Contents

- Frequency and Relative Frequency Tables, their graphical representations
- ECDF
- Frequency, Relative Frequency and Density Histograms
- KDE

Frequency Tables

Here we assume that we have observations from a 1D numerical or categorical variable, i.e., we have a univariate *discrete* numerical or categorical data $x_1, x_2, ..., x_n$.

Definition: The **frequency** of a value t in observations $x_1, x_2, ..., x_n$ is the number of times t occurs in observations:

$$\text{Frequency of } t = \text{number of occurrences of } t \text{ in data.}$$

Definition: The **relative frequency** (or percentage) of a value t in observations $x_1, x_2, ..., x_n$ is the ratio of frequency of t divided by the total number of observations, n:

$$\text{Relative Frequency of } t = \frac{\text{Frequency of } t}{\text{Total Number of Observations}} =$$

$$= \frac{\text{Frequency of } t}{n}.$$

Frequency Tables, Example

Example: Given the following Dataset:

$$1, 2, 4, 7, 2, 3, 2, 1, 2, 1, 4, 1, -1$$

obtain the Frequency and Relative Frequency Tables.

Example: Let's construct the Frequency Table of the above Dataset using **R**:

```r
x <- c(1, 2, 4, 7, 2, 3, 2, 1, 2, 1, 4, 1, -1)
table(x)
```

```
## x
## -1  1  2  3  4  7
##  1  4  4  1  2  1
```

Visualizing Frequency and Relative Frequency Tables

Now, having the Frequency or the Relative Frequency Tables, we can visualize the Dataset by using a BarPlot (BarChart), PieChart, Line Graph or a Frequency Polygon.

Frequency Tables, Example

Now, consider the *iris* dataset in **R**:

```
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

Frequency Tables, Example, Cont'd

To get the *Species* Variable of the iris Dataset, we use

```
iris$Species
```

And to calculate the Frequency of each of the Species, we use

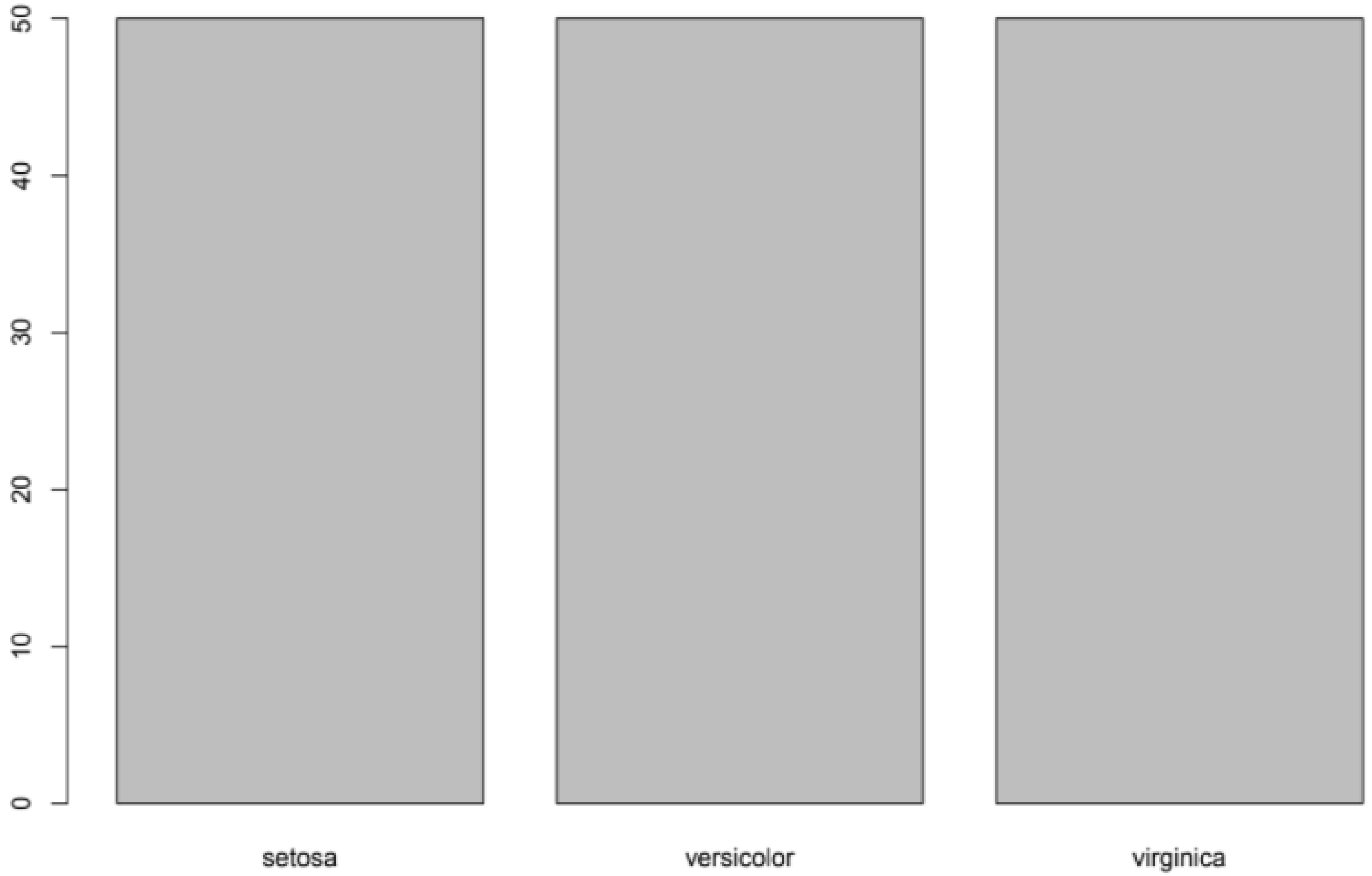
```
table(iris$Species)
```

