GOVERNMENT OF TAMILNADU

Naan Mudhalvan - Project-Based Experiential Learning

Intelligent Admission-The Future Of University Decision Making With Machine Learning

Submitted by

Team ID : NM2023TMID22128

MOHANAPRIYA.K(20326ER018)

MEGALADEVI.G(20326ER017)

NANCY BEAULA.V(20326ER020)

PAVITHRA.U(20326ER021)

Under the guidance of

Mrs. J. SUKANYA, MCA., M.Phil.,

Assistant Professor

PG and Research Department of Computer Science

M.V.MUTHIAH GOVERNMENT ARTS COLLEGE FOR WOMEN

(Affiliated to Mother Teresa Women's University, Kodaikanal)

Reaccredited with 'A' Grade by NAAC

DINDIGUL-624001.

APRIL - 2023

M.V. MUTHIAH GOVERNMENT ARTS COLLEG FOR WOMEN
(Affiliated to Mother Teresa Women's University, Kodaikanal)
Reaccredited with 'A' Grade by NAAC
Dindigul - 624 001

PG & RESEARCH DEPARTMENT OF COMPUTER SCIENCE

BONAFIDE CERTIFICATE

This is to certify that this is a bonafide record of the project entitled, **University Admission The Future Of University Decision Making With Machine Learning** done by **MOHANAPRIYA. K(20326ER018), MEGALADEVI. G(20326ER017), NANCY BEAULA.V(20326ER020), PAVITHRA.U(20326ER021).** This is submitted in partial fulfillment for the award of the degree of **Bachelor of Science in Computer Science in M.V.MUTHIAH GOVERNMENT ARTS COLLEGE FOR WOMEN, DINDIGUL** during the period of December 2022 to April 2023.

Project Mentor(s) **Head of the Department**

Submitted for Viva-Voce Examination held on 11.04.2023

1.INTRODUCTION:

1.1.Overview:

University admission is the process by which students are selected to attend a college or university. The process typically involves several steps, including submitting an application, taking entrance exams, and participating in interviews or other evaluations.

Students are often worried about their chances of admission in University. The university admission process for students can be demanding, but by being well-informed, prepared, and organized, students can increase their chances of being admitted to the university of their choice.

With this project, students can make more informed decisions about which universities to apply to, and universities can make more efficient use of their resources by focusing on the most promising applicants.

The predicted output gives them a fair idea about their admission chances in a particular university. This analysis should also help students who are currently preparing or will be preparing to get a better idea.

1.2.Purpose:

Intelligent Admission The future of university decision making with machine learning is a technique that involves the use of advanced analytical algorithms and statistical models to analyze and interpret the data related to the admissions.

This technique is useful for identifying the patterns and trends in the data, which can provide valuable insights into the students that influence the placement of various university.

Machine learning algorithm can be used to analyze the data related to students academic performance,their technical skills, and their performance in interviews. This analysis should also help students who are currently preparing or will be preparing to get a better idea.

2.PROBLEM DEFINITION&DESIGN THINKING

2.1.Empathy Map:

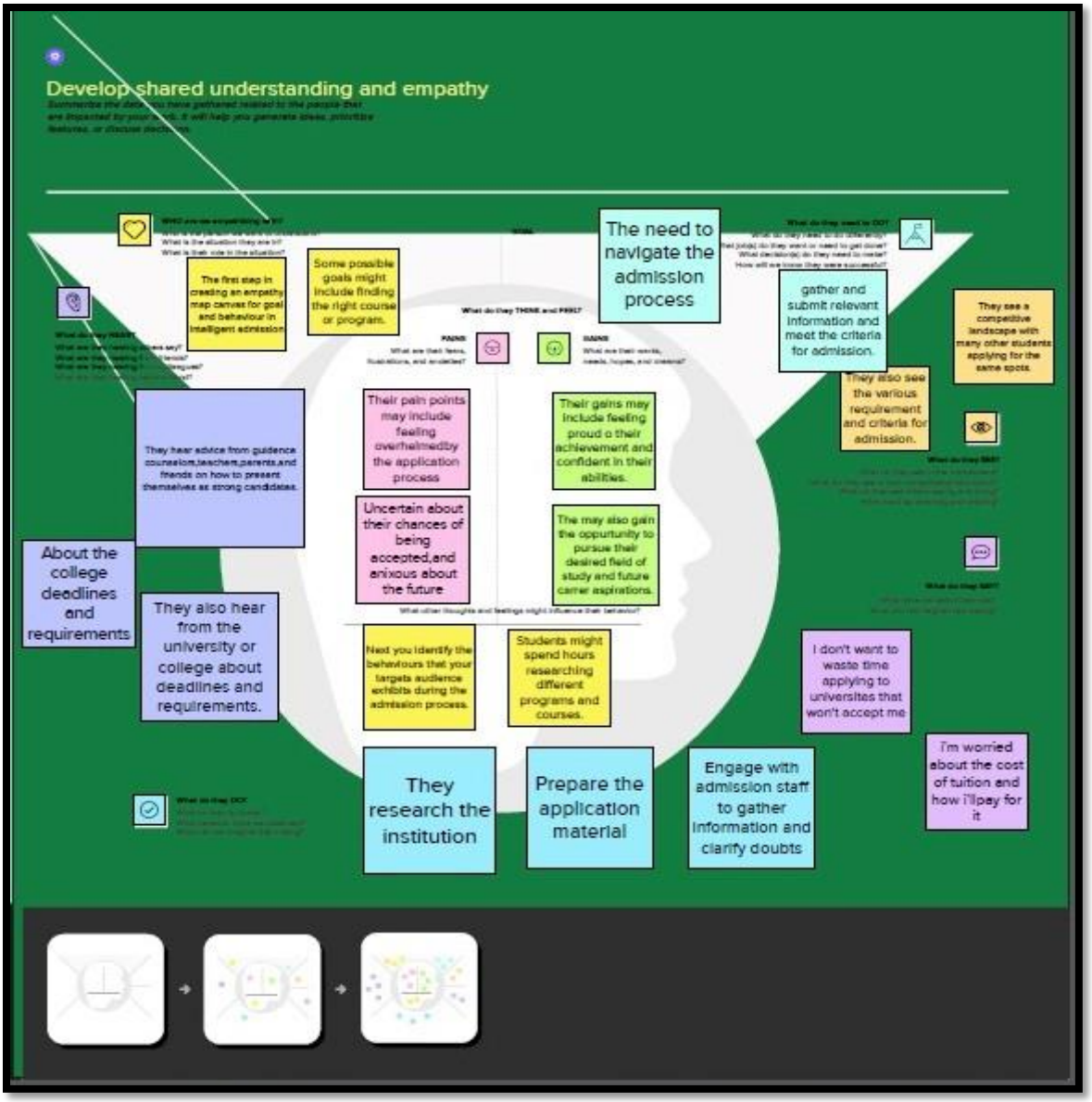
Empathy for Identifying Patterns and Trends in Campus Placement Data using Machine Learning.Empathy in this case can refer to the ability of the machine learning algorithm to understand the context and nuances of the data it is analyzing.

This includes understanding the factors that may impact the placement of students, such as their academic performance, background, and the current job market. To develop empathy in a machine learning algorithm, it is important to train the model on a diverse set of data that includes a wide range of backgrounds and experiences.

This will help the algorithm to identify patterns and trends that may not be immediately obvious, and to recognize the impact of different university outcomes.

In addition to training the algorithm on diverse data, it is also important to include feedback mechanisms that allow the algorithm to learn from its mistakes and adjust its approach over time.

This can help the algorithm to continually refine its understanding of the data and improve its ability to identify meaningful patterns and trends


Develop shared understanding and empathy
The first step in creating an empathy map canvas for goal and behaviour in intelligent admission
Some possible goals might include finding the right course or program.
The need to navigate the admission process
gather and submit relevant information and meet the criteria for admission.
They see a competitive landscape with many other students applying for the same spots.
They also see the various requirement and criteria for admission.
They hear advice from guidance counselors,teachers,parents,and friends on how to present themselves as strong candidates.
Their pain points may include feeling overhelmedby the application process
Their gains may include feeling proud o their achievement and confident in their abilities.
Uncertain about their chances of being accepted,and anxious about the future
The may also gain the oppurtunity to pursue their desired field of study and future carrer aspirations.
About the college deadlines and requirements
They also hear from the university or college about deadlines and requirements.
Next you identify the behaviours that your targets audience exhibits during the admission process.
Students might spend hours researching different programs and courses.
I don't want to waste time applying to universites that won't accept me
I'm worried about the cost of tuition and how i'llpay for it
They research the institution
Prepare the application material
Engage with admission staff to gather information and clarify doubts

2.2.Ideation&Brainstorming map:

Brainstorm Map for the future of university decision making with machine learning.

Here is a brainstorm map that outlines some possible strategies for the future of university decision making with machine learning.

Data preprocessing: Clean and normalize the data by removing duplicates, handling missing values, and transforming variables to be consistent across the dataset.

Feature selection: Identify the most important features that impact placement outcomes, such as academic performance, background, and job market trends.

Dimensionality reduction: Reduce the dimensionality of the data by using techniques such as principal component analysis (PCA) or t-distributed stochastic neighbor embedding (t-SNE) to visualize the data in a lower-dimensional space.

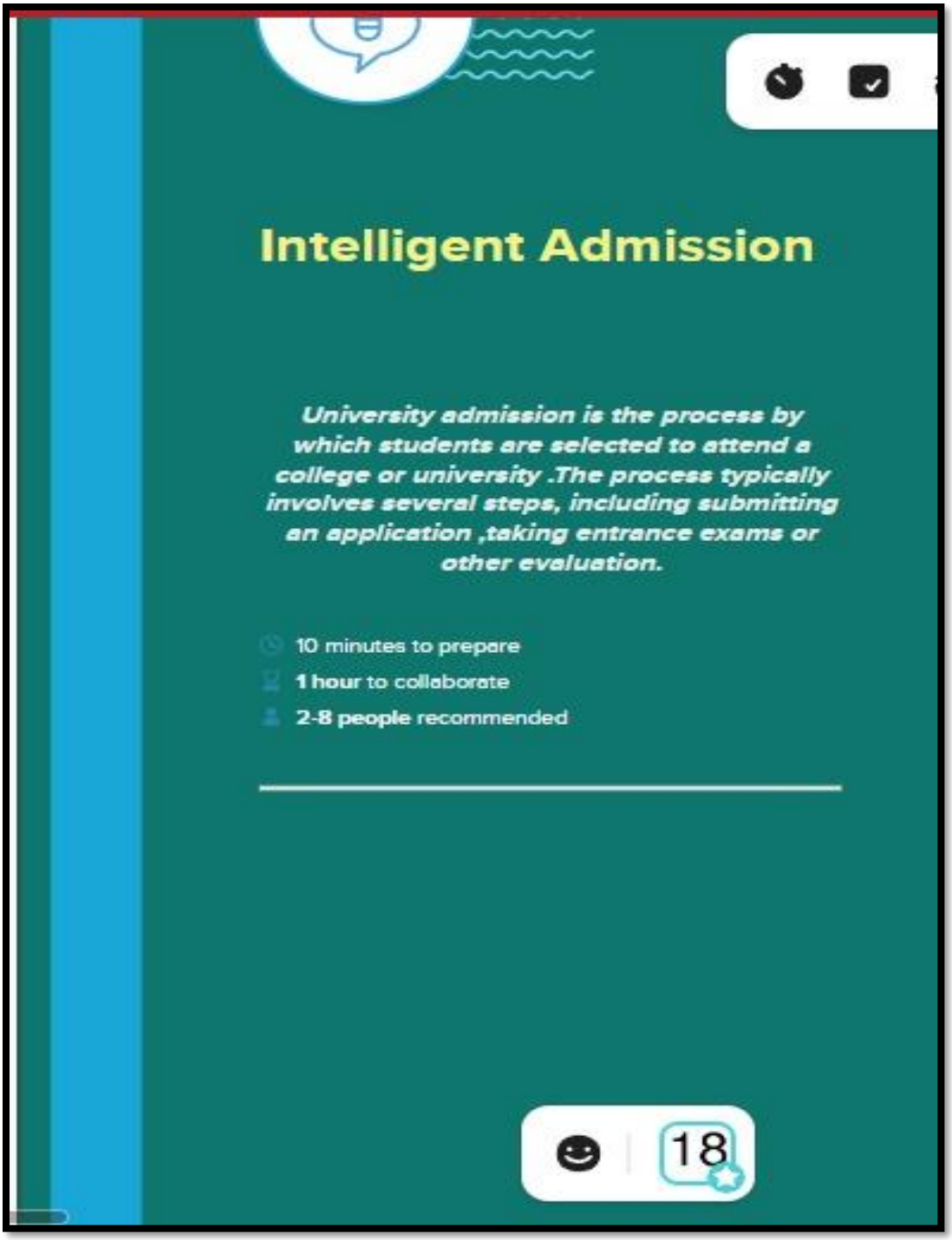
Algorithm selection: Select the most appropriate machine learning algorithm for the task at hand, such as decision trees, random forests, or neural networks.

