IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING

SUBMITTED BY

MENAKA.V(20326ER056)

NITHYASRI.J(20326ER057)

NITHYASRI.K(20326ER058)

PRIYA.S(20326ER059)

CONTENT

- INTRODUCTION
- PROBLEM DEFINITION & DESIGN THINKING
- RESULT
- ADVANTAGES
- APPLICATIONS
- CONCLUSION
- FUTURE SCOPE
- APPENDIX

ABSTRACT

- Campus recruitment is a strategy for sourcing, engaging and hiring young talent for internship and entry-level positions.

- Campus recruitment often involves working with university career services centers and attending career fairs to meet in-person with college students and recent graduates.

- We will be using algorithms such as KNN, SVM and ANN.

1. INTRODUCTION

1.1 **OVERVIEW**

- Campus recruitment often involves working with university career services centers and attending career fairs to meet in-person with college students and recent graduates. Our solution revolves around the placement season of a Business School in India.

- We will train and test the data with these algorithms. From this the best model is selected and saved in .pkl format. We will be doing flask integration and IBM deployment.

1.2 PURPOSE

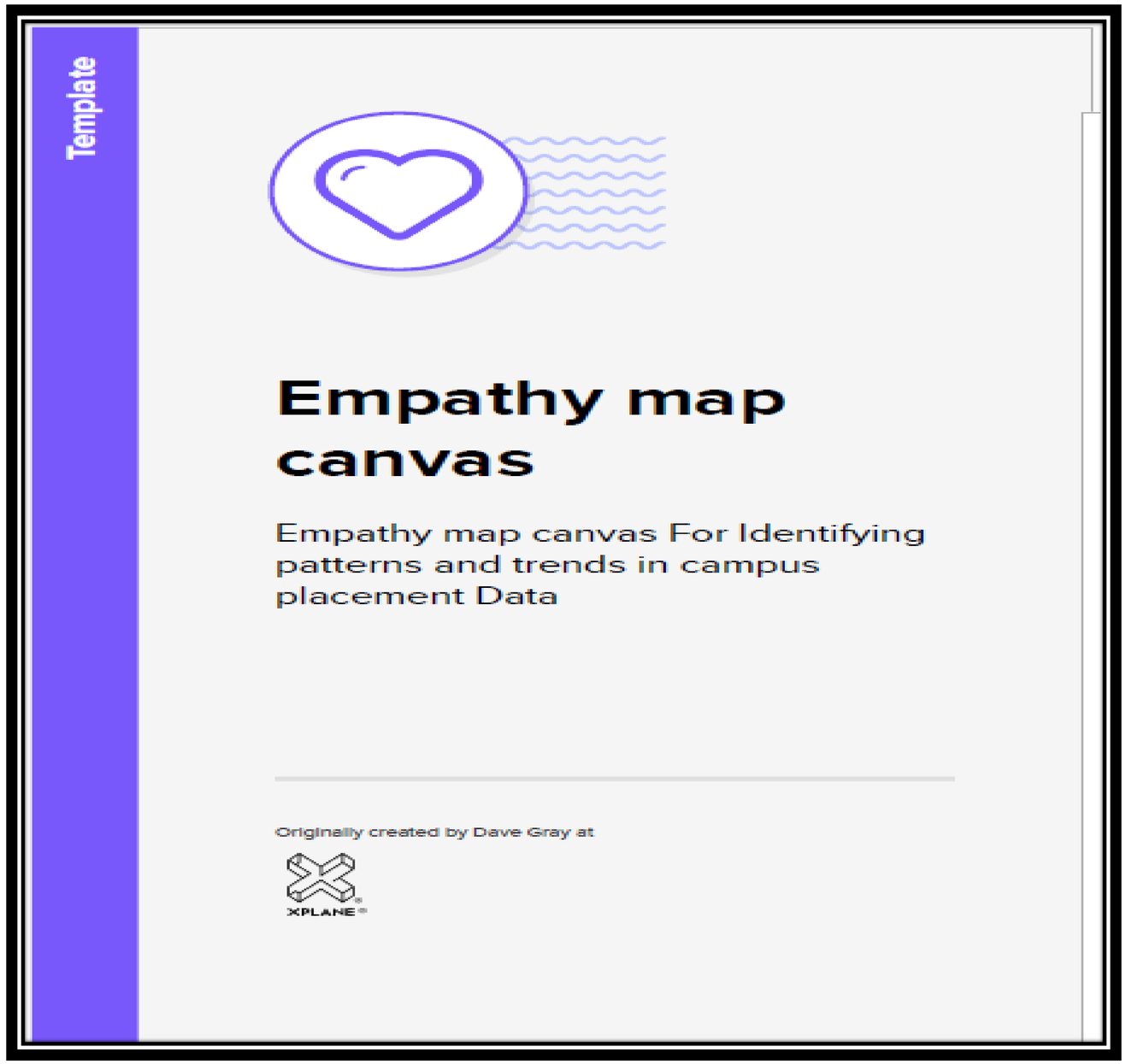
- Identifying patterns and trends in campus placement data using machine learning is a technique that involves the use of advanced analytical algorithms and statistical models to analyze and interpret the data related to campus placement.

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

- Machine learning algorithm can be used to analyze the data related to students academic performance, their technical skills, and their performance in interviews.
- These algorithms can also be used to identify the correlation between various factors and the success rate of students in securing placements.

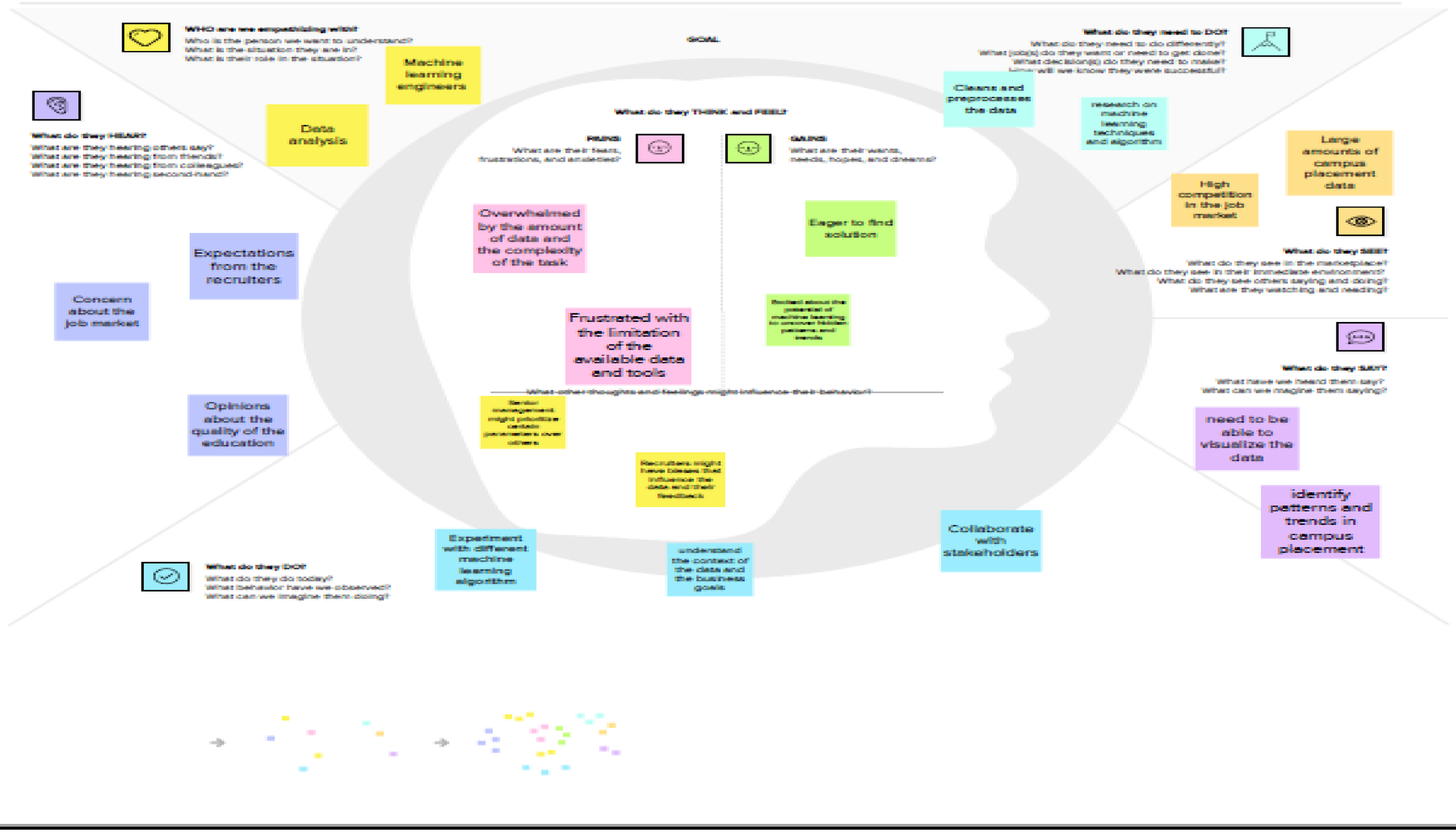
2. PROBLEM DEFINITIO&DESIGNTHINKING

2.1 EMPATHY MAP



Develop shared understanding and empathy

Summarize the data you have gathered related to the people that are impacted by your work. It will help you generate ideas, prioritize features, or discuss decisions.



2.2 IDEATION & BRAINSTORMING MAP

Before collaborate

The main objective of campus placement is to identify the talented and qualified student before they complete the education.

10 minutes

A **Team gathering**
Totally Four Participation are there in session. We invite members through mural link and gathered in this session.

B **Set the goal**
The main objective of campus placement is to identify the talented and qualified student before they complete the education.

C **Learn how to use the facilitation tools**
Facilitation tools can be very helpful for guiding discussions, brainstorming sessions, or decision- making processes.

Open article →

1

problem statement

5 minutes

1)The Main objective of Campus placement is to identify the talented and qualified student before they complete the education.

2)This Project is used to Career opportunities for student in reputed corporate companies.

3)Campus recruitment is a strategy for sourcing, engaging and hiring young talent for internship and entry level position.

4)Campus recruitment often involves working with university career services centers and attending career fairs to meet in person with college students and recent graduates.

5)This Prediction uses a Machine learning algorithm to gives the result.

2

Brainstorm

Here some ideas

10 minutes

Person 1

- KNN algorithm is used.
- High Performance CPU is used.
- Dataset is needed.
- Visualization techniques is used.

Person 2

- ANN algorithm is used.
- 16GB or more of RAM is needed.
- Libraries are imported.
- Python language is used.

Person 3

- SVM algorithm is used.
- need to use tools like Pandas.
- Building Html Pages is used.
- COLAB is used to collaborate on a single notebook.

Person 4

- Machine learning algorithm is used.
- Internet is needed.
- Web Framework is used.
- operating system is required.

3

Group ideas

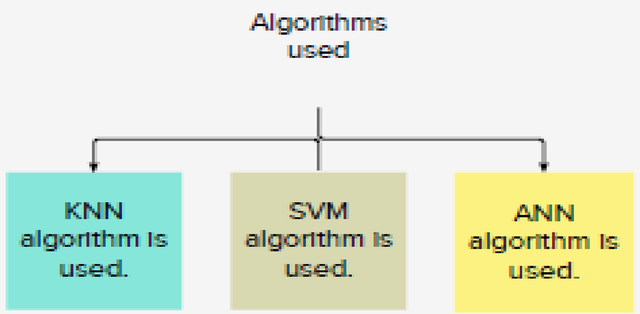
1)Operating system is required.
2)COLAB is used to collaborate on a single notebook.
3)Machine learning algorithm is used.
4)Dataset is needed.
20 minutes
5)KNN,ANN,SVM algorithm is used.

operating system is required.

COLAB is used to collaborate on a single notebook.

Machine learning algorithm is used.

Dataset is needed.

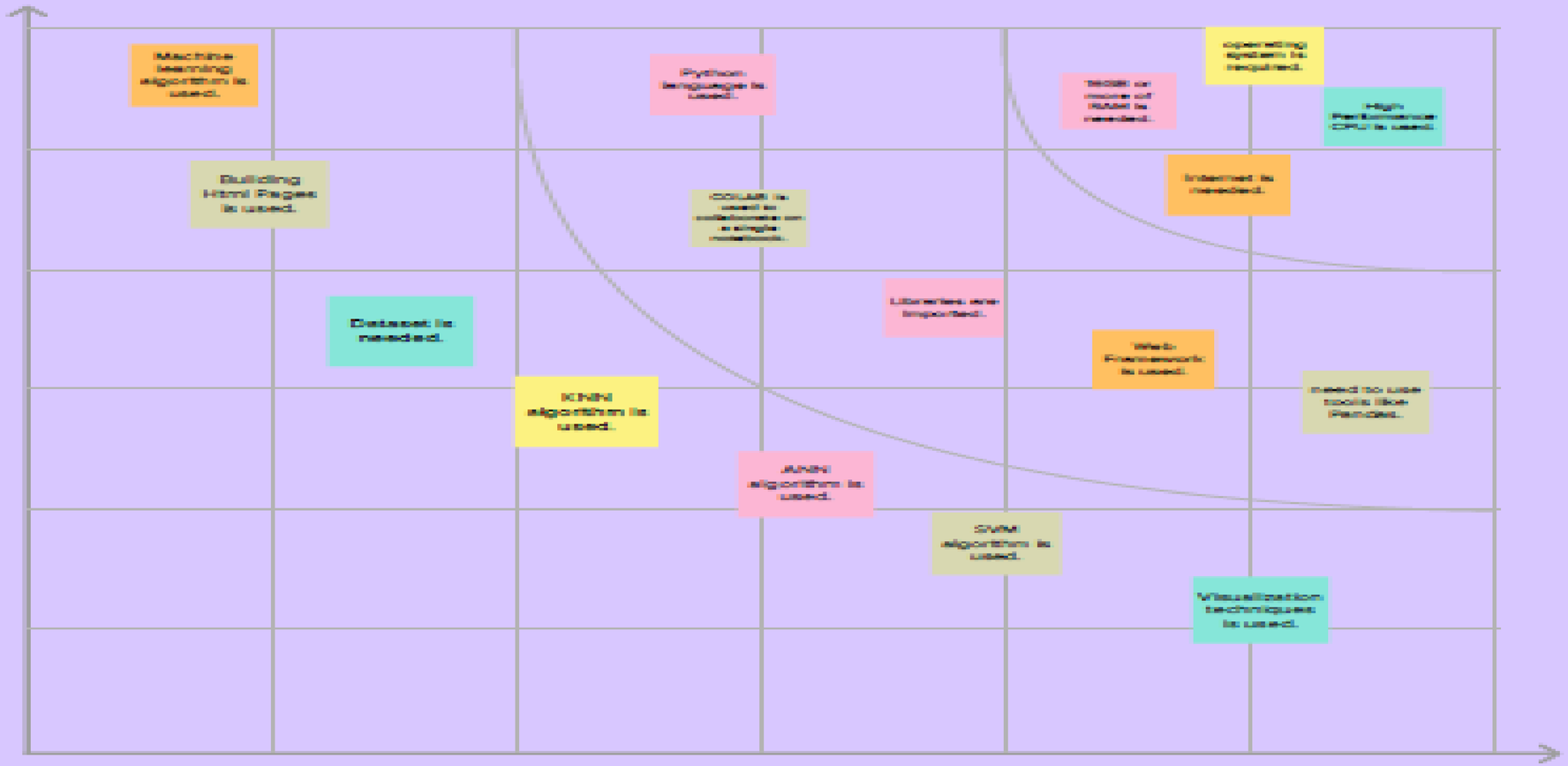


4

Prioritize

Priortize the Ideas

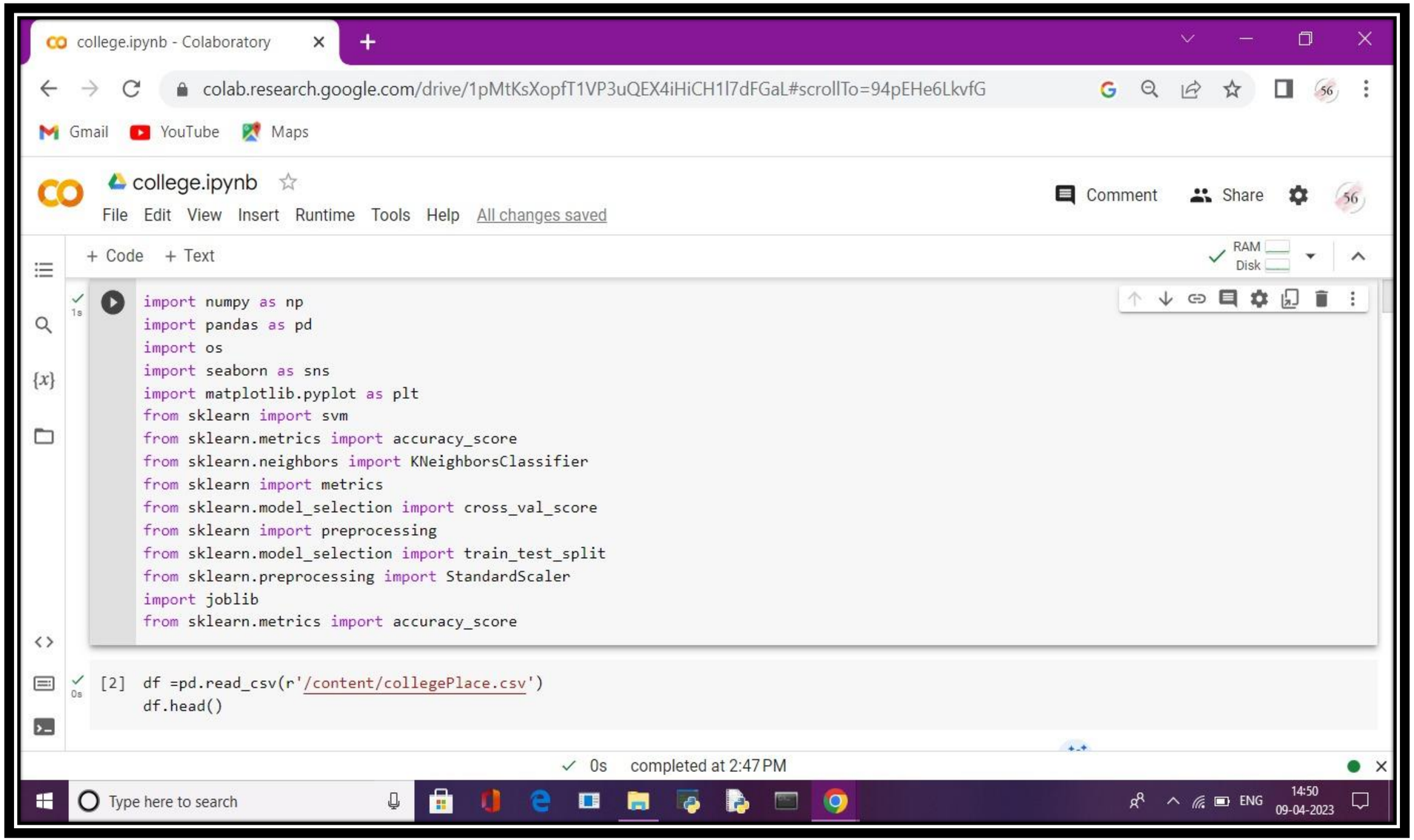
20 minutes



After collaborate

We can export the mural as pdf to share .It is helpful to getting information.

3.RESULT


college.ipynb - Colaboratory
colab.research.google.com/drive/1pMtKsXopfT1VP3uQEX4iHiCH1l7dFGaL#scrollTo=94pEHe6LkvfG
Gmail
YouTube
Maps
college.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 os
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn import svm
from sklearn.metrics import accuracy_score
from sklearn.neighbors import KNeighborsClassifier
from sklearn import metrics
from sklearn.model_selection import cross_val_score
from sklearn import preprocessing
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
import joblib
from sklearn.metrics import accuracy_score
[2] df =pd.read_csv(r'/content/collegePlace.csv')
df.head()
0s completed at 2:47 PM
Type here to search
ENG
14:50
09-04-2023

college.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

Comment Share 56

+ Code + Text

RAM Disk

```
df =pd.read_csv(r'/content/collegePlace.csv')
df.head()
```

	Age	Gender	Stream	Internships	CGPA	Hostel	HistoryOfBacklogs	PlacedOrNot
0	22	Male	Electronics And Communication	1	8	1	1	1
1	21	Female	Computer Science	0	7	1	1	1
2	22	Female	Information Technology	1	6	0	0	1
3	21	Male	Information Technology	0	8	0	1	1
4	22	Male	Mechanical	0	8	1	0	1

```
[3] df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2966 entries, 0 to 2965
Data columns (total 8 columns):
 #   Column             Non-Null Count  Dtype
```

✓ 0s completed at 2:47 PM

college.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

Comment Share 56

+ Code + Text

RAM Disk

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2966 entries, 0 to 2965
Data columns (total 8 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Age               2966 non-null   int64
 1   Gender            2966 non-null   object
 2   Stream            2966 non-null   object
 3   Internships       2966 non-null   int64
 4   CGPA              2966 non-null   int64
 5   Hostel            2966 non-null   int64
 6   HistoryOfBacklogs 2966 non-null   int64
 7   PlacedOrNot       2966 non-null   int64
dtypes: int64(6), object(2)
memory usage: 185.5+ KB
```

```
[4] df.isnull().sum()
```

✓ 0s completed at 2:47 PM


college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
56
+ Code + Text
RAM
Disk
[3]
0s
2 Stream 2966 non-null object
3 Internships 2966 non-null int64
4 CGPA 2966 non-null int64
5 Hostel 2966 non-null int64
6 HistoryOfBacklogs 2966 non-null int64
7 PlacedOrNot 2966 non-null int64
dtypes: int64(6), object(2)
memory usage: 185.5+ KB
df.isnull().sum()
0s
Age 0
Gender 0
Stream 0
Internships 0
CGPA 0
Hostel 0
HistoryOfBacklogs 0
PlacedOrNot 0
dtype: int64
0s completed at 2:47 PM

college.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

Comment Share 56

+ Code + Text

RAM Disk

```
def transformationplot(feature):
  plt.figure(figsize=(12,5))
  plt.subplot(1,2,1)
  sns.distplot(feature)
```

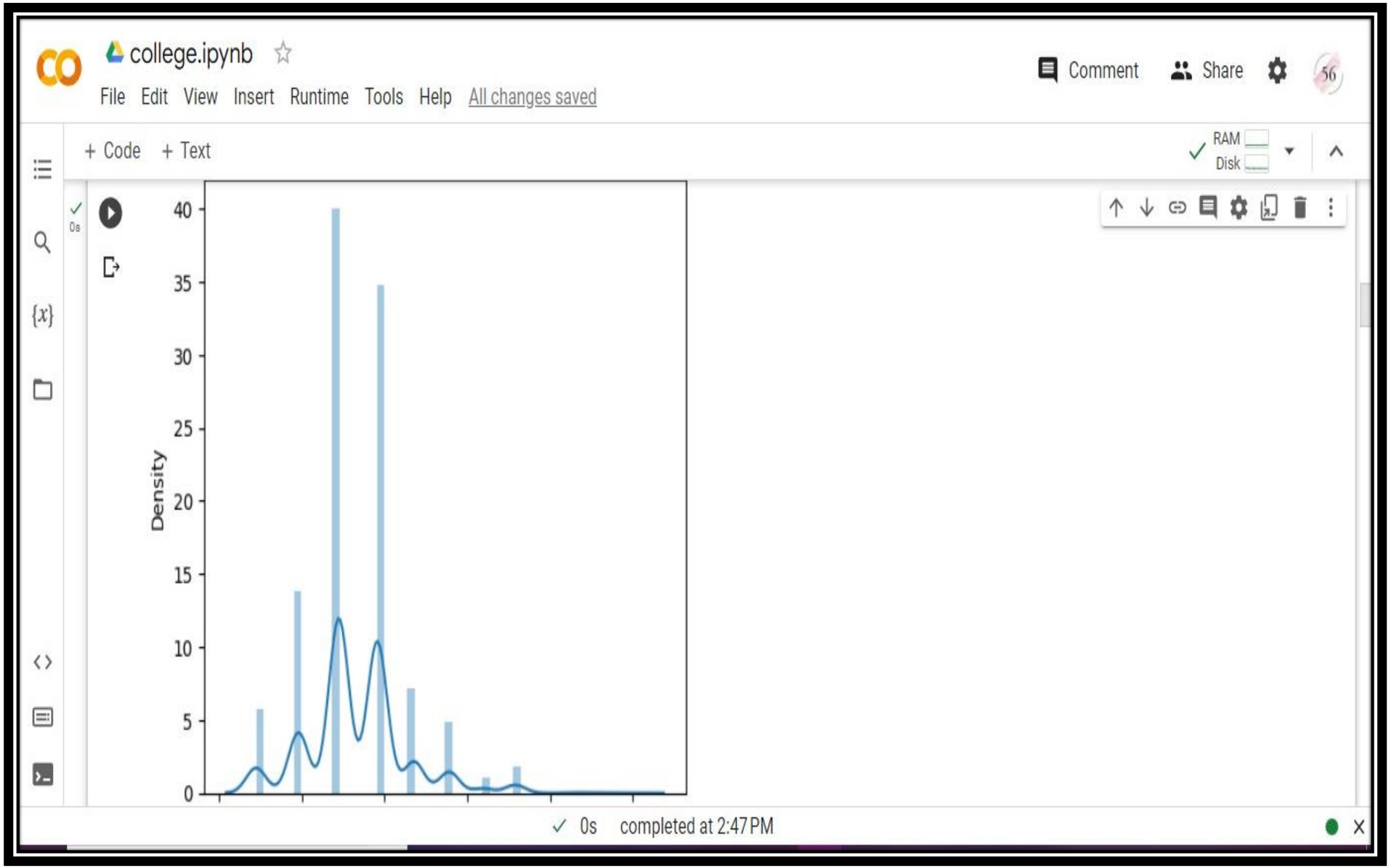
0s

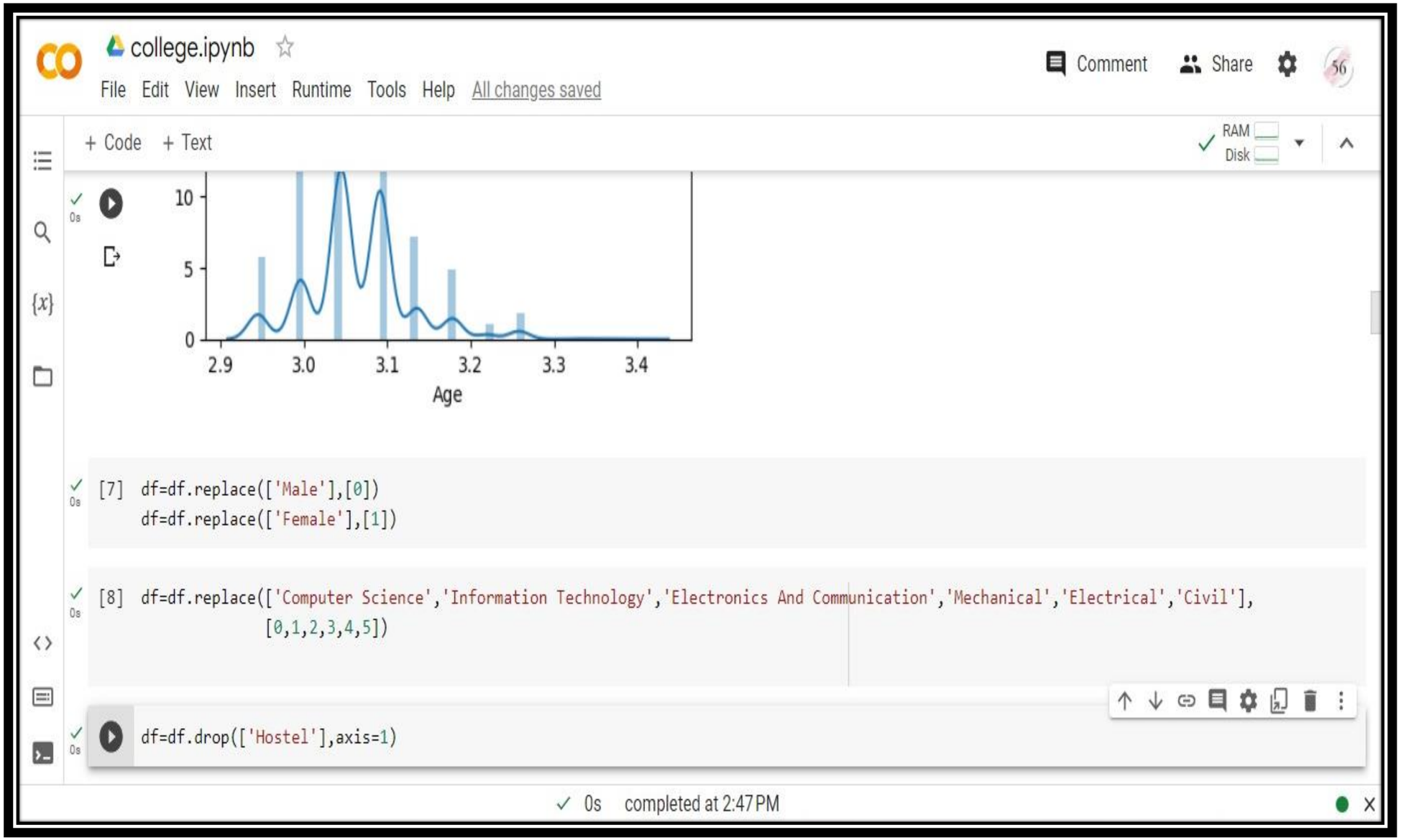
```
[6] transformationplot(np.log(df['Age']))
```

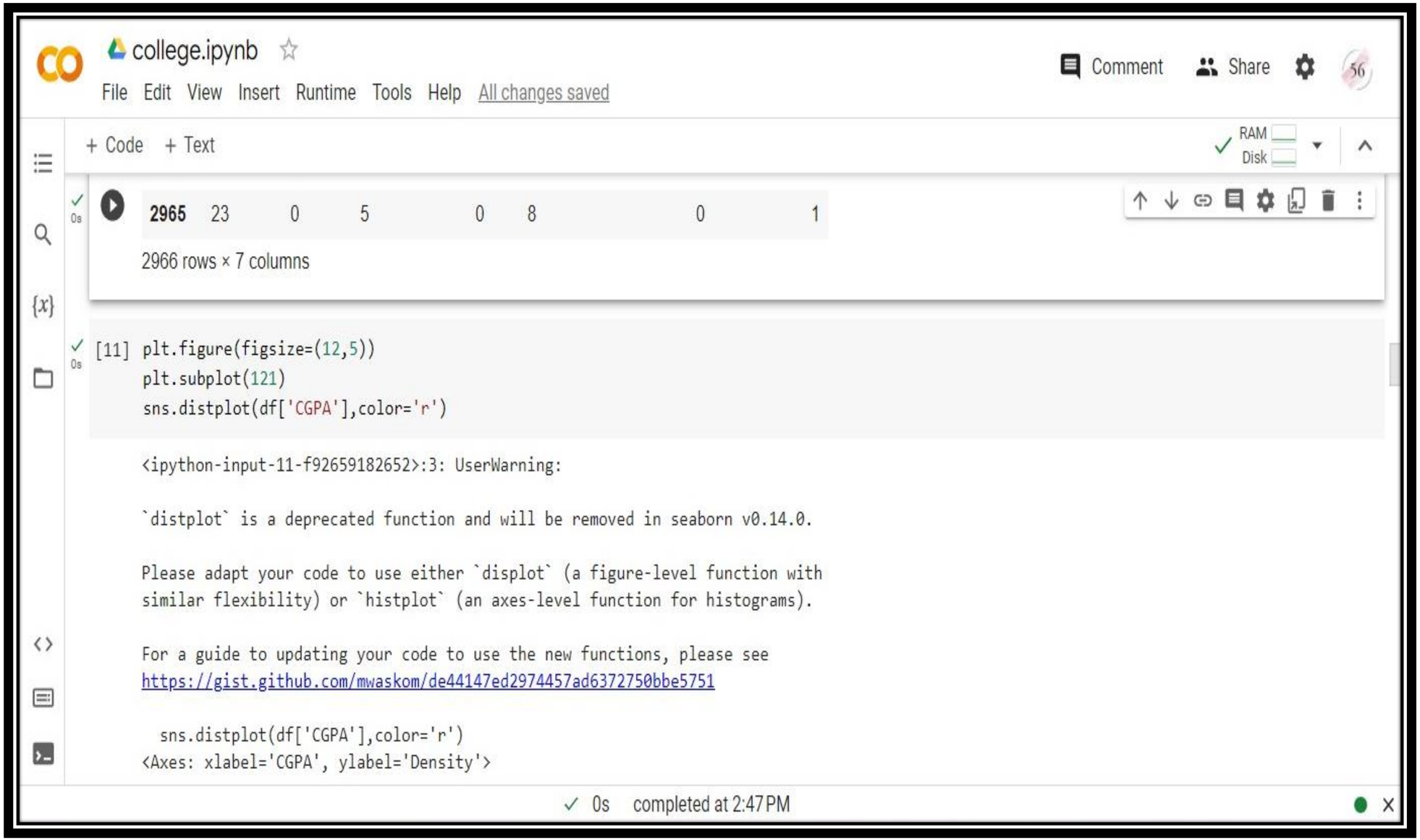
0s

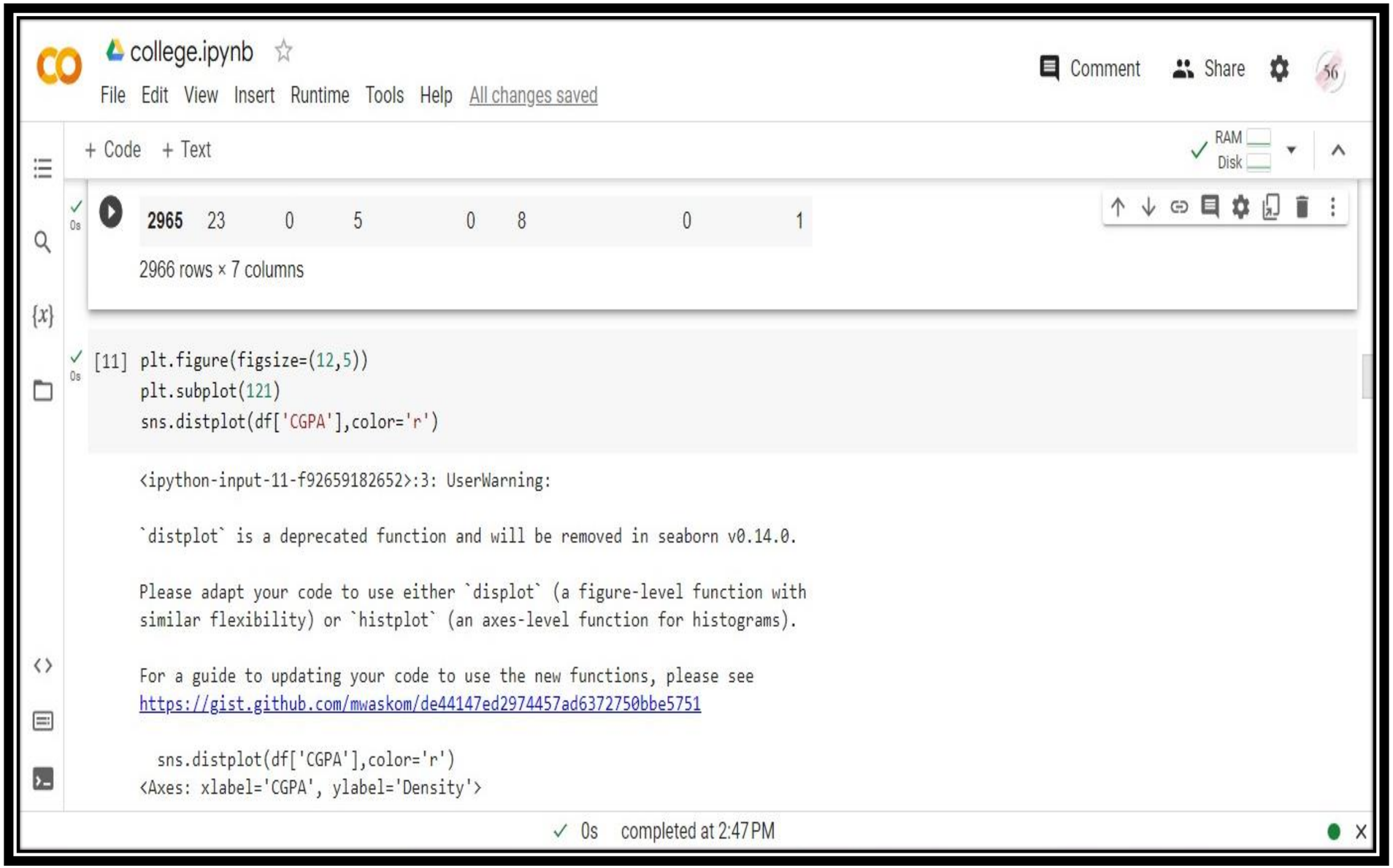
```
<ipython-input-5-4e023cd5df85>:4: UserWarning:

`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(feature)
```

✓ 0s completed at 2:47 PM


college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
56
+ Code + Text
RAM
Disk
0s
40
35
30
25
20
15
10
5
0
Density
✓ 0s completed at 2:47 PM


college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
56
+ Code + Text
RAM
Disk
10
5
0
2.9
3.0
3.1
3.2
3.3
3.4
Age
[7] df=df.replace(['Male'],[0])
df=df.replace(['Female'],[1])
[8] df=df.replace(['Computer Science','Information Technology','Electronics And Communication','Mechanical','Electrical','Civil'],
[0,1,2,3,4,5])
df=df.drop(['Hostel'],axis=1)
0s completed at 2:47 PM


college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
56
+ Code + Text
RAM
Disk
2965 23 0 5 0 8 0 1
2966 rows × 7 columns
[11] plt.figure(figsize=(12,5))
plt.subplot(121)
sns.distplot(df['CGPA'],color='r')
<ipython-input-11-f92659182652>:3: 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(df['CGPA'],color='r')
<Axes: xlabel='CGPA', ylabel='Density'>
0s completed at 2:47 PM


college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
56
+ Code + Text
RAM
Disk
2965 23 0 5 0 8 0 1
2966 rows × 7 columns
[11] plt.figure(figsize=(12,5))
plt.subplot(121)
sns.distplot(df['CGPA'],color='r')
<ipython-input-11-f92659182652>:3: 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(df['CGPA'],color='r')
<Axes: xlabel='CGPA', ylabel='Density'>
0s completed at 2:47 PM

college.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share 56

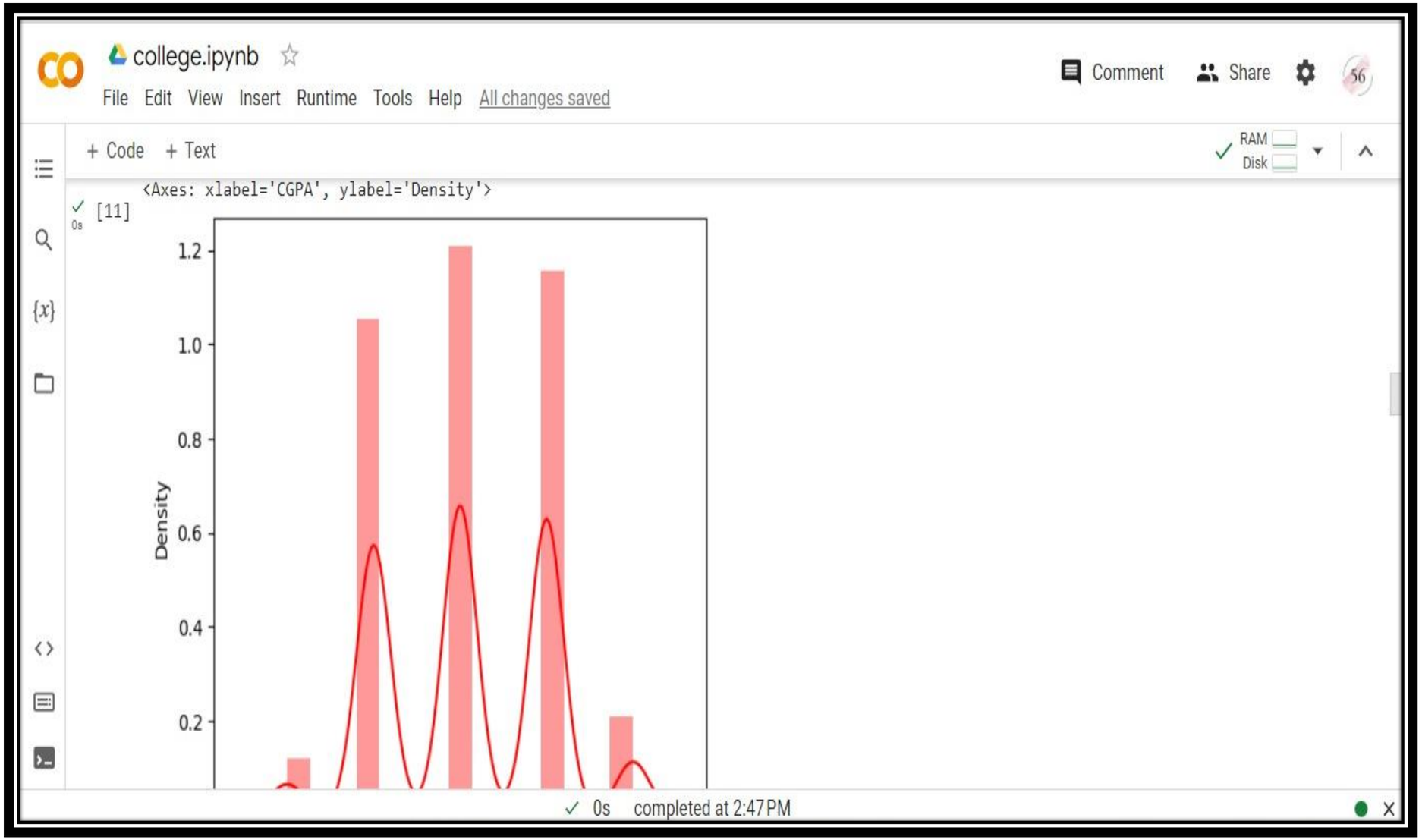
+ Code + Text

RAM Disk

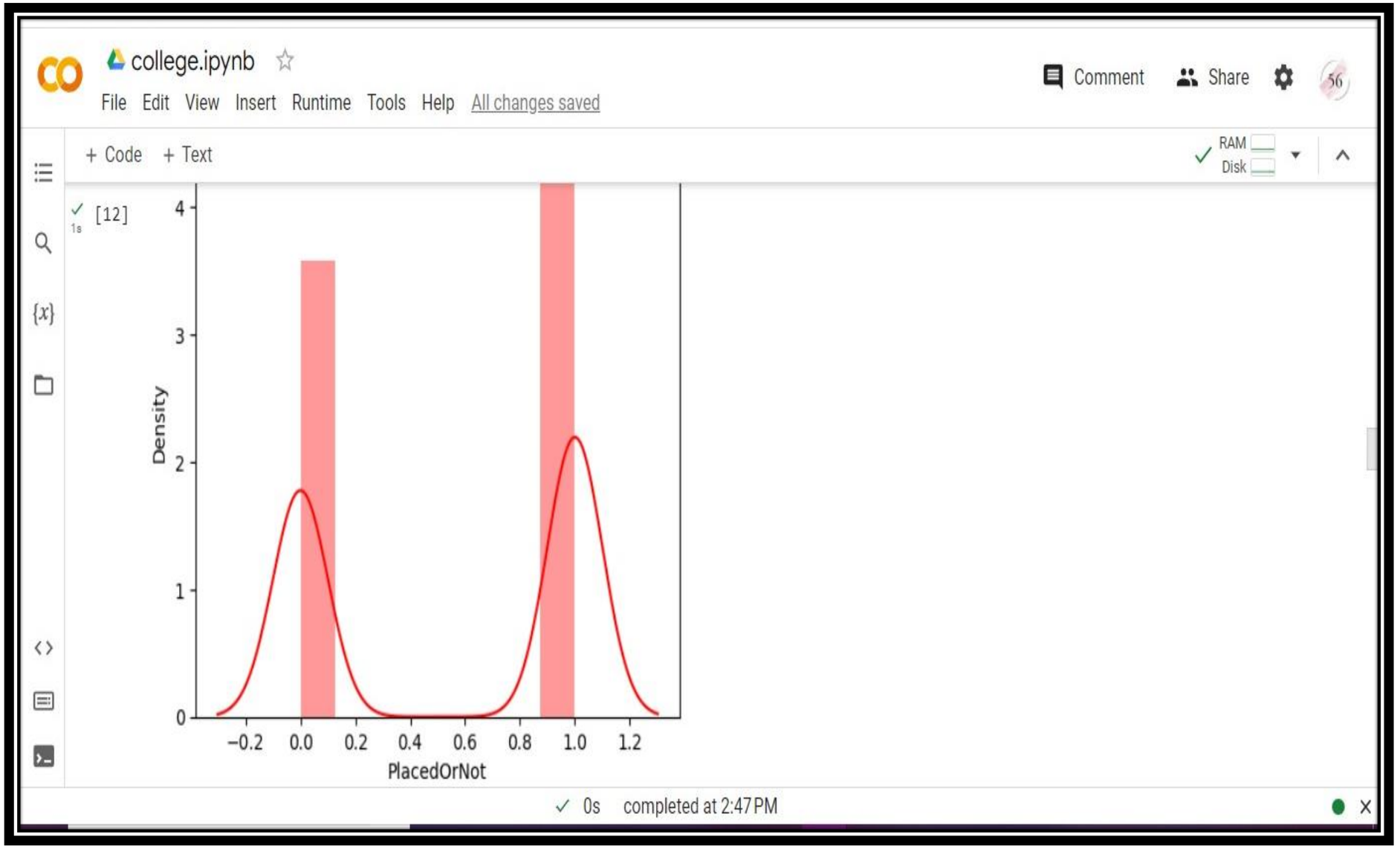
```
df
```

	Age	Gender	Stream	Internships	CGPA	HistoryOfBacklogs	PlacedOrNot
0	22	0	2	1	8	1	1
1	21	1	0	0	7	1	1
2	22	1	1	1	6	0	1
3	21	0	1	0	8	1	1
4	22	0	3	0	8	0	1
...	...	...	...	...	...	...	...
2961	23	0	1	0	7	0	0
2962	23	0	3	1	7	0	0
2963	22	0	1	1	7	0	0
2964	22	0	0	1	7	0	0

✓ 0s completed at 2:47 PM


college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
56
+ Code + Text
RAM
Disk
[11]
0s
<Axes: xlabel='CGPA', ylabel='Density'>
Density
1.2
1.0
0.8
0.6
0.4
0.2
0s completed at 2:47 PM


college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
56
+ Code + Text
RAM
Disk
[12]
plt.figure(figsize=(12,5))
plt.subplot(121)
sns.distplot(df['PlacedOrNot'],color='r')
<ipython-input-12-5e468beb8a0d>:3: 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(df['PlacedOrNot'],color='r')
<Axes: xlabel='PlacedOrNot', ylabel='Density'>
4
0s completed at 2:47 PM


college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
56
+ Code + Text
RAM
Disk
[12]
1s
Density
4
3
2
1
0
−0.2
0.0
0.2
0.4
0.6
0.8
1.0
1.2
PlacedOrNot
0s completed at 2:47 PM

college.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

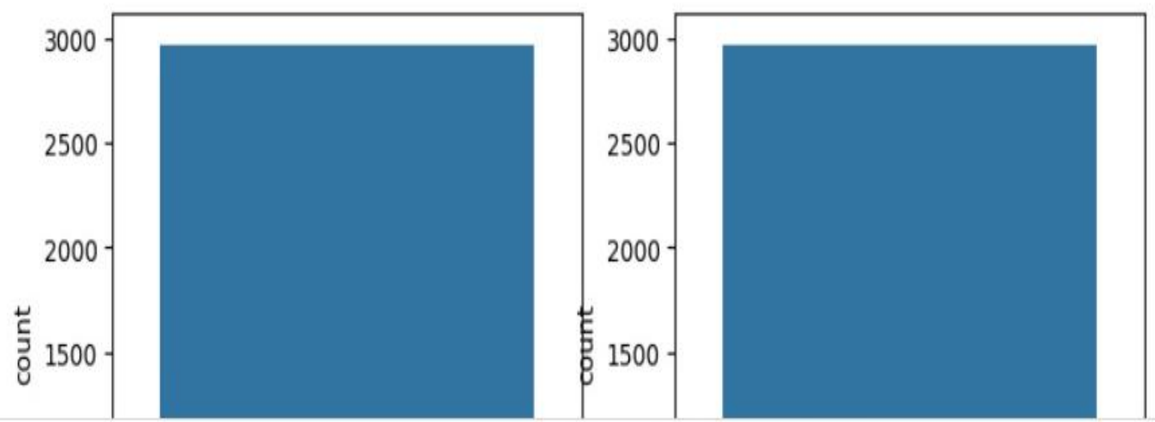
Comment Share 56

+ Code + Text

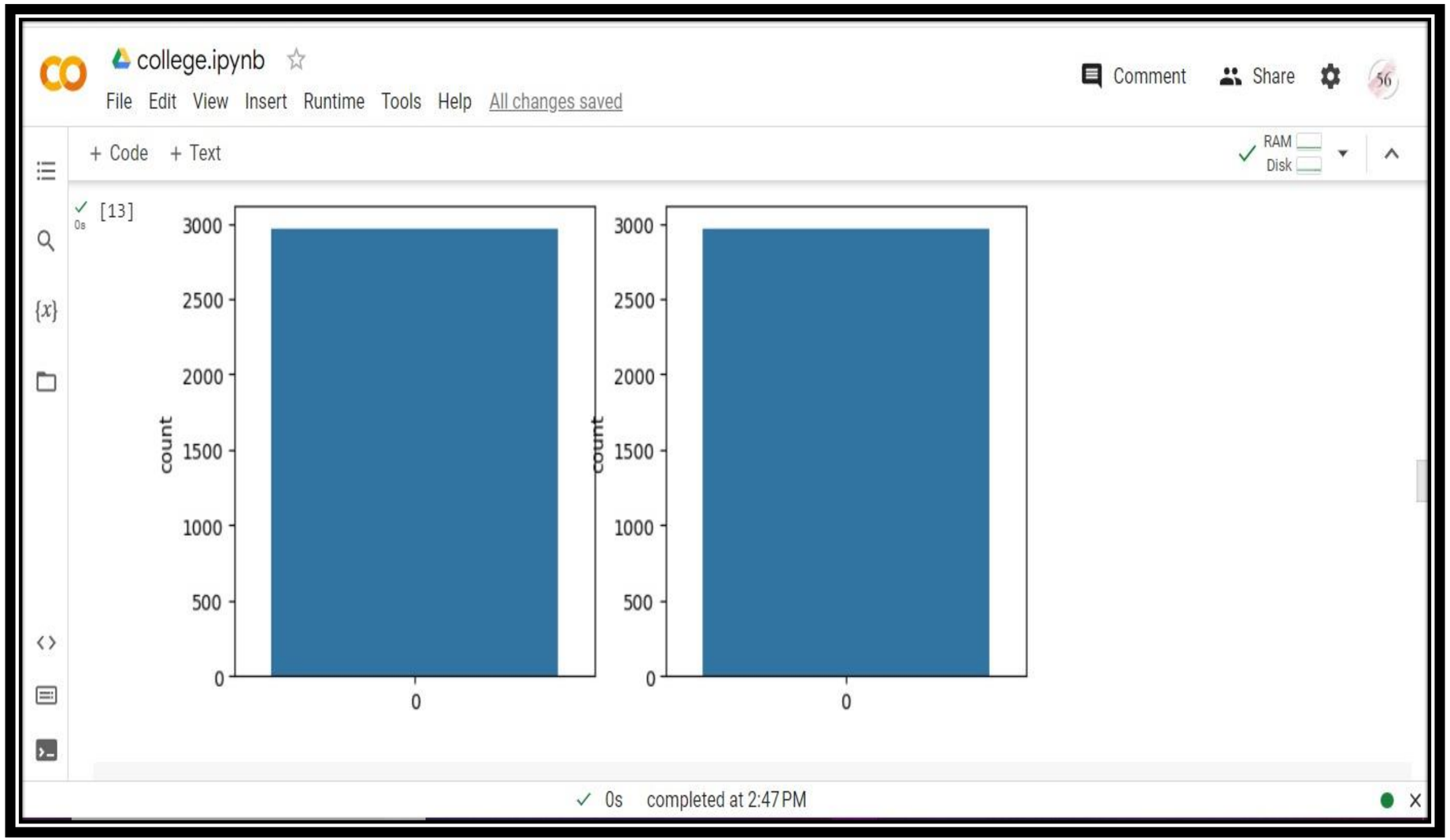
RAM Disk

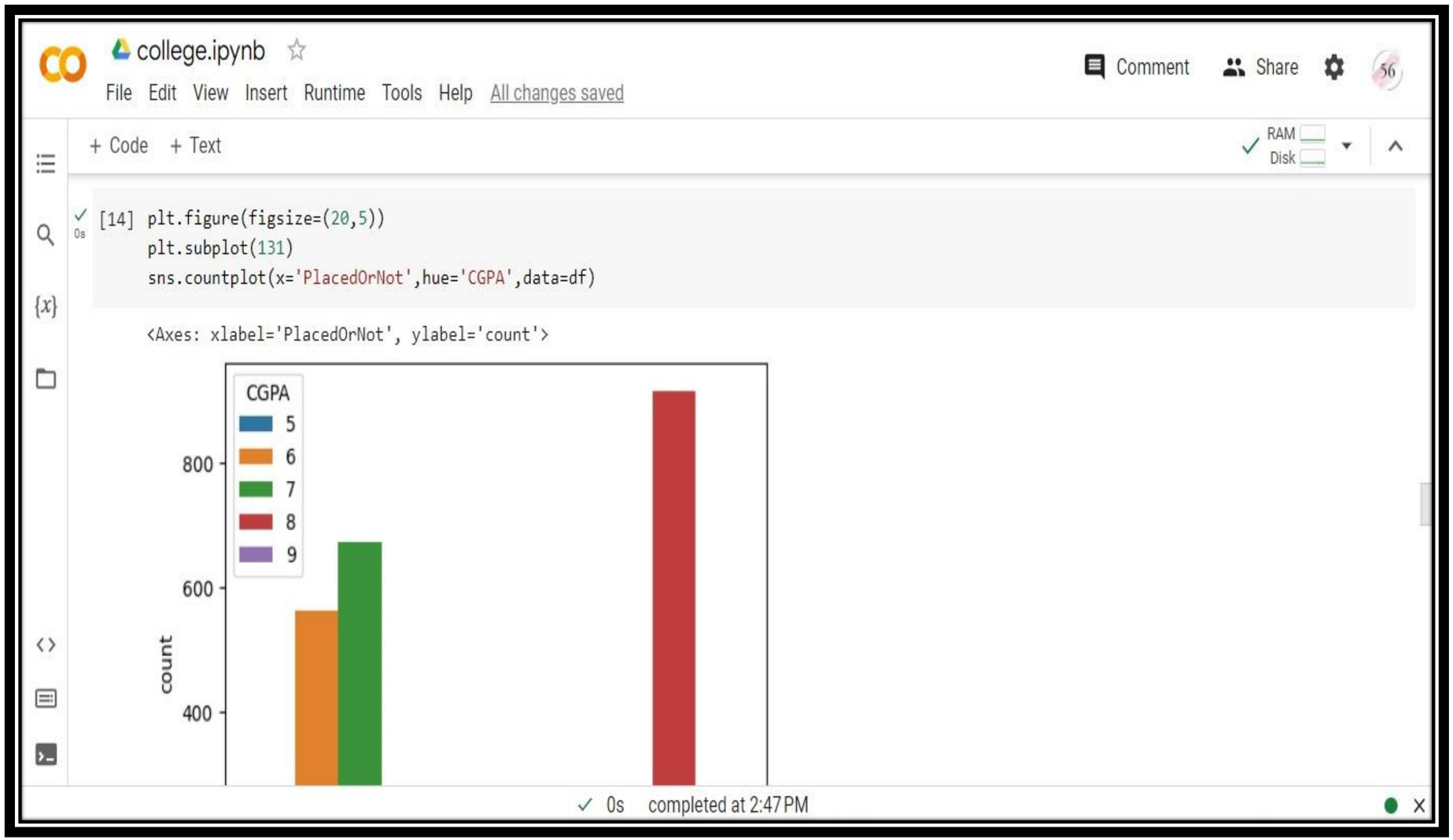
PlacedOrNot

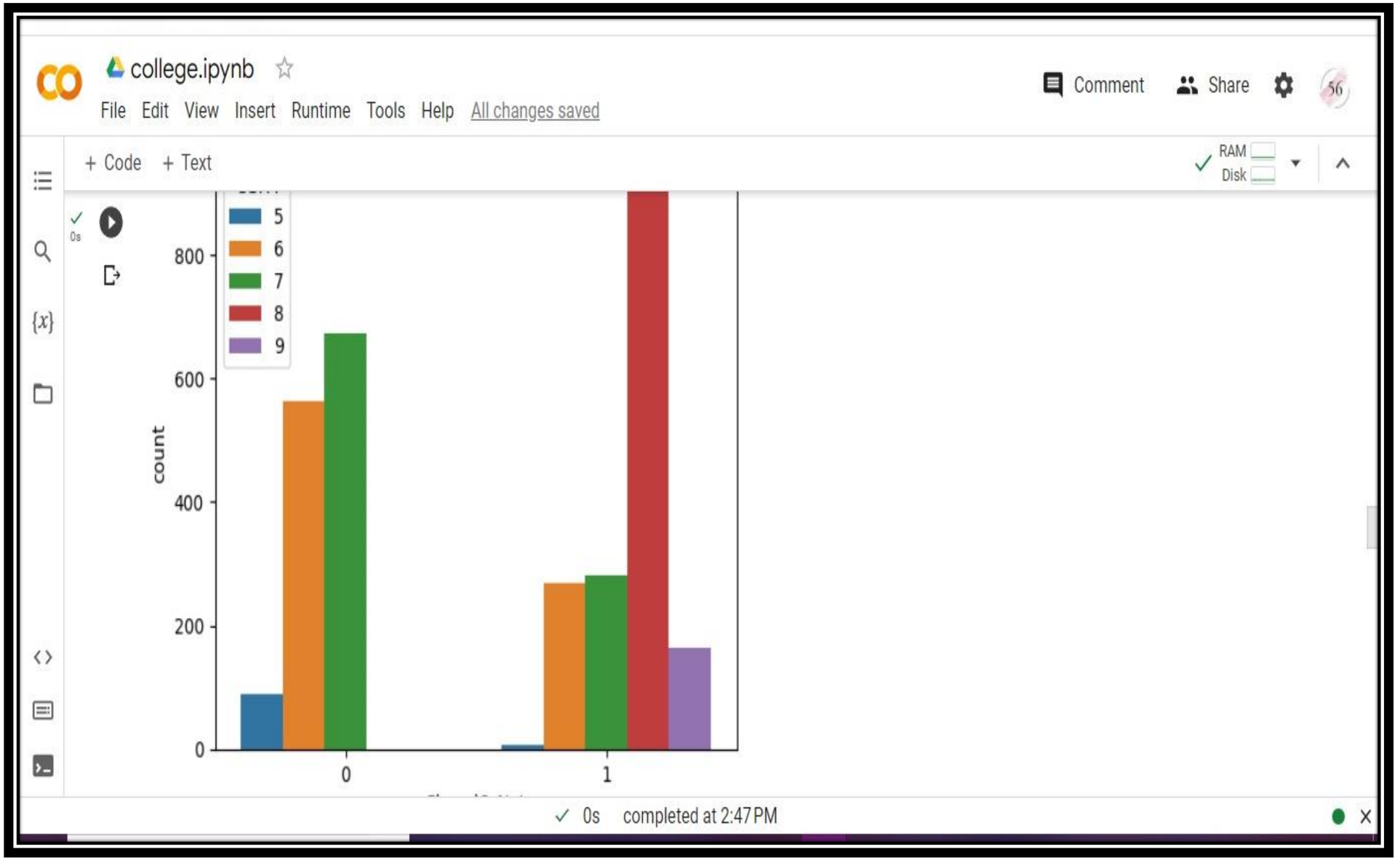
```
[13] plt.figure(figsize=(18,4))
     plt.subplot(1,4,1)
     sns.countplot(df['Gender'])
     plt.subplot(1,4,2)
     sns.countplot(df['Stream'])
     plt.show()
```

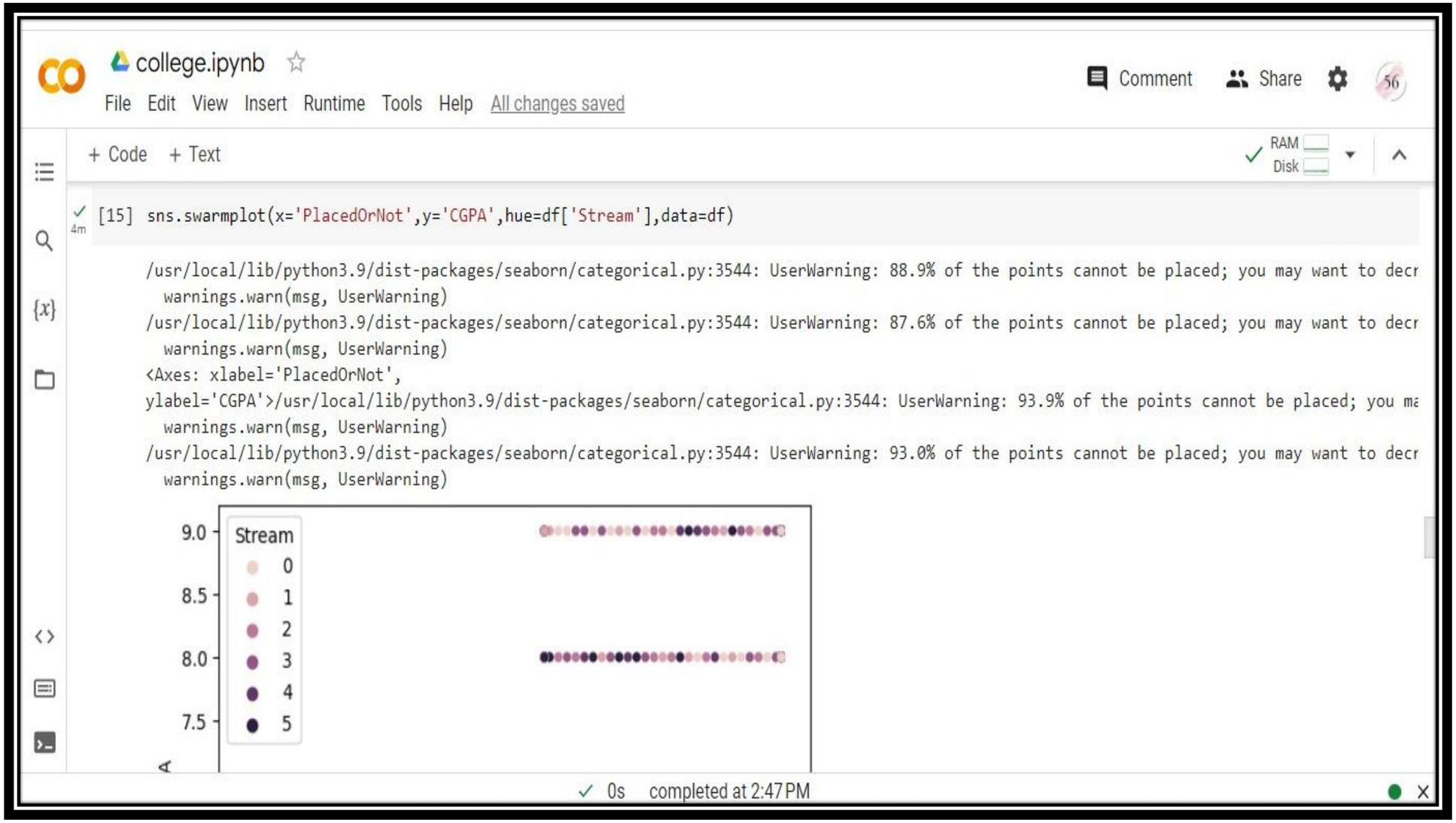


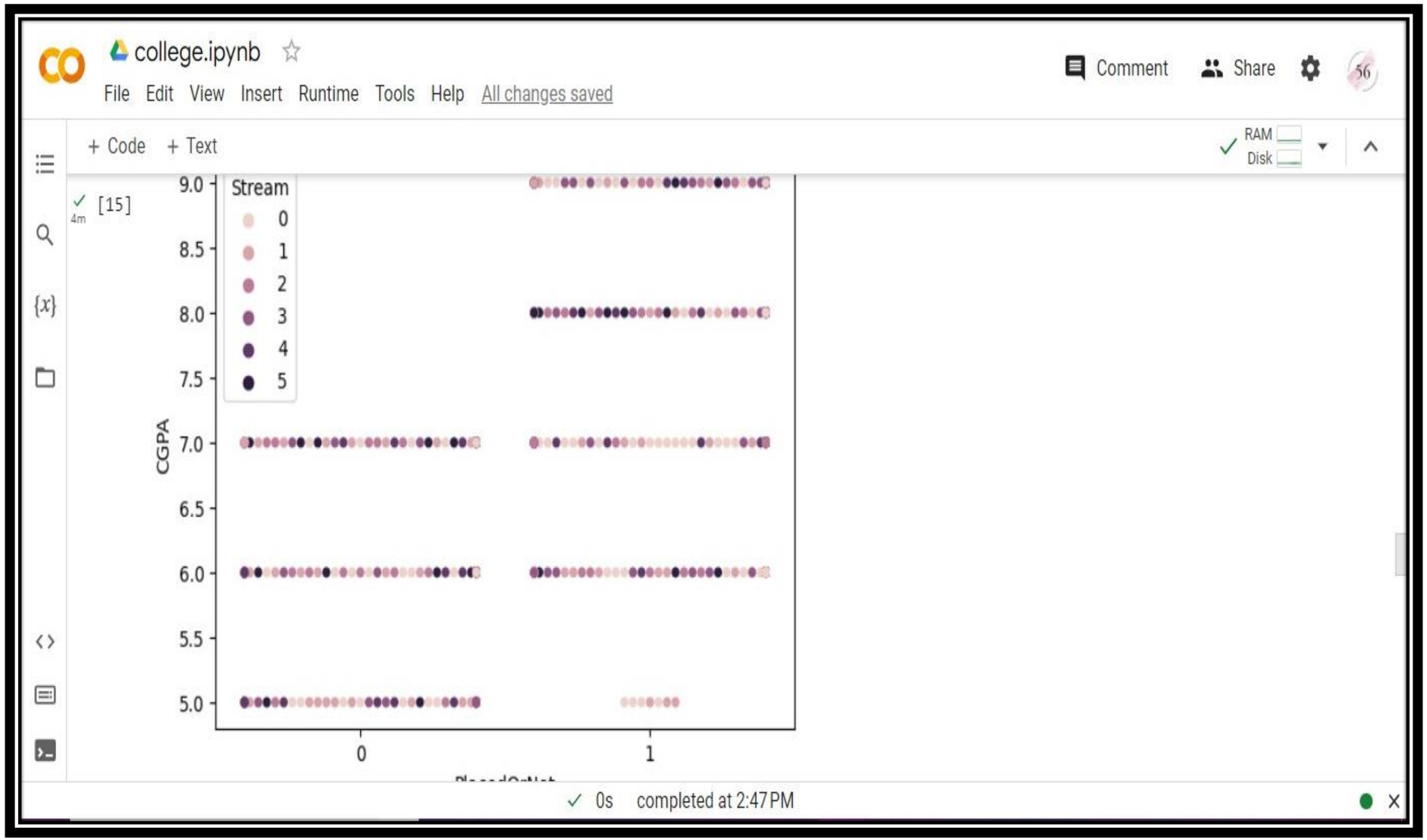
✓ 0s completed at 2:47 PM

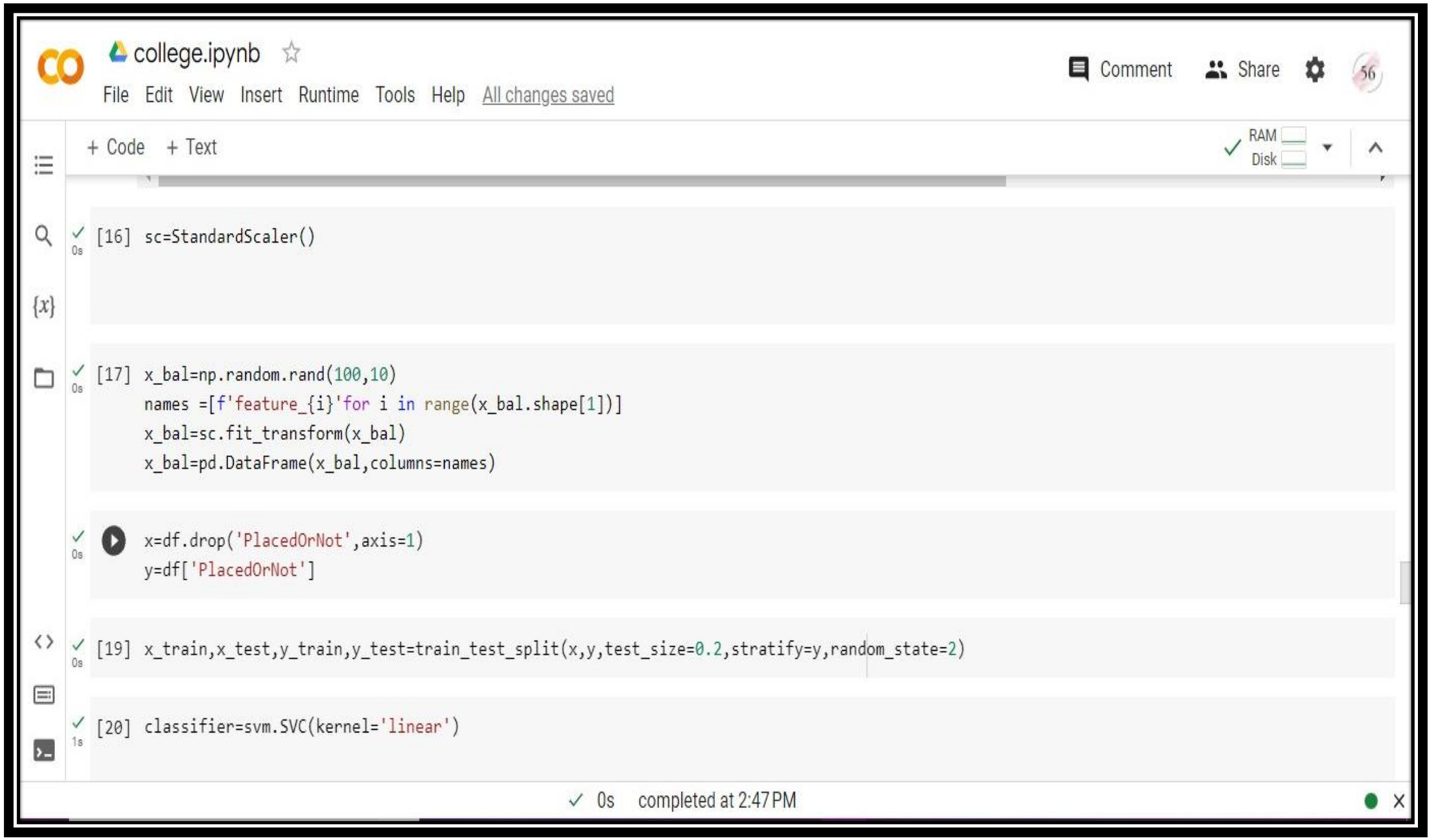

college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
56
+ Code + Text
RAM
Disk
[13]
0s
3000
2500
2000
1500
1000
500
0
count
0
completed at 2:47 PM

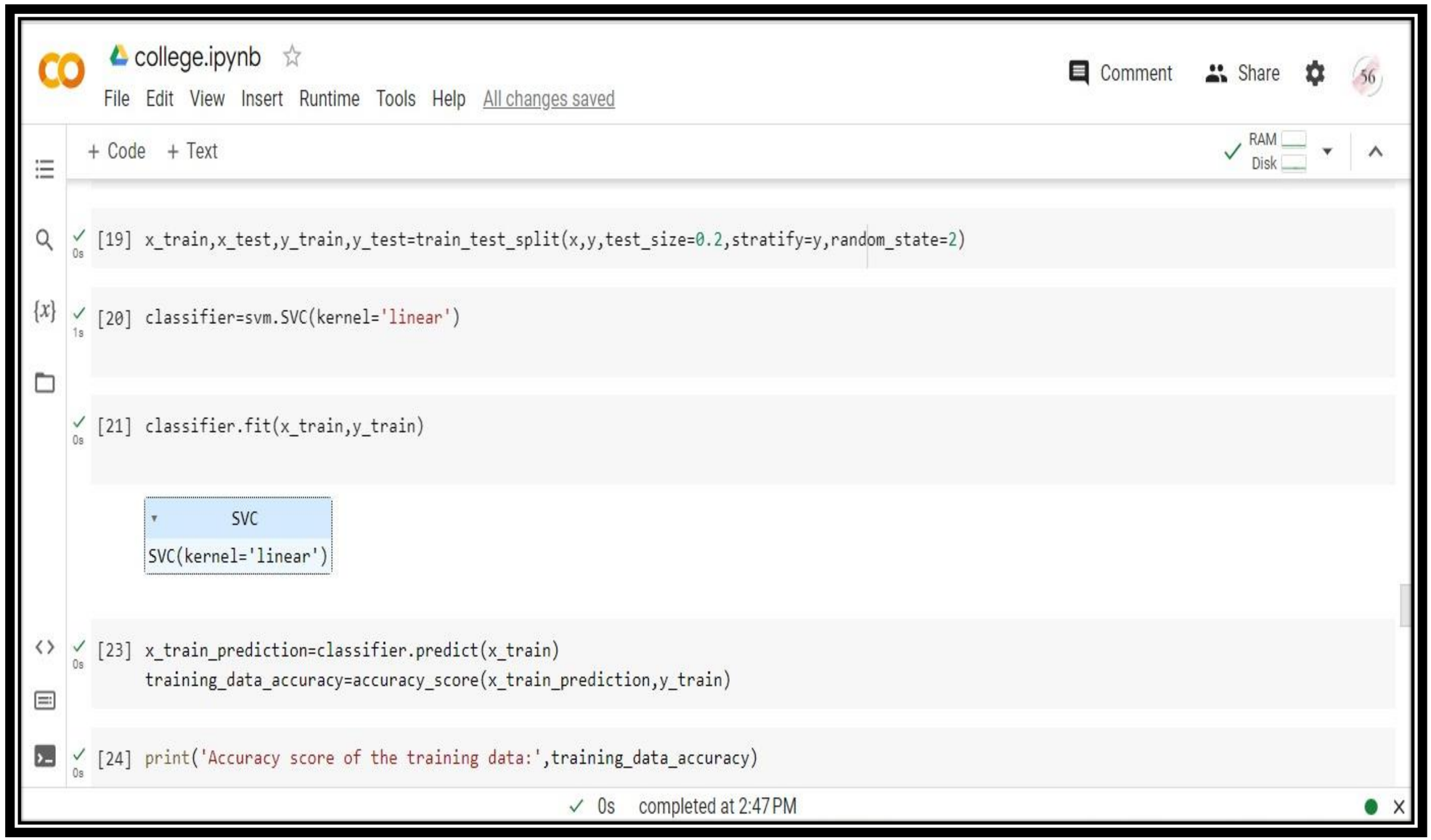

college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
[14] plt.figure(figsize=(20,5))
plt.subplot(131)
sns.countplot(x='PlacedOrNot',hue='CGPA',data=df)
<Axes: xlabel='PlacedOrNot', ylabel='count'>
CGPA
5
6
7
8
9
800
600
400
count
0s completed at 2:47 PM


college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
56
+ Code + Text
RAM
Disk
0s
5
6
7
8
9
800
600
400
200
0
count
0
1
0s completed at 2:47PM


college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
[15] sns.swarmplot(x='PlacedOrNot',y='CGPA',hue=df['Stream'],data=df)
4m
/usr/local/lib/python3.9/dist-packages/seaborn/categorical.py:3544: UserWarning: 88.9% of the points cannot be placed; you may want to decr
warnings.warn(msg, UserWarning)
/usr/local/lib/python3.9/dist-packages/seaborn/categorical.py:3544: UserWarning: 87.6% of the points cannot be placed; you may want to decr
warnings.warn(msg, UserWarning)
<Axes: xlabel='PlacedOrNot',
ylabel='CGPA'>/usr/local/lib/python3.9/dist-packages/seaborn/categorical.py:3544: UserWarning: 93.9% of the points cannot be placed; you ma
warnings.warn(msg, UserWarning)
/usr/local/lib/python3.9/dist-packages/seaborn/categorical.py:3544: UserWarning: 93.0% of the points cannot be placed; you may want to decr
warnings.warn(msg, UserWarning)
9.0
8.5
8.0
7.5
Stream
0
1
2
3
4
5
0s completed at 2:47 PM


college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
56
+ Code + Text
RAM
Disk
[15]
4m
Stream
0
1
2
3
4
5
9.0
8.5
8.0
7.5
7.0
6.5
6.0
5.5
5.0
CGPA
0
1
0s completed at 2:47 PM


college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
56
+ Code + Text
RAM
Disk
[16] sc=StandardScaler()
[17] x_bal=np.random.rand(100,10)
names =[f'feature_{i}'for i in range(x_bal.shape[1])]
x_bal=sc.fit_transform(x_bal)
x_bal=pd.DataFrame(x_bal,columns=names)
x=df.drop('PlacedOrNot',axis=1)
y=df['PlacedOrNot']
[19] x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,stratify=y,random_state=2)
[20] classifier=svm.SVC(kernel='linear')
0s completed at 2:47 PM


college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
56
+ Code + Text
RAM
Disk
[19] x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,stratify=y,random_state=2)
0s
[20] classifier=svm.SVC(kernel='linear')
1s
[21] classifier.fit(x_train,y_train)
0s
SVC
SVC(kernel='linear')
[23] x_train_prediction=classifier.predict(x_train)
training_data_accuracy=accuracy_score(x_train_prediction,y_train)
0s
[24] print('Accuracy score of the training data:',training_data_accuracy)
0s
0s completed at 2:47PM

college.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share 56

+ Code + Text

RAM Disk

```
[24] print('Accuracy score of the training data:',training_data_accuracy)
```

```
Accuracy score of the training data: 0.7685497470489039
```

```
[25] best_k={"Regular":0}
     best_score={"Regular":0}
     for k in range(3,50,2):
       knn_temp = KNeighborsClassifier(n_neighbors=k)
     knn_temp.fit(x_train,y_train)
     knn_temp_pred = knn_temp.predict(x_test)
     score = metrics.accuracy_score(y_test, knn_temp_pred)*100
     if score >= best_score["Regular"]and score < 100:
          best_score["Regular"]=score
          best_k["Regular"]=k
```

```
[26] print("---Results---\nk: {}".format(best_k,best_score))
     knn=KNeighborsClassifier(n_neighbors=best_k["Regular"])
```

✓ 0s completed at 2:47 PM

college.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

Comment Share 56

+ Code + Text

RAM Disk

```
[26] print("---Results---\nk: {}".format(best_k,best_score))
     knn=KNeighborsClassifier(n_neighbors=best_k["Regular"])
     knn.fit(x_train,y_train)
     knn_pred=knn.predict(x_test)
     testd = accuracy_score(knn_pred,y_test)
```

```
---Results---
k: {'Regular': 49}
```

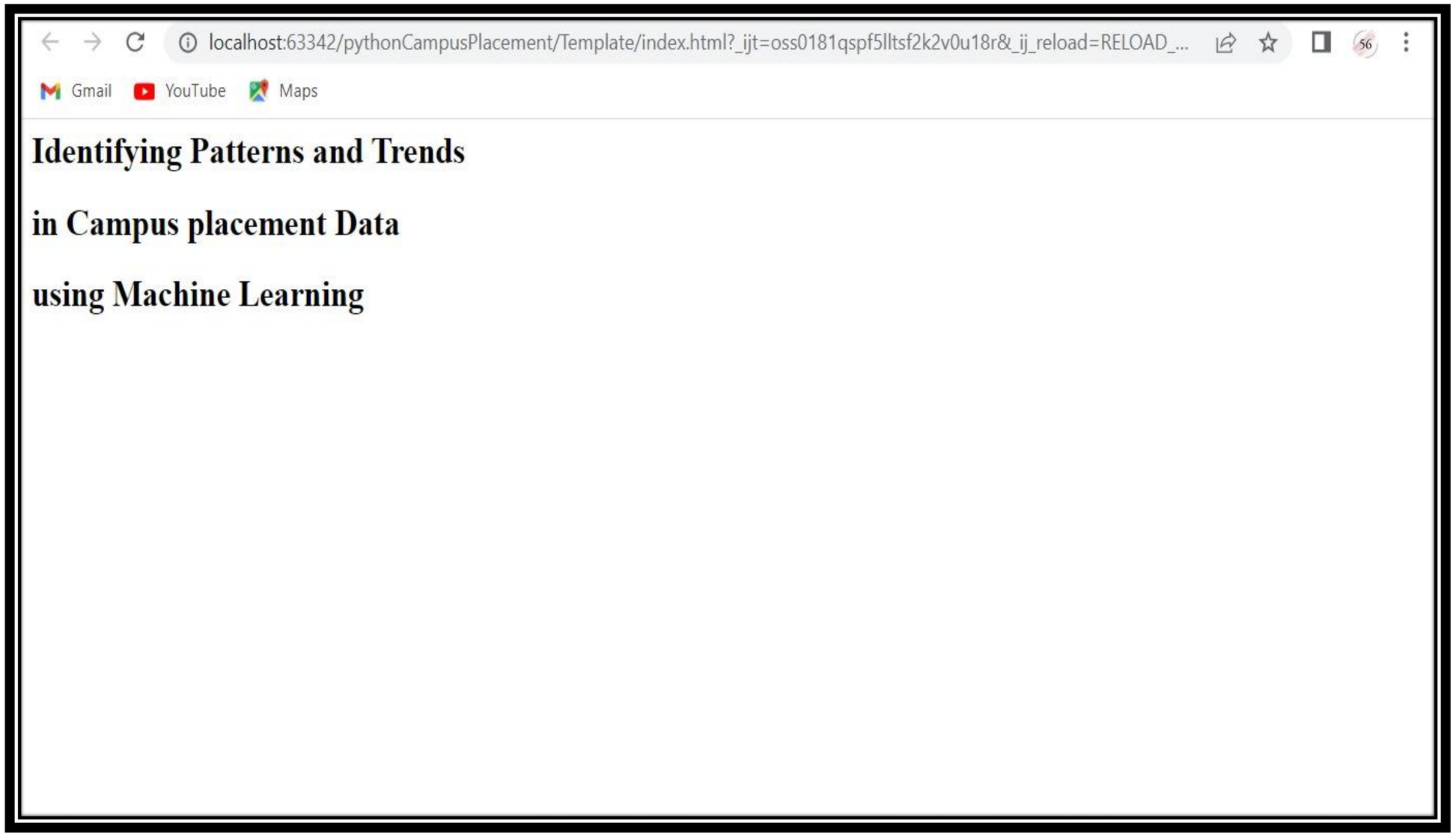
```
[27] import tensorflow as tf
     from tensorflow import keras
     from keras.models import Sequential
     from tensorflow.keras import layers
```

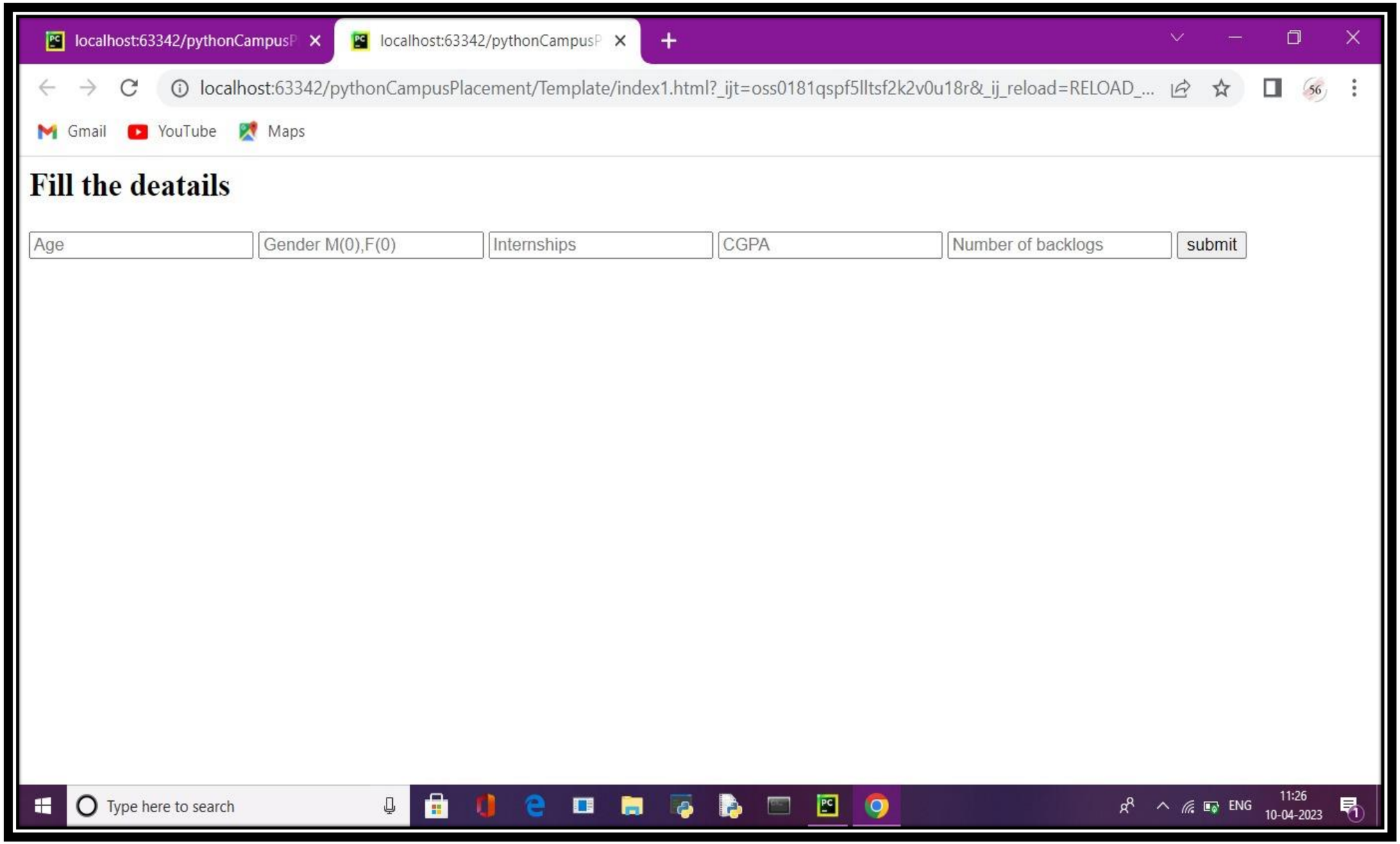
```
[28] classifier = Sequential()
     classifier.add(keras.layers.Dense(6,activation = 'relu',input_dim=6))
     classifier.add(keras.layers.Dropout(0.50))
     classifier.add(keras.layers.Dense(6,activation='relu'))
     classifier.add(keras.layers.Dropout(0.50))
```

✓ 0s completed at 2:47 PM


college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
56
+ Code + Text
RAM
Disk
[30] loss_1=tf.keras.losses.BinaryCrossentropy()
classifier.compile(optimizer = 'Adam',loss = loss_1 , metrics=['accuracy'])
[31] classifier.fit(x_train,y_train,batch_size=20,epochs = 100)
119/119 [==============================] - 0s 2ms/step - loss: 0.6444 - accuracy: 0.6244
Epoch 73/100
119/119 [==============================] - 0s 2ms/step - loss: 0.6434 - accuracy: 0.6244
Epoch 74/100
119/119 [==============================] - 0s 2ms/step - loss: 0.6385 - accuracy: 0.6239
Epoch 75/100
119/119 [==============================] - 0s 2ms/step - loss: 0.6343 - accuracy: 0.6294
Epoch 76/100
119/119 [==============================] - 0s 2ms/step - loss: 0.6381 - accuracy: 0.6341
Epoch 77/100
119/119 [==============================] - 0s 2ms/step - loss: 0.6398 - accuracy: 0.6336
Epoch 78/100
119/119 [==============================] - 0s 2ms/step - loss: 0.6322 - accuracy: 0.6324
Epoch 79/100
119/119 [==============================] - 0s 2ms/step - loss: 0.6327 - accuracy: 0.6383
0s completed at 2:47 PM


college.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
56
+ Code + Text
RAM
Disk
[31] Epoch 98/100
119/119 [==============================] - 0s 2ms/step - loss: 0.6184 - accuracy: 0.6446
Epoch 99/100
119/119 [==============================] - 0s 2ms/step - loss: 0.6108 - accuracy: 0.6513
Epoch 100/100
119/119 [==============================] - 0s 2ms/step - loss: 0.6132 - accuracy: 0.6324
<keras.callbacks.History at 0x7f13b433ceb0>
[32] import pickle
pickle.dump(knn,open("placement.pkl",'wb'))
model=pickle.load(open('placement.pkl','rb'))
0s completed at 2:47 PM


localhost:63342/pythonCampusPlacement/Template/index.html?_ijt=oss0181qspf5lltsf2k2v0u18r&_ij_reload=RELOAD_...
Gmail
YouTube
Maps
Identifying Patterns and Trends
in Campus placement Data
using Machine Learning


localhost:63342/pythonCampusP
localhost:63342/pythonCampusP
localhost:63342/pythonCampusPlacement/Template/index1.html?_ijt=oss0181qspf5lltsf2k2v0u18r&_ij_reload=RELOAD_...
Gmail
YouTube
Maps
Fill the deatails
Age
Gender M(0),F(0)
Internships
CGPA
Number of backlogs
submit
Type here to search
ENG
11:26
10-04-2023

4.ADVANTAGES

- **Increased accuracy**
- **Improved efficiency**
- **Saves Time & Efforts.**
- **Improved Retention Rates.**
- **Getting New Knowledge & Skills.**
- **Quick Learners & Multi-tasking candidates.**
- **Good relationship between Organization & Campus.**
- **High Volume of Talent Pool. ...**
- **Resumes are the only way to select a candidate.**
- **Limited Staff & Time.**

5.CONCLUSION

- Conclusion for Identifying Patterns and Trends in Campus Placement Data using Machine Learning

- Identifying patterns and trends in campus placement data using machine learning can provide valuable insights to educational institutions and employers to improve their recruitment processes and prepare students for the job market. By analyzing factors such as academic performance, skills, and demographics, machine learning algorithms can identify correlations and make predictions about which students are more likely to be successful in obtaining employment.

- Furthermore, this data can help educational institutions tailor their curriculum to meet the demands of employers and the job market. By analyzing which skills and traits are in high demand, institutions can adjust their programs to better prepare students for their careers.

- Overall, the use of machine learning in analyzing campus placement data can provide significant benefits for both educational institutions and employers, ultimately leading to improved job outcomes for students and a more efficient recruitment process for employers.

6.FUTURE ENHANCEMENT

- Incorporate natural language processing (NLP): Many campus placement reports include written feedback from both employers and students. By incorporating NLP techniques, machine learning algorithms could extract insights from this unstructured data to identify patterns and trends in what employers are looking for in candidates and how students are responding to their job offers.

- Use graph analysis techniques: Campus placement data typically involves complex relationships between multiple variables such as colleges, companies, job roles, and students. Graph analysis techniques such as network analysis and graph clustering could be used to identify patterns and trends in these relationships.