Lecture-2
Course: Applied Data Science

Exploratory Data Analysis

By

Dr. Sibarama Panigrahi

Assistant Professor, Department of Computer Sc. & Engineering

National Institute of Technology, Rourkela, Odisha, 769008, India

Mobile No.: +91-7377302566

Email: panigrahis[at]nitrkl[dot]ac[dot]in

panigrahi[dot]sibarama[at]gmail[dot]com

Exploratory Data Analysis

- Exploratory Data Analysis (EDA) involves **summarizing** the main characteristics of the data, often *through visualization and statistical analysis*.

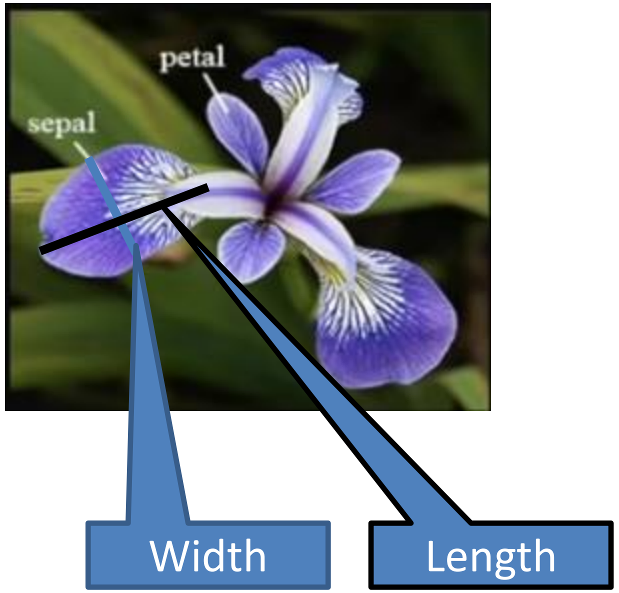
- It is a crucial step in the **data analysis process** where you *examine and understand the data* before diving into further analysis or modeling.

IRIS Flower Dataset

- Iris Dataset:

https://en.wikipedia.org/wiki/Iris_flower_data_set

- Dataset is collected in 1936 by Ronald Fisher.
- 3 flowers of Iris species.
  - 50 flowers of each class. Total 150 flowers.
- Petal (length & width) and Sepal (length & width)



IRIS Flower Dataset

- Iris Dataset:

https://en.wikipedia.org/wiki/Iris_flower_data_set

- Petal (length & width) and Sepal (length & width)
- **Objective:** ***Classify*** a *new flower as belonging to one of the 3 classes* given the 4 features.

Sepal length	Sepal width	Petal length	Petal width	Species
5.1	3.5	1.4	0.2	*I. setosa*
4.9	3.0	1.4	0.2	*I. setosa*
4.7	3.2	1.3	0.2	*I. setosa*
4.6	3.1	1.5	0.2	*I. setosa*

Instance/ Object

Each Column is a Feature/ Attribute
4 input (independent) features, 1 target (dependent) class label

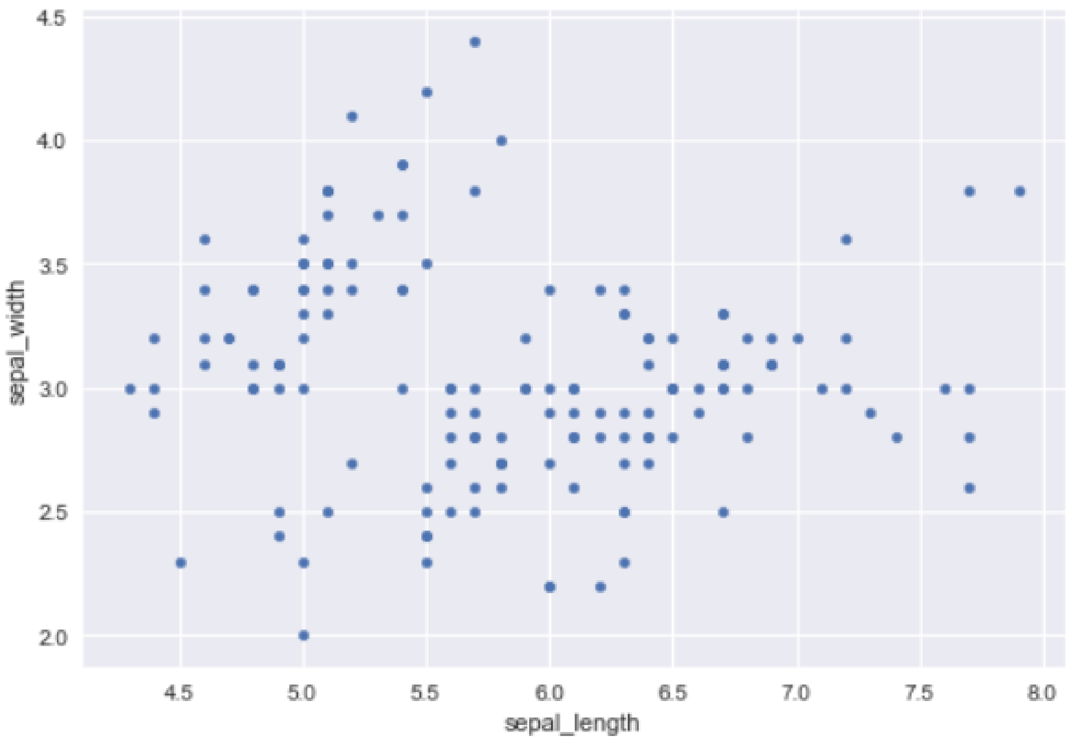

Setosa


Versicolor


Virginica

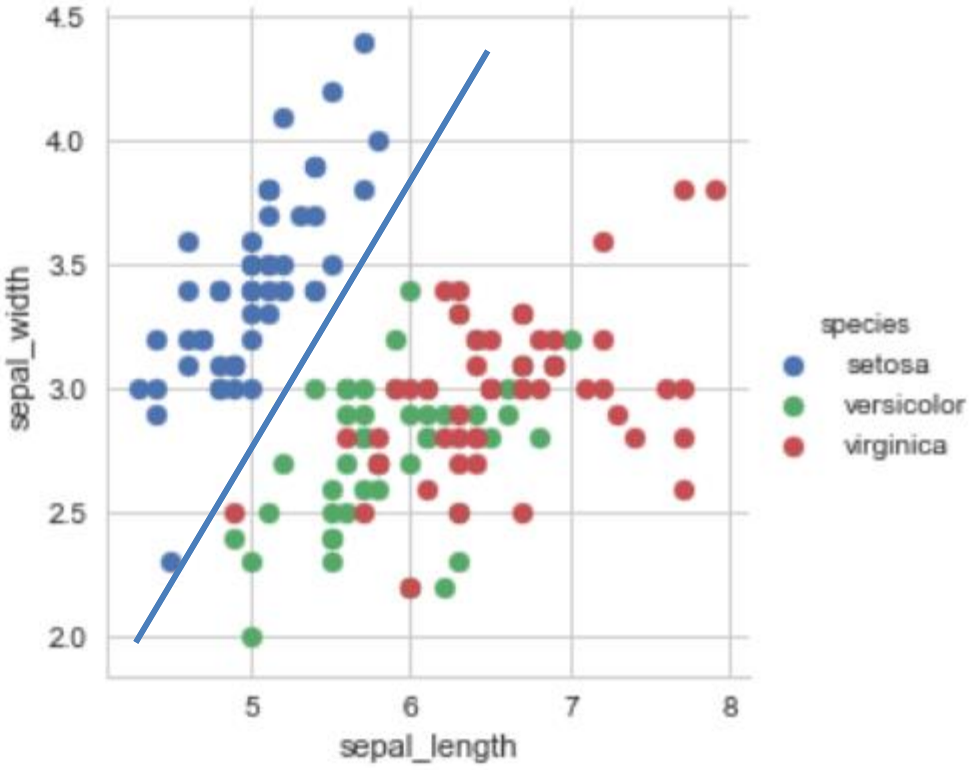
2-D Scatter Plot

- **2-D Scatter Plot:** Visualization used in exploratory data analysis to display the relationship between two numerical variables.
- **Observations:**
  - Look at the axis labels
  - Sepal length values lie in the range 3 to 8.
  - Sepal width values lie in the range 2 to 4.5.
  - No information relating to class of flower

