



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Системы обработки информации и управления (ИУ5)

ОТЧЕТ по лабораторной работе

«Разведочный анализ данных. Исследование и визуализация данных»

ДИСЦИПЛИНА: «Технологии машинного обучения»

Выполнил: студент гр. ИУ5-62Б _____ (Кудрявцев С.Д.)
(Подпись) (Ф.И.О.)

Проверил: _____ (Гапанюк Ю.Е.)
(Подпись) (Ф.И.О.)

2020 г.

▼ Лабораторная работа №1

▼ 1) Текстовое описание набора данных

В качестве набора данных мы будем использовать набор данных Diabets dataset <https://sci-datasets> Для каждого из $n = 442$ больных сахарным диабетом были получены десять исход массы тела, среднее артериальное давление и шесть измерений сыворотки крови, а также количественная мера прогрессирования заболевания через год после исходного уровня.

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")

data = pd.read_csv('data/diabetes.tab.txt', sep="\t")
```

▼ 2) Основные характеристики датасета

```
# Первые 5 строк датасета
data.head()
```

	AGE	SEX	BMI	BP	S1	S2	S3	S4	S5	S6	Y
0	59	2	32.1	101.0	157	93.2	38.0	4.0	4.8598	87	151
1	48	1	21.6	87.0	183	103.2	70.0	3.0	3.8918	69	75
2	72	2	30.5	93.0	156	93.6	41.0	4.0	4.6728	85	141
3	24	1	25.3	84.0	198	131.4	40.0	5.0	4.8903	89	206
4	50	1	23.0	101.0	192	125.4	52.0	4.0	4.2905	80	135

```
# Размер датасета - 442 строки, 11 колонок
data.shape
```

 (442, 11)

```
total_count = data.shape[0]
print('Всего строк: {}'.format(total_count))
```

 Всего строк: 442

```
# Список колонок
data.columns
```

```
Index(['AGE', 'SEX', 'BMI', 'BP', 'S1', 'S2', 'S3', 'S4', 'S5', 'S6', 'Y'], dt
```

```
# Список колонок с типами данных
data.dtypes
```

```
AGE      int64
SEX      int64
BMI      float64
BP      float64
S1      int64
S2      float64
S3      float64
S4      float64
S5      float64
S6      int64
Y      int64
dtype: object
```

```
# Проверим наличие пустых значений
# Цикл по колонкам датасета
for col in data.columns:
    # Количество пустых значений - все значения заполнены
    temp_null_count = data[data[col].isnull()].shape[0]
    print('{} - {}'.format(col, temp_null_count))
```

```
AGE - 0
SEX - 0
BMI - 0
BP - 0
S1 - 0
S2 - 0
S3 - 0
S4 - 0
S5 - 0
S6 - 0
Y - 0
```


```
# Основные статистические характеристики набора данных
data.describe()
```



	AGE	SEX	BMI	BP	S1	S2	S
count	442.000000	442.000000	442.000000	442.000000	442.000000	442.000000	442.000000
mean	48.518100	1.468326	26.375792	94.647014	189.140271	115.439140	49.78846
std	13.109028	0.499561	4.418122	13.831283	34.608052	30.413081	12.93420

min	19.000000	1.000000	18.000000	62.000000	97.000000	41.600000	22.00000
25%	38.250000	1.000000	23.200000	84.000000	164.250000	96.050000	40.25000

