



Internship Report On Drug Classification Using Decision Tree

Submitted by
Kamale Natasha
245521748088
Keshav Memorial Engineering College

Kunapuli Vaishali
245521748094
Keshav Memorial Engineering College

Gali Johnson Rohith Reddy
245521733145
Keshav Memorial Engineering College

Submitted to
Mallika Srivastava
Head, Training Delivery
EISystems Services

&

Mayur Dev Sewak
Head, Internships & Trainings
EISystems Services

Student's Declaration

I, KAMALE NATASHA, a student of B.E program, Roll No. 245521748088 of the Department of CSE (AI & ML), Keshav Memorial Engineering College do hereby declare that I have completed the mandatory internship in EISystems Technologies under the faculty guideship of P. Naresh Kumar, Department of CSE (AI & ML), Keshav Memorial Engineering College.


- 15/07/24
(Signature and Date)

Endorsements


SIGNATURE

P. Naresh Kumar
Computer Science and Engineering (Artificial Intelligence & Machine Learning),
Keshav Memorial Engineering College.

SIGNATURE

Head of the Department:

Dr. B. Devender

Computer Science and Engineering (Artificial Intelligence & Machine Learning),
Keshav Memorial Engineering College.



HEAD
Department of CSE (AI & ML)
Keshav Memorial Engineering College


EISYSTEMS SERVICES

FF-110, Express Greens Plaza, Sector 1
Vaishali - Delhi NCR - India 201010

W: www.eisystems.in | E: info@eisystems.in | T: +91 92122-51000

Student's Declaration

I, KUNAPULI VAISHALI, a student of B.E program, Roll No. 245521748094 of the Department of CSE (AI & ML), Keshav Memorial Engineering College do hereby declare that I have completed the mandatory internship in EISYSTEMS Technologies under the faculty guideship of P. Naresh Kumar, Department of CSE (AI & ML), Keshav Memorial Engineering College.


16/07/24
(Signature and Date)

Endorsements


SIGNATURE

P. Naresh Kumar
Computer Science and Engineering (Artificial Intelligence & Machine Learning),
Keshav Memorial Engineering College.


SIGNATURE

Head of the Department:
Dr. B. Devender
Computer Science and Engineering (Artificial Intelligence & Machine Learning),
Keshav Memorial Engineering College.

HEAD
Department of CSE (AI & ML)
Keshav Memorial Engineering College

EISYSTEMS SERVICES

FF-110, Express Greens Plaza, Sector 1
Vaishali – Delhi NCR – India 201010

W: www.eisystems.in | E: info@eisystems.in | T: +91 92122-51000

Student's Declaration

I, GALI JOHNSON ROHITH REDDY, a student of B.E program, Roll No. 245521733145 of the Department of CSE, Keshav Memorial Engineering College do hereby declare that I have completed the mandatory internship in Eisystems Technologies under the faculty guideship of D.V.S.S Subrahmanyam, Department of CSE, Keshav Memorial Engineering College.

Rohith 29/6/24
(Signature and Date)

Endorsements

D.V.S.S 29/6/24
SIGNATURE
D.V.S.S Subrahmanyam
Computer Science and Engineering
Keshav Memorial Engineering College

D.V.S.S 29/6/24
SIGNATURE
Head of the Department :
D.V.S.S Subrahmanyam
Computer Science and Engineering
Keshav Memorial Engineering College
Dept. of Computer Science & Engineering,
Keshav Memorial Engineering College.

EISYSTEMS SERVICES

FF-110, Express Greens Plaza, Sector 1
Vaishali – Delhi NCR – India 201010

W: www.eisystems.in | E: info@eisystems.in | T: +91 92122-51000

Table of Content

| Serial No | Title | Page No |
|-----------|--|---------|
| 1. | List of Tables | 5 |
| 2. | Executive Summary | 6 |
| 3. | Overview Of Organization | 7 |
| 4. | Project Summary | 8 |
| 5. | Data Flow Diagram\Process Flow | 9 |
| 6. | Code / Program with Supported Screenshots | 10 |
| 7. | Input / Output with Datasets & Supported Screenshots | 17 |
| 8. | References | 20 |
| 9. | Student Self Evaluation of the Short-Term Internship | 21 |
| 10. | Annexure 1 (Daily Activity Report) | 24 |
| 11. | Annexure 2 (Weekly Activity Report) | 32 |

List of Tables

| Serial No | Table Caption | Page No |
|-----------|-------------------------------------|---------|
| 1. | Drug200 dataset | 15 |
| 2. | Annexure 1 (Daily Activity Report) | 22 |
| 3. | Annexure 2 (Weekly Activity Report) | 30 |

Executive Summary

During the internship focused on artificial intelligence, I successfully mastered Python programming concepts, including data types, control statements, loops, dictionaries, tuples, lists, and object-oriented programming (OOP) concepts like class, object, inheritance, encapsulation, polymorphism, abstraction. This foundational knowledge enabled me to effectively manipulate and analyze data using advanced libraries and frameworks.

In the training phase, I gained proficiency in essential AI concepts and tools such as `NumPy`, `pandas`, `scikit-learn`, and `matplotlib`. These tools were instrumental in data preprocessing, visualization, and implementing machine learning models like linear regression, logistic regression, K-Nearest Neighbors (KNN), Support Vector Machines (SVM), and Decision Trees.

Throughout the internship, I engaged in a variety of activities, starting with learning Python fundamentals and progressively applying them to AI projects. I actively participated in data preprocessing tasks, model training, evaluation, and serialization using libraries like `pandas`, `scikit-learn`, and `pickle`. Additionally, I conducted experiments to optimize model performance and gained practical insights into interpreting and visualizing data using `matplotlib`.

The various objectives and outcomes I have achieved during the internship and project phases are:

The internship's primary objective was to master AI concepts and their practical applications. By the end of the internship, I achieved the following outcomes:

- Proficiency in Python programming for AI, encompassing foundational concepts and advanced data manipulation techniques.
- Expertise in utilizing essential AI libraries (`NumPy`, `pandas`, `scikit-learn`, `matplotlib`) for data preprocessing, visualization, and model implementation.
- Hands-on experience with various machine learning algorithms, enhancing my ability to analyze data, train models, and evaluate their performance effectively.

