

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №1
по дисциплине «Машинное обучение»
Тема: Предобработка данных

Студент гр. 8303

Преподаватель

Гришин К. И.

Жангиров Т.Р.

Санкт-Петербург

2021

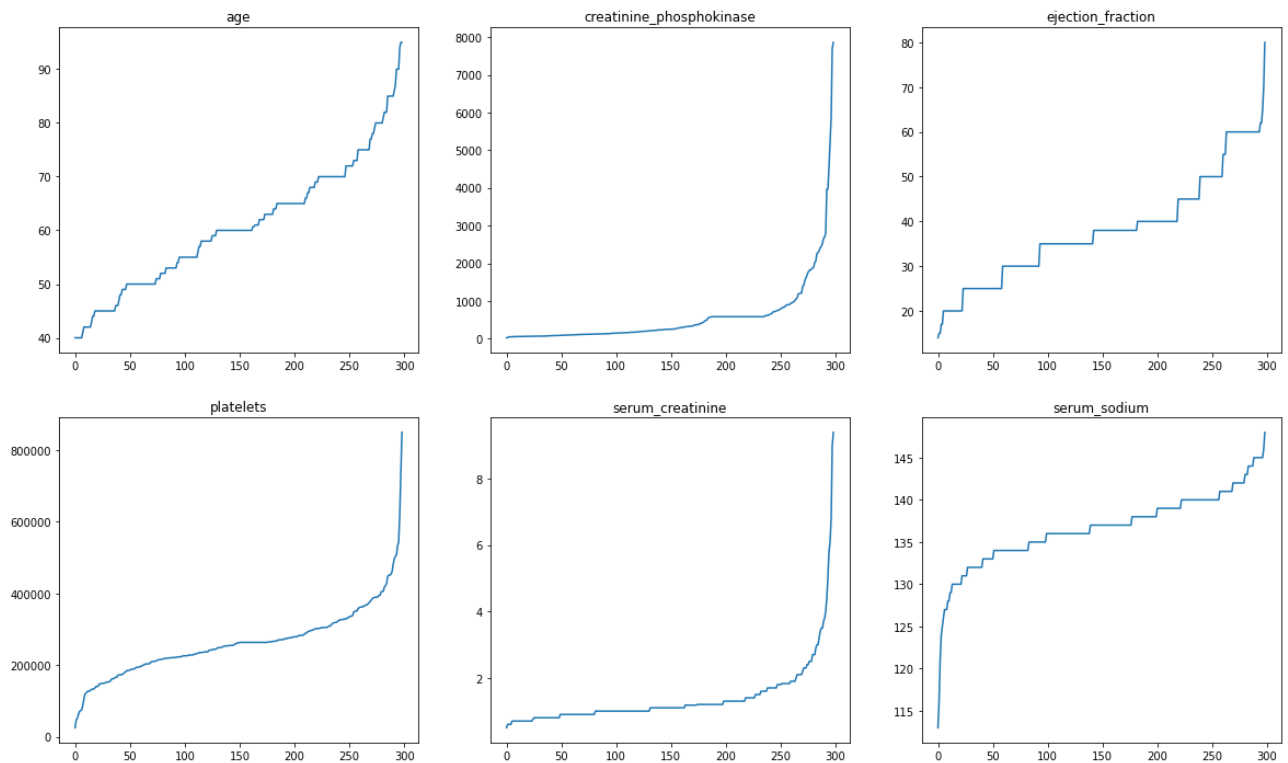
Цель работы

Ознакомиться с методами предобработки данных из библиотеки *Scikit Learn*.

