МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ

РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

**Московский государственный технический университет**

**имени Н. Э. Баумана**

**(национальный исследовательский университет)**

**Центр дополнительного образования**

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

на тему:

**Прогнозирование конечных свойств новых материалов**

**(композиционных материалов)**

**по программе профессиональной переподготовки «Data Science Pro»**

Автор - Лобанова Е.Н.

Москва 2025

Содержание

1. Аналитическая часть

*1.1. Постановка задачи.*

*1.2. Разведочный анализ данных, EDA*

2. Практическая часть

*2.1. Предобработка данных.*

*2.2. Разработка и обучение модели, тестирование модели.*

*2.3. Создание удаленного репозитория и загрузка результатов работы на него.*

3. Список используемой литературы

**1. Аналитическая часть**

* 1. *Постановка задачи*

Целью данной работы является разработка алгоритмов машинного обучения, способных предсказать (спрогнозировать) ключевые механические характеристики композиционных материалов.

Композиционные материалы — это искусственно созданные материалы, состоящие из нескольких других с четкой границей между ними. Композиты обладают теми свойствами, которые не наблюдаются у компонентов по отдельности. При этом композиты являются монолитным материалом, т.е. компоненты материала неотделимы друг от друга без разрушения конструкции в целом. Современные композиты изготавливаются из разнообразных материалов: полимеры, керамика, стеклянные и углеродные волокна, при сохранении данного принципа. У такого подхода есть и недостаток: даже если мы знаем характеристики исходных компонентов, определить характеристики композита, состоящего из этих компонентов, достаточно проблематично. Для решения этой проблемы существуют два подхода: физические испытания образцов материалов или прогнозирование характеристик. Суть прогнозирования путем машинного обучения заключается в симуляции представительного элемента объема композита на основе данных о характеристиках входящих компонентов (связующего и армирующего компонента).

Кейс, рассматриваемый в данной работе, основан на реальных производственных задачах Центра НТИ «Цифровое материаловедение: новые материалы и вещества» (структурное подразделение МГТУ им. Н.Э. Баумана).

На входе имеется массив данных о начальных свойствах компонентов композиционных материалов (количество связующего, наполнителя, температурный режим отверждения и т.д.).

На выходе необходимо спрогнозировать ряд конечных свойств получаемых композиционных материалов.

**Актуальность:**

Созданные прогнозные модели помогут сократить количество проводимых испытаний, а также пополнить базу данных материалов возможными новыми характеристиками материалов и цифровыми двойниками новых композитов.

Характеристика датасета:

В качестве входных данных представлены 2 файла формата exl табличной формы, содержащие записи, каждая из которых представляет собой уникальную комбинацию параметров и соответствующих выходных характеристик.

Файл X\_bp.exl содержит 10 столбцов, из которых входными параметрами являются Соотношение матрица-наполнитель; Плотность, кг/м3; модуль упругости, ГПа; Количество отвердителя, м.%; Содержание эпоксидных групп,%\_2; Температура вспышки, С\_2; Поверхностная плотность, г/м2; Потребление смолы, г/м2, а выходными - Модуль упругости при растяжении, ГПа; Прочность при растяжении, МПа. Каждый столбец имеет 1023 уникальных числовых значения.

|  | **Соотношение матрица-наполнитель** | **Плотность, кг/м3** | **модуль упругости, ГПа** | **Количество отвердителя, м.%** | **Содержание эпоксидных групп,%\_2** | **Температура вспышки, С\_2** | **Поверхностная плотность, г/м2** | **Модуль упругости при растяжении, ГПа** | **Прочность при растяжении, МПа** | **Потребление смолы, г/м2** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 1.857143 | 2030.000000 | 738.736842 | 30.000000 | 22.267857 | 100.000000 | 210.000000 | 70.000000 | 3000.000000 | 220.000000 |
| **1** | 1.857143 | 2030.000000 | 738.736842 | 50.000000 | 23.750000 | 284.615385 | 210.000000 | 70.000000 | 3000.000000 | 220.000000 |
| **2** | 1.857143 | 2030.000000 | 738.736842 | 49.900000 | 33.000000 | 284.615385 | 210.000000 | 70.000000 | 3000.000000 | 220.000000 |
| **3** | 1.857143 | 2030.000000 | 738.736842 | 129.000000 | 21.250000 | 300.000000 | 210.000000 | 70.000000 | 3000.000000 | 220.000000 |
| **4** | 2.771331 | 2030.000000 | 753.000000 | 111.860000 | 22.267857 | 284.615385 | 210.000000 | 70.000000 | 3000.000000 | 220.000000 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **1018** | 2.271346 | 1952.087902 | 912.855545 | 86.992183 | 20.123249 | 324.774576 | 209.198700 | 73.090961 | 2387.292495 | 125.007669 |
| **1019** | 3.444022 | 2050.089171 | 444.732634 | 145.981978 | 19.599769 | 254.215401 | 350.660830 | 72.920827 | 2360.392784 | 117.730099 |
| **1020** | 3.280604 | 1972.372865 | 416.836523 | 110.533477 | 23.957502 | 248.423047 | 740.142791 | 74.734344 | 2662.906040 | 236.606764 |
| **1021** | 3.705351 | 2066.799773 | 741.475517 | 141.397963 | 19.246945 | 275.779840 | 641.468152 | 74.042708 | 2071.715856 | 197.126067 |
| **1022** | 3.808020 | 1890.413468 | 417.316232 | 129.183416 | 27.474763 | 300.952708 | 758.747882 | 74.309704 | 2856.328932 | 194.754342 |

Файл X\_nup.exl содержит 3 столбца (Угол нашивки, град; Шаг нашивки; Плотность нашивки) и 1040 строк с уникальными числовыми значениями.

| **Угол нашивки, град** | **Шаг нашивки** | **Плотность нашивки** |
| --- | --- | --- |
| **0** | 0 | 4.000000 | 57.000000 |
| **1** | 0 | 4.000000 | 60.000000 |
| **2** | 0 | 4.000000 | 70.000000 |
| **3** | 0 | 5.000000 | 47.000000 |
| **4** | 0 | 5.000000 | 57.000000 |
| **...** | ... | ... | ... |
| **1035** | 90 | 8.088111 | 47.759177 |
| **1036** | 90 | 7.619138 | 66.931932 |
| **1037** | 90 | 9.800926 | 72.858286 |
| **1038** | 90 | 10.079859 | 65.519479 |
| **1039** | 90 | 9.021043 | 66.920143 |

Файлы были преобразованы в формат csv, что предпочтительнее для работы в Python для увеличения скорости обработки и обеспечения большей совместимости при работе различных библиотек и инструментов. Далее произведено объединение 2 файлов в один методом INNER, получен массив данных (далее – датасет) на 1023 ячейки и 13 столбцов, все переменные числовые (формат Float64, Int64).