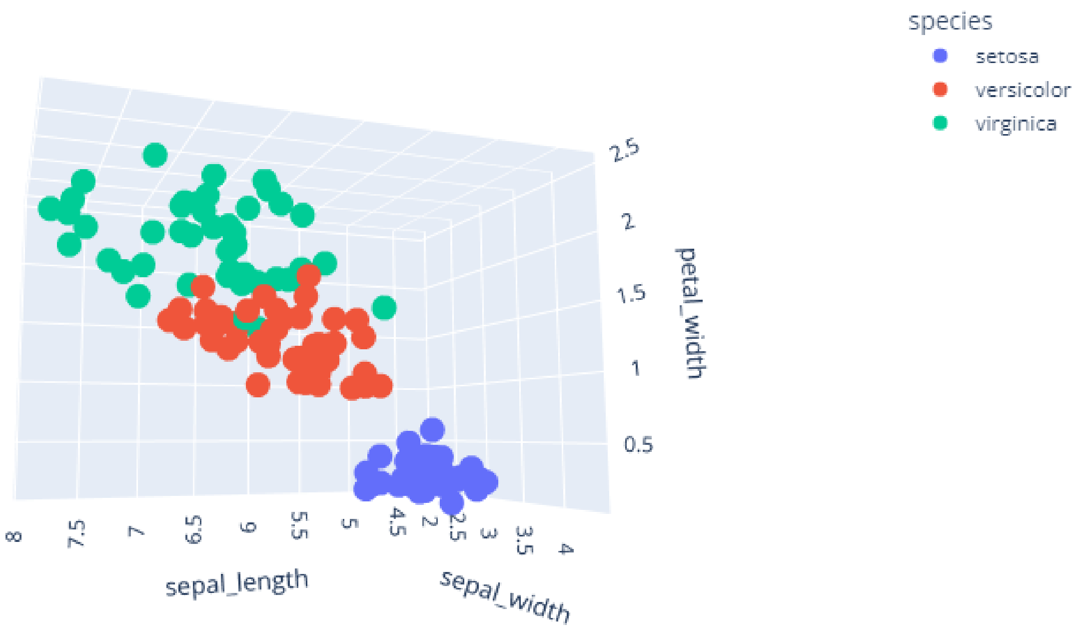


2-D Scatter Plot with Color Coding

- **Observations:**
  - Using sepal_length and sepal_width features, we can distinguish Setosa flowers from others. **[Linearly Separable]**
  - Seperating Versicolor from Viginica is much harder as they have considerable overlap.

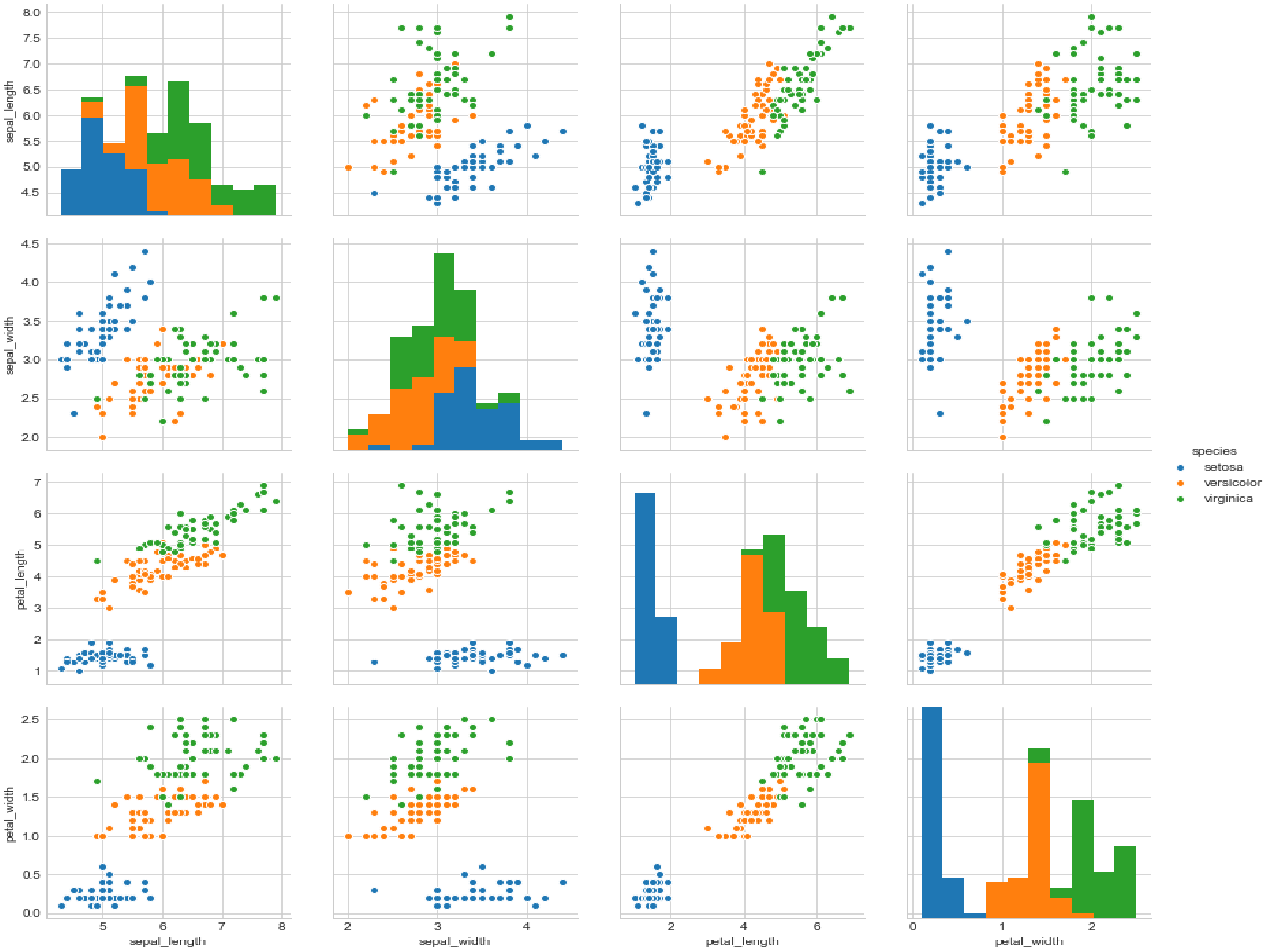


3-D Scatter Plot



- [3d scatter plots in Python (plotly.com)]
- Needs a lot to mouse interaction to interpret data.
- What about 4-D, 5-D or n-D scatter plot?