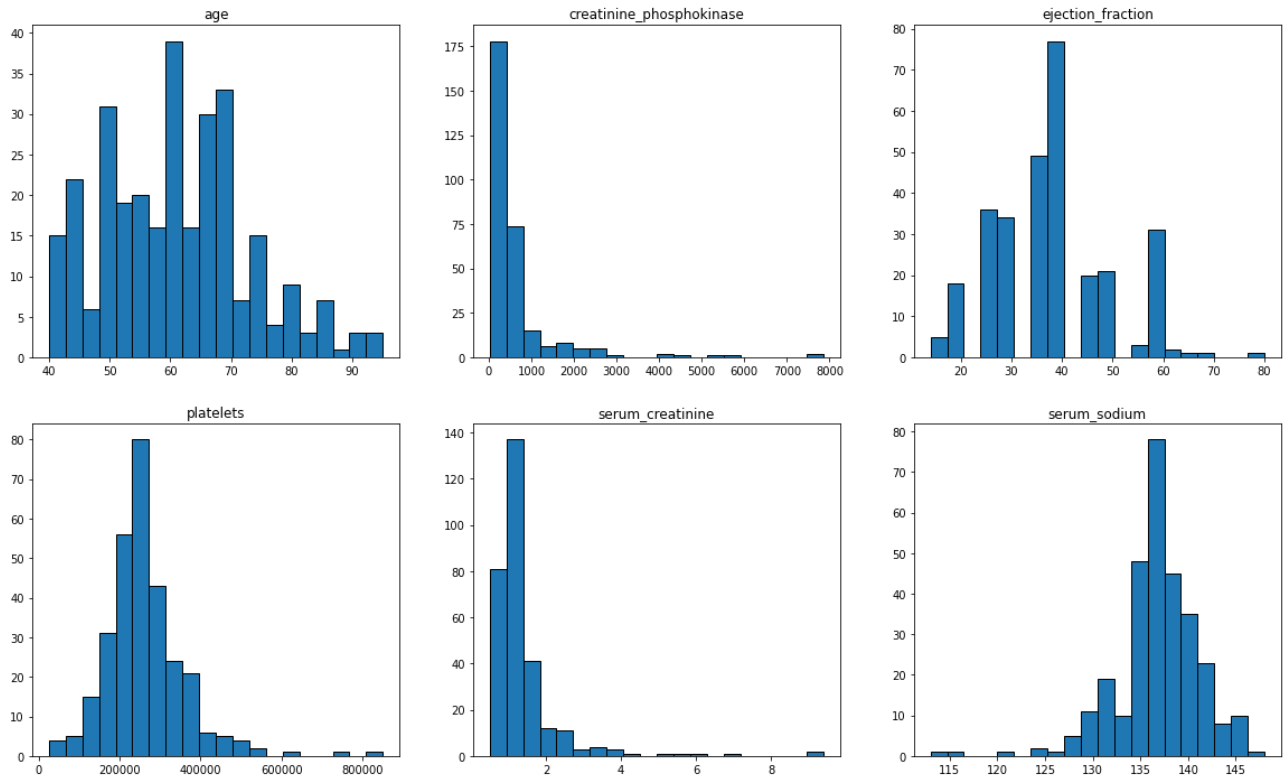
Ход выполнения работы

Загрузка данных

1. Загрузить датасет (<https://www.kaggle.com/andrewmvd/heart-failure-clinical-data>).
2. Создать Python скрипт, загружающий датасет и исключающий нечисловые признаки (отсортированные данные приведены на рис. 1).



3. Построить гистограммы признаков (рис. 2).



4. На основании гистограмм определите диапазоны значений для каждого из признаков, а также возле какого значения лежит наибольшее количество наблюдений (табл. 1).

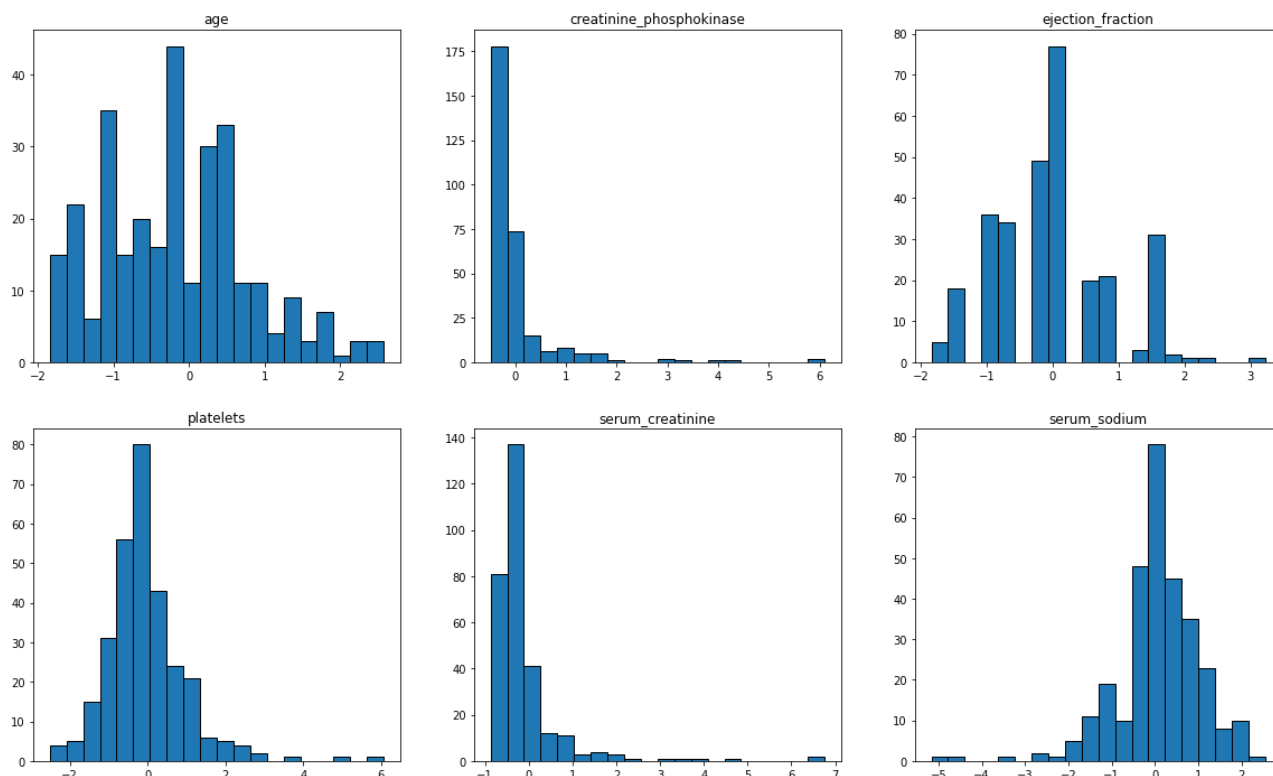
| | Диапазон | Наибольшее количество наблюдений |
|--------------------------|-----------------|----------------------------------|
| age | (40, 95) | 60 |
| creatinine_phosphokinase | (0, 7900) | 0 |
| ejection_fraction | (14, 80) | 40 |
| platelets | (25000, 850000) | 225000 |
| serum_creatinine | (0.5, 9.2) | 1 |
| serum_sodium | (113, 148) | 136 |

5. Преобразовать *Pandas* датафрейм в массив *NumPy*.

Стандартизация данных

1. Подключить модуль *Sklearn* и настроить стандартизацию по первым 150 наблюдениям используя *StandardScaler*.
2. Стандартизировать данные.

3. Построить гистограммы признаков (рис. 3).



4. Сравнить данные до и после стандартизации.

