VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.



LAB REPORT on

Machine Learning (23CS6PCMAL)

Submitted by

Abhishek Gouda (1BM22CS006)

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING



B.M.S. COLLEGE OF ENGINEERING
(Autonomous Institution under VTU)
BENGALURU-560019
Sep-2024 to Jan-2025

B.M.S. College of Engineering,

Bull Temple Road, Bangalore 560019

(Affiliated To Visvesvaraya Technological University, Belgaum)

Department of Computer Science and Engineering



CERTIFICATE

This is to certify that the Lab work entitled "Machine Learning (23CS6PCMAL)" carried out by **Abhishek Gouda (1BM22CS006)**, who is Bonafide student of **B.M.S. College of Engineering.** It is in partial fulfillment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum. The Lab report has been approved as it satisfies the academic requirements in respect of an Machine Learning (23CS6PCMAL) work prescribed for the said degree.

Dr. Seema Patil	Dr. Kavitha Sooda
Assistant Professor	Professor & HOD
Department of CSE, BMSCE	Department of CSE, BMSCE

Index

Sl. No.	Date	Experiment Title	Page No.			
1	4-3-2025	Write a python program to import and export data using Pandas library functions	5-8			
2	11-3-2025	Demonstrate various data pre-processing techniques for a given dataset	9-13			
3	18-3-2025	Use an appropriate data set for building the decision tree (ID3) and apply this knowledge to classify a new sample.	14-16			
4	1-4-2025	5 Implement Linear and Multi-Linear Regression algorithm using appropriate dataset				
5	8-4-2025	Build Logistic Regression Model for a given dataset	21-23			
6	15-4-2025	Build KNN Classification model for a given dataset.	24-27			
7	15-4-2025	Build Support vector machine model for a given dataset	28-30			
8	22-4-2025	Implement Random forest ensemble method on a given dataset.	31-32			
9	22-4-2025	5 Implement Boosting ensemble method on a given dataset.				
10	29-4-2025	Build k-Means algorithm to cluster a set of data stored in a .CSV file.				
11	29-4-2025	Implement Dimensionality reduction using Principal Component Analysis (PCA) method.	40-44			

Github Link: https://github.com/Abhii2404/6thSem-ML-Lab

Write a python program to import and export data using Pandas library functions

Lab-L	solvenia de la
1) Tilling values directly into date frame.	and expect to the state of the state of the
South made at the	MID Home
total grant pulled total	0 44
Nome to white Bob , smarle , many),	i Giv 3.
- 14 gra = 1 (15 c) 11 , 25 c/m),	National Control of the Control of t
City : State de po for angelline	to Downtowing elements from , existing element seper to ale
(1 (1) (1) (1) (1) (1)	43 - Fix Med to V (Moth Oshick to 1)
I have a feel may at the good	mint (A hop fla)
of pholaborane Edubl	minh (Mining 1/2)
mich (M) and a second and a second	0
He me o Just	
what he are and the training in the	
None Recutify	id Ronk Bala doolpris
b Alice is New Yorks	learnest with the
1 Bet 20 Los Angels	impost y finance as of
Contraction of the second	import pondes or pd
Importing classicks from whiteapy dataset	import maliplotlib, poplet as plt
Thom sales elements import book 186	plickens "" HORE BANK 115", "ILI (3 BANK 115"]
of apollab Franc ("harolate " Nime air's Stature)	grint (data head ())
of Charges VI x 1812 - Market	gold (date steps)
print (of head)) 1 a market order his	print (ditariolemni)
- 2 refer D - 1 comment of the control of the control of	plt. fig (lig = (14, 15)
Small handles of the sales and	out of
Spellingth (in) Syptain (in) publication) petel molhica)	Timen KOTHBANH, NA
33 30 1,4 1,00	palee with with low the
1 7.5 3.0 0 0 0 0 1 P-V	data data
of a second relative or supplied to the second	702-01-01 1906-99 1916-201 1891-022 1902-1914-23 9
Importing dobsets from a specific con file	Shop (1, 147, 13)
THE THE THE THE PERSON IS NOT THE PERSON IS	
of agel. Bead try (flagor)	Tolum Lenks
not be to feel a be d	MATINALE (C'KOTAN BANK NS " 'Opm')
	('homkova ns', 'High),
Sand Valer and	3
	homes of Tour (pane)

```
Code:
import yfinance as yf
import pandas as pd
import matplotlib.pyplot as plt
tickers = ["HDFCBANK.NS", "ICICIBANK.NS", "KOTAKBANK.NS"]
data = yf.download(tickers, start="2024-01-01", end="2024-12-30", group by='ticker')
print("First 5 rows of the dataset:")
print(data.head())
print("\nShape of the dataset:")
print(data.shape)
print("\nColumn names:")
print(data.columns)
hdfc data = data['HDFCBANK.NS']
print("\nSummary statistics for HDFC Bank:")
print(hdfc data.describe())
hdfc_data['Daily Return'] = hdfc_data['Close'].pct_change()
icici data = data['ICICIBANK.NS']
print("\nSummary statistics for ICICI Bank:")
print(icici data.describe())
icici data['Daily Return'] = icici data['Close'].pct change()
kotak_data = data['KOTAKBANK.NS']
print("\nSummary statistics for Kotak Mahindra Bank:")
print(kotak data.describe())
```

```
kotak data['Daily Return'] = kotak data['Close'].pct change()
plt.figure(figsize=(14, 10))
plt.subplot(3, 2, 1)
hdfc data['Close'].plot(title="HDFC Bank - Closing Price")
plt.subplot(3, 2, 2)
hdfc data['Daily Return'].plot(title="HDFC Bank - Daily Returns", color='orange')
plt.subplot(3, 2, 3)
icici data['Close'].plot(title="ICICI Bank - Closing Price")
plt.subplot(3, 2, 4)
icici data['Daily Return'].plot(title="ICICI Bank - Daily Returns", color='orange')
plt.subplot(3, 2, 5)
kotak data['Close'].plot(title="Kotak Mahindra Bank - Closing Price")
plt.subplot(3, 2, 6)
kotak data['Daily Return'].plot(title="Kotak Mahindra Bank - Daily Returns", color='orange')
plt.tight layout()
plt.show()
hdfc data.to csv('hdfc bank_data.csv')
icici data.to csv('icici bank data.csv')
kotak data.to csv('kotak bank data.csv')
print("\nHDFC Bank data saved to 'hdfc bank data.csv'.")
print("ICICI Bank data saved to 'icici bank data.csv'.")
print("Kotak Bank data saved to 'kotak bank data.csv'.")
```

