RNW Example File Fall 2019

Name: Michael Huber

Submission Date: 10/17/2019

Data Visualization Document (10/02/2019)

45 Points — Due Friday 10/18/2019 (via Canvas by 11:59pm)

(i) (35 Points) In this question, you have to work with the *faithful* data set from **datasets** in baseR. Be aware that different versions of this data set exist. Do not use any of the other versions or your results will differ slightly as different versions of the data set could be considered to be different samples from the same underlying geological process.

The help page for the *faithful* data set indicates:

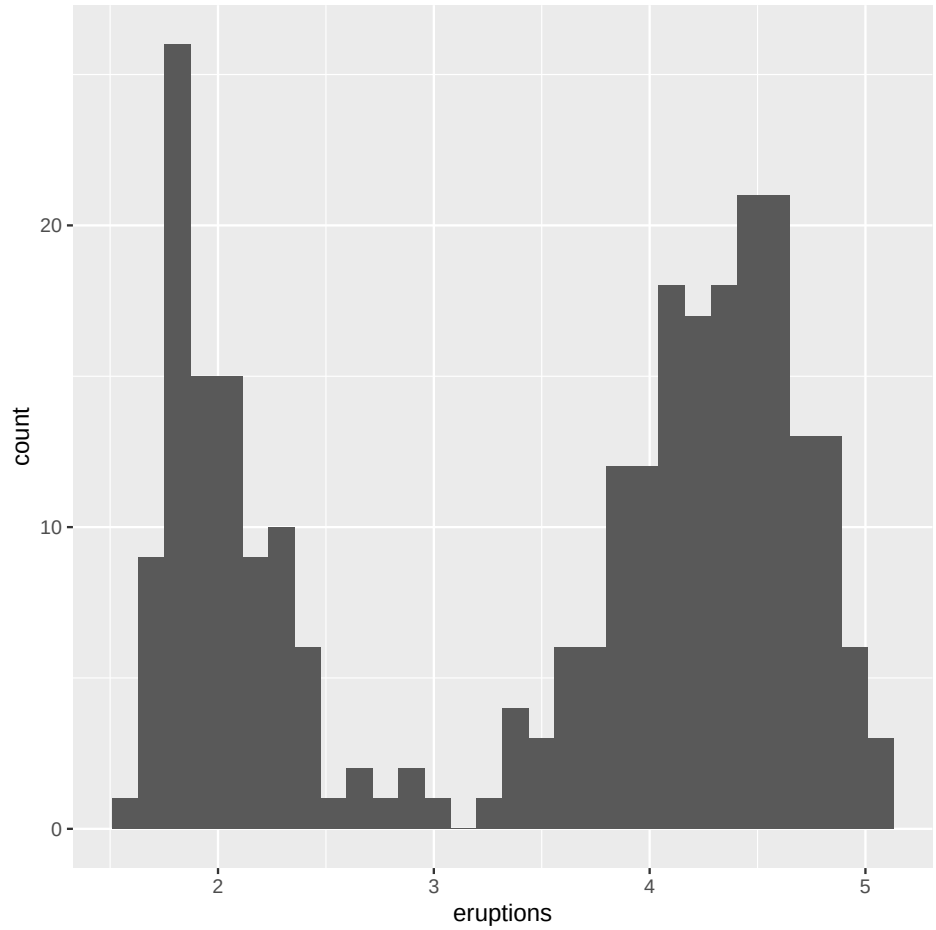
```
Old Faithful Geyser Data

Description
Waiting time between eruptions and the duration of the eruption for
the Old Faithful geyser in Yellowstone National Park, Wyoming, USA.

Usage
faithful

Format
A data frame with 272 observations on 2 variables.
[,1]  eruptions  numeric  Eruption time in mins
[,2]  waiting  numeric  Waiting time to next eruption (in mins)
```

(a) (1 Point) Load all required R packages to answer this question. Show your R code.

```
> library(ggplot2)
> library(gridExtra)
> library(MASS)
> library(vioplot)
```

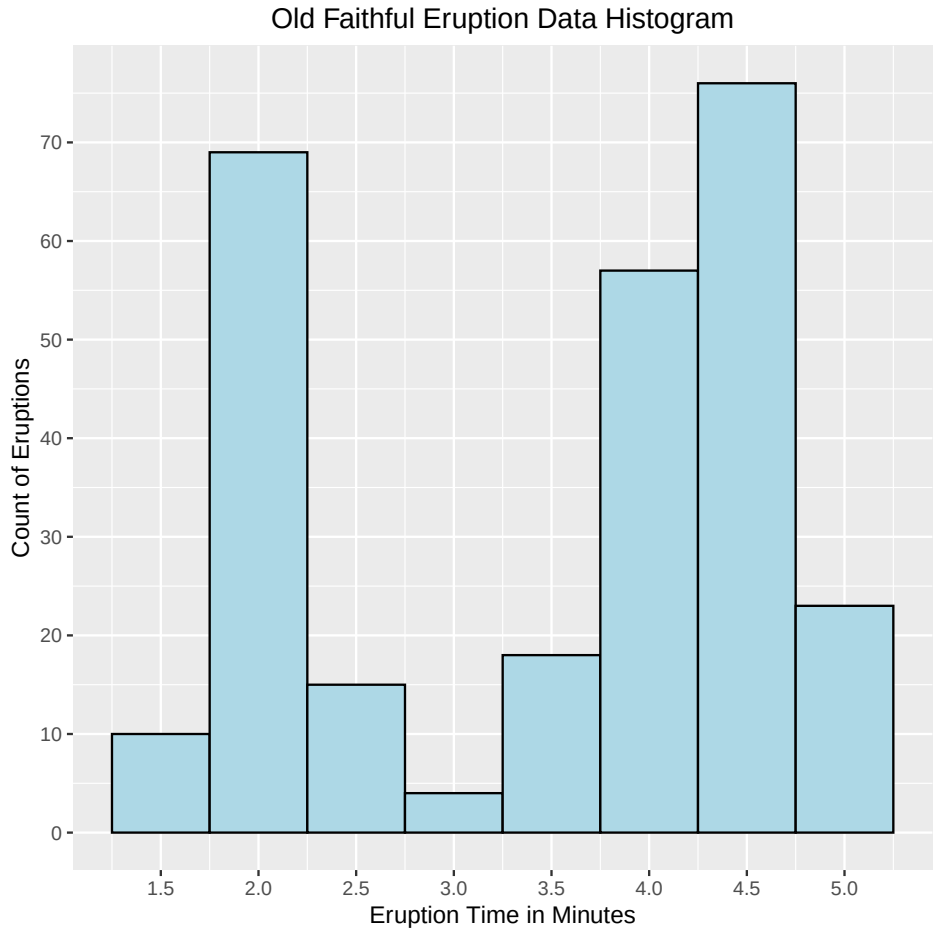
(b) (1 Point) Draw a basic histogram for *eruptions* using ggplot2. Include your R code and the resulting graph.

```
> data <- faithful
> ggplot(data, aes(x = eruptions)) +
+   geom_histogram()
```



(c) (3 Points) Further improve your histogram from (b). You may want to adjust the binwidth, starting points of the intervals, labels, title, etc. Clearly indicate which changes you made and why you made these changes. Include your final graph and the R code for the final graph. No need to include any intermediate graphs and the R code for those.

```
> ggplot(data, aes(x = eruptions)) +
+   scale_x_continuous(breaks = seq(0, 6, .5)) +
+   scale_y_continuous(breaks = seq(0, 100, 10)) +
+   geom_histogram(binwidth = .5, fill = "light blue", color = "black") +
+   xlab("Eruption Time in Minutes") +
+   ylab("Count of Eruptions") +
+   ggtitle("Old Faithful Eruption Data Histogram") +
+   theme(plot.title = element_text(hjust = 0.5))
```