Основные обязательные статистики, такие как количество наблюдений (count), среднее арифметическое (mean), среднее квадратическое отклонение (std), минимальное (min) и максимальное (max) значения, а также первый (25%), второй (50%, медиана) и третий (75%) квартиль каждой количественной переменной, показали, что диапазон значений по признакам находится в адекватных пределах.

Пропущенные значения в датасете отсутствуют, что указывает на допустимое качество данных и возможность перехода к машинному обучению без этапа восстановления пропусков.

Однако в некоторых переменных, таких как, например, температура вспышки и поверхностная плотность, наблюдаются соответственные различия между минимумами и максимумами, что указывает на наличие выбросов.

Таким образом, имеется весь необходимый набор метрик для решения задачи построения регрессионной модели алгоритма машинного обучения, который будет определять значения Модуль упругости при растяжении, ГПа; Прочность при растяжении, МПа., способной учитывать сложные взаимосвязи между характеристиками компонентов композитных материалов и его итоговыми механическими свойствами. Также впоследствии регрессионная модель может быть интегрирована в прикладное программное обеспечение (в данной работе не рассматривается).

*1.2. Разведочный (исследовательский) анализ данных, EDA*

В данном разделе приводится краткое описание методов разведочного анализа данных, которые используются для первоначального анализа с учетом достоинств, недостатков, области применения и априорных предпосылок к работоспособности каждого из них.

Расчет корреляции производится для проверки на предмет линейной взаимозависимости каких-либо атрибутов во всех столбцах с помощью коэффициента Пирсона df.corr(), который является ее мерой. В контексте анализа данных, корреляция помогает понять, как одна переменная изменяется по отношению к другой. Значение коэффициента корреляции варьируется от -1 до 1. Корреляция может быть полезна для выявления скрытых зависимостей в данных и для предварительного анализа перед построением моделей машинного обучения.

Для представления о степени влияния факторов друг на друга и на целевую переменную была построена корреляционная матрица.

Взаимосвязь каждой пары параметров визуализирована с помощью инструмента Матрица рассеяния (scatter plot matrix).

Графические значения параметров представлены в виде тепловой карты (heatmap) из библиотеки Seaborn. Это инструмент визуализации данных, который отображает значения в виде цветовой матрицы, где цвета соответствуют интенсивности значений. Она используется для выявления закономерностей, корреляций, кластеров и аномалий в данных. Тепловые карты особенно полезны для анализа больших наборов данных, где трудно заметить эти закономерности, рассматривая просто цифры. Уровни прозрачности цвета регулируются с помощью параметра альфа. Полученные максимальные значение коэффициента корреляции для целевых переменных являются мерой тесноты и направления линейной связи используются для дальнейшей аналитики:

* Модуль упругости при растяжении/количество отвердителя, м, %, обратная зависимость между переменными -0,066х
* Прочность при растяжении/количество отвердителя, м, %, обратная зависимость между переменными -0,075х
* Соотношение матрица-наполнитель/потребление смолы, прямая зависимость между переменными 0,07х



Фильтрация, осуществленная далее, играет ключевую роль в анализе данных. С помощью оценки важности того или иного признака с помощью алгоритмов машинного обучения и отсечения ненужных производится на тестовой модели. Вначале датафрейм был разделен на train и test. Разделение данных на тренировочные и тестовые помогает избежать переобучения (overfitting) и обеспечивает объективную оценку производительности модели. Разделяя данные, осуществляем проверку, насколько хорошо модель справляется с задачей на данных, которые она не видела во время обучения. Это особенно важно в реальных приложениях, где модель будет использоваться для принятия решений на основе новых данных.

При выделении целевого признака "Модуль упругости при растяжении, ГПа” удалим остальные два “Прочность при растяжении, МПа", "Соотношение матрица-наполнитель” с помощью команды drop.

Результаты произведенного анализа с помощью команды mutual info regression показывают следующие важные признаки по отношению к модели train с целевым признаком Модуль упругости при растяжении, ГПа” по получившимся значениям: 'Плотность, кг/м3', 'модуль упругости, ГПа', 'Количество отвердителя, м.%', 'Содержание эпоксидных групп,%\_2', 'Температура вспышки, С\_2', 'Поверхностная плотность, г/м2', 'Потребление смолы, г/м2', 'Шаг нашивки', 'Прочность при растяжении, МПа'.

Анализ корреляции Пирсона модели для обучения показывает 2 переменных с достаточно сильной линейной связью с целевым признаком, это Потребление смолы, г/м2, Содержание эпоксидных групп,%\_2. Остальные показывают отрицательную зависимость или же несущественную (до 0,05х) положительную.

**2. Практическая часть**

*2.1. Предобработка данных*

Проверка данных модели на наличие выбросов осуществляется с помощью распределения Гаусса. Нормальное или гауссовское распределение – это одно из наиболее распространенных распределений вероятностей в статистике. Оно описывает, как значения случайной величины обычно распределены вокруг среднего значения, причем большая часть значений сосредоточена вблизи среднего, а вероятность отдаленных значений уменьшается. В нормальном (гауссовом) распределении выбросы – это значения, которые сильно отклоняются от среднего значения и находятся в "хвостах" распределения, то есть на больших расстояниях от центра.

Результаты анализа показывают, что все признаки имеют нормальное распределение Гаусса, кроме двух: Поверхностная плотность, г/м2, 'модуль упругости, ГПа' а параметр "угол нашивки" - бимодальное (2 пика на графике). Диаграммы размаха (Boxplot, ящик с усами) визуально показывает распределение для каждой переменной по числовым данным, отображая медиану, квартили и выбросы.

Далее была осуществлена проверка на нормальность распределения выборки в Scipy, которая используется для проведения статистического теста Шапиро-Уилка с помощью которого можно определить, взяты ли данные из нормально распределенной генеральной совокупности и есть ли выбросы. Выяснилось, что некоторые данные, скорее всего, не нормально распределены (меньше альфа=0,05).

