A Project Report on

A NOVEL METHODOLOGY FOR DIAGNOSING CHRONIC KIDNEY DISEASE USING MACHINE LEARNING

Submitted in partial fulfillment of the requirements for the award of the Degree of

Bachelor of Technology

in

Computer Science and Engineering

By

KOPPANATHI MOUNIKA	20A95A0516
VINNAKOTA RAMA SESHU	19A91A05J5
CHINTALAPUDI LAKSHMI DURGA	20A95A0515

Under the Esteemed Supervision Of

Dr. ANAND KUMAR KINJARAPU (M.Tech.,Ph. D)
Professor

Department Of Computer Science and Engineering

ADITYA ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Affiliated to JNTUK Kakinada, Accredited by NAAC with ‘A’ Grade)

Aditya Nagar, ADB Road, Surampalem

2019 – 2023

ADITYA ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Affiliated to JNTUK Kakinada, Accredited by NAAC with ‘A’ Grade)
Aditya Nagar, ADB Road, Surampalem
2019 – 2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the thesis entitled “**A NOVEL MATHODOLOGY FOR DIAGNOSING CHRONIC KIDNEY DISEASE USING MACHINE LEARNING”** is being submitted by

KOPPANATHI MOUNIKA	**20A95A0516**
VINNAKOTA RAMA SESHU	**19A91A05J5**
CHINTALAPUDI LAKSHMI DURGA	**20A95A0515**

in partial fulfillment of the requirements for the award of degree of **B. Tech** in Computer Science and Engineering from **Jawaharlal Nehru Technological University Kakinada** is a record of bonafide work carried out by them at **Aditya Engineering College**

The results embodied in this Project report have not been submitted to any other University orInstitute for the award of any degree or diploma.

PROJECT GUIDE
Dr. Anand Kumar Kinjarapu. M.Tech.,Ph.D
Professor

HEAD OF THE DEPARTMENT
Dr. A. Vanathi M.E.,Ph.D
Associate Professor

EXTERNAL EXAMINER

DECLARATION

We hereby declare that the project entitled **"A NOVEL METHODOLOGY FOR DIAGNOSING CHRONIC KIDNEY DISEASE USING MACHINE LEARNING"** is a genuine project. This work has been submitted to the **ADITYA ENGINEERING COLLEGE,** Surampalem, permanently affiliated to **JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, KAKINADA** in partial fulfillment of the **B. Tech** degree**.** We further declare that this project work has not been submitted in full or part of the award of any degree of this or any other educational institutions.

by

KOPPANATHI MOUNIKA	**(20A95A0516)**
VINNAKOTA RAMA SESHU	**(19A91A05J5)**
CHINTALAPUDI LAKSHMI DURGA	**(20A95A0515)**

ACKNOWLEDGEMENT

It is with immense pleasure that we would like to express our indebted gratitude to our project guide **Dr. Anand Kumar Kinjarapu, Professor, Department of CSE** who has guided us a lot and encouraged us in every step of the project work, his valuable moral support and guidance throughout the project helped us to a greater extent.

Our deepest thanks to our HOD **Dr. A. Vanathi, Associate Professor** for inspiring us all the way and for arranging all the facilities and resources needed for our project.

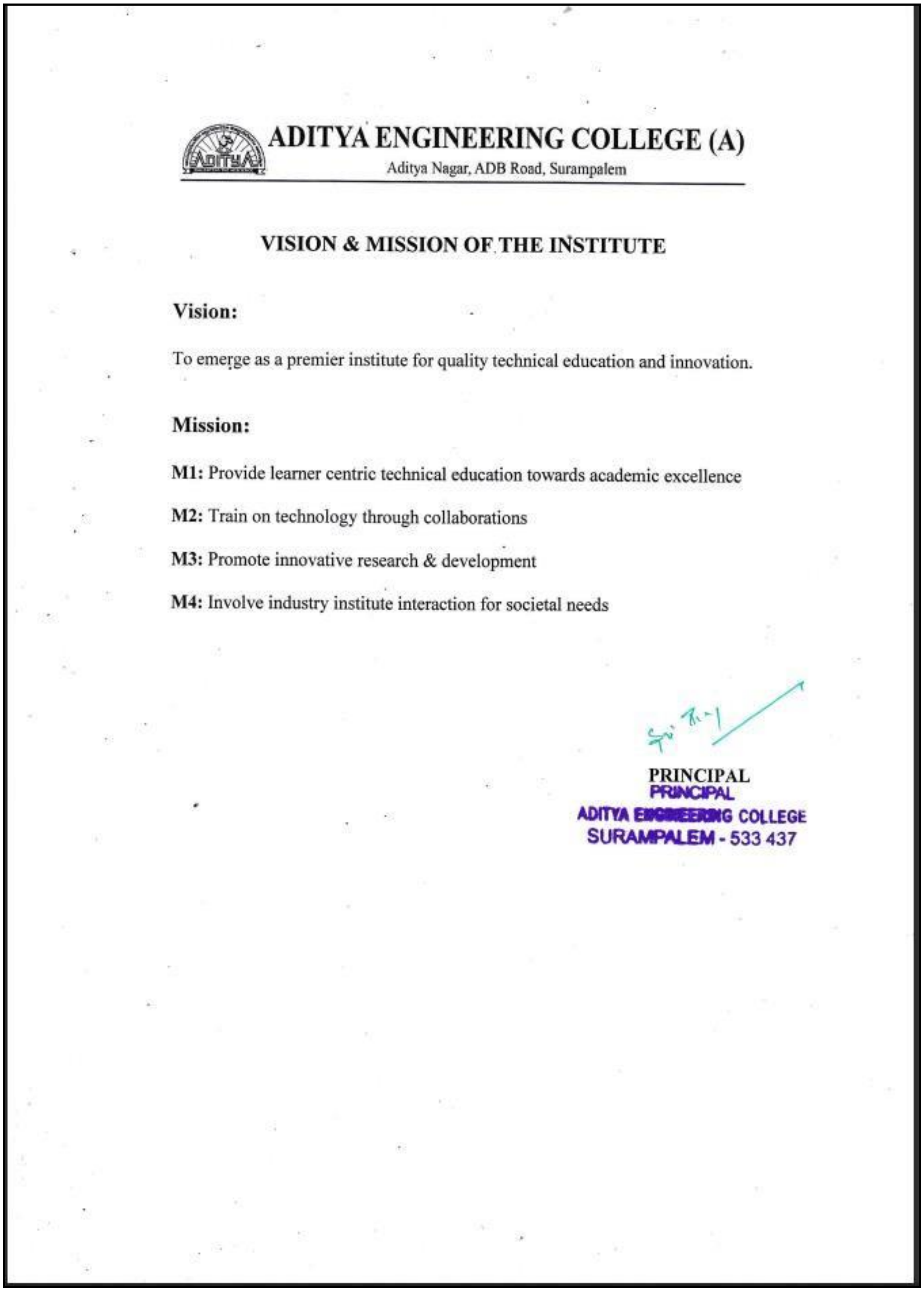
We would like to express our indebted gratitude to our project coordinator **Dr.V. Ravi Kishore, M. Tech Ph.D., Associate Professor, Department of CSE** who has guided us a lot and encouraged us in every step of the project work.

We wish to thank **Dr. S. Rama Sree, Professor** in CSE and Dean (Academics) for her support and suggestions during our project work.

We owe our sincere gratitude to **Dr. M. Sreenivasa Reddy, Principal** for providing a great support and for giving us the opportunity of doing the project.

We are thankful to our **College Management** for providing all the facilities in time to us for completion of our project.

Not to forget, **Faculty, Lab Technicians, non-teaching staff, and our friends** who have directly or indirectly helped and supported us in completing our project in time.



ADITYA ENGINEERING COLLEGE (A)

Aditya Nagar, ADB Road, Surampalem

VISION & MISSION OF THE INSTITUTE

Vision:

To emerge as a premier institute for quality technical education and innovation.

Mission:

M1: Provide learner centric technical education towards academic excellence

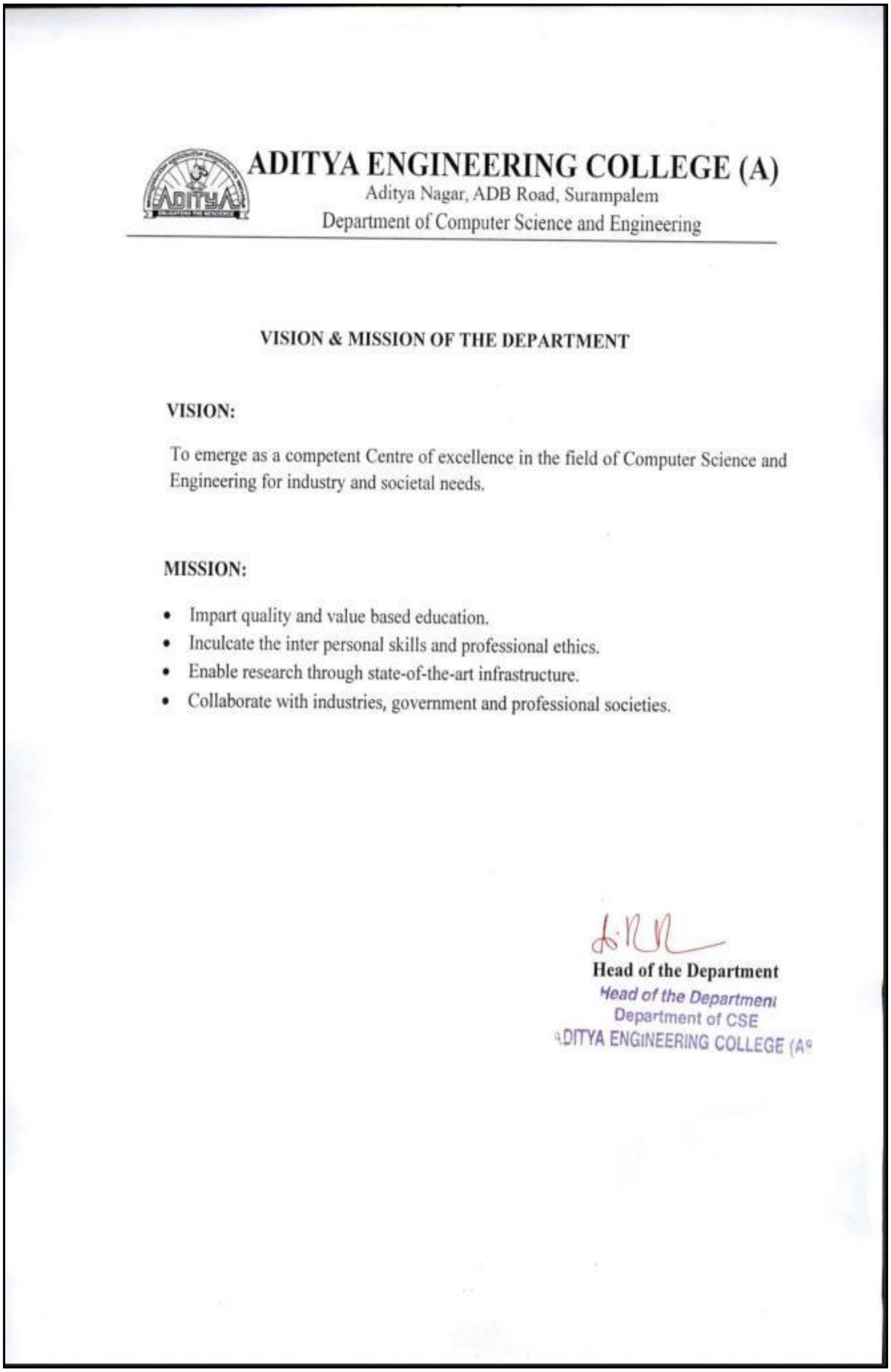
M2: Train on technology through collaborations

M3: Promote innovative research & development

M4: Involve industry institute interaction for societal needs

PRINCIPAL

PRINCIPAL
ADITYA ENGINEERING COLLEGE
SURAMPALEM - 533 437



ADITYA ENGINEERING COLLEGE (A)

Aditya Nagar, ADB Road, Surampalem
Department of Computer Science and Engineering

VISION & MISSION OF THE DEPARTMENT

VISION:

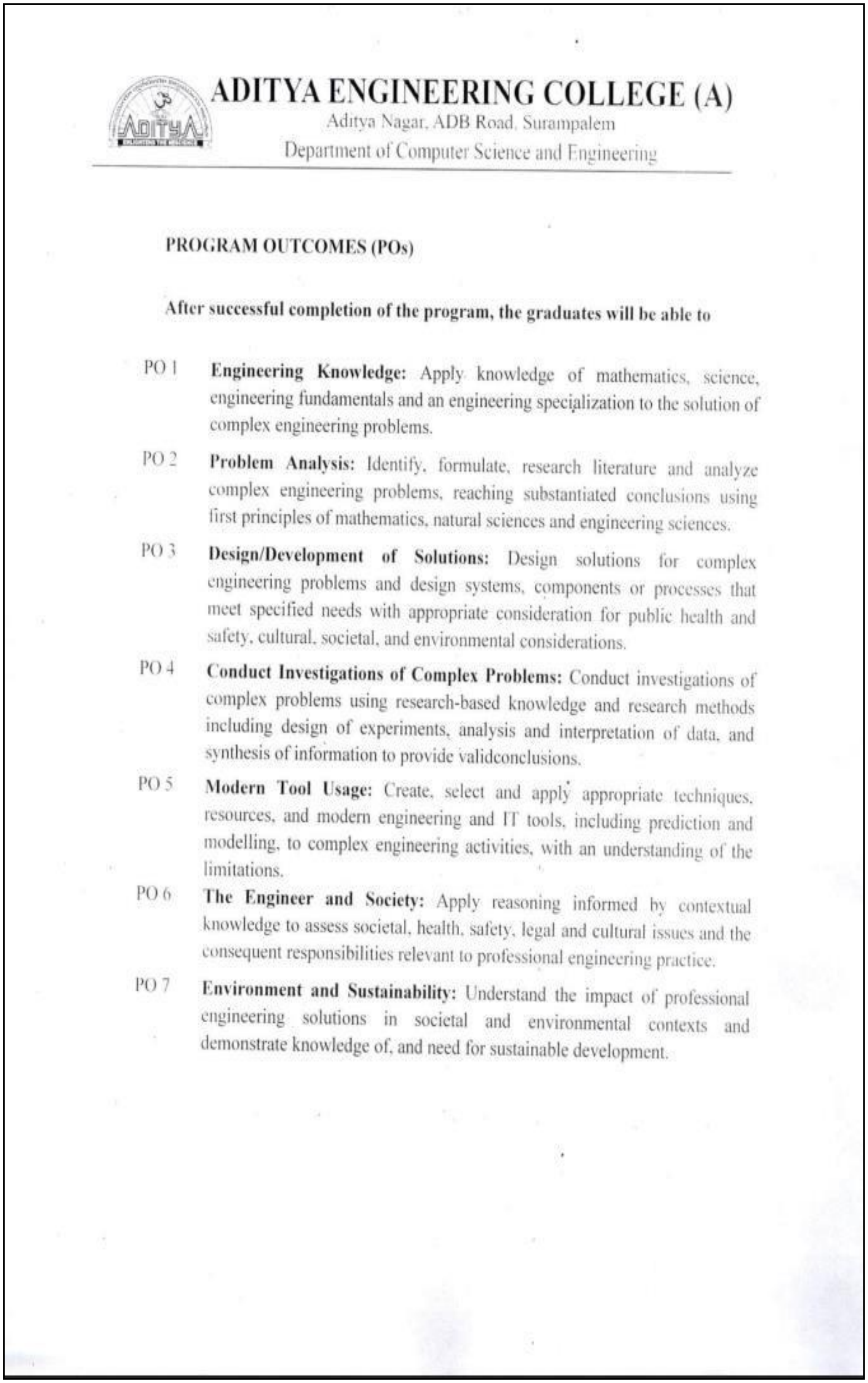
To emerge as a competent Centre of excellence in the field of Computer Science and Engineering for industry and societal needs.

MISSION:

- Impart quality and value based education.
- Inculcate the inter personal skills and professional ethics.
- Enable research through state-of-the-art infrastructure.
- Collaborate with industries, government and professional societies.

Head of the Department

Head of the Department
Department of CSE
ADITYA ENGINEERING COLLEGE (A



ADITYA ENGINEERING COLLEGE (A)
Aditya Nagar, ADB Road, Surampalem
Department of Computer Science and Engineering

PROGRAM OUTCOMES (POs)

After successful completion of the program, the graduates will be able to

PO 1 **Engineering Knowledge:** Apply knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.

PO 2 **Problem Analysis:** Identify, formulate, research literature and analyze complex engineering problems, reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences.

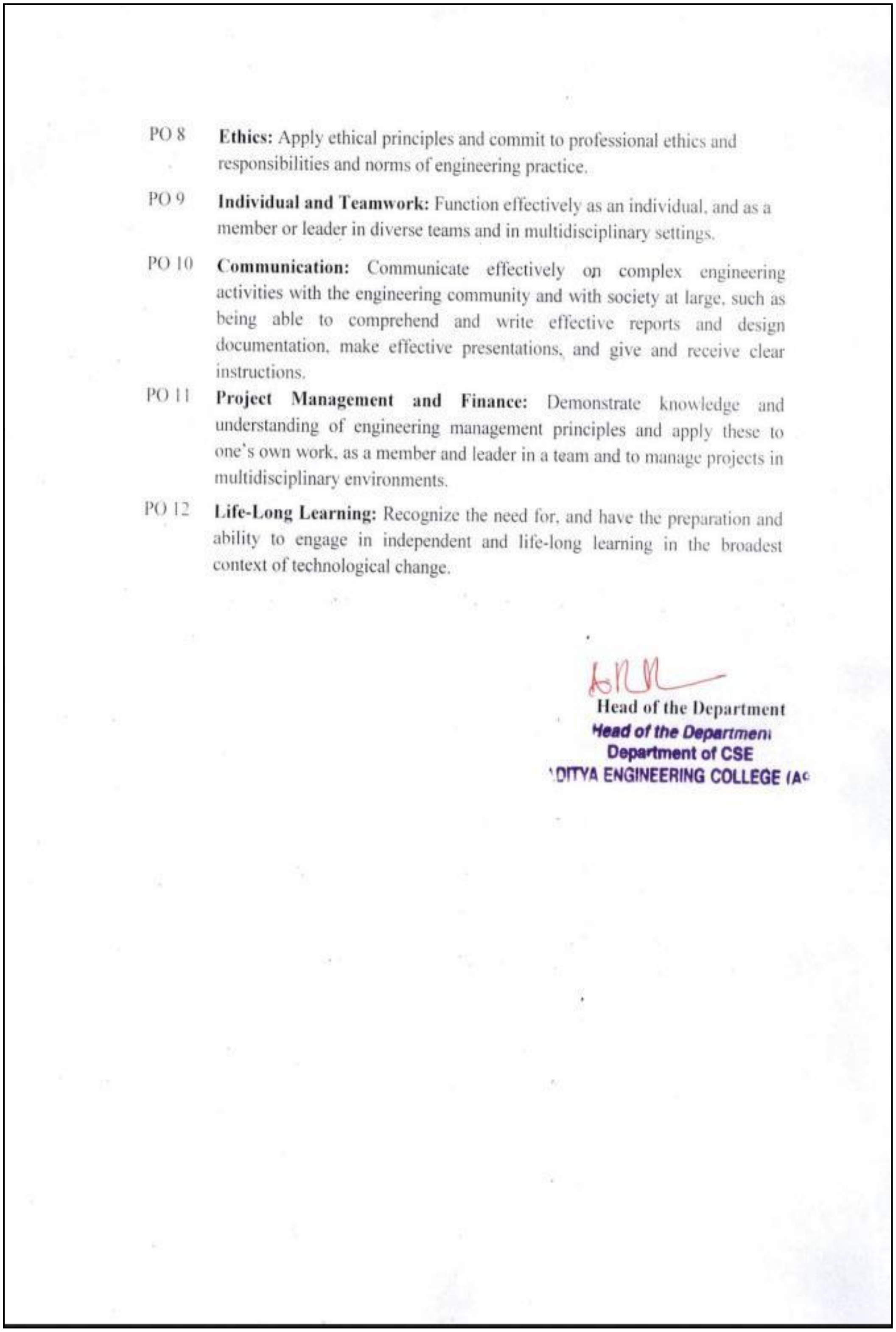
PO 3 **Design/Development of Solutions:** Design solutions for complex engineering problems and design systems, components or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.

PO 4 **Conduct Investigations of Complex Problems:** Conduct investigations of complex problems using research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of information to provide validconclusions.

PO 5 **Modern Tool Usage:** Create, select and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modelling, to complex engineering activities, with an understanding of the limitations.

PO 6 **The Engineer and Society:** Apply reasoning informed by contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to professional engineering practice.

PO 7 **Environment and Sustainability:** Understand the impact of professional engineering solutions in societal and environmental contexts and demonstrate knowledge of, and need for sustainable development.



PO 8 **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of engineering practice.

PO 9 **Individual and Teamwork:** Function effectively as an individual, and as a member or leader in diverse teams and in multidisciplinary settings.

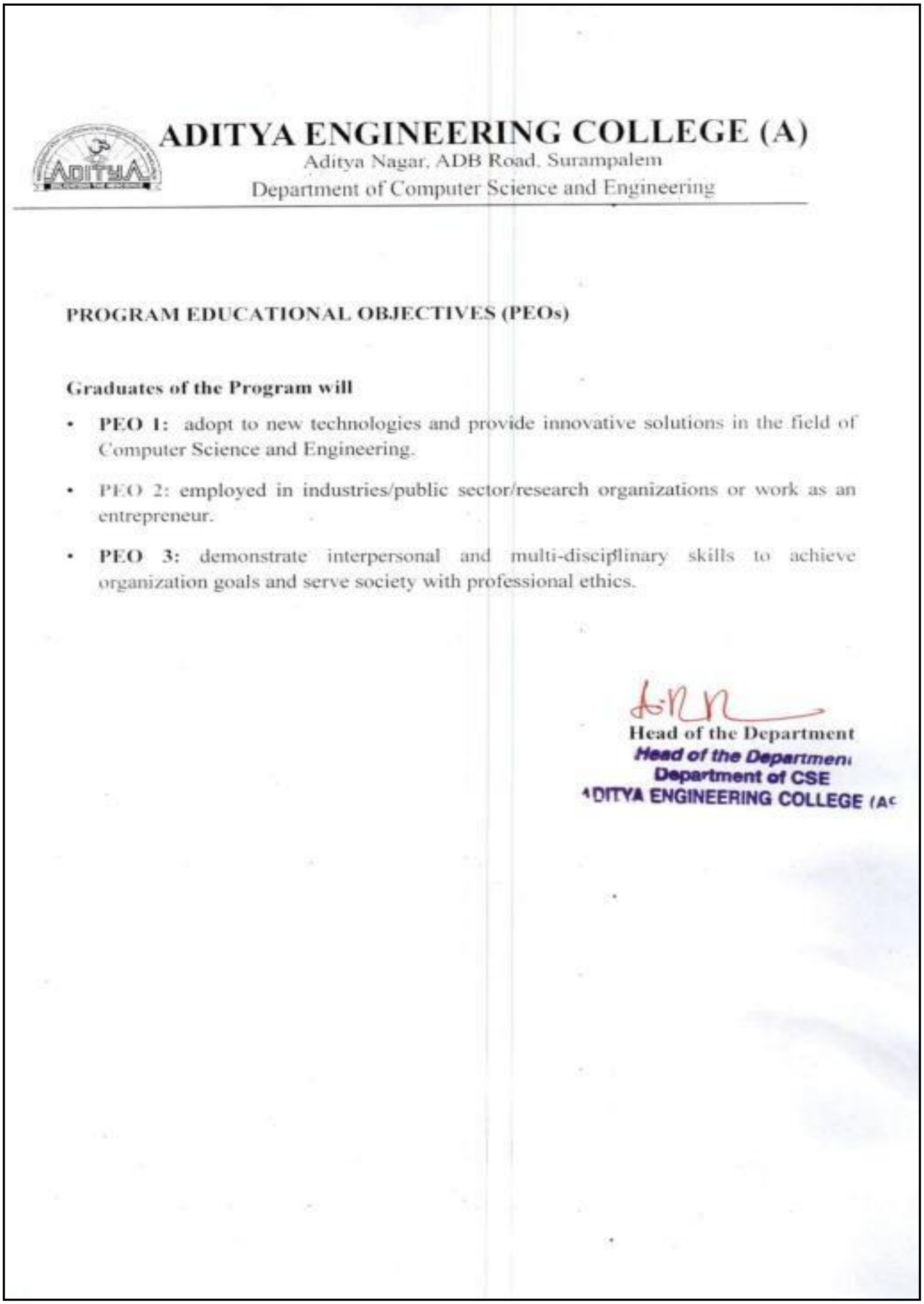
PO 10 **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO 11 **Project Management and Finance:** Demonstrate knowledge and understanding of engineering management principles and apply these to one's own work, as a member and leader in a team and to manage projects in multidisciplinary environments.

