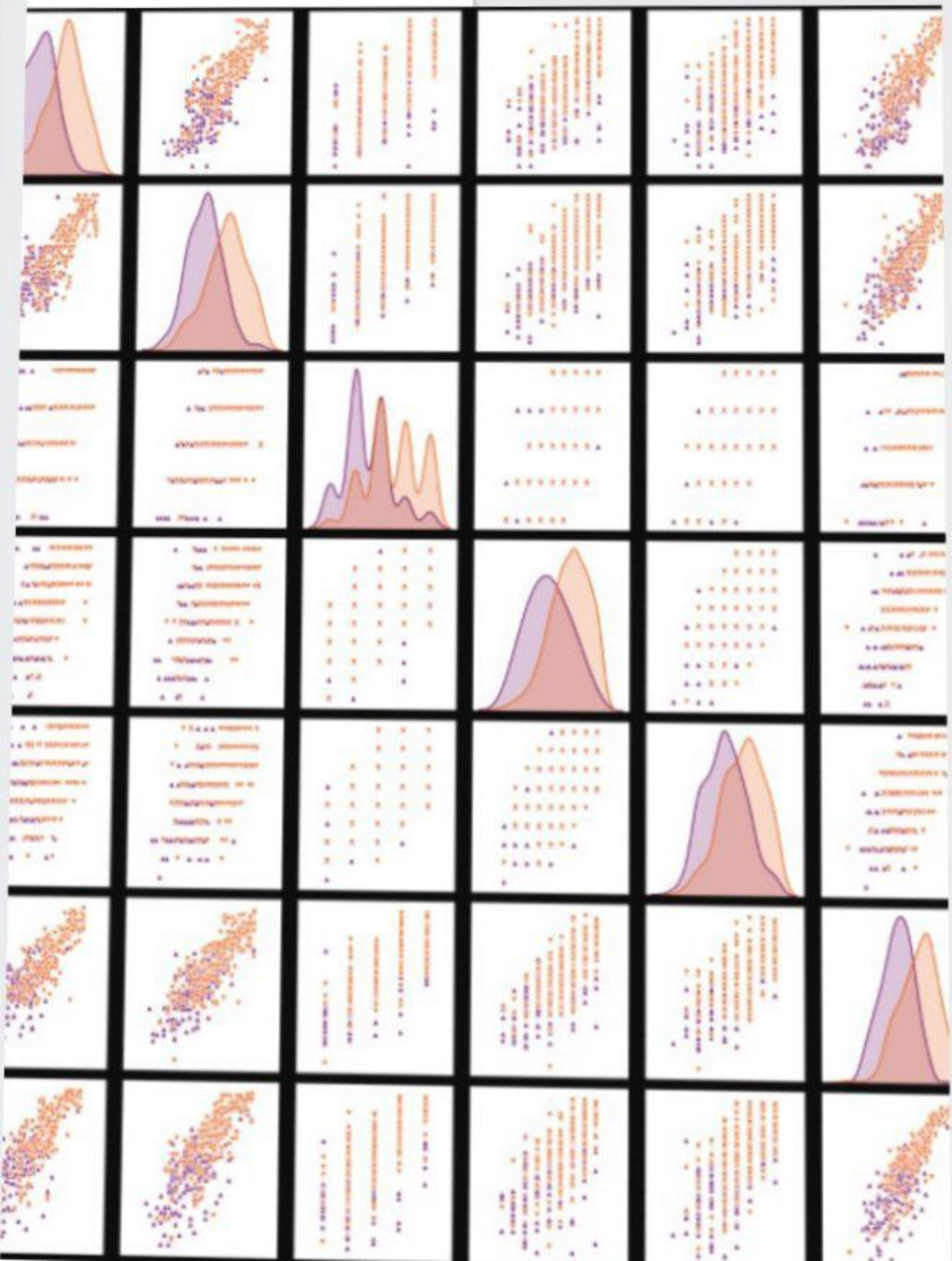
```
data.describe()
```

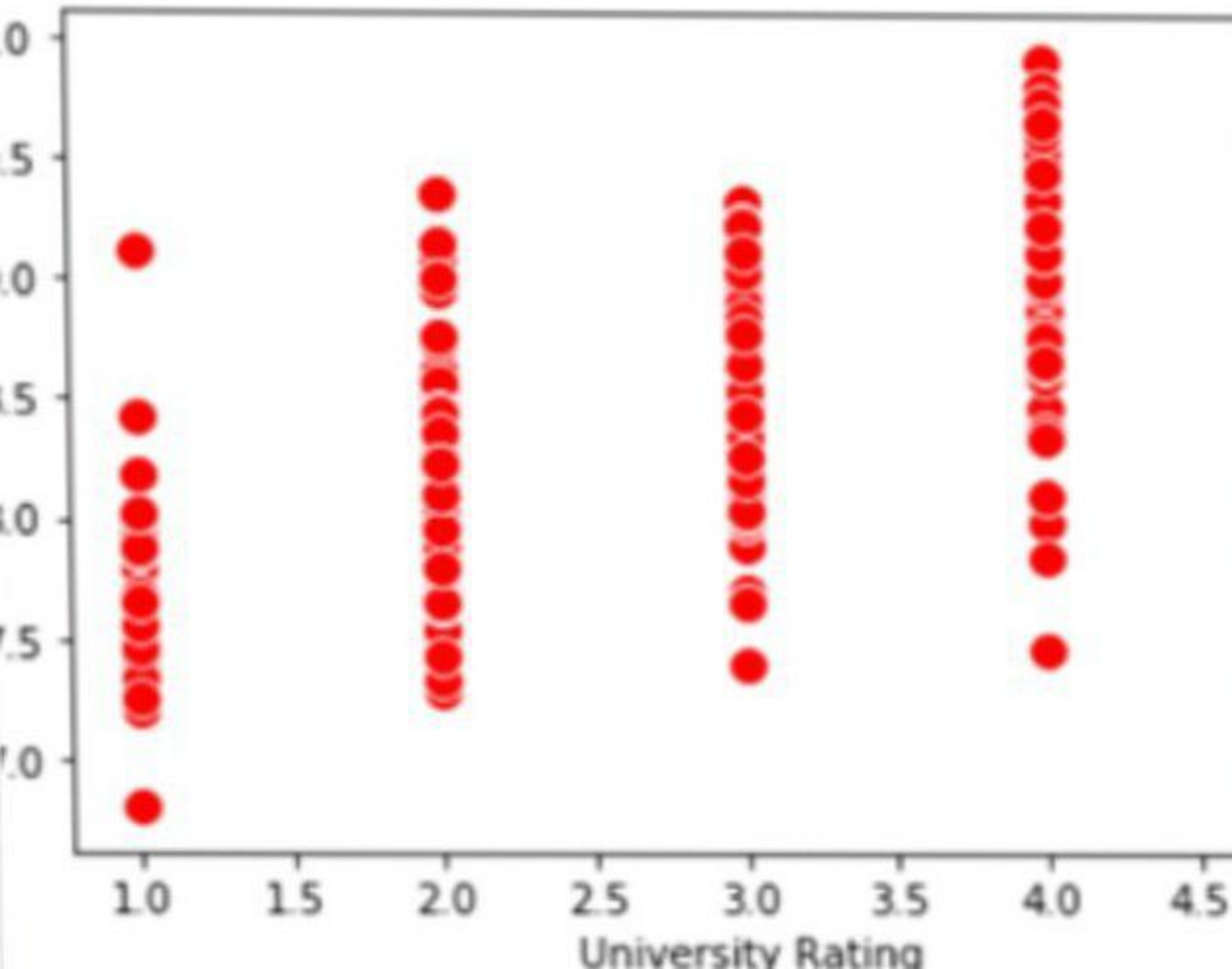
	GRE Score	TOEFL Score	University Rating	SOP	LO
count	400.000000	400.000000	400.000000	400.000000	400.0000
mean	316.807500	107.410000	3.087500	3.400000	3.452500
std	11.473646	6.069514	1.143728	1.006869	0.898478
min	290.000000	92.000000	1.000000	1.000000	1.000000
25%	308.000000	103.000000	2.000000	2.500000	3.000000
50%	317.000000	107.000000	3.000000	3.500000	3.500000
75%	325.000000	112.000000	4.000000	4.000000	4.000000
max	340.000000	120.000000	5.000000	5.000000	5.000000

```
s.pairplot(data=data,hue='Research',markers=["^"
```



```
catterplot(x='University Rating',y='CGPA',data=d
```

