INTERNSHIP REPORT

A report submitted in partial fulfillment of the requirements for the Award of Degree of

BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING
(Artificial Intelligence & Machine Learning)

Submitted by
BABU VIJAYA RAGHAVAN M
Reg. No.: 201012006
B.E CSE (AI & ML) - V Semester

Under the Supervision of
Mr. JOYJIT GUHA
GUIDE
Academy Of Skill Development
Kolkata

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

FACULTY OF ENGINEERING AND TECHNOLOGY

ANNAMALAI UNIVERSITY

ANNAMALAI NAGAR – 608002

NOVEMBER – 2022

ANNAMALAI UNIVERSITY
FACULTY OF ENGINEERING AND TECHNOLOGY
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

<u>CERTIFICATE</u>

This is to certify that the "**Internship Report"** submitted by **Babu Vijaya Raghavan M**

Reg. No.: 201012006 is the work done by him/ her and submitted during the academic year 2022

– 2023, in partial fulfillment of the requirements for the award of the degree of **BACHELOR**

OF ENGINEERING in COMPUTER SCIENCE AND ENGINEERING(Artificial

Intelligence & Machine Learning), at **Academy of Skill Development, Kolkata.**

Internship Coordinator   **Professor & Head**
Department of CSE   **Department of CSE**

Internal Examiner   **External Examiner**

Place:
Date:

Course Code	INDUSTRIAL TRAINING / RURAL INTERNSHIP/ INNOVATION / ENTREPRENEURSHIP	L	TR	S	C
		0	1	2	4

Four weeks during the summer vacation after the IV semester / VI semester of the programme

COURSE OBJECTIVES:
- To expose the students to understand technical and professional skill requirements in IT industries.
- To impart professional skills for solving problems in industries.
- To train the students to design innovative solutions for a problem.
- To motivate the students to become an Entrepreneur.
- To develop communication and technical report writing skill.

The students will work for two periods per week guided by student counselor. They will be asked to present a seminar of not less than 15 minutes and not more than 30 minutes on any technical topic of student's choice related to Computer Science and Engineering and to engage in discussion with audience. They will defend their presentation. A brief copy of their presentation also should be submitted. Evaluation will be done by the student counselor based on the technical presentation, the report and on the interaction shown during the seminar.

The students will individually undertake a training program in reputed concerns in the field of Computer Science and Engineering during summer vacation for a minimum stipulated period of four weeks. At the end of training the student has to submit the detailed report on the training undertaken within ten days from the commencement of the seventh semester. The student will be evaluated by a team of staff members nominated by the Head ofthe Department through a viva-voce examination.

COURSE OUTCOMES:
At the end of this course, the students will be able to
1. Understand the day-to-day job in IT industries, and technical and professional skills needed for an industry.
2. Develop and refine technical and professional skills through hands-on work experience.
3. Design an innovative solution for an Industry requirement by applying the knowledge learned from industry and in academics.
4. Develop a startup for product or services based on the people or industry requirements.
5. Communicate effectively the knowledge learned in internship through document and PowerPoint presentation.

Mapping of Course Outcomes with Programme Outcomes											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	1	-	-	-	-	-	-	-	-	-	-
CO2	1	2	2	-	-	-	-	-	-	-	-	3
CO3	1	-	2	1	2	-	-	-	-	-	-	-
CO4	1	-	-	-	-	-	-	-	2	-	2	1
CO5	1	-	-	-	2	-	-	-	-	3	-	-

Academy of Skill Development

A non-profit trust registered with Govt. of West Bengal U/S 60 and Rule 69 with registration number -190307248

We dream of self-sufficient India
हम आत्मनिर्भर भारत का सपना देखते हैं

Industrial Internship Certificate

This certificate is awarded to

BABU VIJAYA RAGHAVAN M

of

ANNAMALAI UNIVERSITY FACULTY OF ENGINEERING AND TECHNOLOGY

for successfully completing the Industrial Internship on

DATA SCIENCE, AI, AND MACHINE LEARNING USING PYTHON (PROJECT BASED)

from

23rd July, 2022 to 26th August, 2022 (5 Weeks)

and implementing the project titled

STOCK PRICE PREDICTION

ISO 9001:2015 CERTIFIED

ASD
Skills For Employment

Certificate ID: ASD/DAT/ANN/STO/29007
Issue Date: 26th August, 2022

Head
Technology Services

VERIFIED

Head
Operations

Microsoft
Technology Associate

AUTODESK

UDYOG AADHAR
MSME

ISO

ACKNOWLEDGEMENT

First, I would like to thank Mr. JOYJIT GUHA, GUIDE, **ACADEMY OF SKILL DEVELOPMENT**, **Kolkata** for giving me the opportunity to do an internship within the organization.

I also would like to thank all the people who worked along with me **ACADEMY OF SKILL DEVELOPMENT, Kolkata.** With their patience and openness they created an enjoyable working environment. It is indeed with a great sense of pleasure and immense sense of gratitude that I acknowledge the help of these individuals.

I wish to express my immense gratitude to **Dr. A. Murugappan,** Dean**,** Faculty of Engineering and Technology, Annamalai University, for the facilities provided to accomplish this internship.

I would like to convey my heartfelt thanks to Dr**. S. Palanivel**, Professor and Head, Department of Computer Science and Engineering, for his support, encouragement and blessings throughout my internship.

