Data Analysis Project - GRE Score Admission Prediction

September 30, 2020

[140]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

0.0.1 Part 1: Load the data into a DataFrame

[141]:
```
adm = pd.DataFrame()
adm = pd.read_csv('Admission_Predict.csv')
```

[142]:
```
#Getting the top few rows to explore the dataset
adm.head()
```

[142]:
```
   Serial No.  GRE Score  TOEFL Score  University Rating  SOP  LOR   CGPA  \
0           1        337          118                  4  4.5   4.5  9.65
1           2        324          107                  4  4.0   4.5  8.87
2           3        316          104                  3  3.0   3.5  8.00
3           4        322          110                  3  3.5   2.5  8.67
4           5        314          103                  2  2.0   3.0  8.21

   Research  Chance of Admit
0         1             0.92
1         1             0.76
2         1             0.72
3         1             0.80
4         0             0.65
```

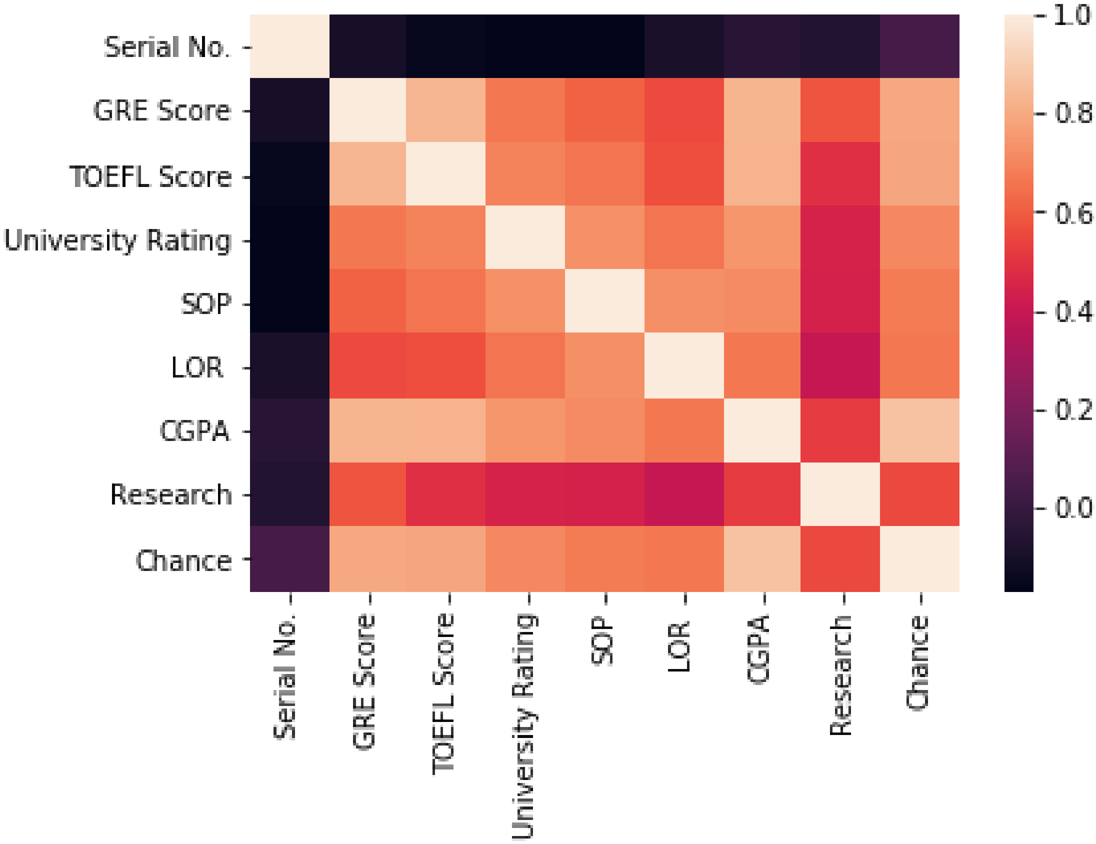
[143]:
```
#Statistical analysis of the dataset
adm.describe()
```

[143]:
```
       Serial No.   GRE Score  TOEFL Score  University Rating         SOP  \
count  400.000000  400.000000   400.000000         400.000000  400.000000
mean   200.500000  316.807500   107.410000           3.087500    3.400000
std    115.614301   11.473646     6.069514           1.143728    1.006869
min      1.000000  290.000000    92.000000           1.000000    1.000000
25%    100.750000  308.000000   103.000000           2.000000    2.500000
50%    200.500000  317.000000   107.000000           3.000000    3.500000
75%    300.250000  325.000000   112.000000           4.000000    4.000000
```

```
max    400.000000  340.000000   120.000000           5.000000    5.000000

             LOR        CGPA    Research  Chance of Admit
count  400.000000  400.000000  400.000000        400.000000
mean     3.452500    8.598925    0.547500          0.724350
std      0.898478    0.596317    0.498362          0.142609
min      1.000000    6.800000    0.000000          0.340000
25%      3.000000    8.170000    0.000000          0.640000
50%      3.500000    8.610000    1.000000          0.730000
75%      4.000000    9.062500    1.000000          0.830000
max      5.000000    9.920000    1.000000          0.970000
```

[144]:
```
#Heatmap of data
heat = adm.corr()
sns.heatmap(heat,
            xticklabels=corr.columns.values,
            yticklabels=corr.columns.values)
```

[144]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2f860624278>
```



0.0.2 Part 2: Calculate the mean GRE score of students with a chance of admission above 85%

[145]:
```
#Changing name of column
adm = adm.rename({"Chance of Admit ":"Chance"}, axis='columns')
adm.head()
```

[145]:
```
   Serial No.  GRE Score  TOEFL Score  University Rating  SOP  LOR   CGPA  \
0           1        337          118                  4  4.5   4.5  9.65
1           2        324          107                  4  4.0   4.5  8.87
2           3        316          104                  3  3.0   3.5  8.00
3           4        322          110                  3  3.5   2.5  8.67
4           5        314          103                  2  2.0   3.0  8.21

   Research  Chance
0         1    0.92
1         1    0.76
2         1    0.72
3         1    0.80
4         0    0.65
```

[146]:
```
adm_filter = adm[adm['Chance']>0.85] #Filtering out students with chance less␣
 ↪than 85% or 0.85
```

[147]:
```
# Resetting index
adm_filter.reset_index(drop=True, inplace = True)
```

[148]:
```
adm_filter.head()
```

[148]:
```
   Serial No.  GRE Score  TOEFL Score  University Rating  SOP  LOR   CGPA  \
0           1        337          118                  4  4.5   4.5  9.65
1           6        330          115                  5  4.5   3.0  9.34
2          23        328          116                  5  5.0   5.0  9.50
3          24        334          119                  5  5.0   4.5  9.70
4          25        336          119                  5  4.0   3.5  9.80

   Research  Chance
0         1    0.92
1         1    0.90
2         1    0.94
3         1    0.95
4         1    0.97
```

[149]:
```
#Calculating mean of GRE Score of students with chance of admission above 85%
print("Mean of GRE scores of students with chance of admissions above 85%:",␣
 ↪adm_filter["GRE Score"].mean())
```

```
Mean of GRE scores of students with chance of admissions above 85%:
331.144578313253
```

0.0.3 Part 3: Calculate the mean chance of admissions of students for which the research column is 1

[150]:
```
adm_filter1 = adm[adm['Research'] == 1]
adm_filter1.reset_index(drop=True, inplace = True)
adm_filter1.head()
```

[150]:
```
   Serial No.  GRE Score  TOEFL Score  University Rating  SOP  LOR   CGPA  \
0           1        337          118                  4  4.5   4.5  9.65
1           2        324          107                  4  4.0   4.5  8.87
2           3        316          104                  3  3.0   3.5  8.00
3           4        322          110                  3  3.5   2.5  8.67
4           6        330          115                  5  4.5   3.0  9.34

   Research  Chance
0         1    0.92
1         1    0.76
2         1    0.72
3         1    0.80
4         1    0.90
```

[151]:
```
adm_filter1.mean()
print("Mean chance of admissions for students with 1 research:",␣
 ↪adm_filter1["Chance"].mean())
```