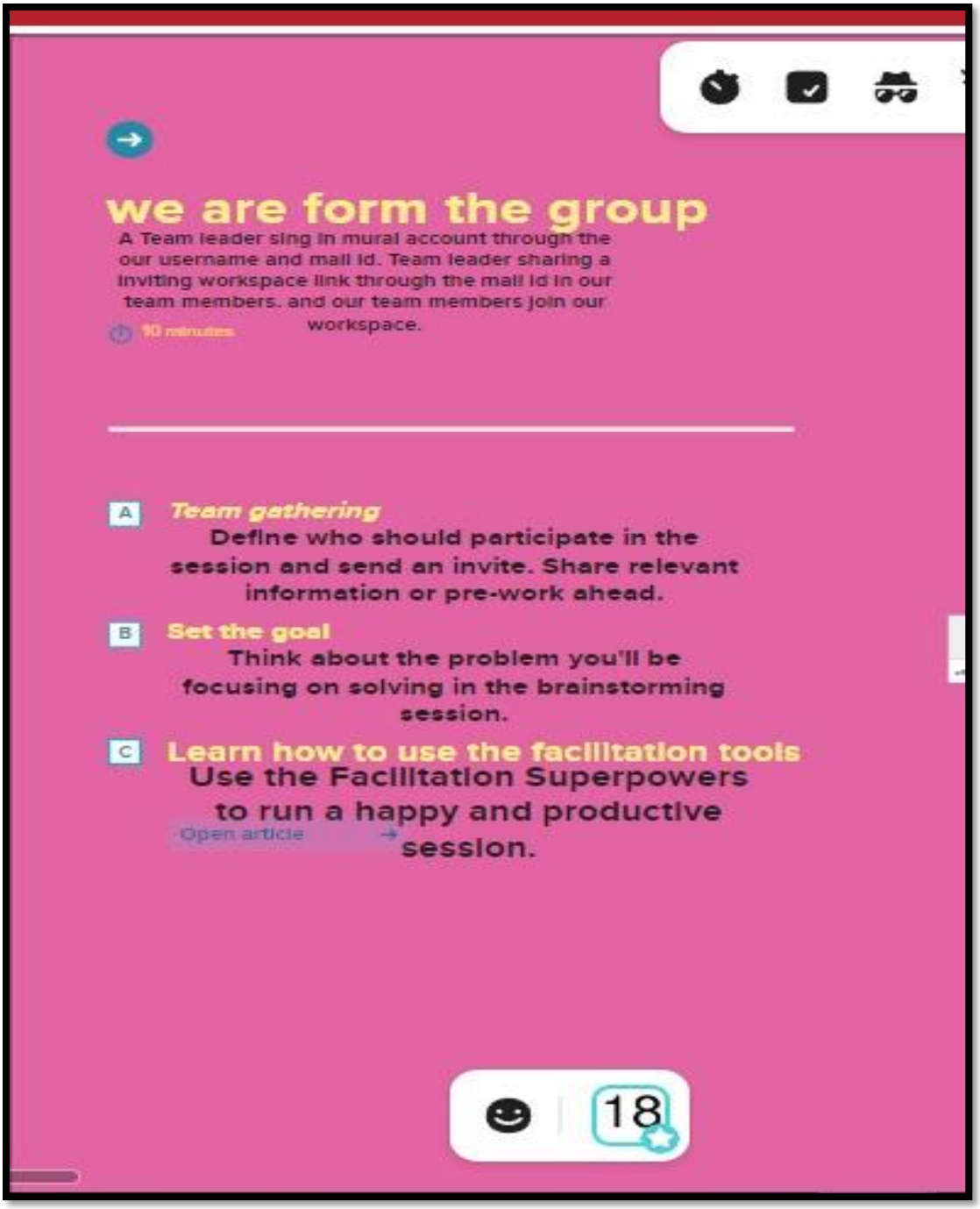
Model training: Train the model on a diverse set of data that includes a wide range of backgrounds and experiences, and use techniques such as cross-validation to ensure that the model is generalizable to new data.

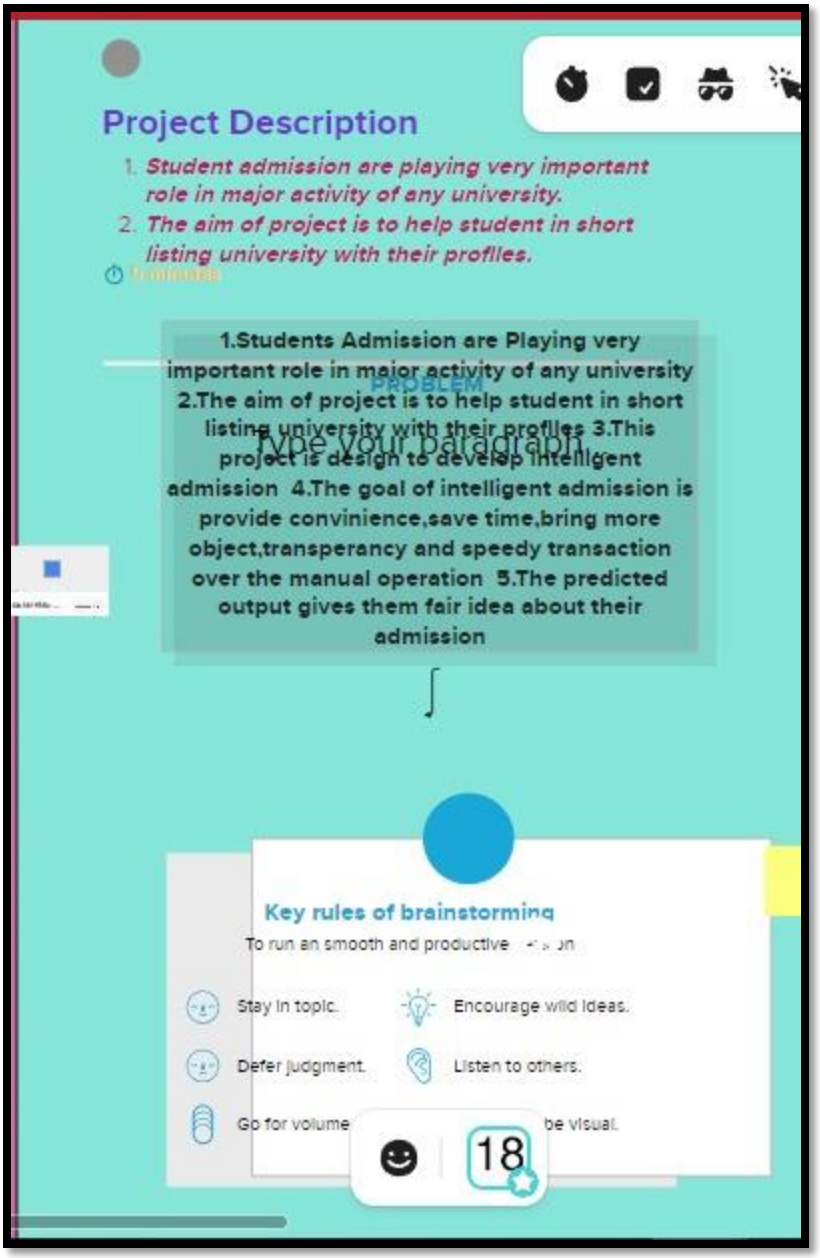
Model evaluation: Evaluate the performance of the model using metrics such as accuracy, precision, recall, and F1 score, and use techniques such as confusion matrices and ROC curves to visualize the results.

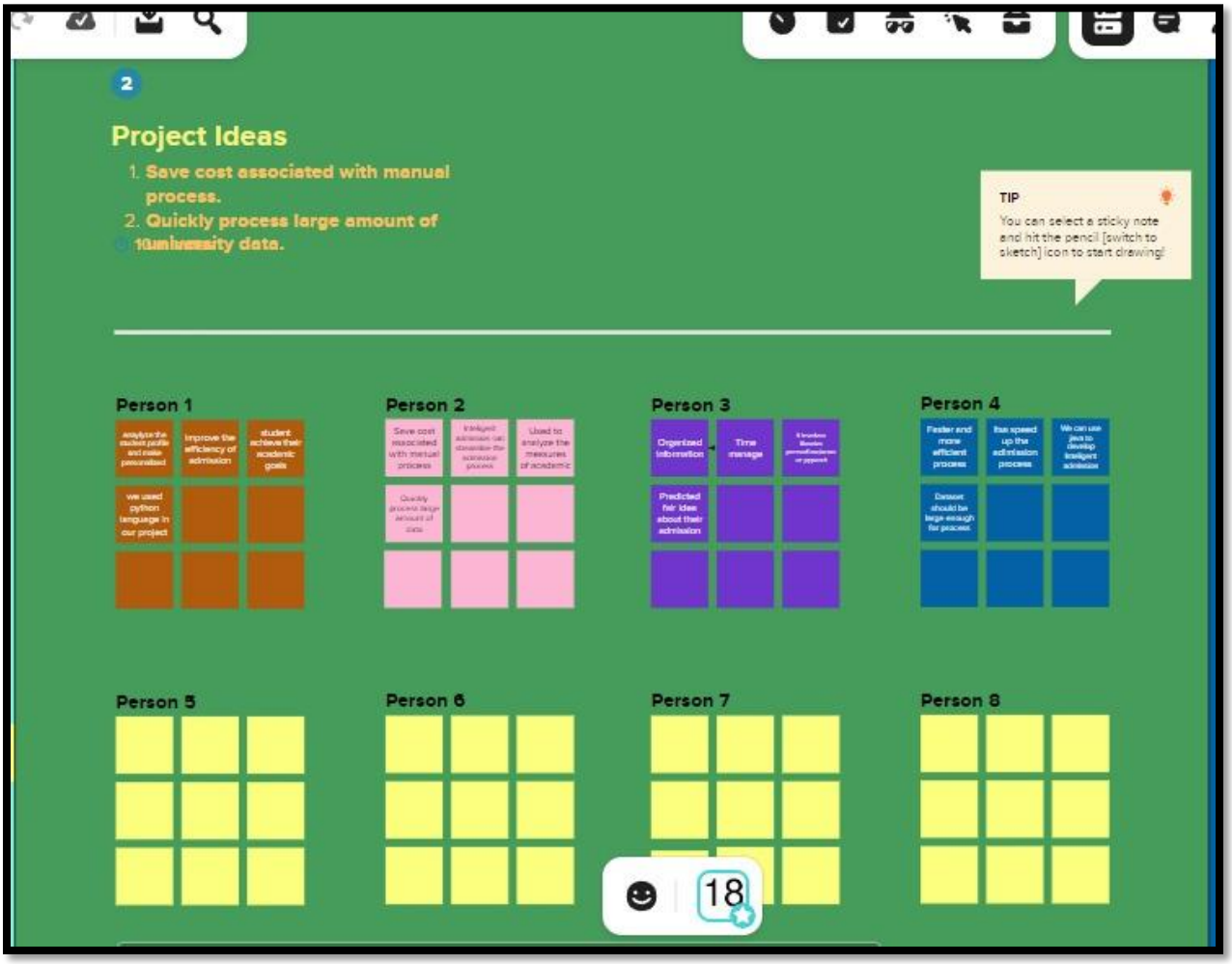
Interpretation: Interpret the results of the model by analyzing the most important features and identifying the key factors that impact placement outcomes.

