

Statistics

Lecture 19: | Python Implementation

Prof. Dr. Mostafa Elhosseini

<https://youtube.com/drmelhosseini>

| How to Plot Histogram?

- Determine the **frequency** and **relative frequency** for each class.
- **Mark the class boundaries on a horizontal** measurement axis.
- **Above** each **class interval**, draw a **rectangle** whose height is the corresponding **relative frequency** (or **frequency**).

| Example

- 1, 3, 27, 32, 5, 63, 26, 25, 18, 16, 4, 45, 29, 19, 22, 51, 58, 9, 42, 6

Bucket	Frequency
0 – 9	6
10 – 19	3
20 – 29	5
30 – 39	1
40 - 49	2
50 - 59	2
60 - 69	1

| Example

Bucket	Frequency
0 – 9	6
10 – 19	3
20 – 29	5
30 – 39	1
40 - 49	2
50 - 59	2
60 - 69	1

Bucket	Frequency
0 – 9 0-<10	6
10 – 19 10-<20	3
20 – 29 20-<30	5
30 – 39 30-<40	1
40 – 49 40-<50	2
50 – 59 50-<60	2
60 – 69 60-<70	1

| Dataset

- The best way to avoid mistakes in buying clothes through online shops
- You can check your predictions using this dataset
- The dataset consists of 4 columns:
 - Weight (kgs)
 - Age
 - Height (cm)
 - Size
- The output parameter is size, and input parameters are Weight, Age and Height

| Dataset

```
df = pd.read_csv("ClothesSizePrediction.csv")
```

```
df.head()
```

	weight	age	height	size
0	62	28.0	172.72	XL
1	59	36.0	167.64	L
2	61	34.0	165.10	M
3	65	27.0	175.26	L
4	62	45.0	172.72	M

| Data Visualization

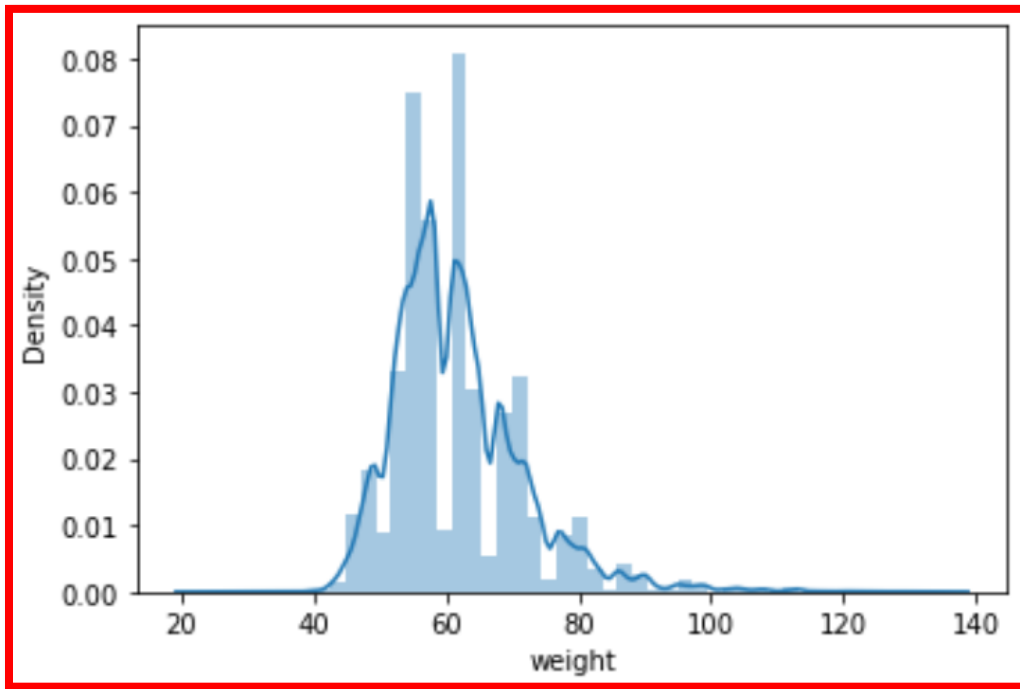
- During the **EDA** workflow, **visualization** of data is a critical step.
- Using it gives you a better understanding of your data and allows for communication of insights to both technical and non-technical stakeholders
- The **seaborn library** allows you to easily create informative and attractive plots. It builds on top of matplotlib

