МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

Национальный исследовательский ядерный университет «МИФИ»

Институт
интеллектуальных кибернетических систем

Кафедра кибернетики (№ 22)

Направление подготовки 09.03.04 Программная инженерия

Пояснительная записка

к научно-исследовательской работе студента на тему:

Создание модели машинного обучения для предсказания состояния работы крутильного вала в режиме реального времени

Группа М25-534

Студент _____ Ворона А.К.
(подпись) (ФИО)

Руководитель _____ Климов В.В.
(подпись) (ФИО)

Научный консультант _____ _____
(подпись) (ФИО)

Оценка руководителя _____ Оценка консультанта _____
(0-30 баллов) (0-30 баллов)

Итоговая оценка _____ ECTS _____
(0-100 баллов)

Комиссия

Председатель _____ _____
(подпись) (ФИО)

_____ _____
(подпись) (ФИО)

_____ _____
(подпись) (ФИО)

_____ _____
(подпись) (ФИО)

Москва 2026

Реферат

Пояснительная записка содержит 35 страниц, 33 рисунка, 27 источников.

Ключевые слова: *ансамблевые модели, машинное обучение, динамические модели, LGMB, диагностика турбоагрегатов.*

Данная работа заключается в исследовании современных методов обработки данных в машинном обучении для предсказания критических ситуаций на инженерном объекте.

Объектом исследования в данной работе являются исторические данные по работе крутильного вала за 2016 год, создание записей которых производилось с интервалом в 1 час.

В первом разделе приводится описание методов и моделей машинного обучения, используемые для выявления закономерностей в данных. Также приводится описание исследуемых исторических данных.

Во втором разделе дается описание алгоритмов, по которому будет выбираться модель и оцениваться результаты полученных закономерностей целевой переменной.

В третьем разделе приводятся описание архитектуры предиктивной системы и системы обучения модели в отдельности.

В четвертом разделе представлены результаты обучения модели LGBM, оценок качества, графиков и их сравнение

Оглавление

Введение

Контроль и диагностика валопроводов турбоагрегатов по крутильным колебаниям является острой проблемой на отечественных электростанциях. Данные установки являются самой важной частью тепловых электростанций, обладающие мощностью до 1200 МВт. Поддержка и ремонт необходимы для постоянной работы предприятий и выявление критических ситуаций позволят оперативно предотвращать катастрофы и сохранять жизни персонала.

В данной работе будет рассматриваться применение методов и моделей машинного обучения для предсказания таких опасных ситуаций и выявления критических значений физических параметров работы турбины.

Также в рамках данной работы будут заложены основы для создания платформы по мониторингу, архивированию и обучению моделей машинного обучения в рамках единой предиктивной системы.

1. Обзор моделей машинного обучения и методов обработки данных

1.1. Регрессионные модели

Регрессионные модели применяются для описания и анализа зависимостей между различными показателями. Они позволяют установить, каким образом изменения одних переменных отражаются на значении другой переменной, а также использовать полученную зависимость для прогнозирования. Подобные модели являются базовым инструментом анализа данных и находят применение при решении прикладных и исследовательских задач в самых разных областях [1-7]. В дальнейшем будут рассмотрены основные типы регрессионных моделей и их особенности.

1.1.1. Статические регрессионные модели

Статические регрессионные модели используются для описания зависимости между переменными в рамках одного фиксированного состояния системы. В таких моделях предполагается, что исследуемая связь не меняется со временем, а сами наблюдения не зависят друг от друга по временной оси [2]. Иными словами, каждый замер рассматривается изолированно, без учёта предшествующих или последующих значений.

Основная характеристика таких моделей: временная независимость, интуитивность модели и простота обучения. Все модели предполагают, что наблюдаемые данные являются независимыми во времени. Они не учитывают временные лаги или последовательность наблюдений.

