

**TRAINING REPORT**

**OF**

**SUMMER TRAINING , UNDERTAKEN AT**

**"NIELIT CHANDIGARH"**

**ON**

**"DIABETES ANALYSIS AND PREDICTION"**

**OF**

**BACHELOR OF TECHNOLOGY**

**IN**

**Computer Science and Engineering**

**Submitted By :-**

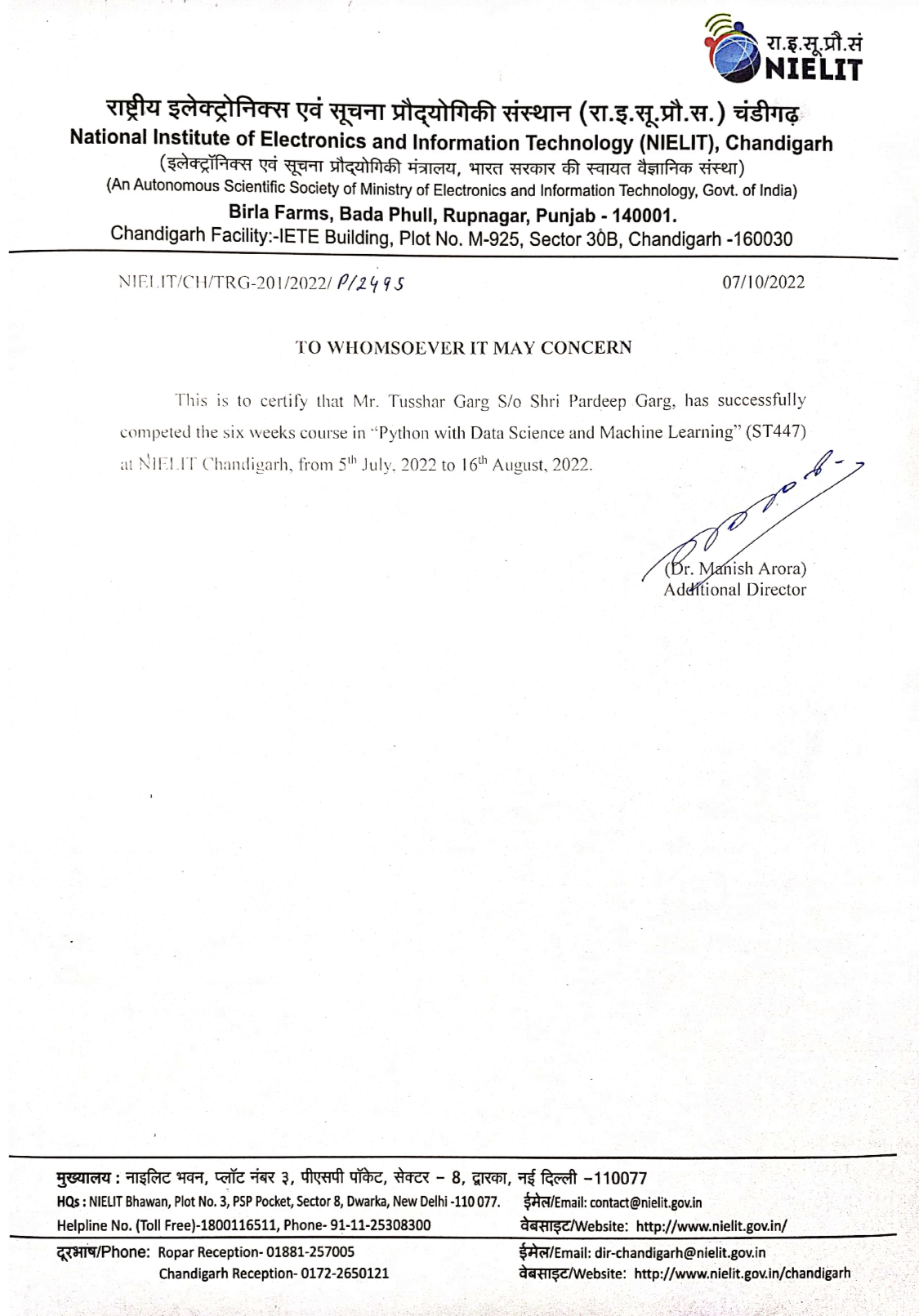
**Name : TUSSHAR GARG**

**Roll No. : 12001148**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**PUNJABI UNIVERSITY PATIALA-(147002)**

**Certificate**

****

**CANDIDATE'S DECLARATION**

I “**Tusshar Garg**” hereby declare that I have undertaken Summer training at “**NIELIT Chandigarh**” during a period from 05-07-2022 to 16-08-2022 in partial fulfilment of requirements for the award of degree of B.tech (Department of Computer Science & Engineering) at Punjabi University, Patiala. The authentic record of my work which is being presented in the training report is submitted to Department of Computer Science & Engineering) at Punjabi University, Patiala is an authentic record of training work.

Signature of the Student

Tusshar Garg

12001148

The summer training Viva–Voce Examination of has

been held on \_\_\_and accepted.

Signature of Examiner

**ABSTRACT**

This work has been done as a part of industrial training. The purpose of industrial training is to

familiarize the students with the present working environment in country and outside.

To provide the students with the opportunity to study the latest technical trends those have been established as well as one that what they are learning during the period of industrial training will certainly help them to develop their potential and technical skills. As part of our industrial training, I undertook a project of **“DIABETES ANALYSIS AND PREDICTION”**. We undertook classes of **Python, Data Science and Machine Learning** where I came to know which problems can occur during making of project.

Machine learning in medicine has recently made headlines. Google has developed a machine learning algorithm to help identify cancerous tumors on mammograms. Stanford is using a deep learning algorithm to identify skin cancer. A recent JAMA article reported the results of a deep machine-learning algorithm that was able to diagnose diabetic retinopathy in retinal images. It’s clear that machine learning puts another arrow in the quiver of clinical decision making. Still, machine learning lends itself to some processes better than others. Algorithms can provide immediate benefit to disciplines with processes that are reproducible or standardized. Also, those with large image datasets, such as radiology, cardiology, and pathology, are strong candidates. Machine learning can be trained to look at images, identify abnormalities, and point to areas that need attention, thus improving the accuracy of all these processes. Long term, machine learning will benefit the family practitioner or internist at the bedside. Machine learning can offer an objective opinion to improve efficiency, reliability, and accuracy. We have applied machine learning to predict whether the patient has diabetes or not.

**ACKNOWLEDGEMENT**

It is a great pleasure to present this report on the project named “DIABETES ANALYSIS AND PREDICTION**”**.” undertaken by us as part of my B.tech (CSE) curriculum. We are thankful to our college University College of Engineering, Punjabi University, Patiala Campus, for offering us such a wonderfully challenging opportunity. We express our deepest thanks to all coordinators for providing all the possible help and assistance and their constant encouragement. We express our sincere gratitude to Respected H.O.D of Computer Science and Engineering Department for providing us an opportunity to undergo this summer training. I would also like to thank my instructors at the company Ms Anita Budhiraja and Mr. Sarwan Singh. They were always by myside and helped me throughout the time I was in the company. They made all the “Python with Data Science and Machine Learning”, easy to understand. Whenever I was stuck at any point in completing my project, they helped me overcome that problem, thus, enhancing my problem solving and analytical skills.

**ABOUT THE COMPANY**

**COMPANY’ S VISION:**

To be the leader in the development of industry-oriented quality education and training and be the country's premier Institution for examination and certification in the field of Information, Electronics and Communications Technology (IECT).To be the single source for quality assurance in computer education among the nation's non-formal institutes, after turning out competent IT professionals in large numbers, NIELIT reach is now being extended to all regions of the country as well as overseas.

