CHRONIC KIDNEY DISEASE ANALYSIS

Using DECISION TREE CLASSIFICATION ALGORITHM

Developed by: Gaurav Soni, Pramod Poojar, Surendra K V, Abhishek B P

SmartBridge - Remote Summer Internship Program

1. INTRODUCTION

Chronic Kidney Disease (CKD) is a major medical problem and can be cured if treated it in the early stages. Usually, people are not aware that medical tests, we take for different purposes could contain valuable information concerning kidney diseases. Consequently, attributes of various medical tests are investigated to distinguish which attributes may contain helpful information about the disease. The information says that it helps us to measure the severity of the problem, the predicted survival of the patient after the illness, the pattern of the disease and work for curing the disease.

Chronic kidney disease can affect your body's ability to clean your blood, filter extra water out of your blood, and help control your blood pressure. It can also affect red blood cell production and vitamin D metabolism needed for bone health. You are born with two kidneys. They're on either side of your spine, just above your waist. When your kidneys are damaged, waste products and fluid can build up in your body. That can cause swelling in your ankles, nausea, weakness, poor sleep, and shortness of breath. Without treatment, the damage can get worse and your kidneys may eventually stop working. That's serious, and it can be life-threatening.

What Your Kidneys Do

Healthy kidneys:

- Keep a balance of water and minerals (such as sodium, potassium, and phosphorus) in your blood.
- Remove waste from your blood after digestion, muscle activity, and exposure to chemicals or medications.
- Make renin, which your body uses to help manage your blood pressure.
- Make a chemical called erythropoietin, which prompts your body to make red blood cells.
- Make an active form of vitamin D, needed for bone health and other things.

Kidney Problems

- Kidney stones
- Chronic kidney disease

Chronic kidney disease causes:

When your kidneys don't work well for longer than 3 months, doctors call it chronic kidney disease. You may not have any symptoms in the early stages, but that's when it’s simpler to treat. Diabetes (types 1 and 2) and high blood pressure are the most common culprits. High blood sugar levels over time can harm your kidneys. And high blood pressure creates wear and tear on your blood vessels, including those that go to your kidneys.

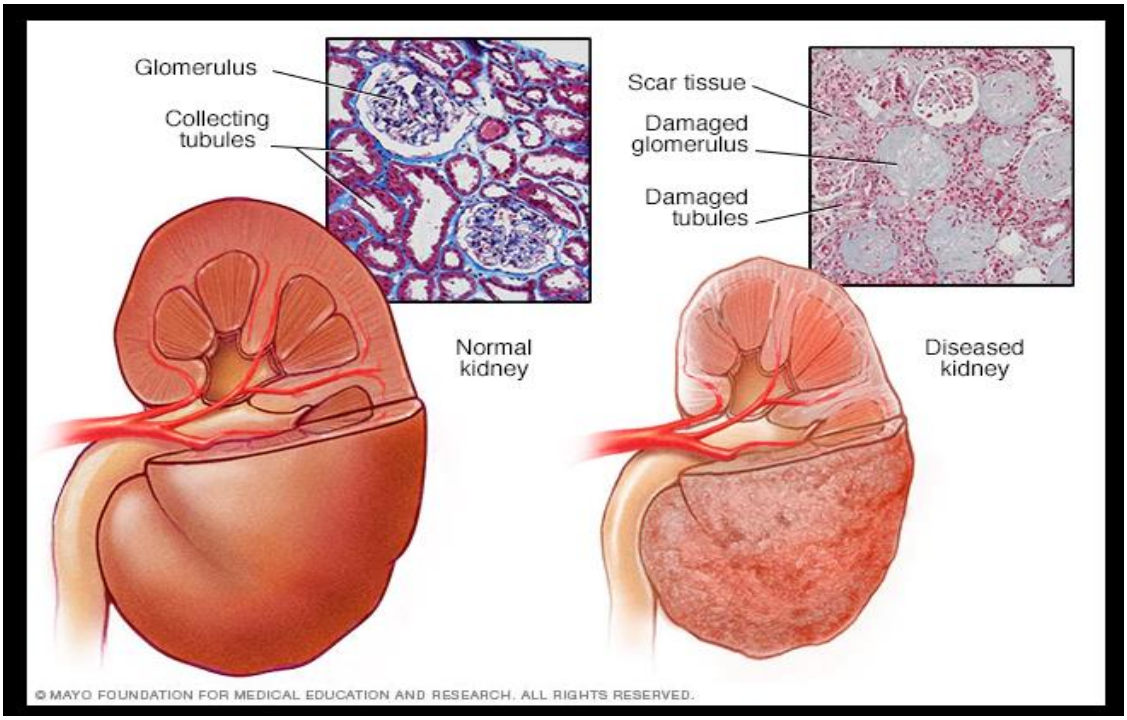


Fig. 1: Difference between normal kidney and diseased kidney

To predict and analyze the chronic diseases such as kidney, diabetic, cancer, and heart diseases, there are several proposed machine learning algorithms that can be used. These algorithms include the decision tree (DT), SVM, ANN, linear regression (LR), KNN, NB, and time series prediction models. Because of the rapid innovation and continuous changes in software engineering, a huge volume of information can be generated. With the development of a healthcare database management system, there will be more opportunities for the enhancement of the healthcare systems. Extracting patterns from these datasets and managing large amounts of dimensionality data have become a major field of machine learning. The machine learning algorithm is considered to be the

classification of healthcare datasets to obtain useful knowledge that can help health officials and communities. To apply machine learning algorithms that enhance the performance of the classification process, the preprocessing of the soft clustering algorithm is required.

1.1 Overview

In the field of biomedical and healthcare communities the accurate prediction plays the major role to find out the risk of the disease in the patient . The prediction gives the benefits of early disease detection. However the analysis accuracy as the relationship with the condition of the medical data, thus the poor condition of the medical data leads to less accuracy of prediction. Here we use a certain machine learning algorithm to state the rate of disease. Prediction process is done using the dataset provided from certain hospitals, the entire dataset will be preprocessed and the missing values will be reconstructed. Compared to several types of prediction algorithms, the Decision Tree algorithm gives the highest accuracy of prediction around 97.5% with a convergence speed.

1.2 Purpose

Our aim from the project is to make use of pandas, matplotlib, numpy, tensorflow, keras libraries from python to extract the libraries for machine learning for the chronic disease prediction. Also, we do the hyperparameter tuning to achieve better accuracy. And finally predict whether the person have chronic disease or not and laying out the conclusion.

2.LITERATURE SURVEY

Many studies explore and analyze chronic diseases using various techniques for early diagnosis. Some of the surveys explore various data mining techniques for their detection accuracy, including logistic regression, multilayer perception, ANN, decision table, radical basic function, naive Bayes, k-nearest neighbor, and sequential minimal optimization. Depending on the type of dataset, such techniques show differences in the level of accuracy, and there is no single rule for the best result.

The research we have done suggests that machine learning provides important insights into data and can help classify data into different classes. The findings indicate that machine learning techniques can produce accurate classification results if used in conjunction with feature selection techniques. Therefore, retaining the benefits of classification results for machine learning techniques,

this study employs a set of the most popular machine learning techniques in combination with feature selection technique to classify normal and kidney disease patients.

2.1 Existing problem

Nowadays the chronic diseases became threat in all countries, in every country one third of people is suffering from chronic diseases. The chronic diseases diagnosis are costlier, it's difficult to poorly people. And in medical a lot of chronic disease dataset are collected and stored, the data mining helps in early detection of the disease. Chronic disease such as kidney disease, diabetes, liver disease, Alzheimer, Parkinson's, are most expensive diagnosis disease. It's a big challenge in medical canter or healthcare center to provide best quality treatment to all patients, as only affordable patients can use good quality of treatment. To minimize the cost the data mining approaches can help to take decision and detect the disease in earlier so that every patient can afford a good quality of treatment.

