A BRIEF ANALYSIS ON INDIAN LITERACY

A

Thesis Submitted

in the partial fulfilment of the requirementsfor the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE & ENGINEERING

BY

SOFIA TARANNUM	**19281A0590**
ARAVAPELLY SHEETHAL	**19281A05B0**
KAYITHA CHISHWANA	**19281A0592**
SARDAR HANSRAJ SINGH	**19281A05B5**

PROJECT GUIDE

Mrs. FATHIMA ANJUM

Assistant Professor, CSE

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

KAMALA INSTITUTE OF TECHNOLOGY & SCIENCE

(Approved by AICTE, New Delhi, Affiliated to JNTU, Hyderabad,
Accredited with B++’Grade by NAAC, Accredited with NBA)
Singapur, Huzurabad, Karimnagar, Telangana- 505468

(2022-2023)

KAMALA INSTITUTE OF TECHNOLOGY & SCIENCE

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

(Approved by AICTE, New Delhi, Affiliated to JNTU, Hyderabad,
Accredited with B++'Grade by NAAC, Accredited with NBA)
Singapur,Huzurabad,Karimnagar,Telangana-505468
(2022-2023)

CERTIFICATE

This is to certify that SOFIA TARANNUM (19281A0590), ARAVAPELLY SHEETHAL (19281A05B0), KAYITHA CHISHWANA (19281A0592), SARDAR HANSRAJ SINGH (19281A05B5) of the B. Tech (CSE) has satisfactorily completed the dissertation work for Mini Project Work entitled **"A BRIEF ANALYSIS ON INDIAN LITERACY"** towards the partial fulfilment of B. Tech degree in the academic year 2022-2023.

Project Guide **Head of the Department**

Principal **External Examiner**

INDEX

ACKNOWLEDGEMENTS

The success of any course depends mainly on the teachers who teach us. Only good teaching can interpret the syllabus and procedure desirable changes and competent citizens. This project was a team effort and many people whose names do not appear on the cover also deserve credit. First, wethank God almighty for his manifold mercies in carrying out our project successfully.

We would like to pay our respects to our principal **Dr. K. SHANKER** and management for providing all the facilities required for completing this project work.

We sincerely thank our Head of the Department and Professor of Computer Science & Engineering **Dr. Capt. RAVINDRA BABU KALLAM**, for encouraging us in doing this project and for his guidance.

We would like to thank our project Co-ordinator **Mr. K. BHASKAR PRAKASH,** Assistant Professor in CSE for the valuable guidance and constant encouragement at each stage of this project. We would like to thank our project guide **Mrs. FATHIMA ANJUM** and Assistant Professor in CSE for the valuable guidance and constant encouragement at each stage of this project.

We thank other teaching and other non-teaching staff of Kamala Institute of Technology & Science, for supporting us in every stage of our Minor Project, entitled **"A BRIEF ANALYSIS ON INDIAN LITERACY ".**

We thank our Parents and Friends for their moral support throughout the project work that helped to strengthen our will.

SOFIA TARANNUM	**19281A0590**
ARAVAPELLY SHEETHAL	**19281A05B0**
KAYITHA CHISHWANA	**19281A0592**
SARDAR HANSRAJ SINGH	**19281A05B5**

ABSTRACT

Learning has been a significant emerging field for several decades since it is a great determinant of the world's civilization and evolution. In general , improving the existing learning activities has a great influence on the global literacy rates.

Literacy must be viewed in the context of its immense potential for bringing about transformation in quality of human life. Development in educational attainment means an increase in literacy level. The main objective of this project is to analyse the Indian Literacy rate state-wise using the machine learning algorithms. It helps us to find the literacy rate for the given year. We are also predicting theliteracy rate of coming years and different graphs are also presented depicting the state-wise literacyrate. It reduces the wastage of manual time in visiting the educational institutions and manually noting them. Therefore, this project is an easy approach to find the literacy rates, through which wecan eliminate risks of manual work and time.

LIST OF FIGURES

LIST OF TABLES

TITLE **PAGE NO.**

CHAPTER 1

INTRODUCTION

1.1 ABOUT THE PROJECT

The literacy rate in India in 2001 and 2011 is examined in this research. An important factor in determining a country's degree of development is literacy. The percentage of adults over the ageof fifteen who are literate is known as the literacy rate. Some emerging nations are attempting to raise the literacy rate, including Bangladesh, Nepal, Laos, and India. In the past ten years, India's literacy rate has increased significantly. India still has lower levels of literacy than manyother nations, though. The literacy rate is 77.70%, with literate males at 84.70% and literate females at 70.30%, according to the National Family Health Survey (NFHS-5) and National Statistical Office: NSO (2021 and 2022). Women appear to have a low literacy rate despite the high percentage of males that are literate. Literacy rate of India is analyzed and predicted using the machine learning algorithms. We have used Linear Regression and Random forest algorithm for the prediction and analysis of the literacy rate. First, we have predicted the population for the year given as input and then predicted population is given as input for predicting the literacy rate for the given year. Apart from predicting the literacy rate of the coming years different graphs are also presented depicting the state-wise literacy rate.

1.2 EXISTING SYSTEM WITH DRAWBACKS

At present, Literacy rate is calculated manually by the government officials which is very burdensome. We also can’t guarantee the prediction made.

Drawbacks:

- Maintaining the records manually is very difficult and it consumes some place for storingthe records alone for many years.
- Contacting the institutions is too difficult as they could be away from the city.

1.3 PROPOSED SYSTEM WITH FEATURES:

We have created a regression model for analysis and prediction, so that we don’t have to make predictionsmanually. We can adapt this model for other purposes just by making little changes.

The experimental results show that the performance of the proposed model was better than the other machine learning models by attaining accuracy of 88.55% and precision of 88.21%.

Features:

- Easy access to the Indian literacy rates.
- Decreases in manual work.
- No need to maintain records manually.

CHAPTER 2

ANALYSIS

The goal of system analysis is to determine where the problem is in an attempt to fix the system. This step involves breaking down the system in different pieces to analyse the situation, analysing project goals, breaking down what needs to be created and attempting to engage users so that definite requirements can be defined.

2.1 HARDWARE AND SOFTWARE REQUIREMENTS

2.1.1 Hardware Requirements (Preferable):

The selection of hardware is very important in the existence and proper working of anysoftware. In the selection of hardware, the size and the capacity requirements are also important.

PROCESSOR	**I3 OR ABOVE**
RAM CAPACITY	**1GB**
KEYBOARD	**NORMAL**
PROCESSORSPEED	**2.40GHZ**

2.1.2 Software Requirements (Preferable):

One of the most difficult tasks is the selection of the software, once a system requirement isknown that is determining whether a particular software package fits the requirements**.**

PROGRAMMINGLANGUAGE	**PYTHON**
VERSION	**PYTHON3.6**
OPERATINGSYSTEM	**WINDOWS/UBUNTU**
TOOLS USED	**JUPYTER NOTEBOOK**

2.2 FUNCTIONAL REQUIREMENTS AND NON-FUNCTIONAL REQUIREMENTS

Functional Requirements:

The following are the functional requirements of our project:

- A training dataset has to be created on which training is performed.
- A testing dataset has to be created on which testing is performed.

Non-Functional Requirements:

- **Maintainability:** Maintainability is used to make future maintenance easier, meet newrequirements.
- **Robustness:** Robustness is the quality of being able to withstand stress, pressures orchanges in procedure or circumstance.
- **Reliability:** Reliability is an ability of a person or system to perform and maintain itsfunctions in circumstances.
- **Size:** The size of a particular application plays a major role, if the size is less thenefficiency will be high.
- **Speed:** If the speed is high then it is good. Since the number of lines in our code isless, hence the speed is high.

Module Description

For predicting the literacy rate of India, our project has been divided into followingmodules:

1. Data Analysis & Pre-processing
2. Model Training & Testing
3. Accuracy Measures
4. Prediction & Visualization

1. Data Analysis & Pre-processing

Data Analysis is done by collecting raw data from different literacy websites.Data pre- processing technique involves transforming raw data into an understandable format. Real- world data is often incomplete, inconsistent, and/or lacking in certain behaviours or trends, and is likely to contain many errors. Data pre-processing is a proven method of resolving suchissues. We use pandas module for Data Analysis and pre-processing real and predicted literacy rate on a graph. Then calculate the accuracy. We use Sklearn andNumPy python module for Training and testing.

Sklearn:

It features various classification, regression and clustering algorithms including support vectormachines, rando **Pandas:**

In order to be able to work with the data in Python, we'll need to read the csv file into PandasDataframe. A Data Frame is a way to represent and work with tabular data. Tabular data has rows and columns, just like our csv file

2. Model Training & Testing

For Literacy rate prediction, we perform “converting into 2D array” and “scaling using normalization” operations on data for further processing. We use fit transform to centre the data in a way that it has 0 mean and 1 standard error. Then, we divide the data into x_train and y_train. Our model will get the 0-th element from x_train and try to predict the 0-th element from y_train. Finally, we reshape the x_train data to match the requirements for training usingkeras. Now we need to train our model using the above data.

For literacy rate prediction:

The algorithm that we have used is Linear Regression

Linear Regression:

Linear Regression is a machine learning algorithm based on supervised learning. It is a statistical approach for modelling relationship between a dependent variable with a given set of independent variables. Here we refer dependent variables as response and independent variables as features for simplicity.

Simple Linear Regression is an approach for predicting a response using a single feature. It isassumed that the two variables are linearly related. Hence, we try to find a linear function thatpredicts the response value(y) as accurately as possible as a function of the feature or independent variable(x). For predicting the literacy rate of any given year, first we need predict the population for thatyear. Then the predicted population is given as input to the model which predict literacy rate . For the algorithm which predict population, year is taken as independent variable. And the predicted population is taken as independent variable for the literacy prediction algorithm.

Testing:

In testing, now we predict the data. Here we have 2 steps: predict the literacy rate and plot it to compare with the real results.Using fit transform to scale the data and then reshape it for the prediction. Predict the data and rescale the predicted data to match its real values. Then plot m forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate withthe Python numerical and scientific libraries NumPy.

NumPy:

NumPy is the core library for scientific computing in Python. It provides a high-performancemultidimensional array object, and tools for working with these arrays.It is used for NumericalCalculations

3. Accuracy Measures

The Accuracy of the model is to be evaluated to figure out the correctness of the prediction. The proposed model got 87% Accuracy.

4. Prediction & Visualization

Using the Proposed model prediction is made for coming years. Graphs are used to visualizestate wise literacy rate predictions. We use Matplotlib python module for Visualization. Here to analyze the literary rate, we have used Random Forest Random Forest Algorithm. Algorithm Random Forest Algorithm is a collection of multiple decision trees and is not sensitive to data.

Random Forest Algorithm

Random forest Algorithm is a combination of decision trees and is not sensitive to data Random forest is a combination of Bootstrapping and Aggregation. The process that we used to create new data from the existing dataset is called bootstrapping and the data is called bootstrapped data.

The process of combining the results from multiple models is called aggregation. Bootstrapping ensures that we are not using the same data for every tree and makes sure that it becomes less sensitive to data. For Classification model, the Random. Forest algorithm takes the majority voting classifier and for regression model, it goes for the average of the predictions. In this project, ours is a regression model, so we go for the average of the predictions.

Matplotlib:

It 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.[3] SciPy makes use of Matplotlib.