```
Mean chance of admissions for students with 1 research: 0.7959817351598172
```

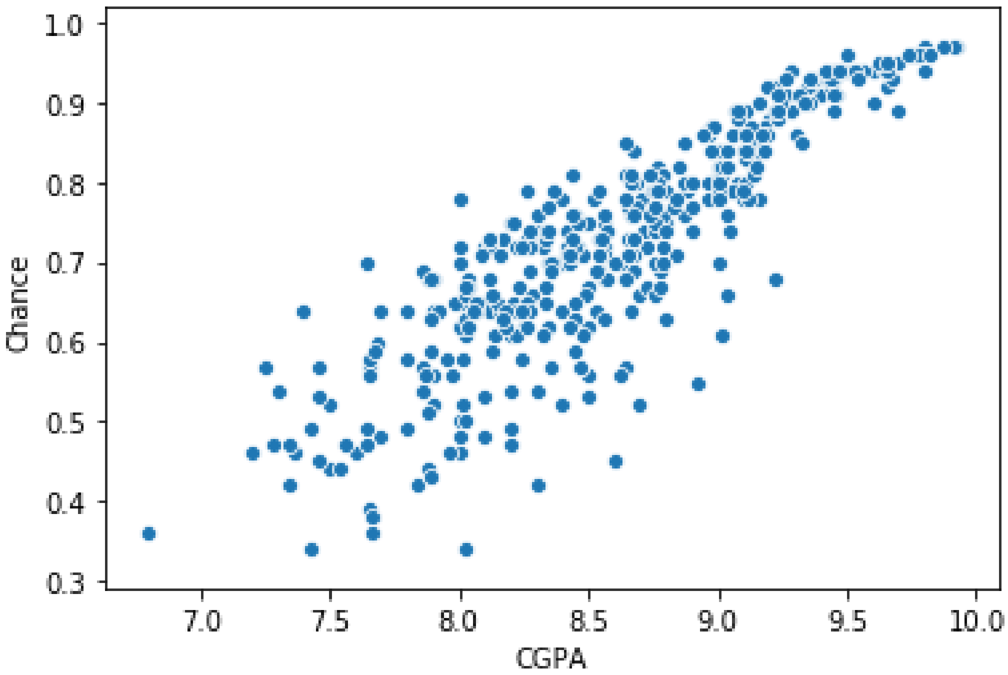
0.0.4 Part 4: Calculate the correlation coefficient between CGPA and Chance of Admission

[152]:
```
print("Correlation coefficient betweeen CGPA and Chance of Admission:␣
 ↪",adm[['CGPA','Chance']].corr()['CGPA']['Chance'])
```

```
Correlation coefficient betweeen CGPA and Chance of Admission:
0.8732890993553011
```

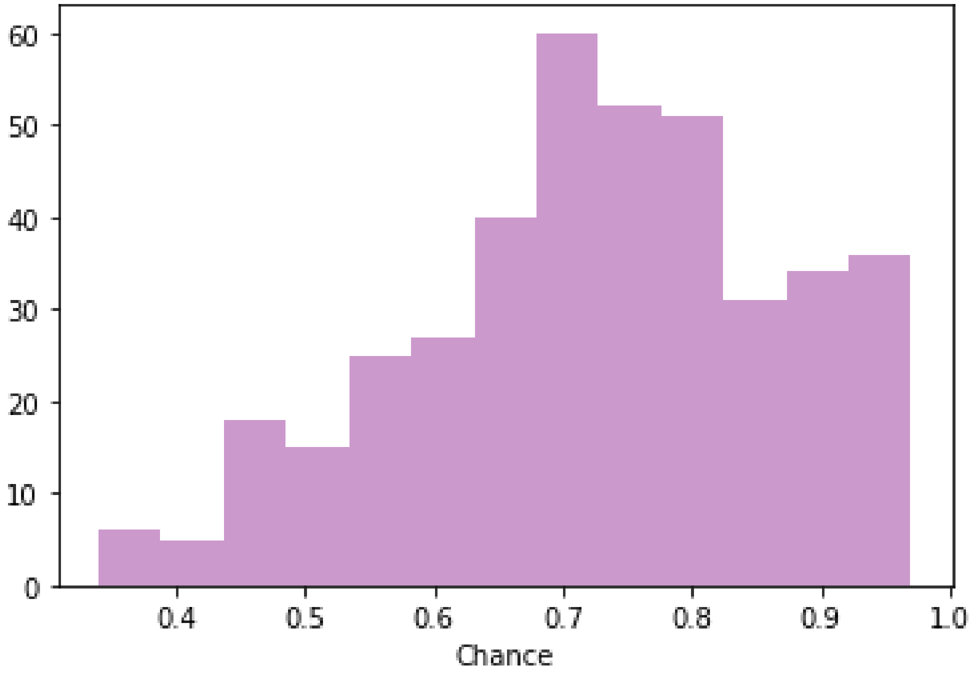
[153]:
```
#Visualising correlation using scatterplot
sns.scatterplot(x= 'CGPA', y = 'Chance', data= adm)
```

[153]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2f860df9320>
```



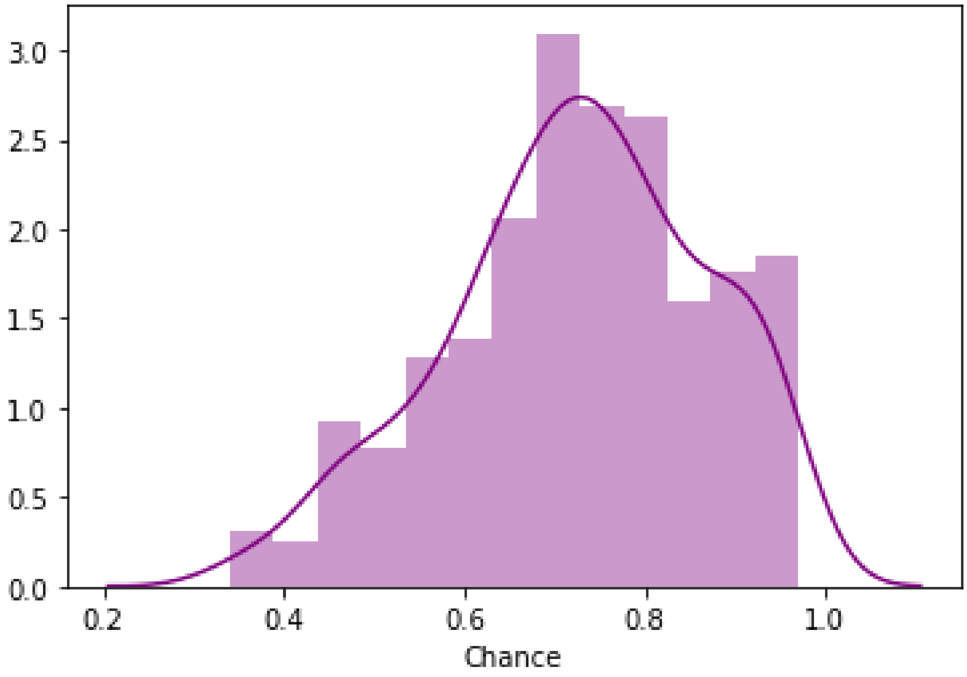
0.0.5 Part 5: Plot a histogram of the chance of admission. What distribution does it seem to have?

