

EARLY PREDICTION FOR CHRONIC KIDNEY DISEASE DETECTION: A PROGRESSIVE APPROACH TO HEALTH MANAGEMENT

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

Bachelor of Computer Science

In the faculty of Computer Science of Bharathiar University, Coimbatore

Submitted by

TEAM ID: NM2023TMID17963

K.BAVADHARANI

S.MUTHAMILSELVI

S.RUJITHA

S.THILAKAVATHI

Under the guidance of

Dr.V.PUNITHAVATHI MCA,M.Phil,DOA.,Ph.D.

Lecturer , Department of Computer Science



DEPARTMENT OF COMPUTER SCIENCE

L.R.G GOVERNMENT ARTS COLLEGE FOR WOMEN

(Affiliated To Bharathiar University)

TIRUPUR-4

APRIL-2023

LRG GOVERNMENT ARTS COLLEGE
NAAN MUDHALVAN PROJECT WORK
(AFFILIATED TO BHARATHIAR UNIVERSITY)
TIRUPUR-641602

**TITLE : EARLY PREDICTION FOR CHRONIC KIDNEY DISEASE
DETECTION: A PROGRESSIVE APPROACH TO HEALTH
MANAGEMENT**

This is to certify that this is a bonafide record of work done by the above students of III B.Sc (CS) Degree **NAAN MUDHALVAN PROJECT** during the year 2022-23

Submitted for the Naan Mudhalvan project work held on.....20

CLASS TUTOR

HEAD OF DEPARTMENT

TABLE OF CONTENTS

S.NO	TITLE DESCRIPTION	P.NO
1	INTRODUCTION	
	1.1 OVERVIEW	
	1.2 PROBLEM DEFINITION	
2	PROBLEM DEFINITION & DESIGN THINKING 2.1 EMPATHY MAP 2.2 IDEATION AND BRAINSTROMING MAP	
3	RESULT	
4	ADVANTAGES & DISADVANTAGES	
5	APPLICATIONS	
6	CONCLUSION	
7	FUTURE SCOPE	
8	APPENDICES A.SOURCE CODE	

INTRODUCTION

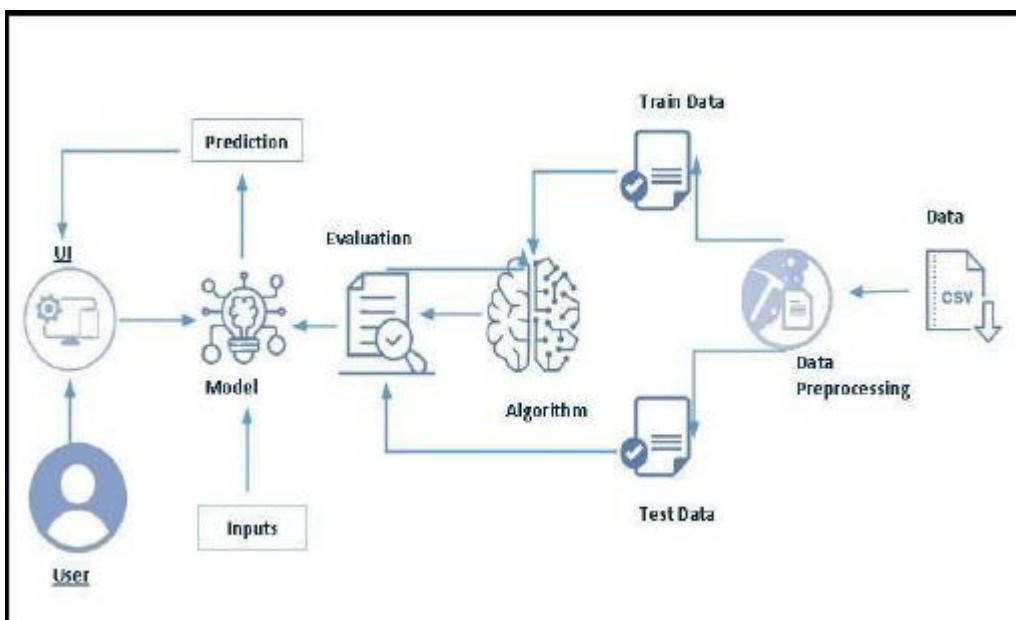
EARLY PREDICTION FOR CHRONIC KIDNEY DISEASE DETECTION: A PROGRESSIVE APPROACH TO HEALTH MANAGEMENT

1.1 OVERVIEW

Chronic Kidney Disease (CKD) is a major medical problem and can be cured if treated 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.

In today's world as we know most of the people are facing so many diseases and as this can be cured if we treat people in early stages this project can use a pretrained model to predict the chronic Kidney disease which can help in treatments of people who are suffer from this disease.

TECHNICAL ARCHITECTURE:

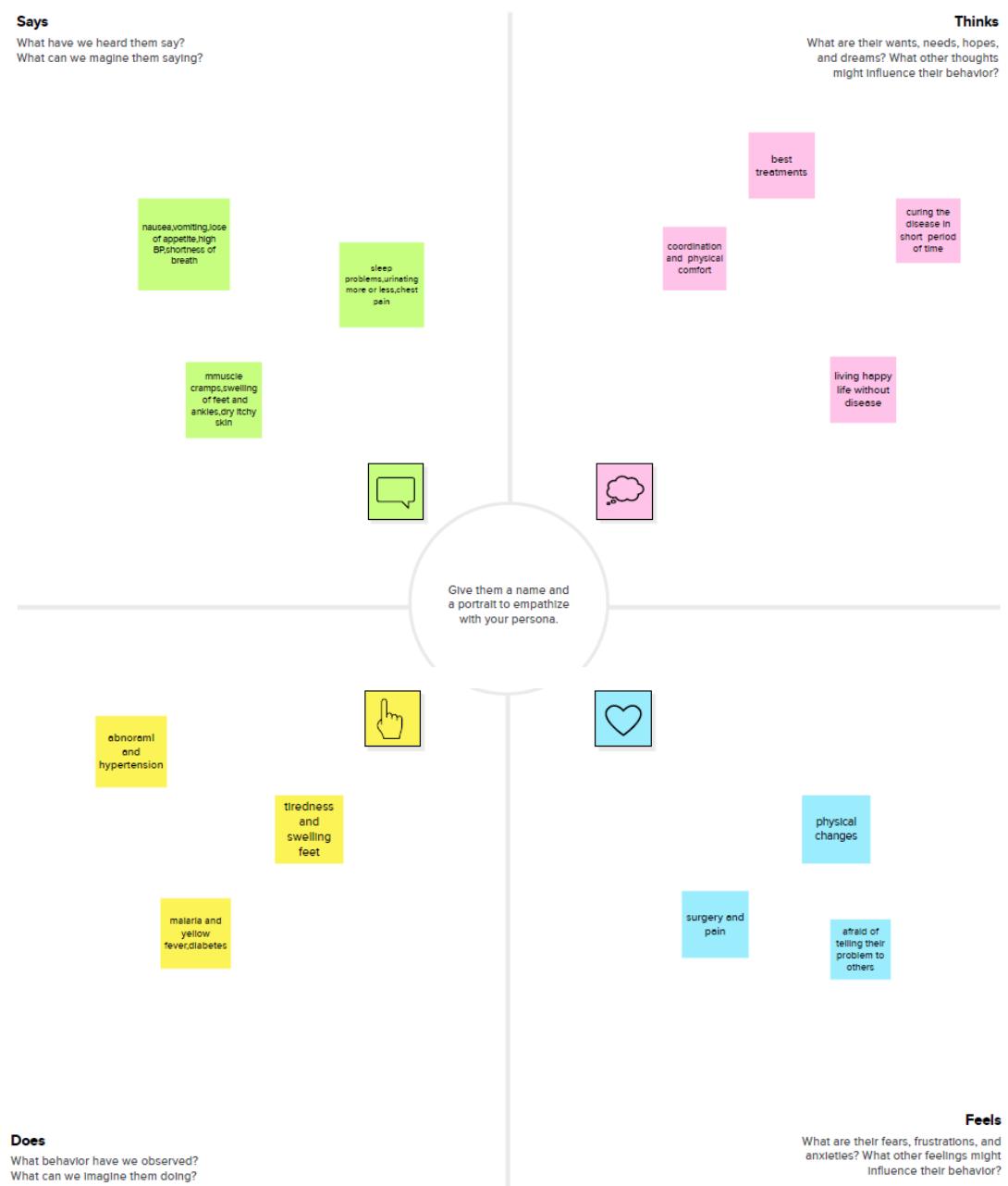


1.2 PROBLEM DEFINITION

- The goal is to build a real time application by using the machine learning techniques (Naive Bayes and KNN algorithms), to detect the CKD at an early stage.
- The proposed system finds the problem to **predicts the chance of presence of a disease that are present in a patient on the basis of their symptoms**. It also recommends necessary precautionary measures required to treat the predicted disease.
- The rationale for testing asymptomatic people for CKD is that earlier detection might **allow for the implementation of therapeutic interventions and avoidance of inappropriate exposure to nephrotoxic agents**, both of which may slow the progression of CKD to end-stage kidney disease.

2.PROBLEM DEFINITION AND DESIGN THINKING

2.1 EMPATHY MAP



2.2 IDEATION AND BRAINSTORMING MAP



Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

⌚ 10 minutes to prepare
💡 1 hour to collaborate
👥 2-8 people recommended

➡ Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

⌚ 10 minutes

A Team gathering
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B Set the goal
Think about the problem you'll be focusing on solving in the brainstorming session.

C Learn how to use the facilitation tools
Use the Facilitation Superpowers to run a happy and productive session.

[Open article](#) ➔

1 Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

⌚ 5 minutes

PROBLEM
How might we [your problem statement]? ↗



Key rules of brainstorming

To run an smooth and productive session

- ⌚ Stay in topic.
- 💡 Encourage wild ideas.
- ⌚ Difer Judgment.
- ⌚ Listen to others.
- ⌚ Go for volume.
- ⌚ If possible, be visual.

2

Brainstorm

Brainstorm

10 minutes

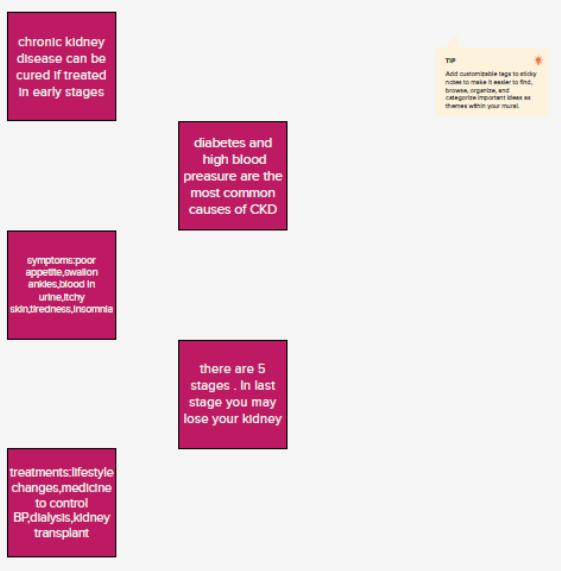


3

Group Ideas

Group Ideas
Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

20 minutes

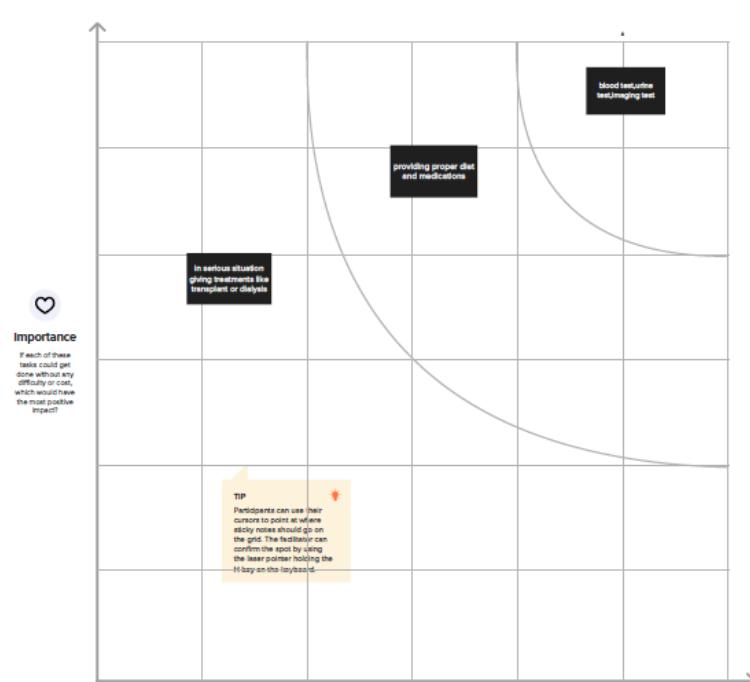


4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

⌚ 20 minutes



5

After you collaborate

You can export the mural as an image or pdf to share with members of your company who might find it helpful.