В обучении и применении данных моделей применяют обычно следующий перечень шагов [3]:

- Сбор данных, отображающие зависимые и независимые переменные;
- Предобработка данных. Например, фильтрация, нормализация, избавление от выбросов, заполнение или удаление пропущенных значений в данных, если собранные данные содержат ошибки или недочеты;
- Обучение модели на данных с помощью метода наименьших квадратов, где параметры модели подбираются таким образом, чтобы сумма разностей предсказанных и истинных значений была минимальной
- Проверка предположений. Делается для того, чтобы мы могли оценить качество модели и чтобы эти оценки не вводили в заблуждение. Из них: линейность, отсутствие мультиколлинеарности, нормальность остатков и независимость наблюдений.

Выделяют следующие типы регрессионных моделей [3]:

- Простая линейная регрессия

Это простейший вид статической модели, который описывает зависимость между одной независимой переменной (x) и зависимой переменной (y). Математически она представляет собой поиск прямой линии, которая наилучшим образом аппроксимирует облако точек.

$y = \beta_0 + \beta_1 x + \epsilon$, где y – зависимая переменная, β_0 – свободный член (интерцепт), β_1 – коэффициент регрессии, ϵ – случайная ошибка (остаток)

- Множественная линейная регрессия

Применяется, когда на целевой показатель влияют одновременно несколько факторов. Это позволяет оценить чистый вклад каждого фактора при условии, что остальные остаются неизменными.

$y = \beta_0 + \beta_1 x + \beta_2 x_2 + \cdots + \beta_n x_n + \epsilon$

- Полиномиальная регрессия

Используется в случаях, когда связь между переменными нелинейна (например, имеет форму дуги или S-образной кривой), но модель все еще может быть представлена как линейная комбинация признаков, возведенных в степень.

- Регрессия с регуляризацией

Это улучшенные версии статической регрессии, которые применяются для борьбы с переобучением и мультиколлинеарностью. В функцию потерь добавляется штраф за слишком большие значения коэффициентов β. Выделяют 2 вида:

  - Ridge: добавляет штраф, пропорциональный квадрату коэффициентов (L_2 - регуляризация).
  - Lasso: добавляет штраф, пропорциональный модулю коэффициентов (L_1 - регуляризация). Она полезна тем, что может занулять коэффициенты при слабых признаках, фактически выполняя их отбор.

1.1.2. Динамические регрессионные модели

Динамические регрессионные модели применяются для более детального и глубокого анализа процессов и систем, развитие которых происходит во времени и характеризуется наличием временной зависимости. В таких моделях предполагается, что текущее состояние исследуемой системы формируется под влиянием не только актуальных внешних факторов,

наблюдаемых в данный момент времени, но и их значений в предыдущие моменты, а также накопленного эффекта прошлых состояний самой системы [4-5]. Иными словами, поведение объекта описывается с учётом его предыстории и инерционности, что позволяет более адекватно отражать реальные закономерности функционирования.

Основное уравнение динамической регрессии в общем виде можно представить так: $y_t = f(x_t, x_{t-1}, \dots, x_{t-k}, y_{t-1}, \dots, y_{t-p}) + \epsilon_t$, где t – индекс времени, а k и p – величины временных лагов (задержек).

Рассмотрим основные виды динамических регрессионных моделей [19-24]:

- Модели с задержками

В этих моделях предполагается, что воздействие независимой переменной x на результат y происходит не мгновенно, а распределено во времени;

$$y_t = \alpha + \sum_{i=0}^{k} \beta_i x_{t-i} + \epsilon_t$$

- Авторегрессионные модели

Эти модели учитывают, что текущее значение зависимой переменной зависит от её собственных прошлых значений. Здесь отсутствуют независимые переменные.