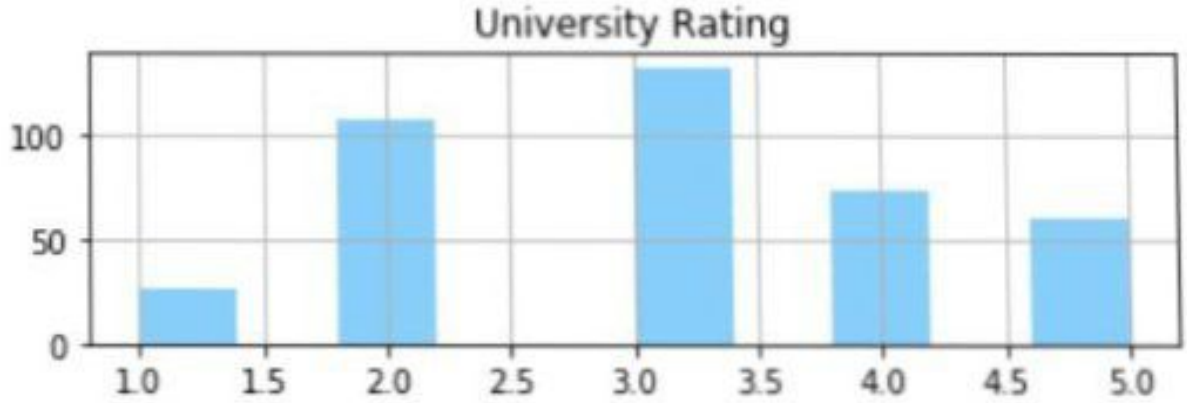
```
plotlib.axes._subplots.AxesSubplot at 0x
```

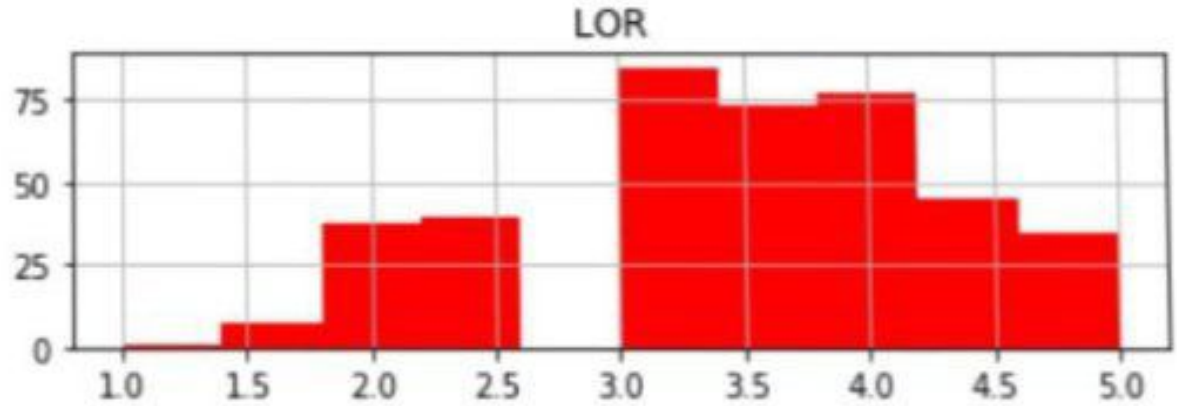


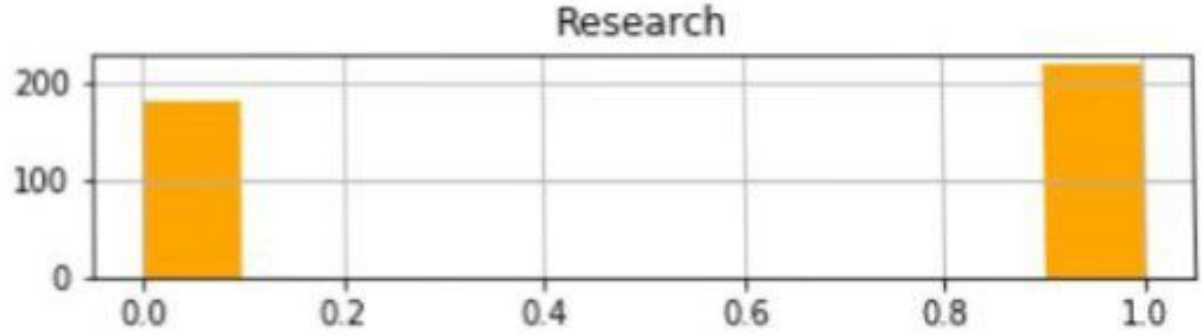
```
category = ['GRE Score','TOEFL Score','University Rating','SOP
color = ['yellowgreen','gold','lightskyblue','pink','red','pur
start = True
for i in np.arange(4):
    fig = plt.figure(figsize=(14,8))
    plt.subplot2grid((4,2),(i,0))
    data[category[2*i]].hist(color=color[2*i],bins=10)
    plt.title(category[2*i])
    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()
```