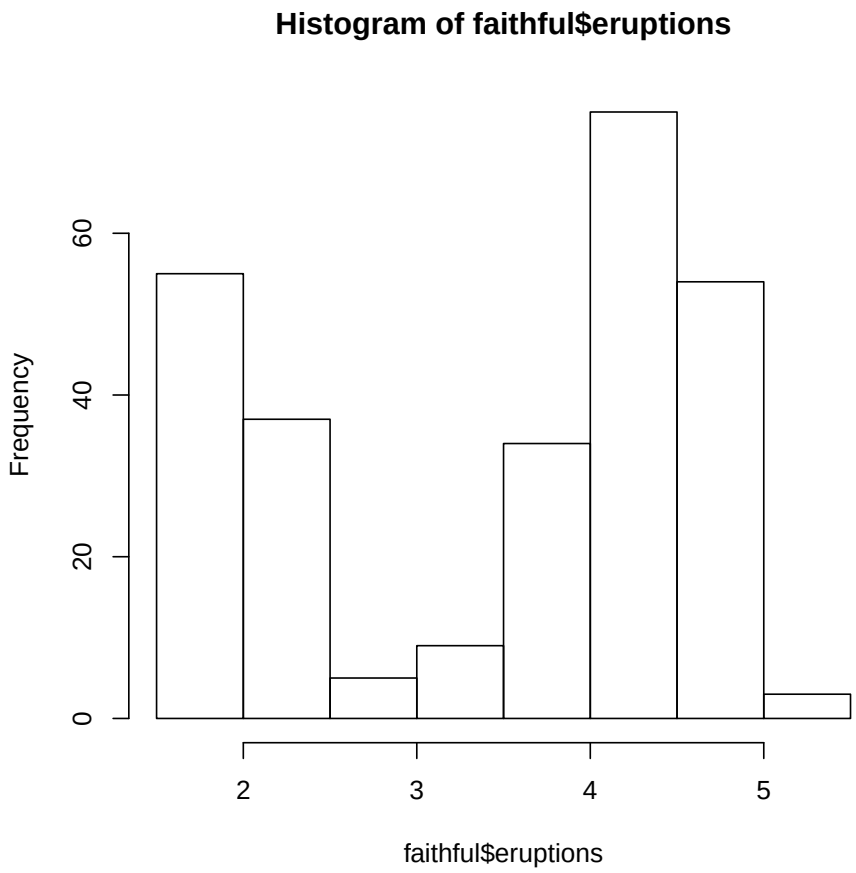
Answer: Changes made:

- Added scale x continuous so that the data would all fit on the x axis and define line breaks
- Added scale y coninuous To be able to define the line breaks on the y axis
- Added xlab to label that the x axis show the duration of the eruption in minutes
- Added ylab to label the number of eruptions that were occuring for each time block
- Added binwidth of .5 to break each bin into 30 second increments.
- Added fill = ”light blue” to add some color to contrast with the background
- Added color = ”black” to define the separation between different bars
- Added ggtitle() to label that this was Old Faithfuls eruption Data

- Added theme(plot.title = element text(hjust = 0.5)) to center the title over the graph

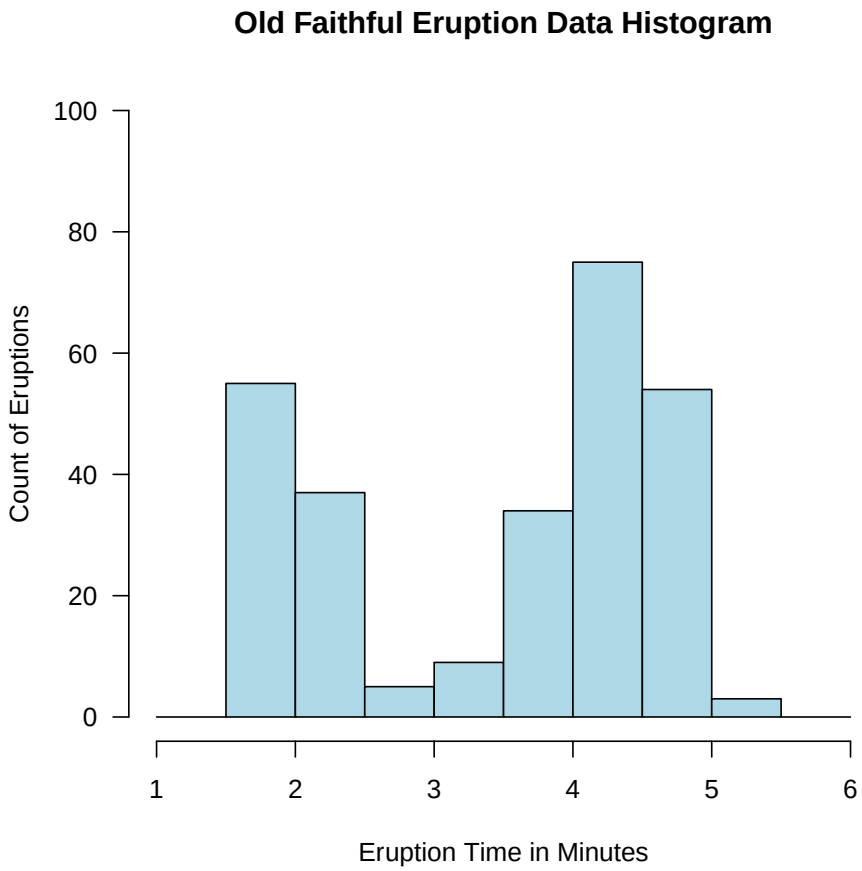
(d) (1 Point) Repeat (b) from above, now using the *hist* function from baseR.

```
> hist(faithful$eruptions)
```



(e) (3 Points) Repeat (c) from above, now using the *hist* function from baseR.

```
> hist(faithful$eruptions,
+      breaks = seq(1, 6, .5),
+      xlab = "Eruption Time in Minutes",
+      ylab = "Count of Eruptions",
+      main = "Old Faithful Eruption Data Histogram",
+      col = "light blue",
+      las = 1,
+      ylim = c(0, 100),
+      xlim = c(1, 6))
```



Answer: Changes made:

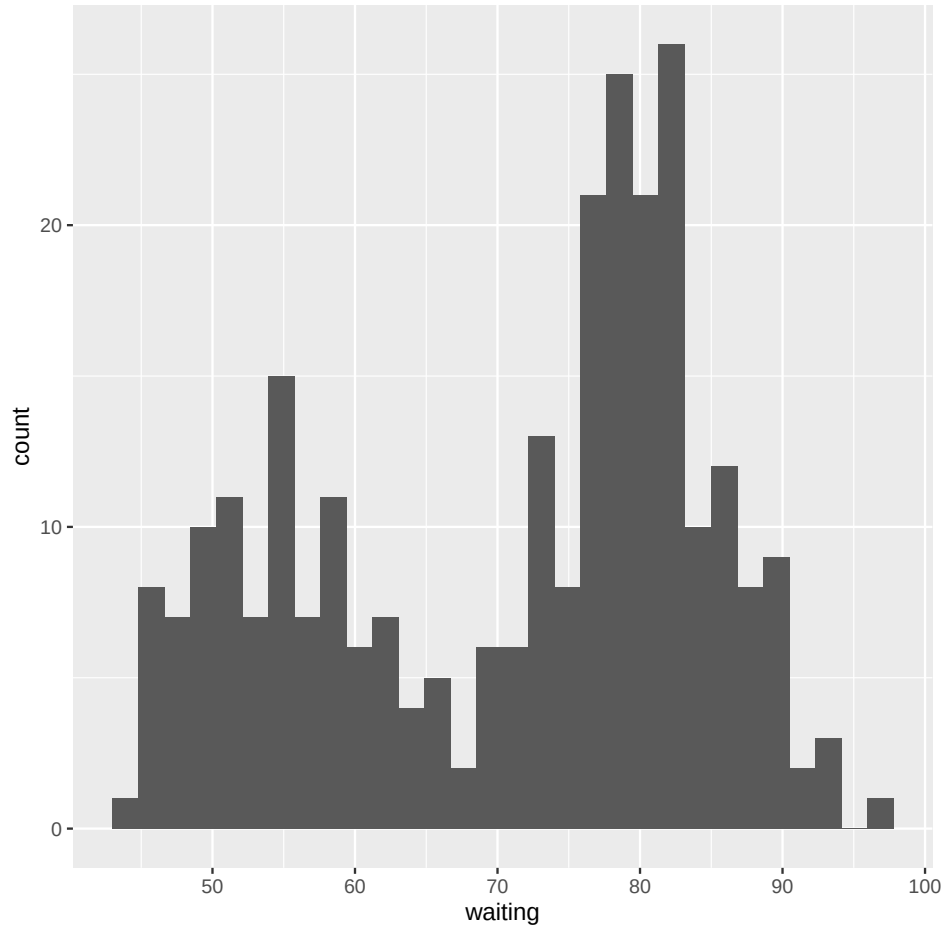
- Added breaks to the hist function which adjusted the groups that the bars were broken in to. It also sets the width of the x-axis.
- Added xlab and ylab to label what was being displayed on each axis.
- Added main to add a title to the graph
- Added col with light blue to add a light blue color to the bins.
- Added las = 1 which flips the numbers on the y axis from being on their sides to standing upright.
- Added ylim so that the y axis would be numbered to the top of the graph.

