"Python for Data Science"
Week 1

Learning Objectives of the Session

- Understand the big picture of Data Science & Machine Learning
- Introduction to Python
- Basic Operations in Python using a Case Study

A few applications

Case 1:
Can you predict which client will default the loan payment based on the client's spending?

- Why does the bank want to know who will default?

- What type of information I would need about the client to know the risk?

Case 2:
Can you predict when an employee will resign from his/her organization?

- Why is this important for a company?

- What type of information do we need to make an informed decision ?

- If my company is a 40-50 years old company, should one use all the available data to proceed with this analysis?

Topics covered so far

1. Python

2. Libraries Used

3. NumPy

4. Pandas

5. Visualization: Matplotlib, Seaborn, etc.

6. Case Study

1. What are the different libraries for data manipulation in Python?

2. What are the key operations that can be performed using NumPy & Pandas?

Python in Data Science

- One of the fastest growing programming languages

- Great functionality to deal with mathematics, statistics and data science applications

- Easy to use, easy to debug language that also caters to people with non-programming backgrounds

- Python libraries can be used as tools to assist you in working with data

- Quantitative analysis: Describes and summarizes data numerically

- Visual analysis: Illustrates data with charts, plots, graphs etc.

Key Libraries for Data Manipulation - NumPy & Pandas

Numpy

- **Num**erical **Py**thon

- Fundamental package for scientific computing

- A powerful N-dimensional array object - ndarray

- Useful in linear algebra, vector calculus, and random number capabilities, etc.

- If you use the Anaconda distribution, you will automatically be able to use the common libraries, NumPy being one of them.
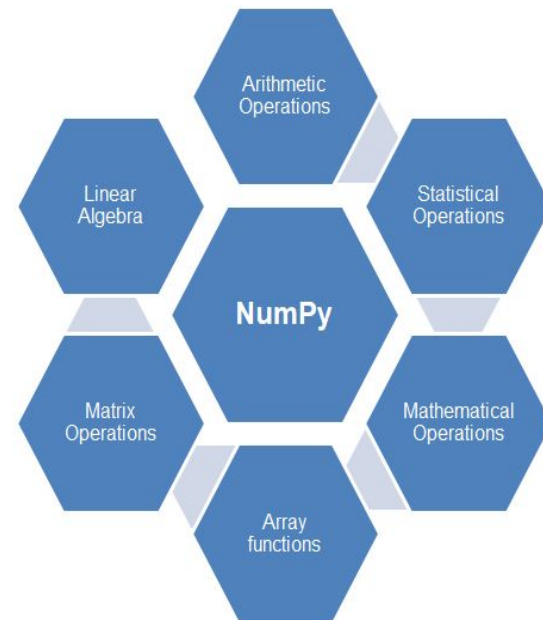
Key Libraries for Data Manipulation - NumPy & Pandas

Pandas

- Extremely useful for data manipulation and exploratory analysis

- Built on top of NumPy

- Offers two major data structures - *Series* & *DataFrame*

- A *DataFrame* is made up of several *Series* - Each column of a *DataFrame* is a *Series*.

- In a DataFrame, each column can have its own data type unlike NumPy array which creates all entries with the same data type.

NumPy Key Operations

NumPy provides many useful operations for data manipulation. Some of the most commonly used operations and functions of NumPy are:

Operation	Numpy Function
Declare a NumPy array or convert a list into a NumPy array	array()
Reshape an n-dimensional array without changing the data inside the array	reshape()
Concatenate two or more arrays along a specified axis	concatenate()
Create evenly spaced elements in an interval, particularly useful while working with loops	arange(), linspace()
Working with matrices and perform different operations on them	dot(), transpose(), eye()

Pandas Key Operations

Pandas is one of the most famous data manipulation tool which is built on top of NumPy. Some of the commonly used operations and functions of Pandas are:

Operation	Pandas Function
Load or import the data from different sources/formats	read_csv(), read_excel(), read_html(), read_json()
Information about the data - dimension, column dtypes, non-null values and memory usage	info()
View of basic statistical details of numeric data - quartiles, min, max, mean, std	describe()
Used to detect missing values of an array-like object	isnull()

Gauge Your Understanding

1. What do you understand by data visualization and why is it important?

2. What are the commonly used libraries to use for data visualization in Python?

3. How do you choose the right visualization for your analysis?

Introduction to Visualization

What is Data Visualization?

- Visual representation of data

- Helps to observe & communicate patterns and trends with naked eye

Why Data Visualization is important?

- Data visualization helps to communicate information in a manner that is universal, fast, and effective.