Demonstrate various data pre-processing techniques for a given dataset

Demonstrake various da la pre processing lachaiques for a given da la sel Pleusing (SV impost ponds as pa file-path = 'housing cev' af a parced-av (file-rotts) print l'Information about all columns:") print l'Information about all columns france Snake: 20 640 entities) ate 2003 g Bala column (total 10 columns): column on multipant aligne tolomis mon multipant aligne 1 latitude 1 latitude 2 newing medienege 5 total sooms 7 total illedecons 7 total illedecons 8 new - volve 9 oca provinity dtypes flout by dtypes flout by	Love - Data Processing 5 1/2/25	
Housings (SV impost ponds as pd File-path & 'bouring csv' df & pd. &cod - av (file-rosh) "psint ('InfoSmotion obeut all solvens'") psint (off info(3)) output: InfoSmotion about all solvens Bange Sndex: 20 640 entities about 20 bype colvens (total 10 colvens): colvens (total 10 colvens): foot colvens (total 10 colvens): lower and incore 1 lotitude 20640 non nost plates hold soons Jotal soons Jotal soons pepulation propulation non-multipath plates non-multipath plates non-multipath plates non-multipath plates non-multipath non-multipath plates non-multipath non-multipath plates non-multipath non-	Demonstrate various do la pre processing tachmiques	
imposet ponds or pd file = path = 'housing.csv' df = pd. sced = csv (file = peth) print Sufermotion about all solvers:") print Sufermotion about all solvers:") print Color color		
Af 2 pa. 2000 - av (file-poth) "" "" "" "" "" "" "" "" ""	Blausing CSV	
print (Information obsert all solvers:") Print (Information obser	1) Vile with - 'tank'	
print ("Information about all columns") print (off info(3)) puteut: Information about all columns Brange Index: 20 640 entities; 040 2063 9 Deba columns (total 10 columns): column non-null point 0 type 0 longitude 20 640 non-null floot 64 1 latitude 11 11 2 housing median-ege 12 13 3 total sound 5 papulation 12 13 6 householdes 12 14 8 hour volue 14 9 oce provinity 1 1 object	de and send and the	
print (Info (3) print (of info (3)) Prin	The state of the posts	
print (of info()) Description Surface modion about all columns Beange Sudex: 20 640 entities; 0 to 2063 9 Dela columns (total 10 columns): column non-mull rount Object O longitude 20 640 non mult blook 64 1 latitude "" 2 housing - medion ege " 3 total - sound on ege " 4 population " 5 population " 6 householdes " 7 median - intome 8 new - volue " 9 oce - provinity " 1 policet		
Surfermotion about all columns Brange Index: 20 640 entities to be 2000 9 Pada columns (total 10 columns): column non-mult poent 0 type O longitude 20 640 non-mult float 64 1 latitude "" housing median-ege not in the second of the sec		
Bonge Index = 20 640 entities = 0 to 20 63 9 Doto columns (total 10 columns) = column non-null rount 10 type 0 longitude 20 640 mon null float 64 1 latitude 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4		
Benge Index: 20 640 entities to be 2063 9 Dela columns (total 10 columns): column non-null rount Otype O longitude 20 640 non-null block 64 1 latitude 4 4 4 2 housing medienege 1 4 4 3 total bedasons 4 4 5 population 4 4 2 median in come 8 hour - volue 4 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1		
column non-hull rount Otype O longitude 20640 non null floot 64 1 lotitude "" 2 housing medienege " 3 total sound " 4 potal iledacome " 5 papulation " 6 householdes " 7 median in lone " 8 hour volue " 9 olen proximity " 1 object		
O longitude 20640 non must float 64 1 latitude "" 2 heusing medien ege " 3 total sooms " 4 population " 5 population " 6 householdes " 7 median intome " 8 hour volue " 9 ottal provinity " 1 policet	Doto columns (total 10 columns):	
1 latitude 2 heusing medienege 3 total sooms 4 hotal incleasons 5 papulation 6 householdes 7 median income 8 heus volue 9 oce provinity 1 noticet	column non-hull rount Otype	
2 housing medianope n 3 total soons 4 total bedooms 5 population 6 householdes 7 median intome 8 hous volue 9 ottal provinity 1 noticet		
5 population "" 6 householdes 2 median-intome 8 housevolue 9 occa-provinity 1 noticet	1 latitude 42 4 annix	
5 papulation 1. " 6 householdes 1. " 7 median - intome 1. " 8 house - volue 1. " 9 otto - proximity 1. " 1. object	= housing = medien-epe n "	
s population 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	3 totala soons	
householdes median intome median volue notice noti	y total - bedagons in the second	
2 median intome 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,	s population " "	
2 median intome 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,	6 householdes	
9 occampaoximity 1 " object	2 median intome For , and the	
9 occampaoximity " Discent	8 has value 1 h	
	9 ocen-proximity ! " abject	
		F
		THE PERSON NAMED IN
		15
		7110
		-

	page No.
522	print Mistal stical Information of numerical columns :")
- sist	print C'statistical Information (2)
	statistical Information of mumerands automas.
	In it to latitude housing all roles 2000 me
	20640 10640 20840 20640
	110 6 35.631 28.638 2635.40 20433
	2.131
	the sample the trade product of the second
10)	Print ("count of unique labels for ocean proximity"
	raint (df E'ocean-Broxinsity 'Javalue - countsis)
	ospul?
	21HD (een 9136 Real Section 1)
	9 NIAND 655 IN THE STATE OF THE
ui -	NEAR OCEAN 2658
	NEPR BAY 2290
	TSLAND SI
19.11	ASCHINE TO THE WAY OF COURT A
V)	point ("columns with missing values:")
	pain+ (df. is null(). sum (1 Edf. is null(). sum() >0))
	adjust "
	columns with missing values
	total-bid sooms 207
	y sulversel &
	Sanda " " Minister and P
	110 1 1 1/10 1 14

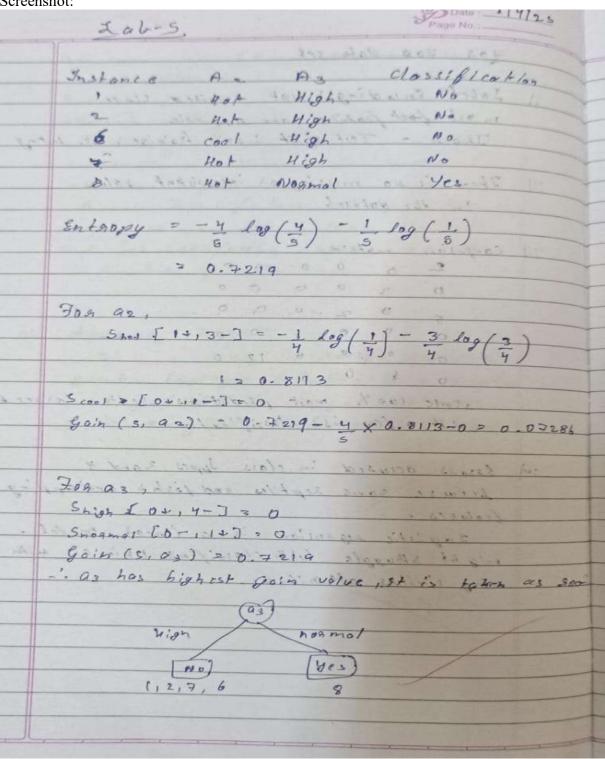
	S Cases
	Diabeles and dult Income data sets
	of the season of the season of the season of
	impost pundos de pol
	import numpy or not Dank to the Dank
	diabeter of a pola scool con ('diabetes eco')
+	adult-df = pd. scad - cx v (adult chvi)
,	H. Y
)	Missing volues
7	paint (diebetes - of ismult () sum()
	print (adult - off isnull () sunt)
1	outputt has gotton you will all the set of house to
1	Missing volumes in diobetes doloset s
	- Spein Marcaffig to fine 1010 11 11 10 To 10
-	No-Pation 0
+	ACT AND ACT OF
#	Hissing volves in Adult income dataset
1	- Colores have to see a sound
	coophelasonad or son market and an execution and
	and out some
	from & Kleann . processing impost minmon scales, standard
	imposit malliplatlib pyplot ospit
	file-path 2 diabetesics 1 2000 as a less to
	of apd. see desv (file-Path)
1	of -num . Uf select - dtypes (include · E'number 1). copy ()
1.4	'nput = simple luputer (strategy = 'mean')
a	f-num: loca [:, :] : in put - fit - transform left -numeric
do	(df-num. golumns) . Uf. numeric
	91 - db - nom. quantile (0.25)
	13 of - numagrantile (0.75)
	TOTA = BS-BI
m	in-man-scalar = min Mox scales (16)