Выбросы были удалены путем применения метода z-score, статистической меры отношения показателя к среднему значению в группе показателей. Z-счет был принят равным 2,5.

*Преобразование данных – нормализация*

В данной работе применен метод Бокс-Кокса для стабилизации дисперсии и улучшения нормальности распределений данных в текущих условиях. Это преобразование повышает достоверность статистических моделей и надежность выводов, сделанных на основе данных. Метод нормализации применен к переменным “Модуль упругости, ГПа”, “Поверхностная плотность”.

|  | **Плотность, кг/м3** | **модуль упругости, ГПа** | **Количество отвердителя, м.%** | **Содержание эпоксидных групп,%\_2** | **Температура вспышки, С\_2** | **Поверхностная плотность, г/м2** | **Потребление смолы, г/м2** | **Угол нашивки, град** | **Шаг нашивки** | **Плотность нашивки** | **Модуль упругости при растяжении, ГПа** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **count** | 911.000000 | 911.000000 | 911.000000 | 911.000000 | 911.000000 | 911.000000 | 911.000000 | 911.000000 | 911.000000 | 911.000000 | 911.000000 |
| **mean** | 1973.860620 | 731.401297 | 111.056368 | 22.207376 | 286.014837 | 477.107194 | 218.179365 | 45.543359 | 6.887547 | 57.096612 | 73.259028 |
| **std** | 70.118028 | 321.923704 | 26.427439 | 2.323873 | 39.651547 | 274.827292 | 57.162421 | 45.021436 | 2.411469 | 11.408243 | 3.009838 |
| **min** | 1801.940695 | 2.436909 | 40.304806 | 16.391595 | 186.508613 | 0.603740 | 72.530873 | 0.000000 | 0.571835 | 26.559894 | 65.553336 |
| **25%** | 1923.787286 | 497.412391 | 93.260237 | 20.583073 | 259.211711 | 264.063336 | 179.766002 | 0.000000 | 5.124226 | 49.860400 | 71.219440 |
| **50%** | 1977.126504 | 731.867638 | 110.615466 | 22.192903 | 285.811108 | 453.919475 | 219.249718 | 90.000000 | 6.908628 | 57.341920 | 73.219286 |
| **75%** | 2019.796983 | 950.526573 | 130.055633 | 23.959367 | 312.885525 | 690.820893 | 257.106878 | 90.000000 | 8.556590 | 64.513479 | 75.255148 |
| **max** | 2158.794836 | 1546.290886 | 181.032809 | 27.920843 | 386.067992 | 1182.326112 | 359.052220 | 90.000000 | 13.246219 | 86.012427 | 81.053293 |

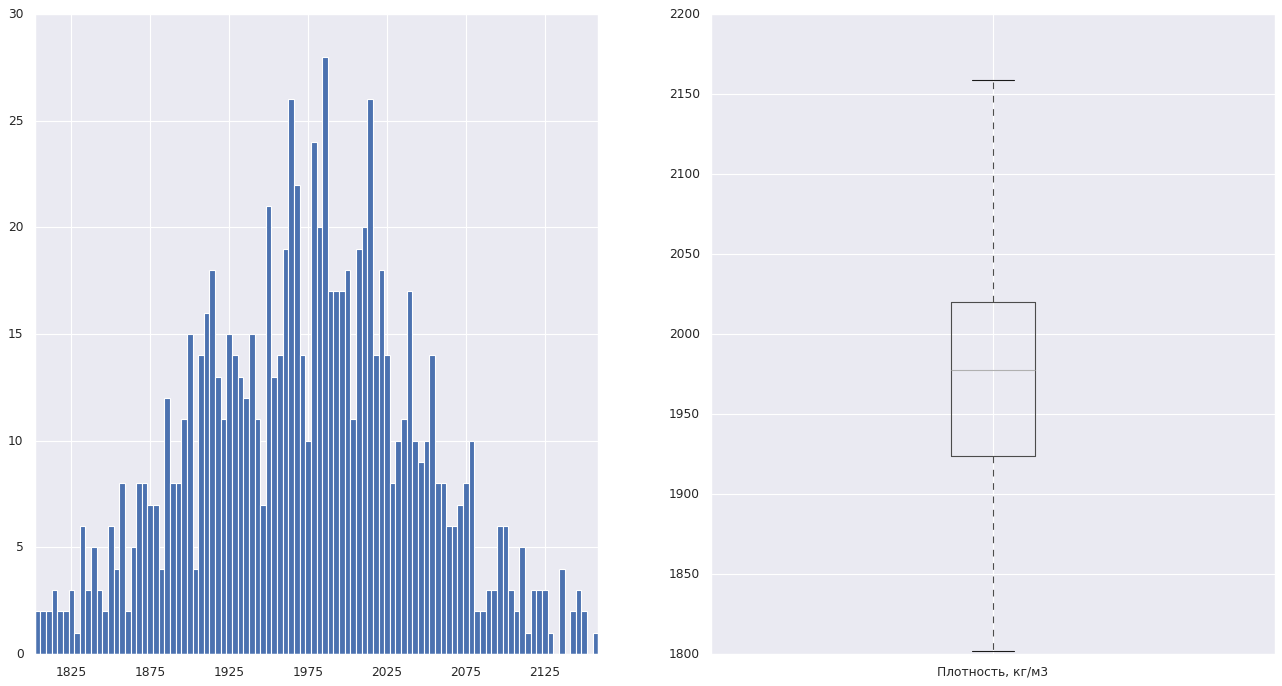
|  | **модуль упругости, ГПа** | **Поверхностная плотность, г/м2** | **Плотность, кг/м3** | **Количество отвердителя, м.%** | **Содержание эпоксидных групп,%\_2** | **Температура вспышки, С\_2** | **Потребление смолы, г/м2** | **Угол нашивки, град** | **Шаг нашивки** | **Плотность нашивки** | **Модуль упругости при растяжении, ГПа** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **count** | 805.000000 | 805.000000 | 805.000000 | 805.000000 | 805.000000 | 805.000000 | 805.000000 | 805.000000 | 805.000000 | 805.000000 | 805.000000 |
| **mean** | 344.595955 | 79.289307 | 1974.447313 | 110.832745 | 22.214658 | 285.588829 | 218.328274 | 39.689441 | 6.887312 | 56.941225 | 73.233486 |
| **std** | 134.151577 | 33.874901 | 71.248427 | 26.408543 | 2.335549 | 39.824273 | 56.924482 | 44.713326 | 2.450508 | 11.733107 | 3.030864 |
| **min** | 1.341736 | -0.430139 | 1801.940695 | 40.304806 | 16.391595 | 186.508613 | 72.530873 | 0.000000 | 0.571835 | 26.559894 | 65.553336 |
| **25%** | 248.942940 | 55.866060 | 1923.811976 | 93.290977 | 20.583309 | 258.991905 | 179.858423 | 0.000000 | 5.030449 | 49.516439 | 71.175111 |
| **50%** | 349.419206 | 80.363849 | 1977.302956 | 110.738213 | 22.208465 | 285.811108 | 219.350696 | 0.000000 | 6.916144 | 57.094641 | 73.181410 |
| **75%** | 435.663145 | 106.443864 | 2020.947411 | 129.947268 | 23.967543 | 312.425316 | 257.147388 | 90.000000 | 8.595968 | 64.578552 | 75.230994 |
| **max** | 660.712926 | 151.922427 | 2158.794836 | 181.032809 | 27.920843 | 386.067992 | 359.052220 | 9 |  |  |  |