```
# Определим уникальные значения для целевого признака
data['SEX'].unique()
```


 array([2, 1], dtype=int64)

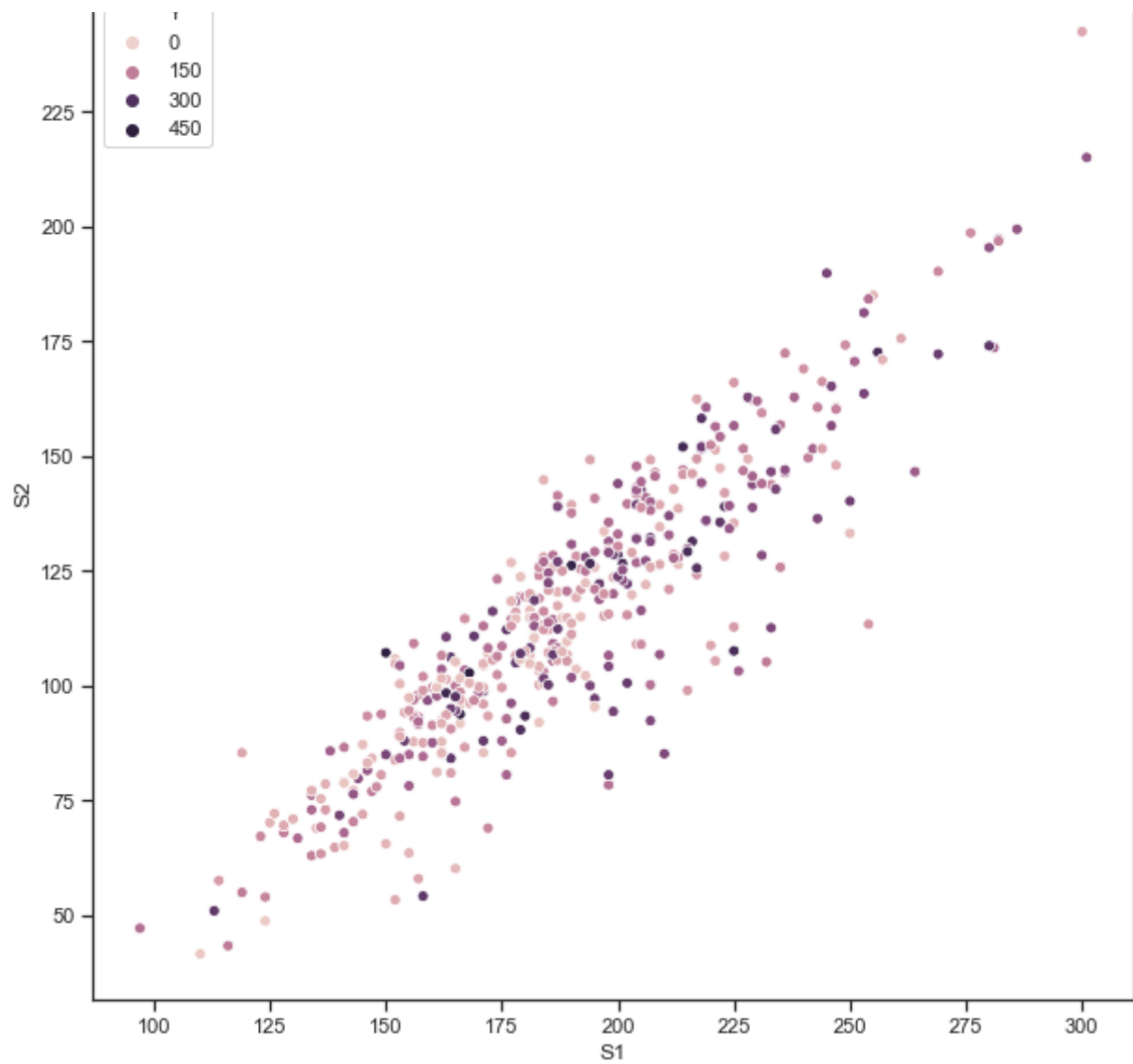
▼ 3) Визуальное исследование датасета

```
fig, ax = plt.subplots(figsize=(10,10))
sns.scatterplot(ax=ax, x='S1', y='S2', data=data)
```



