

РК №1

Рубежный контроль №1. Ищенко А.С. Вариант №6: задание 1, датасет 6

Задание:

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.

Набор данных состоит из данных, необходимых для прогнозирования поступления в аспирантуру в Индии. Набор содержит следующие данные:

Баллы GRE (из 340) Баллы TOEFL (из 120) Рейтинг университета (из 5)
Заявление о целях (из 5) Рекомендательное письмо (из 5) Средний балл бакалавриата (из 10) Исследовательский опыт (0 или 1) Шанс допуска (от 0 до 1)

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsRegressor,
KNeighborsClassifier
```

```
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import f1_score
from sklearn.preprocessing import OrdinalEncoder
from sklearn import preprocessing
from sklearn import utils
```

```
%matplotlib inline
sns.set(style="ticks")
```

```
df = pd.read_csv('Admission_Predict_Ver1.1.csv')
```

```
df.shape
```

```
(500, 9)
```

```
df.head(5)
```

	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

```
df.columns
```

```
Index(['Serial No.', 'GRE Score', 'TOEFL Score', 'University Rating',
'SOP',
      'LOR ', 'CGPA', 'Research', 'Chance of Admit '],
      dtype='object')
```

Название последней колонки имеет незаметный лишний пробел в конце строки, уберем его

```
df.rename(columns={'Chance of Admit ': 'Chance of Admit'},
inplace=True)
```

```
df.columns
```

```
Index(['Serial No.', 'GRE Score', 'TOEFL Score', 'University Rating',
'SOP',
      'LOR ', 'CGPA', 'Research', 'Chance of Admit'],
      dtype='object')
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 500 entries, 0 to 499
Data columns (total 9 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Serial No.            500 non-null   int64
1   GRE Score              500 non-null   int64
2   TOEFL Score            500 non-null   int64
3   University Rating      500 non-null   int64
4   SOP                    500 non-null   float64
5   LOR                    500 non-null   float64
6   CGPA                   500 non-null   float64
7   Research               500 non-null   int64
8   Chance of Admit        500 non-null   float64
dtypes: float64(4), int64(5)
memory usage: 35.3 KB
```

```
df.dtypes
```

```
Serial No.          int64
GRE Score           int64
TOEFL Score         int64
University Rating   int64
SOP                 float64
LOR                 float64
CGPA                float64
Research            int64
Chance of Admit     float64
dtype: object
```