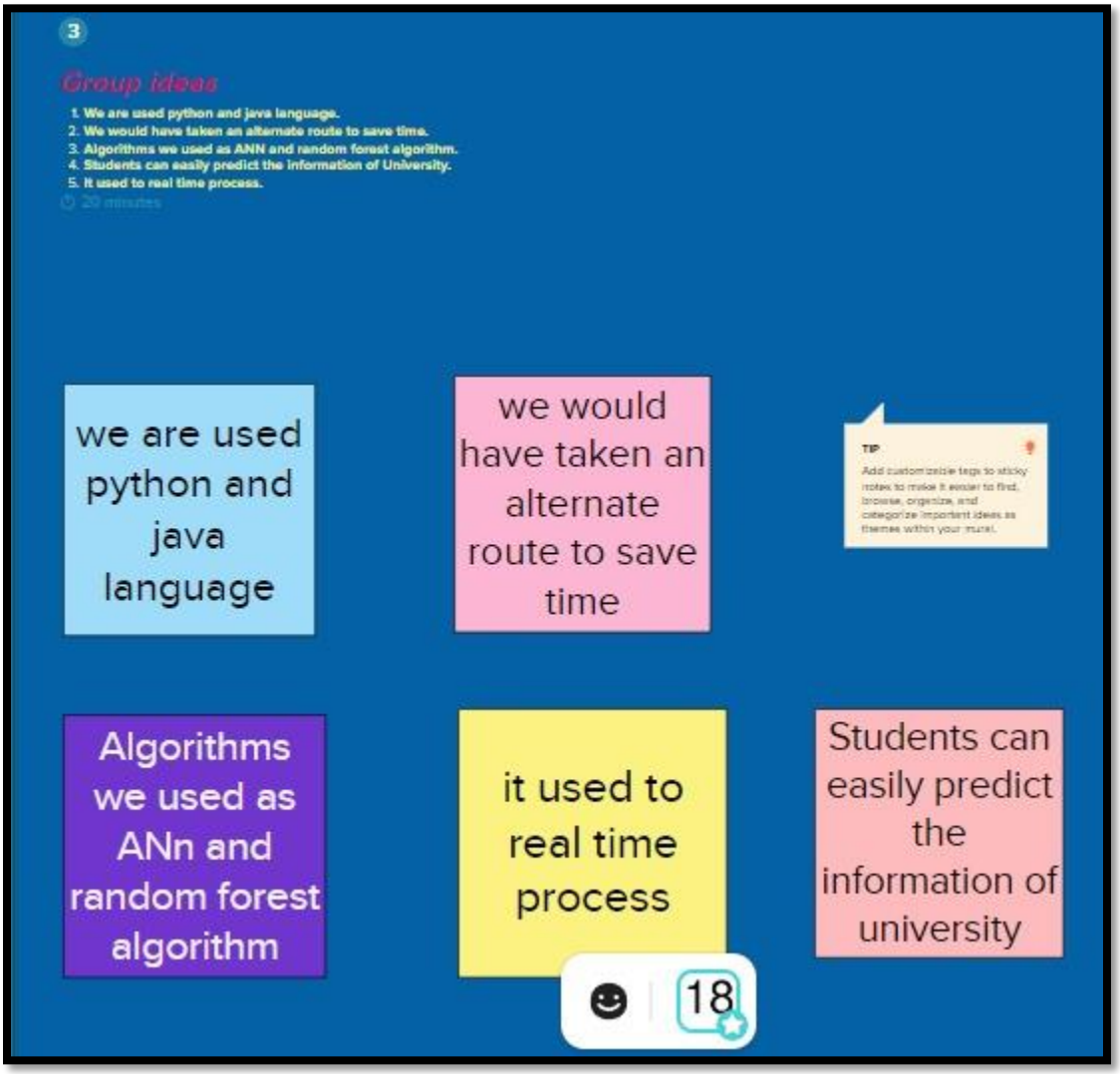
Visualization: Visualize the results of the model using techniques such as heatmaps, scatter plots, and bar charts to identify patterns and trends in the data

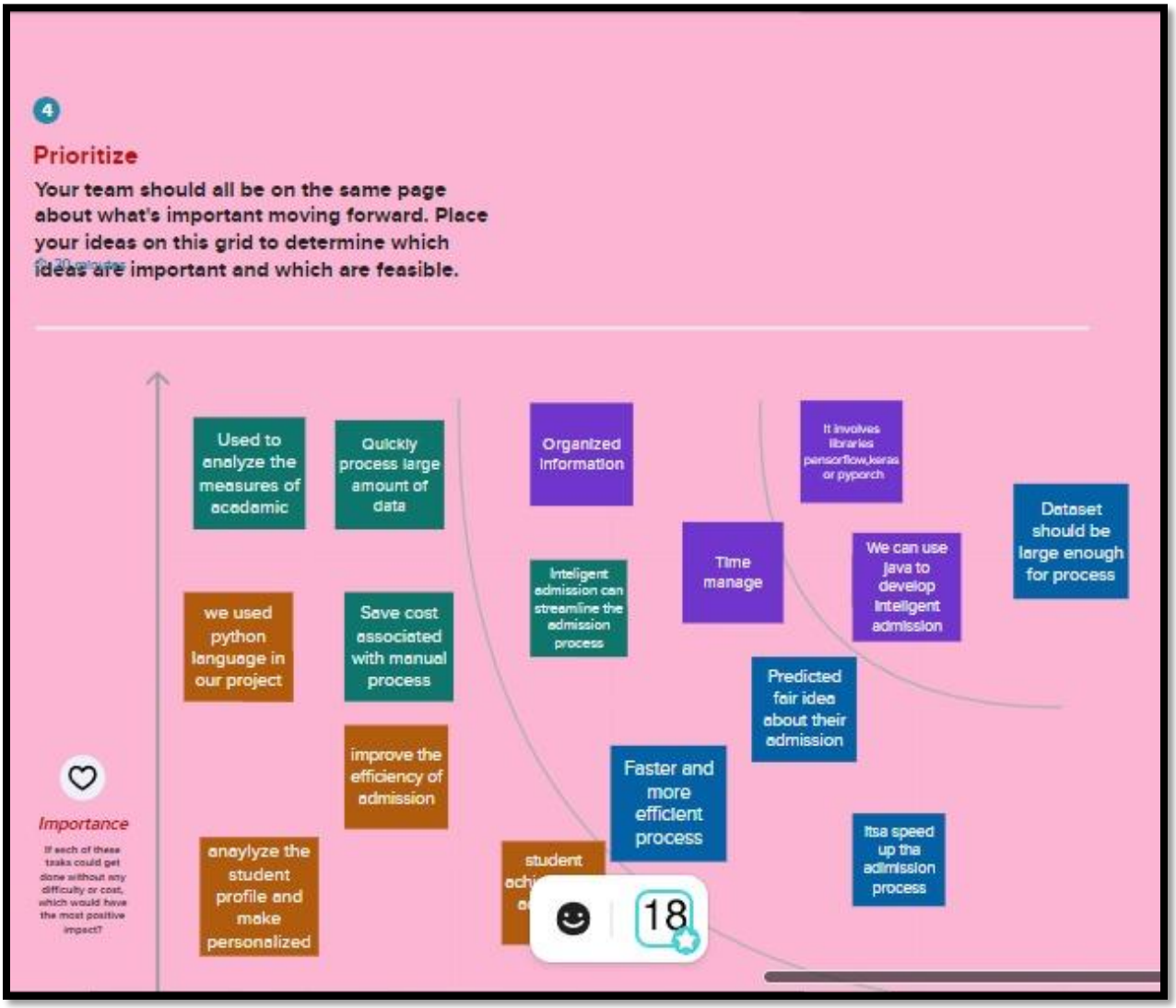

Intelligent Admission
University admission is the process by which students are selected to attend a college or university .The process typically involves several steps, including submitting an application ,taking entrance exams or other evaluation.
10 minutes to prepare
1 hour to collaborate
2-8 people recommended
18

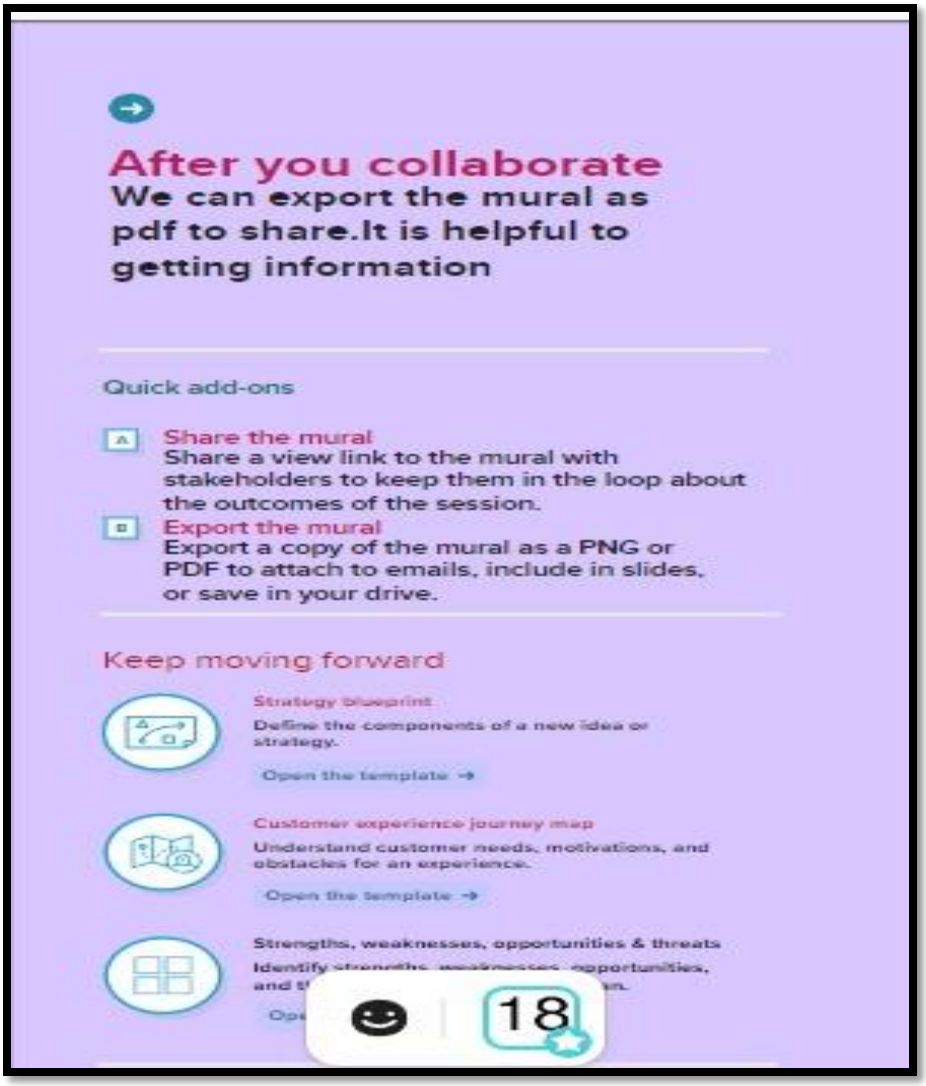

we are form the group
A Team leader sing in mural account through the our username and mail id. Team leader sharing a inviting workspace link through the mail id in our team members. and our team members join our workspace.
10 minutes
A
Team gathering
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.
B
Set the goal
Think about the problem you'll be focusing on solving in the brainstorming session.
C
Learn how to use the facilitation tools
Use the Facilitation Superpowers to run a happy and productive session.
Open article
18


Project Description
1. Student admission are playing very important role in major activity of any university.
2. The aim of project is to help student in short listing university with their profiles.
1.Students Admission are Playing very important role in major activity of any university 2.The aim of project is to help student in short listing university with their profiles 3.This project is design to develop intelligent admission 4.The goal of intelligent admission is provide convinience,save time,bring more object,transperancy and speedy transaction over the manual operation 5.The predicted output gives them fair idea about their admission
PROBLEM
Type your paragraph
Key rules of brainstorming
To run an smooth and productive
Stay in topic.
Encourage wild ideas.
Defer judgment.
Listen to others.
Go for volume
be visual.
18