S. S	
she was dissolver a shine of the and exclose filed to sport in	Popul Latenciae V
Show that dies States a statement of the Mande Statement of the Designation of the State Statement of the St	import parties expl
Al steel and special departs from the States his is	impeat new py as up
good (the Amount Sharet)	Som Mountainpuls import simple inputs
grind (of name of health (should be also se)	from solven processing impored were also sent on s
grid (of whated health)	shorted scatter with France dile party addle some na
pal of thems	of a ph stadent of the aprile
1 14 to talget and colones did you ideal by ?	af sepleta ("1" ne van l'exless «Then)
the despectation to line to the Control of the Cont	erm-inputs a simple Improve Catendropy a name?
	all of whit of your lindade - (186, 1877, column) +
what is the lift the minute scaling and standardigations	Intel-cocation + 13
when well you are one over the other?	for whin difrehet dyper limelate - Labort) a vicence :
when were & you will see you will be lind sever trill	le related Enterfacts
plic Has bed of thankfire	of flow late fit broughoun (of flow)
y . y . y . z	
Yam - Yele	Tabel Maried of Prat To le
and also - dalsol date not follow a normal disk	On a of grantile (and)
- leading have I'll hanges indused to be bound	33 . H. quarter (sat)
Charles galon Maniforns data to have gone recons	Tana Bijada Angelina and and and and
and notions.	min-how a Min Man insteal to
a seretime giving while up	of -11 d = pol - Data time (12d - scale - fil bronsfrom (ab))
1 AY To a second part design	slo scale Slandard Scaled
and while and out fellow a baquestan that	1/-sld - pd Salaframe (the scale fix Doniford (4))
but d subta a daile see fees in a superior and	Colours of the period belongs to depole
- many the sign assume harmality	of Hissing values one superceled by 11" which
of the state of the said	and the second second
/ The glad milly mil	we replace with elast
and the state of t	for behavioral extenses - Septente with mean
and the strength of the	for cangonial column a replace with anda
The san make a	3) worth (law , education , must let - state , econopedity , decision)
	pare isen, rather constant income
of everyon had	threading netherd table treading to become the beginning to
1811 1 1 1 1 1	Third of miles (part 1929)
(Waynester questioned)	numerical framula

```
import pandas as pd
import numpy as np
from sklearn.model selection import train test split
from sklearn.preprocessing import MinMaxScaler, StandardScaler
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import LabelEncoder
import seaborn as sns
import matplotlib.pyplot as plt
diabetes data = pd.read csv('/content/Dataset of Diabetes .csv')
adult income data = pd.read csv('/content/adult.csv')
print("Diabetes Dataset:")
print(diabetes data.head())
print("\nAdult Income Dataset:")
print(adult income data.head())
diabetes numerical cols = diabetes data.select dtypes(include=[np.number]).columns
diabetes categorical cols = diabetes data.select dtypes(include=[object]).columns
diabetes imputer num = SimpleImputer(strategy='median')
diabetes data[diabetes numerical cols] =
diabetes imputer num.fit transform(diabetes data[diabetes numerical cols])
diabetes imputer cat = SimpleImputer(strategy='most frequent')
diabetes data[diabetes categorical cols] =
diabetes imputer cat.fit transform(diabetes data[diabetes categorical cols])
adult income numerical cols = adult income data.select dtypes(include=[np.number]).columns
adult income categorical cols = adult income data.select dtypes(include=[object]).columns
adult income imputer num = SimpleImputer(strategy='median')
adult income data[adult income numerical cols] =
adult income imputer num.fit transform(adult income data[adult income numerical cols])
adult income imputer cat = SimpleImputer(strategy='most frequent')
adult income data[adult income categorical cols] =
adult income imputer cat.fit transform(adult income data[adult income categorical cols])
categorical columns adult = adult income data.select dtypes(include=['object']).columns
label encoder = LabelEncoder()
for col in categorical columns adult:
  adult income data[col] = label encoder.fit transform(adult income data[col])
```

```
def detect and remove outliers(df):
  numerical df = df.select dtypes(include=[np.number])
  Q1 = numerical df.quantile(0.25)
  Q3 = numerical df.quantile(0.75)
  IQR = Q3 - Q1
  return df[\sim((numerical df < (Q1 - 1.5 * IQR)) | (numerical df > (Q3 + 1.5 * IQR))).any(axis=1)]
diabetes data cleaned = detect and remove outliers(diabetes data)
adult income data cleaned = detect and remove outliers(adult income data)
min max scaler = MinMaxScaler()
diabetes numerical cols = diabetes data cleaned.select dtypes(include=[np.number]).columns
diabetes data normalized = diabetes data cleaned.copy()
diabetes data normalized[diabetes numerical cols] =
min max scaler.fit transform(diabetes data cleaned[diabetes numerical cols])
adult income numerical cols =
adult income data cleaned.select dtypes(include=[np.number]).columns
adult income data normalized = adult income data cleaned.copy()
adult income data normalized[adult income numerical cols] =
min max scaler.fit transform(adult income data cleaned[adult income numerical cols])
standard scaler = StandardScaler()
diabetes data standardized = diabetes data cleaned.copy()
diabetes data standardized[diabetes numerical cols] =
standard scaler.fit transform(diabetes data cleaned[diabetes numerical cols])
adult income data standardized = adult income data cleaned.copy()
adult income data standardized[adult income numerical cols] =
standard scaler.fit transform(adult income data cleaned[adult income numerical cols])
```

Use an appropriate data set for building the decision tree (ID3) and apply this knowledge to classify a new sample.



Page No.
Mc - Decision Taxe (Institute)
Sepred of the predict (x. dat)
The live his -12 had
1 1 lat colit (x-plao) 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1
112 = Perlin Tel Prades DA()
y- yeld = att. y redit (x- ks t)
The state of the state of
print (mon alsolution by ustay exacts)
Decision for classificaction for Isi
Dang : 1-0 (2)
Confusion Mokiu!
TO 0 6 7
classication report
Medision second - fl-score support
8tosa 1.00 1.00 to
viggislæ 1.60 (.00 11.1)
a command 12 20 30
mars-aug 30
weightdavd 30

	W UN DY Page No.
	Deisien kee for drug.
	Accusory: 1.0
115	Confesion making
	1660000
	50 3 0 0 0 5
	(0 0 5 50 0 7 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	(0 0 6 Drod (0 0 0)
	(C C 0 0 0 0 0 22)
	Chassiflation Agent's money
10	Friction darath of Stone & support
	Original de de la latera de latera de la latera de latera de la latera de latera de latera de la latera de la latera de latera delatera de latera de latera de latera delatera de latera de latera de latera delatera de latera de latera delatera delat
	B Maio 24 dans days 3
	2 x planet de con la forma de se
	× (1000) 2000 111
	the state of the s
	accuracy index of the 40
	brocko aug il ingland
	acigned gog 100 100 100 40
	Delision In Con Petrol commenption
	Mean Ausolute exton: 98.9
	Meon square 98108 : 17957-3
-	Nort Hoon square squared Esson: 133.6312089
	the transfer and the set have been and
	Ave to
	10 08.12 , 200 yet 10 A Action 12 1.80 of

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model selection import train test split
from sklearn.tree import DecisionTreeClassifier, DecisionTreeRegressor, plot tree
from sklearn.metrics import accuracy score, confusion matrix, classification report,
mean absolute error, mean squared error
from sklearn.preprocessing import LabelEncoder
iris = pd.read csv("/content/iris (4).csv")
drug = pd.read csv("/content/drug.csv")
petrol = pd.read csv("/content/petrol consumption.csv")
X iris = iris.iloc[:, :-1]
y iris = iris.iloc[:, -1]
X train, X test, y_train, y_test = train_test_split(X_iris, y_iris, test_size=0.2, random_state=42)
dtc = DecisionTreeClassifier()
dtc.fit(X train, y train)
y pred = dtc.predict(X test)
print("Decision Tree Classification for IRIS Dataset:")
print("Accuracy:", accuracy_score(y_test, y_pred))
print("Confusion Matrix:\n", confusion matrix(y test, y pred))
print("Classification Report:\n", classification report(y test, y pred))
X drug = drug.iloc[:, :-1]
y drug = drug.iloc[:, -1]
le = LabelEncoder()
for col in X drug.select dtypes(include=['object']).columns:
  X \text{ drug[col]} = \text{le.fit transform}(X \text{ drug[col]})
X train, X test, y train, y test = train test split(X drug, y drug, test size=0.2, random state=42)
dtc = DecisionTreeClassifier()
dtc.fit(X train, y train)
y pred = dtc.predict(X test)
print("\nDecision Tree Classification for Drug Dataset:")
print("Accuracy:", accuracy score(y test, y pred))
print("Confusion Matrix:\n", confusion matrix(y test, y pred))
print("Classification Report:\n", classification report(y test, y pred))
X \text{ petrol} = \text{petrol.iloc}[:, :-1]
```

```
y_petrol = petrol.iloc[:, -1]
X_train, X_test, y_train, y_test = train_test_split(X_petrol, y_petrol, test_size=0.2, random_state=42)
dtr = DecisionTreeRegressor()
dtr.fit(X_train, y_train)
y_pred = dtr.predict(X_test)

