Рубежный контроль №1, Грызин Алексей РТ5-61Б

Задача 1, Вариант 6

- Для заданного набора данных проведите корреляционный анализ. В случае наличия пропусков в данных удалите строки или колонки, содержащие пропуски. Сделайте выводы о возможности построения моделей машинного обучения и о возможном вкладе признаков в модель.

Дополнительные требования по группам:

- Для студентов группы РТ5-61Б - для пары произвольных колонок данных построить график "Jointplot".

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

df = pd.read_csv('data/Admission_Predict.csv')
```

Анализ датасета

In [ ]:
```
df.head()
```

Out[ ]:

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
0	1	337	118	4	4.5	4.5	9.65	1	0.92
1	2	324	107	4	4.0	4.5	8.87	1	0.76
2	3	316	104	3	3.0	3.5	8.00	1	0.72
3	4	322	110	3	3.5	2.5	8.67	1	0.80
4	5	314	103	2	2.0	3.0	8.21	0	0.65

In [ ]:
```
df.describe()
```

Out[ ]:

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	C
count	400.000000	400.000000	400.000000	400.000000	400.000000	400.000000	400.000
mean	200.500000	316.807500	107.410000	3.087500	3.400000	3.452500	8.598
std	115.614301	11.473646	6.069514	1.143728	1.006869	0.898478	0.596
min	1.000000	290.000000	92.000000	1.000000	1.000000	1.000000	6.800
25%	100.750000	308.000000	103.000000	2.000000	2.500000	3.000000	8.170
50%	200.500000	317.000000	107.000000	3.000000	3.500000	3.500000	8.610
75%	300.250000	325.000000	112.000000	4.000000	4.000000	4.000000	9.062
max	400.000000	340.000000	120.000000	5.000000	5.000000	5.000000	9.920

In [ ]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 400 entries, 0 to 399
Data columns (total 9 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   Serial No.         400 non-null    int64  
 1   GRE Score          400 non-null    int64  
 2   TOEFL Score        400 non-null    int64  
 3   University Rating  400 non-null    int64  
 4   SOP                400 non-null    float64
 5   LOR                400 non-null    float64
 6   CGPA               400 non-null    float64
 7   Research           400 non-null    int64  
 8   Chance of Admit    400 non-null    float64
dtypes: float64(4), int64(5)
memory usage: 28.2 KB
```

In [ ]:
```
df.isnull().sum()
```

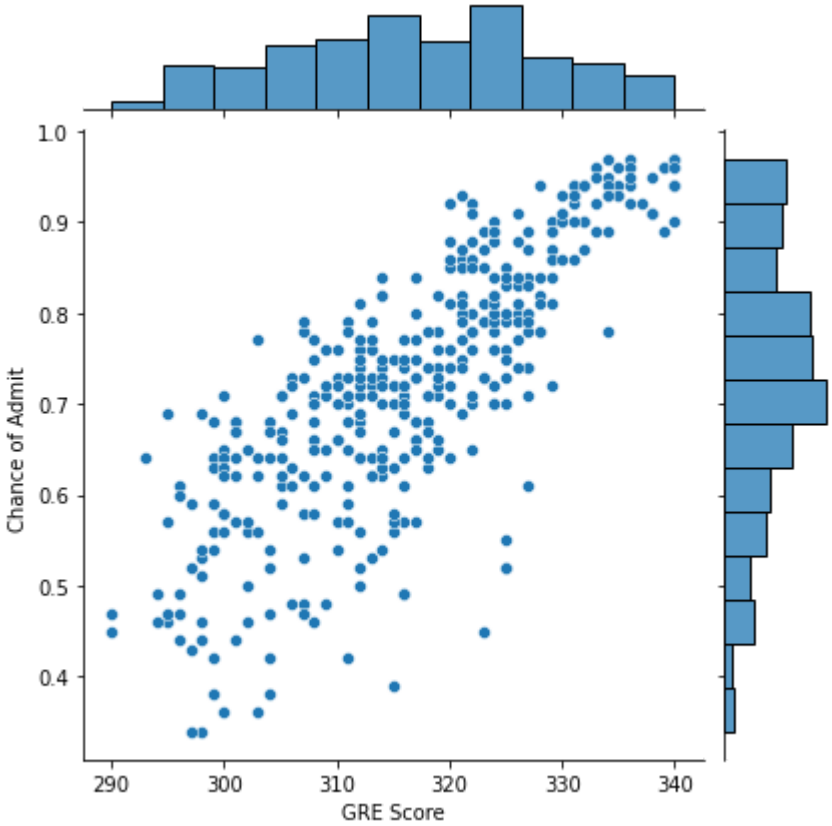
Out[ ]:
```
Serial No.           0
GRE Score            0
TOEFL Score          0
University Rating    0
SOP                  0
LOR                  0
CGPA                 0
Research             0
Chance of Admit      0
dtype: int64
```

> Как видно, пропуски отсутствуют, а значит нет необходимости в удалении колонок или строк.

