Histograms and Density Plots*

Alan T. Arnholt
Department of Mathematical Sciences
Appalachian State University
arnholt@math.appstate.edu

Spring 2006 R Notes

Histograms
Overview of Histograms

Density Plots
Overview of Density Plots

Stem-and-Leaf Plots
Overview of Stem-and-Leaf Plots

Problem
Application

The R Script

Histograms

The **histogram** is a graphical means of illustrating numerical data. Although the barplot and the histogram look similar, the barplot is used for categorical data, while the histogram is used for numerical data. Yet, the bins that either the user specifies or those that R uses by default are in essence categories. Clearly, we can always make quantitative data categorical; however, the reverse is not true.

Using the R function `hist()`

- Histograms created in R with the function `hist(x)` where `x` is a numeric vector are by default frequency histograms.

Using the R function `hist()`

- Histograms created in R with the function `hist(x)` where `x` is a numeric vector are by default frequency histograms.
- To create relative frequency histograms, use the optional argument `prob=TRUE`.

Using the R function `hist()`

- Histograms created in R with the function `hist(x)` where `x` is a numeric vector are by default frequency histograms.
- To create relative frequency histograms, use the optional argument `prob=TRUE`.
- For for information on `hist()`, type `?hist` at the R prompt.

The R function `density()`

The R function `density(x)`, where `x` is a numeric vector, can be used to create a density estimate. The user can optionally specify kernels other than the default Gaussian. The result of the density estimate can be viewed with either the `plot()` or `lines()` function.

Using `plot()` and `lines()`

- The `plot()` is a high-level function.

Using `plot()` and `lines()`

- The `plot()` is a high-level function.
- The `lines()` is a low-level function.

Using `plot()` and `lines()`

- The `plot()` is a high-level function.
- The `lines()` is a low-level function.

That is, `plot()` will create a graph, while `lines()` will add to an existing graph.

The R function `stem()`

- One way to get a quick impression of the data is to use a **stem-and-leaf plot**. When a stem-and-leaf plot is constructed, each observation is split into a stem and a leaf.

The R function `stem()`

- One way to get a quick impression of the data is to use a **stem-and-leaf plot**. When a stem-and-leaf plot is constructed, each observation is split into a stem and a leaf.
- Regardless of where the observation is split, the leaf in a stem-and-leaf plot is represented with a single digit.

The R function `stem()`

- One way to get a quick impression of the data is to use a **stem-and-leaf plot**. When a stem-and-leaf plot is constructed, each observation is split into a stem and a leaf.
- Regardless of where the observation is split, the leaf in a stem-and-leaf plot is represented with a single digit.
- Although it is possible to use a stem-and-leaf plot with a moderately sized data set (more than 100 values), the plot becomes increasingly hard to read as the number of values plotted increases. Consequently, it is recommended that stem-and-leaf plots be used graphically to illustrate smallish data sets (less than 100 values).

The R function `stem()`

- One way to get a quick impression of the data is to use a **stem-and-leaf plot**. When a stem-and-leaf plot is constructed, each observation is split into a stem and a leaf.
- Regardless of where the observation is split, the leaf in a stem-and-leaf plot is represented with a single digit.
- Although it is possible to use a stem-and-leaf plot with a moderately sized data set (more than 100 values), the plot becomes increasingly hard to read as the number of values plotted increases. Consequently, it is recommended that stem-and-leaf plots be used graphically to illustrate smallish data sets (less than 100 values).

Using `stem()`

- `stem(x)` creates a stem-and-leaf plot

Using `stem()`

- `stem(x)` creates a stem-and-leaf plot
- `x` must be a numeric vector

Using `stem()`

- `stem(x)` creates a stem-and-leaf plot
- `x` must be a numeric vector
- `scale` controls plot length

Using `stem()`

- `stem(x)` creates a stem-and-leaf plot
- `x` must be a numeric vector
- `scale` controls plot length
- `stem(x,scale=2)` produces plot roughly twice as long as default

Stem-and-Leaf Example

```
> library(BSDA)
> attach(Entrance)
> stem(score)

  The decimal point is 1 digit(s) to the right of the |

  4 | 38
  5 | 589
  6 | 2346689
  7 | 345579
  8 | 12346
  9 | 1
```

Stem-and-Leaf Example Continued

```
> stem(score,scale=2)

  The decimal point is 1 digit(s) to the right of the |

  4 | 3
  4 | 8
  5 |
  5 | 589
  6 | 234
  6 | 6689
  7 | 34
  7 | 5579
  8 | 1234
  8 | 6
  9 | 1
```