[154]:
```
sns.distplot(adm['Chance'], kde=False, color = 'purple');
```



[155]:
```
sns.distplot(adm['Chance'], color = 'Purple')
#The histogram appears to be left skewed since it has an elongated left tail.
```

[155]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2f860935c88>
```



0.0.6 Part 6: Calculate the mean and standard deviation of the chance of admission.

[156]:
```
chance_mean = adm['Chance'].mean()
print("Mean of chance of admission:", chance_mean)
```

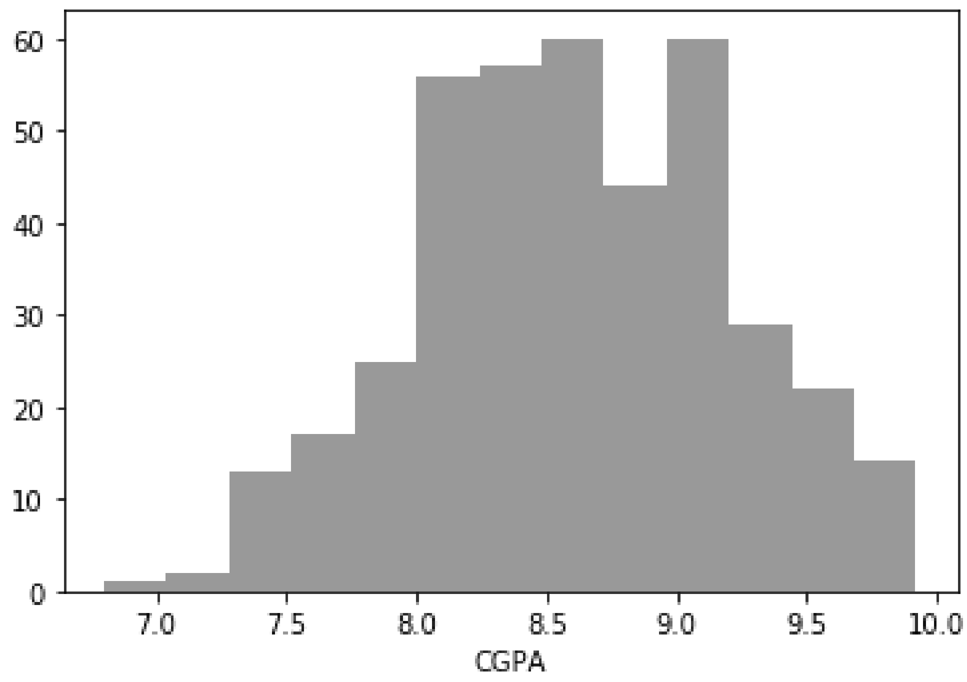
```
Mean of chance of admission: 0.7243499999999996
```

[157]:
```
chance_std = adm['Chance'].std()
print("Standard deviation of chance of admission:", chance_std)
```

```
Standard deviation of chance of admission: 0.14260933017384092
```

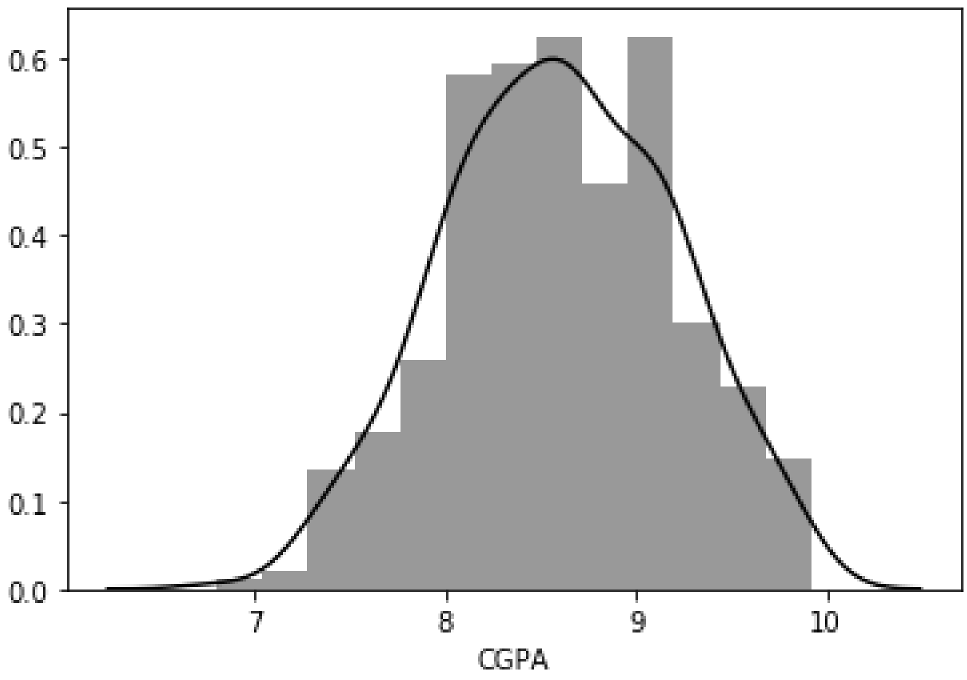
0.0.7 Part 7: Plot a histogram of CGPA

[158]:
```
sns.distplot(adm['CGPA'], kde=False, color = 'black');
```



[159]:
```
sns.distplot(adm['CGPA'], color = 'Black')
#This seems to be normally distributed plot
```

[159]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2f860effa58>
```



0.0.8 Part 8: Calculate the mean and standard deviation of the CGPA

[160]:
```
cgpa_mean = adm['CGPA'].mean()
print("Mean of CGPA:", cgpa_mean)
```

```
Mean of CGPA: 8.598924999999998
```

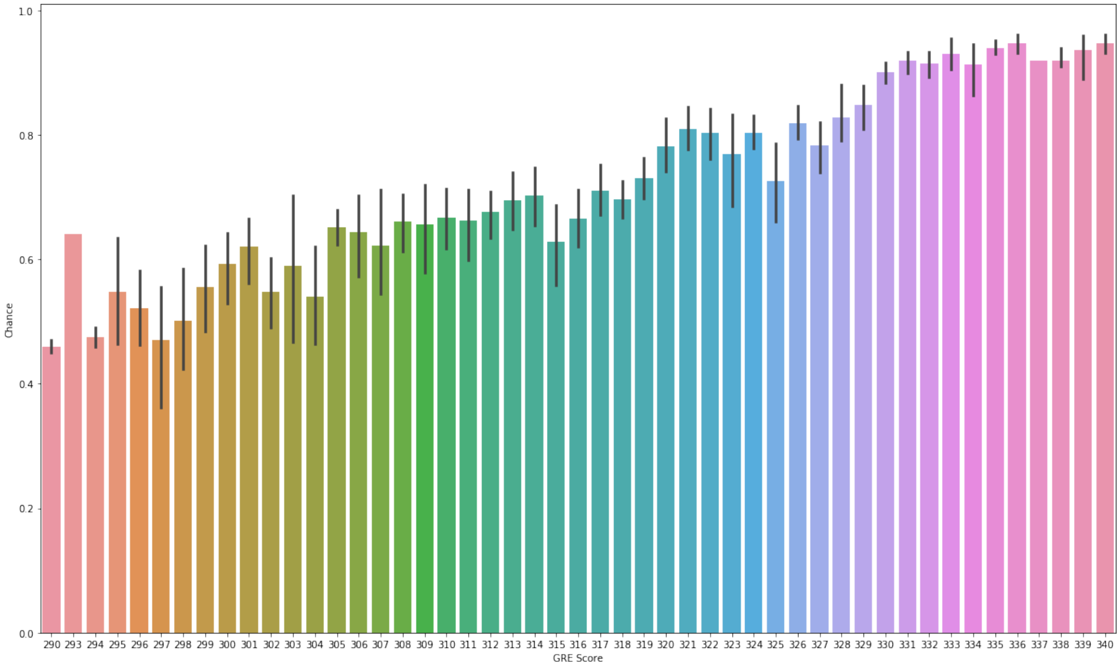
[161]:
```
cgpa_std = adm['CGPA'].std()
print("Standard deviation of CGPA:", cgpa_std)
```

```
Standard deviation of CGPA: 0.5963170964964317
```

0.0.9 Part 9: Plot GRE vs Chance of Admission

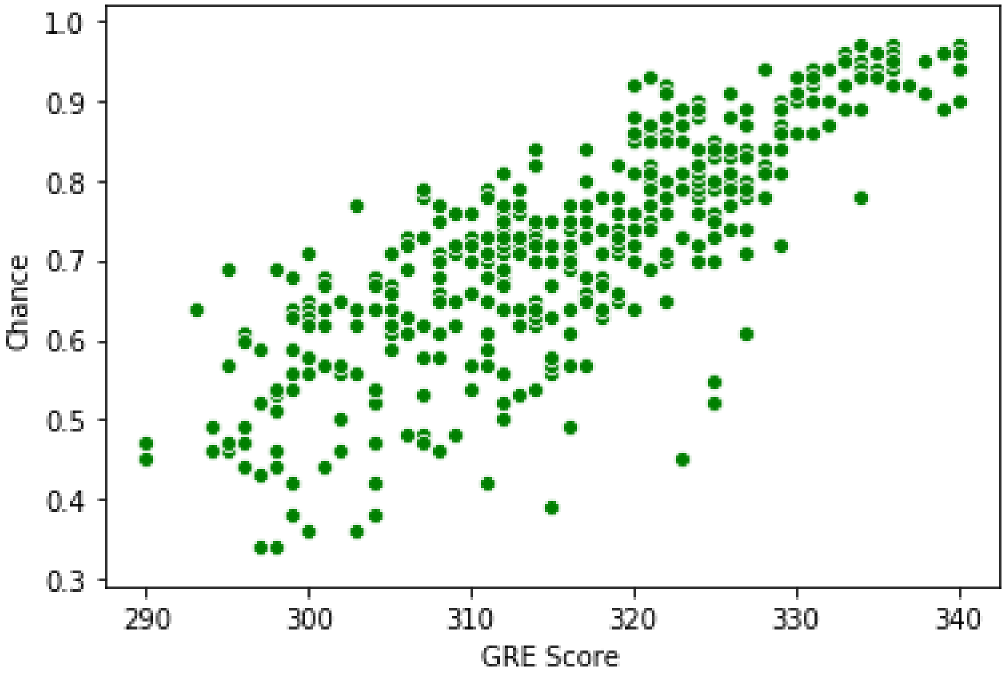
[162]:
```
from matplotlib import pyplot
```

