

IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING

1.INTRODUCTION

Campus Placements/ Campus Recruitment drives are conducted in various educational institutes for providing job opportunities to the students. Campus Recruitment allows students to earn a safe and secure future. Keeping in mind the importance of the placement, it is vital for a student to prepare adequately for these programs and ensure that they put their best foot forward.

2.OVERVIEW

We are sure you all will be aware of what is campus interview is. But, are you well informed about the type of campus placement?

In general, there are mainly two types of campus placements that can be classified under the existing campus placement models. They are:

On-Campus Placement:

In On-Campus placement drives, recruitment companies are officially invited on the college campus to conduct interviews to gauge their potential as future employees. The placement process is centralized. Before the interviews, they make a selection on criteria like student's knowledge, technical abilities, and zeal to work.

Off-Campus Placement:

As the name defines, it is a type of placement conducted outside the college campus. In Off-campus placement, the role of college is absent. This means that there are no rules and regulations of the college, and seek employment on your own. Reputed companies conduct pool placement interviews where students from different colleges assemble at a commonplace.

Students who have failed to get an on-campus placement can get a career opportunity through this type of placement. Here, an individual need to take a lot of effort to find their perfect employment fit.

Pool Campus Placement:

This type of campus placement considers a specific affiliation criterion. The concept of the pool campus placement enables a student to attend campus interviews held at colleges that are affiliated with the same university. Pool Placements adds an advantage for the students of the colleges to join the combined placement drives, which are similar to on-campus drives with a larger scale. Here, one gets to explore a lot of career choices to opt for.

On-Campus Vs. Off-Campus Vs. Pool Campus Placement

The ultimate purpose of the placement drives is to hire candidates that fit the organization's requirements. However, every college campus recruitment has its own pros and cons.

If you talk about on-campus vs. off-campus recruitment drives, on-campus placement focuses on saving your time, provides you limited competition that increases the higher success probabilities, dedicated assistance from the respective college placement team that guarantees placement. Whereas, in off-campus recruitment, one gets to explore an infinite range of opportunities without any interference from the college and feel a sense of self-achievement after getting placed. Apart from pros, off-campus has a few cons, like facing more challenges during the placement process and limited career prospects. Pool campus placement paves the way for

IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING

creating larger talent; therefore, there are higher chances for selection. As this approach encourages the participation of a number of colleges and students, it brings diversity to the company as a whole. On the other side, Pool campus placement involves a large number of participants, which makes appropriate evaluation of the real talent difficult. So, this approach requires detailed planning and precision to manage this placement process efficiently

3.PURPOSE

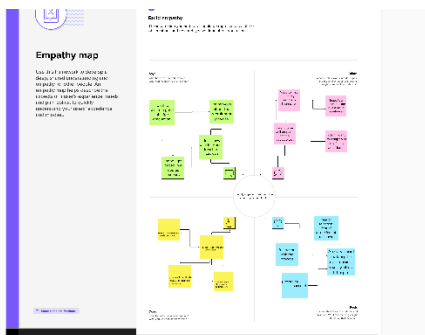
- To identify the talented and qualified students in the college.
- To create promising career opportunities for students in reputed corporate companies.
- To select candidates who are suitable for the current job roles without any biased behaviour.
- To provide roles and duties as per the student's knowledge, expertise, and interest.
- To ensure students start a career and move forward in the right direction for better quality living.
- To provide ultimate satisfaction to students by offering the companies of their choice according to their eligibility.
- To provide career guidance through counselling and interactions with industry experts.
- To evaluate and select the right candidate to meet the organization's requirements.
- To identify the professional traits, real-time skills, and values within the students.

4.PROBLEM DEFINITION & DESIGN THINKING

EMPATHY MAP



campus placement_2023-03-26_09-18-54.pdf



IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING

4.1 IDEATION & BRAINSTORMING MAP



campus_2023-03-30_05-46-15.pdf

Person 1

develop a centralized platform that connects students and recruiters based on their preferences and qualifications

Person 2

create a mentorship program where industry professionals can guide and mentor students in their career paths

Person 3

conduct regular skills assessment tests to identify student strengths and weaknesses and provide personalized training and development programs

Person 4

host career fairs and networking events that bring students and recruiters together in a meaningful way

Person 5

offer internships and apprenticeships that provide hands-on experience to students and bridge the gap between academic knowledge and practical skills

Person 6

provide career counseling and guidance services to help students make informed decisions about their future

Person 7

partner with industry associations and chambers of commerce to align curriculum and industry requirements

Person 8

introduce gamification elements to make the placement process more engaging and interactive for students and recruiters

IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING

5.RESULT

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import sklearn

placement = pd.read_csv("Placement_Data_Full_Class.csv")

placement.head(10)
```

sl_no	gender	ssc_p	ssc_b	hsc_p	hsc_b	hsc_s	degree_p	degree_t	workex	etest_p	specialisation	mba_p	status	salary	
0	1	M	67.00	Others	91.00	Others	Commerce	58.00	Sci&Tech	No	55.00	Mkt&HR	58.80	Placed	270000.0
1	2	M	79.33	Central	78.33	Others	Science	77.48	Sci&Tech	Yes	86.50	Mkt&Fin	66.28	Placed	200000.0
2	3	M	65.00	Central	68.00	Central	Arts	64.00	Comm&Mgmt	No	75.00	Mkt&Fin	57.00	Placed	250000.0
3	4	M	56.00	Central	52.00	Central	Science	52.00	Sci&Tech	No	66.00	Mkt&HR	59.43	Not Placed	NaN
4	5	M	85.00	Central	73.60	Central	Commerce	73.30	Comm&Mgmt	No	96.00	Mkt&Fin	55.50	Placed	425000.0
5	6	M	55.00	Others	49.00	Others	Science	67.25	Sci&Tech	Yes	55.00	Mkt&Fin	51.58	Not Placed	NaN
6	7	F	46.00	Others	49.20	Others	Commerce	79.00	Comm&Mgmt	No	74.28	Mkt&Fin	53.29	Not Placed	NaN
7	8	M	82.00	Central	64.00	Central	Science	66.00	Sci&Tech	Yes	67.00	Mkt&Fin	62.14	Placed	252000.0

```
placement_copy=placement.copy()

placement_copy.shape

(215, 15)

placement_copy.dtypes
```

sl_no	int64
gender	object
ssc_p	float64
ssc_b	object
hsc_p	float64
hsc_b	object
hsc_s	object
degree_p	float64
degree_t	object
workex	object
etest_p	float64
specialisation	object
mba_p	float64
status	object
salary	float64

IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING

```
File Edit Selection View Go Run Terminal Help MainNotebook.ipynb - Visual Studio Code
C:\Users\VIJAY> Downloads > CampusPlacement-main > CampusPlacement-main > MainNotebook.ipynb > import numpy as np
Code | Markdown | Run All | Clear All Outputs | Outline | Select Kernel

placement.head(10)
[142] Python
...
  sl_no  gender  ssc_p  ssc_b  hsc_p  hsc_b  hsc_s  degree_p  degree_t  workex  etest_p  specialisation  mba_p  status  salary
0      1      M   67.00  Others  91.00  Others  Commerce  50.00  SoftTech  No    55.00  Mkt&HR  50.00  Placed  2700000.0
1      2      M  79.33  Central  78.33  Others  Science  77.48  SoftTech  Yes   86.50  Mkt&Fin  66.28  Placed  2000000.0
2      3      M  66.00  Central  68.00  Central  Arts  64.00  Comm&Mgmt  No    75.00  Mkt&Fin  57.00  Placed  2500000.0
3      4      M  56.00  Central  52.00  Central  Science  52.00  SoftTech  No    66.00  Mkt&HR  59.43  Not Placed  NaN
4      5      M  65.00  Central  75.00  Central  Commerce  75.30  Comm&Mgmt  No    96.00  Mkt&Fin  55.50  Placed  4750000.0
5      6      M  55.00  Others  49.00  Others  Science  67.25  SoftTech  Yes   55.00  Mkt&Fin  51.58  Not Placed  NaN
6      7      F  46.00  Others  49.00  Others  Commerce  79.00  Comm&Mgmt  No    74.78  Mkt&Fin  53.79  Not Placed  NaN
7      8      M  82.00  Central  64.00  Central  Science  66.00  SoftTech  Yes   67.00  Mkt&Fin  62.14  Placed  2520000.0
8      9      M  73.00  Central  79.00  Central  Commerce  72.00  Comm&Mgmt  No    91.34  Mkt&Fin  61.79  Placed  2310000.0
9     10      M  58.00  Central  70.00  Central  Commerce  61.00  Comm&Mgmt  No    54.00  Mkt&Fin  52.21  Not Placed  NaN

placement_copy=placement.copy()
[144] Python Python

placement_copy.shape
[145] Python Python
... (215, 15)
```

```
File Edit Selection View Go Run Terminal Help MainNotebook.ipynb - Visual Studio Code
C:\Users\VIJAY> Downloads > CampusPlacement-main > CampusPlacement-main > MainNotebook.ipynb > import numpy as np
Code | Markdown | Run All | Clear All Outputs | Outline | Select Kernel

NAN Handling

placement_copy.isnull().sum()
[177] Python
...
sl_no      0
gender     0
ssc_p      0
ssc_b      0
hsc_p      0
hsc_b      0
hsc_s      0
degree_p   0
degree_t   0
workex     0
etest_p    0
specialisation  0
mba_p      0
status     0
salary     0
dtype: int64

placement_copy['salary'].fillna(value=0 , inplace = True )
[178] Python
```

IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING

```
File Edit Selection View Go Run Terminal Help
MainNotebook.ipynb - Visual Studio Code

> Users > VJAY > Downloads > CampusPlacement-main > CampusPlacement-main > MainNotebook.ipynb > import numpy as np

placement_copy.isnull().sum()

[39]
Python

---
sl_no      0
gender     0
ssc_p      0
ssc_b      0
hsc_p      0
hsc_b      0
hsc_s      0
degree_p   0
degree_t   0
workex     0
etest_p    0
specialisation 0
mba_p      0
status     0
salary     0
dtype: int64

placement_copy.drop(['sl_no','ssc_b','hsc_b'], axis = 1, inplace = True)

[40]
Python

placement_copy.head()

[41]
Python

---
gender ssc_p hsc_p hsc_s degree_p degree_t workex etest_p specialisation mba_p status salary
0  M    67.00  91.00  Commerce  58.00  Sci&Tech  No    55.0  Mkt&HR  58.80  Placed  270000.0
1  M    79.33  78.33   Science  77.48  Sci&Tech  Yes   86.5  Mkt&Fin  66.28  Placed  200000.0
2  M    65.00  68.00    Arts    64.00  Comm&Mgmt No    75.0  Mkt&Fin  57.90  Placed  250000.0
3  M    56.00  52.00   Science  52.00  Sci&Tech  No    66.0  Mkt&HR  59.43  Not Placed  0.0
4  M    85.00  73.00  Commerce  73.00  Comm&Mgmt No    96.0  Mkt&HR  55.50  Placed  425000.0

Cell 1 of 31
```

```
File Edit Selection View Go Run Terminal Help
MainNotebook.ipynb - Visual Studio Code

> Users > VJAY > Downloads > CampusPlacement-main > CampusPlacement-main > MainNotebook.ipynb > import numpy as np

placement_copy.head()

[41]
Python

---
gender ssc_p hsc_p hsc_s degree_p degree_t workex etest_p specialisation mba_p status salary
0  M    67.00  91.00  Commerce  58.00  Sci&Tech  No    55.0  Mkt&HR  58.80  Placed  270000.0
1  M    79.33  78.33   Science  77.48  Sci&Tech  Yes   86.5  Mkt&Fin  66.28  Placed  200000.0
2  M    65.00  68.00    Arts    64.00  Comm&Mgmt No    75.0  Mkt&Fin  57.90  Placed  250000.0
3  M    56.00  52.00   Science  52.00  Sci&Tech  No    66.0  Mkt&HR  59.43  Not Placed  0.0
4  M    85.00  73.00  Commerce  73.00  Comm&Mgmt No    96.0  Mkt&HR  55.50  Placed  425000.0