```
print('Число пропусков')
df.isna().sum()
```

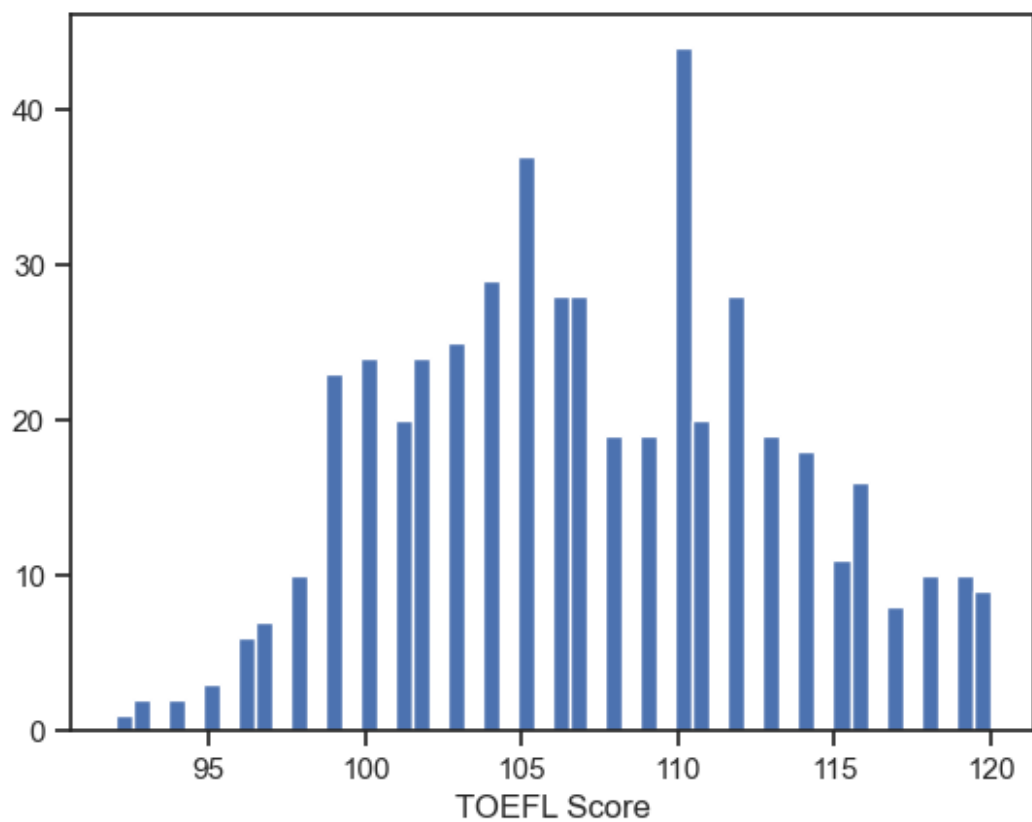
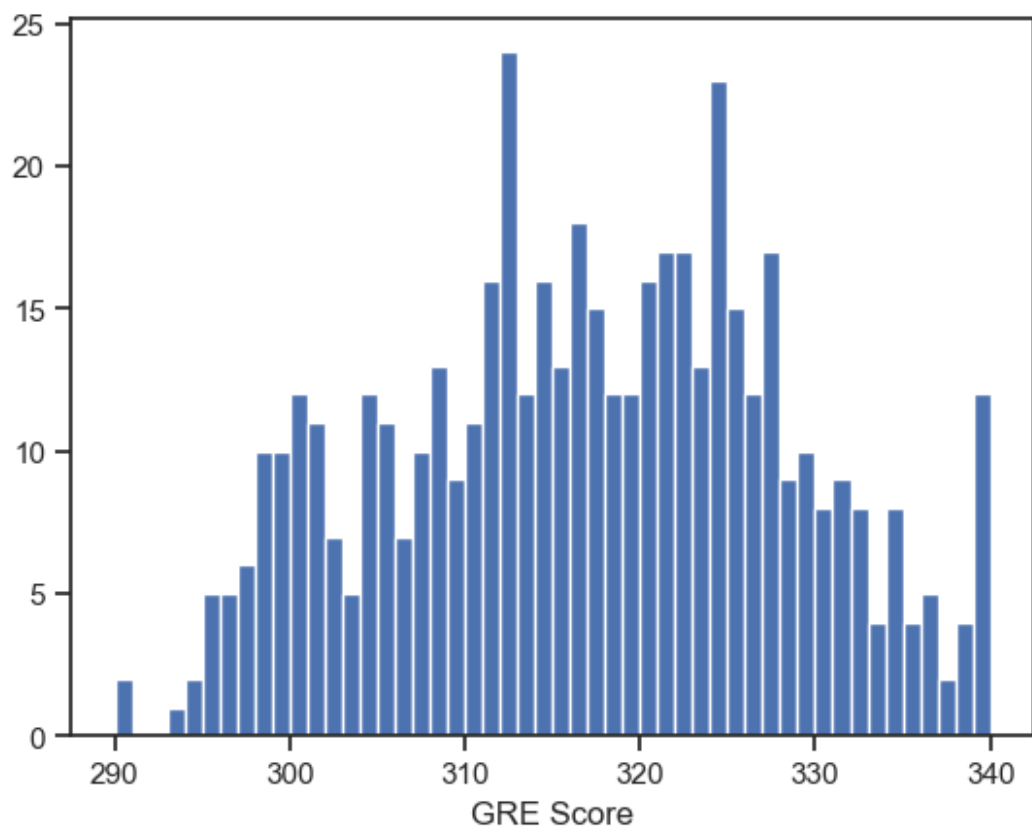
```
