**19CS82C - Internship/In-plant Training Report**

***Submitted by***

# RAGUL T N - 1912021

***in partial fulfillment for the award of the degree of***

# BACHELOR OF ENGINEERING

**IN**

**COMPUTER SCIENCE AND ENGINEERING**



**NATIONAL ENGINEERING COLLEGE**

**(An Autonomous Institution, Affiliated to Anna University, Chennai)**

**K.R. NAGAR, KOVILPATTI - 628 503.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| P a g e | **2**  **TABLE OF CONTENTS** | | | | |
|  | **Serial Number** | **Content Topics** | **Page Number** |  |
|  | 1 | **WEEKLY RECORDS** | 5 |  |
|  | 2 | **CHAPTER 1 - INTRODUCTION** | 10 |  |
|  | 3 | * ABOUT CUREYA | 10 |  |
|  | 4 | * OBJECTIVE | 10 |  |
|  | 5 | * MISSION | 10 |  |
|  | 6 | * COMPANY INFORMATION | 10 |  |
|  | 7 | **CHAPTER 2 – INTERNSHIP SCHEDULE** | 11 |  |
|  | 8 | * INTERNSHIP DURATION | 11 |  |
|  | 9 | * JOB SCOPE | 11 |  |
|  | 10 | **CHAPTER 3** | 12 |  |
|  | 11 | **Skills Every Well-Rounded Programmer Should Know** | 12 |  |
|  | 12 | **Difference Between AI, ML and DL** | 12 |  |
|  | 13 | * What is Data Science | 12 |  |
|  | 14 | **History of AI** | 13 |  |
|  | 15 | **Demand for AI** | 14 |  |
|  | 16 | **What is AI?** | 14 |  |
|  | 17 | **Learn Python Using Jupyter Notebook** | 14 |  |
|  | 18 | **AI Applications** | 15 |  |
|  | 19 | **The Evolutionary Stages of AI** | 15 |  |
|  | 20 | * Artificial Narrow Intelligence (ANI) | 15 |  |
|  | 21 | * Artificial General Intelligence (AGI) | 15 |  |
|  | 22 | * Artificial Super Intelligence (ASI) | 15 |  |
|  | 23 | **Programming Languages for AI** | 16 |  |
|  | 24 | * Python | 16 |  |
|  | 25 | * R | 16 |  |
|  | 26 | * Java | 16 |  |
|  | 27 | * Lisp | 16 |  |
|  | 28 | * Prolog | 16 |  |
|  | 29 | **Python – The New Generation Language** | 16 |  |
|  | 30 | * Why Python? | 16 |  |
|  | 31 | * Features of Python | 17 |  |
|  | 32 | **Python Applications** | 18 |  |
|  | 33 | **Introduction to Machine Learning** | 19 |  |
|  | 34 | * Need for Machine Learning | 19 |  |
|  | 35 | * What is Machine Learning? | 19 |  |
|  | 36 | **Applications of Machine Learning** | 19 |  |
|  | 37 | **Machine Learning Definitions** | 20 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| P a g e | **3** | | | | |
|  | 44. | 2) ML Model | 20 |  |
|  | 45. | 3) Predictive Variable | 20 |  |
|  | 46. | 4) Response Variable | 20 |  |
|  | 47. | 5) Training Data | 20 |  |
|  | 48. | 6) Testing Data | 21 |  |
|  | 49. | **Machine Learning Process** | 21 |  |
|  | 50. | * **Steps in Machine Learning Process** | 21 |  |
|  | 51. | * Define objective | 21 |  |
|  | 52. | * Data Gathering | 22 |  |
|  | 53. | * Preparing Data | 22 |  |
|  | 54. | * Data Exploration | 22 |  |
|  | 55. | * Building a model | 22 |  |
|  | 56. | * Model evaluation | 23 |  |
|  | 57. | * Predictions making | 23 |  |
|  | 58. | **Machine Learning Types** | 23 |  |
|  | 59. | * Supervised Learning | 23 |  |
|  | 60. | * Unsupervised Learning | 24 |  |
|  | 61. | * Reinforcement Learning | 24 |  |
|  | 62. | **Types of Problems in Machine Learning** | 25 |  |
|  | 63. | * Regression | 25 |  |
|  | 64. | * Classification | 25 |  |
|  | 65. | * Clustering | 25 |  |
|  | 66. | **Machine Learning Algorithms** | 26 |  |
|  | 67. | * Training Phase | 26 |  |
|  | 68. | * Test Phase | 26 |  |
|  | 69. | **Explanation of Important Machine Learning Algorithms** | 26 |  |
|  | 70. | * Linear Regression | 26 |  |
|  | 71. | * Logistic regression | 27 |  |
|  | 72. | * Support Vector Machine (SVM) | 28 |  |
|  | 73. | * Decision Trees | 28 |  |
|  | 74. | * Random Forest | 29 |  |
|  | 75. | * Naïve Bayes | 30 |  |
|  | 76. | * KNN (K Nearest Neighbor) | 31 |  |
|  | 77. | * K – Means | 32 |  |
|  | 78. | * Elbow Method | 33 |  |
|  | 79. | * Purpose Based K-Means Clustering | 33 |  |
|  | 80. | **Some Important Considerations in Machine Learning** | 34 |  |
|  | 81. | * Bias Variance Trade-off | 34 |  |
|  | 82. | * Bias Variance Dependencies | 34 |  |
|  | 83. | * Cross-Validation | 34 |  |
|  | 84. | * Types of Cross-Validation | 34 |  |
|  | 85. | * Exhaustive Cross-Validation | 35 |  |
|  | 86. | * Non-exhaustive Cross-Validation | 35 |  |
|  | 87. | * Confusion Matrix | 35 |  |
|  | 88. | * What is a Confusion Matrix? | 35 |  |
|  | 89. | * Accuracy and Components of Confusion Matrix | 36 |  |
|  | 90. | * Accuracy | 36 |  |
|  | 91. | * Recall | 36 |  |
|  | 92. | * Precision | 36 |  |
|  | 93. | * F – Measure | 37 |  |
|  | 94. | **Drawbacks of Machine Learning** | 38 |  |
|  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| 96. | **What is Deep Learning?** | 39 |
| 97. | * Simple Definition of a Neural Network | 40 |
| 98. | * Deep Learning Use Case | 40 |
| 99. | **Limitations of Machine Learning (Need for Deep Learning)** | 41 |
| 100. | * Curse of Dimensionality | 42 |
| 101. | * Manual Feature Engineering | 42 |
| 102. | * Not Ideal for Object Detection and Image Recognition | 42 |
| 103. | **Deep Learning Use Case/ Applications** | 42 |
| 104. | **How does a Neural Network Works?** | 43 |
| 105. | * What is Perceptron? | 43 |
| 106. | * Weight and Bias in Deep Learning | 43 |
| 107. | * Summation Function | 44 |
| 108. | * Activation Function | 44 |
| 109. | **Neural Networks Explained with An Example** | 44 |
| 110. | **Natural Language Processing (NLP)** | 45 |
| 111. | **Applications of Deep Learning** | 45 |
| 112. | * Speech Recognition | 45 |
| 113. | * Automatic Machine Translation | 46 |
| 114. | * Instant Visual Translation | 46 |
| 115. | * Bahaviour: Automated Self-Driven Cars | 47 |
| 116. | **Data Visualization** | 48 |
| 117. | * General Types of Visualizations | 48 |
| 118. | **Tableau** | 49 |
| 119. | * Tableau Intro | 49 |
| 120. | * Tableau Architecture | 50 |
| 121. | * Process Flow of BI Project | 50 |
| 122. | * Tableau Data Types | 50 |
| 123. | * Connections in Tableau | 50 |
| 124. | * Joins in Tableau | 50 |
| 125. | **Feature Engineering** | 51 |
| 126. | **Machine Learning Pipelines** | 52 |
| 127. | **PyTorch** | 53 |
| 128. | **CHAPTER 4** | 53 |
| 130. | **LEARNING OUTCOME** | 54 |
| 133. | **CERTIFICATE OF COMPLETION** | 51 |
| 134. | **CONCLUSION** | 52 |

## WEEKLY RECORDS

|  |  |  |  |
| --- | --- | --- | --- |
| **WEEK NUMBER** | | **DATES** | **ROLE** |
| 1 | | 25/07/2021 - 31/07/2021 | INTERN |
| * ARTIFICIAL INTELLIGENCE | | | |
|  | * Essential skills for well-rounded Programmer * Difference between AI, ML and Deep Learning * AI vs ML vs Deep Learning vs Data Science * AI Full Course * Jupyter Notebook from scratch | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **WEEK NUMBER** | | **DATES** | **ROLE** |
| 2 | | 01/08/2021 - 07/08/2021 | INTERN |
| * MACHINE LEARNING | | | |
|  | * What is Machine Learning? * ML algorithm and its applications * ML-Types of Tasks, applications and approaches * Machine Learning Fundamentals explanation with diagrams and graphs * Machine Learning with Python | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **WEEK NUMBER** | | **DATES** | **ROLE** |
| 3 & 4 | | 08/08/2021 - 14/08/2021 | INTERN |
| * DATA SCIENCE | | | |
|  | * Complete Machine Learning * Statistics in Machine Learning * Feature Engineering * Machine Learning Pipelines * Data Stacks * Complete Deep Learning * PyTorch * Data Visualization * Tableau * AI and Deep Learning Projects | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **WEEK NUMBER** | | **DATES** | **ROLE** |
| 5 & 6 | | 15/08/2021 - 25/08/2021 | INTERN |
| * PYTHON | | | |
|  | * Python for beginners * Django for beginners * Django-Building LMS (Learning Management System) * Python-Django Web Framework * How to build E-commerce website with Django and Python * Advanced Python * Numpy * Pandas * MongoDb with Python * Introduction to Mobile App Development * Top 10 mobile application framework | | |

## CHAPTER 1



### INTRODUCTION

**ABOUT CUREYA**

CUREYA (registered as Aspexx Health solutions Pvt Ltd, under MCA) is DPIIT recognized start up, registered under 'STARTUP INDIA SCHEME'. CUREYA collaborated with stakeholders include - World Yoga associations, Flag bits technologies and many more.

CUREYA provide natural healthcare solutions that aims to reduce the medical health expenditure and eliminate the information asymmetry. Medical practitioners are available at CUREYA via video call, chat, email, or phone call system.

CUREYA provide all kinds of medical services related to Ayurveda, Yoga & Naturopathy, Unani, Siddha, and Homoeopathy.

**OBJECTIVE**

CUREYA’s primary objective is ‘HEALTH FOR ALL’, by reducing the medical expenditure, eliminating the information asymmetry, promoting health awareness and achieving inclusive & holistic approach for healthcare treatments.

The objective is to eliminate the information asymmetry, language barrier, and emphasize to achieve global standards of healthcare delivery systems based on access, equity, affordability and quality, efficiency, sustainability.

**MISSION**

CUREYA’s mission is to achieve the right to "Health for All" and improve the healthcare indicators by dissemination of health education that focuses on health promotion, health prevention, and self- medication.

**COMPANY INFORMATION**

Company Name: **CureYa**

Founder & CEO: Shivani Mishra Industry: Healthcare & Lifesciences

Location: Greater Noida, Uttar Pradesh, India E-mail: [info@cureya.in](mailto:info@cureya.in)

Phone: 9990375133

* **Useful Links:**
  + Website: [https://cureya.in](https://cureya.in/)
  + Facebook: <https://www.facebook.com/cureya7/>
  + LinkedIn: <https://www.linkedin.com/company/cureya/>
  + Instagram: <https://www.instagram.com/cureya.in/?hl=en>
  + YouTube: <https://www.youtube.com/channel/UCjsRwGm--mr1ADln5CB5Siw>

## CHAPTER 2

### INTERNSHIP SCHEDULE

**INTERNSHIP DURATION**

It was a 2 months’ work from home (WFH) internship. The internship started on 25th April, 2021 and completed on 25th June, 2021. During this period of 8 weeks of internship, firstly, I was introduced to basic concepts of artificial intelligence, machine learning, deep learning and data science. And then implement the learning into projects.

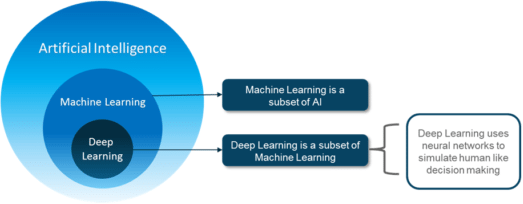
**JOB SCOPE**

I was assigned various tasks and projects during this internship. I completed all the tasks successfully under the guidance of supervisors and mentors. This internship enables me to get relevant jobs in the data industry such as data analyst, data engineer, data scientist, and machine learning engineer.

## CHAPTER 3

#### 5 essential skills every well-rounded programmer should know:

1. Version Control (GIT)
2. Databases (SQL) (NO-SQL) (POSTGRES)
3. Command Line (Terminal)
4. Unit Testing (Bonus: Continuous Integration/Delivery)
5. Learn Multiple Languages

**Difference between AI, ML and DL**

##### Figure 1- Difference between AI, ML and DL

People often tend to think that [Artificial Intelligence](https://www.edureka.co/blog/artificial-intelligence-tutorial/), [Machine Learning](https://www.edureka.co/blog/introduction-to-machine-learning/), and [Deep Learning](https://www.edureka.co/blog/deep-learning-tutorial) are the same since they have common applications. For example, Siri is an application of AI, Machine learning and Deep learning.

So how are these technologies related?

