**CHAPTER 2**

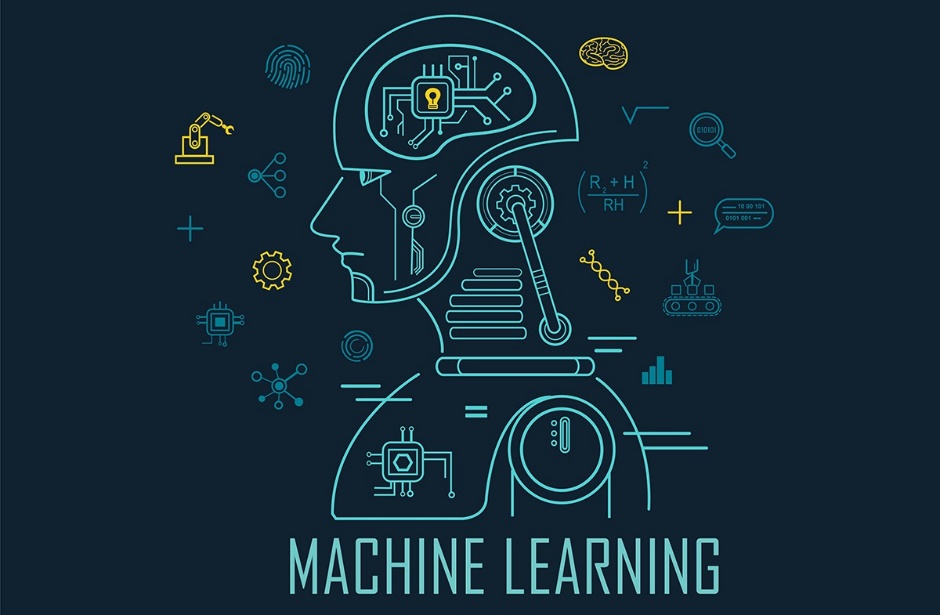
**MACHINE LEARNING**

Machine learning is a subfield of artificial intelligence (AI). The goal of machine learning generally is to understand the structure of data and fit that data into models that can be understood and utilized by people.

Although machine learning is a field within computer science, it differs from traditional computational approaches. [15] In traditional computing, algorithms are sets of explicitly programmed instructions used by computers to calculate or problem solve. Machine learning algorithms instead allow for computers to train on data inputs and use statistical analysis in order to output values that fall within a specific range. Because of this, machine learning facilitates computers in building models from sample data in order to automate decision-making processes based on data inputs.

**2.1 What is Machine Learning?**

Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy. Machine learning is an important component of the growing field of data science. Using statistical methods, algorithms are trained to make classifications or predictions, uncovering key insights within data mining projects. These insights subsequently drive decision-making within applications and businesses, ideally impacting key growth metrics as shown in fig 2.1. As big data continues to expand and grow, the market demand for data scientists will increase, requiring them to assist in the identification of the most relevant business questions and subsequently the data to answer them. A neat diagram of machine learning is shown below in fig. 2.1.



**Fig 2.1 Machine Learning**

**2.2 History of Machine Learning**

Before some years (about 40-50 years), machine learning was science fiction, but today it is the part of our daily life. Machine learning is making our day to day life easy from **self-driving cars** to **Amazon virtual assistant "Alexa"**. However, the idea behind machine learning is so old and has a long history. Below some milestones are given which have occurred in the history of machine learning:

1. The early history of Machine Learning (Pre-1940):
2. 1834: In 1834, Charles Babbage, the father of the computer, conceived a device that could be programmed with punch cards. However, the machine was never built, but all modern computers rely on its logical structure.
3. 1936: In 1936, Alan Turing gave a theory that how a machine can determine and execute a set of instructions.
4. The era of stored program computers:
5. 1940: In 1940, the first manually operated computer, "ENIAC" was invented, which was the first electronic general-purpose computer. After that stored program computer such as EDSAC in 1949 and EDVAC in 1951 were invented.
6. 1943: In 1943, a human neural network was modelled with an electrical circuit. In 1950, the scientists started applying their idea to work and analysed how human neurons might work.
7. Computer machinery and intelligence:
8. 1950: In 1950, Alan Turing published a seminal paper, "Computer Machinery and Intelligence," on the topic of artificial intelligence. In his paper, he asked, "Can machines think?"
9. Machine intelligence in Games:
10. 1952: Arthur Samuel, who was the pioneer of machine learning, created a program that helped an IBM computer to play a checkers game. It performed better more it played.
11. 1959: In 1959, the term "Machine Learning" was first coined by Arthur Samuel.
12. The first "AI" winter:
13. The duration of 1974 to 1980 was the tough time for AI and ML researchers, and this duration was called as AI winter.In this duration, failure of machine translation occurred, and people had reduced their interest from AI, which led to reduced funding by the government to the researches.
14. Machine Learning from theory to reality
15. 1959: In 1959, the first neural network was applied to a real-world problem to remove echoes over phone lines using an adaptive filter.
16. 1985: In 1985, Terry Sejnowski and Charles Rosenberg invented a neural network NETtalk, which was able to teach itself how to correctly pronounce 20,000 words in one week.
17. 1997: The IBM's Deep blue intelligent computer won the chess game against the chess expert Garry Kasparov, and it became the first computer which had beaten a human chess expert.
18. Machine Learning at 21st century
19. 2006: In the year 2006, computer scientist Geoffrey Hinton has given a new name to neural net research as "deep learning," and nowadays, it has become one of the most trending technologies.
20. 2012: In 2012, Google created a deep neural network which learned to recognize the image of humans and cats in YouTube videos.
21. 2014: In 2014, the Chabot "Eugen Goostman" cleared the Turing Test. It was the first Chabot who convinced the 33% of human judges that it was not a machine.
22. 2014: DeepFace was a deep neural network created by Facebook, and they claimed that it could recognize a person with the same precision as a human can do.
23. 2016: AlphaGo beat the world's number second player Lee sedol at Go game. In 2017 it beat the number one player of this game Ke Jie.
24. 2017: In 2017, the Alphabet's Jigsaw team built an intelligent system that was able to learn the online trolling. It used to read millions of comments of different websites to learn to stop online trolling.
25. Machine Learning at present:
26. Now machine learning has got a great advancement in its research, and it is present everywhere around us, such as self-driving cars, Amazon Alexa, Catboats, recommender system, and many more. It includes Supervised, unsupervised, and reinforcement learning with clustering, classification, decision tree, SVM algorithms, etc.
27. Modern machine learning models can be used for making various predictions, including weather prediction, disease prediction, stock market analysis, etc.

**2.3 How does Machine Learning Work?**

UC Berkeley (link resides outside IBM) breaks out the learning system of a machine learning algorithm into three main parts.

A Decision Process: In general, machine learning algorithms are used to make a prediction or classification. Based on some input data, which can be labelled or unlabelled, your algorithm will produce an estimate of a pattern in the data.

An Error Function: An error function serves to evaluate the prediction of the model. If there are known examples, an error function can make a comparison to assess the accuracy of the model. A Model Optimization Process: If the model can fit better to the data points in the training set, then weights are adjusted to reduce the discrepancy between the known example. and the model estimate. The algorithm will repeat this evaluate and optimize process, updating weights autonomously until a threshold of accuracy has been met.

**2.4 Literature Survey**

A core objective of a learner is to generalize from its experience. The computational analysis of machine learning algorithms and their performance is a branch of theoretical computer science known as computational learning theory. Because training sets are finite and the future is uncertain, learning theory usually does not yield guarantees of the performance of algorithms. Instead, probabilistic bounds on the performance are quite common. The bias–variance decomposition is one way to quantify generalization error.

For the best performance in the context of generalization, the complexity of the hypothesis should match the complexity of the function underlying the data. If the hypothesis is less complex than the function, then the model has underfit the data. If the complexity of the model is increased in response, then the training error decreases. But if the hypothesis is too complex, then the model is subject to overfitting and generalization will be poorer.

In addition to performance bounds, learning theorists study the time complexity and feasibility of learning. In computational learning theory, a computation is considered feasible if it can be done in polynomial time. There are two kinds of time complexity results. Positive results show that a certain class of functions can be learned in polynomial time. Negative results show that certain classes cannot be learned in polynomial time.

**2.5 The Challenges Facing Machine Learning**

While there has been much progress in machine learning, there are also challenges. For example, the mainstream machine learning technologies are black-box approaches, making us concerned about their potential risks. To tackle this challenge, we may want to make machine learning more explainable and controllable. As another example, the computational complexity of machine learning algorithms is usually very high and we may want to invent lightweight algorithms or implementations. Furthermore, in many domains such as physics, chemistry, biology, and social sciences, people usually seek elegantly simple equations (e.g., the Schrödinger equation) to uncover the underlying laws behind various phenomena. Machine learning takes much more time. You have to gather and prepare data, then train the algorithm. There are much more uncertainties. That is why, while in traditional website or application development an experienced team can estimate the time quite precisely, a machine learning project used for example to provide product recommendations can take much less or much more time than expected. Why? Because even the best machine learning engineers don’t know how the deep learning networks will behave when analysing different sets of data. It also means that the machine learning engineers and data scientists cannot guarantee that the training process of a model can be replicated.