A huge amount of healthcare data, are available which are not mined in better and accurate way, to discover hidden information for effective decision making. The proposed system uses data mining techniques to diagnose early for the chronic diseases.

2.2 Proposed Solution

Data mining is the process of analyzing data from different perspectives and extracting useful knowledge from it. It is the core of knowledge discovery process. The various steps involved in extracting knowledge from raw data.

Different data mining techniques include classification, clustering, association rule mining, prediction and sequential patterns, neural networks, regression etc. Classification is the most commonly applied data mining technique, which employs a set of pre-classified examples to develop a model that can classify the population of records at large. Fraud detection and credit risk applications are particularly well suited to classification technique.

This approach frequently employs Decision tree based classification Algorithm. In classification, a training set is used to build the model as the classifier which can classify the data items into its appropriate classes. A test set is used to validate the model.

3. THEORETICAL ANALYSIS

While selecting the algorithm that gives an accurate prediction we gone through lot of algorithms which gives the results abruptly accurate and from them we selected only one algorithm for the prediction problem that is Decision tree Classifier, it assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. That's how the prediction work great with the Decision tree Algorithm.

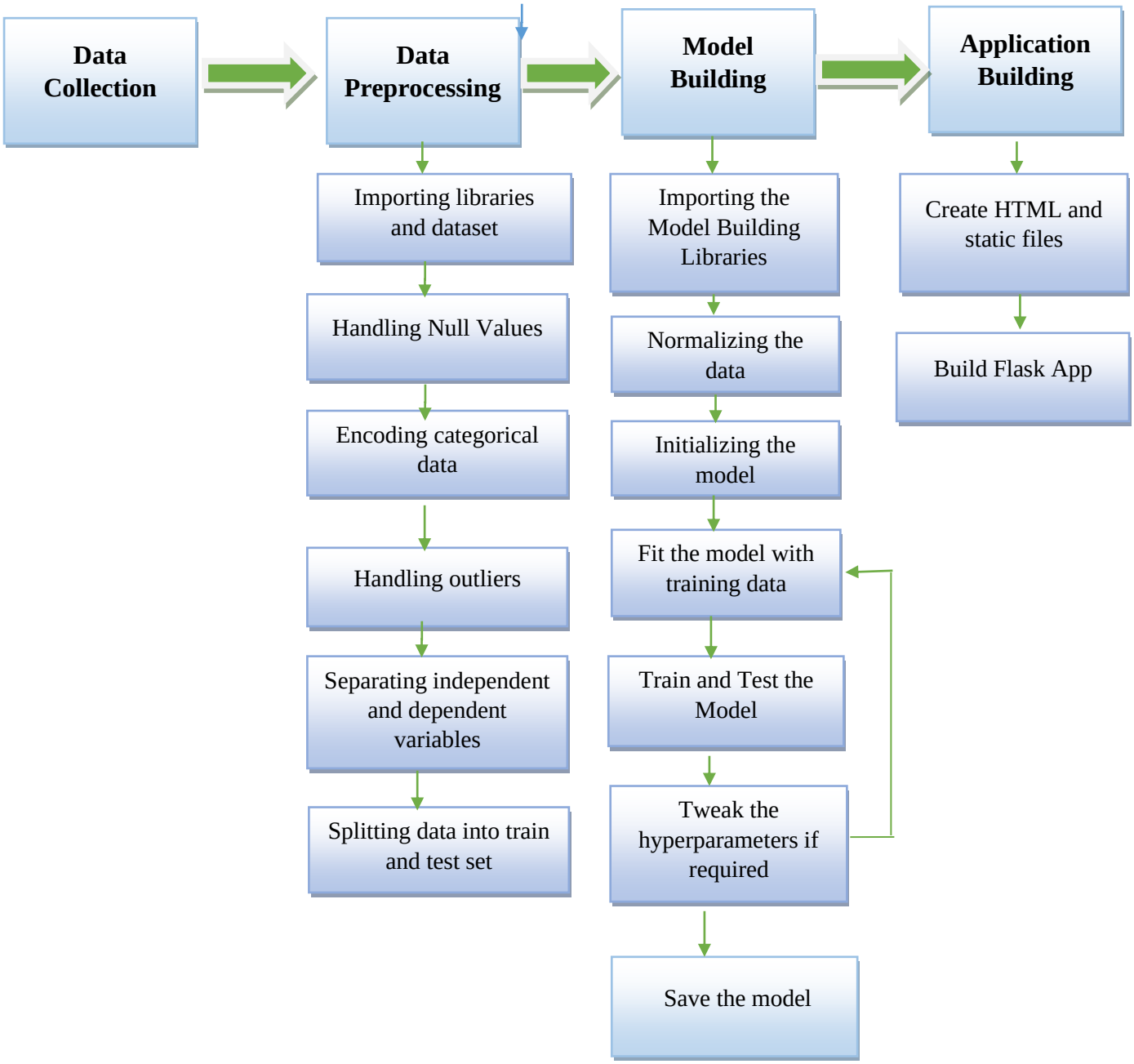
Accuracy is defined as the ratio of the number of samples correctly classified by the classifier to the total number of samples for a given test data set. At first we got like lot of worst accuracies because we tried lot of algorithms for the best accurate algorithm , finally after all of that we tried the best suitable algorithm which gives the prediction accurately is Decision tree Classifier. And developed it to use as a real time prediction problem for the loan status prediction.

In statistics, a receiver operating characteristic (ROC), is a two dimensional graphical plot that illustrates the performance of a binary classifier system. The curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. Using Z-score we have removed outliers to get high accuracy.

3.1 Software Designing

- Jupyter Notebook Environment
- Spyder
- Machine Learning Algorithms
- Python (Pipeline, StandardScaler, DecisionTreeClassifier)
- HTML5, CSS3, Bootstrap
- Flask
- We developed this kidney disease analysis by using the Python language, which is a high level programming language along with Machine Learning Algorithm such as Decision Tree. For coding we used the Jupyter Notebook of Anaconda distributions and Spyder, an integrated scientific programming in python language. Flask is used as a user interface for the prediction. Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser.

3.2 Block Diagram



4. Experimental Investigation

In our project, we have used the chronic kidney disease dataset. The data that is used in this project originally comes from the kaggle machine learning dataset . The data consists of over 400 records from patients with CKD and it was collected from the hospitals in India.