PO 12 **Life-Long Learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Head of the Department

Head of the Departmen
Department of CSE
'DITYA ENGINEERING COLLEGE (A



ADITYA ENGINEERING COLLEGE (A)
Aditya Nagar, ADB Road, Surampalem
Department of Computer Science and Engineering

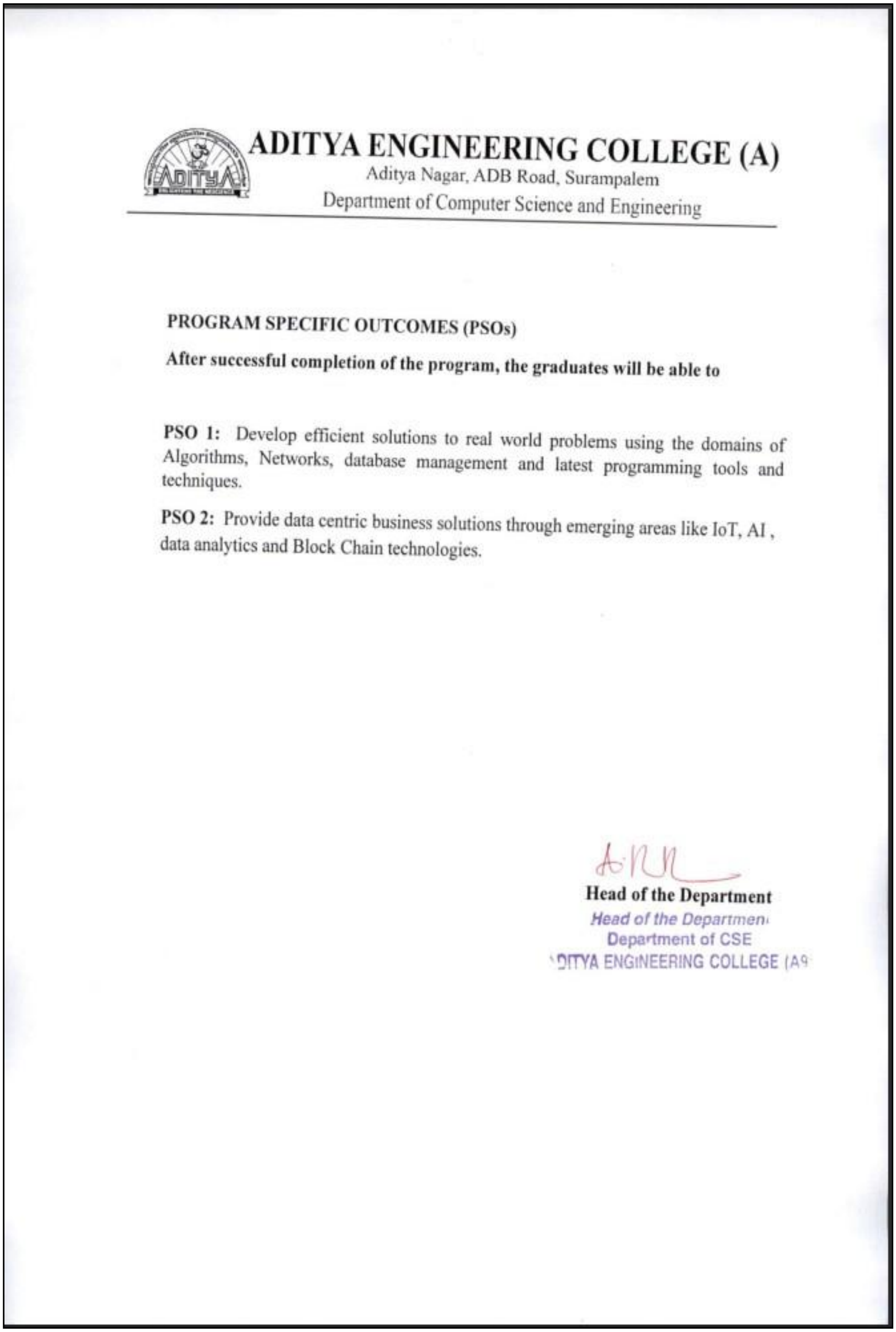
PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

Graduates of the Program will

- **PEO 1:** adopt to new technologies and provide innovative solutions in the field of Computer Science and Engineering.
- PEO 2: employed in industries/public sector/research organizations or work as an entrepreneur.
- **PEO 3:** demonstrate interpersonal and multi-disciplinary skills to achieve organization goals and serve society with professional ethics.

Head of the Department

Head of the Departmen
Department of CSE
ADITYA ENGINEERING COLLEGE (A



ADITYA ENGINEERING COLLEGE (A)

Aditya Nagar, ADB Road, Surampalem
Department of Computer Science and Engineering

PROGRAM SPECIFIC OUTCOMES (PSOs)

After successful completion of the program, the graduates will be able to

PSO 1: Develop efficient solutions to real world problems using the domains of Algorithms, Networks, database management and latest programming tools and techniques.

PSO 2: Provide data centric business solutions through emerging areas like IoT, AI , data analytics and Block Chain technologies.

Head of the Department

Head of the Departmen
Department of CSE
DITYA ENGINEERING COLLEGE (A9

ABSTRACT

Early diagnosis and characterization are the important components in determining the treatment of chronic kidney disease (CKD). CKD is an ailment which tends to damage the kidney and affect their effective functioning of excreting waste and balancing body fluids. Some of the complications included are hypertension, anemia (low blood count), mineral bone disorder, poor nutritional health, acid base abnormalities, and neurological complications. Early and error-free detection of CKD can be helpful in averting further deterioration of patient's health. These chronic diseases are prognosticated using various types of data mining classification approaches and machine learning (ML) algorithms. This Prediction is performed using Logistic Regression, KNN, Random Forest, SVM, Xgboost, Adaboost. The data used is collected from the UCI Repository with 400 data sets with 21 attributes. This data has been fed into Classification algorithms. The experimental results show that DT, RF, Gradient Boosting hands out an accuracy of 98.75%, 98.75% and 97.50% respectively. The Xgboost and Adaboost classifier gives out a maximum accuracy of 100%.

Chronic kidney disease (CKD) is a global health problem with high morbidity and mortality rate, and it induces other diseases. Since there are no obvious symptoms during the early stages of CKD, patients often fail to notice the disease. Early detection of CKD enables patients to receive timely treatment to ameliorate the progression of this disease. Machine learning models can effectively aid clinicians achieve this goal due to their fast and accurate recognition performance. In this study, we propose a machine learning methodology for diagnosing CKD. The CKD data set was obtained from the University of California Irvine (UCI) machine learning repository, which has many missing values. KNN imputation was used to fill in the missing values, which selects several complete samples with the most similar measurements to process the missing data for each incomplete sample. Missing values are usually seen in real-life medical situations because patients may miss some measurements for various reasons. After effectively filling out the incomplete data set, six machine learning algorithms (logistic regression, random forest, support vector machine, k-nearest neighbor, Xgboost, Adaboost) were used to establish models.

INDEX Page No

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 Introduction of Project area/Domain

What is Machine Learning:

Machine learning is often categorized as a subfield of artificial intelligence, but I find that categorization can often be misleading at first brush. The study of machine learning certainly arose from research in this context, but in the data science application of machine learning methods, it is more helpful to think of machine learning as a means of *building models of data*.

Fundamentally, machine learning involves building mathematical models to help understand data. "Learning" enters the fray when we give these models *tunable parameters* that can be adapted to observed data; in this way the program can be "learning" from the data. Once these models have been fit to previously seen data, they can be used to predict and understand aspects of newly observed data. I'll leave to the reader the more philosophical digression regarding the extent to which this type of mathematical, model-based "learning" is similar to the "learning" exhibited by the human brain. Understanding the problem setting in machine learning is essential to using these tools effectively, and so we will start with some broad categorizations of the types of approaches we'll discuss here.

Categories Of Machine Leaning

At the most fundamental level, machine learning can be categorized into two main types: supervised learning, unsupervised learning.

Supervised learning involves somehow modeling the relationship between measured features of data and some label associated with the data; once this model is determined, it can be used to apply labels to new, unknown data. This is further subdivided into *classification* tasks and *regression* tasks: in classification, the labels are discrete categories, while in regression, the labels are continuous quantities. We will see examples of both types of supervised learning in the following section.

Unsupervised learning involves modeling the features of a dataset without reference to any label, and is often described as "letting the dataset speak for itself." These models include tasks such as *clustering* and *dimensionality reduction.* Clustering algorithms identify distinct groups of data, while dimensionality reduction algorithms search for more succinct representations of the data. We will see examples of both types of unsupervised learning in the following section.

Need for Machine Learning

Human beings, at this moment, are the most intelligent and advanced species on earth because they can think, evaluate, and solve complex problems. On the other side, AI is still in its initial stage and have not surpassed human intelligence in many aspects. Then the question is that what is the need to make machine learn? The most suitable reason for doing this is, “to make decisions, based on data, with efficiency and scale”.

Lately, organizations are investing heavily in newer technologies like Artificial Intelligence, Machine Learning and Deep Learning to get the key information from data to perform several real-world tasks and solve problems. We can call it data-driven decisions taken by machines, particularly to automate the process. These data-driven decisions can be used, instead of using programing logic, in the problems that cannot be programmed inherently. The fact is that we can’t do without human intelligence, but other aspect is that we all need to solve real-world problems with efficiency at a huge scale. That is why the need for machine learning arises.

Applications of Machines Learning:

Machine Learning is the most rapidly growing technology and according to researchers we are in the golden year of AI and ML. It is used to solve many real-world complex problems which cannot be solved with traditional approach. Following are some real-world applications of ML

- Emotion analysis
- Sentiment analysis
- Error detection and prevention
- Weather forecasting and prediction
- Stock market analysis and forecasting
- Speech synthesis
- Speech recognition
- Customer segmentation

- Object recognition
- Fraud detection
- Fraud prevention
- Recommendation of products to customer in online shopping

(a) Terminologies of Machine Learning

- **Model –** A model is a specific representation learned from data by applying some machine learning algorithm. A model is also called a hypothesis.
- **Feature –** A feature is an individual measurable property of the data. A set of numeric features can be conveniently described by a feature vector. Feature vectors are fed as input to the model. For example, in order to predict a fruit, there may be features like color, smell, taste, etc.
- **Target (Label) –** A target variable or label is the value to be predicted by our model. For the fruit example discussed in the feature section, the label with each set of input would be the name of the fruit like apple, orange, banana, etc.
- **Training –** The idea is to give a set of inputs(features) and it’s expected outputs(labels), so after training, we will have a model (hypothesis) that will then map new data to one of the categories trained on.
- **Prediction –** Once our model is ready, it can be fed a set of inputs to which it will provide a predicted output(label).

(b) Types of Machine Learning

- **Supervised Learning –** This involves learning from a training dataset with labeled data using classification and regression models. This learning process continues until the required level of performance is achieved.
- **Unsupervised Learning –** This involves using unlabeled data and then finding the underlying structure in the data in order to learn more and more about the data itself using factor and cluster analysis models.
- **Semi-supervised Learning –** This involves using unlabeled data like Unsupervised Learning with a small amount of labeled data. Using labeled data vastly increases the learning accuracy and is also more cost-effective than Supervised Learning.

- **Reinforcement Learning –** This involves learning optimal actions through trial and error. So the next action is decided by learning behaviors that are based on the current state and that will maximize the reward in the future.

Advantages of Machine learning:

1. Easily identifies trends and patterns -

Machine Learning can review large volumes of data and discover specific trends and patterns that would not be apparent to humans. For instance, for an e-commerce website like Amazon, it serves to understand the browsing behaviors and purchase histories of its users to help cater to the right products, deals, and reminders relevant to them. It uses the results to reveal relevant advertisements to them.

2. No human intervention needed (automation)

With ML, you don't need to babysit your project every step of the way. Since it means giving machines the ability to learn, it lets them make predictions and also improve the algorithms on their own. A common example of this is anti-virus software's; they learn to filter new threats as they are recognized. ML is also good at recognizing spam.

3. Continuous Improvement

As **ML algorithms** gain experience, they keep improving in accuracy and efficiency. This lets them make better decisions. Say you need to make a weather forecast model. As the amount of data, you have keeps growing, your algorithms learn to make more accurate predictions faster.

4. Handling multi-dimensional and multi-variety data

Machine Learning algorithms are good at handling data that are multi-dimensional and multi-variety, and they can do this in dynamic or uncertain environments.

5. Wide Applications

You could be an e-tailer or a healthcare provider and make ML work for you. Where it does apply, it holds the capability to help deliver a much more personal experience to customers while also targeting the right customers.

Disadvantages of Machine Learning:

1. Data Acquisition

Machine Learning requires massive data sets to train on, and these should be inclusive/unbiased, and of good quality. There can also be times where they must wait for new data to be generated.

2. Time and Resources

ML needs enough time to let the algorithms learn and develop enough to fulfill their purpose with a considerable amount of accuracy and relevancy. It also needs massive resources to function. This can mean additional requirements of computer power for you.

3. Interpretation of Results

Another major challenge is the ability to accurately interpret results generated by the algorithms. You must also carefully choose the algorithms for your purpose.

4. High error-susceptibility

Machine Learning is autonomous but highly susceptible to errors. Suppose you train an algorithm with data sets small enough to not be inclusive. You end up with biased predictions coming from a biased training set. This leads to irrelevant advertisements being displayed to customers. In the case of ML, such blunders can set off a chain of errors that can go undetected for long periods of time. And when they do get noticed, it takes quite some time to recognize the source of the issue, and even longer to correct it.

1.2 Existing System and Disadvantages

- ❖ Hornelen et al. utilized image registration to detect renal morphologic changes.
- ❖ Vasquez-Morales et al. established a classifier based on neural network using large-scale CKD data, and the accuracy of the model on their test data was 95%. In addition, most of the previous studies utilized the CKD data set that was obtained from the UCI machine learning repository.
- ❖ Chen et al. used k-nearest neighbor (KNN), support vector machine (SVM) and soft independent modeling of class analogy to diagnose CKD, KNN and SVM achieved the highest accuracy of 99.7%. In addition, they used fuzzy rule-building expert system, fuzzy

optimal associative memory and partial least squares discriminant analysis to diagnose CKD, and the range of accuracy in those models was 95.5%-99.6%. Their studies have achieved good results in the diagnosis of CKD.

Disadvantages:

- Most of them suffering from either the method used to impute missing values has a limited application range or relatively low accuracy.
- In the above models, the mean imputation is used to fill in the missing values and it depends on the diagnostic categories of the samples. As a result, their method could not be used when the diagnostic results of the samples are unknown. In reality, patients might miss some measurements for various reasons before diagnosing.
- In addition, for missing values in categorical variables, data obtained using mean imputation might have a large deviation from the actual values.

1.3 Proposed System and Advantages

- Early and error-free detection of CKD can be helpful in averting further deterioration of patient's health. These chronic diseases are prognosticated using various types of data mining classification approaches and machine learning (ML) algorithms. This Prediction is performed using Logistic Regression, KNN, Random Forest, Decision Tree, SVM, Gradient Boosting, Xgboost, Adaboost. The data used is collected from the UCI Repository with 400 data sets with 21 attributes. This data has been fed into Classification algorithms. The experimental results show that DT, RF, Gradient Boosting hands out an accuracy of 98.75%, 98.75% and 97.50% respectively. The Xgboost and Adaboost classifier gives out a maximum accuracy of 100%.

Advantages

- We propose a methodology to extend application range of the CKD diagnostic models.
- At the same time, the accuracy of the model is further improved.

1.4 Objectives of the Project

To summarize the previous CKD diagnostic models, we find that most of them suffering from either the method used to impute missing values has a limited application range or relatively low accuracy. Therefore, in this work, we propose a methodology to extend application range of the CKD diagnostic models. At the same time, the accuracy of the model is further improved. The contributions of the proposed work are as follows.

1) we used KNN imputation to fill in the missing values in the data set, which could be applied to the data set with the diagnostic categories are unknown.

2) Logistic regression (LOG), RF, SVM, KNN, naive Bayes classifier (NB) and feed forward neural network (FNN) were used to establish CKD diagnostic models on the complete CKD data sets. The models with better performance were extracted for misjudgment analysis.

3) An integrated model that combines LOG and RF by using perceptron was established and it improved the performance of the component models in CKD diagnosis after the missing values were filled by KNN imputation. KNN imputation is used to fill in the missing values. To our knowledge, this is the first time that KNN imputation has been used for the diagnosis of CKD. In addition, building an integrated model is also a good way to improve the performance of separate individual models.

4) The proposed methodology might effectively deal with the scene where patients are missing certain measurements before being diagnosed. In addition, the resulting integrated model shows a higher accuracy. Therefore, it is speculated that this methodology might be applicable to the clinical data in the actual medical diagnosis.

1.5 Organization of Project

The Organization of the Project covers the complete synopsis of the procedure that is followed in this project and which are partitioned into chapters with respect to the application of the methods in the process. These chapters are described in brief along with their contents, which are as follows.

Chapter 1: In this chapter, we have introduced about machine learning, python, and advantages along with some process of its working

- We have also discussed about existing system and its disadvantages and proposed system and its advantages.
- Objectives of the project are also discussed.

Chapter 2: In this, we discussed about the necessary requirements we used in building this project along with its specific features and uses.

- Modules and frameworks used in this project are discussed.

- Hardware and software requirements along with its specification are discussed.

Chapter 3: In this, we discussed about all the necessary research papers on earthquake and discussed the methods used and accuracy gained along with places used for developing this project.

Chapter 4: In this chapter different modules that are obtained by dividing the project and implementation of each module is discussed.

Chapter 5: In this chapter the designing of the system for making predictions is discussed along with the algorithm and steps followed. In order to understand it clearly UML diagrams that support higher level development concepts such as collaborations, frameworks, patterns, and components are also drawn.

Chapter 6: This Chapter shows how our project is implemented and the selected software for completing the project and its Sample code.

Chapter 7: The various possible test cases that are possible for this are generated and displayed in this chapter.

Chapter 8: Here, the various output that are generated are displayed with proof i.e., along with appropriate screenshots and images regarding the working of the project.

Chapter 9: In this, we conclude by once again giving a quick summary of the working of this project along with some details about its future scope.

Chapter 10: Finally, all the references are mentioned here that means the books and papers that we have done research on for developing this project.

2. REQUIREMENTS ANALYSIS

2.1 Introduction of Requirement Analysis:

- Good requirements are essential for executing projects.
- Improperly understood or documented requirements lead to cost escalations, late delivery, and poor quality, in short leads to dissatisfied customers.
- Two major activities are requirement analysis and specification and requirements change management.
- Requirement specification activity is done at start of the project and change management is done throughout the project.
- Requirement traceability is another activity that aims to ensure that all requirements can be traced to elements in the outputs produced in the later stages of the project. The goal is to ensure that final software satisfies the customer's requirements is met.

- Good requirements are essential for executing projects.
- Improperly understood or documented requirements lead to cost escalations, late delivery, and poor quality, in short leads to dissatisfied customers.
- Two major activities are requirement analysis and specification and requirements change management.
- Requirement specification activity is done at start of the project and change management is done throughout the project.
- Requirement traceability is another activity that aims to ensure that all requirements can be traced to elements in the outputs produced in the later stages of the project. The goal is to ensure that final software satisfies the customer's requirements is met.

2.2 HARDWARE AND SOFTWARE REQUIREMENTS

SOFTWARE REQUIREMENTS

The functional requirements or the overall description documents include the product perspective and features, operating system and operating environment, graphics requirements, design constraints and user documentation. The appropriation of requirements and implementation constraints gives the general overview of the project in regards to what the areas of strength and deficit are and how to tackle them.

- **Python idle 3.7**

- **Anaconda 3.7**
- **Jupiter**

HARDWARE REQUIREMENTS

Minimum hardware requirements are very dependent on the software being developed by a given Enthought Python / Canopy / VS Code user. Applications that need to store large arrays/objects in memory will require more RAM, whereas applications that need to perform numerous calculations or tasks more quickly will require a faster processor.

- **Operating system : windows, Linux**
- **Processor : minimum intel i3**
- **Ram : minimum 4 gb**
- **Hard disk : minimum 250gb**

2.3 SOFTWARE REQUIREMENT SPECIFICATION

2.3.1 FUNCTIONAL REQUIREMENTS

1.Data Collection

2.Data Pre-processing

3.Training and Testing

4.Modiling

5.Predicting

2.3.2 NON-FUNCTIONAL REQUIREMENTS

NON-FUNCTIONAL REQUIREMENT (NFR) specifies the quality attribute of a software system. They judge the software system based on Responsiveness, Usability, Security, Portability and other non-functional standards that are critical to the success of the software system. Example of nonfunctional requirement, *"how fast does the website load?"* Failing to meet non-functional requirements can result in systems that fail to satisfy user needs. Non- functional Requirements allows you to impose constraints or restrictions on the design of the system across the various agile backlogs. Example, the site should load in 3 seconds when the number of simultaneous users are > 10000. Description of non-functional requirements is just as critical as a functional requirement.

- Usability requirement

- Serviceability requirement
- Manageability requirement
- Recoverability requirement
- Security requirement
- Data Integrity requirement
- Capacity requirement
- Availability requirement
- Scalability requirement
- Interoperability requirement
- Reliability requirement
- Maintainability requirement
- Regulatory requirement
- Environmental requirement

SYSTEM STUDY

FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

♦ ECONOMICAL FEASIBILITY

♦ TECHNICAL FEASIBILITY

♦ SOCIAL FEASIBILITY

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

3. LITERATURE SURVEY

3.1) Diagnosis of patients with chronic kidney disease by using two fuzzy classifiers

AUTHORS: Z. Chen, Z. Zhang, R. Zhu, Y. Xiang, and P. B. Harrington

ABSTRACT:

The feasibility of two in-house fuzzy classifiers, fuzzy rule-building expert system (FuRES) and fuzzy optimal associative memory (FOAM), for diagnosis of patients with chronic kidney disease (CKD) was investigated. A linear classifier, partial least squares discriminant analysis (PLS-DA), was used for comparison. The CKD data used in this work were taken from the UCI Machine Learning Repository. Composite datasets were created by adding different levels of proportional noise to evaluate the robustness of the two fuzzy approaches. Firstly, 11 levels of proportional noises were added to each numeric attribute of the training and prediction sets one after another, and then these simulated training and prediction sets were combined in pairs. Thus, a grid with 121 groups of simulated data was generated, and classification rates for these 121 pairs were compared. Secondly, the performances of two fuzzy classifiers using the simulated datasets, in which 11 levels of noise were randomly distributed to each numeric attribute, were compared and the average prediction rates of FuRES and FOAM were 98.1 ± 0.5% and 97.2 ± 1.2%, respectively, with 200 bootstrap Latin partitions. The PLS-DA can give 94.3 ± 0.8% with the identical evaluation. Confluent datasets comprised of the original and modified datasets were also used to evaluate FuRES, FOAM, and PLS-DA classification models. The average prediction rates of FuRES and FOAM obtained from 200 bootstrapped evaluations were 99.2 ± 0.3% and 99.0 ± 0.3%. PLS-DA yields slightly worse accuracy with 95.9 ± 0.6%. The results demonstrate that both FuRES and FOAM perform well on the identification of CKD patients, while FuRES is more robust than FOAM. These two fuzzy classifiers are useful tools for the diagnosis of CKD patients with satisfactory robustness, and can also be used for other kinds of patients.

3.2) Diagnosis of chronic kidney disease by using random forest

AUTHORS: A. Subasi, E. Alickovic, and J. Kevric

ABSTRACT:

Chronic kidney disease (CKD) is a global public health problem, affecting approximately 10% of the population worldwide. Yet, there is little direct evidence on how CKD can be diagnosed in a systematic and automatic manner. This paper investigates how CKD can be diagnosed by using machine learning (ML) techniques. ML algorithms have been a driving force in detection of abnormalities in different physiological data, and are, with a great success, employed in different classification tasks. In the present study, a number of different ML classifiers are experimentally validated to a real data set, taken from the UCI Machine Learning Repository, and our findings are compared with the findings reported in the recent literature. The results are quantitatively and qualitatively discussed and our findings reveal that the random forest (RF) classifier achieves the near-optimal performances on the identification of CKD subjects. Hence, we show that ML algorithms serve important function in diagnosis of CKD, with satisfactory robustness, and our findings suggest that RF can also be utilized for the diagnosis of similar diseases.

3.3) Prevalence of chronic kidney disease in China: A cross-sectional survey

AUTHORS: L. Zhang

ABSTRACT:

Background: The prevalence of chronic kidney disease is high in developing countries. However, no national survey of chronic kidney disease has been done incorporating both estimated glomerular filtration rate (eGFR) and albuminuria in a developing country with the economic diversity of China. We aimed to measure the prevalence of chronic kidney disease in China with such a survey. Methods: We did a cross-sectional survey of a nationally representative sample of Chinese adults. Chronic kidney disease was defined as eGFR less than 60 mL/min per 1·73 m (2) or the presence of albuminuria. Participants completed a lifestyle and medical history questionnaire and had their blood pressure measured, and blood and urine samples taken. Serum creatinine was measured and used to estimate glomerular filtration rate. Urinary albumin and creatinine were tested to assess albuminuria. The crude and adjusted prevalence of indicators of kidney damage were calculated and factors associated with the presence of chronic kidney disease analyzed by logistic regression.

3.4) Incorporating temporal EHR data in predictive models for risk stratification of renal function deterioration

AUTHORS: A. Singh, G. Nadkarni, O. Gottesman, S. B. Ellis, E. P. Bottinger, andJ. V. Guttag

ABSTRACT:

Predictive models built using temporal data in electronic health records (EHRs) can potentially play a major role in improving management of chronic diseases. However, these data present a multitude of technical challenges, including irregular sampling of data and varying length of available patient history. In this paper, we describe and evaluate three different approaches that use machine learning to build predictive models using temporal EHR data of a patient. The first approach is a commonly used non-temporal approach that aggregates values of the predictors in the patient's medical history. The other two approaches exploit the temporal dynamics of the data. The two temporal approaches vary in how they model temporal information and handle missing data. Using data from the EHR of Mount Sinai Medical Center, we learned and evaluated the models in the context of predicting loss of estimated glomerular filtration rate (eGFR), the most common assessment of kidney function. Our results show that incorporating temporal information in patient's medical history can lead to better prediction of loss of kidney function. They also demonstrate that exactly how this information is incorporated is important. In particular, our results demonstrate that the relative importance of different predictors varies over time, and that using multi-task learning to account for this is an appropriate way to robustly capture the temporal dynamics in EHR data. Using a case study, we also demonstrate how the multi-task learning based model can yield predictive models with better performance for identifying patients at high risk of short-term loss of kidney function.