На основе полученных гистограмм составлена таблица с диапазонами наблюдений и значениями, около которых лежит наибольшее число наблюдений (табл. 2).

| | Диапазон | Наибольшее количество наблюдений |
|--------------------------|--------------|----------------------------------|
| age | (-1.75, 2.6) | -0.2 |
| creatinine_phosphokinase | (-0.5, 6.1) | -0.4 |
| ejection_fraction | (-2.2, 3.2) | 0.1 |
| platelets | (-2.5, 6) | -0.2 |
| serum_creatinine | (-0.9, 6.6) | 0.1 |
| serum_sodium | (-5.1, 2.5) | 0 |

Из табл. 2 видно, что наибольшее количество наблюдений находится возле 0, а сам разброс значений находится в пределах одного порядка.

5. Рассчитать мат. ожидание и СКО до и после стандартизации. Вывести формулы, по которым они стандартизировались (табл. 3).

| | μ до | σ до | μ после | σ после |
|------------------------|------------|-----------|---------|---------|
| age | 60.834 | 11.895 | -0.170 | 0.955 |
| creatinе_phosphokinase | 581.839 | 970.288 | -0.021 | 0.816 |
| ejection_fraction | 38.084 | 11.835 | 0.011 | 0.908 |
| platelets | 263358.029 | 97804.237 | -0.035 | 1.017 |
| serum_creatine | 1.394 | 1.035 | -0.109 | 0.887 |
| serum_sodium | 136.625 | 4.412 | 0.038 | 0.972 |

Исходя из таблицы 3, можно сделать вывод, что стандартизация данных приводит к тому, что математическое ожидание становится равным нулю, а среднеквадратичное отклонение равным 1. Из этого следует, что формула преобразования данных принимает вид:

$$y = \frac{x - \mu(X)}{\sigma(X)}$$

где X – исходный набор данных.

6. Сравнение со значениями объекта настроенной стандартизации (табл. 4).

| | mean_ | var_ | σ |
|------------------------|----------|--------------|----------|
| Age | 62.95 | 155.00 | 12.45 |
| creatinе_phosphokinase | 607.15 | 1415488.82 | 1189.74 |
| ejection_fraction | 37.95 | 170.02 | 13.04 |
| Platelets | 266746.8 | 9.252860e+09 | 96191.79 |
| serum_creatine | 1.52 | 1.36 | 1.17 |
| serum_sodium | 136.45 | 20.61 | 4.54 |

Из таблицы видно, что значения математического ожидания и дисперсии близки к соответствующим значениям исходного набора данных, однако все же отличаются. Из-за этих отличий и наблюдается математическое ожидание и среднеквадратичное отклонение стандартизованного набора данных от 0 и 1 соответственно.

7. Настройка стандартизации на всех данных. Сравнение со стандартизацией для первых 150 наблюдений.

Сравнение исходного набора данных со стандартизированным приведено в таблице 5. Значения объекта настроенной стандартизации приведены в таблице 6.

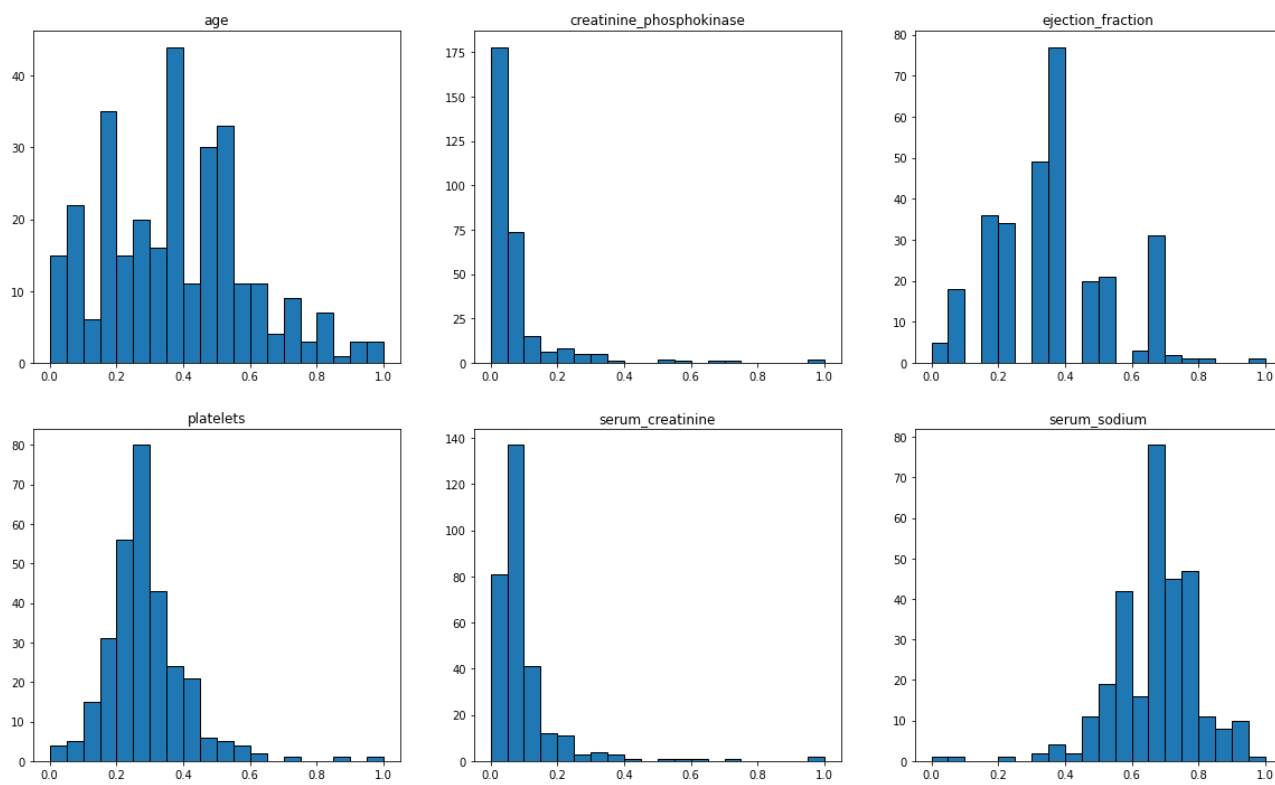
| | μ до | σ до | μ после | σ после |
|------------------------|------------|-------------|-------------|----------------|
| age | 60.834 | 11.895 | 0 | 1.002 |
| creatine_phosphokinase | 581.839 | 970.288 | 0 | 1.002 |
| ejection_fraction | 38.084 | 11.835 | 0 | 1.002 |
| platelets | 263358.029 | 97804.237 | 0 | 1.002 |
| serum_creatine | 1.394 | 1.035 | 0 | 1.002 |
| serum_sodium | 136.625 | 4.412 | 0 | 1.002 |