* + **Artificial Intelligence** is the science of getting machines to mimic the behavior of humans.
  + **Machine learning** is a subset of Artificial Intelligence (AI) that focuses on getting machines to make decisions by feeding them data.
  + **Deep learning** is a subset of Machine Learning that uses the concept of neural networks to solve complex problems.

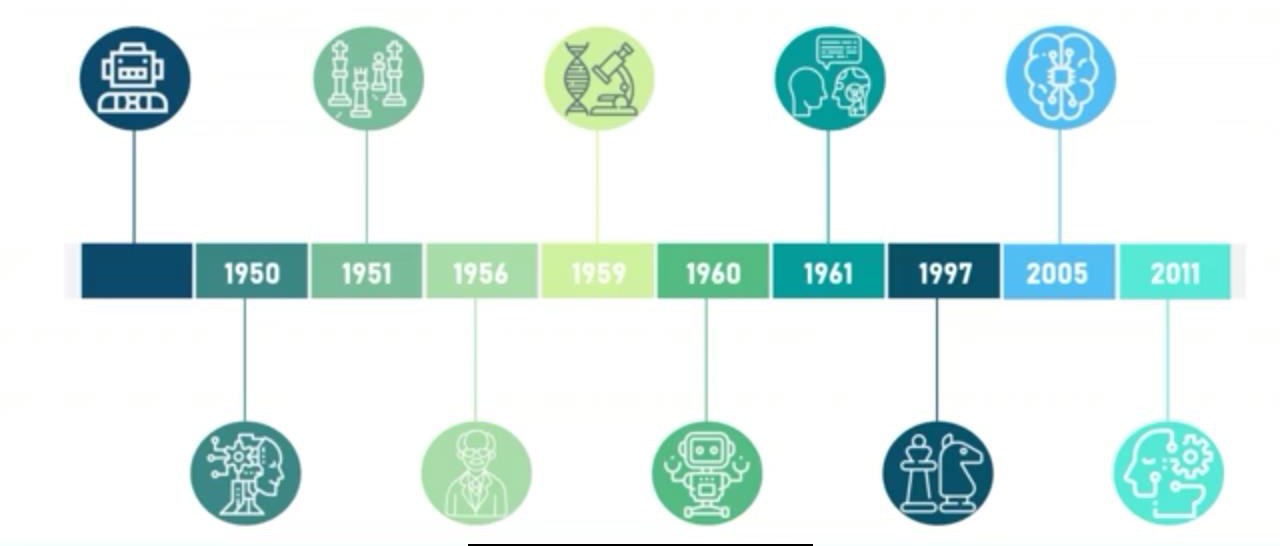
To sum it up AI, Machine Learning and Deep Learning are interconnected fields. Machine Learning and Deep learning aids Artificial Intelligence by providing a set of algorithms and neural networks to solve data- driven problems.

###### What is Data Science?

Data science is a broad field that spans the collection, management, analysis and interpretation of large amounts of data with a wide range of applications. It integrates all the terms above and summarizes or extract insights from data (exploratory data analysis) and make predictions from large datasets (predictive analytics).

## History of AI

* The concept of Artificial Intelligence goes back to the classical ages. Though machine learning models of AI looks complex, its history is quite interesting.



##### Figure 2-History of AI

**Greek Mythology-Talos**

Under Greek mythology, the concept of machines and mechanical men were well thought of. So, an example of this is Talos. Talos was a giant animated bronze warrior programmed to guard the Island of Crete created by the Hephaesius. No one knows if that’s true but concept of AI can be seen here.

**1950-Alan Turing**

Alan Turing published a landmark paper in which he speculated about the possibility of creating machines that think.

**TURING TEST:** Then we come all the way to 1950 when Alan Turing speculated the possibility of creating machines that could think. A certain test known as Turing test was created to test the efficiency of AI systems. The test states that if a machine could converse which would be indistinguishable with human characters, then the machine is said to have passed the rest. But unfortunately, no machines have been able to pass this test.

**1951-Game AI**

Christopher Strachey wrote a checkers program and Dietrich Prinz wrote one for chess.

**1956-The birth of AI**

John McCarthy first coined the term- “Artificial Intelligence” in 1956 at the Dartmouth Conference.

**1959-First AI Laboratory**

MIT (Manchester Institute of Technology) AI lab was first set up in 1959. The research on AI began.

**1960-General Motors Robot**

First robot was introduced to General Motors assembly line.

**1961-First Chatbot**

The first AI chatbot called ELIZA was introduced in 1961.

**1997-IBM Deep Blue**

IBM’s Deep Blue beats world champion Garry Kasparov in the game of chess. This was the first major achievement of AI.

**2005-DARPA Grand Challenge**

Stanford Racing Team’s autonomous robotic car. Stanley wins the 2005 DARPA Grand Challenge.

**2011-IBM Watson**

IBM’s question answering system. Watson, defeated the 2 greatest Jeopardy! Champions Brad rutter & Ken Jennings.

* And from then on AI has grown in a faster pace solving many of the world’s problems in healthcare, education, accidents etc.

 WHAT STARTED AS AN HYPOTHETICAL CONCEPT IS NOW TAKING THE WORLD BY STORM.

* AI includes machine learning, deep learning, neural networks, natural language processing, computer

vision, knowledge base, expert systems, image processing & so on…

###### Why AI now? (Demand for AI)

1. More computational power (GPU)
2. More data
3. Better Algorithms
4. Broad Investment (all big giants invest in AI)

###### What is AI?

John McCarthy first coined the term Artificial Intelligence in the year 1956 at the Dartmouth Conference. John McCarthy defined Artificial Intelligence as the science and engineering of making intelligent machines. In other words, Artificial Intelligence is the theory and development of computer systems able to perform tasks normally requiring human intelligence, such as visual perception, speech recognition, decision-making and translation between languages.

Thus, AI is a term for machines to work and behave like humans. In the recent past AI has been able to do this by creating machines and robots that have been used in wide range of fields including healthcare, robotics, marketing, business analytics and many more.

###### Learn Python using Jupyter Notebook

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modelling, data visualization, machine learning, and much more.

Jupyter Notebooks are a spin-off project from the IPython project, which used to have an IPython Notebook project itself. The name, Jupyter, comes from the core supported programming languages that it supports: Julia, Python, and R.Jupyter ships with the IPython kernel, which allows you to write your programs in Python, but there are currently over 100 other kernels that you can also use.

**AI Applications**

* **Google Predictive Search Engine:** When you begin typing a search term, and Google makes recommendations for you to choose from, that is artificial intelligence in action. So, predictive searches are based on data that Google collects about you, such as your browser history, your location, your age and other personal details. By using AI, Google attempts to guess what you might be trying to find. Behind this, there’s a lot of natural language processing, deep learning and machine learning involved.
* **JP Morgan Chase’s Contract Intelligence (COIN)** platform uses AI, machine learning and image recognition software to analyze legal documents.
* **IBM Watson:** Healthcare organizations use IBM AI (Watson) technology for medical diagnosis.
* **Google’s AI Eye Doctor** can examine retina scans and identify a condition called diabetic retinopathy.
* **Facebook** uses machine learning and deep learning to detect facial features & tag your friends
* **Twitter** is using AI to identify hate speech and terrorist language in tweets. The company discovered & banned 300000 terrorist-linked accounts, 95% of which were found by non-human, artificially intelligent machines.
* **Google Assistant:** The Google Duplex cannot only respond to calls and book appointments for you, it adds a human touch.
* **Tesla:** Tesla will have fully self-driving cars and “robo taxi” version-one that can ferry passengers without anyone behind the wheel-ready for the streets next year (self-driving cars uses computer vision, image detection, and deep learning).
* **Netflix:** Over 75% of what you watch is recommended by Netflix. Recommendations are made by machine learning algorithms.
* **Spam Filtering:** Machine learning and natural language processing are used in spam filtering.

###### Three evolutionary stages of AI:

* **Artificial Narrow Intelligence (ANI):** ANI is also known as Weak AI. It involves applying AI only to specific tasks. E.g., self-driving cars, Alexa, Google’s search engine, Sofia, The Humanoid, AlphaGo. We’re at this stage right now.
* **Artificial General Intelligence (AGI):** AGI is also known as strong AI. It involves machines that possess the ability to perform any intellectual task that a human being can.
* **Artificial Super Intelligence (ASI):** ASI is a term referring to the time when the capability of computers will surpass human beings. E. g., science fiction, movies like terminator. It’s a speculation that this will take over the world by the 2040.

**Programming Languages for AI**

**Python>>** Most famous and effective language for AI because a lot of developers prefer to use Python. Data scientists prefer to use the Python language. This is partly because the syntaxes which belong to Python are very simple and they can be learned very easily. It’s considered to be one of the easiest languages to learn and also many other AI algorithms and machine learning algorithms can be easily implemented in Python, because there are lot of libraries which are predefined functions for these algorithms. So, all you have to do is call that function. You don’t actually have to code your algorithm. So, python is considered the best choice for artificial intelligence.

**R>>** With Python, stands R, which is a statistical programming language. R is one of the most effective language and environment for analyzing and manipulating the data for statistical purpose. It is a statistical programming language. So, using R we can easily produce well designed publication quality plots, including mathematical symbol and formula, wherever needed. If you ask me, I think R is also one of the easiest programming languages to learn. The syntax is very similar to English language, and it also has N number of libraries that support statistics, data science, AI, machine learning, and so on. It also has pre-defined functions for machine learning algorithms, natural language processing, and so on. So, R is also a very good choice if you want to get started with programming languages for machine learning and AI.

**Java>>** Java can also be considered as a good choice for AI development. Artificial Intelligence has a lot to do with search algorithms, artificial neural networks, and genetic programming. Java provides many benefits. It’s easy to use. Debugging is very easy, package services. There’s simplified work with large scale projects. There’s a good user interaction and graphical representation of data. It has something known as the standard widget toolkit, which can be used for making graphs and interfaces. So, graphic virtualization is actually a very important part of AI, or data science, or machine learning for that matter.

**Lisp>>** Shockingly, a lot of people have not heard of this language. This is actually the oldest and the most suited language for the development of artificial intelligence. It is considered to be a language which is very suited for the development of artificial intelligence. Now, let me tell you that this language was invented by John McCarthy, who’s also known as the father of artificial intelligence. He’s the person who coined the term “Artificial Intelligence”. It has the capability of processing symbolic information. It has excellent prototyping capabilities. It is easy, and it creates dynamic objects with a lot of ease. There’s automatic garbage collection in all of that. But over the years, because of advancements, many of these features have migrated into many other languages. That’s why many people don’t go for Lisp. There are a lot of new languages which have more effective features or which have better packages you can see.

**Prolog>>** Prolog is frequently used in knowledge base and expert systems. The features provided by Prolog include pattern matching, freebase data structuring, automatic back tracking and so on. All of these features provide a very powerful and flexible programming framework. Prolog is actually widely used in medical projects and also for designing expert AI systems.

Apart from this, we also have C++, SaaS, JavaScript, which can also be used for AI. We also have MATLAB, Julia. All of these programming languages are considered pretty good for artificial intelligence.

#### Python – The New Generation Language

###### Why Python?

* Python is an interpreted, object-oriented, widely used general-purpose, high level programming language with dynamic semantics.
* Python was created by Guido van Rossum in 1989 and is very easy to learn it.
* It was developed by Python Software Foundation and it can be easily implemented.

###### Features of Python

 **Simple and Easy to Learn:** Python is a simple and easy to learn, read and write.

 **Free and Open Source:** Python is an example of a FLOSS (Free/Libre and Open-Source Software) which means one can distribute copies of this software, read the source code and modify it, etc.

 **High-level Language:** One does not need to bother about the low-level details like memory allocation, etc. while writing Python script.

 **Portable:** Supported by many platforms like Linux, Windows, FreeBSD, Macintosh, OS/2, AS/400, Solaris, BeOS, OS/390, PlayStation, Windows CE, etc.

 **Supports different Programming Paradigm:** Python supports procedure-oriented programming as well as object-oriented programming.

 **Extensible:** Python code can invoke C and C++ libraries, can be called from and C++ programs, can integrate with Java and .NET components.

### Python Applications

Python supports **cross-platform operating systems** which makes building applications with it all the more convenient. Some of the globally known applications such as [YouTube](https://www.youtube.com/user/edurekaIN?sub_confirmation=1), BitTorrent, DropBox, etc. use Python to achieve their functionality.

###### Web Development

Python can be used to make [web-applications](https://www.edureka.co/blog/django-tutorial/) at a rapid rate. Why is that? It is because of the frameworks Python uses to create these applications. There is *common-backend logic* that goes into making these frameworks and a number of libraries that can help integrate protocols such as HTTPS, FTP, SSL etc. and even help in the processing of JSON, XML, E-Mail and so much more.

Some of the most well-known frameworks are [Django](https://www.edureka.co/blog/django-tutorial/), [Flask](https://www.youtube.com/watch?v=lj4I_CvBnt0), Pyramid. Why uses a framework? The ***security***, ***scalability***, ***convenience*** that they provide is commendable if we compare it to starting the development of a website from scratch.

###### Game Development

Python is also used in the development of [interactive games](https://www.edureka.co/blog/python-turtle-module/). There are libraries such as PySoy which is a 3D game engine supporting Python 3, [PyGame](https://www.edureka.co/blog/pygame-tutorial) which provides functionality and a library for game development. Games such as Civilization-IV, Disney’s Toon town Online, Vega Strike etc. have been built using Python.

