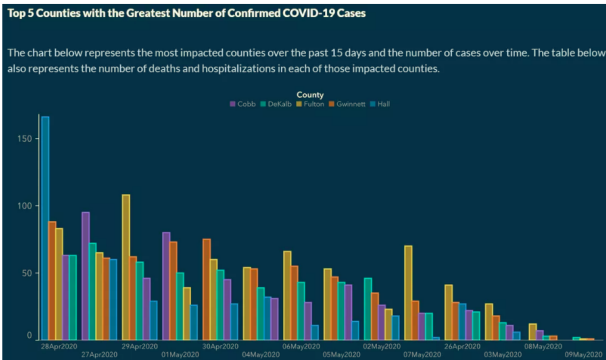


# Quantitative Variables

Dennis Sun (modified by Jonathan Taylor)  
Stanford University

**DATASCI 112**

April 3, 2026



# Quantitative Variables

We have analyzed a quantitative variable already. Where?

In the Colombia COVID data!

```
df_CO = pd.read_csv(url + "colombia_2020-05-28.csv")
df_CO
```

	Departamento	Edad	Sexo	Tipo	Ubicación	Estado	Fecha de inicio de síntomas
0	Bogotá D.C.	19	F	Importado	Recuperado	Leve	2020-02-27
1	Valle del Cauca	34	M	Importado	Recuperado	Leve	2020-03-04
2	Antioquia	50	F	Importado	Recuperado	Leve	2020-02-29
3	Antioquia	55	M	Relacionado	Recuperado	Leve	2020-03-06
4	Antioquia	25	M	Relacionado	Recuperado	Leve	2020-03-08
...	...	...	...	...	...	...	...
25361	Buenaventura D.E.	48	M	En estudio	Hospital	Moderado	2020-05-12
25362	Valle del Cauca	55	F	En estudio	Casa	Leve	2020-05-21
25363	Buenaventura D.E.	39	F	En estudio	Casa	Leve	2020-05-23
25364	Valle del Cauca	13	F	En estudio	Casa	Leve	2020-05-13
25365	Córdoba	0	F	En estudio	Hospital	Moderado	2020-05-11

25366 rows x 10 columns

This example will motivate our discussion of quantitative variables today!



- 1 Visualizing One Quantitative Variable
- 2 Summarizing One Quantitative Variable
- 3 Recap



1 Visualizing One Quantitative Variable

2 Summarizing One Quantitative Variable

3 Recap

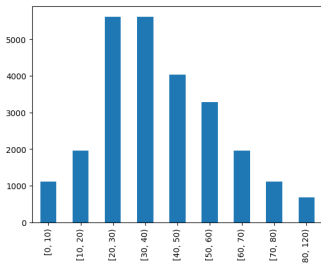


# Visualizing One Quantitative Variable

To visualize the age variable, we did the following:

quantitative variable  $\xrightarrow{\text{binning}}$  categorical variable  $\xrightarrow{\hspace{2cm}}$  make a "bar plot"

```
df_CO["age"] = pd.cut(  
    df_CO["Edad"],  
    bins=[0, 10, 20, 30, 40, 50, 60, 70, 80, 120],  
    labels=["0-9", "10-19", "20-29", "30-39", "40-49", "50-59", "60-69",  
    right=False)  
df_CO["age"].value_counts(sort=False).plot.bar()
```



This is the idea behind a visualization called the **histogram**.

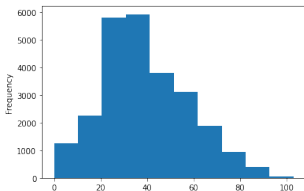


# Histograms

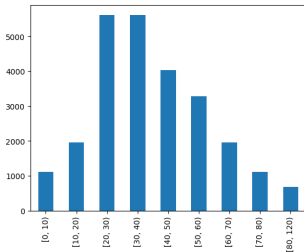
Pandas provides a built-in method for constructing histograms:

`Series.plot.hist()`.

```
df_CO["Edad"].plot.hist()
```



How does this differ from the manual histogram from earlier?



- There are no spaces between the bars.
- The  $x$ -axis is just numbers, rather than bins.

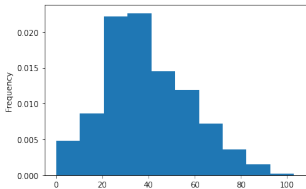


# Distributions

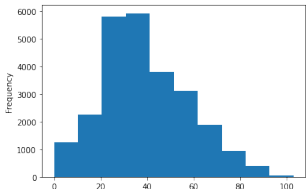
Recall the distribution of a categorical variable.

The **distribution** of a quantitative variable is similar. The counts are scaled so that the total *area* is 1.0 (or 100%).

```
df_CO["Edad"].plot.hist(density=True)
```



How does this differ from the (counts) histogram from earlier?



- Only the y-axis changes.
- The shape is the same!



1 Visualizing One Quantitative Variable

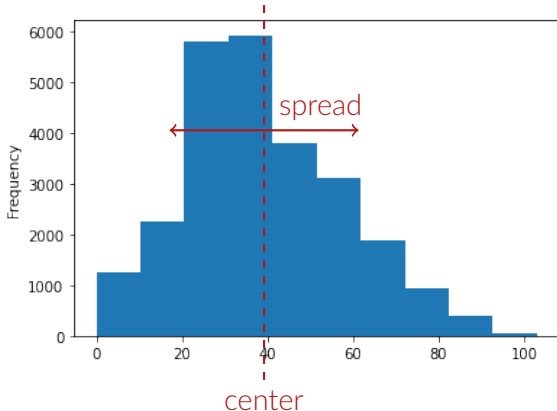
2 Summarizing One Quantitative Variable

3 Recap



# Summarizing a Quantitative Variable

If you had to summarize this data using a single number, what number would you pick?