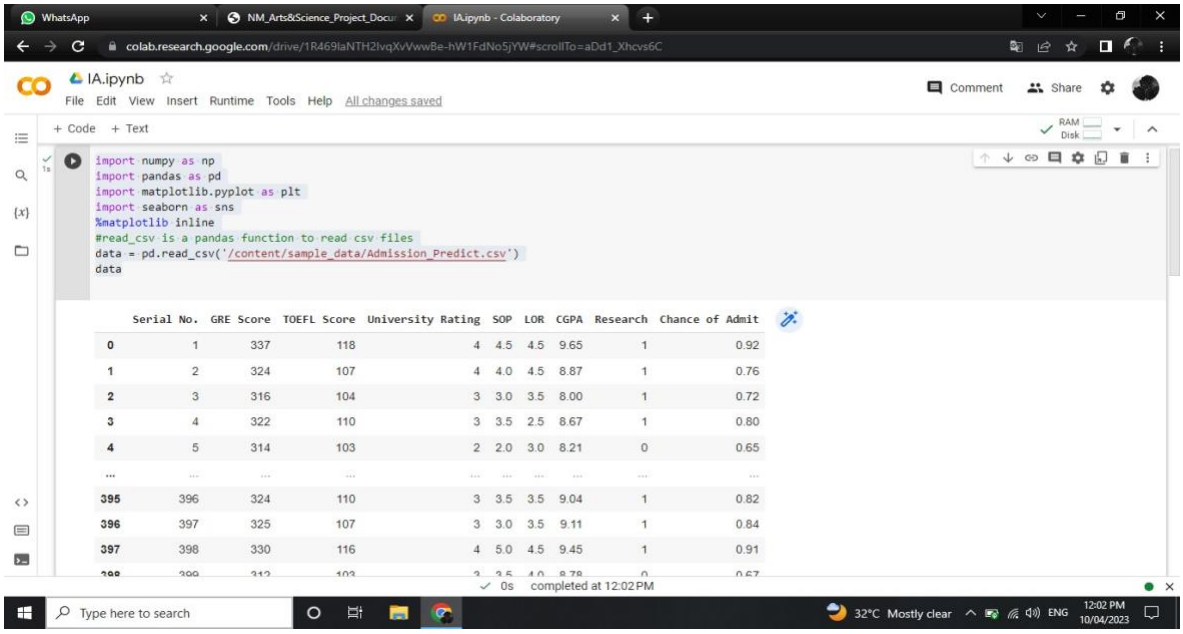

2
Project Ideas
1. Save cost associated with manual process.
2. Quickly process large amount of university data.
TIP
You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!
Person 1
analyze the student profile and make personalized
improve the efficiency of admission
student achieve their academic goals
we used python language in our project
Person 2
Save cost associated with manual process
Intelligent admission can streamline the admission process
Used to analyze the measures of academic
Quickly process large amount of data
Person 3
Organized information
Time manage
Predicted fair idea about their admission
Person 4
Faster and more efficient process
Can speed up the admission process
We can use java to develop Intelligent admission
Dataset should be large enough for process
Person 5
Person 6
Person 7
Person 8
18

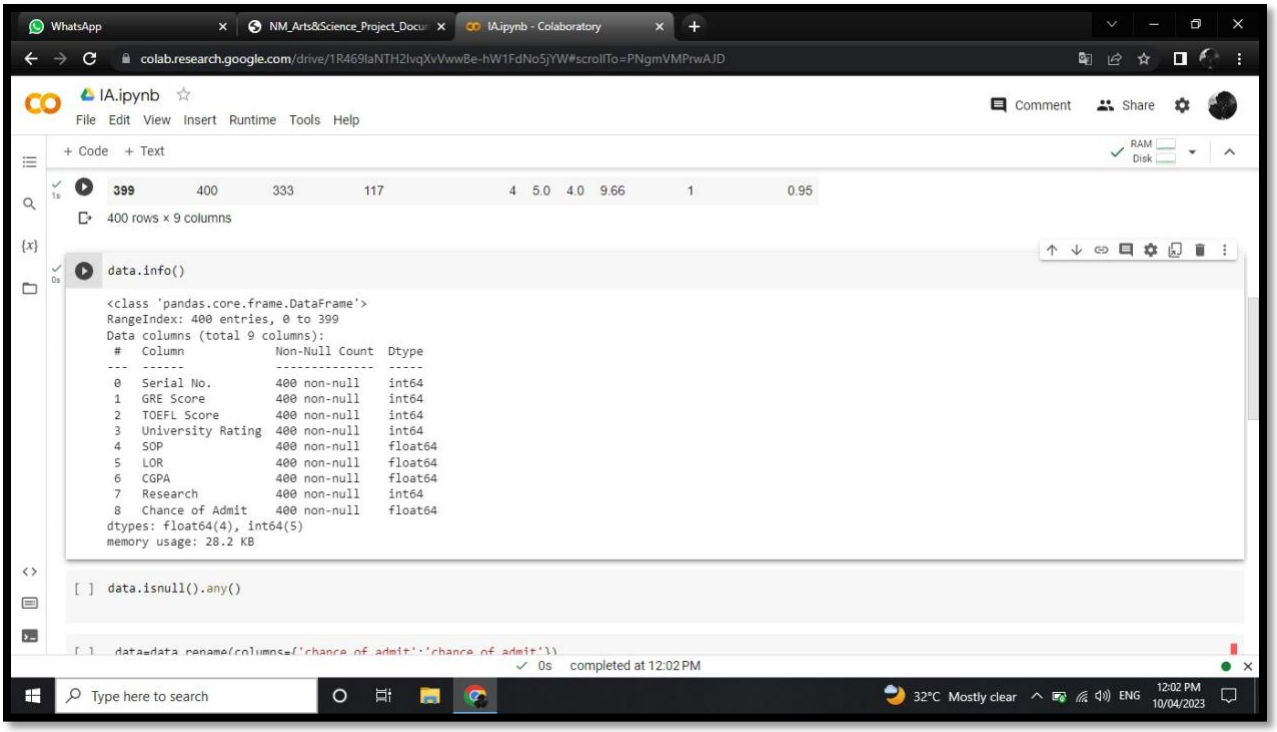

3
Group ideas
1. We are used python and java language.
2. We would have taken an alternate route to save time.
3. Algorithms we used as ANN and random forest algorithm.
4. Students can easily predict the information of University.
5. It used to real time process.
20 minutes
we are used python and java language
we would have taken an alternate route to save time
TIP
Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.
Algorithms we used as ANn and random forest algorithm
it used to real time process
Students can easily predict the information of university
18

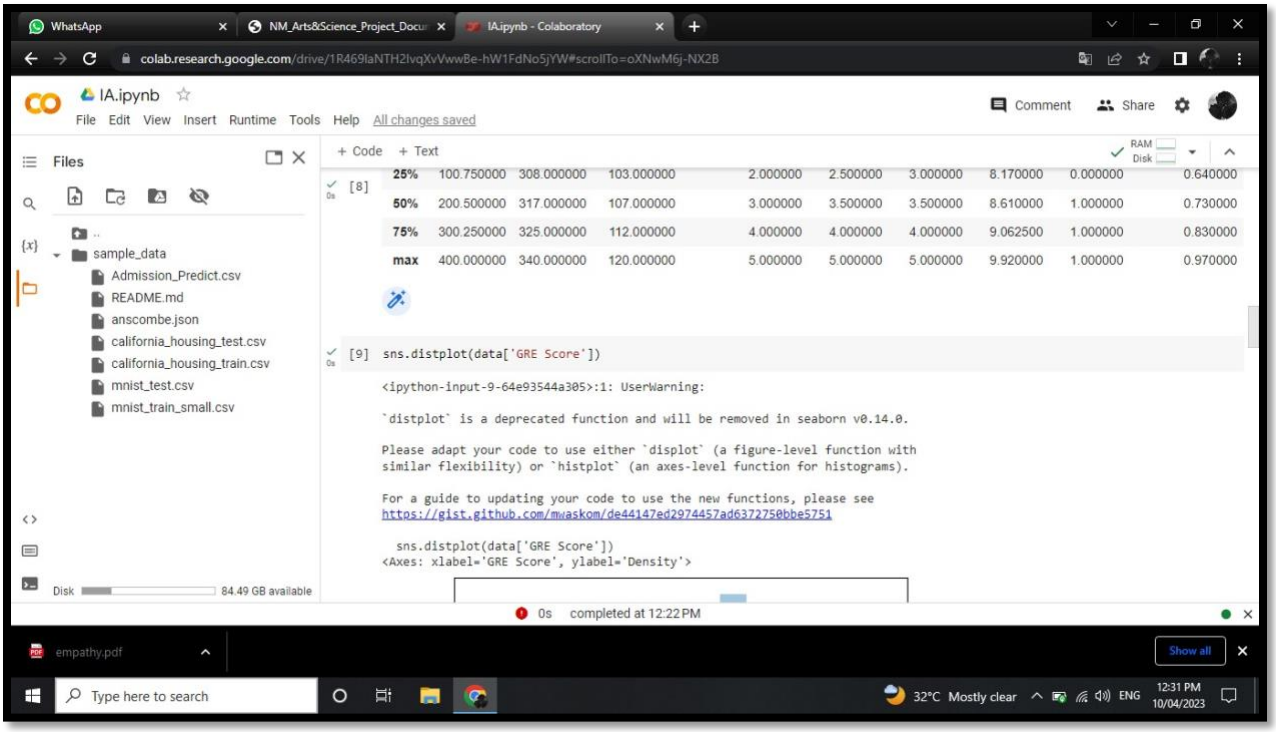

4
Prioritize
Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.
Used to analyze the measures of academic
Quickly process large amount of data
Organized information
It involves libraries pensorflow,keras or pyporch
Dataset should be large enough for process
Inteligent admission can streamline the admission process
Time manage
We can use java to develop Inteligent admission
we used python language in our project
Save cost associated with manual process
Predicted fair idea about their admission
improve the efficiency of admission
Faster and more efficient process
Importance
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?
anaylyze the student profile and make personalized
Itsa speed up the adimission process
18

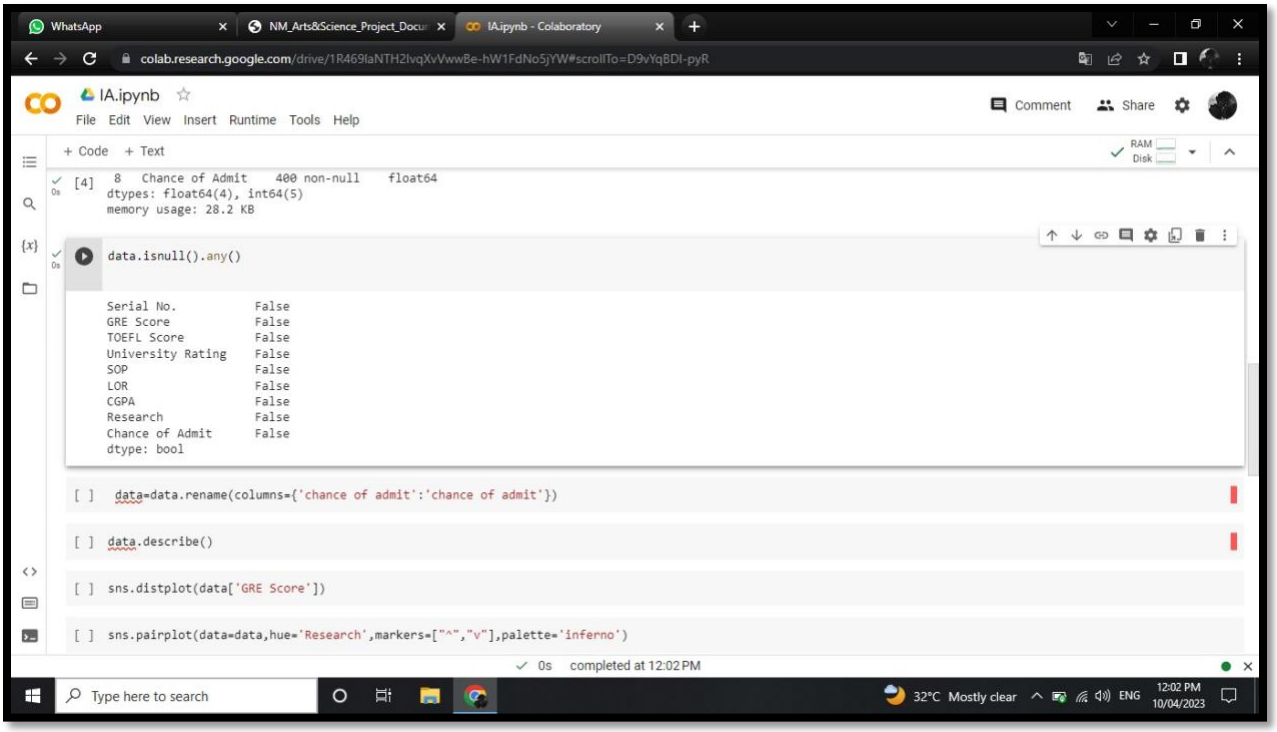

After you collaborate
We can export the mural as pdf to share.It is helpful to getting information
Quick add-ons
Share the mural
Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.
Export the mural
Export a copy of the mural as a PNG or PDF to attach to emails, include in slides, or save in your drive.
Keep moving forward
Strategy blueprint
Define the components of a new idea or strategy.
Open the template
Customer experience journey map
Understand customer needs, motivations, and obstacles for an experience.
Open the template
Strengths, weaknesses, opportunities & threats
18

