Exploratory Data Analysis

for the course *ECE590: Introductory Machine Learning for Data Science*

Frank Xu

Netid: *hx44*

1&2 Data set and the question

Data source: https://www.kaggle.com/mohansacharya/graduate-admissions/home (https://www.kaggle.com/mohansacharya/graduate-admissions/home)

The data set is from Kaggle, featuring 500 observations of chance of admit data with the students' GRE and TOFEL score, GPA and other important attributes for graduate admission.

The question is: Whether GRE, TOFEL, CGPA, SOP, LOR, university ranking and research experience influence the graduate admission, and if so, to what extent do they affect the chance of admit?

3 Checking data

In [11]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# Reading in data
df=pd.read_csv('Admission_Predict.csv', sep=',')

# Determine if there exist any null values
print(df.isnull().any())

# Changing data types and column names
df.rename(columns = {"LOR ": "LOR", "Chance of Admit ": "Chance of Admit"}, inpl
ace = True)

# Describe data
df.describe()
```

```
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
```

Out[11]:

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	C
count	500.000000	500.000000	500.000000	500.000000	500.000000	500.00000	500.00
mean	250.500000	316.472000	107.192000	3.114000	3.374000	3.48400	8.5764
std	144.481833	11.295148	6.081868	1.143512	0.991004	0.92545	0.6048
min	1.000000	290.000000	92.000000	1.000000	1.000000	1.00000	6.8000
25%	125.750000	308.000000	103.000000	2.000000	2.500000	3.00000	8.1275
50%	250.500000	317.000000	107.000000	3.000000	3.500000	3.50000	8.5600
75%	375.250000	325.000000	112.000000	4.000000	4.000000	4.00000	9.0400
max	500.000000	340.000000	120.000000	5.000000	5.000000	5.00000	9.9200

There seems to have no missing values or erroneous values. The column name need to be changed slightly, and 'Research' column should be boolean. The data set is ready for visualization.

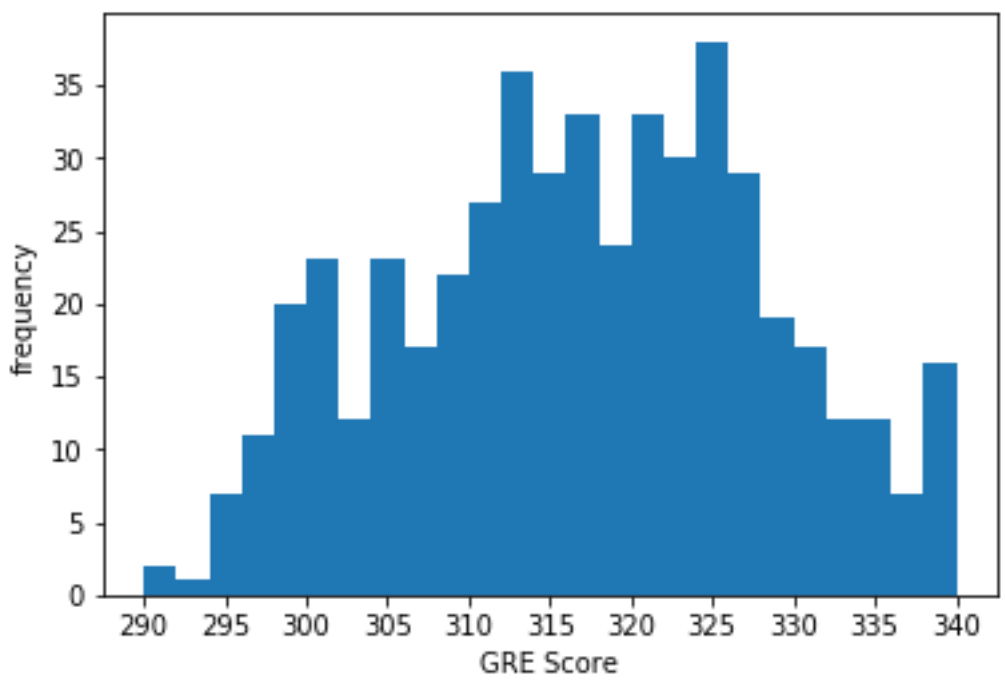
4 Plotting and Exploring

In [2]:

```
# First let's see the distribution of most attributes
plt.xticks(np.arange(290, 341, 5.0))
plt.ylabel('frequency')
plt.xlabel('GRE Score')
plt.hist(df['GRE Score'], bins = 25)
```

Out[2]:

```
(array([ 2.,  1.,  7., 11., 20., 23., 12., 23., 17., 22., 27., 36.,
29.,
        33., 24., 33., 30., 38., 29., 19., 17., 12., 12.,  7., 16.])
,
 array([290., 292., 294., 296., 298., 300., 302., 304., 306., 308.,
310.,
        312., 314., 316., 318., 320., 322., 324., 326., 328., 330.,
332.,
        334., 336., 338., 340.]),
 <a list of 25 Patch objects>)
```

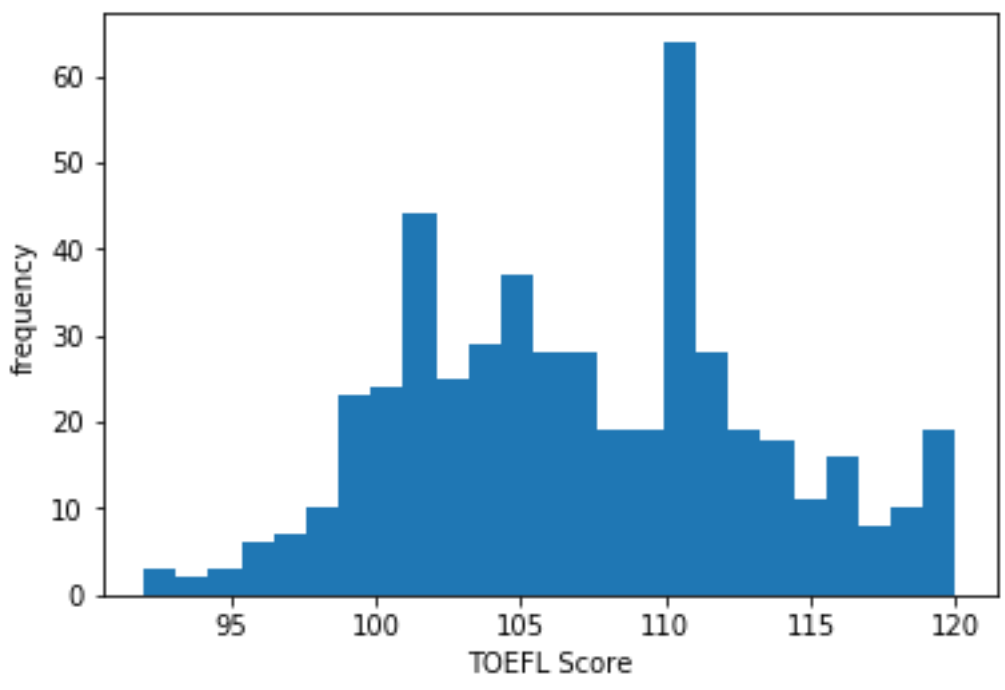


In [3]:

```
plt.xticks(np.arange(90, 121, 5.0))
plt.ylabel('frequency')
plt.xlabel('TOEFL Score')
plt.hist(df['TOEFL Score'], bins = 25)
```

Out[3]:

```
(array([ 3.,  2.,  3.,  6.,  7., 10., 23., 24., 44., 25., 29., 37.,
28.,
        28., 19., 19., 64., 28., 19., 18., 11., 16.,  8., 10., 19.])
,
 array([ 92.  ,  93.12,  94.24,  95.36,  96.48,  97.6 ,  98.72,  99.
84,
        100.96, 102.08, 103.2 , 104.32, 105.44, 106.56, 107.68, 108.
8 ,
        109.92, 111.04, 112.16, 113.28, 114.4 , 115.52, 116.64, 117.
76,
        118.88, 120.  ]),
 <a list of 25 Patch objects>)
```

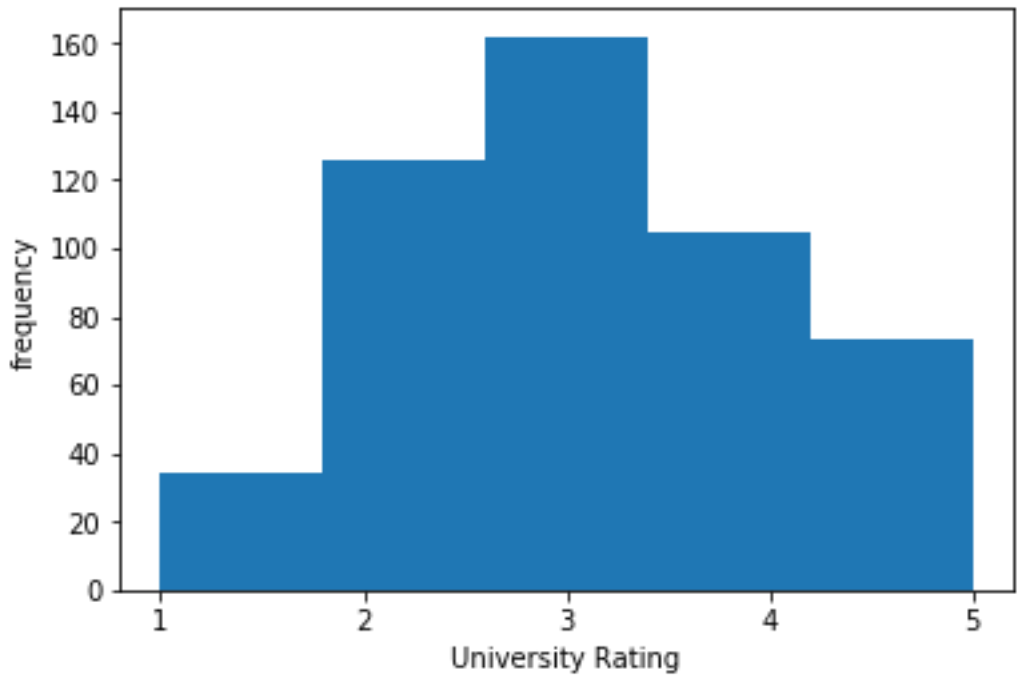


In [4]:

```
plt.xticks(np.arange(1, 6, 1.0))
plt.ylabel('frequency')
plt.xlabel('University Rating')
plt.hist(df['University Rating'], bins = 5)
```

Out[4]:

```
(array([ 34., 126., 162., 105.,  73.]),
 array([1. , 1.8, 2.6, 3.4, 4.2, 5. ]),
 <a list of 5 Patch objects>)
```

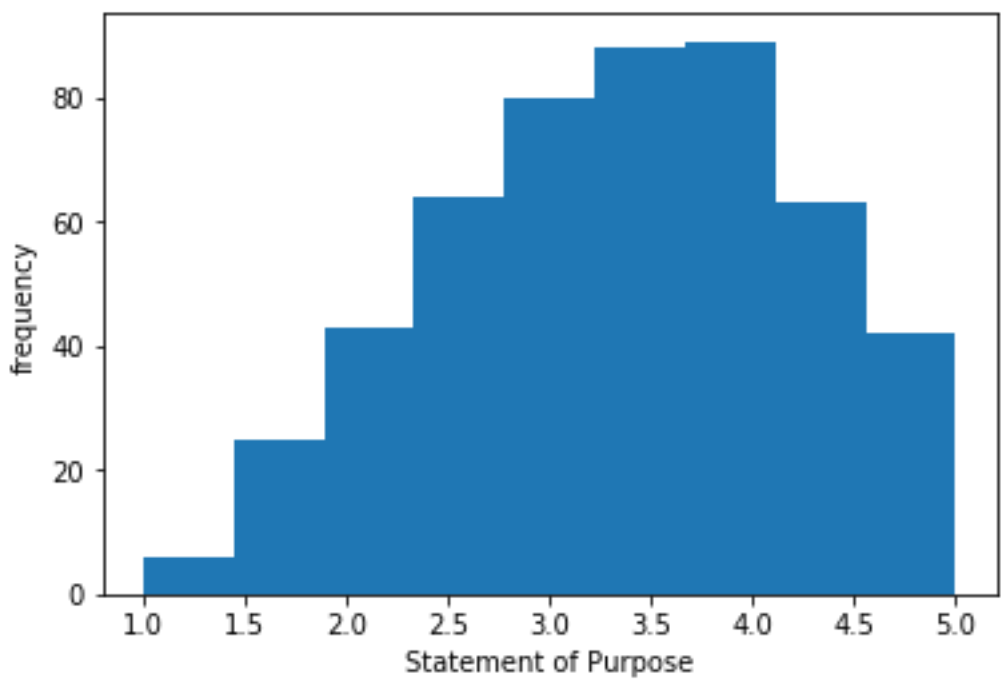


In [5]:

```
plt.xticks(np.arange(1, 6, 0.5))
plt.ylabel('frequency')
plt.xlabel('Statement of Purpose')
plt.hist(df['SOP'], bins = 9)
```

Out[5]:

```
(array([ 6., 25., 43., 64., 80., 88., 89., 63., 42.]),
 array([1.        , 1.44444444, 1.88888889, 2.33333333, 2.77777778,
        3.22222222, 3.66666667, 4.11111111, 4.55555556, 5.        ])
,
 <a list of 9 Patch objects>)
```

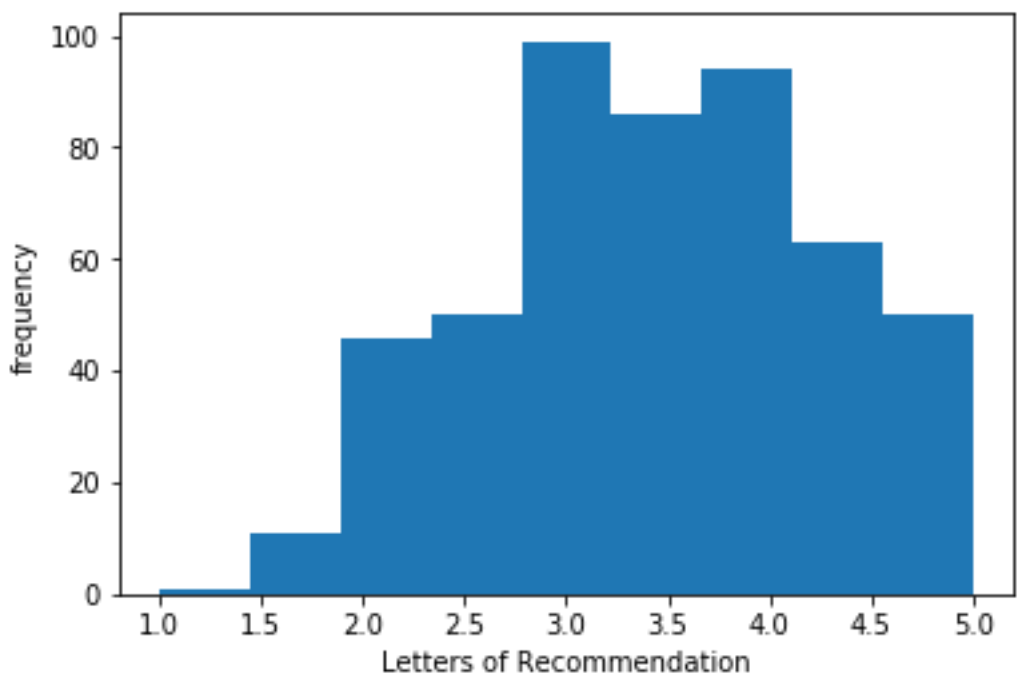


In [6]:

```
plt.xticks(np.arange(1, 6, 0.5))
plt.ylabel('frequency')
plt.xlabel('Letters of Recommendation')
plt.hist(df['LOR'], bins = 9)
```

Out[6]:

```
(array([ 1., 11., 46., 50., 99., 86., 94., 63., 50.]),
 array([1.        , 1.44444444, 1.88888889, 2.33333333, 2.77777778,
        3.22222222, 3.66666667, 4.11111111, 4.55555556, 5.        ])
,
 <a list of 9 Patch objects>)
```

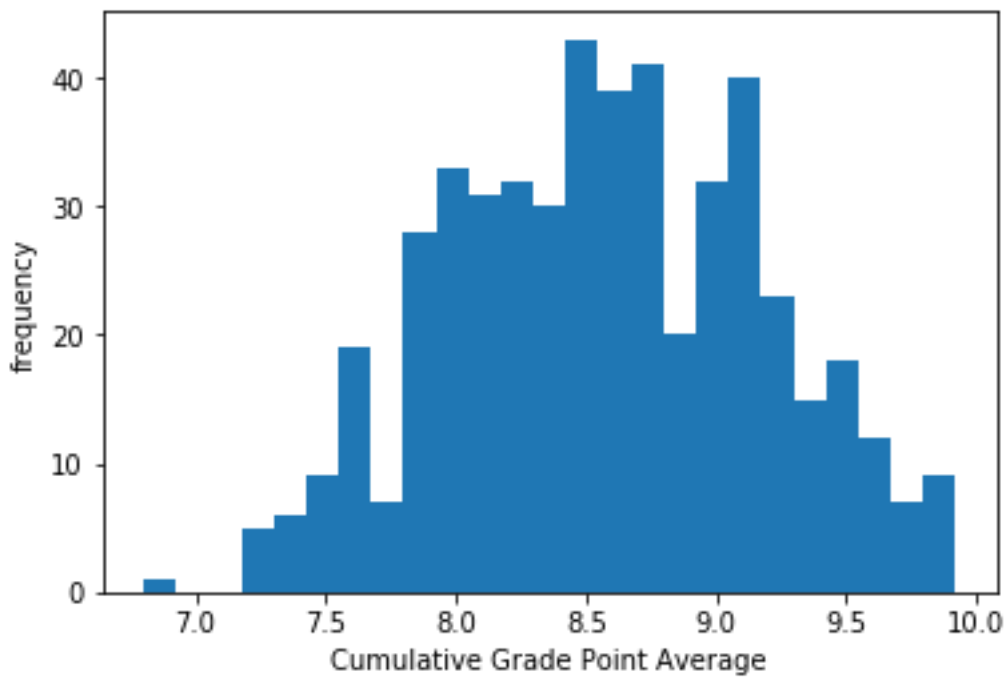


In [7]:

```
plt.xticks(np.arange(6, 11, 0.5))
plt.ylabel('frequency')
plt.xlabel('Cumulative Grade Point Average')
plt.hist(df['CGPA'], bins = 25)
```

Out[7]:

```
(array([ 1.,  0.,  0.,  5.,  6.,  9., 19.,  7., 28., 33., 31., 32.,
30.,
        43., 39., 41., 20., 32., 40., 23., 15., 18., 12.,  7.,  9.])
,
 array([6.8   , 6.9248, 7.0496, 7.1744, 7.2992, 7.424 , 7.5488, 7.67
36,
        7.7984, 7.9232, 8.048 , 8.1728, 8.2976, 8.4224, 8.5472, 8.67
2 ,
        8.7968, 8.9216, 9.0464, 9.1712, 9.296 , 9.4208, 9.5456, 9.67
04,
        9.7952, 9.92  ]),
 <a list of 25 Patch objects>)
```

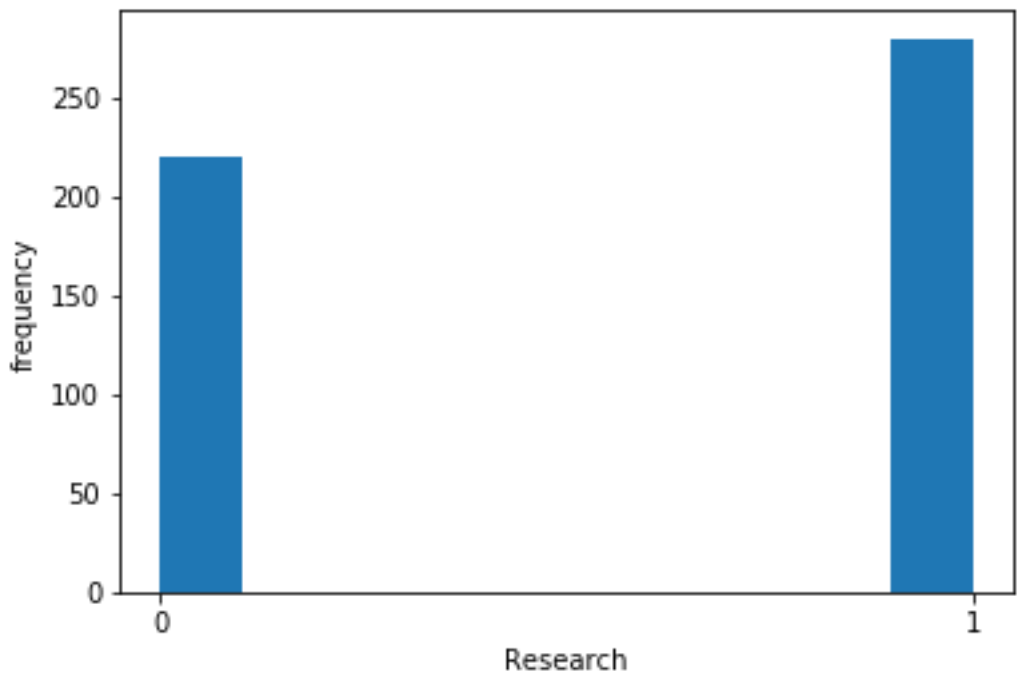


In [8]:

```
plt.xticks([0, 1])
plt.ylabel('frequency')
plt.xlabel('Research')
plt.hist(df['Research'])
```

Out[8]:

```
(array([220.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0., 280.])
,
 array([0. , 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1. ]),
 <a list of 10 Patch objects>)
```

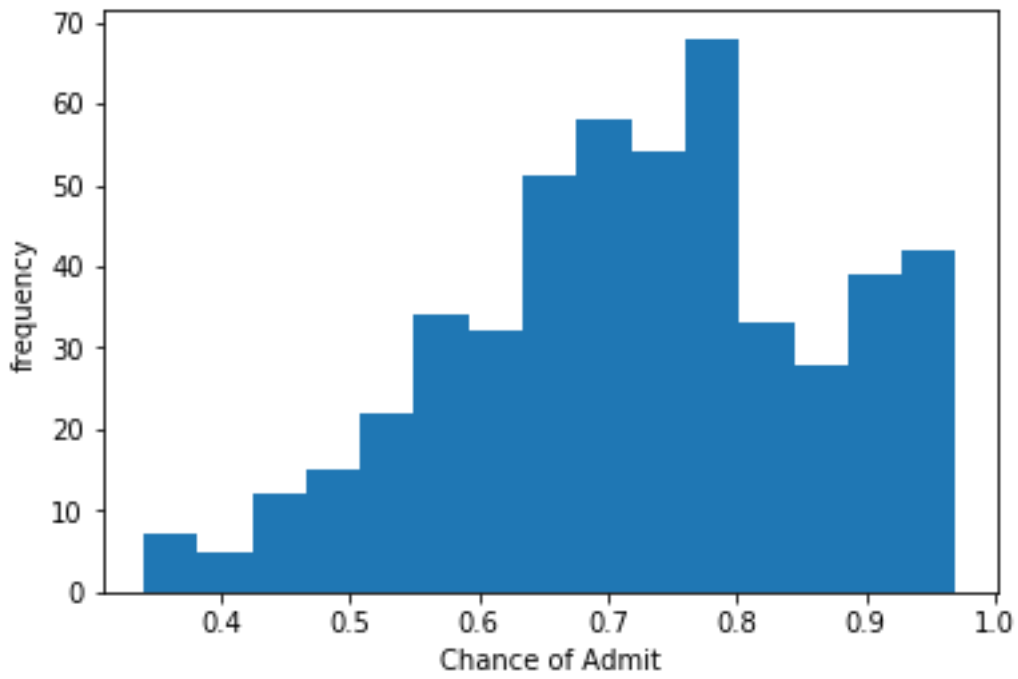


In [9]:

```
plt.xticks(np.arange(0, 1.1, 0.1))
plt.ylabel('frequency')
plt.xlabel('Chance of Admit')
plt.hist(df['Chance of Admit'], bins = 15)
```

Out[9]:

```
(array([ 7.,  5., 12., 15., 22., 34., 32., 51., 58., 54., 68., 33.,
28.,
        39., 42.]),
 array([0.34 , 0.382, 0.424, 0.466, 0.508, 0.55 , 0.592, 0.634, 0.67
6,
        0.718, 0.76 , 0.802, 0.844, 0.886, 0.928, 0.97 ]),
 <a list of 15 Patch objects>)
```



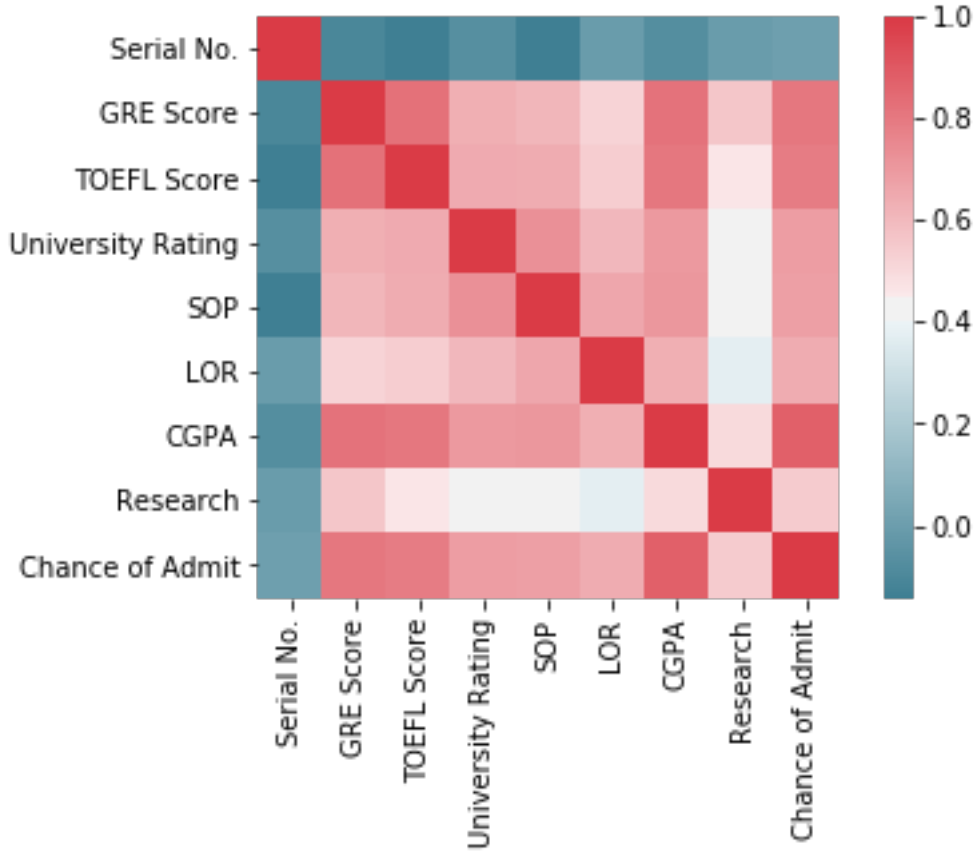
In [10]:

```
import seaborn as sns

# Correlation between columns
sns.heatmap(df.corr(), mask=np.zeros_like(df.corr(), dtype=np.bool), cmap=sns.di
verging_palette(220, 10, as_cmap=True),
            square=True)
```

Out[10]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1a1b2a5550>
```



5 Explaining and Interpretation

For the distributions, we can see some interesting trend for certain columns:

1. GRE score clearly has two peaks in the distribution, between 310-315 and just above 325, repectively. Since GRE score is produced by a normal distribution, those peaks are absolutely abnormal naturely. Combining with our own experience, it proves that certain number of students take GRE multiple times until they reach 310 and 325, which are considered the "Pass" and "Excellent" score in GRE.
2. TOFEL score has a more clear trend of peaks in a normal distribution. Same as GRE, students are trying to get above 100 and 110, which are considered the "Pass" and "Excellent" score in TOFEL.
3. The SOP and LOR are scored by human, more specifically, the admission office. The both have a slightly skewed distribution, with μ around 3.5 - 4.0.
4. The CGPA also shows how students are "trying" to get their GPA to certain levels: 7.5, 8.5 and 9.0.
5. Finally, about chance of admit: The correlation plot shows that among those attribute, the most correlated are CGPA, GRE and TOFEL, sorted by importance. By the way, those three scores themself are also highly correlated, which means a student with a high CGPA has greater chance of having higher GRE and TOFEL scores. In comparison with those three scores, Research experience (whether they have it or not) is the least important (still, a 0.5 correlation for the worst one), followed by letters of recommendation.

Overall speaking, if you are going to apply for a graduate program, you should be aware that certain score "platforms" for GRE, TOFEL and CGPA, that if you still have chance to improve your score, try to beat those levels because that is what most students do.