References Used to Solve this Problem:

- https://www.datacamp.com/community/tutorials/make-histogram-basic-r

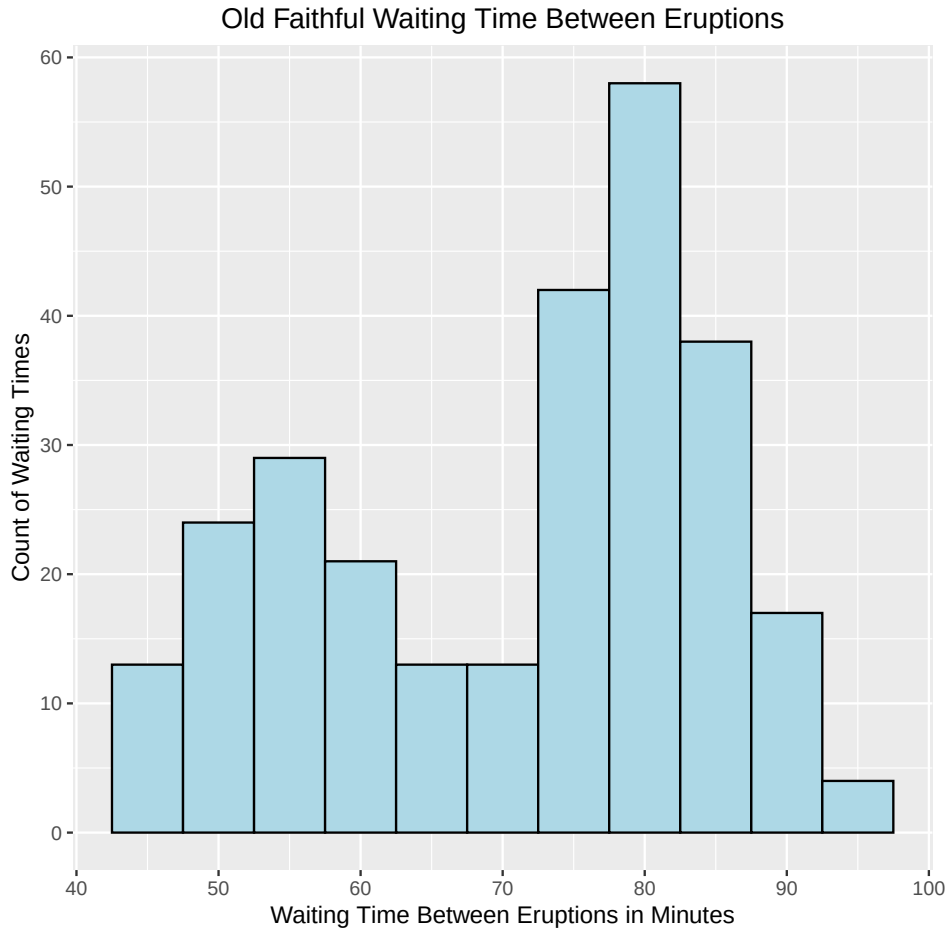
(f) (1 Point) Repeat (b) from above, now for *waiting*.

```
> ggplot(faithful, aes(x = waiting)) +
+   geom_histogram()
```



(g) (3 Points) Repeat (c) from above, now for *waiting*.

```
> ggplot(faithful, aes(x = waiting)) +
+   scale_x_continuous(breaks = seq(0, 100, 10)) +
+   scale_y_continuous(breaks = seq(0, 70, 10)) +
+   geom_histogram(binwidth = 5, fill = "light blue", color = "black") +
+   xlab("Waiting Time Between Eruptions in Minutes") +
+   ylab("Count of Waiting Times") +
+   ggtitle("Old Faithful Waiting Time Between Eruptions") +
+   theme(plot.title = element_text(hjust = 0.5))
```

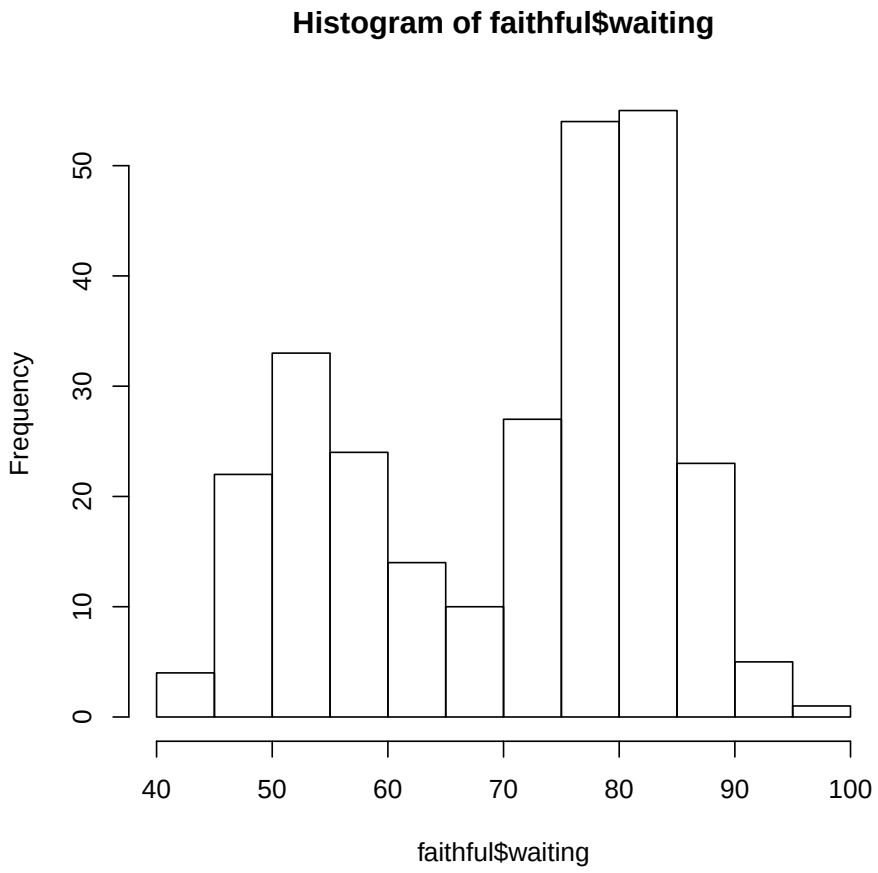


Answer:

- Added scale x continuos to set the x axis number breaks to intervals of 10.
- Added scale y continuous to set the y axis number breaks to intervals of 10.
- Added binwidth = 5 so that each bin represents 5 minute segments.
- Added a fill = light blue and color = black so that it was easier to distinguish between each bin.
- Added a xlab and ylab to describe what each axis was measuring.
- Added ggtitle to give a title to the graph, along with the plot.title to center it over the graph.