Число пропусков
```

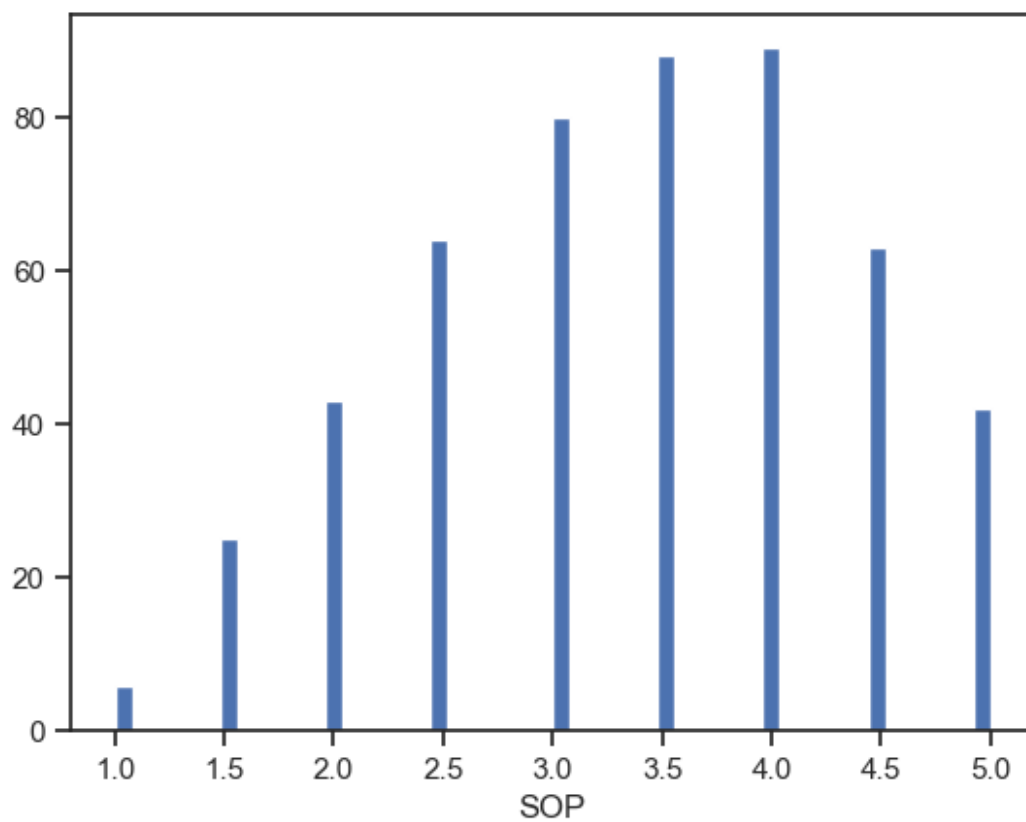
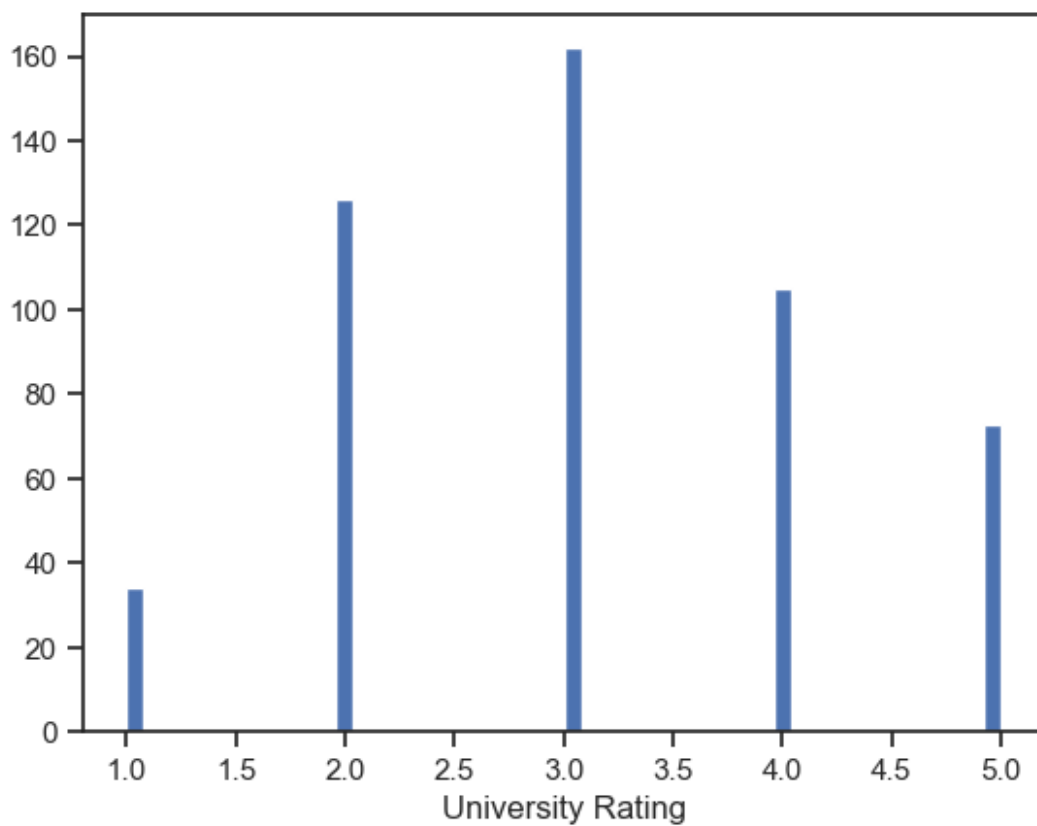
```
Serial No.          0
GRE Score           0
