YSU ASDS, Statistics, Fall 2019

Lecture 04

Michael Poghosyan

09 Sep 2019

Descriptive Statistics

Contents

Last Lecture ReCap

- What is a **Frequency Table**?

Last Lecture ReCap

- What is a **Frequency Table**?
- What is the Definition of the **ECDF**?

Last Lecture ReCap

- What is a **Frequency Table**?
- What is the Definition of the **ECDF**?
- What is it for?

Histograms

Now we want to estimate the PDF of the RV behind our Data, we want to get the *shape* of the Distribution.

Histograms

Now we want to estimate the PDF of the RV behind our Data, we want to get the *shape* of the Distribution.We assume that our 1D dataset $x_1, ..., x_n$ is numerical, coming from an either Discrete or a Continuous Variable.

Histograms

Now we want to estimate the PDF of the RV behind our Data, we want to get the *shape* of the Distribution.We assume that our 1D dataset $x_1, ..., x_n$ is numerical, coming from an either Discrete or a Continuous Variable.

Barplot or LinePlot can help us in some cases, but if we have Continuous Variable, or a Discrete variable with many distinct values, then Barplot/LinePlot will not give the required approximation.

Histograms

Now we want to estimate the PDF of the RV behind our Data, we want to get the *shape* of the Distribution.We assume that our 1D dataset $x_1, ..., x_n$ is numerical, coming from an either Discrete or a Continuous Variable.

Barplot or LinePlot can help us in some cases, but if we have Continuous Variable, or a Discrete variable with many distinct values, then Barplot/LinePlot will not give the required approximation. So people use Histograms.

Histograms

Now we want to estimate the PDF of the RV behind our Data, we want to get the *shape* of the Distribution.We assume that our 1D dataset $x_1, ..., x_n$ is numerical, coming from an either Discrete or a Continuous Variable.

Barplot or LinePlot can help us in some cases, but if we have Continuous Variable, or a Discrete variable with many distinct values, then Barplot/LinePlot will not give the required approximation. So people use Histograms.

To define the Histogram, first we divide the range of our Dataset into *class intervals* or *bins*:

Histograms

Now we want to estimate the PDF of the RV behind our Data, we want to get the *shape* of the Distribution.We assume that our 1D dataset $x_1, ..., x_n$ is numerical, coming from an either Discrete or a Continuous Variable.

Barplot or LinePlot can help us in some cases, but if we have Continuous Variable, or a Discrete variable with many distinct values, then Barplot/LinePlot will not give the required approximation. So people use Histograms.

To define the Histogram, first we divide the range of our Dataset into *class intervals* or *bins*:

- we take first the range: either $I = [\min_i\{x_i\}, \max_i\{x_i\}]$ or I is an interval containing $[\min_i\{x_i\}, \max_i\{x_i\}]$;

Histograms

- we take a finite partition of I: $I_1, I_2, ..., I_k$, i.e. I_j-s are disjoint, and their union is the interval I;

Histograms

- we take a finite partition of I: $I_1, I_2, ..., I_k$, i.e. I_j-s are disjoint, and their union is the interval I; Usually, the intervals I_j have equal legths.

[1] **R** is using the *right-endpoint* convention (i.e., right endpoint is included, but not the left one), by default.

Histograms

- we take a finite partition of I: $I_1, I_2, ..., I_k$, i.e. I_j-s are disjoint, and their union is the interval I; Usually, the intervals I_j have equal legths. And we will assume that I_j includes its left endpoint but not the right endpoint (except the case when I_j is the rightmost interval - in that case I_j includes also the right endpoint)[1].

[1] **R** is using the *right-endpoint* convention (i.e., right endpoint is included, but not the left one), by default.

Histograms

- we take a finite partition of I: $I_1, I_2, ..., I_k$, i.e. I_j-s are disjoint, and their union is the interval I; Usually, the intervals I_j have equal legths. And we will assume that I_j includes its left endpoint but not the right endpoint (except the case when I_j is the rightmost interval - in that case I_j includes also the right endpoint)[1].
- we calculate the number n_j of datapoints x_i lying in I_j:

$$n_j = \text{the number of data points in } I_j \qquad j = 0, 1, 2, ..., k.$$

---

[1] **R** is using the *right-endpoint* convention (i.e., right endpoint is included, but not the left one), by default.

Histograms

Definition: The **frequency histogram** of our continuous (or a grouped) data $x_1, ..., x_n$ is the piecewise constant function

$$h_{freq}(x) = n_j, \qquad \forall x \in I_j, \quad j = 1, 2, ..., k.$$

Histograms

Definition: The **frequency histogram** of our continuous (or a grouped) data $x_1, ..., x_n$ is the piecewise constant function

$$h_{freq}(x) = n_j, \qquad \forall x \in I_j, \quad j = 1, 2, ..., k.$$

Frequency histogram shows the number of observations in our dataset in each bin, in each class interval. One also defines $h_{freq}(x) = 0$ for all $x \notin I$.