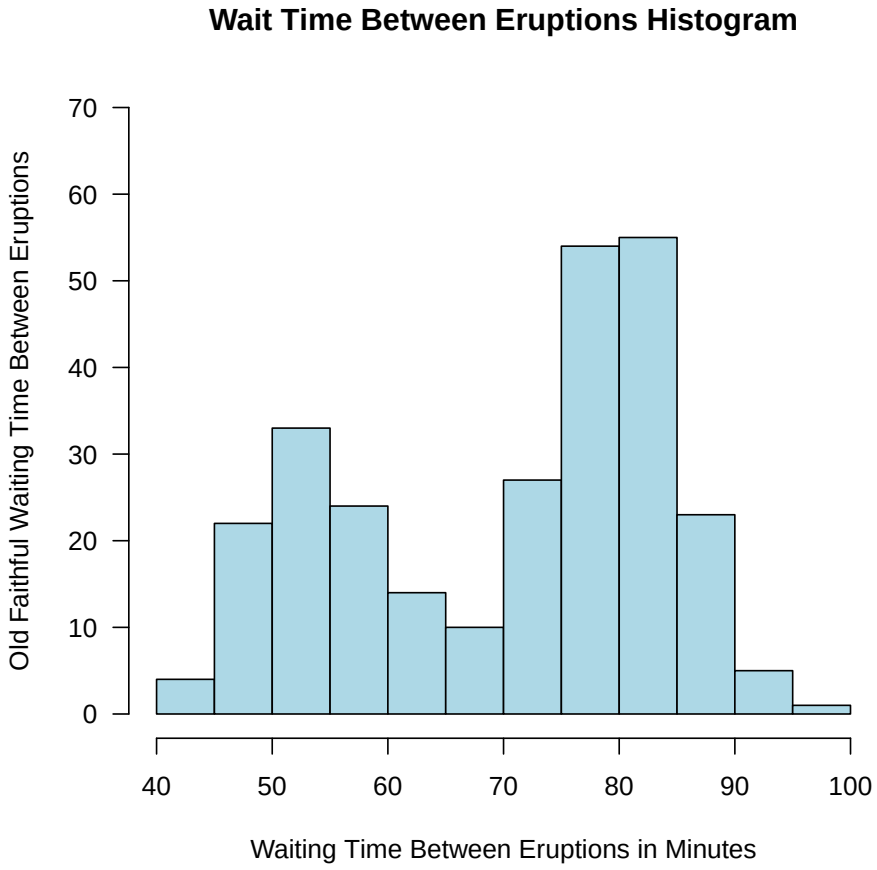
(h) (1 Point) Repeat (d) from above, now for *waiting*.

```
> hist(faithful$waiting)
```



(i) (3 Points) Repeat (e) from above, now for *waiting*.

```
> hist(faithful$waiting,
+      breaks = seq(40, 100, by = 5),
+      xlab = "Waiting Time Between Eruptions in Minutes",
+      ylab = "Old Faithful Waiting Time Between Eruptions",
+      main = "Wait Time Between Eruptions Histogram",
+      col = "light blue",
+      las = 1,
+      ylim = c(0, 70))
```

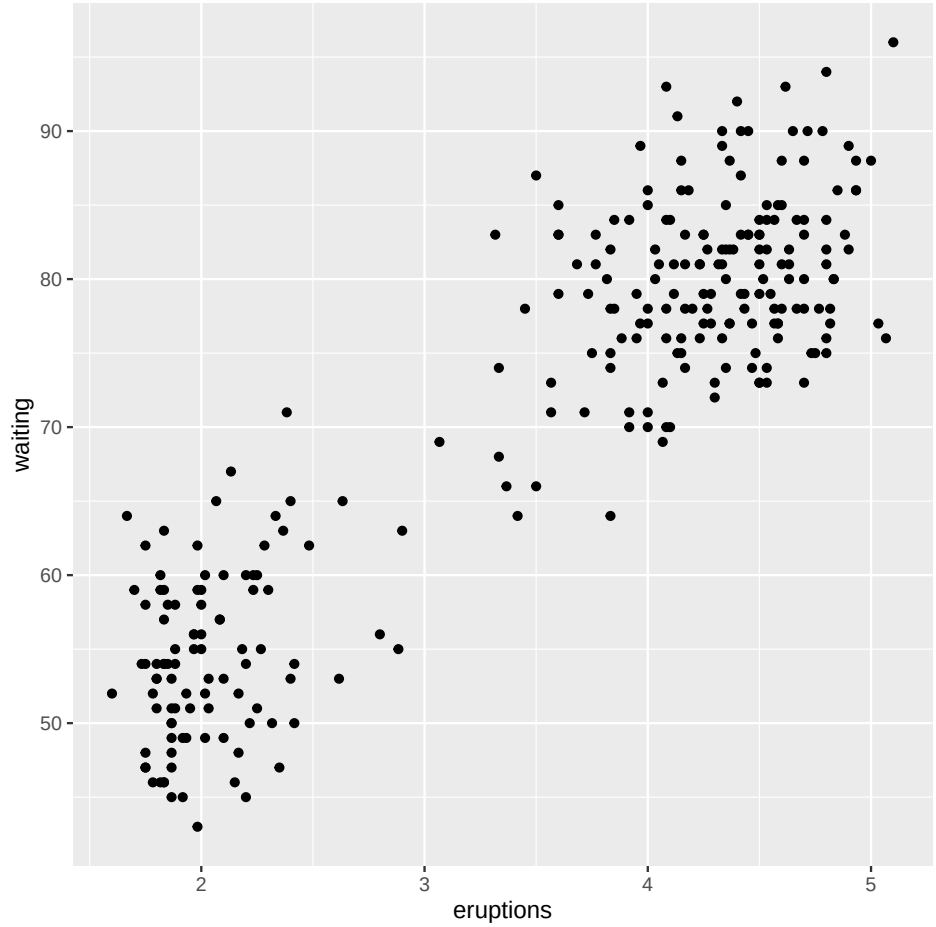


Answer:

- Added breaks into the his function to cover the range of waiting times and give a binwidth of 5 which means each bin is equal to 5 minutes.
- Added xlab and ylab to label each of the axis to tell the reader what they measure.
- Added main to give the graph a title to let the reader know the overall purpose of the graph.
- Added col = light blue so that the bars of the graph would stand out against the page.
- Added las = 1 so that the numbers of the y axis would stand upright.
- Added ylim to define how high the y axis will go.

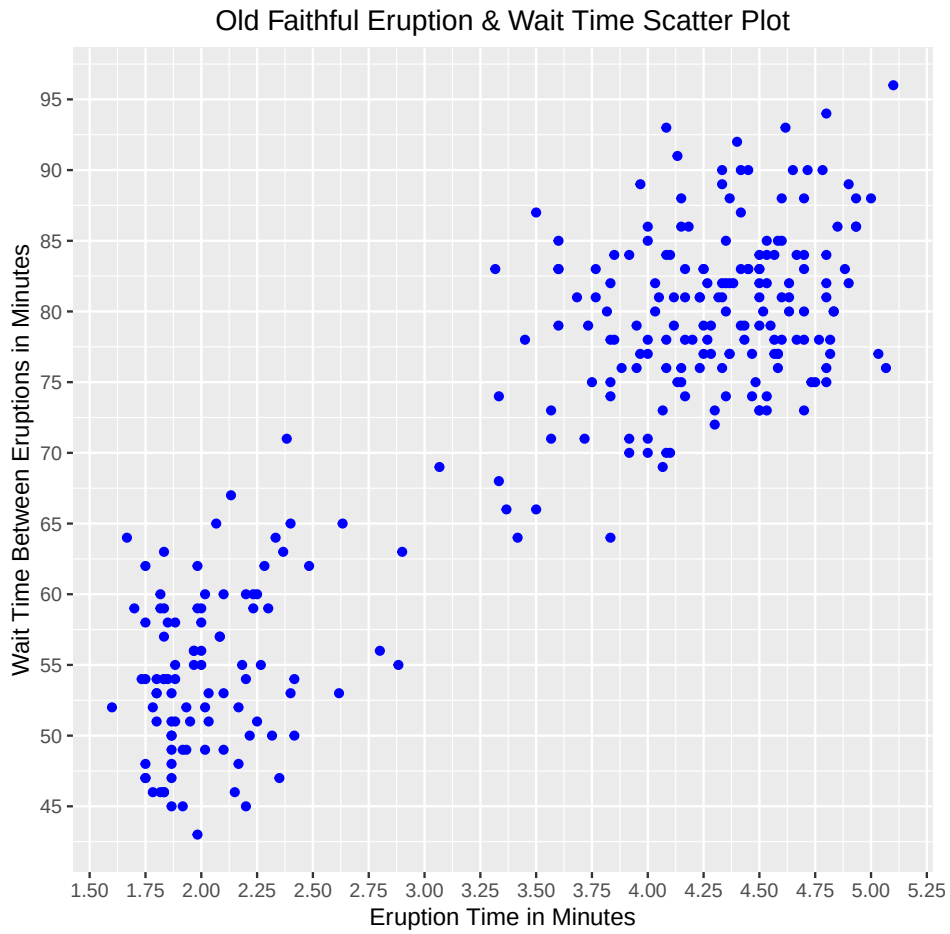
(j) (1 Point) Draw a basic scatterplot of the two variables using ggplot2. Assume that *eruptions* is the explanatory variable. Include your R code and the resulting graph.

```
> ggplot(faithful, aes(x = eruptions, y = waiting)) +
+   geom_point()
```