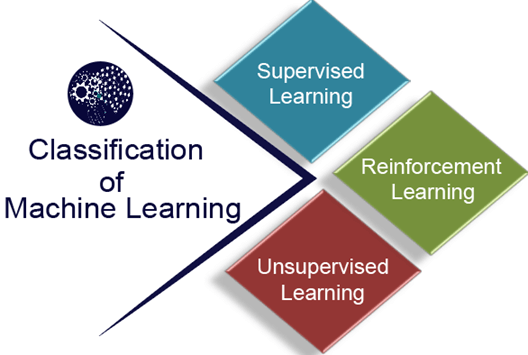
**2.6 Features of Machine Learning**

1. Machine learning uses data to detect various patterns in a given dataset.
2. It can learn from past data and improve automatically.
3. It is a data-driven technology.
4. Machine learning is much similar to data mining as it also deals with a huge amount of data.

**2.7 Types of Machine Learning**

There are three types of machine learning as shown in fig 2.7:

1. Supervised
2. Unsupervised
3. Reinforcement

****

**Fig 2.7 Types of Machine Learning**

### **2.7.1 Supervised Learning**

Supervised learning, also known as supervised machine learning, is a subcategory of machine learning and artificial intelligence. It is defined by its use of labelled datasets to train algorithms that to classify data or predict outcomes accurately. As input data is fed into the model, it adjusts its weights until the model has been fitted appropriately, which occurs as part of the cross-validation process. Supervised learning helps organizations solve for a variety of real-world problems at scale, such as classifying spam in a separate folder from your inbox.

### **2.7.2 Unsupervised Learning**

Unsupervised learning refers to the use of artificial intelligence (AI) algorithms to identify patterns in data sets containing data points that are neither classified nor labelled. Unsupervised learning is commonly used for finding meaningful patterns and groupings inherent in data, extracting generative features, and exploratory purposes.

### **2.7.3 Reinforcement Learning**

Reinforcement learning is a machine learning training method based on rewarding desired behaviours and/or punishing undesired ones. In general, a reinforcement learning agent is able to perceive and interpret its environment, take actions, and learn through trial and error.

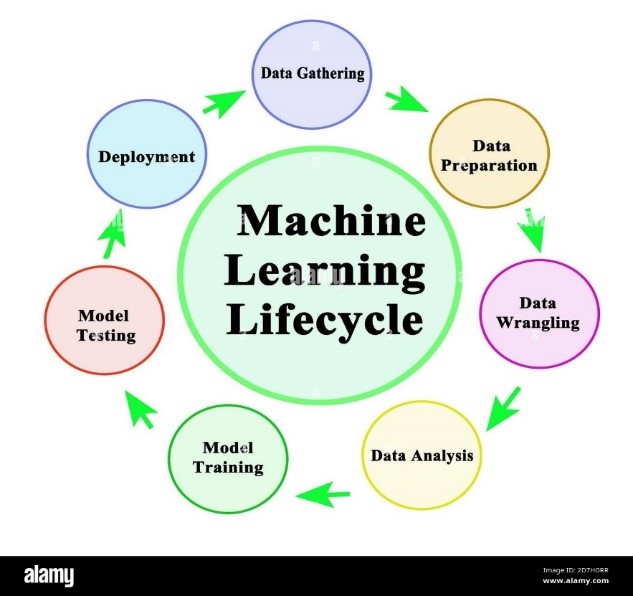
**2.8 Machine Learning Lifecycle**

Machine learning has given the computer systems the abilities to automatically learn without being explicitly programmed. But how does a machine learning system work? So, it can be described using the life cycle of machine learning. Machine learning life cycle is a cyclic process to build an efficient machine learning project. The main purpose of the life cycle is to find a solution to the problem or project. A neat diagram of machine learning lifecycle is shown below in fig. 2.8.