5. DATASET DESCRIPTION

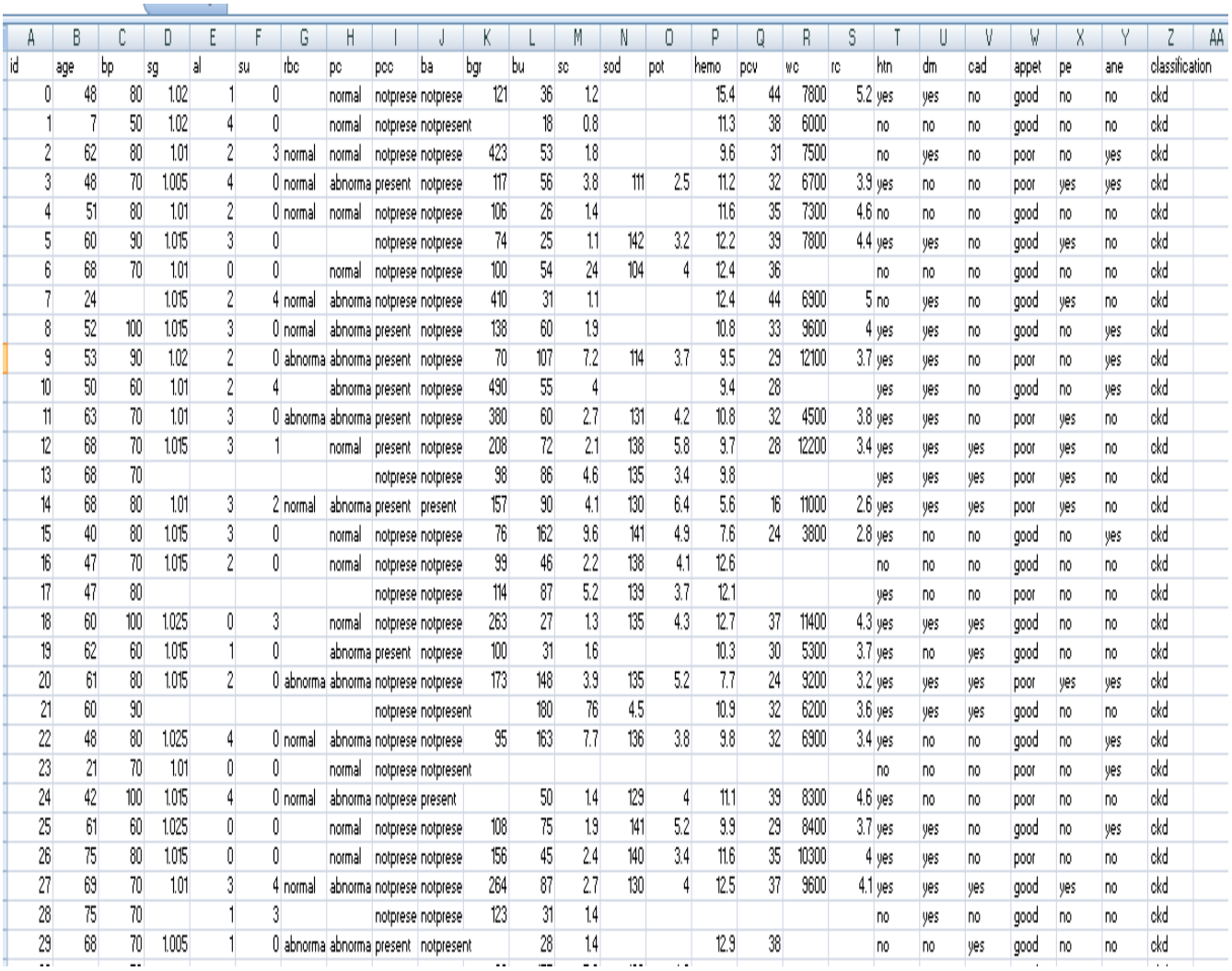
We use the following representation to collect the dataset
age - age
bp - blood pressure
sg - specific gravity
al - albumin
su - sugar
rbc - red blood cells
pc - pus cell
pcc - pus cell clumps
ba - bacteria
bgr - blood glucose random
bu - blood urea
sc - serum creatinine
sod - sodium
pot - potassium
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
classification – classification

Attribute Information:

We use 25 features = 25 (11 numeric ,14 nominal)
1.Age(numerical)
age in years

2.Blood Pressure(nominal)
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.Red Blood Cells(nominal)
rbc - (normal,abnormal)
7.Pus Cell (nominal)
pc - (normal,abnormal)
8.Pus Cell clumps(nominal)
pcc - (present,notpresent)
9.Bacteria(nominal)
ba - (present,notpresent)
10.Blood Glucose Random(numerical)
bgr in mgs/dl
11.Blood Urea(numerical)
bu in mgs/dl
12.Serum Creatinine(numerical)
sc in mgs/dl
13.Sodium(numerical)
sod in mEq/L
14.Potassium(numerical)
pot in mEq/L
15.Hemoglobin(numerical)
hemo in gms
16.Packed Cell Volume(numerical)
17.White Blood Cell Count(numerical)
wc in cells/cumm
18.Red Blood Cell Count(numerical)
rc in millions/cmm
19.Hypertension(ordinal)
htn - (yes,no)
20.Diabetes Mellitus(ordinal)
dm - (yes,no)
21.Coronary Artery Disease(ordinal)
cad - (yes,no)
22.Appetite(ordinal)
appet - (good,poor)
23.Pedal Edema(ordinal)
pe - (yes,no)

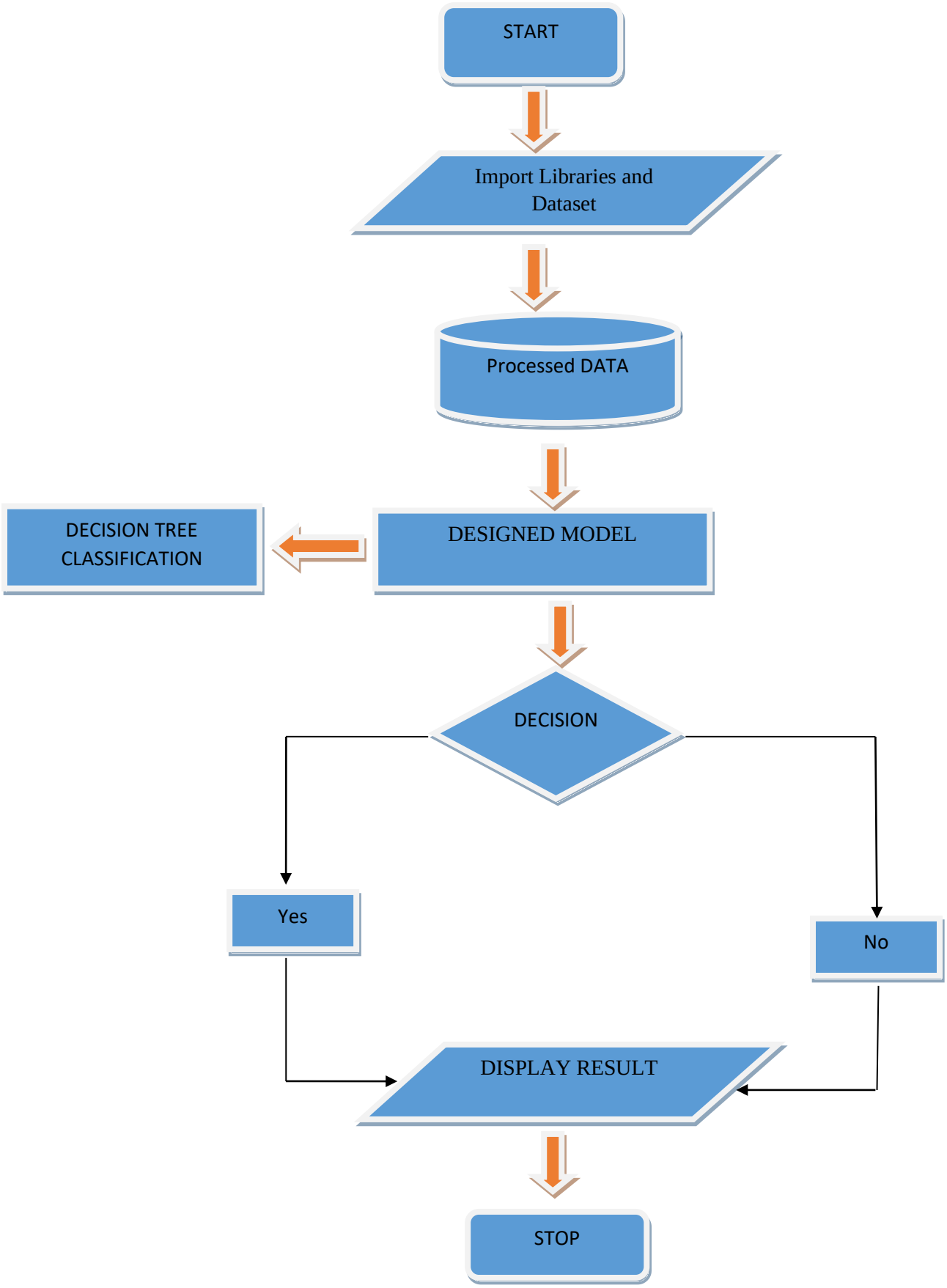
24.Anemia(ordinal)
ane - (yes,no)
25.Classification (ordinal)
classification - (ckd,notckd)