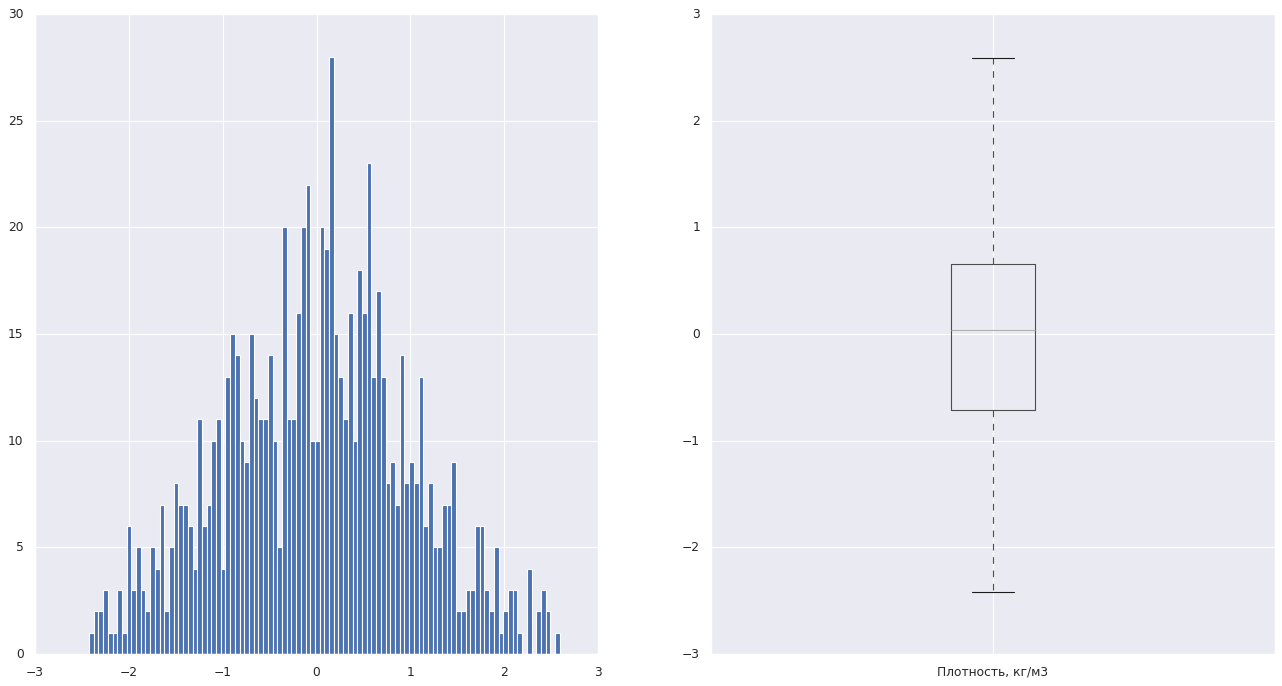
*Преобразование данных – стандартизация*

Стандартизация признаков производится путем вычитания среднего и затем масштабирования к единичной дисперсии, используя команду StandardScaler. Это важный метод, который, в основном, выполняется как этап предварительной обработки перед интеграцией данных их в модель машинного обучения, чтобы стандартизировать диапазон функциональных возможностей входного набора данных, и используется для изменения размера распределения значений так, чтобы среднее значение наблюдаемых значений было равно 0, а [стандартное отклонение](https://en.wikipedia.org/wiki/Standard_deviation) – 1.

Для визуализации стандартизированных данных через библиотеку Seaborn был применен График оценки плотности ядра (KDEplot) — это метод визуализации распределения наблюдений в наборе данных, аналогичный гистограмме, где данные представлены с помощью непрерывной кривой плотности вероятности в одном или нескольких измерениях.

Согласно задачам ВКР, построены гистограммы и ящик с усами до стандартизации и после на примере параметра «Плотность, м2».





Таким образом, была проведена предобработка данных с целью корректировки работы имеющихся алгоритмов в машинное обучение, которое является особенно чувствительным к масштабу входных данных, а также обеспечила корректное масштабирование признаков и очистку данных от выбросов, что является важным этапом для построения модели машинного обучения и позволило исключить аномальные значения, которые могли бы исказить обучение модели.

*2.2. Разработка и обучение модели, тестирование модели.*

При построении модели для предсказания - "Модуль упругости при растяжении, ГПа" столбец "Угол нашивки, град" был удален, т.к. данный параметр имеет бимодальное распределение, т.е. не является нормальным, т.к. либо выборка не является однородной и наблюдения порождены двумя или более "наложенными" распределениями, либо выбранные инструменты не подходят для измерения из-за смещения.

Также был удален столбец "Прочность при растяжении, МПа", т.к для него также потребовалась была бы отдельная модель.