3.5) Prevalence of chronic kidney disease in an adult population

AUTHORS: A. M. Cueto-Manzano, L. Cortés-Sanabria, H. R. Martínez-Ramírez,
E. Rojas-Campos, B. Gómez-Navarro, and M. Castillero-Manzano

ABSTRACT:

Background and aims: One strategy to prevent and manage chronic kidney disease (CKD) is to offer screening programs. The aim of this study was to determine the percentage prevalence and risk factors of CKD in a screening program performed in an adult general population.

Methods: This is a cross-sectional study. Six-hundred ten adults (73% women, age 51 ± 14 years) without previously known CKD were evaluated. Participants were subjected to a questionnaire, blood pressure measurement and anthropometry. Glomerular filtration rate estimated by CKD-EPI formula and urine tested with albuminuria dipstick.

Results: More than 50% of subjects reported family antecedents of diabetes mellitus (DM), hypertension and obesity, and 30% of CKD. DM was self-reported in 19% and hypertension in 29%. During screening, overweight/obesity was found in 75%; women had a higher frequency of obesity (41 vs. 34%) and high-risk abdominal waist circumference (87 vs. 75%) than men. Hypertension (both self-reported and diagnosed in screening) was more frequent in men (49%) than in women (38%). CKD was found in 14.7%: G1, 5.9%; G2, 4.5%; G3a, 2.6%; G3b, 1.1%, G4, 0.3%; and G5, 0.3%. Glomerular filtration rate was mildly/moderately reduced in 2.6%, moderately/severely reduced in 1.1%, and severely reduced in <1%. Abnormal albuminuria was found in 13%. CKD was predicted by DM, hypertension, and male gender.

4. MODULES

4.1 Introduction to Modules

- Data Collection
- Dataset
- Data Preparation
- Model Selection index
- Analyze and Prediction
- Accuracy on test set
- Saving the Trained Model

4.2 Modules Implementation:

Data Collection:

This is the first real step towards the real development of a machine learning model, collecting data. This is a critical step that will cascade in how good the model will be, the more and better data that we get, the better our model will perform.

There are several techniques to collect the data, like web scraping, manual interventions etc.
The dataset used in this chronic kidney disease dataset taken from UCI: https://archive.ics.uci.edu/ml/datasets/chronic_kidney_disease

Dataset:

The dataset consists of 400 individual data. There are 26 columns in the dataset, which are described below.

age	-	age
bp	-	blood pressure
sg	-	specific gravity
al	-	albumin
su	-	sugar
rbc	-	red blood cells
pc	-	pus cell
pcc	-	pus cell clumps

ba	-	bacteria
bgr	-	blood glucose random
bu	-	blood urea
sc	-	serum creatinine
sod	-	sodium
pot	-	potassium
hemo	-	hemoglobin
pcv	-	packed cell volume
wc	-	white blood cell count
rc	-	red blood cell count
htn	-	hypertension
dm	-	diabetes mellitus
cad	-	coronary artery disease
appet	-	appetite
pe	-	pedal edema
ane	-	anemia
class	-	classification

Data Preparation:

we will transform the data. By getting rid of missing data and removing some columns. First, we will create a list of column names that we want to keep or retain. Next, we drop or remove all columns except for the columns that we want to retain. Finally, we drop or remove the rows that have missing values from the data set. Split into training and evaluation sets.

Model Selection:

It is a supervised learning algorithm that includes more dependent variables. The response of this algorithm is in the binary form. Logistics regression can provide the continuous outcome of a specific data. This algorithm consists of statistical model with binary variables.

Analyze and Prediction:

In the actual dataset, we chose only 19 features:

1. Age(numerical) --> age in years
2. Blood Pressure(numerical) bp in mm/Hg
3. Specific Gravity(nominal) sg - (1.005,1.010,1.015,1.020,1.025)
4. Albumin(nominal)al - (0,1,2,3,4,5)
5. Sugar(nominal) su - (0,1,2,3,4,5)
6. Blood Glucose Random(numerical) bgr in mgs/dl
7. Blood Urea(numerical) bu in mgs/dl
8. Serum Creatinine(numerical) sc in mgs/dl
9. Sodium(numerical) sod in mEq/L
10. Potassium(numerical) pot in mEq/L
11. Haemoglobin(numerical) hemo in gms
12. Packed Cell Volume(numerical)
13. White Blood Cell Count(numerical) wc in cells/cumm
14. Hypertension(nominal) htn - (yes,no)
15. Diabetes Mellitus(nominal) dm - (yes,no)
16. Coronary Artery Disease(nominal) cad - (yes,no)
17. Appetite(nominal) ppet - (good,poor)
18. Pedal Edema(nominal) pe - (yes,no)
19. Anemia(nominal)ane - (yes,no)
20. Chronic_kidney_disease: Displays whether the individual is suffering from kidney disease or not

Accuracy on test set

We got a accuracy of 100% on test set.

Saving the Trained Model

Once you are confident enough to take your trained and tested model into the production-ready environment, the first step is to save it into a .h5 or. pkl file using a library like `pickle`. Make sure you have `pickle` installed in your environment. Next, let us import the module and dump the model into. `pkl` file

5. SYSTEM DESIGN

5.1 Introduction to System Design

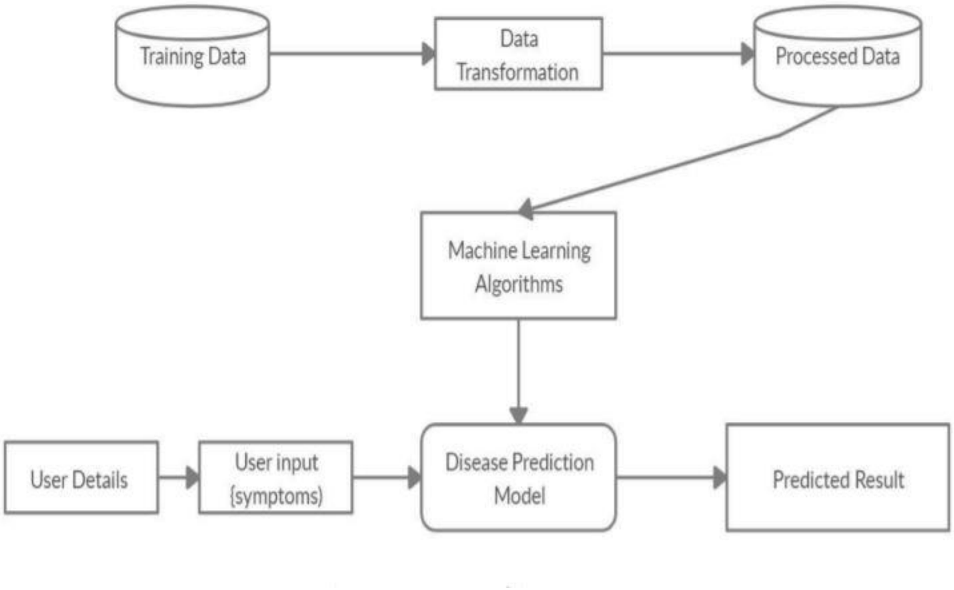


Fig 5.1 System Design for Predicting Result

5.2 DATA FLOW DIAGRAM:

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

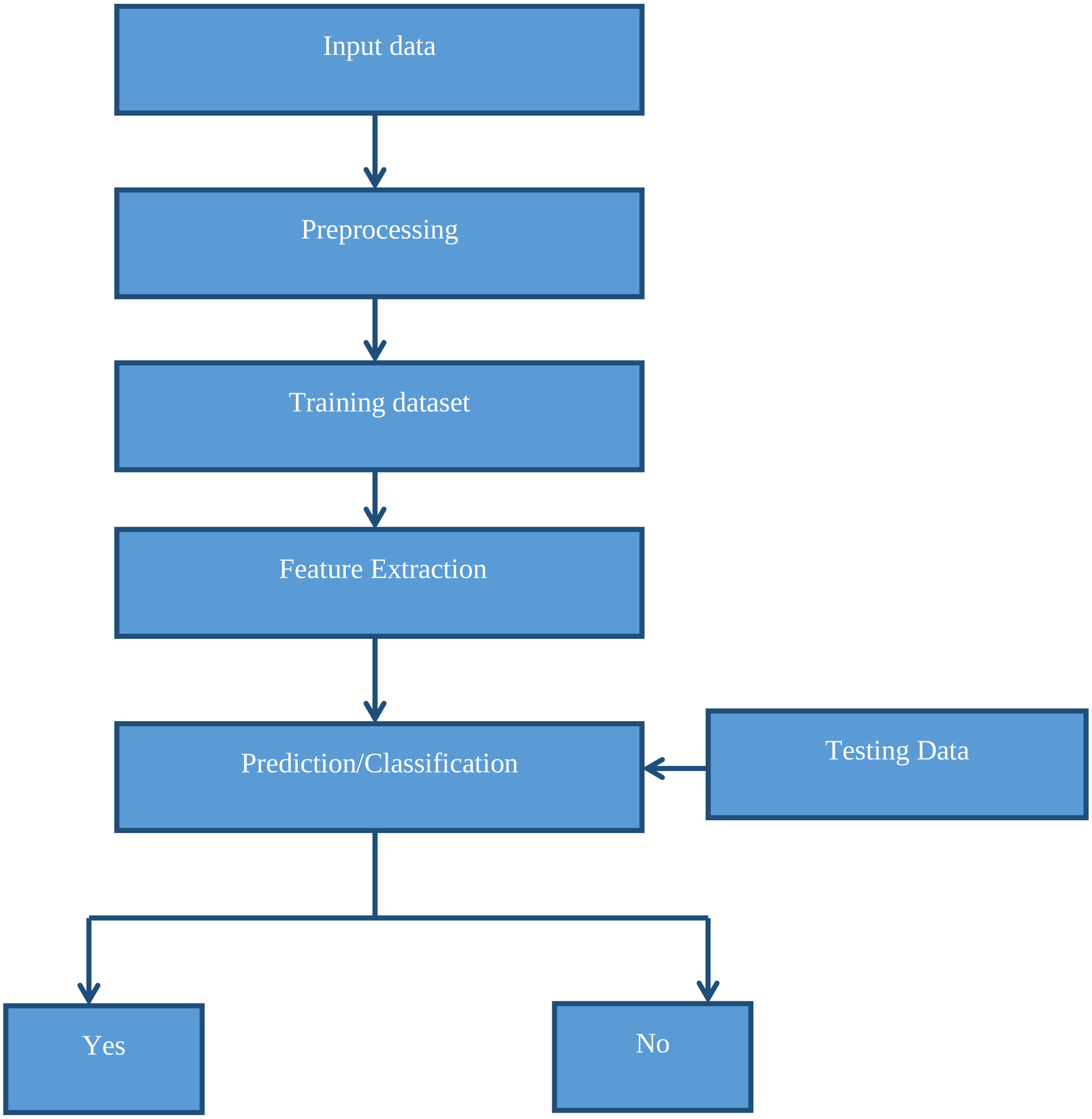


Fig 5.2 Diagrammatic Representation of the Data Flow

5.3 UML Diagrams

Use case diagram

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

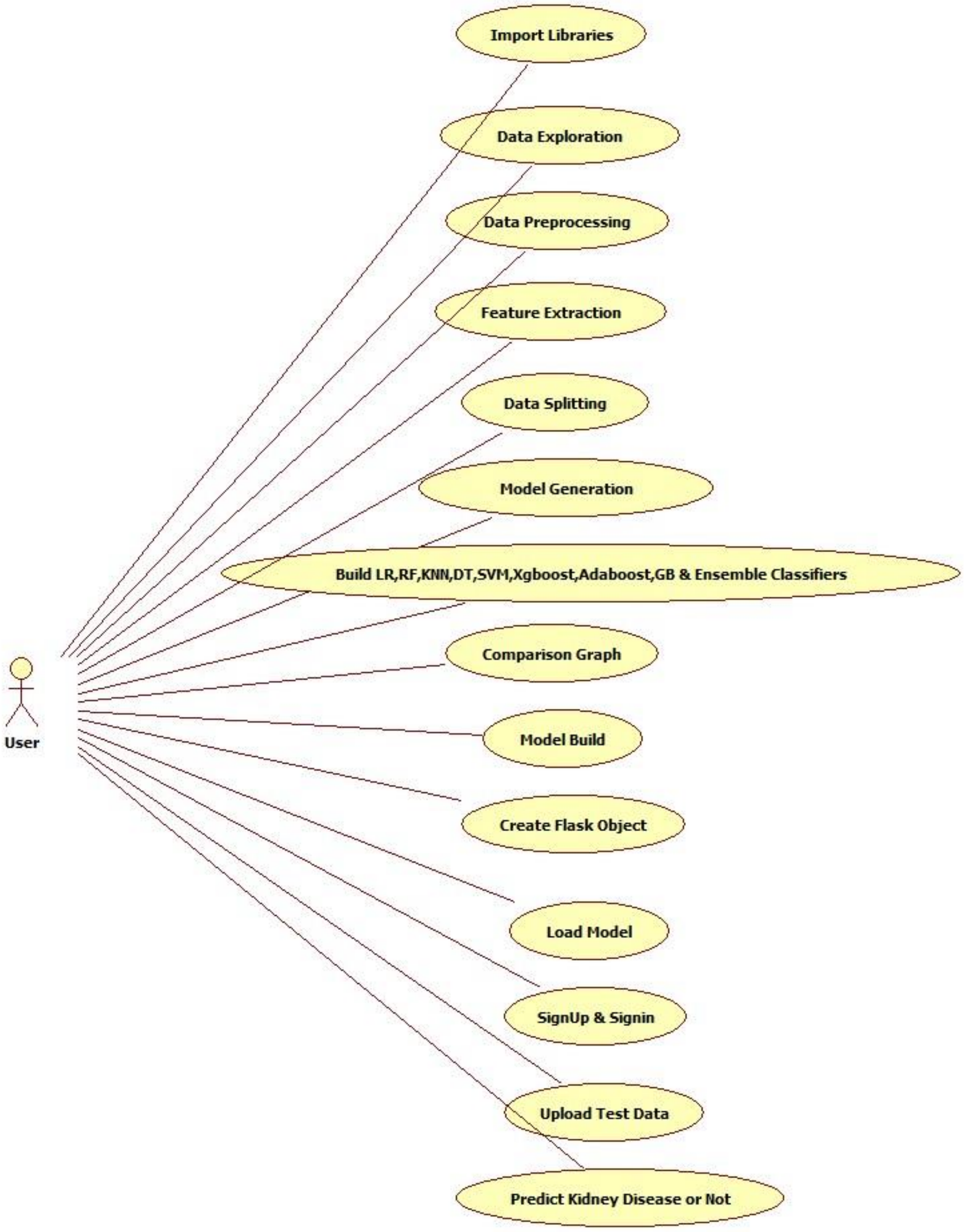


Fig 5.3 Usecase Diagram for Predicting Kidney Disease

Class diagram

The class diagram is used to refine the use case diagram and define a detailed design of the system. The class diagram classifies the actors defined in the use case diagram into a set of interrelated classes. The relationship or association between the classes can be either an "is-a" or "has-a" relationship. Each class in the class diagram may be capable of providing certain functionalities. These functionalities provided by the class are termed "methods" of the class. Apart from this, each class may have certain "attributes" that uniquely identify the class.

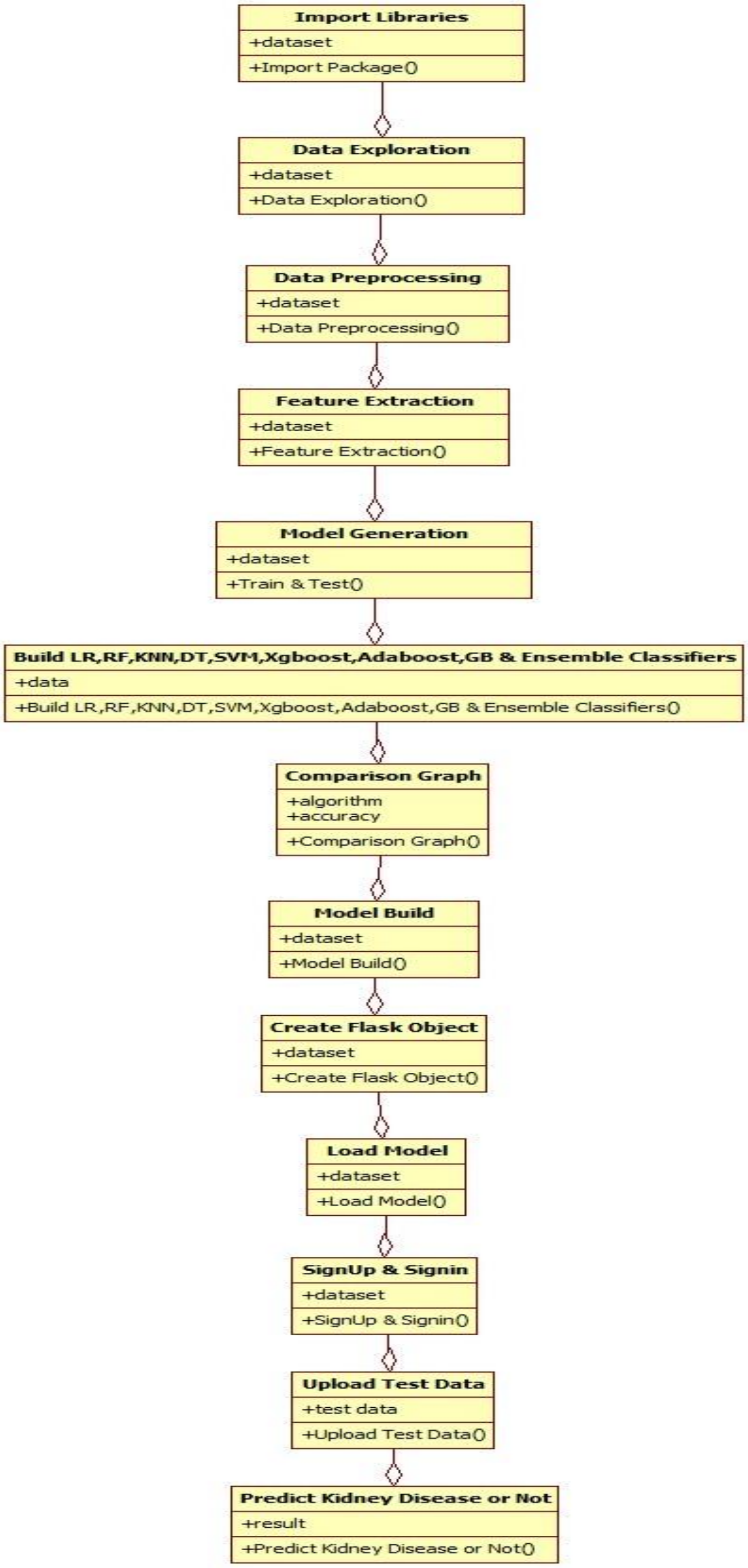


Fig 5.4 Class Diagram for Running Code using ML

Object diagram

The object diagram is a special kind of class diagram. An object is an instance of a class. This essentially means that an object represents the state of a class at a given point of time while the system is running. The object diagram captures the state of different classes in the system and their relationships or associations at a given point of time.

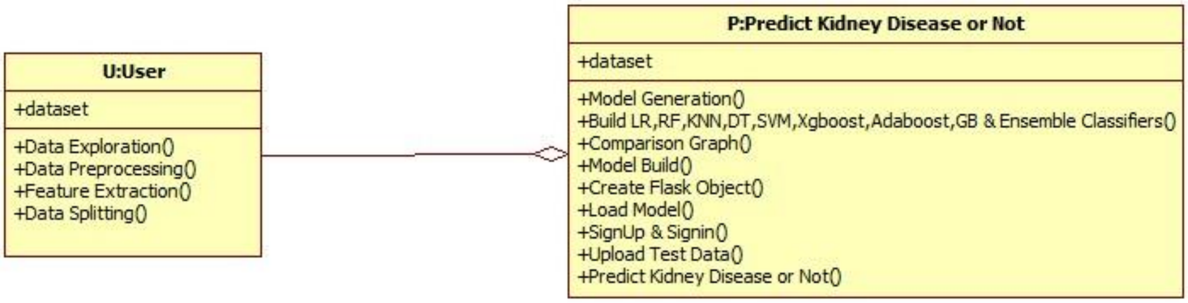


Fig 5.5 Object Diagram for Predicting Kidney Disease

State diagram:

A state diagram, as the name suggests, represents the different states that objects in the system undergo during their life cycle. Objects in the system change states in response to events. In addition to this, a state diagram also captures the transition of the object's state from an initial state to a final state in response to events affecting the system.

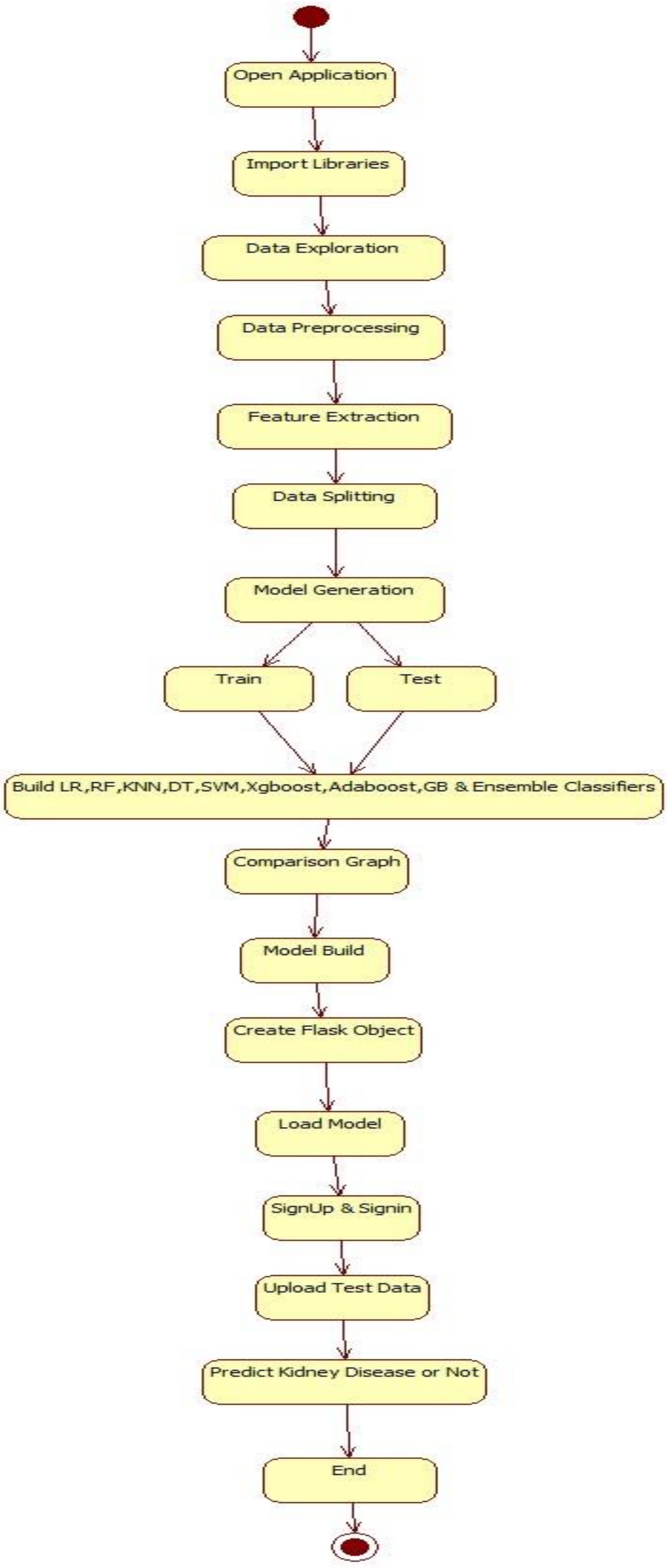


Fig 5.6 State Diagram for Predicting Kidney Disease using Data Visualization

Activity diagram

The process flows in the system are captured in the activity diagram. Similar to a state diagram, an activity diagram also consists of activities, actions, transitions, initial and final states, and guard conditions.