Problem

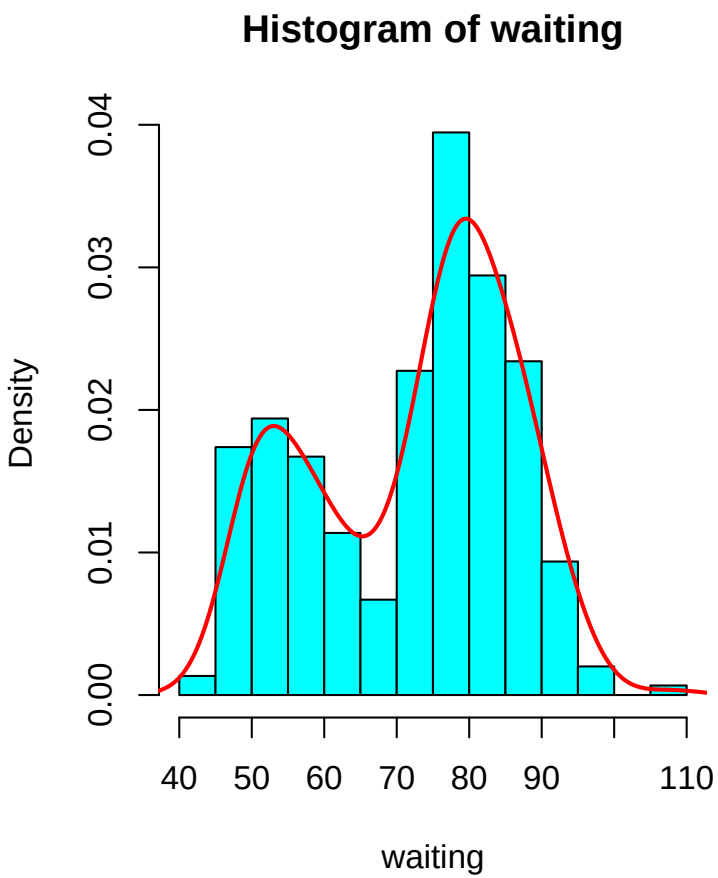
Construct a relative frequency histogram of the waiting time until the next eruption using the data frame `geyser` available in the `MASS` library. Superimpose a density estimate over the relative frequency histogram. In the same graph, show the estimated density without showing the histogram.

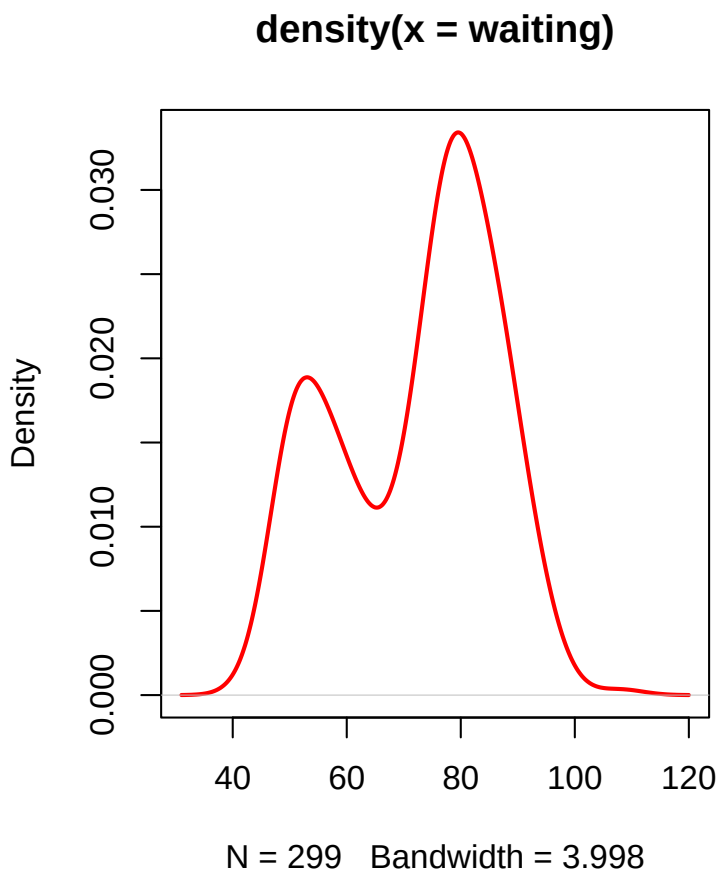
Solution

Note that to superimpose a density over a histogram, the histogram must be a relative frequency histogram. Recall that relative frequency histograms are produced with the optional argument `prob=TRUE`.

```
> library(MASS)
> par(mfrow=c(1,2))          # Make device region 1 by 2
> attach(geyser)
> hist(waiting,prob=TRUE)
> lines(density(waiting)) # Add density to Histogram
> plot(density(waiting))  # Create density by itself
```

The Graphs


Histogram of waiting
Density
0.00
0.01
0.02
0.03
0.04
40
50
60
70
80
90
110
waiting


density(x = waiting)
Density
0.000
0.010
0.020
0.030
40
60
80
100
120
N = 299 Bandwidth = 3.998

Can you get colors in your graphs?

1. Use the argument `col="somecolor"` to change the color of the histogram.

Can you get colors in your graphs?

1. Use the argument `col="somecolor"` to change the color of the histogram.
2. Use the argument `col="somecolor"` to change the density line color.

Can you get colors in your graphs?

1. Use the argument `col="somecolor"` to change the color of the histogram.
2. Use the argument `col="somecolor"` to change the density line color.
3. To change the default titles use the argument `main="YOU TYPE SOMETHING HERE"`.

Can you get colors in your graphs?

1. Use the argument `col="somecolor"` to change the color of the histogram.
2. Use the argument `col="somecolor"` to change the density line color.
3. To change the default titles use the argument `main="YOU TYPE SOMETHING HERE"`.

The actual R code used for the graphs

```
> par(mfrow=c(1,2))        # Make device region 1 by 2
> hist(waiting,prob=TRUE,col="cyan",
+ main="You Type Something Here")
> lines(density(waiting),lwd=3,col="red")
> plot(density(waiting),lwd=3,col="red",main="")
> title(main="Density Plot")
> par(mfrow=c(1,1))
```

Link to the R Script

- Go to my web page Script for Histograms and Density Plots
- Homework: problems 1.45-1.50, 1.54-1.56
- See me if you need help!