I also would like to express my thanks to my department staff members and friends who helped me in successful completion of this internship

INTERNSHIP OBJECTIVES

➢ Internships are generally thought of to be reserved for college students looking to gain experience in a particular field. However, a wide array of people can benefit from training internships in order to receive real world experience and develop their skills.

➢ Internships are utilized in a number of different career fields, including architecture, engineering, healthcare, economics, advertising and many more.

➢ Some internships are used to allow individuals to perform scientific research, while others are specifically designed to allow people to gain first-hand experience working.

➢ Utilizing internships is a great way to build your resume and develop skills that canbe emphasized in your resume for future jobs.

➢ Internship learning objectives should be developed along four dimensions as follows:

1. **Skill development:** Learning and improving skills such as writing, verbal communication, research, technology, teamwork, and leadership. It is the development of these skills that often represent the major benefits of an internship.

2. **Understanding Real-World Application:** Understanding the workplace, operating procedures, the department/company and its products, and other organizational concepts. In addition, this would include knowledge added to existing classroom knowledge, such as new applications or new skills.

3. **Career Awareness:** Internships often provide the opportunity to take a peek at what working for a company or in an industry would be like. Objectives could include learning about career positions and occupations along with the qualities and training required to obtain those positions.

4. **Personal Development:** One of the major benefits of an internship is how it helps you to develop self-confidence, assertiveness, and basic work habits.

INDEX

<u>OVERVIEW</u>

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.

HISTORY OF PYTHON

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. Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, Small Talk, UNIX shell, and other scripting languages. Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL). Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

FEATURES OF PYTHON

Easy-to-learn: Python has few Keywords, simple structure and clearly defined syntax. This allows a student to pick up the language quickly.

Easy-to-Read: Python code is more clearly defined and visible to the eyes.

Easy -to-Maintain: Python's source code is fairly easy-to-maintain.

A broad standard library: Python's bulk of the library is very portable and cross platform compatible on UNIX, Windows, and Macintosh.

Interactive Mode: Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.

Portable: Python can run on the wide variety of hardware platforms and has the same interface on all platforms.

Extendable: You can add low level modules to the python interpreter. These modules enables programmers to add to or customize their tools to be more efficient.

Databases: Python provides interfaces to all major commercial databases.

GUI Programming: Python supports GUI applications that can be created and ported to many system calls, libraries, and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.

Scalable: Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below:
- It support 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 datatypes and supports dynamic type checking.
- It supports automatic garbage collections.
- It can be easily integrated with C, C++, COM, ActiveX, CORBA and JAVA.

ENVIRONMENT SETUP

Open a terminal window and type "python" to find out if it is already installed and which version is installed.

- UNIX (Solaris, Linux, FreeBSD, AIX, HP/UX, SunOS, IRIX, etc.)

- Win 9x/NT/2000

- Macintosh (Intel, PPC, 68K)

- OS/2

- DOS (multiple versions)

- PalmOS

- Nokia mobile phones

- Windows CE

- Acorn/RISC OS

BASIC SYNTAX OF PYTHON PROGRAM

Type the following text at the Python prompt and press the Enter –

>>> print "Hello, Python!"

If you are running new version of Python, then you would need to use print statement with parenthesis as in **print ("Hello, Python!");**.
However in Python version 2.4.3, this produces the following result –

Hello, Python!

Python Identifiers

A Python identifier is a name used to identify a variable, function, class, module or other object. An identifier starts with a letter A to Z or a to z or an underscore (_) followed by zero or more letters, underscores and digits (0 to 9).
Python does not allow punctuation characters such as @, $, and % within identifiers. Python is a case sensitive programming language.

Python Keywords

The following list shows the Python keywords. These are reserved words and you cannot use them as constant or variable or any other identifier names. All the Python keywords contain lowercase letters only.

And, exec, not
Assert, finally, or
Break, for, pass
Class, from, print
continue, global, raise
def, if, return
del, import, try
elif, in, while
else, is, with
except, lambda, yield

Lines & Indentation

Python provides no braces to indicate blocks of code for class and function definitions or flow control. Blocks of code are denoted by line indentation, which is rigidly enforced.
The number of spaces in the indentation is variable, but all statements within the block must be indented the same amount. For example –
if True:
print "True"
else:
print "False"

Command Line Arguments

Many programs can be run to provide you with some basic information about how they should be run. Python enables you to do this with -h –

$ python-h
usage: python [option]...[-c cmd|-m mod | file |-][arg]...

Options and arguments (and corresponding environment variables):

-c cmd: program passed in as string(terminates option list)

-d : debug output from parser (also PYTHONDEBUG=x)

-E : ignore environment variables (such as PYTHONPATH)

-h : print this help message and exit [etc.]

VARIABLE TYPES

Variables are nothing but reserved memory locations to store values. This means that when you create a variable you reserve some space in memory.

Assigning Values to Variables

Python variables do not need explicit declaration to reserve memory space. The declaration happens automatically when you assign a value to a variable. The equal sign (=) is used to assign values to variables
.

```
counter=10          # An integer assignment
weight=10.60        # A floating point
name="Ardent"       # A string
```

Multiple Assignment

Python allows you to assign a single value to several variables simultaneously. For example − a = b = c = 1
a,b,c = 1,2,"hello"

Standard Data Types

The data stored in memory can be of many types. For example, a person's age is stored as a numeric value and his or her address is stored as alphanumeric characters. Python has five standard data types −
- String
- List
- Tuple
- Dictionary
- Number

Data Type Conversion

Sometimes, you may need to perform conversions between the built-in types. To convert between types, you simply use the type name as a function.