**OBJECTIVES:**

1. To generate quality manpower and develop skilled professionals in the area of Information, Electronics and Communications Technology (IECT) and allied, by providing world class education & training and accreditation services;

2. To provide continuing support to learners and trainers through active design & development of innovative curricula and acquisition of content, aligned with the dynamically changing IECT;

3. To establish a quality system of examination and certification that is globally recognised providing a fair assessment of the competency of students;

4. To continue to implement the DOEACC Scheme for computer courses, jointly developed by AICTE and DIT (now MeitY) in the non-formal sector of IT Education and Training;

5. To establish standards in the areas of IECT and to develop markets in the emerging areas;

6. To impart continuing Education for up-gradation of knowledge and skills in view of high obsolescence in the area of IECT;

7. To provide entrepreneurship development programme in the area of IECT;

8. To develop and promote the culture of a market to encourage and nurture industry oriented design and development;

9. To purchase, lease, hire, exchange or otherwise require any immovable property and the sell, lease, dispose off, exchange, invest and deal with bonds, moneys, securities and all kinds of movable or immovable properties for all or any of the objectives for which the Society is establish.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| S.NO | Topic | PAGE NO. |
| 01 | INTRODUCTION | 9 |
| 1.1 | CONTEXT | 9 |
| 1.2 | MOTIVATION | 9 |
| 1.3 | OBJECTIVE | 9 |
| 1.4 | PROBLEM STATEMENT | 9 |
| 1.5 | PYTHON | 10 |
| 02 | TRAINING WORK UNDERTAKEN | 16 |
| 2.1 | PYTHON BASICS | 16 |
| 2.2 | INTRODUCTION TO DATASCIENCE | 23 |
| 2.3 | MACHINE LEARMING | 28 |
| 2.4 | PROJECT WORK UNDERTAKEN | 34 |
| 03 | RESULT AND DISCUSSION | 35 |
| 04 | CONCLUSION AND FUTURE SOPE | 38 |
| 4.1 | CONCLUSION | 38 |
| 4.2 | FUTURE SCOPE | 38 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| Fig. No. | Figure Name | PAGE NO. |
| 01 | Python Library | 11 |
| 02 | SCIKIT-Learning | 14 |
| 03 | Python Function | 21 |
| 04 | Machine Learning | 21 |
| 05 | Numerical Operations Of Arrays | 24 |
| 06 | Arrays | 25 |
| 07 | Linear Regression | 31 |
| 08 | Data Set | 33 |
| 09 | Histogram Plot | 34 |
| 10 | Violin Plot | 34 |
| 11 | Scatter Plot | 35 |
| 12 | Positive Output | 37 |
| 13 | Negative Output | 37 |

**CHAPTER 1: INTRODUCTION**

**1.1 CONTEXT**

This project was made because we were intrigued and we wanted to gain hands-on

experience with the Machine Learning Project.

**1.2 MOTIVATION**

We are highly interested in anything related to Machine Learning, the independent project provided us with the opportunity to study and reaffirm our passion for this subject. The capacity to generate guesses, forecasts, and offer machines the ability to learn on their own is both powerful and infinite in terms of application possibilities. Machine Learning may be applied in finance, medicine, and virtually any other field. That is why we opted to base

our idea on Machine Learning.

**1.3 OBJECTIVE**

As a first project, we intended to make it as instructional as possible by tackling each stage of the machine learning process and attempting to comprehend it well. We have picked DIABETES ANALYSIS ANDPREDICTION project in which we will analyse and predict whether the patients have diabetes or not.

**1.4 PROBLEM STATEMENT**

Machine learning has been used for years to offer image recognition, spam detection, natural

speech comprehension, product recommendations, and medical diagnoses. Today, machine learning algorithms can help us enhance cyber security, ensure public safety, and improve medical outcomes. Machine learning systems can also make customer service better and automobiles safer. When we started experimenting with machine learning, we wanted to come up with an application that would solve a real-world problem but would not be too complicated to implement. We also wanted to practice working with regression algorithms. So I started looking for a problem worth solving.

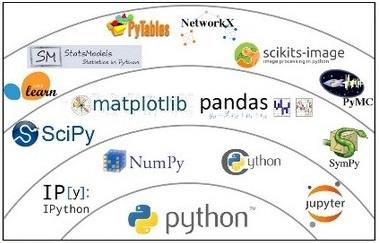
**1.5 Python:**

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

* **Python is Interpreted** − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
* **Python is Interactive** − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
* **Python is Object-Oriented** − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
* **Python is a Beginner's Language** − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands. It is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.

It supports functional and structured programming methods as well as OOP. It can be used as a scripting language or can be compiled to byte-code for building large applications. It provides very high-level dynamic data types and supports dynamic type checking.IT supports automatic garbage collection.It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.



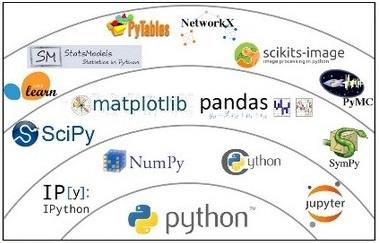


Fig1:Python Libraries

LIBRARIES USED IN THE PROJECT :

1. Pandas
2. Matplotlib
3. Seaborn
4. Tkinter
5. Scikit – Learn

**PANDAS:**

Pandas is an open-source, BSD-licensed Python library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language. Python with Pandas is used in a wide range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc.

In 2008, developer Wes McKinney started developing pandas when in need of high performance, flexible tool for analysis of data.

Prior to Pandas, Python was majorly used for data munging and preparation. It had very little contribution towards data analysis. Pandas solved this problem. Using Pandas, we can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data — load, prepare, manipulate, model, and analyze.

Python with Pandas is used in a wide range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc.

**Key Features of Pandas -**

* Fast and efficient DataFrame object with default and customized indexing.
* Tools for loading data into in-memory data objects from different file formats.
* Data alignment and integrated handling of missing data.
* Reshaping and pivoting of date sets.
* Label-based slicing, indexing and subsetting of large data sets.
* Columns from a data structure can be deleted or inserted.
* Group by data for aggregation and transformations.
* High performance merging and joining of data.
* Time Series functionality.

**MATPLOTLIB:**

Matplotlib is a plotting library for Python. It is used along with NumPy to provide an environment that is an effective open source alternative for MatLab. It can also be used with graphics toolkits like PyQt and wxPython.

Matplotlib module was first written by John D. Hunter. Since 2012, Michael Droettboom is the principal developer.

Matplotlib is probably the single most used Python package for 2D-graphics. It provides both a very quick way to visualize data from Python and publication-quality figures in many formats. We are going to explore matplotlib in interactive mode covering most common cases.

**SEABORN:**

Seaborn is an open source, BSD-licensed Python library providing high level API for visualizing the data using Python programming language. Seaborn is a visualization library in Python. It is built on top of Matplotlib.

It is summarized that if Matplotlib “tries to make easy things easy and hard things possible”, Seaborn tries to make a well-defined set of hard things easy too.”

Seaborn helps resolve the two major problems faced by Matplotlib; the problems are−

* Default Matplotlib parameters
* Working with data frames

As Seaborn compliments and extends Matplotlib, the learning curve is quite gradual. If you know Matplotlib, you are already half way through Seaborn.

**Important Features of Seaborn**

Seaborn is built on top of Python’s core visualization library Matplotlib. It is meant to serve as a complement, and not a replacement. However, Seaborn comes with some very important features. Let us see a few of them here. The features help in − built in themes for styling matplotlib graphics, visualizing univariate and bivariate data, fitting in and visualizing linear regression models, plotting statistical time series data, Seaborn works well with NumPy and Pandas data structures, it comes with built in themes for styling Matplotlib graphics.

**Tkinter:**

Tkinter is the standard GUI library for Python. The name *Tkinter* comes from *Tk interface*. Tkinter was written by Fredrik Lundh. Python when combined with Tkinterprovides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.

Creating a GUI application using Tkinter is an easy task. All you need to do is perform the following steps −

* Import the *Tkinter* module.
* Create the GUI application main window.
* Add one or more of the above-mentioned widgets to the GUI application.
* Enter the main event loop to take action against each event triggered by the user.