- Communicating insights to non-technical decision makers is one of the most critical phases in a data science project

Common Libraries for Visualization

Matplotlib

- Matplotlib is one of the most popular libraries for data visualizations.
- It provides high-quality graphics and a variety of plots such as histograms, bar charts, pie charts, etc.
- Some important functions - plot(), hist(), bar(), pie(), scatter(), text(), legend(), etc.

Seaborn

- Seaborn is complementary to Matplotlib and it specifically targets statistical data visualizations.
- A saying around matplotlib and seaborn is, "matplotlib tries to make easy things easy and hard things possible, seaborn tries to make a well-defined set of hard things easy too."
- Some important functions - displot(), boxplot(), stripplot(), pairplot()

Which visualization to use (1/2)

There are numerous types of plots available in Matplotlib and Seaborn, each has its own usage with certain specific data. Choosing right visualization for right purpose is very important.

Type	X Variable	Y Variable	Purpose of analysis	Type of chart	Example
Univariate	Continuous	-	How the values of the X variable are distributed?	Histogram, Distribution plot	Distribution of cholesterol ranges Distribution of horsepower of cars
Univariate	Categorical	-	What is the count of observations in each category of X variable?	Count Plot	What is the count of employees for each type of degree in an organization?
Bivariate	Continuous	Continuous	How Y is correlated with X?	Scatter plot	How tip varies with the total bill?
Bivariate	Time Related (months, hours, etc.)	Continuous	How Y changes over time?	Line Plot	How sales varies on different days?
Bivariate	Continuous	Categorical	How range of X varies for various category levels?	Box plot, Swarm Plot	How tip varies at lunch and dinner? How tips varies with day of the week?
Bivariate	Categorical	Categorical	What is the number or % of records of X which falls under each category of Y?	Stacked Bar plot	What is the percentage of smokers and non-smokers across fitness levels.?

Note: Univariate plots can also be used to visualize relationships among two or more variables by using arguments like 'hue' in the plot.

Which visualization to use (2/2)

Multivariate analysis is used to study the interaction between more than two variables. Exploring more combination of variables helps to extract deeper insights which could not observed with univariate or bivariate analysis. Examples: Correlation, Regression analysis, etc.

Type	Variables	Purpose of analysis	Type of chart	Example
Multivariate	Continuous (more than two)	How to visualize relationship across multiple combination of variables?	Pair Plot	Relation between three variables - horsepower, weight, and acceleration
Multivariate	Continuous (more than two)	How to visualize the spread of values in the data with color-encoding?	Heatmap	Correlation matrix for three variables horsepower, weight, and acceleration

Note: Pair plot and heatmap can also be used with only two variables but are generally preferred and more useful for visualizing more than two variables.

Gauge Your Understanding

1. What do you mean by Exploratory Data Analysis and why do you need it?

2. What do you mean by Data Preprocessing and why do you need it?

3. What are the steps involved in doing Exploratory Data Analysis?

4. What are two important steps involved in data preprocessing?

Exploratory Data Analysis (EDA)

| What is EDA?	• Combination of visualization techniques and statistical methods • Exploring and summarizing key information within the data

Why of EDA?	• First step in any analysis • Gain good insights into the data • Uncover underlying structure of the data • Detect and figure out the best strategy to handle *unclean data* (missing values, outliers etc.) • Identify initial set of observations and insights

What and Why of Data Preprocessing?

Data preprocessing refers to the process of preparing the raw data into a structured format before building a machine learning model.

- Raw data is often incomplete, inconsistent and has many other fallacies
- This makes it inapt for any statistical analysis
  - It might lead to wrong insights
  - Decisions taken from this data can be counter productive for the organization
- So you can't directly go from data to insights

In real world, the journey looks more like this...