print("\nDecision Tree Regression for Petrol Consumption:")
print("Mean Absolute Error:", mean_absolute_error(y_test, y_pred))
print("Mean Squared Error:", mean_squared_error(y_test, y_pred))
print("Root Mean Squared Error:", np.sqrt(mean_squared_error(y_test, y_pred)))
```

Program 4

Implement Linear and Multi-Linear Regression algorithm using appropriate dataset

Jan 102 X Sicral Greater	£
and the same of th	(2 2 5 2 2 - [1.3 - 0 5] [1 , 1 1 , 1]
	100 ged (15 th)
	1 1 - 6,5 - 15, -05
plat he didn fine on	*** *** *** ***
	Annual back and a second
An administração do dos quebrados	(x x) 1 x) y = 1 0.5 0 -0.1 (2)
a land was desired but but We - IR	(-6.5 +6.1 8.1 3.3) s
The state of the s	Commence and the said that
t y Hala	• [-5,5] n - Inhasuf
7 3	and James
1 1	y = Ps + P. X + C (x = 5)
1 1	y = PO + P. X + C (x = 5)
STEP E'S YOU COME HAT AND BE	the spirit of the section of the section of
grafe was 12 a marky treps	y. (-0.5) 1(0.3)(5) + C
the second stage	4 = -0.5 + 11 = €
272 [1 4 4] - [1 4]	y=10:52 en al 1 have not an
3 2 d all 2 well	Transporting gold
Ter to 3 tons	(Course or
1 1 2 1	
47 4 10 Burney See	And have been
10 Sp. Secure and and the	
10 300	
(97x) 1 So we respect with	Charles College and the Street
	The state of the s
. 1.5 ~0.5 000	
10 to the contract of the cont	
francisco .	
1 mice	
Arms armarates I arms	
STORY TO SHILLIAM T	

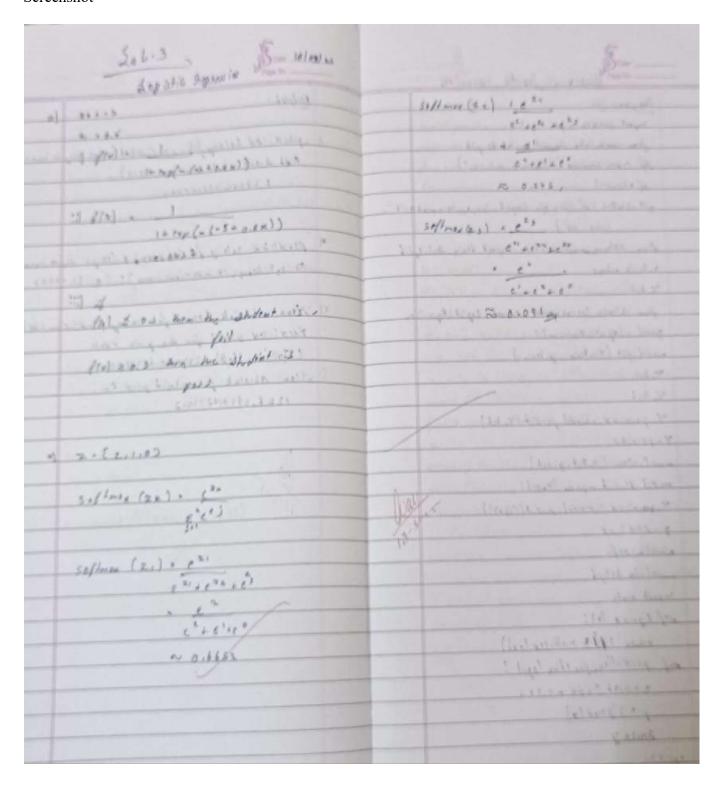
	\$		<u>\$</u>
Sab.	77		Quetous ?
Look ralaise		J	predicted for lopita intenc for
Take 17 1 1 1 1 1 1 1	-	-1	Consold in 2010; 41,017,622,576
in sail from at par			as a real transfer
A STATE OF THE PARTY OF THE PAR		_	The licked holony for 12 your of agreement
from Orlean involve to	real-heads !	_	1 150 18 0 13 7 . 59/ \$3 1 3 9 3 6 3 h
An Colore involve la	el a pl	-	
1 100 1 100	- "	-1	Hean Should Enter (140000) - 3240 39310742
Will was a State Late Late Land Company			VICEN SERVICE
plk while (Giro)		_	Hear Absolute Ennon (Salmy 1: 9519.1406 2251331)
THE RESERVE TO A STATE OF THE PARTY.			Mean Assert these county to
a code to down of	1 still 14184 - 210 14000000	_	
100	2 3 11 3 2 1		
I II a Path	cast stratura 17		Party State Lawrence Barrell
price of the paint	1000 (00)		and the second fine the sea
plas 3 M. Flill	346 11.41 11.8		1
			The Car
sig . lings most Links &	ginanii		200 405
Rig fit (Prov of price)			Logical
my prosett((32003))			
ing latific		-	the are supply services
are interest		-	1000
sig interrupt		-	
7.			
61	(112-121)		
productions squested	(1.18801)		
V			
		-	

\$	Dati	and the state of t
	Colub T	60 hut
Hellple	MAR IN	1111
The same of the sa	The same	1 pack that solony for contident the experien
injust purkes as put		feet Acent , 6 Jahrview Frese)
ingest enough as of	Will I	1 51560,105096851
port week as ap linesteredel	Williak to B. mail	1 11 11
of a poly seed all of howing plicet	The state of the s	Classes of the M
11	1	" facedicked solvey for condidate 1 laga copia
of Middle mediatill and March	D. 1-11-	D 451- 61945 10 2 62 4 610 21020) = 9 6 5 62 400
of udden of bidrown filling (\$0	1 Vo stime	No. with
of volume of bedravery	The same	of Hear Absolute Energy for valory pardiction
13		PARTICIES Wishit for the girls in falls
		TAID ICUS PARTIE GOT THE ST
my amultiple-model historialization	1)	(7/m3/da Slate) 15549 00 6128352
mg lit of day (Prins 1001) 2	(Olumns'),	often distalle for gestil prediction:
mg for or or		12 2 8 , 262 4192 33312
df.ps. 15)	4-13-1	
		2 [1.1.0]
201/-		lib.
rig in hilly !		Value of the problem
		The less
1-4- 424 (((3006, 3; 40 J))		
		12 x (2) x (1)
The state of the state of		In Party
		21/21/1
		7218.0 10

```
import pandas as pd
import numpy as np
from sklearn.model selection import train test split
from sklearn.linear model import LinearRegression
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import mean absolute error
import matplotlib.pyplot as plt
hiring data = pd.read csv('hiring.csv')
print(hiring data.head())
hiring data = hiring data.dropna()
experience mapping = {
  'one': 1, 'two': 2, 'three': 3, 'four': 4, 'five': 5, 'six': 6, 'seven': 7, 'eight': 8,
  'nine': 9, 'ten': 10, 'eleven': 11, 'twelve': 12, 'thirteen': 13, 'fourteen': 14,
hiring data['experience'] = hiring data['experience'].replace(experience mapping)
hiring data['experience'] = pd.to numeric(hiring data['experience'], errors='coerce')
if hiring data['experience'].isnull().any():
  print("Warning: There are still non-numeric values in the 'experience' column.")
  hiring data = hiring data.dropna(subset=['experience'])
X hiring = hiring data[['experience', 'test score(out of 10)', 'interview score(out of 10)']]
y hiring = hiring data['salary($)']
X train hiring, X test hiring, y train hiring, y test hiring = train test split(X hiring, y hiring,
test size=0.2, random state=42)
regressor hiring = LinearRegression()
regressor hiring.fit(X train hiring, y train hiring)
candidate 1 = \text{np.array}([[2, 9, 6]])
candidate 2 = \text{np.array}([[12, 10, 10]])
salary 1 = regressor hiring.predict(candidate 1)
salary 2 = regressor hiring.predict(candidate 2)
print(f"Predicted salary for candidate 1 (2 yr experience, 9 test score, 6 interview score):
{salary 1[0]}")
print(f"Predicted salary for candidate 2 (12 yr experience, 10 test score, 10 interview score):
{salary 2[0]}")
```

```
companies data = pd.read csv('/content/1000 Companies.csv')
print(companies data.head())
companies data = companies data.dropna()
label encoder = LabelEncoder()
companies data['State'] = label encoder.fit transform(companies data['State'])
X companies = companies data[['R&D Spend', 'Administration', 'Marketing Spend', 'State']]
y companies = companies data['Profit']
X train companies, X test companies, y train companies, y test companies =
train test split(X companies, y companies, test size=0.2, random state=42)
regressor companies = LinearRegression()
regressor companies.fit(X train companies, y train companies)
input data = np.array([[91694.48, 515841.3, 11931.24, label encoder.transform(['Florida'])[0]]])
predicted profit = regressor companies.predict(input data)
print(f"Predicted profit for the given inputs (Florida State): {predicted profit[0]}")
y pred hiring = regressor hiring.predict(X test hiring)
mae hiring = mean absolute error(y test hiring, y pred hiring)
print(f"Mean Absolute Error for Salary Prediction: {mae hiring}")
y pred companies = regressor companies.predict(X test companies)
mae companies = mean absolute error(y test_companies, y_pred_companies)
print(f"Mean Absolute Error for Profit Prediction: {mae companies}")
```

Build Logistic Regression Model for a given dataset



```
Code:
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model selection import train test split
from sklearn.preprocessing import StandardScaler
from sklearn.linear model import LogisticRegression
from sklearn.metrics import accuracy score, confusion matrix
file path = 'HR comma sep.csv'
data = pd.read csv(file path)
print(data.info())
print(data.head())
print(data.describe())
plt.figure(figsize=(8, 5))
sns.countplot(x='salary', hue='left', data=data)
plt.title('Impact of Salary on Employee Retention')
plt.xlabel('Salary')
plt.ylabel('Count')
plt.legend(title='Employee Retention', labels=['Stayed', 'Left'])
plt.show()
plt.figure(figsize=(10, 6))
sns.countplot(x='Department', hue='left', data=data)
plt.title('Impact of Department on Employee Retention')
plt.xlabel('Department')
plt.ylabel('Count')
plt.legend(title='Employee Retention', labels=['Stayed', 'Left'])
plt.xticks(rotation=45)
plt.show()
data encoded = pd.get dummies(data, columns=['salary', 'Department'], drop first=True)
print(data encoded.info())
X = data encoded.drop('left', axis=1)
y = data encoded['left']
X train, X test, y train, y test = train test split(X, y, test size=0.2, random state=42)
scaler = StandardScaler()
X train scaled = scaler.fit transform(X train)
X test scaled = scaler.transform(X test)
```

