B G S INSTITUTE OF TECHNOLOGY

B G Nagar – 571448.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

LAB COURSE FILE & MANUAL

Academic Year	:	2020 – 2021 (ODD SEMESTER)
Programme (UG/PG)	:	UG
Year / Semester	:	4th Year / 7th Semester
Course Code	:	17CSL76
Course Title	:	MACHINE LEARNING LABORATORY

Prepared By:

Mr. M J Prasanna Kumar

Assistant Professor,

Department of CS&E

B G S Institute of Technology

B. K. Raghav

HOD

Dept. of Computer Science & Engg.

B. G. S. Institute of Technology

B G Nagar - 571 448

Nagamangala Tq. Mandya Dist

Karnataka (INDIA)

B G S INSTITUTE OF TECHNOLOGY

B G Nagar – 571448.

DEPARTMENT
OF
COMPUTER SCIENCE AND ENGINEERING

LAB COURSE FILE & MANUAL

Academic Year	:	2020 – 2021 (ODD SEMESTER)
Programme (UG/PG)	:	UG
Year / Semester	:	4th Year / 7th Semester
Course Code	:	17CSL76
Course Title	:	MACHINE LEARNING LABORATORY

Prepared By:
Mr. M J Prasanna Kumar
Assistant Professor,
Department of CS&E
B G S Institute of Technology

B G S INSTITUTE OF TECHNOLOGY

VISION

BGSIT is commited to the cause of creating tomorrow's engineers by providing quality education inculcating ethical values.

MISSION

M1: Imparting quality technical education by nurturing a conducive learning environment.

M2: Offering professional training to meet industry requirements.

M3: Providing education with a moral-cultural base and spiritual touch.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VISION

To produce engineers by possessing good technical knowledge and ethics through quality education and research.

MISSION

M1: Achieve excellence by providing good infrastructure and competent faculty.

M2: Strengthening the technical, soft skills, leadership qualities and ethical values to meet the industry requirements.

M3: Facilitate experimental learning through research projects.

PROGRAM OUTCOMES (POs)

Engineering Graduates will be able to:

1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

INSTRUCTIONS TO STUDENTS

Computer Lab Safety Rules for Protecting Equipment

- Turn off the machine once you are done using it.
- Do not plug in external devices without scanning them for computer viruses.
- Try not to touch any of the circuit boards and power sockets when a device is connected to them and switched on.
- Always maintain an extra copy of all your important data files.

General Safety Guidelines to be followed at all times

- All users of the laboratory are to follow the directions of Academic/Laboratory Technician staff member.
- Students should not attempt to repair, open, tamper or interfere with any of the computer, printing, cabling, air conditioning or other equipment in the laboratory.
- Students should be aware of office ergonomic guidelines for correct posture when using computer equipment.
- Please treat fellow users of the laboratory, and all equipment within the laboratory, with the appropriate level of care and respect.

DO's AND DON'TS

Do's

- Enter the log register
- Follow the dress code and ware ID card
- Always keep quiet. Be considerate to other lab users.
- Report any problems with the computer to the person in charge.
- Shut down the computer properly and keep the chairs aligned before leaving the lab.
- Know the location of the fire extinguisher and the first aid box and how to use them in case of an emergency.
- Report any broken plugs or exposed electrical wires to your lecturer/laboratory technician immediately.

Don'ts

- Do not eat or drink in the laboratory.
- Do not use mobile phone.
- Don't damage, remove, or disconnect any labels, parts, cables or equipment.
- Avoid stepping on electrical wires or any other computer cables.
- Do not install or download any software or modify or delete any system files on any lab computers.
- If you leave the lab, do not leave your personal belongings unattended.
- Do not open the system unit casing or monitor casing particularly when the power is turned on.
- Do not insert metal objects such as clips, pins and needles into the computer casings. They may cause fire.

LABORATORY RUBRICS

1. **FOR 25 MARKS (2010 SCHEME)**

Sl. No.	Description	Marks
1	**Continuous Evaluation**	**15**
	a. Observation write up and punctuality	2.5
	b. Conduction of experiment and output	5.0
	c. Viva voice	2.5
	d. Record write up	5.0
2	**Internal Test**	**10**

2. **FOR 20 MARKS (2015 CBCS SCHEME)**

Sl. No.	Description	Marks
1	**Continuous Evaluation**	**12**
	a. Observation write up and punctuality	2.0
	b. Conduction of experiment and output	4.0
	c. Viva voice	2.0
	d. Record write up	4.0
2	**Internal Test**	**08**

3. **FOR 40 MARKS (2017 REVISED CBCS SCHEME)**

Sl. No.	Description	Marks
1	**Continuous Evaluation**	**30**
	a. Observation write up and punctuality	5.0
	b. Conduction of experiment and output	10.0
	c. Viva voice	5.0
	d. Record write up	10.0
2	**Internal Test**	**10**

11. **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

PEO 1: Graduates will be pursuing successful career and higher education.

PEO 2: Graduates will be able to apply the knowledge of programming skills to solve the real-world problems.

PEO 3: Graduates will display professional ethics to work in a team and lead the team by effectively communicating the ideas.

PEO 4: Graduates will practice lifelong learning.

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO 1: Ability to apply Mathematical Methodologies, Management Principles and Ethics, Electronics and Embedded Systems and Programming Technologies to solve real time problems.

PSO 2: Ability to apply software design and development practices to develop software in emerging areas such as Internet of Things, Data Management, Social Networking and Security, Cloud and High-Performance Computing.

"Jai Sri Gurudev"

BGS INSTITUTE OF TECHNOLOGY

Department of CSE

TIME TABLE

NAME:Prasanna Kumar M J

SEMESTER: VII CS 'A'

Academic Year: 2020-21

Days/Time	09.00 am - 09.55 am	09.55 am - 10.50 am	10.50 am - 11.00 am	11.00 am - 11.55 am	11.55 am - 12.50 pm	12.50 pm - 01.45 pm	01.45 pm 04.00 pm		
Monday			TEA BREAK			LUNCH BREAK			
Tuesday									
Wednesday									
Thursday									
Friday									
Saturday	Meachine Learning Lab						Meachine Learning Lab		
DATE	A1	5/9/2020	19/9/2020	3/10/2020	17/10/2020	7/11/2020	21/11/2020	5/12/2020	9/12/2020
	A2	12/9/2020	27/9/2020	10/10/2020	31/10/2020	28/11/2020	12/12/2020	26/12/2020	2/1/2020

Sl No	Subjects:	Sub Code	Staffs:
1	Meachine Learning Lab	17CSL76	Prasanna Kumar M J + DIVYA B M

Prepared by : Dr. Raghavendra B K

Date & Sign : B. K. Ragh

HOD

Dept. of Computer Science & Engg.
Institute of Technology
Nagar 571 448
Mandya Dist

Approved by : Dr. B K Narendra

Date & Sign :

PRINCIPAL

B.G.S. INSTITUTE OF TECHNOLOGY

B.G. NAGAR - 571 448

||Jai Sri Gurudev||

BGS INSTITUTE OF TECHNOLOGY

Department of CSE

VII CS OOD Sem 2020-21

A1			A2		
Sl. No.	USN	NAME	Sl. No.	USN	NAME
1	4BW17CS001	ABHISHEK URS C J	1	4BW17CS040	NAVYASHREE H D
2	4BW17CS002	AISHWARYA D	2	4BW17CS041	NIKITH G S
3	4BW17CS003	AISHWARYA G P	3	4BW17CS042	NOOR AYESHA S
4	4BW17CS004	AISHWARYA K.P	4	4BW17CS043	POOJA D R
5	4BW17CS005	AJAY S	5	4BW17CS044	POOJA K S
6	4BW17CS006	AKANKASHA K P	6	4BW17CS045	POOJASHREE G
7	4BW17CS009	ANUPAMA A M	7	4BW17CS046	PRIYADARSHINI P
8	4BW17CS010	ATHFIA FARHEEN N	8	4BW17CS048	PRIYANKA V L
9	4BW17CS012	BHAVAN A J	9	4BW17CS050	PUNEETH RAJ B S
10	4BW17CS013	BHAVANI N D	10	4BW17CS051	RAHUL B
11	4BW17CS014	BHUMIKA M R	11	4BW17CS052	RAKESH C S
12	4BW17CS015	BINDU H	12	4BW17CS053	RAKSHITHA N
13	4BW17CS016	BRUNDA D	13	4BW17CS054	RAMYA K L
14	4BW17CS017	CHAITHRA R	14	4BW17CS055	RANJITHA B S
15	4BW17CS018	CHAITHRA JAIN H P	15	4BW17CS056	RITESH KUMAR
16	4BW17CS020	DEEKSHITHA C	16	4BW17CS057	ROHIT KUMAR JHA
17	4BW17CS021	DEEPIKA A N	17	4BW17CS058	SAHANA L M
18	4BW17CS022	DIVYA KHYANI	18	4BW17CS059	SANJANA GOWDA N
19	4BW17CS023	DIVYASHREE K H	19	4BW17CS060	SANJAY KUMAR C G
20	4BW17CS024	HARISH GOWDA	20	4BW17CS061	HANDARGAL
21	4BW17CS025	HARSHITHA Y	21	4BW17CS062	SHIFAALI
22	4BW17CS026	HEMA D	22	4BW17CS063	SHRUSTI M
23	4BW17CS027	INDU SHREE G J	23	4BW17CS064	SIDDARTH SINGH
24	4BW17CS028	ISHWARAPPA HAVIN	24	4BW17CS065	SINCHANA B R
25	4BW17CS029	JEEVAN R	25	4BW17CS066	SMITHA B U
26	4BW17CS031	JINASHREE P	26	4BW17CS084	NAMRATHA
27	4BW17CS032	KARTHIK K P	27	4BW17CS085	NAYANA
28	4BW17CS034	LAKSHMIKANTH	28	4BW17CS086	SOWMYA JAKKULA
29	4BW17CS035	MAANYA K V	29	4BW18CS403	DHANANJAYA
30	4BW17CS036	MANJUSHREE C S	30	4BW18CS404	GAGAN B S
31	4BW17CS037	MANOJ S B	31	4BW18CS406	GIRISH REDDY
32	4BW17CS038	MEGHANA K	32	4BW18CS410	VIDYASAGAR
33	4BW17CS039	MEGHANA M V			