[163]:
```
fig, ax = pyplot.subplots(figsize=(20,12))
sns.barplot(x='GRE Score',y='Chance',data=adm)
plt.show()
```



[164]:
```
sns.scatterplot(x='GRE Score',y='Chance',data=adm,color='green')
```

[164]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2f860edde10>
```



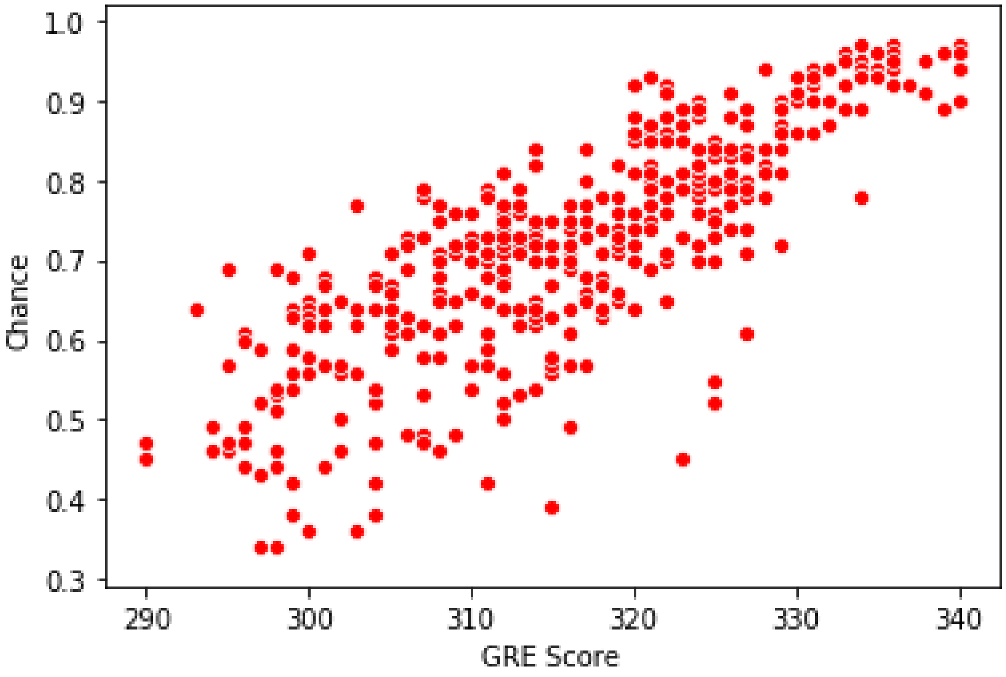
0.0.10 Part 9: Calculate the correlation coefficient between GRE and Chance of Admission

[165]:
```
print("Correlation coefficient between GRE and Chance of Admission: ",adm['GRE␣
 ↪Score'].corr(adm['Chance']))
```

```
Correlation coefficient between GRE and Chance of Admission:  0.8026104595903503
```

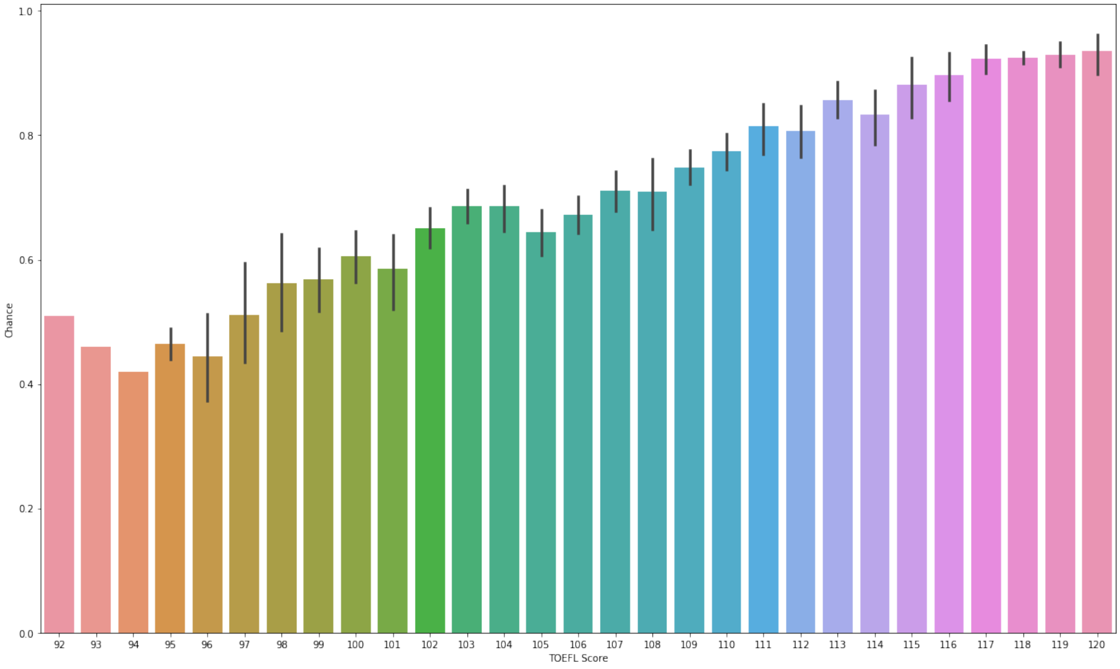
[166]:
```
#Visualising correlation using scatterplot

sns.scatterplot(x= 'GRE Score', y = 'Chance', data= adm, color = 'red')
```

[166]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2f860f05b70>
```



0.0.11 Part 10: Plot TOEFL vs Chance of Admission

[167]:
```
fig, ax = pyplot.subplots(figsize=(20,12))
sns.barplot(x='TOEFL Score',y='Chance',data=adm)
plt.show()
```



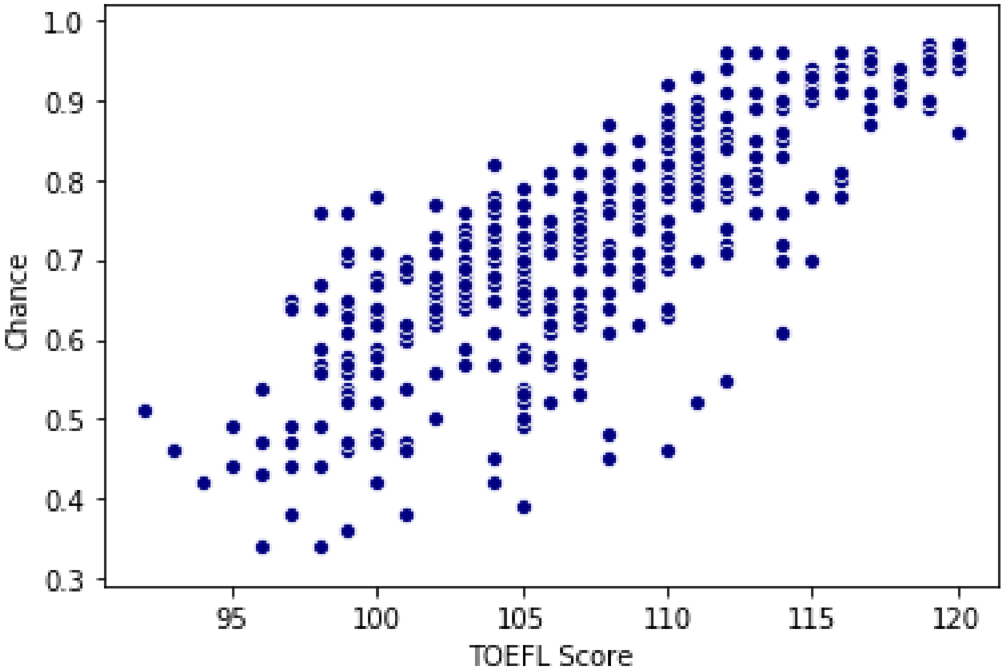
0.0.12 Part 11: Calculate the correlation coefficient between TOEFL and Chance of Admission