Both Tk and Tkinter are available on most Unix platforms, as well as on Windows and Macintosh systems. Starting with the 8.0 release, Tk offers native look and feel on all platforms.

Tkinter consists of a number of modules. The Tk interface is provided by a binary extension module named **\_tkinter**. This module contains the low -level interface to Tk, and should never be used directly by application programmers. It is usually a shared library (or DLL), but might in some cases be statically linked with the Python interpreter.

The public interface is provided through a number of Python modules. The most important interface module is the **Tkinter** module itself.

**SCIKIT– Learning**

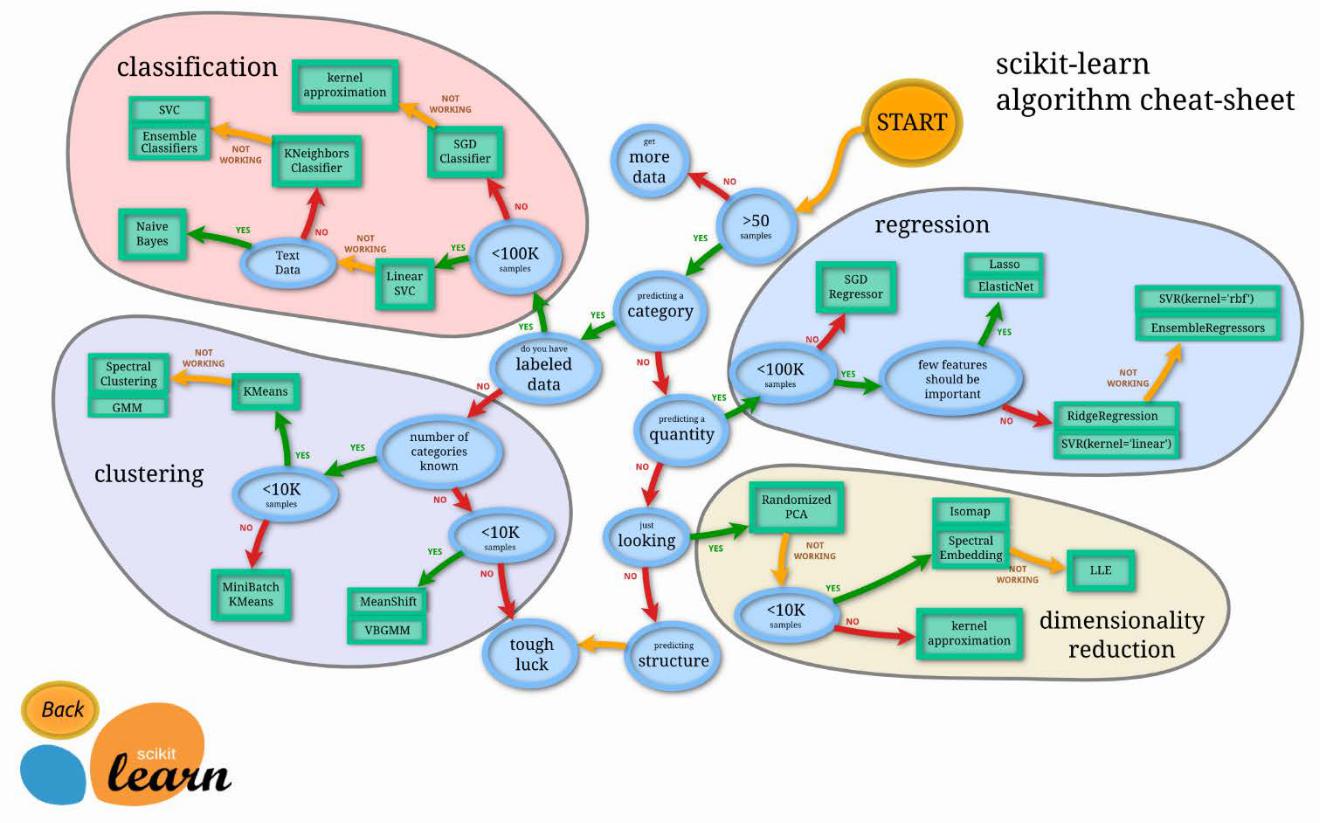


Fig2:SCIKIT– Learning

**Scikit-learn** is a [free software](https://en.wikipedia.org/wiki/Free_software) [machine learning](https://en.wikipedia.org/wiki/Machine_learning) [library](https://en.wikipedia.org/wiki/Library_(computing)) for the programming

language. It features various [classification,](https://en.wikipedia.org/wiki/Statistical_classification) [regression](https://en.wikipedia.org/wiki/Regression_analysis) and [clustering](https://en.wikipedia.org/wiki/Cluster_analysis) algorithms including [support vector machines,](https://en.wikipedia.org/wiki/Support_vector_machine) [random forests,](https://en.wikipedia.org/wiki/Random_forests) [gradient boosting,](https://en.wikipedia.org/wiki/Gradient_boosting) [*k*](https://en.wikipedia.org/wiki/K-means_clustering)

[means](https://en.wikipedia.org/wiki/K-means_clustering) and [DBSCAN,](https://en.wikipedia.org/wiki/DBSCAN) and is designed to interoperate with the Python numerical and scientific libraries [NumPy](https://en.wikipedia.org/wiki/NumPy) and [SciPy.](https://en.wikipedia.org/wiki/SciPy)

Scikit-learn is probably the most useful library for machine learning in Python. It is on NumPy, SciPy and matplotlib, this library contains a lot of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction. Note that *scikit-learn is used to build models*. It should not be used for reading the data, manipulating and summarizing it. There are better libraries for that (e.g. NumPy, Pandas etc.)

Machine learning is where computational and algorithmic skills of data science meet the statistical thinking of data science.

* The result is a collection of approaches to inference and data exploration that are *not about effective theory* so much as *effective computation*.
* Better to think of machine learning as a *means of building models of Data*
* Machine learning along with entire Data Science ecosystem is trying to make this mathematical, modelbased “learning” as same as “learning” exhibited by the human brain.

**Why Machine Learning Matters ?**

With the rise in big data, machine learning has become a key technique for solving problems in areas, such as:

1. Computational finance, *for credit scoring and algorithmic trading.*
2. Image processing and computer vision, *for face recognition, motion detection, and object detection.*
3. Computational biology, *for tumor detection, drug discovery, and DNA sequencing.*
4. Energy production, *for price and load forecasting.*
5. Automotive, aerospace, and manufacturing, *for predictive maintenance.*
6. Natural language processing, *for voice recognition applications.*

**CHAPTER 2: TRAINING WORK UNDERTAKEN**

**2.1 PYTHON BASICS**



Python was derived from **Monty Python's Flying Circus** in 1970 by **Van Rossum**. 1. Later on, **Iron Python** was developed which works with **.NET**.

2. **Jython** was developed which works with **Java Virtual Machine**.

3. **Django** was developed which works with **Web Framework**.

**Ecosystem**

There are various packages like:

1. NumPy

2. Pandas

3. Scikit

4. Matplotlib

5. CartoPy

**Features**

1. Python can be used on a server to create web applications.
2. Python can be used alongside software to create workflows.
3. Python can connect to database systems. It can also read and modify files.
4. Python can be used to handle big data and perform complex mathematics.
5. Python can be used for rapid prototyping, or for production-ready software development.
6. **Shell** interacts with machine. Nothing will be saved permanently. But we have to make files and folders. So, we don’t use shell. We will make use of **IDLE(Integrated Development Environment).** Examples:- **Jupyter, Spyder**

Top 5 IDE's for Data Science:

1.RodeoIDE

2. Pycharm

3. Jupyter

4. Spyder

5. Atom IDE

**Data types in Python**

Every value in Python has a datatype. Since everything is an object in Python programming, data types are actually classes and variables are instance (object) of these classes.

There are various data types in Python. Some of the important types are listed below.