```
<matplotlib.axes._subplots.AxesSubplot at 0xe70c610>
fig, ax = plt.subplots(figsize=(10,10))
sns.scatterplot(ax=ax, x='S1', y='S2', data=data, hue='Y')
```

 <matplotlib.axes._subplots.AxesSubplot at 0xfd81e70>



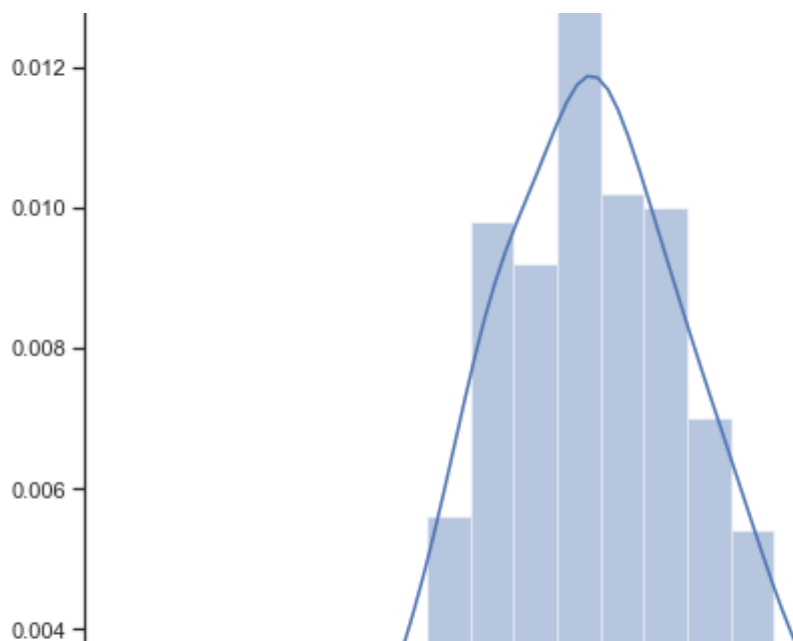
▼ Гистограмма