[168]:
```
print("Correlation coefficient between TOEFL and Chance of Admission:␣
↪",adm['TOEFL Score'].corr(adm['Chance']))
```

```
Correlation coefficient between TOEFL and Chance of Admission:
0.7915939869351044
```

[169]:
```
#Visualising correlation using scatterplot
sns.scatterplot(x= 'TOEFL Score', y = 'Chance', data= adm,color = 'navy' )
```

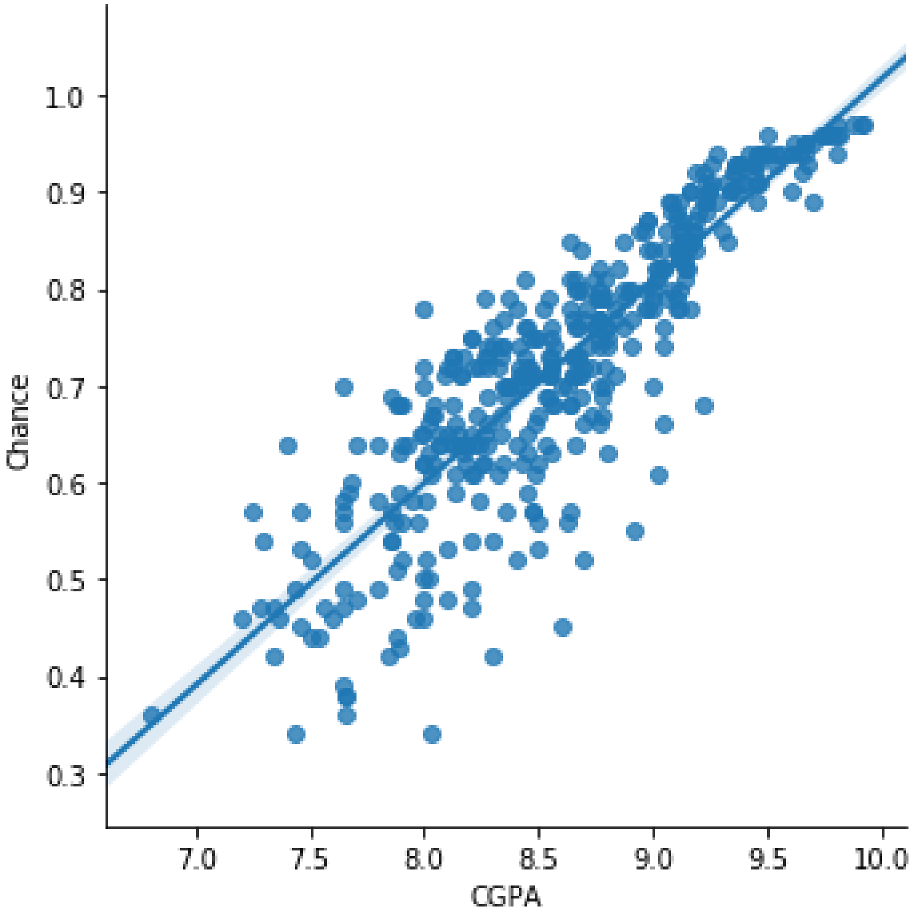
[169]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2f85ec22630>
```



0.0.13 Part 12: Plot CGPA vs Chance of Admission

[170]:
```
sns.lmplot(x='CGPA',y='Chance',data=adm)
```

[170]:
```
<seaborn.axisgrid.FacetGrid at 0x2f85ebcde10>
```



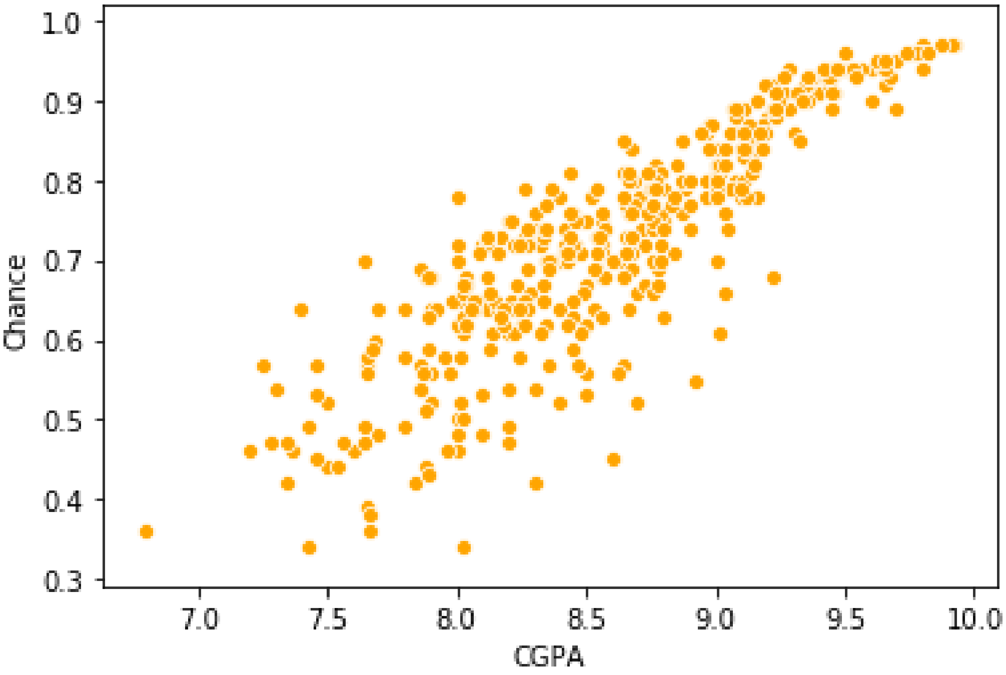
0.0.14 Part 13: Calculate the correlation coefficient between CGPA and Chance of Admission

```
[171]: print("Correlation coefficient between CGPA and Chance of Admission:␣
        ↪",adm['CGPA'].corr(adm['Chance']))
```

```
Correlation coefficient between CGPA and Chance of Admission:
0.8732890993553001
```