| | mean_ | var_ | σ |
|------------------------|-----------|-------------|----------|
| Age | 60.83 | 141.01 | 11.87 |
| creatine_phosphokinase | 581.84 | 938309.88 | 968.66 |
| ejection_fraction | 38.08 | 139.60 | 11.82 |
| Platelets | 263358.03 | 9.533677e+9 | 97640.55 |
| serum_creatine | 1.39 | 1.07 | 1.03 |
| serum_sodium | 136.63 | 1.03 | 4.41 |

Настройка стандартизации по полной выборке привела к тому, что у всех наборов данных математическое ожидание равно 0, а среднеквадратичное отклонение 1 (несмотря на присутствующую погрешность вычислений, все данные имеют одинаковое значение, а значит – стандартизованы).

Приведение к диапазону

1. Привести данные к диапазону, используя *MinMaxScaler*.
2. Построить гистограммы признаков (рис. 4)

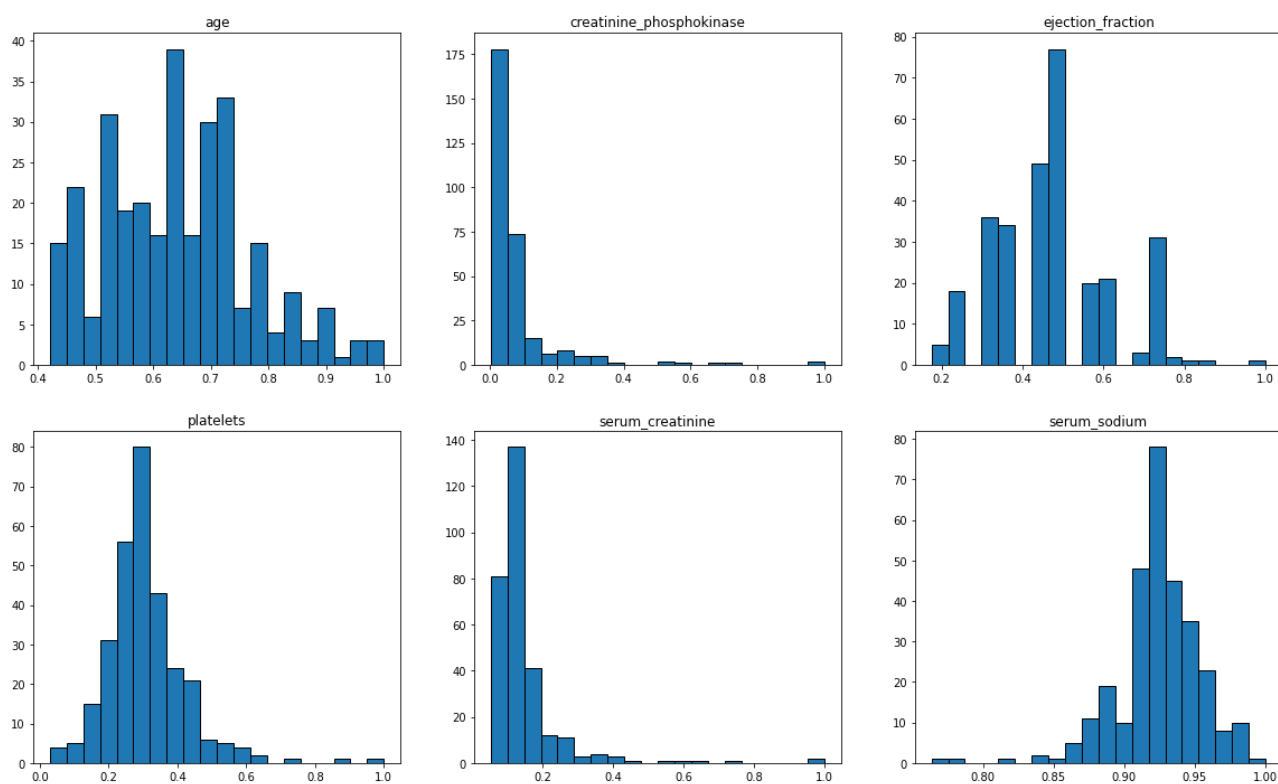


3. Определить минимальное и максимальное значение для каждого признака (табл. 7)

| | min | max |
|--------------------------|-------|--------|
| age | 40 | 95 |
| creatinine_phosphokinase | 23 | 7861 |
| ejection_fraction | 14 | 80 |
| platelets | 25100 | 850000 |
| serum_creatinine | 0.5 | 9.4 |
| serum_sodium | 113 | 148.0 |