id	age	bp	sg	al	su	rbc	pc	pcc	ba	bgr	bu	sc	sod	pot	hemo	pcv	wc	rc	htn	dm	cad	appet	pe	ane	classification
0	48	80	1.02	1	0		normal	notprese	notprese	121	36	1.2			15.4	44	7800	5.2	yes	yes	no	good	no	no	ckd
1	7	50	1.02	4	0		normal	notprese	notpresent		18	0.8			11.3	38	6000		no	no	no	good	no	no	ckd
2	62	80	1.01	2	3	normal	normal	notprese	notprese	423	53	1.8			9.6	31	7500		no	yes	no	poor	no	yes	ckd
3	48	70	1.005	4	0	normal	abnorma	present	notprese	117	56	3.8	111	2.5	11.2	32	6700	3.9	yes	no	no	poor	yes	yes	ckd
4	51	80	1.01	2	0	normal	normal	notprese	notprese	106	26	1.4			11.6	35	7300	4.6	no	no	no	good	no	no	ckd
5	60	90	1.015	3	0			notprese	notprese	74	25	1.1	142	3.2	12.2	39	7800	4.4	yes	yes	no	good	yes	no	ckd
6	68	70	1.01	0	0		normal	notprese	notprese	100	54	24	104	4	12.4	36			no	no	no	good	no	no	ckd
7	24		1.015	2	4	normal	abnorma	notprese	notprese	410	31	1.1			12.4	44	6900	5	no	yes	no	good	yes	no	ckd
8	52	100	1.015	3	0	normal	abnorma	present	notprese	138	60	1.9			10.8	33	9600	4	yes	yes	no	good	no	yes	ckd
9	53	90	1.02	2	0	abnorma	abnorma	present	notprese	70	107	7.2	114	3.7	9.5	29	12100	3.7	yes	yes	no	poor	no	yes	ckd
10	50	60	1.01	2	4		abnorma	present	notprese	490	55	4			9.4	28			yes	yes	no	good	no	yes	ckd
11	63	70	1.01	3	0	abnorma	abnorma	present	notprese	380	60	2.7	131	4.2	10.8	32	4500	3.8	yes	yes	no	poor	yes	no	ckd
12	68	70	1.015	3	1		normal	present	notprese	208	72	2.1	138	5.8	9.7	28	12200	3.4	yes	yes	yes	poor	yes	no	ckd
13	68	70						notprese	notprese	98	86	4.6	135	3.4	9.8				yes	yes	yes	poor	yes	no	ckd
14	68	80	1.01	3	2	normal	abnorma	present	present	157	90	4.1	130	6.4	5.6	16	11000	2.6	yes	yes	yes	poor	yes	no	ckd
15	40	80	1.015	3	0		normal	notprese	notprese	76	162	9.6	141	4.9	7.6	24	3800	2.8	yes	no	no	good	no	yes	ckd
16	47	70	1.015	2	0		normal	notprese	notprese	99	46	2.2	138	4.1	12.6				no	no	no	good	no	no	ckd
17	47	80						notprese	notprese	114	87	5.2	139	3.7	12.1				yes	no	no	poor	no	no	ckd
18	60	100	1.025	0	3		normal	notprese	notprese	263	27	1.3	135	4.3	12.7	37	11400	4.3	yes	yes	yes	good	no	no	ckd
19	62	60	1.015	1	0		abnorma	present	notprese	100	31	1.6			10.3	30	5300	3.7	yes	no	yes	good	no	no	ckd
20	61	80	1.015	2	0	abnorma	abnorma	notprese	notprese	173	148	3.9	135	5.2	7.7	24	9200	3.2	yes	yes	yes	poor	yes	yes	ckd
21	60	90						notprese	notpresent		180	76	4.5		10.9	32	6200	3.6	yes	yes	yes	good	no	no	ckd
22	48	80	1.025	4	0	normal	abnorma	notprese	notprese	95	163	7.7	136	3.8	9.8	32	6900	3.4	yes	no	no	good	no	yes	ckd
23	21	70	1.01	0	0		normal	notprese	notpresent										no	no	no	poor	no	yes	ckd
24	42	100	1.015	4	0	normal	abnorma	notprese	present		50	1.4	129	4	11.1	39	8300	4.6	yes	no	no	poor	no	no	ckd
25	61	60	1.025	0	0		normal	notprese	notprese	108	75	1.9	141	5.2	9.9	29	8400	3.7	yes	yes	no	good	no	yes	ckd
26	75	80	1.015	0	0		normal	notprese	notprese	156	45	2.4	140	3.4	11.6	35	10300	4	yes	yes	no	poor	no	no	ckd
27	69	70	1.01	3	4	normal	abnorma	notprese	notprese	264	87	2.7	130	4	12.5	37	9600	4.1	yes	yes	yes	good	yes	no	ckd
28	75	70		1	3			notprese	notprese	123	31	1.4							no	yes	no	good	no	no	ckd
29	68	70	1.005	1	0	abnorma	abnorma	present	notpresent		28	1.4			12.9	38			no	no	yes	good	no	no	ckd

Fig. 2: Dataset

5.FLOWCHART:


START
Import Libraries and Dataset
Processed DATA
DESIGNED MODEL
DECISION TREE CLASSIFICATION
DECISION
Yes
No
DISPLAY RESULT
STOP

6.RESULT

Here, Decision tree algorithm is used to predict the whether person is suffering from ckd or not. It results 1 or 0 as output.