Pair Plot


sepal_length
sepal_width
petal_length
petal_width
species
setosa
versicolor
virginica

Pair Plot

- Pairwise scatter plot: Pair-Plot
- NOTE: The diagonal elements are histograms for each feature.

- **Observations:**
  - petal_length and petal_width are the most useful features to identify various flower types.
  - While Setosa can be easily identified (linearly seperable), Virginica and Versicolor have some overlap (almost linearly seperable).
  - *We can find "lines" and "if-else" conditions to build a simple model to classify the flower types.*

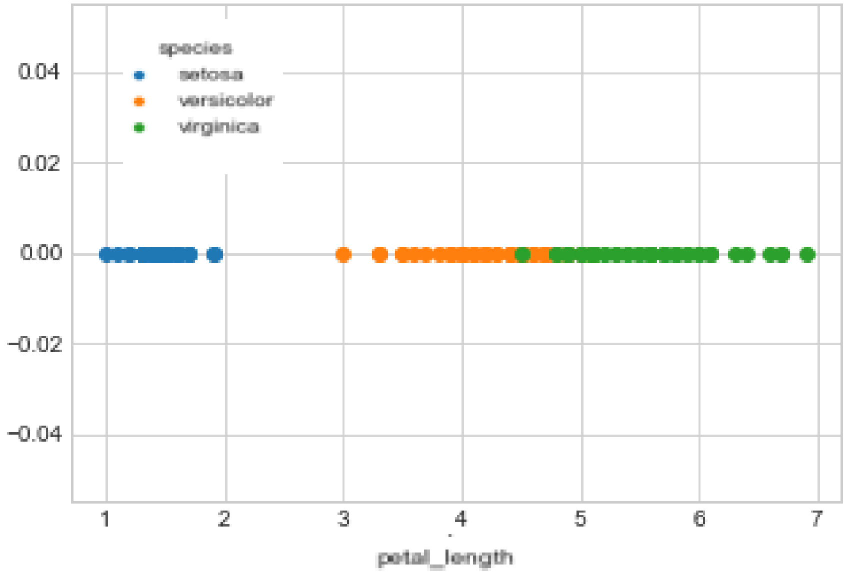
Pair Plot

- Dis-advantages:
  - Can’t be used when number of features are very high.
  - Cannot visualize higher dimensional patterns in 3-D and 4-D.
  - Only possible to view 2-D patterns.

- Will use dimensionality reduction techniques like principal component analysis (PCA), t-distributed stochastic neighborhood embedding (t-SNE) to reduce the dimensions of high dimensional features and employ pair plot to visualize.

1-D Scatter Plot

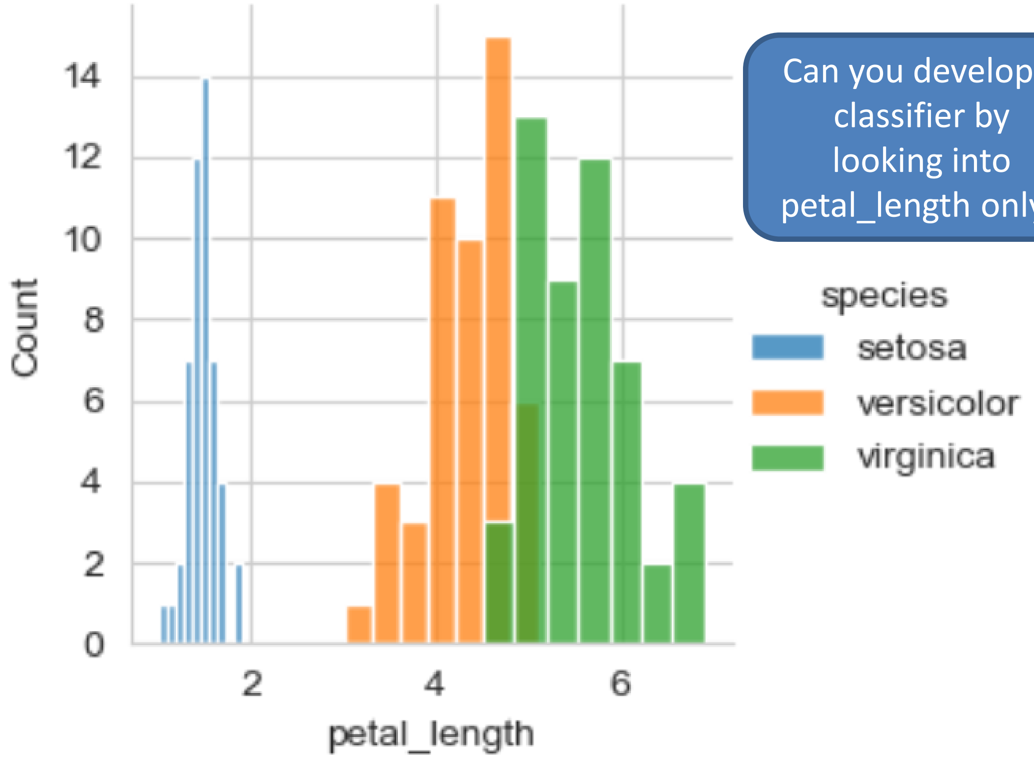
- **x-axis:** Petal length
- **y-axis:** All values zero



- **Observations:**
  - Very hard to make sense as points are overlapping a lot.
  - Are there better way to visualize 1-D scatter plot?

Histogram

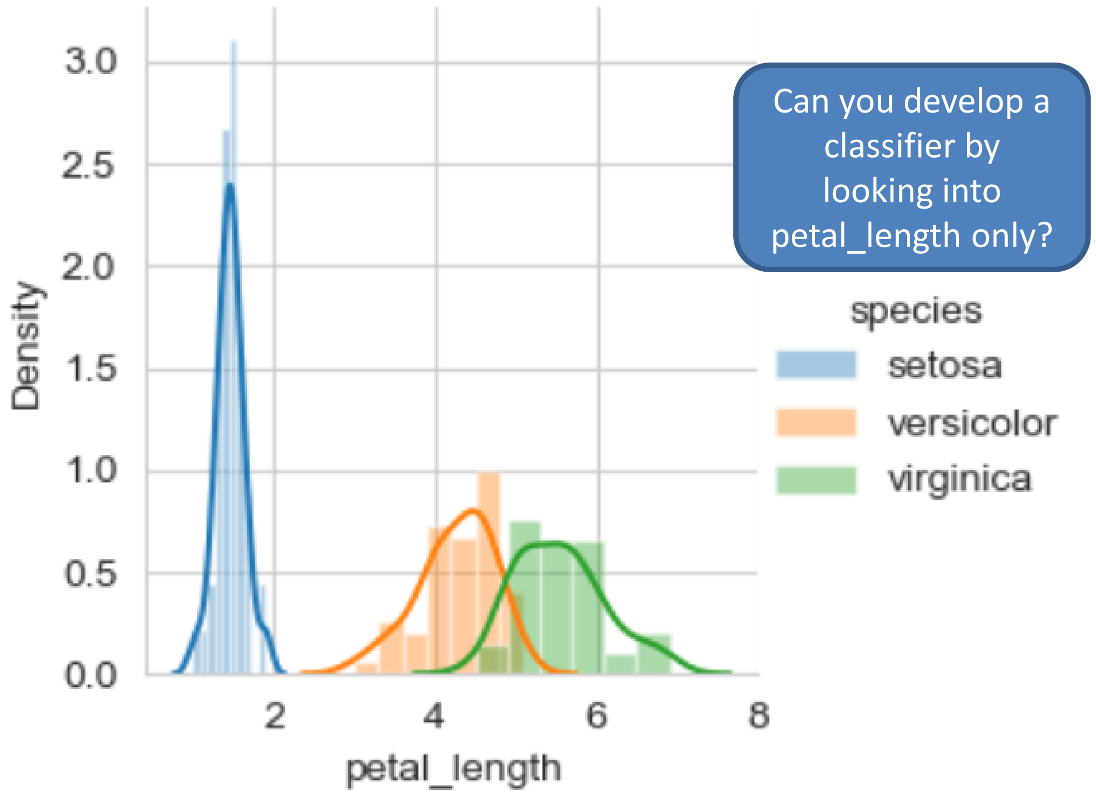
- A histogram is a graphical representation of the distribution of numerical data.
- It consists of a series of bins, where the range of values is divided into intervals, and the height of each bar represents the frequency or count of observations falling into that interval.