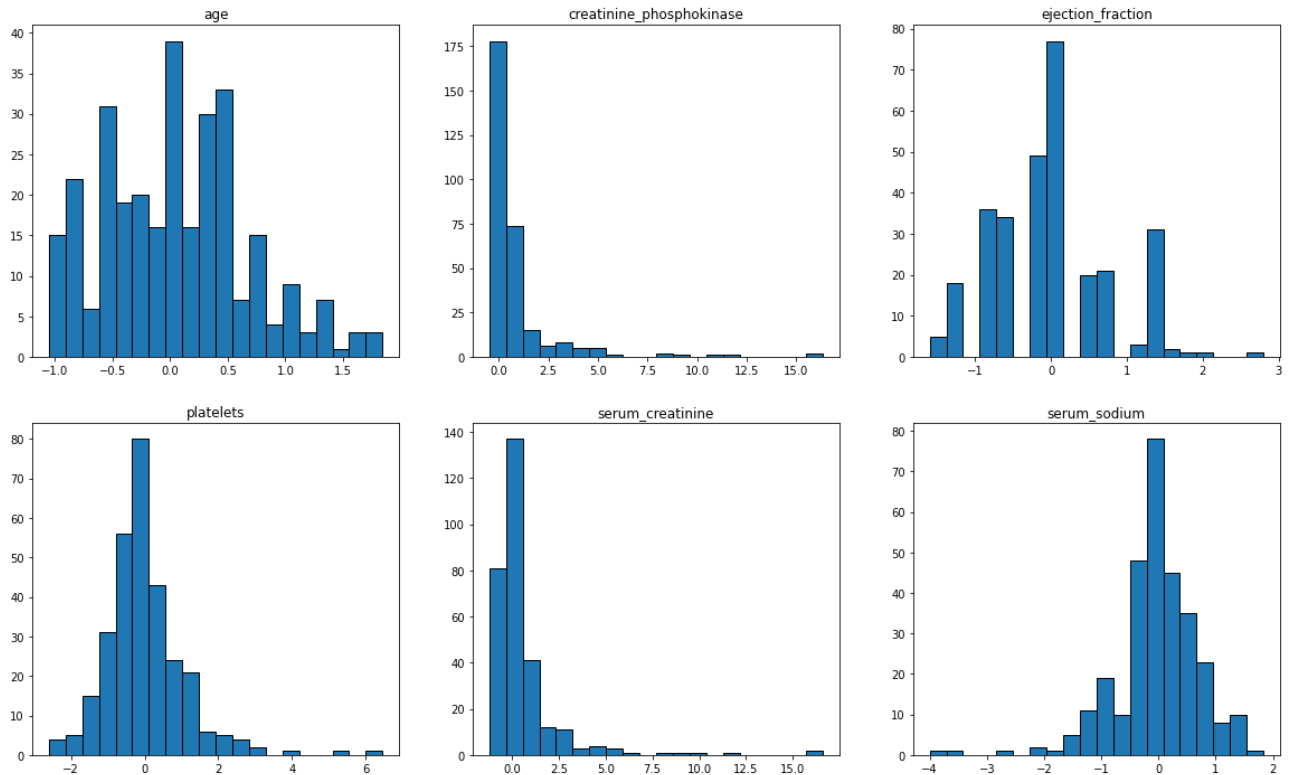
4. Трансформировать данные используя MaxAbsScaler и RobustScaler. Построить гистограммы, определить диапазоны.

Гистограмма трансформации MaxAbsScaler приведена на рисунке 5.



Данные отмасштабированы так, что модуль максимального значения равен единице.

Гистограмма трансформации RobustScaler приведена на рисунке 6.



Данные отмасштабированы следующим образом. Сначала медиана устанавливается в ноль, затем конец первого и начало третьего квантиля становятся минимумом и максимумом соответственно.

5. Написать функцию, которая приведет данные к диапазону [-5, 10].

Для приведения к диапазону [-5, 10] будет использоваться следующая формула:

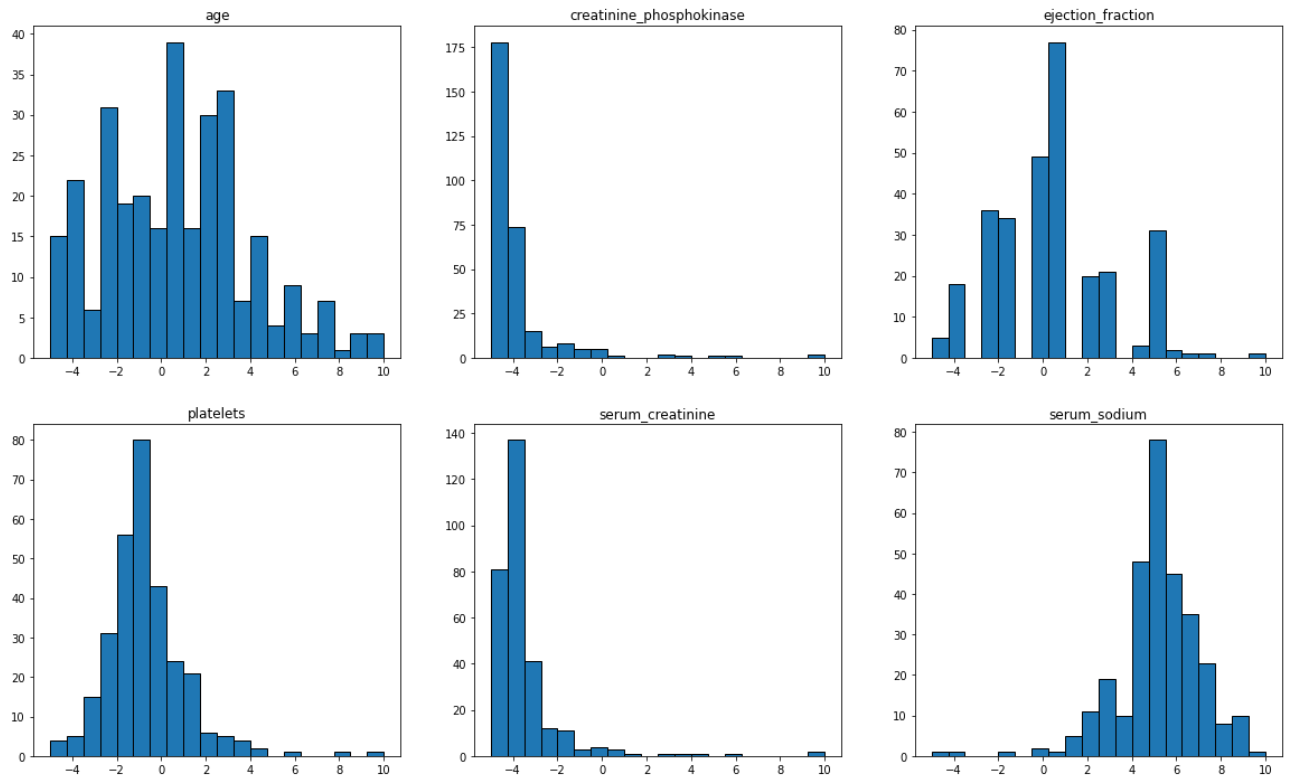
$$y = \frac{x - \min(X)}{\max(X) - \min(X)} (10 + 5) - 5$$

```

1. def min_max_scaler(data, min, max):
2.     d_min, d_max = min, max
3.
4.     def map_range(value, s_min, s_max, d_min, d_max):
5.         return (value - s_min) / (s_max - s_min) * (d_max - d_min) + d_min
6.
7.     def inner_min_max(values):
8.         s_min, s_max = np.min(values), np.max(values)
9.         l = lambda x: map_range(x, s_min, s_max, d_min, d_max)
10.        return np.array(list(map(l, values)))
11.
12.    return np.array(list(map(inner_min_max, data.T))).T

```

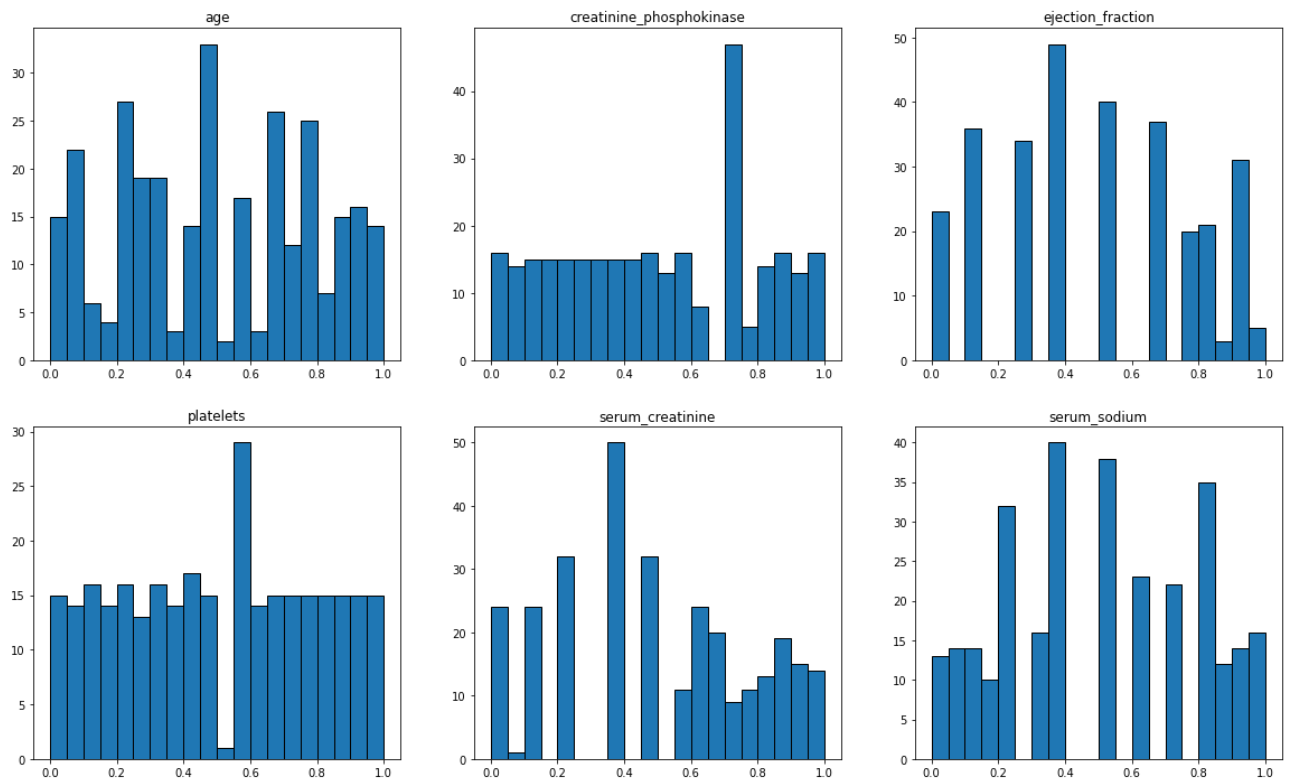
Гистограмма приведена на рисунке 7.



Нелинейные преобразования

1. Привести данные к равномерному распределению, используя *QuantileTransformer*.