There are several built-in functions to perform conversion from one data type to another

Sr. No	Function & Description
1	**int(x[,base])** Converts x to integer, base specifies the base if x is a string
2	**long(x[,base])** Converts x to long integer, base specifies the base if x is a string
3	**float(x)** Converts x to a floating-point number
4	**complex(real[,imag])** Creates a complex number
5	**str(x)** Converts object x to a string representation
6	**repr(x)** Converts object x to expression string
7	**eval(str)** Evaluates a string and returns an object
8	**tuple(s)** Converts s into a tuple
9	**list(s)** Converts s into a list

FUNCTIONS

Defining a Function

```
def function_name( parameters ):
            "function_docstring"
            function_suite
            return [expression]
```

Pass by reference vs Pass by value

All parameters (arguments) in the Python language are passed by reference. It means if you change what a parameter refers to within a function, the change also reflects back in the calling function. For example –

 # Function definition is here

```
def changeme(mylist):
        "This changes a passed list into this function"
        mylist.append([1,2,3,4]);
        print"Values inside the function: ",mylist
        return
```

 # Now you can call changeme function

```
mylist=[10,20,30];
changeme(mylist);
print"Values outside the function: ",mylist
```

Here, we are maintaining reference of the passed object and appending values in the same object. So, this would produce the following result –

```
Values inside the function: [10, 20, 30, [1, 2, 3, 4]]
Values outside the function: [10, 20, 30, [1, 2, 3, 4]]
```
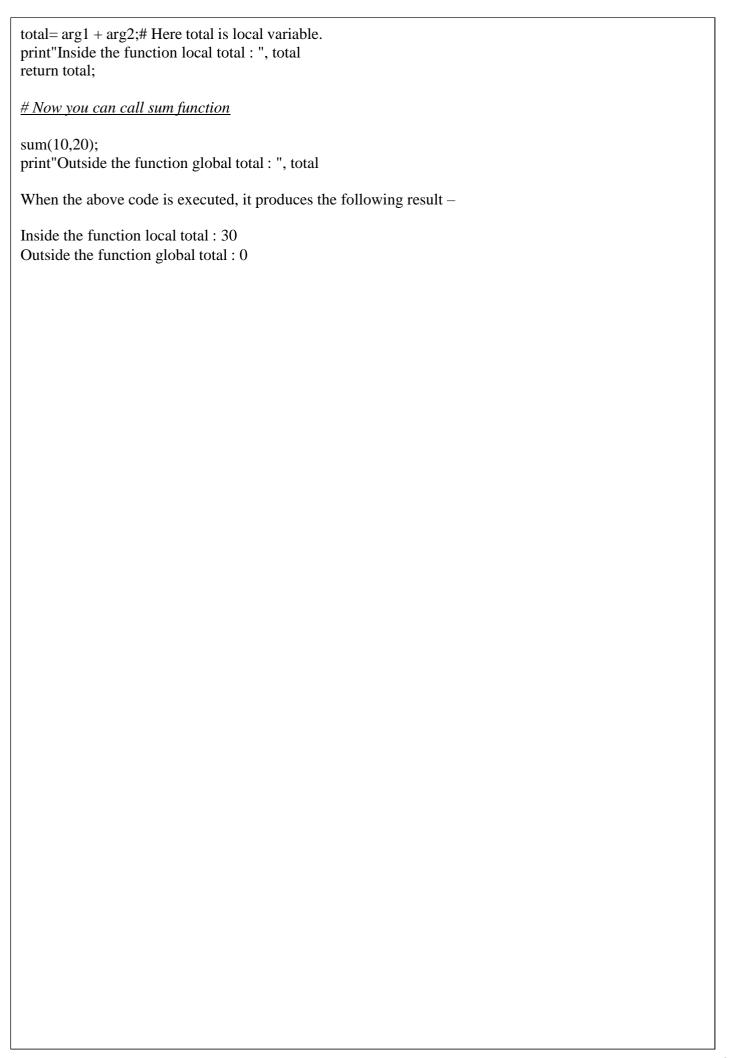
Global vs. Local variables

Variables that are defined inside a function body have a local scope, and those defined outside have a global scope . For Example-
total=0; # This is global variable.

 # Function definition is here

```
def sum( arg1, arg2 ):
```

 # Add both the parameters and return them."

```
total= arg1 + arg2;# Here total is local variable.
print"Inside the function local total : ", total
return total;
```

```
sum(10,20);
print"Outside the function global total : ", total
```

When the above code is executed, it produces the following result –

Inside the function local total : 30
Outside the function global total : 0

MODULES

A module allows you to logically organize your Python code. Grouping related code into a module makes the code easier to understand and use. A module is a Python object with arbitrarily named attributes that you can bind and reference .