TOEFL Score         0
University Rating   0
SOP                 0
LOR                 0
CGPA                0
Research            0
Chance of Admit     0
dtype: int64
```

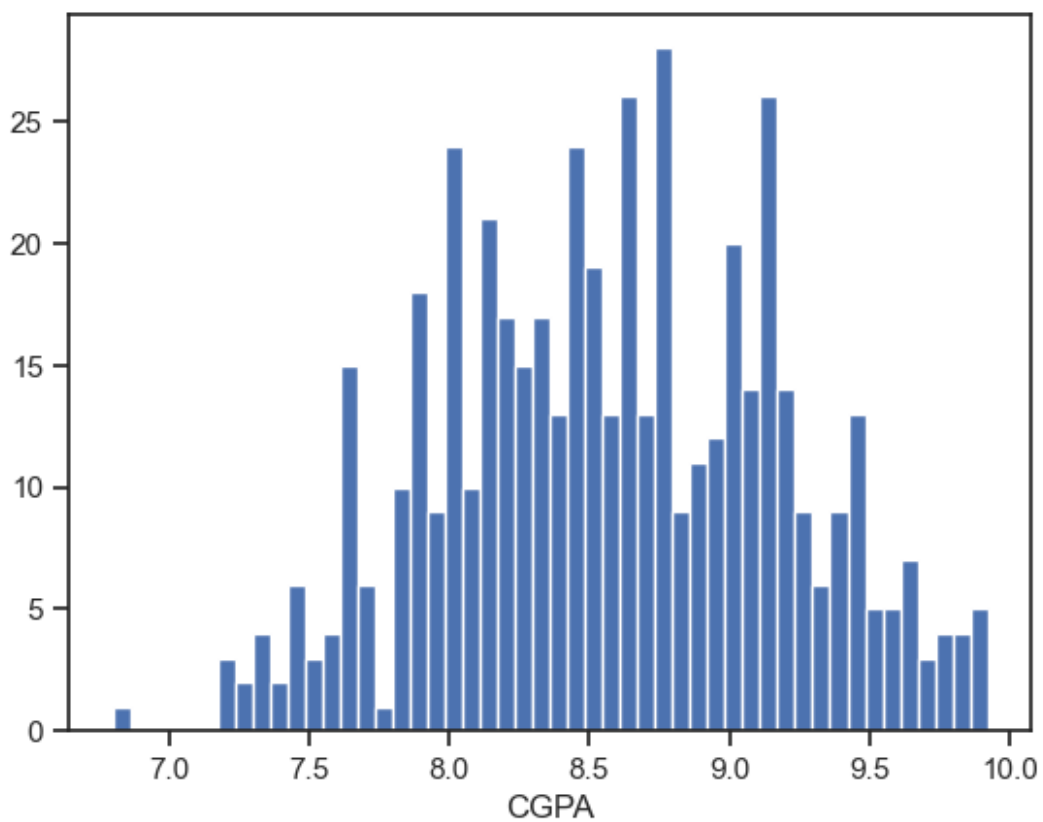
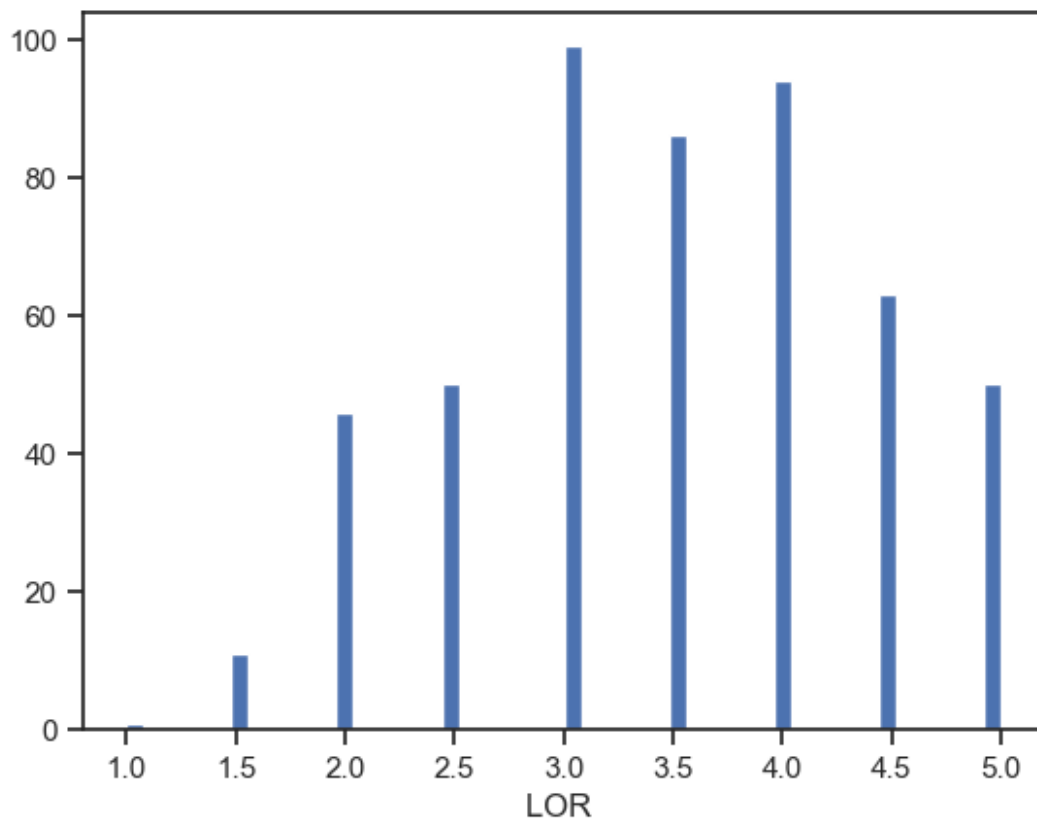
Визуализация данных