These achievements have equipped me with the skills necessary to contribute effectively to AI projects, leveraging Python and its associated libraries to solve complex problems in healthcare and beyond

Overview of Organization

EISystems Services, commonly known as EISystem, is an Indian technology company that specializes in conducting training sessions and workshops (IIT's) in various fields like **Python Programming, Data Science, Artificial Intelligence, Machine Learning, Cloud Computing, Website development with JavaScript, Bootstrap, React Js** and many more.

The Vision of the organization includes becoming leading provider of creative and innovative training solutions. The main aim and mission of EI Systems Services is to empower students with practical skills and knowledge through high-quality training programs.

EI Systems Values quality education, practical experience and skills and promotes solid foundational education.

The Policy of the organization is to encourage and provide practical experience through immersive learning Sessions and to let intern participate in projects where theoretical knowledge is applied to practical and Real world scenarios and also to develop trending and demanding skills that are aligned with current industry standards.

EI Systems has made a significant impact on technology education by training around 50,000 students and reaching approximately 200,000 students through their various initiatives.

EI Systems provides both offline and online training sessions and internships, offering flexibility and accessibility to a wide range of learners.

EI Systems has established its presence in several esteemed institutions including:

- > Indian Institute of Science, Bangalore
- > Indian Institute of Technology, Bombay
- > Indian Institute of Technology, Delhi
- > Indian Institute of Technology, Kanpur
- > Indian Institute of Technology, Madras
- > Indian Institute of Technology, Roorkee
- > Indian Institute of Technology (Banaras Hindu University), Varanasi

And many more

Project Summary

The primary idea behind “**Drug Classification using Decision Tree**” project is to develop a machine learning model that can classify patients into different drug categories based on their age, sex, blood pressure (BP), cholesterol levels, and sodium to potassium ratio (Na_to_K). This can aid healthcare professionals in making informed decisions about drug prescriptions based on patient profiles.

This project involves building a **decision tree classifier** to predict the type of drug that a patient should take. The dataset used contains information about patients, including their age, sex, BP, cholesterol, and Na_to_K ratio. By training a model on this data, we aim to achieve high accuracy in drug classification.

The software used in this project is **Python language**. Python was used as the primary programming language due to its versatility and robust library support. Key libraries included **pandas** for data manipulation and analysis, and **sklearn** for **machine learning** algorithms and preprocessing tasks. The warnings library was utilized to manage and suppress unnecessary warning messages. Additionally, **pickle** was employed for model serialization, allowing the trained model to be saved and reused without retraining. These tools collectively enabled the efficient development and execution of the drug classification model. Software like **VS code** and **google colab** are used for execution of project. The working of the model includes:

- The data is first read and preprocessed.
- Encoding categorical variables (Sex, BP, Cholesterol) using **LabelEncoder**.
- Splitting the dataset into training and testing sets using **train_test_split**.
- A **decision tree classifier** is trained on the training data.
- The model is evaluated on the testing data, achieving an accuracy of approximately **accuracy_score**.
- The trained model is serialized and saved to a file using pickle.

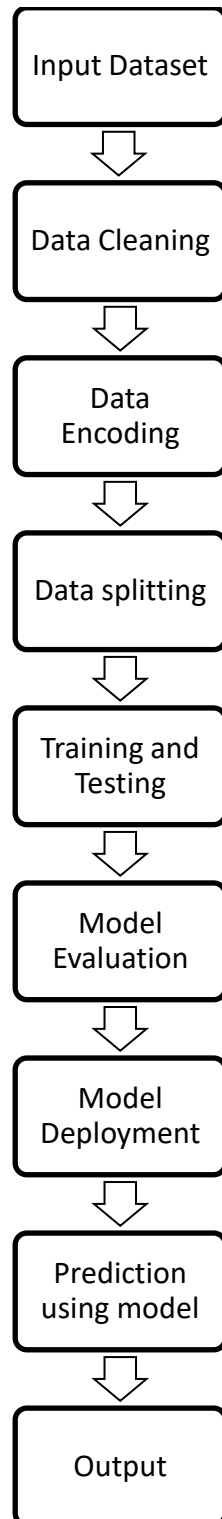
The Decision Tree Classifier in the code is used to classify patients into drug categories based on their features. It learns decision rules from the training data and applies these rules to predict the drug category for new patients.

Research involved understanding the dataset and preprocessing requirements

- **Preprocessing:** Researching methods for encoding categorical variables to numerical values.
- **Model Selection:** Deciding on using a Decision Tree Classifier based on its interpretability and suitability for classification problems.
- **Evaluation Metrics:** Using accuracy as the evaluation metric for model performance.

Further, this project can enhance the model's accuracy by incorporating more advanced algorithms like Random Forest and integrating this model into a healthcare application could provide real-time drug recommendations to medical professionals.

Data Flow Diagram / Process Flow



Code / Program with Supported Screenshots

In Google Colab Importing libraries and reading dataset

```
import pandas as pd
```

```
dt = pd.read_csv(r"/content/drug200.csv")  
dt.head()
```

| | Age | Sex | BP | Cholesterol | Na_to_K | Drug |
|---|-----|-----|--------|-------------|---------|-------|
| 0 | 23 | F | HIGH | HIGH | 25.355 | drugY |
| 1 | 47 | M | LOW | HIGH | 13.093 | drugC |
| 2 | 47 | M | LOW | HIGH | 10.114 | drugC |
| 3 | 28 | F | NORMAL | HIGH | 7.798 | drugX |
| 4 | 61 | F | LOW | HIGH | 18.043 | drugY |

Check if the data is clean by Preprocessing the data


```
[5] dt.info()
```

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 200 entries, 0 to 199  
Data columns (total 6 columns):  
#   Column          Non-Null Count  Dtype  
---  ---  
0   Age             200 non-null   int64  
1   Sex             200 non-null   object  
2   BP              200 non-null   object  
3   Cholesterol      200 non-null   object  
4   Na_to_K         200 non-null   float64  
5   Drug            200 non-null   object  
dtypes: float64(1), int64(1), object(4)  
memory usage: 9.5+ KB
```