2. Построить гистограммы (рис. 8)

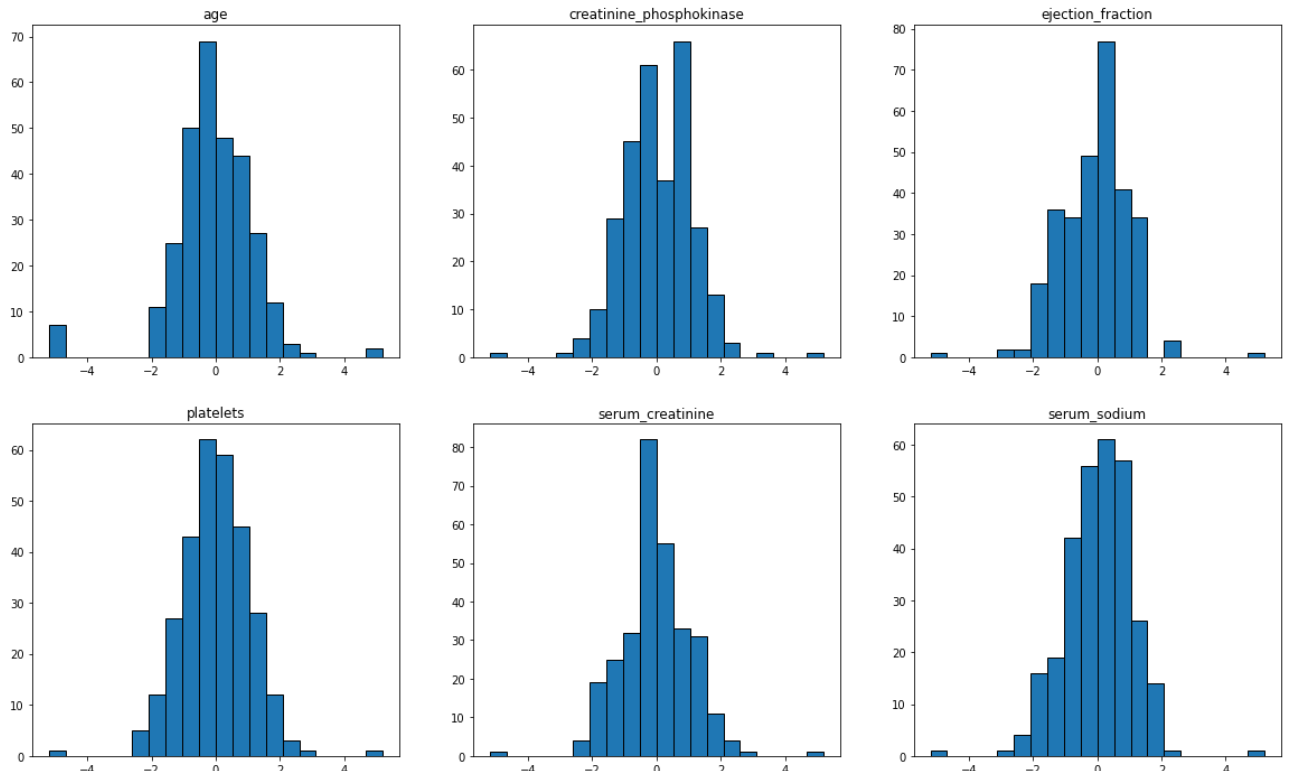


3. Определить влияние параметра n_quantiles

Параметр определяет количество квантилей, которое будет использовано для дискретизации функции распределения.

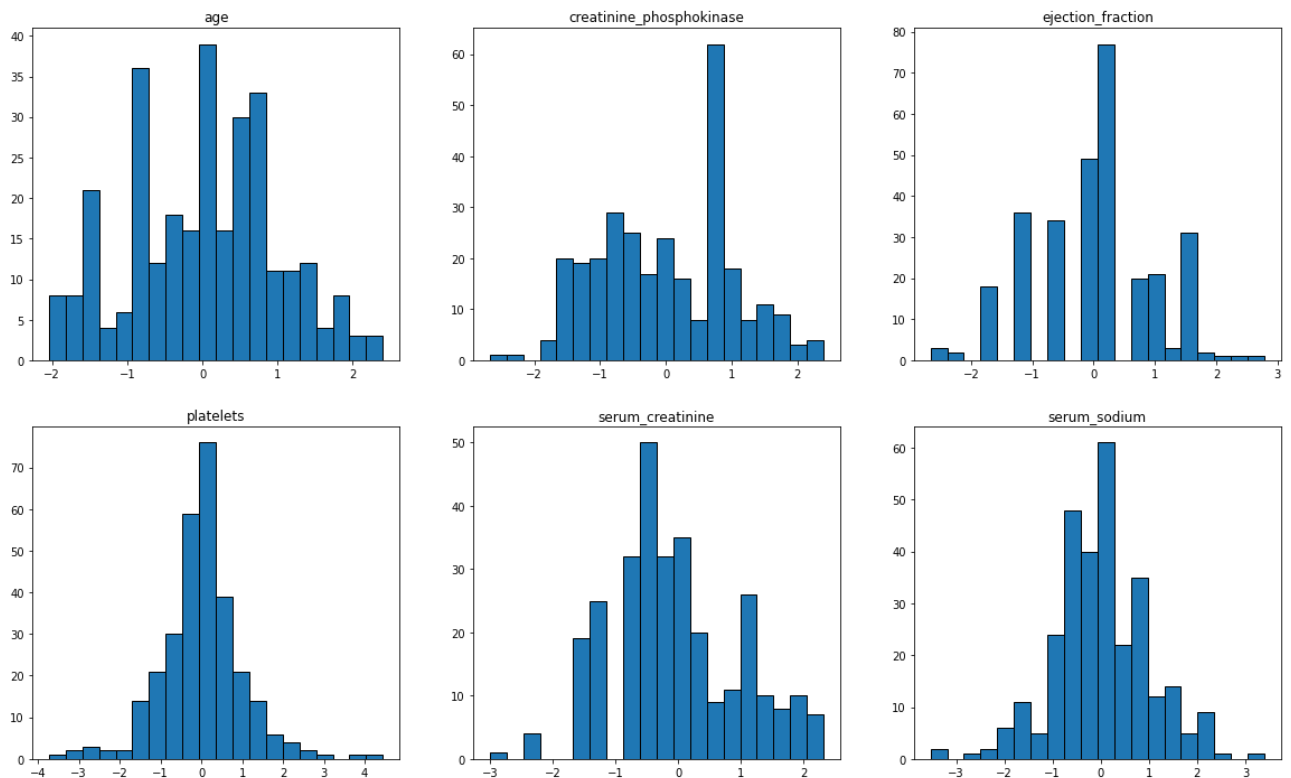
4. Привести данные к нормальному распределению, используя *QuantileTransformer*.