CHAPTER 3

DESIGN

3.1 BLOCK DIAGRAM:

The block diagram is typically used for a higher level, less detailed description aimed more at

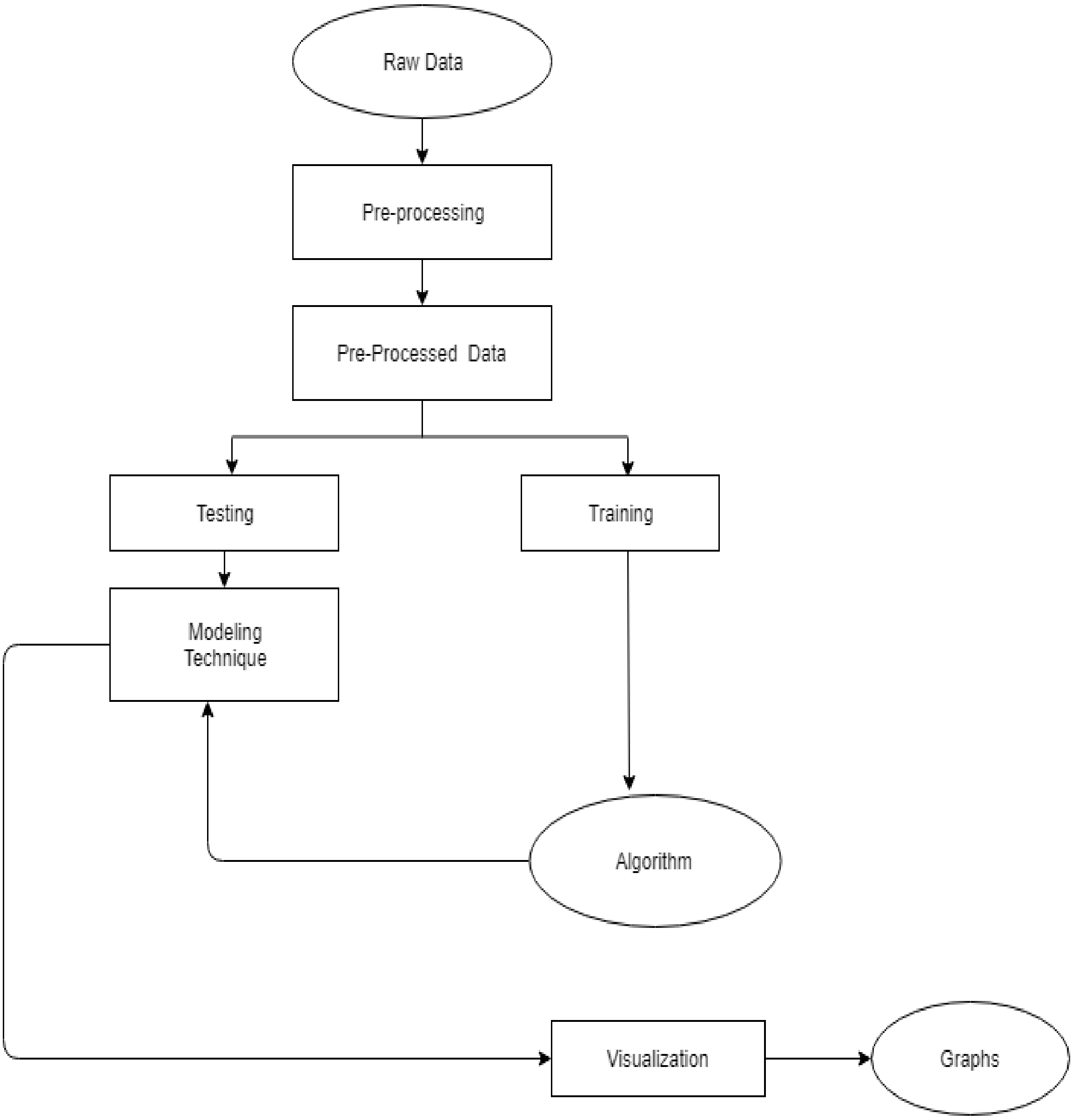


Figure 3.1.1 Block Diagram for Indian literacy Rate Prediction

3.2 DATA FLOW DIAGRAMS:

Data flow diagram (DFD) is a graphical representation of the “flow” of data through an information system, modelling its process concepts. Often, they are a preliminary step used to create an overview of the system which can later be elaborated. DFD’s can also be used for the visualization of data processing (structured design).

A DFD shows what kinds of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It doesn’t show information about timing of processes, or information about whether processes will operate in sequence or parallel. A DFD is also called as “bubble chart”.

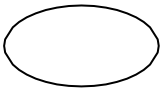
DFD Symbols:

In the DFD, there are four symbols:

- A square defines a source or destination of system data.
- An arrow indicates dataflow. It is the pipeline through which the information flows.
- A circle or a bubble represents transforms dataflow into outgoing dataflow.
- An open rectangle is a store, data at reset or at temporary repository of data.

Dataflow: Data moves in a specific direction from an origin to a destination.

Process: People, procedures or devices that use or produce (Transform) data. The physical component is not identified.



Sources: External sources or destination of data, which may be programs, organizations or otherentity.

Data store: Here data is stored or referenced by a process in the system's #

In our project, we had built the data flow diagrams at the very beginning of business processmodelling in order to model the functions that our project has to carry out and the interaction between those functions together with focusing on data exchanges between processes.

Context level DFD:

A Context level Data flow diagram created using select structured systems analysis and design method (SSADM). This level shows the overall context of the system and its operating environment and showsthe whole system as just one process. It does not usually show data stores, unless they are "owned" byexternal systems, e.g. are accessed by but not maintained by this system, however, these are often shown as external entities. The Context level DFD is shown in fig.3.2.1

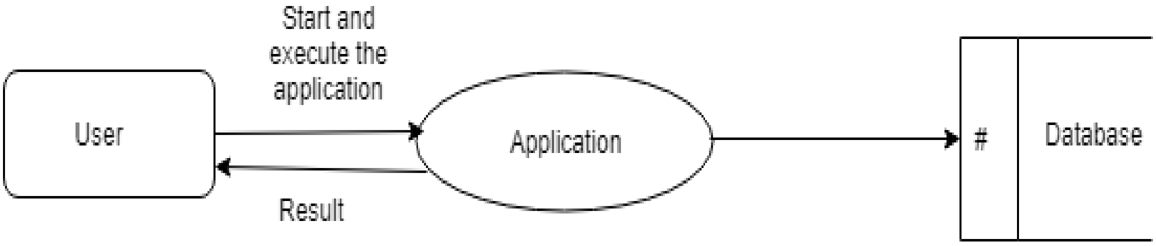


Figure 3.2.1 Context Level DFD for Indian Literacy rate prediction

The Context Level Data Flow Diagram shows the data flow from the application to the databaseand to the system.

Top level DFD:

A data flow diagram is that which can be used to indicate the clear progress of a business venture. In the process of coming up with a data flow diagram, the level one provides an overview of the major functional areas of the undertaking. After presenting the values for most important fields of discussion, it gives room for level two to be drawn.

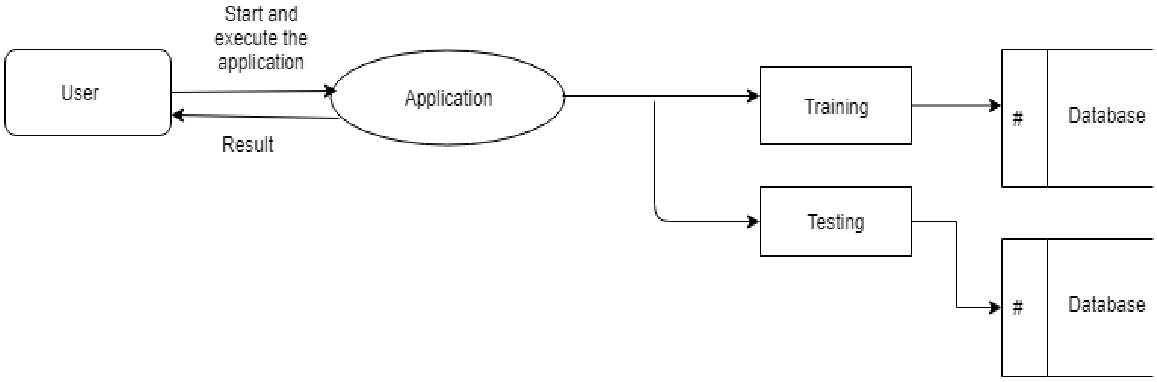


Figure 3.2.2 Top Level DFD for Indian literacy Rate Prediction

After starting and executing the application, training and testing the dataset can be done as shownin the above figure

Detailed Level Diagram

This level explains each process of the system in a detailed manner. In first detailed level DFD (Generation of individual fields): how data flows through individual process/fields in it are shown. In second detailed level DFD (generation of detailed process of the individual fields): how data flows through the system to form a detailed description of the individual processes

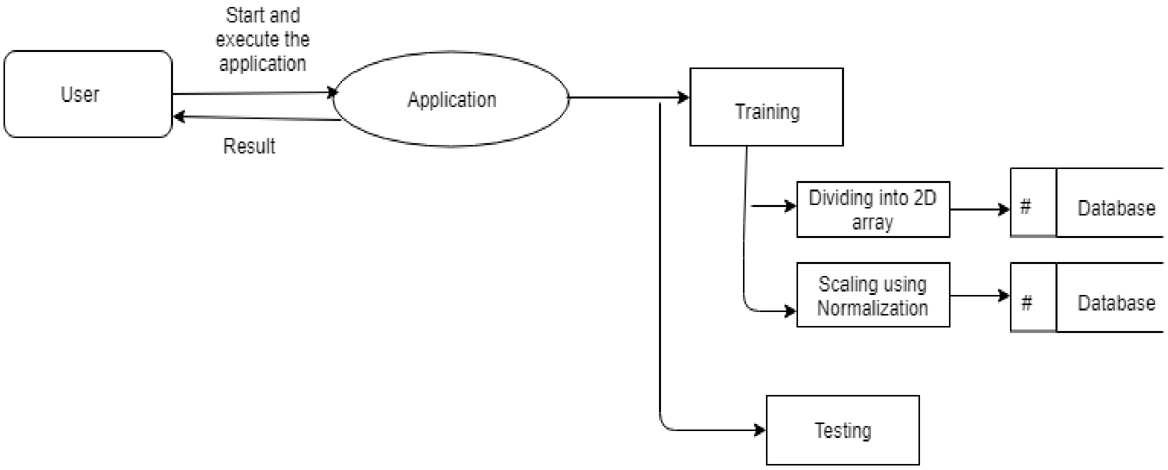


Figure 3.2.3.1 Detailed level DFD for Indian literacy rate prediction

After starting and executing the application, training the dataset is done by using dividing into 2D array and scaling using normalization algorithms, and then testing is done

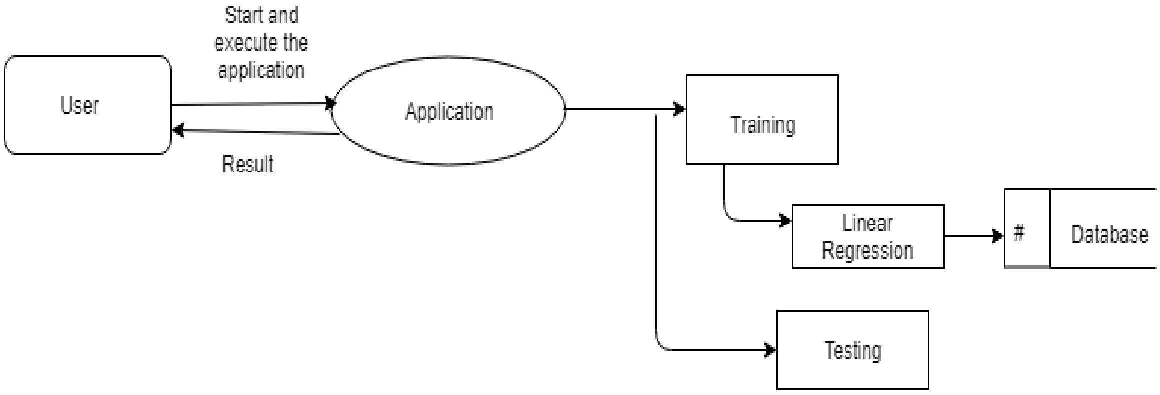


Figure 3.2.3.2 Detailed level dfd for Indian literacy Rate Prediction

After starting and executing the application, training the dataset is done by using linear regressionand then testing is done.