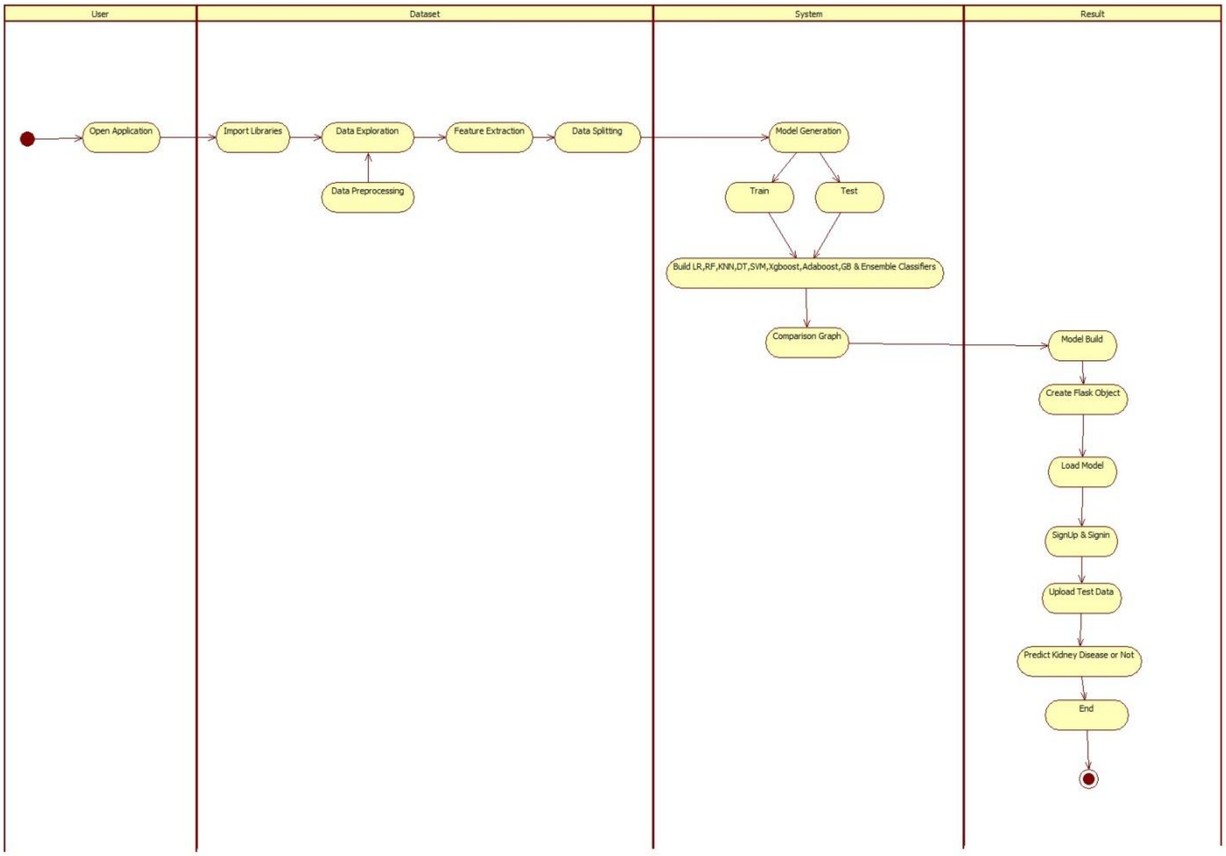


Fig 5.7 Activity Diagram for Web Development Stages

Sequence diagram

A sequence diagram represents the interaction between different objects in the system. The important aspect of a sequence diagram is that it is time-ordered. This means that the exact sequence of the interactions between the objects is represented step by step. Different objects in the sequence diagram interact with each other by passing "messages".

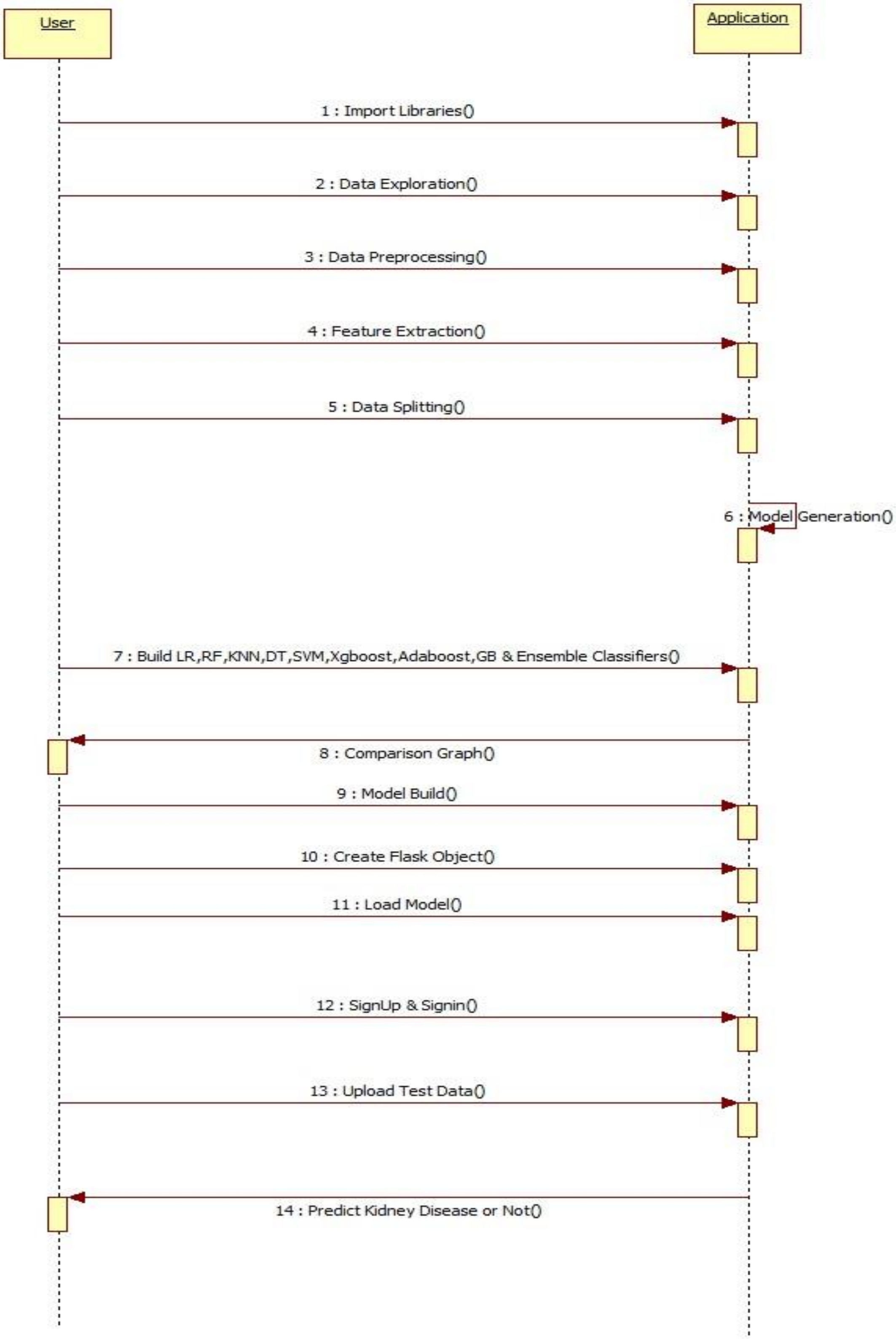


Fig 5.8 Sequence Diagram for Building Model to predict CKD

Collaboration diagram

A collaboration diagram groups together the interactions between different objects. The interactions are listed as numbered interactions that help to trace the sequence of the interactions. The collaboration diagram helps to identify all the possible interactions that each object has with other objects.

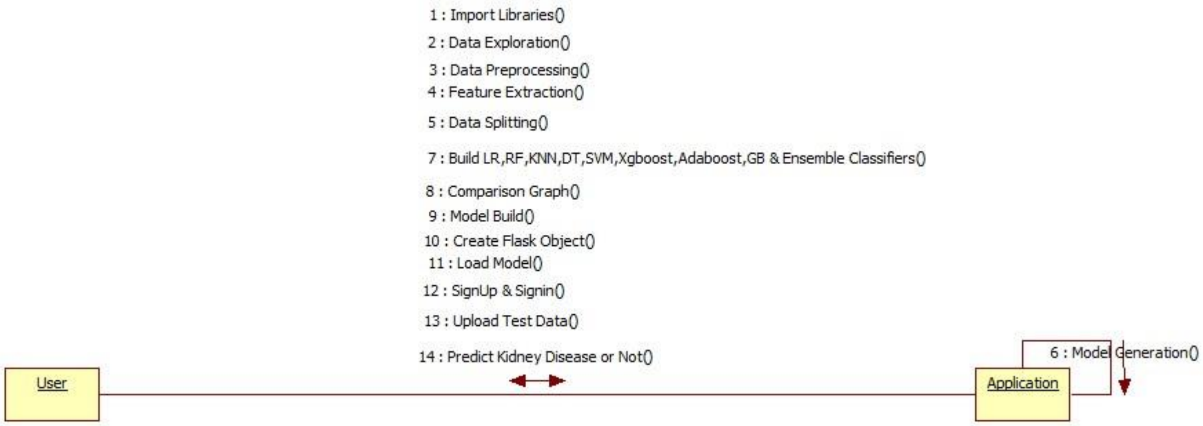


Fig 5.9 Collaboration Diagram for predicting CKD

6. SYSTEM IMPLEMENTATION

6.1 INTRODUCTION

Algorithms

Logistic Regression

The logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval, or ratio-level independent variables.

Random Forest Algorithm

Random Forest algorithm is a supervised classification algorithm. We can see it from its name, which is to create a forest by some way and make it random. There is a direct relationship between the number of trees in the forest and the results it can get: the larger the number of trees, the more accurate the result. But one thing to note is that creating the forest is not the same as constructing the decision with information gain or gain index approach. The decision tree is a decision support tool. It uses a tree-like graph to show the possible consequences. If you input a training dataset with targets and features into the decision tree, it will formulate some set of rules. These rules can be used to perform predictions. When we have our dataset categorized into 3 categories so now Random Forest helps to make classes from the dataset. Random forest is clusters of decision trees all together, if you input a training dataset with features and labels into a decision tree, it will formulate some set of rules, which will be used to make the predictions.

Random forests or random decision forests is an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees.It is called a Random Forest because we use Random subsets of data and features and we end up building a Forest of decision trees (many trees). Random Forest is also a classic example of a bagging approach as we use different subsets of data in each model to make predictions.

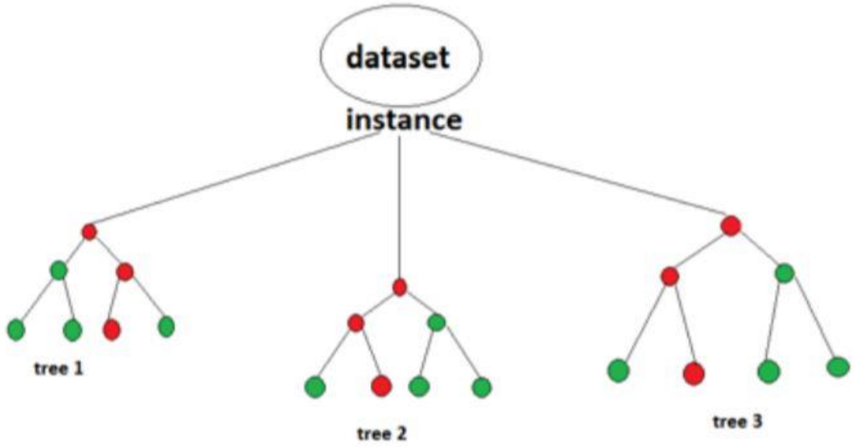


Fig 6.1 Illustration of Random Forest Algorithm

K NEAREST NEIGHBOR ALGORITHM

K-Nearest Neighbors is one of the most basic yet essential classification algorithms in Machine Learning. It belongs to the supervised learning domain and finds intense application in pattern recognition, data mining and intrusion detection. It is widely disposable in real-life scenarios since it is non-parametric, meaning, it does not make any underlying assumptions about the distribution of data (as opposed to other algorithms such as GMM, which assume a Gaussian distribution of the given data).We are given some prior data (also called training data), which classifies coordinates into groups identified by an attribute. As an example, consider the following table of data points containing two features:

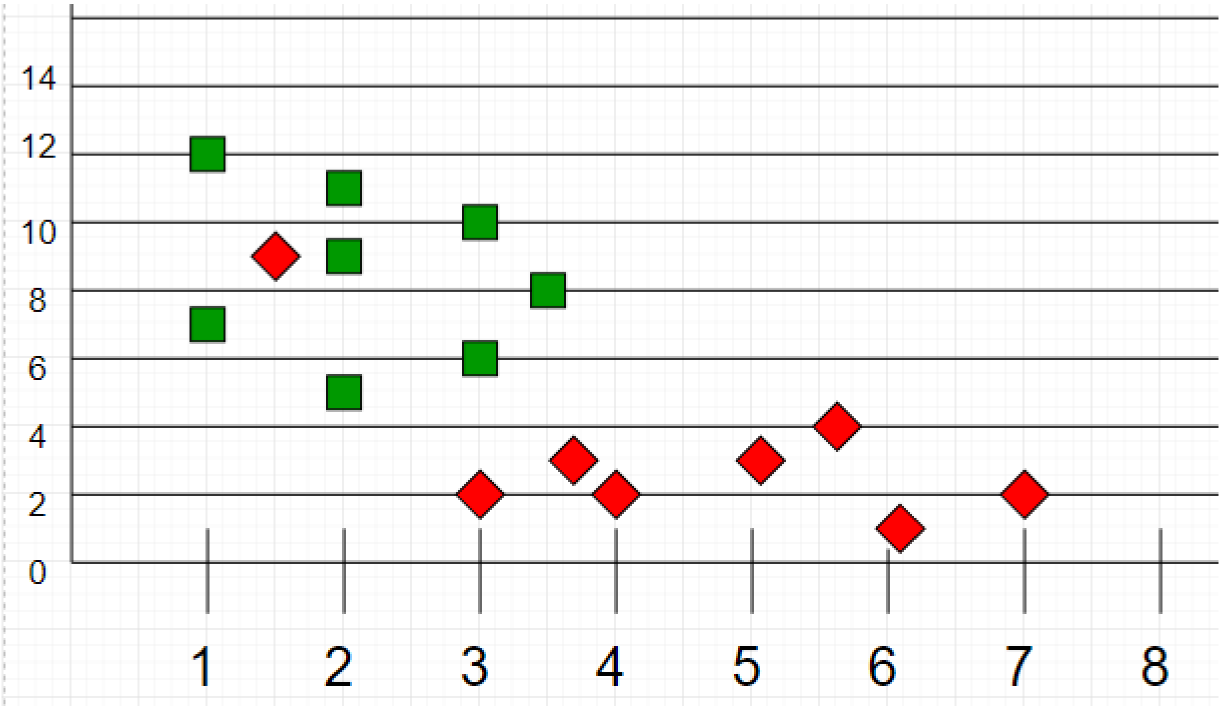


Fig 6.2 Graphical Representation of the trained data points

Now, given another set of data points (also called testing data), allocate these points a group by analyzing the training set. Note that the unclassified points are marked as ‘White’.

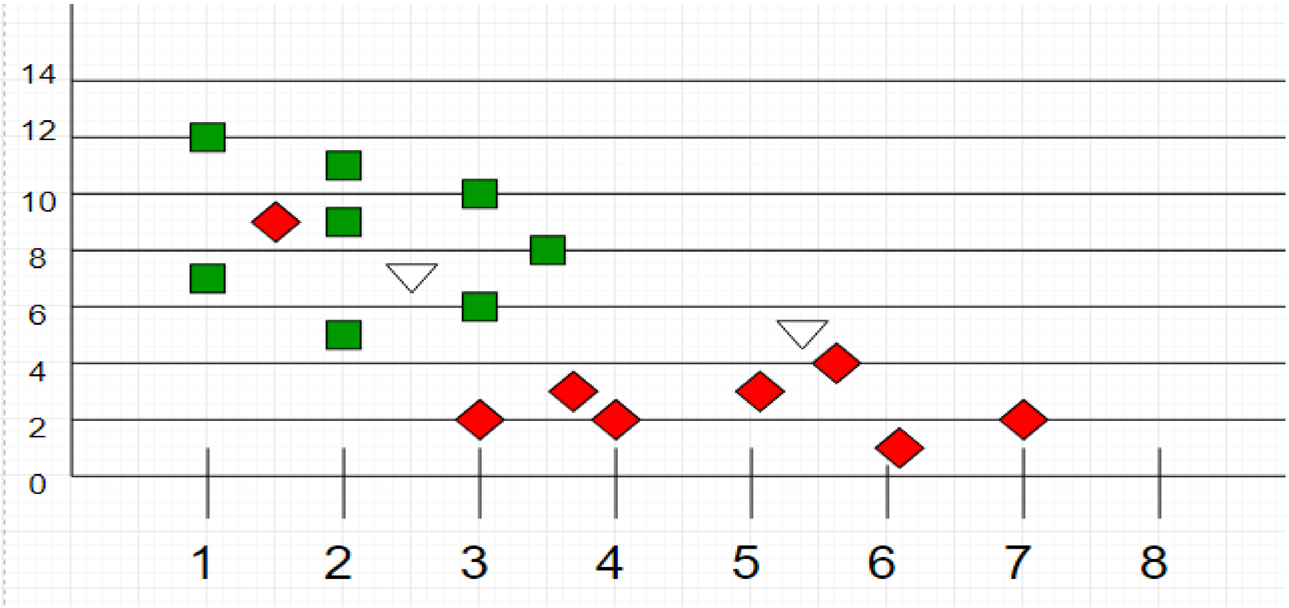


Fig 6.3 Graphical Representation of testing data points

ADABOOST ALGORITHM

AdaBoost also called Adaptive Boosting is a technique in Machine Learning used as an Ensemble Method. The most common algorithm used with AdaBoost is decision trees with one level that means with Decision trees with only 1 split. These trees are also called **Decision Stumps.**

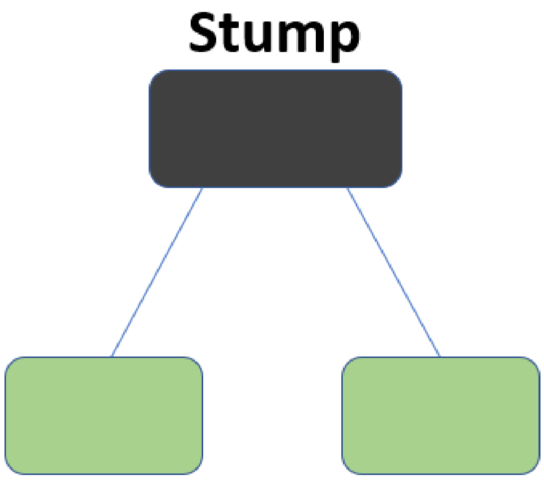


Fig 6.4 Illustration of Decision Stumps

Xgboost Algorithm:

Generally, Xgboost is fast. Really fast when compared to other implementations of gradient boosting.

Szilard Pafka performed some objective benchmarks comparing the performance of Xgboost to other implementations of gradient boosting and bagged decision trees. He wrote up his results in May 2015 in the blog post titled “Implementations”. He also provides a more extensive report of results with hard numbers. His results showed that Xgboost was almost always faster than the other benchmarked implementations from R, Python Spark and H2O.XGBoost dominates structured or tabular datasets on classification and regression predictive modeling problems.

The evidence is that it is the go-to algorithm for competition winners on the Kaggle competitive data science platform. The Xgboost library implements the gradient boosting decision tree algorithm. This algorithm goes by lots of different names such as gradient boosting, multiple additive regression trees, stochastic gradient boosting or gradient boosting machines.

Boosting is an ensemble technique where new models are added to correct the errors made by existing models. Models are added sequentially until no further improvements can be made. A popular example is the AdaBoost algorithm that weights data points that are hard to predict. Gradient boosting is an approach where new models are created that predict the residuals or errors of prior models and then added together to make the final prediction. It is called gradient boosting because it uses a gradient descent algorithm to minimize the loss when adding new models. This approach supports both regression and classification predictive modeling problems.

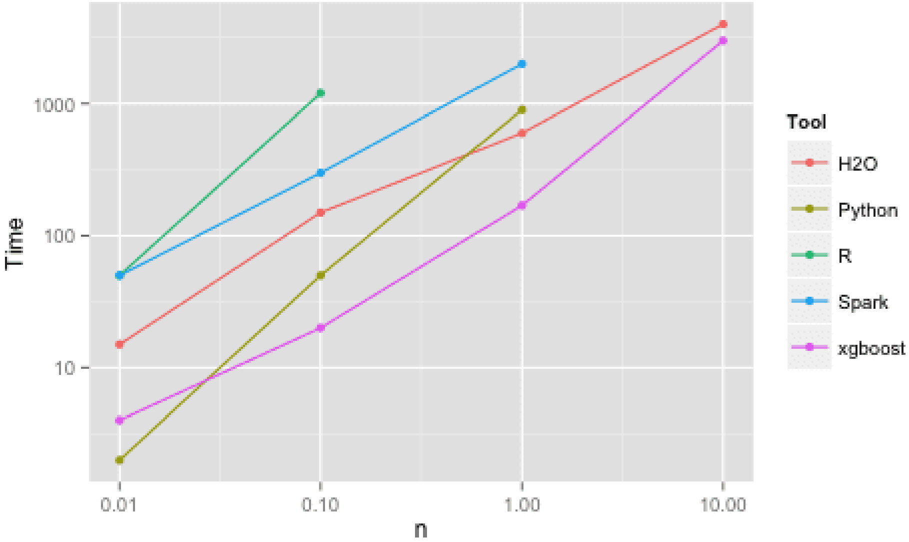


Fig 6.5 Gantt Chart for Xgboost Algorithm Performance

SUPPORT VECTOR MACHINE(SVM)

“Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiate the two classes very well (look at the below snapshot). The SVM algorithm is implemented in practice using a kernel. The learning of the hyperplane in linear SVM is done by transforming the problem using some linear algebra, which is out of the scope of this introduction to SVM. A powerful insight is that the linear SVM can be rephrased using the inner product of any two given observations, rather than the observations themselves. The inner product between two vectors is the sum of the multiplication of each pair of input values. For example, the inner product of the vectors [2, 3] and [5, 6] is 2*5 + 3*6 or 28. The equation for making a prediction for a new input using the dot product between the input (x) and each support vector (xi) is calculated as follows:

$$f(x) = B0 + \text{sum}\ (ai * (x, xi))$$

This is an equation that involves calculating the inner products of a new input vector (x) with all support vectors in training data. The coefficients B0 and ai (for each input) must be estimated from the training data by the learning algorithm.

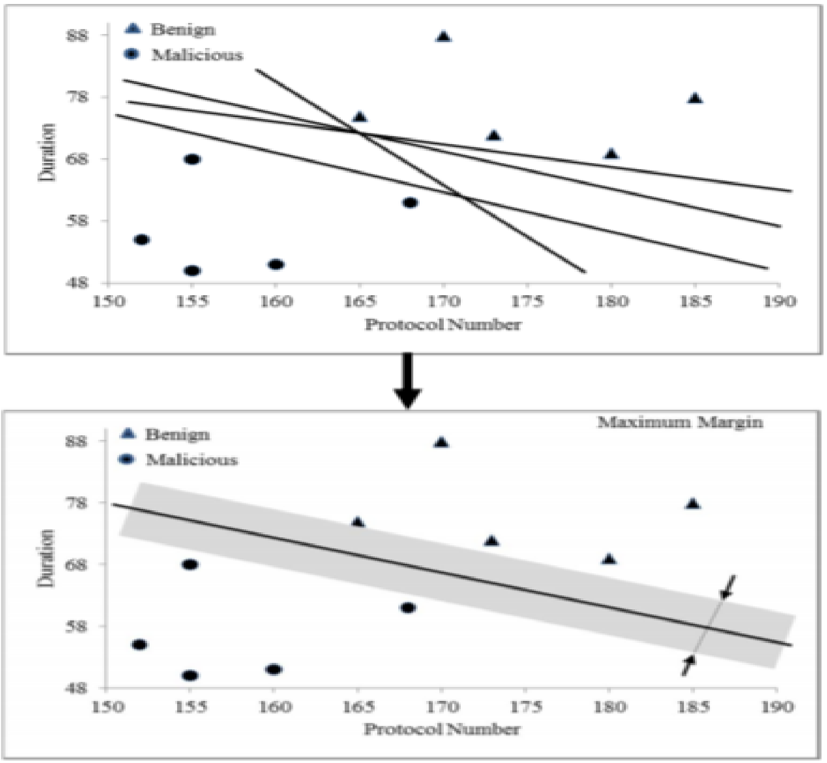


Fig 6.6 Representation of SVM using Duration vs Protocol Graph

6.2 SELECTED SOFTWARE

Python

Python is currently the most widely used multi-purpose, high-level programming language. Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java. Programmers must type relatively less and indentation requirement of the language, makes them readable all the time.

Advantages of Python

Let us see how Python dominates over other languages.

1. Extensive Libraries

Python downloads with an extensive library and it *contain code for various purposes like regular expressions, documentation-generation, unit-testing, web browsers, threading, databases, CGI, email, image manipulation, and more.* So, we don't have to write the complete code for that manually.

2. Extensible

As we have seen earlier, Python can be **extended to other languages**. You can write some of your code in languages like C++ or C. This comes in handy, especially in projects.

3. Embeddable

Complimentary to extensibility, Python is embeddable as well. You can put your Python code in your source code of a different language, like C++. This lets us add **scripting capabilities** to our code in the other language.

4. Improved Productivity

The language's simplicity and extensive libraries render programmers **more productive** than languages like Java and C++ do. Also, the fact that you need to write less and get more things done.

5. IOT Opportunities

Since Python forms the basis of new platforms like Raspberry Pi, it finds the future bright for the Internet Of Things. This is a way to connect the language with the real world.

6. Simple and Easy

When working with Java, you may have to create a class to print **'Hello World'**. But in Python, just a print statement will do. It is also quite **easy to learn, understand,** and **code.**

This is why when people pick up Python, they have a hard time adjusting to other more verbose languages like Java.

7. Readable

Because it is not such a verbose language, reading Python is much like reading English. This is the reason why it is so easy to learn, understand, and code. It also does not need curly braces to define blocks, and **indentation is mandatory.** This further aids the readability of the code.

8. Object-Oriented

This language supports both the **procedural and object-oriented** programming paradigms. While functions help us with code reusability, classes and objects let us model the real world. A class allows the **encapsulation of data** and functions into one.

9. Free and Open-Source

Like we said earlier, Python is **freely available.** But not only can you **download Python** for free, but you can also download its source code, make changes to it, and even distribute it. It downloads with an extensive collection of libraries to help you with your tasks.

10. Portable

When you code your project in a language like C++, you may need to make some changes to it if you want to run it on another platform. But it isn’t the same with Python. Here, you need to **code only once**, and you can run it anywhere. This is called **Write Once Run Anywhere (WORA)**. However, you need to be careful enough not to include any system-dependent features.

11. Interpreted

Lastly, we will say that it is an interpreted language. Since statements are executed one by one, **debugging is easier** than in compiled languages.