Raw Data → Structure Data → Exploratory Data Analysis ⇄ Data Preprocessing → Insights, Visualizations
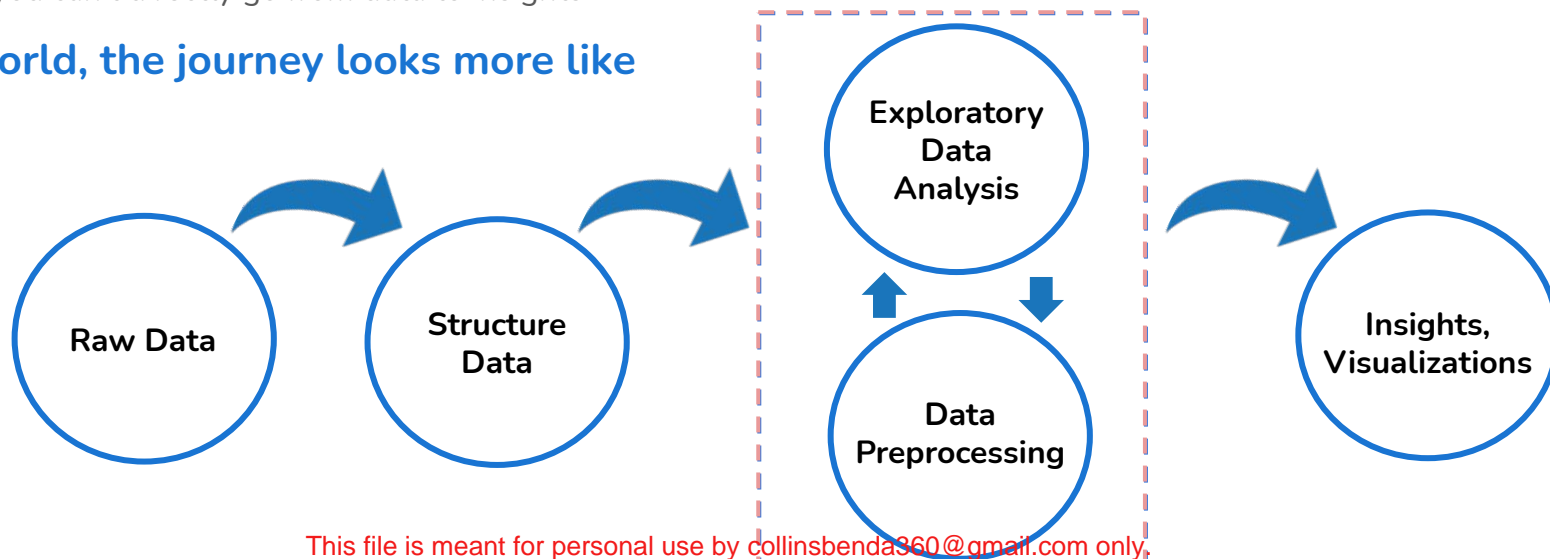
Steps of EDA

Overview of Data	Gain basic understanding of the data - shape, data types, etc.
Summary Statistics	Check descriptive statistics about the data - mean, std, median, etc.
Univariate Analysis	Check distribution of variables in the data, missing values, outliers
Bivariate Analysis	Find the patterns or relationships between different variables
Multivariate Analysis	Explore more combination of variables to unearth deeper insights
Key fixes and summarize	Identify and do the key fixes in the data. Finally, summarize the key findings from EDA

What are missing values?

- Missing values occur when no data value is stored for the variable in an observation

- Missing values can have a significant effect on the inferences that are drawn from the data

What are different types of missing values?

- Values which are not actually missing and represents some information about the data

    ○ Example - Number of hours an employee works everyday. Missing values can mean that employee was absent or on leave that particular day.

- Values which are actually missing and provides no information about the data

    ○ Example - A weighing scale that is running out of batteries. Some data will simply be missing randomly.

How to deal with missing values?

- If values are not actually missing then we can replace the missing values with the value they actually represent in the data.

  - In the example above, we can replace all the missing working hours of an employee with 0

- If values are actually missing, then we must explore the importance and extent of missing values in the data

  - If the variable has large percentage of missing values (say more than 70%) and is not significant for our analysis, then we can drop that variable

  - If the percentage of missing values is small or the variable is significant for our analysis, then we can replace the missing values in that variable using:

    - Mean or median of that variable if the variable is continuous

    - Mode of that variable if the variable is categorical

    - Sometimes we use functions like min, max, etc. to replace the missing values depending on the dataset

Outliers

What are outliers?

Outliers are the observations which are very different from the other observations. In order to analyze the data, sometimes we have to work with outliers.

How to detect outliers?