Quick add-ons

Share the mural

A Share the mural
Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.

Export the mural

Statistik und Daten



Strategy Blueprint



Customer experience journey map



Strengths, weaknesses, opportunities & threats

 Share template feedback

3. RESULT

Kidney.ipynb - kidneydisease - Visual Studio Code

EXPLORER ... model.py app.py Kidney.ipynb gif.gif

File Edit Selection View Go Run Terminal Help

Chronic-Kidney-Disea... + Code + Markdown ...

Images

- gif.gif
- gif2.gif
- kidney2.jpg
- screenshot.png

static

- no.gif
- yes2.webp

templates

- app.py
- Kidney_data.csv
- Kidney.ipynb
- kidney.jpg
- Kidney.pkl
- model.py
- Profile
- README.md
- requirements.txt

Code Cells

[1]

```
# Importing Libraries:  
import pandas as pd  
import numpy as np  
import seaborn as sns  
import matplotlib.pyplot as plt
```

[2]

```
# for displaying all feature from dataset:  
pd.pandas.set_option('display.max_columns', None)
```

[3]

```
# Reading Dataset:  
dataset = pd.read_csv("Kidney_data.csv")  
# Top 5 records:  
dataset.head()
```

[4]

	id	age	bp	sg	al	su	rbc	pc	pcc	ba	bgr	bu	sc	sod	pot	hem	pcv	wc	rc	htn	dm	cad
0	0	48.0	80.0	1.020	1.0	0.0	NaN	normal	notpresent	notpresent	121.0	36.0	1.2	NaN	NaN	15.4	44	7800	5.2	yes	yes	no
1	1	7.0	50.0	1.020	4.0	0.0	NaN	normal	notpresent	notpresent	NaN	18.0	0.8	NaN	NaN	11.3	38	6000	NaN	no	no	no
2	2	62.0	80.0	1.010	2.0	3.0	normal	normal	notpresent	notpresent	423.0	53.0	1.8	NaN	NaN	9.6	31	7500	NaN	no	yes	no
3	3	48.0	70.0	1.005	4.0	0.0	normal	abnormal	present	notpresent	117.0	56.0	3.8	111.0	2.5	11.2	32	6700	3.9	yes	no	no
4	4	51.0	80.0	1.010	2.0	0.0	normal	normal	notpresent	notpresent	106.0	26.0	1.4	NaN	NaN	11.6	35	7300	4.6	no	no	no

Cell 82 of 84

Type here to search

Windows Taskbar: File Explorer, File History, Task View, Start, Taskbar Icons, Taskbar Buttons, Taskbar Notifications, Taskbar Search, Taskbar Icons, Taskbar Buttons, Taskbar Notifications, Taskbar Search.

Kidney.ipynb - kidneydisease - Visual Studio Code

EXPLORER ... model.py app.py Kidney.ipynb gif.gif

File Edit Selection View Go Run Terminal Help

Chronic-Kidney-Disea... + Code + Markdown ...

Images

- gif.gif
- gif2.gif
- kidney2.jpg
- screenshot.png

static

- no.gif
- yes2.webp

templates

- app.py
- Kidney_data.csv
- Kidney.ipynb
- kidney.jpg
- Kidney.pkl
- model.py
- Profile
- README.md
- requirements.txt

Code Cells

[3]

	id	age	bp	sg	al	su	rbc	pc	pcc	ba	bgr	bu	sc	sod	pot	hem	pcv	wc	rc	htn	dm	cad
0	0	48.0	80.0	1.020	1.0	0.0	NaN	normal	notpresent	notpresent	121.0	36.0	1.2	NaN	NaN	15.4	44	7800	5.2	yes	yes	no
1	1	7.0	50.0	1.020	4.0	0.0	NaN	normal	notpresent	notpresent	NaN	18.0	0.8	NaN	NaN	11.3	38	6000	NaN	no	no	no
2	2	62.0	80.0	1.010	2.0	3.0	normal	normal	notpresent	notpresent	423.0	53.0	1.8	NaN	NaN	9.6	31	7500	NaN	no	yes	no
3	3	48.0	70.0	1.005	4.0	0.0	normal	abnormal	present	notpresent	117.0	56.0	3.8	111.0	2.5	11.2	32	6700	3.9	yes	no	no
4	4	51.0	80.0	1.010	2.0	0.0	normal	normal	notpresent	notpresent	106.0	26.0	1.4	NaN	NaN	11.6	35	7300	4.6	no	no	no

[4]

```
# Dropping unnecessary feature:  
dataset = dataset.drop('id', axis=1)
```

[5]

```
# Shape of dataset:  
dataset.shape
```

[6]

```
(400, 25)
```

[7]

```
# Checking Missing (NaN) Values:  
dataset.isnull().sum()
```

[8]

```
age          9
```

Cell 82 of 84

Type here to search

Windows Taskbar: File Explorer, File History, Task View, Start, Taskbar Icons, Taskbar Buttons, Taskbar Notifications, Taskbar Search, Taskbar Icons, Taskbar Buttons, Taskbar Notifications, Taskbar Search.

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb gif.gif

EXPLORER ... model.py app.py Kidney.ipynb gif.gif

KIDNEYDISEASE dney.ipynb > M+Chronic Kidney Disease > M+Heatmap > M+There is no need of Standardization and Normalization of our dataset, as we using Ensemble Technique. > # GradientBoostingClassifier: Select Kernel

Chronic-Kidney-Disea... + Code + Markdown ...

Images gif.gif gif2.gif

static kidney2.jpg screenshot.png

yes2.webp templates app.py Kidney_data.csv

Kidney.ipynb Kidney.pkl model.py Profle README.md requirements.txt

Checking missing (NaN) values:
dataset.isnull().sum()

	age	bp	sg	al	su	bgr	bu	sc	sod	pot	hemo
count	391.000000	388.000000	353.000000	354.000000	351.000000	356.000000	381.000000	383.000000	313.000000	312.000000	348.000000
mean	51.483376	76.469072	1.017408	1.016949	0.450142	148.036517	57.425722	3.072454	137.528754	4.627244	12.526437
std	17.169714	13.683637	0.005717	1.352679	1.099191	79.281714	50.503006	5.741126	10.408752	3.193904	2.912587
min	2.000000	50.000000	1.005000	0.000000	0.000000	22.000000	1.500000	0.400000	4.500000	2.500000	3.100000
25%	42.000000	70.000000	1.010000	0.000000	0.000000	99.000000	27.000000	0.900000	135.000000	3.800000	10.300000
50%	55.000000	80.000000	1.020000	0.000000	0.000000	121.000000	42.000000	1.300000	138.000000	4.400000	12.650000
75%	64.500000	80.000000	1.020000	2.000000	0.000000	163.000000	66.000000	2.800000	142.000000	4.900000	15.000000
max	90.000000	180.000000	1.025000	5.000000	5.000000	490.000000	391.000000	76.000000	163.000000	47.000000	17.800000

Description:
dataset.describe()

Datatypes:
dataset.dtypes

	age	bp	sg	al	su	bgr	bu	sc	sod	pot	hemo
count	391.000000	388.000000	353.000000	354.000000	351.000000	356.000000	381.000000	383.000000	313.000000	312.000000	348.000000
mean	51.483376	76.469072	1.017408	1.016949	0.450142	148.036517	57.425722	3.072454	137.528754	4.627244	12.526437
std	17.169714	13.683637	0.005717	1.352679	1.099191	79.281714	50.503006	5.741126	10.408752	3.193904	2.912587
min	2.000000	50.000000	1.005000	0.000000	0.000000	22.000000	1.500000	0.400000	4.500000	2.500000	3.100000
25%	42.000000	70.000000	1.010000	0.000000	0.000000	99.000000	27.000000	0.900000	135.000000	3.800000	10.300000
50%	55.000000	80.000000	1.020000	0.000000	0.000000	121.000000	42.000000	1.300000	138.000000	4.400000	12.650000
75%	64.500000	80.000000	1.020000	2.000000	0.000000	163.000000	66.000000	2.800000	142.000000	4.900000	15.000000
max	90.000000	180.000000	1.025000	5.000000	5.000000	490.000000	391.000000	76.000000	163.000000	47.000000	17.800000

Cell 82 of 84 ENG 3:25 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb gif.gif

EXPLORER ... model.py app.py Kidney.ipynb gif.gif

KIDNEYDISEASE dney.ipynb > M+Chronic Kidney Disease > M+Heatmap > M+There is no need of Standardization and Normalization of our dataset, as we using Ensemble Technique. > # GradientBoostingClassifier: Select Kernel

Chronic-Kidney-Disea... + Code + Markdown ...

Images gif.gif gif2.gif kidney2.jpg screenshot.png

static yes2.webp templates app.py Kidney_data.csv

Kidney.ipynb Kidney.pkl model.py Profle README.md requirements.txt

Description:
dataset.describe()

Datatypes:
dataset.dtypes

	age	bp	sg	al	su	bgr	bu	sc	sod	pot	hemo
count	391.000000	388.000000	353.000000	354.000000	351.000000	356.000000	381.000000	383.000000	313.000000	312.000000	348.000000
mean	51.483376	76.469072	1.017408	1.016949	0.450142	148.036517	57.425722	3.072454	137.528754	4.627244	12.526437
std	17.169714	13.683637	0.005717	1.352679	1.099191	79.281714	50.503006	5.741126	10.408752	3.193904	2.912587
min	2.000000	50.000000	1.005000	0.000000	0.000000	22.000000	1.500000	0.400000	4.500000	2.500000	3.100000
25%	42.000000	70.000000	1.010000	0.000000	0.000000	99.000000	27.000000	0.900000	135.000000	3.800000	10.300000
50%	55.000000	80.000000	1.020000	0.000000	0.000000	121.000000	42.000000	1.300000	138.000000	4.400000	12.650000
75%	64.500000	80.000000	1.020000	2.000000	0.000000	163.000000	66.000000	2.800000	142.000000	4.900000	15.000000
max	90.000000	180.000000	1.025000	5.000000	5.000000	490.000000	391.000000	76.000000	163.000000	47.000000	17.800000

Cell 82 of 84 ENG 3:26 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb gif.gif

EXPLORER KIDNEYDISEASE dney.ipynb > M+Chronic Kidney Disease > M+Heatmap > M+There is no need of Standardization and Normalization of our dataset, as we using Ensemble Technique. > # GradientBoostingClassifier: Select Kernel

Images gif.gif gif2.gif kidney2.jpg screenshot.png static no.gif yes2.webp templates app.py Kidney_data.csv Kidney.ipynb Kidney.pkl model.py Profle README.md requirements.txt

```
[8]
... age float64
bp float64
sg float64
al float64
su float64
rbc object
pc object
pcc object
ba object
bgr float64
bu float64
sc float64
sod float64
pot float64
hemo float64
pcv object
wc object
rc object
htn object
dm object
cad object
appet object
pe object
ane object
classification object
dtype: object
```

Cell 82 of 84 ENG 3:26 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb gif.gif

EXPLORER KIDNEYDISEASE dney.ipynb > M+Chronic Kidney Disease > M+Heatmap > M+There is no need of Standardization and Normalization of our dataset, as we using Ensemble Technique. > # GradientBoostingClassifier: Select Kernel

Images gif.gif gif2.gif kidney2.jpg screenshot.png static no.gif yes2.webp templates app.py Kidney_data.csv Kidney.ipynb Kidney.pkl model.py Profle README.md requirements.txt

```
[9]
... dtype: object
```

```
[9]
dataset.head()
... age bp sg al su rbc pc pcc ba bgr bu sc sod pot hemo pcv wc rc htn dm cad appet
0 48.0 80.0 1.020 1.0 0.0 NaN normal notpresent notpresent 121.0 36.0 1.2 NaN NaN 15.4 44 7800 5.2 yes yes no goo
1 7.0 50.0 1.020 4.0 0.0 NaN normal notpresent notpresent NaN 18.0 0.8 NaN NaN 11.3 38 6000 NaN no no no goo
2 62.0 80.0 1.010 2.0 3.0 normal normal notpresent notpresent 423.0 53.0 1.8 NaN NaN 9.6 31 7500 NaN no yes no poc
3 48.0 70.0 1.005 4.0 0.0 normal abnormal present notpresent 117.0 56.0 3.8 111.0 2.5 11.2 32 6700 3.9 yes no no poc
4 51.0 80.0 1.010 2.0 0.0 normal normal notpresent notpresent 106.0 26.0 1.4 NaN NaN 11.6 35 7300 4.6 no no no goo
```

Replacing Categorical values with numbers:

1. rbc

```
[10]
dataset['rbc'].value_counts()
... normal 201
abnormal 47
Name: rbc, dtype: int64
```