```
##
##     setosa versicolor  virginica
##         50         50         50
```

BarPlot

Now, let us visualize our Frequency Table by using a BarPlot:

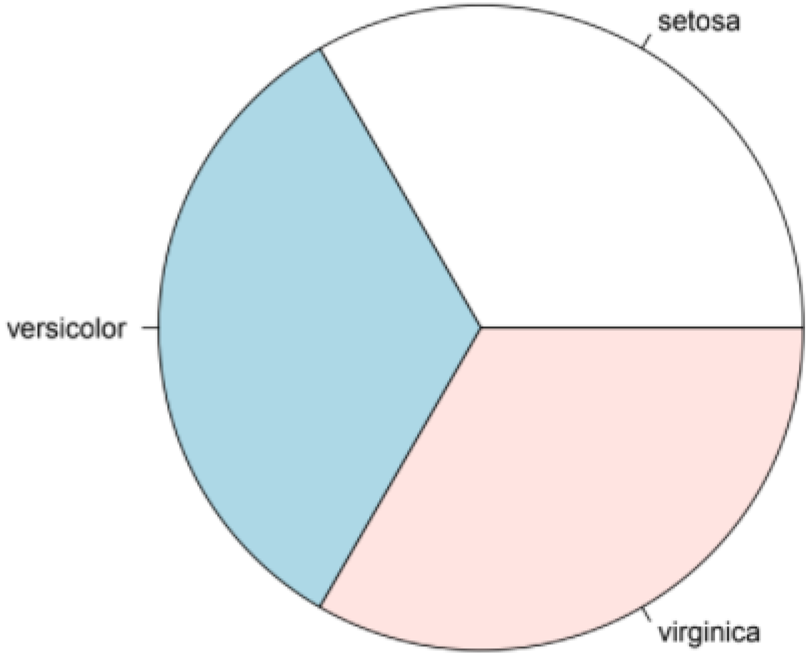
```
barplot(table(iris$Species))
```



PieChart

Also, we can visualize the same Frequency Table (or, in fact, the Relative Frequency Table) using a PieChart:

```
pie(table(iris$Species))
```



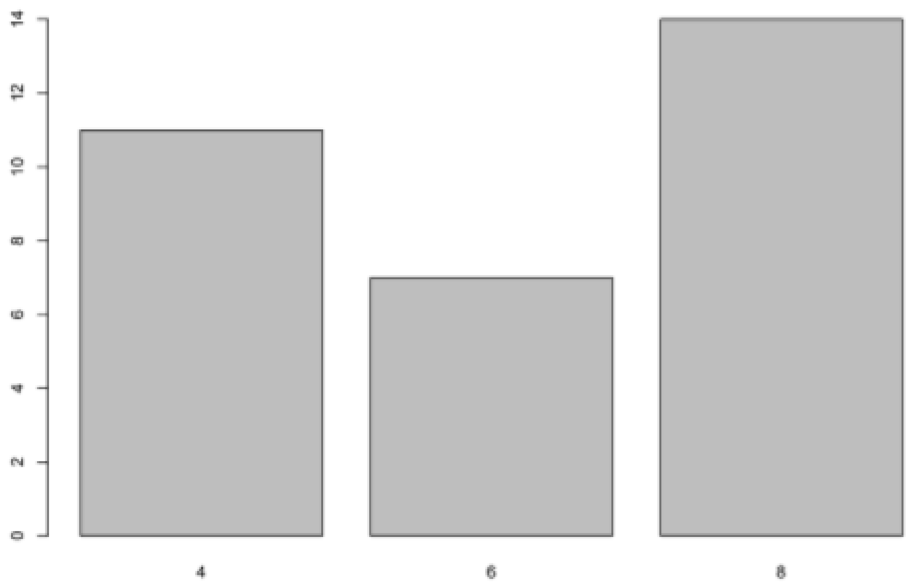
BarPlot

Another standard Dataset, *mtcars*, again about cars ⌣̈:

```
head(mtcars, 3)
```

```
##                mpg cyl disp  hp drat    wt  qsec vs am gear c
## Mazda RX4     21.0   6  160 110 3.90 2.620 16.46  0  1    4
## Mazda RX4 Wag 21.0   6  160 110 3.90 2.875 17.02  0  1    4
## Datsun 710    22.8   4  108  93 3.85 2.320 18.61  1  1    4
```

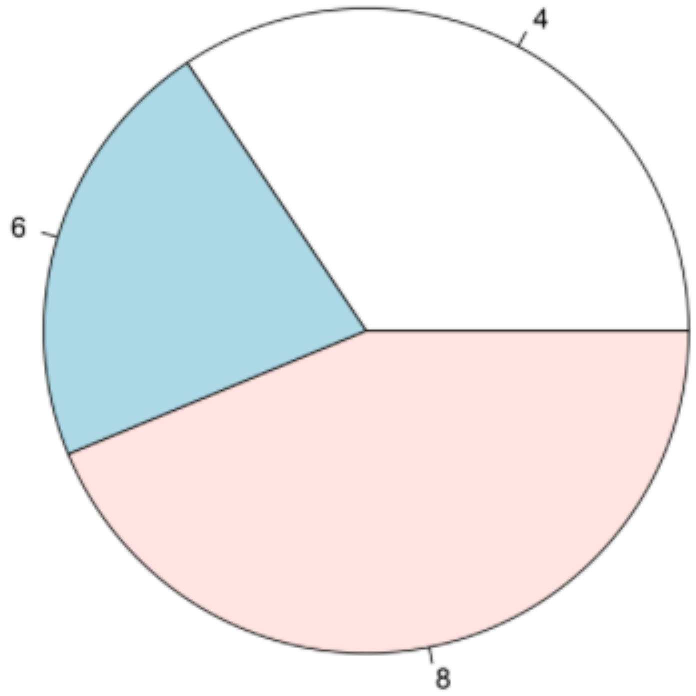
```
barplot(table(mtcars$cyl))