**NUMBERS**

"=" sign is used to assign values. Assignment is always from right to left. var1=1

print(var1) #For printing the value of var1 del var1 #For deleting its value

**Type Conversion/Type Casting** refers to conversion of one data type to another. In python, we make use of **input keyword** also. Input always take value in strings.

a=int(input("Enter value of a:")) #Here int is function print("a:",a);

**STRINGS**

In python, consecutive sequence of characters is known as string. An individual character in a string is accessed using a subscript (index). The subscript should always be an integer(positive or negative). A subscript starts from 0. Ways for writing string:-

e="This is 'python' training"; f='This is "python" training'; g="This is \"python\" training"; h='This is \'python\' training';

The main functions of strings are as follows:

* ord() function returns the numeric value of char.
* chr() function returns character corresponding to ascii code.
* casefold() function converts the string to lowercase
* count() function count how many times a letter occurs in a string
* find() function get the position of character in a string
* endswith() function check if string ends with a specified value
* lstrip() function delete all the leading characters(from left side)
* rstrip() function delete all the trailing characters(from right side)
* isspace() function check if all the characters in the string are whitespace characters
* istitle() function returns true if the string is titlecased string otherwise returns false
* partition() function takes a separator as the argument that separates the string at its first occurrence.
* split() function splits the string in list of words
* str() function converts integer to string.

**LISTS**

In lists, values are enclosed in square brackets. Values in the list can be modified i.e. it is mutable. For accessing an element of the list, indexing is used. Declaration of list is as follows:-

L1=[12,65,78,34,59]; #list of numbers

city=["Chd","Bassi","Ropar"]; #list of strings

aa=[12,13,'Chd'];

The main functions of lists are len(), append(), count(), extend(), insert(), pop(), remove(),

sort(), del(), clear() etc.

 len() function tells the length of string.

 count() function count the occurences of specified object in the list  append() function will add the item at the end

 extend() function add the items of one list to other list

 pop() and remove() functions are used to delete elements. In pop(),we give index value whereas in remove(),we give the element we want to remove. Without index, pop() will delete the last element by default.

 reverse() function reverse the order of list and items.

**List Comprehension**:- A Python list comprehension consists of brackets containing the

expression, which is executed for each element along with the for loop to iterate over each element in the python list. Python List comprehension provides a much more short syntax for creating a new list based on the values of an existing list.

Syntax of list comprehension is:-

newList **= [** expression(element) **for** element **in** oldList **if** condition **]** For example:-

list **=** [i **for** i **in** range(11) **if** i **%** 2 **==** 0]

print(list)

Output**:**

[0, 2, 4, 6, 8, 10]

TUPLES

Tuples are used to store multiple items in a single variable. Tuple is one of 4 built-in data

types in Python used to store collections of data, the other 3 are List, Set and Dictionary, all with different qualities and usage. A tuple is a collection which is ordered and **unchangeable**.

Tuples are same as list but tuples can't be changed and thus are immutable. Declaration of tuples is as follows:-

t1='a','b','c' #tuple without parentheses

t1=('a','b','c') #tuple with parentheses

When we say that tuples are ordered, it means that the items have a defined order, and that order will not change. Tuples are unchangeable, meaning that we cannot change, add or remove items after the tuple has been created.

The difference between list and tuple is that lists are in [] brackets and tuples in () brackets.

Lists can be changed, extended and deleted, but not tuples. Tuples can hold homogenous and heterogenous data, but in list we store homogenous data. When we create a tuple, we

normally assign values to it. This is called "**packing**" a tuple. But, in python, we are also allowed to extract the values back into variables. This is called "**unpacking**" a tuple.

DICTIONARY

Dictionary is an unordered collection of data values, used to store data values like a map, which unlike other Data types that hold only single value as an element, Dictionary holds key:value pair. Dictionary is used for storing values with keys. Dictionaries are like address books. The declaration of dictionary is as follows:-mydict={"name":"anu","phone":234156,"city":"chd"}

Here name is key and anu is value. **Dictionary functions and methods**

 get() function will get the element value

 pop() function will remove/pop the mentioned

 keys() function will print keys only

 values() function will print values only

 items() function will print all items with values

 clear() function will clear all elements of dictionary.

LOOPS

Python programming language provides the following types of loops to handle looping requirements. Python provides three ways for executing the loops. While all the ways provide similar basic functionality, they differ in their syntax and condition checking time.

**While Loop:-**A while loop is used to execute a block of statements repeatedly until a given

condition is satisfied. And when the condition becomes false, the line immediately after the loop in the program is executed. Syntax of while loop:-

#while(condt): #<body> #statement

For example:-Printing i as long as i is less than 6:

i = 1

while i <6:

print(i)

i += 1

**For Loop:-**A for loop is used for iterating over a sequence (that is either a list, a tuple, a

dictionary, a set, or a string).

For example:-Printing first five integers using for loop for i in range(1,6):

print(i, end=" ");

**Nested For Loop:-**A nested loop is a loop inside a loop.The "inner loop" will be executed

one time for each iteration of the "outer loop".For example:-Printing tables from 2 to 5 for i in range(2,6):

for j in range(1,11): print(i,"\*",j,"=",i\*j,end="\n");

print(" ")

FUNCTIONS

**Python Functions** is a block of statements that return the specific task. The idea is to put

some commonly or repeatedly done tasks together and make a function so that instead of writing the same code again and again for different inputs, we can do the function calls to reuse code contained in it over and over again.

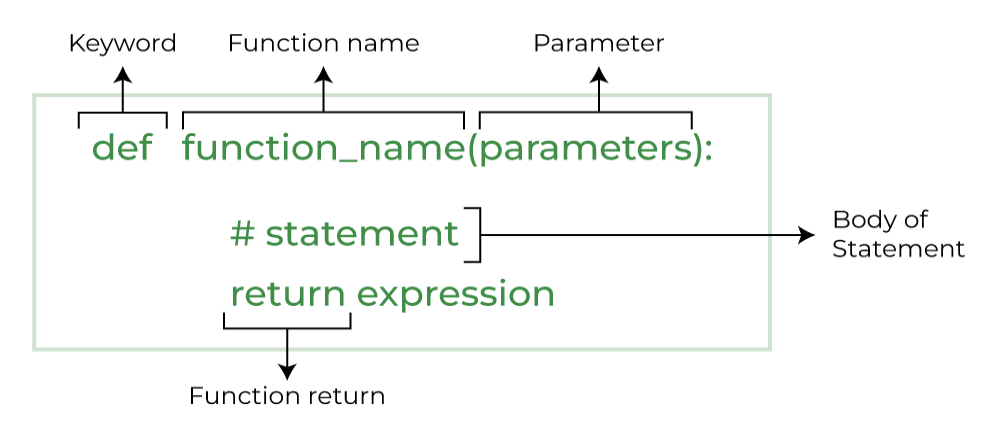


Fig3:PYTHON FUNCTION

For example:-Function for adding two number

Function definition def sumtwonum(a,b):

c=a+b; print(c);

Calling of function sumtwonum(10,7);

**Lambda Function:-** Lambda keyword is used to create small Anonymous function. These

function are called anonymous because they are not declared as standard function. Lambda forms can take any number of arguments but return one value. They cannot contain commands or multiple expressions. Syntax:

lambda[arg1[arg2,....]]:expression For example:- Function definition sum=lambda arg1,arg2:arg1+arg2; Calling sum function

print("Value of total is",sum(10,20));

**LEGB Rule**:-LEGB Rule Multiple namespaces can exist independently from each other and

they can contain the same variable names on different hierarchy levels. The "Scope" defines on which hierarchy level Python searches for a particular "variable name" for its associated object. The order in which Python search the different levels of namespaces before it finds the name-to-object' mapping is-LEGB rule.

**MODULES**

**Math Module:-** Sometimes when working with some kind of financial or scientific projects it becomes necessary to implement mathematical calculations in the project. Python provides the **math module** to deal with such calculations. Math module provides functions to deal with both basic operations such as addition(+), subtraction(-), multiplication(\*), division(/) and advance operations like trigonometric, logarithmic, exponential functions.