(k) (3 Points) Further improve your scatterplot from (j). You may want to adjust the range of the axes, labels, etc. Clearly indicate which changes you made and why you made these changes. Include your final graph and the R code for the final graph. No need to include any intermediate graphs and the R code for those.

```
> ggplot(faithful, aes(x = eruptions, y = waiting)) +
+   scale_x_continuous(breaks = seq(0, 6, .25)) +
+   scale_y_continuous(breaks = seq(40, 100, 5)) +
+   geom_point(color="blue") +
+   xlab("Eruption Time in Minutes") +
+   ylab("Wait Time Between Eruptions in Minutes") +
+   ggtitle("Old Faithful Eruption & Wait Time Scatter Plot") +
+   theme(plot.title = element_text(hjust = 0.5))
```

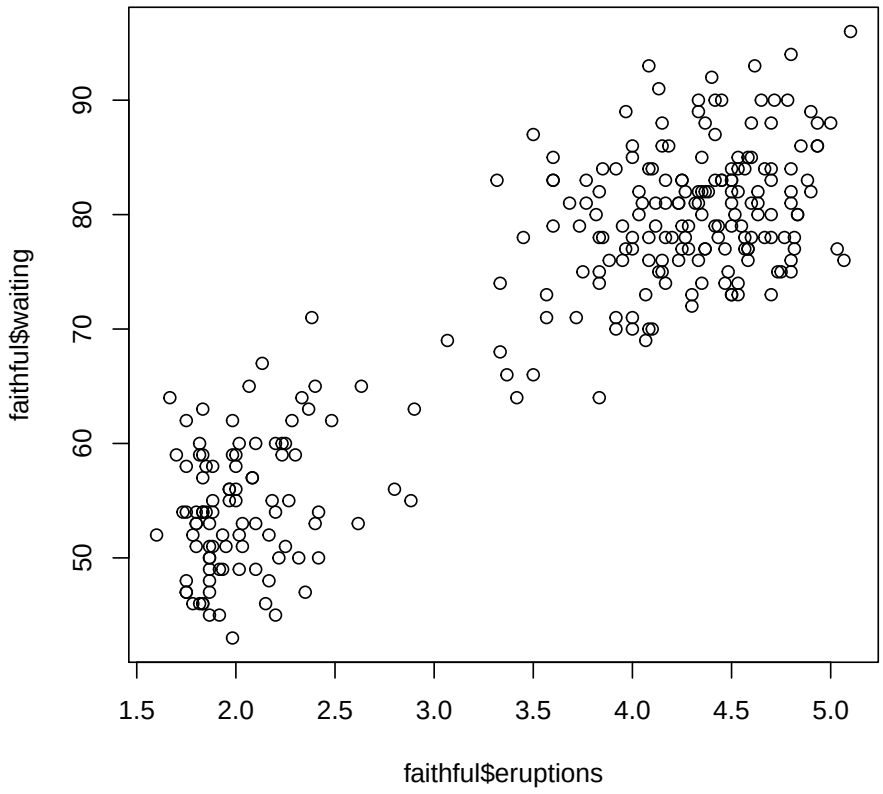


Answer:

- Added scale x continuous to be able to more easily identify what is the time for the eruption values.
- Added scale y continuouse to be able to more easily identify the value for the waiting variable.
- Added xlab and ylab labels to both axes to explain what each axis is showing on the graph.
- Added ggtitle to let the reader know what the graph was displaying.
- Added theme plot.title to center the title over the graph.
- Added color = blue to contrast with the background of the graph.

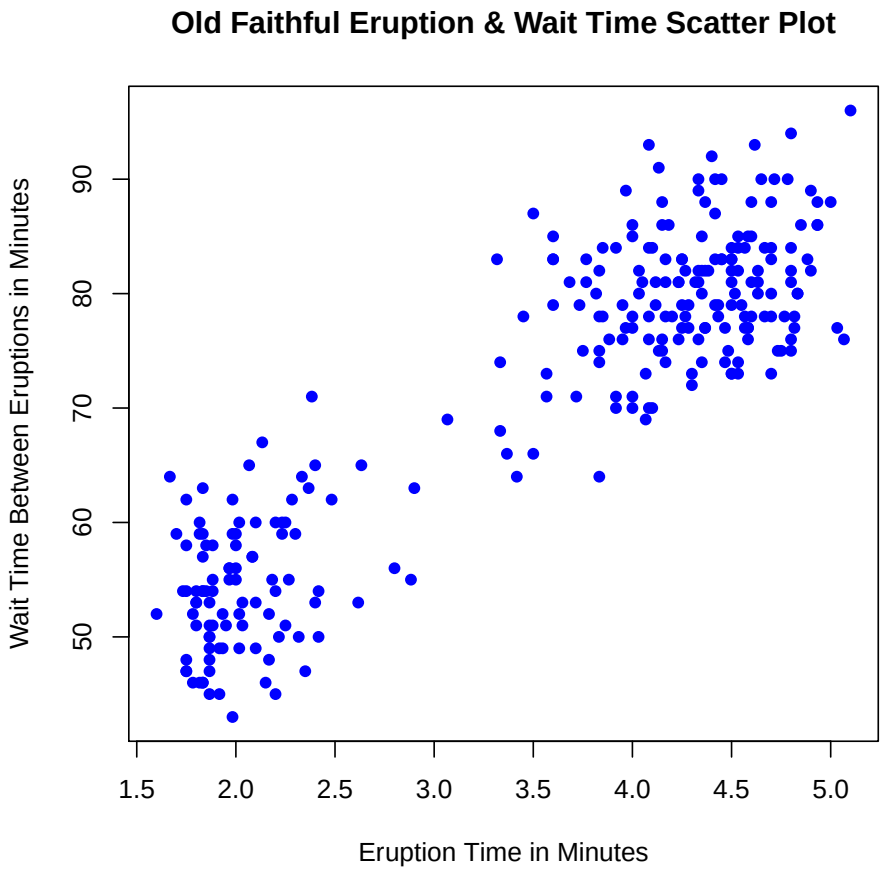
(l) (1 Point) Repeat (j) from above, now using the *plot* function from baseR.

```
> plot(faithful$eruptions, faithful$waiting)
```



(m) (3 Points) Repeat (k) from above, now using the *plot* function from baseR.

```
> plot(faithful$eruptions, faithful$waiting,
+      xlab = "Eruption Time in Minutes",
+      ylab = "Wait Time Between Eruptions in Minutes",
+      main = "Old Faithful Eruption & Wait Time Scatter Plot",
+      col = "blue",
+      pch = 16)
```



Answer:

- Added xlab to label the x axis with a descriptive label
- Added ylab to label the y axis with a descriptive label
- Added main to give the graph a title to explain what the graph is displaying
- Added col = blue so that the points would stand out from the background
- Added pch = 16 so that the points would be filled in circles

References:

- https://thomasleeper.com/Rcourse/Tutorials/plotcolors.html

(n) (5 Points) Provide a careful discussion of the three final graphs, i.e., the final histogram for *eruptions*, the final histogram for *waiting*, and the final scatterplot. It shouldn't matter whether you refer to the ggplot2 or baseR version as both (hopefully) will show similar information. Summarize what is shown in each of these three graphs. You need to tell the reader and verbally