B. K. Raj

HOD
Dept. of Computer Science & Eng
B. G. S. Institute of Technolog
B.G. Nagar - 571 448
Nagara Tq. Mandya D

||Jai Sri Gurudev||

BGS INSTITUTE OF TECHNOLOGY

Department of CSE

VII CS OOD Sem 2020-21

A3		
Sl. No.	**USN**	**NAME**
1	4BW16CS055	RAJU M D
2	4BW17CS067	SMITHA M
3	4BW17CS068	SNEHA N J
4	4BW17CS069	SOWNDARYA L T
5	4BW17CS070	SPOORTHI H
6	4BW17CS071	SPOORTHI R
7	4BW17CS072	SPOORTHI C
8	4BW17CS074	SWATHI D
9	4BW17CS075	TASMIYA
10	4BW17CS076	TEJAS RAHUL R
11	4BW17CS077	THEJAS G C
12	4BW17CS078	VARALAKSHMI C K
13	4BW17CS081	YASHASHWINI H M
14	4BW17CS082	YOGASHREE C R
15	4BW18CS400	ANUSHA K J
16	4BW18CS401	BHAVYA J K
17	4BW18CS402	BINDHUSHREE A C
18	4BW18CS405	GAYITHRI K
19	4BW18CS407	GREESHMA M S
20	4BW18CS408	KALAVATHI R
21	4BW18CS409	KEERTHI B L

B. K. Raghu

H O D

Dept. of Computer Science & Engg

B.G.S. Institute of Technology

B.G. Nagar - 571 448

Ramanagata Tq. Mandya Dist

Karnataka (INDIA)

COURSE OUTCOMES

Upon successful completion of this course, students should be able to:

Course code: 17CSL76	Course Name: Machine Learning Lab
CO1	Understand the implementation procedures for the machine learning algorithms
CO2	Design Python programs for various Learning algorithms.
CO3	Apply appropriate data sets to the Machine Learning algorithms
CO4	Identify and apply Machine Learning algorithms to solve real world problems

CO-PO-PSO MAPPING

COs	Pos												PSOs	
	PO1	PO2	PO3	PO4	PO5	PO6	P07	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO1	**2**	**2**	**2**	**1**					**1**				**2**	1
CO2	**2**	**2**	**2**	**1**					**1**				**2**	1
CO3	**2**	**2**	**2**	**1**					**1**				**2**	1
CO4	**2**	**2**	**2**	**1**					**1**				**2**	1
AVG	**2**	**2**	**2**	**1**					**1**				**2**	**1**

MAPPING	LEVEL	JUSTIFICATION
CO1-PO1	**2**	They are able to meet the engineering problem solving by MLL.
CO1-PO2	**2**	They are able to find the problem statement with trying an algorithm to solve it.
CO1-PO3	**2**	They are able to solve using mathematical functions usage
CO1-PO4	**1**	They are able to conduct with different data sets with either algorithm to check efficiency.
CO1-PO9	**1**	They are able to check the program efficiency individually .
CO1-PSO1	**2**	Now they are able check compatibility of different or suitable algorithm for the real time problem
CO1-PSO2	**1**	Now they are able to solve or apply the ML algorithm to real world problems
CO2-PO1	**2**	They are able to calculate the algorithm complexity with designment of programming method referred

CO2-PO2	**2**	They can analyse the problem and method suitability with python program structure.
CO2-PO3	**2**	Now they can design or program in python for given problem statement
CO2-PO4	**1**	They try with different data set and conduct different test case with program execution time.
CO2-PO9	**1**	They can calculate the time and need of algorithm individually the capability of each algorithm.
CO2-PSO1	**2**	New principles to adopt with the programming with new tool implementation.
CO2-PSO2	**1**	Designing the trend practicing new methods to implement with ML language adopted programming language
CO3-PO1	**2**	Checking from the basic knowledge to adopt the different dataset to implement.
CO3-PO2	**2**	They try to substantiated solution to try or make the machine to understand the need.
CO3-PO3	**2**	Based on socio-economic strategy, they plan to design the algorithm execution.
CO3-PO4	**1**	They will come to conclusion by synthesis of result and work mode
CO3-PO9	**1**	They discuss in team to come to an conclusion , regarding the adopted method or technology
CO3-PSO1	**2**	They think of real time existing problem to solve using the statement in advancement way
CO3-PSO2	**1**	Try to make an IoT or Machine learning programming with different datasets or db's
CO 4-PO1	**2**	They will search the need of ML from basic to advanced need in real life
CO 4-PO2	**2**	Make an literature survey of different way of solving methods to each of them

CO 4-PO3	**2**	Planning of working principal and consideration the appropriate one to reassignment
CO 4-PO4	**1**	Research on the problem as well solved methods usage
CO4-PO9	**1**	Individually and in team they can conclude the work to be done
CO4-PSO1	**2**	Try with updated tools to work to solve the problem statement in the given programming language.
CO4-PSO2	**1**	They can execute the new model with high performance model to solve the real life time problems.

MACHINE LEARNING LABORATORY **[As per Choice Based Credit System (CBCS) scheme]** **(Effective from the academic year 2016 -2017)** **SEMESTER – VII**			
Subject Code	17CSL76	IA Marks	20
Number of Lecture Hours/Week	01I + 02P	Exam Marks	60
Total Number of Lecture Hours	40	Exam Hours	03
CREDITS – 02			

Course objectives: This course will enable students to

1. Make use of Data sets in implementing the machine learning algorithms
2. Implement the machine learning concepts and algorithms in any suitable language of choice.

Description (If any):

1. The programs can be implemented in either JAVA or Python.
2. For Problems 1 to 6 and 10, programs are to be developed without using the built-in classes or APIs of Java/Python.
3. Data sets can be taken from standard repositories (https://archive.ics.uci.edu/ml/datasets.html) or constructed by the students.

1	Implement and demonstrate the **FIND-S algorithm** for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.
2	For a given set of training data examples stored in a .CSV file, implement and demonstrate the **Candidate-Elimination algorithm** to output a description of the set of all hypotheses consistent with the training examples.
3	Write a program to demonstrate the working of the decision tree based **ID3 algorithm**. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
4	Build an Artificial Neural Network by implementing the **Backpropagation algorithm** and test the same using appropriate data sets.
5	Write a program to implement the **naïve Bayesian classifier** for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
6	Assuming a set of documents that need to be classified, use the **naïve Bayesian Classifier** model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.
7	Write a program to construct a **Bayesian network** considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.
8	Apply **EM algorithm** to cluster a set of data stored in a .CSV file. Use the same data set for clustering using ***k*-Means algorithm**. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.
9	Write a program to implement ***k*-Nearest Neighbour algorithm** to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.
10	Implement the non-parametric **Locally Weighted Regression algorithm** in order to

	fit data points. Select appropriate data set for your experiment and draw graphs.

Course outcomes: The students should be able to:

1. Understand the implementation procedures for the machine learning algorithms.
2. Design Java/Python programs for various Learning algorithms.
3. Apply appropriate data sets to the Machine Learning algorithms.
4. Identify and apply Machine Learning algorithms to solve real world problems.

Conduction of Practical Examination:

- All laboratory experiments are to be included for practical examination.
- Students are allowed to pick one experiment from the lot.
- Strictly follow the instructions as printed on the cover page of answer script
- Marks distribution: Procedure + Conduction + Viva:**20 + 50 +10 (80)**

Change of experiment is allowed only once and marks allotted to the procedure part to be made zero.

TABLE OF CONTENTS

1.Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a CSV file.

```
import numpy as np
import pandas as pd
data = pd.DataFrame(data=pd.read_csv('finds.csv'))
concepts = np.array(data.iloc[:,0:-1])
target = np.array(data.iloc[:,-1])
def learn(concepts, target):
    specific_h = concepts[0].copy()
    for i, h in enumerate(concepts):
        if target[i] == "Yes":
            for x in range(len(specific_h)):
                if h[x] != specific_h[x]:
                    specific_h[x] = "?"
    return specific_h
specific_h = learn(concepts, target)
print(specific_h)
```

Input: finds.csv

Example	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2	Sunny	Warm	High	Strong	Warm	Same	Yes
3	Rainy	Cold	High	Strong	Warm	Change	No
4	Sunny	Warm	High	Strong	Cool	Change	Yes

Output:

```
['Sunny' 'Warm' '?' 'Strong' '?' '?']
```

2.For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent

```
import numpy as np
import pandas as pd
data = pd.DataFrame(data=pd.read_csv('finds.csv'))
concepts = np.array(data.iloc[:,0:-1])
target = np.array(data.iloc[:,-1])
def learn(concepts,target):
    specific_h = concepts[0].copy()
    general_h = [["?" for i in range(len(specific_h))] for i in range(len(specific_h))]
    for i, h in enumerate(concepts):
        # Checking if the hypothesis has a positive target
        if target[i] == "Yes":
            for x in range(len(specific_h)):
                # Change values in S & G only if values change
                if h[x] != specific_h[x]:
                    specific_h[x] = '?'
                    general_h[x][x] = '?'

        # Checking if the hypothesis has a negative target
        if target[i] == "No":
            for x in range(len(specific_h)):

                # For negative hyposthesis change values only in G
                if h[x] != specific_h[x]:
                    general_h[x][x] = specific_h[x]
                else:
                    general_h[x][x] = '?'
```

```
# find indices where we have empty rows, meaning those that are unchanged
indices = [i for i,val in enumerate(general_h) if val == ['?', '?', '?', '?', '?', '?']]
for i in indices:
    general_h.remove(['?', '?', '?', '?', '?', '?'])
return specific_h, general_h
```

Input: finds.csv

Example	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2	Sunny	Warm	High	Strong	Warm	Same	Yes
3	Rainy	Cold	High	Strong	Warm	Change	No
4	Sunny	Warm	High	Strong	Cool	Change	Yes

Output:

```
Final G:
[['Sunny', '?', '?', '?', '?', '?'], ['?', 'Warm', '?', '?', '?', '?']]
```