```
[6] dt.isnull().sum()
```

```
Age             0  
Sex             0  
BP              0  
Cholesterol      0  
Na_to_K         0  
Drug            0  
dtype: int64
```

```
[7] dt.describe()
```



| | Age | Na_to_K |
|-------|------------|------------|
| count | 200.000000 | 200.000000 |
| mean | 44.315000 | 16.084485 |
| std | 16.544315 | 7.223956 |
| min | 15.000000 | 6.269000 |
| 25% | 31.000000 | 10.445500 |
| 50% | 45.000000 | 13.936500 |
| 75% | 58.000000 | 19.380000 |
| max | 74.000000 | 38.247000 |


```
dt.duplicated().sum()
```




```
0
```

No duplicate value exists in the data



Create an array of all input features and define it as “x”



```
x = dt[["Age", "Sex", "BP", "Cholesterol", "Na_to_K"]]  
x
```



| | Age | Sex | BP | Cholesterol | Na_to_K |
|-----|-----|-----|--------|-------------|---------|
| 0 | 23 | F | HIGH | HIGH | 25.355 |
| 1 | 47 | M | LOW | HIGH | 13.093 |
| 2 | 47 | M | LOW | HIGH | 10.114 |
| 3 | 28 | F | NORMAL | HIGH | 7.798 |
| 4 | 61 | F | LOW | HIGH | 18.043 |
| ... | ... | ... | ... | ... | ... |
| 195 | 56 | F | LOW | HIGH | 11.567 |
| 196 | 16 | M | LOW | HIGH | 12.006 |
| 197 | 52 | M | NORMAL | HIGH | 9.894 |
| 198 | 23 | M | NORMAL | NORMAL | 14.020 |
| 199 | 40 | F | LOW | NORMAL | 11.349 |

200 rows × 5 columns

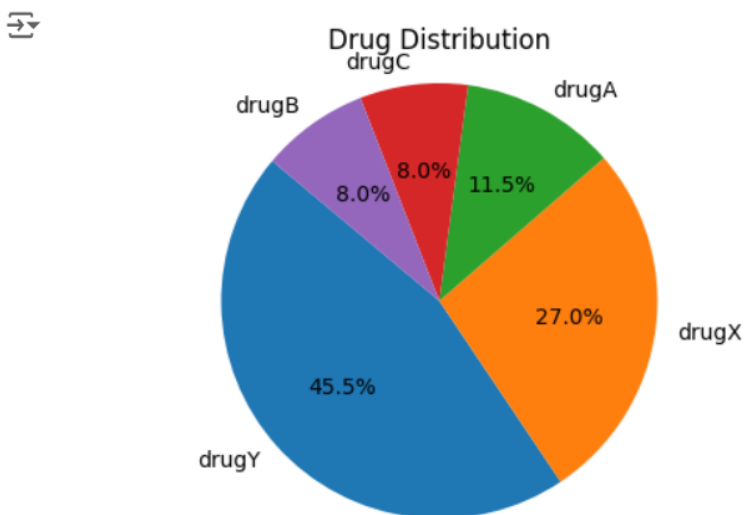
```
▶ y = dt.Drug
y
```

```
⇒ 0    drugY
   1    drugC
   2    drugC
   3    drugX
   4    drugY
   ...
  195   drugC
  196   drugC
  197   drugX
  198   drugX
  199   drugX
Name: Drug, Length: 200, dtype: object
```

```
[11] import warnings
      warnings.filterwarnings("ignore")
```

```
[12] filtered_dt = dt[dt['Drug'].isin(['drugA', 'drugB', 'drugC', 'drugX', 'drugY'])]
      drug_count = filtered_dt['Drug'].value_counts()
```

```
▶ import matplotlib.pyplot as plt
  plt.figure(figsize=(6, 4))
  plt.pie(drug_count.values, labels=drug_count.index, autopct='%1.1f%%', startangle=140)
  plt.title('Drug Distribution')
  plt.axis('equal')
  plt.show()
```



The above pie chart depicts the data balance in drug dataset with balanced proportion of all drugs

Encoding all the variables into numeric format

```
[14] from sklearn.preprocessing import LabelEncoder
      le=LabelEncoder()
      y = le.fit_transform(y)
      y
```

```
array([4, 2, 2, 3, 4, 3, 4, 2, 4, 4, 2, 4, 4, 4, 3, 4, 3, 0, 2, 4, 4, 4,
        4, 4, 4, 4, 4, 3, 4, 4, 3, 1, 3, 4, 3, 3, 0, 3, 3, 3, 4, 1, 4, 3,
        3, 3, 0, 2, 4, 4, 4, 3, 4, 4, 1, 2, 1, 4, 3, 4, 4, 0, 4, 3, 1, 4,
        0, 3, 4, 4, 1, 4, 3, 4, 4, 4, 0, 4, 0, 3, 1, 3, 2, 0, 2, 1, 3, 4,
        4, 4, 4, 4, 4, 4, 3, 4, 4, 4, 0, 0, 2, 3, 4, 3, 3, 4, 1, 4,
        0, 3, 3, 3, 3, 4, 3, 3, 0, 4, 4, 4, 4, 4, 1, 4, 4, 3, 4, 3, 4, 4,
        3, 4, 4, 3, 1, 0, 1, 3, 0, 4, 1, 4, 0, 3, 3, 0, 3, 2, 0, 1, 3, 3,
        4, 2, 0, 4, 2, 3, 3, 1, 3, 4, 4, 4, 4, 3, 4, 0, 3, 3, 4, 4, 0, 4,
        0, 4, 4, 4, 3, 3, 4, 4, 4, 1, 0, 4, 4, 4, 0, 4, 2, 4, 2, 2, 3,
        3, 3])
```