Importing math module:-import math as mt r=mt.sqrt(56)

print(r)

**Random Module:-**Python **Random module** is an in-built module of Python which is used to generate random numbers. These are pseudo-random numbers means these are not truly random. This module can be used to perform random actions such as generating random numbers, print random a value for a list or string, etc.

Importing random module:-

 random function-it gives a random number between 0 and 1 random.random()

 randint(a,b)-it gives a random number between a and b random.randint(15,100)

 uniform(a,b)-it gives a floating point random number between a and b random.uniform(5,10)

**2.2 INTRODUCTION TO DATA SCIENCE**

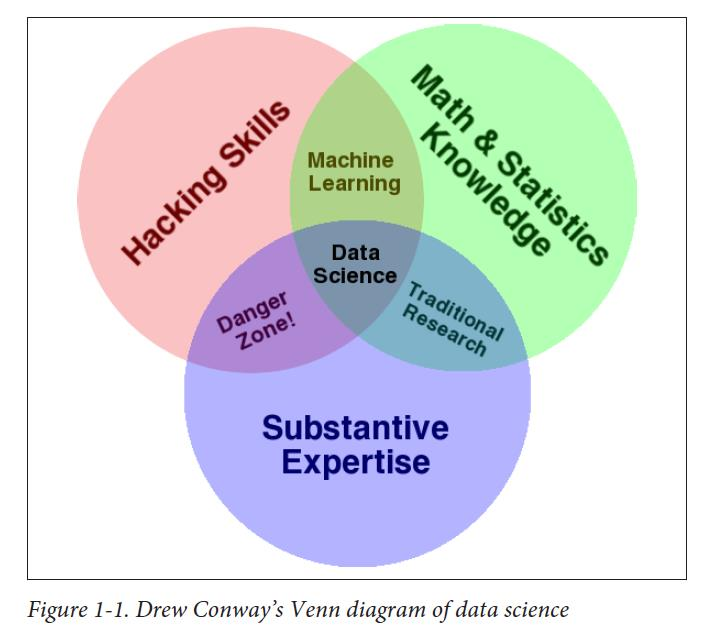
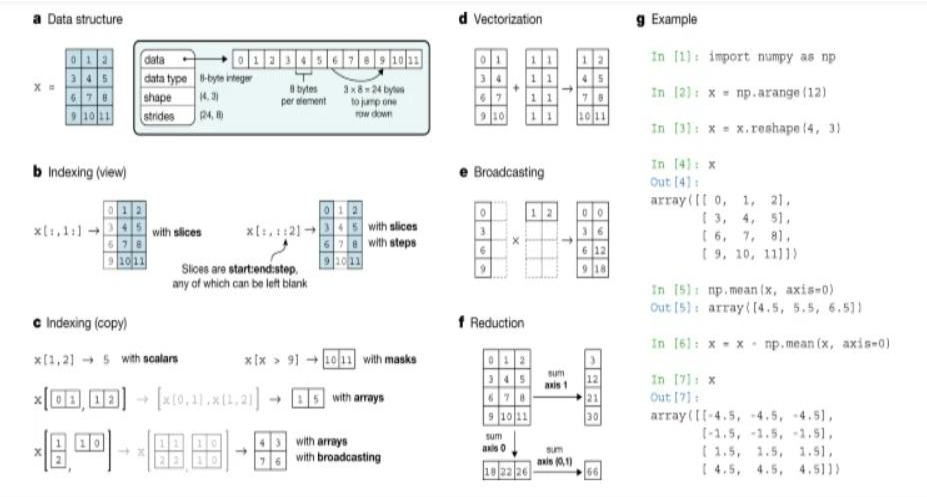
Data science is a multidisciplinary area that uses scientific methods, procedures, tools and systems to extract knowledge and get insights into structured and unstructured data. Data science is related to data analytics, data mining and big data. It understands the phenomenon of the data. It employs techniques and theories drawn from many fields within the context of mathematics, statistics, computer science, and information science.

Fig4: Machine Learning

Statistics is one of the most important disciplines to provide tools and methods to find structure in and to give deeper insight into data, and the most important discipline to analyze an quantify. uncertainty. Python provides various predefined modules to work on Data science projects[13]

NUMPY

In the Python world, NumPyarrays are the standard representation for numerical data. Overall, three techniques are applied to improve performance: vectorizing calculations, avoiding copying data in memory, and minimizing operation counts.



The NumPy array is a data structure that efficiently stores and accesses multidimensional array[s17](https://www.nature.com/articles/s41586-020-2649-2#ref-CR17) (also known as tensors), and enables a wide variety of scientific computation. It consists of a pointer to memory, along with metadata used to interpret the data stored there, notably ‘data type’, ‘shape’ and ‘strides’ .

A NumPy array is a multidimensional, uniform collection of elements. An array is characterized by the type of elements it contains and by its shape. For example, a matrix may be represented as an array of shape (M ×N ) that contains numbers, e.g., ﬂoating point or complex numbers. Unlike matrices, NumPy arrays can have any dimensionality.

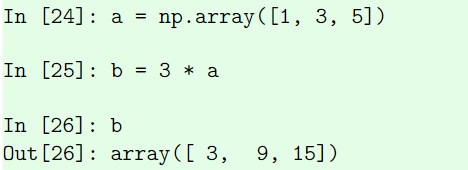
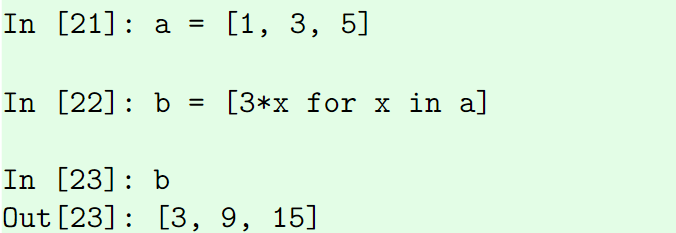
NumPy is imported as follows: import numpy as np

Elements contained in an array can be indexed using the [] operator. In addition, parts of an

array may be retrieved using standard Python slicing of the form start:stop:step. For instance, the ﬁrst two rows of an array x are given by x[:2, :] or columns 1 through 3 by x[:, 1:4].

Fig5:Numerical operations on arrays: vectorization

Grouping these element-wise operations together,a process known as vectorisation, allows NumPy to perform such computations much more rapidly.



Suppose we have a vector a and wish to multiply its magnitude by 3. A traditional for-loop approach would look as follows:

Fig6:Array

The vectorized approach applies this operation to all elements of an array:

Vectorized operations in NumPy are implemented in C, resulting in a signiﬁcant speed improvement. Operations are not restricted to interactions between scalars and arrays.

**Broadcasting**

Before broadcasting two arrays, NumPy veriﬁes that all dimensions are suitably matched. Dimensions match when they are equal, or when either is 1 or None. In the latter case, the dimension of the output array is expanded to the larger of the two.For example, consider arrays x and y with shapes (2,4,3) and (4,1) respectively. These arrays are to be combined in a broadcasting operation such as z = x + y. We match their dimensions as follows:

x (2, 4, 3) y ( 4, 1) z (2, 4, 3)

Therefore, the dimensions of these arrays are compatible, and yield and output of shape (2, 4, 3).[14][15]

**PANDAS**

Pandas is Python package for data analysis. Pandas is an open source python library built on top of Numpy. Pandas can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data:-

 load

 prepare

 manipulate

 model

 analyze

**Why Pandas?**

\* Big part of data science is data cleaning.

\* Pandas is a power tool for data cleaning.

**Key Features of Pandas**

\* Fast and efficient dataframe object with default and customized indexing.

\* Label based slicing, indexing and subsetting of large datasets.

\* Columns from a data structure can be deleted or inserted.

\* High performance merging and joining of data.

\* Time series functionality.

**NumPy vs Pandas**

\* Pandas is designed for working with tabular or heterogenous data.