```
[172]: #Visualising correlation using scatterplot

       sns.scatterplot(x= 'CGPA', y = 'Chance', data= adm, color = 'orange')
```

```
[172]: <matplotlib.axes._subplots.AxesSubplot at 0x2f860c46160>
```

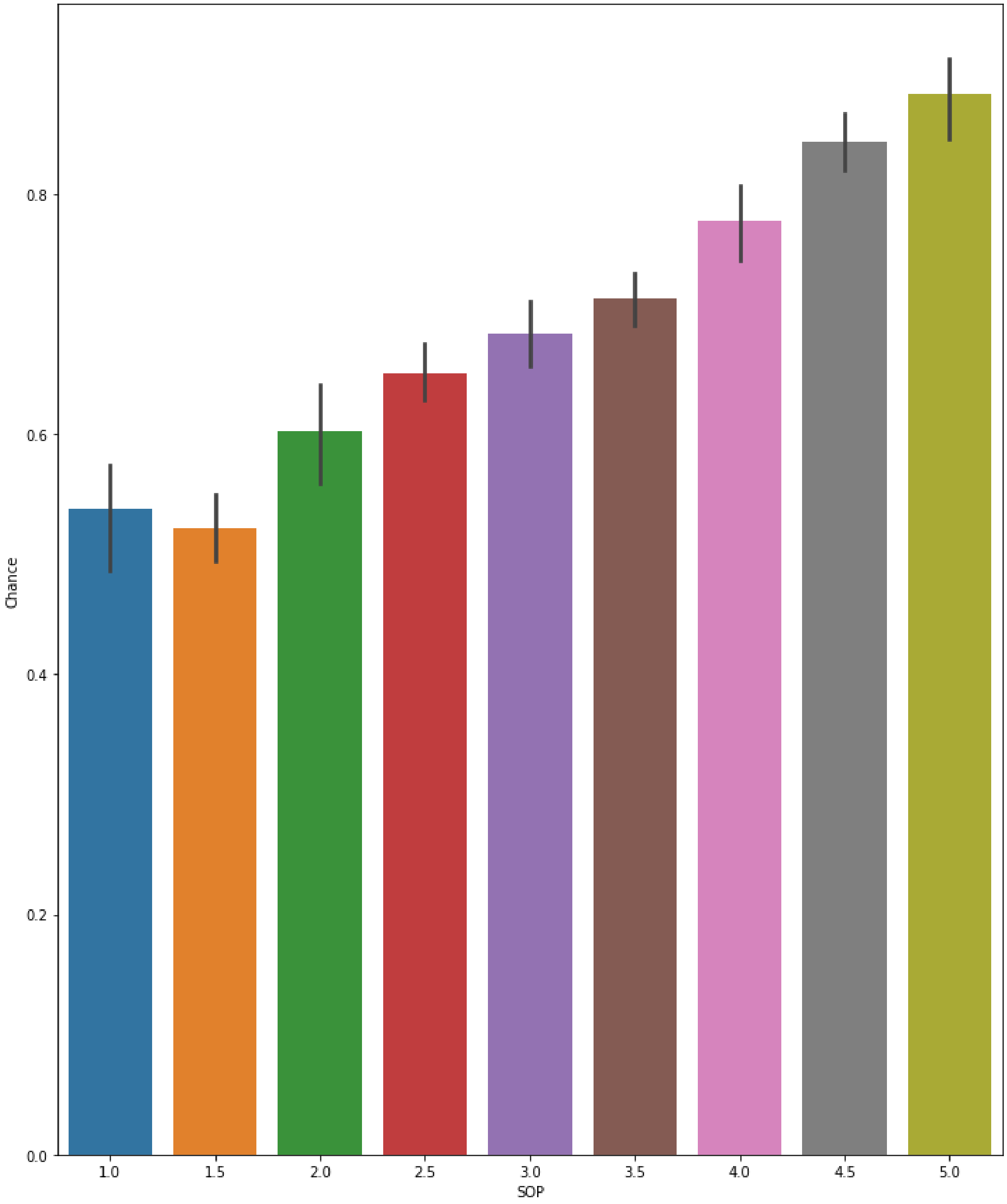


0.0.15 Part 14: Plot one more graph that you think helps the analysis of the data - 1

Plot between SOP Rating and Chance of Admission

[173]:

```
fig, ax = pyplot.subplots(figsize=(12,15))
sns.barplot(x='SOP',y='Chance',data=adm)
plt.show()
```



0.0.16 Part 15: Calculate the correlation coefficient between the variables in Part 14 - 1

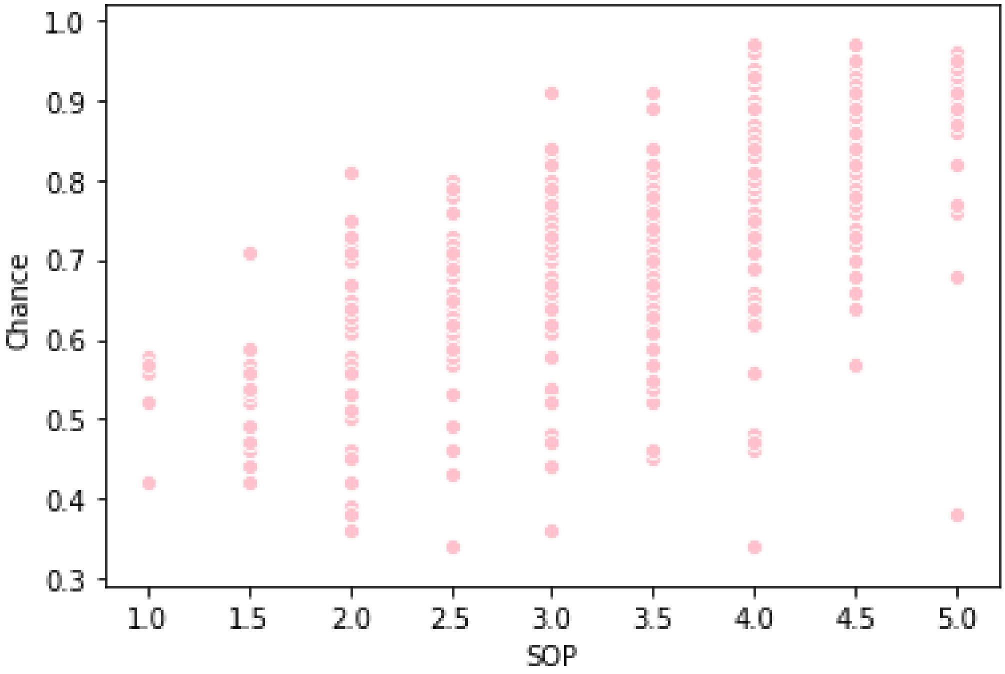
[174]:

```
print("Correlation coefficient between SOP Rating and Chance of Admission:␣
 ↪",adm['SOP'].corr(adm['Chance']))
```

```
Correlation coefficient between SOP Rating and Chance of Admission:
0.6757318583886716
```

[175]:
```
sns.scatterplot(x= 'SOP', y = 'Chance', data= adm, color='pink')
```

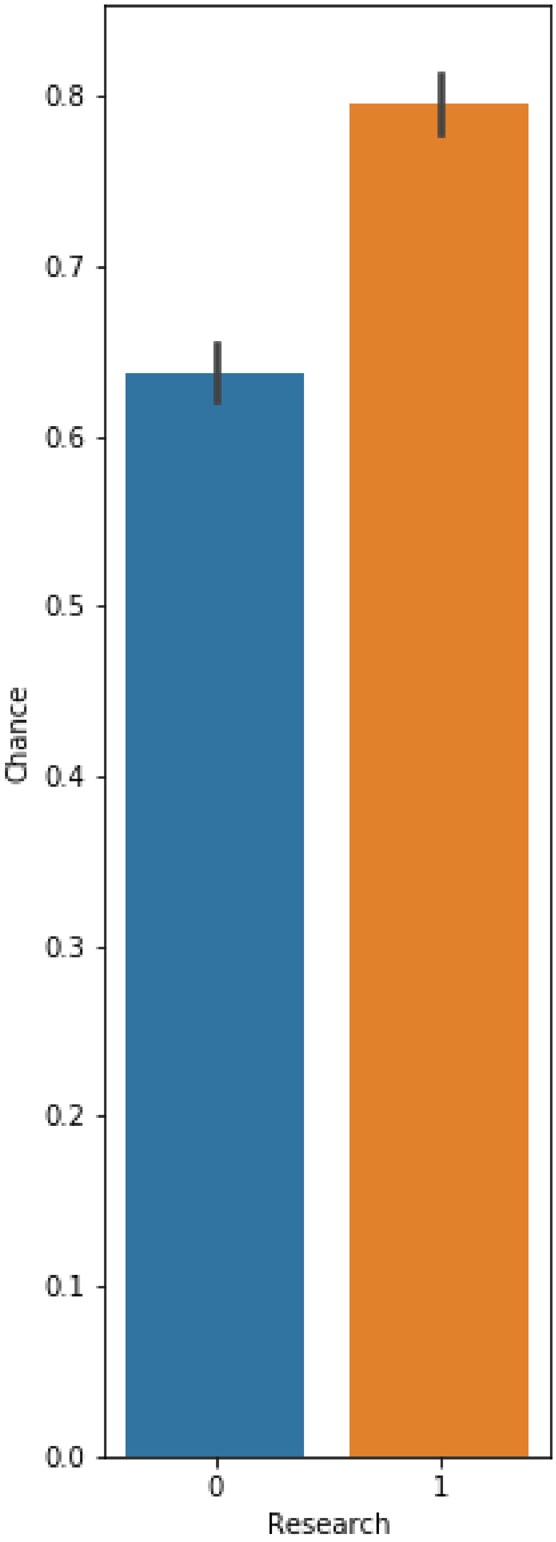
[175]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2f860ea6f60>
```