Cell 82 of 84 ENG 3:26 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb gif.gif

REPL Python

EXPLORER KIDNEYDISEASE dney.ipynb > Chronic Kidney Disease > Heatmap > There is no need of Standardization and Normalization of our dataset, as we using Ensemble Technique. > # GradientBoostingClassifier: Select Kernel

Images gif.gif gif2.gif kidney2.jpg screenshot.png static no.gif yes2.webp templates app.py Kidney_data.csv Kidney.ipynb Kidney.pkl model.py Profle README.md requirements.txt

Replacing Categorical values with numbers:

1. rbc

```
[10] dataset['rbc'].value_counts()
... normal 201
... abnormal 47
Name: rbc, dtype: int64
```

```
[11] dataset['rbc'] = dataset['rbc'].replace(to_replace = {'normal' : 0, 'abnormal' : 1})
```

2. pc

```
[12] dataset['pc'].value_counts()
... normal 259
... abnormal 76
Name: pc, dtype: int64
```

Cell 82 of 84 ENG 3:26 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb gif.gif

REPL Python

EXPLORER KIDNEYDISEASE dney.ipynb > Chronic Kidney Disease > Heatmap > There is no need of Standardization and Normalization of our dataset, as we using Ensemble Technique. > # GradientBoostingClassifier: Select Kernel

Images gif.gif gif2.gif kidney2.jpg screenshot.png static no.gif yes2.webp templates app.py Kidney_data.csv Kidney.ipynb Kidney.pkl model.py Profle README.md requirements.txt

```
[13] dataset['pc'] = dataset['pc'].replace(to_replace = {'normal' : 0, 'abnormal' : 1})
```

3. pcc

```
[14] dataset['pcc'].value_counts()
... notpresent 354
... present 42
Name: pcc, dtype: int64
```

```
[15] dataset['pcc'] = dataset['pcc'].replace(to_replace = {'notpresent':0, 'present':1})
```

4. ba

```
[16] dataset['ba'].value_counts()
... notpresent 374
... present 22
Name: ba, dtype: int64
```

Cell 82 of 84 ENG 3:27 PM IN 4/15/2023

The screenshot shows a Visual Studio Code interface with a Jupyter Notebook open. The notebook has two cells:

- Cell 5 (htn):** Prints the value counts for the 'htn' column.

htn	Count
no	251
yes	147

Name: htn, dtype: int64
- Cell 6 (dm):** Prints the value counts for the 'dm' column.

dm	Count
no	258
yes	134
\tno	3
\tyes	2
yes	1

Name: dm, dtype: int64

The left sidebar shows a file tree with files like model.py, app.py, gif.gif, gif2.gif, kidney2.jpg, screenshot.png, no.gif, yes2.webp, static, and Kidney_data.csv. The bottom status bar shows the cell index (Cell 82 of 84), language (Python), and system status (ENG IN 4/15/2023).

The screenshot shows a Visual Studio Code interface with a Jupyter Notebook open. The notebook contains Python code for data manipulation and visualization related to kidney disease.

Code Snippets:

- [21] dataset['dm'] = dataset['dm'].replace(to_replace = {'yes' : 1, 'no' : 0})
- [22] dataset['cad'].value_counts()

cad	Count
no	362
yes	34
\tno	2

Name: cad, dtype: int64
- [23] dataset['cad'] = dataset['cad'].replace(to_replace = {'\tno':'no'})
- [24] dataset['cad'] = dataset['cad'].replace(to_replace = {'yes' : 1, 'no' : 0})

Explorer: Shows files like model.py, app.py, gif.gif, gif2.gif, kidney.jpg, screenshot.png, static, no.gif, yes2.webp, templates, app.py, Kidney.data.csv, and Kidney.ipynb.

Status Bar: Cell 82 of 84, ENG IN 3:27 PM 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb gif.gif

EXPLORER KIDNEYDISEASE dney.ipynb > Chronic Kidney Disease > Heatmap > There is no need of Standardization and Normalization of our dataset, as we using Ensemble Technique. > # GradientBoostingClassifier: Select Kernel

Images gif.gif gif2.gif kidney2.jpg screenshot.png static no.gif yes2.webp templates app.py Kidney_data.csv Kidney.ipynb Kidney.pkl model.py Profle README.md requirements.txt

dataset['cad'] = dataset['cad'].replace(to_replace = {'yes' : 1, 'no' : 0})

8. appet

dataset['appet'].unique()

[26] Python

... array(['good', 'poor', nan], dtype=object)

[27] Python

dataset['appet'] = dataset['appet'].replace(to_replace={ 'good':1, 'poor':0, 'no':np.nan})

9. pe

dataset['pe'].value_counts()

[28] Python

... no 323
yes 76
Name: pe, dtype: int64

Cell 82 of 84 ENG 3:28 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb gif.gif

EXPLORER KIDNEYDISEASE dney.ipynb > Chronic Kidney Disease > Heatmap > There is no need of Standardization and Normalization of our dataset, as we using Ensemble Technique. > # GradientBoostingClassifier: Select Kernel

Images gif.gif gif2.gif kidney2.jpg screenshot.png static no.gif yes2.webp templates app.py Kidney_data.csv Kidney.ipynb Kidney.pkl model.py Profle README.md requirements.txt

dataset['ane'].value_counts()

[30] Python

... no 339
yes 60
Name: ane, dtype: int64

[31] Python

dataset['ane'] = dataset['ane'].replace(to_replace = {'yes' : 1, 'no' : 0})

10. ane

11. classification

dataset['classification'].value_counts()

[32] Python

... ckd 248
notckd 150
ckd\+t 2
Name: classification, dtype: int64

[33] Python

dataset['classification'] = dataset['classification'].replace(to_replace={'ckd\+t':'ckd'})

Cell 82 of 84 ENG 3:28 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb gif.gif

EXPLORER KIDNEYDISEASE dney.ipynb > Chronic Kidney Disease > Heatmap > There is no need of Standardization and Normalization of our dataset, as we using Ensemble Technique. > # GradientBoostingClassifier: Select Kernel Python

Images gif.gif gif2.gif kidney2.jpg screenshot.png static no.gif yes2.webp templates app.py Kidney_data.csv Kidney.ipynb Kidney.pkl model.py Profle README.md requirements.txt

[34] dataset.head()

Python

	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	48.0	80.0	1.020	1.0	0.0	NaN	0.0	0.0	0.0	121.0	36.0	1.2	NaN	NaN	15.4	44	7800	5.2	1.0	1.0	0.0	1.0	0.0	0.0	
1	7.0	50.0	1.020	4.0	0.0	NaN	0.0	0.0	0.0	NaN	18.0	0.8	NaN	NaN	11.3	38	6000	NaN	0.0	0.0	0.0	1.0	0.0	0.0	
2	62.0	80.0	1.010	2.0	3.0	0.0	0.0	0.0	0.0	423.0	53.0	1.8	NaN	NaN	9.6	31	7500	NaN	0.0	1.0	0.0	0.0	0.0	1.0	
3	48.0	70.0	1.005	4.0	0.0	0.0	1.0	1.0	0.0	117.0	56.0	3.8	111.0	2.5	11.2	32	6700	3.9	1.0	0.0	0.0	0.0	1.0	1.0	
4	51.0	80.0	1.010	2.0	0.0	0.0	0.0	0.0	0.0	106.0	26.0	1.4	NaN	NaN	11.6	35	7300	4.6	0.0	0.0	0.0	1.0	0.0	0.0	

[35] # Datatypes: dataset.dtypes

Python

	age	bp	sg	al	su	rbc	pc	pcc	ba
0	float64								
1	float64								
2	float64								
3	float64								
4	float64								

[36] Cell 82 of 84

Type here to search ENG 3:28 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb gif.gif

EXPLORER KIDNEYDISEASE dney.ipynb > Chronic Kidney Disease > Heatmap > There is no need of Standardization and Normalization of our dataset, as we using Ensemble Technique. > # GradientBoostingClassifier: Select Kernel Python

Images gif.gif gif2.gif kidney2.jpg screenshot.png static no.gif yes2.webp templates app.py Kidney_data.csv Kidney.ipynb Kidney.pkl model.py Profle README.md requirements.txt

[34] dataset.head()

Python

	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	48.0	80.0	1.020	1.0	0.0	NaN	0.0	0.0	0.0	121.0	36.0	1.2	NaN	NaN	15.4	44	7800	5.2	1.0	1.0	0.0	1.0	0.0	0.0	
1	7.0	50.0	1.020	4.0	0.0	NaN	0.0	0.0	0.0	NaN	18.0	0.8	NaN	NaN	11.3	38	6000	NaN	0.0	0.0	0.0	1.0	0.0	0.0	
2	62.0	80.0	1.010	2.0	3.0	0.0	0.0	0.0	0.0	423.0	53.0	1.8	NaN	NaN	9.6	31	7500	NaN	0.0	1.0	0.0	0.0	0.0	1.0	
3	48.0	70.0	1.005	4.0	0.0	0.0	1.0	1.0	0.0	117.0	56.0	3.8	111.0	2.5	11.2	32	6700	3.9	1.0	0.0	0.0	0.0	1.0	1.0	
4	51.0	80.0	1.010	2.0	0.0	0.0	0.0	0.0	0.0	106.0	26.0	1.4	NaN	NaN	11.6	35	7300	4.6	0.0	0.0	0.0	1.0	0.0	0.0	

[35] # Datatypes: dataset.dtypes

Python

	age	bp	sg	al	su	rbc	pc	pcc	ba
0	float64								
1	float64								
2	float64								
3	float64								
4	float64								

[36] Cell 82 of 84

Type here to search ENG 3:28 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb gif.gif

EXPLORER ... model.py app.py Kidney.ipynb gif.gif

KIDNEYDISEASE dney.ipynb > Chronic Kidney Disease > Heatmap > There is no need of Standardization and Normalization of our dataset, as we using Ensemble Technique. > # GradientBoostingClassifier: Select Kernel

Images gif.gif gif2.gif kidney2.jpg screenshot.png static no.gif yes2.webp templates app.py Kidney_data.csv Kidney.ipynb Kidney.jpg Kidney.pkl model.py Profile README.md requirements.txt

```
# Datatypes:  
dataset.dtypes
```

[38]

	age	bp	sg	al	su	rbc	pc	pcc	ba	bgr	bu	s
count	391.000000	388.000000	353.000000	354.000000	351.000000	248.000000	335.000000	396.000000	396.000000	356.000000	381.000000	383.000000
mean	51.483376	76.469072	1.017408	0.16949	0.450142	0.189516	0.226866	0.106061	0.055556	148.036517	57.425722	3.07245
std	17.169714	13.683637	0.005717	1.352679	1.099191	0.392711	0.419431	0.308305	0.229351	79.281714	50.503006	5.74112
min	2.000000	50.000000	1.005000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	22.000000	1.500000	0.40000
25%	42.000000	70.000000	1.010000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	99.000000	27.000000	0.90000
50%	55.000000	80.000000	1.020000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	121.000000	42.000000	1.30000
75%	64.500000	80.000000	1.020000	2.000000	0.000000	0.000000	0.000000	0.000000	0.000000	163.000000	66.000000	2.80000
max	90.000000	180.000000	1.025000	5.000000	5.000000	1.000000	1.000000	1.000000	1.000000	490.000000	391.000000	76.00000

Python

[38]

```
# Checking Missing (NaN) Values:  
dataset.isnull().sum().sort_values(ascending=False)
```

[40]

	rbc	rc	wc	pot	sod	pcv	pc	hemo	an
...	152	131	106	88	87	71	65	52	41

Python

[40]

```
Cell 82 of 84
```

Type here to search

Windows Taskbar: File Explorer, Edge, Google Chrome, Microsoft Edge, File, Task View, Start, Taskbar icons.