Advantages of Python Over Other Languages

1. Less Coding

Almost all the tasks done in Python requires less coding when the same task is done in other languages. Python also has an awesome standard library support, so you don't have to search for any third-party libraries to get your job done. This is the reason that many people suggest learning Python to beginners.

2. Affordable

Python is free therefore individuals, small companies or big organizations can leverage the free available resources to build applications. Python is popular and widely used so it gives you better community support.The 2019 GitHub annual survey showed us that Python has overtaken Java in the most popular programming language category.

Disadvantages of Python

So far, we have seen why Python is a great choice for your project. But if you choose it, you should be aware of its consequences as well. Let us now see the downsides of choosing Python over another language.

1. Speed Limitations

We have seen that Python code is executed line by line. But since Python is interpreted, it often results in **slow execution**. This, however, isn't a problem unless speed is a focal point for the project. In other words, unless high speed is a requirement, the benefits offered by Python are enough to distract us from its speed limitations.

2. Weak in Mobile Computing and Browsers

While it serves as an excellent server-side language, Python is much rarely seen on the **client-side**. Besides that, it is rarely ever used to implement smartphone-based applications. One such application is called **Carbonnelle.** The reason it is not so famous despite the existence of Brython is that it is not that secure.

3. Design Restrictions

As you know, Python is **dynamically-typed**. This means that you don't need to declare the type of variable while writing the code. It uses **duck-typing**. But wait, what is that? Well, it just means that if it looks like a duck, it must be a duck. While this is easy on the programmers during coding, it can **raise run-time errors**.

4. Underdeveloped Database Access Layers

Compared to more widely used technologies like **JDBC (Java Database Connectivity)** and **ODBC (Open Database Connectivity)**, Python's database access layers are a bit underdeveloped. Consequently, it is less often applied in huge enterprises.

5. Simple

No, we are not kidding. Python's simplicity can indeed be a problem. Take my example. I do not do Java, I'm more of a Python person. To me, its syntax is so simple that the verbosity of Java code seems unnecessary.

What is Machine Learning

The study of machine learning certainly arose from research in this context, but in the data science application of machine learning methods, it's more helpful to think of machine learning as a means of *building models of data*.

Fundamentally, machine learning involves building mathematical models to help understand data. "Learning" enters the fray when we give these models *tunable parameters* that can be adapted to observed data; in this way the program can be "learning" from the data. Once these models have been fit to previously seen data, they can be used to predict and understand aspects of newly observed data. I will leave to the reader the more philosophical digression regarding the extent to which this type of mathematical, model-based "learning" is like the "learning" exhibited by the human brain. Understanding the problem setting in machine learning is essential to using these tools effectively, and so we will start with some broad categorizations of the types of approaches we will discuss here.

Challenges in Machines Learning

While Machine Learning is rapidly evolving, making significant strides with cybersecurity and autonomous cars, this segment of AI as whole still has a long way to go. The reason behind is that ML has not been able to overcome number of challenges. The challenges that ML is facing currently are −

Quality of data − Having good-quality data for ML algorithms is one of the biggest challenges. Use of low-quality data leads to the problems related to data preprocessing and feature extraction.

Time-Consuming task − Another challenge faced by ML models is the consumption of time especially for data acquisition, feature extraction and retrieval.

Lack of specialist persons − As ML technology is still in its infancy stage, availability of expert resources is a tough job.

No clear objective for formulating business problems − Having no clear objective and well-defined goal for business problems is another key challenge for ML because this technology is not that mature yet.

Issue of overfitting & underfitting − If the model is overfitting or underfitting, it cannot be represented well for the problem.

Curse of dimensionality − Another challenge ML model faces is too many features of data points. This can be a real hindrance.

Difficulty in deployment − Complexity of the ML model makes it quite difficult to be deployed in real life.

Applications of Machines Learning

Machine Learning is the most rapidly growing technology and according to researchers we are in the golden year of AI and ML. It is used to solve many real-world complex problems which cannot be solved with traditional approach. Following are some real-world applications of ML

- Emotion analysis
- Sentiment analysis

- Error detection and prevention
- Weather forecasting and prediction
- Stock market analysis and forecasting
- Speech synthesis
- Speech recognition
- Customer segmentation
- Object recognition
- Fraud detection
- Fraud prevention
- Recommendation of products to customer in online shopping

(a) Learn Linear Algebra and Multivariate Calculus

Both Linear Algebra and Multivariate Calculus are important in Machine Learning. However, the extent to which you need them depends on your role as a data scientist. If you are more focused on application heavy machine learning, then you will not be that heavily focused on math's as there are many common libraries available. But if you want to focus on R&D in Machine Learning, then mastery of Linear Algebra and Multivariate Calculus is very important as you will have to implement many ML algorithms from scratch.

(b) Learn Statistics

Data plays a huge role in Machine Learning. In fact, around 80% of your time as an ML expert will be spent collecting and cleaning data. And statistics is a field that handles the collection, analysis, and presentation of data. So it is no surprise that you need to learn it!!!Some of the key concepts in statistics that are important are Statistical Significance, Probability Distributions, Hypothesis Testing, Regression, etc. Also, Bayesian Thinking is also a very important part of ML which deals with various concepts like Conditional Probability, Priors, and Posteriors, Maximum Likelihood, etc.

(c) Learn Python

Some people prefer to skip Linear Algebra, Multivariate Calculus and Statistics and learn them as they go along with trial and error. But the one thing that you absolutely cannot skip is Python! While there are other languages you can use for Machine Learning like R, Scala, etc. Python is currently the most popular language for ML. In fact, there are many Python libraries that are specifically useful for Artificial Intelligence and Machine Learning such as Kera's, TensorFlow, Scikit-learn, etc. So, if you want to learn ML, it's best if you learn Python! You can do that using various online resources and courses such as **Fork Python** available Free on GeeksforGeeks.

Step 2 – Learn Various ML Concepts

Now that you are done with the prerequisites, you can move on to learning ML (Which is the fun part!!!) It's best to start with the basics and then move on to the more complicated stuff. Some of the basic concepts in ML are:

(a) Terminologies of Machine Learning

- **Model –** A model is a specific representation learned from data by applying some machine learning algorithm. A model is also called a hypothesis.
- **Feature –** A feature is an individual measurable property of the data. A set of numeric features can be conveniently described by a feature vector. Feature vectors are fed as input to the model. For example, in order to predict a fruit, there may be features like color, smell, taste, etc.
- **Target (Label)** – A target variable or label is the value to be predicted by our model. For the fruit example discussed in the feature section, the label with each set of input would be the name of the fruit like apple, orange, banana, etc.
- **Training –** The idea is to give a set of inputs(features) and it's expected outputs(labels), so after training, we will have a model (hypothesis) that will then map new data to one of the categories trained on.
- **Prediction –** Once our model is ready, it can be fed a set of inputs to which it will provide a predicted output(label).

(b) Types of Machine Learning

- **Supervised Learning –** This involves learning from a training dataset with labeled data using classification and regression models. This learning process continues until the required level of performance is achieved.
- **Unsupervised Learning –** This involves using unlabeled data and then finding the underlying structure in the data in order to learn more and more about the data itself using factor and cluster analysis models.
- **Semi-supervised Learning –** This involves using unlabeled data like Unsupervised Learning with a small amount of labeled data. Using labeled data vastly increases the learning accuracy and is also more cost-effective than Supervised Learning.
- **Reinforcement Learning –** This involves learning optimal actions through trial and error. So, the next action is decided by learning behaviors that are based on the current state and that will maximize the reward in the future.

Advantages of Machine learning

1. Easily identifies trends and patterns -

Machine Learning can review large volumes of data and discover specific trends and patterns that would not be apparent to humans. For instance, for an e-commerce website like Amazon, it serves to understand the browsing behaviors and purchase histories of its users to help cater to the right products, deals, and reminders relevant to them. It uses the results to reveal relevant advertisements to them.

2. No human intervention needed (automation)

With ML, you don’t need to babysit your project every step of the way. Since it means giving machines the ability to learn, it lets them make predictions and also improve the algorithms on their own. A common example of this is anti-virus software’s; they learn to filter new threats as they are recognized. ML is also good at recognizing spam.

3. Continuous Improvement

As ML algorithm**s** gain experience, they keep improving in accuracy and efficiency. This lets them make better decisions. Say you need to make a weather forecast model. As the amount of data, you have keeps growing, your algorithms learn to make more accurate predictions faster.

4. Handling multi-dimensional and multi-variety data

Machine Learning algorithms are good at handling data that are multi-dimensional and multi-variety, and they can do this in dynamic or uncertain environments.

5. Wide Applications

You could be an e-tailer or a healthcare provider and make ML work for you. Where it does apply, it holds the capability to help deliver a much more personal experience to customers while also targeting the right customers.

Disadvantages of Machine Learning

1. Data Acquisition

Machine Learning requires massive data sets to train on, and these should be inclusive/unbiased, and of good quality. There can also be times where they must wait for new data to be generated.

2. Time and Resources

ML needs enough time to let the algorithms learn and develop enough to fulfill their purpose with a considerable amount of accuracy and relevancy. It also needs massive resources to function. This can mean additional requirements of computer power for you.

3. Interpretation of Results

Another major challenge is the ability to accurately interpret results generated by the algorithms. You must also carefully choose the algorithms for your purpose.

4. High error-susceptibility

Machine Learning is autonomous but highly susceptible to errors. Suppose you train an algorithm with data sets small enough to not be inclusive. You end up with biased predictions coming from a biased training set. This leads to irrelevant advertisements being displayed to customers. In the case of ML, such blunders can set off a chain of errors that can go undetected for long periods of time. And when they do get noticed, it takes quite some time to recognize the source of the issue, and even longer to correct it.

Python Development Steps

Purpose

We demonstrated that our approach enables successful segmentation of intra-retinal layers—even with low-quality images containing speckle noise, low contrast, and different intensity ranges throughout—with the assistance of the ANIS feature.

Python

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library. 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 also acknowledges that speed of development is important. Readable and terse code is part of this, and so is access to powerful constructs that avoid tedious repetition of code. Maintainability also ties into this may be an all but useless metric, but it does say something about how much code you have to scan, read and/or understand to troubleshoot problems or tweak behaviors. This speed of development, the ease with which a programmer of other languages can pick up basic Python skills and the huge standard library is key to another area where Python excels. All its tools have been quick to implement, saved a lot of time, and several of them have later been patched and updated by people with no Python background - without breaking.

Modules Used in Project

TensorFlow

TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library, and is also used for machine learning applications such as neural_networks. It is used for both research and production at Google. TensorFlow was developed by the Google_Brain team for internal Google use. It was released under the Apache 2.0 open-source license on November 9, 2015.

NumPy

NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.It is the fundamental package for scientific computing with Python. It contains various features including these important ones:

- A powerful N-dimensional array object
- Sophisticated (broadcasting) functions
- Tools for integrating C/C++ and Fortran code
- Useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined using NumPy which allows NumPy to integrate with a wide variety of databases seamlessly and speedily.

Pandas

Pandas is an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data structures. Python was majorly used for data munging and preparation. It had very little contribution towards data analysis. Pandas solved this problem. Using Pandas, we can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data load, prepare, manipulate, model, and analyze. Python with Pandas is used in a wide range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc.

Matplotlib

Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and I Python shells, the Jupyter Notebook, web application servers, and four graphical user interface toolkits. Matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, error charts, scatter plots, etc., with just a few lines of code. For examples, see the sample plots and thumbnail gallery.

For simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with I Python. For the power user, you have full control of line styles, font properties, axes properties, etc., via an object-oriented interface or via a set of functions familiar to MATLAB users.

Scikit – learn

Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. It is licensed under a permissive simplified BSD license and is distributed under many Linux distributions, encouraging academic and commercial use. **Python**

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library. 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 also acknowledges that speed of development is important. Readable and terse code is part of this, and so is access to powerful constructs that avoid tedious repetition of code. Maintainability also ties into this may be an all but useless metric, but it does say something about how much code you have to scan, read and/or understand to troubleshoot problems or tweak behaviors. This speed of development, the ease with which a programmer of other languages can pick up basic Python skills and the huge standard library is key to another area where Python excels. All its tools have been quick to implement, saved a lot of time, and several of them have later been patched and updated by people with no Python background - without breaking.

Install Python Step-by-Step in Windows and Mac

Python a versatile programming language does not come pre-installed on your computer devices. Python was first released in the year 1991 and until today it is a very popular high-level programming language. Its style philosophy emphasizes code readability with its notable use of great whitespace.The object-oriented approach and language construct provided by Python enables programmers to write both clear and logical code for projects. This software does not come pre-packaged with Window

How to Install Python on Windows and Mac

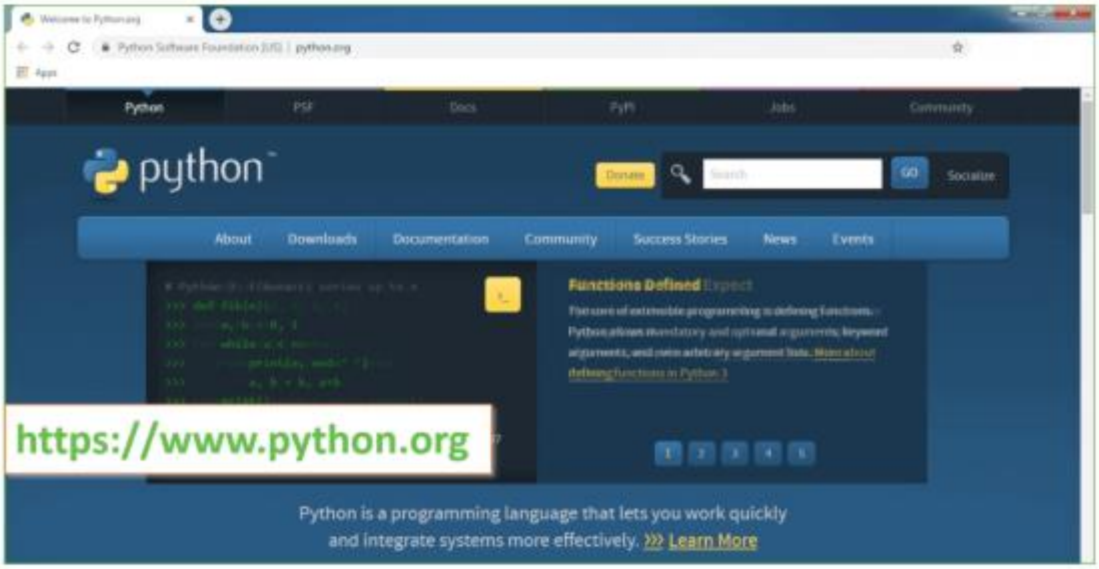
There have been several updates in the Python version over the years. The question is how to install Python? It might be confusing for the beginner who is willing to start learning Python but this tutorial will solve your query. The latest or the newest version of Python is version 3.7.4 or in other words, it is Python 3.

Note: The python version 3.7.4 cannot be used on Windows XP or earlier devices.

Before you start with the installation process of Python. First, you need to know about your **System Requirements**. Based on your system type i.e. operating system and based processor, you must download the python version. My system type is a **Windows 64-bit operating system**. So the steps below are to install python version 3.7.4 on Windows 7 device or to install Python 3. Download the Python Cheatsheet here.The steps on how to install Python on Windows 10, 8 and 7 are **divided into 4 parts** to help understand better.

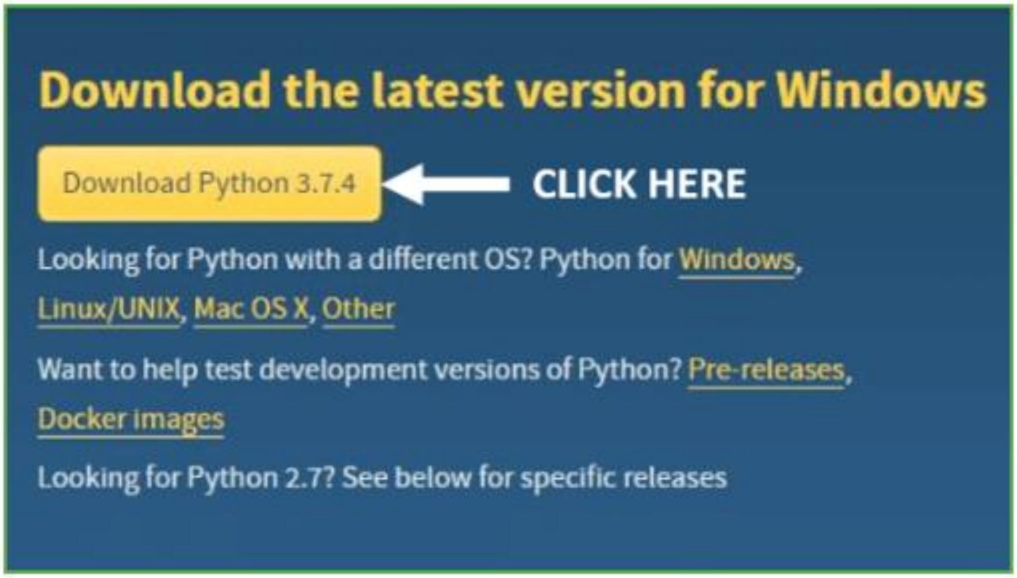
Download the Correct version into the system

Step 1: Go to the official site to download and install python using Google Chrome or any other web browser. OR Click on the following link: **https://www.python.org**



Now, check for the latest and the correct version for your operating system.

Step 2: Click on the Download Tab.



Download the latest version for Windows

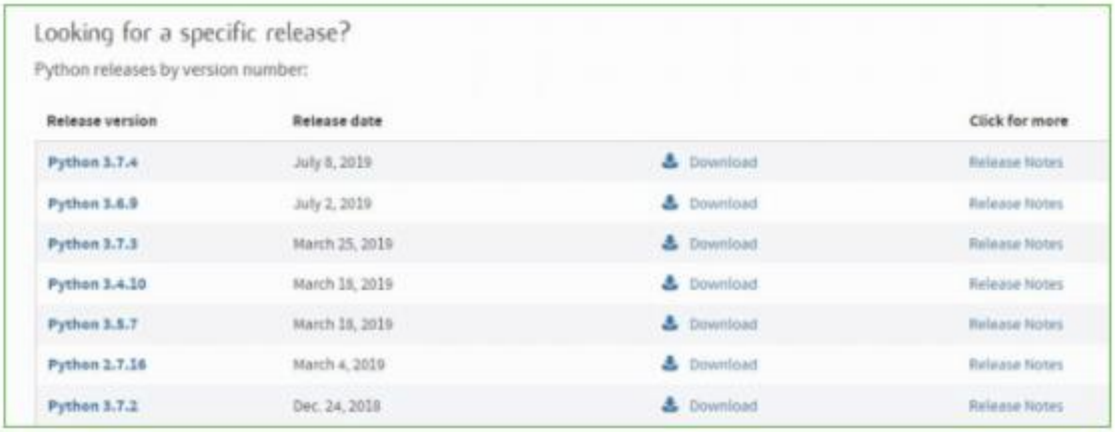
Download Python 3.7.4 CLICK HERE

Looking for Python with a different OS? Python for Windows, Linux/UNIX, Mac OS X, Other

Want to help test development versions of Python? Pre-releases, Docker images

Looking for Python 2.7? See below for specific releases

Step 3: You can either select the Download Python for windows 3.7.4 button in Yellow Color or you can scroll further down and click on download with respective to their version. Here, we are downloading the most recent python version for windows 3.7.4



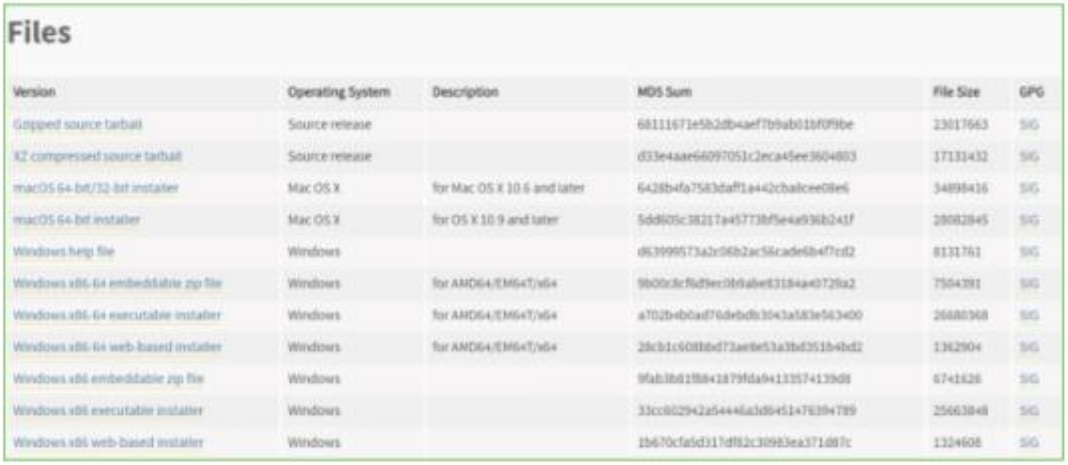
Looking for a specific release?

Python releases by version number:

Release version	Release date		Click for more
Python 3.7.4	July 8, 2019	Download	Release Notes
Python 3.6.9	July 2, 2019	Download	Release Notes
Python 3.7.3	March 25, 2019	Download	Release Notes
Python 3.4.10	March 18, 2019	Download	Release Notes
Python 3.5.7	March 18, 2019	Download	Release Notes
Python 2.7.16	March 4, 2019	Download	Release Notes
Python 3.7.2	Dec. 24, 2018	Download	Release Notes

Step 4: Scroll down the page until you find the Files option.

Step 5: Here you see a different version of python along with the operating system.



Files

Version	Operating System	Description	MD5 Sum	File Size	GPG
Gzipped source tarball	Source release		68111671e5b2db4aef7b9ab01bf0f9be	23017663	SIG
XZ compressed source tarball	Source release		d33e4aae66097051c2eca45ee3604803	17131432	SIG
macOS 64-bit/32-bit installer	Mac OS X	for Mac OS X 10.6 and later	6428b4fa7583daff1a442cba8cee08e6	34898416	SIG
macOS 64-bit installer	Mac OS X	for OS X 10.9 and later	5dd605c38217a45773bf5e4a936b241f	28082845	SIG
Windows help file	Windows		d63999573a2c56b2ac56cade6b4f7cd2	8131761	SIG
Windows x86-64 embeddable zip file	Windows	for AMD64/EM64T/x64	9b00c8cf6d9ec0b9abe83184a40729a2	7504391	SIG
Windows x86-64 executable installer	Windows	for AMD64/EM64T/x64	a702b4b0ad76debdb3043a583e563400	26680368	SIG
Windows x86-64 web-based installer	Windows	for AMD64/EM64T/x64	28cb1c608bbd73ae8e53a3bd351b4bd2	1362904	SIG
Windows x86 embeddable zip file	Windows		9fab3b81f8841879fda94133574139d8	6741626	SIG
Windows x86 executable installer	Windows		33cc602942a54446a3d6451476394789	25663848	SIG
Windows x86 web-based installer	Windows		1b670cfa5d317df82c30983ea371d87c	1324608	SIG

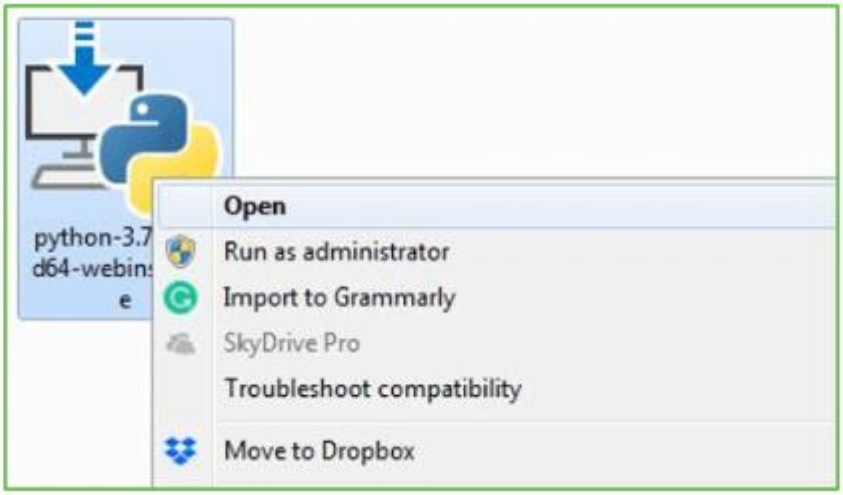
- To download Windows 32-bit python, you can select any one from the three options: Windows x86 embeddable zip file, Windows x86 executable installer or Windows x86 web-based installer.
- To download Windows 64-bit python, you can select any one from the three options: Windows x86-64 embeddable zip file, Windows x86-64 executable installer or Windows x86-64 web-based installer.