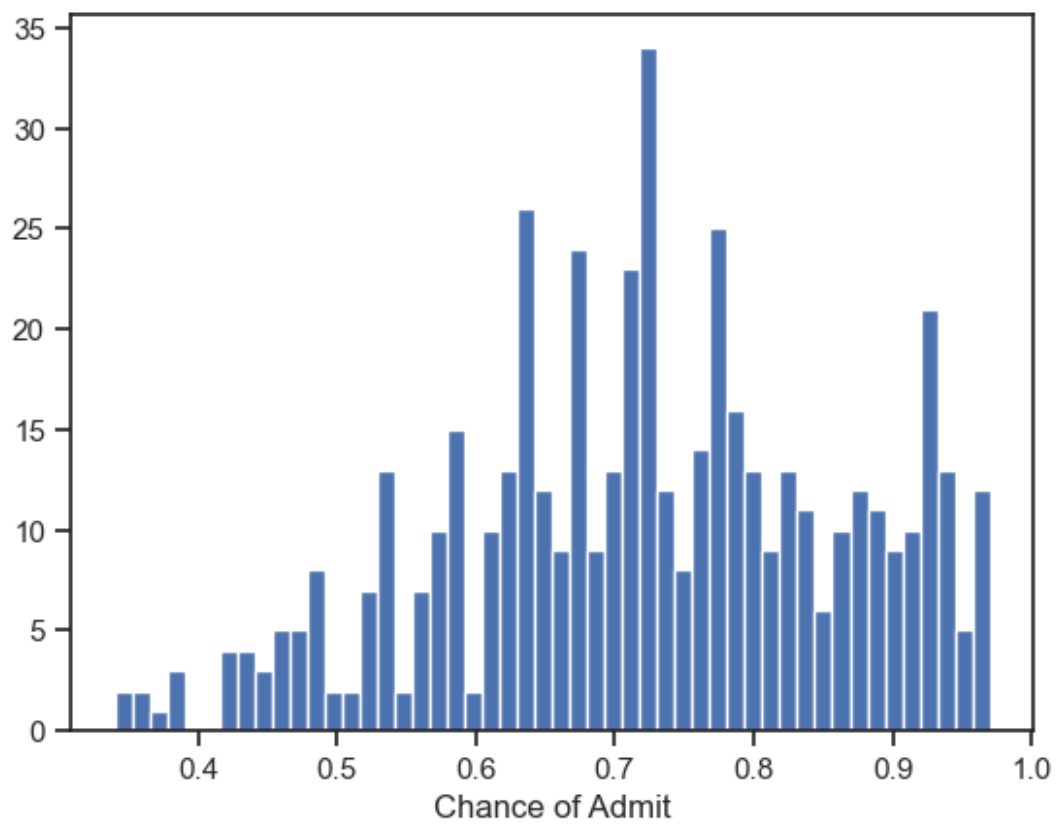
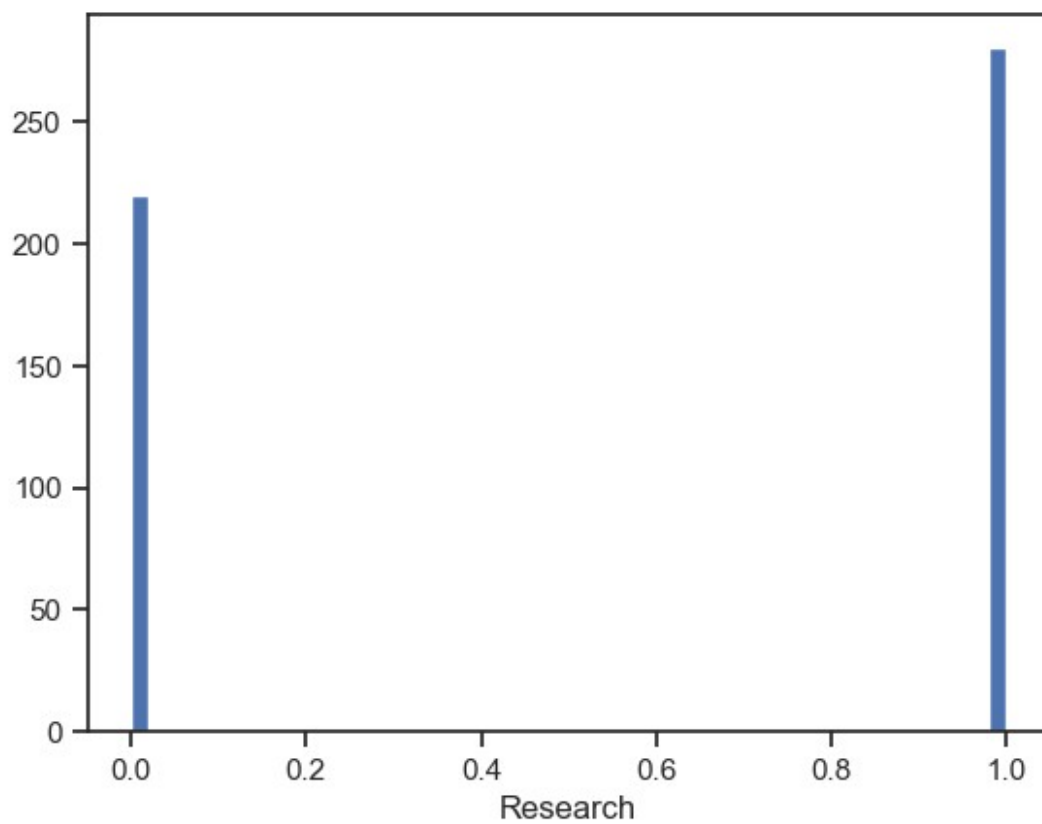
```
df_new = df.drop(columns='Serial No.')
```