Outlier Handling

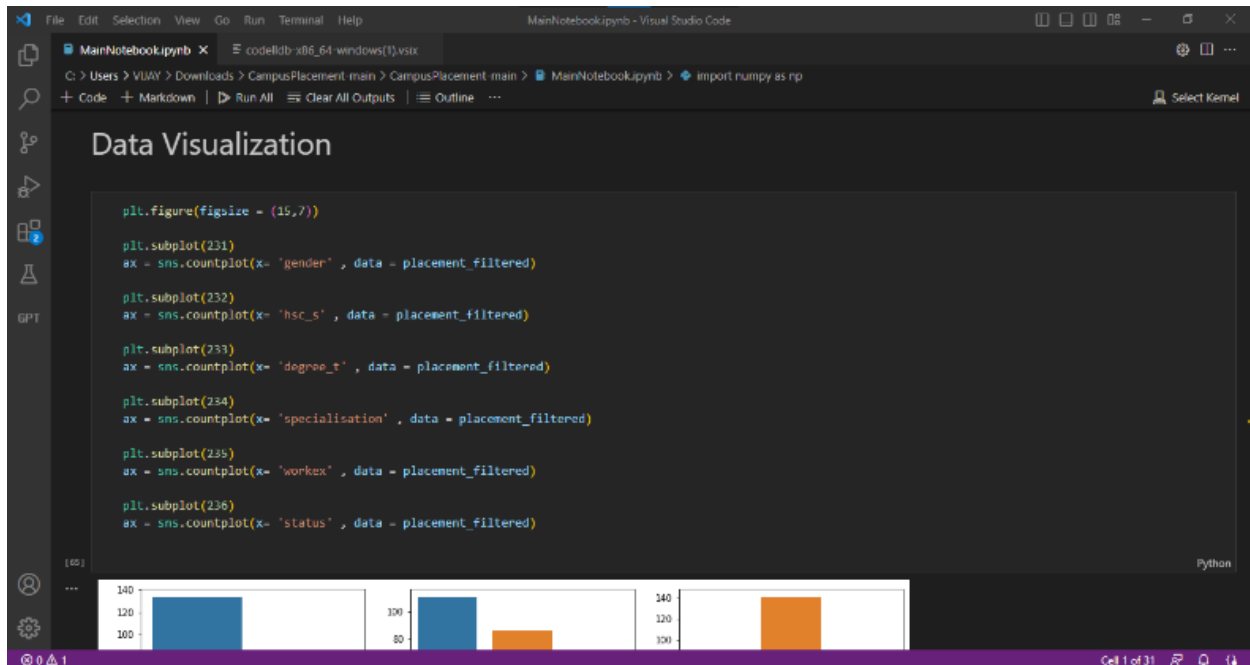
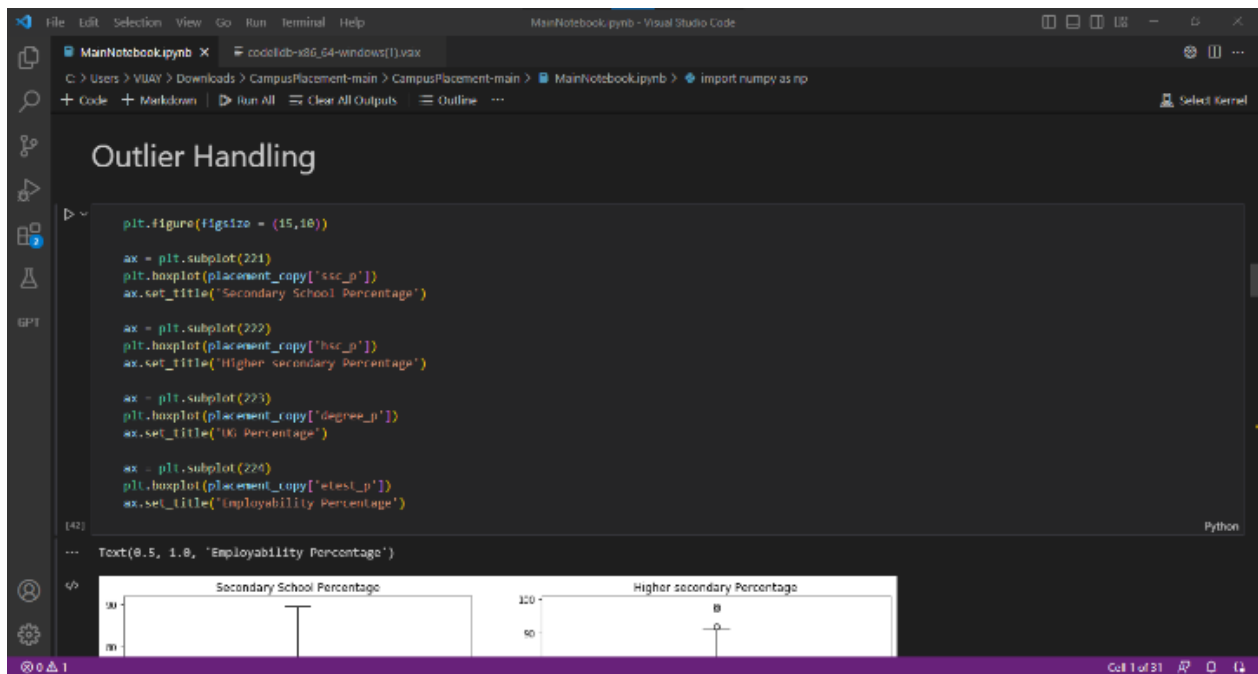
plt.figure(figsize = (15,10))

ax = plt.subplot(221)
plt.boxplot(placement_copy['ssc_p'])
ax.set_title('Secondary School Percentage')

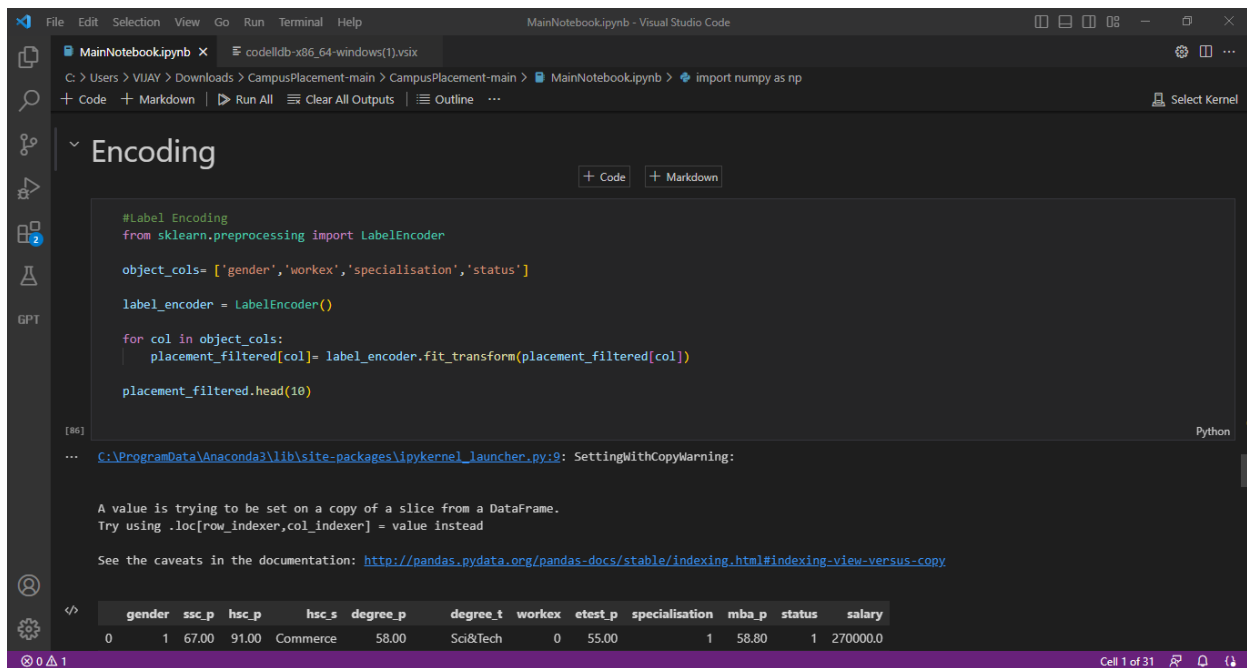
ax = plt.subplot(222)
plt.boxplot(placement_copy['hsc_p'])
ax.set_title('Higher secondary Percentage')

ax = plt.subplot(223)
plt.boxplot(placement_copy['degree_p'])
```

IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING



IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING



The screenshot shows a Jupyter Notebook cell titled "Encoding" in Visual Studio Code. The code imports `LabelEncoder` from `sklearn.preprocessing` and applies it to categorical columns: `gender`, `workex`, `specialisation`, and `status`. The output shows the first 10 rows of the filtered data with these columns encoded.