```
[15] bp=le.fit_transform(dt[["BP"]])
      bp
```

```
array([0, 1, 1, 2, 1, 2, 2, 1, 2, 1, 1, 0, 1, 1, 2, 0, 1, 0, 1, 0, 1, 2,
        1, 1, 1, 0, 0, 2, 1, 1, 2, 0, 1, 0, 2, 2, 0, 1, 2, 2, 2, 0, 2, 2,
        2, 2, 0, 1, 2, 1, 0, 2, 1, 0, 0, 1, 0, 0, 2, 0, 1, 0, 1, 1, 0, 2,
        0, 2, 2, 0, 0, 2, 2, 2, 0, 1, 0, 0, 0, 1, 0, 2, 1, 0, 1, 0, 2, 1,
        0, 2, 2, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 2, 0, 1, 2, 1, 0, 2,
        0, 2, 1, 1, 2, 0, 2, 2, 0, 0, 2, 0, 2, 2, 0, 0, 0, 2, 1, 2, 2, 1,
        1, 2, 0, 1, 0, 0, 0, 2, 0, 1, 0, 0, 0, 2, 1, 0, 1, 1, 0, 0, 2, 1,
        1, 1, 0, 1, 1, 1, 2, 0, 2, 0, 0, 1, 1, 2, 1, 0, 2, 1, 2, 1, 0, 0,
        0, 2, 2, 2, 0, 2, 1, 0, 0, 2, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 2,
        2, 1])
```

```
Sex=le.fit_transform(dt[["Sex"]])
Sex
```


```
array([0, 1, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 1, 0, 0, 0, 1, 1, 1, 0, 1, 1,
        1, 0, 0, 0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 1, 0, 0, 1, 1,
        0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 1, 1, 1, 1, 0, 1, 1, 1, 0, 0,
        1, 1, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 1,
        0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1,
        1, 0, 1, 0, 0, 1, 1, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0, 0, 1,
        1, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1, 1, 0,
        1, 1, 1, 1, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0,
        1, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1,
        1, 0])
```

```
[17] Cholesterol=le.fit_transform(dt[["Cholesterol"]])
      Cholesterol
```




```
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0,
        1, 0, 0, 1, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 1, 1, 0,
        1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 1,
        0, 1, 0, 1, 0, 0, 0, 1, 1, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0,
        1, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0,
        0, 1, 1, 1, 1, 0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1,
        1, 0, 0, 1, 0, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 1, 1, 1, 0,
        1, 0, 0, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 1, 1, 1, 1, 0,
        1, 0, 0, 0, 1, 0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0,
        1, 1])
```

Appending the encoded values to the assigned array for model processing

```
x["BP"]=bp
x["Sex"]=Sex
x["Cholesterol"]=Cholesterol
x
```



| | Age | Sex | BP | Cholesterol | Na_to_K |
|-----|-----|-----|-----|-------------|---------|
| 0 | 23 | 0 | 0 | 0 | 25.355 |
| 1 | 47 | 1 | 1 | 0 | 13.093 |
| 2 | 47 | 1 | 1 | 0 | 10.114 |
| 3 | 28 | 0 | 2 | 0 | 7.798 |
| 4 | 61 | 0 | 1 | 0 | 18.043 |
| ... | ... | ... | ... | ... | ... |
| 195 | 56 | 0 | 1 | 0 | 11.567 |
| 196 | 16 | 1 | 1 | 0 | 12.006 |
| 197 | 52 | 1 | 2 | 0 | 9.894 |
| 198 | 23 | 1 | 2 | 1 | 14.020 |
| 199 | 40 | 0 | 1 | 1 | 11.349 |


200 rows × 5 columns

Split the dataset for training and testing and apply decision tree classifier

```
[19] from sklearn.model_selection import train_test_split
      train_x,test_x,train_y,test_y=train_test_split(x,y,test_size=0.2,random_state=10)
```

```
[20] from sklearn.tree import DecisionTreeClassifier
      model=DecisionTreeClassifier()
```

```
[21] model.fit(train_x,train_y)
```



▾ DecisionTreeClassifier
 DecisionTreeClassifier()

```
[22] from sklearn.metrics import accuracy_score, classification_report
      y_pred = model.predict(test_x)
      accuracy = accuracy_score(test_y, y_pred)
      accuracy
```

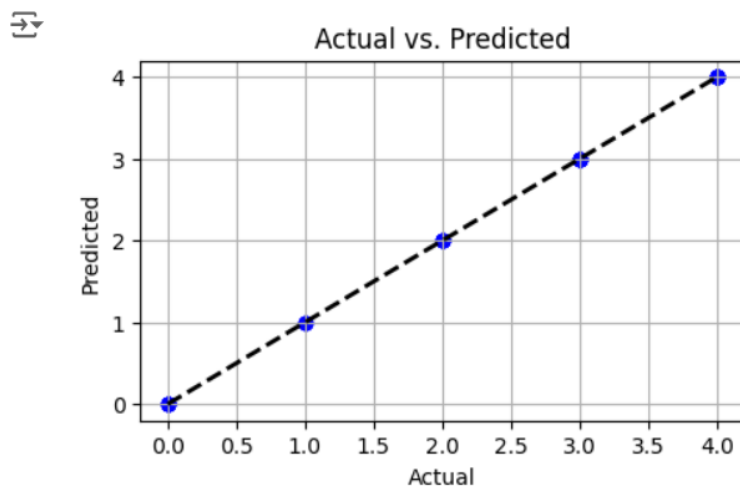
⇒ 1.0

```
[23] model.predict(test_x)==test_y
```

⇒ array([True, True, True, True, True, True, True, True, True, True,
 True, True, True, True, True, True, True, True, True, True,
 True, True, True, True, True, True, True, True, True, True,
 True, True, True, True])

Check for model accuracy

```
plt.figure(figsize=(5, 3))
plt.scatter(test_y, y_pred, color='blue')
plt.plot([test_y.min(), test_y.max()], [test_y.min(), test_y.max()], 'k--', lw=2)
plt.xlabel('Actual')
plt.ylabel('Predicted')
plt.title('Actual vs. Predicted')
plt.grid(True)
plt.show()
```



The above graph depicts that the model is 100% accurate.

Deploying the model