Here we will install Windows x86-64 web-based installer. Here your first part regarding which version of python is to be downloaded is completed. Now we move ahead with the second part in installing python i.e. Installation

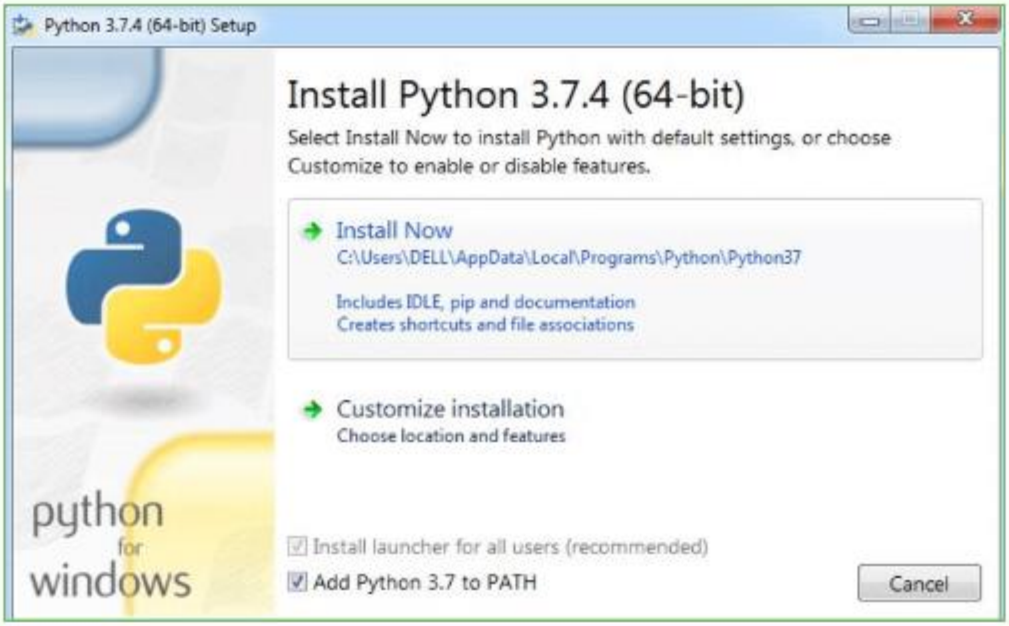
Note: To know the changes or updates that are made in the version you can click on the Release Note Option.

Installation of Python

Step 1: Go to Download and Open the downloaded python version to carry out the installation process.



Step 2: Before you click on Install Now, Make sure to put a tick on Add Python 3.7 to PATH.



Python 3.7.4 (64-bit) Setup

Install Python 3.7.4 (64-bit)

Select Install Now to install Python with default settings, or choose Customize to enable or disable features.

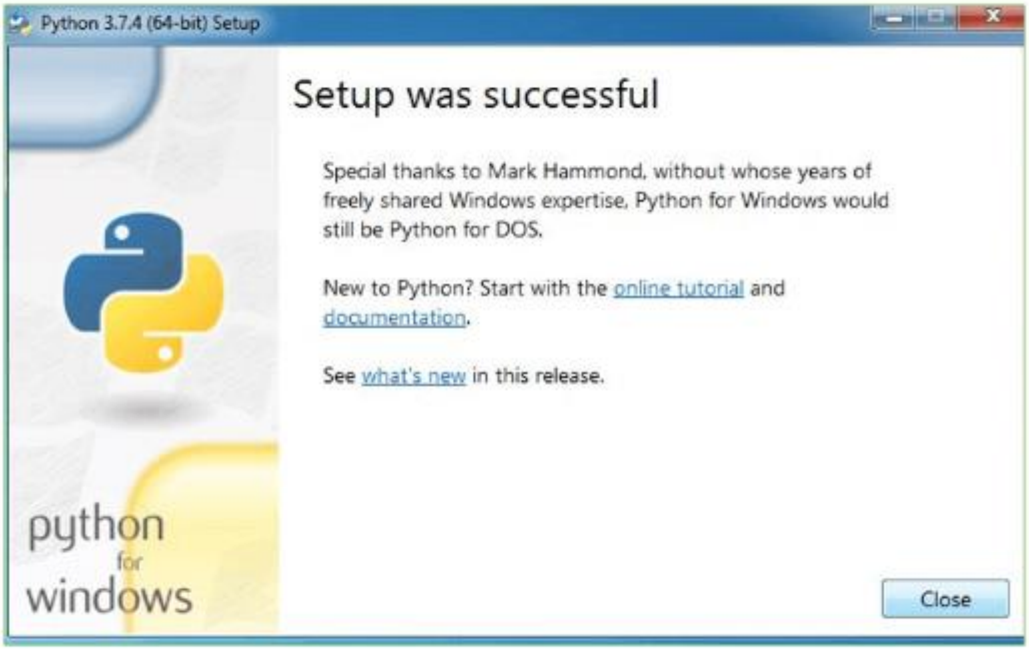
Install Now
C:\Users\DELL\AppData\Local\Programs\Python\Python37

Includes IDLE, pip and documentation
Creates shortcuts and file associations

Customize installation
Choose location and features

☑ Install launcher for all users (recommended)
☑ Add Python 3.7 to PATH

Step 3: Click on Install NOW After the installation is successful. Click on Close.



Python 3.7.4 (64-bit) Setup

Setup was successful

Special thanks to Mark Hammond, without whose years of freely shared Windows expertise, Python for Windows would still be Python for DOS.

New to Python? Start with the online tutorial and documentation.

See what's new in this release.

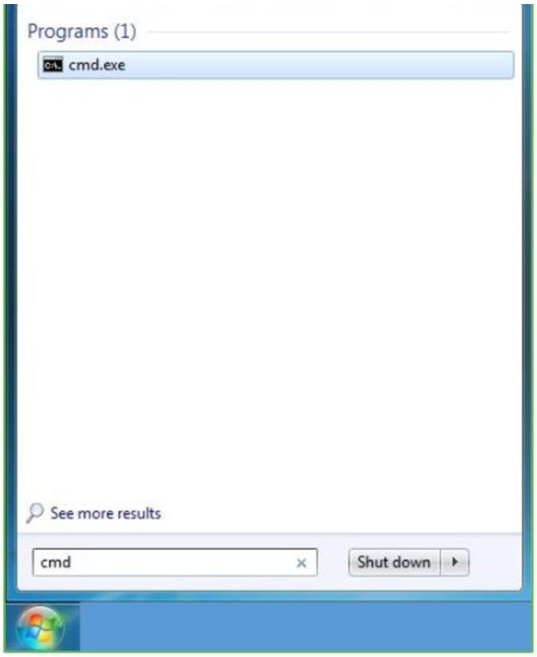
With these above three steps on python installation, you have successfully and correctly installed Python. Now is the time to verify the installation.

Note: The installation process might take a couple of minutes.

Verify the Python Installation

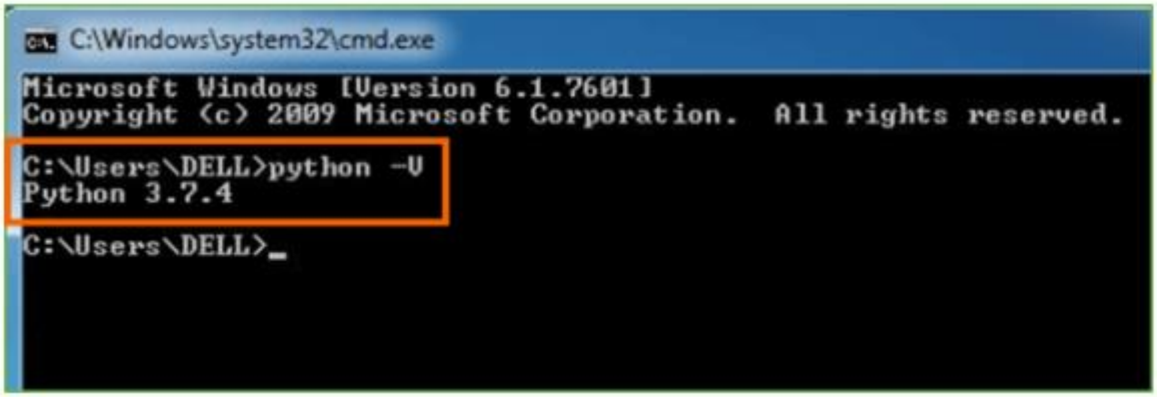
Step 1: Click on Start

Step 2: In the Windows Run Command, type “cmd”.



Step 3: Open the Command prompt option.

Step 4: Let us test whether the python is correctly installed. Type **python –V** and press Enter.



```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\DELL>python -V
Python 3.7.4

C:\Users\DELL>_
```

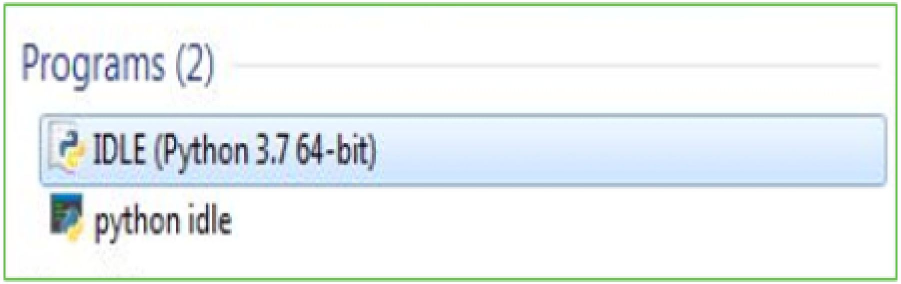
Step 5: You will get the answer as 3.7.4

Note: If you have any of the earlier versions of Python already installed. You must first uninstall the earlier version and then install the new one.

Check how the Python IDLE works

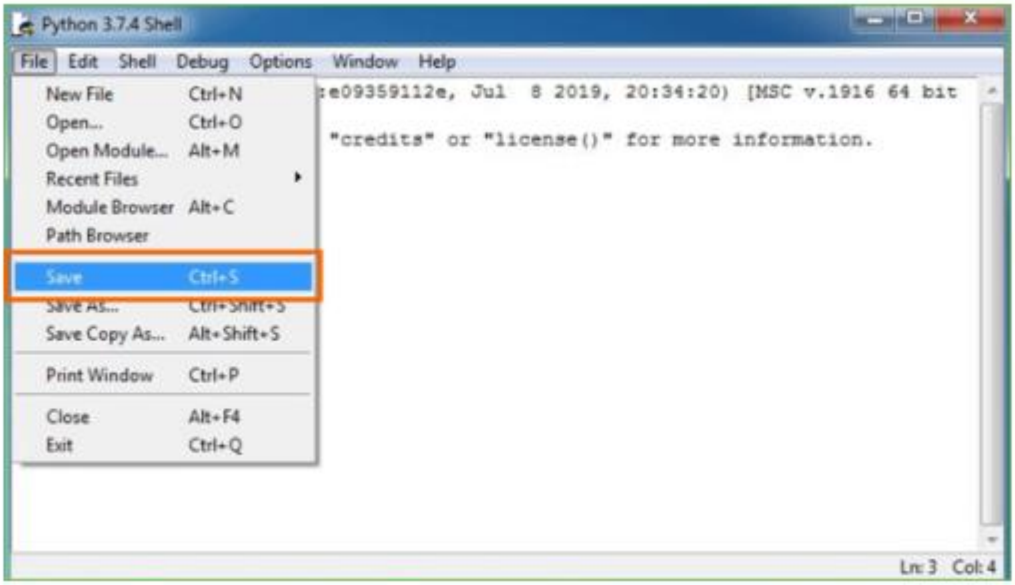
Step 1: Click on Start

Step 2: In the Windows Run command, type "python idle".



Step 3: Click on IDLE (Python 3.7 64-bit) and launch the program

Step 4: To go ahead with working in IDLE you must first save the file. **Click on File > Click on Save**



```
:e09359112e, Jul  8 2019, 20:34:20) [MSC v.1916 64 bit
"credits" or "license()" for more information.
```

Step 5: Name the file and save as type should be Python files. Click on SAVE. Here I have named the files as Hey World.

Step 6: Now for e.g. **enter print**

6.3 SAMPLE CODE:

Code for Web Application using Python language:

```
from flask import Flask,request, url_for, redirect, render_template
import pandas as pd
import numpy as np
import pickle
import sqlite3
from keras.models import load_model
from sklearn.preprocessing import normalize
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
app = Flask(__name__)
model_path2 = 'model.h5' # load .h5 Model
model = load_model(model_path2)
@app.route('/')
def hello_world():
    return render_template("home.html")
@app.route('/login')
def login():
                return render_template('signin.html')
@app.route("/signin")
def signin():
    mail1 = request.args.get('user',")
    password1 = request.args.get('password',")
    con = sqlite3.connect('signup.db')
    cur = con.cursor()
    cur.execute("select `user`, `password` from info where `user` = ? AND `password` = ?",(mail1,password1,))
    data = cur.fetchone()
    if data == None:
        return render_template("signin.html")

    elif mail1 == 'admin' and password1 == 'admin':
        return render_template("index.html")

    elif mail1 == str(data[0]) and password1 == str(data[1]):
        return render_template("index.html")
    else:
        return render_template("signup.html")
@app.route('/predict',methods=['POST','GET'])
def predict():
    text[1 to 24] = request.form['1' to ‘24’]
    row_df                                                                                    =
pd.DataFrame([text1,text2,text3,text4,text5,text6,text7,text8,text9,text10,text11,text12,text13,text14,text15,text16,text17,text18
,text19,text20,text21,text22,text23,text24])
    row_df = sc.fit_transform(row_df)
    row_df  = row_df.reshape(1,-1)
    print(row_df.shape)
    prediction = model.predict(row_df)
    print(prediction)
    prediction = (prediction > 0.5)
    if prediction:
        return render_template('result.html',pred=f'You have chance of having Kidney Disease')
    else:
        return render_template('result.html',pred=f'You are safe! You do not have Disease')
@app.route('/index')
def index():
                return render_template('index.html')
if __name__ == '__main__':
    app.run(debug=True)
```

Code for Implementing Machine Learning Techniques to Predict CKD:

Importing Libraries:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
from sklearn.metrics import confusion_matrix, accuracy_score, classification_report
from sklearn import metrics
from sklearn.model_selection import train_test_split
import pickle
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import cross_val_score
```

Importing Datasets:

```
dataset = pd.read_csv("Dataset/kidney_disease.csv")
```

Dataset Handling:

```
dataset.head()
dataset.shape
dataset.info()
dataset.isnull().sum()
dataset.describe().T
dataset.drop('id', axis = 1, inplace = True)
```

Data Visualization:

```
sns.countplot(x = 'class',data = dataset)
--------------------------
plt.figure(figsize = (20, 15))
plotnumber = 1
for column in num_cols:
    if plotnumber <= 14:
        ax = plt.subplot(3, 5, plotnumber)
        sns.distplot(dataset[column],color='black',)
        plt.xlabel(column)
    plotnumber += 1
--------------------------
plt.tight_layout()
plt.show()
plt.figure(figsize = (20, 15))
plotnumber = 1
for column in cat_cols:
    if plotnumber <= 11:
        ax = plt.subplot(3, 4, plotnumber)
        sns.countplot(dataset[column], palette = 'rocket',color='black')
        plt.xlabel(column)
    plotnumber += 1
plt.tight_layout()
plt.show()
```

Implementing Machine Learning Techniques using scikit:

```
from sklearn.linear_model import LogisticRegression
LR = LogisticRegression(random_state=0)
LR.fit(X_train, y_train)
```

7. TESTING

7.1 INTRODUCTION

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

TYPES OF TESTS

Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

7.2 TEST CASES

PATIENT DETAILS

Table 7.1 Representation of Test Cases using Patient Details

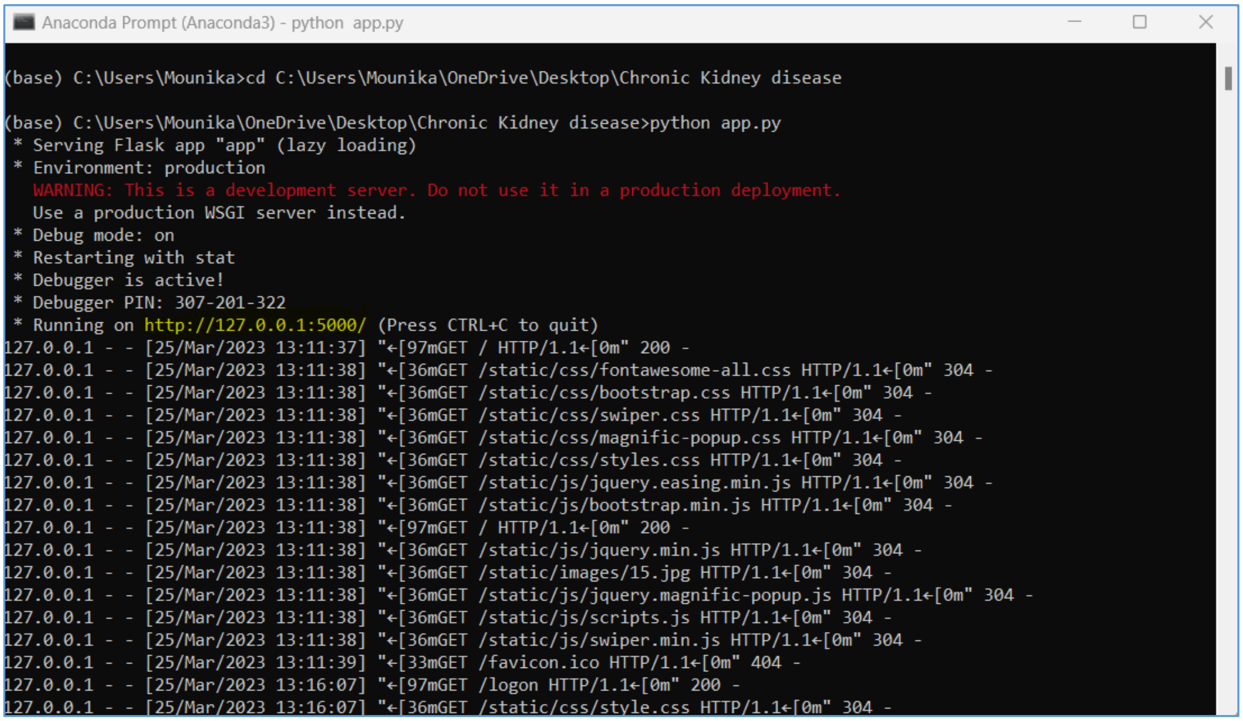
Test case ID	Name of the Test Case	Usability	Test Case	Output
1	Predict Diseases	To detect the diseases with accuracy.	The user gives the input in the form of upload patient details.	An output predicts the patient has no kidney disease. So, the patient is not safe
2	Predict Diseases	To detect the diseases with accuracy.	The user gives the input in the form of upload patient details.	An output predicts the patient has kidney disease. So, the patient is safe.

PREDICTION DISEASES

Table 7.2 Representation of Test Cases to Predict Disease

Test case ID	Test case name	Purpose	Test Case	Output
1	Clinical Dataset of Patients	To detect the diseases with accuracy.	The user gives the input in the form of upload patient details.	An output predicts diseases with patient details are done.
2	Predict Diseases	To detect the diseases with accuracy.	The user gives the input in the form of upload patient details.	An output predicts diseases with patient details are done.

8.SCREENS & REPORTS



```
(base) C:\Users\Mounika>cd C:\Users\Mounika\OneDrive\Desktop\Chronic Kidney disease