```



mtcars CYL with PieChart

The same, but with PieChart:

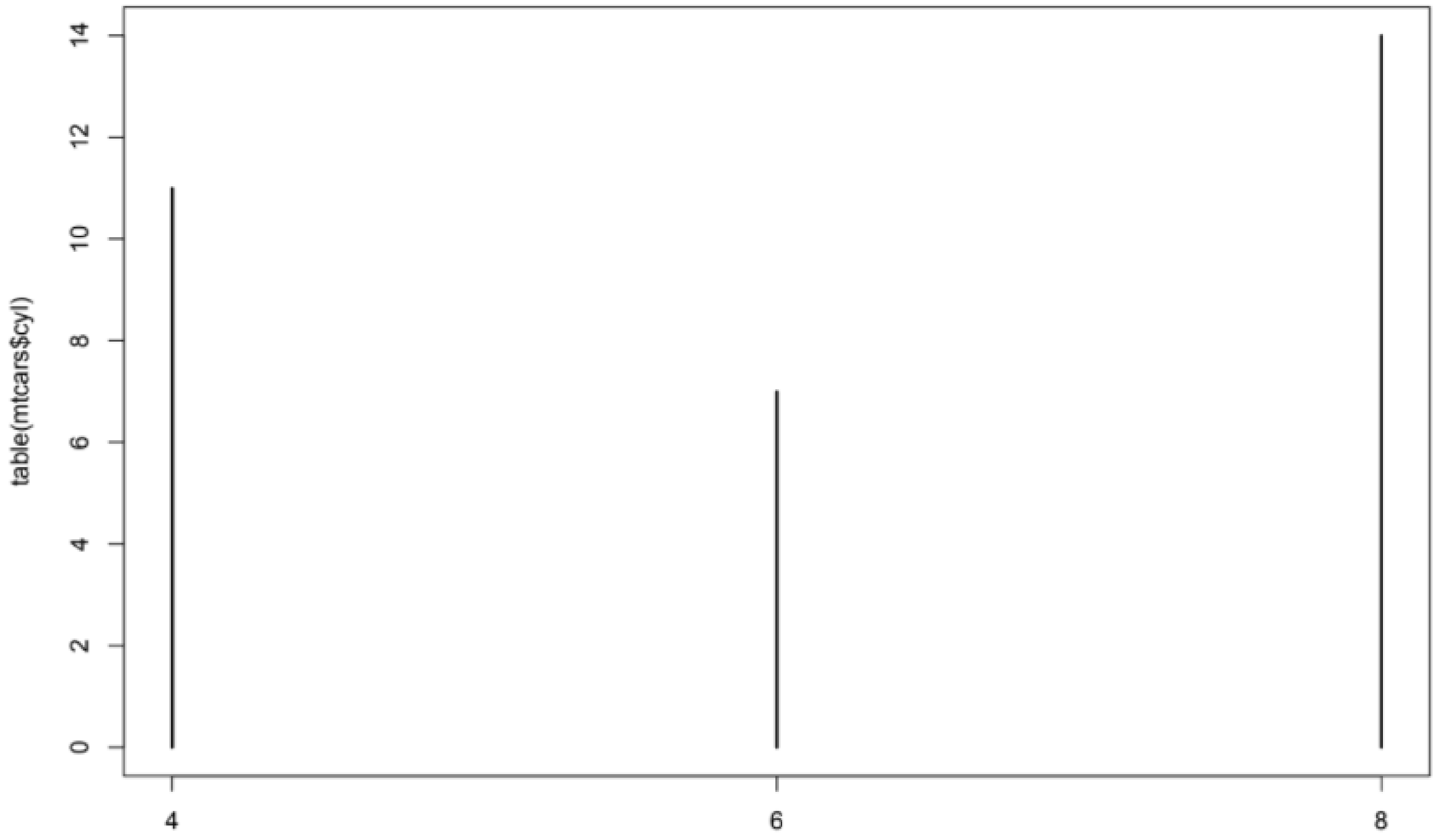
```
pie(table(mtcars$cyl))
```



LineGraph and Barplot

Now, with the Line Graph:

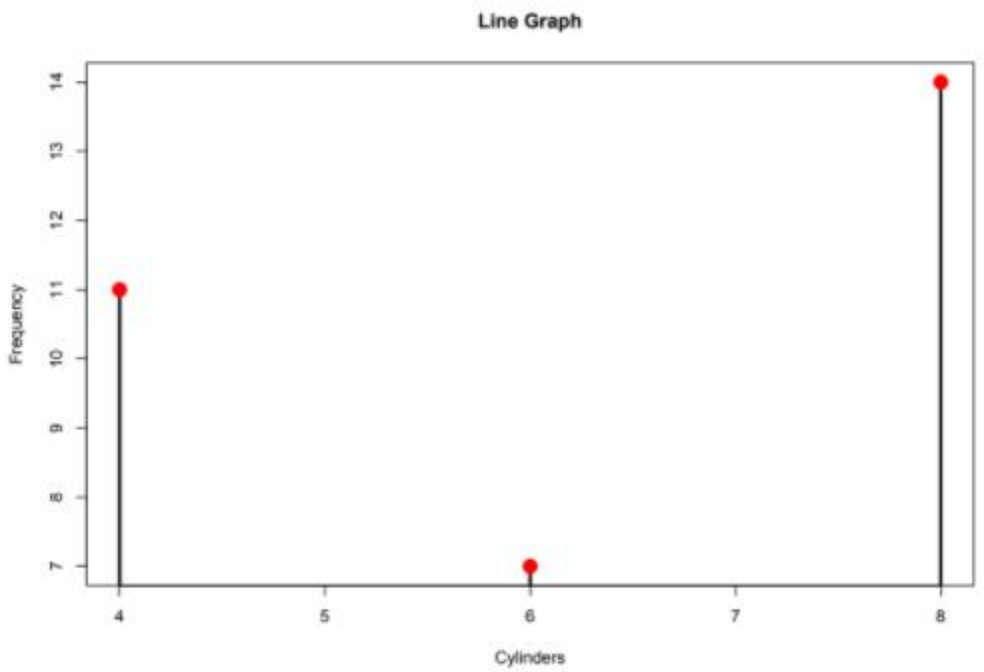
```
plot(table(mtcars$cyl))
```



LineGraph and Barplot

More sophisticated (titiz) version:

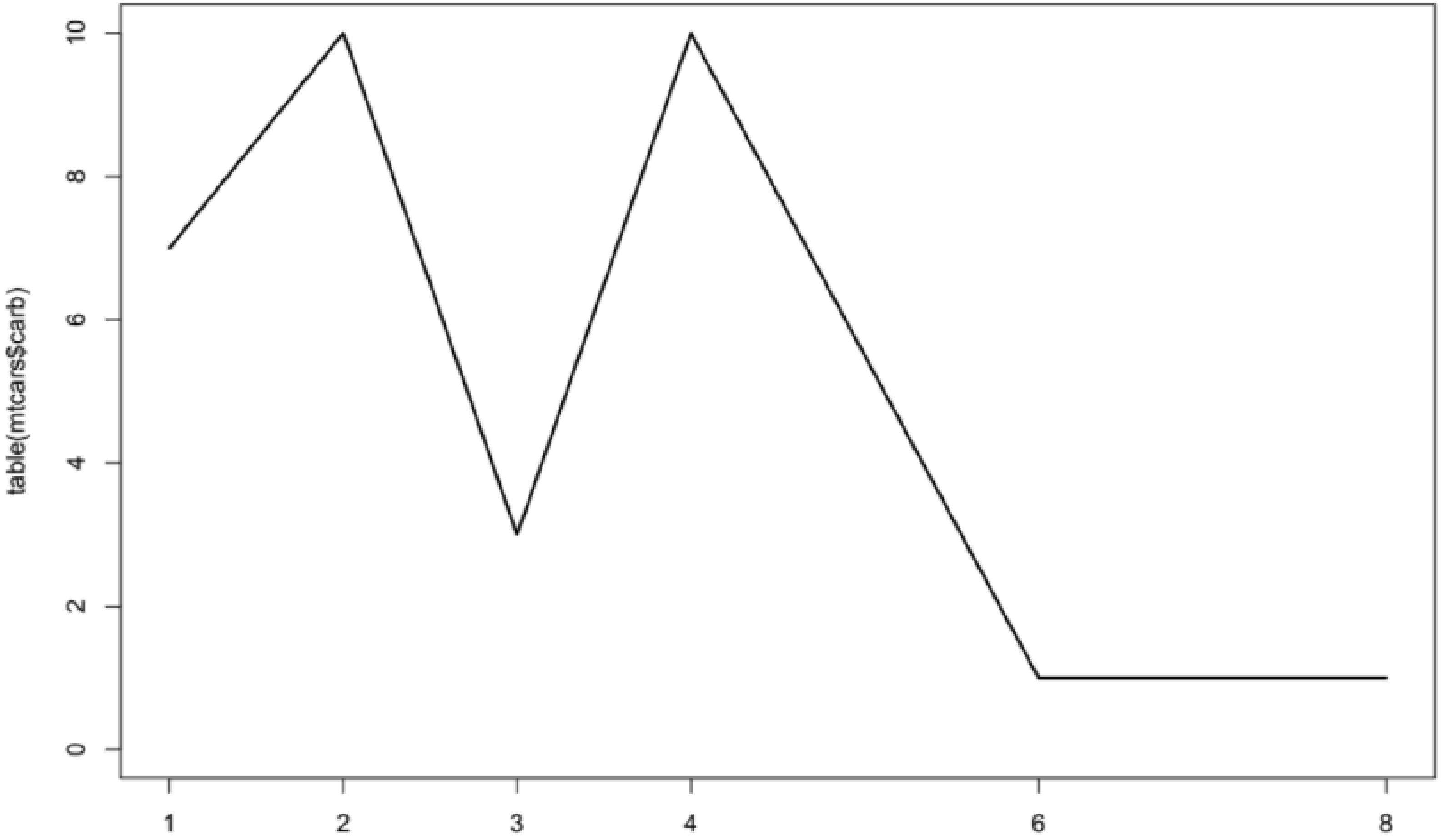
```
x <- mtcars$cyl; y <- as.data.frame(table(x))
a <- as.numeric(as.character(y$x)); b <- y$Freq
plot(a,b,type="h", lwd=3,  xlab = "Cylinders",
     ylab = "Frequency", main = "Line Graph")
points(a,b, pch=16, cex=2, col="red")
```



The Frequency Polygon

Again, same cars, but now the *carb* Variable Frequencies:

```
plot(table(mtcars$carb), type = "l")
```



Supplements

If our Dataset has more complex structure, say, we have categories, and categories can be separated by some groups, then we can use **Stacked** or **Grouped BarPlots** to visualize the Dataset.

Describing the Data Distribution

Assume we have a 1D numerical dataset x: $x_1, x_2, ..., x_n$. We assume that our dataset comes as a set of realizations of some Random Variable.

In Statistics, this is very common. We assume that there is some RV behind our observations, we do not know the Distribution of that RV, but we have some observations from that Distribution. And our aim is to find (estimate) that Distribution.

Say, when we talk about the height distribution of persons between the ages 20-30, we assume that there is some unknown process that generates that heights. And we assume *Height* is our r.v., and we have some observations from that r.v.

From the Probability course, we know two complete characteristics of a Random Variable: the **CDF and PD(M)F**. So to describe our Data Distribution, we can try to describe the CDF and/or PD(M)F behind the Data.

Empirical CDF

First let's estimate the CDF. We will estimate CDF by the Empirical CDF:

Definition: The **Empirical Distribution Function, ECDF** or the **Cumulative Histogram** $ecdf(x)$ of our data $x_1, ..., x_n$ is defined by

$$ecdf(x) = \frac{\text{number of elements in our dataset} \leq x}{\text{the total number of elements in our dataset}} =$$

$$= \frac{\text{number of elements in our dataset} \leq x}{n}, \qquad \forall x \in \mathbb{R}.$$

Example

Example: Construct the ECDF (analytically and graphically) of the following data:

$$-1, 4, 7, 5, 4$$

- Analytical Part - on the board

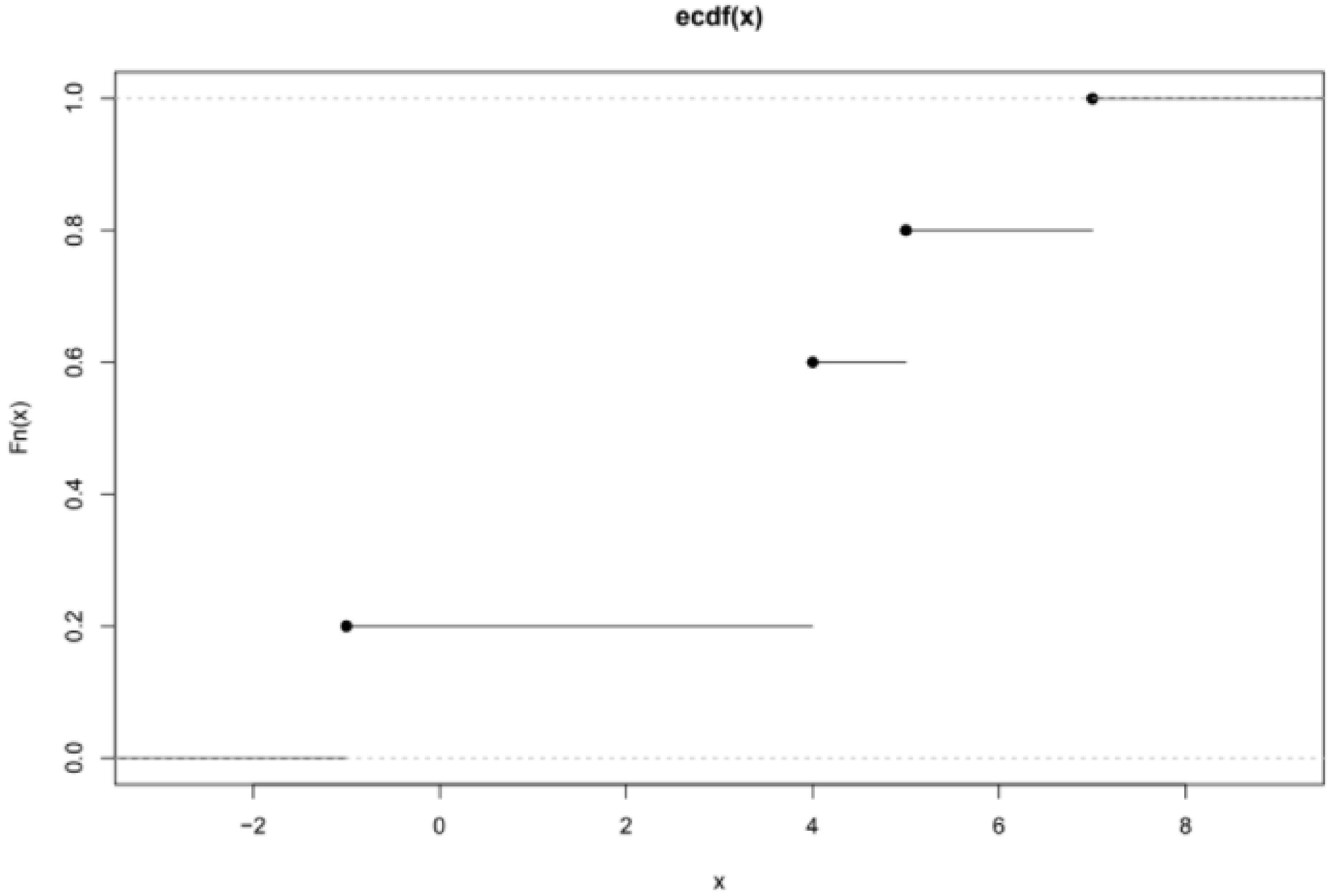
To do the graphical part, we

- Sort our Dataset from the lowest to the largest values
- Plot the Data points on the OX axis
- ECDF is 0 for values of x less that the smallest Datapoint, and is 1 for values of x bigger than the largest Datapoint
- For each Data point, calculate the Relative Frequency of that Datapoint (the number of times it occurs in our Dataset over the total number of Datapoints). At that Datapoint, do a Jump of the size of the Relative Frequency, and draw a horizontal line up to the next Datapoint.

Example

Now, using **R**:

```
x <-c(-1, 4, 7, 5, 4)
f <- ecdf(x)
plot(f)
```



Note: It is easy to see that the ECDF satisfies all properties of a CDF.

Note: It is easy to see that the ECDF for a Dataset

$$-1, 4, 7, 5, 4$$

coincides with the CDF of a r.v.

X	-1	4	5	7
$\mathbb{P}(X = x)$	$\frac{1}{5}$	$\frac{2}{5}$	$\frac{1}{5}$	$\frac{1}{5}$

Glivenko-Cantelli Theorem

How do we know that the ECDF is representing (estimating) the unknown CDF behind the Data good enough?

Well, this was proved by Glivenko and Cantelli: if our data $x_1, ..., x_n$ comes from the Distribution with the CDF $F(x)$,and if we will denote by $F_n(x)$ the ECDF constructed for $x_1, ..., x_n$, then

$$F_n(x) \to F(x) \quad \text{uniformly on } \mathbb{R}.$$

This Theorem says that if you will have enough datapoints from a Distribution, you can approximate the unknown CDF of your Distribution pretty well by using the ECDF.

Above, we need to be more precise about in which sense the convergence holds.

Glivenko-Cantelli Theorem

In fact, the following Theorem Holds:

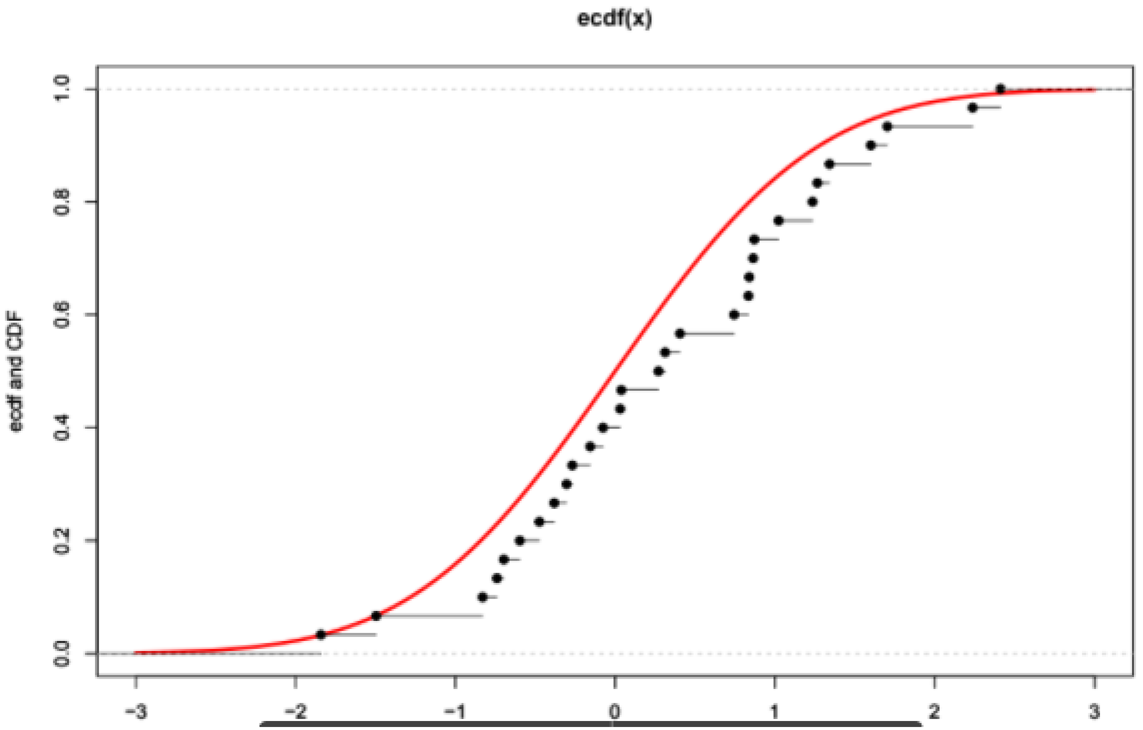
Theorem (Glivenko, Cantelli): If $X_1, ..., X_n$ are IID r.v.s from the Distribution with the CDF $F(x)$, and $F_n(x)$ is the ECDF constructed by using $X_1, ..., X_n$, then

$$\sup_x |F_n(x) - F(x)| \to 0 \qquad a.s.$$

Estimation of the CDF through ECDF

Let us check this theorem using **R**:

```
plot(pnorm, lwd = 3, col = 'red', xlim = c(-3,3),
     ylim = c(0,1), ylab = "ecdf and CDF")
n <- 30 ; x <- rnorm(n) #Taking a sample of size n from N(0,1)
f <- ecdf(x) #f will be the ECDF of our data x
par(new = TRUE) #this is to keep the previous graph
plot(f, xlim = c(-3,3), ylim = c(0,1), ylab = "ecdf and CDF")
```



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; 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:

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

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.

Here is the header:

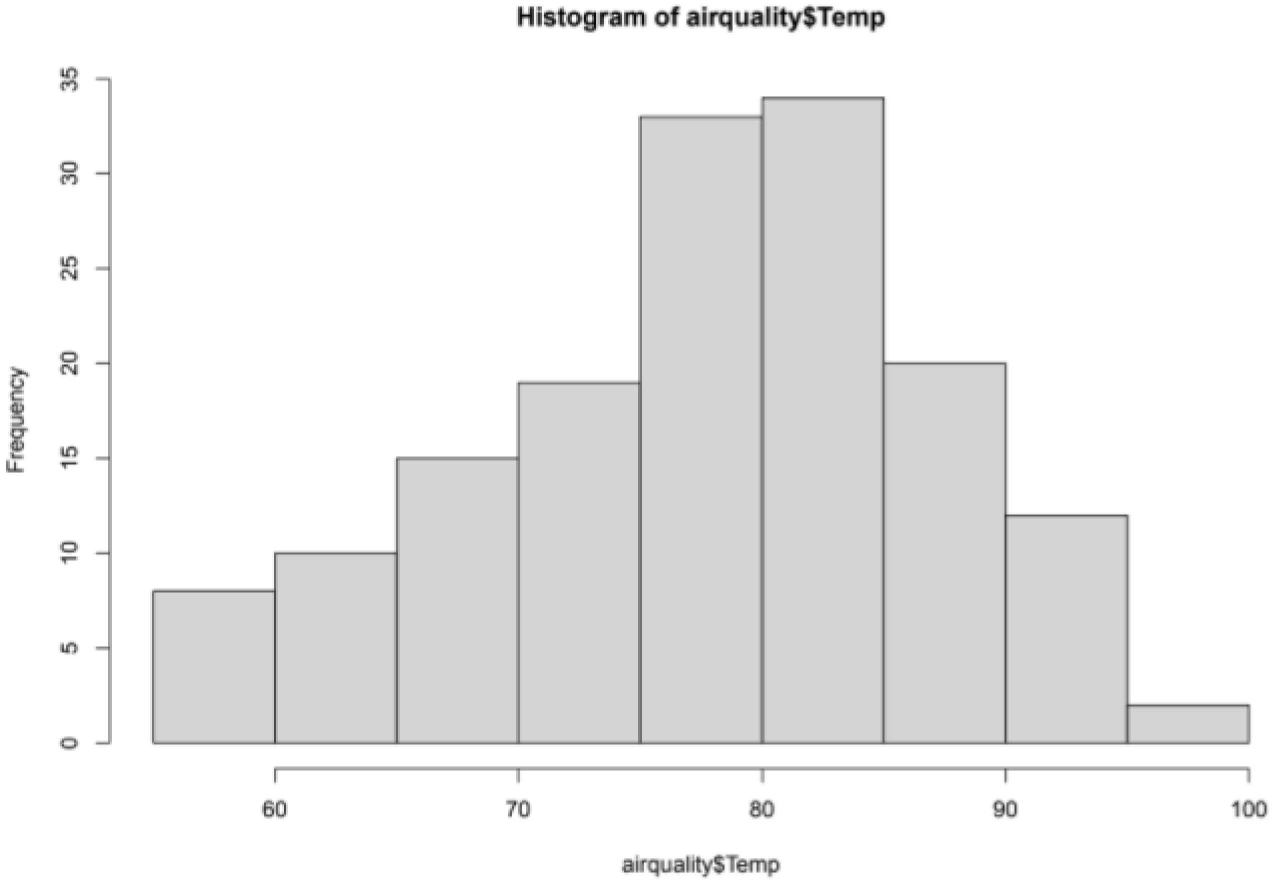
```
head(airquality)
```

```
##   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:

- **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

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

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:

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 $h_{dens}(x) = 0$, 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

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

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.

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