Диаграмма Jointplot

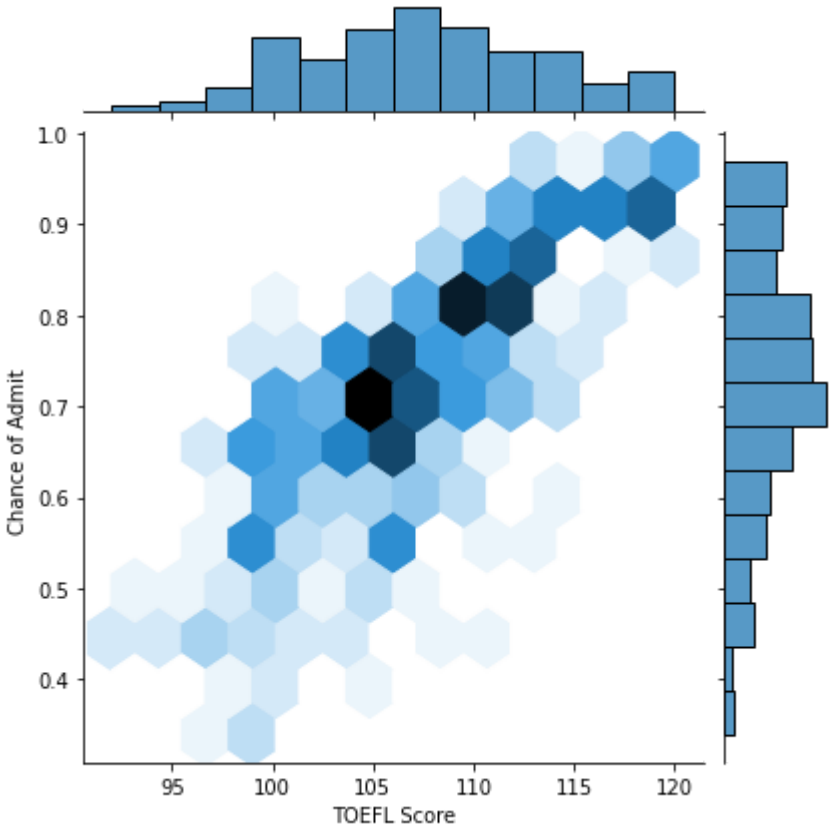
In [ ]:
```
sns.jointplot(x="GRE Score", y="Chance of Admit ", data=df)
```

Out[ ]:
```
<seaborn.axisgrid.JointGrid at 0x286c03340>
```



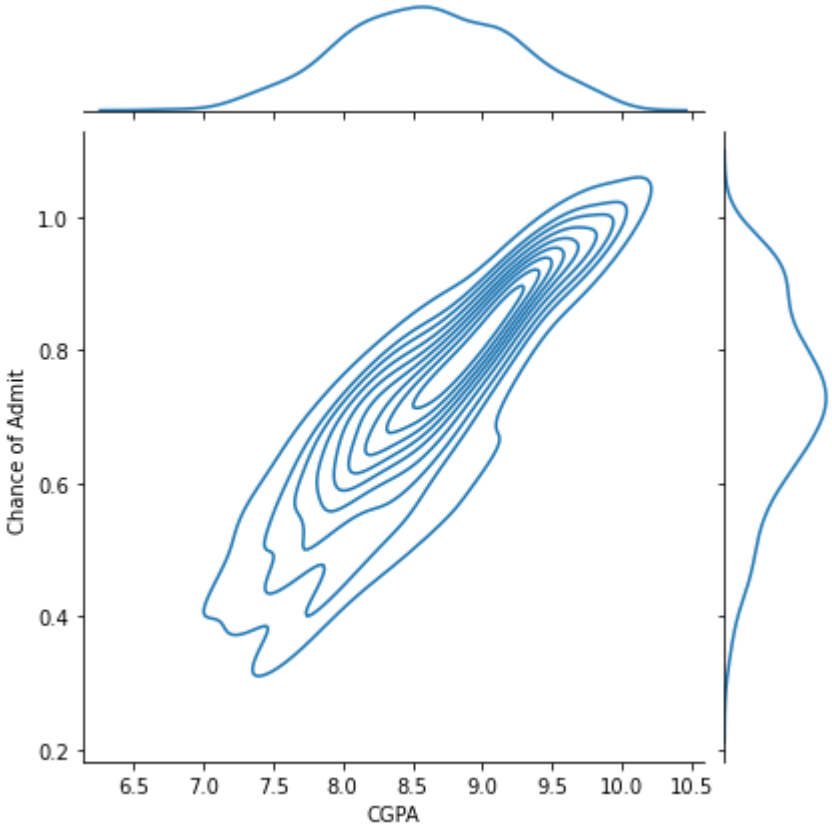
In [ ]:
```
sns.jointplot(x="TOEFL Score", y="Chance of Admit ", data=df, kind='hex')
```