###### Machine Learning and Artificial Intelligence

[Machine Learning](https://www.edureka.co/blog/machine-learning-career/) and [Artificial Intelligence](https://www.edureka.co/blog/pros-and-cons-of-ai/) are the talks of the town as they yield the most promising careers for the future. We make the computer learn based on past experiences through the data stored or better yet, create algorithms which makes the computer learn by itself. The programming language that mostly everyone chooses? It’s Python. Why? Support for these domains with the **libraries** that exist already such as [Pandas](https://www.edureka.co/blog/python-pandas-tutorial/), [Scikit-Learn,](https://www.edureka.co/blog/scikit-learn-machine-learning/) [NumPy](https://www.edureka.co/blog/python-numpy-tutorial/) and so many more.

Learn the algorithm, use the library and you have your solution to the problem. It is that simple. But if you want to go the hardcore way, you can design your own code which yields a better solution, which still is much easier when we compare it to other languages.

###### Data Science and Data Visualization

Data is *money* if you know how to extract relevant information which can help you take calculated risks and increase profits. You study the data you have, perform operations and extract the information required. [Libraries](https://www.edureka.co/blog/python-libraries/) such as Pandas, NumPy help you in extracting information.

You can even visualize the data libraries such as [Matplotlib](https://www.edureka.co/blog/python-matplotlib-tutorial/), [Seaborn](https://www.edureka.co/blog/python-seaborn-tutorial/), which are helpful in **plotting graphs** and much more. This is what Python offers you to become a [Data Scientist](https://www.edureka.co/blog/who-is-a-data-scientist/).

###### Desktop GUI

We use Python to program **desktop applications**. It provides the [Tkinter](https://www.edureka.co/blog/tkinter-tutorial/) library that can be used to develop user interfaces. There are some other useful toolkits such as the wxWidgets, [Kivy](https://www.edureka.co/blog/kivy-tutorial/), PYQT that can be used to create applications on several platforms.

You can start out with creating simple applications such as Calculators, To-Do apps and go ahead and create much more complicated applications.

###### Web Scraping Applications

Python is a savior when it comes to pull a large amount of data from websites which can then be helpful in various real-world processes such as price comparison, job listings, research and development and much more.

Python has BeautifulSoup which we use to pull such data. Here’s a full-fledged guide to learn [Web scraping](https://www.edureka.co/blog/web-scraping-with-python/) [with Python](https://www.edureka.co/blog/web-scraping-with-python/).

###### Business Applications

Business Applications are different than our normal applications covering domains such as e-commerce, ERP and many more. They require applications which are scalable, extensible and easily readable and Python provides us with all these features. Platforms such as Tryton is available to develop such business applications.

###### Audio and Video Applications

We use Python to develop applications that can multi-task and also output media. Video and audio applications such as TimPlayer, Cplay have been developed using Python libraries. They provide better stability and performance in comparison to other media players.

###### CAD Applications

Computer-Aided Designing is quite challenging to make as many things have to be taken care of. Objects and their representation, functions are just the tip of the iceberg when it comes to something like this. Python makes this simple too and the most well-known application for CAD is Fandango.

###### Embedded Applications

Python is based on C which means that it can be used to create [Embedded C](https://www.edureka.co/blog/embedded-c-programming/) software for embedded applications. This helps us to perform higher-level applications on smaller devices which can compute Python.

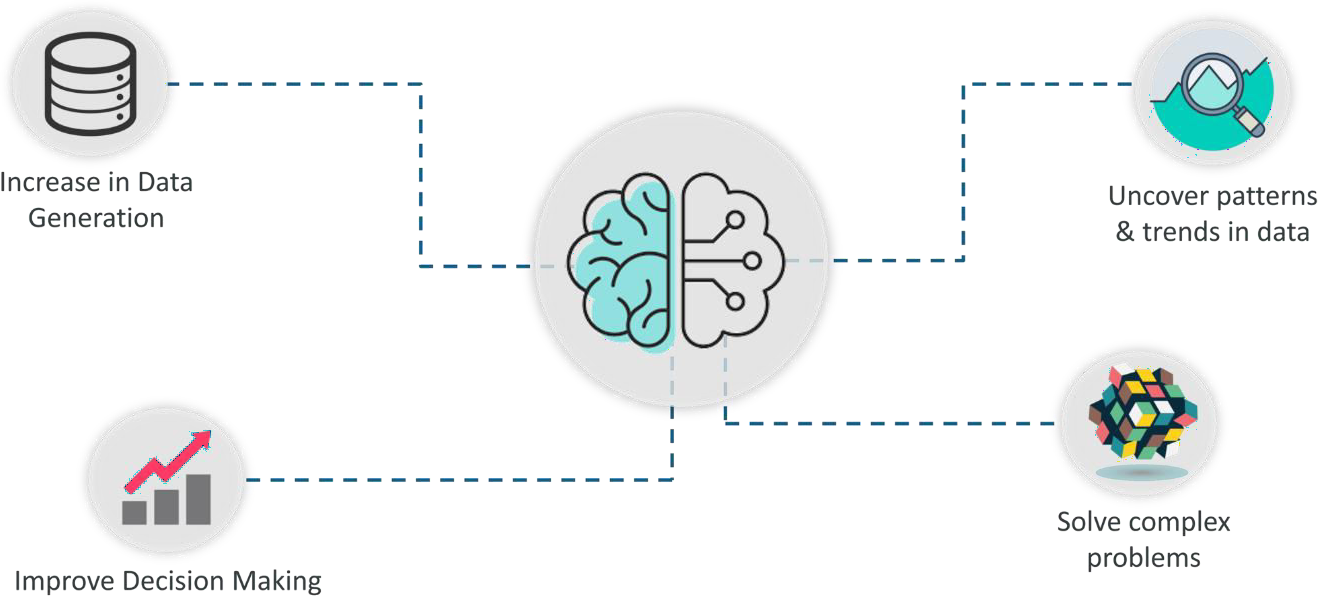
The most well-known embedded application could be the [Raspberry Pi](https://www.edureka.co/blog/raspberry-pi-tutorial/) which uses Python for its computing. We can also use it as a computer or like a simple embedded board to perform high-level computations.

We use python in a variety of applications. No matter what field you take up, Python is rewarding.

## Introduction to Machine Learning

###### Need for ML

* 1. Increase in data generation
  2. Improve decision making
  3. Uncover patterns and trends in data
  4. Solve complex problems



**Brief history of machine learning**

*Figure 3 – Need for ML*

Arthur Samuel first coined the term Machine Learning in the year 1959.

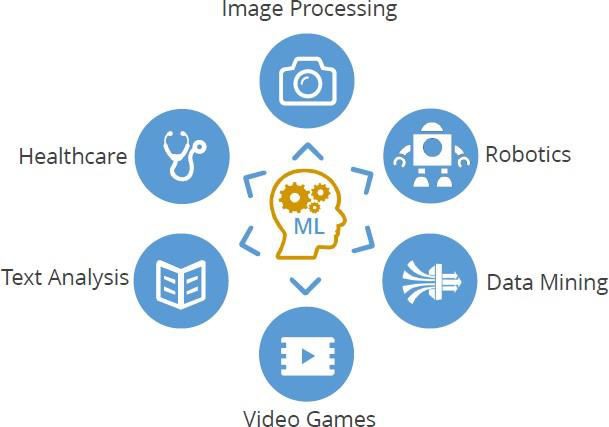
###### What is Machine Learning?

**“**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.”

In simple terms, Machine Learning is a subset of Artificial Intelligence (AI) which provides machines the ability to learn automatically & improve from experience without being explicitly programmed to do so.

In the sense, it is about getting machines to solve problems by gaining the ability to think.

**Applications of Machine Learning**



##### Figure 4 – ML Applications

*Image Processing*

* Optical Character Recognition (OCR)
* Self-driving cars
* Image tagging and recognition

*Robotics*

* Industrial robotics
* Human simulation

*Data Mining*

* Association rules
* Anomaly detection
* Grouping and Predictions

*Video games*

* Pokémon
* PUBG

*Text Analysis*

* Spam Filtering
* Information Extraction
* Sentiment Analysis

*Healthcare*

* Emergency Room & Surgery
* Research
* Medical Imaging & Diagnostics

###### Machine Learning Definitions

**Algorithm:** Set of rules and statistical techniques that are used to learn patterns of data and draw significant information from it. The logic behind the ML model is the ML algorithm. An example of a ML algorithm is linear regression, decision tree or a random forest. All of these are ML algorithms which define the logic behind ML model.

**ML Model:** A model is the main component of a ML process. A model is trained by using the ML algorithm. The difference between a ML model and an algorithm is that an algorithm maps all the decisions that a model I supposed to take based on the given input in order to get the correct output. So, the model will use the ML algorithm in order to draw useful insights from the input and give you an outcome that is very precise.

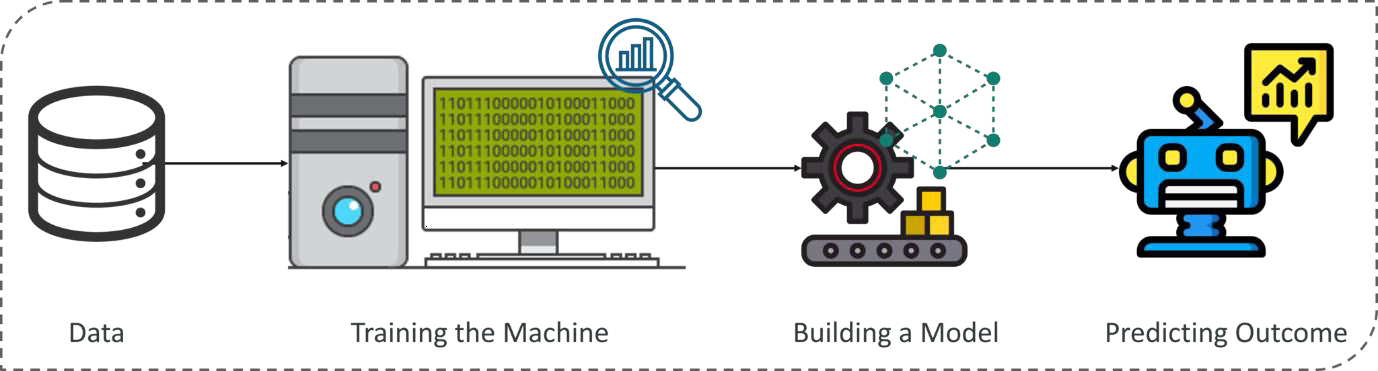
**Predictive Variable:** A predictive variable is any feature of the data, that can be used to predict the output.

E. g., if you’re trying to predict the height of a person depending on his weight. So, here, the predictive variable is weight because we’re using the weight of person to predict the person’s height.

**Response Variable:** In the same example, height would be the response variable. It is also known as target variable or output variable. This is the variable we’re trying to predict using the predictive variable. So, the response variable is the feature or the output variable that means to be predicted using the predictive variable.

**Training Data:** Training and testing data are terminologies that you come across very often in the ML process. Training data is basically the data that is used to create the ML model. Basically, in a ML process, when you feed data to the machine, it’ll be divided into two parts. So, splitting the data into two parts is also known as data slicing. So, you’ll take your input data and divide it into two sections. One you’ll call the training data and the other you’ll call the testing data.

**Testing Data:** The training data is basically used to create the ML model. The training data helps the model to identify key trends and patterns which are essential to predict the output. Now, the testing data is after the model is trained, it must be tested in order to evaluate how accurately it can predict an outcome. This is done by testing data. So, basically the training data is used to train the model. The testing data is used to test the efficiency of the model.



##### Figure 5 – Machine Learning Process

The above figure shows that a machine learning process begins by feeding machine lots of data, by using this the machine is trained to detect hidden insights and trends. These insights are then used to build a Machine Learning Model by using an algorithm in order to solve a problem.

###### Machine Learning Process

A machine learning process involves building a predictive model that can be used to find the solution for a problem statement.

Steps to solve a problem in ML process

1. Define objective
2. Data gathering
3. Preparing Data
4. Data Exploration
5. Building a model
6. Model evaluation
7. Predictions making

To understand the Machine Learning process let’s assume that you have been given a problem that needs to be solved by using Machine Learning.

*The problem is to predict the occurrence of rain in your local area by using Machine Learning.*

The below steps are followed in a Machine Learning process:

###### Step 1: Define the objective of the problem

1. What are we trying to predict?

The output is going to be a continuous variable or a desirable variable?

1. What are the target features?

Which is the target variable and what are the predictive variables in order to predict this outcome? Here, our target variable will be whether it’s going to rain or not.

1. What is the input data?

Input data is we’ll need such as temperature on a particular day or the humidity level and precipitation and so on.

1. What kind of problem are we facing? Binary classification, clustering or regression?

###### Step 2: Data Gathering

At this stage, you must be asking questions such as,

* What kind of data is needed to solve this problem?
* Is the data available?
* How can I get the data?

Once you know the types of data that is required, you must understand how you can derive this data. Data collection can be done manually or by web scraping. However, if you’re a beginner and you’re just looking to learn Machine Learning you don’t have to worry about getting the data. There are 1000s of data resources on the web, you can just download the data set and get going.

Coming back to the problem at hand, the data needed for weather forecasting includes measures such as humidity level, temperature, pressure, locality etc are either collected manually or scraped from the web.

###### Step 3: Preparing data also known as data cleaning

Data cleaning involves getting rid of inconsistencies in data such as missing values or redundant variables.

* Data Cleaning
  + Missing values
  + Corrupted data
  + Remove unnecessary data

Removing such inconsistencies is very essential because they might lead to wrongful computations and predictions. Therefore, at this stage, you scan the data set for any inconsistencies and you fix them then and there.

###### Step 4: Exploratory Data Analysis

Data exploration involves understanding the patterns and trends in the data. At this stage all the useful insights are drawn and correlations between the variables are understood.

You become a detective at this stage. It’s the brain storming stage in ML. It’s the most important part of ML because you’ll know exactly what your data is and how you can form the solution.