(base) C:\Users\Mounika\OneDrive\Desktop\Chronic Kidney disease>python app.py
 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 307-201-322
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [25/Mar/2023 13:11:37] "←[97mGET / HTTP/1.1←[0m" 200 -
127.0.0.1 - - [25/Mar/2023 13:11:38] "←[36mGET /static/css/fontawesome-all.css HTTP/1.1←[0m" 304 -
127.0.0.1 - - [25/Mar/2023 13:11:38] "←[36mGET /static/css/bootstrap.css HTTP/1.1←[0m" 304 -
127.0.0.1 - - [25/Mar/2023 13:11:38] "←[36mGET /static/css/swiper.css HTTP/1.1←[0m" 304 -
127.0.0.1 - - [25/Mar/2023 13:11:38] "←[36mGET /static/css/magnific-popup.css HTTP/1.1←[0m" 304 -
127.0.0.1 - - [25/Mar/2023 13:11:38] "←[36mGET /static/css/styles.css HTTP/1.1←[0m" 304 -
127.0.0.1 - - [25/Mar/2023 13:11:38] "←[36mGET /static/js/jquery.easing.min.js HTTP/1.1←[0m" 304 -
127.0.0.1 - - [25/Mar/2023 13:11:38] "←[36mGET /static/js/bootstrap.min.js HTTP/1.1←[0m" 304 -
127.0.0.1 - - [25/Mar/2023 13:11:38] "←[97mGET / HTTP/1.1←[0m" 200 -
127.0.0.1 - - [25/Mar/2023 13:11:38] "←[36mGET /static/js/jquery.min.js HTTP/1.1←[0m" 304 -
127.0.0.1 - - [25/Mar/2023 13:11:38] "←[36mGET /static/images/15.jpg HTTP/1.1←[0m" 304 -
127.0.0.1 - - [25/Mar/2023 13:11:38] "←[36mGET /static/js/jquery.magnific-popup.js HTTP/1.1←[0m" 304 -
127.0.0.1 - - [25/Mar/2023 13:11:38] "←[36mGET /static/js/scripts.js HTTP/1.1←[0m" 304 -
127.0.0.1 - - [25/Mar/2023 13:11:38] "←[36mGET /static/js/swiper.min.js HTTP/1.1←[0m" 304 -
127.0.0.1 - - [25/Mar/2023 13:11:39] "←[33mGET /favicon.ico HTTP/1.1←[0m" 404 -
127.0.0.1 - - [25/Mar/2023 13:16:07] "←[97mGET /logon HTTP/1.1←[0m" 200 -
127.0.0.1 - - [25/Mar/2023 13:16:07] "←[36mGET /static/css/style.css HTTP/1.1←[0m" 304 -
```

Fig 8.1 Running of Python File in Anaconda Prompt

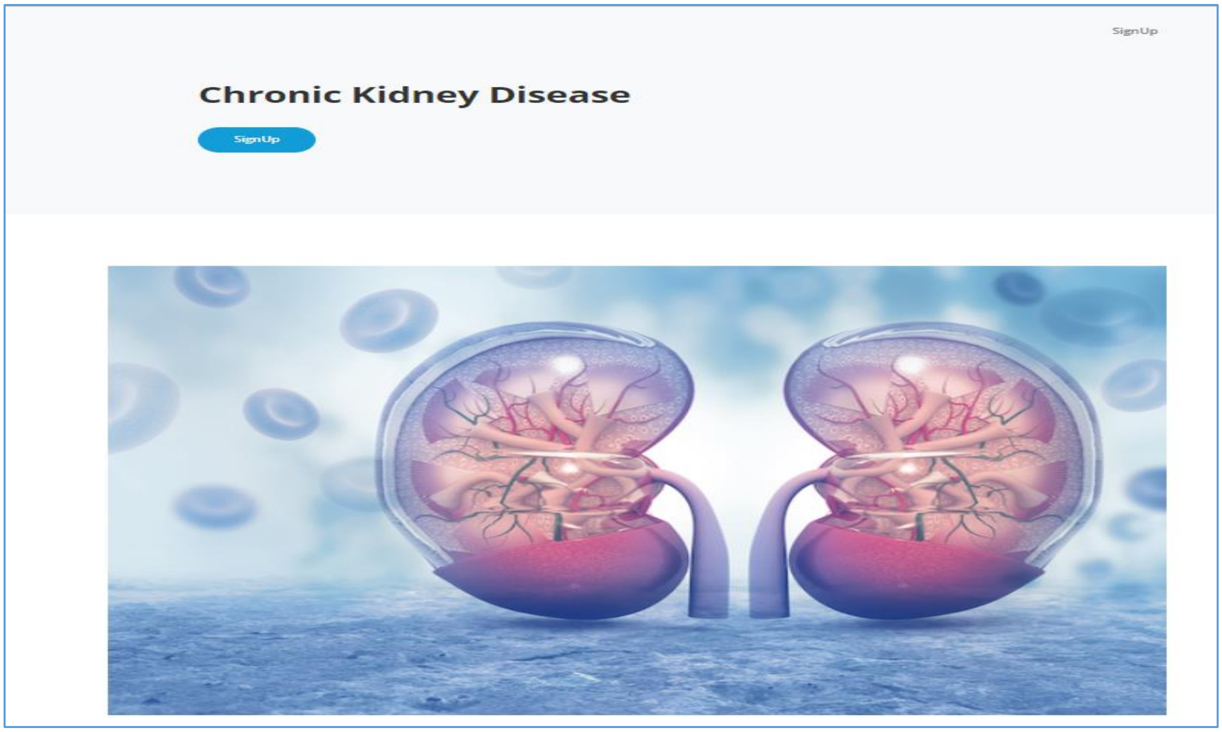


Fig 8.2 Initial Interface of the Web Page

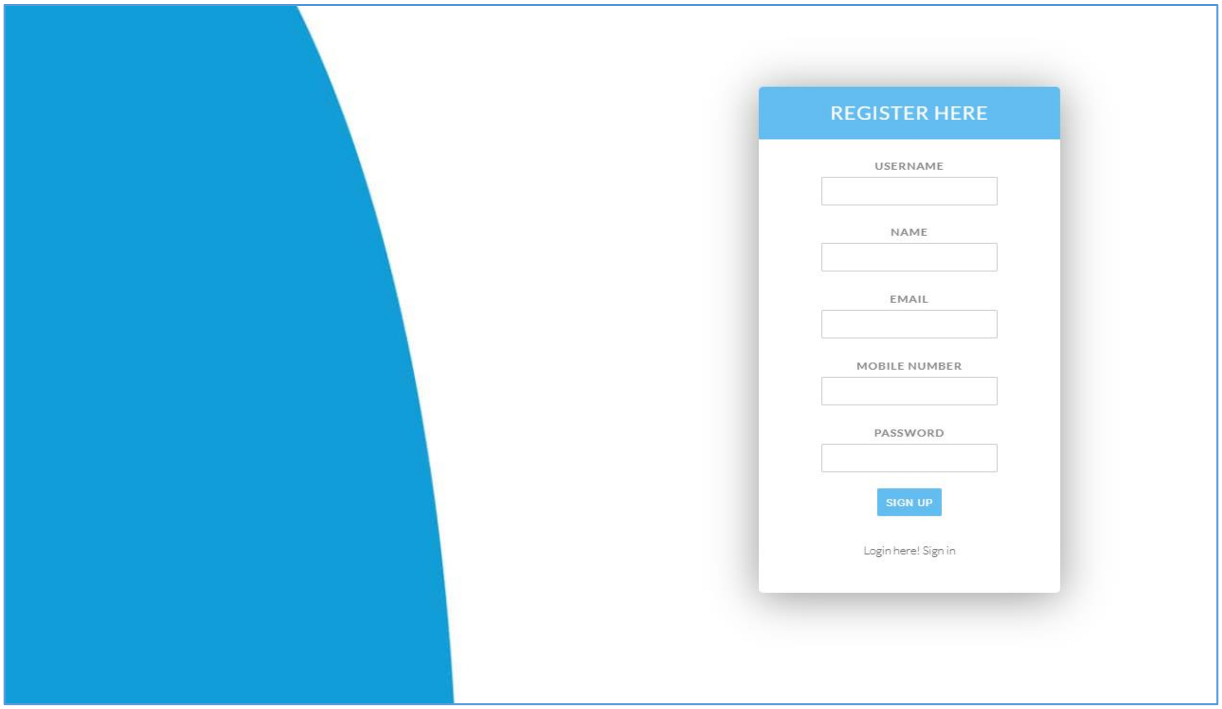


Fig 8.3 Registration Interface of the User

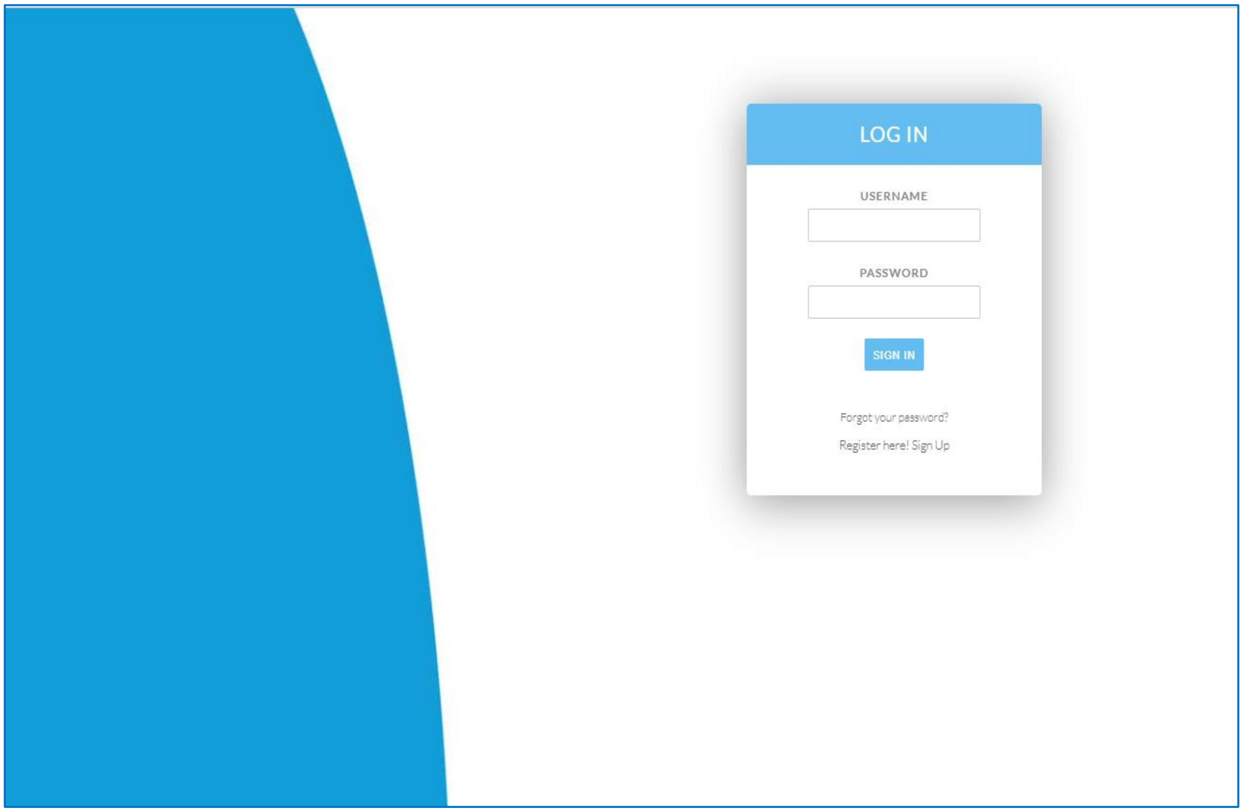
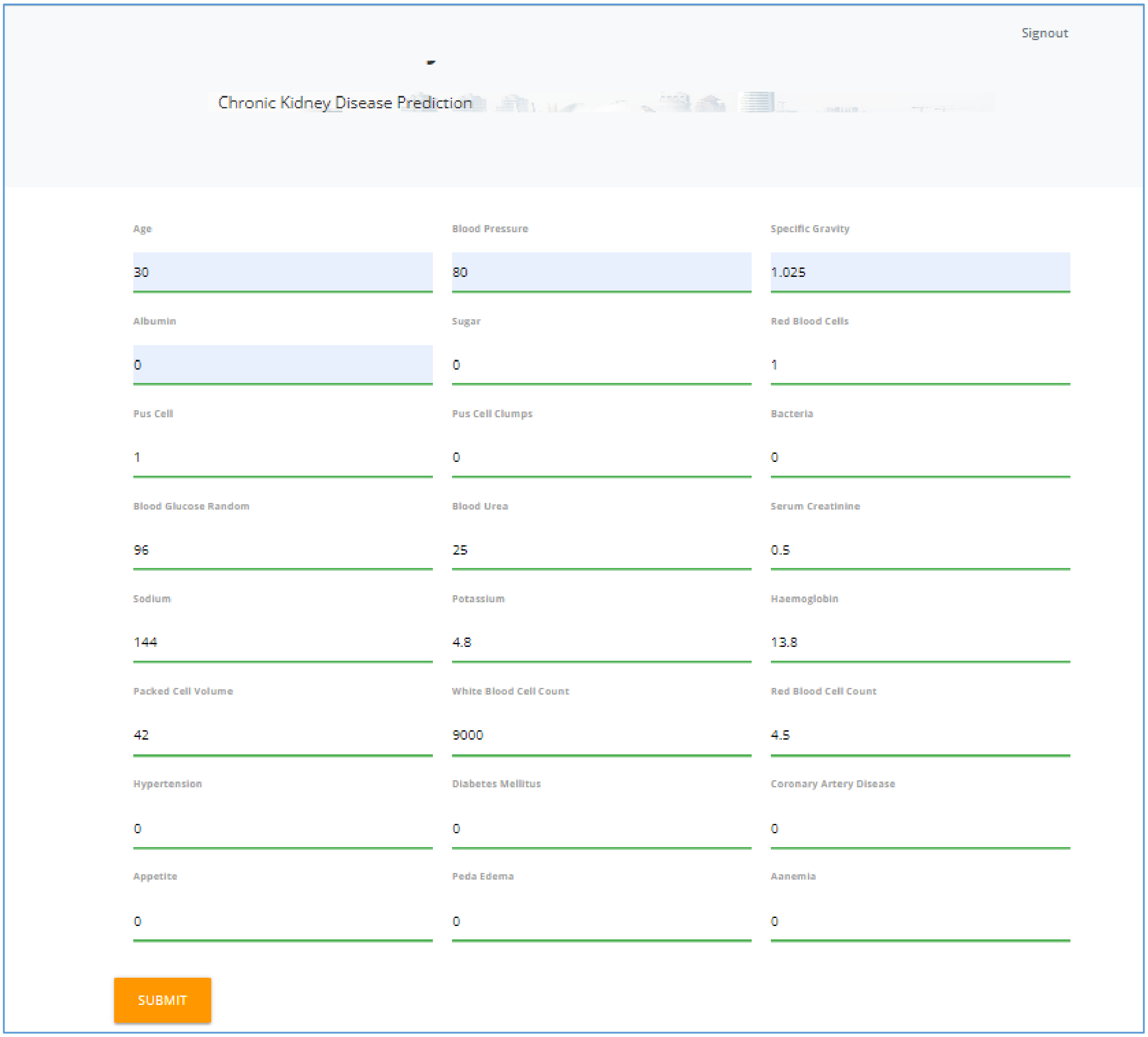


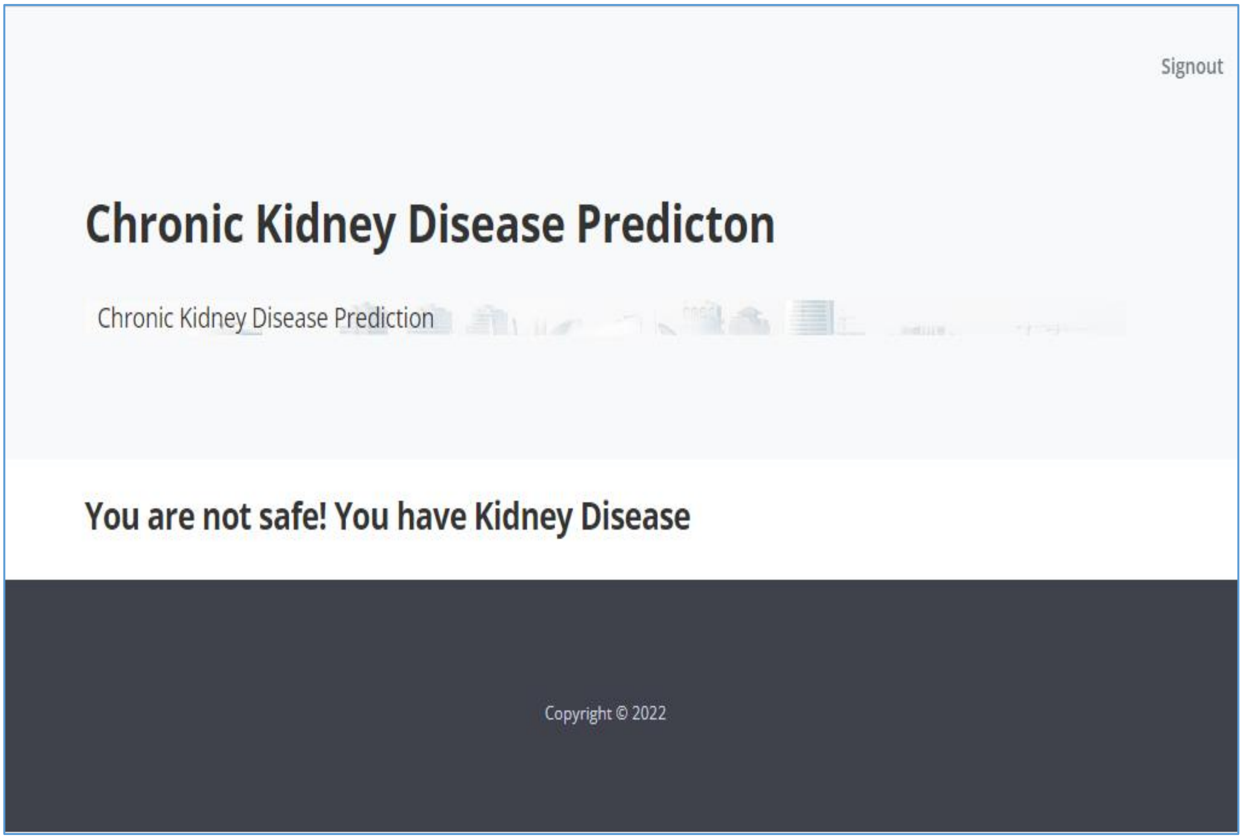
Fig 8.4 Log In interface in the Web Page



Chronic Kidney Disease Prediction

Age	Blood Pressure	Specific Gravity
30	80	1.025
Albumin	Sugar	Red Blood Cells
0	0	1
Pus Cell	Pus Cell Clumps	Bacteria
1	0	0
Blood Glucose Random	Blood Urea	Serum Creatinine
96	25	0.5
Sodium	Potassium	Haemoglobin
144	4.8	13.8
Packed Cell Volume	White Blood Cell Count	Red Blood Cell Count
42	9000	4.5
Hypertension	Diabetes Mellitus	Coronary Artery Disease
0	0	0
Appetite	Peda Edema	Aanemia
0	0	0

Fig 8.5 Acquiring the Patient Details from the User



Signout

Chronic Kidney Disease Predicton

Chronic Kidney Disease Prediction

You are not safe! You have Kidney Disease

Copyright © 2022

Fig 8.6 Result Page of Chronic Kidney Disease Prediction

9. CONCLUSION and FUTURE SCOPE

9.1 Conclusion:

Early and error-free detection of CKD can be helpful in averting further deterioration of patient's health. These chronic diseases are prognosticated using various types of data mining classification approaches and machine learning (ML) algorithms. This Prediction is performed using Logistic Regression ,Random Forest, SVM, Xgboost ,Adaboost . The data used is collected from the UCI Repository with 400 data sets with 21 attributes. This data has been fed into Classification algorithms. The experimental results show that DT, RF, Gradient Boosting hands out an accuracy of 98.75%, 98.75% and 97.50% respectively. The Xgboost and Adaboost classifier gives out a maximum accuracy of 100%.

9.2 Future Scope:

This Project is implemented using the web-based simulating environment created using Python and with the usage of Machine Learning Techniques such as Random Forest, KNN, Support Vector Machine, Logistic Regression which is combined with Perceptron, to obtain better accuracy in the Diagnostics of Chronic Kidney Disease.

10. BIBLIOGRAPHY

10.1 WEB LINKS:

[1]https://www.researchgate.net/publication/304342832_Clinical_risk_assessment_of_patients_with_chronic_kidney_disease_by_using_clinical_data_and_multivariate_models

[2]https://www.researchgate.net/publication/299499558_Prediction_and_detection_models_for_acute_kidney_injury_in_hospitalized_older_adults

[3]https://www.researchgate.net/publication/314985466_Diagnosis_of_Chronic_Kidney_Disease_by_Using_Random_Forest

[4]https://www.researchgate.net/publication/297891670_Diagnosis_of_patients_with_chronic_kidney_disease_by_using_two_fuzzy_classifiers

[5] https://youtube.com/watch?v=Ou-7G9VQugg&feature=shares

[6] https://youtube.com/watch?v=WkFtIqWmX9o&feature=shares

10.2 REFERENCES:

1. Z. Chen, Z. Zhang, R. Zhu, Y. Xiang, and P. B. Harrington, ``Diagnosis of patients with chronic kidney disease by using two fuzzy classifiers," Chemometrics Intell. Lab. Syst., vol. 153, pp. 140145, Apr. 2016.

2. A. Subasi, E. Alickovic, and J. Kevric, ``Diagnosis of chronic kidney disease by using random forest," in Proc. Int. Conf. Med. Biol. Eng.,Mar. 2017, pp. 589594.

3. L. Zhang, ``Prevalence of chronic kidney disease in China: A crosssectionalsurvey," Lancet, vol. 379, pp. 815822, Mar. 2012.

4. A. Singh, G. Nadkarni, O. Gottesman, S. B. Ellis, E. P. Bottinger, andJ. V. Guttag, ``Incorporating temporal EHR data in predictive models for risk stratication of renal function deterioration," J. Biomed. Informat.,vol. 53, pp. 220228, Feb. 2015.

5. A. M. Cueto-Manzano, L. Cortés-Sanabria, H.w R. Martínez-Ramírez,E. Rojas-Campos, B. Gómez-Navarro, and M. Castillero-Manzano,``Prevalence of chronic kidney disease in an adult population," Arch. Med.Res., vol. 45, no. 6, pp. 507513, Aug. 2014.

6 H. Polat, H. D. Mehr, and A. Cetin, ``Diagnosis of chronic kidney disease based on support vector machine by feature selection methods," J. Med.Syst., vol. 41, no. 4, p. 55, Apr. 2017.

7. C. Barbieri, F. Mari, A. Stopper, E. Gatti, P. Escandell-Montero,J. M. Martínez-Martínez, and J. D. Martín-Guerrero, ``A new machine learning approach for predicting the response to anemia treatment in a large cohort of end stage renal disease patients undergoing dialysis," Comput.Biol. Med., vol. 61, pp. 5661, Jun. 2015.

8. V. Papademetriou, E. S. Nylen, M. Doumas, J. Probsteld, J. F. Mann,R. E. Gilbert, and H. C. Gerstein, ``Chronic kidney disease, basal insulin glargine, and health outcomes in people with dysglycemia: The ORIGIN Study," Amer. J. Med., vol. 130, no. 12, pp. 1465.e271465.e39, Dec. 2017.

9. N. R. Hill, ``Global prevalence of chronic kidney disease A systematic review and meta-analysis," PLoS ONE, vol. 11, no. 7, Jul. 2016,Art. no. e0158765.

10. M. M. Hossain, R. K. Detwiler, E. H. Chang, M. C. Caughey, M.W. Fisher,T. C. Nichols, E. P. Merricks, R. A. Raymer, M. Whitford, D. A. Bellinger,L. E. Wimsey, and C. M. Gallippi, ``Mechanical anisotropy assessment in kidney cortex using ARFI peak displacement: Preclinical validation and pilot in vivo clinical results in kidney allografts," IEEE Trans. Ultrason.,Ferroelectr., Freq. Control, vol. 66, no. 3, pp. 551562, Mar. 2019.

11. M. Alloghani, D. Al-Jumeily, T. Baker, A. Hussain, J. Mustana, andA. J. Aljaaf, ``Applications of machine learning techniques for software engineering learning and early prediction of students' performance," in Proc. Int. Conf. Soft Comput. Data Sci., Dec. 2018, pp. 246258.

12. D. Gupta, S. Khare, and A. Aggarwal, ``A method to predict diagnostic codes for chronic diseases using machine learning techniques," in Proc.Int. Conf. Comput., Commun. Autom.(ICCCA), Apr. 2016, pp. 281287.

13. L. Du, C. Xia, Z. Deng, G. Lu, S. Xia, and J. Ma, ``A machine learning based approach to identify protected health information in Chinese clinical text," Int. J. Med. Informat., vol. 116, pp. 2432, Aug. 2018.

14. R. Abbas, A. J. Hussain, D. Al-Jumeily, T. Baker, and A. Khattak, ``Classification of foetal distress and hypoxia using machine learning approaches,"in Proc. Int. Conf. Intell.Comput., Jul. 2018, pp. 767776.

15. M. Mahyoub, M. Randles, T. Baker, and P. Yang, ``Comparison analysis of machine learning algorithms to rank alzheimer's disease risk factors by importance," in Proc. 11th Int. Conf. Develop. eSyst. Eng. (DeSE),Sep. 2018, pp. 111.

ADITYA ENGINEERING COLLEGE (A)
Aditya Nagar, ADB Road, Surampalem
Department of Computer Science and Engineering

Academic Year: 2022-23

Project Title: **A Novel Methodology for Diagnosing Chronic Kidney Disease Using Machine Learning**

Type of Project: **Application Oriented/Design Oriented/Societal based/ Research Oriented/Industry Oriented**

Project Guide: Dr. Anand Kumar Kinjarapu

Project Team:

K. Mounika	(20A95A0516)
V. Rama Seshu	(19A91A05J5)
Ch. Lakshmi Durga	(20A95A0515)

ABSTRACT

Early diagnosis and characterization are the important components in determining the treatment of chronic kidney disease (CKD). CKD is an ailment which tends to damage the kidney and affect their effective functioning of excreting waste and balancing body fluids. Some of the complications included are hypertension, anemia (low blood count), mineral bone disorder, poor nutritional health, acid base abnormalities, and neurological complications. Early and error-free detection of CKD can be helpful in averting further deterioration of patient's health. These chronic diseases are prognosticated using various types of data mining classification approaches and machine learning (ML) algorithms. This Prediction is performed using Logistic Regression, KNN, Random Forest, SVM, Xgboost, Adaboost. The data used is collected from the UCI Repository with 400 data sets with 21 attributes. This data has been fed into Classification algorithms. The experimental results show that DT, RF, Gradient Boosting hands out an accuracy of 98.75%, 98.75% and 97.50% respectively. The Xgboost and Adaboost classifier gives out a maximum accuracy of 100%.

Chronic kidney disease (CKD) is a global health problem with high morbidity and mortality rate, and it induces other diseases. Since there are no obvious symptoms during the early stages of CKD, patients often fail to notice the disease. Early detection of CKD enables

patients to receive timely treatment to ameliorate the progression of this disease. Machine learning models can effectively aid clinicians achieve this goal due to their fast and accurate recognition performance. In this study, we propose a machine learning methodology for diagnosing CKD. The CKD data set was obtained from the University of California Irvine (UCI) machine learning repository, which has many missing values. KNN imputation was used to fill in the missing values, which selects several complete samples with the most similar measurements to process the missing data for each incomplete sample. Missing values are usually seen in real-life medical situations because patients may miss some measurements for various reasons. After effectively filling out the incomplete data set, six machine learning algorithms (logistic regression, random forest, support vector machine, k-nearest neighbor, Xgboost, Adaboost) were used to establish models.

Signature of the Team members

COURSE OUTCOMES

PROJECT PART 1

Regulation: AR19	L	T	P	C
Course Code:191CS7P04	0	0	4	2

CO1 Identify a real life / engineering problem

CO2 Perform extensive investigation with prior knowledge

CO3 Interpret problem formulation and solution through critical thinking

CO4 Develop the work plan, schedule and estimate the cost

CO5 Identify the resources required to initiate project work

PROJECT PART 2

Regulation: AR19	L	T	P	C
Course Code:191CS8P05	0	0	14	7

CO6 Apply the domain knowledge to arrive at a framework to solve the problem

CO7 Design solution using research-based knowledge and modern tools and interpret the results

CO8 Assess the obtained solution in the context of engineering framework addressing the societal and environmental concerns adhering to professional ethics

CO9 Demonstrate communication skills effectively to work as a team, for guide interaction and presentations.

CO10 Prepare technical documentation/reports with effective written communication skills

CO-PO MAPPING

PROJECT PART-I 191CS7P04

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	1								1		1	2	2
CO2	1	1	1								3	1	1	2
CO3		2				2	3					1	2	2
CO4	1	1									3	1	2	2
CO5	1	1									3	1	2	2

PROJECT PART-II191CS8P05

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO6	1	3	2	2	1				2	2	2	3	2	1
CO7	1	3	2		1	1			2	2	2	2	2	1
CO8	1	3	3	2			1	1	2	2	2	2	2	1
CO9	1	3	2	1					2	2	2	2	2	1
CO10	1	3	2	1	1	1		1	1	2	1	1	2	1

CONCLUSION

Early and error-free detection of CKD can be helpful in averting further deterioration of patient's health. These chronic diseases are prognosticated using various types of data mining classification approaches and machine learning (ML) algorithms. This Prediction is performed using Logistic Regression, Random Forest, SVM, Xgboost, Adaboost. The data used is collected from the UCI Repository with 400 data sets with 21 attributes. This data has been fed into Classification algorithms. The experimental results show that DT, RF, Gradient Boosting hands out an accuracy of 98.75%, 98.75% and 97.50% respectively. The Xgboost and Adaboost classifier gives out a maximum accuracy of 100%.

FUTURE SCOPE

This Project is implemented using the web-based simulating environment created using Python and with the usage of Machine Learning Techniques such as Random Forest, KNN, Support Vector Machine, Logistic Regression which is combined with Perceptron, to obtain better accuracy in the Diagnostics of Chronic Kidney Disease.

ADITYA ENGINEERING COLLEGE (A)

Aditya Nagar, ADB Road, Surampalem
Department of Computer Science and Engineering

Academic Year: 2022-23

Project Title: **A Novel Methodology for Diagnosing Chronic Kidney Disease using Machine Learning.**

Type of Project: **Application Oriented/Design Oriented/Societal based/ Research Oriented/Industry Oriented**

Project Guide: Dr. Anand Kumar Kinjarapu

PROGRAM OUTCOMES			
PO1	Engineering knowledge	PO7	Environment and sustainability
PO2	Problem analysis	PO8	Ethics
PO3	Design/development of solutions	PO9	Individual and team work
PO4	Conduct investigations of complex problems	PO10	Communication
PO5	Modern tool usage	PO11	Project management and finance
PO6	The engineer and society	PO12	Lifelong learning

PROGRAM SPECIFIC OUTCOMES	
PSO1	Develop efficient solutions to the real-world problems using the domains of algorithms, networks, database management and latest programming tools and techniques.
PSO2	Provide the data centric business solutions through emerging areas like IoT, AI ,data analytics and Block chain technologies.

CONCLUSION STATEMENTS

S.No	Description	Attained COs	Attained POs/PSOs
1	Identified various causes and effects of Chronic Kidney Disease through research and study.	CO1,CO2,CO5	PO1,PO2
2	Identified the most affecting cause for this disease.	CO1, CO3	PO2
3	Researched about ways to let the patients and clinicians to predict the chances of getting affected by CKD	CO3CO7	PO2,PO4, PSO1
4	Literatures were reviewed about the Prediction of the Kidney Disease by various authors.	CO2,CO5	PO2,PO4
5	A new methodology for predicting the Kidney Disease presence is proposed.	CO4, CO6,CO3	PO3,PO5 PO8
6	Compared this method with existing methods.	CO2,CO9	PO4,PO6
7	New Algorithms and ML Techniques were added to make a new and better solution for the problem.	CO5,CO8,	PO3, PO8,PO12 PSO1
8	Concluded and analysed the solution and presented it.	CO9,CO4 CO5	PO9,PO10PO11, PSO2

Signature of the Team Members

K. Mounika (20A95A0516)

V. Rama Seshu (19A91A05J5)

Ch. Lakshmi Durga (20A95A0515)

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

A Novel Methodology for Diagnosing Chronic Kidney Disease Using Machine Learning

[1]Koppanathi Mounika, [2]Vinnakota Rama Seshu, [3]Chintalapudi Lakshmi Durga, [4]Dr. Anand Kumar Kinjarapu, [5]Mr. B.R.S.S. Raju

[1,2,3]CSE, B. Tech, Aditya Engineering College, Surampalem
[4]Professor, Aditya Engineering
College, Surampalem [5]Assistant
Professor,Aditya Engineering
College, Surampalem

ABSTRACT:

To characterize the therapy for chronic kidney disease (CKD), early determination and characterisation are fundamental. The kidneys are harmed in CKD, making it harder for them to take out squander and keep a sound liquid equilibrium. The results incorporate hypertension, pallor (low blood count), mineral bone infection, nourishing lacks, corrosive base irregularities, and neurological issues. Patients might profit from an early and blunder free conclusion of CKD to keep away from additional medical conditions. Various information mining arrangement strategies and machine learning (ML) calculations are used to anticipate these chronic diseases. This figure uses Logistic Regression, KNN, Random forest, decision Tree, SVM, Gradient Boosting, Xgboost, Adaboost, and Troupe. The UCI Archive contains 400 informational indexes with 21 properties, which are utilized for this review. Characterization calculations were taken care of this data. The consequences of the investigation demonstrate that Inclination Helping, DT, and RF each have a accuracy of 97.50%. The Xgboost and Adaboost classifiers have a greatest exactness of 100%.

***Keywords** – Gradient Boosting, Xgboost, Adaboost, Ensemble, Logistic Regression, KNN, Random Forest, Decision Tree, SVM, and Random Forest.*

1. INTRODUCTION

In the present society, chronic kidney disease (CKD) is viewed as a huge wellbeing risk. There are treatments for ongoing kidney infection that can slow the progression of the disease, reduce the effects of a lower Glomerular Filtration Rate (GFR) and the risk of cardiovascular disease, and further improve endurance and personal satisfaction. CKD may be achieved by a shortfall of hydration, smoking, an uncalled-for eating routine, a shortfall of rest, and different various issues. This condition impacted 753 million individuals overall in 2016, with 417 million females and 336 million guys impacted. More often than not, the illness is found in its last stages, which can prompt renal disappointment. The ongoing technique for analysis depends on examining pee utilizing serum creatinine levels. For this reason, different clinical systems, like ultrasonography and screening, are used. Tests are finished on individuals who have hypertension, a background marked by cardiovascular illness, sickness before, and family members who have kidney infection. In a first-morning pee test, this technique consolidates assessing GFR from blood creatinine levels and estimating the albumin-to-creatinine ratio (ACR) in the pee.

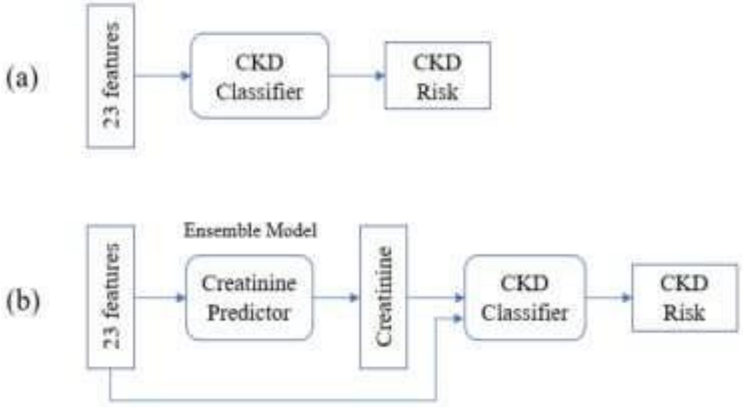


Fig.1: Example figure

The kidneys are two organs with the state of beans and the size of a clench hand [1-3]. Each side of the spine, straightforwardly underneath the rib confine, has one. The kidneys channel somewhere in the range of 120 and 150 quarts of blood each day to create somewhere in the range of 1 and 2 quarts of pee. The essential capability of the kidneys is to create pee, which is utilized to dispose of abundance liquid and waste from the body. The cycles of discharge and re-assimilation that make up pee creation are very unpredictable. This framework is supposed to keep the body's manufactured agreement reliable. The body's salt, potassium, and corrosive levels are constrained by the kidneys, which additionally produce chemicals that influence how different organs work. For example, a chemical created by the kidneys controls calcium digestion, manages circulatory strain, and drives the blend of red platelets. 14% of the total populace is impacted by CKD, a condition described by a dynamic lessening in kidney capability over the long run. Albeit this number may just address 10% of the individuals who expect treatment to make due, roughly 2 million individuals overall required dialysis or a kidney relocate. A larger number of individuals kick the bucket from constant kidney sickness than from bosom or prostate malignant growth [2]. The deliberate or estimated glomerular filtration rate (eGFR), still up in the air by creatinine level [4], orientation, race, and age, is generally answerable for deciding the periods of CKD. There are five stages to kidney capability [5]. The capability is ordinary in stage one and somewhat decreased in stage two, yet most cases happen in stage 3.

2. LITERATURE REVIEW

Diagnosis of patients with chronic kidney disease by using two fuzzy classifiers:

Two in-house fluffy classifiers, fuzzy optimal associative memory (FOAM) and fuzzy rule-building expert system (FuRES), were examined for their suitability for CKD conclusion. The purpose of the examination was to employ a straight classifier known as partial least squares discriminant analysis (PLS-DA). The UCI artificial intelligence Vault provided the CKD data for this study. By adding changing levels of corresponding commotion, composite datasets were made to test the strength of the two fluffy strategies. The reenacted preparing and expectation sets were then participated two by two after 11 degrees of corresponding commotion were acquainted each in turn with each mathematical quality. A lattice of 121 arrangements of recreated information was made subsequently, and grouping rates for these 121 pairings were looked at. Second, the normal forecast paces of FuRES and Froth with 200 bootstrap Latin parcels were 98.1 0.5 percent and 97.2 1.2 percent, separately, on mimicked datasets with 11 degrees of irregular clamor appropriated to each mathematical property. The PLS-DA might furnish 94.3 0.8% with a similar assessment. Intersecting datasets contained the first and changed datasets were likewise used for the assessment of FuRES, Froth, and PLS-DA grouping models. FuRES and Foam ordinary assumption rates from 200 bootstrapped assessments were 99.2 0.3% and 99.0 0.3%, independently. The exactness of PLS-DA is 95.9 0.6 percent lower. According to the disclosures, FuRES and Foam are both effective at recognizing CKD patients, with FuRES being more generous than Foam. These two feathery classifiers can be used to examine different patients despite CKD patients. They are amazing tools for analysis.