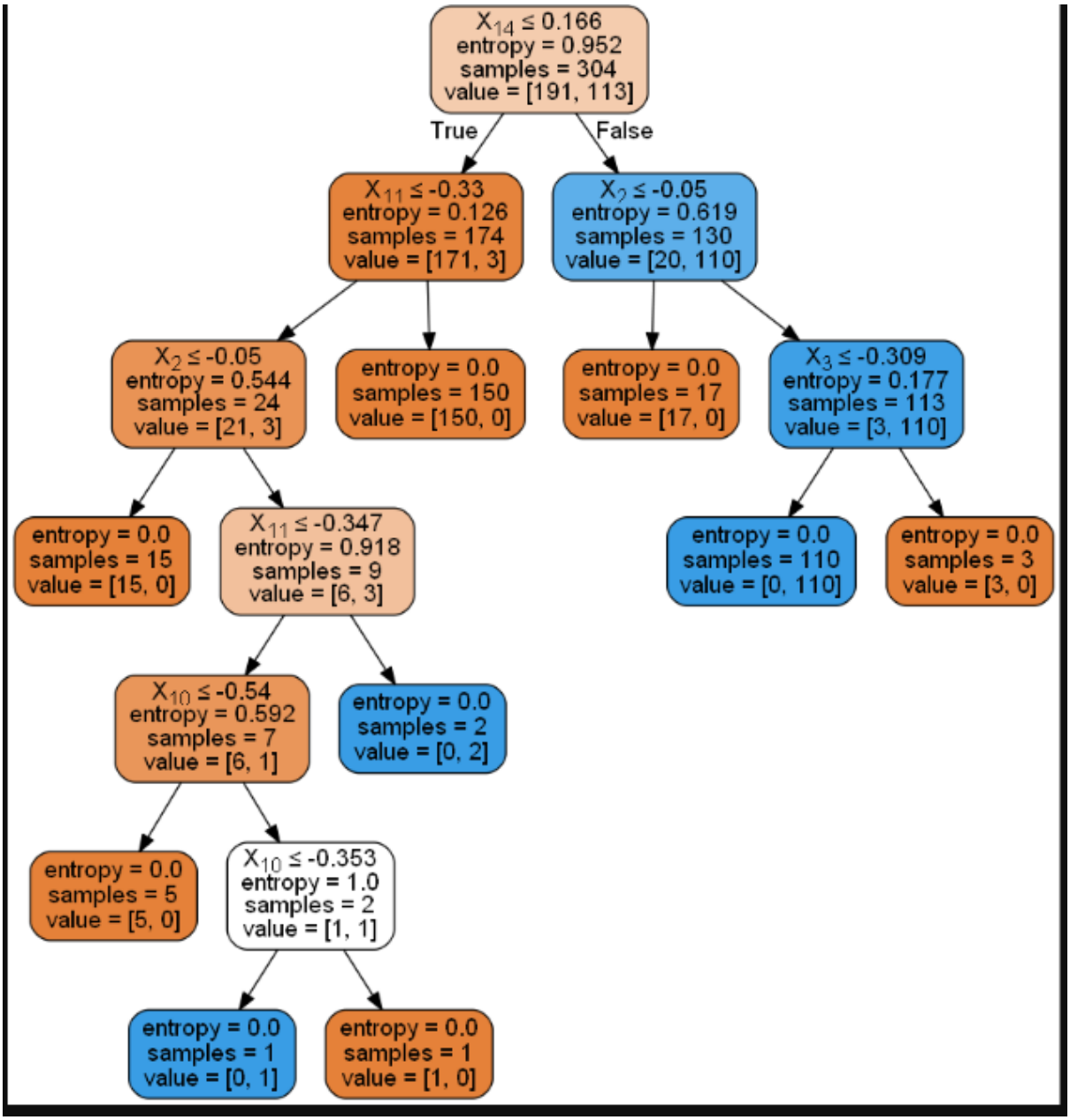


Fig. 3: Decision tree

In this project, the DTC algorithm is used to predict its performance. The results show 97.4% accuracy.

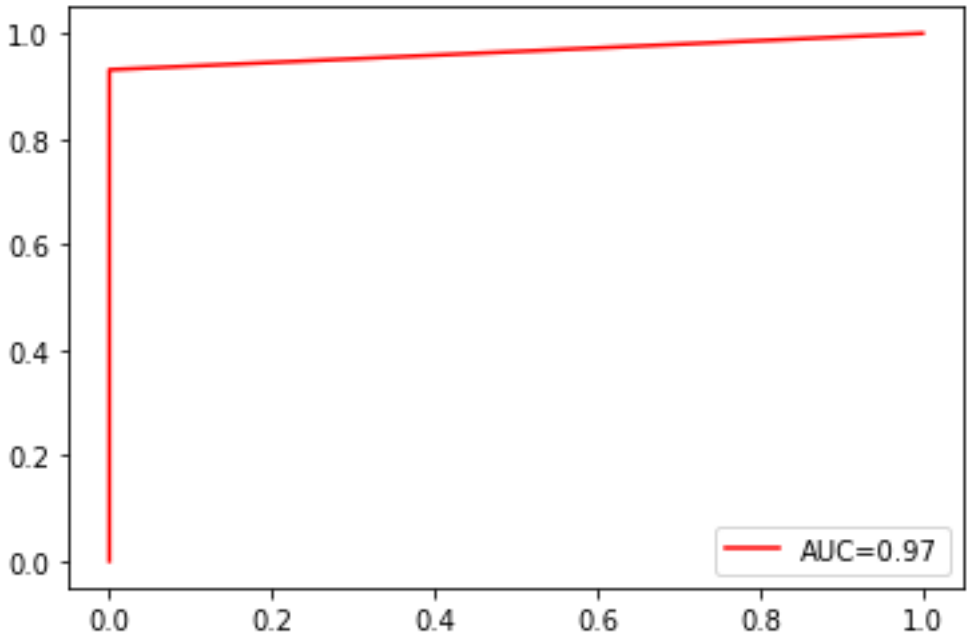


Fig. 4: AUC – ROC Curve

Snapshots:

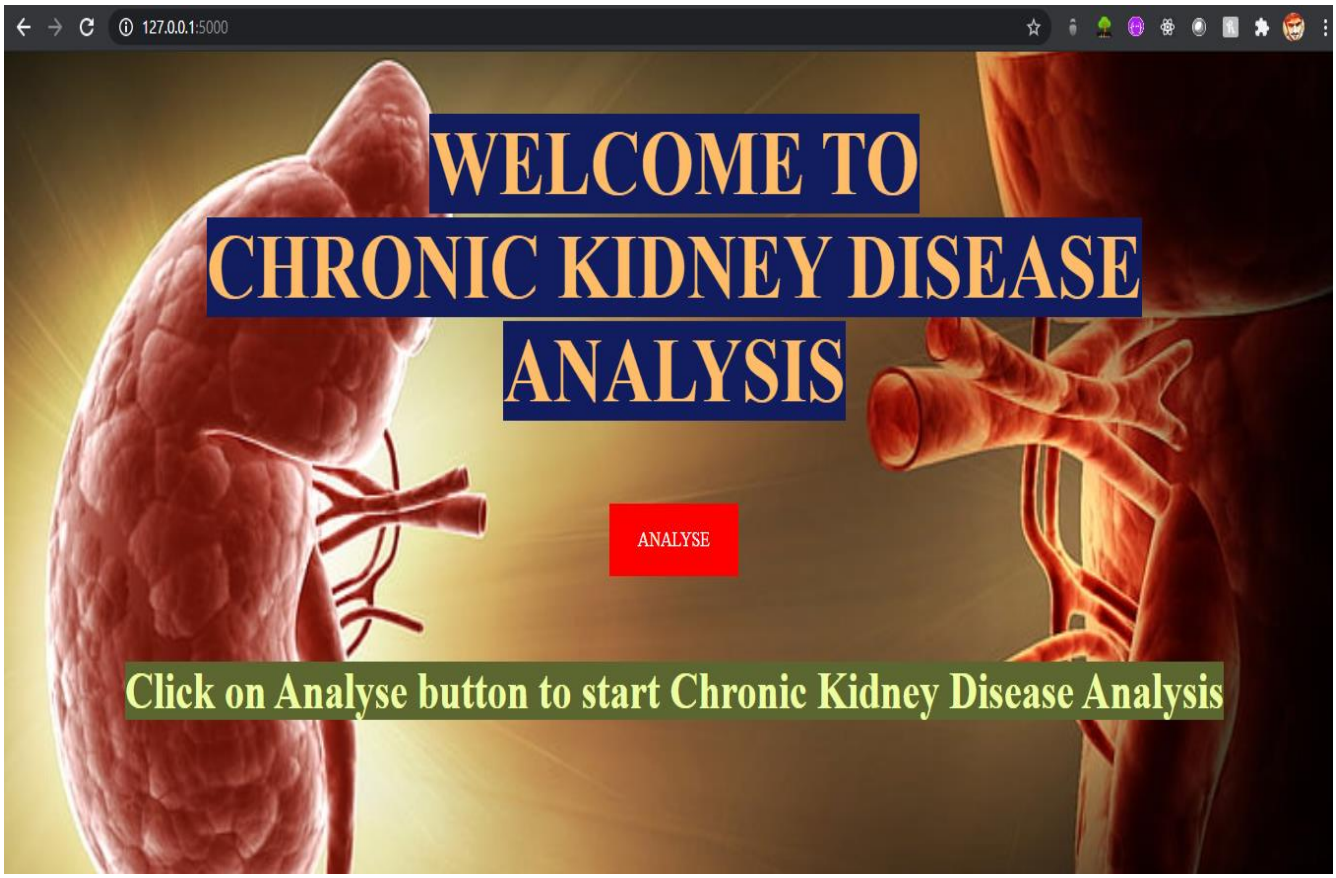


Fig:5 - Home Page

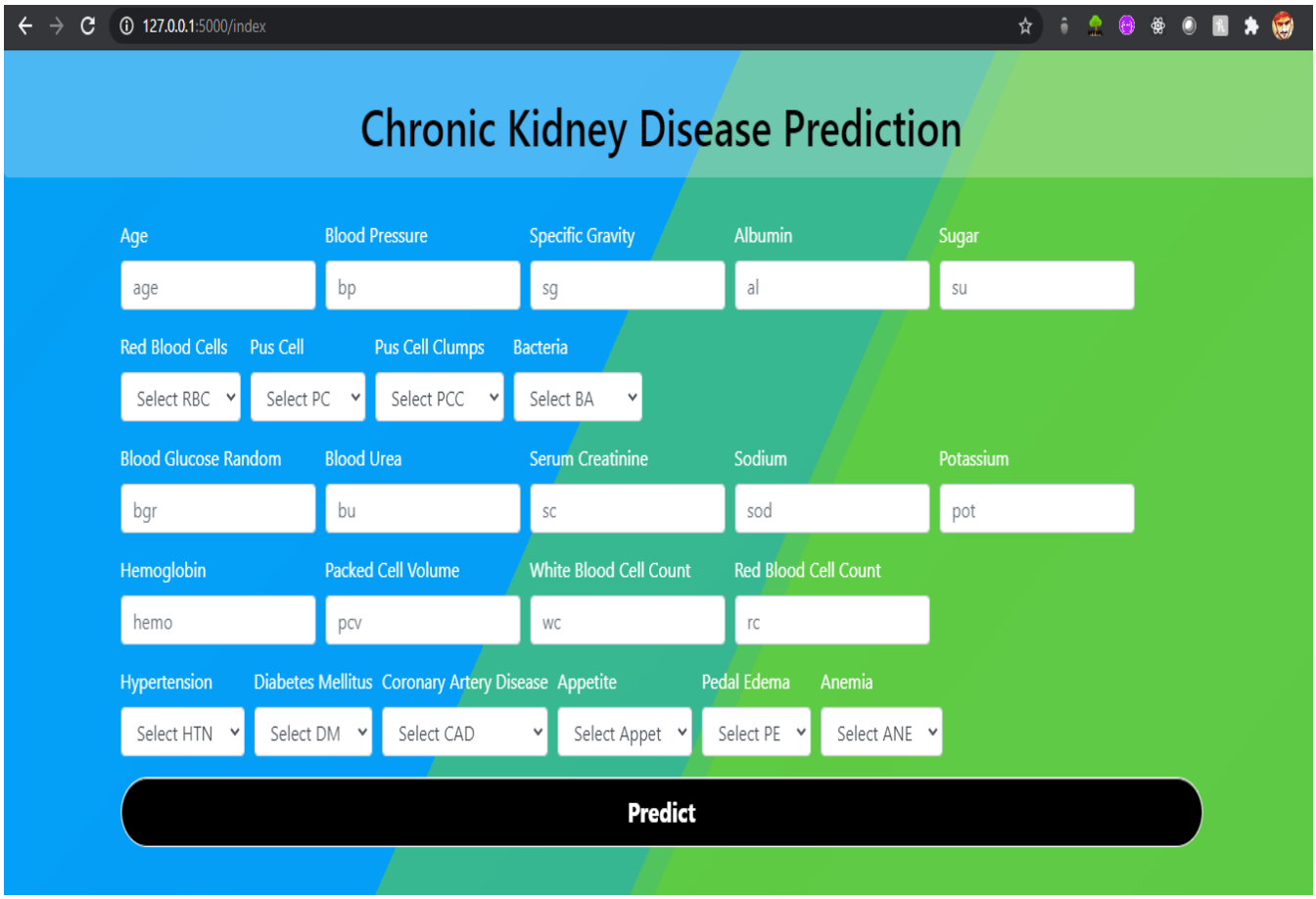


Fig:6 - When you clicked on ANALYSE in Fig:5, it will redirect to http://127.0.0.1:5000/index .

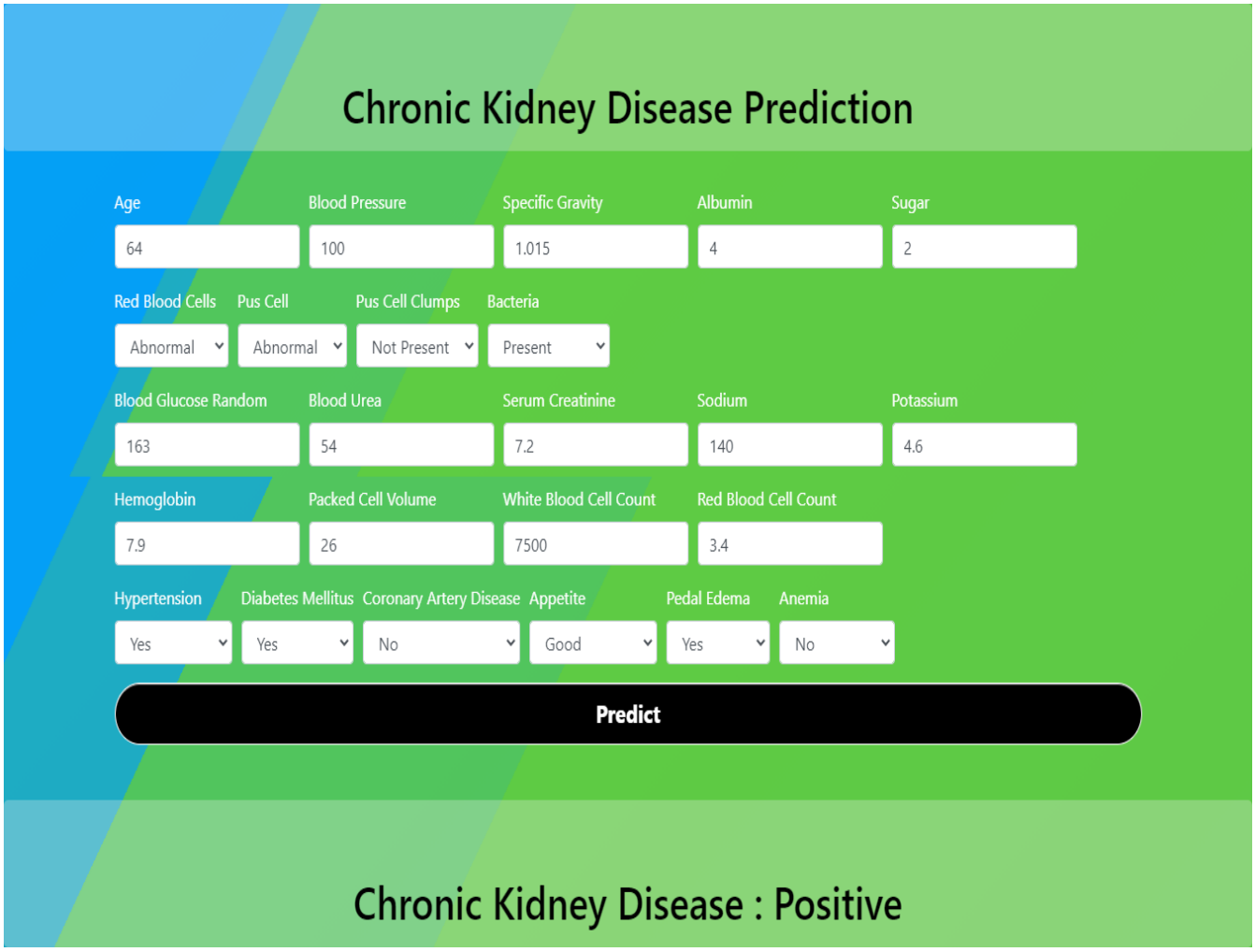


Fig:7 - Positive Classification Example : indicates that the patient has Chronic Kidney Disease.