```
import pickle
pickle.dump(model, open("Drug.pkl", "wb"))
```


Streamlit is used for construct User Interface of the model

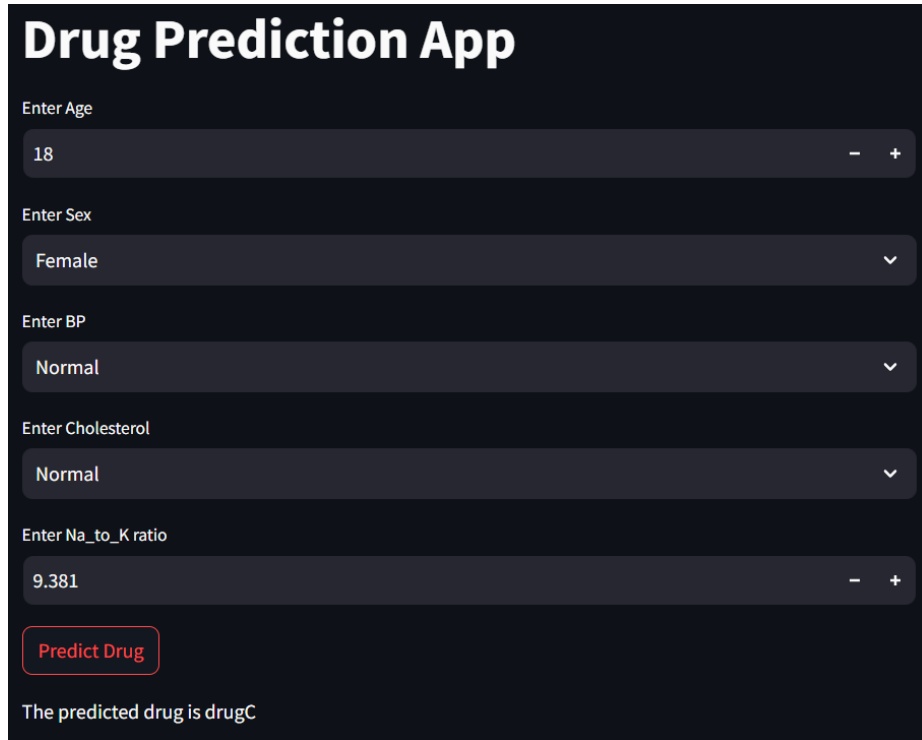
```
import streamlit as st
import pickle
import numpy as np
# Load the trained model
model = pickle.load(open("Drug.pkl", "rb"))
# Streamlit interface
st.title("Drug Prediction App")
# Input fields
age = st.number_input("Enter Age", min_value=0)
sex = st.selectbox("Enter Sex", ["Male", "Female"])
bp = st.selectbox("Enter BP", ["Low", "Normal", "High"])
cholesterol = st.selectbox("Enter Cholesterol", ["Normal", "High"])
na_to_k = st.number_input("Enter Na_to_K ratio", min_value=0.0, format="%.2f")
# Encode categorical inputs
sex_encoded = 1 if sex == "Male" else 0
bp_encoded = {"Low": 0, "Normal": 1, "High": 2}[bp]
cholesterol_encoded = 1 if cholesterol == "High" else 0
# Prediction button
if st.button("Predict Drug"):
    # Prepare input for prediction
    features = np.array([age, sex_encoded, bp_encoded, cholesterol_encoded, na_to_k])
    prediction = model.predict(features)
    # Mapping the predictions to drug name
    predicted_drug = ""
    if prediction[0] == 0:
        predicted_drug = 'drugA'
    elif prediction[0] == 1:
        predicted_drug = 'drugB'
    elif prediction[0] == 2:
        predicted_drug = 'drugC'
    elif prediction[0] == 3:
        predicted_drug = 'drugX'
    elif prediction[0] == 4:
        predicted_drug = 'drugY'
    # Display the result
    st.write(f"The predicted drug is {predicted_drug}")
```

Input / Output with Datasets & Supported Screenshots

Input used for the model is the drug200.csv dataset:

"C:\Users\NATHASHA K\Downloads\drug200.csv"

Output window



Drug Prediction App

Enter Age
18 - +

Enter Sex
Female ▾

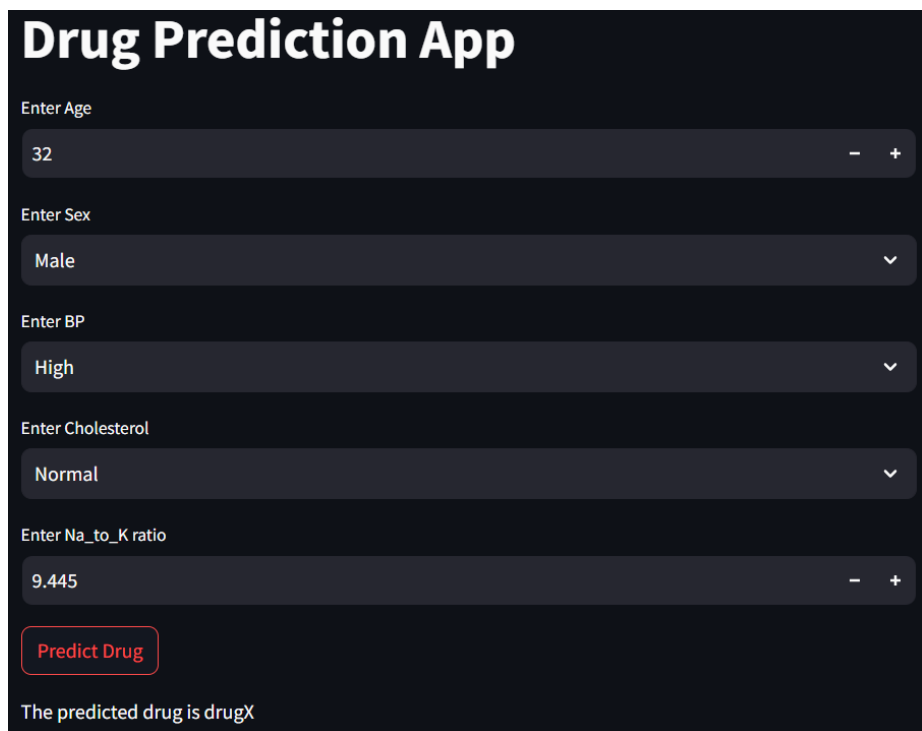
Enter BP
Normal ▾

Enter Cholesterol
Normal ▾

Enter Na_to_K ratio
9.381 - +

Predict Drug

The predicted drug is drugC



Drug Prediction App

Enter Age
32 - +

Enter Sex
Male ▾

Enter BP
High ▾

Enter Cholesterol
Normal ▾

Enter Na_to_K ratio
9.445 - +

Predict Drug

The predicted drug is drugX

Drug Prediction App

Enter Age

32

-

+

Enter Sex

Female

▼

Enter BP

Low

▼

Enter Cholesterol

High

▼

Enter Na_to_K ratio

9.712

-

+

Predict Drug

The predicted drug is drugA

Drug Prediction App

Enter Age

59

-

+

Enter Sex

Female

▼

Enter BP

Low

▼

Enter Cholesterol

High

▼

Enter Na_to_K ratio

10.444

-

+

Predict Drug

The predicted drug is drugB

EISYSTEMS SERVICES

FF-110, Express Greens Plaza, Sector 1
 Vaishali – Delhi NCR – India 201010

W: www.eisystems.in | E: info@eisystems.in | T: +91 92122-51000

Drug Prediction App

Enter Age

23

-

+

Enter Sex

Female

▼

Enter BP

Normal

▼

Enter Cholesterol

High

▼

Enter Na_to_K ratio

45.675

-

+

Predict Drug

The predicted drug is drugY

EISYSTEMS SERVICES

FF-110, Express Greens Plaza, Sector 1
Vaishali – Delhi NCR – India 201010

W: www.eisystems.in | E: info@eisystems.in | T: +91 92122-51000

References

- Jordan, Michael I., and Tom M. Mitchell. "Machine learning: Trends, perspectives, and prospects." *Science* 349, no. 6245 (2015): 255-260.
- El Naqa, Issam, and Martin J. Murphy. *What is machine learning?*. Springer International Publishing, 2015.
- Bi, Qifang, Katherine E. Goodman, Joshua Kaminsky, and Justin Lessler. "What is machine learning? A primer for the epidemiologist." *American journal of epidemiology* 188, no. 12 (2019): 2222-2239.
- Montgomery, Douglas C., Elizabeth A. Peck, and G. Geoffrey Vining. *Introduction to linear regression analysis*. John Wiley & Sons, 2021.
- Seber, George AF, and Alan J. Lee. *Linear regression analysis*. John Wiley & Sons, 2012.
- Sperandei, Sandro. "Understanding logistic regression analysis." *Biochemia medica* 24, no. 1 (2014): 12-18.
- Peduzzi, Peter, John Concato, Elizabeth Kemper, Theodore R. Holford, and Alvan R. Feinstein. "A simulation study of the number of events per variable in logistic regression analysis." *Journal of clinical epidemiology* 49, no. 12 (1996): 1373-1379.
- Sun, Shiliang, and Rongqing Huang. "An adaptive k-nearest neighbor algorithm." In *2010 seventh international conference on fuzzy systems and knowledge discovery*, vol. 1, pp. 91-94. IEEE, 2010.
- Chomboon, Kittipong, Pasapitch Chujai, Pongsakorn Teerarassamee, Kittisak Kerdprasop, and Nittaya Kerdprasop. "An empirical study of distance metrics for k-nearest neighbor algorithm." In *Proceedings of the 3rd international conference on industrial application engineering*, vol. 2, p. 4. 2015.