Machine learning life cycle involves seven major steps, which are given below:

1. Gathering Data
2. Data preparation
3. Data Wrangling
4. Analyse Data
5. Train the model
6. Test the model
7. Deployment

The most important thing in the complete process is to understand the problem and to know the purpose of the problem. Therefore, before starting the life cycle, we need to understand the problem because a good result depends on a better understanding of the problem. In the complete life cycle process, to solve a problem, we create a machine learning system called a "model", and this model is created by providing "training". But to train a model, we need data, hence, the life cycle starts by collecting data.



**Fig. 2.8 Machine Learning Lifecycle**

The most important thing in the complete process is to understand the problem and to know the purpose of the problem. Therefore, before starting the life cycle, we need to understand the problem because a good result depends on a better understanding of the problem. In the complete life cycle process, to solve a problem, we create a machine learning system called a "model", and this model is created by providing "training". But to train a model, we need data, hence, the life cycle starts by collecting data:

1. **Data Gathering**

Data Gathering is the first step of the machine learning life cycle. The goal of this step is to identify and obtain all data-related problems.

In this step, we need to identify the different data sources, as data can be collected from various sources such as **files**, **databases**, the **internet**, or **mobile devices**. It is one of the most important steps of the life cycle. The quantity and quality of the collected data will determine the efficiency of the output. The more will be the data, the more accurate will be the prediction.

1. **Data preparation**

After collecting the data, we need to prepare it for further steps. Data preparation is a step where we put our data into a suitable place and prepare it to use in our machine learning training.

1. **Data Wrangling**

Data wrangling is the process of cleaning and converting raw data into a useable format. It is the process of cleaning the data, selecting the variable to use, and transforming the data in a proper format to make it more suitable for analysis in the next step. It is one of the most important steps of the complete process. Cleaning of data is required to address the quality issues.

1. **Analyse Data**

The aim of this step is to build a machine learning model to analyse the data using various analytical techniques and review the outcome. It starts with the determination of the type of the problems, where we select the machine learning techniques such as Classification, Regression, Cluster analysis, Association, etc. then build the model using prepared data, and evaluate the model.

1. **Train the model**

Now the next step is to train the model, in this step we train our model to improve its performance for a better outcome of the problem. We use datasets to train the model using various machine learning algorithms. Training a model is required so that it can understand the various patterns, rules, and features.

1. **Test the model**

Once our machine learning model has been trained on a given dataset, then we test the model. In this step, we check for the accuracy of our model by providing a test dataset to it. Testing the model determines the percentage accuracy of the model as per the requirement of project or problem.

1. **Deployment**

The last step of machine learning life cycle is deployment, where we deploy the model in the real-world system. If the above-prepared model is producing an accurate result as per our requirement with acceptable speed, then we deploy the model in the real system. But before deploying the project, we will check whether it is improving its performance using available data or not. The deployment phase is similar to making the final report for a project.

**2.9 Applications of Machine Learning**

1. **Web Search Engine:** One of the reasons why search engines like google, bing etc work so well is because the system has learnt how to rank pages through a complex learning algorithm.
2. **Photo tagging Applications:** Be it Facebook or any other photo tagging application, the ability to tag friends makes it even more happening. It is all possible because of a face recognition algorithm that runs behind the application.
3. **Spam Detector:** Our mail agent like Gmail or Hotmail does a lot of hard work for us in classifying the mails and moving the spam mails to spam folder. This is again achieved by a spam classifier running in the back end of mail application.
4. **Database Mining for growth of automation:** Typical applications include Web-click data for better UX, Medical records for better automation in healthcare, biological data and many more.
5. **Applications that cannot be programmed:** There are some tasks that cannot be programmed as the computers we use are not modelled that way. Examples include Autonomous Driving, Recognition tasks from unordered data (Face Recognition/ Handwriting Recognition), Natural language Processing, computer Vision etc.
6. **Understanding Human Learning:** This is the closest we have understood and mimicked the human brain. It is the start of a new revolution, The real AI. Now, after a brief insight lets come to a more formal definition of Machine Learning

**2.10 Future Scope**

` Future of Machine Learning is as vast as the limits of human mind. We can always keep learning, and teaching the computers how to learn. And at the same time, wondering how some of the most complex machine learning algorithms have been running in the back of our own mind so effortlessly all the time. There is a bright future for machine learning. Companies like Google, Quora, and Facebook hire people with machine learning. There is intense research in machine learning at the top universities in the world. The global machine learning as a service market is rising expeditiously mainly due to the Internet revolution. The process of connecting the world virtually has generated vast amount of data which is boosting the adoption of machine learning solutions. Considering all these applications and dramatic improvements that ML has brought us, it doesn't take a genius to realize that in coming future we will definitely see more advanced applications of ML, applications that will stretch the capabilities of machine learning to an unimaginable level.