3.3 UNIFIED MODELLING LANGUAGE DIAGRAMS:

The Unified Modelling Language (UML) is a Standard language for specifying, visualizing, constructing and documenting the software system and its components. The UML focuses on theconceptual and physical representation of the system. It captures the decisions and understandingsabout systems that must be constructed. A UML system is represented using five different views that describe the system from a distinctly different perspective. Each view is defined by a set of diagrams, which are as follows.

- **User Model View**

i. This view represents the system from the user’s perspective.

ii. The analysis representation describes a usage scenario from the end-users perspective.

- **Structural Model View**

i. In this model the data and functionality are arrived from inside the system.

ii. This model view models the static structures

- **Behavioural Model View**

It represents the dynamic of behavioural as parts of the system, depicting the interactions of collection between various structural elements described in the user model and structural model view.

- **Implementation model View**

In this the structural and behavioural as parts of the system are represented as they are to be built.

- **Environmental Model View**

In these the structural and behavioural aspects of the environment in which the system is to be implemented are represented.

3.3.1 Class Diagram:

A “Class Diagram” shows a set of classes, interfaces and collaborations and their relationships. These diagrams are the most common diagrams in modelling object-oriented systems. The class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing and documenting different aspects of a system but also for constructing executable code of the software application.

The class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modelling of object-oriented systems because they are the only UML diagrams which can be mapped directly with object oriented languages.

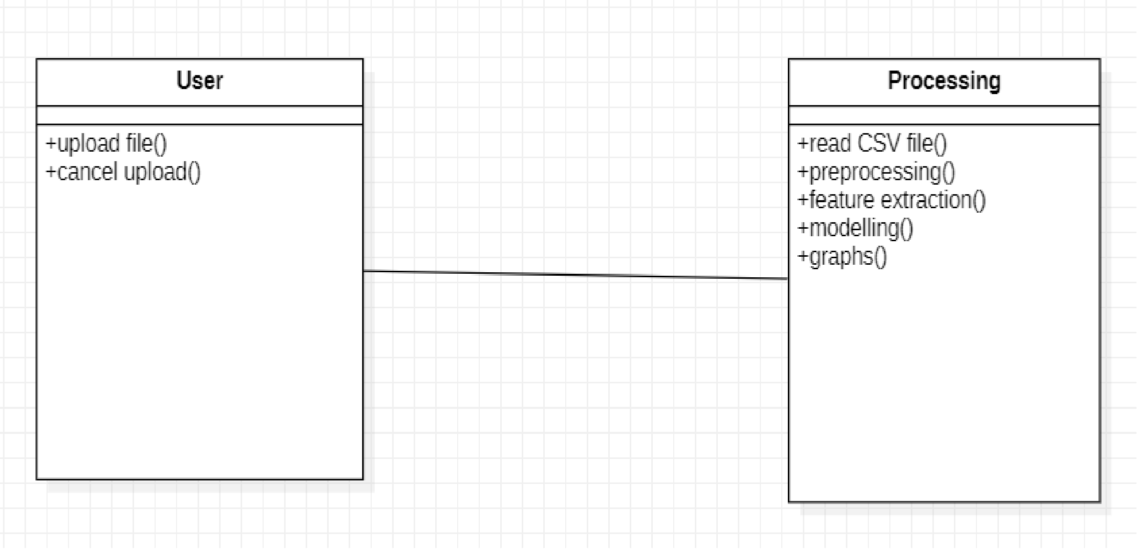


Figure 3.3.1 Class Diagram

3.3.2 Use Case Diagram:

Use case diagrams are one of the five diagrams in the UML for modelling the dynamic aspects of the systems (activity diagrams, sequence diagram, state chart diagram, collaboration diagram are the four other kinds of diagrams in the UML for modelling the dynamic aspects of systems).Use case diagrams are central to modelling the behaviour of the system, a sub-system, or a class. Each one shows a set ofuse cases and actors and relations

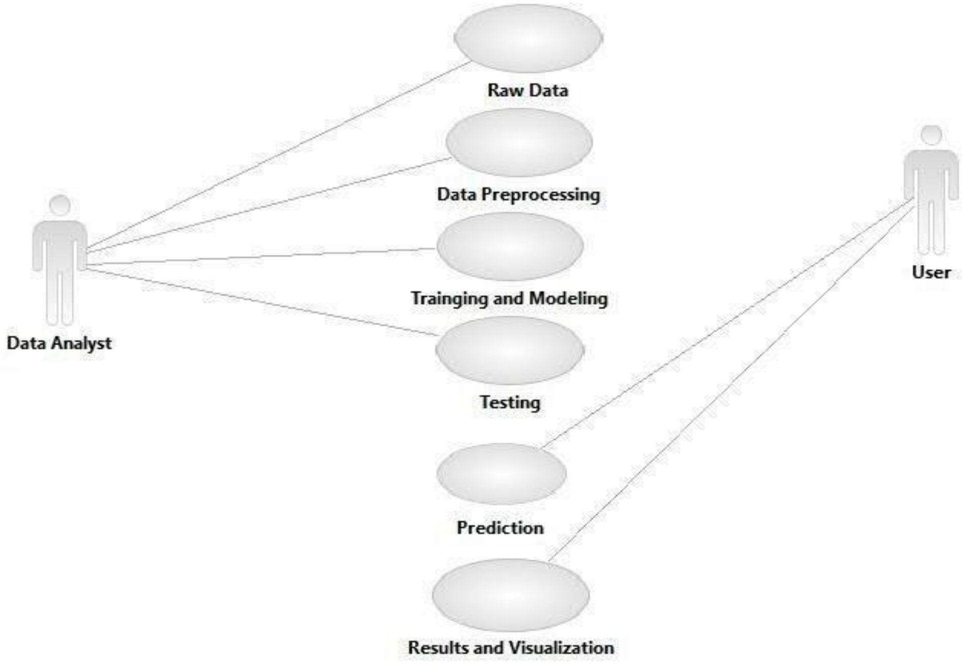


Figure 3.3.2 Use case Diagram

3.3.3 Sequence Diagram:

Sequence diagram is an interaction diagram which focuses on the time ordering of messages. It shows a set of objects and messages exchanged between these objects. This diagram illustrates the dynamic view of a system.

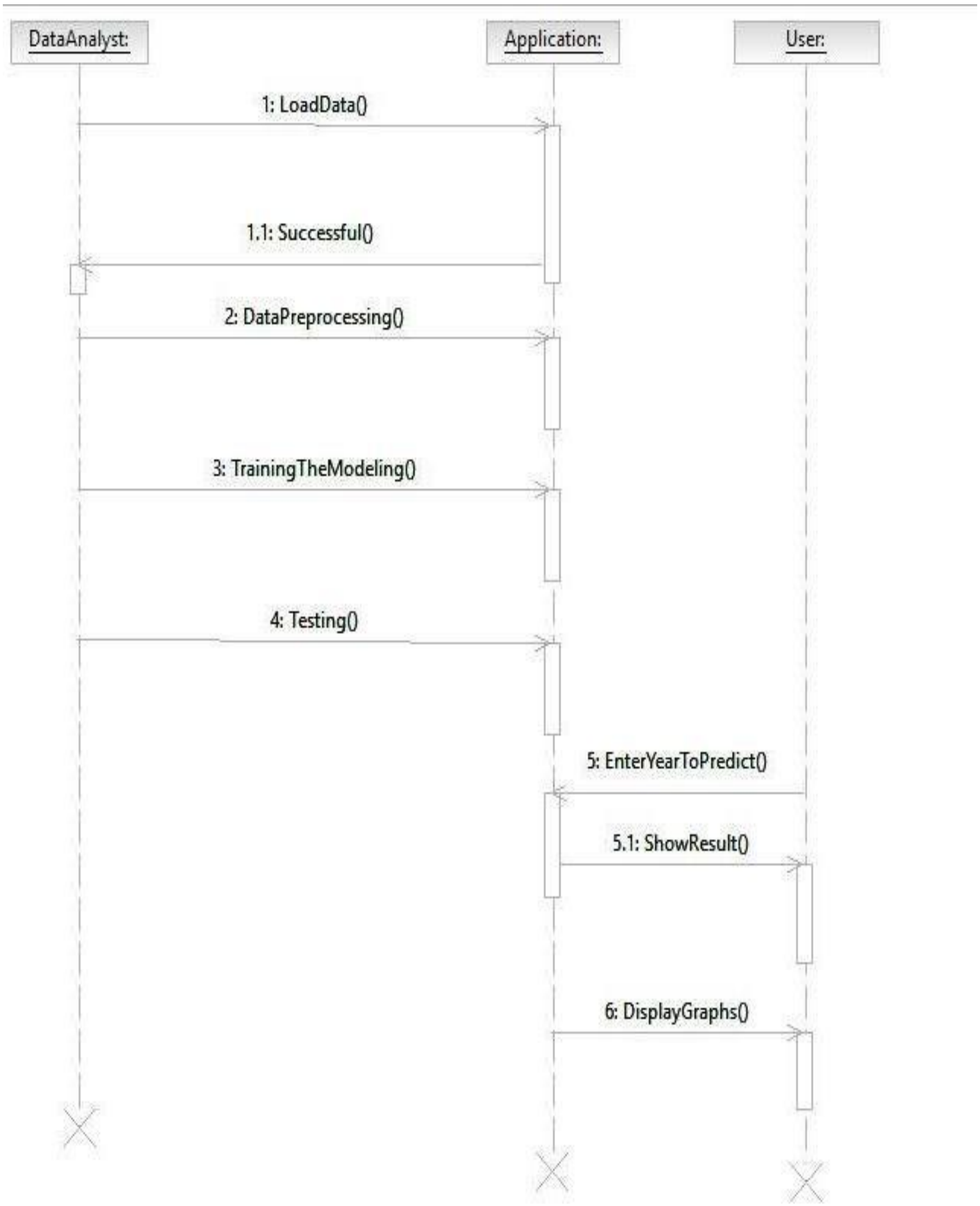


Figure3.3.3 Sequence Diagram

3.3.4 Collaboration Diagram:

Collaboration diagram is an interaction diagram that emphasizes the structural organization of the objects that send and receive messages. Collaboration diagram and sequence diagram are isomorphic.