The Python code for a module named *aname* normally resides in a file named *aname.py*. Here's an example of a simple module, support.py

```
def print_func( par ):
print"Hello : ", par
return
```

The *import* Statement

You can use any Python source file as a module by executing an import statement in some other Python source file. The *import* has the following syntax –

import module1[, module2[,… moduleN]

PACKAGES

A package is a hierarchical file directory structure that defines a single Python application environment that consists of modules and sub packages and sub-subpackages, and so on.

Consider a file *Pots.py* available in *Phone* directory. This file has following line of source code −
```
def Pots():
    print "I'm Pots Phone"
```

Similar way, we have another two files having different functions with the same name as above −

- *Phone/Isdn.py* file having function Isdn()
- *Phone/G3.py* file having function G3()

Now, create one more file__init__.py in *Phone* directory –

- Phone/__init__.py

To make all of your functions available when you've imported Phone, you need to put explicit import statements in init .py as follows −
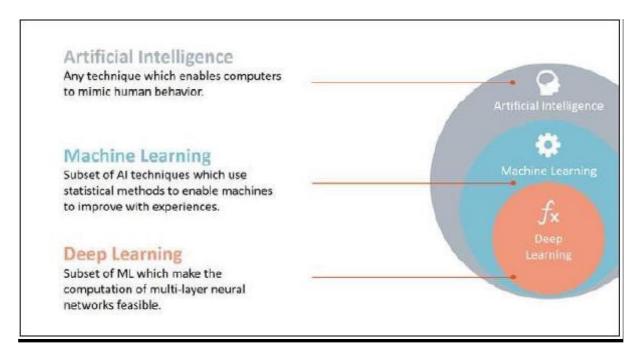```
from Pots import Pots
from Isdn import Isdn
from G3 import
```

# ARTIFICIAL INTELLIGENCE
## Introduction



Artificial Intelligence
Any technique which enables computers to mimic human behavior.

Machine Learning
Subset of AI techniques which use statistical methods to enable machines to improve with experiences.

Deep Learning
Subset of ML which make the computation of multi-layer neural networks feasible.

According to the father of Artificial Intelligence, John McCarthy, it is *"The science and engineering of making intelligent machines, especially intelligent computer programs".*

Artificial Intelligence is a way of **making a computer, a computer-controlled robot, or a software think intelligently**, in the similar manner the intelligent humans think.

AI is accomplished by studying how human brain thinks, and how humans learn, decide, and work while trying to solve a problem, and then using the outcomes of this study as a basis of developing intelligent software and systems.

The development of AI started with the intention of creating similar intelligence in machines that we find and regard high in humans.

## Goals of AI

**To Create Expert Systems** − The systems which exhibit intelligent behaviour, learn, demonstrate, explain, and advice its users.
**To Implement Human Intelligence in Machines** − Creating systems that understand, think, learn, and behave like humans.

# Applications of AI

AI has been dominant in various fields such as :-

**Gaming** − AI plays crucial role in strategic games such as chess, poker, tic-tac-toe, etc., where machine can think of large number of possible positions based on heuristic knowledge.

**Natural Language Processing** − It is possible to interact with the computer that understands natural language spoken by humans.

**Expert Systems** − There are some applications which integrate machine, software, and special information to impart reasoning and advising. They provide explanation and advice to the users.

**Vision Systems** − These systems understand, interpret, and comprehend visual input on the computer.
For example: A spying aeroplane takes photographs, which are used to figure out spatial information or map of the areas.
Doctors use clinical expert system to diagnose the patient.
Police use computer software that can recognize the face of criminal with the stored portrait made by forensic artist.

**Speech Recognition** − Some intelligent systems are capable of hearing and comprehending the language in terms of sentences and their meanings while a human talks to it. It can handle different accents, slang words, noise in the background, change in human's noise due to cold, etc.

**Handwriting Recognition** − The handwriting recognition software reads the text written on paper by a pen or on screen by a stylus. It can recognize the shapes of the letters and convert it into editable text.
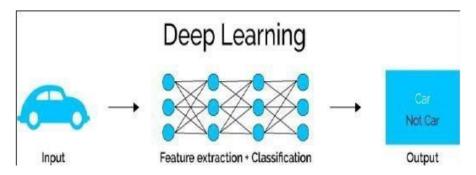
**Intelligent Robots** − Robots are able to perform the tasks given by a human. They have sensors to detect physical data from the real world such as light, heat, temperature, movement, sound, bump, and pressure. They have efficient processors, multiple sensors and huge memory, to exhibit intelligence. In addition, they are capable of learning from their mistakes and they can adapt to the new environment.
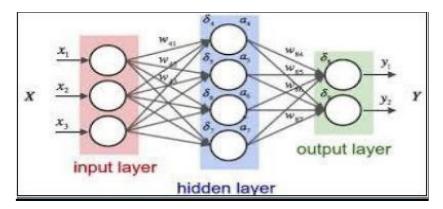
# Application of AI

## Deep Learning

Deep learning is a subset of machine learning. Usually, when people use the term deep learning, they are referring to deep artificial neural networks, and somewhat less frequently to deep reinforcement learning. Deep learning is a class of machine learning algorithms that:

- use a cascade of multiple layers of nonlinear processing units for feature extraction and transformation. Each successive layer uses the output from the previous layer as input.
- learn in supervised (e.g., classification) and/or unsupervised (e.g., pattern analysis) manners.
- learn multiple levels of representations that correspond to different levels of abstraction; the levels form a hierarchy of concepts.
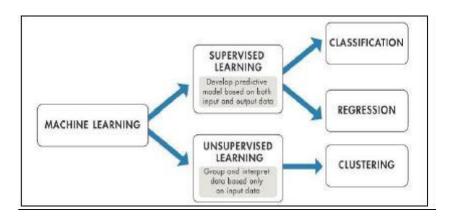- use some form of gradient descent for training via backpropagation.



### NEURAL NETWORKING

**Artificial neural networks** (**ANNs**) or **connectionist systems** are computing systems inspired by the biological neural networks that constitute animal brains. Such systems learn (progressively improve performance on) tasks by considering examples, generally without task-specific programming



An ANN is based on a collection of connected units or nodes called artificial neurons (analogous to biological neurons in an animal brain). Each connection between artificial neurons can transmit a signal from one to another.

# MACHINE LEARNING



Machine learning is a field of computer science that gives computers the ability to learn without being explicitly programmed.

Evolved from the study of pattern recognition and computational learning theory in artificial intelligence, machine learning explores the study and construction of algorithms that can learn from and make predictions on data.

# INTRODUCTION TO MACHINE LEARNING

**Machine learning** is a field of computer science that gives computers the ability to learn without being explicitly programmed.

**Arthur Samuel**, an American pioneer in the field of computer gaming and artificial intelligence, coined the term "Machine Learning" in 1959 while at IBM. Evolved from the study of pattern recognition and computational learning theory in artificial intelligence, machine learning explores the study and construction of algorithms that can learn from and make predictions on data.

Machine learning tasks are typically classified into two broad categories, depending on whether there is a learning "signal" or "feedback" available to a learning system:-

## SUPERVISED LEARNING

**Supervised learning** is the machine learning task of inferring a function from *labeled training data*.[1] The training data consist of a set of *training examples*. In supervised learning, each example is a *pair* consisting of an input object (typically a vector) and a desired output value.

A supervised learning algorithm analyses the training data and produces an inferred function, which can be used for mapping new examples. An optimal scenario will allow for the algorithm to correctly determine the class labels for unseen instances. This requires the learning algorithm to generalize from thetraining data to unseen situations in a "reasonable" way.

## UNSUPERVISED LEARNING

**Unsupervised learning** is the machine learning task of inferring a function to describe hidden structure from "unlabelled" data (a classification or categorization is not included in the observations). Since the examples given to the learner are unlabelled, there is no evaluation of the accuracy of the structure that is output by the relevant algorithm—which is one way of distinguishing unsupervised learning from supervised learning and reinforcement learning.

A central case of unsupervised learning is the problem of density estimation in statistics, though unsupervised learning encompasses many other problems (and solutions) involving summarizing and explaining key features of the data.

## NUMPY

NumPy is a library for the Python programming language, adding support for large, multi- dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. The ancestor of NumPy, Numeric, was originally created by Jim Hugunin.

NumPy targets the CPython reference implementation of Python, which is a non-optimizing bytecode interpreter. Mathematical algorithms written for this version of Python often run much slower than compiled equivalents.

Using NumPy in Python gives functionality comparable to MATLAB since they are both interpreted, and they both allow the user to write fast programs as long as most operations work on arrays or matrices instead of scalars.

# NUMPY ARRAY

NumPy's main object is the homogeneous multidimensional array. It is a table of elements (usually numbers), all of the same type, indexed by a tuple of positive integers. In NumPy dimensions are called *axes*. The number of axes is *rank*.

For example, the coordinates of a point in 3D space [1, 2, 1] is an array of rank 1, because it has one axis. That axis has a length of 3. In the example pictured below, the array has rank 2 (it is 2- dimensional). The first dimension (axis) has a length of 2, the second dimension has a length of 3.

[[ 1., 0., 0.],
 [ 0., 1., 2.]]

NumPy's array class is called *ndarray*. It is also known by the alias.

# SLICING NUMPY ARRAY

**import numpy as np**
**a = np.array([[1,2,3],[3,4,5],[4,5,6]])**