System tray: ENG 3:29 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb gif.gif

EXPLORER ... model.py app.py Kidney.ipynb gif.gif

KIDNEYDISEASE dney.ipynb > Chronic Kidney Disease > Heatmap > There is no need of Standardization and Normalization of our dataset, as we using Ensemble Technique. > # GradientBoostingClassifier: Select Kernel

Images gif.gif gif2.gif kidney2.jpg screenshot.png static no.gif yes2.webp templates app.py Kidney_data.csv Kidney.ipynb Kidney.jpg Kidney.pkl model.py Profile README.md requirements.txt

```
dataset.describe()
```

[39]

	age	bp	sg	al	su	rbc	pc	pcc	ba	bgr	bu	s
count	391.000000	388.000000	353.000000	354.000000	351.000000	248.000000	335.000000	396.000000	396.000000	356.000000	381.000000	383.000000
mean	51.483376	76.469072	1.017408	0.16949	0.450142	0.189516	0.226866	0.106061	0.055556	148.036517	57.425722	3.07245
std	17.169714	13.683637	0.005717	1.352679	1.099191	0.392711	0.419431	0.308305	0.229351	79.281714	50.503006	5.74112
min	2.000000	50.000000	1.005000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	22.000000	1.500000	0.40000
25%	42.000000	70.000000	1.010000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	99.000000	27.000000	0.90000
50%	55.000000	80.000000	1.020000	2.000000	0.000000	0.000000	0.000000	0.000000	0.000000	121.000000	42.000000	1.30000
75%	64.500000	80.000000	1.020000	2.000000	0.000000	0.000000	0.000000	0.000000	0.000000	163.000000	66.000000	2.80000
max	90.000000	180.000000	1.025000	5.000000	5.000000	1.000000	1.000000	1.000000	1.000000	490.000000	391.000000	76.00000

Python

[39]

```
# Checking Missing (NaN) Values:  
dataset.isnull().sum().sort_values(ascending=False)
```

[40]

	rbc	rc	wc	pot	sod	pcv	pc	hemo	an
...	152	131	106	88	87	71	65	52	41

Python

[40]

```
Cell 82 of 84
```

Type here to search

Windows Taskbar: File Explorer, Edge, Google Chrome, Microsoft Edge, File, Task View, Start, Taskbar icons.

System tray: ENG 3:29 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb gif.gif

EXPLORER ... model.py app.py Kidney.ipynb gif.gif

KIDNEYDISEASE dney.ipynb > Chronic Kidney Disease > Heatmap > There is no need of Standardization and Normalization of our dataset, as we using Ensemble Technique. > # GradientBoostingClassifier: Select Kernel

Images gif.gif gif2.gif kidney2.jpg screenshot.png

static rbc 152
rc 131
wc 106
pot 88
sod 87
pcv 71
pc 65
hemo 52
su 49
sg 47
al 46
bgr 44
bu 19
sc 17
bp 12
age 9
ba 4
pcc 4
htn 2
dm 2
cad 2
ane 1
appet 1
pe 1
classification 0
dtvne: int64

OUTLINE
TIMELINE

Type here to search Cell 82 of 84 ENG 3:29 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb gif.gif

EXPLORER ... model.py app.py Kidney.ipynb gif.gif

KIDNEYDISEASE dney.ipynb > Chronic Kidney Disease > Heatmap > There is no need of Standardization and Normalization of our dataset, as we using Ensemble Technique. > # GradientBoostingClassifier: Select Kernel

Images gif.gif gif2.gif kidney2.jpg screenshot.png

static dataset.columns
... Index(['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'],
... dtype='object')

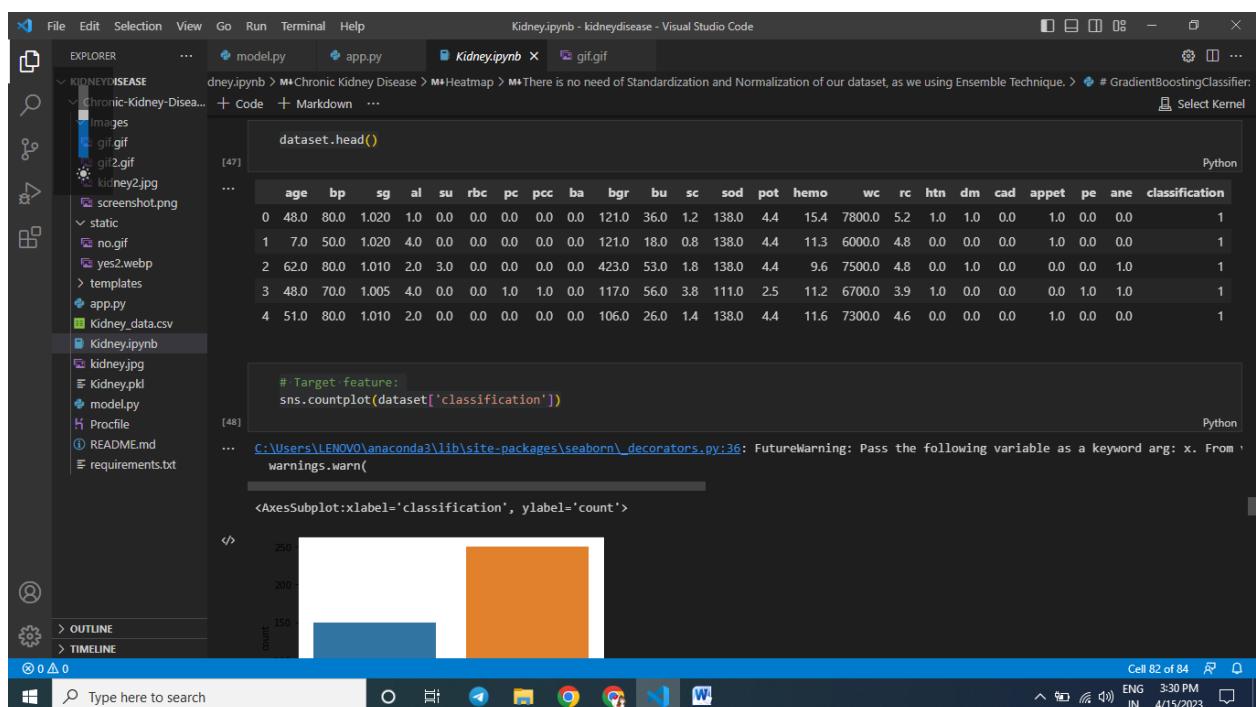
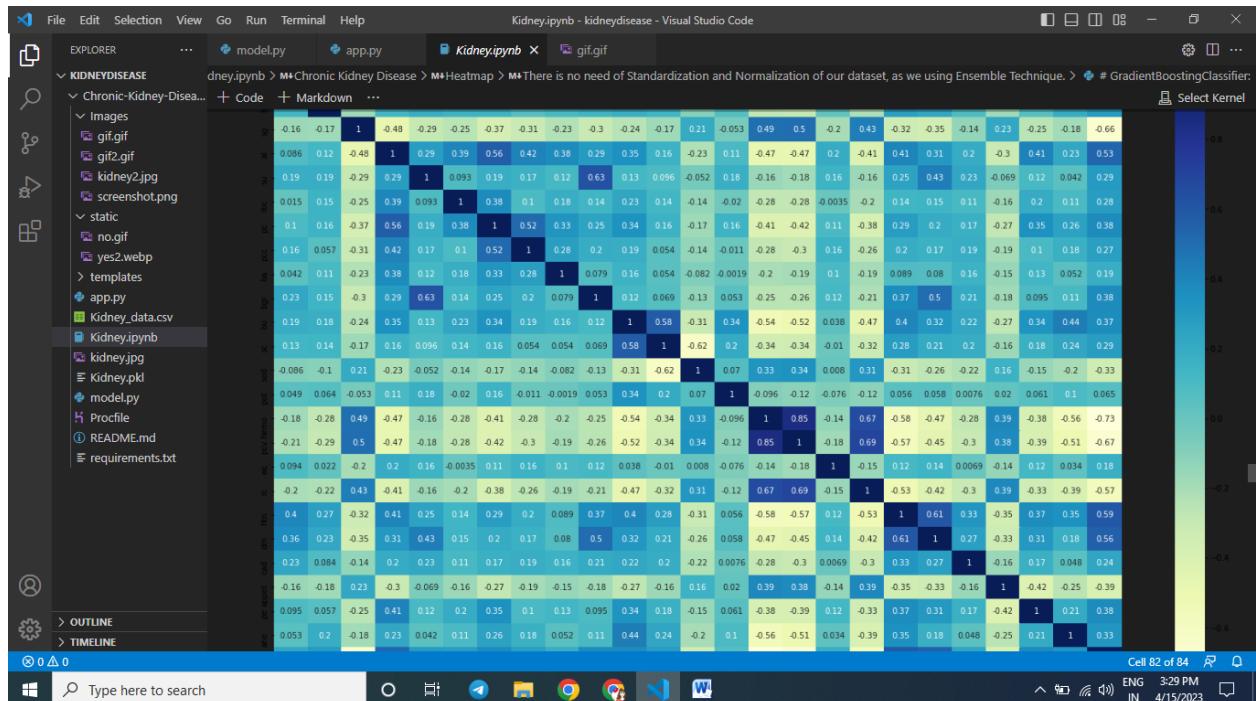
dataset.columns
... Index(['age', 'bp', 'sg', 'al', 'su', 'rbc', 'pc', 'pcc', 'ba', 'bgr', 'bu',
... 'sc', 'sod', 'pot', 'hemo', 'pcv', 'wc', 'rc', 'htn', 'dm', 'cad',
... 'appet', 'pe', 'ane'],
... dtype='object')

for feature in features:
... dataset[feature] = dataset[feature].fillna(dataset[feature].median())

dataset.isnull().any().sum()
... 0

Heatmap

Type here to search Cell 82 of 84 ENG 3:29 PM IN 4/15/2023



File Edit Selection View Go Run Terminal Help Kidney.ipynb - kidneydisease - Visual Studio Code

EXPLORER model.py app.py Kidney.ipynb gif.gif

KIDNEYDISEASE dney.ipynb > Chronic-Kidney-Disea... + Code + Markdown ... C:\Users\LENOVO\anaconda3\lib\site-packages\seaborn_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From 'warnings.warn(

Images gif.gif gif2.gif kidney2.jpg screenshot.png static no.gif yes2.webp yes2.webp templates app.py Kidney_data.csv Kidney.ipynb Kidney.jpg Kidney.pkl model.py Profile README.md requirements.txt

<AxesSubplot:xlabel='classification', ylabel='count'>

```
# Independent and Dependent Feature:
X = dataset.iloc[:, :-1]
y = dataset.iloc[:, -1]
```

[49] Python

OUTLINE [50] X.head()

Timeline Python

Type here to search Cell 82 of 84 ENG 3:30 PM IN 4/15/2023

File Edit Selection View Go Run Terminal Help Kidney.ipynb - kidneydisease - Visual Studio Code

EXPLORER model.py app.py Kidney.ipynb gif.gif

KIDNEYDISEASE dney.ipynb > Chronic-Kidney-Disea... + Code + Markdown ... C:\Users\LENOVO\anaconda3\lib\site-packages\seaborn_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From 'warnings.warn(

Images gif.gif gif2.gif kidney2.jpg screenshot.png static no.gif yes2.webp yes2.webp templates app.py Kidney_data.csv Kidney.ipynb Kidney.jpg Kidney.pkl model.py Profile README.md requirements.txt

X.head()

	age	bp	sg	al	su	rbc	pc	pcc	ba	bgr	bu	sc	sod	pot	hemo	wc	rc	htn	dm	cad	appet	pe	ane
0	48.0	80.0	1.020	1.0	0.0	0.0	0.0	0.0	0.0	121.0	36.0	1.2	138.0	4.4	15.4	7800.0	5.2	1.0	1.0	0.0	1.0	0.0	0.0
1	7.0	50.0	1.020	4.0	0.0	0.0	0.0	0.0	0.0	121.0	18.0	0.8	138.0	4.4	11.3	6000.0	4.8	0.0	0.0	0.0	1.0	0.0	0.0
2	62.0	80.0	1.010	2.0	3.0	0.0	0.0	0.0	0.0	423.0	53.0	1.8	138.0	4.4	9.6	7500.0	4.8	0.0	1.0	0.0	0.0	0.0	1.0
3	48.0	70.0	1.005	4.0	0.0	0.0	1.0	1.0	0.0	117.0	56.0	3.8	111.0	2.5	11.2	6700.0	3.9	1.0	0.0	0.0	0.0	1.0	1.0
4	51.0	80.0	1.010	2.0	0.0	0.0	0.0	0.0	0.0	106.0	26.0	1.4	138.0	4.4	11.6	7300.0	4.6	0.0	0.0	0.0	1.0	0.0	0.0

```
# Feature Importance:
from sklearn.ensemble import ExtraTreesClassifier
import matplotlib.pyplot as plt
model=ExtraTreesClassifier()
model.fit(X,y)

plt.figure(figsize=(8,6))
ranked_features=pd.Series(model.feature_importances_,index=X.columns)
ranked_features.nlargest(24).plot(kind='barh')
plt.show()
```

[51] Python

OUTLINE [52] Type here to search Cell 82 of 84 ENG 3:30 PM IN 4/15/2023

File Edit Selection View Go Run Terminal Help Kidney.ipynb - kidneydisease - Visual Studio Code

EXPLORER model.py app.py Kidney.ipynb gif.gif

KIDNEYDISEASE dney.ipynb > M+Chronic Kidney Disease > M+Heatmap > M+There is no need of Standardization and Normalization of our dataset, as we are using Ensemble Technique. > # GradientBoostingClassifier: Select Kernel

Chronic-Kidney-Dise... + Code + Markdown ...

