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)»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА  
по курсу  
«Data Science»

Тема: «Исследование датасета “Оставшееся количество  
 циклов зарядки аккумулятора”»

Слушатель Ларин Петр Михайлович

Москва, 2023

**Содержание**

[Введение 3](#_Toc133325724)

[1. Описание датасета и постановка задачи 4](#_Toc133325725)

[2. Используемые программные средства 5](#_Toc133325726)

[3. Разведочный анализ и предобработка 5](#_Toc133325727)

[4. Методика исследования 11](#_Toc133325728)

[5. Нейронная сеть 14](#_Toc133325729)

[6. Сравнение регрессоров 14](#_Toc133325730)

[7. Оптимальные наборы признаков 16](#_Toc133325731)

[8. Выводы 16](#_Toc133325732)

[9. Приложение 17](#_Toc133325733)

[Список литературы 19](#_Toc133325734)

# Введение

Данная работа посвящена исследованию датасета «Оставшееся количество циклов зарядки аккумулятора». Датасет связывает параметры циклов зарядки и разрядки (продолжительность, напряжение) распространенной модели литий-ионных аккумуляторов с оставшимся полезным сроком службы аккумулятора. Актуальность темы обусловлена динамичным ростом глобального рынка литий-ионных батарей (совокупный среднегодовой темп роста в перспективе до 2030 г. оценивается от 13% до 20%).