**CHAPTER 3**

**MACHINE LEARNING LANGUAGES**

Machine learning is a growing area of computer science and several programming languages support ML framework and libraries. Among all of the programming languages, Python is the most popular choice followed by C++, Java, JavaScript, and C#.

**3.1 Python – The New Generation Language**

Python is a widely used general-purpose, high level programming language. It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation. It was mainly developed for an emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code. Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.

**3.1.1 Features**

1. **Interpreted:** In Python there is no separate compilation and execution steps like C/C++. It directly run the program from the source code. Internally, Python converts the source code into an intermediate form called bytecodes which is then translated into native language of specific computer to run it.
2. **Platform Independent:** Python programs can be developed and executed on the multiple operating system platform. Python can be used on Linux, Windows, Macintosh, Solaris and many more.
3. **Multi- Paradigm:** Python is a multi-paradigm programming language. Object-oriented programming and structured programming are fully supported, and many of its features support functional programming and aspect oriented programming.
4. **Simple:** Python is a very simple language. It is a very easy to learn as it is closer to English language. In python more emphasis is on the solution to the problem rather than the syntax.
5. **Rich Library Support:** Python standard library is very vast. It can help to do various things involving regular expressions, documentation generation, unit testing, threading, databases, web browsers, CGI, email, XML, HTML, WAV files, cryptography, GUI and many more.
6. **Free and Open Source:** Firstly, Python is freely available. Secondly, it is open-source. This means that its source code is available to the public. We can download it, change it, use it, and distribute it. This is called FLOSS (Free/Libre and Open-Source Software). As the Python community, we’re all headed toward one goal- an ever-bettering Python.

**3.1.2 Why Python Is a Suitable Language for Machine Learning?**

1. **A great library ecosystem:** A great choice of libraries is one of the main reasons Python is the most popular programming language used for AI. A library is a module or a group of modules published by different sources which include a pre-written piece of code that allows users to reach some functionality or perform different actions. Python libraries provide base level items so developers don’t have to code them from the very beginning every time. ML requires continuous data processing, and Python’s libraries let us access, handle and transform data. These are some of the most widespread libraries you can use for ML and AI:
2. **Scikit-learn** for handling basic ML algorithms like clustering, linear and logistic regressions, regression, classification, and others. o Pandas for high-level data structures and analysis. It allows merging and filtering of data, as well as gathering it from other external sources like Excel, for instance.
3. **Keras** for deep learning. It allows fast calculations and prototyping, as it uses the GPU in addition to the CPU of the computer. o TensorFlow for working with deep learning by setting up, training, and utilizing artificial neural networks with massive datasets.
4. **Matplotlib** for creating 2D plots, histograms, charts, and other forms of visualization. o NLTK for working with computational linguistics, natural language recognition, and processing.
5. **Scikit** image for image processing.
6. **PyBrain** for neural networks, unsupervised and reinforcement learning.
7. **Caffe** for deep learning that allows switching between the CPU and the GPU and processing 60+ mln images a day using a single NVIDIA K40 GPU.
8. **StatsModels** for statistical algorithms and data exploration.

In the PyPI repository, we can discover and compare more python libraries.

1. **A low entry barrier:** Working in the ML and AI industry means dealing with a bunch of data that we need to process in the most convenient and effective way. The low entry barrier allows more data scientists to quickly pick up Python and start using it for AI development without wasting too much effort into learning the language. In addition to this, there’s a lot of documentation available, and Python’s community is always there to help out and give advice.
2. **Flexibility:** Python for machine learning is a great choice, as this language is very flexible:
3. It offers an option to choose either to use OOPs or scripting.
4. There’s also no need to recompile the source code, developers can implement any changes and quickly see the results.
5. Programmers can combine Python and other languages to reach their goals.
6. **Good Visualization Options:** For AI developers, it’s important to highlight that in artificial intelligence, deep learning, and machine learning, it’s vital to be able to represent data in a human-readable format. Libraries like Matplotlib allow data scientists to build charts, histograms, and plots for better data comprehension, effective presentation, and visualization. Different application programming interfaces also simplify the visualization process and make it easier to create clear reports.
7. **Community Support:** It’s always very helpful when there’s strong community support built around the programming language. Python is an open-source language which means that there’s a bunch of resources open for programmers starting from beginners and ending with pros. A lot of Python documentation is available online as well as in Python communities and forums, where programmers and machine learning developers discuss errors, solve problems, and help each other out. Python programming language is absolutely free as is the variety of useful libraries and tools.
8. **Growing Popularity:** As a result of the advantages discussed above, Python is becoming more and more popular among data scientists. According to Stack Overflow, the popularity of Python is predicted to grow until 2020, at least. This means it’s easier to search for developers and replace team players if required. Also, the cost of their work maybe not as high as when using a less popular programming language.