3.Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.

```
import numpy as np
import pandas as pd
from pprint import pprint
data = pd.read_csv("playTennis.csv")
data_size= len(data)
treenodes = []
tree = {"ROOT": data}

def total_entropy(data, col):
    mydict = {}
    for elem in data[col]:
        if elem in mydict.keys():
            mydict[elem] += 1
        else:
            mydict[elem] = 1
    total = sum(mydict.values())
    E = 0
    for key in mydict.keys():
        E += entropy(mydict[key], total)
    return E

def entropy(num, denom):
    return -(num/denom) * np.log2(num/denom)

def get_sorted_data(data, column):
    sort = {}
    for column_name in get_attributes(data, column):
        sort[column_name] = data.loc[data[column]==column_name]
    return sort


def get_attributes(data, column):
    return data[column].unique().tolist()

def InfoGain(total_entropy, sorted_data, entropy_by_attribute):
    length = data_size
    total = 0
    for col, df in sorted_data.items():
        total += (len(df) / length) * entropy_by_attribute[col]
    return total_entropy - total


def get_entropy_by_attribute(sorted_data):
    entropies = {}
```

```
    for key, df in sorted_data.items():
        entropies[key] = total_entropy(df, 'Decision')
    return entropies


def drop_node(data, column):
    return data.drop(column, axis=1)

def id3(tree):
    for branch, data in tree.items():
        # Make sure it's a DataFrame
        if not isinstance(data, pd.DataFrame):
            continue

        #Fetch column names so you can use them to iterate later
        columns = data.columns

        # Calculate the Entropy for the entire dataset
        total_entropy_for_data = total_entropy(data.values, -1)

        # If only one column is left, it means we're done.
        if len(columns) == 1:
            break


        # Keep track of information gain to choose the attribute with maximum info gain.
        info_gain_list = []

        # Now iterate over each column to calculate information gain w.r.t o/p column
        for i in range(0, len(data.columns)-1):

            # Sort the rows w.r.t o/p
            sorted_rows = get_sorted_data(data, columns[i])

            # Calculate the entropy w.r.t to each attribute based on sorted columns
            entropy_by_attribute =
get_entropy_by_attribute(sorted_rows)

            # get the info gain
            info_gain = InfoGain(total_entropy_for_data, sorted_rows,
entropy_by_attribute)

            # save it
            info_gain_list.append(info_gain)

        # Find index of max info gain
        node = info_gain_list.index(max(info_gain_list))

        # sort the data into branches based on the new node
        branches = get_sorted_data(data, columns[node])
```

```
        # If we've reached the end of iterations, just assign the
value,
        #else drop the sorted column
        for attr, df in branches.items():
            if (total_entropy(df, columns[-1]) == 0):
                branches[attr] = df.iloc[0,-1]
            else:
                branches[attr] = df.drop(columns[node], axis=1)

        # Keep track of nodes already done
        treenodes.append(columns[node])

        # add the new branches to the tree
        child = {columns[node]: {}}
        tree[branch] = child
        tree[branch][columns[node]] = branches
        # ID3
        id3(tree[branch][columns[node]])
x=id3(tree)
pprint(tree, depth=5)
```

Input: playTennis.csv

Outlook	Temp	Humidity	Wind	Decision
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Mild	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Mild	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Mild	Normal	Weak	Yes
Sunny	Mild	Normal	Strong	Yes
Overcast	Mild	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Mild	High	Strong	No

Output:

```
{'ROOT': {'Outlook': {'Overcast': 'Yes',
                      'Rain': {'Wind': {'Strong': 'No', 'Weak': 'Yes'}},
                      'Sunny': {'Humidity': {'High': 'No', 'Normal': 'Ye
s'}}}}}
```

4.Build an Artificial Neural Network by implementing the Back propagation algorithm and test the same using appropriate dataset

```
import numpy as np
X = np.array(([2, 9], [1, 5], [3, 6]), dtype=float)
y = np.array(([92], [86], [89]), dtype=float)
X = X/np.amax(X,axis=0) # maximum of X array longitudinally
y = y/100
#Sigmoid Function
def sigmoid (x):
    return 1/(1 + np.exp(-x))

#Derivative of Sigmoid Function
def derivatives_sigmoid (x):
    return x*(1-x)

#Variable initialization
epoch=7000 #Setting training iterations
lr=0.1 #Setting learning rate
inputlayer_neurons = 2 #number of features in data set
hiddenlayer_neurons = 3 #number of hidden layers neurons
output_neurons = 1 #number of neurons at output layer

#weight and bias initialization
wh=np.random.uniform(size=(inputlayer_neurons,hiddenlayer_neurons))
bh=np.random.uniform(size=(1,hiddenlayer_neurons))
wout=np.random.uniform(size=(hiddenlayer_neurons,output_neurons))
bout=np.random.uniform(size=(1,output_neurons))

#for i in range(epoch):
    #Forward Propogation
    hinp1=np.dot(X,wh)
    hinp=hinp1 + bh
    hlayer_act = sigmoid(hinp)

    outinp1=np.dot(hlayer_act,wout)
    outinp= outinp1+ bout
    output = sigmoid(outinp)

    #Backpropagation
    EO = y-output
    outgrad = derivatives_sigmoid(output)
    d_output = EO* outgrad

    EH = d_output.dot(wout.T)
    hiddengrad = derivatives_sigmoid(hlayer_act)
    #how much hidden layer wts contributed to error
    d_hiddenlayer = EH * hiddengrad
```

```
    wout += hlayer_act.T.dot(d_output) *lr# dotproduct of nextlayererror and currentlayerop
    wh += X.T.dot(d_hiddenlayer) *lr

print("Input: \n" + str(X))
print("Actual Output: \n" + str(y))
print("Predicted Output: \n" ,output)
```

Output:

```
Input:
[[0.66666667 1.        ]
 [0.33333333 0.55555556]
 [1.         0.66666667]]
Actual Output:
[[0.92]
 [0.86]
 [0.89]]
Predicted Output:
 [[0.89704366]
 [0.87644182]
 [0.89586104]]
```

5. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.

```
import csv
import random
import math

def loadCsv(filename):
    lines=csv.reader(open(filename,"r"))
    dataset=list(lines)
    for i in range(len(dataset)):
        dataset[i]=[float(x)for x in dataset[i]]
    return dataset

def splitDataset(dataset,splitRatio):
    trainSize=int(len(dataset)*splitRatio)
    trainSet=[]
    copy=list(dataset)
    while len(trainSet)<trainSize:
        index=random.randrange(len(copy))
        trainSet.append(copy.pop(index))
    return[trainSet,copy]

def seperateByClass(dataset):
    seperated={}
    for i in range(len(dataset)):
        vector=dataset[i]
        if(vector[-1] not in seperated):
            seperated[vector[-1]]=[]
        seperated[vector[-1]].append(vector)
    return seperated

def mean(numbers):
    return sum(numbers)/float(len(numbers))

def stdev(numbers):
    avg=mean(numbers)
    variance=sum([pow(x-avg,2) for x in
numbers])/float(len(numbers)-1)
    return math.sqrt(variance)

def summarize(dataset):
    summaries=[(mean(attribute),stdev(attribute)) for attribute in zip(*dataset)]
    del summaries[-1]
    return summaries

def summarizeByClass(dataset):
    seperated=seperateByClass(dataset)
```

```
    summaries={}
    for classValue,instances in seperated.items():
         summaries[classValue]=summarize(instances)
    return summaries


def calculateProbability(x,mean,stdev):
    exponent=math.exp(-(math.pow(x-mean,2)/(2*math.pow(stdev,2))))
    return(1/(math.sqrt(2*math.pi)*stdev))*exponent

def calculateClassProbabilities(summaries,inputVector):
    probabilities={}
    for classValue,classSummaries in summaries.items():
        probabilities[classValue]=1
        for i in range(len(classSummaries)):
            mean,stdev=classSummaries[i]
            x=inputVector[i]

probabilities[classValue]*=calculateProbability(x,mean,stdev)
        return probabilities

def predict(summaries,inputVector):
    probabilities=calculateClassProbabilities(summaries,inputVector)
    bestLabel,bestProb=None,-1
    for classValue,probability in probabilities.items():
        if bestLabel is None or probability>bestProb:
                bestProb=probability
                bestLabel=classValue
    return bestLabel

def getPredictions(summaries,testSet):
    predictions=[]
    for i in range(len(testSet)):
            result=predict(summaries,testSet[i])
            predictions.append(result)
    return predictions

def getAccuracy(testSet,predictions):
    correct=0
    for i in range(len(testSet)):
        if testSet[i][-1]==predictions[i]:
            correct+=1

        return(correct/float(len(testSet)))*100.0

def main():
    filename='5_pima-indians-diabetes.data.csv'
    splitRatio=0.67
```

```
    dataset=loadCsv(filename)
    trainingSet,testSet=splitDataset(dataset,splitRatio)
    print('Split {0} rows into train={1} and test={2} rows'.
          format(len(dataset),len(trainingSet),len(testSet)))
        summaries=summarizeByClass(trainingSet)
    predictions=getPredictions(summaries,testSet)
    accuracy=getAccuracy(testSet,predictions)
    print('Accuracy:{0}%'.format(accuracy))
main()
```

Input: 5 pima-indians-diabetes.data.csv

Output:

```
Split 768 rows into train=514 and test=254 rows
Accuracy:0.39370078740157477%
```

6.Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.

```
import pandas as pd
msg=pd.read_csv('6pg.csv',names=['message','label'])
print('The dimensions of the dataset',msg.shape)
msg['labelnum']=msg.label.map({'pos':1,'neg':0})
X=msg.message
y=msg.labelnum
print(X)
print(y)
#splitting the dataset into train and test data
from sklearn.model_selection import train_test_split
xtrain,xtest,ytrain,ytest=train_test_split(X,y)
print(xtest.shape)
print(xtrain.shape)
print(ytest.shape)
print(ytrain.shape)
#output of count vectoriser is a sparse matrix
from sklearn.feature_extraction.text
import CountVectorizer
count_vect = CountVectorizer()
xtrain_dtm = count_vect.fit_transform(xtrain)
xtest_dtm=count_vect.transform(xtest)
print(count_vect.get_feature_names())
df=pd.DataFrame(xtrain_dtm.toarray(),columns=count_vect.get_feature_
names())
print(df)#tabular representation
print(xtrain_dtm) #sparse matrix representation
# Training Naive Bayes (NB) classifier on training data.
from sklearn.naive_bayes import MultinomialNB
clf = MultinomialNB().fit(xtrain_dtm,ytrain)
predicted = clf.predict(xtest_dtm)
#printing accuracy metrics
from sklearn import metrics
print('Accuracy metrics')
print('Accuracy of the classifer
is',metrics.accuracy_score(ytest,predicted))
print('Confusion matrix')
print(metrics.confusion_matrix(ytest,predicted))
print('Recall and Precison ')
print(metrics.recall_score(ytest,predicted))
print(metrics.precision_score(ytest,predicted))
```