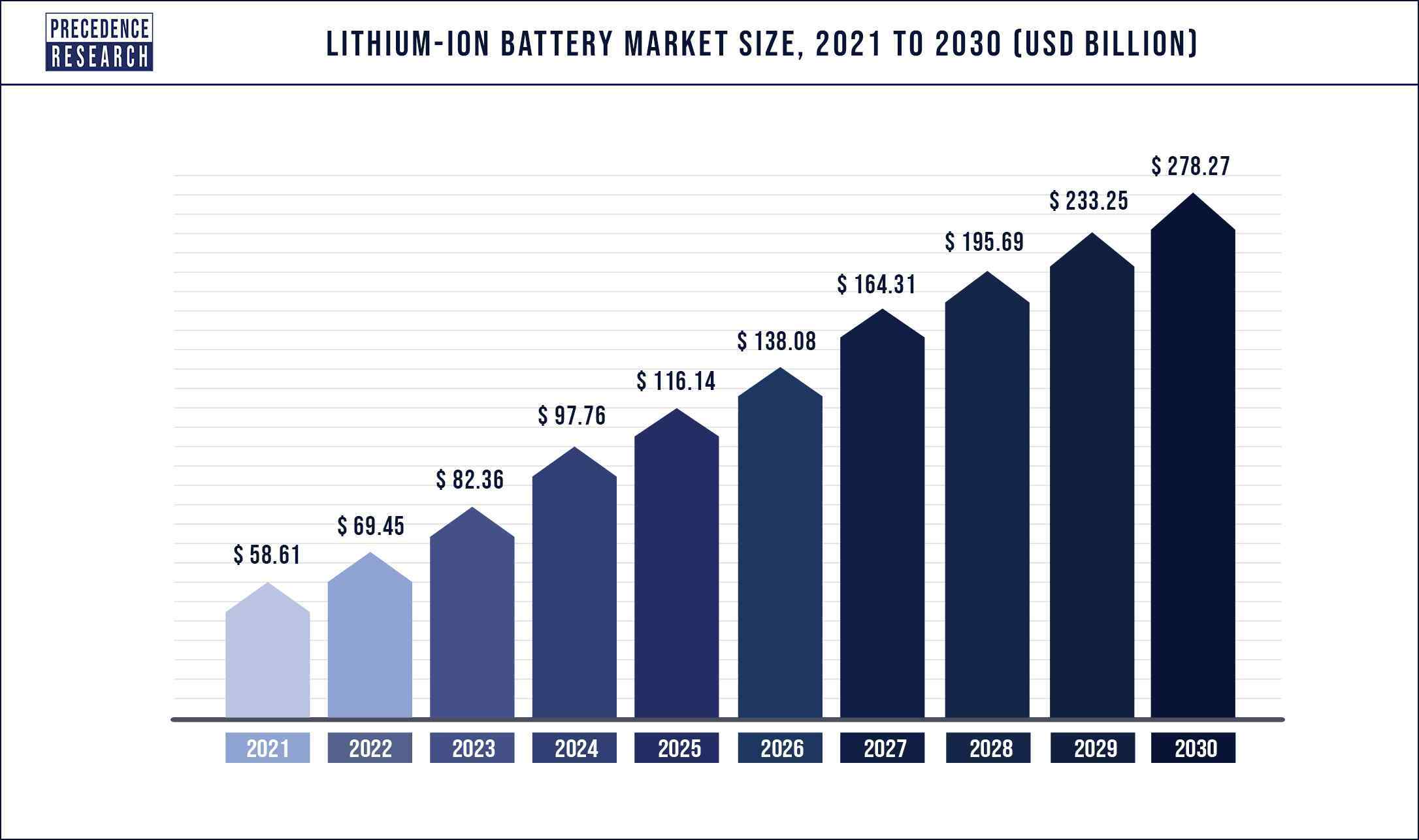


Рисунок 1 – Глобальный рынок литий-ионных батарей.  
 Источник – PrecedenceResearch.com

# Описание датасета и постановка задачи

Датасет «Оставшееся количество циклов зарядки аккумулятора» (оригинальное название Battery Remaining Useful Life – RUL) создан аргентинским специалистом по машинному обучению Игнасио Виньюалесом (Ignacio Viñuales) на основе исследований, выполненных Гавайским Институтом природной энергии (HNEI). Исследование включало тестирование 14 аккумуляторов распространенного типа NMC-LCO 18650 номинальной емкости 2.8 А·ч. (Данный тип широко используется в самых разных изделиях, от ноутбуков до электромобилей.) Каждый аккумулятор был подвергнут более чем 1000 циклам разрядки/зарядки при температуре 25°C, включающим CC-CV зарядку в режиме C/2 и разрядку в режиме 1.5C. Переработав первичные данные исследования, автор датасета создал новые признаки, описывающие динамику напряжения и силы тока для каждого цикла, с целью предсказывания оставшегося количества циклов аккумулятора. Датасет содержит сводную информацию о тестировании всех 14 аккумуляторов.

Датасет и сопутствующая информация размещены на [1, 2].

В датасете 9 столбцов и 15064 строки, то есть более 1000 циклов для каждого из протестированных аккумуляторов. В столбцах приводится следующая информация:

* Индекс цикла (нумерация не сквозная, а повторяется для каждого аккумулятора)
* F1: Полная продолжительность разрядки (с)
* F2: Продолжительность разрядки с 3.6 В до 3.4 В (с)
* F3: Стартовое напряжение при разрядке (В)
* F4: Стартовое напряжение при зарядке (В)
* F5: Продолжительность зарядки до 4.15 В (с)
* F6: Продолжительность зарядки постоянным током (с)
* F7: Полная продолжительность зарядки (с)
* RUL: Целевая переменная (оставшееся количество циклов зарядки аккумулятора)

# Используемые программные средства

Для анализа датасета использовалась среда Jupyter Notebook в составе пакета Anaconda для Windows. Для разработки приложения также использовалась среда разработки Microsoft Visual Studio. Серверная часть приложения – пакет TensorFlow Serving, размещенный в контейнере Docker Desktop.

# Разведочный анализ и предобработка

В датасете нет пропусков, но даже при беглом осмотре очевидно наличие выбросов. Поскольку строки датасета содержат результаты последовательных измерений параметров циклов зарядки и разрядки, отличия в значениях соседних строк крайне невелики, порядка единиц процентов. Приведем примеры строк, которые выбиваются из данной нормы.

Graphical user interface, table

Description automatically generated

Рисунок 2 – пример выброса (строка 14)

На рисунке 2 приведен пример выброса, где значения у пяти признаков отличаются от соседних строк на 2 порядка. Данные признаки содержат время зарядки и разрядки в секундах, и в соседних с выбросом строках это время имеет порядок тысяч секунд, то есть нескольких часов. В строке с выбросом время имеет порядок суток. Нет никаких разумных способов объяснения таких величин чем-либо, кроме ошибки измерения или обработки.