```
logreg = LogisticRegression(max_iter=1000)

logreg.fit(X_train_scaled, y_train)

y_pred = logreg.predict(X_test_scaled)

accuracy = accuracy_score(y_test, y_pred)

print(f"Accuracy of the Logistic Regression Model: {accuracy * 100:.2f}%")

cm = confusion_matrix(y_test, y_pred)

plt.figure(figsize=(6, 5))

sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', cbar=False, xticklabels=['Stayed', 'Left'],

yticklabels=['Stayed', 'Left'])

plt.title('Confusion Matrix')

plt.ylabel('Predicted')

plt.ylabel('Actual')

plt.show()
```

Build KNN Classification model for a given dataset.

Screen	shot					
	Lab-6	K-NN		.5	Deres 1/4	12
-	Person				e No.	
	AILES	Age 18	Salon	1	Jarget	
	B		0 56	3.1	yv /4	
	0 C 1018	24	12		~	
	b vaco		0 4 5 0	17.50	N	
		91	60	17	Y	49.00
		43 14	3 440	8.4	V ,	
	AF +5 +2	38	0 41 0	4.5	Y	
	×		100	1.0	2 3	
		61 41 5	2 - 5	C 1 24		
	(xe, xi) =1	(33,106)		1 - 14		
	for (18, 50) 6	d = V (x =)	47 - + (4:	8,7 2		
		e V (34.	1572 4/100	- 12	*	
		e 52.8	?!	1230)		
	P-17-31 THE STA					
	f. 2 6 2 11)	/-	106.5	ו לאדה	rail th	
	Jos (3,33)	2 1 (35, 33	1) = + (10	0 -257	Lode 2	1.
		= 46.57	F G	0 8	•	
1	105 (24,70)	= V (31-2	247 7+ (110	0 2071	<	
		1221-95	13 -	0 0	9	
	for (4.7,60) 10	e V(35-47)) & Bon1	10)4		
	graver lessen	2 40,44			3	3
	las rivariant y	1 1 2 2 2 3	2 / / 1	- 1170	Jon And	
	for (431701 x					
	the cast of the	231-041	1 2000	laga!	of the	
	for (38, 401 a	V(32-30)	+ (103-4	022	Bu L	
	- 6	0.07		40 VV.	Q 100	
	MAN TON	a. Viine	and the s			
	in particularly from					
7.4	regration die	2869 00	15 Hick	72 25	7.700.	
assigned layed or ever wind values. They was						
	1114					1/2/4
				ali		

13/1				K. Dr	Page No.
	NAM	pge	salony	Donget	distance Ro
B	(4)	23	55	N	31.91 2
D	8	71	60	414	40.44 3
	N.	38	70	A 84	60.07 6
K	5	3.1	780	Ass	2
			, 2 - Y		= (x , ex)
			- N		195 (08. 80)
1914	80	X 6 3	5, 100) 3	y o	
			18.	25 .	

A selection of the selection in the selection of the sele	John KNN S	10 to 15 15 15 15 15 15 15 15 15 15 15 15 15
desurory on but dolo : 91,80%	promote profit of the profit of the special state of the special state of the special	A securery stoof lass pole date 200.00%. Accuracy on district the date 169.40%. By playing accusary and coron asks various to value select the having highest accuracy and lawist essent sole of the having considered and entires a feet dataset. Talunce scalling courses that all feeders contacted a paterns afeet dataset. Talunce scalling courses that all feeders contacted adjusted to the contacted appearably for desirable to algorithms like them. It proved feetens with law walves from the work of professor if some form feature to have the sole of sole in the contacted and according to have the sole of sole

```
import pandas as pd
import numpy as np
from sklearn.model selection import train test split
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy score, confusion matrix, classification report
import matplotlib.pyplot as plt
import seaborn as sns
iris df = pd.read csv('/content/iris (3).csv')
print(iris df.head())
X iris = iris df.drop(columns=['species'])
y iris = iris df['species']
X train iris, X test iris, y train iris, y test iris = train test split(X iris, y iris, test size=0.2,
random state=42)
scaler = StandardScaler()
X train iris = scaler.fit transform(X train iris)
X test iris = scaler.transform(X test iris)
knn iris = KNeighborsClassifier(n neighbors=3)
knn iris.fit(X train iris, y train iris)
y pred iris = knn iris.predict(X test iris)
accuracy iris = accuracy score(y test iris, y pred iris)
print(f"Accuracy on Iris test data: {accuracy iris * 100:.2f}%")
cm iris = confusion matrix(y test iris, y pred iris)
sns.heatmap(cm_iris, annot=True, fmt="d", cmap="Blues", xticklabels=knn_iris.classes,
yticklabels=knn iris.classes )
plt.title("Confusion Matrix for Iris Dataset")
plt.xlabel('Predicted')
plt.ylabel('True')
plt.show()
print("Classification Report for Iris Dataset:")
print(classification report(y test iris, y pred iris))
diabetes df = pd.read csv('diabetes.csv')
print(diabetes df.head())
```

```
X diabetes = diabetes df.drop(columns=['Outcome'])
y diabetes = diabetes df['Outcome']
X train diabetes, X test diabetes, y train diabetes, y test diabetes = train test split(X diabetes,
y diabetes, test size=0.2, random state=42)
scaler = StandardScaler()
X train diabetes = scaler.fit transform(X train diabetes)
X test diabetes = scaler.transform(X test diabetes)
knn diabetes = KNeighborsClassifier(n neighbors=5)
knn diabetes.fit(X train diabetes, y train diabetes)
y pred diabetes = knn diabetes.predict(X test diabetes)
accuracy diabetes = accuracy score(y test diabetes, y pred diabetes)
print(f"Accuracy on Diabetes test data: {accuracy diabetes * 100:.2f}%")
cm diabetes = confusion matrix(y test diabetes, y pred diabetes)
sns.heatmap(cm_diabetes, annot=True, fmt="d", cmap="Blues", xticklabels=knn_diabetes.classes ,
yticklabels=knn diabetes.classes )
plt.title("Confusion Matrix for Diabetes Dataset")
plt.xlabel('Predicted')
plt.ylabel('True')
plt.show()
print("Classification Report for Diabetes Dataset:")
print(classification report(y test diabetes, y pred diabetes))
```

Program 7

Build Support vector machine model for a given dataset

121-7 SVM Sayario 114/20	Som
hvers blance	Dx, + 4 dx + 9 dg = 1
(11) (21) (1-1) (2-1) (4.01 (511) (51-1) (610)	4d, + (dr + 9d) 2 1
7 was the site of	9 d, s 9 d 2 + 12 d 3 2 - 1
Le court forth to the sour tours	d1 +1314 d2 + 13/4
1 23 %	124/2
	w = E x; 5;
$S_{1} = \begin{pmatrix} e \\ 1 \end{pmatrix}$ $S_{2} = \begin{pmatrix} e \\ 1 \end{pmatrix}$ $S_{3} = \begin{pmatrix} e \\ 1 \end{pmatrix}$ $S_{4} = \begin{pmatrix} e \\ 1 \end{pmatrix}$ $S_{5} = \begin{pmatrix} e \\ 1 \end{pmatrix}$	= 13 (2) 13 (2) 1 (-2) (4)
St. > (£) St (£)	e/1
53 • (4)	-3
d13, 3, 2 de 3, 56 7 do 5, 56 10. 24 10.	w = / t)
a. 55 p. d. 55 55 + d. 56 55 . 195	1-2-3 1-43-20
#1 55 5, + d2 \$2 52 + d3 53 53 5-1	plints (1) verticle line 111 to your
4 (1 (1) 2 2 (2) (2) 2 2 1 (7) (2) 2 3	at x : 3
(1)(1) (1)(1) (1)(1)	
21/2 (1) + 20(0)/2) + 23/4/4) = -1	81
17 17 17111 (17 17	