```
print 'Our array is:'
print a
print '\n'
print 'The items in the second column are:'
print a[...,1]
print '\n'
print 'The items in the second row are:'
print a[1,...]
print '\n'
print 'The items column 1 onwards are:'
print a[...,1:]
```

OUTPUT
Our array is:
[[1 2 3]
[3 4 5]
[4 5 6]]
The items in the second column are: [2 4 5]
The items in the second row are: [3 4 5]
The items column 1 onwards are: [[2 3]
                                 [4 5]
                                 [5 6]]

SCIPY

Modules for optimization, linear algebra, integration, interpolation, special functions, FFT, signal and image processing, ODE solvers and other tasks common in science and engineering.

SciPy builds on the NumPy array object and is part of the NumPy stack which includes tools like Matplotlib, pandas and SymPy, and an expanding set of scientific computing libraries. This NumPy stack has similar users to other applications such as MATLAB, GNU Octave, and Scilab. The NumPy stack is also sometimes referred to as the SciPy stack.

The SciPy Library/Package

The SciPy package of key algorithms and functions core to Python's scientific computing capabilities. Available sub-packages include:

- **constants:** physical constants and conversion factors (since version 0.7.0)
- **cluster:** hierarchical clustering, vector quantization, K-means
- **fftpack:** Discrete Fourier Transform algorithms
- **integrate:** numerical integration routines
- **interpolate:** interpolation tools
- **io:** data input and output
- **lib:** Python wrappers to external libraries
- **linalg:** linear algebra routines
- **misc:** miscellaneous utilities (e.g. image reading/writing)
- **ndimage:** various functions for multi-dimensional image processing
- **optimize:** optimization algorithms including linear programming
- **signal:** signal processing tools
- **sparse:** sparse matrix and related algorithms
- **spatial:** KD-trees, nearest neighbours, distance functions
- **special:** special functions
- **stats:** statistical functions
- **weave:** tool for writing C / C++ code as Python multiline strings
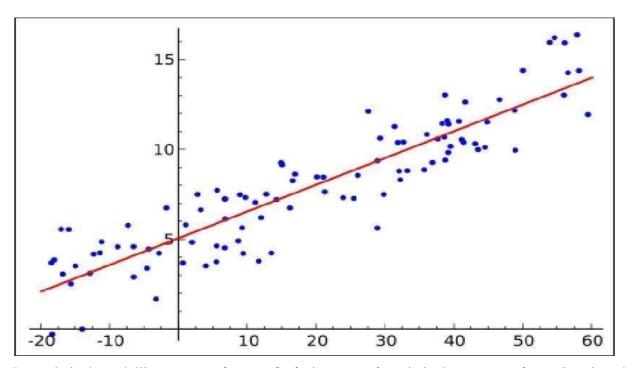
Data Structures

The basic data structure used by SciPy is a multidimensional array provided by the NumPy module. NumPy provides some functions for linear algebra, Fourier transforms and random number generation, but not with the generality of the equivalent functions in SciPy. NumPy can also be used as an efficient multi-dimensional container of data with arbitrary data-types. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases. Older versions of SciPy used Numeric as an array type, which is now deprecated in favour of the newer NumPy array code.

SCIKIT-LEARN

Scikit-learn is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, *k*-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

The scikit-learn project started as scikits.learn, a Google Summer of Code project by David Cournapeau. Its name stems from the notion that it is a "SciKit" (SciPy Toolkit), a separately-developed and distributed third-party extension to SciPy.[4] The original codebase was later rewritten by other developers. In 2010 Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort and Vincent Michel, all from INRIA took leadership of the project and made the first public release on February the 1st 2010[5]. Of the various scikits, scikit-learn as well as scikit-image weredescribed as "well- maintained and popular" in November 2012.

REGRESSION ANALYSIS

In statistical modelling, **regression analysis** is a set of statistical processes for estimating the relationships among variables. It includes many techniques for modelling and analysing several variables, when the focus is on the relationship between a dependent variable and one or more independent variables (or 'predictors'). More specifically, regression analysis helps one understand how the typical value of the dependent variable (or 'criterion variable') changes whenany one of the independent variables is varied, while the other independent variables are held fixed.

Regression analysis is widely used for prediction and forecasting, where its use has substantial overlap with the field of machine learning. Regression analysis is also used to understand which among the independent variables are related to the dependent variable, and to explore the forms of these relationships. In restricted circumstances, regression analysis can be used to infer casual relationships between the independent and dependent variables. However this can lead to illusions or false relationships, so caution is advisable

LINEAR REGRESSION

Linear regression is a linear approach for modelling the relationship between a scalar dependent variable y and one or more explanatory variables (or independent variables) denoted X. The case of one explanatory variable is called *simple linear regression*. For more than one explanatory variable, the process is called *multiple linear regression*. In linear regression, the relationships are modelled using linear predictor functions whose unknown model parameters are estimated from the data. Such models are called *linear models*.

LOGISTIC REGRESSION