1. **Box plot:** We can visualize the outliers by using box plot.
2. **Scatter plot:** We can check outliers using scatter plot by identifying the data point the lies far from other observations.
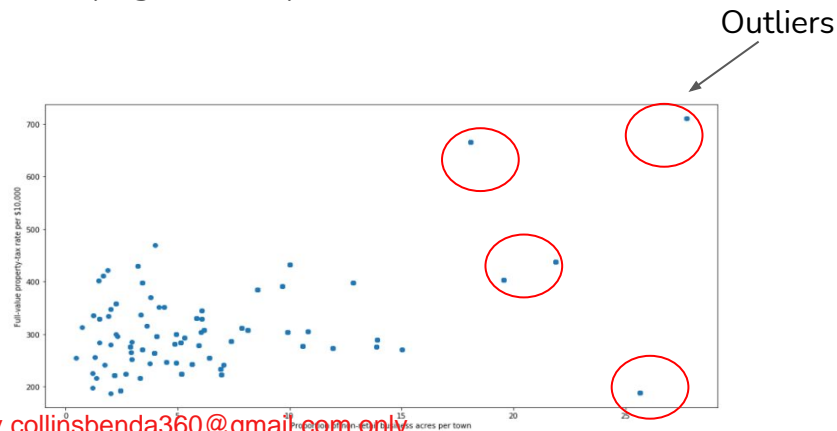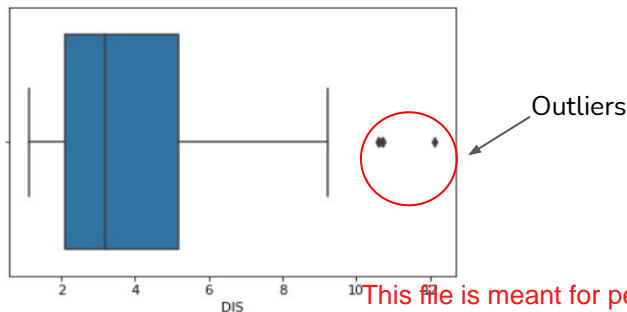
Outliers

Outliers

How to deal with outliers?

Handling outliers is subjective to the business problem we are trying to solve but some general practices are as follows:

- We should analyze outliers before treating them.

- If an outlier represents the general trend then there is no need to treat it
  - Example - Income is generally a skewed variable but all extreme points might not be outliers

- If we decide to treat the outlier after analyzing it, then:
  - we can drop them but we would lose information in other columns of the data
  - we can cap outliers at certain values, say 5th percentile or 95th percentile
  - We can set a threshold using IQR and remove the outliers greater than that threshold value

Case Study

Appendix

Scatter Plot

- A **scatter plot** uses dots to represent values for two different numeric variables.
- The position of each dot on the horizontal and vertical axis indicates values for an individual data point.
- Scatter plots are used to observe relationships between continuous variables.
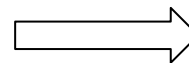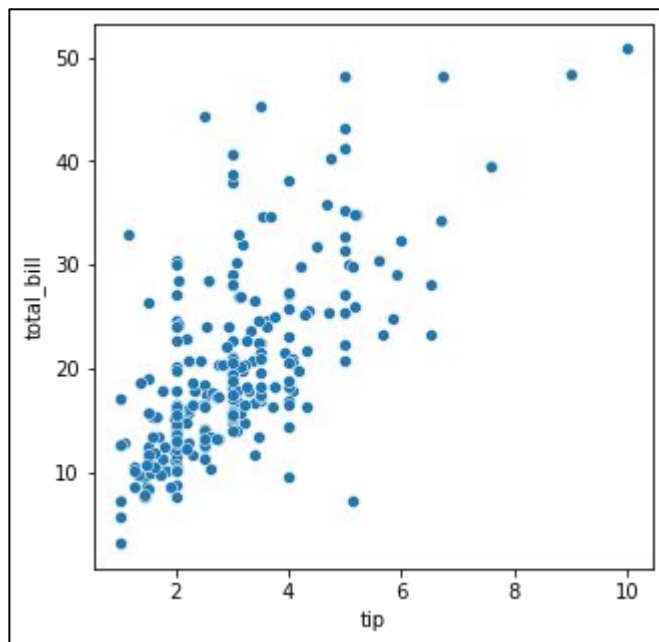
- This plot shows the relationship between the tip and the total bill.
- We can say if the total bill is large, the tip can also be large
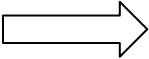
Bar Plot

- A bar chart is a chart that presents categorical data with rectangular bars with heights or lengths proportional to the values that they represent.
- The bars can be plotted vertically or horizontally.