Для оценки производительности моделей могут теоретически быть использованы следующие метрики (в работе применены некоторые из них):

*Коэффициент детерминации*, также называемый *оценкой R2*. Оценка r2 как мера оценки регрессионных моделей для идеальной модели, показывающей силу связи между двумя случайными величинами, составляет 1,0. Оценка r2 для модели, которая прогнозирует среднее значение, или для тривиальной (крайне упрощенной) модели равна 0,0. Значение метрики может быть отрицательно, если модель предсказывает хуже, чем тривиальная, т. е. при плохой обобщающей способности модели, при этом оценка r2 для худшей модели -2,0

*RMSE (Root Mean Squared Error, корень из среднеквадратичной ошибки)* –это разброс между предсказанными и истинными значениями. Это, по сути, корень из MSE (Mean Squared Error (среднеквадратичная ошибка), которая используется в качестве функции потерь при обучении моделей машинного обучения, чтобы минимизировать ошибку предсказания, и рассчитывается как среднее значение квадратов разностей между предсказанными и фактическими значениями. RMSE часто предпочтительнее из-за более удобной интерпретируемости и более чувствительна к большим ошибкам, к аномалиям и выбросам, чем MSE, выражается в тех же единицах, что и целевая переменная. Идеальное значение для RMSE= 0. Низкое значение метрики RMSE указывает на то, что модель хорошо предсказывает, а высокое - говорит о том, что модель имеет большие ошибки в предсказаниях

*MAE (средняя абсолютная ошибка)* показывает средний квадрат отклонений предсказанных значений от реальных и показывает оценку точности модели регрессии, т.е. насколько близки предсказываемые параметры к фактическим. Чем выше значение MAE, тем модель хуже. У идеальной модели MAE=0

*MAPE (средняя абсолютная ошибка в процентах)* выражается в процентах и имеет хорошую интерпретируемость. Идеальная модель имеет MAPE=0. Верхний предел - не ограничен.

При выбросах мы можем применить также:

*ME (максимальная ошибка)* показывает наихудший случай предсказания модели. В некоторых задачах важно, чтобы модель не ошибалась сильно, а небольшие отклонения не критичны. Зачастую эта метрика используется как вспомогательная совместно с другими через команду max\_error(y\_true, y\_pred).

В работе применены метрики RMSE(e), MSE, R2, MAPE.

В данном разделе работы показана ошибка каждой модели на тренировочной и тестирующей части выборки, обоснован выбор модели.

Для применения в работе были импортированы следующие пакеты и классы Python

from pandas import read\_csv, DataFrame

import numpy as np

from sklearn.linear\_model import LinearRegression, LogisticRegression

from sklearn.neighbors import KNeighborsRegressor

from sklearn.svm import SVR

from sklearn.ensemble import RandomForestRegressor

from sklearn.metrics import r2\_score

from sklearn.model\_selection import train\_test\_split

from sklearn import metrics

from sklearn.utils import shuffle

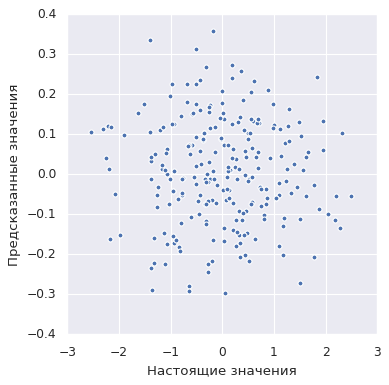
from sklearn.model\_selection import cross\_val\_score

from sklearn.linear\_model import Lasso

from sklearn.linear\_model import LassoCV

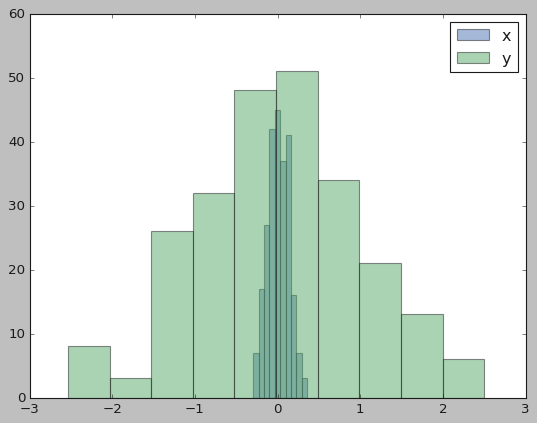
*Линейная регрессия (Linear regression)*

# Входы (регрессоры, x) и выход (предиктор, y) представлены массивами данных train и test моделей. С помощью команды .fit() вычислены оптимальные значение весов, используя существующие вход и выход (x и y) в качестве аргументов, определены вышеуказанные метрики оценки достоверности и качества данной модели.



# Данные возвращены в исходный масштаб с помощью метода .inverse\_transform() и визуализированы в гистограмме.

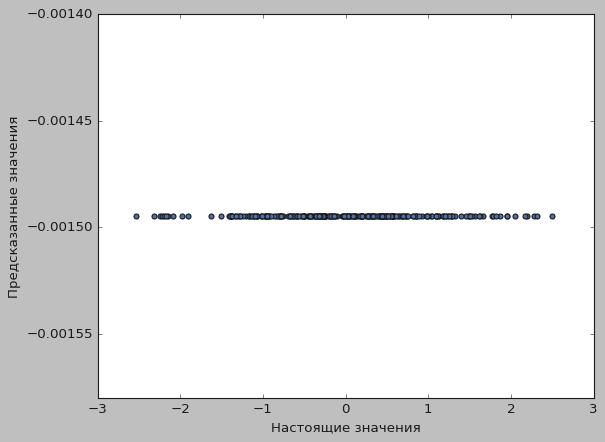
# C:\Users\1\Desktop\Без названия (4).png

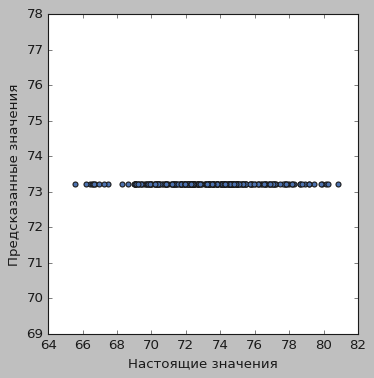


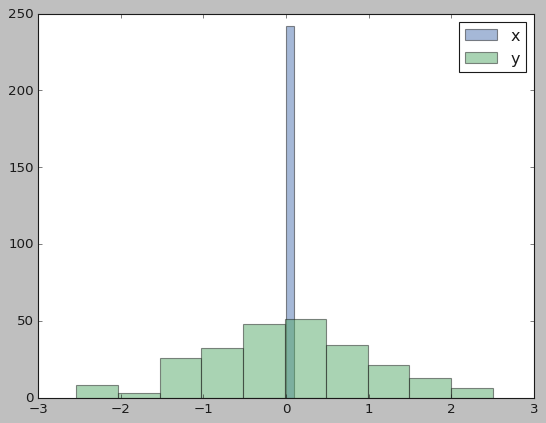
*Регресссия Lasso*

Данный тип регрессии был применен в работе для регуляризации линейной модели с помощью метода lasso\_regression.predict(X\_test) как способ уменьшить переобучение и упорядочить модель, т.к. чем меньше у нее будет степеней свободы, тем сложнее будет переобучить данные. Также, определены вышеуказанные метрики оценки достоверности и качества данной модели.