**3.2 Introduction to R**

R is a language and environment for statistical computing and graphics. It is a GNU project which is similar to the S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. R can be considered as a different implementation of S. There are some important differences, but much code written for S runs unaltered under R.

R provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, …) and graphical techniques, and is highly extensible. The S language is often the vehicle of choice for research in statistical methodology, and R provides an Open-Source route to participation in that activity.

One of R’s strengths is the ease with which well-designed publication-quality plots can be produced, including mathematical symbols and formulae where needed. Great care has been taken over the defaults for the minor design choices in graphics, but the user retains full control.

R is available as Free Software under the terms of the Free Software Foundation’s GNU General Public License in source code form. It compiles and runs on a wide variety of UNIX platforms and similar systems (including FreeBSD and Linux), Windows and MacOS.

**3.2.1 Features**

1. **Open-source**

R is an open-source software environment. It is free of cost and can be adjusted and adapted according to the user’s and the project’s requirements. You can make improvements and add packages for additional functionalities. R is freely available. You can learn how to install R, Download and start practicing it.

1. **Strong Graphical Capabilities**

R can produce static graphics with production quality visualizations and has extended libraries providing interactive graphic capabilities.

1. **Highly Active Community**

R has an open-source library which is supported by its growing number of users. The R environment is continuously growing. This growth is due to its large user-base.

1. **A Wide Selection of Packages**

CRAN or Comprehensive R Archive Network houses more than 10,000 different packages and extensions that help solve all sorts of problems in data science. High-quality interactive graphics, web application development, quantitative analysis or machine learning procedures, there is a package for every scenario available. R contains a sea of packages for all the forms of disciplines like astronomy, biology, etc. While R was originally used for academic purposes, it is now being used in industries as well.

1. **Comprehensive Environment**

R has a very comprehensive development environment meaning it helps in statistical computing as well as software development. R is an object-oriented programming language. It also has a robust package called Rshiny which can be used to produce full-fledged web apps. Combined with data analysis and data visualization, R can be used for highly interactive online data-driven storytelling.

1. **Can Perform Complex Statistical Calculations**

R can be used to perform simple and complex mathematical and statistical calculations on data objects of a wide variety. It can also perform such operations on large data sets.

1. **Running Code Without a Compiler**

R is an interpreted language which means that it does not need a compiler to make a program from the code. R directly interprets provided code into lower-level calls and pre-compiled code.

1. **Data Variety**

R can handle a variety of structured and unstructured data. It also provides various data modelling and data operation facilities due to its interaction with databases.

1. **Cross-platform Support**

Cross Platform compatible with R. R is machine-independent. It supports the cross-platform operation. Therefore, it can be used on many different operating systems.

**3.2.2 Why R Is a Suitable Language for Machine Learning?**

1. It provides good explanatory code. For example, if you are at the early stage of working with a machine learning project and you need to explain the work you do, it becomes easy to work with R language comparison to python language as it provides the proper statistical method to work with data with fewer lines of code.
2. R language is perfect for data visualization. R language provides the best prototype to work with machine learning models.
3. R language has the best tools and library packages to work with machine learning projects. Developers can use these packages to create the best pre-model, model, and post-model of the machine learning projects. Also, the packages for R are more advanced and extensive than python language which makes it the first choice to work with machine learning projects.
4. Suitable for Analysis — if the data analysis or visualization is at the core of your project then R can be considered as the best choice as it allows rapid prototyping and works with the datasets to design machine learning models.
5. The bulk of useful libraries and tools — Similar to Python, R comprises of multiple packages which help to improve the performance of the machine learning projects. For instance — Caret boosts the machine learning capabilities of the R with its special set of functions which helps to create predictive models efficiently. R developers gain advantage from the advanced data analysis packages which cover the pre- and post-modelling stages which are directed at specific tasks like model validation or data visualization.
6. Suitable for exploratory work — If you require any exploratory work in statistical models at the beginning stages of your project then R makes it easier to write them as the developers just need to add a few lines of code.