| Data Visualization

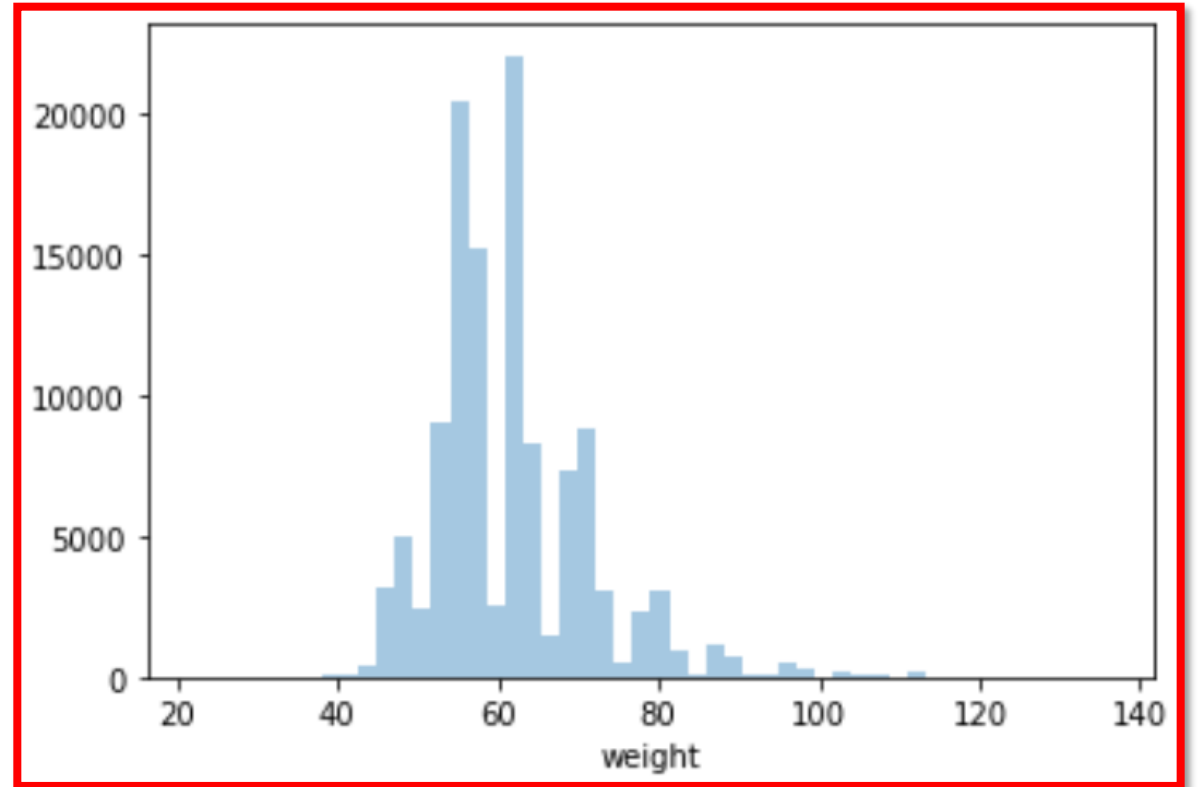
- In many machine learning algorithms, assumption is made about **how the data is distributed**, so before you apply those algorithms, it's critical to understand how your own dataset is distributed.
- It is easy to generate a **histogram** with **Seaborn's distplot** function that displays a distribution chart of a variable

| Histogram

```
sns.distplot(df["weight"])
```



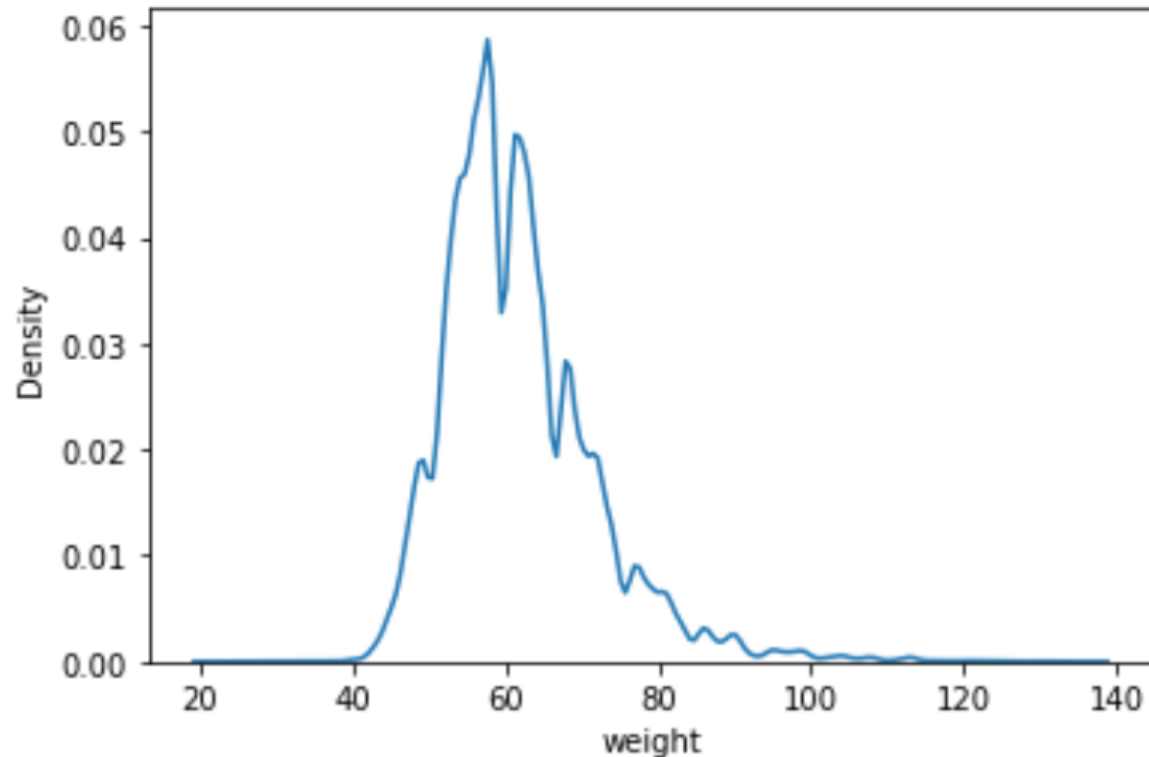
```
sns.distplot(df["weight"], kde = False)
```



- **KDE:** Kernel Density function

| Histogram

```
sns.distplot(df["weight"], hist = False )
```



- Resembles a bell curve - normal distribution
- Almost all things we measure in our lives are approximated well by the normal distribution
- Most models assume you have normally distributed data

| Histogram

```
sns.distplot(df["weight"], kde = False, bins = 15)
```