Images

gif.gif gif2.gif kidney2.jpg screenshot.png

static

no.gif yes2.webp

templates

app.py Kidney_data.csv

Kidney.ipynb

kidney.jpg

Kidney.pkl

model.py

Profile

README.md

requirements.txt

```
plt.figure(figsize=(8,6))
ranked_features=pd.Series(model.feature_importances_,index=X.columns)
ranked_features.nlargest(24).plot(kind='barh')
plt.show()
```

[51]

Python

Feature	Importance
ba	0.005
wc	0.005
cad	0.005
pot	0.005
pcc	0.005
age	0.005
su	0.005
sgot	0.005
tnp	0.005
bu	0.005
rbc	0.005
ane	0.005
bgr	0.005
sc	0.005
pe	0.005
rc	0.005
pt	0.005
appet	0.005
al	0.005
dm	0.005
hemoglobin	0.005
htin	0.115
sg	0.005

• We take top 8 feature only.

The screenshot shows a Visual Studio Code interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Title Bar:** Kidney.ipynb - kidneydisease - Visual Studio Code.
- Explorer:** Shows the file structure:
 - KIDNEYDISEASE
 - Chronic-Kidney-Diseas...
 - Images
 - static
 - templates
 - app.py
 - Kidney_data.csv
 - Kidney.ipynb
- Code Editor:** Displays three code cells in Python.
 - `X.head()`: Prints the first 53 rows of the dataset. The columns are sg, htn, hemo, dm, al, appet, rc, pc. The data shows values ranging from 0.0 to 15.8.
 - `X.tail()`: Prints the last 62 rows of the dataset. The columns are sg, htn, hemo, dm, al, appet, rc, pc. The data shows values ranging from 0.0 to 15.8.
 - `y.head()`: Prints the first 54 rows of the target variable y. The columns are sg, htn, hemo, dm, al, appet, rc, pc. The data shows values ranging from 0.0 to 15.8.
- Status Bar:** Cell 82 of 84, ENG IN 3:31 PM 4/15/2023.

Screenshot of Visual Studio Code showing a Jupyter notebook cell for a Random Forest Classifier. The code imports RandomForestClassifier from sklearn.ensemble, fits it to the dataset, and prints the accuracy and classification report.

```
# RandomForestClassifier:  
from sklearn.ensemble import RandomForestClassifier  
RandomForest = RandomForestClassifier()  
RandomForest = RandomForest.fit(X_train,y_train)  
  
# Predictions:  
y_pred = RandomForest.predict(X_test)  
  
# Performance:  
print('Accuracy:', accuracy_score(y_test,y_pred))  
print(confusion_matrix(y_test,y_pred))  
print(classification_report(y_test,y_pred))
```

The output shows an accuracy of 0.975 and a classification report:

	precision	recall	f1-score	support
0	1.00	0.95	0.97	58
1	0.95	1.00	0.98	62
accuracy			0.97	120
macro avg	0.98	0.97	0.97	120
weighted avg	0.98	0.97	0.97	120

Screenshot of Visual Studio Code showing a Jupyter notebook cell for an AdaBoost Classifier. The code imports AdaBoostClassifier from sklearn.ensemble, fits it to the dataset, and prints the accuracy and classification report.

```
print(confusion_matrix(y_test,y_pred))  
print(classification_report(y_test,y_pred))
```

The output shows an accuracy of 0.975 and a classification report:

	precision	recall	f1-score	support
0	1.00	0.95	0.97	58
1	0.95	1.00	0.98	62
accuracy			0.97	120
macro avg	0.98	0.97	0.97	120
weighted avg	0.98	0.97	0.97	120

Below this cell, another cell is partially visible, containing code for GradientBoostingClassifier.

```
# GradientBoostingClassifier:  
from sklearn.ensemble import GradientBoostingClassifier  
GradientBoost = GradientBoostingClassifier()  
GradientBoost = GradientBoost.fit(X_train,y_train)  
  
# Predictions:  
y_pred = GradientBoost.predict(X_test)  
  
# Performance:  
print('Accuracy:', accuracy_score(y_test,y_pred))  
print(confusion_matrix(y_test,y_pred))  
print(classification_report(y_test,y_pred))
```

The screenshot shows a Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Title Bar:** Kidney.ipynb - kidneydisease - Visual Studio Code.
- Explorer Panel:** Shows a file tree with a folder named "KIDNEYDISEASE". Inside "KIDNEYDISEASE" are subfolders "Chronic-Kidney-Disea...", "Images", "static", and files "app.py", "Kidney.ipynb", "Kidney.pkl", "model.py", "Profile", "README.md", and "requirements.txt".
- Code Editor:** A Python code cell containing the following code:

```
# GradientBoostingClassifier:  
from sklearn.ensemble import GradientBoostingClassifier  
GradientBoost = GradientBoostingClassifier()  
GradientBoost = GradientBoost.fit(X_train,y_train)  
  
# Predictions:  
y_pred = GradientBoost.predict(X_test)  
  
# Performance:  
print('Accuracy:', accuracy_score(y_test,y_pred))  
print(confusion_matrix(y_test,y_pred))  
print(classification_report(y_test,y_pred))
```
- Output Panel:** Displays the execution results:

```
... Accuracy: 0.975  
[[55  3]  
 [ 0 62]]  
 precision    recall   f1-score   support  
          0       1.00      0.95      0.97      58  
          1       0.95      1.00      0.98      62  
  
accuracy         0.98      0.97      0.97     120  
macro avg       0.98      0.97      0.97     120  
weighted avg    0.98      0.97      0.97     120
```
- Bottom Status Bar:** Shows "Cell 81 of 84", system icons (Wi-Fi, battery), and the date/time "4/15/2023".

SAMPLE OUTPUT

File Edit Selection View Go Run Terminal Help Kidney.ipynb - kidneydisease - Visual Studio Code

EXPLORER KIDNEYDISEASE ... model.py app.py Kidney.ipynb Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > Chronic Kidney Disease + Code + Markdown ... Select Kernel

Images gif.gif gif2.gif kidney2.jpg screenshot.png static no.gif yes2.webp templates index.html result.html app.py Kidney_data.csv Kidney.ipynb kidney.jpg Kidney.pkl model.py Profiler README.md requirements.txt

Importing Libraries:
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

[1]

for displaying all feature from dataset:
pd.pandas.set_option('display.max_columns', None)

[2]

Reading Dataset:
dataset = pd.read_csv("Kidney_data.csv")
Top 5 records:
dataset.head()

[3]

	id	age	bp	sg	al	su	rbc	pc	pcc	ba	bgr	bu	sc	sod	pot	hemo	pcv	wc	rc	htn
0	0	48.0	80.0	1.020	1.0	0.0	NaN	normal	notpresent	notpresent	121.0	36.0	1.2	NaN	NaN	15.4	44	7800	5.2	yes
1	1	7.0	50.0	1.020	4.0	0.0	NaN	normal	notpresent	notpresent	NaN	18.0	0.8	NaN	NaN	11.3	38	6000	NaN	no
2	2	62.0	80.0	1.010	2.0	3.0	normal	normal	notpresent	notpresent	423.0	53.0	1.8	NaN	NaN	9.6	31	7500	NaN	no
3	3	48.0	70.0	1.005	4.0	0.0	normal	abnormal	present	notpresent	117.0	56.0	3.8	111.0	2.5	11.2	32	6700	3.9	yes
4	4	51.0	80.0	1.010	2.0	0.0	normal	normal	notpresent	notpresent	106.0	26.0	1.4	NaN	NaN	11.6	35	7300	4.6	no

Cell 1 of 84 ENG 2:45 PM IN 4/15/2023

Type here to search

File Edit Selection View Go Run Terminal Help Kidney.ipynb - kidneydisease - Visual Studio Code

EXPLORER KIDNEYDISEASE ... model.py app.py Kidney.ipynb Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > Chronic Kidney Disease + Code + Markdown ... Select Kernel

Images gif.gif gif2.gif kidney2.jpg screenshot.png static no.gif yes2.webp templates index.html result.html app.py Kidney_data.csv Kidney.ipynb kidney.jpg Kidney.pkl model.py Profiler README.md requirements.txt

Dropping unnecessary feature :
dataset = dataset.drop('id', axis=1)

[4]

Shape of dataset:
dataset.shape

[5]

(400, 25)

Checking Missing (NaN) Values:
dataset.isnull().sum()

[6]

	age	9
bp	12	
sg	47	
al	46	
su	49	
rbc	152	
pc	65	
pcc	4	
ba	4	
bgr	44	
bu	19	
sc	17	

Cell 1 of 84 ENG 2:46 PM IN 4/15/2023

Type here to search

The screenshot shows a Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Title Bar:** Kidney.ipynb - kidneydisease - Visual Studio Code.
- Left Sidebar (Explorer):**
 - KIDNEYDISEASE
 - Chronic-Kidney-Disease-Prediction-Project-main
 - Images (gif.gif, gif2.gif, kidney2.jpg, screenshot.png)
 - static (no.gif, yes2.webp)
 - templates (index.html, result.html)
 - app.py
 - Kidney_data.csv
 - Kidney.ipynb (selected)
 - model.pkl
 - Procfile
 - README.md
 - requirements.txt
- Central Area:**
 - Code Editor: A Jupyter cell containing the code: `# Description:
dataset.describe()`.
 - Output Panel: Displays the descriptive statistics for the dataset columns.
- Bottom Status Bar:** Cell 1 of 84, 2:46 PM, 4/15/2023, ENG IN.

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help

EXPLORER KIDNEYDISEASE model.py app.py Kidney.ipynb

Chronic-Kidney-Disease-Prediction-P... Chronic Kidney Disease

Select Kernel Python

Datatypes:
dataset.dtypes

	age	float64
...	bp	float64
	sg	float64
	al	float64
	su	float64
	rbc	object
	pc	object
	pcc	object
	ba	object
	bgr	float64
	bu	float64
	sc	float64
	sod	float64
	pot	float64
	hemto	float64
	pcv	object
	wc	object
	rc	object
	htn	object
	dm	object
	cad	object
	appet	object
	pe	object
	ane	object
	classification	object
	dtype:	object

Cell 1 of 84

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb X

EXPLORER model.py app.py Kidney.ipynb X

Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > Chronic Kidney Disease

+ Code + Markdown ... Select Kernel

dataset.head()

Python

	age	bp	sg	al	su	rbc	pc	pcc	ba	bgr	bu	sc	sod	pot	hemo	pcv	wc	rc	htn	dm	cad	appe
0	48.0	80.0	1.020	1.0	0.0	NaN	normal	notpresent	notpresent	121.0	36.0	1.2	NaN	NaN	15.4	44	7800	5.2	yes	yes	no	good
1	7.0	50.0	1.020	4.0	0.0	NaN	normal	notpresent	notpresent	NaN	18.0	0.8	NaN	NaN	11.3	38	6000	NaN	no	no	no	good
2	62.0	80.0	1.010	2.0	3.0	normal	normal	notpresent	notpresent	423.0	53.0	1.8	NaN	NaN	9.6	31	7500	NaN	no	yes	no	poor
3	48.0	70.0	1.005	4.0	0.0	normal	abnormal	present	notpresent	117.0	56.0	3.8	111.0	2.5	11.2	32	6700	3.9	yes	no	no	poor
4	51.0	80.0	1.010	2.0	0.0	normal	normal	notpresent	notpresent	106.0	26.0	1.4	NaN	NaN	11.6	35	7300	4.6	no	no	no	good

Replacing Categorical values with numbers:

1. rbc

dataset['rbc'].value_counts()

Python

	normal	201
...	abnormal	47

Name: rbc, dtype: int64

Type here to search Cell 1 of 84

OUTLINE TIMELINE

ENG 2:47 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb X

EXPLORER model.py app.py Kidney.ipynb X

Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > Chronic Kidney Disease

+ Code + Markdown ... Select Kernel

dataset['rbc'] = dataset['rbc'].replace(to_replace = {'normal' : 0, 'abnormal' : 1})

Python

2. pc

dataset['pc'].value_counts()

Python

	normal	259
...	abnormal	76

Name: pc, dtype: int64

dataset['pc'] = dataset['pc'].replace(to_replace = {'normal' : 0, 'abnormal' : 1})

Python

3. pcc

dataset['pcc'].value_counts()

Python

	notpresent	354
...	present	42

Name: pcc, dtype: int64

Type here to search Cell 1 of 84

OUTLINE TIMELINE

ENG 2:48 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb X

EXPLORER ... model.py app.py Kidney.ipynb X

Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > Chronic Kidney Disease

+ Code + Markdown ...