For example, in the case of predicting rainfall, we know that there is a strong possibility of rain if the temperature has fallen low. Such correlations must be understood and mapped at this stage.

###### Step 5: Building a ML model

At this stage a predictive model is built by using machine learning algorithms such as linear regression, decision trees etc. We always use the training data to build the model.

Training data is the same input data that you’re feeding to the machine. The only difference is that you’re splitting the data sets in two. You’re randomly picking 80% of your data and you’re assigning it for training purpose and the rest 20% you’ll probably assign for the testing purpose.

The training data is always much more than your testing data obviously because you need to train your machine and more the data you feed the machine during the training phase is, the better it’ll be during the testing phase. It’ll predict the better outcomes if it’s been trained on more data.

Machine learning model is built by using the training data set. The model is the machine learning algorithm that predicts the output by using the data fed to it. Output variable will be a categorical variable. It can take two values of Yes and No. Our outcome is a classification of a categorical variable.

In the case of predicting rainfall, since the output will be in the form of True (if it will rain tomorrow) or False (no rain tomorrow), we’ll use the classification algorithm such as logistic regression. You can also use support vector machine, K-nearest neighbor, naïve bayes and so on. Choosing the right algorithm depends on the type of problem you’re trying to solve, the data set and the level of complexity of the problem.

###### Step 6: Model evaluation and optimization

The efficiency of the model is evaluated and further improvement in the model are implemented. After you’re done building the model using the training data set, it is finally time to put the model to a test. The testing data set is used to check the efficiency of the model and how accurately it can predict the outcome.

Machine learning model is evaluated by using the testing data set. The accuracy of the model is calculated here. Further improvement in the model is done by using techniques like parameter tuning or cross validation. Here, we ‘re testing how model can predict the outcome.

###### Step 7: Predictions

The final outcome is predicted after performing parameter tuning and improving the accuracy of the model. The final outcome can either be a categorical variable (e. g., True or False) or it can be a continuous quantity (e.g., the predicted value of a stock).

In our case, for predicting the occurrence of rainfall, the output will be a categorical variable. So that was the entire Machine Learning process. Now it’s time to learn about the different ways in which Machines can learn.

###### Machine Learning Types

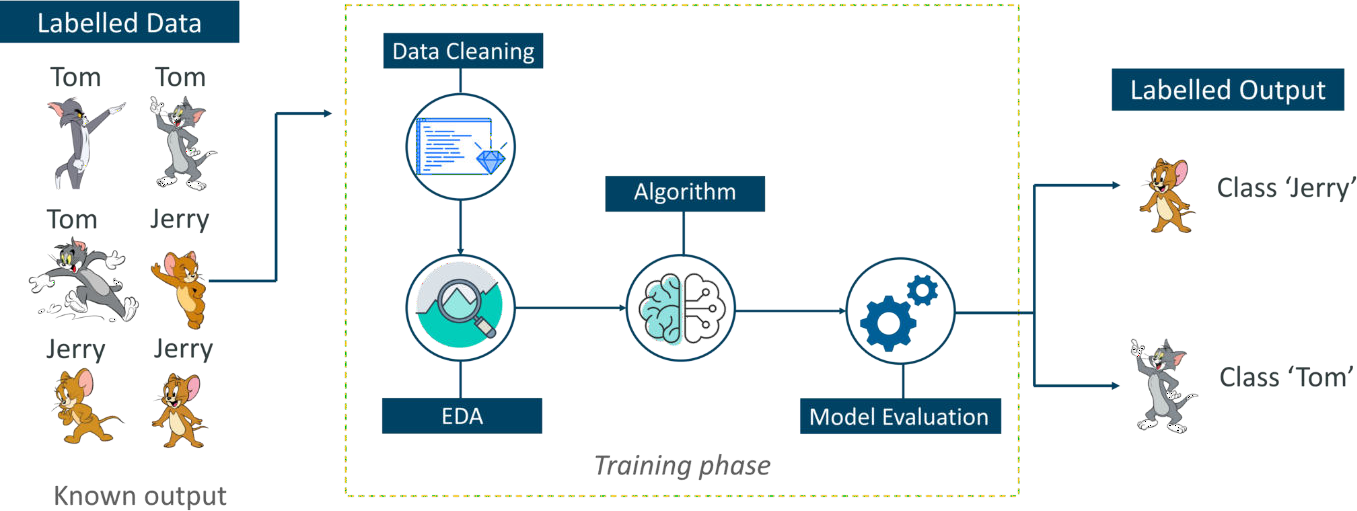
A machine can learn to solve a problem by following any one of the following three approaches. These are the ways in which a machine can learn:

1. Supervised Learning
2. Unsupervised Learning
3. Reinforcement Learning

###### Supervised Learning

*“Supervised learning is a technique in which we teach or train the machine using data which is well labelled.”*

To understand Supervised Learning let’s consider an analogy. As kids we all needed guidance to solve math problems. Our teachers helped us understand what addition is and how it is done. Similarly, you can think of supervised learning as a type of Machine Learning that involves a guide. The labeled data set is the teacher that will train you to understand patterns in the data. The labeled data set is nothing but the training data set.



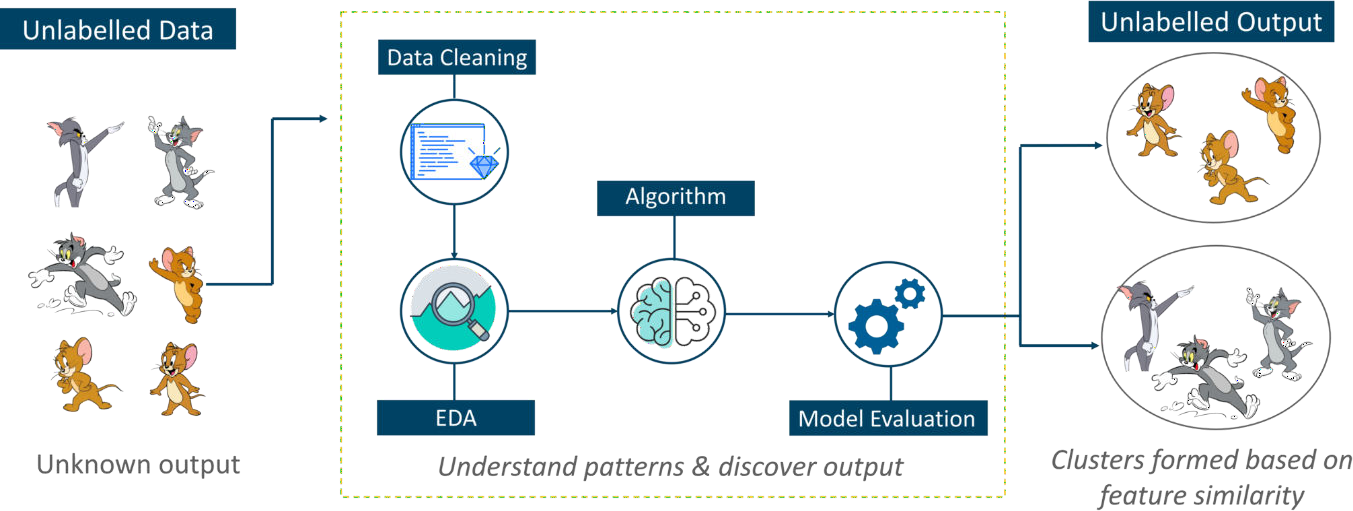
##### Figure 6 -- Supervised Learning – Introduction To Machine Learning

Consider the above figure. Here we’re feeding the machine images of Tom and Jerry and the goal is for the machine to identify and classify the images into two groups (Tom images and Jerry images). The training data set that is fed to the model is labeled, as in, we’re telling the machine, ‘This is how Tom looks and this is Jerry’. By doing so you’re training the machine by using labeled data. In Supervised Learning, there is a well-defined training phase done with the help of labeled data.

###### Unsupervised Learning

*“Unsupervised learning involves training by using unlabelled data and allowing the model to act on that information without guidance.”*

Think of unsupervised learning as a smart kid that learns without any guidance. In this type of Machine Learning, the model is not fed with labeled data, as in the model has no clue that ‘this image is Tom and this is Jerry’, it figures out patterns and the differences between Tom and Jerry on its own by taking in tons of data.



*Figure 7 -- Unsupervised Learning – Introduction To Machine Learning*

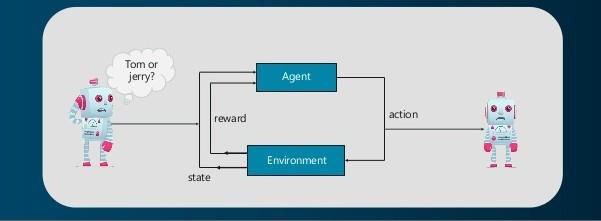
For example, it identifies prominent features of Tom such as pointy ears, bigger size, etc, to understand that this image is of type 1. Similarly, it finds such features in Jerry and knows that this image is of type 2. Therefore, it classifies the images into two different classes without knowing who Tom is or Jerry is.

###### Reinforcement Learning

*“Reinforcement Learning is a part of Machine learning where an agent is put in an environment and he learns to behave in this environment by performing certain actions and observing the rewards which it gets from those actions.”*

This type of Machine Learning is comparatively different. Imagine that you were dropped off at an isolated island! What would you do?

Panic? Yes, of course, initially we all would. But as time passes by, you will learn how to live on the island. You will explore the environment, understand the climate condition, the type of food that grows there, the dangers of the island, etc. This is exactly how Reinforcement Learning works, it involves an Agent (you, stuck on the island) that is put in an unknown environment (island), where he must learn by observing and performing actions that result in rewards.

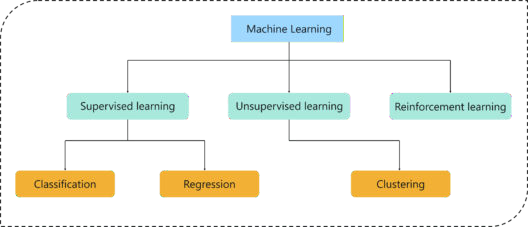


##### Figure 8 -- Reinforcement Learning – Introduction To Machine Learning

Reinforcement Learning is mainly used in advanced Machine Learning areas such as self-driving cars, AplhaGo, etc.

So that sums up the types of Machine Learning. Now, let’s look at the type of problems that are solved by using Machine Learning.

**Type Of Problems In Machine Learning**

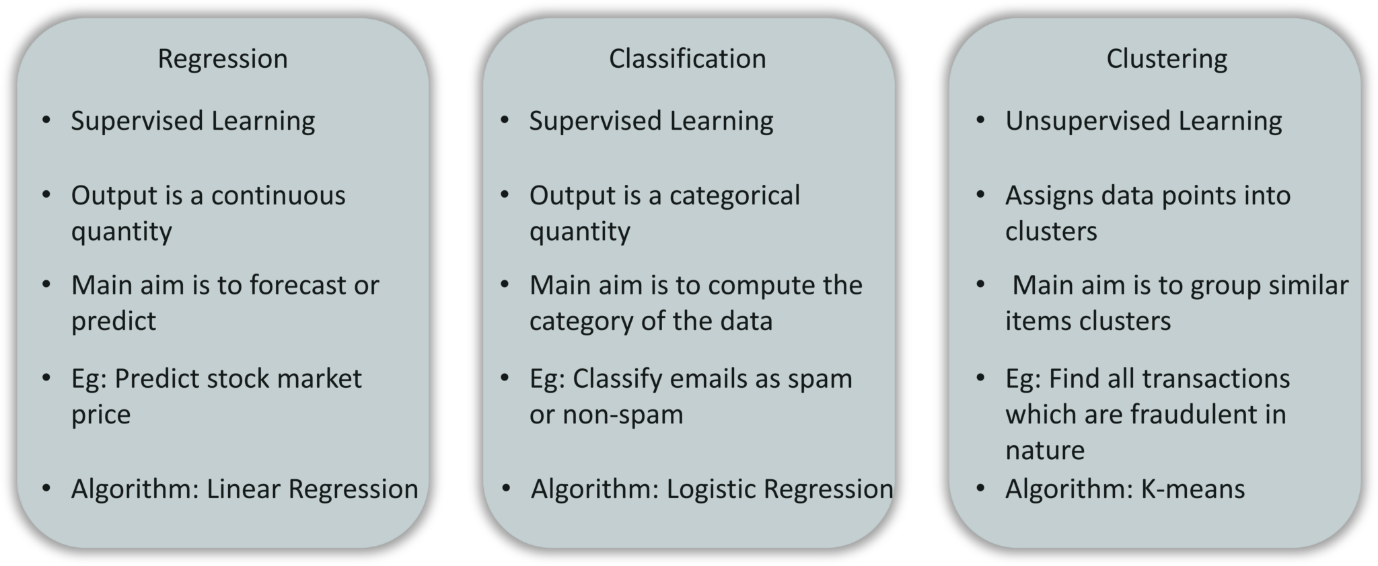


##### Figure 9 -- Type of Problems Solved Using Machine Learning – Intro To Machine Learning

Consider the above figure, there are three main types of problems that can be solved in Machine Learning:

1. **Regression:** In this type of problem the output is a continuous quantity. So, for example, if you want to predict the speed of a car given the distance, it is a Regression problem. Regression problems can be solved by using Supervised Learning algorithms like Linear Regression.
2. **Classification:** In this type, the output is a categorical value. Classifying emails into two classes, spam and non-spam is a classification problem that can be solved by using Supervised Learning classification algorithms such as Support Vector Machines, Naive Bayes, Logistic Regression, K Nearest Neighbour, etc.
3. **Clustering:** This type of problem involves assigning the input into two or more clusters based on feature similarity. For example, clustering viewers into similar groups based on their interests, age, geography, etc can be done by using Unsupervised Learning algorithms like K-Means Clustering.