\* NumPy is best suited for working with homogenous numerical array data.

\* **NumPy** is required by **Pandas**(and by virtually all numerical tools for Python)

\* NumPy is faster than Pandas.

\* Both are faster than normal Python arrays.

\* **Essential difference**:-

\* The NumPy array has an **implicitly defined integer index** used to access the values.

\* The Pandas Series has an **explicitly defined index** associated with the values.

**Pandas and NumPy**

\* Pandas and NumPy both hold data.

\* Pandas has column names as well.

\* Makes it easier to manipulate data.

**Pandas NumPy Scikit-Learn Workflow**

\* Start with CSV

\* Convert to Pandas

\* Slice and dice in Pandas

\* Convert to NumPy array to feed to scikit-learn.

**File access in pandas**

Pandas library work with different type of files with extension like. xlsx, csv, pdf format. The below shown practical import excel file in pandas and display the content of file in structured format using pandas frame class library.[16] Importing the pandas library and aliasing as pd: import pandas as pd

**Data Structures**

\* Pandas deals with the following two data structures:-

\* Series

\* DataFrame

\* These data structures are built on top of NumPy array, which means they are fast.

\* DataFrame is a container of Series.

**Pandas Series**

\* Series is a **1D** labeled **homogenous** array,size **immutable**.

\* Syntax:- pandas.Series(data,index,dtype,copy)

\* A series can be created using various inputs-

\* Array

\* Dict

\* Scalar value or constant

**Pandas Dataframe**

\* General **2D** labeled(rows and columns),size **mutable**,**tabular** structure with potentially **heterogeneously** typed columns

\* Syntax:- pandas.DataFrame(data,index,columns,dtype,copy) \* pandas DataFrame can be created using various inputs like-

\* Lists

\* Dict

\* Series

\* Numpy ndarrays

\* Another Dataframe

**MATPLOTLIB**

Data visualization is very important. Data visualization helps us to better understand the data, discover things that wouldn’t discover in raw format and communicate findings more efficiently to others. The best and most well-known Python data visualization library is Matplotlib. It is not easy to use, but usually it provides many functions like barchart, scatterplot, piechart, histogram etc which are useful for projecting many dimensions of data.

**SCIPY**

Mathematics deals with a huge number of concepts that are very important but at the same time, complex and time-consuming. However, Python provides the fullfledged scipy library that resolves this issue for us. In this scipy, we will be learning how to make use of this library along with a few functions and their examples.

**2.3 MACHINE LEARNING**

The term machine learning refers to the automated detection of meaningful patterns in data. In the past couple of decades it has become a common tool in almost any task that requires information extraction from large data sets. We are surrounded by a machine learning based technology: search engines learn how to bring us the best results (while placing profitable ads), anti-spam software learns to filter our email messages, and credit card transactions are secured bysoftwarethat learns how to detect frauds. Digital cameras learn to detect faces and intelligent personal assistance applications on smartphones learn to recognize voice commands. Cars are equipped with accident prevention systems that are built using machine learning algorithms. Machine learning is also widely used in scientific applications such as bioinformatics, medicine, and astronomy. One common feature of all of these applications is that, in contrast to more traditional uses of computers, in these cases, due to the complexity of the patterns that need to be detected, a human programmer cannot providean explicit, fine detailed specification of how such tasks should be executed. Taking example from intelligent beings, many of our skills are acquired or refined through learning from our experience (rather than following explicit instructions given to us). Machine learning tools are concerned with endowing programs with the ability to “learn “and adapt. Because machine learning is typically used to process large volumes of data, you may want to choose a powerful low-level language. However, if you’re only just beginning to explore this field, it might be better to start with Python. Python is beginner-friendly, and can do the same thing that other coding language.

can, but in fewer lines of code. If you are interested in exploring machine learning with Python, this paper will serve as your guide.

Machine learning is a sub-domain of computer science which evolved from the study of pattern recognition in data, and also from the computational learning theory in artificial intelligence. Basically, 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. 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. Machine Learning can be thought of as the study of a list of sub-problems, viz: decision making, clustering, classification, forecasting, deep-learning etc. In Supervised learning, we have a training set, and a test set. The training and test set consists of a set of examples consisting of input and output vectors, and the goal of the supervised learning algorithm is to infer a function that maps the input vector to the output vector with minimal error. In layman’s terms, supervised learning can be termed as the process of concept learning, where a brain is exposed to a set of inputs and result vectors and the brain learns the concept that relates said inputs to outputs. A wide array of supervised machine learning algorithms are available to the machine learning enthusiast, for example Neural Networks, Decision Trees, Support Vector Machines, Random Forest, Naïve Bayes Classifier, Bayes Net etc. As a field, machine learning is closely related to computational statistics, so having a background knowledge in statistics is useful for understanding and leveraging machine learning algorithms. **Correlation** is a measure of association between two variables that are not designated as either dependent or independent. **Regression** at a basic level is used to examine the relationship between one dependent and one independent variable.

Machine learning algorithms are broadly classified into three categories - *supervised learning, unsupervised learning, and reinforcement learning.*

1. **Supervised Learning** - Learning where data *is* labelled and the motivation is to classify something or predict a value. *Example: Detecting fraudulent transactions from a list of credit card transactions.*

2. **Unsupervised Learning** - Learning where data is *not* labelled and the motivation is to find patterns in given data. In this case, you are asking the machine learning model

process the data from which you can then draw conclusions. *Example: Customer segmentation based on spend data.*

3. **Reinforcement Learning** - Learning by trial and error. This is the closest to how humans learn. The motivation is to find optimal policy of how to act in a given environment. The machine learning model examines all possible actions, makes a policy that maximizes benefit, and implements the policy(trial). If there are errors from the initial policy, apply reinforcements back into the algorithm and continue to do this until you reach the optimal policy. *Example: Personalized recommendations on streaming platforms like YouTube.*

**What are the two types of supervised learning?**

As supervised learning is used to classify something or predict a value, naturally there are two types of algorithms for supervised learning - *classification models and regression models.*

1. **Classification model** - In simple terms, a classification model predicts possible

outcomes.-

2. **Regression model** - Are used to predict a numerical value.

**Classification**

The variable to be predicted, will take two values either 1 or 0 • 1 Positive Class

• 0 Negative Class

Two types

• Two Class Classification/Binary Classification: takes two values 0 or 1 Examples

• Spam or Not Spam • Cancer Malignant or Benign • Rain Yes or No • Admission Yes or No • Multi Class Classification Problem: takes more than 2 values

Examples

• Identifying Digits • Identifying some images • Identifying some categories*.*

**Regression model**

1. Regression is a technique for investigating the relationship between independent variables or features and a dependent variable or outcome. It’s used as a method for predictive modelling in machine learning, in which an algorithm is used to predict continuous outcomes.
2. Solving regression problems is one of the most common applications for machine learning models, especially in supervised machine learning. Algorithms are trained to understand the relationship between independent variables and an outcome or dependent variable. The model can then be leveraged to predict the outcome of new and unseen input data, or to fill a gap in missing data.
3. Regression analysis is an integral part of any forecasting or predictive model, so is a common method found in machine learning powered predictive analytics. Alongside classification, regression is a common use for supervised machine learning models. This approach to training models required labelled input and output training data. Machine learning regression models need to understand the relationship between features and outcome variables, so accurately labelled training data is vital.

**LINEAR REGRESSION MODEL**

1. Linear regression is one of the easiest and most popular Machine Learning algorithms. It is a statistical method that is used for predictive analysis. Linear regression makes predictions for continuous/real or numeric variables such as **sales, salary, age, product price,** etc.
2. Linear regression algorithm shows a linear relationship between a dependent (y) and one or more independent (y) variables, hence called as linear regression. Since linear regression shows the linear relationship, which means it finds how the value of the dependent variable is changing according to the value of the independent variable.