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \epsilon;$$

- Модели авторегрессии с распределенным лагом (ARDL)

Это комбинированный тип моделей, объединяющий влияние прошлых значений самой переменной и прошлых значений внешних факторов

$$y_t = c + \sum_{i=0}^{k} \gamma_i y_{t-i} + \epsilon_t + \sum_{j=0}^{q} \beta_j x_{t-j} + \epsilon_t$$

1.1.3. Ансамблевые модели

Ансамблевое обучение (Ensemble Learning) представляет собой парадигму машинного обучения, в которой для повышения прогностической точности и робастности системы синтезируется решение совокупности базовых моделей (гипотез) [27]. С теоретической точки зрения эффективность ансамблей обосновывается декомпозицией ошибки на смещение (Bias), разброс (Variance) и неустранимый шум (σ).

$$\text{Error} = \text{Bias}^2 + \text{Variance} + \sigma^2$$

Рассмотрим основные виды ансамблевых моделей [26-27]:

- Бэггинг (Bootstrap Aggregating)

Метод основан на генерации M независимых подвыборок с помощью процедуры бутстрапа (случайный отбор с возвращением). На каждой подвыборке обучается отдельный регрессор $h_m(x)$. Итоговая функция прогноза определяется как математическое ожидание предсказаний отдельных моделей:

$$y = \frac{1}{M}\sum_{m=1}^{M} h_m(x)$$

Классическим примером данного подхода является алгоритм Random Forest, который дополнительно использует метод случайных подпространств (random subspace method), что минимизирует корреляцию между базовыми деревьями решений.

- Бустинг (Boosting)

  Бустинг представляет собой итерационный процесс построения аддитивной модели. В отличие от бэггинга, модели обучаются последовательно, где каждый последующий регрессор $h_m(x)$ минимизирует функционал ошибки относительно текущего ансамбля.

  В градиентном бустинге (Gradient Boosting) каждая новая модель аппроксимирует антиградиент функции потерь $L(y, \hat{y})$ по отношению к предсказаниям предыдущего шага:

  $h_m(x) \approx -\left[\frac{\partial L(y, F_{m-1}(x))}{\partial F_{m-1}(x)}\right]$, где

  F_m – текущее состояние ансамбля на итерации m

  Финальная композиция имеет вид:

  $F_M(x) = F_0(x) + \sum_{m=1}^{M} \eta\gamma_m h_m(x)$, где

  η - параметр регуляризации, контролирующий темп обучения;

  γ_m - оптимальный множитель, найденный путем одномерной оптимизации.

- Стекинг (Stacking)

  Стекинг — это метод мета-обучения, при котором предсказания нескольких базовых моделей первого уровня (base learners) служат входными признаками для модели второго уровня (meta-learner).

  Математически это выражается как двухуровневая оптимизация:

  $y = g\big(h_1(x), h_2(x), \dots, h_k(x)\big)$, где

  g - мета-регрессор, который обучается минимизировать ошибку обобщения, оптимально взвешивая вклады различных алгоритмов

1.1.4. Проверка адекватности данных и спецификация признаков

При использовании непараметрических методов, таких как градиентный бустинг над деревьями решений, классические предположения Гаусса-Маркова (линейность, нормальность распределения) не являются обязательными условиями для построения состоятельного прогноза [12-14]. Однако специфика анализа временных рядов с применением лаговых переменных и скользящих окон требует верификации данных по следующим критериям:

- Стационарность временного ряда

Стационарность является фундаментальным условием для обеспечения устойчивости модели во времени. Временной ряд признается стационарным, если его математическое ожидание, дисперсия и автокорреляционная структура инвариантны относительно сдвига во времени. Примером метода проверки ряда на стационарность является тест Дики-Фуллера (ADF-test). Использование нестационарных данных при формировании лагов может привести к эффекту «ложной регрессии», когда модель находит закономерности в случайных трендах, не обладающих прогностической силой на тестовом интервале.

- Отсутствие автокорреляции остатков.

После обучения модели градиентного бустинга необходимо убедиться, что остатки (ошибки) представляют собой «белый шум». Методом проверки в данном случае является тест Льюнга-Бокса или визуальный анализ графика автокорреляции остатков. Наличие значимой автокорреляции в остатках свидетельствует о том, что выбранная структура лагов и скользящих средних не полностью детерминировала динамическую компоненту ряда, и в данных остались невыявленные закономерности.

- Отсутствие мультиколлинеарности и избыточности признаков.