```
fig, ax = plt.subplots(figsize=(10,10))
sns.distplot(data['S1'])
```



<matplotlib.axes._subplots.AxesSubplot at 0xfd816b0>





► Jointplot

Комбинация гистограмм и диаграмм рассеивания.

↳ 3 cells hidden

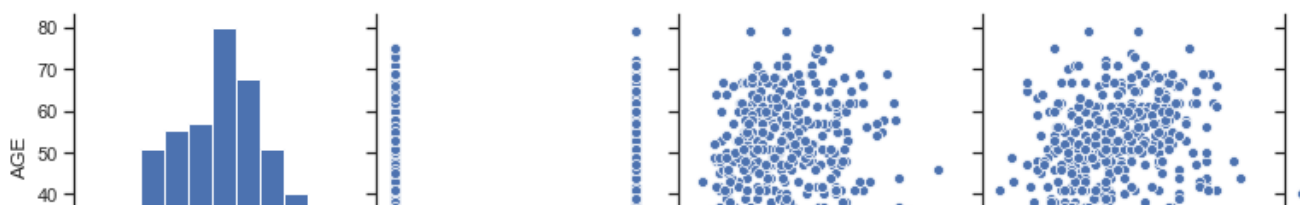


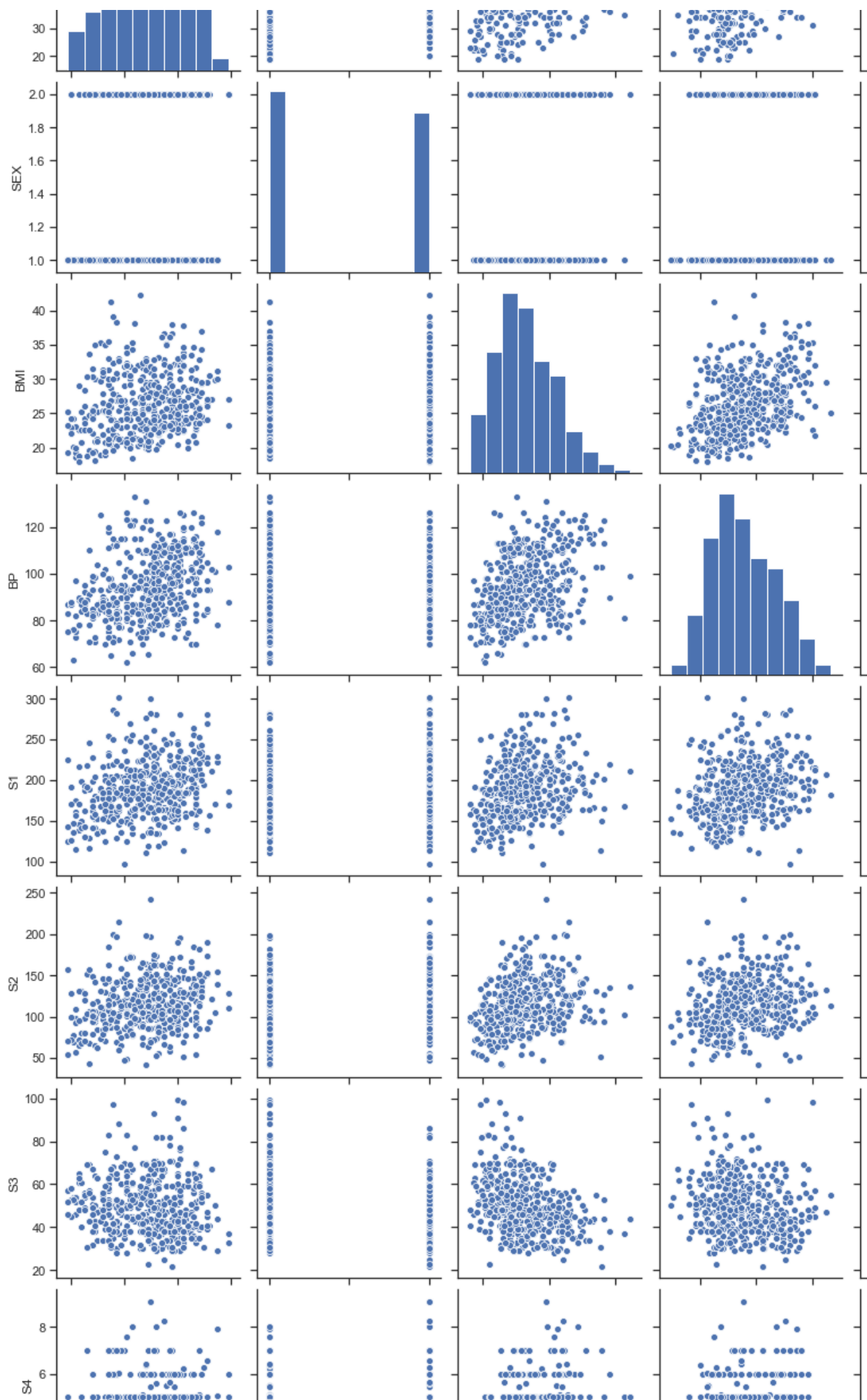
▼ "Парные диаграммы"

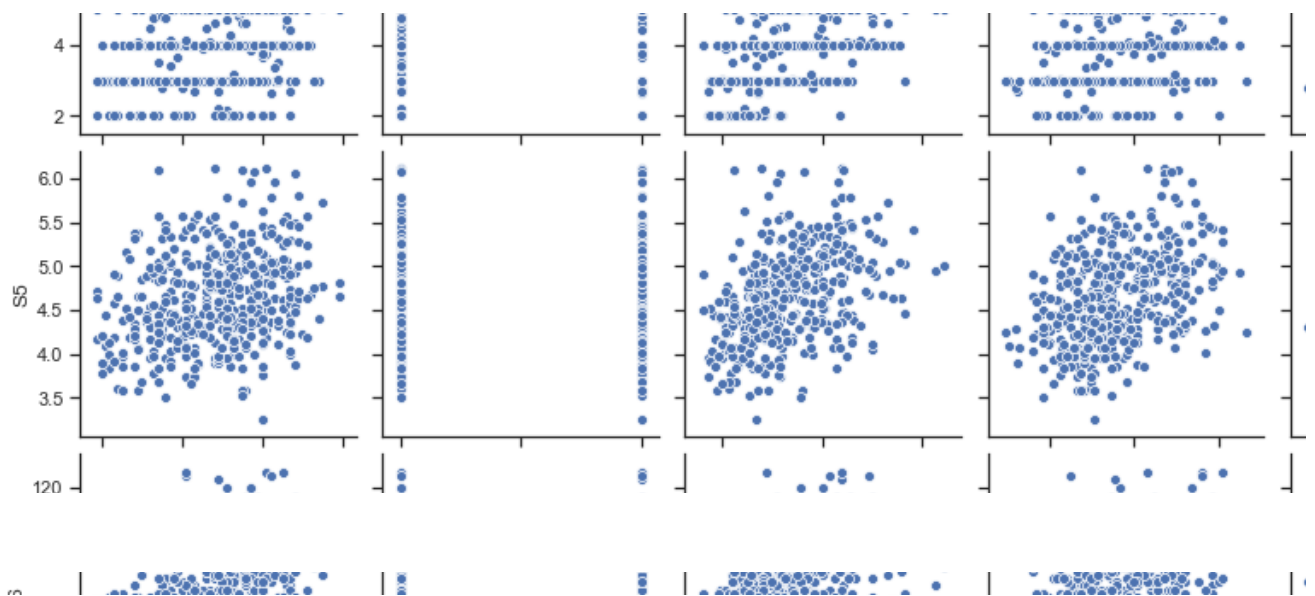
`sns.pairplot(data)`



<seaborn.axisgrid.PairGrid at 0x207eb810>







► Ящик с усами

Отображает одномерное распределение вероятности.



► Violin plot

Похоже на предыдущую диаграмму, но по краям отображаются распределения плотности



▼ 4) Информация о корреляции признаков

`data.corr()`



	AGE	SEX	BMI	BP	S1	S2	S3	S4
AGE	1.000000	0.173737	0.185085	0.335428	0.260061	0.219243	-0.075181	0.203841
SEX	0.173737	1.000000	0.088161	0.241010	0.035277	0.142637	-0.379090	0.332115
BMI	0.185085	0.088161	1.000000	0.395411	0.249777	0.261170	-0.366811	0.413807

BP 0.335428 0.241010 0.395411 1.000000 0.242464 0.185548 -0.178762 0.257650