[15] Select Kernel Python

4. ba

```
dataset['ba'].value_counts()
```

[16] Python

```
... notpresent 374
present 22
Name: ba, dtype: int64
```

```
dataset['ba'] = dataset['ba'].replace(to_replace = {'notpresent':0,'present':1})
```

[17] Python

5. htn

```
dataset['htn'].value_counts()
```

[18] Python

```
... no 251
yes 147
Name: htn, dtype: int64
```

Cell 1 of 84 ENG 2:48 PM IN 4/15/2023

Type here to search

OUTLINE TIMELINE

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb X

EXPLORER ... model.py app.py Kidney.ipynb X

Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > Chronic Kidney Disease

+ Code + Markdown ...

[19] Select Kernel Python

6. dm

```
dataset['dm'].value_counts()
```

[20] Python

```
... no 258
yes 134
\tno 3
\tyes 2
yes 1
Name: dm, dtype: int64
```

```
dataset['dm'] = dataset['dm'].replace(to_replace = {'\tyes':'yes', ' yes':'yes', '\tno':'no'})
```

[21] Python

```
dataset['dm'] = dataset['dm'].replace(to_replace = {'yes' : 1, 'no' : 0})
```

[22] Python

7. cad

```
dataset['cad'].value_counts()
```

[23] Python

Cell 1 of 84 ENG 2:48 PM IN 4/15/2023

Type here to search

OUTLINE TIMELINE

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help

Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > Chronic Kidney Disease

SELECT KERNEL Python

EXPLORER model.py app.py Kidney.ipynb

Chronic-Kidney-Disease-Prediction-Project-main

- Images gif.gif gif2.gif kidney2.jpg screenshot.png
- static no.gif yes2.webp
- templates index.html result.html

app.py Kidney_data.csv Kidney.ipynb

kidney.jpg Kidney.pkl model.py Profle README.md requirements.txt

8. appet

```
[24] dataset['cad'] = dataset['cad'].replace(to_replace = {'\tno': 'no'})
```

```
[25] dataset['cad'] = dataset['cad'].replace(to_replace = {'yes' : 1, 'no' : 0})
```

Python

dataset['appet'].unique()

```
[26] ... array(['good', 'poor', nan], dtype=object)
```

```
[27] dataset['appet'] = dataset['appet'].replace(to_replace={'good':1,'poor':0,'no':np.nan})
```

Python

9. pe

```
[28] dataset['pe'].value_counts()
```

```
[29] ... no 323  
... yes 76
```

Name: pe, dtype: int64

```
[29] dataset['pe'] = dataset['pe'].replace(to_replace = {'yes' : 1, 'no' : 0})
```

Python

Type here to search Cell 1 of 84 ENG 2:49 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help

Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > Chronic Kidney Disease

SELECT KERNEL Python

EXPLORER model.py app.py Kidney.ipynb

Chronic-Kidney-Disease-Prediction-Project-main

- Images gif.gif gif2.gif kidney2.jpg screenshot.png
- static no.gif yes2.webp
- templates index.html result.html

app.py Kidney_data.csv Kidney.ipynb

kidney.jpg Kidney.pkl model.py Profle README.md requirements.txt

9. pe

```
[28] dataset['pe'].value_counts()
```

```
[29] ... no 323  
... yes 76
```

Name: pe, dtype: int64

```
[29] dataset['pe'] = dataset['pe'].replace(to_replace = {'yes' : 1, 'no' : 0})
```

Python

10. ane

```
[30] dataset['ane'].value_counts()
```

```
[30] ... no 339  
... yes 60
```

Name: ane, dtype: int64

Type here to search Cell 1 of 84 ENG 2:49 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb

EXPLORER model.py app.py Kidney.ipynb

Chronic-Kidney-Disease-Project-main > Kidney.ipynb > Chronic Kidney Disease > Replacing Categorical values with numbers: > model.py > dataset['pe'].value_counts()

Select Kernel

11. classification

```
dataset['classification'].value_counts()
```

[32]

```
... ckd    248  
notckd  150  
ckd\tn     2  
Name: classification, dtype: int64
```

Python

```
dataset['classification'] = dataset['classification'].replace(to_replace={'ckd\tn': 'ckd'})
```

[33]

```
dataset["classification"] = [1 if i == "ckd" else 0 for i in dataset["classification"]]
```

[34]

```
dataset.head()
```

[35]

```
... 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   48.0    80.0    1.020   1.0    0.0      NaN    0.0    0.0      0.0    121.0    36.0    1.2      NaN    NaN    15.4    44    7800    5.2    1.0    1.0    0.0    1.0    0.0    0.0  
1   7.0     50.0    1.020   4.0    0.0      NaN    0.0    0.0      0.0      NaN    18.0    0.8      NaN    NaN    11.3    38    6000    NaN    0.0    0.0    0.0    1.0    0.0    0.0  
2   62.0    80.0    1.010   2.0    3.0      0.0    0.0    0.0      0.0    423.0    53.0    1.8      NaN    NaN    9.6     31    7500    NaN    0.0    1.0    0.0    0.0    0.0    0.0    1.0
```

Cell 40 of 84

Type here to search

ENG 2:49 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb

EXPLORER model.py app.py Kidney.ipynb

Chronic-Kidney-Disease-Project-main > Kidney.ipynb > Chronic Kidney Disease > Replacing Categorical values with numbers: > model.py > # Datatypes:

Select Kernel

```
# Datatypes:  
dataset.dtypes
```

[36]

```
... age      float64  
bp      float64  
sg      float64  
al      float64  
su      float64  
rbc      float64  
pc      float64  
pcc      float64  
ba      float64  
bgr      float64  
bu      float64  
sc      float64  
sod      float64  
pot      float64  
hemo      float64  
pcv      object  
wc      object  
rc      object  
htn      float64  
dm      float64  
cad      float64  
appet    float64  
pe      float64  
ane      float64  
classification  int64  
dtype: object
```

Cell 40 of 84

Type here to search

ENG 2:49 PM IN 4/15/2023

File Edit Selection View Go Run Terminal Help Kidney.ipynb - kidneydisease - Visual Studio Code

EXPLORER model.py app.py Kidney.ipynb

KIDNEYDISEASE Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > Chronic Kidney Disease > Replacing Categorical values with numbers: > m+pe > dataset['pe'].value_counts() Select Kernel

CONVERTING OBJECT VALUES INTO NUMERIC VALUES:

```
[37] dataset['pcv'] = pd.to_numeric(dataset['pcv'], errors='coerce')
dataset['wc'] = pd.to_numeric(dataset['wc'], errors='coerce')
dataset['rc'] = pd.to_numeric(dataset['rc'], errors='coerce')
```

[38] # Datatypes:
dataset.dtypes

```
[38] ... age float64
bp float64
sg float64
al float64
su float64
rbc float64
pc float64
pcc float64
ba float64
bgr float64
bu float64
sc float64
sod float64
pot float64
hemo float64
pcv float64
... float64
```

Cell 40 of 84 ENG 2:50 PM IN 4/15/2023

File Edit Selection View Go Run Terminal Help Kidney.ipynb - kidneydisease - Visual Studio Code

EXPLORER model.py app.py Kidney.ipynb

KIDNEYDISEASE Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > Chronic Kidney Disease > Replacing Categorical values with numbers: > m+pe > dataset['pe'].value_counts() Select Kernel

Description:
dataset.describe()

	age	bp	sg	al	su	rbc	pc	pcc	ba	bgr	bu	s
count	391.000000	388.000000	353.000000	354.000000	351.000000	248.000000	335.000000	396.000000	396.000000	356.000000	381.000000	383.00000
mean	51.483376	76.469072	1.017408	1.016949	0.450142	0.189516	0.226866	0.106061	0.055556	148.036517	57.425722	3.07245
std	17.169714	13.683637	0.005717	1.352679	1.099191	0.392711	0.419431	0.308305	0.229351	79.281714	50.503006	5.74112
min	2.000000	50.000000	1.005000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	22.000000	1.500000	0.40000
25%	42.000000	70.000000	1.010000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	99.000000	27.000000	0.90000
50%	55.000000	80.000000	1.020000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	121.000000	42.000000	1.30000
75%	64.500000	80.000000	1.020000	2.000000	0.000000	0.000000	0.000000	0.000000	0.000000	163.000000	66.000000	2.80000
max	90.000000	180.000000	1.025000	5.000000	5.000000	1.000000	1.000000	1.000000	1.000000	490.000000	391.000000	76.00000

[40] # Checking Missing (NaN) Values:
dataset.isnull().sum().sort_values(ascending=False)

```
[40] ... rbc 152
rc 131
wc 106
pot 88
sod 87
pcv 71
pc 65
```

Cell 40 of 84 ENG 2:50 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb

EXPLORER model.py app.py Kidney.ipynb

KIDNEYDISEASE Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > M+Chronic Kidney Disease > M+Replacing Categorical values with numbers: > M+pe > dataset['pe'].value_counts()

Select Kernel

```
# Checking Missing (NaN) Values:
dataset.isnull().sum().sort_values(ascending=False)
```

Python

[48]

	rbc	rc	WC	pot	sod	pcv	pc	hemo	su	sg	al	bgr	bu	sc	bp	age	ba	pcc	htn	dm	cad	ane	appet	pe	classification
0	152	131	106	88	87	71	65	52	49	47	46	44	19	17	12	9	4	4	2	2	2	1	1	0	8
1	0.14	-0.16	0.086	0.19	0.015	0.1	0.16	0.042	0.23	0.19	0.13	0.086	0.049	0.18	-0.21	0.094	0.2	0.4	0.36	0.23	-0.16	0.095	0.053	0.23	
2	0.14	-0.17	0.12	0.19	0.15	0.16	0.057	0.11	0.15	0.18	0.14	-0.1	0.064	-0.28	-0.29	0.022	-0.22	0.27	0.23	0.084	-0.18	0.057	0.2	0.29	
3	-0.16	-0.17	1	-0.48	-0.29	-0.25	-0.37	-0.31	-0.23	-0.3	-0.24	-0.17	0.21	-0.053	0.49	0.5	-0.2	0.43	-0.32	-0.35	-0.14	0.23	0.25	-0.18	-0.66
4	0.086	0.12	-0.48	1	0.29	0.39	0.56	0.42	0.38	0.29	0.35	0.16	-0.23	0.11	-0.47	-0.47	0.2	-0.41	0.41	0.31	0.2	-0.3	0.41	0.23	0.53
5	0.19	0.15	-0.29	0.29	1	0.093	0.19	0.17	0.11	0.63	0.13	0.096	-0.052	0.18	-0.16	-0.18	0.16	-0.16	0.25	0.43	0.23	-0.069	0.12	0.042	0.29
6	0.015	0.15	-0.25	0.39	0.093	1	0.38	0.1	0.18	0.14	0.23	0.14	-0.14	-0.02	-0.28	-0.28	-0.0035	0.2	0.14	0.15	0.11	-0.16	0.2	0.11	0.28
7	0.1	0.16	-0.37	0.56	0.19	0.38	1	0.52	0.33	0.25	0.34	0.16	-0.17	0.16	-0.41	-0.42	0.11	-0.38	0.29	0.2	0.17	-0.27	0.35	0.26	0.38
8	0.16	0.057	-0.31	0.42	0.17	0.1	0.52	1	0.28	0.2	0.19	0.054	-0.14	-0.011	-0.28	-0.3	0.16	-0.26	0.2	0.17	0.19	-0.19	0.1	0.18	0.27
9	0.042	0.11	-0.23	0.38	0.12	0.18	0.33	0.28	1	0.079	0.16	0.054	-0.082	-0.0019	-0.2	-0.19	0.1	-0.19	0.089	0.08	0.16	-0.15	0.13	0.052	0.19
10	0.23	0.15	0.3	0.29	0.63	0.14	0.25	0.2	0.079	1	0.12	0.069	-0.13	0.053	-0.25	-0.26	0.12	-0.21	0.37	0.5	0.21	-0.18	0.095	0.11	0.38
11	0.19	0.18	0.24	0.35	0.13	0.23	0.34	0.19	0.16	0.12	1	0.58	-0.31	0.34	-0.54	-0.52	0.038	-0.47	0.4	0.32	0.22	-0.27	0.34	0.44	0.37
12	0.13	0.14	0.17	0.16	0.096	0.14	0.16	0.054	0.054	0.069	0.58	1	-0.62	0.2	-0.34	-0.34	-0.01	0.32	0.28	0.21	0.2	-0.16	0.18	0.24	0.29
13	-0.086	-0.1	0.21	-0.23	-0.052	-0.14	-0.17	-0.14	-0.082	-0.13	-0.31	-0.62	1	0.07	0.33	0.34	0.008	0.31	-0.31	-0.26	-0.22	0.16	-0.15	-0.2	-0.33
14	0.049	0.064	-0.053	0.11	0.18	-0.02	0.16	-0.011	0.0019	0.053	0.34	0.2	0.07	1	-0.096	-0.12	0.056	0.058	0.0076	0.02	0.061	0.1	0.065		
15	-0.18	-0.28	0.49	-0.47	-0.16	-0.28	-0.41	-0.28	-0.2	-0.25	-0.54	-0.34	0.33	-0.096	1	0.85	-0.14	0.67	-0.58	-0.47	-0.28	0.39	-0.38	-0.56	-0.73
16	-0.21	-0.29	0.5	-0.47	-0.18	-0.28	-0.42	-0.3	-0.19	-0.26	-0.52	-0.34	0.34	-0.12	0.85	1	-0.18	0.69	-0.57	-0.45	-0.3	0.38	-0.39	-0.51	-0.67
17	0.094	0.022	-0.2	0.2	0.16	-0.0035	0.11	0.16	0.1	0.12	0.038	-0.01	0.008	-0.076	-0.14	-0.18	1	-0.15	0.12	0.14	0.0069	-0.14	0.12	0.034	0.18
18	-0.2	-0.22	0.43	-0.41	-0.16	-0.2	-0.38	-0.26	-0.19	-0.21	-0.47	-0.32	0.31	-0.12	0.67	0.69	-0.15	1	0.53	0.42	-0.3	0.39	0.33	-0.39	-0.57