Can you develop a classifier by looking into petal_length only?

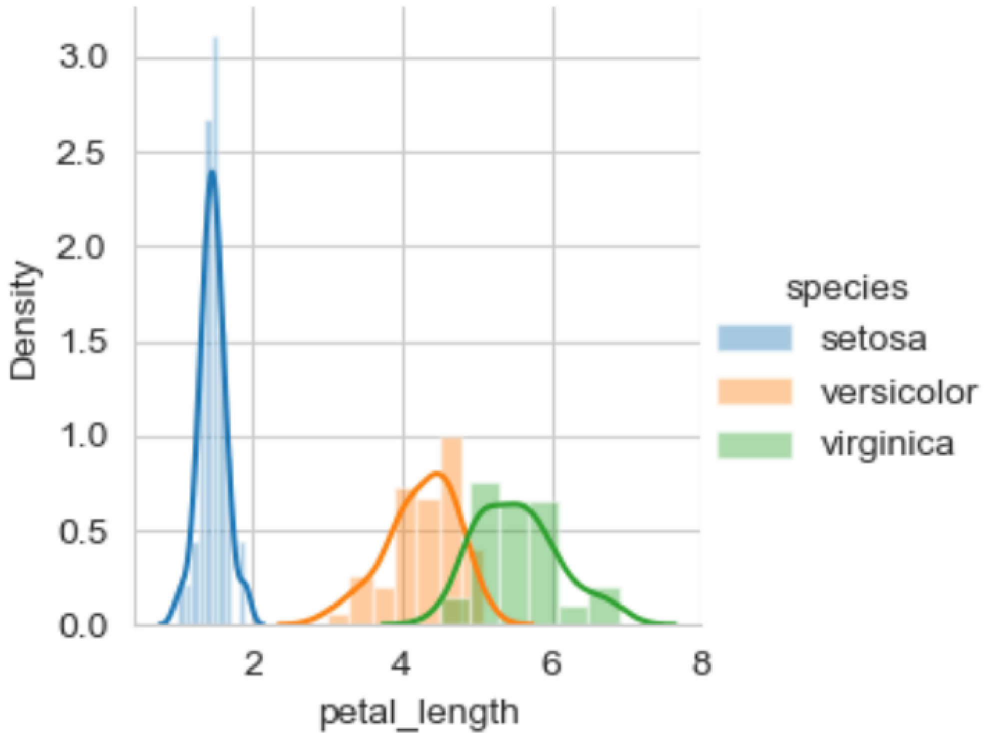
Probability Density Function(PDF)

- In terms of a histogram, the PDF represents the relative frequencies of different intervals or bins in a histogram when the data is continuous.
- Essentially, it's a smooth curve that approximates the distribution of the data.



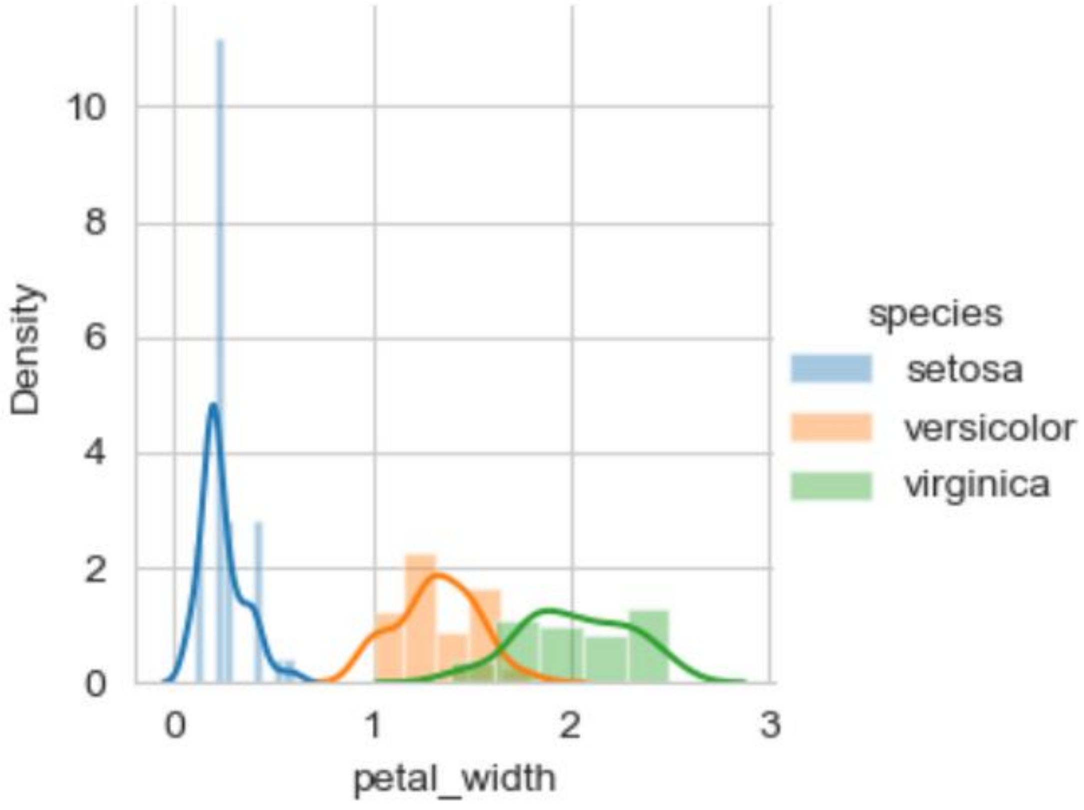
Univariate Analysis

- Analysis based on single variable is called univariate analysis.
- It can be done using Histograms and PDFs.
- **Which particular feature is more useful than other features in classifying iris dataset?**



