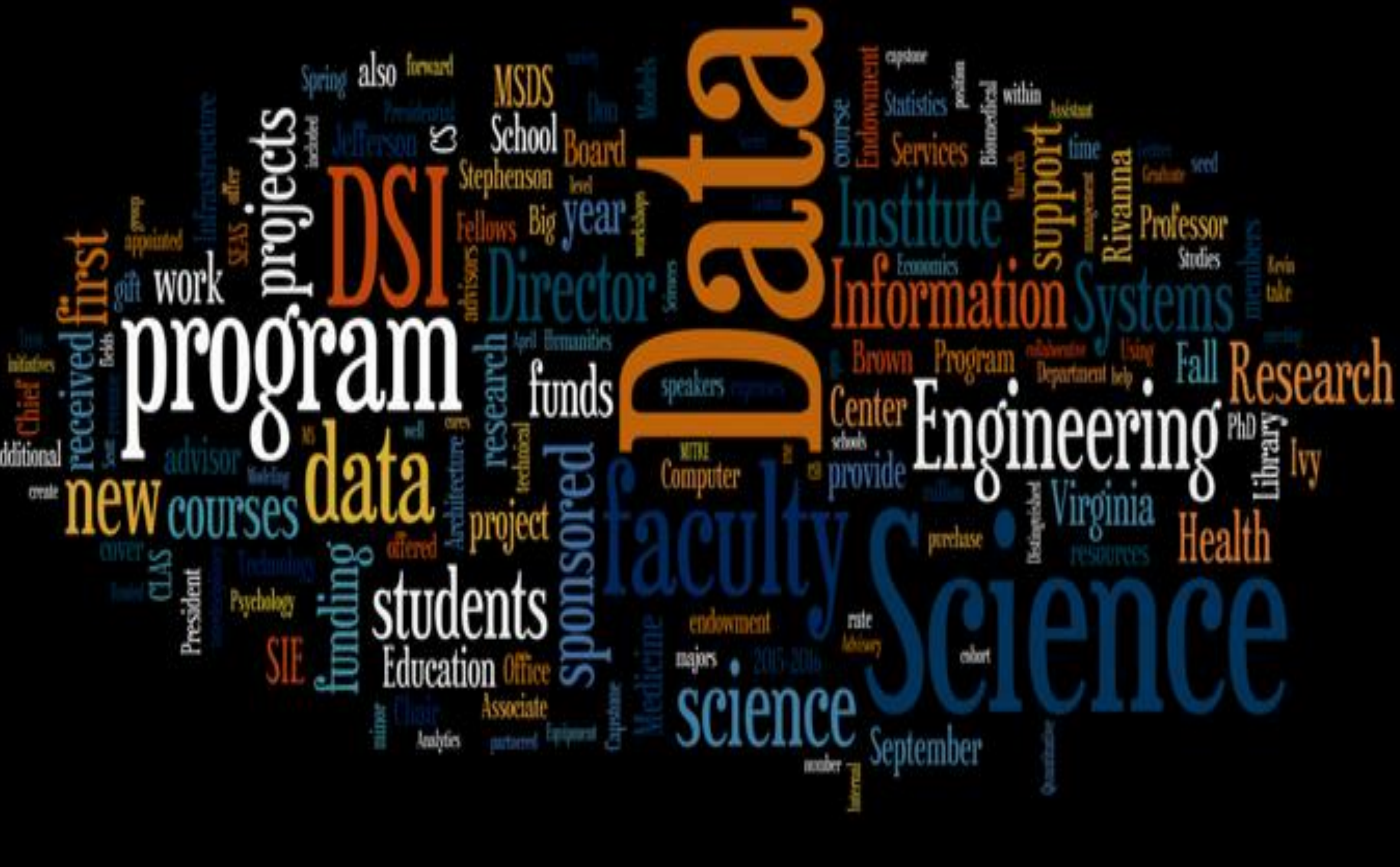
TOOLS FOR DATA SCIENCE (JULY, 2021)