Graphical user interface, application, table, Excel

Description automatically generated

Рисунок 3 – пример выброса (строка 7607)

На рисунке 3 приводится выброс не только с отличием значений на 2 порядка, но и с отрицательным значением.

Graphical user interface, table

Description automatically generated

Рисунок 4 – пример выброса (строка 11826)

На рисунке 4 приводится выброс с отличием значений на 2 порядка в меньшую сторону. В отношении данных рисунков также можно сделать вывод об ошибке измерения либо обработки.

Поскольку у каждого выброса аномальные значения присутствуют сразу у 5 признаков, единственным разумным решением будет удаление таких выбросов.

A picture containing calendar

Description automatically generated

Рисунок 5 – распределения до удаления выбросов

На рисунках 5 и 6 хорошо видно присутствие масштабных выбросов у признаков F1, F2, F5, F6 и F7.

Для выбора порогов удаления выбросов используем графики квартилей данных пяти признаков. В итоге, после достаточно консервативного выбора порогов, в датасете остается 14845 из 15064 значений, т. е. отброшено 1.45% или 219 значений.

Видно, что выбросы можно было удалить более агрессивно, однако мы остановимся на этом.

Chart, box and whisker chart

Description automatically generated

Рисунок 6 – графики boxplot до удаления выбросов

Chart

Description automatically generated

Рисунок 7 – распределения после удаления выбросов

Сравним ковариационные матрицы до и после удаления выбросов. Видно, что удаление выбросов радикально улучшило корреляции. Если до удаления выбросов признаки F1, F2, F5, F6 и F7 являлись фактически константами и их корреляция с целевой переменной лежала в диапазоне 1–18%, то после удаления корреляции лежат в диапазоне 73–98%.

Chart, box and whisker chart

Description automatically generated

Рисунок 8 – графики boxplot после удаления выбросов

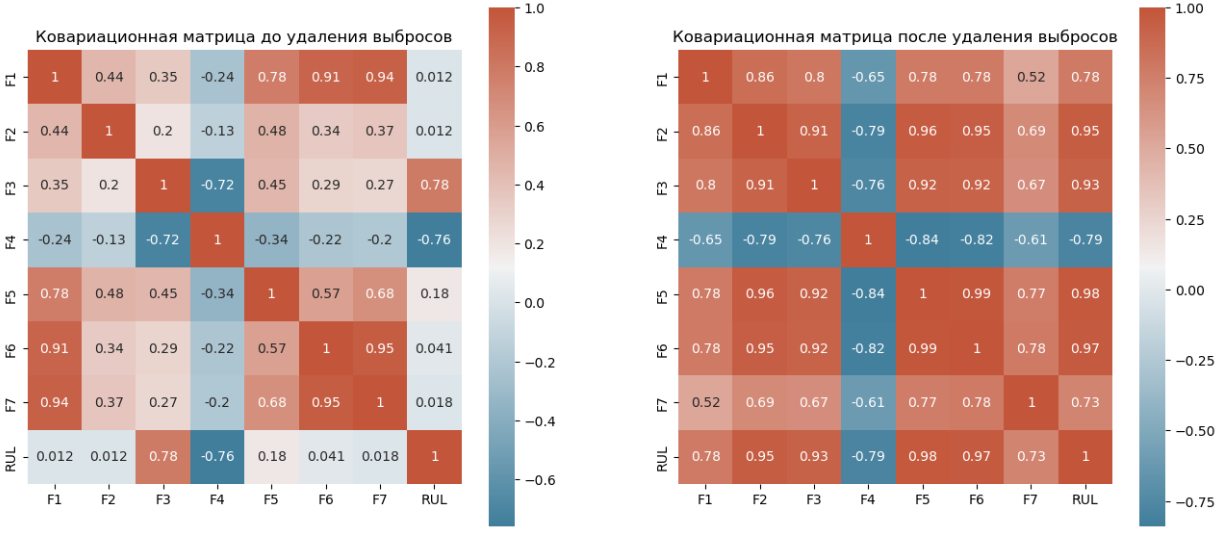


Рисунок 9 – ковариационные матрицы

Также мы наблюдаем высокую корреляцию между признаками, что естественно приводит нас к анализу главных компонент.