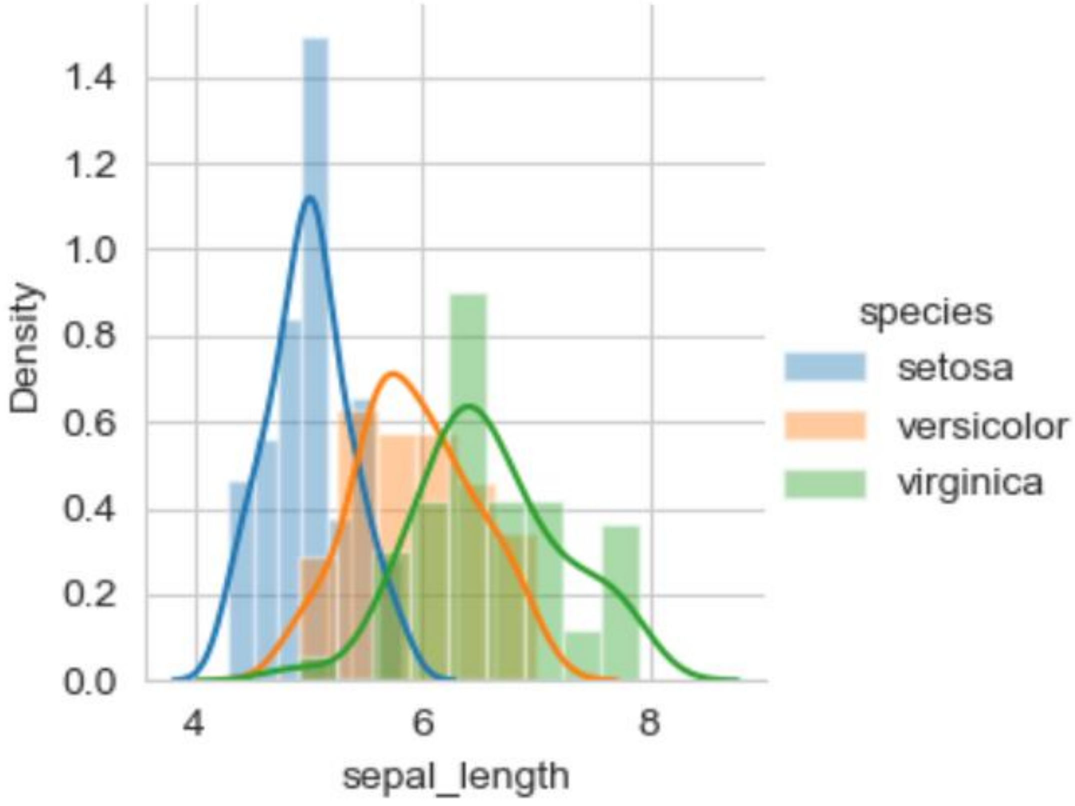
Univariate Analysis

- Analysis based on single variable is called univariate analysis.
- It can be done using Histograms and PDFs.
- **Which particular feature is more useful than other features in classifying iris dataset?**



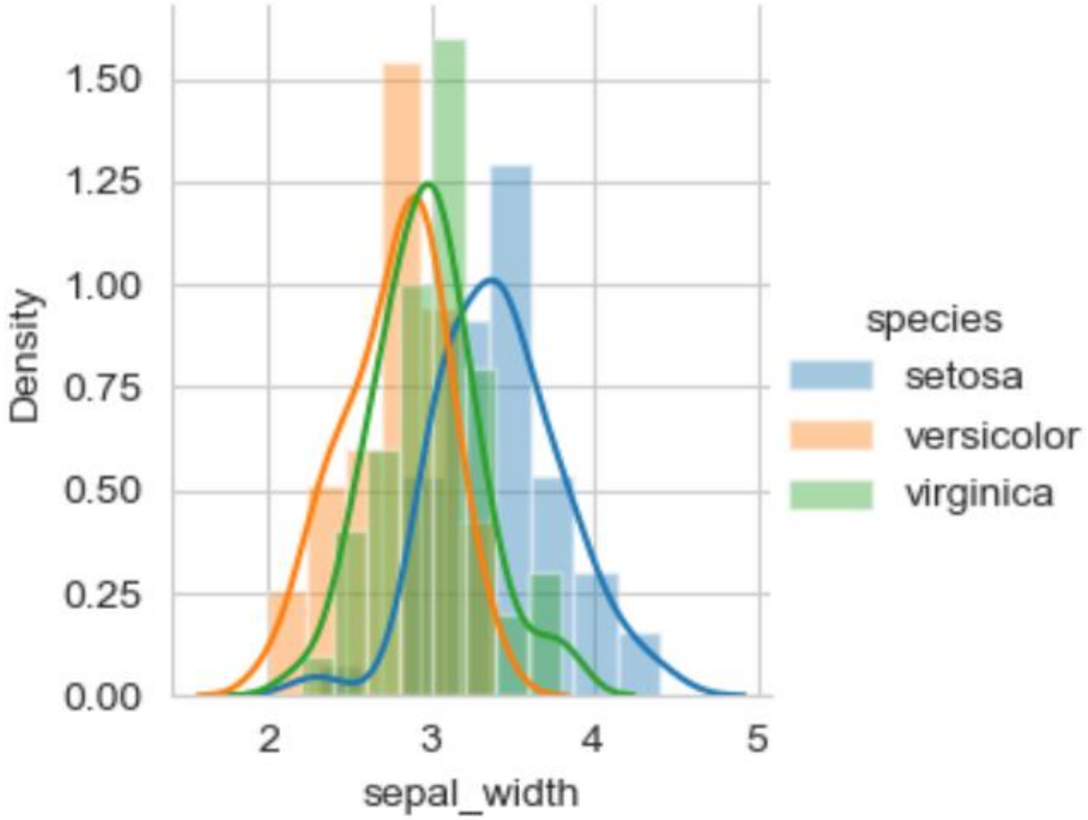
Univariate Analysis

- Analysis based on single variable is called univariate analysis.
- It can be done using Histograms and PDFs.
- **Which particular feature is more useful than other features in classifying iris dataset?**



Univariate Analysis

- Analysis based on single variable is called univariate analysis.
- It can be done using Histograms and PDFs.
- **Which particular feature is more useful than other features in classifying iris dataset?**