```
for column in df_new:
    plt.hist(df_new[column], 50)
    plt.xlabel(column)
    plt.show()
```







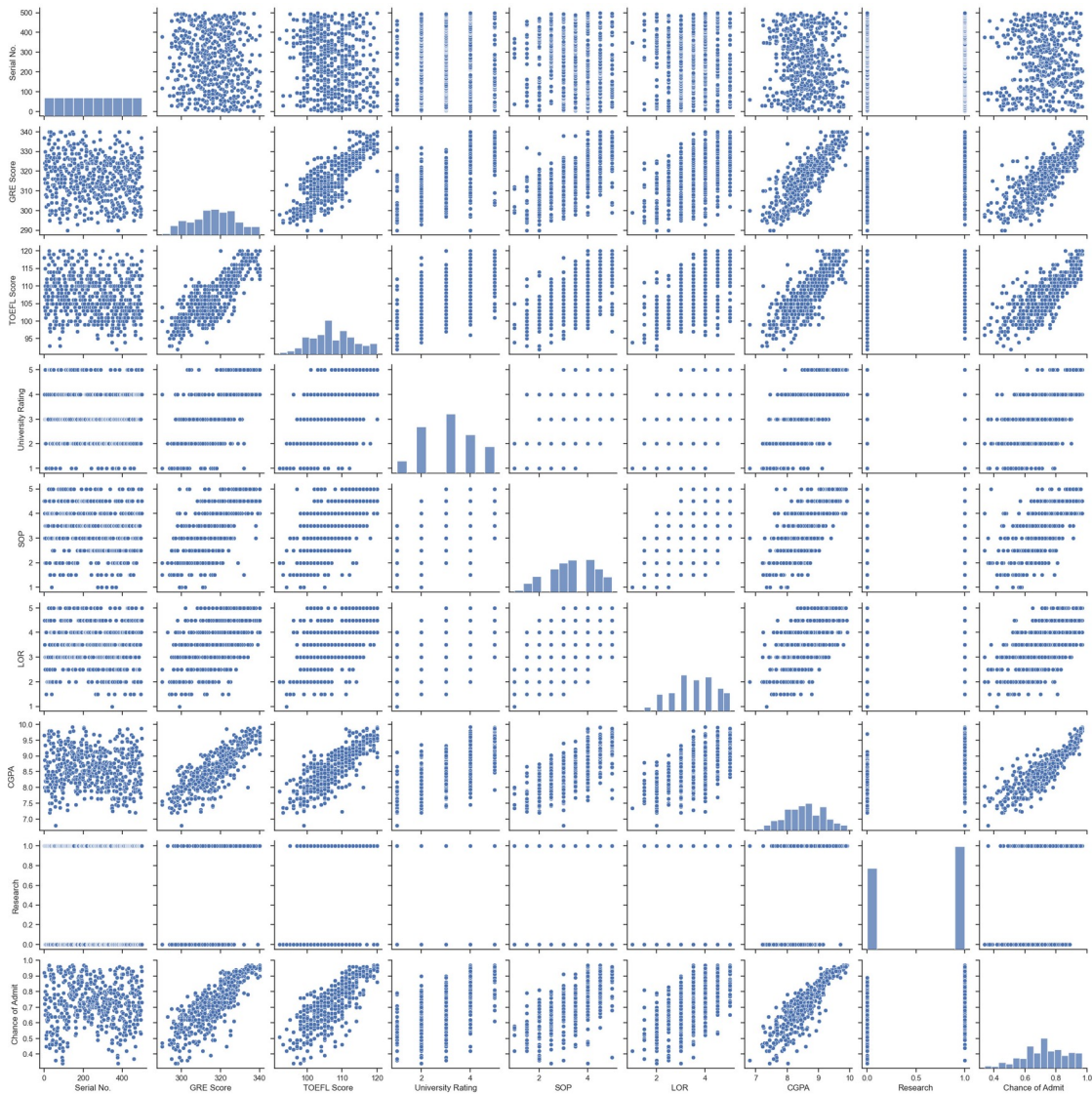


Парный анализ

Все данные в наборе являются числовыми, поэтому в подготовке для парного анализа не нуждаются

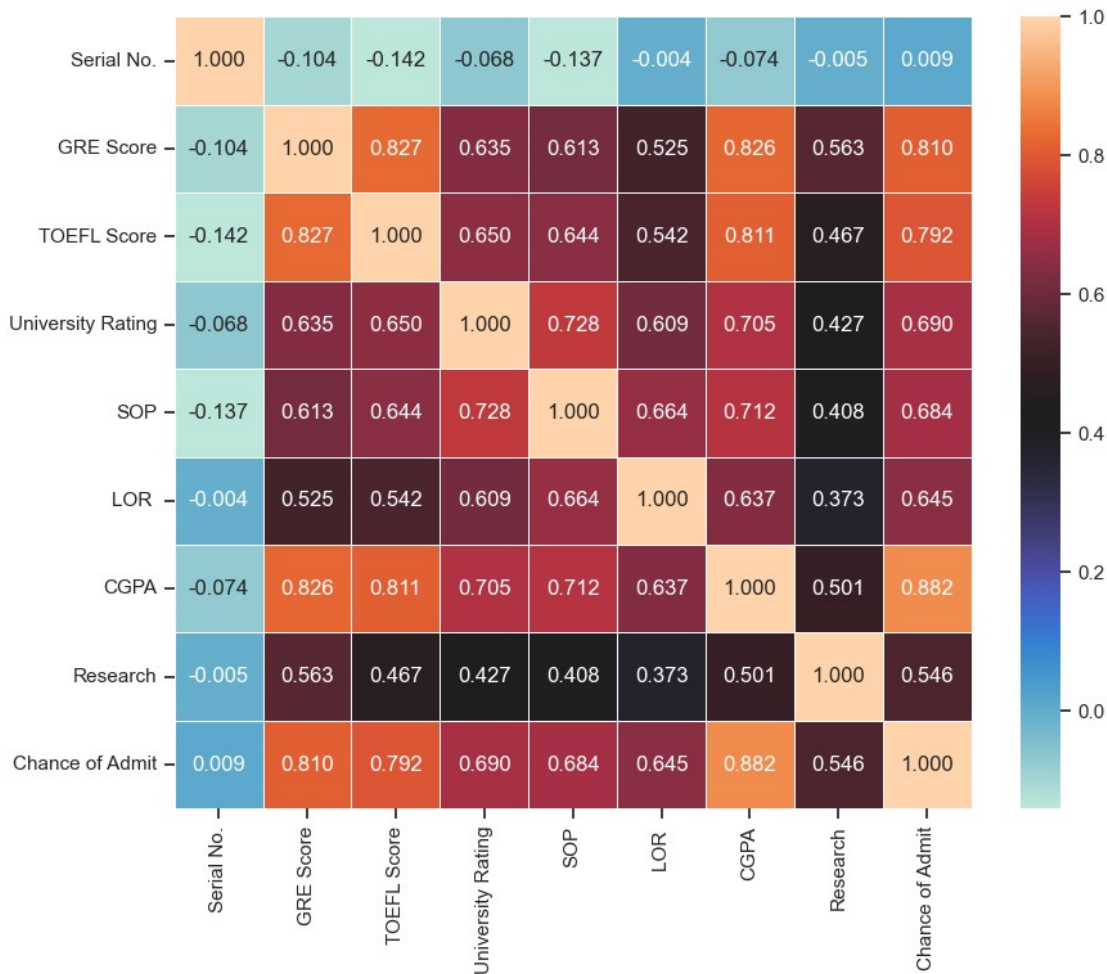
```
sns.pairplot(df)
```