# Данные возвращены в исходный масштаб с помощью метода .inverse\_transform() и визуализированы в гистограмме.







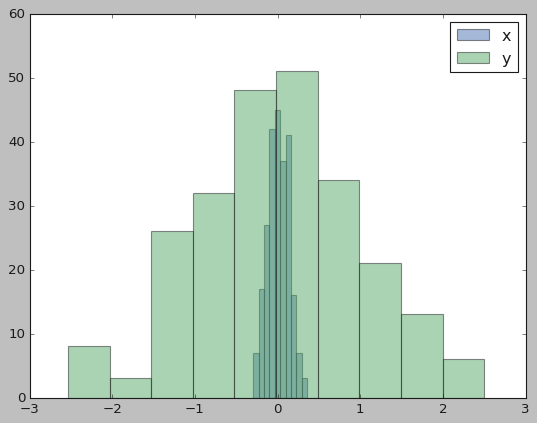
*Регрессия LassoCV*

Это также линейная модель, оценивающая разреженные коэффициенты. Она полезна благодаря своей тенденции отдавать предпочтение решениям с меньшим количеством ненулевых коэффициентов, что эффективно уменьшает количество признаков, от которых зависит данное решение, то есть данный метод, по сути, тесно связан с устранением шума методом преследований (BPDN)). Таким образом, можно отнести данный метод к [математической задаче оптимизации](https://en-m-wikipedia-org.translate.goog/wiki/Mathematical_optimization?_x_tr_sl=en&_x_tr_tl=ru&_x_tr_hl=ru&_x_tr_pto=rq).

По этой причине Lasso и его варианты играют основополагающую роль для многомерных наборов данных с большим количеством коллинеарных признаков, являясь одним из предпочтительных.

Также, определены вышеуказанные метрики оценки достоверности и качества данной модели.

# Данные возвращены в исходный масштаб с помощью метода .inverse\_transform() и визуализированы в гистограмме.

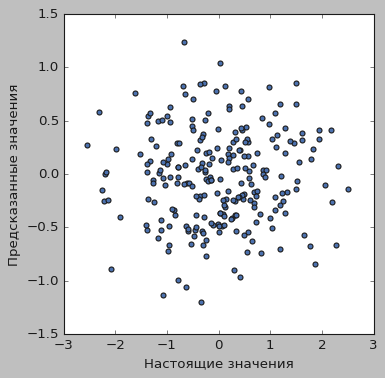


*Kneighborsregressor*

Метод k ближайших соседей (K-Nearest Neighbors, KNN) — алгоритм классификации и регрессии, основанный на гипотезе компактности, которая предполагает, что расположенные близко друг к другу объекты в пространстве признаков имеют схожие значения целевой переменной или принадлежат к одному классу. Преимуществами данного метода являются простота в реализации и интерпретации; возможность применения во многих задачах, особенно в рекомендательных системах; высокая точность прогнозов при правильном подборе k и метрики расстояния

Изначально, на обучающем множестве с помощью вызван метод fit объекта knn, который принимает в качестве аргументов массив x\_train, содержащий обучающие данные, и массив y\_train соответствующих обучающих меток.

Также, определены вышеуказанные метрики оценки достоверности и качества данной модели.



Для осуществления предсказаний вызывается метод predict, который в качестве аргументов принимает тестовые данные, при этом оптимальные значения параметров подобраны с помощью метода сетки GridSearchCV, который используется для непосредственно поиска набора возможных значений для каждого гиперпараметра и перекрёстной проверки (кросс-валидации). После извлечения наилучших значений гиперпараметров строятся прогнозы. для каждой комбинации подобранных значений.

Используемый словарь гиперпараметров - стандартный в сочетании с методом GridSearchCV для систематического поиска оптимальной комбинации этих параметров, обеспечивающей наилучшую производительность модели, и представляет собой следующее.

Параметр n\_neighbors представляет собой количество соседей (k), учитываемых при прогнозировании. Диапазон list(range(1, 50)) указывает, что k будет проверяться от 1 до 49.

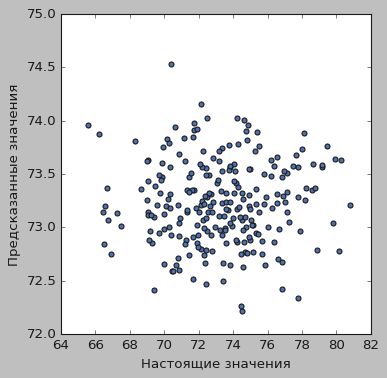
Параметр p относится к метрике расстояния Минковского:p=1 соответствует манхэттенскому расстоянию (норма L1), p=2 соответствует евклидову расстоянию (норма L2).

Параметр weights определяет, как взвешиваются вклады соседей: "uniform"(все соседи весят одинаково), "distance" (вес соседей определяется обратной величиной расстояния, то есть более близкие соседи оказывают большее влияние)

Параметр metric определяет метрику расстояния, используемую для расчета расстояния между точками данных: 'mahalanobis' (расстояние Махаланобиса), 'minkowski' (расстояние Минковского (обобщение евклидова и манхэттенского расстояний, управляемое параметром p), 'cosine' (косинусное сходство (часто используется для текстовых данных или данных высокой размерности), 'chebyshev' (расстояние Чебышёва (максимальная абсолютная разность между координатами)), 'correlation' (корреляционное расстояние), 'euclidean' (Евклидово расстояние).