Chart, bar chart

Description automatically generated

Рисунок 10 – метод главных компонент

Анализ главных компонент показывает, что датасет представляет собой практически одномерное многообразие в многомерном пространстве признаков, и что признаки в значительной степени избыточны.

# Методика исследования

Для исследования датасета мы будем применять классические регрессоры, ансамблевые методы и полносвязную нейронную сеть.

Из библиотеки Scikit-learn использованы:

* + Линейная регрессия
  + Ridge (*L*2 – регуляризация, или регуляризация Тихонова)
  + Полиномиальная регрессия
  + Метод *k*-ближайших соседей
  + RandomForest
  + Градиентный бустинг

Из библиотеки TensorFlow:

* + Полносвязная нейронная сеть

Замечание: включение Ridge-регрессора обусловлено сильной корреляцией признаков. В ситуациях сильной корреляции рекомендуется применять методы регуляризации: Lasso (*L*1-регуляризация) либо Ridge (*L*2- регуляризация), причем Lasso с байесовской точки зрения соответствует априорному распределению Лапласа для коэффициентов регрессии, а Ridge – нормальному распределению [3].

Для каждого регрессора и каждого *n* ∈ [1, 7] проводится подбор *n* признаков, оптимальных по критерию *R*2, из всех наборов из *n* признаков. На полученных оптимальных наборах проводится кросс-валидация и по ее результатам – усреднение *R*2 и MAE. Таким образом, для каждого цикла исследования мы получаем следующий выходной набор данных:

* 7 наборов признаков, оптимальных по *R*2
* 7 соответствующих значений *R*2
* 7 соответствующих значений MAE

Заметим, что оптимизация производится по метрике *R*2, поэтому значения MAE не обязательно будут оптимальными и должны рассматриваться как значения, соответствующие оптимальным значениям *R*2. Из указанного выше набора сохраняются следующие значения:

* Лучшее значение *R*2 из 7 полученных
* Лучшее значение MAE из 7 полученных
* Значения *R*2, полученные при кросс-валидации на одном, двух и трех признаках, и соответствующие им значения MAE. Интерес к данным параметрам вызван сильной корреляцией между признаками и естественно возникающим при этом желанием оценить разброс качества модели на минимальных и максимальных наборах признаков.

Таким образом, мы получаем 8 параметров для каждого полного цикла подбора параметров и кросс-валидации.

Для линейной регрессии и Ridge-регрессии проводится по одному циклу. Для полиномиальной регрессии предварительно создаются дополнительные признаки до 7-й степени включительно, и также проводится один цикл. При этом оказывается, что лучшими признаками все равно остаются первые степени F5 и F6.

Для регрессора kNN проводится 5 циклов с количеством соседей от 2 до 6, при этом оптимальным оказывается цикл с 3 соседями.

Для RandomForest проводится 5 циклов с количеством деревьев от 60 до 1000, при этом оптимальным оказывается максимальное число деревьев, но с крайне незначительным перевесом.

A picture containing text

Description automatically generated

Рисунок 11 – циклы подбора признаков

Для градиентного бустинга предварительно проводится подбор оптимального значения параметра learning\_rate, и затем 5 циклов кросс-валидации в окрестности этого оптимального значения.

Для всех регрессоров, кроме нейронной сети, подбор признаков производится с помощью метода SequentialFeatureSelector.

На рисунке 11 приводится пример двух циклов подбора признаков.

# Нейронная сеть

Для нейронной сети была выбрана архитектура с двумя полносвязными скрытыми слоями. Опытным путем был подобран размер слоев (по 50 нейронов), оптимизатор – Adam, который оказался заметно лучше остальных (за исключением разве что Adamax) и функция активации relu, которая показала себя лучше первоначально планировавшегося tanh.