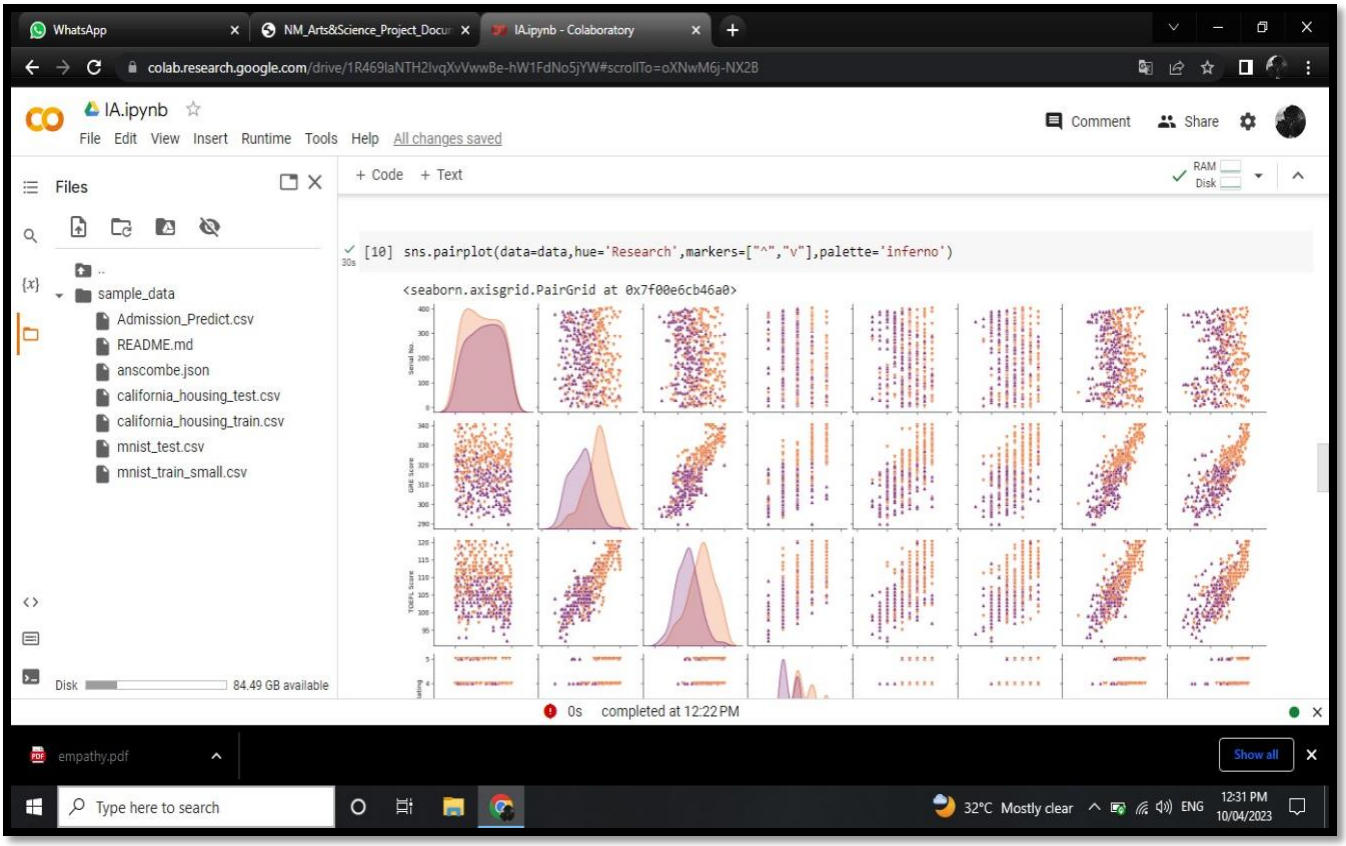
3.Result:

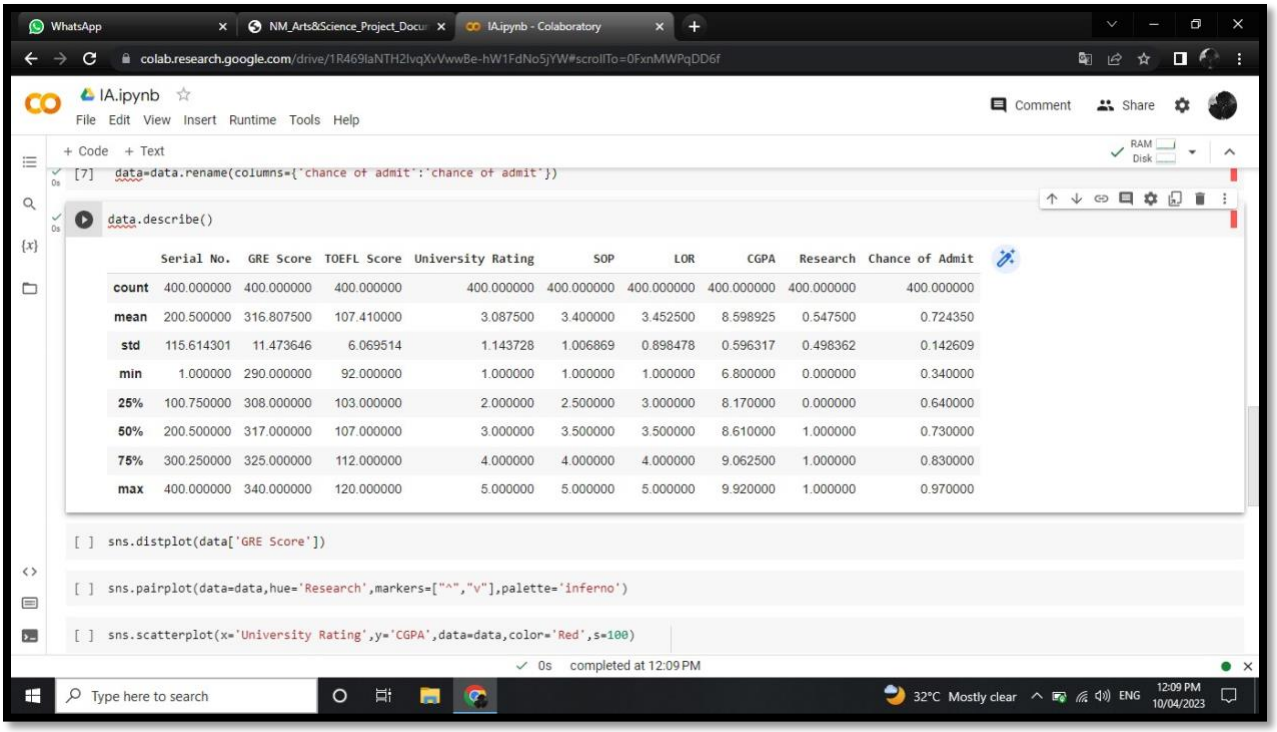

WhatsApp
NM_Arts&Science_Project_Docu
IA.ipynb - Colaboratory
colab.research.google.com/drive/1R469laNTH2lvqXvVwwBe-hW1FdNo5jYW#scrollTo=aDd1_Xhcvs6C
IA.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
#read_csv is a pandas function to read csv files
data = pd.read_csv('/content/sample_data/Admission_Predict.csv')
data
Serial No. GRE Score TOEFL Score University Rating SOP LOR CGPA Research Chance of Admit
0 1 337 118 4 4.5 4.5 9.65 1 0.92
1 2 324 107 4 4.0 4.5 8.87 1 0.76
2 3 316 104 3 3.0 3.5 8.00 1 0.72
3 4 322 110 3 3.5 2.5 8.67 1 0.80
4 5 314 103 2 2.0 3.0 8.21 0 0.65
395 396 324 110 3 3.5 3.5 9.04 1 0.82
396 397 325 107 3 3.0 3.5 9.11 1 0.84
397 398 330 116 4 5.0 4.5 9.45 1 0.91
0s completed at 12:02 PM
Type here to search
32°C Mostly clear
ENG
12:02 PM
10/04/2023

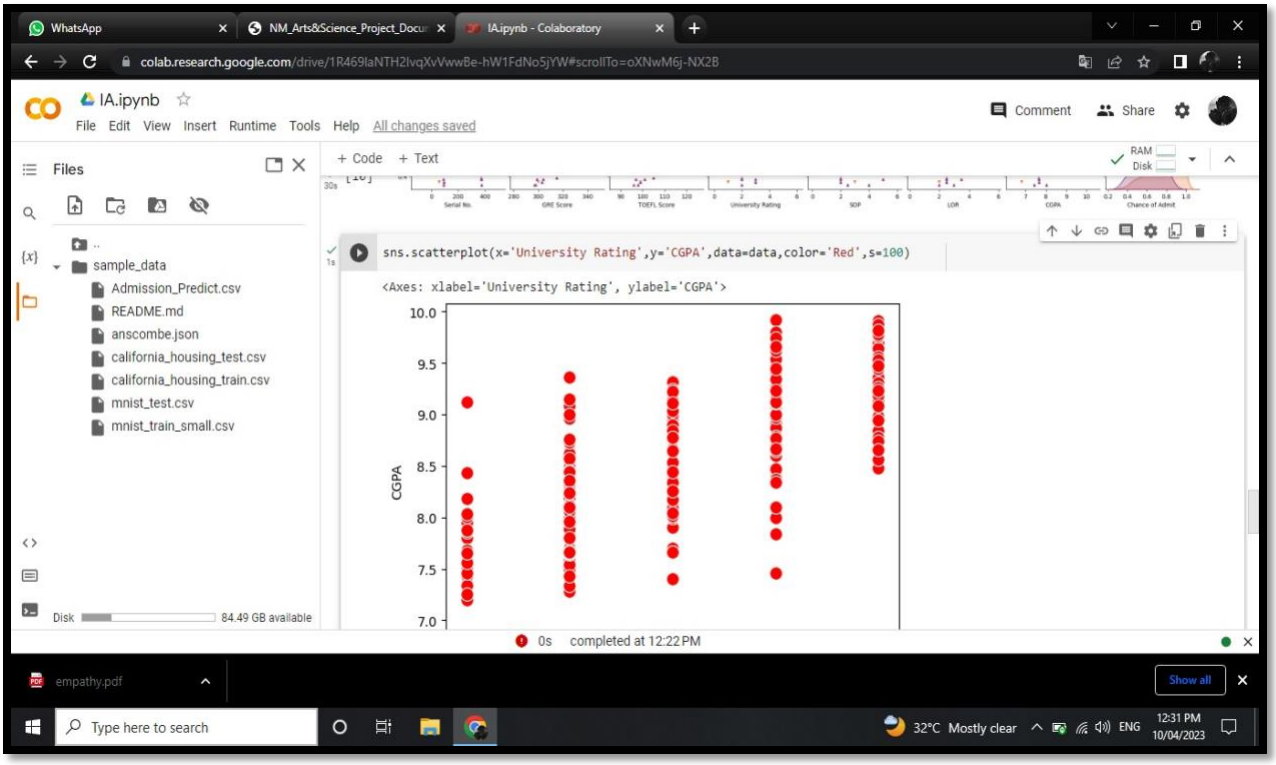

WhatsApp
NM_Arts&Science_Project_Docu
IA.ipynb - Colaboratory
colab.research.google.com/drive/1R469laNTH2lvqXvVwwBe-hW1FdNo5jYW#scrollTo=PNgmVMPrwAJD
IA.ipynb
File Edit View Insert Runtime Tools Help
Comment
Share
+ Code + Text
RAM
Disk
399 400 333 117 4 5.0 4.0 9.66 1 0.95
400 rows × 9 columns
data.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 400 entries, 0 to 399
Data columns (total 9 columns):
Column Non-Null Count Dtype
--- ------ -------------- -----
0 Serial No. 400 non-null int64
1 GRE Score 400 non-null int64
2 TOEFL Score 400 non-null int64
3 University Rating 400 non-null int64
4 SOP 400 non-null float64
5 LOR 400 non-null float64
6 CGPA 400 non-null float64
7 Research 400 non-null int64
8 Chance of Admit 400 non-null float64
dtypes: float64(4), int64(5)
memory usage: 28.2 KB
[ ] data.isnull().any()
0s completed at 12:02 PM
Type here to search
32°C Mostly clear
ENG
12:02 PM
10/04/2023

