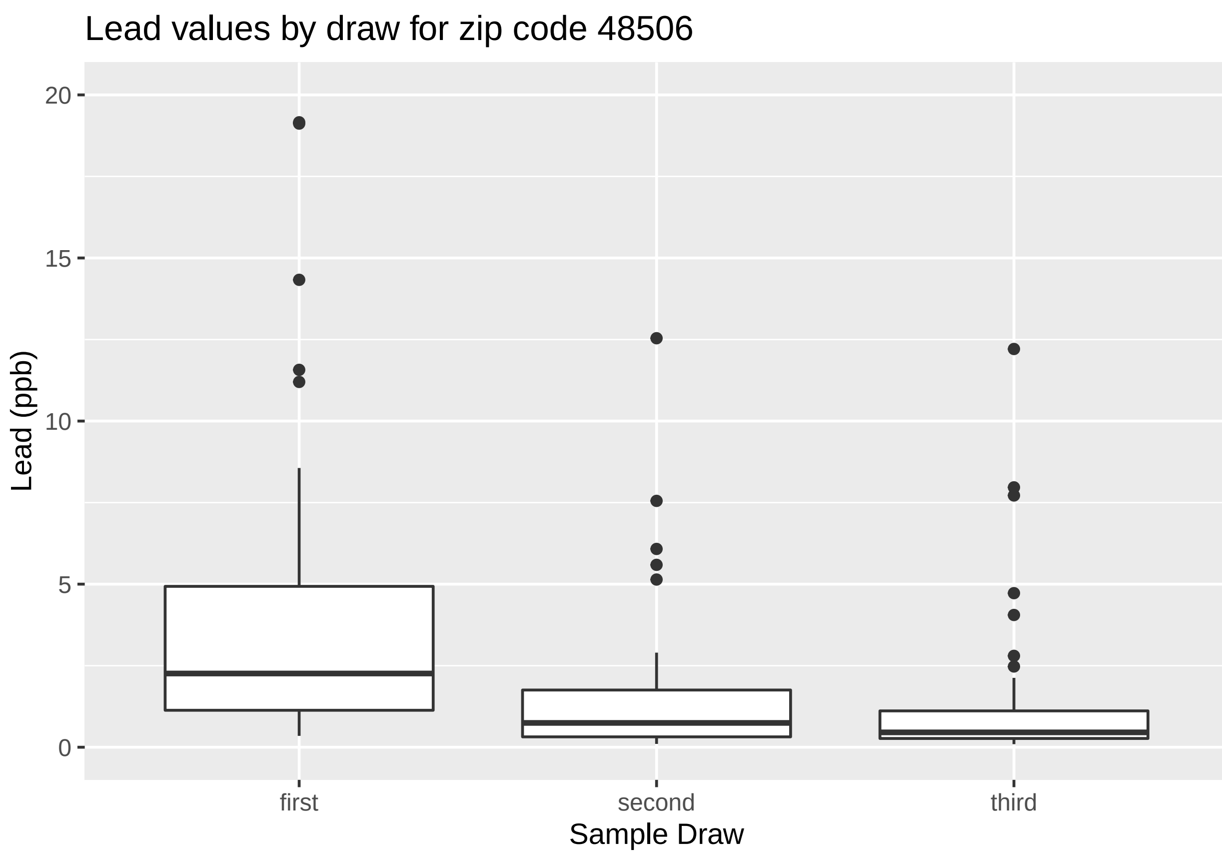
Question 9

Based on your analysis in questions 1-8, does it seem that flushing the water decreases the lead content? You may include further code and visualizations.