Хотя ансамблевые методы на основе деревьев устойчивы к мультиколлинеарности, наличие сильно коррелированных признаков затрудняет интерпретацию значимости признаков. Методом проверки на автокорреляцию может выступать расчет коэффициента инфляции дисперсии (VIF) или анализ матрицы корреляций признаков. Исключение коллинеарных переменных позволяет стабилизировать процедуру выбора разбиений в узлах деревьев и повысить обобщающую способность ансамбля.

- Исключение эффекта «заглядывания вперед»

При формировании признаков на основе сдвигов по времени и агрегированных показателей критически важно исключить утечку данных из будущего в обучающую выборку. В отличие от независимых наблюдений в статической регрессии, данные

временных рядов имеют жесткую хронологию. Наблюдения в обучающей выборке должны предшествовать наблюдениям в валидационной выборке, тем самым исключая утечку данных целевого признака.

1.1.5. Предобработка данных для обучения

Предобработка данных — это важный этап, который существенно влияет на качество и точность работы моделей машинного обучения. Она включает в себя очистку, преобразование и подготовку данных для обучения модели [17-18].

Первым этапом является очистка данных. Этот этап направлен на устранение проблем с данными, которые могут негативно повлиять на модель. Здесь данные избавляются от неполноты и пропущенные значения заменяются на среднее, моду, медиану, интерполяцию или удаляются, если доля пропусков большая. Также, если для модели крайне неестественно появление дубликатов в данных, то их можно удалить. Для построения регрессионных моделей важны предположения о ненулевой дисперсии и отсутствии коррелированных признаков, соответственно, такие предикторы из данных тоже удаляются.

Следующий этап – удаление выбросов – некоторых записей, значение независимых переменных которой сильно отличается от общей тенденции. Такие записи могут сильно испортить результаты модели машинного обучения. Для этого применяется метод межквартильного размаха (IQR), позволяющий выделить все данные в пределах от $Q_1 - 1.5 \cdot IQR$ до $Q_3 + 1.5 \cdot IQR$. Значения вне этого диапазона считается выбросом.

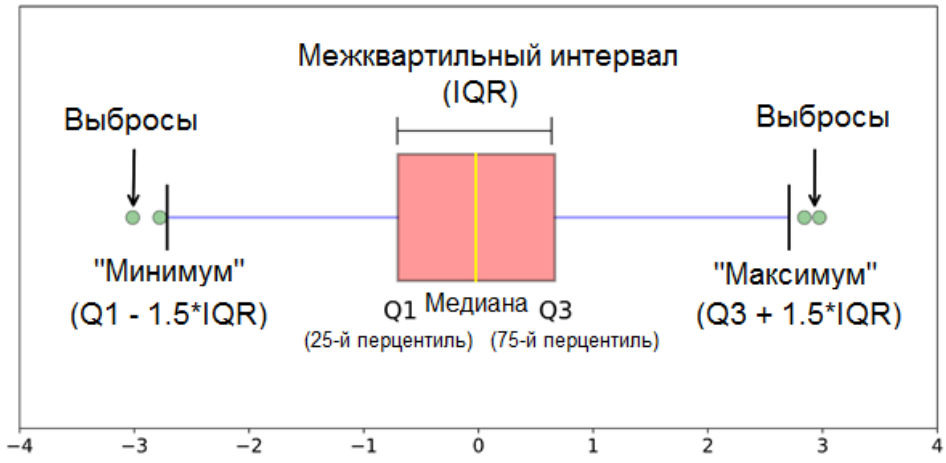

Рис. 1.1 – метод межквартильного размаха (IQR)

Также при использовании z-оценки (z-scores) у выбросов устанавливается значение больше 3. Формула для z-score следующая:

$z = \frac{X-\mu}{\sigma}$, где X – значение, μ – среднее значение выборки, σ — стандартное отклонение выборки.

Выбросы можно заменить на ближайшие краевые значения, либо просто удалить,

если выбросов не так много.

Третий этап – преобразование данных. В этом этапе мы приводим данные к определенному диапазону значений, чтобы с ними было удобней работать. Из наиболее известных методов применяются стандартное (с z-scores), minmax и робастное масштабирование.