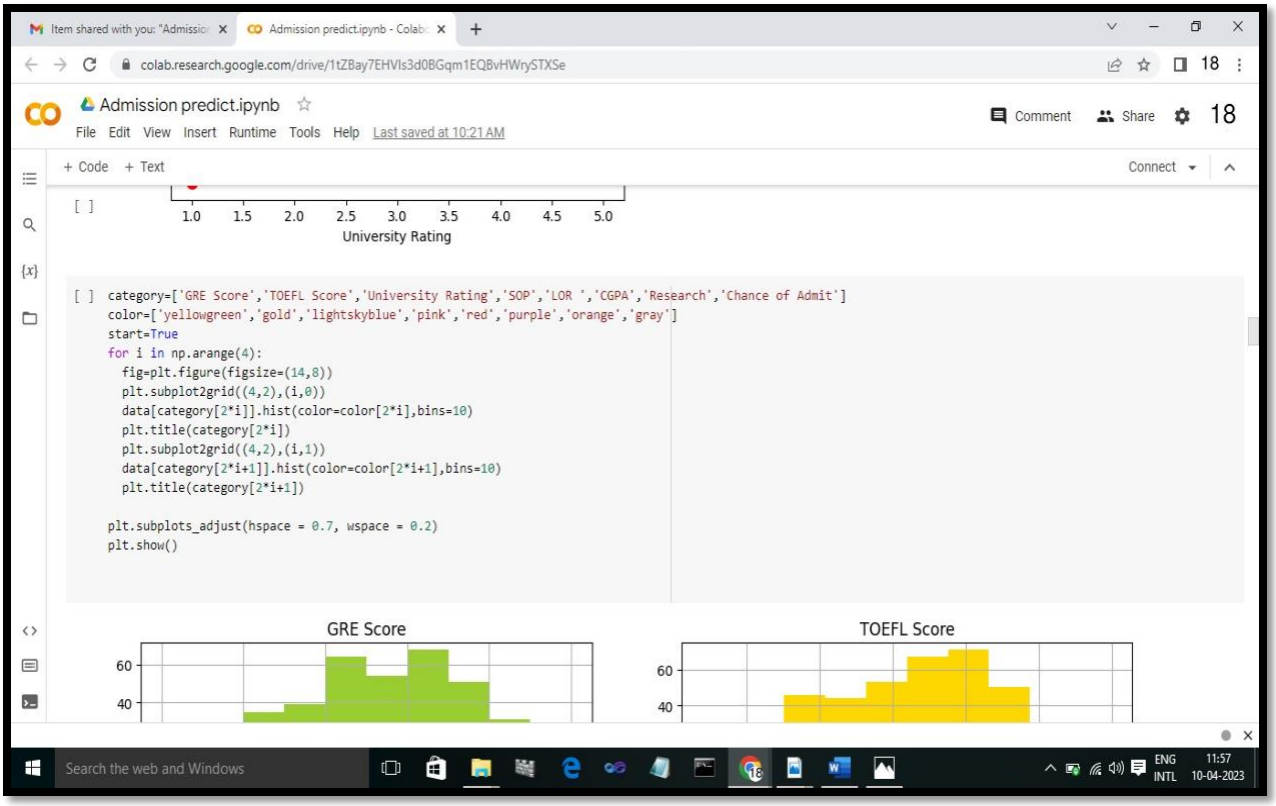

WhatsApp
NM_Arts&Science_Project_Docu
IA.ipynb - Colaboratory
colab.research.google.com/drive/1R469laNTH2lvqXvVwwBe-hW1FdNo5jYW#scrollTo=oXNwM6j-NX2B
IA.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
Files
+ Code + Text
RAM
Disk
sample_data
Admission_Predict.csv
README.md
anscombe.json
california_housing_test.csv
california_housing_train.csv
mnist_test.csv
mnist_train_small.csv
Disk 84.49 GB available
25% 100.750000 308.000000 103.000000 2.000000 2.500000 3.000000 8.170000 0.000000 0.640000
50% 200.500000 317.000000 107.000000 3.000000 3.500000 3.500000 8.610000 1.000000 0.730000
75% 300.250000 325.000000 112.000000 4.000000 4.000000 4.000000 9.062500 1.000000 0.830000
max 400.000000 340.000000 120.000000 5.000000 5.000000 5.000000 9.920000 1.000000 0.970000
[8]
[9] sns.distplot(data['GRE Score'])
<ipython-input-9-64e93544a305>:1: UserWarning:
`distplot` is a deprecated function and will be removed in seaborn v0.14.0.
Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).
For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751
sns.distplot(data['GRE Score'])
<Axes: xlabel='GRE Score', ylabel='Density'>
0s completed at 12:22 PM
empathy.pdf
Show all
Type here to search
32°C Mostly clear
ENG
12:31 PM
10/04/2023

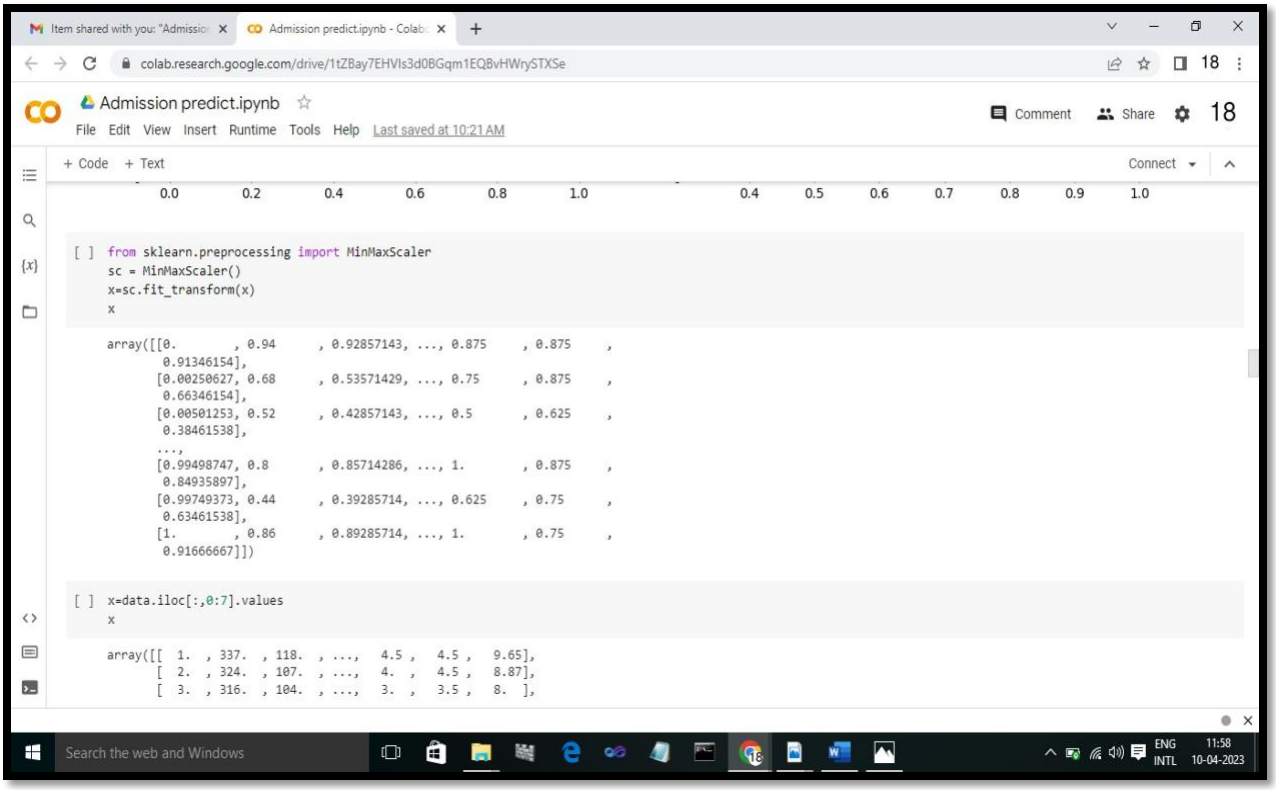

WhatsApp
NM_Arts&Science_Project_Docu
IA.ipynb - Colaboratory
colab.research.google.com/drive/1R469laNTH2lvqXvVwwBe-hW1FdNo5jYW#scrollTo=D9vYqBDI-pyR
IA.ipynb
File Edit View Insert Runtime Tools Help
Comment
Share
+ Code + Text
RAM
Disk
[4] 8 Chance of Admit 400 non-null float64
dtypes: float64(4), int64(5)
memory usage: 28.2 KB
data.isnull().any()
Serial No. False
GRE Score False
TOEFL Score False
University Rating False
SOP False
LOR False
CGPA False
Research False
Chance of Admit False
dtype: bool
[ ] data=data.rename(columns={'chance of admit':'chance of admit'})
[ ] data.describe()
[ ] sns.distplot(data['GRE Score'])
[ ] sns.pairplot(data=data,hue='Research',markers=["^","v"],palette='inferno')
0s completed at 12:02 PM
Type here to search
32°C Mostly clear
ENG
12:02 PM
10/04/2023