Birthday of Students by Month
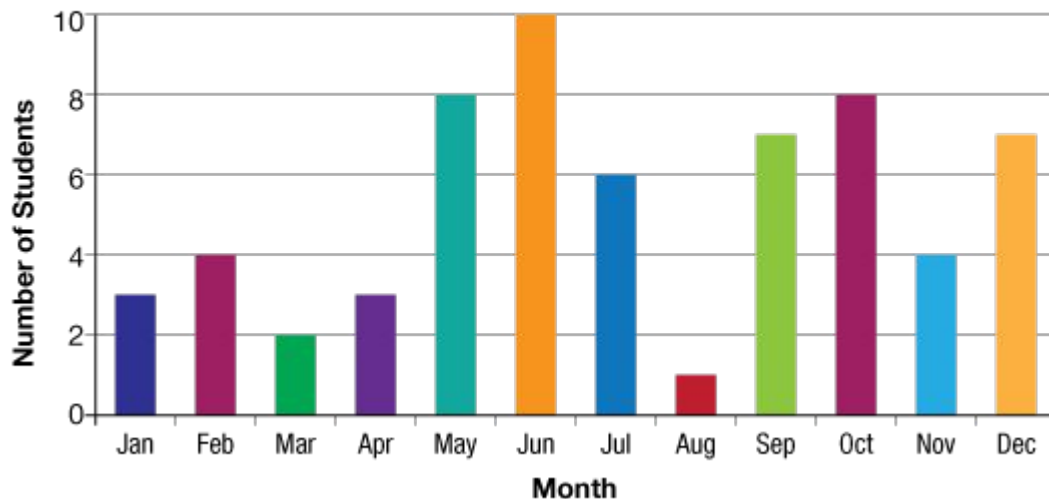
- Most of the students celebrated their birthday in June.
- In August, very less students celebrated their birthdays.

Back

Stacked Bar plot

Stacked Bar plots are used to show how a larger category is divided into smaller categories and what relationship each category of one variable has with each category of another variable.
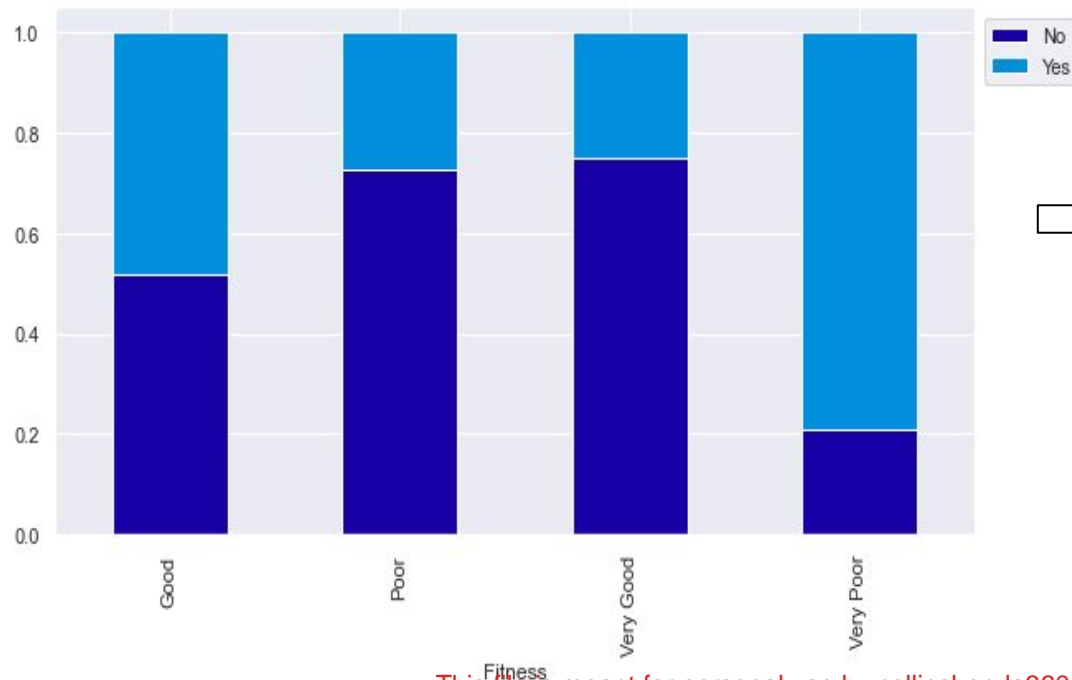
- This plot shows the percentage of smoker and non-smoker for different fitness levels
- The percentage of smokers is very high for people with very poor fitness.