```
data.corr(method='pearson')
```



	AGE	SEX	BMI	BP	S1	S2	S3	S4
AGE	1.000000	0.173737	0.185085	0.335428	0.260061	0.219243	-0.075181	0.203841
SEX	0.173737	1.000000	0.088161	0.241010	0.035277	0.142637	-0.379090	0.332115
BMI	0.185085	0.088161	1.000000	0.395411	0.249777	0.261170	-0.366811	0.413807
BP	0.335428	0.241010	0.395411	1.000000	0.242464	0.185548	-0.178762	0.257650
S1	0.260061	0.035277	0.249777	0.242464	1.000000	0.896663	0.051519	0.542207
S2	0.219243	0.142637	0.261170	0.185548	0.896663	1.000000	-0.196455	0.659817
S3	-0.075181	-0.379090	-0.366811	-0.178762	0.051519	-0.196455	1.000000	-0.738493
S4	0.203841	0.332115	0.413807	0.257650	0.542207	0.659817	-0.738493	1.000000
S5	0.270774	0.149916	0.446157	0.393480	0.515503	0.318357	-0.398577	0.617859
S6	0.301731	0.208133	0.388680	0.390430	0.325717	0.290600	-0.273697	0.417212
Y	0.187889	0.043062	0.586450	0.441482	0.212022	0.174054	-0.394789	0.430453