Параметр algorithm определяет алгоритм, используемый для вычисления ближайших соседей: 'ball\_tree' (древовидная структура данных, оптимизированная для данных высокой размерности), 'kd\_tree' (древовидная структура данных, оптимизированная для данных низкой размерности), «brute»: метод полного перебора, который вычисляет расстояния до всех точек, подходящий для небольших наборов данных или когда методы на основе деревьев неприменимы.

Данные возвращены в исходный масштаб.



Также, определены вышеуказанные метрики оценки достоверности и качества данной модели.

*RandomForestRegressor*

Алгоритм случайного леса (Random Forest) — универсальный алгоритм машинного обучения, суть которого состоит в использовании ансамбля решающих деревьев, каждое из которых немного отличается от других из-за случайности при обучении, и предсказание получается путем объединения результатов всех деревьев. Это один из немногих алгоритмов, который можно использовать в абсолютном большинстве задач. Благодаря своей гибкости Random Forest применяется для решения практически любых проблем в области машинного обучения. Сюда относятся классификации (RandomForestClassifier) и регрессии (RandomForestRegressor), а также более сложные задачи (отбор признаков, поиск выбросов/аномалий и кластеризации).

Сетка param\_grid содержит следующие комбинации гиперпараметров:

n\_estimators - список определяет количество деревьев в случайном лесу. Значения генерируются с использованием np.linspace в диапазоне от 100 до 600, с тремя равномерно распределенными числами. В результате n\_estimators принимает значения 100, 350 и 600.

max\_thought (depth) - этот список определяет максимальную глубину каждого дерева в случайном лесу. Значения генерируются с использованием метода np.linspace (равномерно распределенной последовательности элементов в заданном интервале) в диапазоне от 10 до 110, с тремя равномерно распределенными числами, в результате чего получаются значения 10, 60 и 110. Кроме того, к этому списку добавляется None, что означает, что узлы будут расширяться до тех пор, пока все листья не станут чистыми или пока все листья не будут содержать меньше min\_samples\_split выборок.

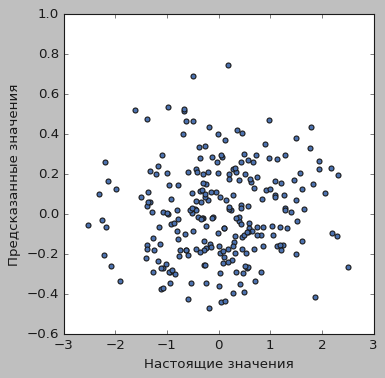
min\_samples\_leaf - этот список определяет минимальное количество выборок (образцов), необходимое для листового узла. Значения: 1, 2 и 4.

При использовании для настройки гиперпараметров в данной модели алгоритм, подобный GridSearchCV, обучает модель для всех возможных комбинаций этих значений гиперпараметров, что позволяет выбрать наиболее эффективный набор гиперпараметров для заданного набора данных и метрики оценки

cv=10 задаёт количество кросс-валидаций для перекрёстной проверки K-слоёв. Модель обучена и оценена 10 раз, при этом каждая серия использует разное подмножество данных для проверки. Это помогает получить более надёжную оценку производительности модели.

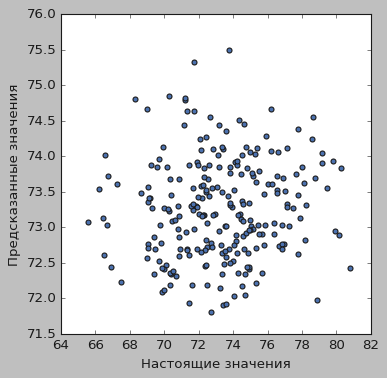
verbose=1 управляет подробностью вывода в процессе поиска. Значение 1 указывает на то, что на консоль будут выводиться некоторые сообщения о ходе поиска, предоставляя обновления. Более высокие значения обычно обеспечивают более подробный вывод.

n\_jobs=-1 задаёт количество ядер, используемых для параллельной обработки. Значение -1 означает, что будут использоваться все доступные ядра, что значительно ускоряет процесс настройки гиперпараметров, особенно при больших значениях param\_grid и cv.



Определены вышеуказанные метрики оценки достоверности и качества данной модели.

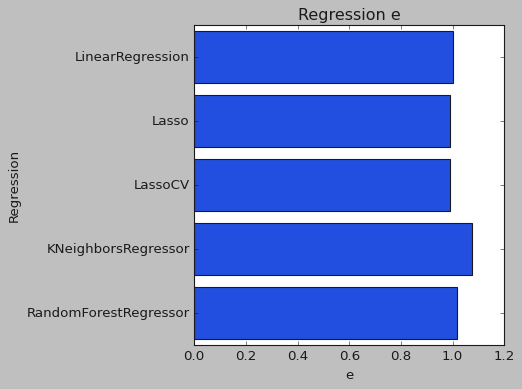
# Данные возвращены в исходный масштаб с помощью метода .inverse\_transform() и визуализированы в гистограмме.

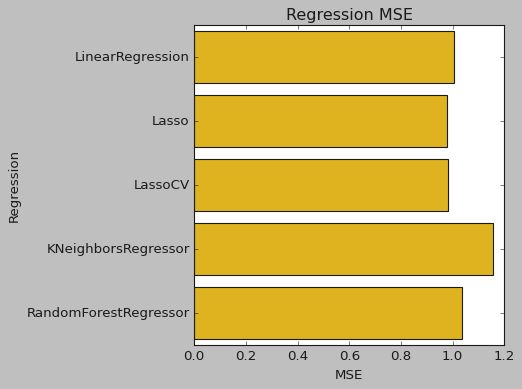


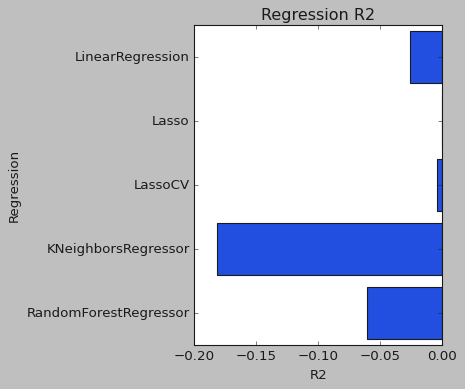
*Сравнение результатов*

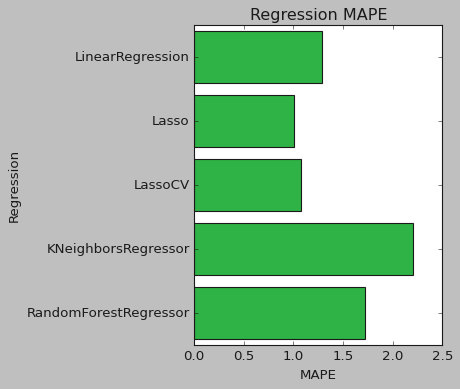
С помощью метода log проведено сопоставление полученных результатов метрик качества предиктивных моделей. Данные сведены в таблицу.