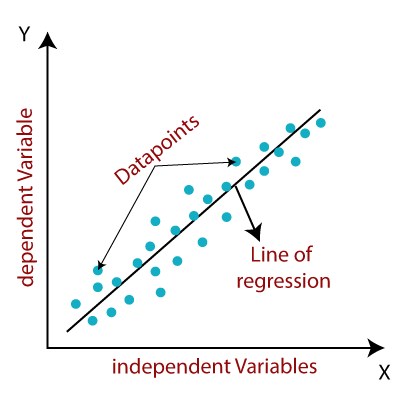


Fig7: Linear regression Graph

1. Mathematical expression for linear regression is :-

y= a0+a1x+ ε o

1. Y=DependentVariable(TargetVariable)

X=IndependentVariable(predictorVariable)

a0=intercept

1. a1 =Linear regression coefficient , ε = random error
2. The values for x and y variables are training datasets for Linear Regression model representation



Fig7:Linear Regression

**2.4 Project Work Undertaken**

Machine learning in medicine has recently made headlines. [Google has developed a](https://www.mercurynews.com/2017/03/03/google-computers-trained-to-detect-cancer/) [machine learning algorithm](https://www.mercurynews.com/2017/03/03/google-computers-trained-to-detect-cancer/) to help identify cancerous tumors on mammograms. [Stanford is using a deep learning algorithm](https://news.stanford.edu/2017/01/25/artificial-intelligence-used-identify-skin-cancer/) to identify skin cancer. A [recent JAMA article](https://jamanetwork.com/journals/jama/article-abstract/2588763) reported the results of a deep machine -learning algorithm that was able to diagnose diabetic retinopathy in retinal images. It’s clear that machine learning puts another arrow in the quiver of clinical decision making.

Still, machine learning lends itself to some processes better than others. Algorithms can provide immediate benefit to disciplines with processes that are reproducible or standardized. Also, those with large image datasets, such as radiology, cardiology, and pathology, are strong candidates. Machine learning can be trained to look at images, identify abnormalities, and point to areas that need attention, thus improving the accuracy of all these processes. Long term, machine learning will benefit the family practitioner or internist at the bedside. Machine learning can offer an objective opinion to improve efficiency, and reliability.

**CHAPTER 3-RESULTS AND DISCUSSION**

In this project we will analyse and predict whether the patients have diabetes or not. So first of all we will read the csv file of patients from which we get the data of the patients which contains the following columns-

* Pregnancies
* Blood pressure
* Skin thickness
* Insulin
* BMI
* Diabetes Pedigree Function
* Age
* Outcome

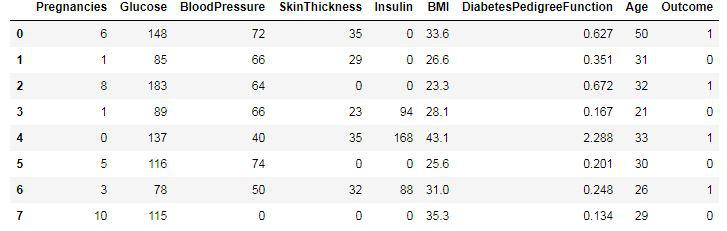
Sampled datasheet is given below

Fig8:Dataset

With the help of this datasheet we analysed the data using various graphs like –

* Histogram plot

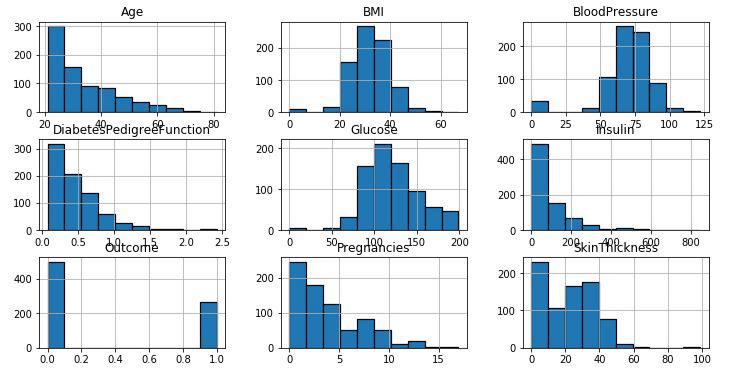


Fig9:Histogram Plot

* Violin plot

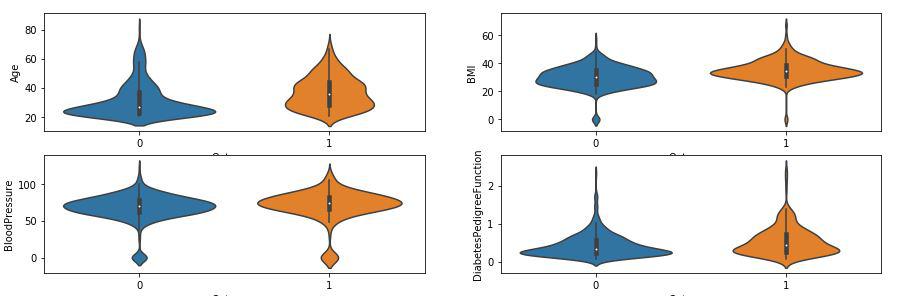


Fig10: Violin Plot

* Scatter plot

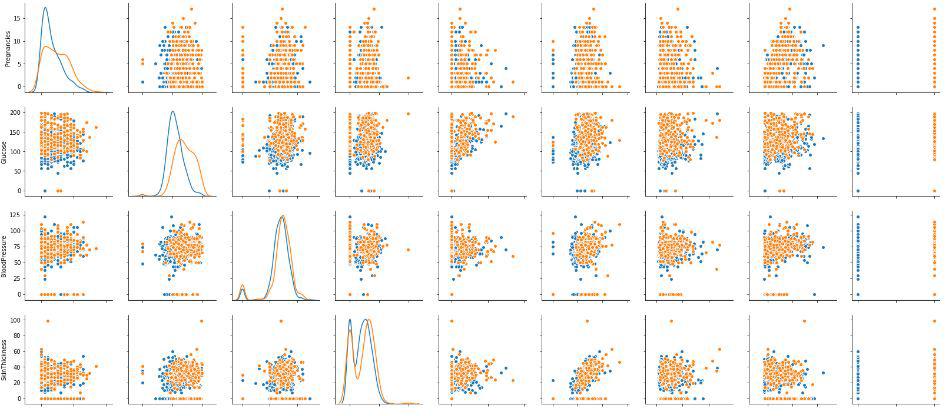
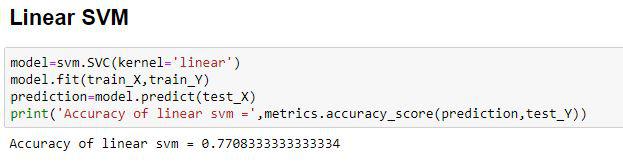
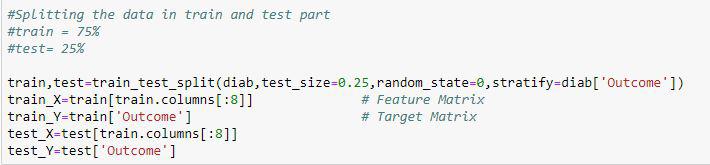


Fig11:Scatter Plot

The diagonal grouping of all pairs of attributes, suggests a low correlation and poor predictable relationship. That’s why it will be difficult to get a high accuracy.

Now we have performed the following standard method for prediction-

1. Choose a class of model by importing the appropriate estimator class from Scikit-Learn.
2. Choose model hyperparameters by instantiating this class with desired values.
3. Arrange data into a features matrix and target vector
4. Fit the model to your data by calling the fit() method of the model instance.
5. Apply the model to new data:
   * For supervised learning, often we predict labels for unknown data using the predict() method.
   * For unsupervised learning, we often transform or infer properties of the data using the transform() or predict() method .