Cell 40 of 84

Type here to search

File Edit Selection View Go Run Terminal Help Kidney.ipynb

EXPLORER model.py app.py Kidney.ipynb

KIDNEYDISEASE Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > M+Chronic Kidney Disease > M+Replacing Categorical values with numbers: > M+pe > dataset['pe'].value_counts()

Select Kernel

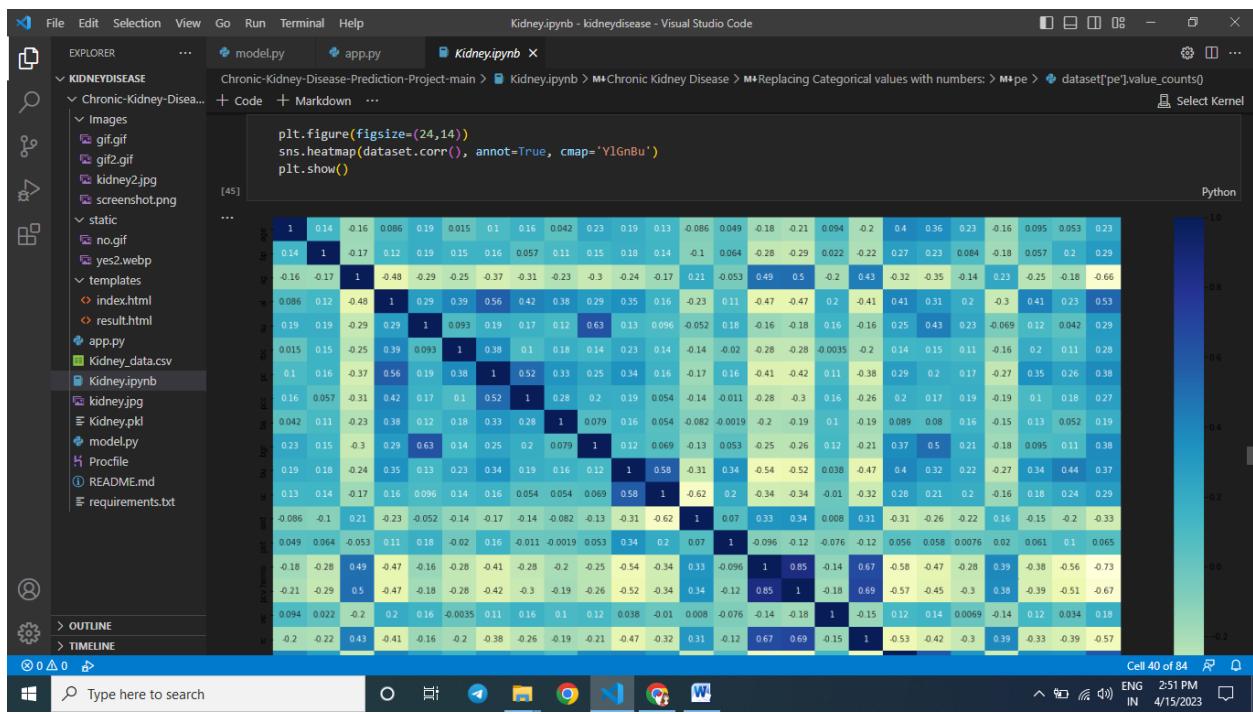
```
plt.figure(figsize=(24,14))
sns.heatmap(dataset.corr(), annot=True, cmap='YlGnBu')
plt.show()
```

Python

[45]

Cell 40 of 84

Type here to search



Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb

EXPLORER model.py app.py Kidney.ipynb

KIDNEYDISEASE Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > M+Chronic Kidney Disease > M+Replacing Categorical values with numbers: > M+pe > dataset['pe'].value_counts()

Images gif.gif gif2.gif kidney2.jpg screenshot.png static no.gif yes2.webp templates index.html result.html app.py Kidney_data.csv Kidney.ipynb kidney.jpg Kidney.pkl model.py Procfile README.md requirements.txt

dataset.head()

	age	bp	sg	al	su	rbc	pc	pcc	ba	bgr	bu	sc	sod	pot	hemo	wc	rc	htn	dm	cad	appet	pe	ane	classification
0	48.0	80.0	1.020	1.0	0.0	0.0	0.0	0.0	0.0	121.0	36.0	1.2	138.0	4.4	15.4	7800.0	5.2	1.0	1.0	0.0	1.0	0.0	0.0	1
1	7.0	50.0	1.020	4.0	0.0	0.0	0.0	0.0	0.0	121.0	18.0	0.8	138.0	4.4	11.3	6000.0	4.8	0.0	0.0	0.0	1.0	0.0	0.0	1
2	62.0	80.0	1.010	2.0	3.0	0.0	0.0	0.0	0.0	423.0	53.0	1.8	138.0	4.4	9.6	7500.0	4.8	0.0	1.0	0.0	0.0	0.0	1.0	1
3	48.0	70.0	1.005	4.0	0.0	0.0	1.0	1.0	0.0	117.0	56.0	3.8	111.0	2.5	11.2	6700.0	3.9	1.0	0.0	0.0	0.0	1.0	1.0	1
4	51.0	80.0	1.010	2.0	0.0	0.0	0.0	0.0	0.0	106.0	26.0	1.4	138.0	4.4	11.6	7300.0	4.6	0.0	0.0	0.0	1.0	0.0	0.0	1

```
# Target feature:  
sns.countplot(dataset['classification'])
```

C:\Users\LENOVO\anaconda3\lib\site-packages\seaborn_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From 'warnings.warn(

```
<AxesSubplot:xlabel='classification', ylabel='count'>
```

Type here to search Cell 40 of 84 ENG 2:51 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb

EXPLORER model.py app.py Kidney.ipynb

KIDNEYDISEASE Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > M+Chronic Kidney Disease > M+Replacing Categorical values with numbers: > M+pe > dataset['pe'].value_counts()

Images gif.gif gif2.gif kidney2.jpg screenshot.png static no.gif yes2.webp templates index.html result.html app.py Kidney_data.csv Kidney.ipynb kidney.jpg Kidney.pkl model.py Procfile README.md requirements.txt

```
<AxesSubplot:xlabel='classification', ylabel='count'>
```

```
# Independent and Dependent Feature:  
X = dataset.iloc[:, :-1]  
y = dataset.iloc[:, -1]
```

X.head()

Type here to search Cell 40 of 84 ENG 2:52 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb

EXPLORER model.py app.py Kidney.ipynb

KIDNEYDISEASE Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > Chronic Kidney Disease > Replacing Categorical values with numbers: > m>pe > dataset['pe'].value_counts()

Select Kernel

```
# Feature Importance:  
from sklearn.ensemble import ExtraTreesClassifier  
import matplotlib.pyplot as plt  
model=ExtraTreesClassifier()  
model.fit(X,y)  
  
plt.figure(figsize=(8,6))  
ranked_features=pd.Series(model.feature_importances_,index=X.columns)  
ranked_features.nlargest(24).plot(kind='barh')  
plt.show()
```

Python

Cell 40 of 84 ENG 2:52 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb

EXPLORER model.py app.py Kidney.ipynb

KIDNEYDISEASE Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > Chronic Kidney Disease > Replacing Categorical values with numbers: > m>pe > dataset['pe'].value_counts()

Select Kernel Python

```
... Index(['sg', 'htn', 'hemo', 'dm', 'al', 'appet', 'rc', 'pc'], dtype='object')
```

```
X = dataset[['sg', 'htn', 'hemo', 'dm', 'al', 'appet', 'rc', 'pc']]  
X.head()
```

Python

Cell 40 of 84 ENG 2:52 PM IN 4/15/2023

```
... sg htn hemo dm al appet rc pc  
0 1.020 1.0 15.4 1.0 1.0 1.0 5.2 0.0  
1 1.020 0.0 11.3 0.0 4.0 1.0 4.8 0.0  
2 1.010 0.0 9.6 1.0 2.0 0.0 4.8 0.0  
3 1.005 1.0 11.2 0.0 4.0 0.0 3.9 1.0  
4 1.010 0.0 11.6 0.0 2.0 1.0 4.6 0.0
```

```
X.tail()
```

Python

Cell 40 of 84 ENG 2:52 PM IN 4/15/2023

```
... sg htn hemo dm al appet rc pc  
395 1.020 0.0 15.7 0.0 0.0 1.0 4.9 0.0  
396 1.025 0.0 16.5 0.0 0.0 1.0 6.2 0.0  
397 1.020 0.0 15.8 0.0 0.0 1.0 5.4 0.0  
398 1.025 0.0 14.2 0.0 0.0 1.0 5.9 0.0  
399 1.025 0.0 15.8 0.0 0.0 1.0 6.1 0.0
```

Cell 40 of 84 ENG 2:52 PM IN 4/15/2023

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb X

EXPLORER model.py app.py Kidney.ipynb X

KIDNEYDISEASE Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > M+Chronic Kidney Disease > M+Replacing Categorical values with numbers: > M+pe > dataset['pe'].value_counts() Select Kernel

```
# Train Test Split:
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test = train_test_split(X,y, test_size=0.3, random_state=33)
```

[55] Python

```
print(X_train.shape)
print(X_test.shape)
```

[56] Python

... (280, 8)
(120, 8)

There is no need of Standardization and Normalization of our dataset, as we using Ensemble Technique.

```
# Importing Performance Metrics:
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
```

[57] Python

```
# RandomForestClassifier:
from sklearn.ensemble import RandomForestClassifier
RandomForest = RandomForestClassifier()
RandomForest = RandomForest.fit(X_train,y_train)

# Predictions:
```

[58] Python

```
y_pred = RandomForest.predict(X_test)

# Performance:
print('Accuracy:', accuracy_score(y_test,y_pred))
print(confusion_matrix(y_test,y_pred))
print(classification_report(y_test,y_pred))
```

Cell 40 of 84 ENG 2:53 PM IN 4/15/2023

Type here to search

Kidney.ipynb - kidneydisease - Visual Studio Code

File Edit Selection View Go Run Terminal Help Kidney.ipynb X

EXPLORER model.py app.py Kidney.ipynb X

KIDNEYDISEASE Chronic-Kidney-Disease-Prediction-Project-main > Kidney.ipynb > M+Chronic Kidney Disease > M+Replacing Categorical values with numbers: > M+pe > dataset['pe'].value_counts() Select Kernel

There is no need of Standardization and Normalization of our dataset, as we using Ensemble Technique.

```
# Importing Performance Metrics:
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
```

[57] Python

```
# RandomForestClassifier:
from sklearn.ensemble import RandomForestClassifier
RandomForest = RandomForestClassifier()
RandomForest = RandomForest.fit(X_train,y_train)

# Predictions:
y_pred = RandomForest.predict(X_test)

# Performance:
print('Accuracy:', accuracy_score(y_test,y_pred))
print(confusion_matrix(y_test,y_pred))
print(classification_report(y_test,y_pred))
```

[58] Python

```
... Accuracy: 0.975
[[55  3]
 [ 0 62]]
 precision    recall   f1-score   support
      0       1.00     0.95     0.97      58
      1       0.95     1.00     0.98      62
accuracy                           0.97      120
```

Cell 40 of 84 ENG 2:53 PM IN 4/15/2023

Type here to search

The screenshot shows a Visual Studio Code interface with the following details:

- File Explorer:** Shows files like model.py, app.py, gif.gif, Kidney.ipynb, and various static and template files.
- Code Editor:** Displays Python code for an AdaBoostClassifier, including imports, model fitting, predictions, and performance metrics.

```
# AdaBoostClassifier:  
from sklearn.ensemble import AdaBoostClassifier  
AdaBoost = AdaBoostClassifier()  
AdaBoost = AdaBoost.fit(X_train,y_train)  
  
# Predictions:  
y_pred = AdaBoost.predict(X_test)  
  
# Performance:  
print('Accuracy:', accuracy_score(y_test,y_pred))  
print(confusion_matrix(y_test,y_pred))  
print(classification_report(y_test,y_pred))
```
- Output:** Shows the execution results for the AdaBoost classifier, including accuracy, precision, recall, f1-score, and support for each class.

	precision	recall	f1-score	support
0	1.00	0.95	0.97	58
1	0.95	1.00	0.98	62
accuracy			0.97	120
macro avg	0.98	0.97	0.97	120
weighted avg	0.98	0.97	0.97	120
- Terminal:** Shows the command "Cell 40 of 84" and the date/time "4/15/2023".

The screenshot shows a Visual Studio Code interface with the following details:

- File Explorer:** Shows files like model.py, app.py, gif.gif, Kidney.ipynb, and various static and template files.
- Code Editor:** Displays Python code for a GradientBoostingClassifier, including imports, model fitting, predictions, and performance metrics.

```
# GradientBoostingClassifier:  
from sklearn.ensemble import GradientBoostingClassifier  
GradientBoost = GradientBoostingClassifier()  
GradientBoost = GradientBoost.fit(X_train,y_train)  
  
# Predictions:  
y_pred = GradientBoost.predict(X_test)  
  
# Performance:  
print('Accuracy:', accuracy_score(y_test,y_pred))  
print(confusion_matrix(y_test,y_pred))  
print(classification_report(y_test,y_pred))
```
- Output:** Shows the execution results for the GradientBoostingClassifier, including accuracy, precision, recall, f1-score, and support for each class.

	precision	recall	f1-score	support
0	1.00	0.95	0.97	58
1	0.95	1.00	0.98	62
accuracy			0.97	120
macro avg	0.98	0.97	0.97	120
weighted avg	0.98	0.97	0.97	120
- Terminal:** Shows the command "Cell 40 of 84" and the date/time "4/15/2023".

4.ADVANTAGES AND DISADVANTAGES

ADVANTAGES

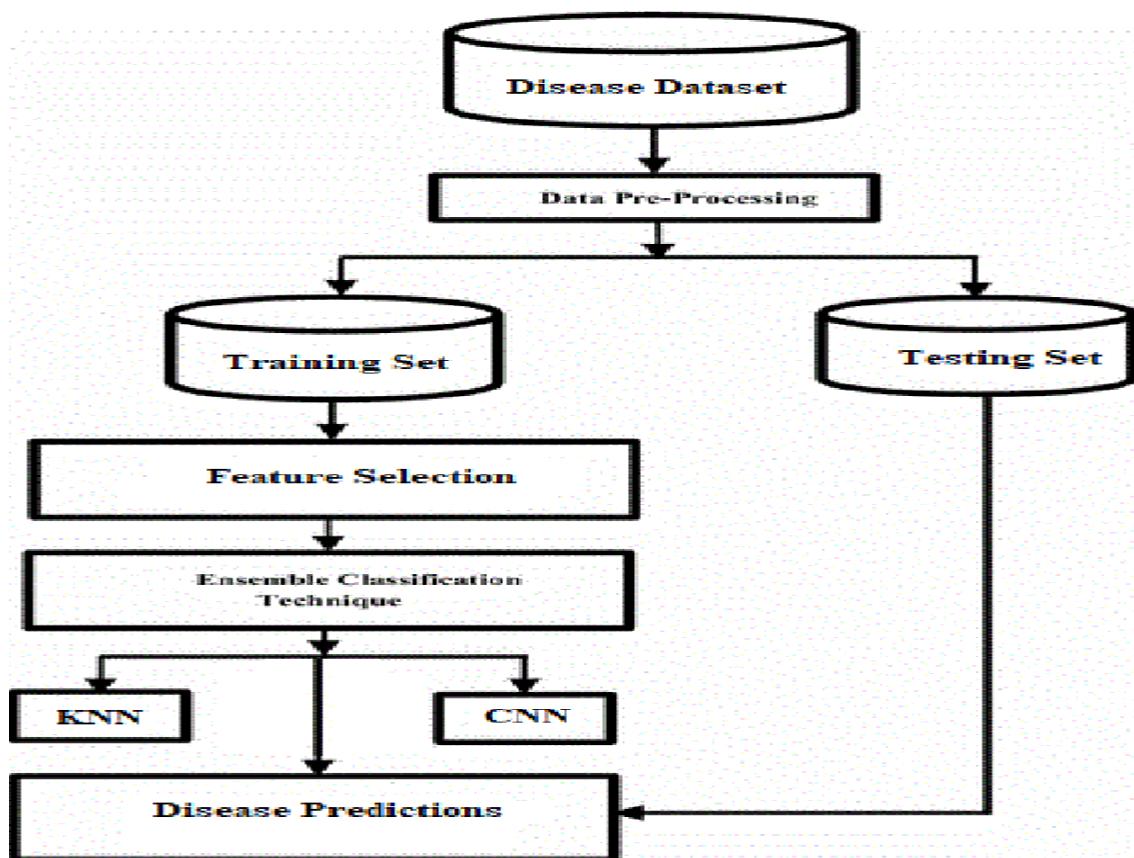
- It helps physicians to identify best treatment for particular disease.
- It can produce fast analysis report
- This system is Operational efficient
- Operational cost is reduced.

DISADVANTAGES

- Data ownership problems
- Privacy and security related issues
- Human data administration and so on.,

5.APPLICATIONS

Machine learning algorithms are used to diagnose and predict the kidney disease in human. The learning-based algorithms plays a vital role in supporting decision-making in disease diagnosis and prediction. This paper investigates traditional classification algorithms and neural network-based machine learning for the chronic kidney disease dataset which is taken from Kaggle dataset.



6.CONCLUSION

This project is a medical sector application high helps the medical practitioners in predicting the Chronic Kidney Disease (CKD) disease based on the CKD parameters. It is automation for CKD disease prediction and it identifies the disease, its stages in an efficient and economically manner. It is successfully accomplished by applying the k-Nearest Neighbour (KNN) and Naive Bayes algorithms for classification. This classification technique comes under data mining technology. This algorithm takes CKD parameters as input and predicts the disease based on old CKD patient's data.

7.FUTURE SCOPE

SMS/Email module

In the proposed system, admin assigns Id and password for doctors and receptionists and is intimated manually, so we can add SMS/Email module as a future enhancement where doctors and receptionists receive an SMS or Email regarding the Id and password.

Query module

We can add the query module as a future enhancement to the application where doctor, receptionist and admin of the application can interact with each other

APPENDIES

A. SOURCE CODE

```
# Importing Libraries:  
import pandas as pd  
import numpy as np  
import seaborn as sns  
import matplotlib.pyplot as plt  
  
# for displaying all feature from dataset:  
pd.pandas.set_option('display.max_columns', None)  
# Reading Dataset:  
dataset = pd.read_csv("Kidney_data.csv")  
# Top 5 records:  
dataset.head()  
# Dropping unneccsary feature :  
dataset = dataset.drop('id', axis=1)  
# Shape of dataset:  
dataset.shape  
# Cheaking Missing (NaN) Values:  
dataset.isnull().sum()  
# Description:  
dataset.describe()  
# Datatypes:  
dataset.dtypes  
dataset.head()  
dataset['rbc'].value_counts()  
dataset['rbc'] = dataset['rbc'].replace(to_replace = {'normal' : 0, 'abnormal' : 1})  
dataset['pc'].value_counts()  
dataset['pc'] = dataset['pc'].replace(to_replace = {'normal' : 0, 'abnormal' : 1})  
dataset['pcc'].value_counts()  
dataset['pcc'] = dataset['pcc'].replace(to_replace = {'notpresent':0,'present':1})  
dataset['ba'].value_counts()  
dataset['ba'] = dataset['ba'].replace(to_replace = {'notpresent':0,'present':1})  
dataset['htn'].value_counts()  
dataset['htn'] = dataset['htn'].replace(to_replace = {'yes' : 1, 'no' : 0})  
dataset['dm'].value_counts()  
dataset['dm'] = dataset['dm'].replace(to_replace = {'\tyses':'yes', ' yes':'yes', '\tno':'no'})
```

```

dataset['dm'] = dataset['dm'].replace(to_replace = {'yes' : 1, 'no' : 0})
dataset['cad'].value_counts()
dataset['cad'] = dataset['cad'].replace(to_replace = {'\tno':'no'})
dataset['cad'] = dataset['cad'].replace(to_replace = {'yes' : 1, 'no' : 0})
dataset['appet'].unique()
dataset['appet'] =
dataset['appet'].replace(to_replace= {'good':1,'poor':0,'no':np.nan})
dataset['pe'].value_counts()
dataset['pe'] = dataset['pe'].replace(to_replace = {'yes' : 1, 'no' : 0})
dataset['ane'].value_counts()
dataset['ane'] = dataset['ane'].replace(to_replace = {'yes' : 1, 'no' : 0})
dataset['classification'].value_counts()
dataset['classification'] =
dataset['classification'].replace(to_replace= {'ckd\t':'ckd'})
dataset["classification"] = [1 if i == "ckd" else 0 for i in
dataset["classification"]]
dataset.head()
# Datatypes:
dataset.dtypes
dataset['pcv'] = pd.to_numeric(dataset['pcv'], errors='coerce')
dataset['wc'] = pd.to_numeric(dataset['wc'], errors='coerce')
dataset['rc'] = pd.to_numeric(dataset['rc'], errors='coerce')
# Datatypes:
dataset.dtypes
# Description:
dataset.describe()
# Cheaking Missing (NaN) Values:
dataset.isnull().sum().sort_values(ascending=False)
dataset.columns
features = ['age', 'bp', 'sg', 'al', 'su', 'rbc', 'pc', 'pcc', 'ba', 'bgr', 'bu',
'sc', 'sod', 'pot', 'hemo', 'pcv', 'wc', 'rc', 'htn', 'dm', 'cad',
'appet', 'pe', 'ane']
for feature in features:
    dataset[feature] = dataset[feature].fillna(dataset[feature].median())
dataset.isnull().any().sum()
plt.figure(figsize=(24,14))
sns.heatmap(dataset.corr(), annot=True, cmap='YlGnBu')
plt.show()
dataset.drop('pcv', axis=1, inplace=True)
dataset.head()
# Target feature:
sns.countplot(dataset['classification'])
# Independent and Dependent Feature:

```

```

X = dataset.iloc[:, :-1]
y = dataset.iloc[:, -1]
X.head()
# Feature Importance:
from sklearn.ensemble import ExtraTreesClassifier
import matplotlib.pyplot as plt
model=ExtraTreesClassifier()
model.fit(X,y)

plt.figure(figsize=(8,6))
ranked_features=pd.Series(model.feature_importances_,index=X.columns)
ranked_features.nlargest(24).plot(kind='barh')
plt.show()
ranked_features.nlargest(8).index
X = dataset[['sg', 'htn', 'hemo', 'dm', 'al', 'appet', 'rc', 'pc']]
X.head()
X.tail()
y.head()
# Train Test Split:
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test = train_test_split(X,y, test_size=0.3,
random_state=33)
print(X_train.shape)
print(X_test.shape)
# Importing Performance Metrics:
from sklearn.metrics import accuracy_score, confusion_matrix,
classification_report
# RandomForestClassifier:
from sklearn.ensemble import RandomForestClassifier
RandomForest = RandomForestClassifier()
RandomForest = RandomForest.fit(X_train,y_train)

# Predictions:
y_pred = RandomForest.predict(X_test)

# Performance:
print('Accuracy:', accuracy_score(y_test,y_pred))
print(confusion_matrix(y_test,y_pred))
print(classification_report(y_test,y_pred))
# AdaBoostClassifier:
from sklearn.ensemble import AdaBoostClassifier
AdaBoost = AdaBoostClassifier()
AdaBoost = AdaBoost.fit(X_train,y_train)

```

```
# Predictions:  
y_pred = AdaBoost.predict(X_test)  
  
# Performance:  
print('Accuracy:', accuracy_score(y_test,y_pred))  
print(confusion_matrix(y_test,y_pred))  
print(classification_report(y_test,y_pred))  
# GradientBoostingClassifier:  
from sklearn.ensemble import GradientBoostingClassifier  
GradientBoost = GradientBoostingClassifier()  
GradientBoost = GradientBoost.fit(X_train,y_train)  
  
# Predictions:  
y_pred = GradientBoost.predict(X_test)  
  
# Performance:  
print('Accuracy:', accuracy_score(y_test,y_pred))  
print(confusion_matrix(y_test,y_pred))  
print(classification_report(y_test,y_pred))
```