ASDS Statistics, YSU, Fall 2020

Lecture 04

Michael Poghosyan

23 Sep 2020

Contents

- Histogram

Last Lecture Recap

- For which type of variables BarPlot will work?

Last Lecture Recap

- For which type of variables BarPlot will work?
- Give the Definition of the Empirical CDF.

Last Lecture Recap

- For which type of variables BarPlot will work?
- Give the Definition of the Empirical CDF.
- State the Glivenko-Cantelli theorem.

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: $I_1, I_2, \ldots, I_k$, 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:

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

[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:

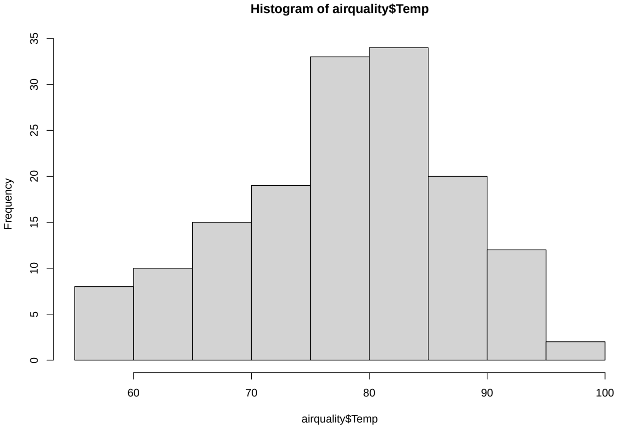
```r
head(airquality)
```

```
##   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:

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

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!