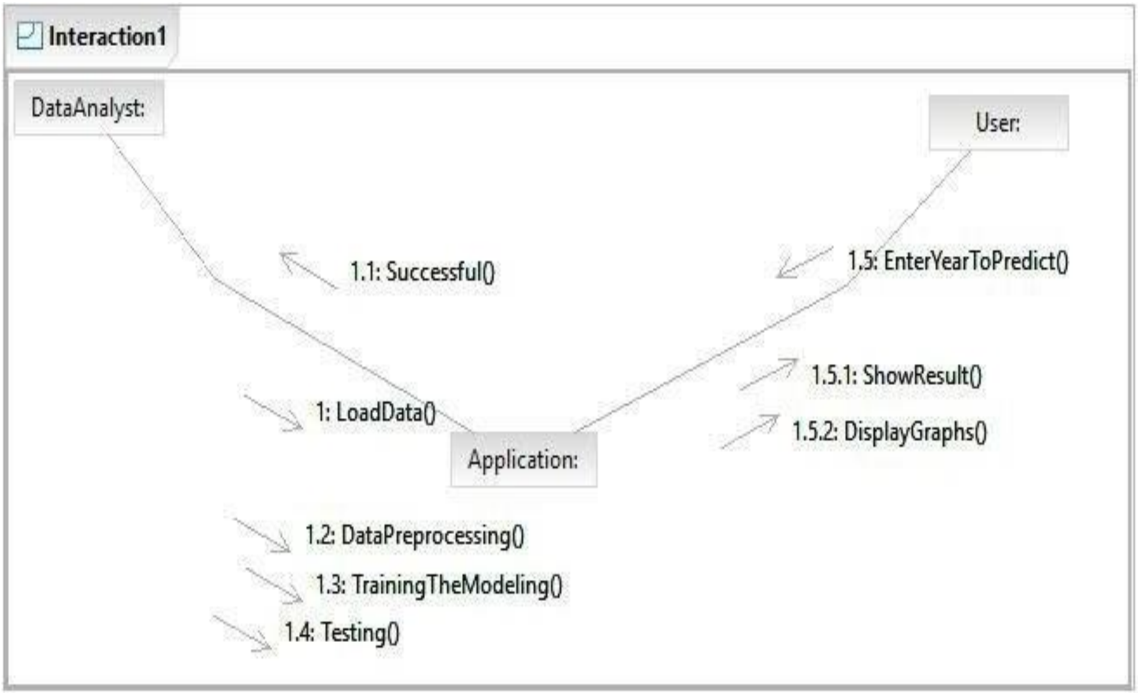


Figure 3.3.4 Collaboration Diagram

3.3.5 Activity Diagram:

An activity diagram shows the flow from activity to activity within a system it emphasizes the flowof control among objects**.**

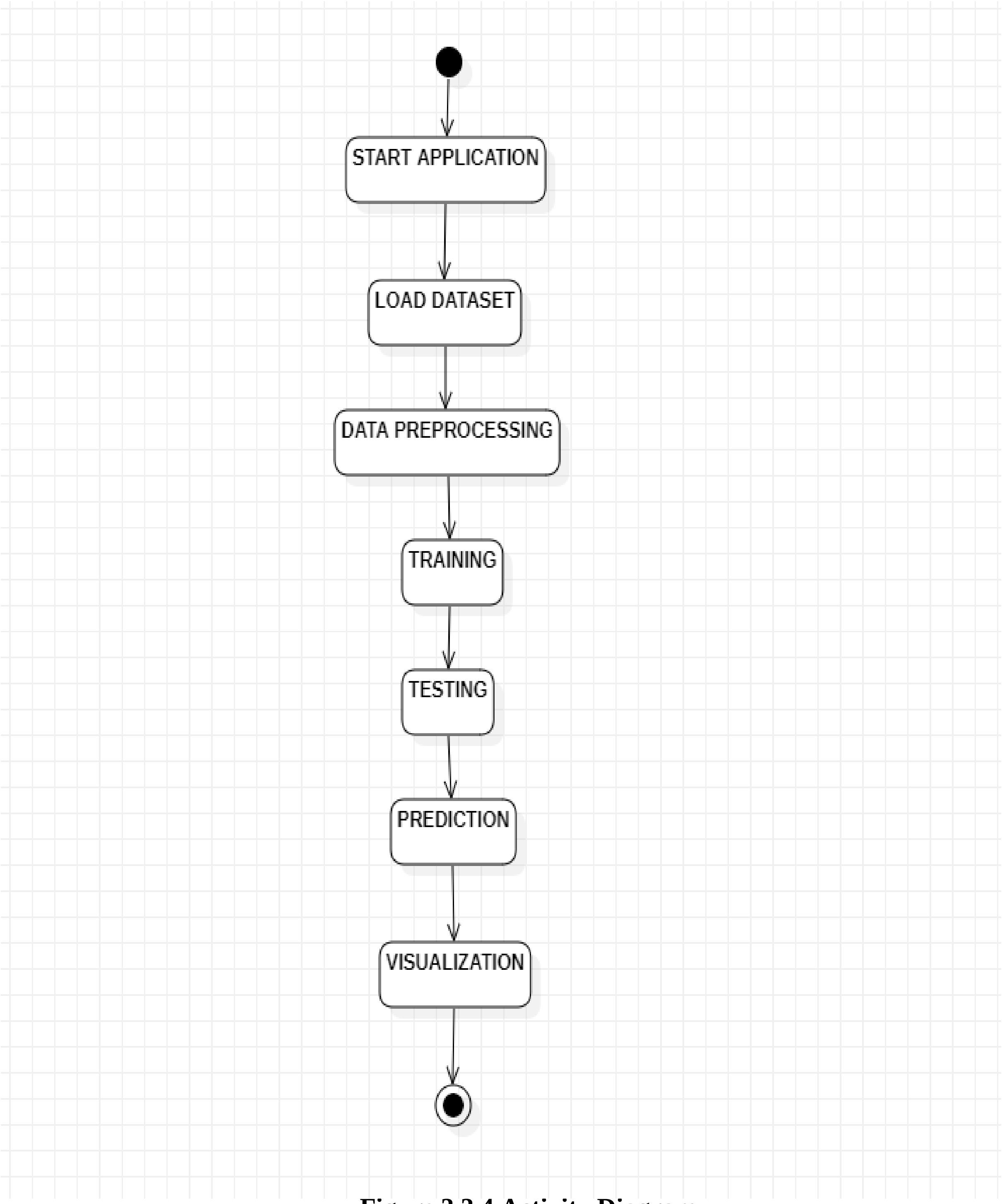
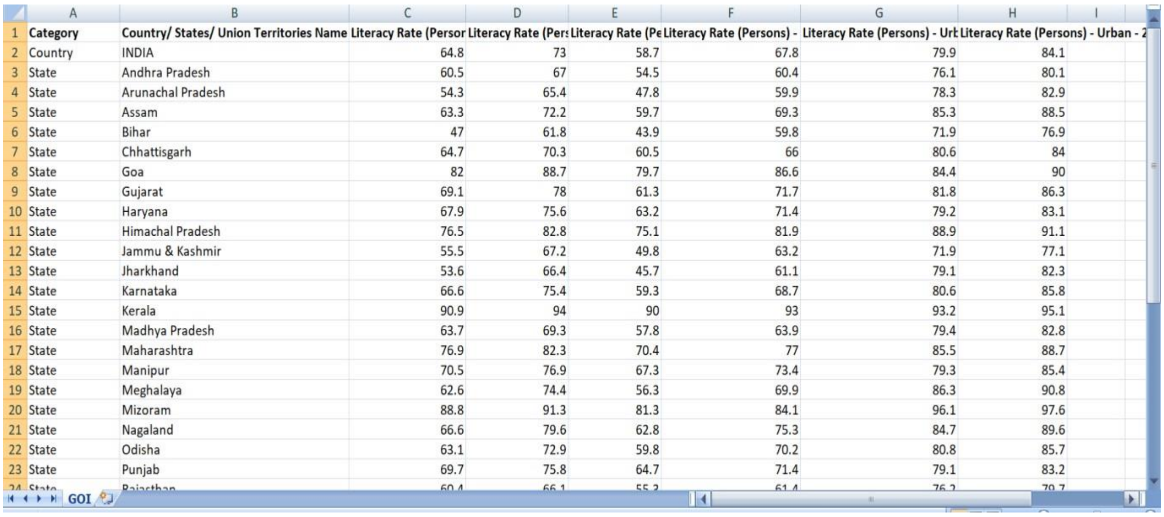


Figure 3.3.4 Activity Diagram

3.4 DATA DICTIONARY



	A	B	C	D	E	F	G	H	I
1	Category	Country/ States/ Union Territories Name	Literacy Rate (Persor	Literacy Rate (Pers	Literacy Rate (Pe	Literacy Rate (Persons) -	Literacy Rate (Persons) - Urt	Literacy Rate (Persons) - Urban - 2	
2	Country	INDIA	64.8	73	58.7	67.8	79.9	84.1	
3	State	Andhra Pradesh	60.5	67	54.5	60.4	76.1	80.1	
4	State	Arunachal Pradesh	54.3	65.4	47.8	59.9	78.3	82.9	
5	State	Assam	63.3	72.2	59.7	69.3	85.3	88.5	
6	State	Bihar	47	61.8	43.9	59.8	71.9	76.9	
7	State	Chhattisgarh	64.7	70.3	60.5	66	80.6	84	
8	State	Goa	82	88.7	79.7	86.6	84.4	90	
9	State	Gujarat	69.1	78	61.3	71.7	81.8	86.3	
10	State	Haryana	67.9	75.6	63.2	71.4	79.2	83.1	
11	State	Himachal Pradesh	76.5	82.8	75.1	81.9	88.9	91.1	
12	State	Jammu & Kashmir	55.5	67.2	49.8	63.2	71.9	77.1	
13	State	Jharkhand	53.6	66.4	45.7	61.1	79.1	82.3	
14	State	Karnataka	66.6	75.4	59.3	68.7	80.6	85.8	
15	State	Kerala	90.9	94	90	93	93.2	95.1	
16	State	Madhya Pradesh	63.7	69.3	57.8	63.9	79.4	82.8	
17	State	Maharashtra	76.9	82.3	70.4	77	85.5	88.7	
18	State	Manipur	70.5	76.9	67.3	73.4	79.3	85.4	
19	State	Meghalaya	62.6	74.4	56.3	69.9	86.3	90.8	
20	State	Mizoram	88.8	91.3	81.3	84.1	96.1	97.6	
21	State	Nagaland	66.6	79.6	62.8	75.3	84.7	89.6	
22	State	Odisha	63.1	72.9	59.8	70.2	80.8	85.7	
23	State	Punjab	69.7	75.8	64.7	71.4	79.1	83.2	
24	State	Rajasthan	60.4	66.1	55.3	61.4	76.2	79.7	

Table 3.4.1 Dataset for predicting literacy rate per year

CHAPTER 4

IMPLEMENTATION

4.1 INTRODUCTION

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus, it can be the most critical stage in achieving a successful new system and in givingthe user confidence that the new system will work and be effective. The implementation stage involves careful planning, investigation of the existing system and its constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

The project is implemented by accessing simultaneously from more than one system and more than one window in one system. The application is implemented in the Internet Information Services 5.0 web server under the Windows XP and accessed from various clients.