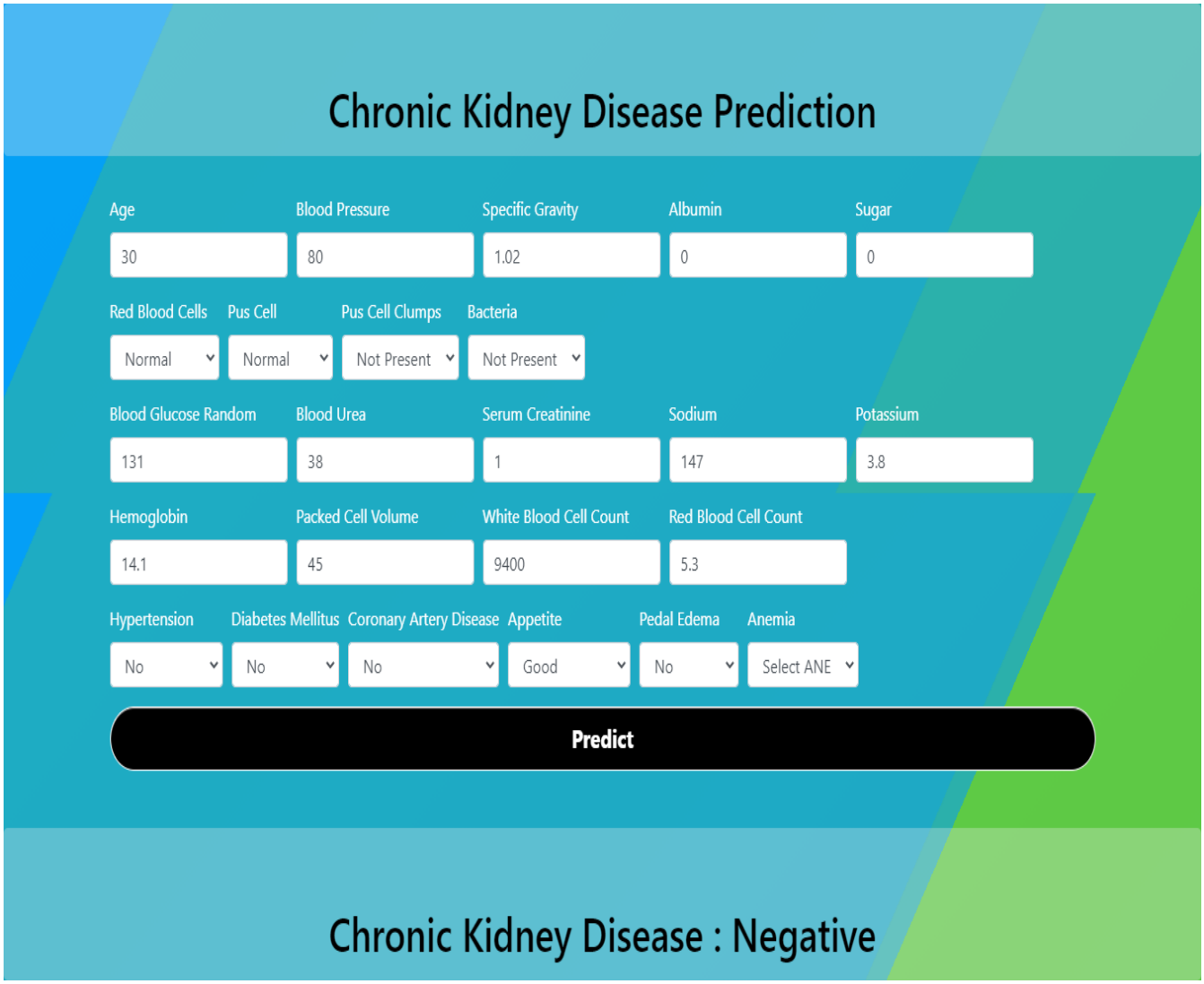


Fig:8 - Negative Classification Example : indicates that the patient doesn't has Chronic Kidney Disease.

7. ADVANTAGES AND DISADVANTAGES

Advantages:

1. Decision Tree Classification Algorithm is easy to implement and understand.

2. It operate in real-time due to low time complexity.

3. It is applicable in training and test-time

4. A Decision trees model is very intuitive and easy to explain to technical teams as well as stakeholders.

Disadvantages:

1. A small change in the data can cause a large change in the structure of the decision tree causing instability.

2. Decision tree often involves higher time to train the model.

8. APPLICATIONS:

1. "Chronic Kidney Disease analysis using Decision Tree Classification algorithm" simplifies the management process of check-up by deploying a web interface to the users.

2. Fast processing and immediate results with high accuracy.

3. Minimizing human effort and cost efficient databases.

4. Navigation through the site is easy.

9. CONCLUSION:

This projects consists of the details about the model which was used for the prediction of CKD using the patients' data .From the resultant graphs, it is proven that the accuracy of the model has reached good level, if it is deployed in the real-time scenario then it will help many people in diagnosing the CKD without wasting the money on check-up. If the CKD is confirmed by the model, then the person can reach the nearest hospital to get the treatment. It can be the best way of practice for people to save money. As we know that the data plays a crucial role in every machine learning model, if the data is more specific and accurate about the symptoms of the CKD then that can help in reaching greater accuracy with better results in real-time applications.

10. FUTURE SCOPE:

Future scope lies in coming up with a prediction model that would factor in the more clinical data (use of specific drugs / associated comorbidities / dietary interventions / degree of blood pressure control / degree of blood sugar control) in predicting the outcomes and providing a possible chance for us to tailor the therapeutic interventions accordingly.

We need to realise that the ML algorithms our study provides need to be considered as a possible screening tool to predict the time frame of progression of CKD patient before he/she would need RRT.

11. BIBLIOGRAPHY

1. Hill NR, Fatoba ST, Oke JL, Hirst JA, O'Callaghan CA, Lasserson DS, et al. Global prevalence of chronic kidney disease—A systematic review and meta-analysis. PloS one. 2016;11(7):e0158765. pmid:27383068

2. Tsai MH, Hsu CY, Lin MY, Yen MF, Chen HH, Chiu YH, et al. Incidence, prevalence, and duration of chronic kidney disease in Taiwan: Results from a community-based screening program of 106,094 individuals. Nephron. 2018;140(3):175–184. pmid:30138926

3. Wu MY, Wu MS. Taiwan renal care system: A learning health-care system. Nephrology. 2018;23:112–115.

4. Eknoyan G, Lameire N, Barsoum R, Eckardt KU, Levin A, Levin N, et al. The burden of kidney disease: Improving global outcomes. Kidney International. 2004;66(4):1310–1314. pmid:15458424

5. Saran R, Robinson B, Abbott KC, Agodoa LY, Albertus P, Ayanian J, et al. US renal data system 2016 annual data report: Epidemiology of kidney disease in the United States.