Input: 6pg.csv

Output:

```
Accuracy metrics
Accuracy of the classifer is 0.4

Confusion matrix
[[1 2]
 [1 1]]
Recall and Precison
0.5
0.3333333333333333
```

7.Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.

```
import bayespy as bp
import numpy as np
import csv
from colorama import init
from colorama import Fore,Back,Style
init()

ageEnum = {'SuperSeniorCitizen':0,'SeniorCitizen':1,'MiddleAged':2,'Youth':3,'Teen':4}
genderEnum = {'Male':0,'Female':1}
familyHistoryEnum={'Yes':0,'No':1}
dietEnum = {'High':0,'Medium':1,'Low':2}
lifeStyleEnum = {'Athlete':0,'Active':1,'Moderate':2,'Sedetary':3}
cholestrolEnum ={'High':0,'BorderLine':1,'Normal':2}
heartDiseaseEnum = {'Yes':0,'No':1}

# reading csv file
with open('heart_disease_data.csv') as csvfile:
    # creating a csv reader object
    lines = csv.reader(csvfile)
    dataset = list(lines)
    data =[]
    for x in dataset:
      data.append([ageEnum[x[0]],genderEnum[x[1]],familyHistoryEnum[x[2]],dietEnum[x[3]],lif
      eStyleEnum[x[4]],cholestrolEnum[x[5]],heartDiseaseEnum[x[6]]])

data = np.array(data)
N = len(data)

# Group assignment probabilities
p_age = bp.nodes.Dirichlet(1.0*np.ones(5))
# Group assignments for nodes
age = bp.nodes.Categorical(p_age,plates=(N,))
age.observe(data[:,0])

p_gender = bp.nodes.Dirichlet(1.0*np.ones(2))
gender = bp.nodes.Categorical(p_gender,plates = (N,))
gender.observe(data[:,1])

p_familyhistory = bp.nodes.Dirichlet(1.0*np.ones(2))
familyhistory = bp.nodes.Categorical(p_familyhistory,plates = (N,))
familyhistory.observe(data[:,2])

p_diet = bp.nodes.Dirichlet(1.0*np.ones(3))
diet = bp.nodes.Categorical(p_diet,plates = (N,))
diet.observe(data[:,3])
```

```
p_lifestyle = bp.nodes.Dirichlet(1.0*np.ones(4))
lifestyle = bp.nodes.Categorical(p_lifestyle,plates = (N,))
lifestyle.observe(data[:,4])

p_cholestrol = bp.nodes.Dirichlet(1.0*np.ones(3))
cholestrol = bp.nodes.Categorical(p_cholestrol,plates = (N,))
cholestrol.observe(data[:,3])

p_heartdisease = bp.nodes.Dirichlet(np.ones(2), plates=(5, 2, 2, 3,
4, 3))
heartdisease = bp.nodes.MultiMixture([age, gender, familyhistory,
diet, lifestyle, cholestrol], bp.nodes.Categorical, p_heartdisease)
heartdisease.observe(data[:,6])
p_heartdisease.update()

m = 0
while m == 0:
    print("\n")
      res = bp.nodes.MultiMixture([int(input('Enter Age: y' +
      str(ageEnum))),   int(input('Enter Gender: ' +
      str(genderEnum))), int(input('Enter FamilyHistory: ' +
      str(familyHistoryEnum))), int(input('Enter dietEnum: ' +
      str(dietEnum))), int(input('Enter LifeStyle: ' +
      str(lifeStyleEnum))), int(input('Enter Cholesterol: ' +
      str(cholestrolEnum)))], bp.nodes.Categorical,
      p_heartdisease).get_moments()[0][heartDiseaseEnum['No']]
    print("Probability(HeartDisease) = " +  str(res))
    m = int(input("Enter for Continue:0, Exit :1  "))
```

Input: heart disease data.csv

```
Enter Age: y{'SuperSeniorCitizen': 0, 'SeniorCitizen': 1, 'MiddleAged':
2, 'Youth': 3, 'Teen': 4}
1

Enter Gender: {'Male': 0, 'Female': 1}
1

Enter FamilyHistory: {'Yes': 0, 'No': 1}
0

Enter dietEnum: {'High': 0, 'Medium': 1, 'Low': 2}
1

Enter LifeStyle: {'Athlete': 0, 'Active': 1, 'Moderate': 2, 'Sedetary':
3} 3

Enter Cholesterol: {'High': 0, 'BorderLine': 1, 'Normal': 2}
1
```

Output:

```
Enter Age: y{'SuperSeniorCitizen': 0, 'SeniorCitizen': 1, 'MiddleAged': 2, 'Youth': 3, 'Tee
n': 4}1
Enter Gender: {'Male': 0, 'Female': 1}1
Enter FamilyHistory: {'Yes': 0, 'No': 1}0
Enter dietEnum: {'High': 0, 'Medium': 1, 'Low': 2}1
Enter LifeStyle: {'Athlete': 0, 'Active': 1, 'Moderate': 2, 'Sedetary': 3}3
Enter Cholesterol: {'High': 0, 'BorderLine': 1, 'Normal': 2}1
Probability(HeartDisease) = 0.5

Enter for Continue:0, Exit :1   1
```

8.Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using *k*-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.

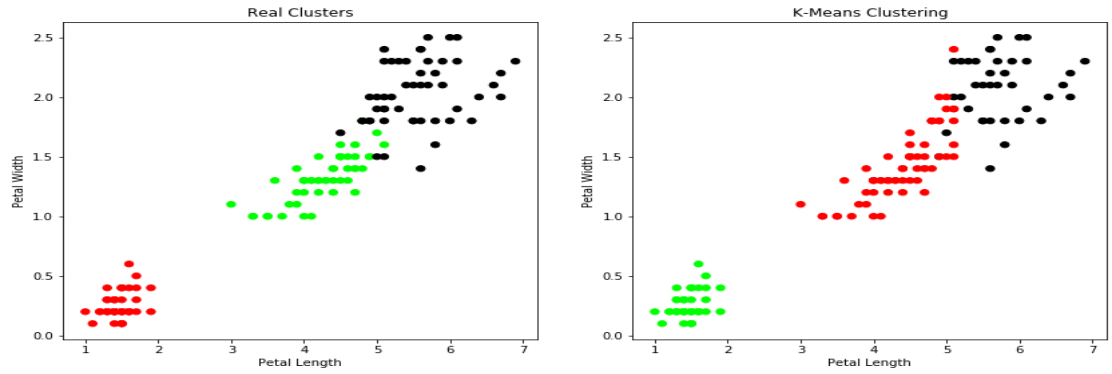
```
import matplotlib.pyplot as plt
from sklearn import datasets
from sklearn.cluster
import KMeans
import pandas as pd
import numpy as np

# import some data to play with
iris = datasets.load_iris()
X = pd.DataFrame(iris.data)
X.columns =
['Sepal_Length','Sepal_Width','Petal_Length','Petal_Width']
y = pd.DataFrame(iris.target)
y.columns = ['Targets']

# Build the K Means Model
model = KMeans(n_clusters=3)
model.fit(X) # model.labels_ : Gives cluster no for which samples
belongs to

# Visualise the clustering results
plt.figure(figsize=(14,14))
colormap = np.array(['red', 'lime', 'black'])

# Plot the Original Classifications using Petal features
plt.subplot(2, 2, 1)
plt.scatter(X.Petal_Length, X.Petal_Width, c=colormap[y.Targets],
s=40)
plt.title('Real Clusters')
plt.xlabel('Petal Length')
plt.ylabel('Petal Width')

# Plot the Models Classifications
plt.subplot(2, 2, 2)
plt.scatter(X.Petal_Length, X.Petal_Width,
c=colormap[model.labels_], s=40)
plt.title('K-Means Clustering')
plt.xlabel('Petal Length')
plt.ylabel('Petal Width')

# General EM for GMM
from sklearn import preprocessing
scaler = preprocessing.StandardScaler()
```

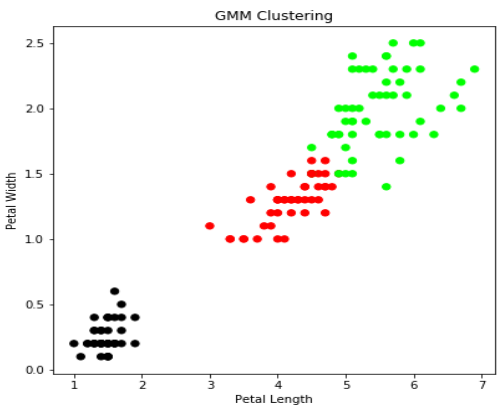
```
scaler.fit(X)
xsa = scaler.transform(X)
xs = pd.DataFrame(xsa, columns = X.columns)

from sklearn.mixture import GaussianMixture
gmm = GaussianMixture(n_components=3)
gmm.fit(xs)
gmm_y = gmm.predict(xs)


plt.subplot(2, 2, 3)
plt.scatter(X.Petal_Length, X.Petal_Width, c=colormap[gmm_y], s=40)
plt.title('GMM Clustering')
plt.xlabel('Petal Length')
plt.ylabel('Petal Width')

print('Observation: The GMM using EM algorithm based clustering
matched the true labels more closely than the Kmeans.')
```

Input: iris.data

Output:

```
Observation: The GMM using EM algorithm based clustering matched the true labels more closely than the Kmeans.
```





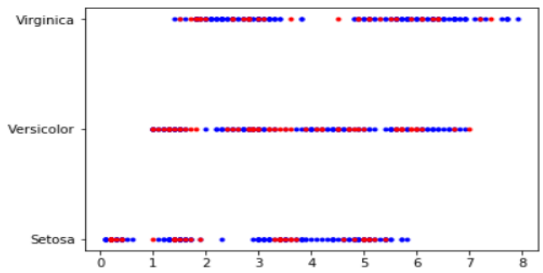
9.Write a program to implement ***k*-Nearest Neighbour algorithm** to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# Read dataset to pandas dataframe
dataset = pd.read_csv("iris.csv")