```
#Label Encoding
from sklearn.preprocessing import LabelEncoder

object_cols= ['gender','workex','specialisation','status']

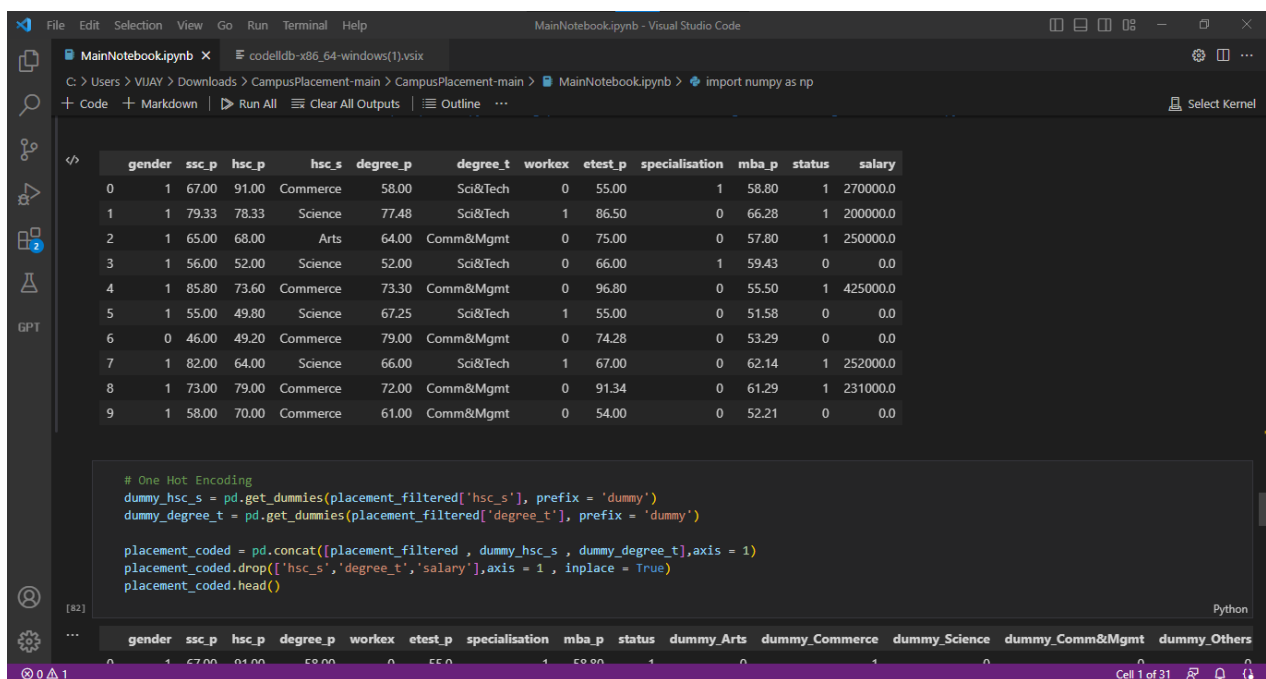
label_encoder = LabelEncoder()

for col in object_cols:
    placement_filtered[col]= label_encoder.fit_transform(placement_filtered[col])

placement_filtered.head(10)
```

Output:

	gender	ssc_p	hsc_p	hsc_s	degree_p	degree_t	workex	etest_p	specialisation	mba_p	status	salary
0	1	67.00	91.00	Commerce	58.00	Sci&Tech	0	55.00	1	58.80	1	270000.0



The screenshot shows a Jupyter Notebook cell titled "One Hot Encoding". The code uses `pd.get_dummies` to create dummy variables for `hsc_s` and `degree_t`, concatenates them with the original data, and drops the original categorical columns. The output shows the first 10 rows of the resulting dataset with dummy variables.

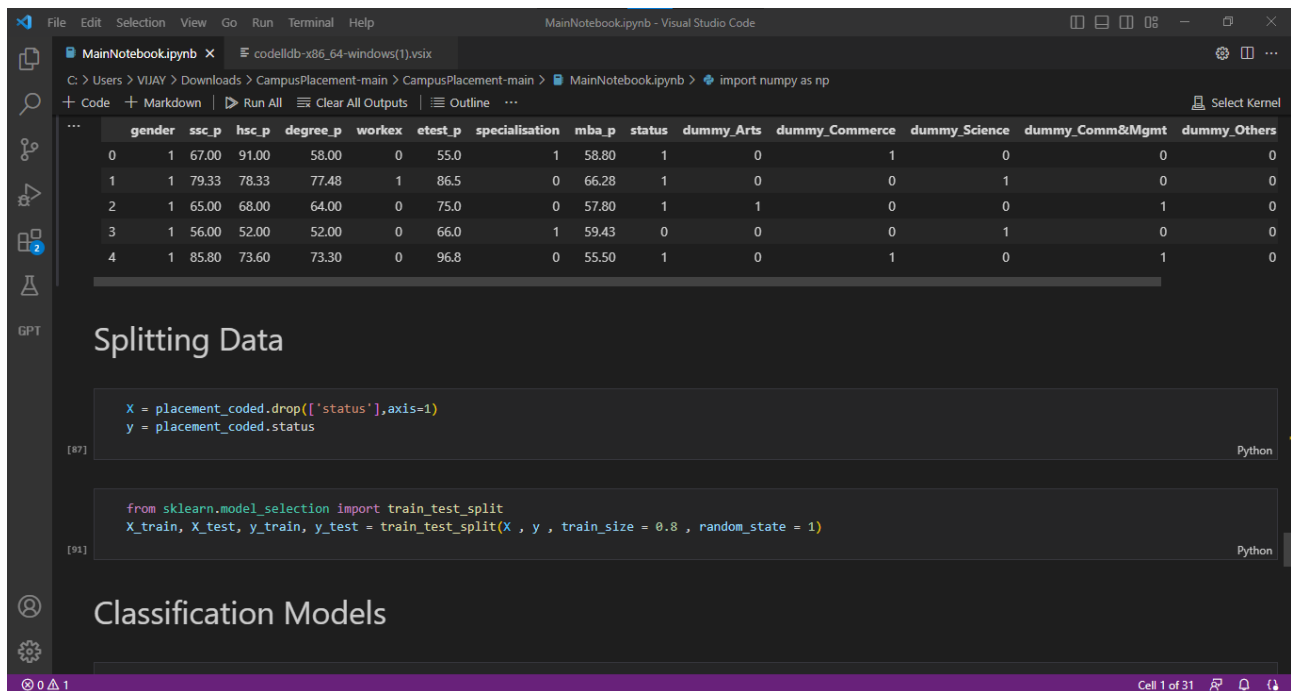
```
# One Hot Encoding
dummy_hsc_s = pd.get_dummies(placement_filtered['hsc_s'], prefix = 'dummy')
dummy_degree_t = pd.get_dummies(placement_filtered['degree_t'], prefix = 'dummy')