Logistic regression, or logit regression, or logit model [1] is a regression model where the dependent variable (DV) is categorical. This article covers the case of a binary dependent variable—that is, where theoutput can take only two values, "0" and "1", which represent outcomes such as pass/fail, win/lose, alive/dead or healthy/sick. Cases where the dependent variable has more than two outcome categories maybe analysed in multinomial logistic regression, or, if the multiple categories are ordered, in ordinal logisticregression. In the terminology of economics, logistic regression is an example of a qualitative response/discrete choice model.

POLYNOMIAL REGRESSION

Polynomial regression is a form of regression analysis in which the relationship between the independent variable x and the dependent variable y is modelled as an nth degree polynomial in x.
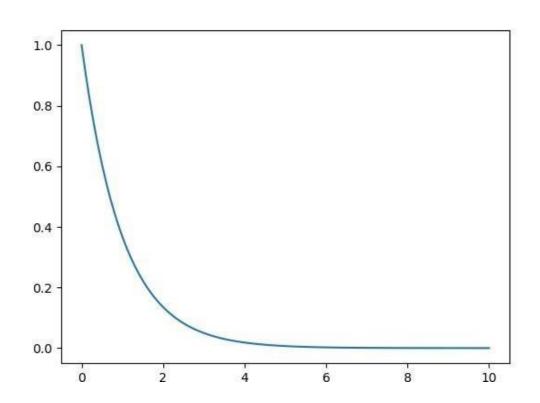Polynomial regression fits a nonlinear relationship between the value of x and the corresponding conditional mean of y , denoted $E(y \mid x)$, and has been used to describe nonlinear phenomena such as the growth rate of tissues, the distribution of carbon isotopes in lake sediments, and the progression of diseaseepidemics. Although *polynomial regression* fits a nonlinear model to the data, as a statistical estimation problem it is linear, in the sense that the regression function $E(y \mid x)$ is linear in the unknown parameters that are estimated from the data.

MATPLOTLIB

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter,wxPython, Qt, or GTK+. There is also a procedural "pylab" interface based on a state machine (like OpenGL), designed to closely resemble that of MATLAB, though its use is discouraged .SciPy makes use of matplotlib.

EXAMPLE
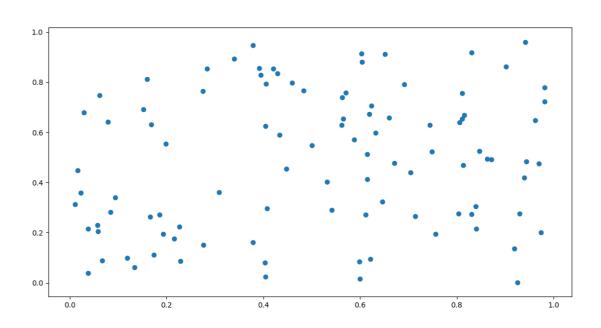
- **LINE PLOT**
  ```python
  import matplotlib.pyplot as plt
  import numpy as np
  a=np.linspace(0,10,100)
  b=np.exp(-a)
  plt.plot(a,b)
  plt.show()
  ```

- **SCATTER PLOT**
  - ➜ **import** matplotlib.pyplot **as** plt
  - ➜ **from** numpy.random **import** rand
  - ➜ a=rand(100)
  - ➜ b=rand(100)
  - ➜ plt.scatter(a,b)
  - ➜ plt.show()

PANDAS

In computer programming, **pandas** is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license. "Panel data", an econometrics term for multidimensional, structured data sets.

LIBRARY FEATURES
- Data Frame object for data manipulation with integrated indexing.
- Tools for reading and writing data between in-memory data structures and different file formats.
- Data alignment and integrated handling of missing data.
- Reshaping and pivoting of data sets.
- Label-based slicing, fancy indexing, and sub setting of large data sets.
- Data structure column insertion and deletion.
- Group by engine allowing split-apply-combine operations on data sets.
- Data set merging and joining.
- Hierarchical axis indexing to work with high-dimensional data in a lower-dimensional data structure.
- Time series-functionality: Date range generation.

CLUSTERING

Cluster analysis or **clustering** is the task of grouping a set of objects in such a way that objects in the same group (called a **cluster**) are more similar (in some sense or another) to each other than to those in other groups (clusters). It is a main task of exploratory data mining, and a common technique for statistical data analysis, used in many fields, including machine learning, pattern recognition, image analysis, information retrieval, bioinformatics, data compression, and computer graphics.

Cluster analysis itself is not one specific algorithm, but the general task to be solved. It can be achieved by various algorithms that differ significantly in their notion of what constitutes a cluster and how to efficiently find them. Popular notions of clusters include groups with small distances among the cluster members, dense areas of the data space, intervals or particular statistical distributions.

Clustering can therefore be formulated as a multi-objective optimization problem. The appropriate clustering algorithm and parameter settings (including values such as the distance function to use, a density threshold or the number of expected clusters) depend on the individual data set and intended useof the results. Cluster analysis as such is not an automatic task, but an iterative process of knowledge discovery or interactive multi-objective optimization that involves trial and failure. It is often necessary to modify data pre-processing and model parameters until the result achieves the desired properties.

ALGORITHM

- ➢ Data Collection

- ➢ Data Formatting

- ➢ Model Selection

- ➢ Training

- ➢ Testing

Data Collection: We have collected data sets of Uniqlo Stock Market(2012-2016) from online website(www.kaggle.com). We have downloaded the .csv files in which information was present.
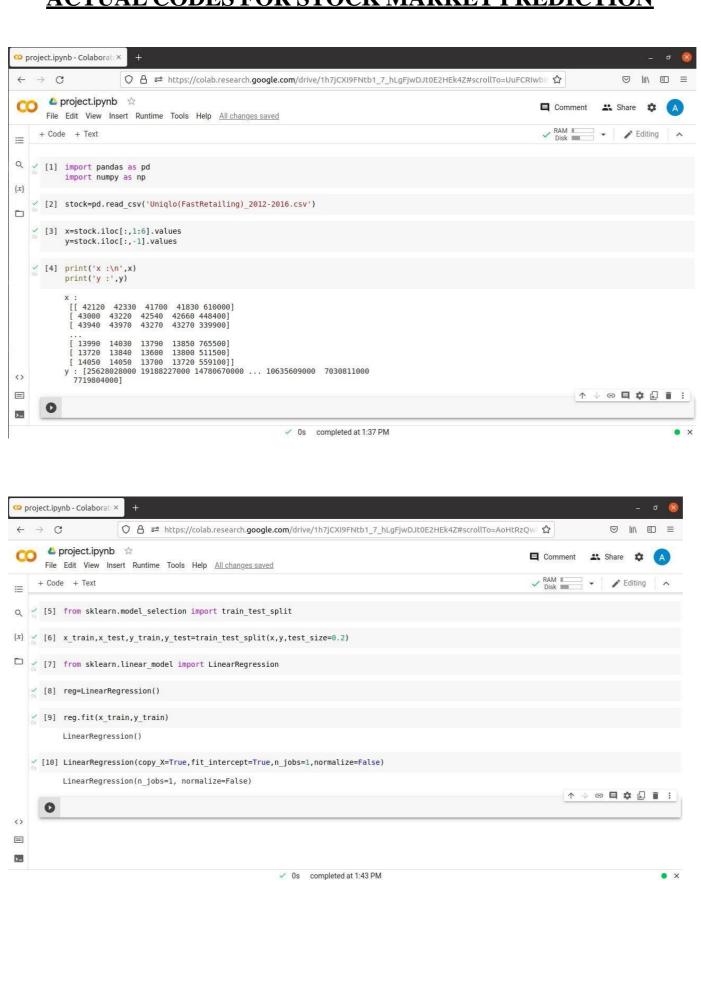
Data Formatting: The collected data is formatted into suitable data sets. We check the collinearity with mean stock value. The data sets which have collinearity nearer to 1.0 has been selected.
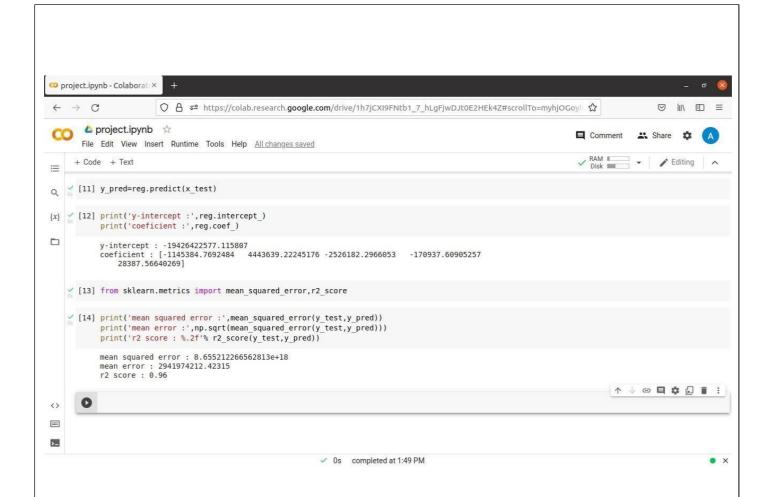
Model Selection: We have selected different models to minimize the error of the predicted value. The different models used are Linear Regression Linear Model, Ridge Linear model, Lasso Linear Model and Bayesian Ridge Linear Model.

Training: The data sets was divided such that x_train is used to train the model with corresponding x_test values and some y_train kept reserved for testing.

Testing: The model was tested with y_train and stored in y_predict . Both y_train and y_predict was compared.

ACTUAL CODES FOR STOCK MARKET PREDICTION