X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, 4].values

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.20)
plt.plot(X_train,y_train,'b.',X_test,y_test,'r.')

from sklearn.neighbors import KNeighborsClassifier
classifier = KNeighborsClassifier(n_neighbors=5)
classifier.fit(X_train, y_train)

accuracy=classifier.score(X_test,y_test)
accuracy1=classifier.score(X_train,y_train)

print(accuracy)
print(accuracy1)

example=np.array([7.7,2.6,6.9,2.3])
example=example.reshape(1,-1)
print(example)
```

Input: iris.csv

Output:

```
0.9333333333333333
0.975
[[7.7 2.6 6.9 2.3]]
['Virginica']
```



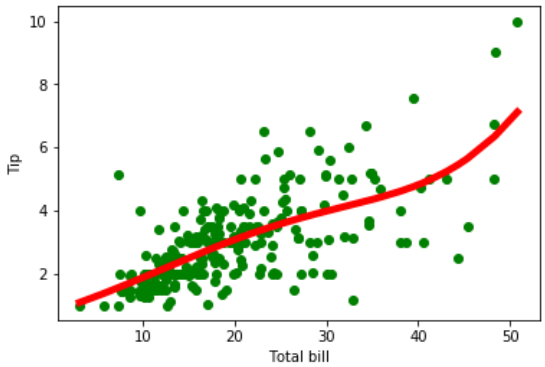
10. Implement the non-parametric **Locally Weighted Regression algorithm** in order to fit data points. Select appropriate data set for your experiment and draw graphs.

```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np

def kernel(point,xmat, k):
    m,n = np.shape(xmat)
    weights = np.mat(np.eye((m))) # eye - identity matrix
    for j in range(m):
        diff = point - X[j]
        weights[j,j] = np.exp(diff*diff.T/(-2.0*k**2))
    return weights

def localWeight(point,xmat,ymat,k):
    wei = kernel(point,xmat,k)
    W = (X.T*(wei*X)).I*(X.T*(wei*ymat.T))
    return W

def localWeightRegression(xmat,ymat,k):
    m,n = np.shape(xmat)
    ypred = np.zeros(m)
    for i in range(m):
        ypred[i] = xmat[i]*localWeight(xmat[i],xmat,ymat,k)
    return ypred

def graphPlot(X,ypred):
    sortindex = X[:,1].argsort(0) #argsort - index of the smallest
    xsort = X[sortindex][:,0]
    fig = plt.figure()
    ax = fig.add_subplot(1,1,1)
    ax.scatter(bill,tip, color='green')
    ax.plot(xsort[:,1],ypred[sortindex], color = 'red', linewidth=5)
    plt.xlabel('Total bill')
    plt.ylabel('Tip')
    plt.show();

# load data points
data = pd.read_csv('data10_tips.csv')
bill = np.array(data.total_bill) # We use only Bill amount and Tips
data
tip = np.array(data.tip)
mbill = np.mat(bill) # .mat will convert nd array is converted in 2D
array
mtip = np.mat(tip)
m= np.shape(mbill)[1]
one = np.mat(np.ones(m))
```

```
X = np.hstack((one.T,mbill.T)) # 244 rows, 2 cols
ypred = localWeightRegression(X,mtip,8) # increase k to get smooth
curves
graphPlot(X,ypred)
```

Input: data10 tips

Output:



Sample Viva Questions

1. What is machine learning?

2. Define supervised learning

3. Define unsupervised learning

4. Define semi supervised learning

5. Define reinforcement learning

6. What do you mean by hypotheses

7. What is classification

8. What is clustering

9. Define precision, accuracy and recall

10.Define entropy

11.Define regression

12.How Knn is different from k-means clustering

13.What is concept learning

14.Define specific boundary and general boundary

15.Define target function 16.Define decision tree

17.What is ANN

18.Explain gradient descent approximation

19.State Bayes theorem

20.Define Bayesian belief networks

21.Differentiate hard and soft clustering

22.Define variance

23.What is inductive machine learning

24.Why K nearest neighbour algorithm is lazy learning algorithm

25.Why naïve Bayes is naïve 26.Mention classification algorithms

26.Mention classification algorithms

27.Define pruning

28.Differentiate Clustering and classification

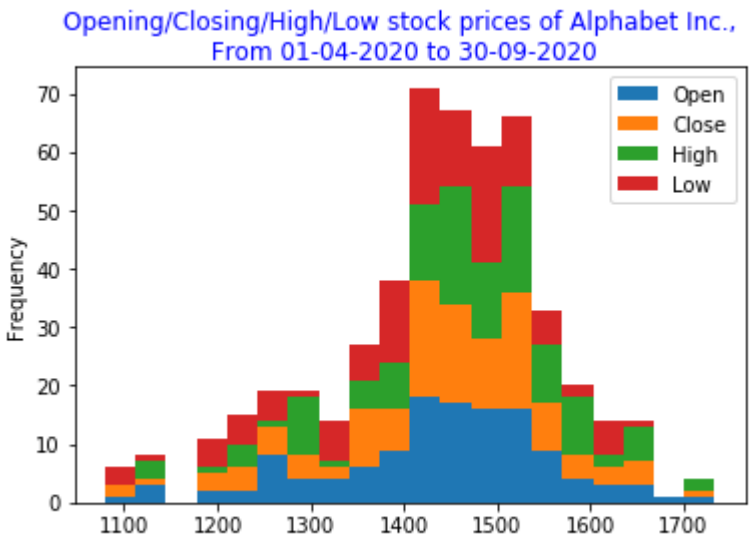
29.Mention clustering algorithms

30.Define Bias

11. **Write a Pandas program to create a stacked histograms plot of opening, closing, high, low stock prices of Alphabet Inc. between two specific dates.**