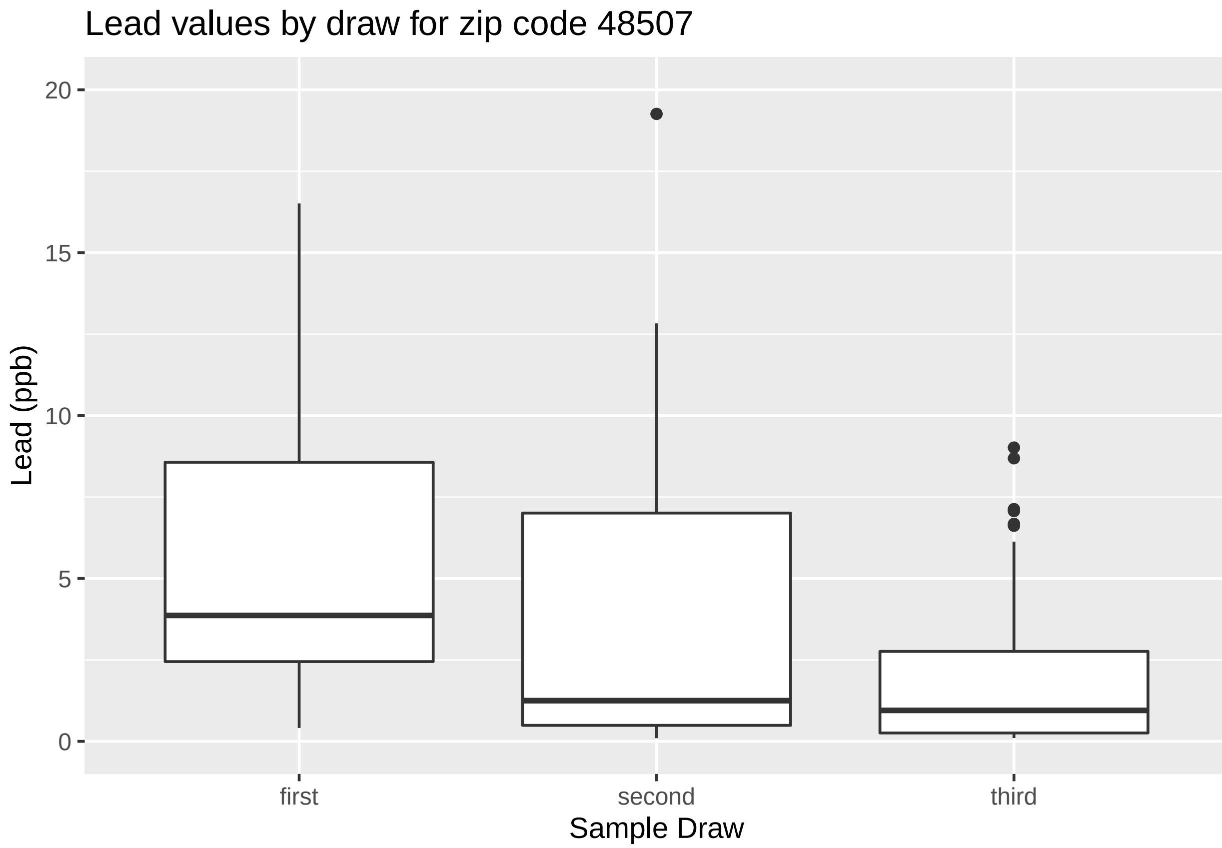
```
flint%>%
  subset(zip == 48504) %>%
  ggplot(mapping=aes(x=as.factor(draw), y=lead), color="black")+
    geom_boxplot()+
    ylim(0,20)+
    labs(x="Sample Draw", y="Lead (ppb)",
    title="Lead values by draw for zip code 48504")
```




```
flint%>%
  subset(zip == 48506) %>%
  ggplot(mapping=aes(x=as.factor(draw), y=lead), color="black")+
    geom_boxplot()+
    ylim(0,20)+
    labs(x="Sample Draw", y="Lead (ppb)",
    title="Lead values by draw for zip code 48506")
```




```
flint%>%
  subset(zip == 48507) %>%
  ggplot(mapping=aes(x=as.factor(draw), y=lead), color="black")+
    geom_boxplot()+
    ylim(0,20)+
    labs(x="Sample Draw", y="Lead (ppb)",
    title="Lead values by draw for zip code 48507")
```



Using the data and visualization above, flushing the water does decrease the amount of lead that is in the water. The box plots for all zipcodes continually show the median to be decreasing after each draw which also represents each flush.

Essential details

Deadline and submission

- The deadline to submit Homework 2 is **11:00pm on Saturday, October 30th, 2021.** This is a individual assignment.
- Submit your work by uploading your RMD and HTML/PDF files through D2L. Kindly **double check your submission to note whether the everything is displayed in the uploaded version of the output in D2L or not.** If submitting HTML outputs, please zip the .rmd and html files for submission.
- Please enable the `echo=TRUE` option in your code chunks and name each code chunk.
- **Late work will not be accepted except under certain extraordinary circumstances.**

Help

- Post general questions in the Teams HW 2 channel. If you are trying to get help on a code error, explain your error in detail
- Feel free to visit us in during our virtual office hours or make an appointment.

- Communicate with your classmates, but do not share snippets of code.
- The instructional team **will not answer any questions within the first 24 hours of this homework being assigned, and we will not answer any questions after 6 P.M of the due date**.

Academic integrity

This is an individual assignment.You may discuss ideas, how to debug code, and how to approach a problem with your classmates in the discussion board forum. You may not copy-and-paste another's code from this class. As a reminder, below is the policy on sharing and using other's code.

> Similar reproducible examples (reprex) exist online that will help you answer many of the questions posed on group assignments, and homework assignments. Use of these resources is allowed unless it is written explicitly on the assignment. You must always cite any code you copy or use as inspiration. Copied code without citation is plagiarism and will result in a 0 for the assignment.

Grading

Use the R Markdown blank file that is provided. If you want, you can use your own formatting. Self-formatting is at your discretion but is graded. Use the in-class assignments and resources available online for inspiration. Another useful resource for R Markdown formatting is available at: https://holtzy.github.io/Pimp-my-rmd/

Topic	Points
Questions(total 15)	75
R Markdown formatting and knitting	7
Communication of results	10
Code style	8
Total	**100**

Please note: Code style includes code efficiency.

Reference

1. http://www.cdc.gov/nchs/data_access/Vitalstatsonline.htm
2. Langkjr-Bain, R. (2017), The murky tale of Flint's deceptive water data. Significance, 14: 16-21.
3. https://holtzy.github.io/Pimp-my-rmd/