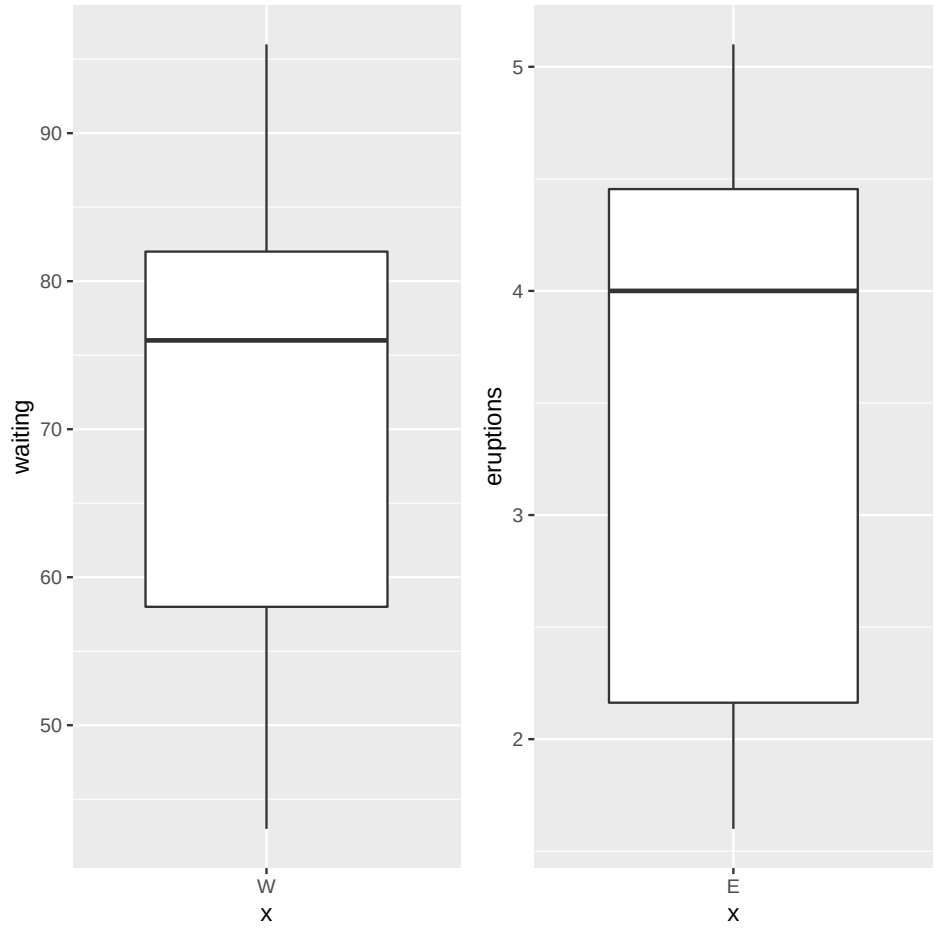
describe the most important features. Use the proper statistical language and refer back to your previously created graphs.

Answer:

- Final Histogram Eruptions: The data in the histogram has a bimodal distribution. With a heavier grouping on the right side of the graph. The eruption times go from 1.625 minutes to 5.125 minutes. The first grouping of eruption times goes from 1.75 minutes to 2.625 minutes. This is the smaller of the two groups in the bimodal distribution. The second grouping of eruption times starts around 3.25 minutes and goes to around 5 minutes. The peak of the first group which is a count of the eruptions that lasted that long is at 24 centered between 1.75 minutes to 1.875 minutes. The second peak reaches a height of 15 and is centered over 4.5 minutes.
- Final Histogram Waiting: The final histogram has a bimodal distribution, with a heavier grouping in the right peak.The left peak starts a 43 minutes and ends at 67 minutes. With a high of 9 centered on 54 minutes. The right peak starts at 68 minutes and continues to 96 minutes. With a high of 15 centered over 78 minutes.
- Final ScatterPlot: The scatterplot shows a positive linear association between eruption time in minutes and the wait time between eruptions.There is some spread to the data so it is not a perfect correlation. For example two plots near the two minute eruption time took anywhere from aound 42.5 minute wait time to a 62.5 minute wait time. The graph also is clearly split into two groups. With a cluster in the 1.75 to 2.5 minute eruption time and the next cluster grouping around the 3.75 to 5 minute wait time.

(o) (2 Points) Would boxplots be good replacements for the two histograms? First create two basic boxplots with a package of your choice. As always, include your R code and the resulting graphs. Then answer yes or no. Justify your answer!

```
> b1 <- ggplot(faithful, aes("W", waiting)) +
+   geom_boxplot()
> b2 <- ggplot(faithful, aes("E", eruptions)) +
+   geom_boxplot()
> grid.arrange(b1, b2, ncol = 2)
```



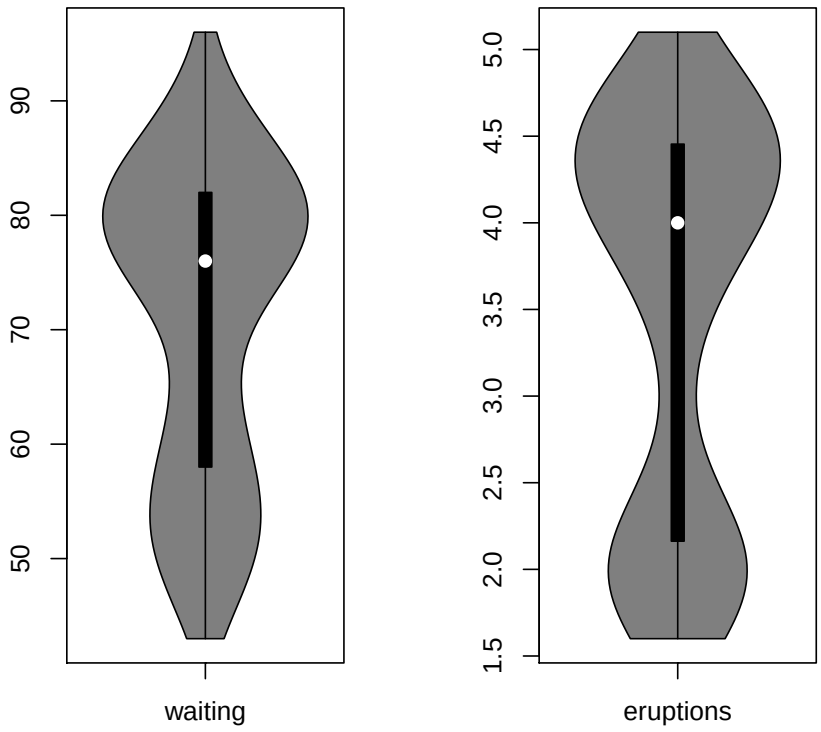
Answer:

- No, boxplots would not be good replacements for the two histograms. Both histograms show a bimodal distribution for the waiting and eruption data. But in the boxplot you are not able to see this split in the data very well

(p) (3 Points) Have you ever heard of violin plots? If not, google them! Find a suitable R package that creates violin plots or see how they can be created in ggplot2. Would violin plots be good replacements for the two histograms? First create two basic violin plots with a package of your choice. As always, include your R code and the resulting graphs. Then answer yes or no. Justify your answer!

```
> par(mfrow = c(1, 2))
> vioplot(faithful$waiting,
+         names = "waiting")
```

```
[1] 43 96
> vioplot(faithful$eruptions,
+         names = "eruptions")
[1] 1.6 5.1
```



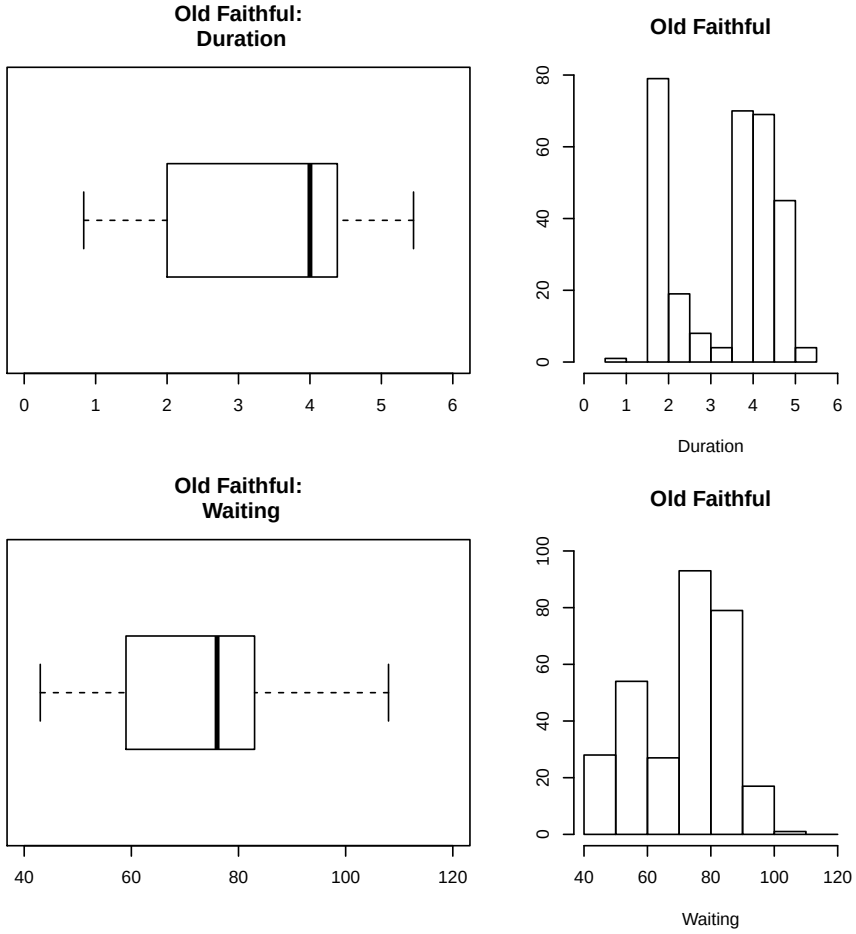
Answer: Yes violion plots would be good replacements for the histograms. They display similar data to a box plot but they allow you to show the distribution of the values so you can see where most of the values are clustering. They also contain a boxplot so you can get the benefit of both graphs.