Back

Line Plot

A **line graph** is a graphical display of information that changes continuously over time.
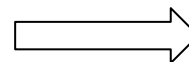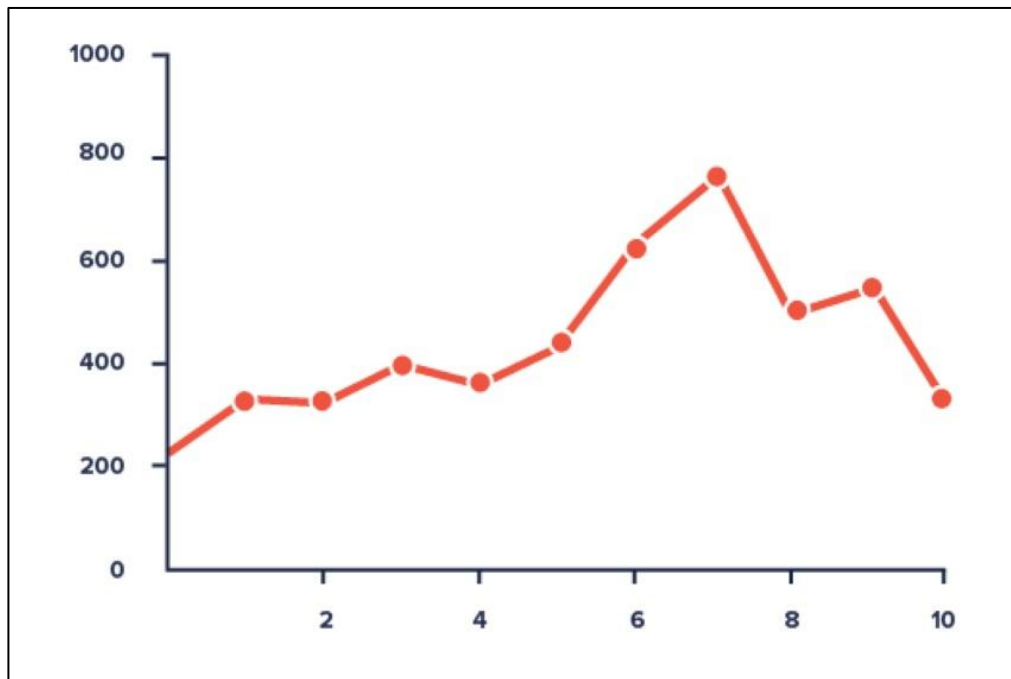
- This plot shows the relationship between the sales and the number of days
- We can say that sales has been the highest on day 7

Back

Histogram and skewness in data

- A **histogram** is a graphical display of data using bars of different heights.
- In a histogram, each bar groups numbers into ranges

Skewness refers to distortion or asymmetry in a symmetrical bell curve in a set of data

- If the curve is shifted to the left, it is called left skewed. (leftmost curve in the below fig.)
- If the curve is shifted to the right, it is called right skewed. (rightmost curve in the below fig.)
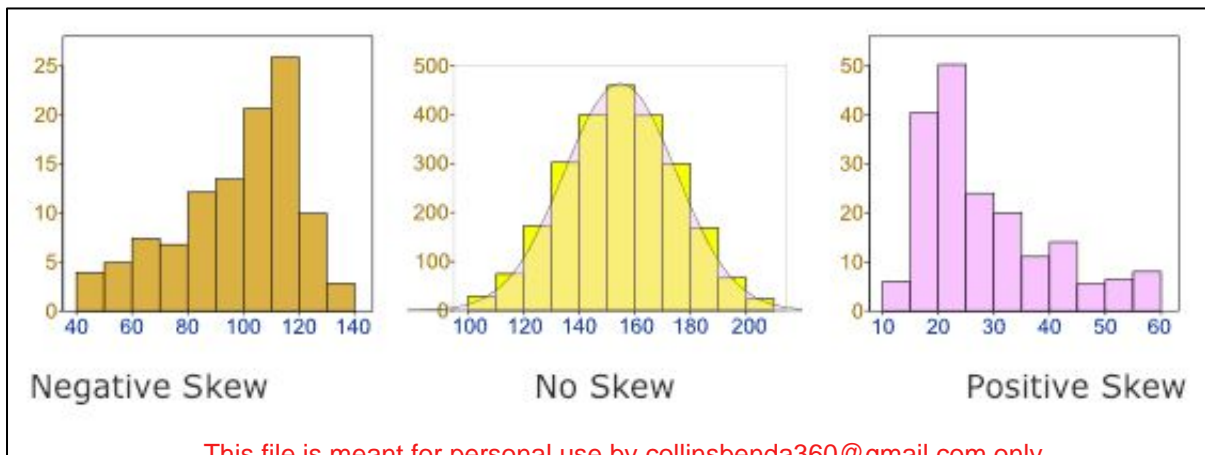
Back