American Journal of Kidney Diseases. 2017;69(3):A7–A8. pmid:28236831

6. Jha V, Garcia-Garcia G, Iseki K, Li Z, Naicker S, Plattner B, et al. Chronic kidney disease: Global dimension and perspectives. The Lancet. 2013;382(9888):260–272.

7. United States Renal Data System. 2015 USRDS annual data report: Epidemiology of kidney disease in the United States; 2015.

8. Gøransson LG, Bergrem H. Consequences of late referral of patients with end-stage renal disease. Journal of Internal Medicine. 2001;250(2):154–159.

APPENDIX

HTML FILE:

```
<!doctype html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

    <!-- Bootstrap CSS -->
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" integrity="sha384-JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z" crossorigin="anonymous">
    <link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">
    <title>ML API</title>
  </head>
  <body>
    <div class="bg"></div>
    <div class="bg bg2"></div>
    <div class="bg bg3"></div>
    <h1 class="jumbotron" style="text-align: center;">Chronic Kidney Disease Prediction</h1>
    <div class="container">

    <form action="{{ url_for('y_predict')}}"method="post">
        <div class="form-row">
            <div class="form-group col-auto">
                <label>Age</label>
                <input type="number" class="form-control" name="age" placeholder="age" required="required" />
```

```
        </div>
        <div class="form-group col-auto">
          <label>Blood Pressure</label>
          <input type="number" class="form-control" name="bp" placeholder="bp" required="required" />
        </div>
        <div class="form-group col-auto">
          <label>Specific Gravity</label>
          <input type="float" class="form-control" name="sg" placeholder="sg" required="required" />
        </div>
        <div class="form-group col-auto">
          <label>Albumin</label>
          <input type="number" class="form-control" name="al" placeholder="al" required="required" />
        </div>
        <div class="form-group col-auto">
          <label>Sugar</label>
          <input type="number" class="form-control" name="su" placeholder="su" required="required" />
        </div>
      </div>
      <div class="form-row">
        <div class="form-group col-auto">
          <label>Red Blood Cells</label>
              <select class="form-control" name="rbc" required="required">
                <option value="">Select RBC </option>
                <option value=1>Normal</option>
                <option value=0>Abnormal</option>
              </select>
        </div>
        <div class="form-group col-auto">
```

```
<label>Pus Cell</label>
<select class="form-control" name="pc" required="required">
<option value="">Select PC </option>
<option value=1>Normal</option>
<option value=0>Abnormal</option>
</select>
</div>
<div class="form-group col-auto">
<label>Pus Cell Clumps</label>
<select class="form-control" name="pcc" required="required">
<option value="">Select PCC </option>
<option value=0>Not Present</option>
<option value=1>Present</option>
</select>
</div>
<div class="form-group col-auto">
<label>Bacteria</label>
<select class="form-control" name="ba" required="required">
<option value="">Select BA </option>
<option value=0>Not Present</option>
<option value=1>Present</option>
</select>
</div>
</div>
<div class="form-row">
<div class="form-group col-auto">
<label>Blood Glucose Random</label>
<input type="number" class="form-control" name="bgr" placeholder="bgr" required="required" />
</div>
<div class="form-group col-auto">
```

```
<label>Blood Urea</label>
<input type="number" class="form-control" name="bu" placeholder="bu" required="required" />
</div>
<div class="form-group col-auto">
<label>Serum Creatinine</label>
<input type="float" class="form-control" name="sc" placeholder="sc" required="required" />
</div>
<div class="form-group col-auto">
<label>Sodium</label>
<input type="number" class="form-control" name="sod" placeholder="sod" required="required" />
</div>
<div class="form-group col-auto">
<label>Potassium</label>
<input type="float" class="form-control" name="pot" placeholder="pot" required="required" />
</div>
<div class="form-group col-auto">
<label>Hemoglobin</label>
<input type="float" class="form-control" name="hemo" placeholder="hemo" required="required" />
</div>
<div class="form-group col-auto">
<label>Packed Cell Volume</label>
<input type="number" class="form-control" name="pcv" placeholder="pcv" required="required" />
</div>
<div class="form-group col-auto">
<label>White Blood Cell Count</label>
<input type="number" class="form-control" name="wc" placeholder="wc" required="required" />
```

```
        </div>
        <div class="form-group col-auto">
          <label>Red Blood Cell Count</label>
          <input type="float" class="form-control" name="rc" placeholder="rc" required="required" />
        </div>
      </div>
      <div class="form-row">
            <div class="form-group col-auto">
        <label>Hypertension</label>
        <select class="form-control" name="htn" required="required">
            <option value="">Select HTN </option>
            <option value=1>Yes</option>
            <option value=0>No</option>
          </select>
      </div>
      <div class="form-group col-auto">
        <label>Diabetes Mellitus</label>
        <select class="form-control" name="dm" required="required">
            <option value="">Select DM </option>
            <option value=1>Yes</option>
            <option value=0>No</option>
          </select>
      </div>
      <div class="form-group col-auto">
        <label>Coronary Artery Disease</label>
            <select class="form-control" name="cad" required="required">
            <option value="">Select CAD </option>
            <option value=1>Yes</option>
            <option value=0>No</option>
          </select>
```

```
        </div>
        <div class="form-group col-auto">
           <label>Appetite</label>
                <select class="form-control" name="appet" required="required">
                  <option value="">Select Appet </option>
                  <option value=0>Good</option>
                  <option value=1>Poor</option>
                </select>
        </div>
        <div class="form-group col-auto">
           <label>Pedal Edema</label>
                <select class="form-control" name="pe" required="required">
                  <option value="">Select PE </option>
                  <option value=1>Yes</option>
                  <option value=0>No</option>
                </select>
        </div>
        <div class="form-group col-auto">
           <label>Anemia</label>
                <select class="form-control" name="ane" required="required">
                  <option value="">Select ANE </option>
                  <option value=1>Yes</option>
                  <option value=0>No</option>
                </select>
        </div>
        </div>
    <button type="submit" class="btn btn-primary btn-block btn-large submitButton">Predict</button>


  </form>
```

```
  <br>
  <br>
  </div>
  <h1 class="jumbotron predict-text" style="text-align: center;">{{ prediction_text }}<h1>
   <!-- Optional JavaScript -->
   <!-- jQuery first, then Popper.js, then Bootstrap JS -->
   <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj" crossorigin="anonymous"></script>
   <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity="sha384-9/reFTGAW83EW2RDu2S0VKaIzap3H66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN" crossorigin="anonymous"></script>
   <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js" integrity="sha384-B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57BaghqFfPlYxofvL8/KUEfYiJOMMV+rV" crossorigin="anonymous"></script>

  </body>
</html>
```

app.py file

```
        # -*- coding: utf-8 -*-
import numpy as np
from flask import Flask, request, render_template
from joblib import load
app = Flask(__name__)
model = load("modeldtc.save")

@app.route('/')
def home():
    return render_template('homepage.html')
```

```
@app.route('/index')
def index():
    return render_template('index.html')


@app.route('/y_predict',methods=['POST'])
def y_predict():
    '''
    For rendering results on HTML GUI
    '''
    x_test = [[x for x in request.form.values()]]
    print(x_test)
    x_test = np.array(x_test, dtype='float64')
    print(x_test)
    prediction = model.predict(x_test)
    print(prediction)
    if prediction>0.5:
        output="Negative"
    else:
        output="Positive"



    return render_template('index.html', prediction_text='Chronic Kidney Disease : {}'.format(output))


if __name__ == "__main__":
    app.run(debug=True)
```