References: https://www.tutorialgateway.org/r-ggplot2-violin-plot/

(ii) (10 Points) This question makes use of the *geyser* data set from the **MASS** R package, an alternative version of the Old Faithful data set used in the previous question. The settings used here may or may not be suitable for the graphs in the previous question. These graphs may not be perfect and may need some further adjustments, but those are not required to get full points in this question.

(a) (6 Points) Recreate the graphs (and layout) below using baseR. Use a ruler to check that the width and height proportions in your graphs match the ones I have used. I worked with integer multiples! Include your R code and the resulting graphs. Hint: You can create a new line via `\n` without any extra spaces before/after `\n`.

```
> grid <- matrix(c(1, 1, 1, 2, 2, 3, 3, 3, 4, 4),
+                nrow = 2, ncol = 5, byrow = TRUE)
> layout(grid)
> par(mar = c(4, 3, 4, 2))
> boxplot(geyser$duration,
+         horizontal = TRUE,
+         main = "Old Faithful:\n Duration",
+         ylim = c(0, 6))
> hist(geyser$duration,
+      main = "Old Faithful",
+      xlab = "Duration",
+      ylab = "Count",
+      xlim = c(0, 6))
> boxplot(geyser$waiting,
+         horizontal = TRUE,
+         main = "Old Faithful:\n Waiting",
+         ylim = c(40, 120))
> hist(geyser$waiting,
+      main = "Old Faithful",
+      xlab = "Waiting",
+      ylab = "Count",
+      xlim = c(40, 120),
+      ylim = c(0, 100),
+      breaks = seq(40, 120, by = 10))
```



Refernces:

- https://www.statmethods.net/advgraphs/layout.html
- https://stackoverflow.com/questions/31319942/change-the-size-of-a-plot-when-plotting-multiple-plots-in-r
- https://www.youtube.com/watch?v=Z3V4Pbxeahg

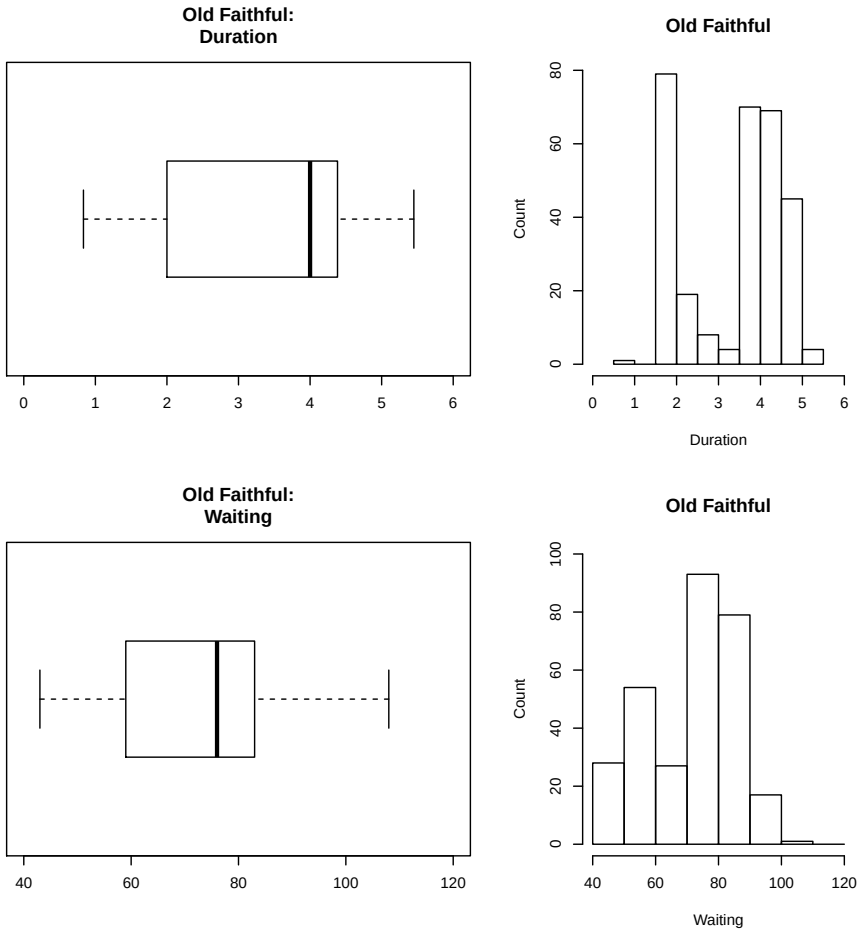
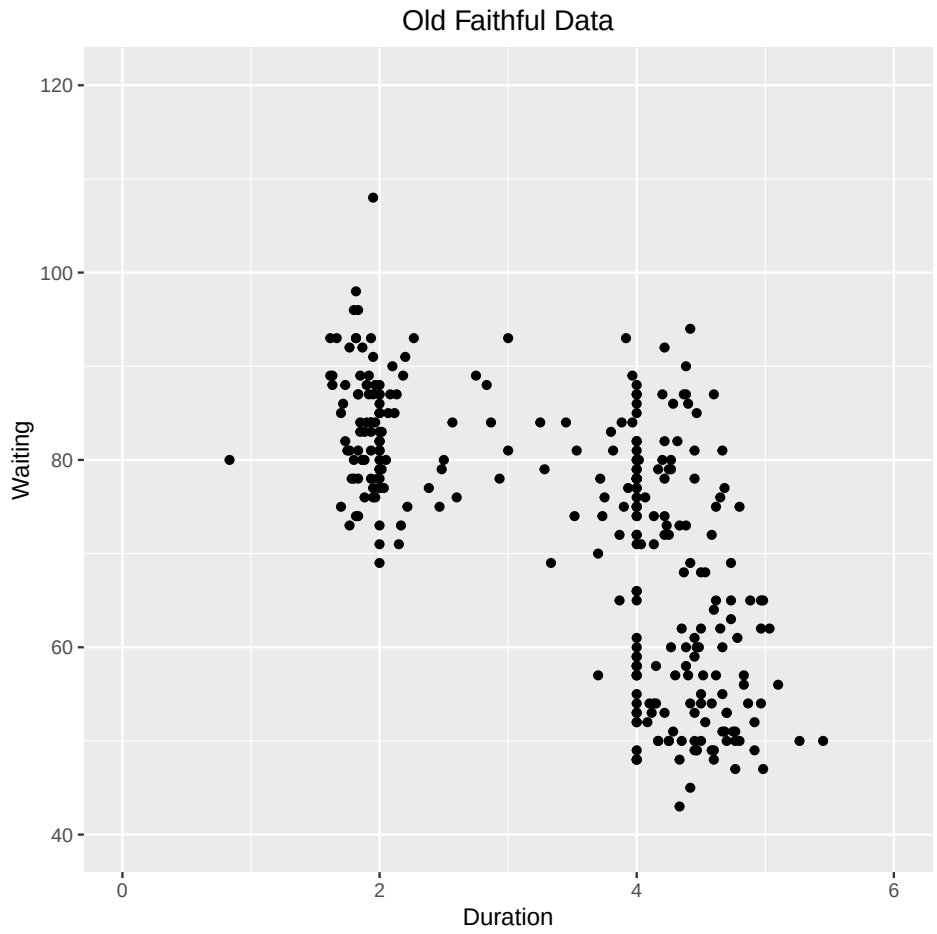


Figure 1: Graph created with *baseR*.