Diagnosis of chronic kidney disease by using random forest:

The worldwide general medical condition known as chronic kidney disease (CKD) influences around 10% of the total populace. Notwithstanding, there is inadequate substantial data with respect to a calculated and mechanized

CKD finding. This study researches how CKD can be recognized utilizing machine learning (ML) methods. The successful application of ML calculations, which have served as the primary impetus for the identification of anomalies in various physiological data, is being realized by projects of varying order. A genuine informational collection from the UCI AI Vault is utilized to experimentally test an assortment of ML classifiers in this review, and our discoveries are contrasted with those in the current writing. Numerically and subjectively, we see that as the random forest (RF) classifier performs almost ideally with regards to recognizing CKD cases. Thusly, we demonstrate the way that RF can likewise be utilized to analyze comparable illnesses and that ML calculations assume a critical part in the determination of CKD with satisfactory vigor.

Prevalence of chronic kidney disease in China: A cross sectional survey:

Foundation: Renal illness is common in agricultural countries. In any case, there has never been a cross-country study of chronic kidney disease that includes both albuminuria and estimated glomerular filtration rate (eGFR) in a wealthy agricultural nation like China. We expected to sort out how typical continuous renal disorder is in China through this survey. Strategies: From an extensively delegate test, we did a cross-sectional outline of Chinese individuals. Continuous renal disease was assessed using albuminuria or an eGFR of less than 60 mL/min per 1•73 m(2). Blood and pee tests were gathered, members had their pulse checked, and they finished a poll about their way of life and clinical history. The glomerular filtration not entirely set in stone by estimating the degrees of serum creatinine. To decide albuminuria, the degrees of creatinine and egg whites in the pee were estimated. The unrefined and changed predominance marks of kidney harm were determined, and calculated relapse was utilized to research factors related with the event of chronic kidney disease.

Incorporating temporal EHR data in predictive models for risk stratification of renal function deterioration:

In electronic health records (EHRs), prescient models worked with transient information can possibly fundamentally work on the administration of constant sicknesses. In any case, these information present various specialized difficulties, like conflicting information assortment and changing lengths of available patient history. Three unmistakable ways to deal with utilizing AI to produce expectation models from a patient's fleeting EHR information are the focal point of this review. The most common approach combines the positive aspects of the patient's clinical history's indicators. The worldly elements of the information are utilized in the other two methodologies. One-of-a-kind is the way that the two fleeting methods model global information and handle missing information. We developed and evaluated models for anticipating loss of estimated glomerular filtration rate (eGFR), the most widely recognized estimation of kidney capability, making use of data from Mount Sinai Clinical Center's electronic health records (EHR). According to our disclosures, coordinating common information into a patient's clinical history could assist with predicting a lessening in renal capacity. In addition, they emphasize the significance of applying this information. Because our findings demonstrate that the overall value of various indicators fluctuates over time, perform multiple tasks learning is an appropriate method for identifying the worldly elements in EHR data. We illustrate, through a contextual investigation, how the perform various tasks learning-based model can possibly work on the exhibition of expectation models with regards to recognizing individuals who are at a high gamble of transient renal capability misfortune.

Prevalence of chronic kidney disease in an adult population:

Foundation and targets: Screening programs are one way to prevent and monitor chronic kidney disease (CKD). A grown-up all inclusive community screening project's rate commonness and hazard factors for CKD were the focal point of this review. The strategy for the review is cross-sectional. 600 and ten individuals, 73 percent of whom were ladies and between the ages of 51 and 14 years of age, were assessed. The members were given a survey, a pulse test, and anthropometry. The CKD-EPI computation was utilized to work out the glomerular filtration rate, and an albuminuria dipstick was utilized to break down the pee. Over portion of individuals had diabetes mellitus (DM), hypertension, or stoutness in their families, and 30% had ongoing kidney illness. Diabetes and hypertension were self-detailed by 29% and 19%, separately. 75% of the individuals who were screened were viewed as stout or overweight; Ladies were more probable than men to have a high-risk stomach midriff periphery (87 versus 75 percent) and a pervasiveness of corpulence (41 versus 34%). Men (49%) had more self-detailed and analyzed hypertension than women (38%). G1, 5.9%; G2, 4.5%; G3a, 2.6%; G3b, 1.1%; G4, 0.3%; Additionally, CKD was present in 0.3% of the population in G5. In 2.6%, the glomerular filtration rate diminished gently or reasonably, in 1.1%, tolerably or essentially, and altogether. Strange albuminuria was available in 13% of the patients. Diabetes, hypertension, and male direction all expected CKD.

3. METHODOLOGY

Picture enrollment was used by Hodneland et al. to track down changes in the morphology of the kidney. Vasquez-Spirits and others utilized huge scope CKD information to foster a brain network-based classifier, and the model was 95% precise on their test information. Moreover, the CKD informational index from the UCI ML archive was used in most of past exploration. The researchers, Chen et al. used support vector machines (SVM), k-nearest neighbor (KNN), and delicate autonomous displaying of class similarities, with KNN and SVM achieving 99.7% accuracy. Also, to analyze CKD, they utilized fluffy rule-building master frameworks, fluffy ideal acquainted memory, and fractional least squares discriminant investigation, with models that were precise somewhere in the range of 95.5 percent to 99.6 percent. The finding of CKD has been worked on by their examination.

Disadvantages:

1. The vast majority of them experience the evil impacts of either a confined application range or rather sad accuracy in the strategy used to credit missing information.
2. The mean attribution, which is based on the information's symptomatic classes, is used to fill in the gaps left by the previous models. When the analytic results of the examples are unclear, their method cannot be utilized. In fact, patients may fail certain tests before being analyzed for a variety of reasons.
3. Besides, while missing qualities are available in straight out classes, mean ascription determined information may essentially go astray from the real qualities.

Patients might profit from an early and mistake free conclusion of CKD to keep away from additional medical conditions. These ongoing illnesses are anticipated utilizing different information mining order methods and machine learning (ML) calculations. Logistic Regression, KNN, Random Forest, Decision Tree, SVM, Gradient Boosting, Xgboost, Adaboost, and Ensemble are used in this prediction. The UCI Archive contains 400 informational collections with 21 properties, which are utilized for this review. Characterization calculations were taken care of this data. The aftereffects of the examination show that gradient Supporting, DT, and RF each have an exactness of 97.50%. The Xgboost and Adaboost classifiers have a most extreme precision of 100%.

Advantages:

1. We recommend a way for growing the application scope of CKD indicative models while likewise working on model precision.

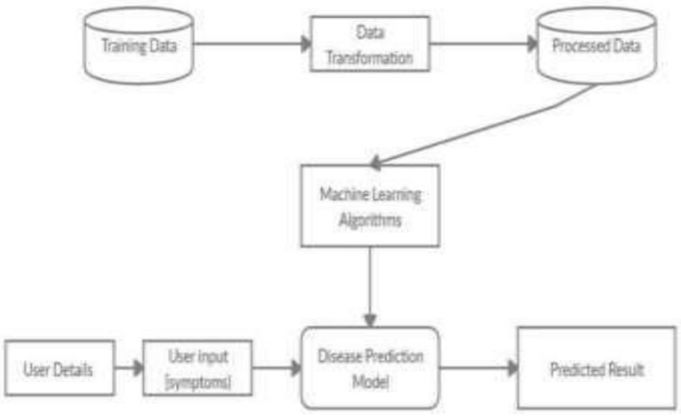


Fig.2: System architecture

MODULES:

We created the accompanying modules for this project.

Data Collection:

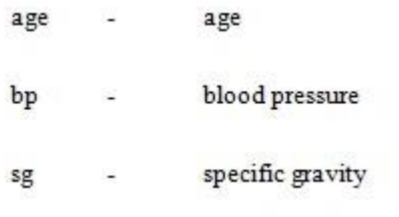
The first significant step toward the actual construction of a machine learning model is the collection of data. This is a crucial phase that will have an impact on the model's success; Our model will perform better the more and better the data we have.

There are various techniques for get-together information, like web based scratching, manual intercessions, etc. This chronic kidney disease dataset was provided by UCI: Chronic kidney disease https://archive.ics.uci.edu/ml/datasets

Dataset:

There are 400 distinct data points in the dataset. The dataset has 26 sections.

Example:



age	-	age
bp	-	blood pressure
sg	-	specific gravity

Data Preparation:

Changes will be made to the information. by erasing information that is absent and certain sections. From that point onward, we will gather a rundown of the segment names that we mean to keep.

The excess segments, in the event that any, are dropped or eliminated.

Last but not least, we get rid of or get rid of the rows in the data collection that don't have any values.

There are training and assessment sets in each set.

Model Selection:

It is a technique for managed learning with additional reliant factors. The consequence of this strategy is a parallel one. A specific piece of information might yield a consistent outcome utilizing coordinated factors relapse. A factual model with parallel factors supports this methodology.

Analyze and Prediction:

In the actual dataset, only 19 characteristics were used: reveals whether the individual has renal disease.

Accuracy on test set:

Our accuracy in the test set was 92.7 percent.

Saving the Trained Model:

Using a library like pickle, you can save your trained and tested model as an.h5 or.pkl file before putting it into a production-ready environment.

4. IMPLEMENTATION

Logistic Regression: A prescient examination is the logistic regression. Data can be presented and the relationship between a single paired variable and at least one apparent, ordinal, stretch, or proportion level free factor can be understood using logistic regression.

Random Forest Algorithm

A regulated order calculation is the Random Forest one. We can tell from its name, which means to create an irregular woodland in a few directions. The potential outcomes are directly proportional to the number of trees present in the woods: The result is more accurate the more trees there are. However, one thing to keep in mind is that using a data gain or gain list approach to build the choice is not the same as building the forest. A choice-assistance tool is the decision tree. The potential outcomes are displayed using a chart that looks like a tree. The decision tree will construct a set of rules if you provide it with a preparation dataset containing targets and elements. Forecasts can be made using these principles. We currently use Random Forest to help us create classes from our dataset because we have it divided into three categories. When you input a preparation dataset with highlights and names into a Random Forest, it will figure out some arrangement of rules that will be used to make the expectations. Random forests are generally groups of choice trees.

Decision Tree Classifier:

Decision Tree is an overseen ML computation used to deal with portrayal issues. The expectation of the target class based on the choice rule derived from previous data is the primary objective of using DecisionTree in this examination work. The character and expectation are served by hubs and internodes. Root hubs arrange the examples according to various features. Leaf hubs deal with order, whereas root hubs can have at least two branches. Decision trees select each hub at each stage by determining which qualities have the highest data gain. Evaluation of the Decision Tree method's display

K NEAREST NEIGHBOR ALGORITHM

One of the most fundamental order calculations in ML is K-Nearest Neighbors. It belongs in the controlled learning environment and has important applications in design recognition, information mining, and interruption location.

It is by and large disposable, in fact, circumstances since it is non-parametric, meaning, it makes no secret assumptions about the scattering of data (as opposed to various estimations, for instance, GMM, which anticipate a Gaussian scattering of the given data).

Adaboost:

AdaBoost, also known as versatile supporting, is a method in machine learning that is used as a gathering technique. Choice trees with one level, or choice trees with just one split, is the most well-known calculation used with AdaBoost. Decision Stumps are another name for these trees.

Xgboost Algorithm:

XGBoost is generally quick. When compared to other strategies for inclination support, this one is extremely quick.

Szilard Pafka played out a few objective benchmarks contrasting the exhibition of XGBoost with different executions of inclination helping and stowed choice trees. He reviewed his outcomes in May 2015 in the blog entry named "Benchmarking Arbitrary Woodland Executions".

He likewise gives a greater report of results with hard numbers.

His outcomes showed that XGBoost was quite often quicker than the other benchmarked executions from R, Python Flash and H2O.

XGBoost rules or even datasets on grouping and relapse mental display issues are organized.

The fact that it is used by contest winners on the Kaggle cutthroat information science stage is evidence of this. The inclination helping choice tree calculation is carried out by the XGBoost library. Numerous different names have been given to this calculation, such as inclination helping, numerous additional substance relapse trees, stochastic slope supporting, and angle helping machines. Helping is an outfit procedure wherein new models are added to address past models' blunders. Models are added consistently until any longer upgrades can't be made. The AdaBoost computation, which loads data centers that are hard to foresee, is a notable model. A technique known as tendency aiding includes making new models that expect the residuals — or blunders — of past models and afterward adding them together to create a definitive forecast. Since it utilizes tendency drop computations to restrict the blunder while adding new models, it is known as incline supporting. Both backslide and arrange perceptive exhibiting issues are upheld by this system.

Ensemble Algorithm

Experimentally, troupes will generally yield improved results when there is a huge variety among the models. Numerous troupe strategies, subsequently, look to advance variety among the models they combineAlthough maybe non-natural, more irregular calculations (like random choice trees) can be utilized to deliver a more grounded group than exceptionally purposeful calculations (like entropy-lessening decision trees).Using an assortment of solid learning calculations, in any case, has been demonstrated to be more powerful than involving procedures that endeavor to simplify the models to elevate diversity.It is feasible to increment variety in the preparation phase of the model involving relationship for relapse undertakings or utilizing data estimates, for example, cross entropy for characterization errands.

SUPPORT VECTOR MACHINE(SVM):

An oversaw ML estimation known as "Support Vector Machine" (SVM) can be utilized to take care of issues with gathering and backslide. Notwithstanding, more often than not, it is utilized in game plan issues. We plot each snippet of data as a point in n-layered space for this computation, where n is the quantity of components you have and the worth of every part is equivalent to the worth of a particular bearing. From that point forward, we perform game plan by finding the hyperplane that plainly isolates the two classes (see the outline underneath). Utilizing a little, the SVM estimation is done. The learning of the hyperplane in direct SVM is finished by changing the issue utilizing some straight polynomial math, which is out of the level of this prelude to SVM.

5. EXPERIMENTAL RESULTS

Fig.3: Home screen

Fig.4: User registration

Fig.5: User login

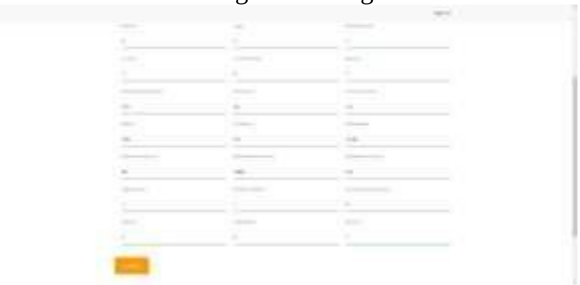


Fig.6: User input

Chronic Kidney Disease Predicton

You are safe! You do not have Disease

Fig.7: Prediction result

6. CONCLUSION

Patients might profit from an early and blunder free determination of CKD to keep away from additional medical issues. These chronic diseases are anticipated utilizing various information mining arrangement procedures and machine learning (ML) calculations. Logistic Regression, KNN, Random Forest, Decision Tree, SVM, Gradient Boosting, Xgboost, Adaboost, and Ensemble are utilized in this expectation. This review makes use of 400 informational indexes with 21 properties from the UCI Store. Classification algorithms were fed this information. The results of the experiment indicate that Gradient Boosting, DT, and RF each have an accuracy of 97.50%. The Xgboost and Adaboost classifiers have a maximum accuracy of 100 percent.

REFERENCES

1. Z. Chen, Z. Zhang, R. Zhu, Y. Xiang, and P. B. Harrington, ``Diagnosis of patients with chronic kidney disease by using two fuzzy classifiers," Chemometrics Intell. Lab. Syst., vol. 153, pp. 140145, Apr. 2016.

2. A. Subasi, E. Alickovic, and J. Kevric, ``Diagnosis of chronic kidney disease by using random forest," in Proc. Int. Conf. Med. Biol. Eng.,Mar. 2017, pp. 589594.

3. L. Zhang, ``Prevalence of chronic kidney disease in China: A crosssectionalsurvey," Lancet, vol. 379, pp. 815822, Mar. 2012.

4. A. Singh, G. Nadkarni, O. Gottesman, S. B. Ellis, E. P. Bottinger, andJ. V. Guttag, ``Incorporating temporal EHR data in predictive models for risk stratication of renal function deterioration," J. Biomed. Informat.,vol. 53, pp. 220228, Feb. 2015.

5. A. M. Cueto-Manzano, L. Cortés-Sanabria, H. R. Martínez-Ramírez,E. Rojas-Campos, B. Gómez-Navarro, and M. Castillero-Manzano,``Prevalence of chronic kidney disease in an adult population," Arch. Med.Res., vol. 45, no. 6, pp. 507513, Aug. 2014.

6 H. Polat, H. D. Mehr, and A. Cetin, ``Diagnosis of chronic kidney disease based on support vector machine by feature selection methods," J. Med.Syst., vol. 41, no. 4, p. 55, Apr. 2017.

7. C. Barbieri, F. Mari, A. Stopper, E. Gatti, P. Escandell-Montero,J. M. Martínez-Martínez, and J. D. Martín-Guerrero, ``A new machine learning approach for predicting the response to anemia treatment in a large cohort of end stage renal disease patients undergoing dialysis," Comput.Biol. Med., vol. 61, pp. 5661, Jun. 2015.

8. V. Papademetriou, E. S. Nylen, M. Doumas, J. Probsteld, J. F. Mann,R. E. Gilbert, and H. C. Gerstein, ``Chronic kidney disease, basal insulin glargine, and health outcomes in people with dysglycemia: The ORIGIN Study,'' Amer. J. Med., vol. 130, no. 12, pp. 1465.e271465.e39, Dec. 2017.

9. N. R. Hill, ``Global prevalence of chronic kidney disease A systematic review and meta-analysis,'' PLoS ONE, vol. 11, no. 7, Jul. 2016,Art. no.
e0158765.

10. M. M. Hossain, R. K. Detwiler, E. H. Chang, M. C. Caughey, M.W. Fisher,T. C. Nichols, E. P. Merricks, R. A. Raymer, M. Whitford, D. A. Bellinger,L. E. Wimsey, and C. M. Gallippi, ``Mechanical anisotropy assessment in kidney cortex using ARFI peak displacement: Preclinical validation and pilot in vivo clinical results in kidney allografts,'' IEEE Trans. Ultrason.,Ferroelectr., Freq. Control, vol. 66, no. 3, pp. 551562, Mar. 2019.

11. M. Alloghani, D. Al-Jumeily, T. Baker, A. Hussain, J. Mustana, andA. J. Aljaaf, ``Applications of machine learning techniques for software engineering learning and early prediction of students' performance,'' in Proc. Int. Conf. Soft Comput. Data Sci., Dec. 2018, pp. 246258.

12. D. Gupta, S. Khare, and A. Aggarwal, ``A method to predict diagnostic codes for chronic diseases using machine learning techniques,'' in Proc.Int. Conf. Comput., Commun. Autom.(ICCCA), Apr. 2016, pp. 281287.

13. L. Du, C. Xia, Z. Deng, G. Lu, S. Xia, and J. Ma, ``A machine learning based approach to identify protected health information in Chinese clinical text,'' Int. J. Med. Informat., vol. 116, pp. 2432, Aug. 2018.

14. R. Abbas, A. J. Hussain, D. Al-Jumeily, T. Baker, and A. Khattak, ``Classification of foetal distress and hypoxia using machine learning approaches,''in Proc. Int. Conf. Intell.Comput., Jul. 2018, pp. 767776.

15. M. Mahyoub, M. Randles, T. Baker, and P. Yang, ``Comparison analysis of machine learning algorithms to rank alzheimer's disease risk factors by importance,'' in Proc. 11th Int. Conf. Develop. eSyst. Eng. (DeSE),Sep. 2018, pp. 111.