```
[1] import pandas as pd
    import numpy as np

[2] stock=pd.read_csv('Uniqlo(FastRetailing)_2012-2016.csv')

[3] x=stock.iloc[:,1:6].values
    y=stock.iloc[:,-1].values

[4] print('x :\n',x)
    print('y :',y)

    x :
    [[ 42120  42330  41700  41830 610000]
     [ 43000  43220  42540  42660 448400]
     [ 43940  43970  43270  43270 339900]
     ...
     [ 13990  14030  13790  13850 765500]
     [ 13720  13840  13600  13800 511500]
     [ 14050  14050  13700  13720 559100]]
    y : [25628028000 19188227000 14780670000 ... 10635609000  7030811000
      7719804000]
```

✓ 0s   completed at 1:37 PM


```
[5] from sklearn.model_selection import train_test_split

[6] x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2)

[7] from sklearn.linear_model import LinearRegression

[8] reg=LinearRegression()

[9] reg.fit(x_train,y_train)

    LinearRegression()

[10] LinearRegression(copy_X=True,fit_intercept=True,n_jobs=1,normalize=False)

    LinearRegression(n_jobs=1, normalize=False)
```

✓ 0s   completed at 1:43 PM

```
[11] y_pred=reg.predict(x_test)

[12] print('y-intercept :',reg.intercept_)
     print('coeficient :',reg.coef_)

     y-intercept : -19426422577.115807
     coeficient : [-1145384.7692484   4443639.22245176 -2526182.2966053   -170937.60905257
         28387.56640269]

[13] from sklearn.metrics import mean_squared_error,r2_score

[14] print('mean squared error :',mean_squared_error(y_test,y_pred))
     print('mean error :',np.sqrt(mean_squared_error(y_test,y_pred)))
     print('r2 score : %.2f'% r2_score(y_test,y_pred))

     mean squared error : 8.655212266562813e+18
     mean error : 2941974212.42315
     r2 score : 0.96
```

<u>CONCLUSION</u>

We have collected the raw data from online sources. Then we take this raw data and format it.

Now we have selected few models for error detection. We have used four models namely linear regression model, linear ridge model, linear lasso model and Linear BayesianRidge model.

Linear Regression model is good one as the prediction accuracy was 96%.

FUTURE SCOPE

The data taken was limited. The project could be extended to more number of columns. The stock trading was predicted only by taking high,open,close,low and volume.

The error can be minimised as well using other algorithms.

THANK YOU