If you had to summarize this data using two numbers, what number would you pick second?



## Summaries of Center: Mean

One summary of the center of a quantitative variable is the **mean**.

To calculate the mean of a quantitative variable  $\mathbf{x}$  with values  $x_1, x_2, x_3, \dots, x_n$ , we use the formula:

$$\bar{\mathbf{x}} = \text{mean}(\mathbf{x}) = \frac{\sum_{i=1}^n x_i}{n}$$

You can calculate it manually...

```
df_C0["Edad"].sum() / df_C0.shape[0]
```

```
39.04742568792872
```

...or using a built-in Python function.

```
df_C0["Edad"].mean()
```

```
39.04742568792872
```



# Summaries of Center: Mean

Don't be fooled by the humble mean,

$$\bar{\mathbf{x}} = \text{mean}(\mathbf{x}) = \frac{\sum_{i=1}^n x_i}{n}.$$

It is not entirely obvious that this formula should give a summary of center!

Let's investigate one reason in Colab.





## Summaries of Center: Median

Another summary of center is the **median**, which is the “middle” of the *sorted* values.

To calculate the median of a quantitative variable  $\mathbf{x}$  with values  $x_1, x_2, x_3, \dots, x_n$ , we do the following steps:

- 1 Sort the values from smallest to largest:

$$x_{(1)}, x_{(2)}, x_{(3)}, \dots, x_{(n)}.$$

Statisticians call the sorted values the **order statistics**.

- 2 The “middle” value depends on whether we have an odd or an even number of observations.
  - If  $n$  is odd, then the middle value is  $x_{(\frac{n+1}{2})}$ .
  - If  $n$  is even, then there are two middle values,  $x_{(\frac{n}{2})}$  and  $x_{(\frac{n}{2}+1)}$ . It is conventional to report the mean of the two values (but you can actually pick any value between them).



## Summaries of Center: Median

We can implement these steps in Python code manually. I asked ChatGPT, and it generated this code:



But it's easier to use the built-in Python function.

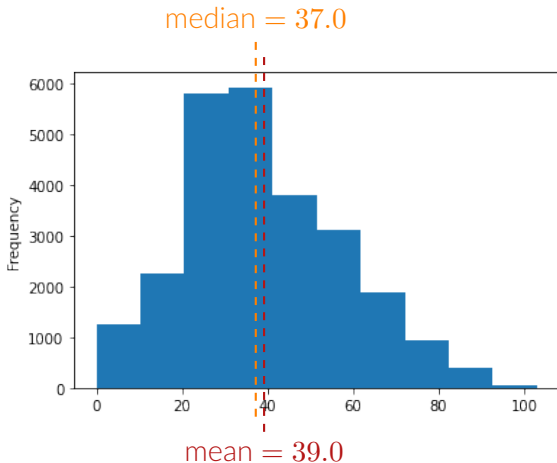
```
df_CO["Edad"].median()
```

37.0



# Summaries of Center: Mean vs. Median

We now have two summaries of center. How do they compare?



How would we summarize spread now?



## Summaries of Spread: Variance

One measure of spread is the **variance**.

The variance of a variable  $\mathbf{x}$  whose values are  $x_1, x_2, x_3, \dots, x_n$  is calculated using the formula

$$\text{var}(\mathbf{x}) = \frac{\sum_{i=1}^n (x_i - \bar{\mathbf{x}})^2}{n - 1}$$

You can implement this formula manually...

```
((df_CO["Edad"] - df_CO["Edad"].mean()) ** 2).sum() /  
(df_CO.shape[0] - 1)
```

```
348.0870469898451
```

...or using a built-in Python function.

```
df_CO["Edad"].var()
```

```
348.0870469898451
```

What are the units? *years*<sup>2</sup>



# Summaries of Spread: Standard Deviation

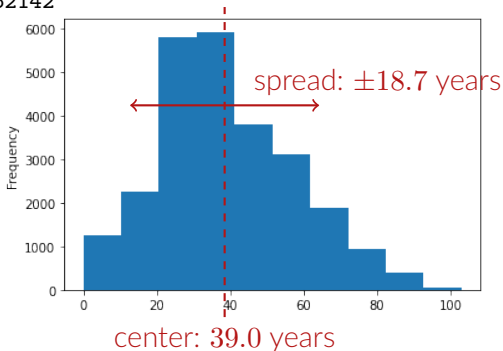
To fix the units, we take the square root to get the **standard deviation**:

$$\underset{\text{years}}{\text{sd}(\mathbf{x})} = \sqrt{\underset{\text{years}^2}{\text{var}(\mathbf{x})}}$$

We can calculate it using the built-in Pandas method `Series.std()`:

```
df_CO["Edad"].std()
```

```
18.65709106452142
```



- 1 Visualizing One Quantitative Variable
- 2 Summarizing One Quantitative Variable
- 3 Recap



# What We Learned Today

- visualizing a quantitative variable **using a histogram**
  - We've now seen several plots that can be made within Pandas: `.plot.bar()`, `.plot.hist()`, and `.plot.line()`.
- summarizing a quantitative variable
  - summarizing the center **By the mean or median**
  - summarizing the spread **By the standard deviation**
- some new Python tricks