```
import pandas as pd
from sklearn.model selection import train_test_split
from sklearn.svm import SVC
from sklearn.metrics import accuracy score, confusion matrix, roc auc score, roc curve
from sklearn.preprocessing import LabelEncoder, label binarize
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
df = pd.read csv("/content/letter-recognition.csv")
top classes = df['letter'].value counts().head(5).index.tolist()
df = df[df]'letter'].isin(top classes)]
X = df.iloc[:, 1:]
y = df.iloc[:, 0]
label encoder = LabelEncoder()
y encoded = label encoder.fit transform(y)
y bin = label binarize(y encoded, classes=np.unique(y encoded))
n classes = y bin.shape[1]
X train, X test, y train, y test bin = train test split(X, y bin, test size=0.2, random state=42)
svm model = SVC(kernel='linear', probability=True)
svm_model.fit(X_train, y_train.argmax(axis=1))
y score = svm model.predict proba(X test)
y pred = svm model.predict(X test)
y true = y test bin.argmax(axis=1)
print("Accuracy:", accuracy_score(y_true, y_pred))
print("Confusion Matrix:\n", confusion matrix(y true, y pred))
plt.figure()
for i in range(n classes):
  fpr, tpr, _ = roc_curve(y_test_bin[:, i], y_score[:, i])
  auc = roc_auc_score(y_test_bin[:, i], y_score[:, i])
  plt.plot(fpr, tpr, label=f"{label encoder.inverse transform([i])[0]} AUC={auc:.2f}")
plt.plot([0, 1], [0, 1], 'k--')
plt.title("ROC Curve (Top 5 Classes)")
plt.xlabel("False Positive Rate")
plt.ylabel("True Positive Rate")
plt.legend(loc="lower right")
plt.tight layout()
```

```
plt.show()
macro_auc = roc_auc_score(y_test_bin, y_score, average="macro")
print("Macro AUC Score:", macro_auc)
```

Implement Random forest ensemble method on a given dataset.

	204-8	\$ Date _15/84/2.5			
19	get the difference in	destruction of			
(0.5)	forest classifier.	decision tree and random			
-7	Acision tare	Barte			
	- A decision tree is a single	Rondom fosist			
	tree attacker where				
	decisions are made by	method that build's multiple			
	Speiting the dolaset at	nesults			
	each node				
	· These one highly prone to	These oreduce overfitting by antaging multiple decision thees which			
	overfitting, especially with	generally improves performance			
	templex defeates	- A models land to have low			
	- A model has high vasiance	vasiants and moderate bias as they			
	end las vias.	combain the dardictions from			
LEV.	the same and the same of	many hous			
-	G. S. A. S. C. C. C. C.				
8	Discuss all the parameters	of hundomforest (dossified			
4	n-estimatons: The no of l	rees in the loses t			
		measure the quality of asplit			
	max-depth : The max depth				
	-min sample split - The min no of sample sequired to split on				
	internal nede				
T	- min samples - 1 cof! The min no of samples sequined to				
	be, featured in Real nodo				
	- max, features: The no of features to consider				
	when looking for the lest split				
-	- buolskrap & whether he use brotstrapping when				
	chealing frees				
-	DOUS COSC ! Whether to use	out of tog complex to estimate			
	he generalization as accuracy	y ·			
د	nojoks: The na of jobs to	nun in proholled for both			
	1:41) and predict1)				
		he son domner of the estimates			
	The state of the s	the same of the sa			

B) Algorithm of Random lovest Skyl! The data set is divided into input features () and the torget louds (y) sky 2. Jos each beer trak a bootstrop sample brom the doloset our > . Tox cock lane, gordomly skleet a subset of features . It is deter mined by more flatures porometre style Build a doision but on the book strapped dotoset using the Delected subset of beauties shipsin once all frees can built make prediction by aggregating the paid actions of all the kings such Driving broining, the data points that were not model's generalization per los mance shet! The final model is evalu valed using mellices like accorded 1 confusion motsix and AVC Score depending on the task at hond, a & The way death if the tree

```
import pandas as pd
from sklearn.model selection import train test split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy score, confusion matrix
from sklearn import preprocessing
df = pd.read csv('/content/train.csv')
X = df.iloc[:, :-1]
y = df.iloc[:, -1]
for column in X.columns:
  if X[column].dtype == 'object':
   le = preprocessing.LabelEncoder()
   X[column] = le.fit transform(X[column])
if y.dtype == 'object':
 le = preprocessing.LabelEncoder()
 y = le.fit transform(y)
X train, X test, y train, y test = train test split(X, y, test size=0.2, random state=42)
rf classifier = RandomForestClassifier(random state=42)
rf classifier.fit(X train, y train)
y pred = rf classifier.predict(X test)
accuracy = accuracy_score(y_test, y_pred)
conf matrix = confusion matrix(y test, y pred)
print(f"Accuracy: {accuracy}")
print(f"Confusion Matrix:\n{conf matrix}")
```

Program 9

Implement Boosting ensemble method on a given dataset.

Screenshot

Later : Proposet	5- mmu
D) Selhot is Boarding:	
a Dook ling is an in it	
town being is an enemable frames to	og Mindigue Hout
Training It was by having mas	farm a strong
tuck trying he correct our essue	en experiency
merica ones	made by Me
A discuss at the possession of Adaba	est desirinte
of mater! The love often dat de la d	logity,
- nextindor . The wan so of a Kins let	s do be used
- Iconning role whalinks the contribute	on of each dossites
- also then The booting algorithm . a	etions for SAMME
and sommer typically performe	better of it was
generalities	
- non down bloke controls the nondom	humber generales for
2. 6. pilo duct it ility.	
- hore estimates : previously used to	Specify the
track terrer	and a store
The state of the s	Street Land
D. White an algerith of Ada Boost.	and the same of the same
- sept : In tidige the weights for all	training instances
	de contract to
olye! For each iteration & from	1673
Steps. Their a wrok (warris botton)	
with weighted samples. I want	
sky 42 compass the earth rate: eno	
sugar compact models weight of the	
w-10 w-1 - ex (-1-+ · y-1 ·	
sky r' Normaliza the weights so the	
she s: Herlasian (Zont + h+(N)	5
	1-1 1-27-120-120-120-120-120-120-120-120-120-120

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.datasets import load iris
from sklearn.ensemble import AdaBoostClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.linear model import LogisticRegression
from sklearn.model selection import train test split
from sklearn.metrics import accuracy score
iris = load iris()
X = iris.data
y = iris.target
X train, X test, y train, y test = train test split(X, y, test size=0.2, random state=42)
results = []
n estimators list = [10, 50, 100]
learning rates = [0.01, 0.1, 1]
for n in n estimators list:
  for lr in learning rates:
     tree base = DecisionTreeClassifier(max depth=1)
     model = AdaBoostClassifier(estimator=tree base, n estimators=n, learning rate=lr,
random state=42)
     model.fit(X train, y train)
     y pred = model.predict(X test)
     acc = accuracy_score(y_test, y_pred)
     results.append({
       'Base': 'DecisionTree',
       'n estimators': n,
       'learning rate': lr,
       'Accuracy': acc
     })
for n in n estimators list:
  for lr in learning rates:
     log reg base = LogisticRegression(max iter=1000)
     model = AdaBoostClassifier(estimator=log reg base, n estimators=n, learning rate=lr,
random state=42)
     model.fit(X train, y train)
     y pred = model.predict(X test)
     acc = accuracy score(y test, y pred)
     results.append({
       'Base': 'LogisticRegression',
```

```
'n_estimators': n,
    'learning_rate': lr,
    'Accuracy': acc
})

results_df = pd.DataFrame(results)
print(results_df)

import seaborn as sns
plt.figure(figsize=(12, 6))
sns.barplot(x='n_estimators', y='Accuracy', hue='Base', data=results_df, ci=None)
plt.title('AdaBoost Accuracy with Different Estimators and n_estimators')
plt.show()
```

Program 10

Build k-Means algorithm to cluster a set of data stored in a .CSV file.

Screenshot

	E
	Lot-10 K-means Page No.:
	Niek on Alpasition of K mions?
	spect the number to decide the no of clusters
	=) soled grandom & proints on centraids
	>) Assign each data points to their closest controid
	which will form the productioned to clusters
134	4) tallulake the variance and place a new entracid
	of each dustre
	5) Repeat the Moild steps restich moons george each
	date point do new closest censuid of each cluster.
	5) If any reassignment occurs then go to skp-4 als go
	Jo finish
	7) The model is neady
	The state of the s
	How so desmine no of clusters?
	Ellow method Silhoutic store
	Jop shakistic
	Jop Statisfic
	What is she formula of sum of squared espess ?
- 61	
	Blot SSE VS no of clusurs.
7	SSE 2 E 1/21-Ux1)2
	The plot of SSE VS no of cluster
	17 Bun the process olgerithm for a hange of cluster volves
	es tongue the SSE for each no of cluses
	of Mal SIE us no of Clusters 4) Calculate the variance and place a new contraid of
	coch theres
	5) Pycat Hesby 3 rubish means massign coch datappial
	to me new closest centred of rath cluster
	of If any seassign ment orward, there go to sky y else gate
	Finish.