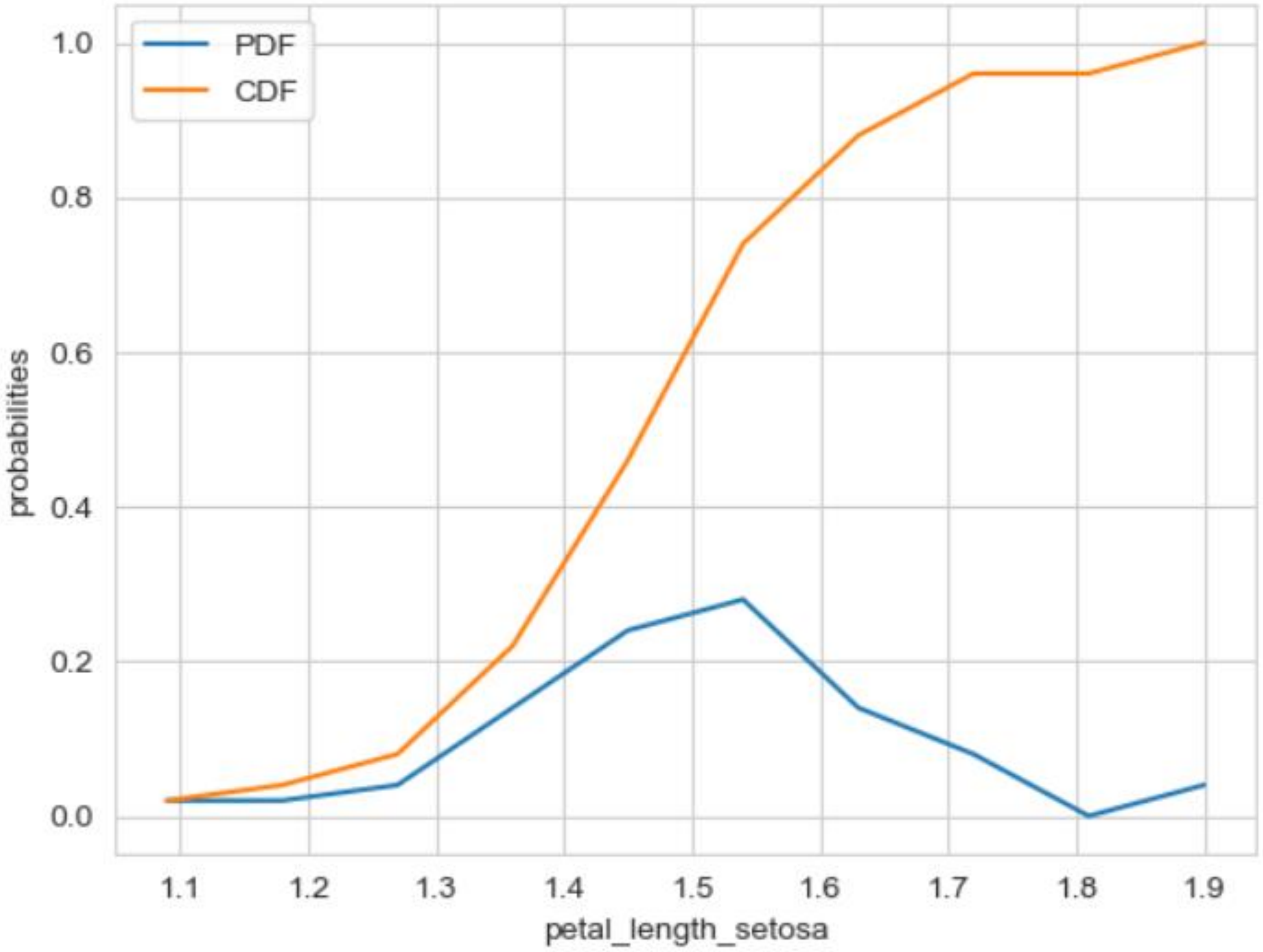
Univariate Analysis

- Analysis based on single variable is called univariate analysis.
- It can be done using Histograms and PDFs.
- **Which particular feature is more useful than other features in classifying iris dataset?**
- Petal_length > Petal_width > Sepal_length > Sepal_width
  - > denotes more appropriate

Cumulative Distribution Function

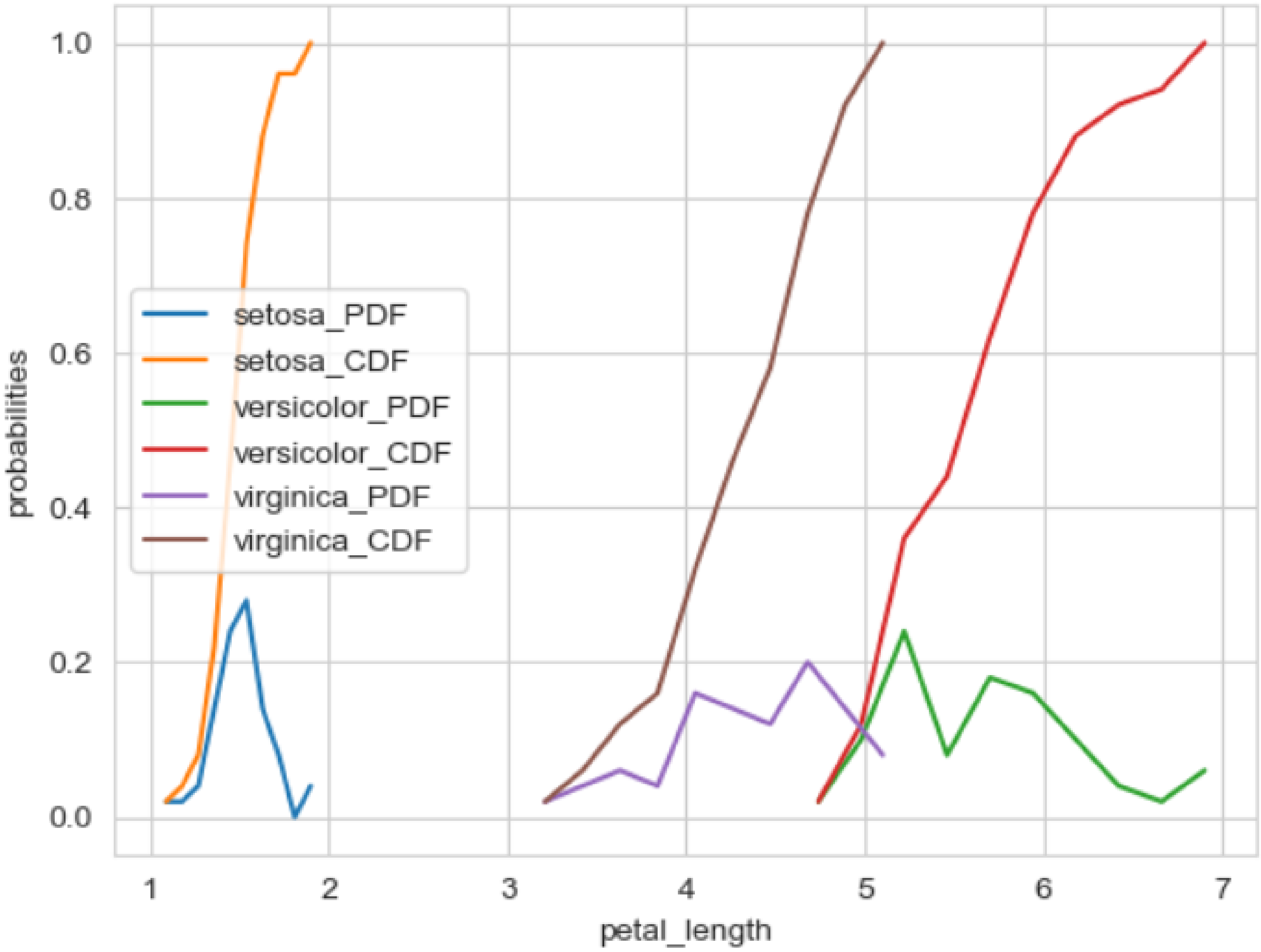
- A cumulative distribution function (CDF) is another way to describe the probability distribution of a random variable, but **it provides cumulative probabilities instead of probability densities.**
- The CDF gives the probability that a random variable takes on a value less than or equal to a given value.
- CDF at a particular point is the Area under the curve of PDF until that point.
- *Hence if you differentiate your CDF you will get your PDF. If you do integration on your PDF you will get CDF.*

Cumulative Distribution Function



Univariate Analysis

- **Observations:** Build a simple classifier based on univariate analysis and what can you say about its accuracy?



Mean, Standard Dev., Variance

- Mean

Mathematically, the mean (μ) of a dataset with n values (x_1, x_2, ..., xn) is calculated as:

$$\mu = \frac{1}{n} \sum_{i=1}^{n} x_i$$

In simple terms, to find the mean:

1. Add up all the values in the dataset.
2. Divide the sum by the total number of values.

The mean is sensitive to extreme values, also known as outliers, because it takes into account every value in the dataset. If there are outliers, they can significantly influence the value of the mean.