```
import pandas as pd
import matplotlib.pyplot as plt
df = pd.read_csv("alphabet_stock_data.csv")
start_date = pd.to_datetime('2020-4-1')
end_date = pd.to_datetime('2020-9-30')
df['Date'] = pd.to_datetime(df['Date'])
new_df = (df['Date']>= start_date) & (df['Date']<= end_date)
df1 = df.loc[new_df]
df2 = df1[['Open','Close','High','Low']]
plt.figure(figsize=(25,25))
df2.plot.hist(stacked=True, bins=20)
plt.suptitle('Opening/Closing/High/Low stock prices of Alphabet
Inc.,\n From 01-04-2020 to 30-09-2020', fontsize=12, color='blue')
plt.show()
```

OUTPUT:



||Jai Sri Gurudev||

BGS Institute of Technology

BG Nagara-571448, Mandya

Result Analysis

Course Coordinator: Mr. M J Prasanna Kumar

Course Name & Code: Machine Learning Lab 17CS

Semester & Sec: 7th Sem

Academic Year: 2020-2021

Sl. No.	USN	NAME	IA	SEE	Total	Result
1	4BW17CS001	ABHISHEK URS C J	40	59	99	Pass
2	4BW17CS002	AISHWARYA D	36	51	87	Pass
3	4BW17CS003	AISHWARYA G P	35	50	85	Pass
4	4BW17CS004	AISHWARYA K.P	38	51	89	Pass
5	4BW17CS005	AJAY S	38	59	97	Pass
6	4BW17CS006	AKANKASHA K P	35	50	85	Pass
7	4BW17CS009	ANUPAMA A M	35	50	85	Pass
8	4BW17CS010	ATHFIA FARHEEN N	35	50	85	Pass
9	4BW17CS012	BHAVAN A J	38	59	97	Pass
10	4BW17CS013	BHAVANI N D	38	51	89	Pass
11	4BW17CS014	BHUMIKA M R	35	49	84	Pass
12	4BW17CS015	BINDU H	35	50	85	Pass
13	4BW17CS016	BRUNDA D	35	49	84	Pass
14	4BW17CS017	CHAITHRA R	35	49	84	Pass
15	4BW17CS018	CHAITHRA JAIN H P	38	59	97	Pass
16	4BW17CS020	DEEKSHITHA C	35	58	93	Pass
17	4BW17CS021	DEEPIKA A N	35	49	84	Pass
18	4BW17CS022	DIVYA KHYANI	38	59	97	Pass
19	4BW17CS023	DIVYASHREE K H	35	56	91	Pass
20	4BW17CS024	HARISH GOWDA	38	59	97	Pass
21	4BW17CS025	HARSHITHA Y	35	48	83	Pass
22	4BW17CS026	HEMA D	35	50	85	Pass
23	4BW17CS027	INDU SHREE G J	35	59	94	Pass
24	4BW17CS028	ISHWARAPPA HAVIN	35	47	82	Pass
25	4BW17CS029	JEEVAN R	38	59	97	Pass
26	4BW17CS031	JINASHREE P	35	49	84	Pass
27	4BW17CS032	KARTHIK K P	35	51	86	Pass
28	4BW17CS034	LAKSHMIKANTH GOWDA M	39	59	98	Pass
29	4BW17CS035	MAANYA K V	35	47	82	Pass

30	4BW17CS036	MANJUSHREE C S	35	59	94	Pass
31	4BW17CS037	MANOJ S B	35	40	75	Pass
32	4BW17CS038	MEGHANA K	35	52	87	Pass
33	4BW17CS039	MEGHANA M V	35	49	84	Pass
34	4BW17CS040	NAVYASHREE H D	35	57	92	Pass
35	4BW17CS041	NIKITH G S	38	56	94	Pass
36	4BW17CS042	NOOR AYESHA S	35	47	82	Pass
37	4BW17CS043	POOJA D R	35	51	86	Pass
38	4BW17CS044	POOJA K S	35	49	84	Pass
39	4BW17CS045	POOJASHREE G	35	49	84	Pass
40	4BW17CS046	PRIYADARSHINI P	35	46	81	Pass
41	4BW17CS048	PRIYANKA V L	35	55	90	Pass
42	4BW17CS050	PUNEETH RAJ B S	35	46	81	Pass
43	4BW17CS051	RAHUL B	35	39	74	Pass
44	4BW17CS052	RAKESH C S	35	43	78	Pass
45	4BW17CS053	RAKSHITHA N	35	49	84	Pass
46	4BW17CS054	RAMYA K L	35	48	83	Pass
47	4BW17CS055	RANJITHA B S	35	48	83	Pass
48	4BW17CS056	RITESH KUMAR CHANDA	35	51	86	Pass
49	4BW17CS057	ROHIT KUMAR JHA	35	48	83	Pass
50	4BW17CS058	SAHANA L M	35	48	83	Pass
51	4BW17CS059	SANJANA GOWDA N C	40	57	97	Pass
52	4BW17CS060	SANJAY KUMAR C G	35	55	90	Pass
53	4BW17CS061	SHANKREPPA HANDARGAL	35	52	87	Pass
54	4BW17CS062	SHIFAALI	35	48	83	Pass
55	4BW17CS063	SHRUSTI M	35	48	83	Pass
56	4BW17CS064	SIDDARTH SINGH	35	55	90	Pass
57	4BW17CS065	SINCHANA B R	35	56	91	Pass
58	4BW17CS066	SMITHA B U	35	48	83	Pass
59	4BW17CS084	NAMRATHA	35	49	84	Pass
60	4BW17CS085	NAYANA	35	46	81	Pass
61	4BW17CS086	SOWMYA JAKKULA	35	48	83	Pass
62	4BW18CS403	DHANANJAYA	35	44	79	Pass
63	4BW18CS404	GAGAN B S	35	55	90	Pass
64	4BW18CS406	GIRISH REDDY	35	40	75	Pass
65	4BW18CS410	VIDYASAGAR	30	38	68	Pass
66	4BW16CS055	RAJU M D	35	54	89	Pass
67	4BW17CS067	SMITHA M	35	48	83	Pass
68	4BW17CS068	SNEHA N J	40	59	99	Pass
69	4BW17CS069	SOWNDARYA L T	35	48	83	Pass
70	4BW17CS070	SPOORTHI H	35	48	83	Pass
71	4BW17CS071	SPOORTHI R	35	48	83	Pass
72	4BW17CS072	SPOORTHI C	35	48	83	Pass
73	4BW17CS074	SWATHI D	35	48	83	Pass

74	4BW17CS075	TASMIYA	35	49	84	Pass
75	4BW17CS076	TEJAS RAHUL R	35	48	83	Pass
76	4BW17CS077	THEJAS G C	35	45	80	Pass
77	4BW17CS078	VARALAKSHMI C K	35	48	83	Pass
78	4BW17CS081	YASHASHWINI H M	35	45	80	Pass
79	4BW17CS082	YOGASHREE C R	35	41	76	Pass
80	4BW18CS400	ANUSHA K J	35	40	75	Pass
81	4BW18CS401	BHAVYA J K	35	45	80	Pass
82	4BW18CS402	BINDHUSHREE A C	35	42	77	Pass
83	4BW18CS405	GAYITHRI K	35	41	76	Pass
84	4BW18CS407	GREESHMA M S	35	45	80	Pass
85	4BW18CS408	KALAVATHI R	35	56	91	Pass
86	4BW18CS409	KEERTHI B L	35	41	76	Pass

No. of Students Attended	86
No. of Students Absent	Nil
No. of Students Passed	86
No. of Students Failed	0
Pass Percentage	100%

M. I Prasad

Signature of Course Coordinator

B. K. Raghu

Signature of HoD

H O D
Dept. of Computer Science & Engg
S. Institute of Technology
B G. Nagar - 571 448
ogala Tq. Mandya Dist
taka (INDIA)

ML Lab 2020-21 Batch 2017 Scheme

	60%		30%	10%	
	CIE		SEE	CES	TOTAL
CO1	2.99		2.66	2.9	2.88
CO2	2.99		2.66	2.9	2.88
CO3	2.99		2.66	2.9	2.88
CO4	2.99		2.66	2.9	2.88

CO-PO/PSO Mapping Table

PO/PSO		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2.88	2	2	2	1					1				2	1
CO2	2.88	2	2	2	1					1				2	1
CO3	2.88	2	2	2	1					1				2	1
CO4	2.88	2	2	2	1					1				2	1
Sum		2	2	2	1					4				8	4
Weighted Sum		23	23	23	11.5					12				23	11.5
Target Value		2	2	2	1					1				2	1
PO Attainment		1.9	1.92	1.92	0.96					1				1.92	0.96

B. K. Raj

HOD
Dept. of Computer Science &
C. S. Institute of Techn
B.G. Nagar - 571 4
Bellangala Tq. Mandy
Karnataka (INDIA

BGS Institute of Technology

VTU

Bengaluru – Hassan National Highway (NH-75), Nagamangala
Taluk, Mandya District, B G Nagar, Karnataka 571448
Ph: 08234 - 288418, Fax:288419
Email: principalbgsit@rediffmail.com, Web:www.bgsit.ac.in

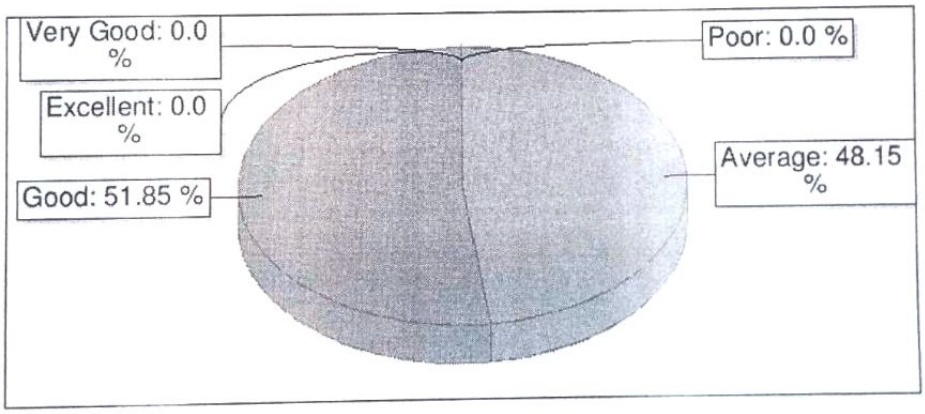
Student Feedback On Faculty 2020-21

Batch : BE , 2017-2021
Staff Name : Mr Prasanna Kumar M J
Subject Code : 17CSL76
Subject Name : MACHINE LEARNING LABORATORY

Department : Computer Science and Engineering
Semester 7 , Sec : A , Batch: 1
Date : 12 Feb 2021

No	Questions	Poor	Average	Good	Very Good	Excellent	Feedback Percentage	Average Score (5)
		1	2	3	4	5		
	Time Sense							
1	Teacher conducts the classes regularly	0	3	4	0	0	51.4	2.6
2	Syllabus of this course is completed in time	0	4	3	0	0	48.6	2.4
3	Assignments, class tests, quizzes and seminars were conducted as per schedule	0	3	4	0	0	51.4	2.6
4	Alternate arrangements were made during his/her absence	0	3	4	0	0	51.4	2.6
	Class Control/Management							
1	The faculty is effective in controlling and conducting the class	0	3	4	0	0	51.4	2.6
2	The faculty invites student participation.	0	4	3	0	0	48.6	2.4
3	The faculty rightfully addresses inappropriate behaviour of students	0	3	4	0	0	51.4	2.6
4	The faculty has a tendency of inviting opinion and questions on subject matter from students	0	3	4	0	0	51.4	2.6
5	The faculty enhances learning by judicious reinforcement mechanism	0	4	3	0	0	48.6	2.4
	Subject Command							
1	The faculty focused on the defined syllabus	0	3	4	0	0	51.4	2.6
2	The faculty conducted and involved students in classroom discussions	0	3	4	0	0	51.4	2.6
3	The faculty had good communication skills	0	4	3	0	0	48.6	2.4
4	The lectures were well structured	0	3	4	0	0	51.4	2.6
5	The faculty related the subject to real life applications of concepts	0	3	4	0	0	51.4	2.6
6	The faculty referred to latest developments in the fields	0	4	3	0	0	48.6	2.4
	Use of Teaching Aid							
1	The faculty used different teaching aids like PPT's, Blackboard, Overhead Projectors etc	0	4	3	0	0	48.6	2.4
2	The blackboard/whiteboard work was clear in terms of legibility, visibility and structure	0	3	4	0	0	51.4	2.6

3	The faculty used different teaching methods in conducting the class. (Example group discussion, seminars, student presentations, etc)	0	3	4	0	0	51.4	2.6
4	The faculty shared and discussed the answers to class tests or sessional tests	0	4	3	0	0	48.6	2.4
5	The faculty allowed the review of answer scripts of class tests	0	4	3	0	0	48.6	2.4
6	The faculty made sure all students are able to understand him/her.	0	3	4	0	0	51.4	2.6
	Helping Attitude							
1	The faculty has a helping attitude towards varied academic interests of students.	0	3	4	0	0	51.4	2.6