Here’s a table that sums up the difference between Regression, Classification, and Clustering.



*Figure 10 -- Regression vs Classification vs Clustering – Introduction To Machine Learning*

**Machine Learning Algorithms: What is a Machine Learning Algorithm?**

Machine Learning algorithm is an evolution of the regular algorithm. It makes your programs “smarter”, by allowing them to automatically learn from the data you provide. The algorithm is mainly divided into:

* Training Phase
* Testing phase

So, building upon the example I had given a while ago, let’s talk a little about these phases.

###### Training Phase

You take a randomly selected specimen of apples from the market (**training data**), make a table of all the physical characteristics of each apple, like color, size, shape, grown in which part of the country, sold by which vendor, etc (**features**), along with the sweetness, juiciness, ripeness of that apple (**output variables**). You feed this data to the machine learning algorithm (**classification/regression**), and it learns a model of the correlation between an average apple’s physical characteristics, and its quality.

###### Testing Phase

Next time when you go shopping, you will measure the characteristics of the apples which you are purchasing (**test data**) and feed it to the Machine Learning algorithm. It will use the model which was computed earlier to predict if the apples are sweet, ripe and/or juicy. The algorithm may internally use the rules, similar to the one you manually wrote earlier (e.g., a **decision tree**). Finally, you can now shop for apples with great confidence, without worrying about the details of how to choose the best apples.

###### Machine Learning Algorithms: List of Machine Learning Algorithms

Here is the list of 8 most commonly used machine learning algorithms.

1. Linear Regression
2. Logistic Regression
3. SVM (Support Vector Machine)
4. Decision Tree
5. Random Forest
6. Naive Bayes
7. kNN (k Nearest Neighbour)
8. k-Means Clustering

###### Linear Regression

It is used to estimate real values (cost of houses, number of calls, total sales etc.) based on continuous variables. Here, we establish a relationship between the independent and dependent variables by fitting the best line. This best fit line is known as the *regression line* and represented by a linear equation **Y= aX + b**.

The best way to understand [***linear regression***](https://www.edureka.co/blog/understanding-linear-regression-in-r/) is to relive this experience of childhood. Let us say, you ask a child in fifth grade to arrange people in his class by increasing order of weight, without asking them their weights! What do you think the child will do? He/she would likely look (visually analyze) at the height and build of people and arrange them using a combination of these visible parameters. This is a linear regression in real life! The child has actually figured out that height and build would be correlated to the weight by a relationship, which looks like the equation above.

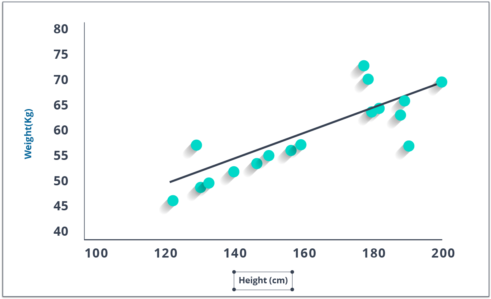
In this equation **Y= aX + b**:

###### Y – Dependent Variable

* + **a – Slope**

###### X – Independent variable

* + **b – Intercept**

*Figure 11 – Linear Regression* These coefficients **a** and **b** are derived based on minimizing the ‘sum of squared differences’ of distance between data points and regression line.

Look at the plot given. Here, we have identified the best fit having linear equation **y=0.2811x+13.9**. Now

using this equation, we can find the weight, knowing the height of a person.

###### Logistic Regression

Don’t get confused by its name! It is a classification, and not a regression algorithm. It is used to estimate discrete values (Binary values like 0/1, yes/no, true/false) based on a given set of independent variable(s). In simple words, it predicts the probability of occurrence of an event by fitting data to a *logit function*. Hence, it is also known as logit regression. Since it predicts the probability, its output values lie between 0 and 1.

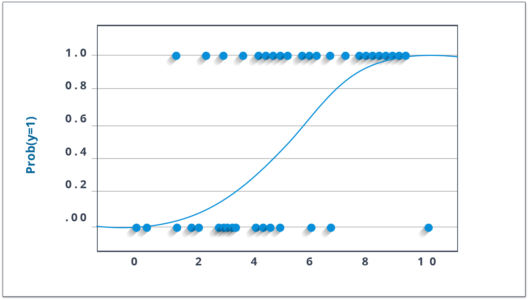
Again, let us try and understand this through a simple example.

Let’s say your friend gives you a puzzle to solve. There are only 2 outcome scenarios – either you solve it or you don’t. Now imagine, that you are being given a wide range of puzzles/quizzes in an attempt to understand which subjects you are good at. The outcome of this study would be something like this – if you are given a trigonometry based tenth-grade problem, you are 70% likely to solve it. On the other hand, if it is grade fifth history question, the probability of getting an answer is only 30%. This is what Logistic Regression provides you.

Coming to the math, the log odds of the outcome is modeled as a linear combination of the predictor variables.

odds= p/ (1-p) = probability of event occurrence / probability of not event occurrence ln(odds) = ln(p/(1-p)) logit(p) = ln(p/(1-p)) = b0+b1X1+b2X2+b3X3 +bkXk

Above, *p* is the probability of the presence of the characteristic of interest. It chooses parameters that maximize the likelihood of observing the sample values rather than that minimize the sum of squared errors (like in ordinary regression).



*Figure 12 – Logistic Regression*

Now, you may ask, why take a log? For the sake of simplicity, let’s just say that this is one of the best mathematical ways to replicate a step function. I can go in more details, but that will beat the purpose of this blog.

###### SVM

SVM (Support Vector Machine) is a supervised machine learning algorithm which is mainly used to classify data into different classes. Unlike most algorithms, SVM makes use of a hyperplane which acts like a decision boundary between the various classes.

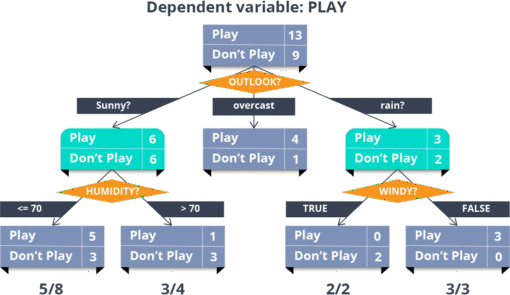
SVM can be used to generate multiple separating hyperplanes such that the data is divided into segments and each segment contains only one kind of data.

Before moving further, let’s discuss the features of SVM:

1. SVM is a supervised learning algorithm. This means that SVM trains on a set of labeled data. SVM studies the labeled training data and then classifies any new input data depending on what it learned in the training phase.
2. A main advantage of SVM is that it can be used for both classification and regression problems. Though SVM is mainly known for classification, the SVR (Support Vector Regressor) is used for regression problems.
3. SVM can be used for classifying non-linear data by using the kernel trick. The kernel trick means transforming data into another dimension that has a clear dividing margin between classes of data. After which you can easily draw a hyperplane between the various classes of data.

###### Decision Tree

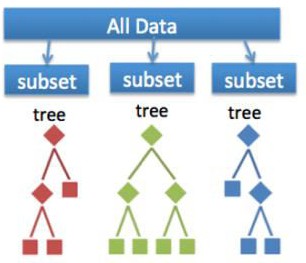
Now, this is one of my favorite algorithms. It is a type of supervised learning algorithm that is mostly used for classification problems. Surprisingly, it works for both categorical and continuous dependent variables. In this algorithm, we split the population into two or more homogeneous sets. This is done based on the most significant attributes/ independent variables to make as distinct groups as possible.

*Figure 13 – Decision Tree*

In the image above, you can see that population is classified into four different groups based on multiple attributes to identify ‘if they will play or not’.

###### Random Forest

Random forest algorithm is a supervised classification and regression algorithm. As the name suggests, this algorithm randomly creates a forest with several trees.

Generally, the more trees in the forest the more robust the forest looks like. Similarly, in the random forest classifier, the higher the number of trees in the forest, greater is the accuracy of the results.

##### Figure 14 – Random Forest

In simple words, random forest builds multiple decision trees (called the forest) and glues them together to get a more accurate and stable prediction. The forest it builds is a collection of Decision Trees, trained with the bagging method.

###### Why Use Random Forest?

You might be wondering why we use Random Forest when we can solve the same problems using Decision trees. Let me explain.

* Even though Decision trees are convenient and easily implemented, they lack accuracy. Decision trees work very effectively with the training data that was used to build them, but they’re not flexible when it comes to classifying the new sample. Which means that the accuracy during testing phase is very low.
* This happens due to a process called Over-fitting.

Over-fitting occurs when a model studies the training data to such an extent that it negatively influences the performance of the model on new data.

* This means that the disturbance in the training data is recorded and learned as concepts by the model. But the problem here is that these concepts do not apply to the testing data and negatively impact the model’s ability to classify the new data, hence reducing the accuracy on the testing data.

This is where Random Forest comes in. It is based on the idea of bagging, which is used to reduce the variation in the predictions by combining the result of multiple Decision trees on different samples of the data set.

Bootstrapping is an estimation method used to make predictions on a data set by re-sampling it. To create a bootstrapped data set, we must randomly select samples from the original data set. A point to note here is that we can select the same sample more than once.

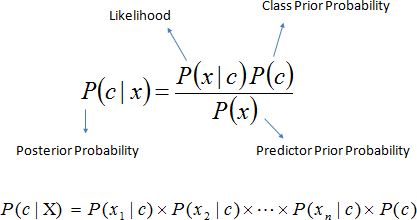
###### Naive Bayes

This is a classification technique based on *Bayes’ theorem* with an assumption of independence between predictors. In simple terms, a ***Naive Bayes classifier*** assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

For example, a fruit may be considered to be an apple if it is red, round, and about 3 inches in diameter. Even if these features depend on each other or upon the existence of the other features, a naive Bayes classifier would consider all of these properties to independently contribute to the probability that this fruit is an apple.

Naive Bayesian model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

Bayes theorem provides a way of calculating posterior probability **P(c|x)** from **P(c)**, **P(x)** and **P(x|c)**. Look at the equation below:



##### Figure 15 – Naïve Bayes

Here,

* ***P*(*c|x*) is the posterior probability of *class* (*target*) given *predictor* (*attribute*).**

###### *P*(*c*) is the prior probability of *class*.

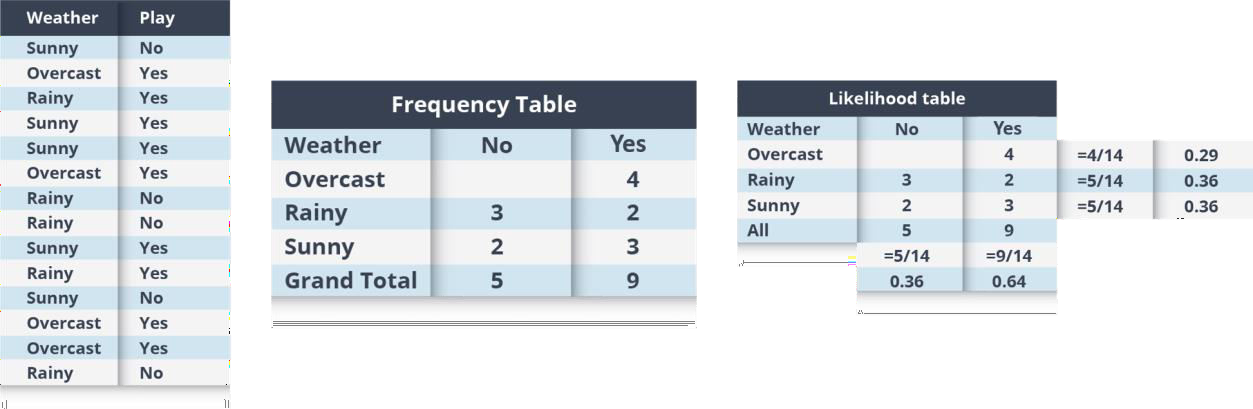
* ***P*(*x|c*) is the likelihood which is the probability of *predictor* given *class*.**
* ***P*(*x*) is the prior probability of *predictor*.**

**Example:** Let’s understand it using an example. Below I have a training data set of weather and corresponding target variable ‘Play’. 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 to the frequency table

**Step 2:** Create a Likelihood table by finding the probabilities like **Overcast probability =**

**0.29** and **probability of playing is 0.64**.



##### Figure 16 – Naïve Bayes Example

**Step 3:** Now, use the 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 pay if the weather is sunny, is this statement is correct?

We can solve it using above discussed method, so **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.

###### kNN (k- Nearest Neighbors)

It can be used for both classification and regression problems. However, it is more widely used in classification problems in the industry. ***K nearest neighbors*** is a simple algorithm that stores all available cases and classifies new cases by a majority vote of its k neighbors. The case being assigned to the class is most common amongst its K nearest neighbors measured by a distance function.

These distance functions can be Euclidean, Manhattan, Minkowski and Hamming distance. First three functions are used for continuous function and the fourth one (Hamming) for categorical variables. If **K = 1**, then the case is simply assigned to the class of its nearest neighbor. At times, choosing K turns out to be a challenge while performing kNN modeling.

Things to consider before selecting KNN:

* KNN is computationally expensive
* Variables should be normalized else higher range variables can bias it
* Works on pre-processing stage more before going for kNN like an outlier, noise removal

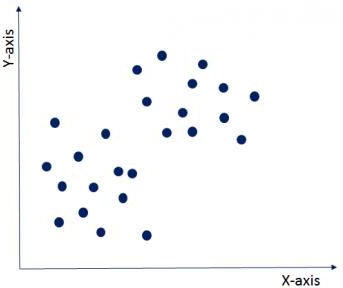
##### Figure 17 - KNN

KNN can easily be mapped to our real lives. If you want to learn about a person, of whom you have no information, you might like to find out about his close friends and the circles he moves in and gain access to his/her information!