Count Plot

Count plot shows the count of observations in each category of a categorical variable using bars. A count plot can be thought of as a histogram across a categorical, instead of continuous, variable.
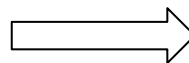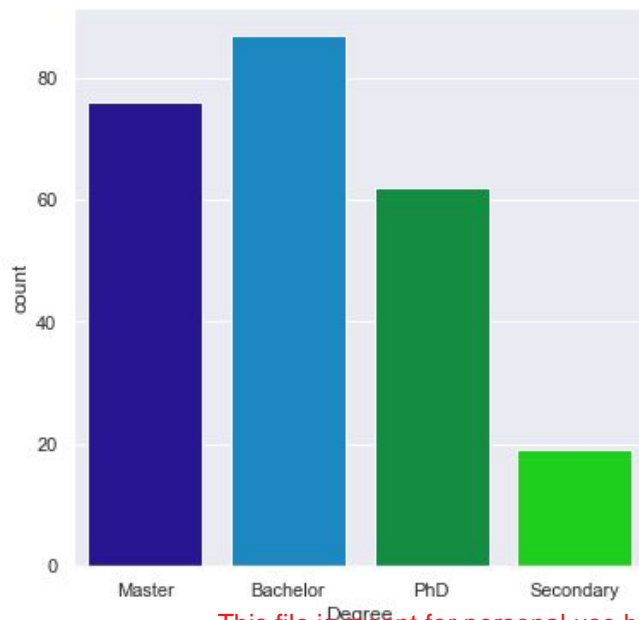
- This plot shows the count of employees for each type of degree in an organization
- We can see that majority of the employees have bachelor degree followed by master.

Back

A box plot is a type of chart often used in exploratory data analysis to visualize the distribution of numerical data and get an idea about the skewness and outliers in the data by displaying the items included in the five point summary. The five point summary includes:

- The minimum
- Q1 (the first quartile, or the 25% mark)
- The median (the second quartile, or the 50% marks)
- Q3 (the third quartile, or the 75% mark)
- The maximum
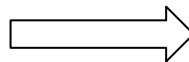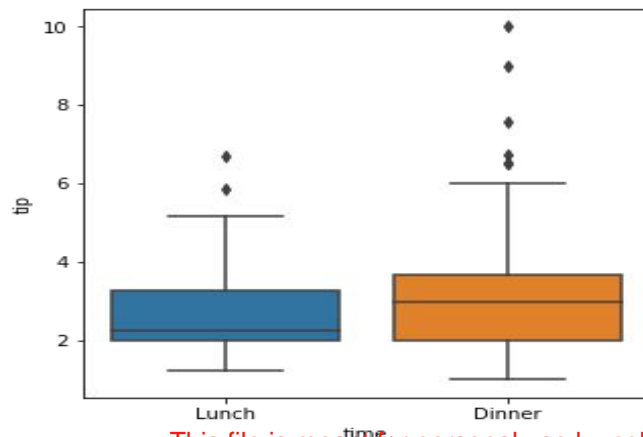
- This plot shows how tip varies at lunch and dinner times in a restaurant.
- We can see that median value of tip is larger at the time of dinner.

Back

Swarm Plot

Swarm is like a categorical scatterplot with non-overlapping points. The data points are adjusted so that they don't overlap. This gives a better representation of the distribution and spread of values.
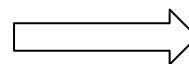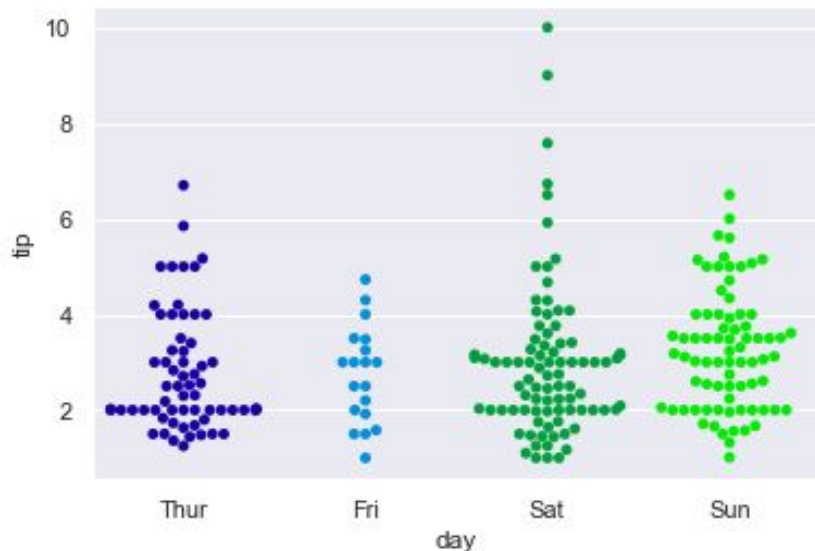