|  | **Regression** | **RMSE(e)** | **MSE** | **R2** | **MAPE** |
| --- | --- | --- | --- | --- | --- |
| **0** | LinearRegression | 1.001900 | 1.003803 | -0.025842 | 1.289533 |
| **1** | Lasso | 0.989212 | 0.978541 | -0.000025 | 1.000562 |
| **2** | LassoCV | 0.991235 | 0.982546 | -0.004118 | 1.074596 |
| **3** | KNeighborsRegressor | 1.075398 | 1.156480 | -0.181871 | 2.200788 |
| **4** | RandomForestRegressor | 1.009462 | 1.019014 | -0.041387 | 1.682908 |









| **RMSE(e)** | **MSE** | **R2** | **MAPE** |
| --- | --- | --- | --- |

Итак, основными метриками, используемыми для оценки точности прогноза и оценки эффективности моделей машинного обучения на основе регрессии стали из среднеквадратичной ошибки (RMSE(е)), среднеквадратичная ошибка (MSE), коэффициент детерминации [(R²)](https://arize.com/blog-course/r-squared-understanding-the-coefficient-of-determination/) и средняя абсолютная процентная ошибка (MAPE).

Полученные значения квадратного корня среднеквадратичной ошибки (RMSE(е)) показывают, насколько в среднем далеки прогнозы от фактических значений (имеется большая погрешность и меньшая точность прогнозов), в том же масштабе, что и у целевой переменной. Более низкое значение RMSE относительно других предполагает меньшие средние ошибки прогнозирования и присутствует в моделях Lasso, Lasso CV

Полученные значения средняя абсолютная ошибка (MSE) также указывают на невысокую точность прогнозирования представленных моделей. Более низкое значение MSE относительно других предполагает лучшую точность относительно других и присутствует в моделях Lasso, Lasso CV

Полученные значения коэффициента детерминации [(R²)](https://arize.com/blog-course/r-squared-understanding-the-coefficient-of-determination/) показывают, что модель слабо объясняет дисперсию целевых переменных. Это может быть связано с высокой сложностью данных, наличием скрытых факторов или ограничением размеров выборки.

Полученные значения средней абсолютной процентной ошибки (MAPE) также нерелевантны, необходима последующая доработка модели

*2.3. Создание удаленного репозитория и загрузка результатов работы на него.*

https://github.com/elenallie?tab=repositories

**3. Список используемой литературы**

1. BABOK. Руководство к Своду знаний по бизнес-анализу. Версия 3.0. – М.: Издательство Олимп - Бизнес, 2024. – 626 с.
2. Андерсон, К. Аналитическая культура От сбора данных до бизнес-результатов. – М.: Издательство Манн, Иванов и Фербер, 2023. – 336 с.
3. Вигерс, К., Битти, Дж. Разработка требований к программному обеспечению. - М.: Издательство BHV, 2020. – 736 с.
4. [Гмурман, В. Е. Теория вероятностей и математическая статистика. Изд. 4-е, доп. учеб. пособие для вузов. – М.: Издательство Высшая школа, 1972. – 368 с.](https://www.ozon.ru/product/teoriya-veroyatnostey-i-matematicheskaya-statistika-174708489/)
5. Лутц, М. Изучаем Python. Том 1. 5-е изд. - М.: Издательство Вильямс, 2019. – 832 с.
6. Брюс, П., Брюс, Э., Гедек, П. Практическая статистика для специалистов Data Science. – М, Издательство БХВ, 2021. – 352 с.
7. Рашка, С., Мирджалили, В. Python и машинное обучение. - М, Издательство [Диалектика-Вильямс](https://www.ozon.ru/publisher/dialektika-vilyams-135585305/), 2021. – 848 с.

# 8. Бурков, А. Машинное обучение без лишних слов. – С-Пб, Издательство Питер, 2020. – 192 с.

# 9. Пасхавер, Б. Pandas в действии. - С-Пб, Издательство Питер, 2023. – 512 с.

# 10. Плас, Дж. Вандер. Python для сложных задач: наука о данных и машинное обучение. - С-Пб, Издательство Питер, 2018 – 576 с.

# 11. Кудрявцев, В. Б. Интеллектуальные системы: учебник и практикум для бакалавриата и магистратуры, 2017

# 12. Кудрявцев, В. Б. Интеллектуальные системы: учебник и практикум для вузов — М.: Издательство Юрайт, 2021. — 165 с.

# 13. Шарден, Б. Крупномасштабное машинное обучение вместе с Python: учебное пособие — М.: Издательство ДМК Пресс, 2018. — 358 с.

# 14. Жерон, О. Прикладное машинное обучение с помощью Scikit-Learn, Keras и TensorFlow : концепции, инструменты и техники для создания интеллектуальных систем. O’Reilly Media, 2019

# 15. Груздев, А. В. Прогнозное моделирование в IBM SPSS Statistics, R и Python: метод деревьев решений и случайный лес: руководство / А. В. Груздев. — М.: ДМК Пресс, 2018. — 642 с.

# 16. Подкорытова, О. А. Анализ временных рядов: учебное пособие для вузов — М.: Издательство Юрайт, 2021. — 267 с.

# Станкевич, Л. А. Интеллектуальные системы и технологии: учебник и практикум для вузов — М.: Издательство Юрайт, 2021. — 397 с.

# 18. Мхитарян, В. С. [и др.]. Статистика. В 2 ч. Часть 1: учебник и практикум для вузов – М.: Издательство Юрайт, 2021. — 249 с.

19. Мхитарян, В. С., Агапова, Т. Н., Ильенкова, С. Д., Суринов, А. Е. Статистика. В 2 ч. Часть 2: учебник и практикум для вузов – М.: Издательство Юрайт, 2021.— 270 с.