	Page No.
	Describe Elbow Jechniquel
when	1) Run K. mount for singe of volves of h
	ii) compute the within cluster sum of squarts for
Saint	each values of h
1000	: !! Plot the ux ss against the no of dus ho (h)
1	illow for the chow point - the value of to oth which the
	Don of decrease in was slows significally
des	The state of the s
	Discuss all paramens used in unears of
	il n-clusus: no d clustors to form
	il h-init: no of Finns the th-mous olgosithm will be
	aun without different leathoid Breds
	"I max-iks: max no of illrations of the K-means.
	VI dol : Pela five toles once with signeds do in estile to
	decleur convergence
	vil handom-station that's the argues to ineating to
	defless convergence
	VIII Algorithm - 4 mount implementation
	viil ven bose senables ventose output for delugging
	pur posel
a street	

			N. S.	Al .					\$	
a dust eight sout with large seprending landing						Sha ciusheo a				
		(1)14 , 61 (80				tensa del	lake and a	Griner.	11.4	
Ala.	1 196/51	V . A2(1/2)	14. (4.2)			1 7 7 7	-4+4111	1 65 11		
		1200 00				Tot ches		(100)	(5/2) /2)	
+ tolant	les do	the kale Way	ries de	1, (0) Ing		1=(13	3-11-17	T F HO / I	1 (512) /2]	
6/2/						3	+	1 temp	Chr of Jan	etue-ol
P. (+)	n 1 /20 -	×17 + 136 + 51	-/1-Z/ :	110-18 =0	Swill	r-02		200	100	1.4
		/ A. (2910)				giren Pls	2016	Det pin		
		21 718-101					L.			Till belog
		s blo Aplanto				And Entry)	5	Ćv.	(3	de closes
		上 1/6-21 增制				AL [=15]	.5 .	8	17	,Ct
		the Mideries &		ide from		A3(814)	12	3	L	£3
		of Believes	Till the			P. (5.1)	5	3	7	(,
		A Fig. 1		25		As (1/3)			1	Tr.
trobat	a - 0 (About 6				As (114)	10	1	7	(1
1		whit A famil		mot	-		10		5	Ce
					-	A+(1.4)	q	9	2	(3
		mains he some			-	Fy(1,4)	3	5	8	CI
		Land Market		1			14	EA	- 0	
	1	in Admiration		1	Fire	Center · (1)	44/12/1/4	13/14/2	3.9.5)	
Artares	(1			W	John	Consus (1	Pasport	1/4 /642	+524)/4)	- les
£1(11/0)	0	5	9	12 (1)	20110	Cenus - Je	util. 1	is)Al .	(15) 331	
£2(1/3)	5	1 1	4	-6					112	
P3/114)	/t-	1 7	19	()		/		113	10 .	0.013
Ax (512)	3	b	10	/s	1		E		11.01	
As [5.5]	10	5	0/	Cs			3			
or tend	11	3	1/4	1		11	.,		1/2	
A) Cha)	9	14	0	6)						100
Ax (+ +)		10		r ci						14.4
471 277			10	14						
						La Prod		141 111		

Stration - 03 Sinta NS Distract Dist from 9 ist Point Monge An (1,10) 125 - 9,00 - 9 An (2008 S.3 - 4.75 - 2 - 02 An (2008 S.3 - 4.75 - 2 - 02 An (2008 S.3 - 4.75 - 2 - 02 An (2008 S.4 - 02 An (2008 S.4 - 02 An (1,10) 9.5 - 1.75 - 2 An (1,10) 1.75 - 1.75 - 1.75 - 1.75 An (1,10) 1.75 - 1.75 - 1.75 - 1.75 An (1,10) 1.75 - 1.75 - 1.75 An (1,10) 1.75 - 1.75 - 1.75 An (1,10) 1.75 - 1.75 An (1		-				Spale	
Give No. Printer Print from gist Print Hooge from the Claster from a fro					ì	Page No.	
Give No. Printer Print from gist Print Hooge from the Claster from a fro			- 07			and the second	
Given Pist Friends (2) 1000 (1) 15 1000 (1) 1000 (1) 15 1000 (1) 1000 (1) 15 1000 (1) 1000 (1) 10		5 4 so tion -	03	0111		Point belongs	
		given Pls		Prot from			
A(1) 10 1 5 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1							
Price S. S. S. S. S. S. S. S		A1(1)10)	123 2 2	(4:02)			
Palsial 10.3 2.33 2 C2 Palsial 8.6 1.25 8 C2 Palsial 8.6 1.25 8 C2 Palsial 9.5 8.35 2 C2 Palsial 9.5 8.35 2 C3 Palsial 9.5 8.35 2 C3 Palsial (2 15 24) 12 2 3.66 1 (10 2 8.9) /3 - 9 Porci (2 15) 12 21.5 (3.3) Phanes (2 25) 12 21.5 (3.3) Palsial (2 25) 12 21.5 (3.3) Palsial (2 25) 2 C3 Palsial Point Po	(==	P. (205)	5.3 -1	4.75			
Autorial Pist Diet Diet Bonco Donies Privation 1.56 10.62 8 C) Privation				0.33			1 117
28(2,3) 8 4 0136 5 5 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6				4,75			
Priling 9.5 8.35 2 CS Priling 9.5 8.5 8.35 2 CS Priling 9.5 8.5 8.5 2 CS Priling 9.5 8.5 8.5 8.5 8.5 8.5 8.5 8.5 8.5 8.5 8	- 1.1		15. N.	0. 14	21.0		
Production (122) 9.5 8.75 8.75 8 C. 1 Production (122+6) 12 2 3 66 1 1 02 829/3 - 9 Plenation (122+6) 12 2 3 (19 254) 12 14.33 Production (122+6) 12 2 3 (19 254) 12 14.33 Production (122+6) 12 2 3 (19 254) 12 14.33 Production (122+6) 12 2 3 (19 254) 12 14.33 Production (122+6) 12 2 3 4 (23 254) 12 14.33 Production (122+6) 12 2 2 3 4 (23 254) 12					5 17		
Proce (200) 1.5 (100 800)/3 - 9 Dence (200) 12-21-5 (3-5) Plention of clashes are (3-66) 9 (7,7-35) (105,3.5) Proce (200) 1.56 (3.6) Proce (200) 1.66 (3.6) Proce (200) 1.66 (3.6) Proces (200) 1.66 (3.6) Proce (3.6) Pro					20	(C) 14	
Desce The ((2 15 x 4) 12 = 3.66 i (1 0 x 8 x 9)/3 - 9 Desce (3 x 3 x 6) 72 x 7 (9 x 5 x 4) 13 x 14.32 Desce (2 x 3) 12 21.5 (3 - 3 Desce (2 x 3) 12 21.5					8 1	(6)	
Desce (1231) 12-21-5 (3-3) Denies (231) 12-21-5 (3-3) Persolion ** Given points Pist Diel Points Denies Poi		p. (4, a)	121.3		4.1	10-17-4	
Derice (713 + 6) /2 = 3 ((4 5 5 4) /3) Derice (231) 12-21-5 (3-3) Percolion ** Given points Pist Diel Points Derice Points Derice (231) 12-21-5 (3-3) Derice Points Derice Points			The second		/	1 - 9	
Derice (713 + 6) /2 = 3 ((4 5 5 4) /3) Derice (231) 12-21-5 (3-3) Percolion ** Given points Pist Diel Points Derice Points Derice (231) 12-21-5 (3-3) Derice Points Derice Points	baci	The Ca 15	241/2 2	3.60 1 (1	0+ 8+461	0.3 (. 2) 3	
Denis (231) 1221.5 (333 General Pict Diet Diet Donce Donce Di(210) 1.66 10.63 \$ C1 Di(210) 2.66 5.67 2 C2 Di(31) 2.34 6.32 2 C2 Di(5.1) 2.34 6.62 \$ C1 Di(5.1) 2.34 6.67 7 C2 Di(6.4) 2.34 6.33 6 C2	0.00	(31)	16) 13 2:	(91)	14) 15	. 52	
9 teration " Given points Prist Diel Promes	0	(231)	12-21-5	13-3	101		
9 teration * Given points Prist Dones fronts Dones fron	162,63			1 6 4	I I STATE		
Gira points Piet Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Domes Dome			9	2	3	1 (0 1/2/2)	
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	Herati	249	1	0.1	nut.	roigh	
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$		given pri	oli Best			The state of the state of	-
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$			bone	Money	Table 1		
P.(2.15) P.(3.14) P.(5.17) P.(6.17) P.(6.1		D1(210)			-		
P3(3,4) 7.34 (.3) Q (2 Pm(5,1) 2.34 5.62 8 C1 P3(2,5) 2.34 5.67 7 C2 P3(8,4) 2.34 (.33 6 C2 P3(112) 9,66 8.33 2 C3 P3(4,9) 0.37 2.66 8 C1 P3(4,9) 0.37 2.66 8 C1		P.(2,3)	1 3.66	5.67	2	()	
PM(5.1) 2.39 5.62 8 C1 PS(7,3) 2.39 5.67 7 C2 PS(8,4) 2.34 (.33 6 C2 PS(112) 9.66 8.33 2 C3 PS(4,9) 0.31 2.66 3 C1 PS(4,9) 0.31 2.66 3 C1						C2	
PS(2,5) 2.34 8.67 7 62 PS(2,5) 2.34 (.33 6 62 PS(112) 9,66 8.33 2 63 PS(4,9) 0.37 2.66 8 61 9 (7,725) (15,35) To center of clasters are (3.86,19) (7,725) (15,35)					8	(2)	
106 (6,4) 2.54 (.33 6 C2 A3 (112) 9.66 8.33 2 C3 A5 (4.9) 0.37 2.66 3 C1 A5 (4.9) 0.37 2.66 3 C1 To Center of Clusters are (3.86.19) (7.7.35) (1.5.35)					7	C2	
Pa(112) 9,66 6.33 2 (3 Pa(112) 9,66 3 C1 Pa(4,9) 0.31 2.66 3 C1 Pa(4,9) 0.31 2.66 3 C1						Cz	
-> (cohe of clusters are (3066, 9) (7,7,35) (15,35)		DG (B, 4)					
T (enter of clusters are (3086, 9) (7,7,35) (115, 35)		As(IIV	9,66	6.33			
7 Centre of Clusters are (3086, 9) (7,7.35) (1,5,3.5)			55 7% M	1 2.66	1 3	1	
		PSILI			/	10 10 10 10	
		- Carlo	of clarke	S AM (3	86,91	(7,4.35)(1.5	13.5)
Custr		7 LENE					
(45)						1	
					U	145/2	