Mean, Standard Dev., Variance

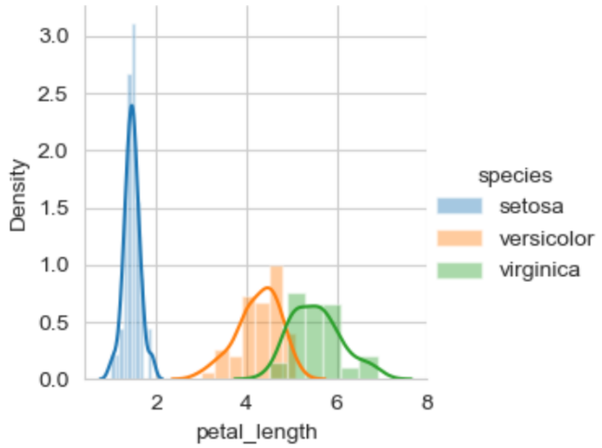
- **Standard Deviation:** The standard deviation is a measure of the dispersion or spread of a dataset around its mean.

Mathematically, the standard deviation (σ) of a dataset with n values (x_1, x_2, ..., xn) and mean μ is calculated as:

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - \mu)^2}$$

In simple terms, to find the standard deviation:

1. Calculate the difference between each value and the mean.
2. Square each of these differences.
3. Take the average of the squared differences.
4. Take the square root of the result.



Mean, Standard Dev., Variance

- **Variance:** A statistical measure that quantifies the spread or dispersion of a set of data points.

Mathematically, the variance (σ^2) of a dataset with n values (x_1, x_2, ..., xn) and mean μ is calculated as:

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2$$

Or equivalently:

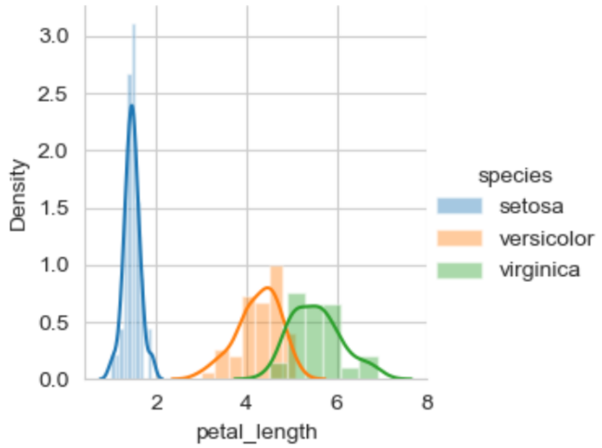
$$\sigma^2 = \frac{\sum_{i=1}^{n}(x_i-\mu)^2}{n}$$

Where:

- x_i are the individual data points,
- μ is the mean of the dataset,
- n is the number of data points.

In simple terms, to find the variance:

1. Calculate the difference between each value and the mean.
2. Square each of these differences.
3. Take the average of the squared differences.



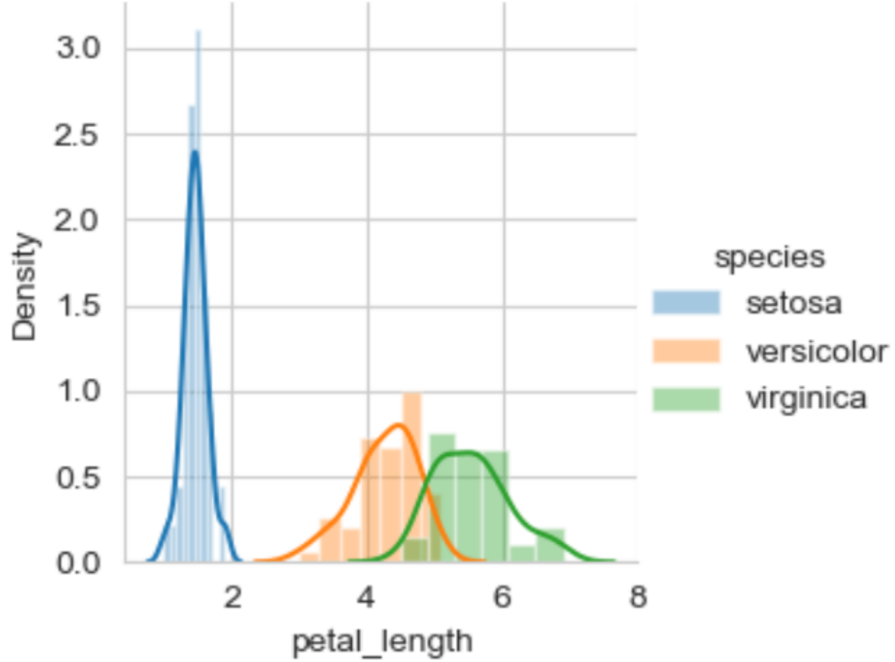
Mean, Standard Dev., Variance

```
print("Means:")
print(np.mean(iris_setosa["petal_length"]))
print(np.mean(iris_virginica["petal_length"]))
print(np.mean(iris_versicolor["petal_length"]))

print("\nStd-dev:");
print(np.std(iris_setosa["petal_length"]))
print(np.std(iris_virginica["petal_length"]))
print(np.std(iris_versicolor["petal_length"]))
```

```
Means:
1.464
5.5520000000000005
4.26

Std-dev:
0.17176728442867115
0.5463478745268441
0.4651881339845204
```



Median

1, 3, 3, **6**, 7, 8, 9

Median = $\underline{\underline{6}}$

ount=7 mean=5.14

1, 2, 3, **4**, **5**, 6, 8, 9

Median = (4 + 5) ÷ 2

= $\underline{\underline{4.5}}$

Mean is more sensitive to outliers.

Median is less sensitive to outliers.

Percentile

- $percentile(x) = \frac{Numbe\ of\ values\ below\ x}{Total\ number\ of\ values}$

- 50^{th} percentile is the median.

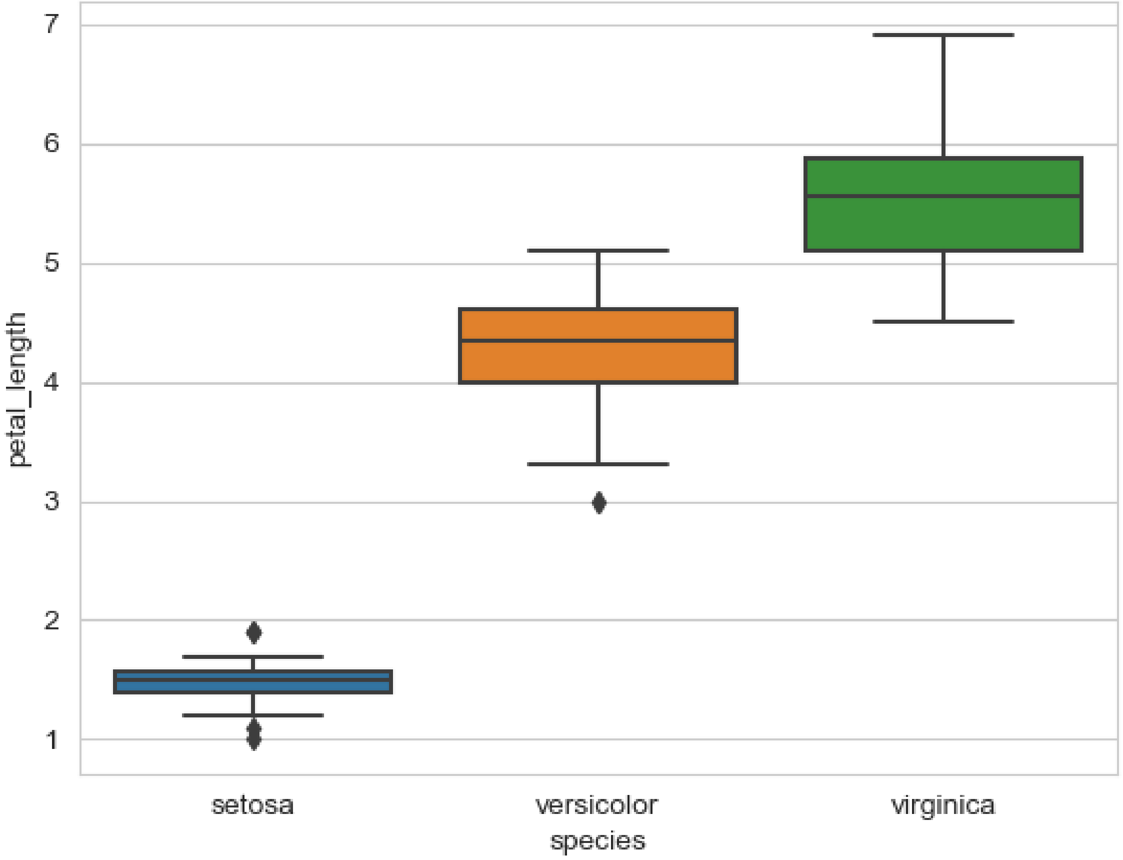
- 25^{th} , 50^{th}, 75^{th} and 100^{th} percentiles are called Quntiles.