placement_coded = pd.concat([placement_filtered , dummy_hsc_s , dummy_degree_t],axis = 1)
placement_coded.drop(['hsc_s','degree_t','salary'],axis = 1 , inplace = True)
placement_coded.head()
```

Output:

	gender	ssc_p	hsc_p	degree_p	workex	etest_p	specialisation	mba_p	status	dummy_Arts	dummy_Commerce	dummy_Science	dummy_Comm&Mgmt	dummy_Others	salary
0	1	67.00	91.00	58.00	0	55.00	1	58.80	1	0	1	0	0	0	270000.0

IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING



The screenshot shows a Jupyter Notebook titled 'MainNotebook.ipynb' in Visual Studio Code. The notebook is open to a cell containing Python code for data splitting. The code imports 'train_test_split' from 'sklearn.model_selection' and splits the data into training and testing sets based on the 'status' variable. The notebook interface includes a sidebar with icons for Explorer, Search, Source Control, and GPT, and a top bar with menu options like File, Edit, Selection, View, Go, Run, Terminal, and Help. The status bar at the bottom indicates 'Cell 1 of 31'.

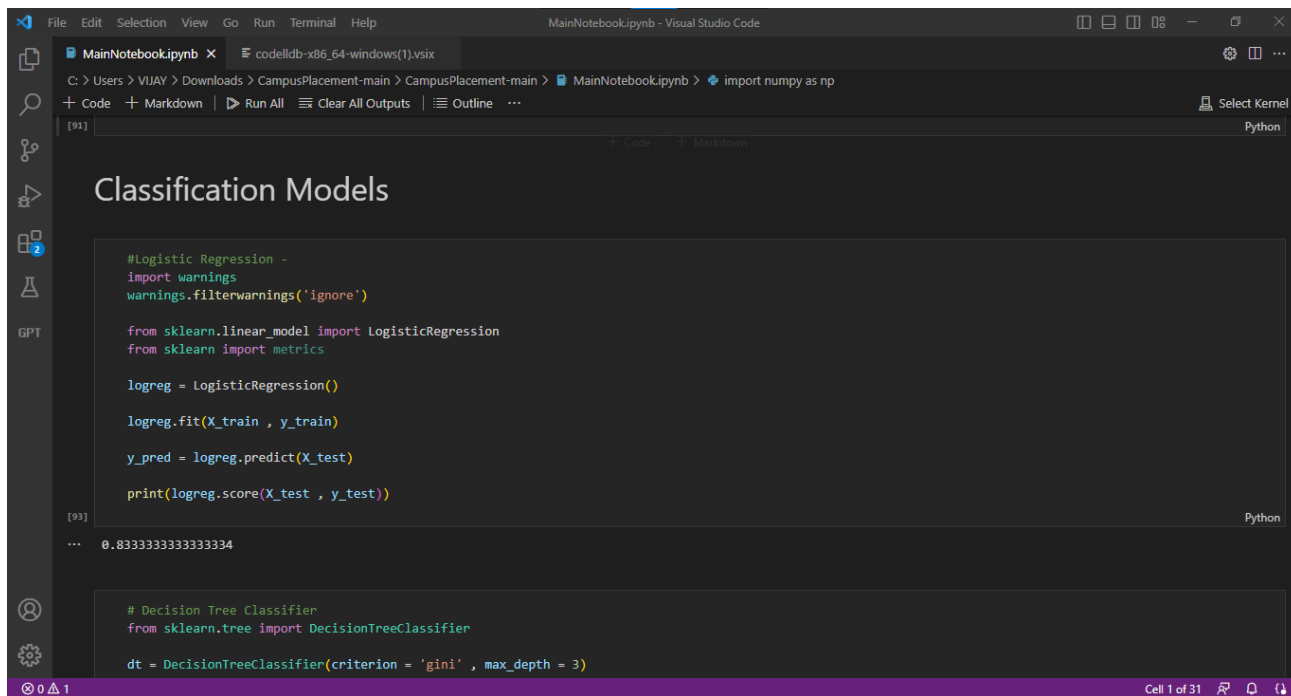
	gender	ssc_p	hsc_p	degree_p	workex	etest_p	specialisation	mba_p	status	dummy_Arts	dummy_Commerce	dummy_Science	dummy_Comm&Mgmt	dummy_Others
0	1	67.00	91.00	58.00	0	55.0	1	58.80	1	0	1	0	0	0
1	1	79.33	78.33	77.48	1	86.5	0	66.28	1	0	0	1	0	0
2	1	65.00	68.00	64.00	0	75.0	0	57.80	1	1	0	0	1	0
3	1	56.00	52.00	52.00	0	66.0	1	59.43	0	0	0	1	0	0
4	1	85.80	73.60	73.30	0	96.8	0	55.50	1	0	1	0	1	0

Splitting Data

```
X = placement_coded.drop(['status'],axis=1)
y = placement_coded.status
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X , y , train_size = 0.8 , random_state = 1)
```

Classification Models



The screenshot shows the same Jupyter Notebook in Visual Studio Code, now displaying code for training and evaluating a Logistic Regression model. The code imports 'LogisticRegression' from 'sklearn.linear_model' and 'metrics' from 'sklearn'. It then creates a 'LogisticRegression' object, fits it to the training data, and uses it to predict on the test data. The notebook interface is consistent with the previous screenshot, showing the same sidebar and top bar. The status bar at the bottom indicates 'Cell 1 of 31'.

Classification Models

```
#Logistic Regression -
import warnings
warnings.filterwarnings('ignore')

from sklearn.linear_model import LogisticRegression
from sklearn import metrics

logreg = LogisticRegression()

logreg.fit(X_train , y_train)

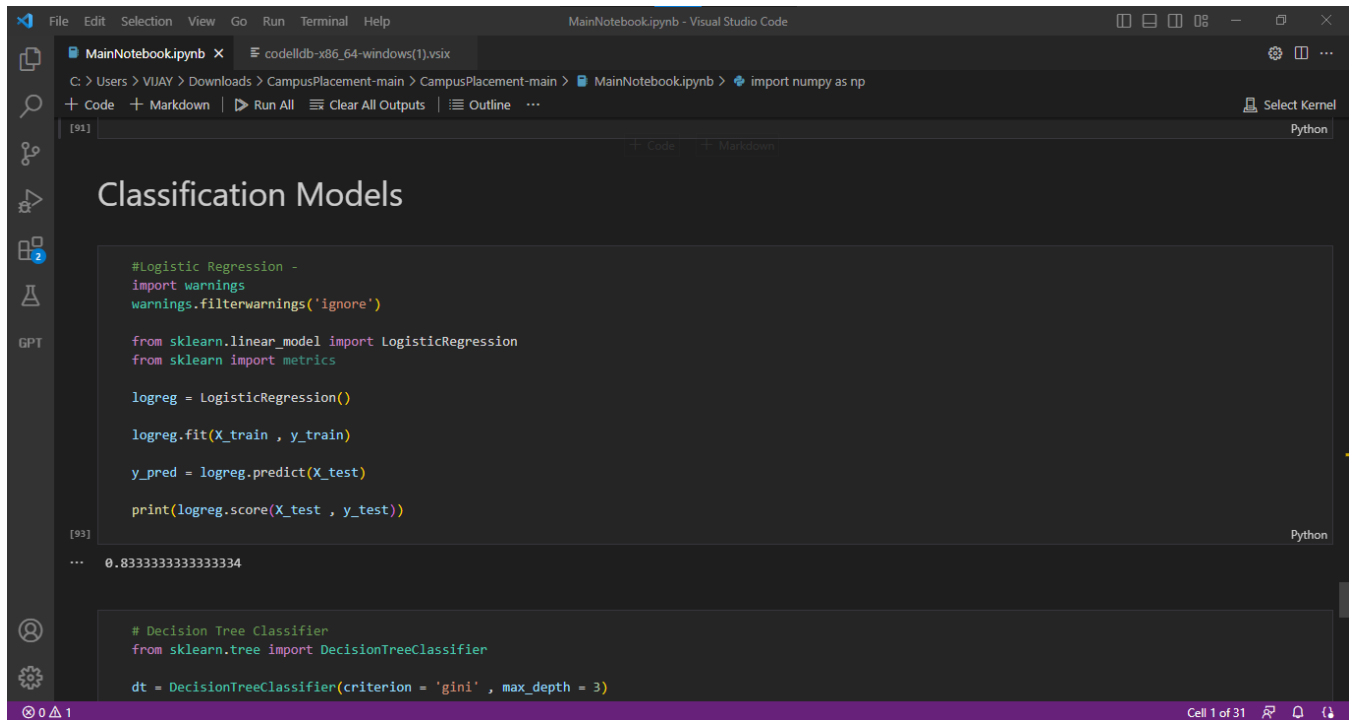
y_pred = logreg.predict(X_test)