```
data.corr(method='kendall')
```



	AGE	SEX	BMI	BP	S1	S2	S3	S4
AGE	1.000000	0.146580	0.136535	0.242111	0.182220	0.153612	-0.073846	0.160898
SEX	0.146580	1.000000	0.080424	0.215733	0.022809	0.110208	-0.326188	0.297335
BMI	0.136535	0.080424	1.000000	0.281770	0.194171	0.198583	-0.249831	0.335625
BP	0.242111	0.215733	0.281770	1.000000	0.188067	0.140253	-0.131014	0.205948
S1	0.182220	0.022809	0.194171	0.188067	1.000000	0.717229	0.010695	0.393367
S2	0.153612	0.110208	0.198583	0.140253	0.717229	1.000000	-0.133332	0.503579
S3	-0.073846	-0.326188	-0.249831	-0.131014	0.010695	-0.133332	1.000000	-0.638633
S4	0.160898	0.297335	0.335625	0.205948	0.393367	0.503579	-0.638633	1.000000
S5	0.180544	0.143172	0.344720	0.268863	0.356268	0.242250	-0.311775	0.485410
S6	0.201784	0.168199	0.266373	0.264566	0.227139	0.194082	-0.200545	0.307397
Y	0.130709	0.030630	0.391195	0.289352	0.154016	0.129665	-0.278884	0.324734

```
data.corr(method='spearman')
```



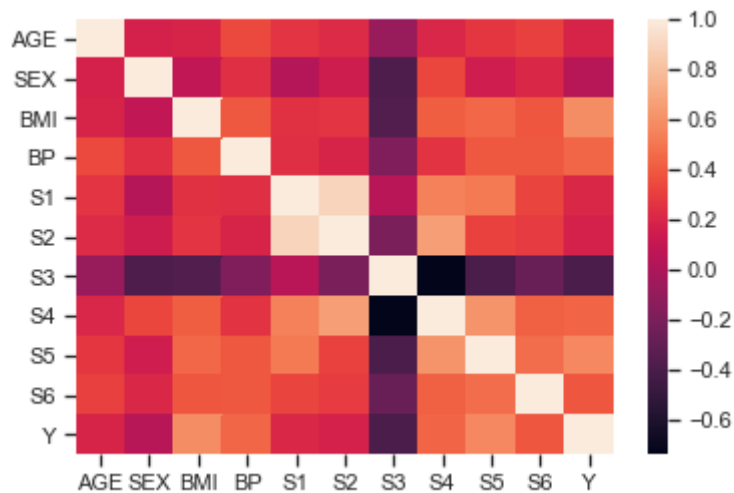
	AGE	SEX	BMI	BP	S1	S2	S3	S4
AGE	1.000000	0.177463	0.200554	0.350859	0.262524	0.221711	-0.106973	0.221017
SEX	0.177463	1.000000	0.098079	0.261508	0.027790	0.134695	-0.394584	0.337524

SEX	0.0271700	1.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
BMI	0.200554	0.098079	1.000000	0.397985	0.287829	0.295494	-0.371172	0.459068	0.280799
BP	0.350859	0.261508	0.397985	1.000000	0.275224	0.205638	-0.191033	0.280799	0.280799
S1	0.262524	0.027790	0.287829	0.275224	1.000000	0.878793	0.015308	0.520674	0.520674
S2	0.221711	0.134695	0.295494	0.205638	0.878793	1.000000	-0.197435	0.652283	0.652283
S3	-0.106973	-0.394584	-0.371172	-0.191033	0.015308	-0.197435	1.000000	-0.789694	-0.789694
S4	0.221017	0.337524	0.459068	0.280799	0.520674	0.652283	-0.789694	1.000000	1.000000
S5	0.265176	0.174625	0.491609	0.396071	0.512864	0.349947	-0.450420	0.640390	0.640390
S6	0.296235	0.203277	0.384664	0.381219	0.332173	0.286483	-0.290863	0.413700	0.413700
Y	0.197822	0.037401	0.561382	0.416241	0.232429	0.195834	-0.410022	0.448931	0.448931

```
sns.heatmap(data.corr())
```



```
<matplotlib.axes._subplots.AxesSubplot at 0x31bc55d0>
```

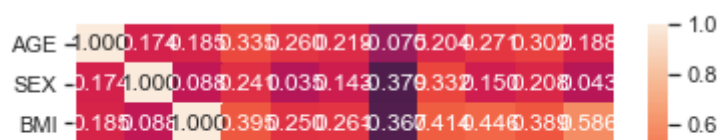