We have applied the following ML algorithms :-

* Linear SVM
* RBF SVM
* Decision Tree Classifier
* Logistic Regression
* K-Neighbour Classifier

Accuracy of above algorithms :-

* Accuracy of linear SVM = 77%
* Accuracy of RBF SVM = 65%
* Accuracy of Decision Tree Classifier = 77%
* Accuracy of Logistic Regression = 75%
* Accuracy of K-Neighbour Classifier (i=7) = 79%

Therefore we have chosen K-Neighbour Classifier (i=7) for our prediction.

GUI APPLICATION USING TKINTER

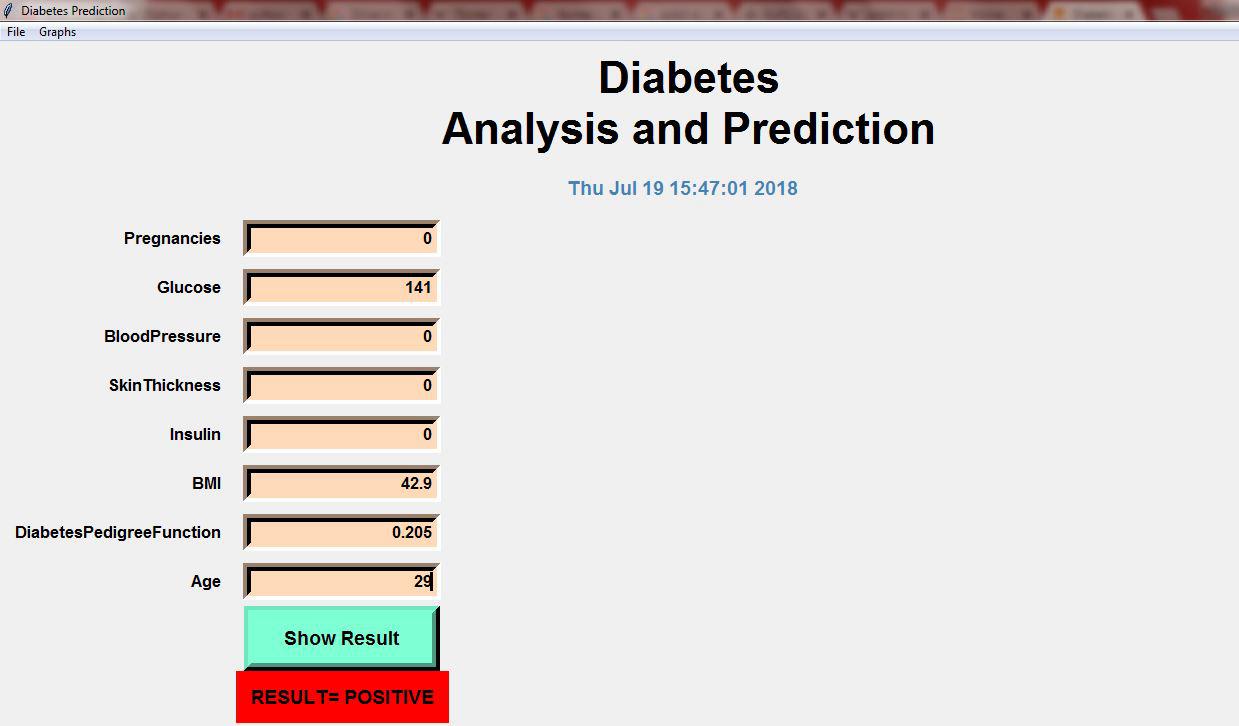
Patient with diabetes prediction **positive** :-

Fig12: Positive Output

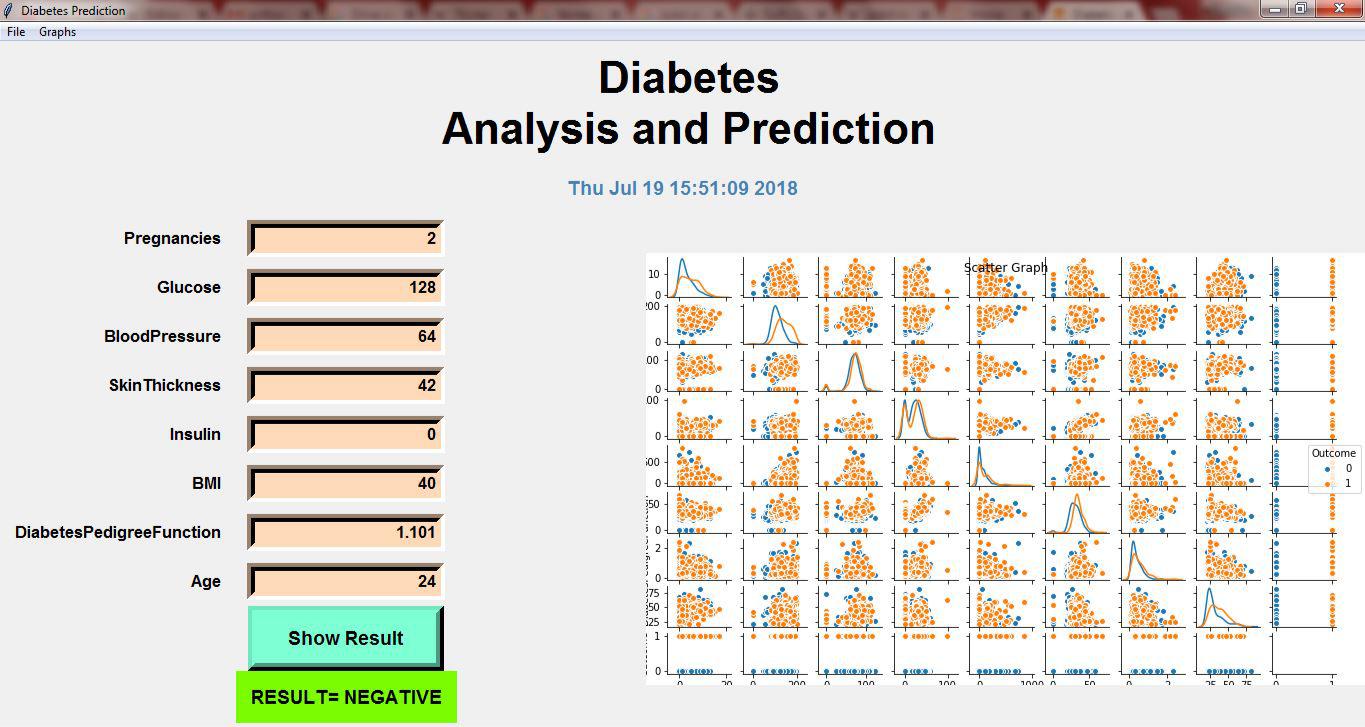
Patient with diabetes prediction **negative** :-

Fig13: Negetive Output

**CHAPTER 4-CONCLUSION AND FUTURE SCOPE**

**4.1 CONCLUSION**

Diabetes is a heterogeneous group of diseases. It’s characterized by chronic elevation of glucose in the blood. The main motto of the American diabetes association [46] is “To prevent and cure diabetes and to improve the lives of all people affected by diabetes”. To support the lives of the people all over the world, we are trying to detect and prevent the complications of diabetes at the early stage through predictive analysis by improving the classification techniques. Our proposed work also performs the analysis of the features in the dataset and selects the optimal features based on the correlation values. The decision tree algorithm and Random forest giving the highest specificity of 98.20% and 98.00%, respectively holds best for the analysis of diabetic data. Support vector machine and NB techniques give the accuracy of 77.73% and 73.48% respectively from the existing method and the proposed method improves the accuracy of the classification techniques. Improved SVM accuracy is 77% and NB accuracy is 82.30%, hence it is able to map the features effectively from low dimensions to high dimensions. It gives the best fit to the data with respect to the diabetic and non-diabetic patients. The Disease prevalence percentage is measured highest from the SVM is 45.7% .

**4.2 FUTURE SCOPE**

We suggest that working on large data sets, better result can be found. We have taken only regression algorithm for diabetes analysis and prediction by applying different regression algorithm to improve the performance of model. This model is useful for development in the field of medical science.