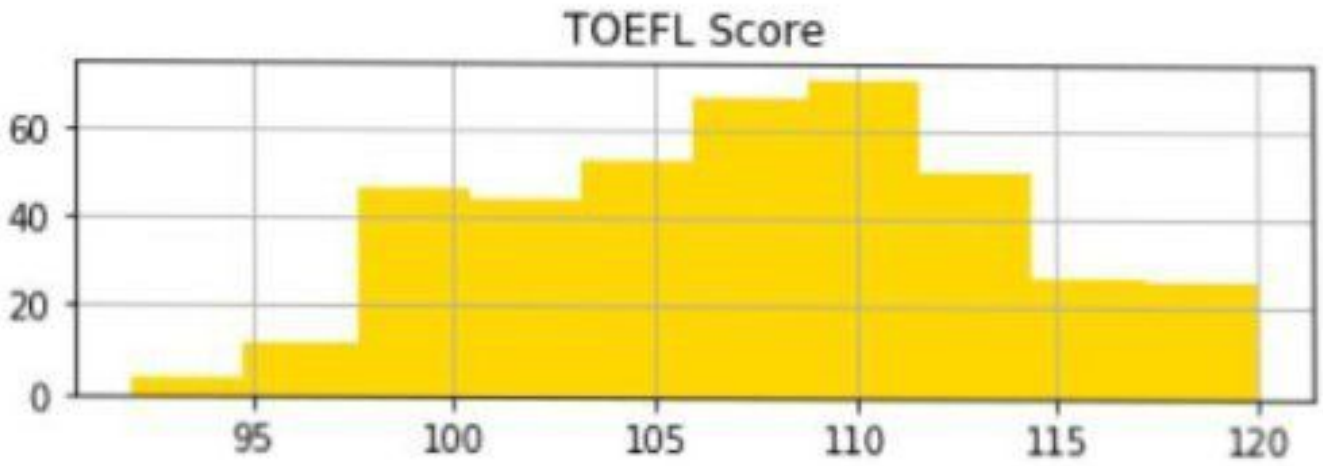
```
plt.show()
```

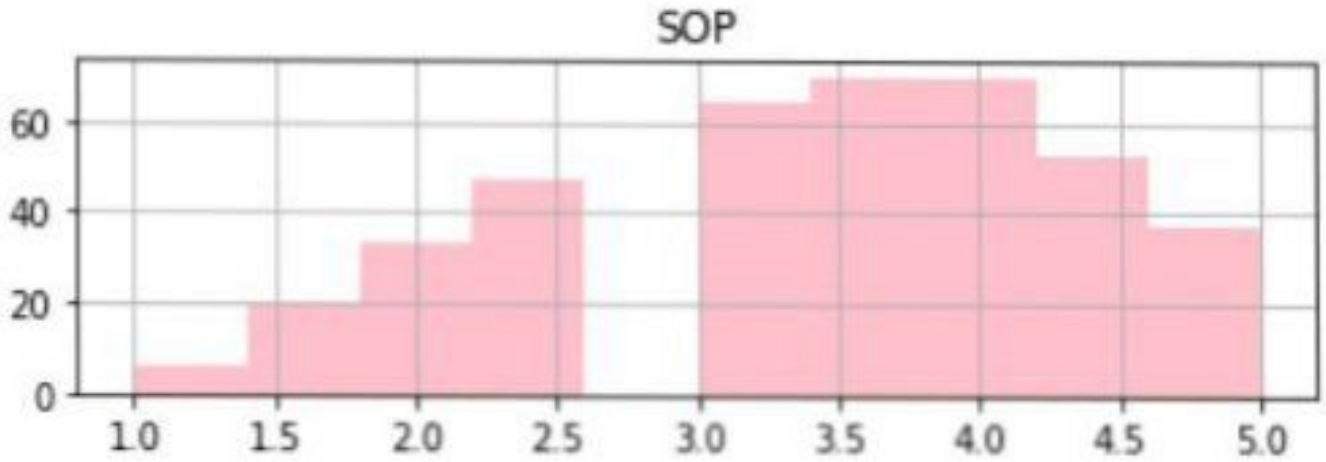

GRE Score
60
40
20
0
290
300
310
320
330
340

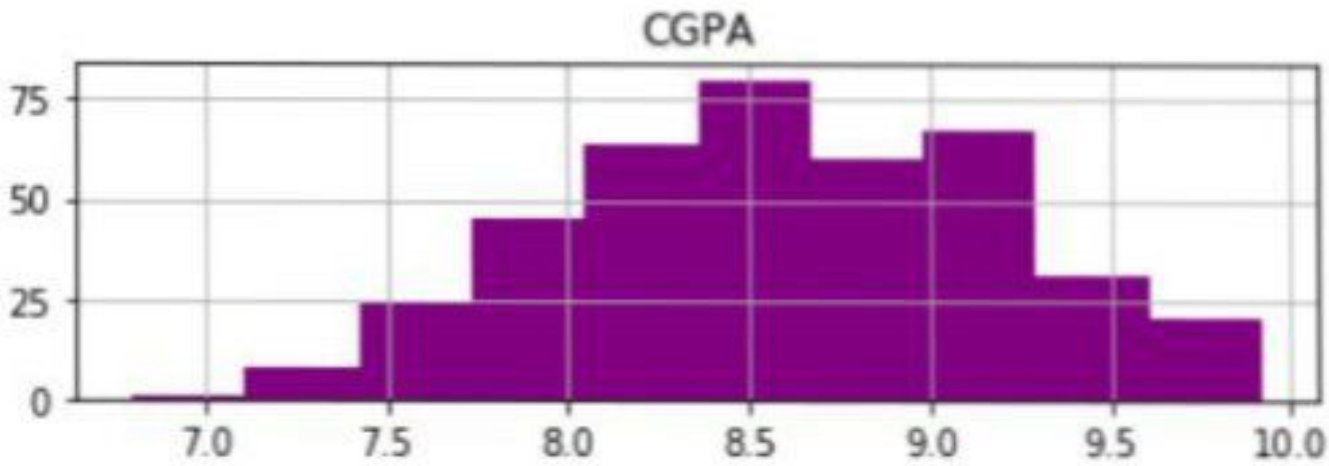

University Rating
100
50
0
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0

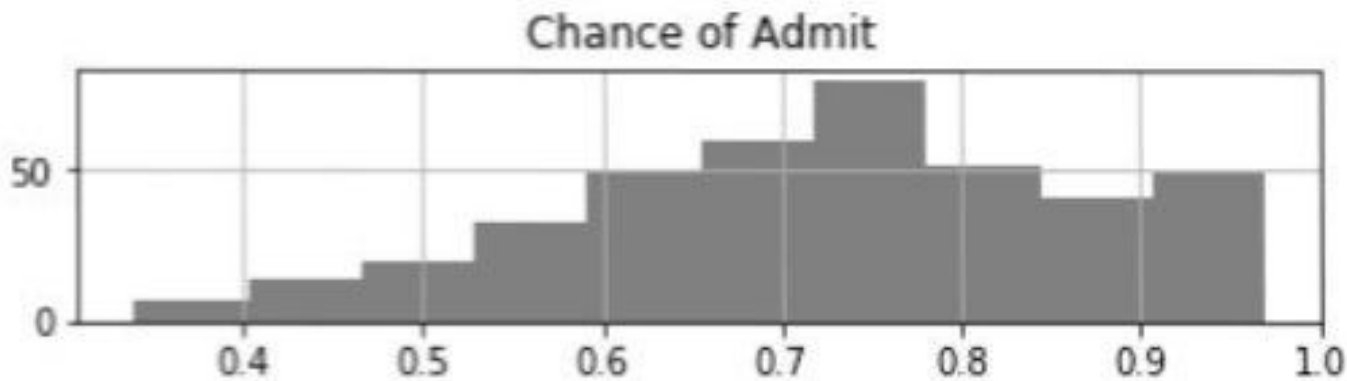

LOR
75
50
25
0
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0


Research
200
100
0
0.0
0.2
0.4
0.6
0.8
1.0


TOEFL Score
60
40
20
0
95
100
105
110
115
120


SOP
60
40
20
0
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0


CGPA
75
50
25
0
7.0
7.5
8.0
8.5
9.0
9.5
10.0


Chance of Admit
50
0
0.4
0.5
0.6
0.7
0.8
0.9
1.0

```
from sklearn.preprocessing import MinMaxSca
sc = MinMaxScaler()
x=sc.fit_transform(x)
x
```

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

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

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x,
#random_state acts as the seed for the random number g
```

```
y_train=(y_train>0.5)
y_train
```

```
y_test=(y_test>0.5)
```