Example

airquality is a Dataset (standard Dataset in **R**) about the daily air quality measurements in New York, May to September 1973.

Example

airquality is a Dataset (standard Dataset in **R**) about the daily air quality measurements in New York, May to September 1973.

Here is the header:

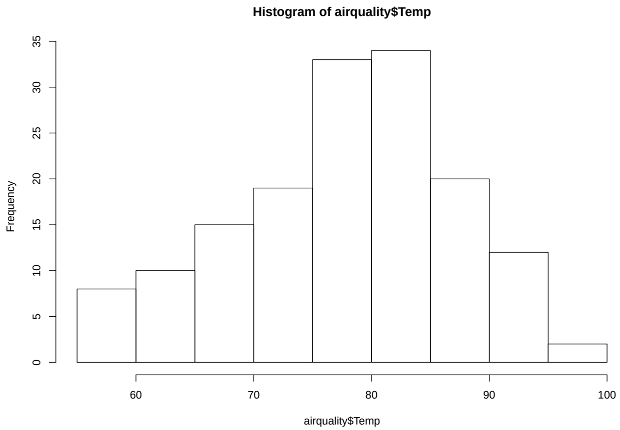
```r
head(airquality)
```

```
##   Ozone Solar.R Wind Temp Month Day
## 1    41     190  7.4   67     5   1
## 2    36     118  8.0   72     5   2
## 3    12     149 12.6   74     5   3
## 4    18     313 11.5   62     5   4
## 5    NA      NA 14.3   56     5   5
## 6    28      NA 14.9   66     5   6
```

Example

Let's Plot the histogram of the *Temp* (Temperature) Variable:

```
hist(airquality$Temp)
```



Notes on the Example

Some Notes:

Notes on the Example

Some Notes:

- **R**, by default, is choosing some appropriate bins;

Notes on the Example

Some Notes:

- **R**, by default, is choosing some appropriate bins;
- **R**'s *hist* command default bins have equal lengths;

Notes on the Example

Some Notes:

- **R**, by default, is choosing some appropriate bins;
- **R**'s *hist* command default bins have equal lengths;
- **R** is adding the default *OX* axis name and the Figure Title.

Histograms

Next is the Relative Frequency Histogram definition:

Definition The **relative frequency histogram** of our continuous data $x_1, ..., x_n$ is the piecewise constant function

$$h_{relfreq}(x) = \frac{n_j}{n}, \qquad \forall x \in I_j, \quad j = 1, 2, ..., k.$$

Histograms

Next is the Relative Frequency Histogram definition:

Definition The **relative frequency histogram** of our continuous data $x_1, ..., x_n$ is the piecewise constant function

$$h_{relfreq}(x) = \frac{n_j}{n}, \qquad \forall x \in I_j, \quad j = 1, 2, ..., k.$$

or, which is the same,

$$h_{relfreq}(x) = \frac{h_{freq}(x)}{n}, \qquad \forall x \in \mathbb{R}.$$

Histograms

Next is the Relative Frequency Histogram definition:

Definition The **relative frequency histogram** of our continuous data $x_1, ..., x_n$ is the piecewise constant function

$$h_{relfreq}(x) = \frac{n_j}{n}, \qquad \forall x \in I_j, \quad j = 1, 2, ..., k.$$

or, which is the same,

$$h_{relfreq}(x) = \frac{h_{freq}(x)}{n}, \qquad \forall x \in \mathbb{R}.$$

The Default **R** package has no Relative Frequency Histogram Plotting command (or I do not know $\ddot\smile$).

Histograms

Next is the Relative Frequency Histogram definition:

Definition The **relative frequency histogram** of our continuous data $x_1, ..., x_n$ is the piecewise constant function

$$h_{relfreq}(x) = \frac{n_j}{n}, \qquad \forall x \in I_j, \quad j = 1, 2, ..., k.$$

or, which is the same,

$$h_{relfreq}(x) = \frac{h_{freq}(x)}{n}, \qquad \forall x \in \mathbb{R}.$$

The Default **R** package has no Relative Frequency Histogram Plotting command (or I do not know $\ddot{\smile}$). But you can use, say, the *lattice* library's *histogram* command:

```
library(lattice)
histogram(airquality$Temp)
```

The Density or Normalized Relative Frequency Histogram

Next, and maybe the most important type of the Histogram is the Density Histogram:

The Density or Normalized Relative Frequency Histogram

Next, and maybe the most important type of the Histogram is the Density Histogram:

Definition: The **Density Histogram** or the **Normalized Relative Frequency Histogram** of our Data $x_1, ..., x_n$ is the piecewise constant function

$$h_{dens}(x) = \frac{n_j}{n} \cdot \frac{1}{length(I_j)}, \qquad \forall x \in I_j.$$

The Density or Normalized Relative Frequency Histogram

Next, and maybe the most important type of the Histogram is the Density Histogram:

Definition: The **Density Histogram** or the **Normalized Relative Frequency Histogram** of our Data $x_1, ..., x_n$ is the piecewise constant function

$$h_{dens}(x) = \frac{n_j}{n} \cdot \frac{1}{length(I_j)}, \qquad \forall x \in I_j.$$

Here $length(I_j)$ is the length of the interval I_j. Also we define $h(x) = 0$, if $x \notin I$.

Note

In the case (which is the mostly used one) when all intervals I_j have the same length:

$$length(I_j) = h,$$

then

Note

In the case (which is the mostly used one) when all intervals I_j have the same length:

$$length(I_j) = h,$$

then

$$h_{dens}(x) = \frac{h_{relfreq}(x)}{h} = \frac{n_j}{n \cdot h}, \qquad \forall x \in I_j.$$

Idea of the Density Histogram

The idea of dividing to the lenght of the corresponding interval, in the definition of the Density Histogram, is that in this case, the Total Area of all rectangles of our Histogram is 1.

Idea of the Density Histogram

The idea of dividing to the lenght of the corresponding interval, in the definition of the Density Histogram, is that in this case, the Total Area of all rectangles of our Histogram is 1.

Recall that all PDF functions integrate to 1.

Idea of the Density Histogram

The idea of dividing to the lenght of the corresponding interval, in the definition of the Density Histogram, is that in this case, the Total Area of all rectangles of our Histogram is 1.

Recall that all PDF functions integrate to 1. And the Density Histogram is approximating (estimating) the unknown PDF behind our Data!

Example

To draw the Density Histogram, we will use the *freq=FALSE* parameter in the *hist* command.

Example

To draw the Density Histogram, we will use the *freq=FALSE* parameter in the *hist* command.

We use here the *discoveries* Standard Dataset from **R**, which gives us the numbers of “great” inventions and scientific discoveries in each year from 1860 to 1959:

Example

To draw the Density Histogram, we will use the *freq=FALSE* parameter in the *hist* command.

We use here the *discoveries* Standard Dataset from **R**, which gives us the numbers of “great” inventions and scientific discoveries in each year from 1860 to 1959:

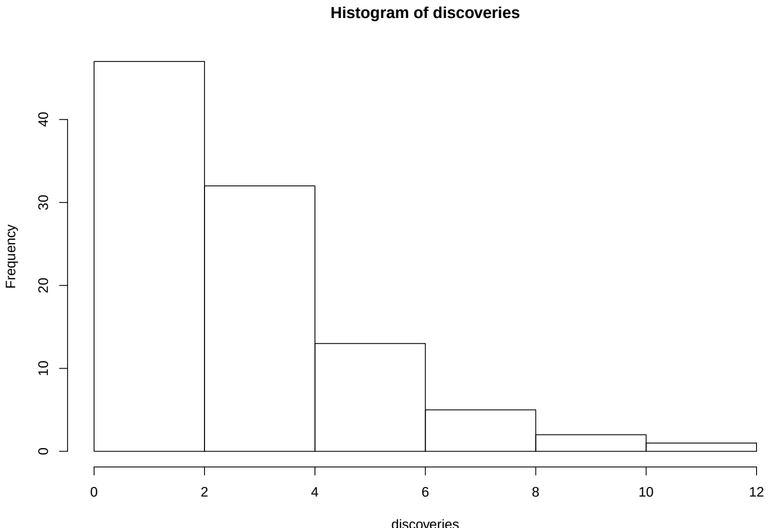
```
discoveries
```

```
## Time Series:
## Start = 1860
## End = 1959
## Frequency = 1
##   [1]  5  3  0  2  0  3  2  3  6  1  2  1  2  1  3  3
##  [24]  3  7 12  3 10  9  2  3  7  7  2  3  3  6  2  4
##  [47]  2  5  2  3  3  6  5  8  3  6  6  0  5  2  2  2
##  [70]  7  5  3  3  0  2  2  2  1  3  4  2  2  1  1  1
##  [93]  4  1  1  1  0  0  2  0
```

Example

First, the Frequency Histogram:

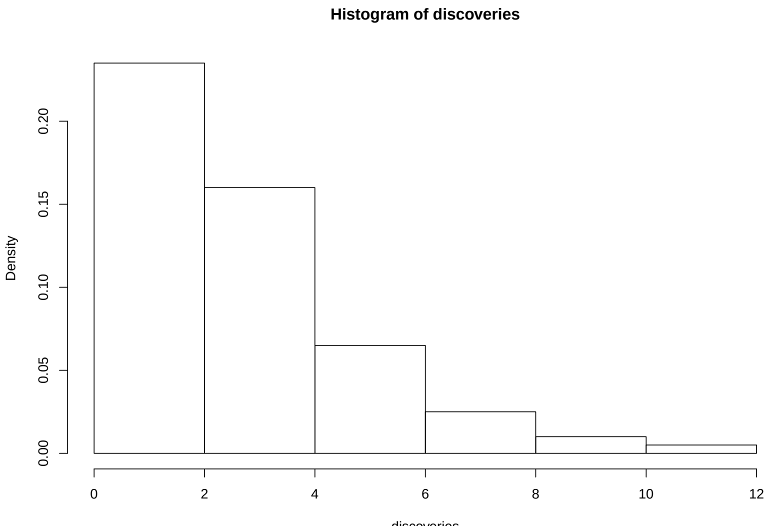
```r
hist(discoveries)
```



Example

Now, the Density Histogram:

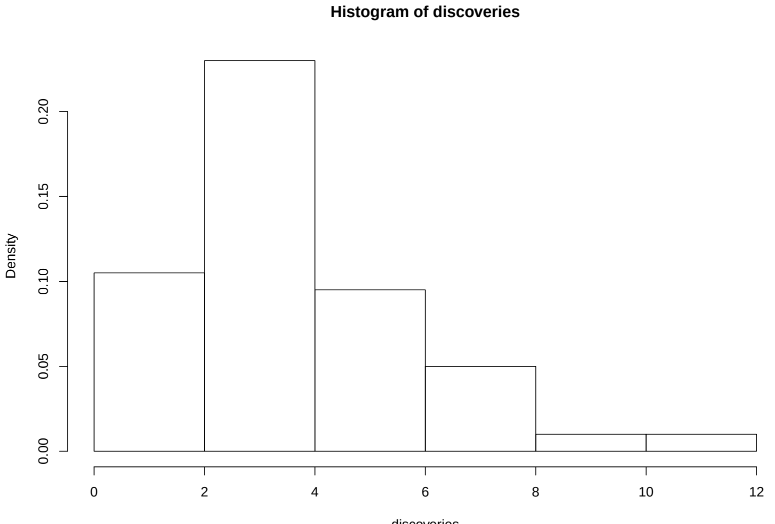
```
hist(discoveries, freq = FALSE)
```



Example

Finally, the Density Histogram with the Bins left-endpoints included:

```r
hist(discoveries, freq = FALSE, right = FALSE)
```



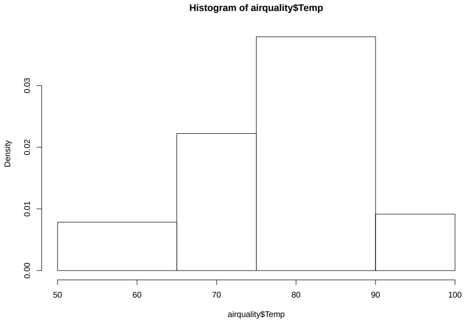
Example

Now let us change the default bins for a Histogram.

Example

Now let us change the default bins for a Histogram. We can use the following - first define the vector of our class interval (Bins) endpoints: (note that you need to cover all Datapoints!)

```
bins.endpoitns <- c(50, 65, 75, 90, 100)
hist(airquality$Temp, breaks = bins.endpoitns)
```



Notes

- By default, if we give custom bins with non-equal lengths, **R** is plotting the Density Histogram!

Notes

- By default, if we give custom bins with non-equal lengths, **R** is plotting the Density Histogram!
- You can give the *breaks* parameter either the vector of Bins' endpoints or the number of (equal-length) intervals

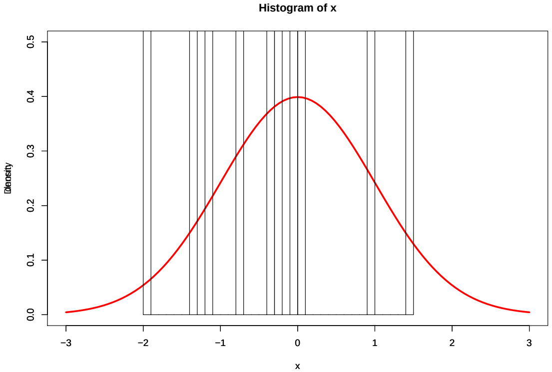
Estimation of the PDF through the Density Histogram

As it was stated above, the Density Histogram is an approximation (estimate) of the PDF of the Data unknown Distribution. To check this, let us take a synthetic Dataset from the Distribution we know:

Estimation of the PDF through the Density Histogram

As it was stated above, the Density Histogram is an approximation (estimate) of the PDF of the Data unknown Distribution. To check this, let us take a synthetic Dataset from the Distribution we know:

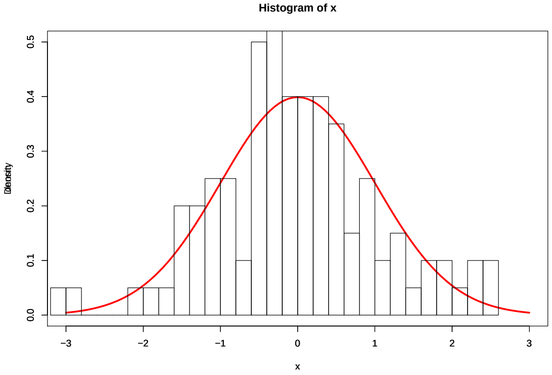
```
plot(dnorm, lwd = 3, col= "red", xlim=c(-3,3), ylim=c(0,0.5))
x <- rnorm(10)
par(new = TRUE)
hist(x, breaks = 40, freq = FALSE, xlim=c(-3,3), ylim=c(0,0.5))
```



Estimation of the PDF through the Density Histogram

As it was stated above, the Density Histogram is an approximation (estimate) of the PDF of the Data unknown Distribution. To check this, let us take a synthetic Dataset from the Distribution we know:

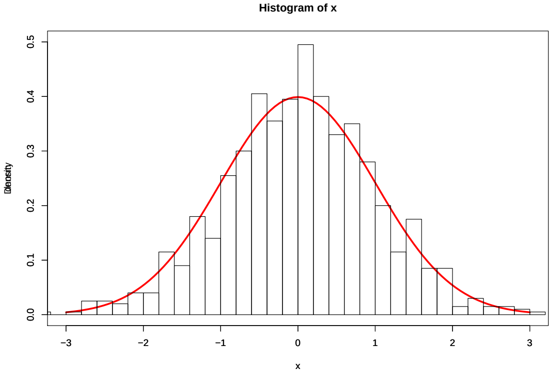
```
plot(dnorm, lwd = 3, col= "red", xlim=c(-3,3), ylim=c(0,0.5))
x <- rnorm(100)
par(new = TRUE)
hist(x, breaks = 40, freq = FALSE, xlim=c(-3,3), ylim=c(0,0.5))
```



Estimation of the PDF through the Density Histogram

As it was stated above, the Density Histogram is an approximation (estimate) of the PDF of the Data unknown Distribution. To check this, let us take a synthetic Dataset from the Distribution we know:

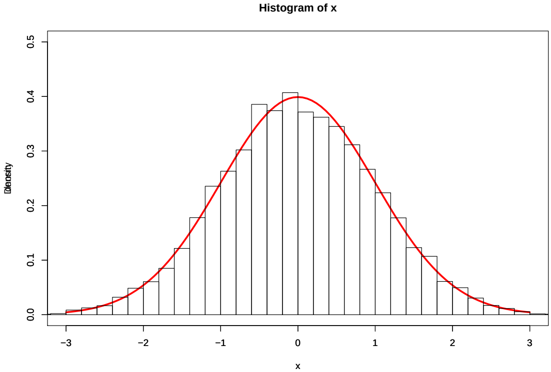
```
plot(dnorm, lwd = 3, col= "red", xlim=c(-3,3), ylim=c(0,0.5))
x <- rnorm(1000)
par(new = TRUE)
hist(x, breaks = 40, freq = FALSE, xlim=c(-3,3), ylim=c(0,0.5))
```



Estimation of the PDF through the Density Histogram

As it was stated above, the Density Histogram is an approximation (estimate) of the PDF of the Data unknown Distribution. To check this, let us take a synthetic Dataset from the Distribution we know:

```
plot(dnorm, lwd = 3, col= "red", xlim=c(-3,3), ylim=c(0,0.5))
x <- rnorm(10000)
par(new = TRUE)
hist(x, breaks = 40, freq = FALSE, xlim=c(-3,3), ylim=c(0,0.5))
```



Choosing Bin sizes correctly

It is important to choose the Bin sizes (lengths of the Bin, class, intervals) wisely. Otherwise you will skip some info or you will not get any valuable info.

Choosing Bin sizes correctly

It is important to choose the Bin sizes (lengths of the Bin, class, intervals) wisely. Otherwise you will skip some info or you will not get any valuable info.

Let us use another **R** standard dataset to show the effect of the choice of the bin size: *precip*. This Dataset shows the average amount of precipitation (rainfall) in inches for each of 70 United States (and Puerto Rico) cities.