При подборе признаков и кросс-валидации модель перекомпилировалась перед каждым запуском, чтобы исключить продолжение обучения сети, уже обученной на предыдущих циклах кросс-валидации.

# Сравнение регрессоров

В результате проведения процедур подбора признаков получается следующая сводная таблица и графики параметров *R*2 и MAE.

Таблица 1 – Сводная таблица оценки регрессоров

Graphical user interface, application

Description automatically generated

Chart, bar chart

Description automatically generated

Рисунок 12 – сравнение регрессоров по *R*2

Chart, bar chart

Description automatically generated

Рисунок 13 – сравнение регрессоров по MAE

# Оптимальные наборы признаков

Интересным фактом является то, что регрессоры (за исключением нейронной сети) разделились на две группы по тому, какие наборы признаков они сочли оптимальными – см. рис. 14.

Timeline

Description automatically generated with low confidence

Рисунок 14 – Оптимальные наборы признаков

В то же время, у нейронной сети подобных устойчивых предпочтений не наблюдалось.

Также следует отметить, что у всех регрессоров, за исключением нейронной сети, лучшие значения *R*2 наблюдались на максимальном наборе признаков (7), в то время как у нейронной при повторении расчетов сети это значение варьировалось от 5 до 7.

# Выводы

* Ключевым моментом в исследовании датасета явилось удаление выбросов. Остается неизвестным, почему автор датасета сам не произвел подобное удаление или не прокомментировал их наличие.
* Высокая избыточность данных позволяет получать качественные предсказания на малых наборах признаков (что будет использовано при создании приложения).
* Регрессоры разделились на 3 группы: семейство линейных регрессоров, включающее Ridge и полиномиальную регрессию, оказалось чуть хуже нейронной сети, а семейство ансамблевых методов, дополненное kNN, оказалось чуть лучше. При этом разделение произошло не только по качеству предсказаний, но и по набору оптимальных признаков (см. выше).
* Наилучшие показатели демонстрируют регрессоры RandomForest и kNN, их показатели очень близки. При этом вычислительные ресурсы, требующиеся для RandomForest, существенно (на 1–2 порядка) выше ресурсов для kNN. Поэтому лидером данного исследования следует признать регрессор kNN.

# Приложение

Использование всех 7 признаков для предсказания представляется избыточным. Для приложения построим нейронную сеть, которая будет делать предсказания по 3 признакам F1, F4 и F7, и предсказывать не только целевую переменную, но и остальные 4 признака.

Diagram

Description automatically generated

Рисунок 15 – Схема приложения

В приложении используется архитектура «Клиент – сервер», где сервером служит пакет TensorFlow Serving, размещенный в Docker Desktop для Windows, а клиентом – приложение для Windows, написанное на C++ с использованием новейшего фреймворка Microsoft WinUI3 для Windows 10/11. Изображение окна приложения приводится на рис. 16. Для использования приложения необходимо загрузить всю папку по адресу <https://github.com/petr-larin/BMSTU-Graduate-Qualifying-Project/tree/main/tensorflow-serving-client-win/release>, и, если будет выдано соответствующее сообщение, обновить библиотеку времени исполнения (скачать и запустить файл WindowsAppRuntimeInstall.exe).

Graphical user interface

Description automatically generated

Рисунок 16 – Окно клиента приложения

Также имеется клиент на основе Jupyter Notebook (файл tfs\_client.ipynb), обладающий той же функциональностью, но без графического интерфейса.

# Список литературы

[1] *Ignacio Viñuales.* Battery RUL prediction using PyTorch – Режим доступа: <https://github.com/ignavinuales/Battery_RUL_Prediction> (дата обращения: 25.04.2023).

[2] *Ignacio Viñuales.* Battery Remaining Useful Life (RUL) – Режим доступа: <https://www.kaggle.com/datasets/ignaciovinuales/battery-remaining-useful-life-rul> (дата обращения: 25.04.2023).

[3] Lasso (statistics) – Режим доступа: <https://en.wikipedia.org/wiki/Lasso_(statistics)> (дата обращения: 25.04.2023).