```
# Вывод значений в ячейках
```

```
sns.heatmap(data.corr(), annot=True, fmt='.3f')
```




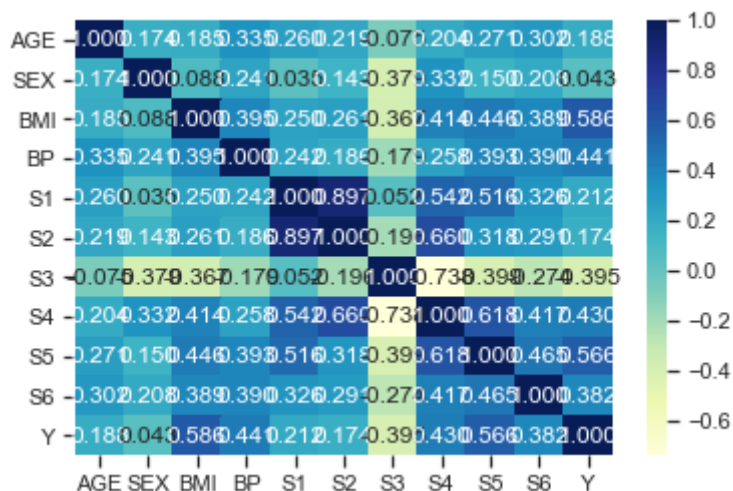
```
<matplotlib.axes._subplots.AxesSubplot at 0x2ccbdf70>
```




```
# Изменение цветовой гаммы
```

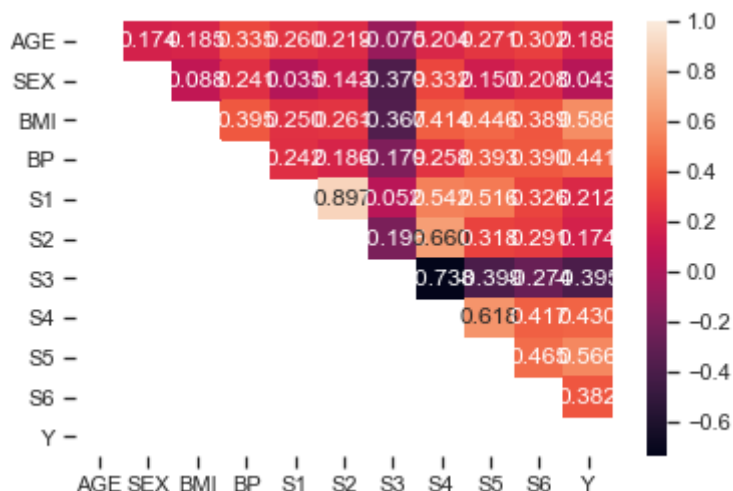
```
sns.heatmap(data.corr(), cmap='YlGnBu', annot=True, fmt='.3f')
```

 <matplotlib.axes._subplots.AxesSubplot at 0x31c7b790>



```
# Треугольный вариант матрицы
mask = np.zeros_like(data.corr(), dtype=np.bool)
# чтобы оставить нижнюю часть матрицы
# mask[np.triu_indices_from(mask)] = True
# чтобы оставить верхнюю часть матрицы
mask[np.tril_indices_from(mask)] = True
sns.heatmap(data.corr(), mask=mask, annot=True, fmt='.3f')
```

 <matplotlib.axes._subplots.AxesSubplot at 0x31ae5f10>



```
fig, ax = plt.subplots(1, 3, sharex='col', sharey='row', figsize=(15,5))
sns.heatmap(data.corr(method='pearson'), ax=ax[0], annot=True, fmt='.2f')
sns.heatmap(data.corr(method='kendall'), ax=ax[1], annot=True, fmt='.2f')
sns.heatmap(data.corr(method='spearman'), ax=ax[2], annot=True, fmt='.2f')
fig.suptitle('Корреляционные матрицы, построенные различными методами')
ax[0].title.set_text('Pearson')
ax[1].title.set_text('Kendall')
ax[2].title.set_text('Spearman')
```



Корреляционные матрицы, построенные различными методами