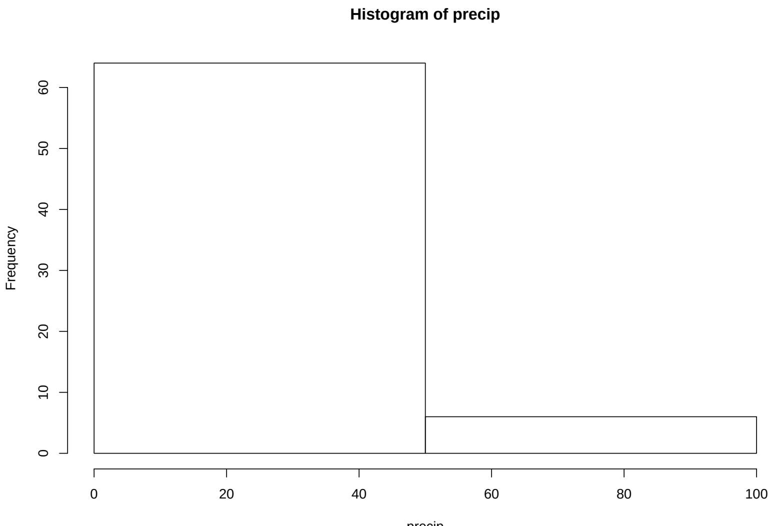
```
head(precip)
```

```
##      Mobile      Juneau     Phoenix Little Rock Los Ange
##        67.0        54.7         7.0        48.5        1
```

Version 1, Small bins

Here, we just use 2 bins:

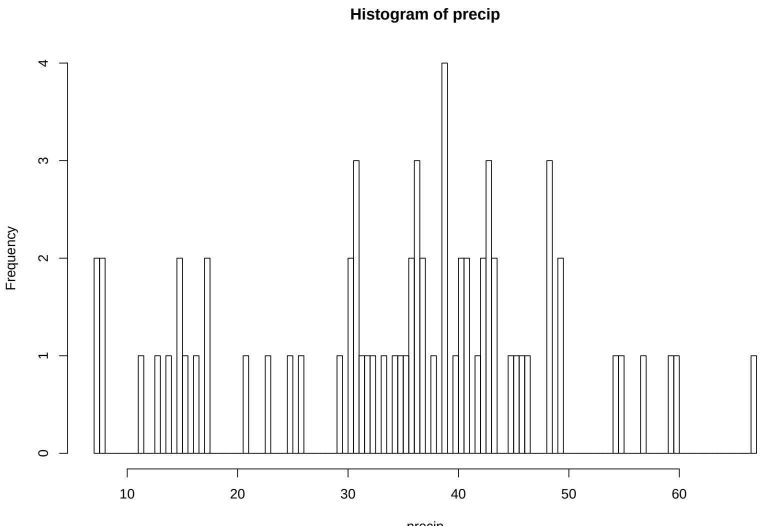
```r
hist(precip, breaks = 2)
```



Version 2, large bins

Here, we use 200 bins:

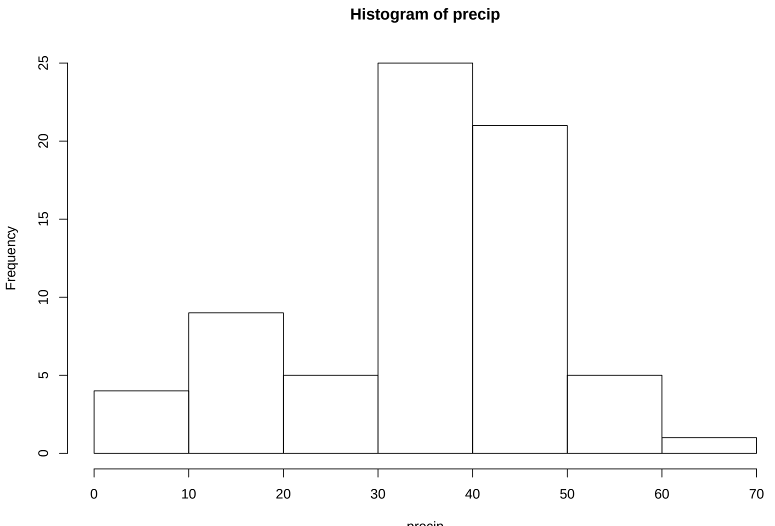
```
hist(precip, breaks = 200)
```



Version 2, large bins

Now, the default:

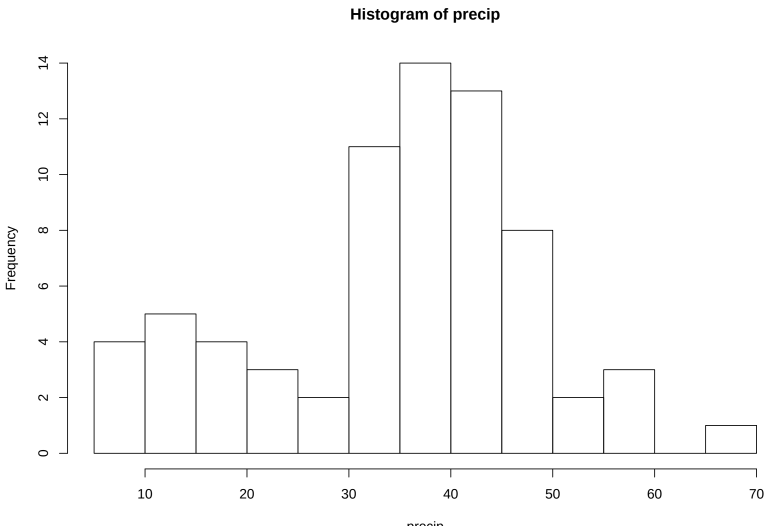
```r
hist(precip)
```



Version 3

Now, let us change to 20 bin intervals:

```
hist(precip, breaks = 20)
```



Choosing the Bin Length

In fact, choosing the correct Bin width is not an easy job. See, for example, the Histogram Wiki page.

Differences between the Barplot and Histogram

- Can you give some differences?

Differences between the Barplot and Histogram

- Can you give some differences?

Here are some:

- *Barplot*'s rectangles widths are arbitrary, do not mean anything, rectangles are not adjacent; *Histogram*'s rectangles are adjacent, and the choice of the Bin widths is changing the graph

Differences between the Barplot and Histogram

- Can you give some differences?

Here are some:

- *Barplot*'s rectangles widths are arbitrary, do not mean anything, rectangles are not adjacent; *Histogram*'s rectangles are adjacent, and the choice of the Bin widths is changing the graph
- *Barplot* is for a categorical or Discrete Data, *Histogram* is for both Discrete and Continuous

Differences between the Barplot and Histogram

- Can you give some differences?

Here are some:

- *Barplot*'s rectangles widths are arbitrary, do not mean anything, rectangles are not adjacent; *Histogram*'s rectangles are adjacent, and the choice of the Bin widths is changing the graph
- *Barplot* is for a categorical or Discrete Data, *Histogram* is for both Discrete and Continuous
- We can exactly reconstruct the Dataset from the *Barplot*, but not the *Histogram*

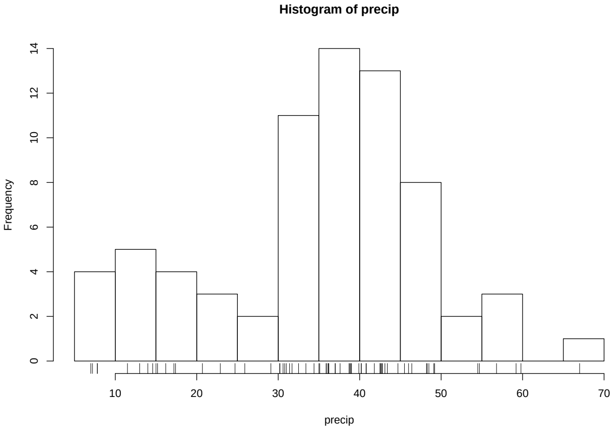
Addition to the Histogram

Nice addition to your Histogram Plot is to add, in some way, the Datapoints:

Addition to the Histogram

Nice addition to your Histogram Plot is to add, in some way, the Datapoints:

```r
hist(precip, breaks = 20)
rug(precip)
```



What we can see from the Histogram

If we will not look at the Histogram as being an estimate for the unknown Distribution behind the Data, and if we will just try to get some info about our Dataset, Histogram is helping us to say if the Data:

What we can see from the Histogram

If we will not look at the Histogram as being an estimate for the unknown Distribution behind the Data, and if we will just try to get some info about our Dataset, Histogram is helping us to say if the Data:

- is symmetric about some point or is skewed to the left or right

What we can see from the Histogram

If we will not look at the Histogram as being an estimate for the unknown Distribution behind the Data, and if we will just try to get some info about our Dataset, Histogram is helping us to say if the Data:

- is symmetric about some point or is skewed to the left or right
- is spread out or concentrated at some point

What we can see from the Histogram

If we will not look at the Histogram as being an estimate for the unknown Distribution behind the Data, and if we will just try to get some info about our Dataset, Histogram is helping us to say if the Data:

- is symmetric about some point or is skewed to the left or right
- is spread out or concentrated at some point
- has some gaps

What we can see from the Histogram

If we will not look at the Histogram as being an estimate for the unknown Distribution behind the Data, and if we will just try to get some info about our Dataset, Histogram is helping us to say if the Data:

- is symmetric about some point or is skewed to the left or right
- is spread out or concentrated at some point
- has some gaps
- has values far apart from others, has outliers (anomalies)

What we can see from the Histogram

If we will not look at the Histogram as being an estimate for the unknown Distribution behind the Data, and if we will just try to get some info about our Dataset, Histogram is helping us to say if the Data:

- is symmetric about some point or is skewed to the left or right
- is spread out or concentrated at some point
- has some gaps
- has values far apart from others, has outliers (anomalies)
- is unimodal, bimodal or multimodal

KDE

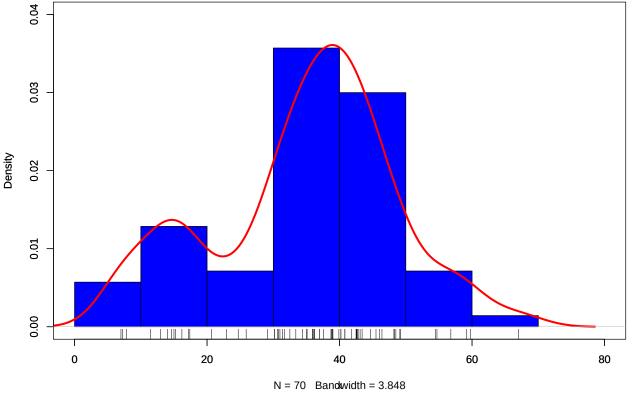
Another estimate for the unknown Distribution PDF is the **Kernel Density Estimator**, KDE.

KDE

Another estimate for the unknown Distribution PDF is the **Kernel Density Estimator**, KDE. It is, in some sense, the smoothed version of the Histogram: Histogram is a piecewise-constant function, with jumps, so it is not a smooth function.

KDE Example

```r
x <- precip; d <- density(x)
hist(x, freq = FALSE, xlim = c(0, 80), ylim = c(0,0.04),
     col = "blue", main = "")
rug(x); par(new = TRUE)
plot(d, lwd = 3, col = "red", xlim = c(0,80), ylim = c(0,0.04),
     main = "")
```



KDE

To define the KDE, we first choose a smooth Kernel function $K(t)$, here, a function with

$$K(t) \geq 0, t \in \mathbb{R}, \qquad \text{and} \qquad \int_{-\infty}^{+\infty} K(t)dt = 1.$$

KDE

To define the KDE, we first choose a smooth Kernel function $K(t)$, here, a function with

$$K(t) \geq 0, t \in \mathbb{R}, \qquad \text{and} \qquad \int_{-\infty}^{+\infty} K(t)dt = 1.$$

For example, we can take the Gaussian Kernel

$$K(t) = \frac{1}{\sqrt{2\pi}} \cdot e^{-t^2/2}, \qquad t \in \mathbb{R},$$

or any other PDF.

KDE

To define the KDE, we first choose a smooth Kernel function $K(t)$, here, a function with

$$K(t) \geq 0, t \in \mathbb{R}, \qquad \text{and} \qquad \int_{-\infty}^{+\infty} K(t)dt = 1.$$

For example, we can take the Gaussian Kernel

$$K(t) = \frac{1}{\sqrt{2\pi}} \cdot e^{-t^2/2}, \qquad t \in \mathbb{R},$$

or any other PDF.

Next, one defines the Kernel Density Estimator with Kernel K as

$$KDE_K(x) = KDE(x) = \frac{1}{nh} \cdot \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right).$$

KDE

It is easy to see that $KDE(x)$ will give a PDF, i.e., will be nonnegative and will integrate to 1:

$$\int_{-\infty}^{+\infty} KDE(x)dx =$$

KDE

It is easy to see that $KDE(x)$ will give a PDF, i.e., will be nonnegative and will integrate to 1:

$$\int_{-\infty}^{+\infty} KDE(x)dx = \frac{1}{nh} \cdot \sum_{i=1}^{n} \int_{-\infty}^{+\infty} K\left(\frac{x - x_i}{h}\right) dx =$$

$$= \frac{1}{n} \cdot \sum_{i=1}^{n} \int_{-\infty}^{+\infty} K\left(\frac{x - x_i}{h}\right) d\frac{x - x_i}{h} \overset{u=\frac{x-x_i}{h}}{=}$$

$$= \frac{1}{n} \cdot \sum_{i=1}^{n} \int_{-\infty}^{+\infty} K(u)du = \frac{1}{n} \cdot \sum_{i=1}^{n} 1 = 1.$$

KDE

Note: Like in the case of the Density histogram, where that histogram was depending on the bins choice, the KDE depends on the choice of $h > 0$. h is called the **bandwidth**, and its estimation is another story.