2	The faculty helps students gain access to material not readily available in text books, through e-resources, e-journals, reference books etc	0	4	3	0	0	48.6	2.4
3	The faculty has helps students facing physical, emotional and learning challenges.	0	3	4	0	0	51.4	2.6
4	The faculty's approach is towards development of professional skills among students	0	3	4	0	0	51.4	2.6
5	The faculty helps students in realizing career goals	0	3	4	0	0	51.4	2.6
6	The faculty helps students in realizing their strengths and development needs	0	4	3	0	0	48.6	2.4
	Total Count	**0**	**91**	**98**	**0**	**0**	**50.4**	**2.53**



Comments
.
Good teaching
Very Good
.
Good
Good
Good

K. K. Rag

HOD
Dept. of Computer Science & Engg
B.G.S. Institute of Technology.
B.G. Nagar - 571 448
Nagamangala Tq. Mandya Dist

BGS Institute of Technology

VTU

Bengaluru – Hassan National Highway (NH-75), Nagamangala
Taluk, Mandya District, B G Nagar, Karnataka 571448
Ph: 08234 - 288418, Fax:288419
Email: principalbgsit@rediffmail.com, Web:www.bgsit.ac.in

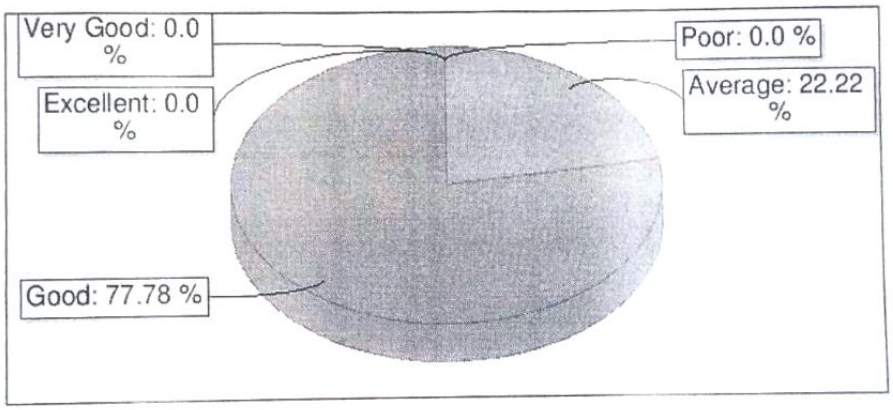
Student Feedback On Faculty 2020-21

Batch : BE , 2017-2021
Staff Name : Mr Prasanna Kumar M J
Subject Code : 17CSL76
Subject Name : MACHINE LEARNING LABORATORY

Department : Computer Science and Engineering
Semester 7 , Sec : A , Batch: 2
Date : 12 Feb 2021

No	Questions	Poor	Average	Good	Very Good	Excellent	Feedback Percentage	Average Score (5)
		1	2	3	4	5		
	Time Sense							
1	Teacher conducts the classes regularly	0	1	5	0	0	56.7	2.8
2	Syllabus of this course is completed in time	0	1	5	0	0	56.7	2.8
3	Assignments, class tests, quizzes and seminars were conducted as per schedule	0	2	4	0	0	53.3	2.7
4	Alternate arrangements were made during his/her absence	0	1	5	0	0	56.7	2.8
	Class Control/Management							
1	The faculty is effective in controlling and conducting the class	0	2	4	0	0	53.3	2.7
2	The faculty invites student participation.	0	1	5	0	0	56.7	2.8
3	The faculty rightfully addresses inappropriate behaviour of students	0	1	5	0	0	56.7	2.8
4	The faculty has a tendency of inviting opinion and questions on subject matter from students	0	3	3	0	0	50	2.5
5	The faculty enhances learning by judicious reinforcement mechanism	0	1	5	0	0	56.7	2.8
	Subject Command							
1	The faculty focused on the defined syllabus	0	1	5	0	0	56.7	2.8
2	The faculty conducted and involved students in classroom discussions	0	1	5	0	0	56.7	2.8
3	The faculty had good communication skills	0	2	4	0	0	53.3	2.7
4	The lectures were well structured	0	1	5	0	0	56.7	2.8
5	The faculty related the subject to real life applications of concepts	0	1	5	0	0	56.7	2.8
6	The faculty referred to latest developments in the fields	0	2	4	0	0	53.3	2.7
	Use of Teaching Aid							
1	The faculty used different teaching aids like PPT's, Blackboard, Overhead Projectors etc	0	1	5	0	0	56.7	2.8
2	The blackboard/whiteboard work was clear in terms of legibility, visibility and structure	0	1	5	0	0	56.7	2.8

3	The faculty used different teaching methods in conducting the class (Example group discussion, seminars, student presentations, etc.)	0	1	5	0	0	56.7	2.8
4	The faculty shared and discussed the answers to class tests or sessional tests	0	1	5	0	0	56.7	2.8
5	The faculty allowed the review of answer scripts of class tests	0	1	5	0	0	56.7	2.8
6	The faculty made sure all students are able to understand him/her	0	2	4	0	0	53.3	2.7
	Helping Attitude							
1	The faculty has a helping attitude towards varied academic interests of students.	0	2	4	0	0	53.3	2.7
2	The faculty helps students gain access to material not readily available in text books, through e-resources, e-journals, reference books etc	0	1	5	0	0	56.7	2.8
3	The faculty has helps students facing physical, emotional and learning challenges.	0	1	5	0	0	56.7	2.8
4	The faculty's approach is towards development of professional skills among students	0	2	4	0	0	53.3	2.7
5	The faculty helps students in realizing career goals	0	1	5	0	0	56.7	2.8
6	The faculty helps students in realizing their strengths and development needs	0	1	5	0	0	56.7	2.8
	Total Count	**0**	**36**	**126**	**0**	**0**	**55.6**	**2.76**



Comments
Good
They teach us very well
Good teaching
Teaching was good.
Good
Good

Make the students to do the mini project using this lab. and implement this for the next batch of students.

B. K. Raghu

HOD
Dept. of Computer Science & Engg
B.G.S. Institute of Technology.
B.G. Nagar - 571 448

Page 2 of 2

||Jai Sri Gurudev||

BGS Institute of Technology

Department of Computer Science and Engineering

Academic year 2020-21 (ODD / EVEN)	
Name of the Faculty with Designation	M. J. PRASANNA KUMARA
Course Name with code	Machine Learning Lab 17CSL76

Feed Back Report			
No. of Students participated	29	**Overall Feedback**	98%

Course End Survey						
CO's	CO.1	CO.2	CO.3	CO.4	CO.5	CO.6
Av. Rating	2.9	2.9	2.9	2.9		

CO Attainment						
CO's	CO.1	CO.2	CO.3	CO.4	CO.5	CO.6
Attainment	2.88	2.88	2.88	2.88		

PO / PSO Attainment														
PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
Attainment	1.92	1.92	1.92	0.96					0.96				1.92	0.96

Analysis of CO, PO/PSO Attainment [Review of attainment (course attainment)]

CO, PO/PSO Attainment of the lab is satisfactory

B. K. Raghu

HOD

Dept. of Computer Science & Engg.
B.G.S. Institute of Technology,
B.G. Nagar - 571 448.
Nagamangala Tq, Mandya Dist
Karnataka (INDIA)

||Jai Sri Gurudev||

BGS Institute of Technology

Department of Computer Science and Engineering

Result Analysis CIE				
	Test-1	Test-2	Test-3	IA Final
22 (≥76%)				86
12-22 (≥41% ≤75%)				
12 (≤40%)				
Total No of Students				86

Action taken for Slow learners:

Test-1

Test-2

Result Analysis SEE					
Course name with Code	Total Appeared	FCD	FC	Pass %	Failed
Machine Learning Lab 17CS76	86	85	1	100%	–

Remarks: Performance is good and satisfactory

M.J Prof

Faculty

B. K. Raghu

HOD

Dept. of Computer Science & Engg.
B.G.S. Institute of Technology

BGS Institute of Technology

VTU

Bengaluru – Hassan National Highway (NH-75), Nagamangala
Taluk, Mandya District, B G Nagar, Karnataka 571448
Ph: 08234 - 288418, Fax:288419
Email: principalbgsit@rediffmail.com, Web:www.bgsit.ac.in

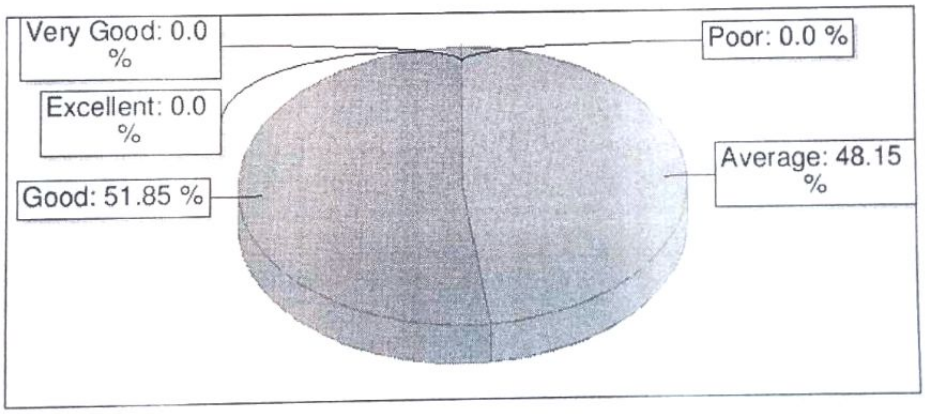
Student Feedback On Faculty 2020-21

Batch : BE , 2017-2021
Staff Name : Mr Prasanna Kumar M J
Subject Code : 17CSL76
Subject Name : MACHINE LEARNING LABORATORY

Department : Computer Science and Engineering
Semester 7 , Sec : A , Batch: 1
Date : 12 Feb 2021

No	Questions	Poor	Average	Good	Very Good	Excellent	Feedback Percentage	Average Score (5)
		1	2	3	4	5		
	Time Sense							
1	Teacher conducts the classes regularly	0	3	4	0	0	51.4	2.6
2	Syllabus of this course is completed in time	0	4	3	0	0	48.6	2.4
3	Assignments, class tests, quizzes and seminars were conducted as per schedule	0	3	4	0	0	51.4	2.6
4	Alternate arrangements were made during his/her absence	0	3	4	0	0	51.4	2.6
	Class Control/Management							
1	The faculty is effective in controlling and conducting the class	0	3	4	0	0	51.4	2.6
2	The faculty invites student participation.	0	4	3	0	0	48.6	2.4
3	The faculty rightfully addresses inappropriate behaviour of students	0	3	4	0	0	51.4	2.6
4	The faculty has a tendency of inviting opinion and questions on subject matter from students	0	3	4	0	0	51.4	2.6
5	The faculty enhances learning by judicious reinforcement mechanism	0	4	3	0	0	48.6	2.4
	Subject Command							
1	The faculty focused on the defined syllabus	0	3	4	0	0	51.4	2.6
2	The faculty conducted and involved students in classroom discussions	0	3	4	0	0	51.4	2.6
3	The faculty had good communication skills	0	4	3	0	0	48.6	2.4
4	The lectures were well structured	0	3	4	0	0	51.4	2.6
5	The faculty related the subject to real life applications of concepts	0	3	4	0	0	51.4	2.6
6	The faculty referred to latest developments in the fields	0	4	3	0	0	48.6	2.4
	Use of Teaching Aid							
1	The faculty used different teaching aids like PPT's, Blackboard, Overhead Projectors etc	0	4	3	0	0	48.6	2.4
2	The blackboard/whiteboard work was clear in terms of legibility, visibility and structure	0	3	4	0	0	51.4	2.6