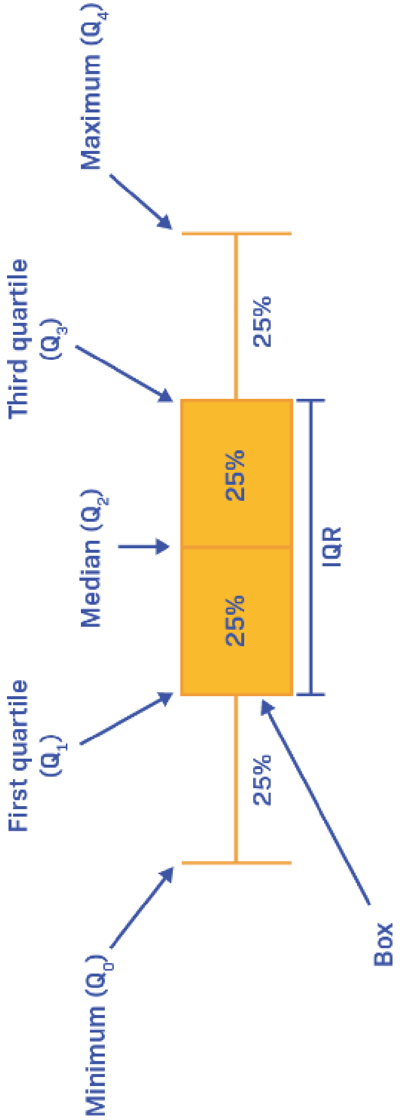
- 1^{st} Quantile: 25^{th} Percentile
- 2^{nd} Quantile: 50^{th} Percentile
- 3^{rd} Quantile: 75^{th} Percentile

- Interquartile Range (IQR)= 75^{th} Percentile – 25^{th} Percentile

Box Plot

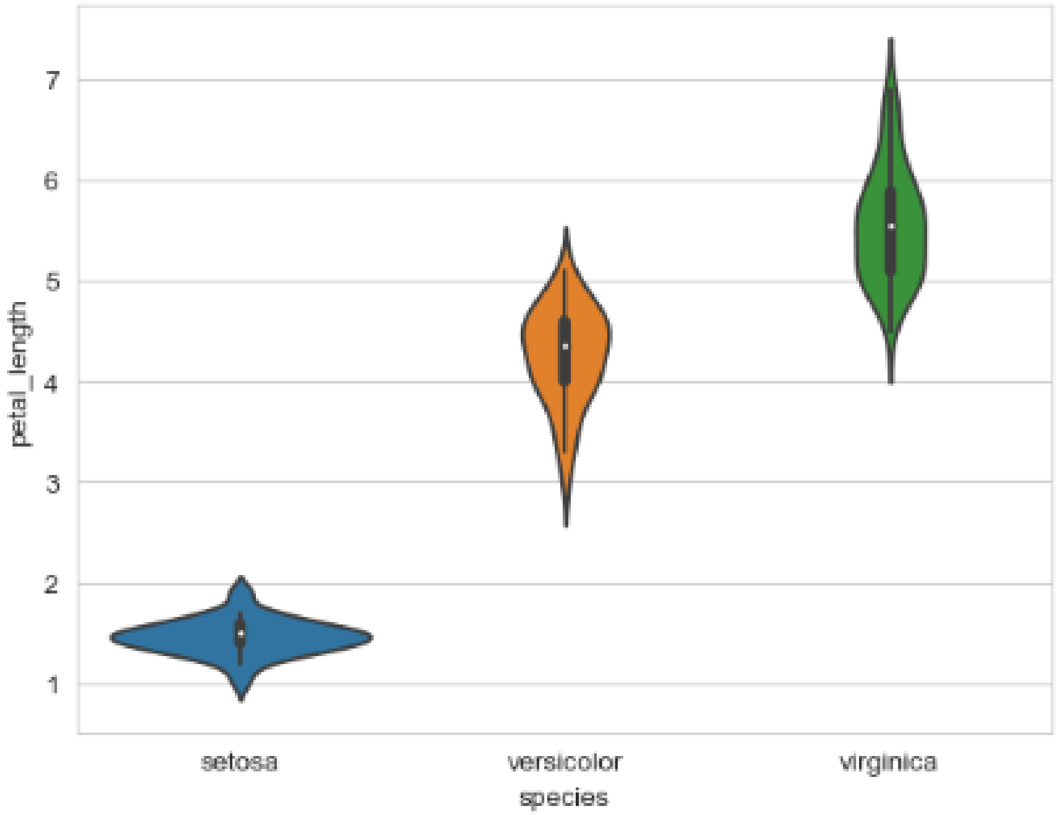
- Box plot with whiskers: another method of visualizing the 1-D scatter plot more intuitively.

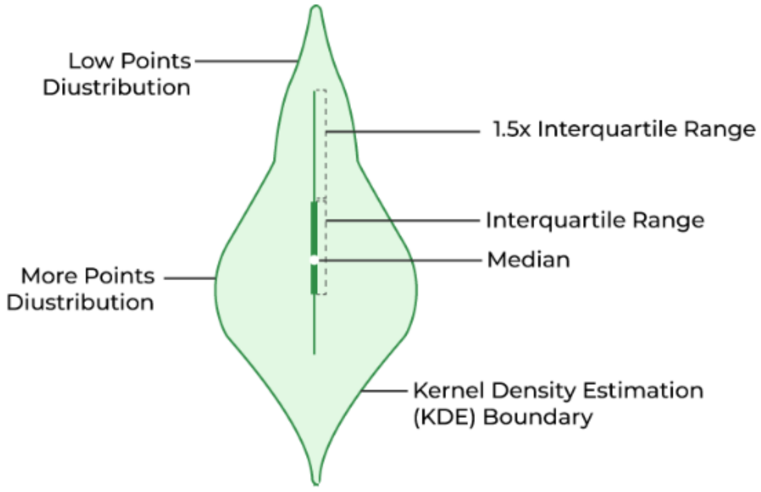




Violin Plots

- A violin plot combines the benefits of Box Plot and PDF and simplifies them.
- Denser regions of the data are fatter, and sparser ones thinner in a violin plot.





Thank You

For Your Valuable Time.