- This plot shows the amount of tip for each day in the data
- We can see that the most number of tips are on Saturday and Sunday
- The amount of tips is maximum on Saturday
- The most common tip on all days is 2 dollars

Back

Distribution Plot

A distribution plot is a method for visualizing the distribution of observations in data. Relative to a histogram, a distribution plot can produce a graph that is less cluttered and more interpretable, especially when drawing multiple distributions
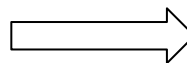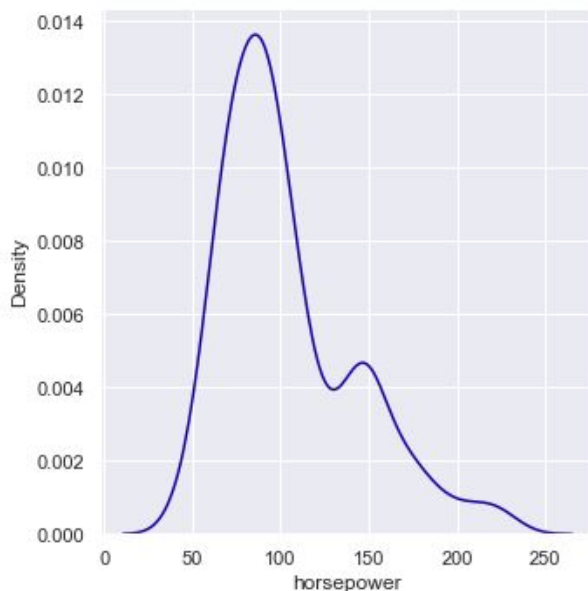
- This plot shows the distribution of horsepower for different types of cars
- We can see that the distribution is slightly right skewed
- Majority of values are less than 100
- The range of values if high. It varies from less than 50 to approx 250.

Pair Plot

- It is used to visualize relationship across multiple combination of variables in a dataset.

- It gives a square matrix of plots where each numeric variable in data will by shared across the y-axes across a single row and the x-axes across a single column.

- The diagonal plots are univariate distribution plot.

- The plot in the figure shows the pairwise relation between all three variables of the mpg data from Searborn - horsepower, weight, and acceleration.

- We can see that horsepower is positively correlated with weight and negatively correlated with acceleration.

- The distribution plot for acceleration shows that it follows a normally distribution.
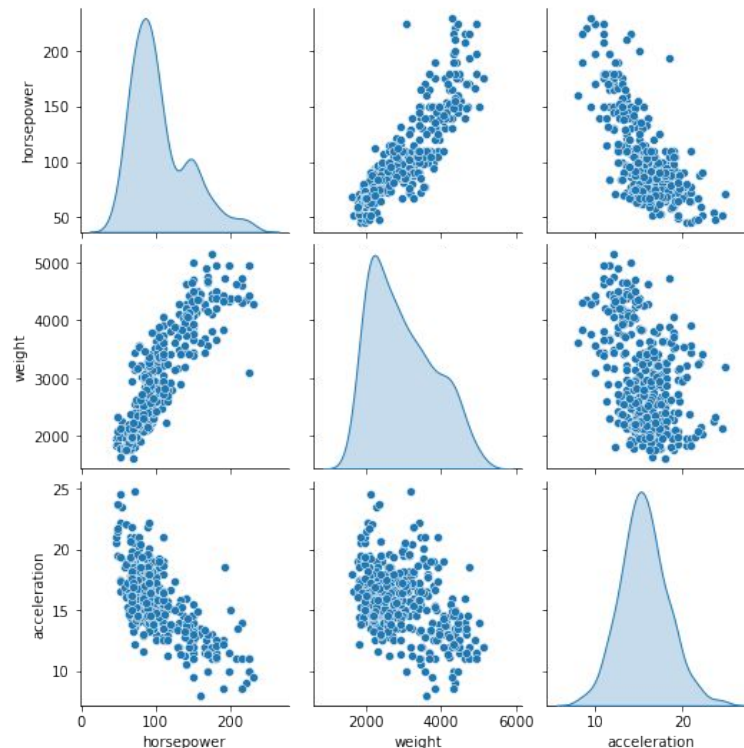
- It is used to visualize the spread of values as a rectangular table using color-encoding to highlight very low and very high values.

- The plot in the figure shows that heatmap for the correlation coefficient between three variables - horsepower, weight, and acceleration.

- The plot shows that acceleration is negatively correlated with horsepower and weight.

- The variable horsepower is positively correlated with weight.

Back

Happy Learning !