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.model selection import train test split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy score
data = {
  'Name': [f'Person \{i+1\}' for i in range(50)],
  'Age': np.random.randint(18, 70, size=50),
  'Income': np.random.randint(20000, 120000, size=50)
df = pd.DataFrame(data)
df.to csv('income.csv', index=False)
df = pd.read csv('income.csv')
X = df[['Age', 'Income']]
scaler = StandardScaler()
X scaled = scaler.fit transform(X)
X train, X test = train test split(X scaled, test size=0.2, random state=42)
sse = []
k range = range(1, 11)
for k in k range:
  kmeans = KMeans(n clusters=k, random state=42)
  kmeans.fit(X train)
  sse.append(kmeans.inertia)
plt.plot(k range, sse, marker='o')
plt.title('SSE vs Number of Clusters')
plt.xlabel('Number of Clusters')
plt.ylabel('Sum of Squared Errors (SSE)')
plt.show()
optimal k = 3
kmeans = KMeans(n clusters=optimal k, random state=42)
kmeans.fit(X train)
y pred = kmeans.predict(X test)
print(f'Predicted Clusters for Test Data: {y pred}')
```

Program 11

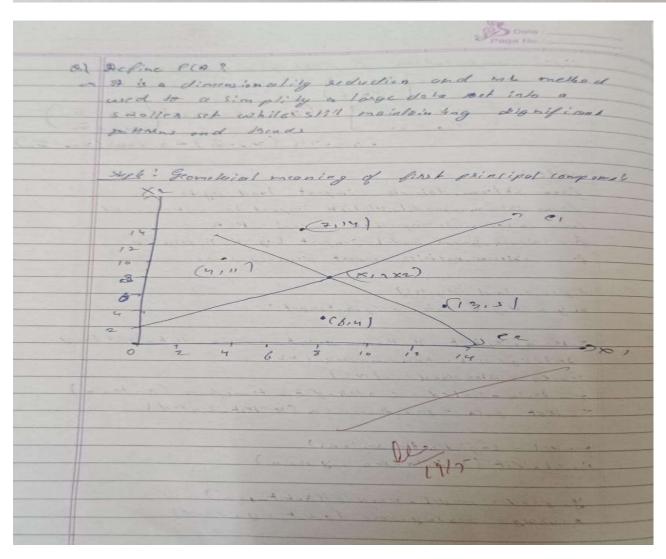
Implement Dimensionality reduction using Principal Component Analysis (PCA) method.

Screenshot

Solut PIA and Di Proprie	\$
i) colculate menn	(av (v, v) = (av (x, v))
a colonation of revosionce mobile	2-11
of collaborated Eigenvalue of the coval back making	1,11
a computation of the eigenvectors queil eigen vectors	(ov (V(1)) . 1 5 (x 5)
si computation of first grincipal component	Major
1) geometrial meaning of list principle companols	3 ((1)-1.) (+(4)-5) (5-1.) (4-1.) (4-1.)
Read to the second the disconstant from	2 23
Glora the data in table reduce the direction from	And the second
2 to 1 using the principle compared analysis	= The lova sience matrix is
algorithm	Sa Standard
Deallist margle 1 2 3 4	(5) [(60 (x, 1x, 1) (60 (x, 1, x, 2)) (60 (x, 1, x, 2))
N, 4 8 13 7	(00 (X13X1) (00 (X13X1))
X2 11 4 5 14	· [[14 -15]
Constitution of the Consti	11 73
skpl: cololek meen	52/31
V, 241811133 28	0 = det (5- AI)
4.,	> [14-> -11 -11 23->)
X= = 11 +4+5+1H = 8.5	
4	2 (14-2) (25-2) -(-11) + (-11)
The state of the s	E X2 -24 > 4201
sheet colculate the covasioner mattin	The second secon
N 2	solving the charactestics rejection we get
(1 1 5 (n - 7))2	> 1 (>> Vas)
(0)(V, x) + 1 5 (N/H - 2)2	7
3 (14-8) 24 (9-8) 24 (13-8) 34 (7-8) 4)	2 30.3849 , 6.6131 17
3	(), () () () ()
	· X1 / X2 (304)
2 14	
(xxx - xx) (xxx - xx)	Shout computiation of the rigen unton
NT ker year	705 1
100(V1,V1) = 1 = (2, x - \(\frac{1}{2}\) (\(\frac{1}{2}\) \(\frac{1}\) \(\frac{1}{2}\) \(\frac{1}{2}\) \(\frac{1}{2}\) \(\fr	V (u,)

(4->1) a1 + 114200 Loking tis only neal number Laking += 1 (14,11 2 Vp1 2+ (14-11)2 2 /112 + (14-30-3849) 2) e' = | 10 | 114,71 er 2 10.3574 - 0.4303 2) by computing similar steps we get er 2 0.8303 1 0.5574

Page No.:
Stys! Computation of first principal component
PT (MIM = 7,] = (0.5)74-0.83 / MIN-4.)
20.5574 (21x-x1) - 6.8303(224-X1



```
import pandas as pd
import numpy as np
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.model selection import train test split
from sklearn.linear model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import SVC
from sklearn.metrics import accuracy score
from sklearn.decomposition import PCA
from scipy import stats
df = pd.read csv('heart (2).csv')
z scores = np.abs(stats.zscore(df.select dtypes(include=[np.number])))
df no outliers = df[(z \text{ scores} < 3).all(axis=1)]
df cleaned = df no outliers.copy()
for col in df cleaned.select dtypes(include='object').columns:
  df cleaned[col] = LabelEncoder().fit transform(df cleaned[col])
X = df cleaned.drop('HeartDisease', axis=1)
y = df cleaned['HeartDisease']
scaler = StandardScaler()
X scaled = scaler.fit transform(X)
X train, X test, y train, y test = train test split(X scaled, y, test size=0.2, random state=42,
stratify=y)
models = {
  "Logistic Regression": Logistic Regression (max iter=1000),
  "Random Forest": RandomForestClassifier(),
  "SVM": SVC()
}
print("Accuracy without PCA:")
for name, model in models.items():
  model.fit(X train, y train)
  y pred = model.predict(X test)
  acc = accuracy score(y test, y pred)
  print(f"{name}: {acc:.4f}")
pca = PCA(n components=5)
X pca = pca.fit transform(X scaled)
X train pca, X test pca, y train, y test = train test split(X pca, y, test size=0.2, random state=42,
stratify=y)
```

```
print("\nAccuracy with PCA:")
for name, model in models.items():
    model.fit(X_train_pca, y_train)
    y_pred = model.predict(X_test_pca)
    acc = accuracy_score(y_test, y_pred)
    print(f"{name}: {acc:.4f}")
```