ATUL KUMAR VERMA
19B090004
MENTOR:DEV DESAI

FINAL REPORT



INTRODUCTION

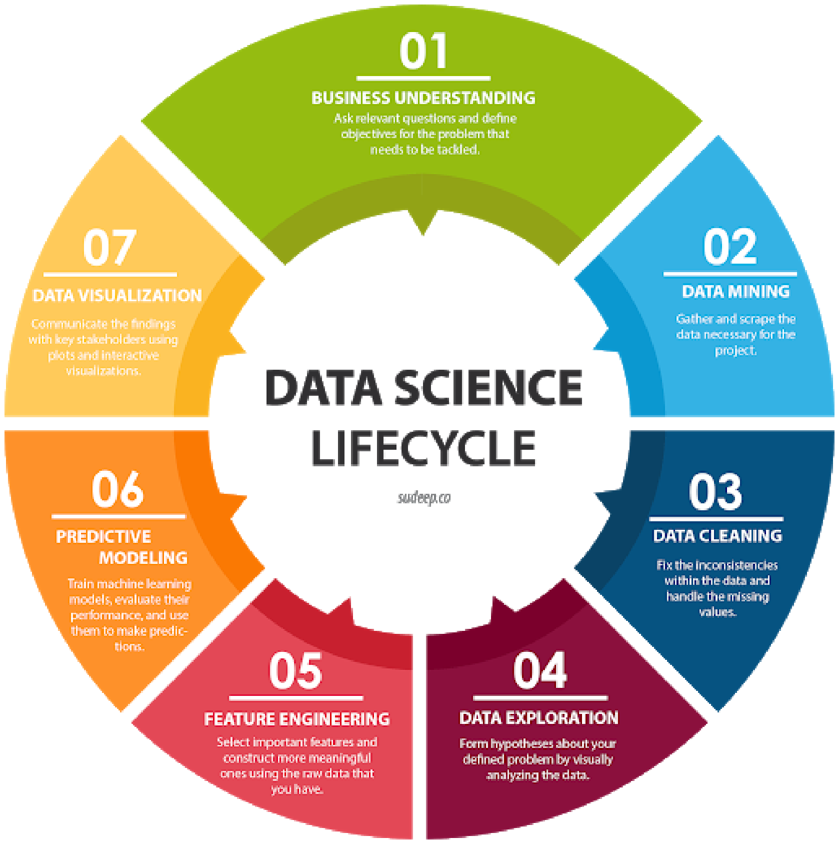
Data science is the practice of mining large data sets of raw data, both structured and unstructured, to identify patterns and extract actionable insight from them. This is an interdisciplinary field, and the foundations of data science include statistics, inference, computer science, predictive analytics, machine learning algorithm development, and new technologies to gain insights from big data.

To define data science and improve data science project management, start with its life cycle. The first stage in the data science pipeline workflow involves capture: acquiring data, sometimes extracting it, and entering it into the system. The next stage is maintenance, which includes data warehousing, data cleansing, data processing, data staging, and data architecture.

Data processing follows, and constitutes one of the data science fundamentals. It is during data exploration and processing that data scientists stand apart from data engineers. This stage involves data mining, data classification and clustering, data modeling, and summarizing insights gleaned from the data—the processes that create effective data.

Next comes data analysis, an equally critical stage. Here data scientists conduct exploratory and confirmatory work, regression, predictive analysis, qualitative analysis, and text mining. This stage is why there is no such thing as cookie cutter data science—when it's done properly.

During the final stage, the data scientist communicates insights. This involves data visualization, data reporting, the use of various business intelligence tools, and assisting businesses, policymakers, and others in smarter decision making.


DATA SCIENCE LIFECYCLE
sudeep.co
01
BUSINESS UNDERSTANDING
Ask relevant questions and define objectives for the problem that needs to be tackled.
02
DATA MINING
Gather and scrape the data necessary for the project.
03
DATA CLEANING
Fix the inconsistencies within the data and handle the missing values.
04
DATA EXPLORATION
Form hypotheses about your defined problem by visually analyzing the data.
05
FEATURE ENGINEERING
Select important features and construct more meaningful ones using the raw data that you have.
06
PREDICTIVE MODELING
Train machine learning models, evaluate their performance, and use them to make predictions.
07
DATA VISUALIZATION
Communicate the findings with key stakeholders using plots and interactive visualizations.

MACHINE LEARNING LIBRARIES

A. NumPy

NumPy is a very popular python library for large multi-dimensional array and matrix processing, with the help of a large collection of high-level mathematical functions. It is very useful for fundamental scientific computations in Machine Learning. It is particularly useful for linear algebra, Fourier transform, and random number capabilities. High-end libraries like TensorFlow uses NumPy internally for manipulation of Tensors.

```
# Python program using NumPy
# for some basic mathematical
# operations

import numpy as np

# Creating two arrays of rank 2
x = np.array([[1, 2], [3, 4]])
y = np.array([[5, 6], [7, 8]])

# Creating two arrays of rank 1
v = np.array([9, 10])
w = np.array([11, 12])

# Inner product of vectors
print(np.dot(v, w), "\n")

# Matrix and Vector product
print(np.dot(x, v), "\n")

# Matrix and matrix product
print(np.dot(x, y))
```

Output:

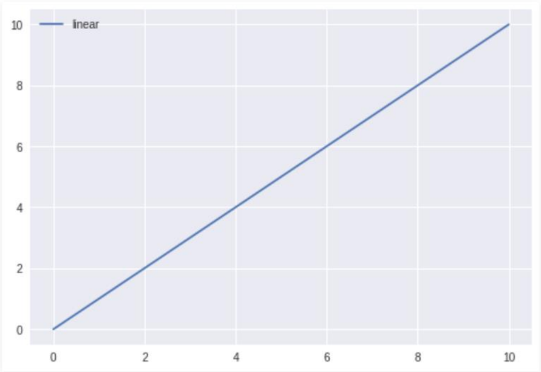
```
219 [29 67] [[19 22] [43
50]]
```

B. MATPLOTLIB

Matpoltlib is a very popular Python library for data visualization. Like Pandas, it is not directly related to Machine Learning. It particularly comes in handy when a programmer wants to visualize the patterns in the data. It is a 2D plotting library used for creating 2D graphs and plots. A module named pyplot makes it easy for programmers for plotting as it provides features to control line styles, font properties, formatting axes, etc. It provides various kinds of graphs and plots for data visualization, viz., histogram, error charts, bar chats, etc,

```
#  Python program using Matplotib
# for forming a linear plot

# importing the necessary packages and modules
import matplotlib.pyplot as plt
import numpy as np

# Prepare the data
x = np.linspace(0, 10, 100)

# Plot the data
plt.plot(x, x, label ='linear')

# Add a legend
plt.legend()

# Show the plot
plt.show()
```