Out[ ]:
```
<seaborn.axisgrid.JointGrid at 0x286c44310>
```



In [ ]:
```
sns.jointplot(x="CGPA", y="Chance of Admit ", data=df, kind="kde")
```

Out[ ]:
```
<seaborn.axisgrid.JointGrid at 0x286e63c10>
```



Корреляционный анализ

- В данном датасете целевым признаком является параметр "Chance of Admit". Рассмотрим, как остальные параметры с ним коррелируют.

In [ ]:
```
df.corr()
```

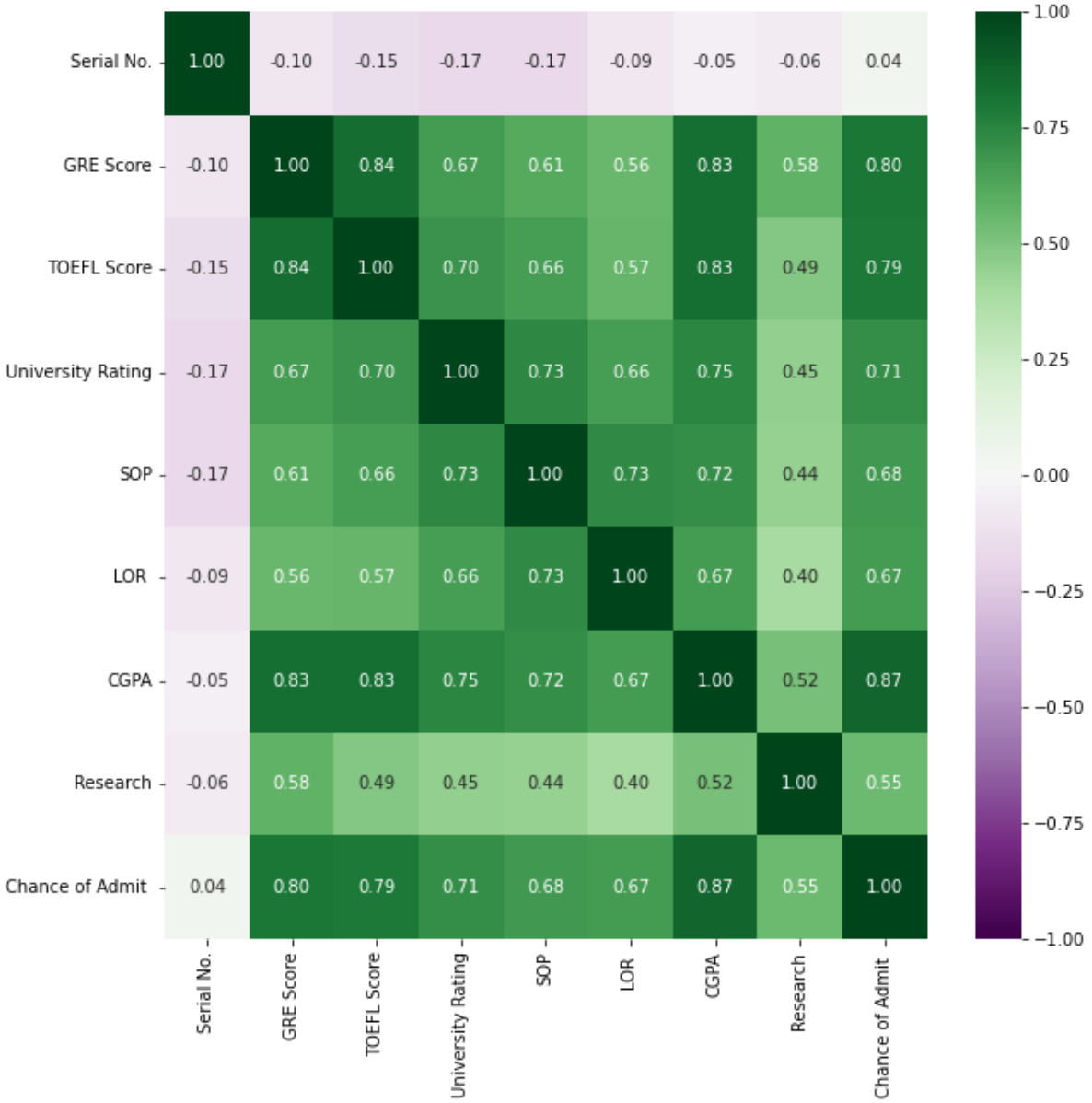
Out[ ]:

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	R
Serial No.	1.000000	-0.097526	-0.147932	-0.169948	-0.166932	-0.088221	-0.045608	-(
GRE Score	-0.097526	1.000000	0.835977	0.668976	0.612831	0.557555	0.833060	(
TOEFL Score	-0.147932	0.835977	1.000000	0.695590	0.657981	0.567721	0.828417	0
University Rating	-0.169948	0.668976	0.695590	1.000000	0.734523	0.660123	0.746479	(
SOP	-0.166932	0.612831	0.657981	0.734523	1.000000	0.729593	0.718144	0
LOR	-0.088221	0.557555	0.567721	0.660123	0.729593	1.000000	0.670211	0
CGPA	-0.045608	0.833060	0.828417	0.746479	0.718144	0.670211	1.000000	(
Research	-0.063138	0.580391	0.489858	0.447783	0.444029	0.396859	0.521654	1
Chance of Admit	0.042336	0.802610	0.791594	0.711250	0.675732	0.669889	0.873289	(

In [ ]:
```
fig, ax = plt.subplots(1, 1, sharex='col', sharey='row', figsize=(10, 10))

sns.heatmap(df.corr(), annot=True,  fmt='.2f', cmap=plt.cm.PRGn, vmin=-1, vm
```

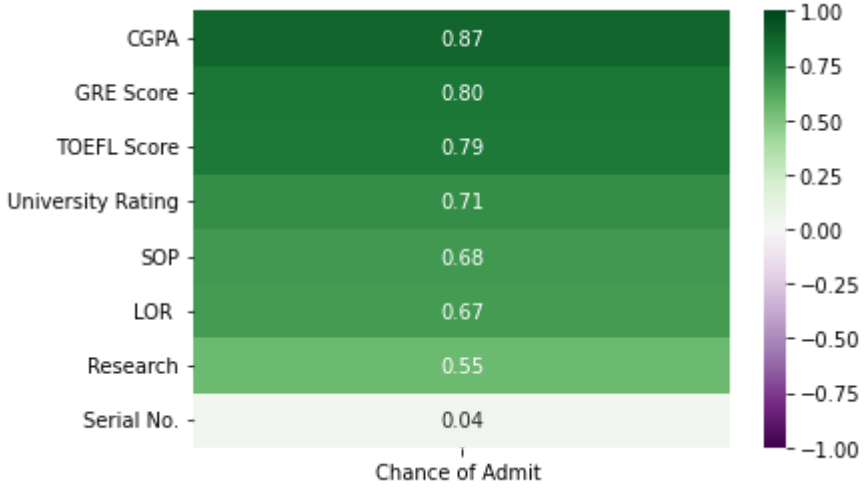
Out[ ]: <AxesSubplot:>



In [ ]:
```
chance_of_admit = pd.DataFrame(df.corr()["Chance of Admit "].sort_values(asc

sns.heatmap(chance_of_admit, annot=True,  fmt='.2f', cmap=plt.cm.PRGn, vmin=
```

Out[ ]: <AxesSubplot:>



Выше представлены матрица корреляций признаков между собой, а также

матрица корреляции для целевого признака. Из этих матриц можно сделать следующие выводы:

- Значение параметра "Serial No" никак не коррелирует со всеми остальными параметрами. В дальнейшем этот столбец можно будет опустить.
- Целевой признак достаточно неплохо коррелирует (положительно) со всеми параметрами. Очень высокая положительная корреляция наблюдается с "CGPA", "GRE Score", "TOEFL Score".
- Также высокая корреляция наблюдается между парами этих параметров, а значит во избежании мультиколлинеарности необходимо выбрать один из этих признаков. Логичнее всего оставить "CGPA", т.к. с ним у целевого признака наблюдается наибольшая связь.

> В результате корреляционного анализа было принято решение в моделях машинного обучения для прогноза целевого признака использовать параметры: "CGPA", "University Rating", "SOP", "LOR" и "Research".