###### k-means Clustering

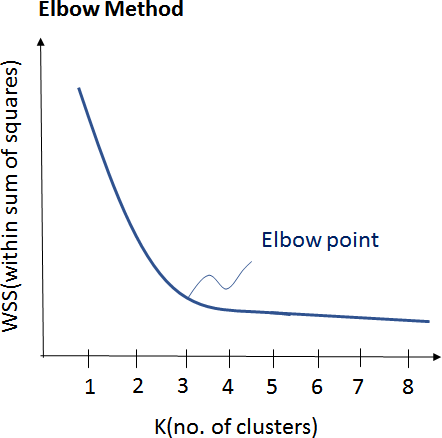
k-means clustering is one of the simplest algorithms which uses unsupervised learning method to solve known clustering issues. k-means clustering require following two inputs.

1. k = number of clusters
2. Training set(m) = {x1, x2, x3, , xm}

Let’s say you have an unlabeled data set like the one shown below and you want to group this data into clusters.

##### Figure 18 – K-Means Clustering

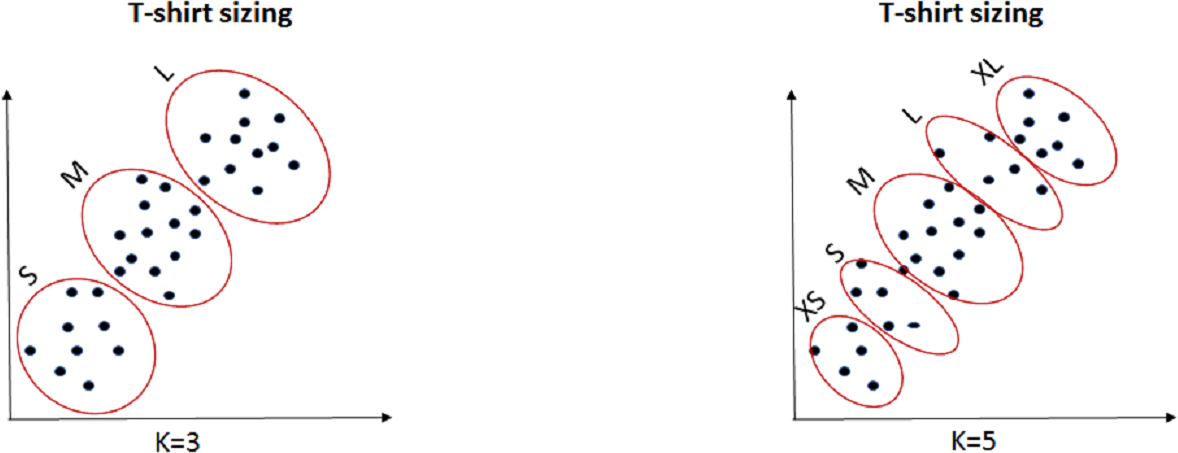
Now, the important question is how should you choose the optimum number of clusters? There are two possible ways for choosing the number of clusters.

1. **Elbow Method:** Here, you draw a curve between WSS (within sum of squares) and the number of clusters. It is called elbow method because the curve looks like a human arm and the elbow point gives us the optimum number of clusters. As you can see that after the elbow point, there is a very slow change in the value of WSS, so you should take the elbow point value as the final number of clusters.

*Fig. 19 – Elbow Method (K-Means)*

1. **Purpose Based:** You can run k-means clustering algorithm to get different clusters based on a variety of purposes. You can partition the data on different metrics and see how well it performs for that particular case. Let’s take an example of marketing T-shirts of different sizes. You can partition the dataset into different number of clusters depending upon the purpose that you want to meet. In the following example, I have taken two different criteria, price and comfort.

Let’s see these two possibilities as shown in the image below. *Fig. 20 – Purpose Based K-Means Clustering*



* 1. 1. K=3: If you want to provide only 3 sizes (S, M, L) so that prices are cheaper, you will divide the data set into 3 clusters.
  2. K=5: Now, if you want to provide more comfort and variety to your customers with more sizes (XS, S, M, L, XL), then you will divide the data set into 5 clusters.

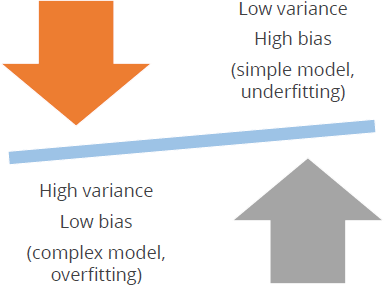
###### Some important considerations in Machine Learning

Bias & Variance Tradeoff

* + - Bias refers to error in the machine learning model due to wrong assumptions. A high-bias model will underfit the training data.
    - Variance refers to problems caused due to overfitting. This is a result of over- sensitivity of the model to small variations in the training data. A model with many degrees of freedom (such as a high-degree polynomial model) is likely to have high variance and thus overfit the training data.

Bias & Variance Dependencies

* + - Increasing a model’s complexity will reduce its bias and increase its variance.



##### Figure 21 - Bias & Variance

* + - Conversely, reducing a model’s complexity will increase its bias and reduce its variance. This is why it is called a trade-off.

###### What is Cross-Validation?

For any model in Machine Learning, *it is considered as a best practice if the model is tested with an independent data set.* Normally, any prediction model works on a known data set which is also known as the training set.

But in a real-life scenario, the model will be tested for its efficiency and accuracy with an altogether different and unique data set. Under those circumstances, you’d want your model to be efficient enough or at least to be at par with the same efficiency that it shows for the training set. Basically, this testing is known as cross- validation in Machine Learning so that it is fit to work with any model in the future.

We can also call it a technique for **asserting how the statistical model generalizes to an independent data set**. Now that we know what cross-validation stands for, let us try to understand cross-validation in simple terms.

The basic purpose of cross-validation is to assess how the model will perform with an unknown data set. For instance, you are trying to score a goal in an empty goal. It looks pretty easy, and you could even score from a considerable distance too. But the real test starts when there is a goalkeeper and a bunch of defenders. That’s why you need to get trained in a real match facing all the heat and still score the goal.

Similarly, a statistical model is trained in such a way that it excels in its efficiency with other unknown data sets using cross-validation.

Types Of Cross-Validation

There are two types of cross-validation techniques in Machine Learning.

1. **Exhaustive Cross-Validation** – This method basically involves testing the model in all possible ways, it is done by dividing the original data set into training and validation sets. Example: Leave-p- out Cross-Validation, Leave-one-out Cross-validation.
2. **Non-Exhaustive Cross-Validation** – In this method, the original data set is not separated into all the possible permutations and combinations. Example: K-fold Cross-Validation, Holdout Method.

**Confusion Matrix:** In [Supervised Machine Learning](https://www.edureka.co/blog/introduction-to-supervised-learning/), we usually have 2 different types of use cases, [Regression,](https://www.edureka.co/blog/linear-regression-in-python/) and [Classification](https://www.edureka.co/blog/classification-algorithms/) problem. Confusion Matrix helps in calculating the accuracy of the classification model which indirectly helps us to describe the performance of the classification model. It is the most important step when it comes to evaluating a model.

###### What is a Confusion Matrix?

A Confusion matrix is the comparison summary of the predicted results and the actual results in any classification problem use case. The comparison summary is extremely necessary to determine the performance of the model after it is trained with some training data.

For a binary classification use case, a Confusion Matrix is a 2×2 matrix which is as shown below

|  |  |  |
| --- | --- | --- |
|  | **Predicted Class 1 Value EG: 1** | **Predicted Class 2 Value EG:0** |
| **Actual Class 1 Value EG:1** | **TP (True Positive)** | **FN (False Negative)** |
| **Actual Class 2 Value EG:0** | **FP (False Positive)** | **TN (True Negative)** |

From the above figure, we have:

##### Figure 22 – A 2x2 Confusion Matrix

* Actual Class 1 value= 1 which is similar to Positive value in a binary outcome.
* Actual Class 2 value = 0 which is similar to a negative value in binary outcome.

The left side index of the confusion matrix basically indicates the Actual Values and the top column indicates the Predicted Values.

There are various components that exist when we create a confusion matrix. The components are mentioned below

**Positive(P):** The predicted result is Positive (Example: Image is a cat)

**Negative(N):** the predicted result is Negative (Example: Images is not a cat)

**True Positive (TP):** Here TP basically indicates the predicted and the actual values is 1(True)

**True Negative (TN):** Here TN indicates the predicted and the actual value is 0(False)

**False Negative (FN):** Here FN indicates the predicted value is 0(Negative) and Actual value is 1. Here both values do not match. Hence it is False Negative.

**False Positive (FP):** Here FP indicates the predicted value is 1(Positive) and the actual value is 0. Here again both values mismatches. Hence it is False Positive.

### Accuracy and Components of Confusion Matrix

After the confusion matrix is created and we determine all the components values, it becomes quite easy for us to calculate the accuracy. So, let us have a look at the components to understand this better.

* **Classification Accuracy**



*Figure 23 - Accuracy*

From the above formula, the sum of TP (True Positive) and the TN (True Negative) are the correct predicted results. Hence in order to calculate the accuracy in percentage, we divide with all the other components. However, there are some problems in the accuracy and we cannot completely depend on it.

**Precision, Recall, and F-Measure**

###### Recall:

A recall is similar to the True Positive Rate and it is the ratio of the Total number of correctly predicted positive values (TP) to all the Positive Values.



*Figure 24 - Recall*

###### Precision:

The Precision basically indicates all the points the model predicted to be positive and what percentage of them are actually Positive.



##### Figure 25 - Precision

Precision and Recall are metrics results which focus on the positive class as shown from the above formulas.

### F-Measure

So, F-Measure is a technique which combines both the Precision and Recall technique and it uses Harmonic Mean in the place of the usual Arithmetic Mean, due to which the extreme values are punished. The F- measure is also called as F1- score and is given by the below formula.



##### Figure 26 – (F – Measure)

Let us consider an example and see how we can compute the Accuracy, Precision, Recall and the F1-score.

o

o

|  |  |  |
| --- | --- | --- |
| **N = 165** | **Predicted YES** | **Predicted NO** |
| **Act**o**ual YES**  o | **TP = 150** | **FN = 10** |
| **Act**o**ual NO** | **FP = 20** | **TN = 100** |

##### Figure 27 – A 2x2 Confusion Matrix Example for F-Measure

o **Accuracy = (TP + TN) / (TP + TN + FP + FN)** = (150 + 100) / (150+100+20+10) = **0.89**

o **Recall= TP/ (TP+FN)** = 150/ (150+10) = **0.93**

o **Precision: TP/(TP+FP)** = 150/ (150+20) = **0.88**

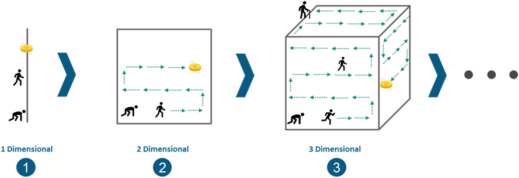
###### F-measure=(2\*Recall\*Precision)/(Recall+Presision)

= (2\*0.93\*0.88)/ (0.93+0.88) = **0.90**

### Drawbacks of the Machine Learning

##### Figure 28 -ML Drawbacks

Machine Learning is not capable of handling high dimensional data that is where input & output is quite large. Handling and processing such type of data becomes very complex and resource exhaustive. This is termed as **Curse of Dimensionality**. To understand this in simpler terms, let’s consider the following image:



##### Figure 29 - Curse of Dimensionality



Consider a line of 100 yards and you have dropped a coin somewhere on the line. Now, it’s quite convenient for you to find the coin by simply walking on the line. This very line is a single dimensional entity.



Next, consider you have a square of side 100 yards each as shown in the above image and yet again, you dropped a coin somewhere in between. Now, it is quite evident that you are going to take more time to find the coin within that square as compared to the previous scenario. This square is a 2- dimensional entity.



Let’s take it a step ahead by considering a cube of side 100 yards each and you have dropped a coin somewhere in between. Now, it is even more difficult to find the coin this time. This cube is a 3- dimensional entity.

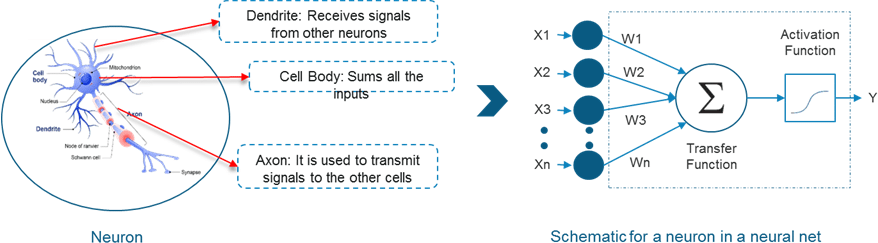
Hence, you can observe the complexity is increasing as the dimensions are increasing. And in real-life, the

high dimensional data that we were talking about has thousands of dimensions that makes it very complex to handle and process. The high dimensional data can easily be found in use-cases like Image processing, NLP, Image Translation etc.

Machine learning was not capable of solving these use-cases and hence, Deep learning came to the rescue. Deep learning is capable of handling the high dimensional data and is also efficient in focusing on the right features on its own. This process is called feature extraction. Now, let’s move ahead in this Deep Learning Tutorial and understand how deep learning works.

### How Deep Learning Works?

In an attempt to re-engineer a human brain, Deep Learning studies the basic unit of a brain called a brain cell or a neuron. Inspired from a neuron an artificial neuron or a perceptron was developed. Now, let us understand the functionality of biological neurons and how we mimic this functionality in the perceptron or an artificial neuron:



##### Figure 30 - Deep Learning Working

* If we focus on the structure of a biological neuron, it has dendrites which is used to receive inputs. These inputs are summed in the cell body and using the Axon it is passed on to the next biological neuron as shown in the above image.
* Similarly, a perceptron receives multiple inputs, applies various transformations and functions and provides an output.
* As we know that our brain consists of multiple connected neurons called neural network, we can also have a network of artificial neurons called perceptrons to form a Deep neural network. So, let’s move ahead in this Deep Learning Tutorial to understand how a Deep neural network looks like.

### Deep Learning Tutorial: What is Deep Learning?

#### Simple Definition of a Neural Network

Modeled in accordance with the human brain, a Neural Network was built to mimic the functionality of a human brain. The human brain is a neural network made up of multiple neurons, similarly, an Artificial Neural Network (ANN) is made up of multiple perceptrons.

##### Figure 31 – A Simple Neural Network

* Any Deep neural network will consist of three types of layers:
  + The Input Layer
  + The Hidden Layer
  + The Output Layer



In the above diagram, the first layer is the input layer which receives all the inputs and the last layer is the output layer which provides the desired output.

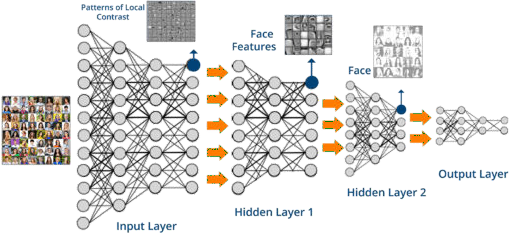
All the layers in between these layers are called hidden layers. There can be n number of hidden layers thanks to the high end resources available these days.

The number of hidden layers and the number of perceptrons in each layer will entirely depend on the use-case you are trying to solve.

Now that you have a picture of a Deep Neural Networks, let’s move ahead to get a high-level view of how Deep Neural Networks solves a problem of Image Recognition.

### Deep Learning Use – Case

We want to perform Image recognition using Deep Networks:



##### Figure 32 – Deep Learning Use-Case



Here, we are passing the high dimensional data to the input layer. To match the dimensionality of the input data, the input layer will contain multiple sub-layers of perceptrons so that it can consume the entire input.

The output received from the input layer will contain patterns and will only be able to identify the edges of the images based on the contrast levels.

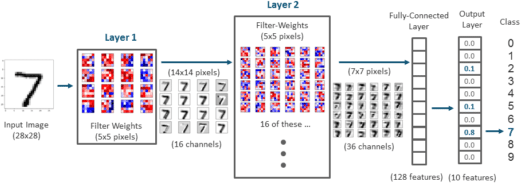
This output will be fed to the Hidden layer 1 where it will be able to identify various face features like eyes, nose, ears etc.

Now, this will be fed to the hidden layer 2 where it will able to form the entire faces. Then, the output of layer 2 is sent to the output layer.



Finally, the output layer performs classification based on the result obtained from the previous and predicts the name.

*Let me ask you a question, what will happen if any of these layers is missing or the neural network is not deep enough?* Simple, we will not be able to accurately identify the images. This is the very reason why these use-cases did not have a solution all these years prior to Deep Learning. Just to take this further, we will try to apply Deep networks on a MNIST Data-set.



##### Figure 33 – Deep Network Implementation On MNIST Data-Set

* The Mnist data-set consists of 60,000 training samples and 10,000 testing samples of handwritten digit images. The task here is to train a model which can accurately identify the digit present on the image.
* To solve this use-case a Deep network will be created with multiple hidden layers to process all the 60,000 images pixel by pixel and finally we will receive an output layer.
* The output layer will be an array of index 0 to 9, where each index corresponds to the respective digit. Index 0 contains the probability for 0 being the digit present on the input image.
* Similarly, index 2 which has a value of 0.1, actually represents the probability of 2 being the digit present on the input image. So, if we see the highest probability in this array is 0.8 which is present at index 7 of the array. Hence the number present on the image is 7.

###### Need For Deep Learning: Limitations of Traditional Machine Learning Algorithms and Techniques

Machine Learning was a major breakthrough in the technical world, it led to the automation of monotonous and time-consuming tasks, it helped in solving complex problems and making smarter decisions. However, there were a few drawbacks in Machine learning that led to the emergence of Deep Learning.

##### Figure 34 – ML Limitations

Here are some limitations of Machine Learning:

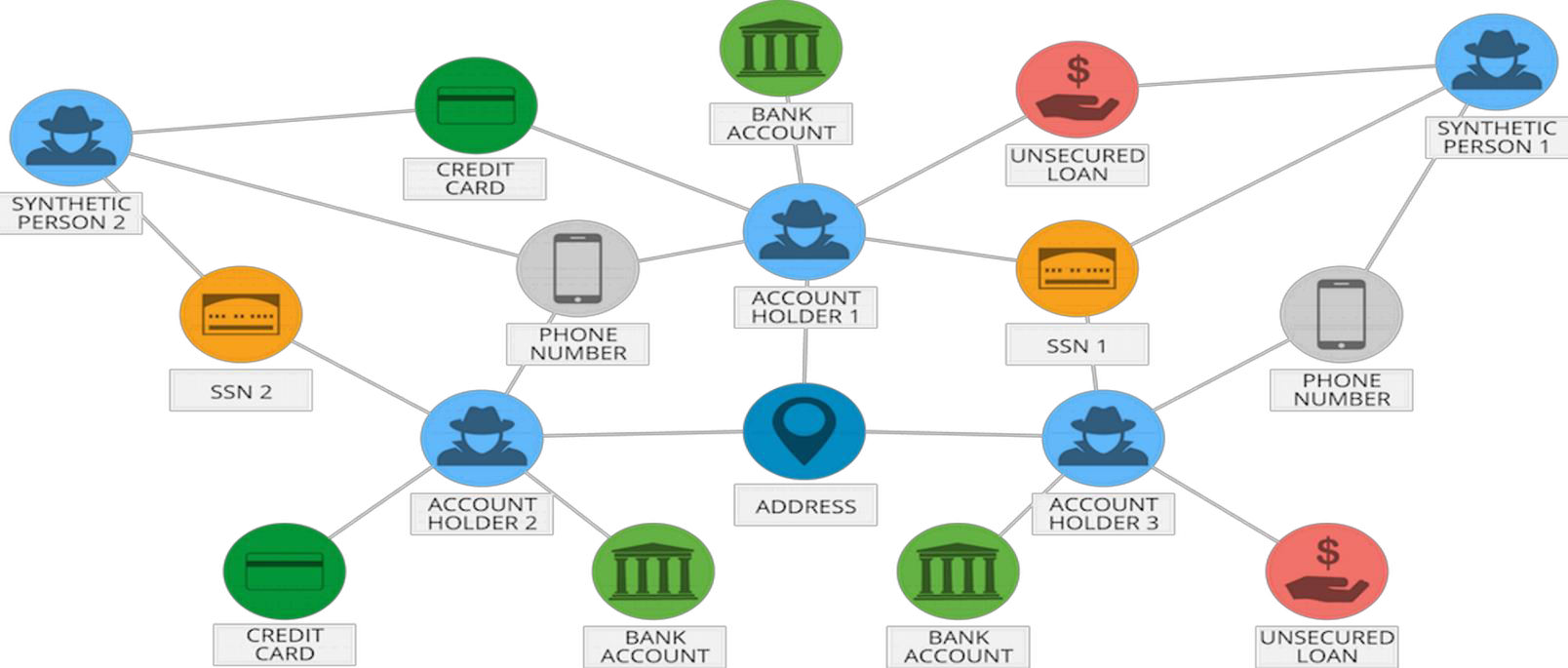
1. **Unable to process high dimensional data:** Machine Learning can process only small dimensions of data that contain a small set of variables. If you want to analyse data containing 100s of variables, then Machine Learning cannot be used.
2. **Feature engineering is manual:** Consider a use case where you have 100 predictor variables and you need to narrow down only the significant ones. To do this you have to manually study the relationship between each of the variables and figure out which ones are important in predicting the output. This task is extremely tedious and time-consuming for a developer.
3. **Not ideal for performing object detection and image processing:** Since object detection requires high-dimensional data, Machine Learning cannot be used to process image data sets, it is only ideal for data sets with a restricted number of features.

Before we get into the depths of Neural Networks, let’s consider a real-world use case where Deep Learning is implemented.

**Deep Learning Use Case/ Applications**

Did you know that PayPal processes over $235 billion in payments from four billion transactions by its more than 170 million customers? It uses this vast amount of data to identify possible fraudulent activities among other reasons.

With the help of Deep Learning algorithms, PayPal mined data from their customer’s purchasing history in addition to reviewing patterns of likely fraud stored in its databases to predict whether a particular transaction is fraudulent or not.



##### Figure 35 – Deep Learning Use-Case for Paypal

The company has been relying on Deep Learning & Machine Learning technology for around 10 years. Initially, the fraud monitoring team used simple, linear models. But over the years the company switched to a more advanced Machine Learning technology called, Deep Learning.

Fraud risk manager and Data Scientist at PayPal, Ke Wang, quoted:

*“What we enjoy from more modern, advanced machine learning is its ability to consume a lot more data, handle layers and layers of abstraction and be able to ‘see’ things that a simpler technology would not be able to see, even human beings might not be able to see.”*

A simple linear model is capable of consuming around 20 variables. However, with Deep Learning technology one can run thousands of data points. Therefore, by implementing Deep Learning technology, PayPal can finally analyze millions of transactions to identify any fraudulent activity.

Now let’s go into the depths of a Neural Network and understand how they work.

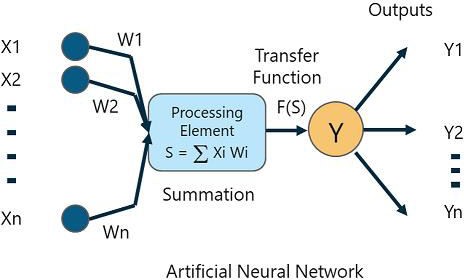
### How Does a Neural Network Work?

To understand neural networks, we need to break it down and understand the most basic unit of a Neural Network, i.e., a Perceptron.

###### What is a Perceptron?

A Perceptron is a single layer neural network that is used to classify linear data. It has 4 important components:

1. Inputs
2. Weights and Bias
3. Summation Function
4. Activation or transformation Function

The basic logic behind a Perceptron is as follows:

*Figure 36 - ANN*

*The inputs (x) received from the input layer are multiplied with their assigned weights w. The multiplied values are then added to form the Weighted Sum. The weighted sum of the inputs and their respective weights are then applied to a relevant Activation Function. The activation function maps the input to the respective output.*

###### Weights and Bias in Deep Learning

Why do we have to assign weights to each input?

Once an input variable is fed to the network, a randomly chosen value is assigned as the weight of that input. The weight of each input data point indicates how important that input is in predicting the outcome.

The bias parameter, on the other hand, allows you to adjust the activation function curve in such a way that a precise output is achieved.

###### Summation Function

Once the inputs are assigned some weight, the product of the respective input and weight is taken. Adding all these products gives us the Weighted Sum. This is done by the summation function.

###### Activation Function

The main aim of the activation functions is to map the weighted sum to the output. Activation functions such as tanh, ReLU, sigmoid and so on are examples of transformation functions.

### Neural Networks Explained with An Example

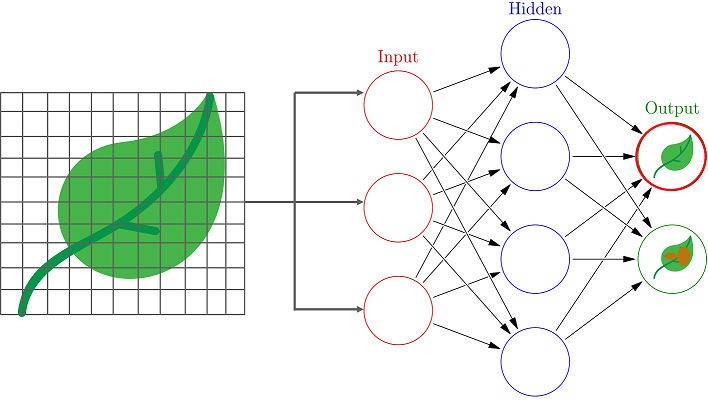
Consider a scenario where you are to build an Artificial Neural Network (ANN) that classifies images into two classes:

* Class A: Containing images of non-diseased leaves
* Class B: Containing images of diseased leaves

So how do you create a Neural network that classifies the leaves into diseased and non-diseased crops?

The process always begins with processing and transforming the input in such a way that it can be easily processed. In our case, each leaf image will be broken down into pixels depending on the dimension of the image.

For example, if the image is composed of 30 by 30 pixels, then the total number of pixels will be 900. These pixels are represented as matrices, which are then fed into the input layer of the Neural Network.



##### Figure 37 – Neural Network Example

Just like how our brains have neurons that help in building and connecting thoughts, an ANN has perceptrons that accept inputs and process them by passing them on from the input layer to the hidden and finally the output layer.

As the input is passed from the input layer to the hidden layer, an initial random weight is assigned to each input. The inputs are then multiplied with their corresponding weights and their sum is sent as input to the next hidden layer.

Here, a numerical value called bias is assigned to each perceptron, which is associated with the weightage of each input. Further, each perceptron is passed through activation or a transformation function that determines whether a particular perceptron gets activated or not.