4.2 TECHNOLOGIES USED WHAT IS PYTHON?

Python is an interpreter, high-level programming language for general-purpose programming by "Guido van Rossum" and first released in 1991, Python has a design philosophy that emphasizes code readability, and a syntax that allows programmers to express concepts in fewer lines of code,notably using significant whitespace. It provides constructs that enable clear programming on bothsmall and large scales. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional, procedural, and has a large and comprehensive standard library.

Python interpreters are available for many operating systems. Python, the reference implementation of Python, is open-source software and has a community-based development model, as do nearly all its variant implementations. Python is managed by the non-profit Python Software Foundation.

Python is a general purpose, dynamic, high level and interpreted programming language. It supports object-oriented programming approach to develop applications. It is simple and easy to learn and provides lots of high-level data structures.

- Windows XP
- Python Programming
- Open-source libraries: Pandas, NumPy, SciPy, matplotlib, OpenCV

Python Versions

Python 2.0 was released on 16 October 2000 and had many major new features, including a cycle-detecting, garbage collector, and support for Unicode. With this release, the development process became more transparent, and community backed. Python 3.0 (initially called Python 3000or py3k) was released on 3 December 2008 after a long testing period. It is a major revision of the language that is not completely backward compatible with previous versions. However, many of its major features have been back ported to the Python 2.6.x and 2.7.x version series, and releasesof Python 3 include the 2to3 utility, which automates the translation of Python 2 code to Python

3. Python 2.7's end-of-life date (a.k.a. EOL, sunset date) was initially set at 2015, then postponedto 2020 out of concern that a large body of existing code could not easily be forward-ported to Python3.

In January 2017, Google announced work on a Python 2.7 to go Trans compiler to improve performance under concurrent workloads. Python 3.6 had changes regarding UTF-8 (in Windows,PEP 528 and PEP 529) and Python 3.7.0b1 (PEP 540) adds a new "UTF-8 Mode" (and overrides POSIX locale).

Why Python?

- Python is a scripting language like PHP, Perl, and Ruby.
- No licensing, distribution, or development fees
- It is a Desktop application.
- Linux, windows
- Excellent documentation
- Trans compiler to improve performance under concurrent workloads.

Python 3.6 had changes regarding UTF-8 (in Windows, PEP 528 and PEP 529) and Python 3.7. (PEP 540) adds a new "UTF-8 Mode" (and overrides POSIX locale).

Libraries Of Python:

Python's large standard library, commonly cited as one of its greatest strengths,provides tools suited too many tasks. For Internet-facing applications, many standard formats and protocols such as MIME and HTTP are supported. It includes modules for creating graphical user interfaces, connecting to relational databases, generating pseudorandom numbers, arithmetic with arbitrary precision decimals, manipulating regular expressions, and unit testing. Some parts of the standardlibrary are covered by specifications (for example, the Web Server Gateway Interface (WSGI) implementation wsgiref follows PEP 33), but most modules are not.They are specified by their code, internal documentation, and test suites (if supplied). However, because most of the standard library is cross-platform Python code, only a few modules need altering or rewriting for variant implementations.

As of March 2018, the Python Package Index (PyPI), the official repository for third-party Python software, contains over 130,000 packages with a wide range of functionality, including:

Graphical user interfaces

- Web frameworks
- Multimedia
- Databases
- Networking
- Test frameworks
- Automation
- Web scraping
- Documentation
- System administration

4.3 MACHINE LEARNING

Machine Learning is an application of artificial intelligence (AI) that provides system the ability to automatically learn and improve from experience without being explicitly programmed.

Machine learning focuses on the development of computer programs that can access data and useit learn for themselves.

Basics of python machine learning:

- You'll know how to use Python and its libraries to explore your data with the help of matplotliband Principal Component Analysis (PCA).
- And you'll pre-process your data with normalization, and you'll split your data into training andtest sets.
- Next, you'll work with the well-known K-Means algorithm to construct an unsupervised model,fit this model to your data, predict values, and validate the model that you have built.
- As an extra, you'll also see how you can also use Support Vector Machines (SVM) to constructanother model to classify your data.

Why Machine Learning?

- It was born from pattern recognition and theory that computers can learn without beingprogrammed to specific tasks.
- It is a method of Data analysis that automates analytical model building.

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. They are **Supervised learning:** The computer is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs. As special cases, the input signal can be only partially available, or restricted to special feedback: **Semi-supervised learning:** the computer is given only an incomplete training signal: a training set with some (often many) of the target outputs missing.

Active learning: The computer can only obtain training labels for a limited set of instances (basedon a budget), and also has to optimize its choice of objects to acquire labels for. When used interactively, these can be presented to the user for labelling.

Reinforcement learning: training data (in form of rewards and punishments) is given only as feedback to the program's actions in a dynamic environment, such as driving a vehicle or playinggame against an opponent.

Unsupervised learning: No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning). In regression, also a supervised problem, the outputs are continuous rather than discrete.

Regression: The analysis or measure of the association between one variable (the dependent variable) and one or more other variables (the independent variables), usually formulated in an equation in which the independent variables have parametric coefficients, which may enable future values of the dependent variable to be predicted.

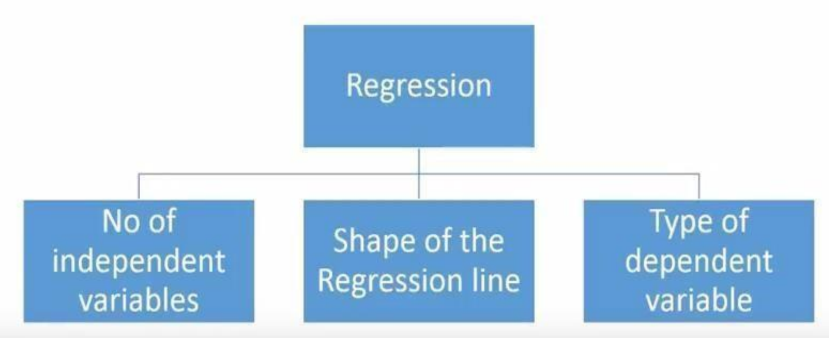

Regression
No of independent variables
Shape of the Regression line
Type of dependent variable

Figure 4.1.1 Regression Structure

Random forest Algorithm

Random forest Algorithm is a combination of decision trees and is not sensitive to data Random forest is a combination of Bootstrapping and Aggregation. The process that we used to create new data from the existing dataset is called bootstrapping and the data is called bootstrapped data.

The process of combining the results from multiple models is called aggregation. Bootstrapping ensures that we are not using the same data for every tree and makes sure that it becomes less sensitive to data. For Classification model, the Random. Forest algorithm takes the majority voting classifier and for regression model, it goes for the average of the predictions. In this project, ours is a regression model, so we go for the average of the predictions.

What is Regression Analysis?

Regression analysis is a form of predictive modelling technique which investigates the relationship between a dependent(target) and independent variable **(s)** (predictor). This technique is used for forecasting, time series modelling and finding the causal effect relationship between the variables

For example, relationship between rash driving and number of road accidents by

Types of Regression:

1. Linear Regression
2. Logistic Regression
3. Polynomial Regression
4. Stepwise Regression
5. Ridge Regression
6. Lasso Regression
7. Elastic Net Regression

1. Linear Regression: -It is one of the most widely known modelling techniques. Linear regression is usually among the first few topics which people pick while learning predictive modelling. In this technique, the dependent variable is continuous, independent variable(s) can be continuous or discrete, and nature of regression line is linear.

Linear Regression establishes a relationship between dependent variable (Y) and one or more independent variables (X) using a best fit straight line (also known as regression line).

2. Logistic Regression: -Logistic regression is used to find the probability of event=Success andevent=Failure. We should use logistic regression when the dependent variable is binary (0/ 1, True/ False, Yes/ No) in nature. Here the value of Y ranges from 0 to 1 and it can have representedby following equation.

odds= p/ (1-p) = probability of event occurrence / probability of not event occurrence

ln(odds) = ln(p/(1-p)) logit(p) = ln(p/(1-p)) =

b0+b1X1+b2X2+b3X3.
....+bkX
k

3. Polynomial Regression: -A regression equation is a polynomial regression equation if the power of independent variable is more than 1. The equation below represents a polynomial equation:

$$y=a+b*x^2$$

4. Stepwise Regression: -This form of regression is used when we deal with multiple independentvariables. In this technique, the selection of independent variables is done with the help of an automatic process, which involves *no* human intervention.

This feat is achieved by observing statistical values like R-square, t-stats and AIC metric to discern significant variables. Stepwise regression basically fits the regressionmodel by adding/dropping co-variants one at a time based on a specified criterion. Some of the most commonly used Stepwise regression methods are listed below:

- Standard stepwise regression does two things. It adds and removes predictors as needed for eachstep.
- Forward selection starts with most significant predictor in the model and adds variable for eachstep.
- Backward elimination starts with all predictors in the model and removes the least significantvariable for each step.

The aim of this modelling technique is to maximize the prediction power with minimumnumber of predictor variables. It is one of the methods to handle higher dimensionality of data set.

5. Ridge Regression: -Ridge Regression is a technique used when the data suffers from multi collinearity (independent variables are highly correlated). In multi collinearity, even though the least squares estimate (OLS) are unbiased; their variances are large which deviates the observed value far from the true value. By adding a degree of bias to the regression estimates, ridge regression reduces the standard errors.

Above, we saw the equation for linear regression. Remember? It can be represented as:

$$y=a+ b*x$$

This equation also has an error term. The complete equation becomes:

y=a+b*x+e (error term), [error term is the value needed to correct for a prediction error between the observed and predicted value]

=> y=a+y= a+ b1x1+ b2x2+ +e, for multiple independent variables.

In a linear equation, prediction errors can be decomposed into two sub components. First is due to the biased and second is due to the variance. Prediction error can occur due to any one of these two or bothcomponents. Here, we'll discuss about the error caused due to variance.

Ridge regression solves the multi collinearity problem throughshrinkage parameterλ (lambda).

Look at the equation below.

$$= \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \underbrace{\|y - X\beta\|_2^2}_{\text{Loss}} + \lambda \underbrace{\|\beta\|_2^2}_{\text{Penalty}}$$

In this equation, we have two components. First one is least square term and other one is lambda of the summation of β2 (beta- square) where β is the coefficient. This is added to least square termin order to shrink the parameter to have a very low variance.

Important Points:

- The assumption of this regression is same as least squared regression except normality is not tobe assumed
- It shrinks the value of coefficients but doesn't reach zero, which suggests no feature selectionfeature
- This is a regularization method and uses l2 regularization.

6. Lasso Regression: -Similar to Ridge Regression, Lasso (Least Absolute Shrinkage and Selection Operator) also penalizes the absolute size of the regression coefficients. In addition, it can reduce the variability and improving the accuracy of linear regression models.

Look at the equation below:

$$= \underset{\beta\in \mathrm{R}^p}{\operatorname{argmin}} \underbrace{\|y - X\beta\|_2^2}_{\text{Loss}} + \lambda \underbrace{\|\beta\|_1}_{\text{Penalty}}$$

Lasso regression differs from ridge regression in a way that it uses absolute values in the penalty function, instead of squares.

This leads to penalizing (or equivalently constraining the sum of the absolute values of the estimates) values which causes some of the parameter estimates to turn out exactly zero. Larger the penalty applied, further the estimates get shrunk towards absolute zero. This results to variable selection, out of given n variables.