3	The faculty used different teaching methods in conducting the class. (Example group discussion, seminars, student presentations, etc)	0	3	4	0	0	51.4	2.6
4	The faculty shared and discussed the answers to class tests or sessional tests	0	4	3	0	0	48.6	2.4
5	The faculty allowed the review of answer scripts of class tests	0	4	3	0	0	48.6	2.4
6	The faculty made sure all students are able to understand him/her.	0	3	4	0	0	51.4	2.6
	Helping Attitude							
1	The faculty has a helping attitude towards varied academic interests of students.	0	3	4	0	0	51.4	2.6
2	The faculty helps students gain access to material not readily available in text books, through e-resources, e-journals, reference books etc	0	4	3	0	0	48.6	2.4
3	The faculty has helps students facing physical, emotional and learning challenges.	0	3	4	0	0	51.4	2.6
4	The faculty's approach is towards development of professional skills among students	0	3	4	0	0	51.4	2.6
5	The faculty helps students in realizing career goals	0	3	4	0	0	51.4	2.6
6	The faculty helps students in realizing their strengths and development needs	0	4	3	0	0	48.6	2.4
	Total Count	**0**	**91**	**98**	**0**	**0**	**50.4**	**2.53**



Comments
.
Good teaching
Very Good
.
Good
Good
Good

K. K. Rag

HOD
Dept. of Computer Science & Engg
B.G.S. Institute of Technology.
B.G. Nagar - 571 448
Nagamangala Tq. Mandya Dist

BGS Institute of Technology

VTU

Bengaluru – Hassan National Highway (NH-75), Nagamangala
Taluk, Mandya District, B G Nagar, Karnataka 571448
Ph: 08234 - 288418, Fax:288419
Email: principalbgsit@rediffmail.com, Web:www.bgsit.ac.in

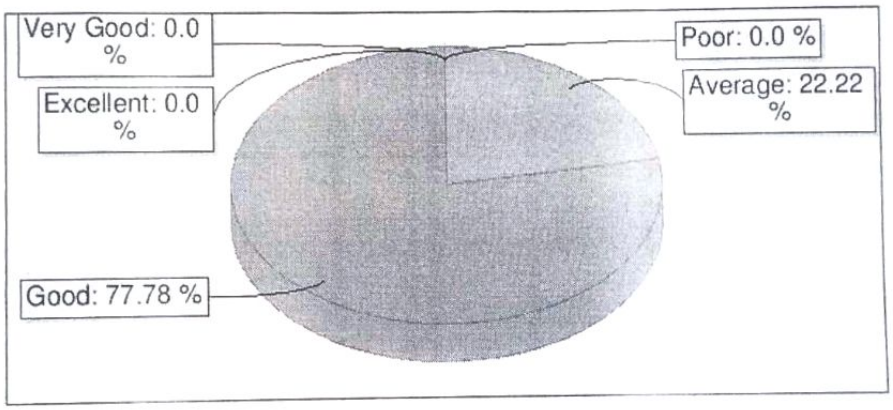
Student Feedback On Faculty 2020-21

Batch : BE , 2017-2021
Staff Name : Mr Prasanna Kumar M J
Subject Code : 17CSL76
Subject Name : MACHINE LEARNING LABORATORY

Department : Computer Science and Engineering
Semester 7 , Sec : A , Batch: 2
Date : 12 Feb 2021

No	Questions	Poor	Average	Good	Very Good	Excellent	Feedback Percentage	Average Score (5)
		1	2	3	4	5		
	Time Sense							
1	Teacher conducts the classes regularly	0	1	5	0	0	56.7	2.8
2	Syllabus of this course is completed in time	0	1	5	0	0	56.7	2.8
3	Assignments, class tests, quizzes and seminars were conducted as per schedule	0	2	4	0	0	53.3	2.7
4	Alternate arrangements were made during his/her absence	0	1	5	0	0	56.7	2.8
	Class Control/Management							
1	The faculty is effective in controlling and conducting the class	0	2	4	0	0	53.3	2.7
2	The faculty invites student participation.	0	1	5	0	0	56.7	2.8
3	The faculty rightfully addresses inappropriate behaviour of students	0	1	5	0	0	56.7	2.8
4	The faculty has a tendency of inviting opinion and questions on subject matter from students	0	3	3	0	0	50	2.5
5	The faculty enhances learning by judicious reinforcement mechanism	0	1	5	0	0	56.7	2.8
	Subject Command							
1	The faculty focused on the defined syllabus	0	1	5	0	0	56.7	2.8
2	The faculty conducted and involved students in classroom discussions	0	1	5	0	0	56.7	2.8
3	The faculty had good communication skills	0	2	4	0	0	53.3	2.7
4	The lectures were well structured	0	1	5	0	0	56.7	2.8
5	The faculty related the subject to real life applications of concepts	0	1	5	0	0	56.7	2.8
6	The faculty referred to latest developments in the fields	0	2	4	0	0	53.3	2.7
	Use of Teaching Aid							
1	The faculty used different teaching aids like PPT's, Blackboard, Overhead Projectors etc	0	1	5	0	0	56.7	2.8
2	The blackboard/whiteboard work was clear in terms of legibility, visibility and structure	0	1	5	0	0	56.7	2.8

3	The faculty used different teaching methods in conducting the class (Example group discussion, seminars, student presentations, etc.)	0	1	5	0	0	56.7	2.8
4	The faculty shared and discussed the answers to class tests or sessional tests	0	1	5	0	0	56.7	2.8
5	The faculty allowed the review of answer scripts of class tests	0	1	5	0	0	56.7	2.8
6	The faculty made sure all students are able to understand him/her	0	2	4	0	0	53.3	2.7
	Helping Attitude							
1	The faculty has a helping attitude towards varied academic interests of students.	0	2	4	0	0	53.3	2.7
2	The faculty helps students gain access to material not readily available in text books, through e-resources, e-journals, reference books etc	0	1	5	0	0	56.7	2.8
3	The faculty has helps students facing physical, emotional and learning challenges.	0	1	5	0	0	56.7	2.8
4	The faculty's approach is towards development of professional skills among students	0	2	4	0	0	53.3	2.7
5	The faculty helps students in realizing career goals	0	1	5	0	0	56.7	2.8
6	The faculty helps students in realizing their strengths and development needs	0	1	5	0	0	56.7	2.8
	Total Count	**0**	**36**	**126**	**0**	**0**	**55.6**	**2.76**



Comments
Good
They teach us very well
Good teaching
Teaching was good.
Good
Good

Make the students to do the mini project using this lab. and implement this for the next batch of students.

B. K. Raghu

HOD
Dept. of Computer Science & Engg
B.G.S. Institute of Technology.
B.G. Nagar - 571 448

Page 2 of 2

BGS Institute of Technology

Department of Computer science & Engineering

INTERNAL AUDITING

DATE: 12/01/2021

Name of the Faculty:	M. J. Prasanna Kumar
Designation:	Asst professor
Subject Name with Code	Machine Learning 17CS73 and machine Learning Lab 17CS76

SL. No.	Contents	(2020-21) ODD Theory	Lab
1	Faculty Profile	✓	✓
2	Vision and Mission of the Institute	✓	✓
3	Vision and Mission of the Department	✓	✓
4	Department PEO's and PSOs,	✓	✓
5	Course Outcome	✓	✓
6	Mapping of COs and POs, PEOs, PSOs	✓	✓
7	Assessment Tools and Procedure for Assessment of Cos (IA Test, Quiz, Surprise test, Assignment, University Examination)	✓	✓
8	Previous University Question Papers	✓	X
9	COE of Institute and COE of the Department (COE= Calendar of Events)	✓	✓
10	Time Table (Class and Individual)	✓	✓
11	Course Plan (Syllabus Copy along with CO and hours)	✓	✓
12	List of Text and Reference Books	✓	✓
13	Lesson Plan	✓	✓
14	Batch wise Assignments Batchwise assignment needs to be given	X	X
15	Students Roll Call with phone numbers (Procter Details batch wise)	X	X
16	Report of Guest Lectures Needs to conduct guest lecture	X	X
17	Notes	✓	X
18	Question Bank	✓	✓
19	FEED Back Report (Mid of the semester & End of the Semester)	✓	✓
20	Communications with Faculty and Students	✓	✓
21	Academic Diary	✓	✓
22	Course end survey	✓	✓

Signature Of External Auditor

Signature Of Academic Incharge

B. K. R. HOD
Dept. of Computer Science & Engg.
B.G.S. Institute of Technology
B.G. Nagar - 571 448
Mandya Dist.

Signature Of HoD

Signature Of principal
Principal
BGSIT

||Jai Sri Gurudev||

BGS Institute of Technology

Department of Computer Science and Engineering

Academic year 2020-21 (ODD / EVEN)	
Name of the Faculty with Designation	M. J. PRASANNA KUMARA
Course Name with code	Machine Learning Lab 17CSL76

Feed Back Report			
No. of Students participated	29	**Overall Feedback**	98%

Course End Survey						
CO's	CO.1	CO.2	CO.3	CO.4	CO.5	CO.6
Av. Rating	2.9	2.9	2.9	2.9		

CO Attainment						
CO's	CO.1	CO.2	CO.3	CO.4	CO.5	CO.6
Attainment	2.88	2.88	2.88	2.88		

PO / PSO Attainment														
PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
Attainment	1.92	1.92	1.92	0.96					0.96				1.92	0.96

Analysis of CO, PO/PSO Attainment [Review of attainment (course attainment)]

CO, PO/PSO Attainment of the lab is satisfactory

B. K. Raghu

HOD

Dept. of Computer Science & Engg.
B.G.S. Institute of Technology,
B.G. Nagar - 571 448.
Nagamangala Tq, Mandya Dist
Karnataka (INDIA)

||Jai Sri Gurudev||

BGS Institute of Technology

Department of Computer Science and Engineering

Result Analysis CIE				
	Test-1	Test-2	Test-3	IA Final
22 (≥76%)				86
12-22 (≥41% ≤75%)				
12 (≤40%)				
Total No of Students				86

Action taken for Slow learners:

Test-1

Test-2

Result Analysis SEE					
Course name with Code	Total Appeared	FCD	FC	Pass %	Failed
Machine Learning Lab 17CS76	86	85	1	100%	–

Remarks: Performance is good and satisfactory

M.J Prof

Faculty

B. K. Raghu

HOD

Dept. of Computer Science & Engg.
B.G.S. Institute of Technology