0.0.17 Part 14: Plot one more graph that you think helps the analysis of the data - 2

Plot between Research and Chance of Admission

[176]:
```
fig, ax = pyplot.subplots(figsize=(3,10))
sns.barplot(x='Research',y='Chance',data=adm)
plt.show()
```


0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
Chance
0
1
Research

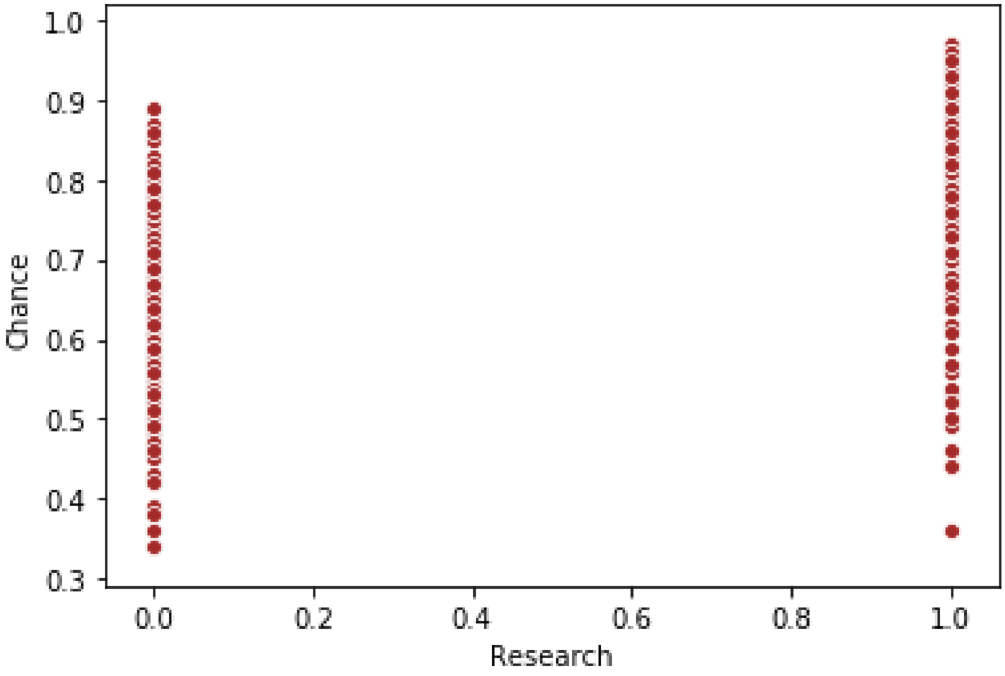
0.0.18 Part 15: Calculate the correlation coefficient between the variables in Part 14 - 2

[177]:
```
print("Correlation coefficient between Research and Chance of Admission:␣
↪",adm['Research'].corr(adm['Chance']))
```

```
Correlation coefficient between Research and Chance of Admission:
0.5532021370190395
```

[178]:
```
sns.scatterplot(x= 'Research', y = 'Chance', data= adm, color='brown')
```

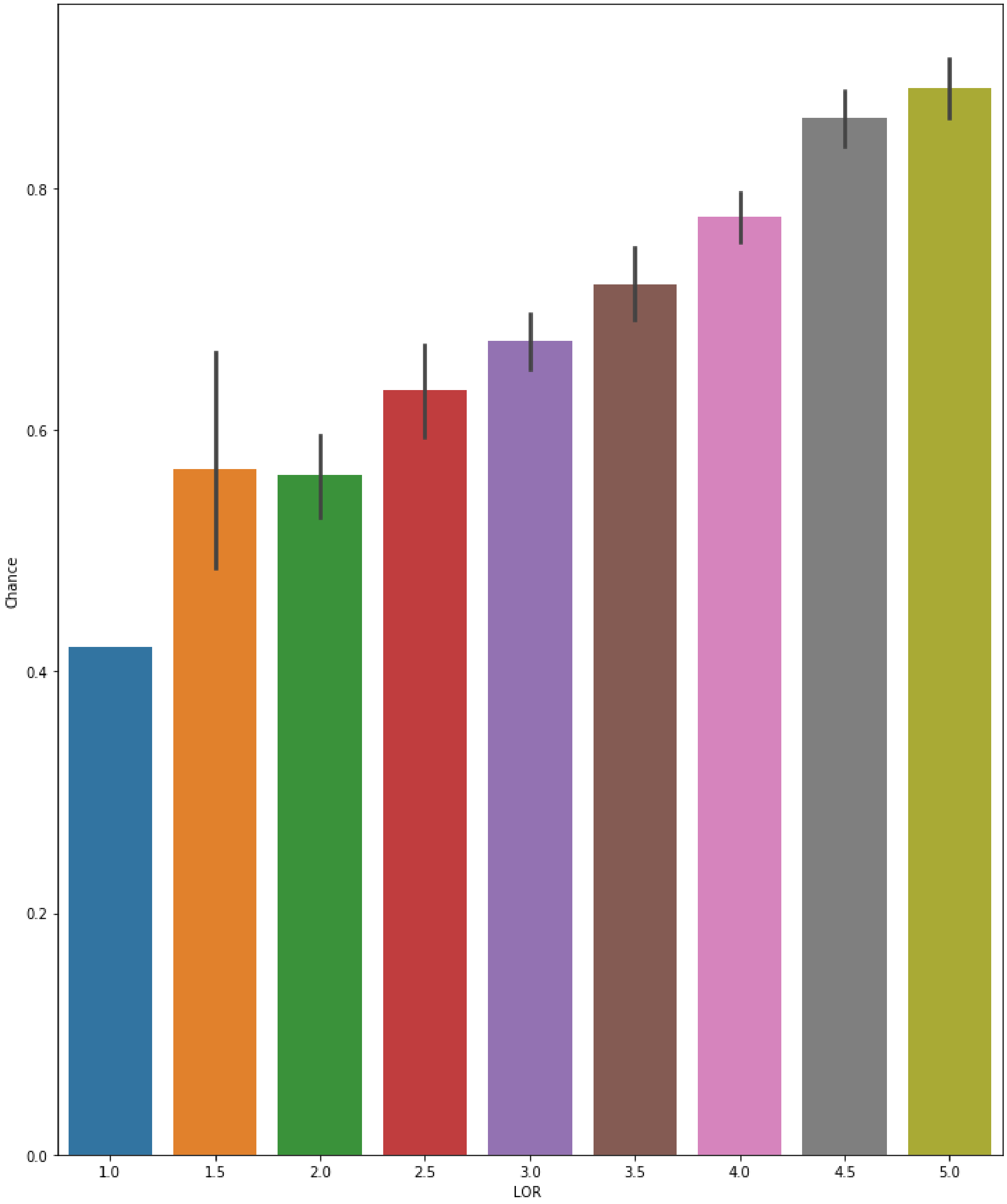
[178]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2f86005a3c8>
```



0.0.19 Part 14: Plot one more graph that you think helps the analysis of the data - 3

Plot between LOR and Chance of Admission

[179]:
```
fig, ax = pyplot.subplots(figsize=(12,15))
sns.barplot(x='LOR ',y='Chance',data=adm)
plt.show()
```



0.0.20 Part 15: Calculate the correlation coefficient between the variables in Part 14 - 3

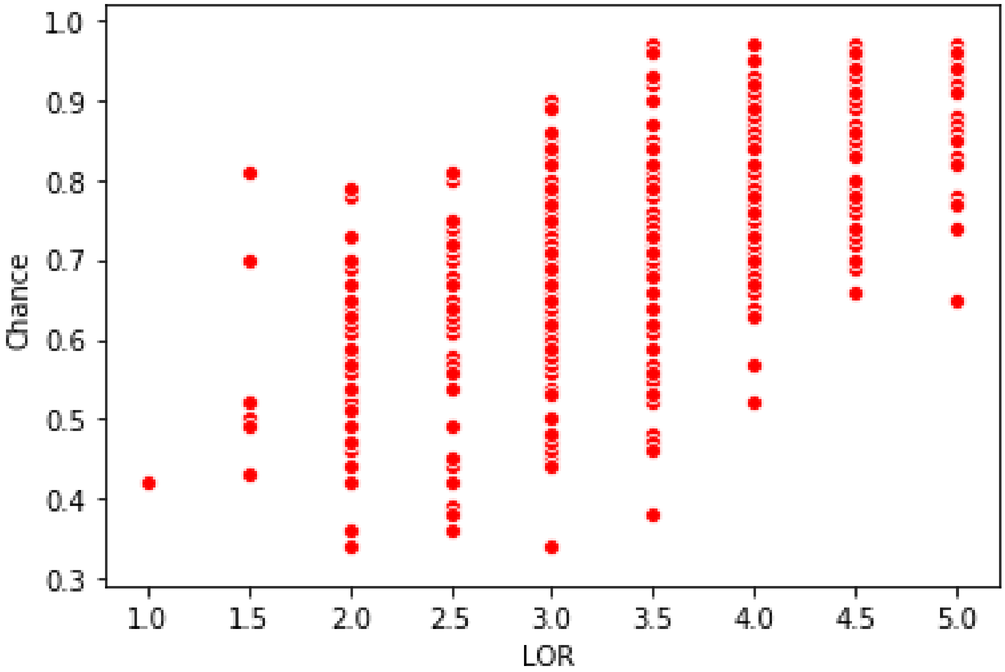
[180]:

```
print("Correlation coefficient between LOR and Chance of Admission: ",adm['LOR␣
↪'].corr(adm['Chance']))
```

```
Correlation coefficient between LOR and Chance of Admission:  0.6698887920106934
```

[181]:
```
sns.scatterplot(x= 'LOR ', y = 'Chance', data= adm, color='red')
```

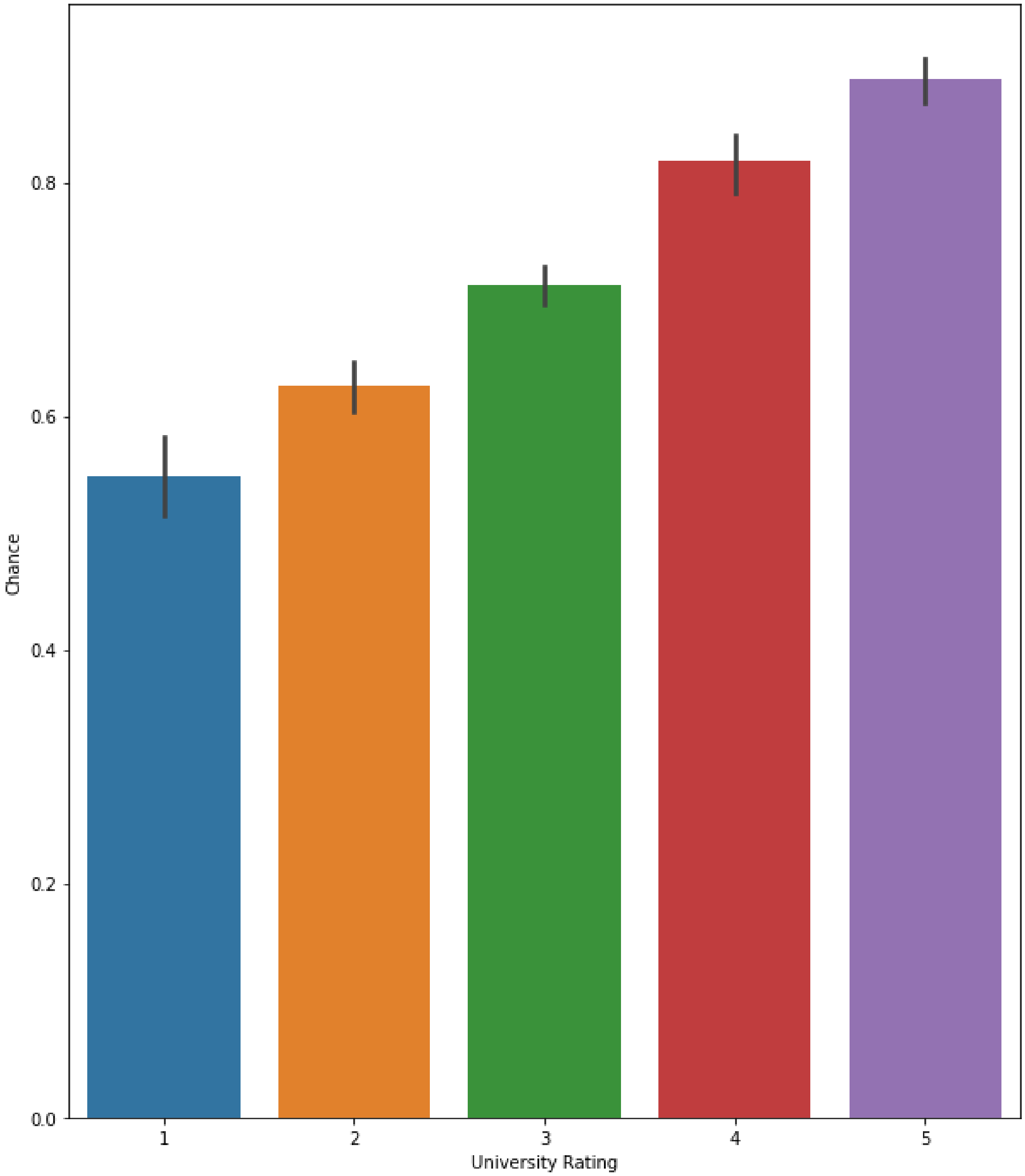
[181]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2f862534e10>
```



0.0.21 Part 14: Plot one more graph that you think helps the analysis of the data - 4

Plot between University Rating and Chance of Admission

[182]:
```
fig, ax = pyplot.subplots(figsize=(10, 12))
sns.barplot(x='University Rating',y='Chance',data=adm)
plt.show()
```



0.0.22 Part 15: Calculate the correlation coefficient between the variables in Part 14 - 4

[183]:
```
print("Correlation coefficient between University Rating and Chance of␣
 ↪Admission: ",adm['University Rating'].corr(adm['Chance']))
```

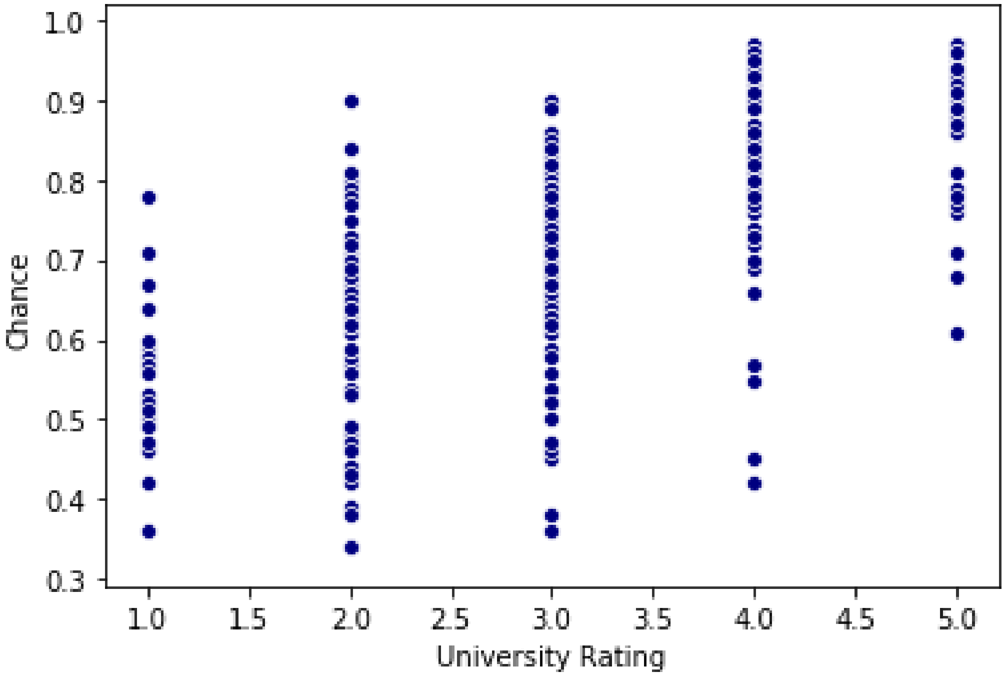
```
Correlation coefficient between University Rating and Chance of Admission:
0.7112502503917218
```

[184]:

```
sns.scatterplot(x= 'University Rating', y = 'Chance', data= adm, color='navy')
```

[184]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x2f862597240>
```


Chance
University Rating