- Charbuty, Bahzad, and Adnan Abdulazeez. "Classification based on decision tree algorithm for machine learning." *Journal of Applied Science and Technology Trends* 2, no. 01 (2021): 20-28.
- Su, Jiang, and Harry Zhang. "A fast decision tree learning algorithm." In *Aaai*, vol. 6, pp. 500-505. 2006.
- Geeks for Geeks: <https://www.geeksforgeeks.org/decision-tree>
- Wikipedia: https://en.wikipedia.org/wiki/Decision_tree
- JavatPoint: <https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm>

Student Self Evaluation of the Short-Term Internship

Please rate your performance in the following areas:

| | | | | | |
|---|---|---|---|---|---|
| 1) Oral communication | 1 | 2 | 3 | 4 | 5 |
| 2) Written communication | 1 | 2 | 3 | 4 | 5 |
| 3) Initiative | 1 | 2 | 3 | 4 | 5 |
| 4) Interaction with staff | 1 | 2 | 3 | 4 | 5 |
| 5) Attitude | 1 | 2 | 3 | 4 | 5 |
| 6) Dependability | 1 | 2 | 3 | 4 | 5 |
| 7) Ability to learn | 1 | 2 | 3 | 4 | 5 |
| 8) Planning and organization | 1 | 2 | 3 | 4 | 5 |
| 9) Professionalism | 1 | 2 | 3 | 4 | 5 |
| 10) Creativity | 1 | 2 | 3 | 4 | 5 |
| 11) Quality of work | 1 | 2 | 3 | 4 | 5 |
| 12) Productivity | 1 | 2 | 3 | 4 | 5 |
| 13) Progress of learning | 1 | 2 | 3 | 4 | 5 |
| 14) Adaptability to organization's culture/policies | 1 | 2 | 3 | 4 | 5 |
| 15) OVERALL PERFORMANCE | 1 | 2 | 3 | 4 | 5 |

Rating Scale: 5 will be Best while 1 will be Worst



Signature of the Student

Student Self Evaluation of the Short-Term Internship

Please rate your performance in the following areas:

| | | | | | |
|---|---|---|---|---|---|
| 1) Oral communication | 1 | 2 | 3 | 4 | 5 |
| 2) Written communication | 1 | 2 | 3 | 4 | 5 |
| 3) Initiative | 1 | 2 | 3 | 4 | 5 |
| 4) Interaction with staff | 1 | 2 | 3 | 4 | 5 |
| 5) Attitude | 1 | 2 | 3 | 4 | 5 |
| 6) Dependability | 1 | 2 | 3 | 4 | 5 |
| 7) Ability to learn | 1 | 2 | 3 | 4 | 5 |
| 8) Planning and organization | 1 | 2 | 3 | 4 | 5 |
| 9) Professionalism | 1 | 2 | 3 | 4 | 5 |
| 10) Creativity | 1 | 2 | 3 | 4 | 5 |
| 11) Quality of work | 1 | 2 | 3 | 4 | 5 |
| 12) Productivity | 1 | 2 | 3 | 4 | 5 |
| 13) Progress of learning | 1 | 2 | 3 | 4 | 5 |
| 14) Adaptability to organization's culture/policies | 1 | 2 | 3 | 4 | 5 |
| 15) OVERALL PERFORMANCE | 1 | 2 | 3 | 4 | 5 |

Rating Scale: 5 will be Best while 1 will be Worst



Signature of the Student

EISYSTEMS SERVICES

FF-110, Express Greens Plaza, Sector 1
Vaishali - Delhi NCR - India 201010

W: www.eisystems.in | E: info@eisystems.in | T: +91 92122-51000

Student Self Evaluation of the Short-Term Internship

Please rate your performance in the following areas:

| | | | | | |
|---|---|---|---|---|---|
| 1) Oral communication | 1 | 2 | 3 | 4 | 5 |
| 2) Written communication | 1 | 2 | 3 | 4 | 5 |
| 3) Initiative | 1 | 2 | 3 | 4 | 5 |
| 4) Interaction with staff | 1 | 2 | 3 | 4 | 5 |
| 5) Attitude | 1 | 2 | 3 | 4 | 5 |
| 6) Dependability | 1 | 2 | 3 | 4 | 5 |
| 7) Ability to learn | 1 | 2 | 3 | 4 | 5 |
| 8) Planning and organization | 1 | 2 | 3 | 4 | 5 |
| 9) Professionalism | 1 | 2 | 3 | 4 | 5 |
| 10) Creativity | 1 | 2 | 3 | 4 | 5 |
| 11) Quality of work | 1 | 2 | 3 | 4 | 5 |
| 12) Productivity | 1 | 2 | 3 | 4 | 5 |
| 13) Progress of learning | 1 | 2 | 3 | 4 | 5 |
| 14) Adaptability to organization's culture/policies | 1 | 2 | 3 | 4 | 5 |
| 15) OVERALL PERFORMANCE | 1 | 2 | 3 | 4 | 5 |

Rating Scale: 5 will be Best while 1 will be Worst

Rohith
29/6/24

Signature of the Student

EISYSTEMS SERVICES

FF-110, Express Greens Plaza, Sector 1
Vaishali - Delhi NCR - India 201010

W: www.eisystems.in | E: info@eisystems.in | T: +91 92122-51000

Annexure 1

Daily Activity Report

Week No: 1
 (1/2/3/4/5/6/7/8)

| Day & Date | Brief Description of Daily Activity | Learning Outcome | Person In-Charge |
|--|---|--|--------------------|
| 06/05/24-Monday | Introduction to python, prerequisites', important thigs about python, features of python. | Introduction to python and its prerequisites and important things about python' | Mallika Srivastava |
| 07/05/24-Tuesday | System requirements of python, how to install python, IDE's, creating first python program. | System requirements for installation of python, and process of installation of python, knowing about various IDE's and creating first python program | Mallika Srivastava |
| 08/05/24-Wednesday | Variables, constants, conventions, print functions. | Various variable usages, constants, convections used, print functions | Mallika Srivastava |
| 09/05/24-Thursday | Sep and end, comments like Single comment, multiline comment. | Usage of sep and end, comments usage and various types of comments | Mallika Srivastava |
| 10/05/24-Friday | Data type number int datatype float datatype, complex datatype | Various datatypes like number int, float, complex | Mallika Srivastava |
| Revision/Homework 11/05/24-Saturday | practice | Practiced all above concepts | Mallika Srivastava |

Week No: 2
 (1/2/3/4/5/6/7/8)

| Day & Date | Brief Description of Daily Activity | Learning Outcome | Person In-Charge |
|--|---|---|--------------------|
| 13/05/24-Monday | String datatype, functions like upper(), index, lower | Various string functions like upper index lower and other | Mallika Srivastava |
| 14/05/24-Tuesday | Title, capitalize, lists | Some other functions of string include title capitalize and lists | Mallika Srivastava |
| 15/05/24-Wednesday | List methods | List methods like append(), extend(), insert(), remove() | Mallika Srivastava |
| 16/05/24-Thursday | Pop(), index(), count(), rename(), sort(), copy | Some other are Pop(), index(), count(), rename(), sort(), copy | Mallika Srivastava |
| 17/05/24- Friday | Tuples and its properties | Various tuple and its functions | Mallika Srivastava |
| Revision/Homework 18/05/24-Saturday | Practice | Practicing the above concepts. | Mallika Srivastava |

Week No: 3
 (1/2/3/4/5/6/7/8)

| Day & Date | Brief Description of Daily Activity | Learning Outcome | Person In-Charge |
|--|-------------------------------------|--|--------------------|
| 20/05/24-Monday | Dictionary and its properties | Various Dictionaries and its functions | Mallika Srivastava |
| 21/05/24-Tuesday | Set and its properties | Various sets and its properties | Mallika Srivastava |
| 22/05/24-Wednesday | Boolean and its properties | Boolean and its various function. | Mallika Srivastava |
| 23/05/24-Thursday | User input and typecasting | Type casting used for changing datatype of element for one datatype to other | Mallika Srivastava |
| 24/05/24-Friday | Control flow statements if, if else | Control flow statements like if, if else | Mallika Srivastava |
| Revision/Homework 25/05/24-Saturday | Practice | Practicing the above concepts | Mallika Srivastava |

Week No: 4
(1/2/3/4/5/6/7/8)

| Day & Date | Brief Description of Daily Activity | Learning Outcome | Person In-Charge |
|--|-------------------------------------|--|--------------------|
| 27/05/24-Monday | If elif- else, nested if | Continuation of control flow statements like if elif-else, nested if | Mallika Srivastava |
| 28/05/24-Tuesday | Project quiz | Doing project quiz | Mallika Srivastava |
| 29/05/24-Wednesday | Loops like for while | Introduction to loops like for and while | Mallika Srivastava |
| 30/05/24-Thursday | Functions in python | Introduction to functions in python language. Functions like inbuilt functions and user defined functions are introduced | Mallika Srivastava |
| 31/05/24-Friday | Types of user defined functions | Introducing various types of user defined functions like Default, parameterized, anonymous functions. | Mallika Srivastava |
| Revision/Homework 01/06/24-Saturday | Practice | Practicing the above concepts | Mallika Srivastava |

Week No: 5
 (1/2/3/4/5/6/7/8)

| Day & Date | Brief Description of Daily Activity | Learning Outcome | Person In-Charge |
|--|---|---|--------------------|
| 03/06/24-Monday | Lambda, global variable, local variable | Functions include various properties like lambda, local variables, and global variables. | Mallika Srivastava |
| 04/06/24-Tuesday | Package, module, library, | Packages, modules and libraries are very important aspect of the programming language. They help in building program more efficiently and accurately and make it easy for understanding. | Mallika Srivastava |
| 05/06/24-Wednesday | Object oriented programming class, object | Object oriented programming is a important aspect of programming language. it can be used to enhance efficiency and readability of the program. Initial properties include object, class. | Mallika Srivastava |
| 06/06/24-Thursday | Abstraction, inheritance | Continuation of object-oriented programming include concepts like Abstraction, inheritance | Mallika Srivastava |
| 07/06/24-Friday | Encapsulation, polymorphism | More topics include object-oriented programming are encapsulation and polymorphism | Mallika Srivastava |
| Revision/Homework 08/06/24-Saturday | Practice | Practicing the above concepts. | Mallika Srivastava |

Week No: 6
(1/2/3/4/5/6/7/8)

| Day & Date | Brief Description of Daily Activity | Learning Outcome | Person In-Charge |
|--|-------------------------------------|--|--------------------|
| 10/05/24-Monday | Constructor and Methods | Constructor is an important aspect of programming language used to write program effectively and increasing readability of the program. The types include default and parameterized. Methods help in building the structure of the code. The types include instance class, static. | Mallika Srivastava |
| 11/06/24-Tuesday | NumPy, arrange, linspace, logspace | NumPy is the first concept used in building a well-trained model. Some of its functions include arrange, linspace, logspace. | Mallika Srivastava |
| 12/06/24-Wednesday | Arrays 1D, 2D, 3D, reshape | Arrays are used as a datatype to store homogeneous elements. The size of the array varies like 1D, 2D, 3D. | Mallika Srivastava |
| 13/06/24-Thursday | Pandas dataframe | Pandas is used for the model training in the form of data frames. It allows importing of files | Mallika Srivastava |
| 14/06/24-Friday | Matplotlib | Matplotlib is used for visualization of the model. This is a pictographic representation of the data. | Mallika Srivastava |
| Revision/Homework 15/06/24-Saturday | Practice | Practicing the model building terminologies. | Mallika Srivastava |

Week No: 7
 (1/2/3/4/5/6/7/8)

| Day & Date | Brief Description of Daily Activity | Learning Outcome | Person In-Charge |
|--|--|--|--------------------|
| 17/06/24-Monday | Model demonstration | Model demonstration include final blueprint of the model. | Mallika Srivastava |
| 18/06/24-Tuesday | Model and project introduction | Model and project construction include building projects using various algorithms depending on the model usage. | Mallika Srivastava |
| 19/06/24-Wednesday | Linear regression, logistic regression | Linear and logistic regression basic model building algorithms. They are done on labeled data. | Mallika Srivastava |
| 20/06/24-Thursday | K – nearest neighbor | This is a type of classification algorithm used to find distance between its nearest similar point and can be used to find outliers. | Mallika Srivastava |
| 21/06/24-Friday | Support vector machine | Support vector machine is a classification algorithm and it is used to find a hyper plane for classification process. | Mallika Srivastava |
| Revision/Homework 22/06/24-Saturday | Decision tree | Decision tree is used when we have to make a conclusion based on more than 2 attributes. | Mallika Srivastava |

Week No: 8
 (1/2/3/4/5/6/7/8)

| Day & Date | Brief Description of Daily Activity | Learning Outcome | Person In-Charge |
|--|-------------------------------------|--|--------------------|
| 24/06/24-Monday | Model deployment | Model deployment is used to take input and give predictions as output on test data. The properties like pickle and streamlit can be used for this. | Mallika Srivastava |
| 25/06/24-Tuesday | Project making | Using above concepts building model. | Mallika Srivastava |
| 26/06/24-Wednesday | Data preprocessing | Preparing the dataset. | Mallika Srivastava |
| 27/06/24-Thursday | Train test splitting | Training testing and splitting of the data for the model | Mallika Srivastava |
| 28/06/24-Friday | Model evaluation | Evaluating the accuracy of the model. | Mallika Srivastava |
| Revision/Homework 29/06/24-Saturday | Model deployment | Deploying the model. | Mallika Srivastava |

Annexure 2

Weekly Progress Report

Week No: 1-8
(1/2/3/4/5/6/7/8)

| Week(s) | Summary of Weekly Activity |
|---------|--|
| Week 1 | Introduction to python, its features variables, constants, conventions, print statements, comments, datatypes like numeric, int float complex. |
| Week 2 | Introduction to Strings and its functionalities, list and its list methods and tuple |
| Week 3 | Dictionary set Boolean, user input typecasting, control flow statements |
| Week 4 | Loops functions various types of functions. |
| Week 5 | Lambda, global variable, local variable, package module library object oriented programming class , object, abstraction, inheritance, polymorphism, encapsulation. |
| Week 6 | Constructor methods NumPy, arrays, pandas, matplotlib. |
| Week 7 | Model demonstration, linear regression, logistic regression, k-nearest neighbor, support vector machine, decision tree |
| Week 8 | Model deployment, project making, data processing, train test split model evaluation, model deployment |