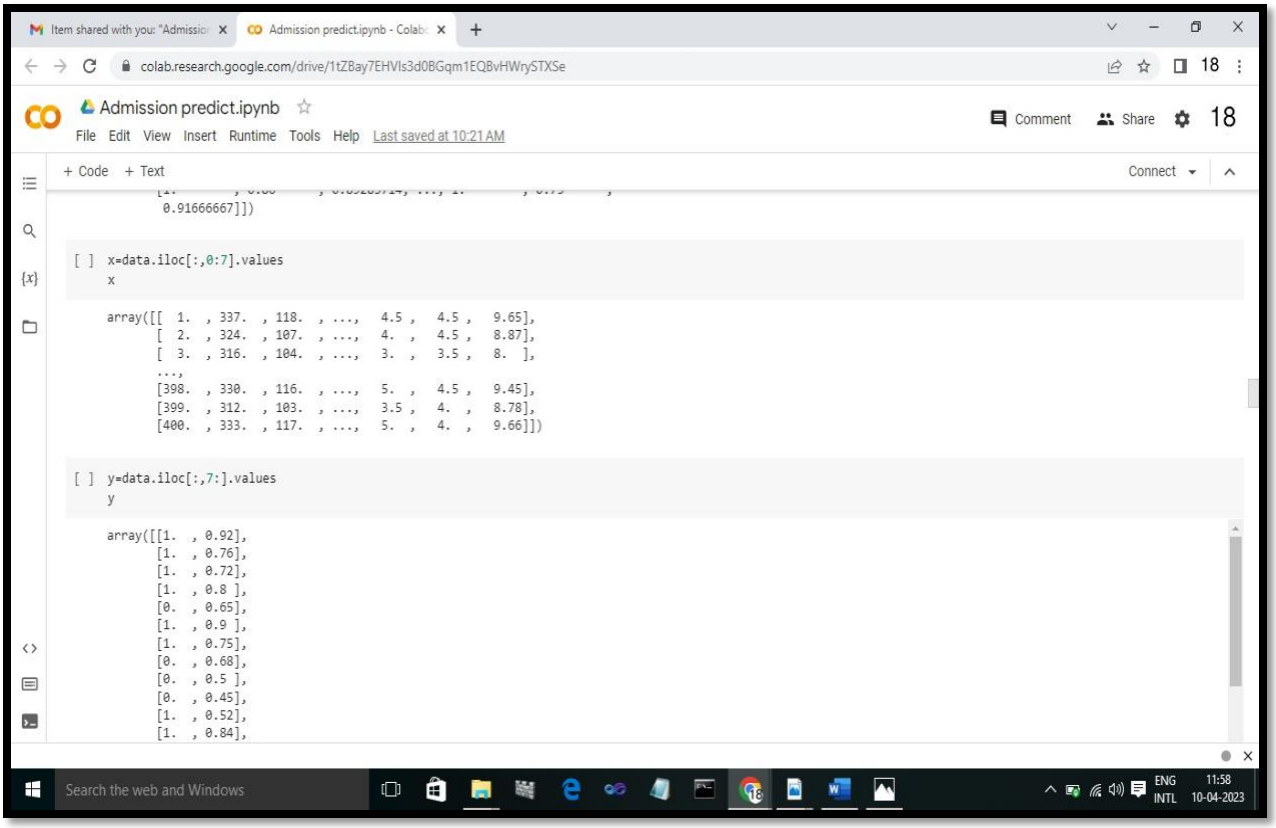

WhatsApp
NM_Arts&Science_Project_Docu
IA.ipynb - Colaboratory
colab.research.google.com/drive/1R469laNTH2lvqXvVwwBe-hW1FdNo5jYW#scrollTo=oXNwM6j-NX2B
IA.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
Files
+ Code + Text
RAM
Disk
[10] sns.pairplot(data=data,hue='Research',markers=["^","v"],palette='inferno')
<seaborn.axisgrid.PairGrid at 0x7f00e6cb46a0>
sample_data
Admission_Predict.csv
README.md
anscombe.json
california_housing_test.csv
california_housing_train.csv
mnist_test.csv
mnist_train_small.csv
Disk 84.49 GB available
0s completed at 12:22 PM
empathy.pdf
Show all
Type here to search
32°C Mostly clear
ENG
12:31 PM
10/04/2023

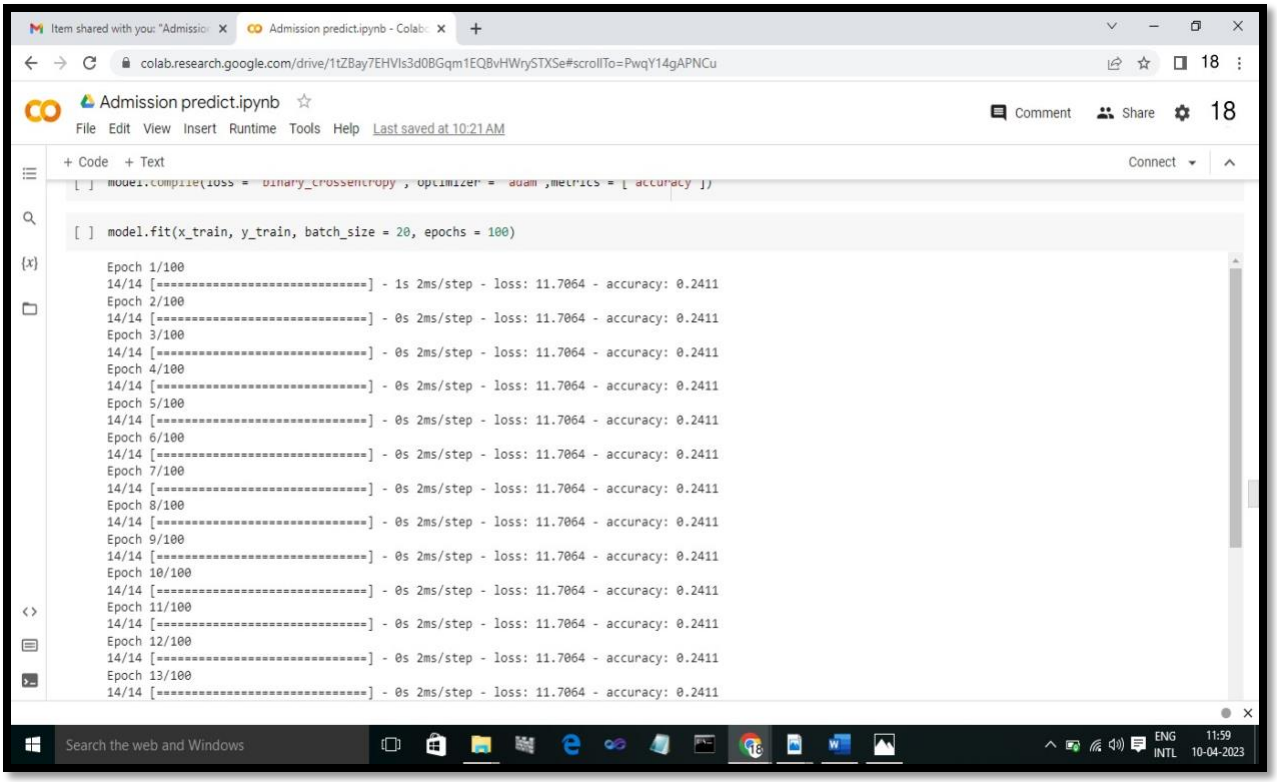

WhatsApp
NM_Arts&Science_Project_Docu
IA.ipynb - Colaboratory
colab.research.google.com/drive/1R469laNTH2lvqXvVwwBe-hW1FdNo5jYW#scrollTo=0FxnMWPqDD6f
IA.ipynb
File Edit View Insert Runtime Tools Help
Comment
Share
+ Code + Text
RAM
Disk
[7] data=data.rename(columns={'chance of admit':'chance of admit'})
data.describe()
Serial No. GRE Score TOEFL Score University Rating SOP LOR CGPA Research Chance of Admit
count 400.000000 400.000000 400.000000 400.000000 400.000000 400.000000 400.000000 400.000000 400.000000
mean 200.500000 316.807500 107.410000 3.087500 3.400000 3.452500 8.598925 0.547500 0.724350
std 115.614301 11.473646 6.069514 1.143728 1.006869 0.898478 0.596317 0.498362 0.142609
min 1.000000 290.000000 92.000000 1.000000 1.000000 1.000000 6.800000 0.000000 0.340000
25% 100.750000 308.000000 103.000000 2.000000 2.500000 3.000000 8.170000 0.000000 0.640000
50% 200.500000 317.000000 107.000000 3.000000 3.500000 3.500000 8.610000 1.000000 0.730000
75% 300.250000 325.000000 112.000000 4.000000 4.000000 4.000000 9.062500 1.000000 0.830000
max 400.000000 340.000000 120.000000 5.000000 5.000000 5.000000 9.920000 1.000000 0.970000
[ ] sns.distplot(data['GRE Score'])
[ ] sns.pairplot(data=data,hue='Research',markers=["^","v"],palette='inferno')
[ ] sns.scatterplot(x='University Rating',y='CGPA',data=data,color='Red',s=100)
0s completed at 12:09 PM
Type here to search
32°C Mostly clear
ENG
12:09 PM
10/04/2023

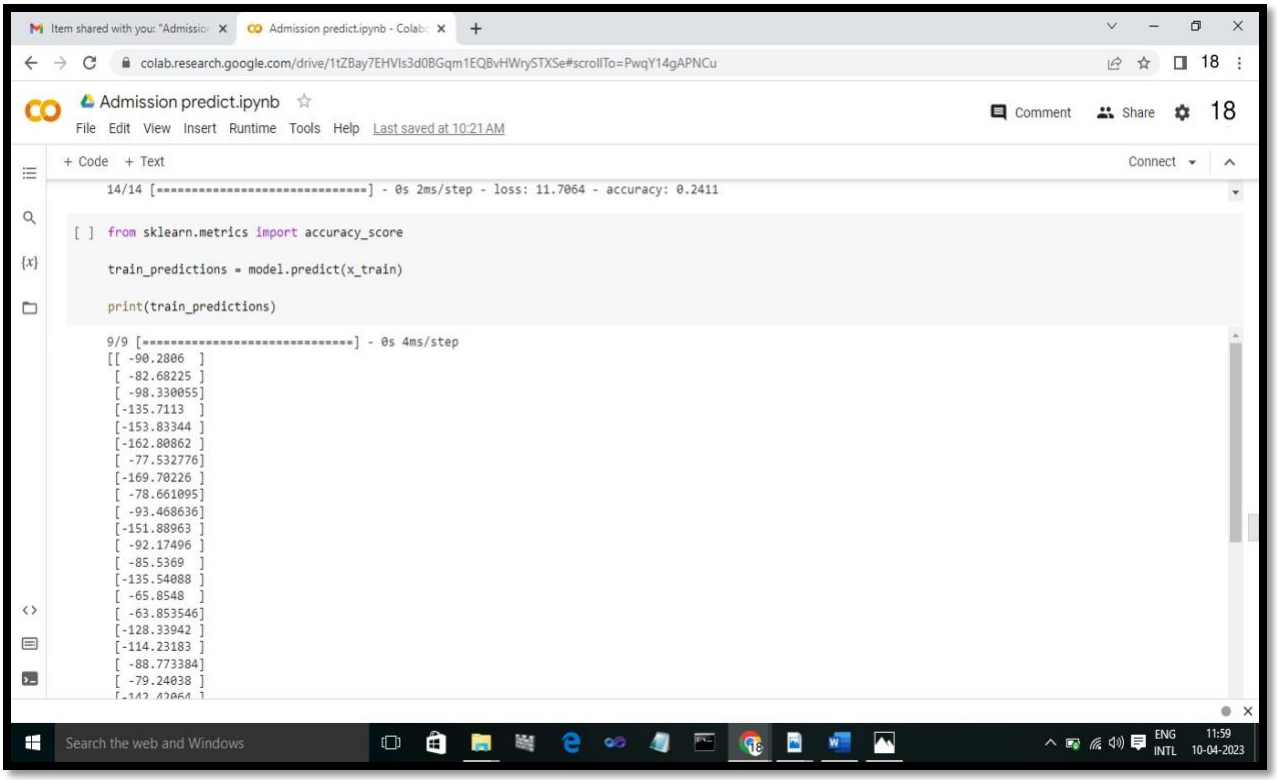

WhatsApp
NM_Arts&Science_Project_Docu
IA.ipynb - Colaboratory
colab.research.google.com/drive/1R469laNTH2lvqXvVwwBe-hW1FdNo5jYW#scrollTo=oXNwM6j-NX2B
IA.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
Files
+ Code + Text
RAM
Disk
sample_data
Admission_Predict.csv
README.md
anscombe.json
california_housing_test.csv
california_housing_train.csv
mnist_test.csv
mnist_train_small.csv
Disk 84.49 GB available
sns.scatterplot(x='University Rating',y='CGPA',data=data,color='Red',s=100)
<Axes: xlabel='University Rating', ylabel='CGPA'>
CGPA
10.0
9.5
9.0
8.5
8.0
7.5
7.0
0s completed at 12:22 PM
empathy.pdf
Show all
Type here to search
32°C Mostly clear
ENG
12:31 PM
10/04/2023


Admission predict.ipynb
File Edit View Insert Runtime Tools Help Last saved at 10:21 AM
Comment Share 18
+ Code + Text
Connect
1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0
University Rating
category=['GRE Score','TOEFL Score','University Rating','SOP','LOR ','CGPA','Research','Chance of Admit']
color=['yellowgreen','gold','lightskyblue','pink','red','purple','orange','gray']
start=True
for i in np.arange(4):
fig=plt.figure(figsize=(14,8))
plt.subplot2grid((4,2),(i,0))
data[category[2*i]].hist(color=color[2*i],bins=10)
plt.title(category[2*i])
plt.subplot2grid((4,2),(i,1))
data[category[2*i+1]].hist(color=color[2*i+1],bins=10)
plt.title(category[2*i+1])
plt.subplots_adjust(hspace = 0.7, wspace = 0.2)
plt.show()
GRE Score
TOEFL Score
60
40
Search the web and Windows
ENG INTL
11:57
10-04-2023