print(logreg.score(X_test , y_test))
```

```
# Decision Tree Classifier
from sklearn.tree import DecisionTreeClassifier

dt = DecisionTreeClassifier(criterion = 'gini' , max_depth = 3)
```

IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING



The screenshot shows a Jupyter Notebook titled 'MainNotebook.ipynb' in Visual Studio Code. The notebook is open to a cell containing Python code for Logistic Regression. The code imports necessary libraries, creates a LogisticRegression model, fits it to training data, and predicts on test data. The output of the cell shows the score of the model on the test data.

```
#Logistic Regression -
import warnings
warnings.filterwarnings('ignore')

from sklearn.linear_model import LogisticRegression
from sklearn import metrics

logreg = LogisticRegression()

logreg.fit(X_train , y_train)

y_pred = logreg.predict(X_test)

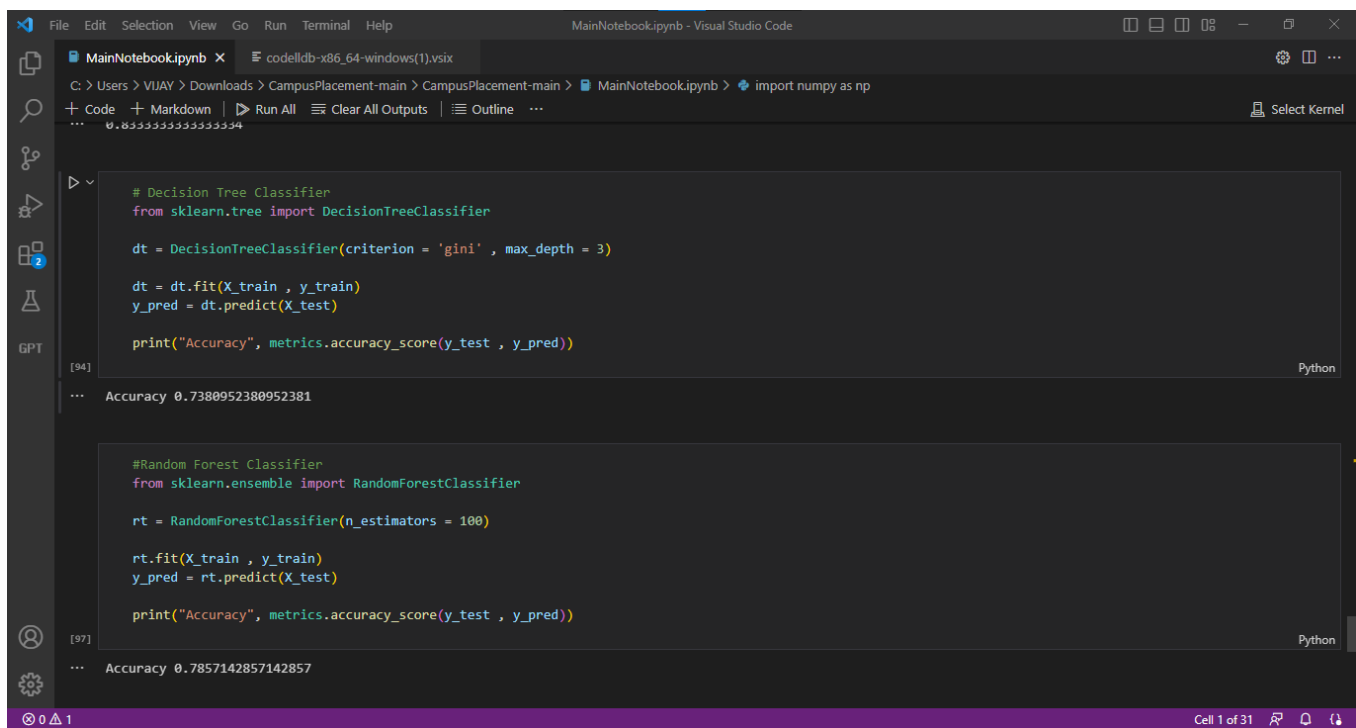
print(logreg.score(X_test , y_test))
```

Output: 0.8333333333333334

The notebook also shows the start of a Decision Tree Classifier code block.

```
# Decision Tree Classifier
from sklearn.tree import DecisionTreeClassifier

dt = DecisionTreeClassifier(criterion = 'gini' , max_depth = 3)
```



The screenshot shows the continuation of the Jupyter Notebook. It displays the code for a Decision Tree Classifier, which is fitted to the training data and used to predict on the test data. The output shows the accuracy of the model. Below this, the code for a Random Forest Classifier is shown, which is also fitted to the training data and used to predict on the test data. The output shows the accuracy of this model as well.

```
# Decision Tree Classifier
from sklearn.tree import DecisionTreeClassifier

dt = DecisionTreeClassifier(criterion = 'gini' , max_depth = 3)

dt = dt.fit(X_train , y_train)
y_pred = dt.predict(X_test)

print("Accuracy", metrics.accuracy_score(y_test , y_pred))
```

Output: Accuracy 0.7380952380952381

```
#Random Forest Classifier
from sklearn.ensemble import RandomForestClassifier

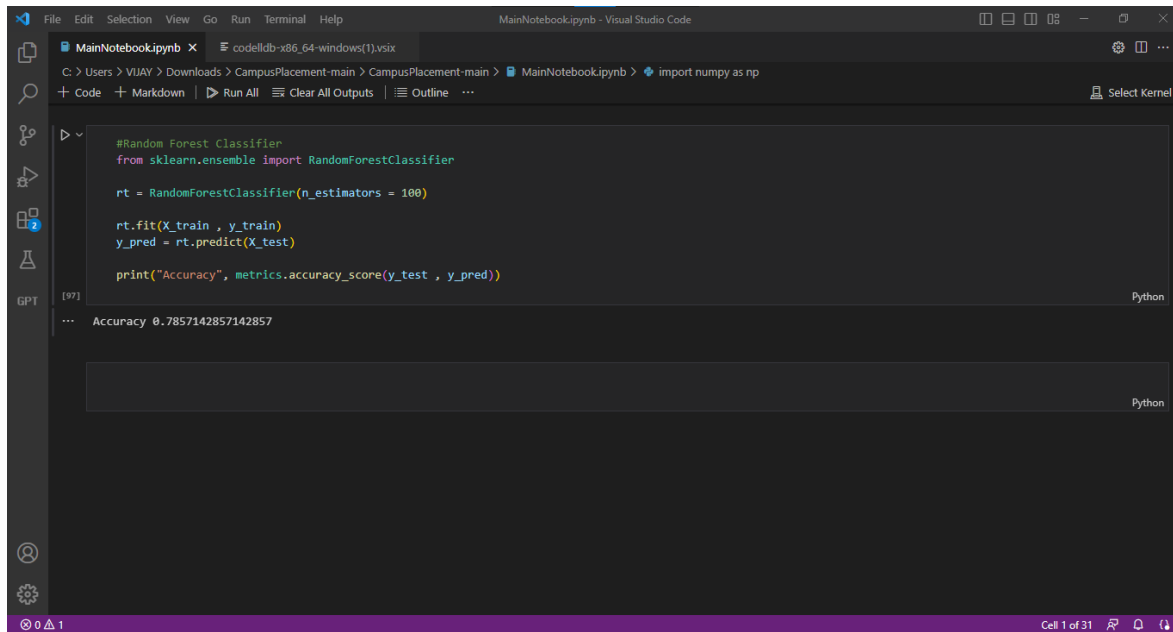
rt = RandomForestClassifier(n_estimators = 100)