5. Построить гистограммы (рис. 9)



6. Привести данные к нормальному распределению, используя *PowerTransformer*.

Гистограмма приведена на рисунке 10.



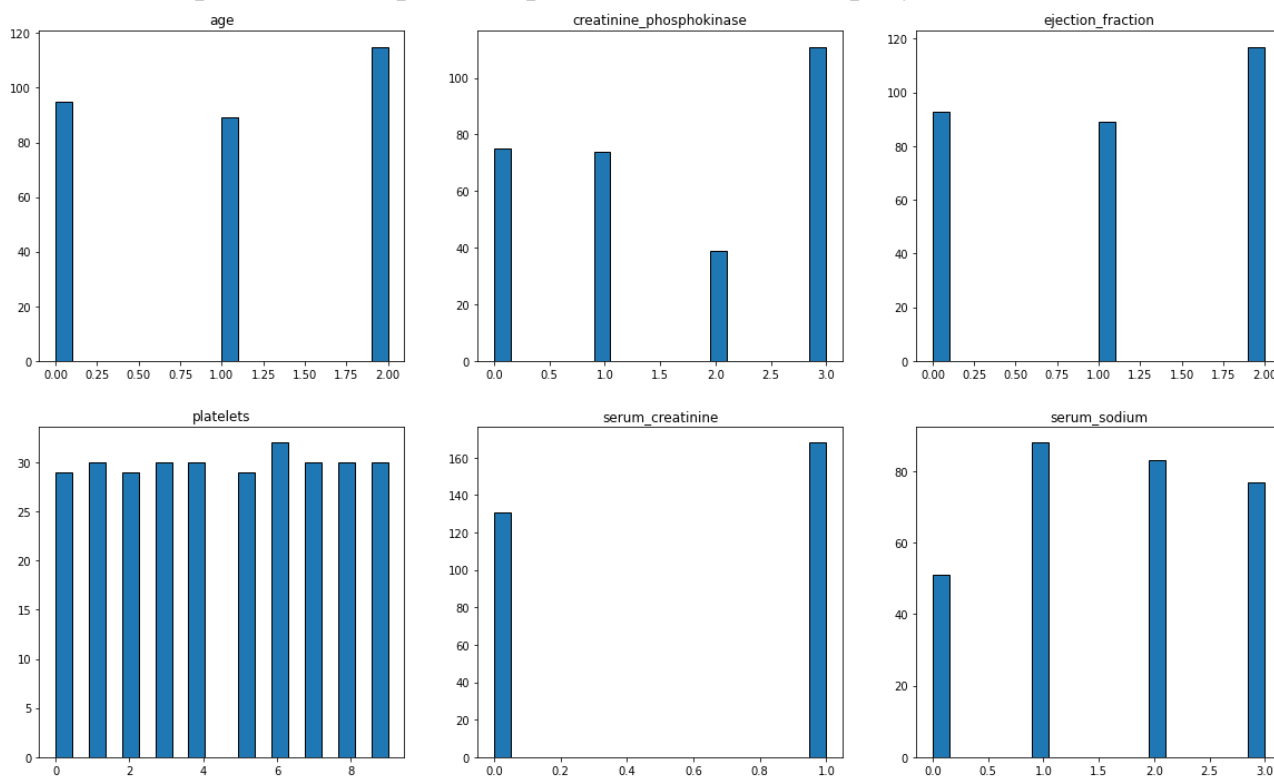
Дискретизация признаков

1. Провести дискретизацию признаков, используя KBinsDiscretizer.

Данные дискретизированы на следующее количество диапазонов:

- age – 3
- creatine_phosphokinase – 4
- ejection_fraction – 3
- platelets – 10
- serum_creatine – 2
- serum_sodium – 4

2. Построить гистограммы (рис. 11). Объяснить результат.



3. Вывести диапазоны каждого интервала (табл. 8).

| | |
|--------------------------|---|
| age | [40, 55, 65, 95] |
| creatinine_phosphokinase | [23, 116.5, 250, 582, 7861] |
| ejection_fraction | [14, 35, 40, 80] |
| platelets | [25100, 153000, 196000, 221000, 237000, 262000, 265000, 285200, 319800, 374600, 850000] |
| serum_creatine | [0.5, 1.1, 9.4] |
| serum_sodium | [113, 134, 137, 140, 148] |

Вывод

В ходе выполнения лабораторной работы было изучено большое количество методов безусловной минимизации. Среди них были первого порядка и второго, одношаговые и двухшаговые. Порядок метода определяется порядком производной, которую он использует. Количество шагов метода определяет, сколько точек, полученных на предыдущих итерациях, использует метод. Из полученных результатов можно сделать вывод, что метод Ньютона является наиболее быстрым среди одношаговых методов, что и подтверждается теорией.