Output

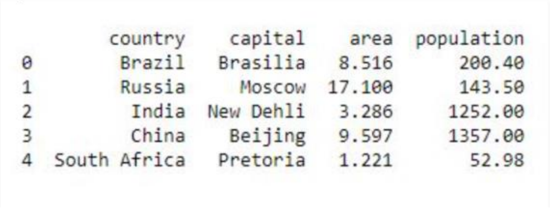


C. PANDAS

Pandas is a popular Python library for data analysis. It is not directly related to Machine Learning. As we know that the dataset must be prepared before training. In this case, Pandas comes handy as it was developed specifically for data extraction and preparation. It provides high-level data structures and wide variety tools for data analysis. It provides many inbuilt methods for groping, combining and filtering data.

```
# Python program using Pandas for
# arranging a given set of data
# into a  table

# importing pandas as pd
import pandas as pd

data = {"country": ["Brazil", "Russia", "India", "China", "South Africa"],
       "capital": ["Brasilia", "Moscow", "New Dehli", "Beijing", "Pretoria"],
       "area": [8.516, 17.10, 3.286, 9.597, 1.221],
       "population": [200.4, 143.5, 1252, 1357, 52.98] }

data_table = pd.DataFrame(data)
print(data_table)
```

Output

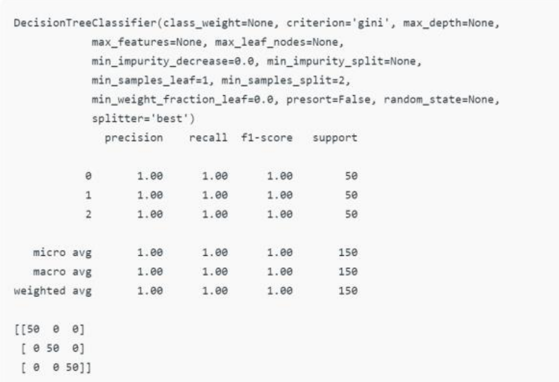


```
        country    capital    area  population
0        Brazil   Brasilia   8.516      200.40
1        Russia     Moscow  17.100      143.50
2         India  New Dehli   3.286     1252.00
3         China    Beijing   9.597     1357.00
4  South Africa   Pretoria   1.221       52.98
```

D. Scikit-learn

Skikit-learn is one of the most popular ML libraries for classical ML algorithms. It is built on top of two basic Python libraries, viz., NumPy and SciPy. Scikit-learn supports most of the supervised and unsupervised learning algorithms. Scikit-learn can also be used for data-mining and data-analysis, which makes it a great tool who is starting out with ML.

```
# Python script using Scikit-learn
# for Decision Tree Clasifier

# Sample Decision Tree Classifier
from sklearn import datasets
from sklearn import metrics
from sklearn.tree import DecisionTreeClassifier

# load the iris datasets
dataset = datasets.load_iris()

# fit a CART model to the data
model = DecisionTreeClassifier()
model.fit(dataset.data, dataset.target)
print(model)

# make predictions
expected = dataset.target
predicted = model.predict(dataset.data)

# summarize the fit of the model
print(metrics.classification_report(expected, predicted))
print(metrics.confusion_matrix(expected, predicted))
```



```
DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=None,
            max_features=None, max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, presort=False, random_state=None,
            splitter='best')
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        50
           1       1.00      1.00      1.00        50
           2       1.00      1.00      1.00        50

   micro avg       1.00      1.00      1.00       150
   macro avg       1.00      1.00      1.00       150
weighted avg       1.00      1.00      1.00       150

[[50  0  0]
 [ 0 50  0]
 [ 0  0 50]]
```

1 Introduction

What is machine learning? We probably use it dozens of times a day without even knowing it. Each time us do a web search on Google or Bing, that works so well because their machine learning software has figured out how to rank what pages. When Facebook or Apple's photo application recognizes our friends in our pictures, that's also machine learning. Each time we read our email and a spam filter saves us from having to wade through tons of spam, again, that's because our computer has learned to distinguish spam from non-spam email. So, that's machinelearning.

One of the reasons I'm excited about this is the **AI**, or artificial intelligence problem.Building truly intelligent machines, we can do just about anything that you or I can do. Many scientists think the best way to make progress on this is through learning algorithms called **neural networks**, which mimic how the human brain works.

1. Defintion

The introduction given above gives us a broad view on machine learning. Even among machine learning practitioners, there isn't a well accepted definition of what is and what isn't machine learning. People tried to define it in different ways. Here aresome:

- **Arthur Samuel(1959):**

  Field of study that gives computers the ability to learn without being explicitly programmed.This is an older, informal definition.

- **Tom Mitchell(1998):**

  A computer program is said to learn from experience **E** with respect to some task **T** and some performance **P**, if its performance on **T**, as measured by **P**, improves with experience **E**.