rt.fit(X_train , y_train)
y_pred = rt.predict(X_test)

print("Accuracy", metrics.accuracy_score(y_test , y_pred))
```

Output: Accuracy 0.7857142857142857

IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING



The screenshot shows a Jupyter Notebook titled 'MainNotebook.ipynb' in Visual Studio Code. The notebook contains a Python cell with the following code:

```
#Random Forest Classifier
from sklearn.ensemble import RandomForestClassifier

rt = RandomForestClassifier(n_estimators = 100)

rt.fit(X_train , y_train)
y_pred = rt.predict(X_test)

print("Accuracy", metrics.accuracy_score(y_test , y_pred))
```

The output of the cell is: Accuracy 0.7857142857142857. The status bar at the bottom indicates 'Cell 1 of 31'.

6.ADVANTAGES &DISADVANTAGES

ADVANTAGES:

1. A student who gets recruited in Campus would have saved time, Even before completing their degree they getting recruited that is lot's of Time saved.
2. A Students finishes Degree in a happy mood as they have bagged a Job. They do not have to think about the job search strategies
3. A student gets Top Branded Organizations and attracts good offers, Top MNC's recruit students from collage and offer Good Salary Package.
4. A student who gets recruited during campus selection enjoy a better social status with the collages, class mates & family.

DISADVANTAGES:

1. Limited opportunities, assume if your collage invited 10 companies for campus selection then you are exposed to those 10 opportunities and if you are selected in campus then you had limited opportunities in front of you. but If you are not selected then you will search job in the open market then you would be having all companies in the domain as your opportunity.

IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING

2. If you are selected in campus then there are least chances for you to choose your special area of your interest. Assume if you are an Engineer and have dreams of making Career as IoT Developer then there are least chances you getting placed in special area of your interest in Campus.

3. Most of the Campus recruitments are conducted by big organizations like MNC's, but small sector and startups companies offer an excellent growth and learning curve, if you want to get expertise in a specific area then startup helps building your career better than an MNC. MNC's may provide a better salary and facilities but small companies and startup provide better projects & work experience and better foundation for your career. You may not get social status but you will be satisfied with your project & Work Experiences.

7.APPLICATIONS

- 1.network with companies
- 2.network with alumni
- 3.Personal development Training
- 4.industrial visits
- 5.online assessments
- 6.cooperate guest lectures
- 7.skill assessment tests 8.linkedIn

8.CONCLUSION:

In this project, we utilize two powerful machine learning models - Random Forest and Decision Tree - to predict placements of engineering students. We apply various techniques such as outlier detection and removal, correlation analysis, and categorical variable encoding to preprocess the data and improve model performance. Additionally, we conduct in-depth data analysis and visualization to gain deeper insights into the data. Feel free to experiment with other models or tune hyperparameters to further enhance the model's accuracy. Let's dive in and keep exploring

9.FUTURE SCOPE:

- 1.social and mobile recruitment
- 2.talent optimization via AI
- 3.competitive Hiring
- 4.online applications
- 5.mobile recruiting

IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING

6.video interview

7.Digital Structured Interviews

10.APPENDIX:

The screenshot shows a Jupyter Notebook with the following code and output:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import sklearn
```

```
placement = pd.read_csv("Placement_Data_Full_Class.csv")
```

```
placement.head(10)
```

	sl_no	gender	ssc_p	ssc_b	hsc_p	hsc_b	hsc_s	degree_p	degree_t	workex	etest_p	specialisation	mba_p	status	salary
0	1	M	67.00	Others	91.00	Others	Commerce	58.00	Sci&Tech	No	55.00	Mkt&HR	58.80	Placed	270000.0
1	2	M	79.33	Central	78.33	Others	Science	77.48	Sci&Tech	Yes	86.50	Mkt&Fin	66.28	Placed	200000.0
2	3	M	65.00	Central	68.00	Central	Arts	64.00	Comm&Mgmt	No	75.00	Mkt&Fin	57.80	Placed	250000.0
3	4	M	56.00	Central	52.00	Central	Science	52.00	Sci&Tech	No	66.00	Mkt&HR	59.43	Not Placed	NaN
4	5	M	85.80	Central	73.60	Central	Commerce	73.30	Comm&Mgmt	No	96.80	Mkt&Fin	55.50	Placed	425000.0
5	6	M	55.00	Others	49.80	Others	Science	67.25	Sci&Tech	Yes	55.00	Mkt&Fin	51.58	Not Placed	NaN
6	7	F	46.00	Others	49.20	Others	Commerce	79.00	Comm&Mgmt	No	74.28	Mkt&Fin	53.29	Not Placed	NaN
7	8	M	82.00	Central	64.00	Central	Science	66.00	Sci&Tech	Yes	67.00	Mkt&Fin	62.14	Placed	252000.0

The screenshot shows a Jupyter Notebook with the following code and output:

```
placement_copy=placement.copy()
```

```
placement_copy.shape
```

```
(215, 15)
```

```
placement_copy.dtypes
```

sl_no	int64
gender	object
ssc_p	float64
ssc_b	object
hsc_p	float64
hsc_b	object
hsc_s	object
degree_p	float64
degree_t	object
workex	object
etest_p	float64
specialisation	object
mba_p	float64
status	object
salary	float64

IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING

```
File Edit Selection View Go Run Terminal Help MainNotebook.ipynb - Visual Studio Code
C:\Users> VIJAY > Downloads > CampusPlacement-main > CampusPlacement-main > MainNotebook.ipynb > import numpy as np
+ Code + Markdown | Run All | Clear All Outputs | Outline ...
Select Kernel

placement.head(10)
[33] Python
...
  sl_no  gender  ssc_p  ssc_b  hsc_p  hsc_b  hsc_s  degree_p  degree_t  workex  etest_p  specialisation  mba_p  status  salary
0      1      M  67.00  Others  91.00  Others  Commerce  58.00  Sci&Tech  No    55.00  Mkt&HR    58.80  Placed  270000.0
1      2      M  79.33  Central  78.33  Others  Science    77.48  Sci&Tech  Yes   86.50  Mkt&Fin  66.28  Placed  200000.0
2      3      M  65.00  Central  68.00  Central  Arts      64.00  Comm&Mgmt  No    75.00  Mkt&Fin  57.80  Placed  250000.0
3      4      M  56.00  Central  52.00  Central  Science   52.00  Sci&Tech  No    66.00  Mkt&HR    59.43  Not Placed  NaN
4      5      M  85.80  Central  73.60  Central  Commerce  73.30  Comm&Mgmt  No    96.80  Mkt&Fin  55.50  Placed  425000.0
5      6      M  55.00  Others  49.80  Others  Science   67.25  Sci&Tech  Yes   55.00  Mkt&Fin  51.58  Not Placed  NaN
6      7      F  46.00  Others  49.20  Others  Commerce  79.00  Comm&Mgmt  No    74.28  Mkt&Fin  53.29  Not Placed  NaN
7      8      M  82.00  Central  64.00  Central  Science   66.00  Sci&Tech  Yes   67.00  Mkt&Fin  62.14  Placed  252000.0
8      9      M  73.00  Central  79.00  Central  Commerce  72.00  Comm&Mgmt  No    91.34  Mkt&Fin  61.29  Placed  231000.0
9     10      M  58.00  Central  70.00  Central  Commerce  61.00  Comm&Mgmt  No    54.00  Mkt&Fin  52.21  Not Placed  NaN

placement_copy=placement.copy()
[34] Python Python

placement_copy.shape
[35] Python Python
... (215, 15)
```