```
<seaborn.axisgrid.PairGrid at 0x7fcb27593940>
```



Тепловая карта

```
f,ax = plt.subplots(figsize=(10, 8))
sns.heatmap(df.corr(), annot=True, linewidths=.5, fmt=
'.3f',ax=ax,cmap="icefire");
```

По тепловой карте видно, что такие признаки, как CGPA, Баллы GRE и TOEFL сильно коррелируют с шансом поступления. На основании этого можно сделать модель предсказания шанса поступления судя по этим параметрам.

Для этого можно использовать классификацию (или регрессию) на основе метода k-ближайших соседей (также можно использовать SVM и дерево решений).

Найдем наилучшее количество соседей для метода k-ближайших соседей

```
data_train, data_test, train_target, test_target =
train_test_split(df, df['Chance of Admit'], test_size=0.1,
random_state=42)
dtest = data_test[['GRE Score', 'University Rating', 'Chance of
Admit']]
```

```
lab = preprocessing.LabelEncoder()
test_target_transformed = lab.fit_transform(test_target)
```

```
knn = KNeighborsClassifier()
parameters = {'n_neighbors': range(1, 10)}
grid = GridSearchCV(knn, parameters, cv=4, scoring='f1')
grid.fit(dtest, test_target_transformed)
grid.best_params_

{'n_neighbors': 1}
```