Item shared with you: "Admissio
Admission predict.ipynb - Colab
colab.research.google.com/drive/1tZBay7EHVIs3d0BGqm1EQBvHWrySTXSe
Admission predict.ipynb
File Edit View Insert Runtime Tools Help Last saved at 10:21 AM
Comment
Share
+ Code + Text
Connect
0.0 0.2 0.4 0.6 0.8 1.0 0.4 0.5 0.6 0.7 0.8 0.9 1.0
[ ] from sklearn.preprocessing import MinMaxScaler
sc = MinMaxScaler()
x=sc.fit_transform(x)
x
array([[0. , 0.94 , 0.92857143, ..., 0.875 , 0.875 ,
0.91346154],
[0.00250627, 0.68 , 0.53571429, ..., 0.75 , 0.875 ,
0.66346154],
[0.00501253, 0.52 , 0.42857143, ..., 0.5 , 0.625 ,
0.38461538],
...,
[0.99498747, 0.8 , 0.85714286, ..., 1. , 0.875 ,
0.84935897],
[0.99749373, 0.44 , 0.39285714, ..., 0.625 , 0.75 ,
0.63461538],
[1. , 0.86 , 0.89285714, ..., 1. , 0.75 ,
0.91666667]])
[ ] x=data.iloc[:,0:7].values
x
array([[1. , 337. , 118. , ..., 4.5 , 4.5 , 9.65],
[2. , 324. , 107. , ..., 4. , 4.5 , 8.87],
[3. , 316. , 104. , ..., 3. , 3.5 , 8.],
Search the web and Windows
ENG
INTL
11:58
10-04-2023


Item shared with you: "Admissio
Admission predict.ipynb - Colab
colab.research.google.com/drive/1tZBay7EHVIs3d0BGqm1EQBvHWrySTXSe
Admission predict.ipynb
File Edit View Insert Runtime Tools Help Last saved at 10:21 AM
Comment
Share
18
+ Code + Text
Connect
0.91666667]])
[ ] x=data.iloc[:,0:7].values
x
array([[1. , 337. , 118. , ..., 4.5 , 4.5 , 9.65],
[2. , 324. , 107. , ..., 4. , 4.5 , 8.87],
[3. , 316. , 104. , ..., 3. , 3.5 , 8.],
...,
[398. , 330. , 116. , ..., 5. , 4.5 , 9.45],
[399. , 312. , 103. , ..., 3.5 , 4. , 8.78],
[400. , 333. , 117. , ..., 5. , 4. , 9.66]])
[ ] y=data.iloc[:,7:].values
y
array([[1. , 0.92],
[1. , 0.76],
[1. , 0.72],
[1. , 0.8],
[0. , 0.65],
[1. , 0.9],
[1. , 0.75],
[0. , 0.68],
[0. , 0.5],
[0. , 0.45],
[1. , 0.52],
[1. , 0.84],
Search the web and Windows
ENG
INTL
11:58
10-04-2023


Item shared with you: "Admissio
Admission predict.ipynb - Colab
colab.research.google.com/drive/1tZBay7EHVIs3d0BGqm1EQBvHWrySTXSe#scrollTo=PwqY14gAPNCu
Admission predict.ipynb
File Edit View Insert Runtime Tools Help Last saved at 10:21 AM
Comment
Share
18
+ Code + Text
Connect
[ ] model.compile(loss = 'binary_crossentropy', optimizer = 'adam',metrics = ['accuracy'])
[ ] model.fit(x_train, y_train, batch_size = 20, epochs = 100)
Epoch 1/100
14/14 [==============================] - 1s 2ms/step - loss: 11.7064 - accuracy: 0.2411
Epoch 2/100
14/14 [==============================] - 0s 2ms/step - loss: 11.7064 - accuracy: 0.2411
Epoch 3/100
14/14 [==============================] - 0s 2ms/step - loss: 11.7064 - accuracy: 0.2411
Epoch 4/100
14/14 [==============================] - 0s 2ms/step - loss: 11.7064 - accuracy: 0.2411
Epoch 5/100
14/14 [==============================] - 0s 2ms/step - loss: 11.7064 - accuracy: 0.2411
Epoch 6/100
14/14 [==============================] - 0s 2ms/step - loss: 11.7064 - accuracy: 0.2411
Epoch 7/100
14/14 [==============================] - 0s 2ms/step - loss: 11.7064 - accuracy: 0.2411
Epoch 8/100
14/14 [==============================] - 0s 2ms/step - loss: 11.7064 - accuracy: 0.2411
Epoch 9/100
14/14 [==============================] - 0s 2ms/step - loss: 11.7064 - accuracy: 0.2411
Epoch 10/100
14/14 [==============================] - 0s 2ms/step - loss: 11.7064 - accuracy: 0.2411
Epoch 11/100
14/14 [==============================] - 0s 2ms/step - loss: 11.7064 - accuracy: 0.2411
Epoch 12/100
14/14 [==============================] - 0s 2ms/step - loss: 11.7064 - accuracy: 0.2411
Epoch 13/100
14/14 [==============================] - 0s 2ms/step - loss: 11.7064 - accuracy: 0.2411
Search the web and Windows
ENG
INTL
11:59
10-04-2023


Item shared with you: "Admissio
Admission predict.ipynb - Colab
colab.research.google.com/drive/1tZBay7EHVIs3d0BGqm1EQBvHWrySTXSe#scrollTo=PwqY14gAPNCu
Admission predict.ipynb
File Edit View Insert Runtime Tools Help Last saved at 10:21 AM
Comment
Share
+ Code + Text
Connect
14/14 [==========] - 0s 2ms/step - loss: 11.7064 - accuracy: 0.2411
from sklearn.metrics import accuracy_score
train_predictions = model.predict(x_train)
print(train_predictions)
9/9 [==========] - 0s 4ms/step
[[-90.2806]
[-82.68225]
[-98.330055]
[-135.7113]
[-153.83344]
[-162.80862]
[-77.532776]
[-169.70226]
[-78.661095]
[-93.468636]
[-151.88963]
[-92.17496]
[-85.5369]
[-135.54088]
[-65.8548]
[-63.853546]
[-128.33942]
[-114.23183]
[-88.773384]
[-79.24038]
Search the web and Windows
ENG
INTL
11:59
10-04-2023

4.Advantages

Machine Learning can review large volumes of data and discover specific trends and patterns that would not be apparent to humans. For instance, for an e-commerce website like Amazon, it serves to understand the browsing behaviors and purchase histories of its users to help cater to the right products, deals, and reminders relevant to them. It uses the results to reveal relevant advertisements to them.

With ML, you don't need to babysit your project every step of the way. Since it means giving machines the ability to learn, it lets them make predictions and also improve the algorithms on their own. A common example of this is anti-virus softwares; they learn to filter new threats as they are recognized. ML is also good at recognizing spam.

Disadvantages

Machine Learning requires massive data sets to train on, and these should be inclusive/unbiased, and of good quality. There can also be times where they must wait for new data to be generated.

ML needs enough time to let the algorithms learn and develop enough to fulfill their purpose with a considerable amount of accuracy and relevancy. It also needs massive resources to function. This can mean additional requirements of computer power for you.

5.Application:

Intelligent Admission you've weighed up all the factors and carefully made your decision, it's time for the really fun part: applying.

Though this might seem obvious, ensure you take care over this. You don't want to miss out simply because you forgot to submit the required evidence or applied too late."Students should check entry requirements and deadlines before applying to make sure that they have the best possible chance of gaining a place on their chosen course," Berry confirms.

"If they are unsure whether their qualifications are acceptable, they might like to contact the admissions office or international office in their chosen institution to check before submitting a full application."

She emphasizes the importance of applying in good time: "Students should try to make an application as early as possible as this will give them plenty of time to make all the necessary arrangements for a move abroad, including organizing their finances, applying for scholarships and obtaining a student visa."

6.Conclusion:

Artificial Intelligence has the potential to transform all organizations. The process by which this transformation happens can vary .But the steps will tend to follow the roadmap we have listed in this book.

The predicted output gives them a fair idea about their admission chances in a particular university. This analysis should also help students who are currently preparing or will be preparing to get a better idea.

7.Future Scope of Artificial Intelligence

The adoption of Artificial Intelligence in India is promising. However, currently, it is at a nascent stage. While there are a few industries such as IT, manufacturing, automobile, etc, that are leveraging the prowess of AI, there are still many areas in which its potential is unexplored.

The immense potential that AI holds can be understood by the various other technologies that are covered under the umbrella of AI. Some of the examples of such technologies include self-improving algorithms, Machine Learning, Pattern Recognition, Big Data, and many others. In the next few years, it is predicted that there will hardly be any industry left untouched by this powerful tool. This is the reason why AI has so much potential to grow in India.

Sourcecode:

```
import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

%matplotlib inline

data = pd.read_csv('/content/sample_data/Admission_Predict.csv')

data

data.info()

data.isnull().any()

data=data.rename(columns={‘change of admit ‘:’Change of Admit’})

data.describe()

sns.distplot(data[‘GRE Score’])

sns.pairplot(data=data,hue=’Research’,markers=[“^” , “v”],palette=’inferno’)

sns.scatterplot(x=’University Rating’,y=’CGPA’,data=data,color=’Red’,s=100)

from sklearn.preprocessing import  MinMaxScaller

sc=MinMaxScaller()

x=sc.fit_transform(x)

x

x=data.iloc[:,0:7].values

x

y=data.iloc[:,7:].values

y
```

```
y_train=(y_train>0.5)

y_train

y_test=(y_test>0.5)

import tensorflow as tf

from tesorflow import keras
```

home.html

```
<!doctype html>

<html>

<head>


<title>

UNIVERSITY ADMISSION PREDICTION SYSTEM

</title>

</head>

<body>

<br>

<form>

<label>enter QRE score</label>

<input type="text" name="enter QRE score" size="15"><br><br>

<label>enter TOEFL score</label>

<inut type="text" name="Enter TOEFL Score"> size="15"><br><br>

<label>

select university on:

</label>

<select>
```

```
<option value="select University on">select university on</option>
<option value="1">1</option>
<option value="2">2</option>
<option value="3">3</option>
<option value="4">4</option>
<option value="5">5</option>
</select>
<br>
<br>
<label>Enter SOP</label>
<input type="text" name ="Enter SOP"size="15"><br><br>
<label>Enter LOR</label>
<input type="text" name ="Enter LOR"size="15"><br><br>
<label>Enter CGPA</label>
<input type="text" name ="Enter CGPA"size="15"><br><br>
<label>
Research
</label><br>
<input type ="radio" name="Research"/>Research<br>
<input type ="radio" name="No Research"/>No Research<br>
<br>
<br>
<input type="submit" value="submit">
<input type="reset" value="Reset">
</form>
```

```
</body>
</html>
```

Project .py

```
import numby as np
from flask import Flask, request, jsonify, render_template
import pickle
app = Flask(name)
from tensorflow.keras.models import load_model
model = load_model('model.h5')
@app.route('/')
def home():
    return render_template('Demo2.html')
@app.route('/')
def home():
    return render_template('Demo2.html')
@app.route('/y_predict',methods=['POST'])
def y_predict():
    '''
    for rendering results on HTML GUI
    '''
    min1=[290.0, 92.0, 1.0, 1.0, 1.0, 6.8, 0.0]
    max1=[340.0, 120.0, 5.0, 5.0, 5.0, 9.92, 1.0]
    k= [float(x) for x in request.form.values()]
    p=[]
    for i in range(7):
```

```
l=(k[i]-min1[i])/(max1[i]-min[i])

p.append(1)

prediction = model.predict([p])

print(prediction)

output=prediction[0]

if(output==False):

    return render_template('noChance.html', prediction_text='you dont have a chance of getting admission')

else:

    return render_template('chance.html', prediction_text='you have a chance of getting admission')

if name == "main":

    app.run(debug=False)
```