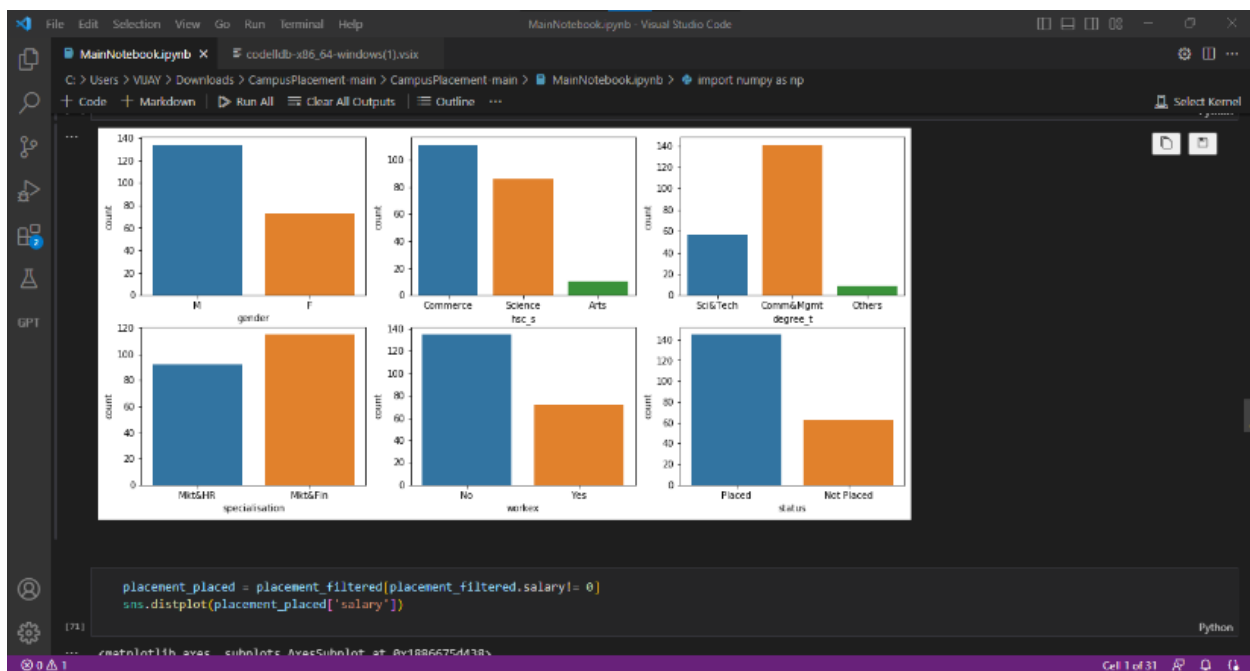
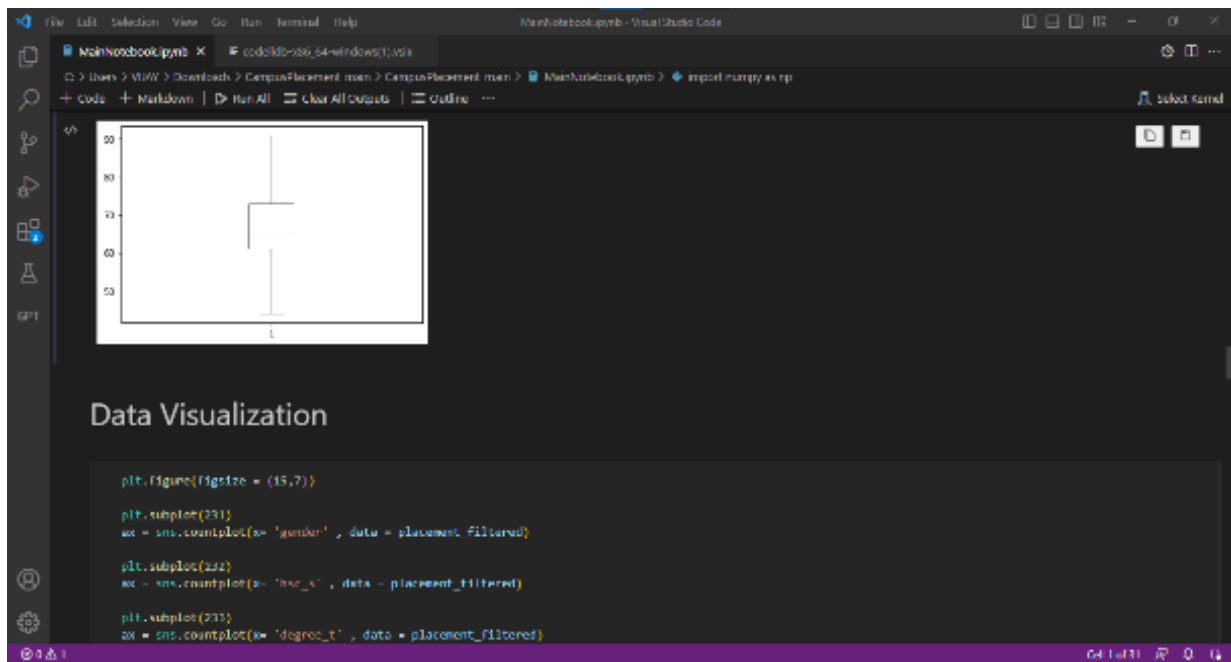
```
File Edit Selection View Go Run Terminal Help MainNotebook.ipynb - Visual Studio Code
C:\Users> VIJAY > Downloads > CampusPlacement-main > CampusPlacement-main > MainNotebook.ipynb > import numpy as np
+ Code + Markdown | Run All | Clear All Outputs | Outline ...
Select Kernel

placement_copy.dtypes
[36] Python Python
...
sl_no      int64
gender      object
ssc_p      float64
ssc_b      object
hsc_p      float64
hsc_b      object
hsc_s      object
degree_p   float64
degree_t   object
workex     object
etest_p    float64
specialisation  object
mba_p      float64
status     object
salary     float64
dtype: object

NAN Handling

placement_copy.isnull().sum()
[37] Python
...
sl_no      0
gender     0
ssc_p      0
```

IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING



IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING