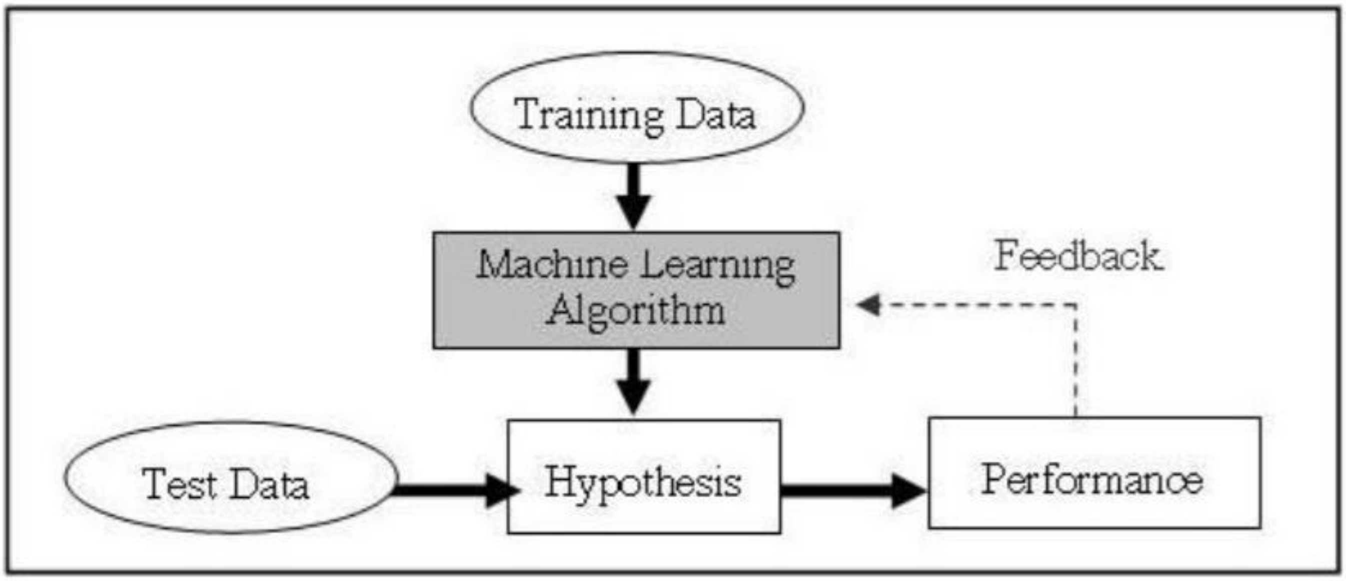
2. Types

In general, any machine learning problem can be assigned to one of two broad classifications:

- **Supervised Learning**
- **Unsupervised Learning**

INTRODUCTION TO MACHINE LEARNING

Machine learning in general is a superset of Neural networks which involves literally anything which makes a machine learn by providing data. A more formal definition by Tom Mitchell of machine learning can be scripted in the following way: "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E."



Machine Learning Algorithms:

❑ **Supervised machine learning**

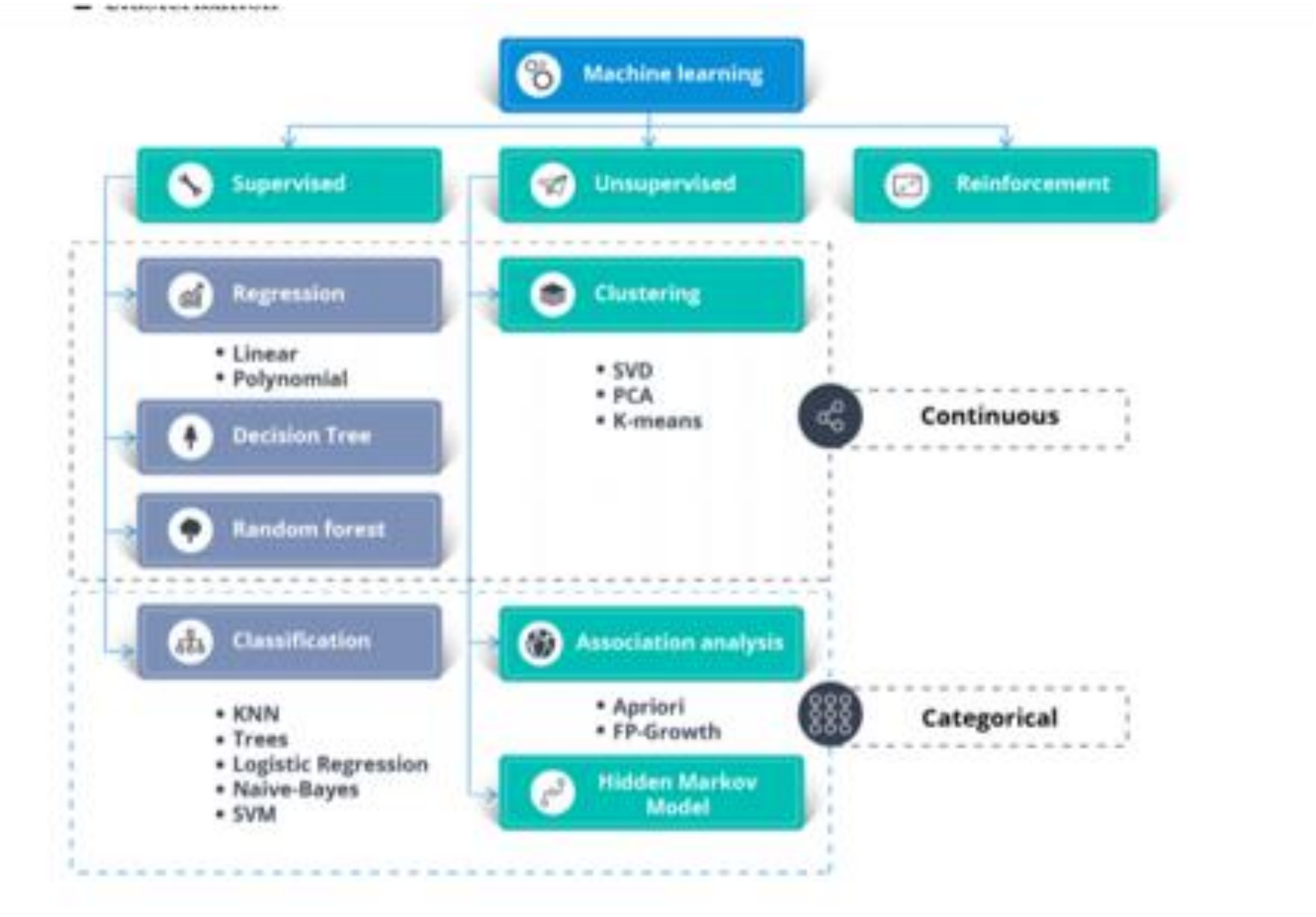
❑ **Unsupervised machine learning**

❑ **Semi-supervised machine learning**

❑ **Reinforcement machine learning**

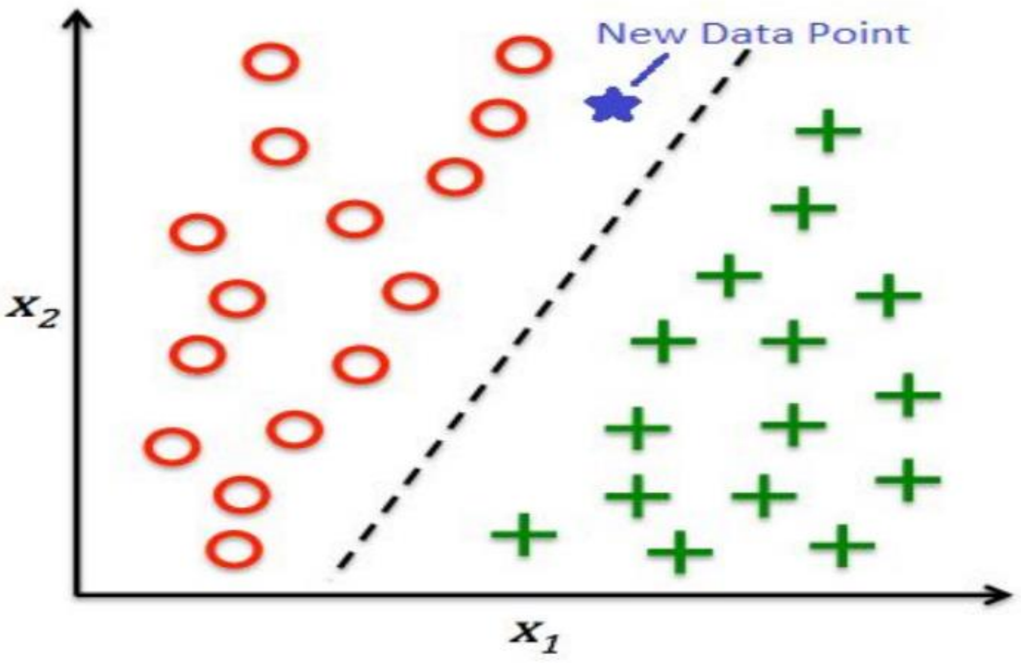
They are often also classified as:

- ❑ **Regressions**
- ❑ **Classification**
- ❑ **Clusterisation**

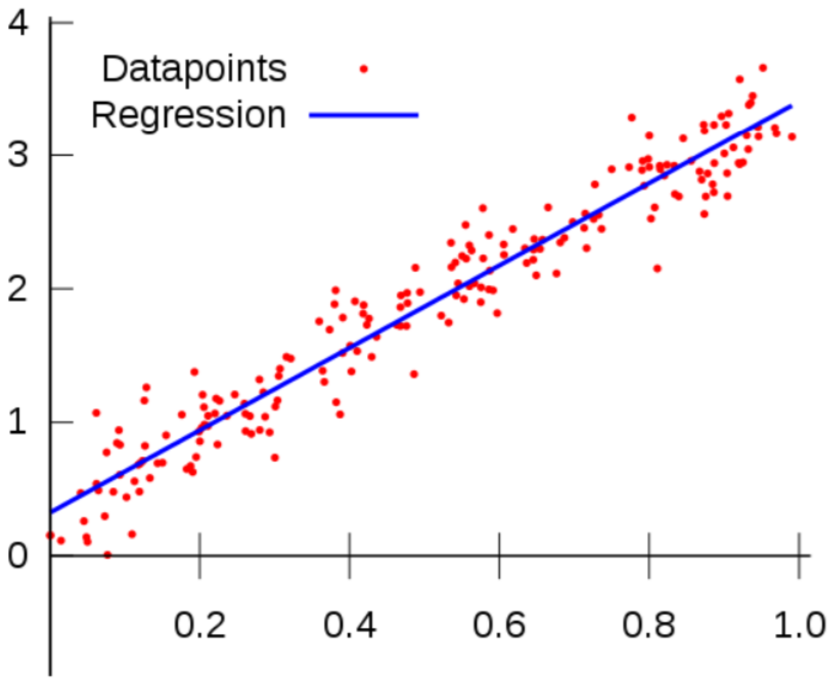


1. Supervised Learning: The main goal in supervised learning is to learn a model from labeled training data that allows us to make predictions about unseen or future data. Here, the term supervised refers to a set of samples where the desired output signals (labels) are already known.Classification and Regression are two subcategories of supervised learning.

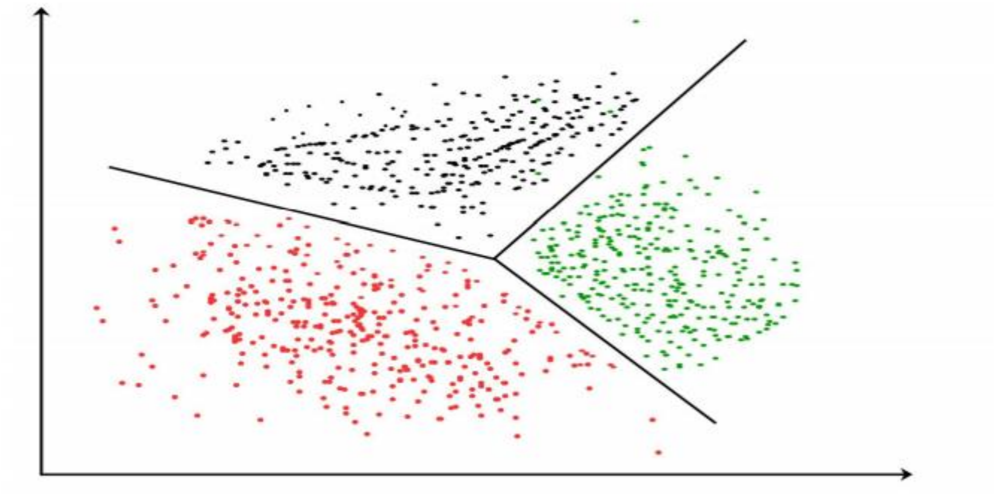
➢ **Classification**: Classification is a subcategory of supervised learning where the goal is to predict the categorical class labels of new instances based on past observations.Those class labels are discrete, unordered values that can be understood as the group memberships of the instances. Eg:- we can train a model using a supervised machine learning algorithm on a corpus of labeled email, e-mail that are correctly marked as spam or not-spam, to predict whether a new email belongs to either of the two categories.



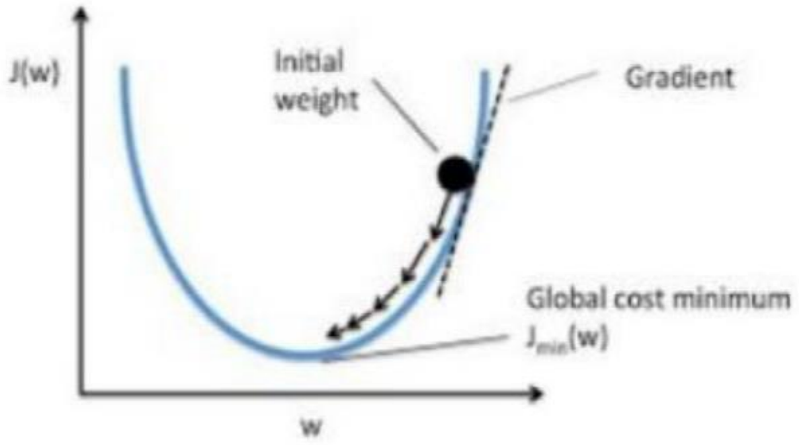
➢ **Regression:** In regression analysis, we are given a number of predictor (explanatory) variables and a continuous response variable(outcome), and we try to find a relationship between those variables that allows us to predict an outcome. Eg:- let's assume that we are interested in predicting the Math SAT scores of our students. If there is a relationship between the timespent studying for the test and the final scores, we could use it as training data to learn a model that uses the study time to predict the test scores of future students who are planning to take this test