(b) (2 Points) Recreate the graph below using ggplot2. Include your R code and the resulting graph.

```
> ggplot(geyser, aes(x=duration, y=waiting)) +
+   geom_point() +
+   xlab("Duration") +
+   ylab("Waiting") +
+   xlim(0, 6) +
+   ylim(40, 120) +
+   ggtitle("Old Faithful Data") +
+   theme(plot.title = element_text(hjust = 0.5))
```


Old Faithful Data
120
100
80
60
40
Waiting
0
2
4
6
Duration

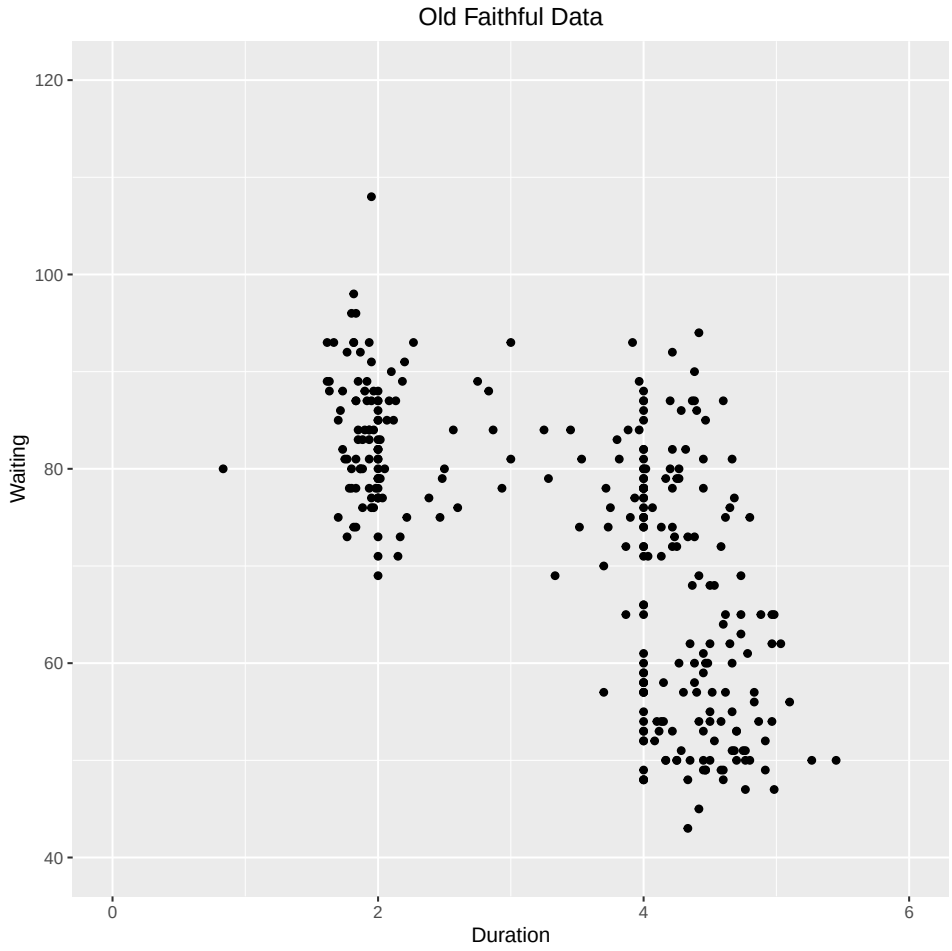
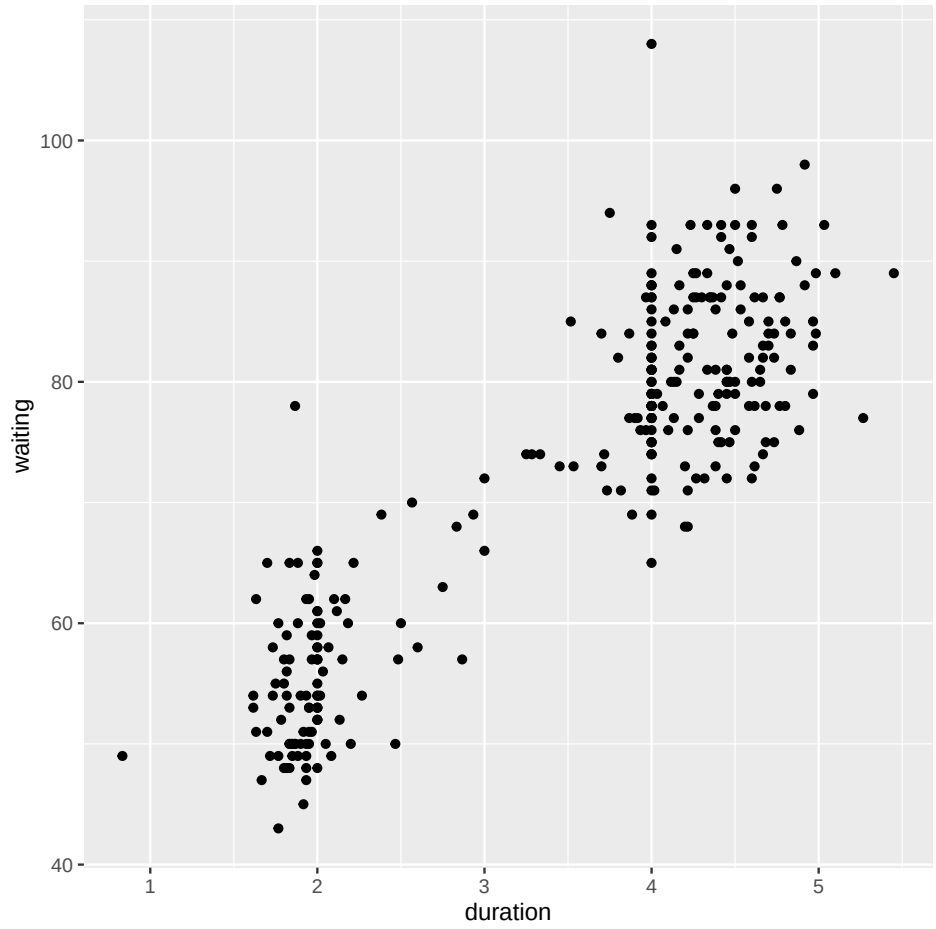


Figure 2: Graph created with *ggplot2*.

(c) (2 Points) Doesn't the scatterplot in (b) above look rather different than the scatterplot in Question 1 (j)? Note that the help page for *geyser* states `waiting numeric Waiting time for this eruption` and `The waiting time was incorrectly described as the time to the next eruption in the original files, and corrected for MASS version 7.3-30.` Use this information to create a basic scatterplot for the *geyser* data that matches the overall appearance in Question 1 (j). Include your R code and the resulting graph. No need to refine this scatterplot.

```
> duration <- geyser$duration[1:298]
> waiting <- geyser$waiting[2:299]
> df <- data.frame(duration, waiting)
> ggplot(df, aes(x = duration, y = waiting )) +
```

```
+            geom_point()
```



Answer: The geyser data appears to have a negative correlation with three clusters where the faithful data has a positive correlation with 2 clusters.

General Instructions

(i) Create a single pdf document, using R Markdown, Sweave, or knitr. You only have to submit this one document to Canvas.

(ii) Include a title page that contains your name, your A–number, the number of the assignment, the submission date, and any other relevant information.

(iii) Start your answers to each main question on a new page (continuing with the next part of a question on the same page is fine). Clearly label each question and question part. Your answer to question (i) should start on page 2!

(iv) Show your R code and resulting graph(s) for each question part!

(v) Before you submit your homework, check that you follow all recommendations from Google's R Style Guide (see `http://web.stanford.edu/class/cs109l/unrestricted/resources/google-style.html`). Moreover, make sure that your R code is consistent, i.e., that you use the same type of assignments and the same type of quotes throughout your entire homework.

(vi) Give credit to external sources, such as stackoverflow or help pages. Be specific and include the full URL where you found the help (or from which help page you got the information). Consider R code from such sources as "legacy code or third–party code" that does not have to be adjusted to Google's R Style (even though it would be nice, in particular if you only used a brief code segment).

(vii) **Not following the general instructions outlined above will result in point deductions!**

(viii) For general questions related to this homework, please use the corresponding discussion board in Canvas! I will try to reply as quickly as possible. Moreover, if one of you knows an answer, please post it. It is fine to refer to web pages and R commands, but do not provide the exact R command with all required arguments or which of the suggestions from a stackoverflow web page eventually worked for you! This will be the task for each individual student!

(ix) Submit your single pdf file via Canvas by the submission deadline. Late submissions will result in point deductions as outlined on the syllabus.