An activated perceptron is used to transmit data to the next layer. In this manner, the data is propagated (Forward propagation) through the neural network until the perceptrons reach the output layer.

At the output layer, a probability is derived which decides whether the data belongs to class A or class B. Sounds simple, doesn’t it? Well, the concept behind Neural Networks is purely based on the functioning of the human brain. You require in-depth knowledge of various mathematical concepts and algorithms. Here’s a list of blogs to get you started:

### Natural Language Processing (NLP)

Natural language processing (NLP) is a subfield of artificial intelligence and linguistics. It studies the problems of automated generation and understanding of natural human languages. Natural language generation systems convert information from computer databases into normal-sounding human language, and natural language understanding systems convert samples of human language into more formal representations that are easier for computer programs to manipulate.

Natural Language Processing is the artificial intelligent concept where the machines understand Natural Languages like English, Korean, French, Telugu, Hindi etc.

## Applications of Deep Learning

Moving ahead in this what is deep learning blog, let us look at some of the real-life applications of Deep Learning to understand its true powers.

###### Speech Recognition

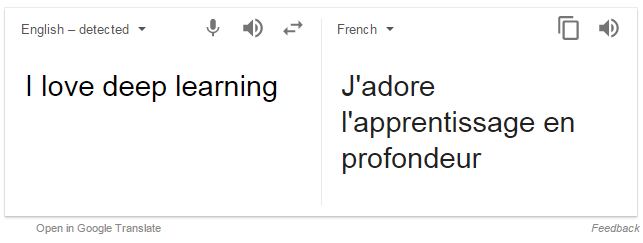
All of you would have heard about Siri, which is Apple’s voice controlled intelligent assistant. Like other big giants, Apple has also started investing on Deep Learning to make its services better than ever.



*Figure 38 – Speech Recognition “Siri”*

In the area of speech recognition and voice controlled intelligent assistant like Siri, one can develop more accurate acoustic model using a deep neural network and is currently one of the most active fields for deep learning implementation. In simple words, you can build such system that can learn new features or adapt itself according to you and therefore, provide better assistance by predicting all possibilities beforehand.

* **Automatic Machine Translation**



##### Figure 39 – Automatic Machine Translation

We all know that Google can instantly translate between 100 different human language, that too very quickly as if by magic. The technology behind *Google Translate* is called **Machine Translation** and has been savior for people who can’t communicate with each other because of the difference in the speaking language. Now, you would be thinking that this feature has been there for a long time, so, what’s new in this? Let me tell you that over the past two years, with the help of deep learning, Google has totally reformed the approach to machine translation in its Google Translate. In fact, deep learning researchers who know almost nothing about language translation are putting forward relatively simple machine learning solutions that are beating the best expert-built language translation systems in the world. Text translation can be performed without any pre-processing of the sequence, allowing the algorithm to learn the dependencies between words and their mapping to a new language. Stacked networks of large recurrent neural networks are used to perform this translation.

* + **Instant Visual Translation**



##### Figure 40 – Instant Visual Translation

As you know, deep learning is used to identify images that have letters and where the letters are on the scene. Once identified, they can be turned into text, translated and the image recreated with the translated text. This is often called **instant visual translation**.

Now, imagine a situation where you have visited any other country whose native language is not known to you. Well, no need to worry, using various apps like Google Translate you can go ahead and perform instant visual translations to read signs or shop boards written in another language. This has been possible only because of Deep Learning.

***Note:*** You can go ahead and download Google Translate App and check out the amazing instant visual translation using the above image.

###### Behaviour: Automated Self Driven Cars

Google is trying to take their self-driving car initiative, known as WAYMO, to a whole new level of perfection using Deep Learning. Therefore, rather than using old hand-coded algorithms, they can now program system that can learn by themselves using data provided by different sensors. Deep learning is now the best approach to most perception tasks, as well as to many low-level control tasks. Hence, now even people who do not know to drive or are disabled, can go ahead and take the ride without depending on anyone else.

##### Figure 41 – Behaviour: Automated Self Driven Cars

Here, I have only mentioned few famous real-life use cases where Deep Learning is being used extensively and showing promising results. There are many other applications of deep learning along with many fields which is yet to be explored.

So, this is all about deep learning in a nutshell. I am sure that by now, you would have realized the difference between Machine Learning and Deep Learning as well as how Deep Learning can be very useful for various real – life application. Now, in my next blog in this deep learning tutorial series, we will deep dive into various concepts and algorithms Deep Learning along with their application in detail.

## Data Visualization

Data visualization is the graphical representation of information and data. By using [visual elements like](https://www.tableau.com/learn/articles/data-visualization/glossary) [charts, graphs, and maps](https://www.tableau.com/learn/articles/data-visualization/glossary), data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data.

In the world of Big Data, data visualization tools and technologies are essential to analyze massive amounts of information and make data-driven decisions.

Data visualization is another form of visual art that grabs our interest and keeps our eyes on the message. When we see a chart, we [quickly see trends and outliers](https://www.tableau.com/reports/business-intelligence-trends). If we can see something, we internalize it quickly. It’s storytelling with a purpose. If you’ve ever stared at a massive spreadsheet of data and couldn’t see a trend, you know how much more effective a visualization can be.

Common general types of data visualization:

* Charts
* Tables
* Graphs
* Maps
* Infographics
* Dashboards

## Tableau

To better understand the data, Visualization tools such as Tableau, Microsoft Power BI, QlikView ae used by BI professionals, data analysts for analysing and forecasting the data.

Tableau is the data visualization tool software widely used by many companies to analyse their data and provide better insights on the data that helps the business grow. The best thing is, you can customize the dashboard according to your requirements.

###### There are 5 main products in Tableau product family:

1. Tableau Desktop
2. Tableau Server
3. Tableau Online
4. Tableau Reader
5. Tableau Online

###### Tableau is used in various departments such as:

* Aerospace & Defence
* Associations and Non-profits
* Higher Education
* K-12 Education
* Banking
* Media & Entertainment
* Pharmaceuticals and much more…..

**Tableau Intro:**

**What is Tableau?**

-Tableau is a Business Intelligence (BI) tool. BI (Business Intelligence) is a process of converting raw data into meaningful information. It is a set of process, architecture, and technologies that drives profitable business actions.

-**Business Intelligence (BI):** It is a method of collecting, storing, and analysing data from business operations or activities to optimize performance.

BI has a direct impact on the organizational strategic and operational business decision. It impacts the revenue and financial model of the business.

###### Why Tableau is a market leader?

* Because it’s
  + Easy
  + Powerful
  + Fast

And Tableau doesn’t require any tech skills or coding to operate it.

**Tableau Architecture:**

Create

Tableau

**Tableau has two components:**

Tableau Reader

Tableau Online

Share

Tableau Server

Tableau Desktop

*Figure 42 - Tableau Architecture*

1. Create: It is used to create the visualization and uses Tableau Desktop for this purpose.
2. Share: It is used to share what is created and it uses Tableau Server, Tableau Online, and Tableau Reader for this purpose.

**Process flow of BI project**

* Business Understanding: - It refers to understanding the domain of the business.
* Data Understanding: - It refers to understanding of the data which may be structured or unstructured, short or long, ambiguous or null.
* Data Preparation: - It refers to the tables, dimensions, relationships
* Modelling: - It refers to the creation of formulas, columns, functions, and advance functions
* Evaluation: - It refers to the creation and evaluation of reports, dashboards. If evaluation fails, process is repeated.
* Deployment: - It refers to the deployment of the project.

###### Tableau has 7 data types:

1. Text (String) Values
2. Date Values
3. Date & Time Values
4. Numerical Values
5. Boolean Values (relational only)
6. Geographic values (used with maps)
7. Cluster Group (Used with “find clusters” in Data)

###### There are 2 types of connections in Tableau:

1. **Live:** It is used when data is too large and it’s remote data. It is used to connect to the dataset and work on the dynamic data.
2. **Extract:** It has the ability to copy the data to local system. It is used for function modelling. We can refresh both functions manually.

###### There are four types of joins to join the tables in Tableau:

1. Inner Join
2. Left Join
3. Right Join
4. Full Outer Join

## Feature Engineering

* 1. Feature engineering is the process of transforming raw data into features that better represent the underlying problem to the predictive models, resulting in improved model accuracy on unseen data.
  2. Feature engineering involves leveraging data mining techniques to extract features from raw data along with the use of domain knowledge. Feature engineering is useful to improve the performance of machine learning algorithms and is often considered as applied machine learning
  3. Feature engineering is the process of using data's domain knowledge to create features that make machine learning algorithms work (Wikipedia). It's the act of extracting important features from raw data and transforming them into formats that are suitable for machine learning.
  4. Feature Selection: Select a subset of input features from the dataset. Unsupervised: Do not use the target variable (e.g. remove redundant variables). Correlation. Supervised: Use the target variable (e.g. remove irrelevant variables). Wrapper: Search for well-performing subsets of features. RFE.
  5. Feature engineering creates features from the existing raw data in order to increment the predictive power of the machine learning algorithms. Generally, the feature engineering process is applied togenerate additional features from the raw data.
  6. Engineering and selecting the correct features for a model will not only significantly improve its predictive power, but will also offer the flexibility to use less complex models that are faster to run and more easily understood.
  7. The most common techniques of feature scaling are Normalization and Standardization. Normalization is used when we want to boundour values between two numbers, typically, between [0,1] or [-1,1]. While Standardization transforms the data to have zero mean and a variance of 1, they make our data unitless.

## Machine Learning Pipelines

1. A machine learning pipeline is used to help automate machine learning workflows. They operate by enabling a sequence of data to be transformed and correlated together in a model that can be testedand evaluated to achieve an outcome, whether positive or negative.
2. Machine learning pipeline. One definition of a machine learning pipeline is a means of automating the machine learning workflow by enabling data to be transformed and correlated into a model that can then be analysed to achieve outputs. This type of ML pipeline makes the process of inputting data into the ML model fully automated.
3. Create the resources required to run an ML pipeline. Set up a data store used to access the data needed in the pipeline steps. Configure a Dataset object to point to persistent data that lives in, or is accessible in, a datastore. Set up the compute targets on which yourpipeline steps will run.
4. Data collection. Funnelling incoming data into a data store is the first step of any ML workflow. The key point is that data is persisted without undertaking any transformation at all, to allow us to have animmutable record of the original dataset.
5. Scikit-learn's pipeline class is a useful tool for encapsulating multipledifferent transformers alongside an estimator into one object, so that you only have to call your important methods once (fit (), predict (),etc).

## What is PyTorch

1. PyTorch is an open-source machine learning library based on the Torch library, used for applications such as computer vision and natural language processing.
2. As you might be aware, PyTorch is an open-source machine learning library used primarily for applications such as computer vision and natural language processing. PyTorch is a strong player in the field ofdeep learning and artificial intelligence, and it can be considered primarily as a research-first library.
3. So, both TensorFlow and PyTorch provide useful abstractions to reduce amounts of boilerplate code and speed up model development.
4. The main difference between them is that PyTorch may feel more “pythonic” and has an object-oriented approach while TensorFlow has several options from which you may choose.
5. PyTorch is an open-source machine learning library based on the Torch library, used for applications such as computer vision and natural language processing, primarily developed by Facebook's AIResearch lab (FAIR).

# CHAPTER 4

## Reason for choosing Data Science/ Machine Learning

Data Science has become a revolutionary technology that everyone seems to talk about. Hailed as the ‘sexiest job of the 21st century’. Data Science is a buzzword with very few people knowing about the technology in its true sense.

* Data Science/Machine Learning Jobs are on the rise
* The major hiring is happening in all top tech companies in search of those special kind of people (machine learning engineers) who can build a hammer (machine learning algorithms).
* The job market for machine learning engineers is not just hot but it’s sizzling. Machine Learning Jobs on Indeed.com - 2,500+(India) & 12,000+(US).

**Advantages**: -

1. It’s in Demand
2. Abundance of Positions
3. A Highly Paid Career
4. Data Science/ ML is Versatile

**Disadvantages**: -

1. Mastering Data Science is near to impossible
2. A large Amount of Domain Knowledge Required
3. Arbitrary Data May Yield Unexpected Results
4. The problem of Data Privacy

## LEARNING OUTCOME

#### After completing the training, I am able to:

* Develop relevant python programming abilities.
* Demonstrate proficiency with statistical analysis of data.
* Develop the skill to build and assess data-based model.
* Demonstrate skill in data management.
* Apply data science concepts and methods to solve problem in real-world contexts and will communicate these solutions effectively.
* Have a good understanding of the fundamental issues and challenges of machine learning: data, model selection, model complexity, etc.
* Have an understanding of the strengths and weaknesses of many popular machine learning approaches.
* Appreciate the underlying mathematical relationships within and across Machine Learning algorithms and the paradigms of supervised and un- supervised learning.
* Be able to design and implement various machine learning algorithms in a range of real-world applications.
* Ability to integrate machine learning libraries and mathematical and statistical tools with modern technologies
* Ability to understand and apply scaling up machine learning techniques and associated computing techniques and technologies.

## CERTIFICATE OF COMPLETION



## CONCLUSION

Being part of a very successful health software solution company, like CureYa, gave me a different perspective on the role of health information in the workforce. The experience that I have gained from working as a data science intern was very educational. In addition, working for the company showed me the variety of tasks that I could handle. Being able to utilize many of my technical skills, such as python programming, ML algorithms, DL combined with the statistical knowledge to handle the tasks assigned to me, proved to me that my informatics skills and knowledge could be applied in the workforce. The most rewarding part of working at CureYa was the satisfaction of knowing that I worked for a company whose primary objective is “Health for All”.