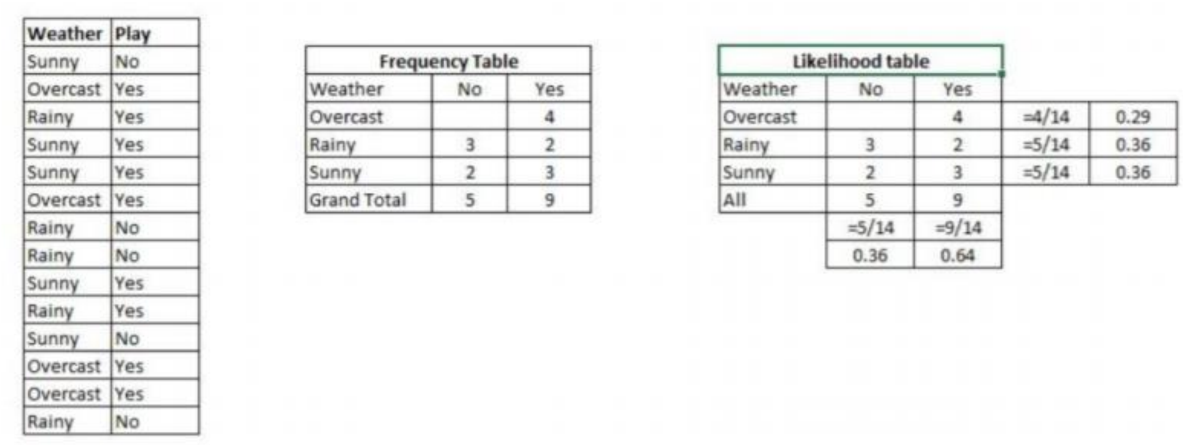
2. **Unsupervised Learning:** In supervised learning, we know the right answer beforehand when we train our model, and in reinforcement learning, we define a measure of rewardfor particular actions by the agent. In unsupervised learning, however, we are dealing with unlabeled data or data of unknown structure. Using unsupervised learning techniques, we are able to explore the structure of our data to extract meaningful information without the guidance of a known outcome variable or reward function. Clustering is the main algorithm used in unsupervised learning. ➢ Clustering: It is an exploratory data analysis technique that allows us to organize a pile of information into meaningful subgroups (clusters) without having any prior knowledge of their group memberships. Each cluster that may arise during the analysis defines a group of objects that share a certain degree of similarity but are more dissimilar to objects in other clusters, which is why clustering is also sometimes called "unsupervised classification." Clustering is a great technique for structuring information and deriving meaningful relationships among data. Eg:- It allows marketers to discover customer groups based on their interests in order to develop distinct marketing programs.



MACHINE LEARNING ALGORITHMS Although all of the below mentioned stuff is just one line of code in many advanced libraries like TensorFlow and Scikit Learn but still getting inside the heart of everything never hurts. 1. Linear Regression: Linear regression is used for finding linear relationship between target and one or more predictors . The core idea is to obtain a line that best fits the data. The best fit line is the one for which total prediction error (all data points) are as small as possible. Error is the distance between the points to the regression line. This is called linear hypothesis, here θi ’s are called parameters and xi ’s are input variables.



2. Naïve Bayes Classifier Algorithm Let's understand it using an example. Below I have a training data set of weather and corresponding target variable 'Play' (suggesting possibilities of playing).



Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9

Likelihood table				
Weather	No	Yes		
Overcast		4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		

Now, we need to classify whether players will play or not based on weather condition. Let's follow the below steps to perform it.

Step 1: Convert the data set into a frequency table

Step 2: Create Likelihood table by finding the probabilities like Overcast probability = 0.29 and probability of playing is 0.64.

Step 3: Now, use Naive Bayesian equation to calculate the posterior probability for each class. The class with the highest posterior probability is the outcome of prediction.

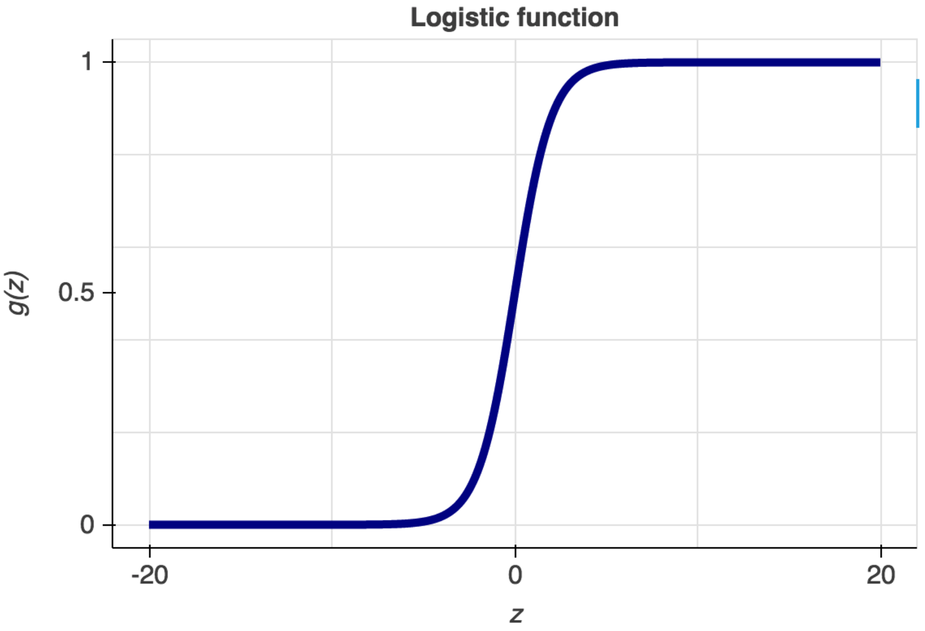
Problem: Players will play if weather is sunny. Is this statement is correct? We can solve it using above discussed method of posterior probability. P(Yes | Sunny) = P(Sunny | Yes) * P(Yes) / P (Sunny) Here we have P (Sunny |Yes) = 3/9 = 0.33, P(Sunny) = 5/14 = 0.36, P(Yes)= 9/14 = 0.64 Now, P (Yes | Sunny) = 0.33 * 0.64 / 0.36 = 0.60, which has higher probability.Naive Bayes uses a similar method to predict the probability of different class based on various attributes. This algorithm is mostly used in text classification and with problems having multiple classes.