Important Points:

- The assumption of this regression is same as least squared regression except normality is not tobe assumed
- It shrinks coefficients to zero (exactly zero), which certainly helps in featureselection
- This is a regularization method and usesl1 regularization
- If group of predictors are highly correlated, lasso picks only one of them and shrinks the othersto zero

7. Elastic Net Regression: -Elastic Net is hybrid of Lasso and Ridge Regression techniques. It is trained with L1 and L2 prior as regularize. Elastic-net is useful when there are multiple features which are correlated. Lasso is likely to pick one of these at random, while elastic-net is likely to pick both

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}}(\|y - X\beta\|^2 + \lambda_2\|\beta\|^2 + \lambda_1\|\beta\|_1).$$

A practical advantage of trading-off between Lasso and Ridge is that, it allows Elastic-Net to inherit some of Ridge’s stability under rotation.

Important Points:

- It encourages group effect in case of highly correlated variables
- There are no limitations on the number of selected variables
- It can suffer with double shrinkage

Beyond these 7 most commonly used regression techniques, you can also look at other models likeBayesian,EcologicalandRobust regression.

Modules in pythonModule: -

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 that you canbind and reference.

Pandas: -

Pandas is a Python package providing fast, flexible, and expressive data structures designed to make working with “relational” or “labelled” data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, real world data analysis in Python. Additionally, it has the broader goal of becoming the most powerful and flexible open-source data analysis / manipulation tool available in any language. It is already well on its way toward this goal.

Pandas is well suited for many kinds of data:

- Tabular data with heterogeneously typed columns, as in an SQL table or Excel spread sheet
- Ordered and unordered (not necessarily fixed frequency) time series data.
- Arbitrary matrix data (homogeneously typed or heterogeneous) with row and column labels
- Any other form of observational / statistical data sets. The data actually need not be labelled at all to be placed into a panda’s data structure
- The two primary data structures of pandas,Series(1-dimensional) and Data Frame (2dimensional), handle the vast majority of typical use cases in finance, statistics, social science, and many areas of engineering. For R users,Data Frameprovides everything that R’s data frame provides and much more. Pandas is built on top of NumPy and is intended to integrate well within a scientific computing environment with many other 3rd party libraries. Few of the things that pandas do well:
- Easy handling of missing data (represented as Nan) in floating point as well as non floating-point data

- **Size mutability:** columns can be inserted and deleted from Data Frame and higher dimensionalobjects
- **Automatic and explicit data alignment**: objects can be explicitly aligned to a set of labels, or the user can simply ignore the labels and let *Series*, *Data Frame*, etc. automatically align the datafor you in computations
- Powerful, flexible group by functionality to perform split-apply-combine operations on data sets,for both aggregating and transforming data
- Make it easy to convert ragged, differently indexed data in other Python and NumPy data structures into Data Frame objects
- Intelligent label-based slicing**,** fancy indexing, and sub setting of large data sets
- Intuitive merging and joining data sets
- Flexible reshaping and pivoting of data sets
- Hierarchical labelling of axes (possible to have multiple labels per tick)
- Robust IO tools for loading data from flat files (CSV and delimited), Excel files, databases, andsaving / loading data from the ultrafast HDF5 format
- **Time series-specific functionality:** date range generation and frequency conversion, movingwindow statistics, moving window linear regressions, date shifting and lagging, etc.

Many of these principles are here to address the shortcomings frequently experienced using otherlanguages / scientific research environments. For data scientists, working with data is typically divided into multiple stages: munging and cleaning data, analysing / modelling it, then organizingthe results of the analysis into a form suitable for plotting or tabular display. pandas is the ideal tool for all of these tasks.

- pandas is fast**.** Many of the low-level algorithmic bits have been extensively improved in Python code. However, as with anything else generalization usually sacrifices performance. So, if you focus on one feature for your application you may be able to create a faster specialized tool.

- pandas are a dependency of stats models, making it an important part of the statistical computingecosystem in Python.
- pandas have been used extensively in production in financial applications.

NumPy: -

NumPy ,which stands for Numerical Python ,is a library consisting of multidimensional array objects and a collection of routines for processing those arrays .Using NumPy ,mathematical and logical operations on arrays can be performed .This tutorial explains the basics of NumPy such as its architecture and environment. It also discusses the various array functions , types of indexing ,etc. An introduction to Matplotlib is also provided.

All this is explained with the help of examples for better understanding.

NumPy is a Python package. It stands for 'Numerical Python'. It is a library consisting of multidimensional array objects and a collection of routines for processing of array. Numeric, the ancestor of NumPy, was developed by Jim Humulin. Another package Numara was also developed, having some additional functionalities. In 2005, Travis Oliphant created NumPy package by incorporating the features of Numara into Numeric package. There are many contributors to this open-source project.

Operations using NumPy: -

Using NumPy, a developer can perform the following operations −

- Mathematical and logical operations on arrays.
- Fourier transforms and routines for shape manipulation.
- Operations related to linear algebra. NumPy has in-built functions for linear algebra and randomnumber generation.

NumPy – A Replacement for MATLAB

NumPy is often used along with packages like SciPy (Scientific Python) and Mat−plotid (plotting library). This combination is widely used as a replacement forMATLAB, a popular platform for technical computing. However, Python alternative to MATLAB is now seen as a more modern and complete programming language. It is open source, which is an added advantage of NumPy.

Sickit-learn: -

Scikit-learn (formerly scikits. 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 Courmayeur. Its name stems from the notion that it is a "SciKit"(SciPy Toolkit), a separately- developed and distributed third-party extension to SciPy. The original codebase was later rewritten by other developers. In 2010 Fabian Pedrosa, Gael Viroqua, Alexandre Gram fort and Vincent Michel, all from INRIA took leadership of the project and made the first public release on February the 1st 2010 .Of the various scikits, scikit-learn as well as scikit-image were described as "well-maintained and popular" in November 2012.

Scikit-learn is largely written in Python, with some core algorithms written in Python to achieve performance. Support vector machines are implemented by a Python wrapper around LIBSVM; logistic regression and linear support vector machines by a similar wrapper around LIBLINEAR.

Some popular groups of models provided by scikit-learn include:

- **Ensemble methods:** for combining the predictions of multiple supervised models.
- **Feature extraction**: for defining attributes in image and text data.
- **Feature selection:** for identifying meaningful attributes from which to create supervised models.
- **Parameter Tuning:** for getting the most out of supervised model
- **Manifold Learning:** For summarizing and depicting complex multi-dimensional data.
- **Supervised Models:** a vast array not limited to generalize linear models, discriminate analysis,naive bayes, lazy methods, neural networks, support vector machines and decision trees.

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.

CHAPTER 5

TESTING

It is the process of testing the functionality and it is the process of executing a program with the intent of finding an error. A good test case is one that has a high probability of finding an as at undiscovered error. A successful test is one that uncovers an as at undiscovered error. Software testing is usually performed for one of two reasons:

- Defect Detection
- Reliability estimation

BLACK BOX TESTING:

The base of the black box testing strategy lies in the selection of appropriate data as per functionality and testing it against the functional specifications in order to check for normal and abnormal behaviour of the system. Now a days, it is becoming to route the testing work to a third party as the developer of the system knows too much of the internal logic and coding of the system, which makes it unfit to test application by the developer.

The following are different types of techniques involved in black box testing.

They are:

- Decision Table Testing
- All pairs testing
- State transition tables testing
- Equivalence Partitioning

Software testing is used in association with Verification and Validation. Verification is the checking of or testing of items, including software, for conformance and consistency with an associated specification. Software testing is just one kind of verification, which also uses techniques as reviews, inspections, walk-through. Validation is the process of checking what has been specified is what the user wanted.

- Validation: Are we doing the right job?
- Verification: Are we doing the job right?

In order to achieve consistency in the Testing style, it is imperative to have and follow a set of testing principles. This enhances the efficiency of testing within SQA team members and thus contributes to increased productivity. The purpose of this document is to provide overview of the testing, plus the techniques. Here, after training is done on the training dataset, testing is done.

After testing is done, the result will be displayed in the form of a graph.

WHITE BOX TESTING:

White box testing [10] requires access to source code. Though white box testing [10] can be performed any time in the life cycle after the code is developed, it is a good practice to perform white box testing [10] during unit testing phase. In designing of database, the flow of specific inputs through the code, expected output and the functionality of conditional loops are tested.

At SDEI, 3 levels of software testing is done at various SDLC phases

- **UNIT TESTING**: in which each unit (basic component) of the software is tested to verify that thedetailed design for the unit has been correctly implemented
- **INTEGRATION TESTING**: in which progressively larger groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a whole.
- **SYSTEM TESTING**: in which the software is integrated to the overall product and tested to showthat all requirements are met. A further level of testing is also done, in accordance with requirements:
- **REGRESSION TESTING**: is used to refer the repetition of the earlier successful tests to ensure that changes made in the software have not introduced new bugs/side effects.
- **ACCEPTANCE TESTING:** Testing to verify a product meets customer specified requirements. The acceptance test suite is run against supplied input data. Then the results obtained are compared with the expected results of the client. A correct match was obtained.

TEST CASES

- To make sure whether system meets user requirements or not.
- To make sure that during the operation, incorrect input, processing and output will be detected.
- To see that when correct inputs are fed to the system the outputs are correct.To verify that the controls are incorporated in the same system as intended.

Test case1: year wise literacy rate prediction Test case: year wise literacy rate Dataset

Test Description: Provide population, literacy rate and year.

Action Performed: give a year as input

Expected Results: predict literacy rate

Result: Success. With accuracy 82.56 percentage

Test case2: state wise literacy rate Prediction Test case: State wise literacy rate Dataset.

Test Description: Provide state names and population

Action Performed: Linear regression algorithms.

Expected Results: Visualization graph.

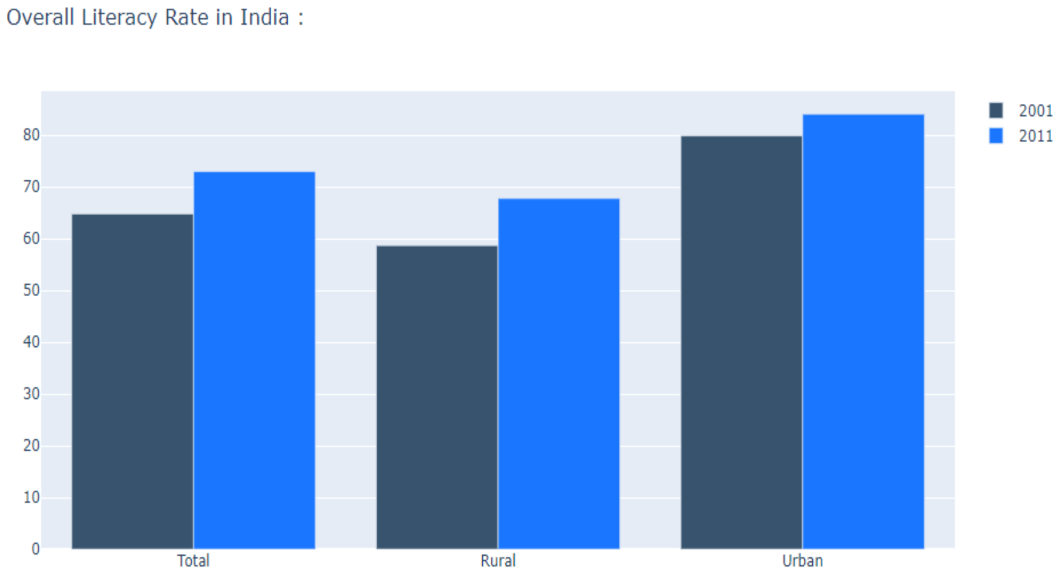
Result: Success.