3. **K MEANS CLUSTERING ALGORITHM** k-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed apriori. The main idea is to define k centers, one for each cluster. These centers should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest center. When no point is pending, the first step is completed and an early group age is done. At this point we need to re-calculate k new centroids as barycenter of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new center. A loop has been generated. As a result of this loop we may notice that the k centers change their location step by step until no more changes are done or in other words centers do not move any more. Finally, this algorithm aims at minimizing an objective function know as squared error function given by:

$$J(V)=\sum_{i=1}^{c}\sum_{j=1}^{c_i}\left(\left\|x_i-v_j\right\|\right)^2$$

‘||xi - vj ||’ is the Euclidean distance between xi and vj . ‘ci ’ is the number of data points in i th cluster. ‘c’ is the number of cluster centers.

4. **Logistic regression** It is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model (a form of binary regression).

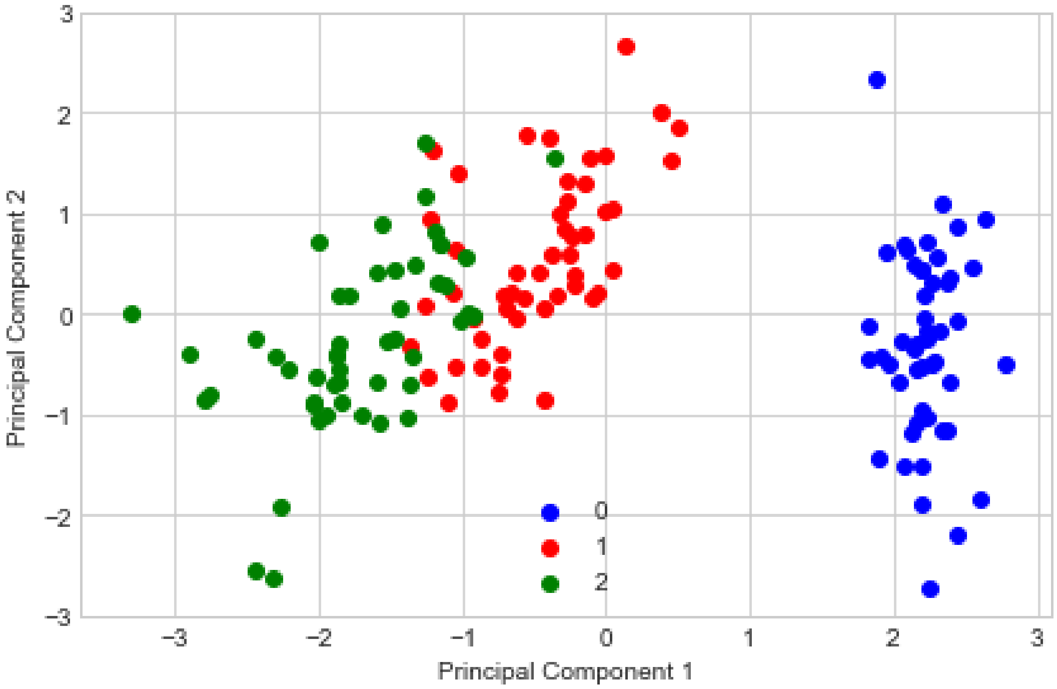


Principal Component Analysis(PCA):

Principal Component Analysis is an unsupervised learning algorithm that is used for the dimensionality reduction in machine learning. It is a statistical process that converts the observations of correlated features into a set of linearly uncorrelated features with the help of orthogonal transformation. These new transformed features are called the **Principal Components**. It is one of the popular tools that is used for exploratory data analysis and predictive modeling. It is a technique to draw strong patterns from the given dataset by reducing the variances.

PCA generally tries to find the lower-dimensional surface to project the high-dimensional data.

PCA works by considering the variance of each attribute because the high attribute shows the good split between the classes, and hence it reduces the dimensionality. Some real-world applications of PCA are ***image processing, movie recommendation system, optimizing the power allocation in various communication channels.*** It is a feature extraction technique, so it contains the important variables and drops the least important variable.



GitHub REPOS:

FOR THE TASKS AND READING MATERIALS:
https://github.com/Tools-For-Data-Science-SOC
https://github.com/atul4411/SoC-Tasks

FOR THE FINAL PROJECT:
https://github.com/atul4411/OCR-Pytesseract