CHAPTER 6

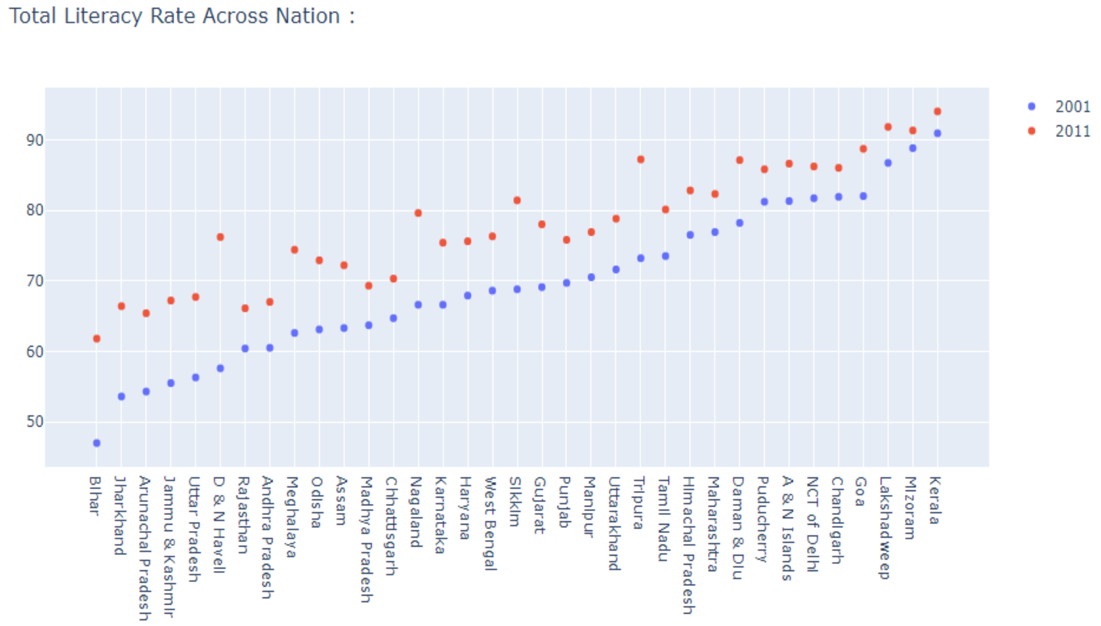
RESULTS

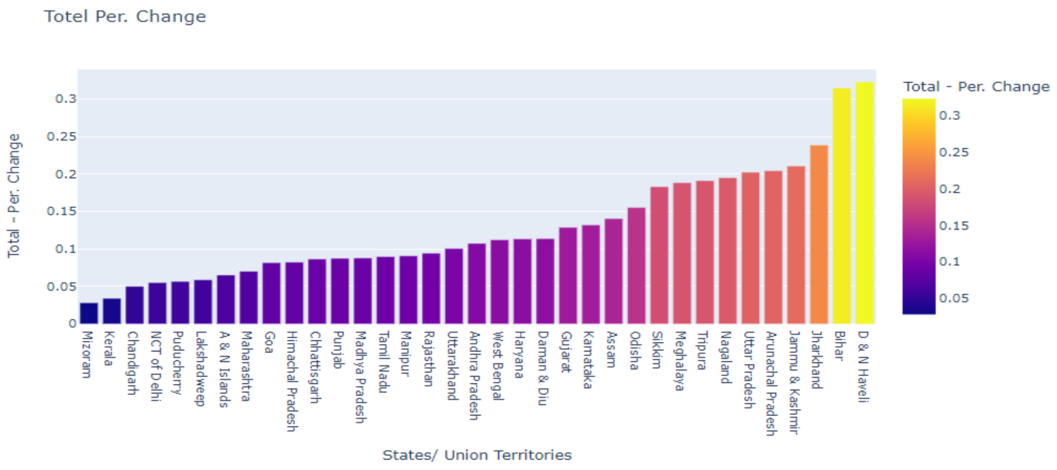
The result screenshots are as follows

Overall Literacy Rates in India.



Total Literacy Rate Across Nation:

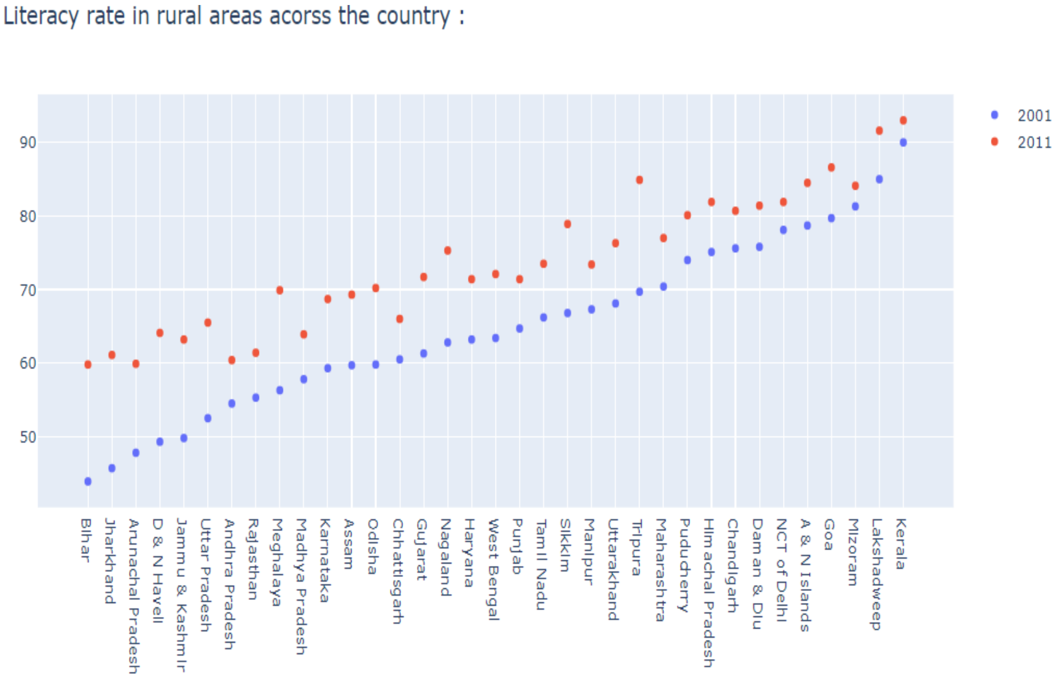




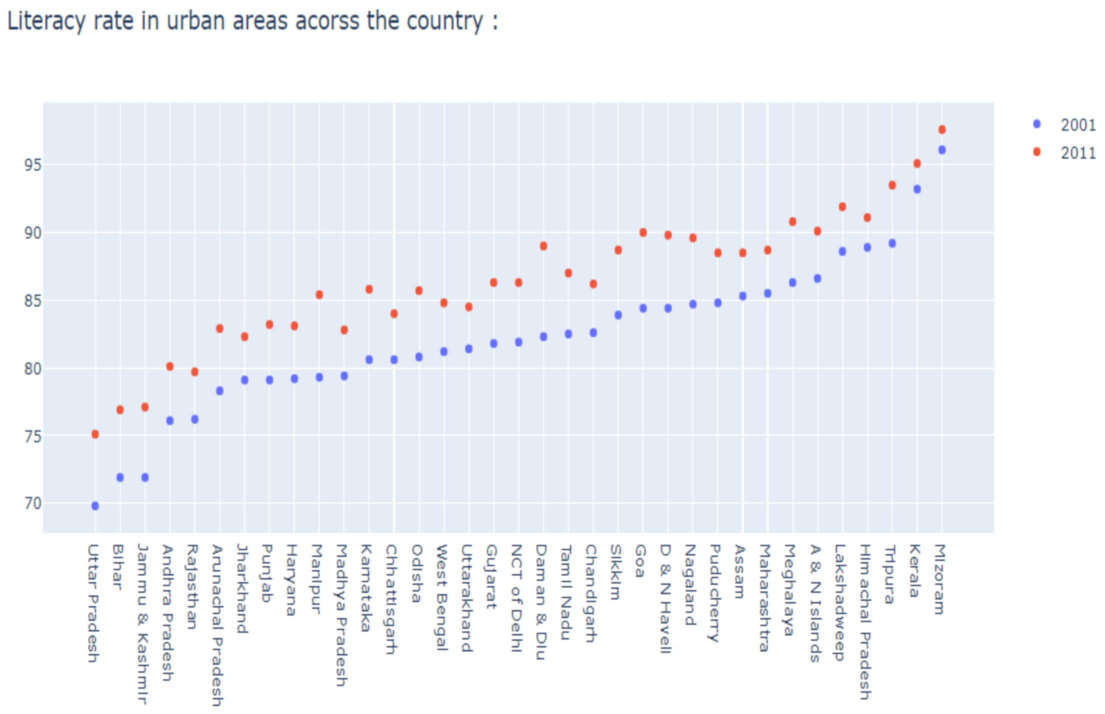
INSIGHTS :

- Bihar, Jharkhand, Arunachal Pradesh, Jammu & Kashmir and Uttar Pradesh were the least literate states/Union Territories in 2001.
- Kerala, Mizoram, Lakshadweep, Goa and Chandigarh are the most literate states/Union territories in 2001.
- Rajasthan and Andhra Pradesh Couldn't keep up with other states and fell in 5 least literate states with Bihar, Arunachal Pradesh and Jharkhand. Whereas Jammu & kashmir and Uttar pradesh managed to improve in 2011.
- Tripura managed to increase it's literacy rate to 5 most literate states along with Kerala, Lakshadweep, mizoram and Goa in 2011.
- Mizoram, Kerala, Chandigarh, NCT of Delhi and Ponducherry have least percentage increase in literacy rate.
- Percentage Increse in Total Literacy is highest in D & N Haveli, Bihar, Jharkhand, Jammu & Kashmir and Arunachal Pradesh.
- In Year 2001 total 13 States/Union Territories had lesser literacy rate than overall indian literacy rate.
- In Year 2011 total 11 States/Union Territories had lesser literacy rate than overall indian literacy rate. Meghalaya and D & N Haveli managed to increse their literacy rate.
- Bihar, Jharkhand, Arunachal Pradesh, Jammu & Kashmir, Uttar Pradesh, Rajasthan, Andhra Pradesh, Odisha, Assam, Madhya Pradesh and Chhattisgarh still have lesser Total literacy rate than overall literacy rate of the Country.

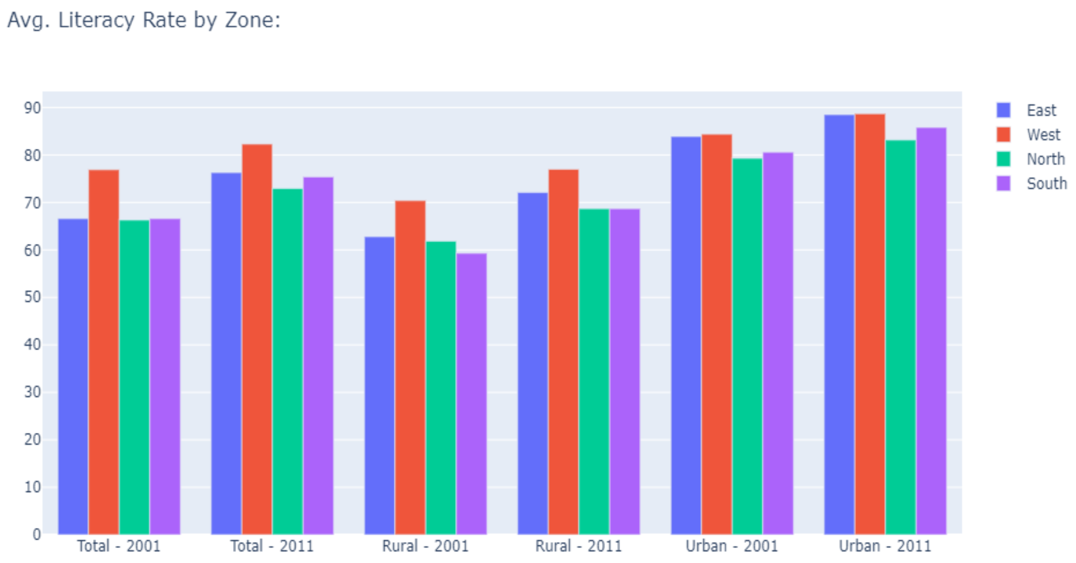
Rural Literacy Rate Across Nation:

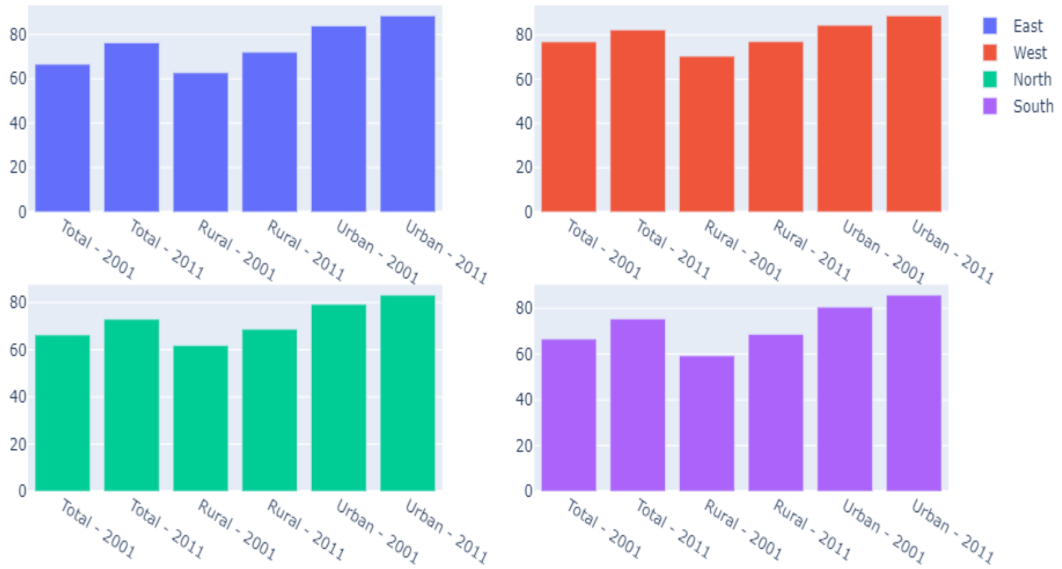


Urban Literacy Rate Across Nation:



Average literacy rate by zone:

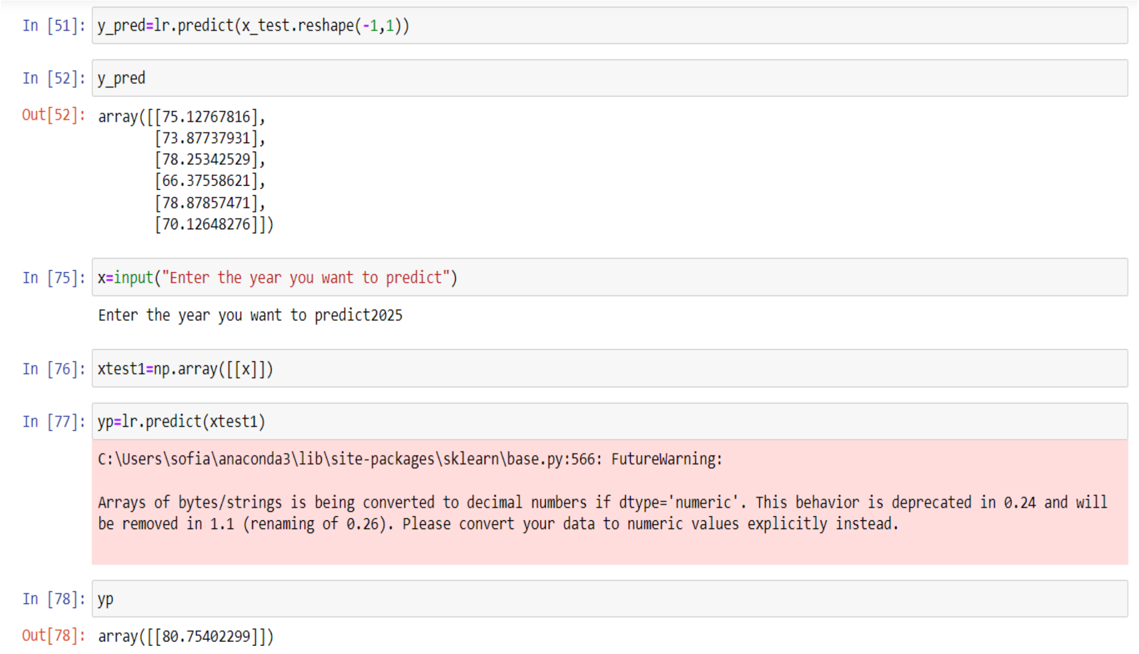




Predicting and analyzing the Indian literacy rate

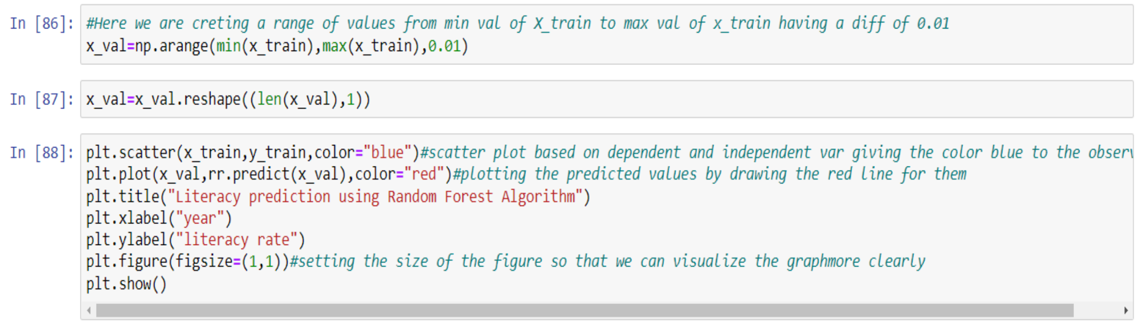
We are using previous dataset of state-wise literacy rate and predicting the future Indian literacy rate. For this we should provide some dataset to process by giving the path of it to predict the result. The figure below shows the results:

Using linear Regression



```
In [51]: y_pred=lr.predict(x_test.reshape(-1,1))

In [52]: y_pred

Out[52]: array([[75.12767816],
                [73.87737931],
                [78.25342529],
                [66.37558621],
                [78.87857471],
                [70.12648276]])

In [75]: x=input("Enter the year you want to predict")

         Enter the year you want to predict2025

In [76]: xtest1=np.array([[x]])

In [77]: yp=lr.predict(xtest1)

         C:\Users\sofia\anaconda3\lib\site-packages\sklearn\base.py:566: FutureWarning:

         Arrays of bytes/strings is being converted to decimal numbers if dtype='numeric'. This behavior is deprecated in 0.24 and will
         be removed in 1.1 (renaming of 0.26). Please convert your data to numeric values explicitly instead.

In [78]: yp

Out[78]: array([[80.75402299]])
```

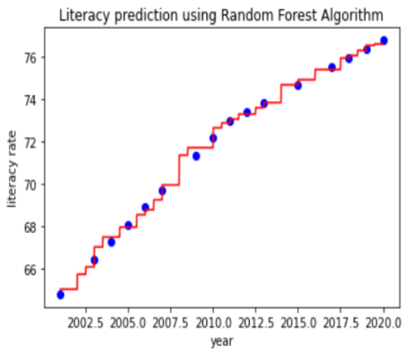
Using Random Forest Algorithm



In [86]:
```
#Here we are creting a range of values from min val of X_train to max val of x_train having a diff of 0.01
x_val=np.arange(min(x_train),max(x_train),0.01)
```

In [87]:
```
x_val=x_val.reshape((len(x_val),1))
```

In [88]:
```
plt.scatter(x_train,y_train,color="blue")#scatter plot based on dependent and independent var giving the color blue to the obser
plt.plot(x_val,rr.predict(x_val),color="red")#plotting the predicted values by drawing the red line for them
plt.title("Literacy prediction using Random Forest Algorithm")
plt.xlabel("year")
plt.ylabel("literacy rate")
plt.figure(figsize=(1,1))#setting the size of the figure so that we can visualize the graphmore clearly
plt.show()
```



```
<Figure size 72x72 with 0 Axes>
```

CHAPTER 7

CONCLUSION

Indian literacy rate can be predicted. It can be used by the Indian government to come up with plans based on the predicted literacy rate. The prediction made is for coming years i.e., till 2050. Based on the prediction and analysis made the Indian Government can conduct campaigns to increase the rate anddecrease dropout rates.

CHAPTER 8

FUTURE SCOPE AND ENHANCEMENTS

Predictions can be done beyond 2050.Prediction can be made to every district in each state.

Age vs state wise prediction can also be made. For example in every state 10-15 years literacy can bepredicted. And also age vs state vs gender wise prediction can also be made.

CHAPTER 9

BIBLIOGRAPHY

[1] A Machine Learning Model for Stock Market Prediction Article by Osman Hegazy and MustafaAbdul Salam

[2] The Unified Modeling Language User Guide, Low Price Edition Grady Booch, James Rumbaugh, Ivar Jacob,ISBN 81-7808-769-5, 91, 1997

[3] The Elements of UM Scott Ambler-Cambridge University Press [4] Newyork@2005,ISBN: 978-0-07-52616-7-82, 2005

[5] Software Testing, CemKaner, James beach, BretPettichord, ISBN 978-0-471-120940,2001

[6] A Practitioner's Approach Roger S. Pressman, Software Engineering,3rd Edition,ISBN: 978-007-126782-3,482, 2014

[7] Black Box Testing: Techniques for Functional Testing of Software and Systems Boris Beizer -WileyPublications, ISBN: 978-0-471-120-940, 1995

[8] Machine Learing in python : Essential Techniques for Predictive Analysis-Michael Bowles

[9] Numpy in Python- https://www.geeksforgeeks.org/numpy-in-python-set-1-introduction/

[10] Pandas Basics, https://www.leampython.org/en/Pandas Basics