$minmax = (X - min)/(max - min)$, где min и max – минимальное и максимальное значения выборки соответственно.

$robust = (X - Q_1)/(Q_3 - Q_1)$, где Q_1 и Q_3 первый и третий квантили соответственно.

Следующий этап предобработки – введение новых предикторов, как некоторые комбинации из ранее используемых, например, введение полиномов. Важно после введения новых предикторов проверить на автокорреляцию.

После всех этих этапов данные готовы к работе и в следующем этапе проходит разбиение данных на тестовую и обучающую выборки. Это делается для того, чтобы обучить модель на одних данных и проверить обобщающие возможности и оценить результаты модели на других. Выделяют несколько стратегий для этого[1]:

- Repeated random subsampling CV (Monte-Carlo CV)

Случайное разбиение на тестовые и обучающие выборки k раз

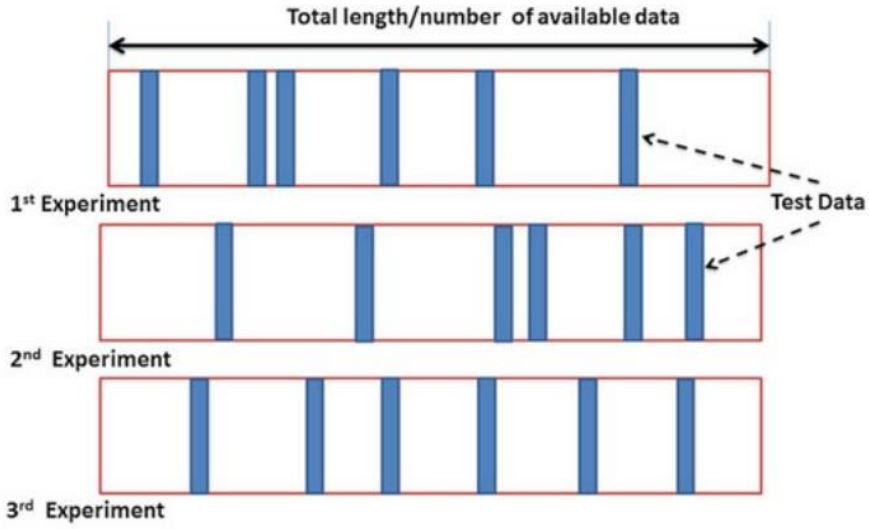

Рис. 1.2 – случайное разбиение кросс-валидации [1]

- k-fold CV

Разбиение на k равных частей. В каждом эксперименте одна из секций выбирается как тестовая выборка. Каждая секция выбирается ровно один раз и только в одном эксперименте.

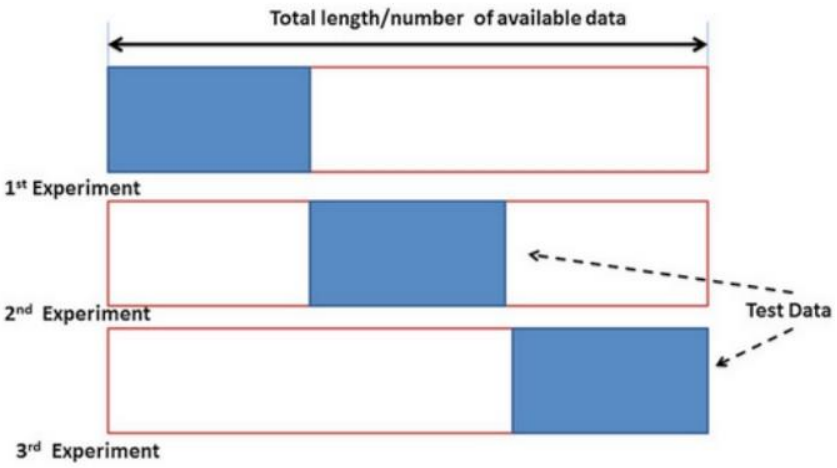

Рис. 1.3 – k-fold валидация [1]

- Leave-one-out CV (LOOCV)

Частный случай k-fold, где k равно размеру всей выборки

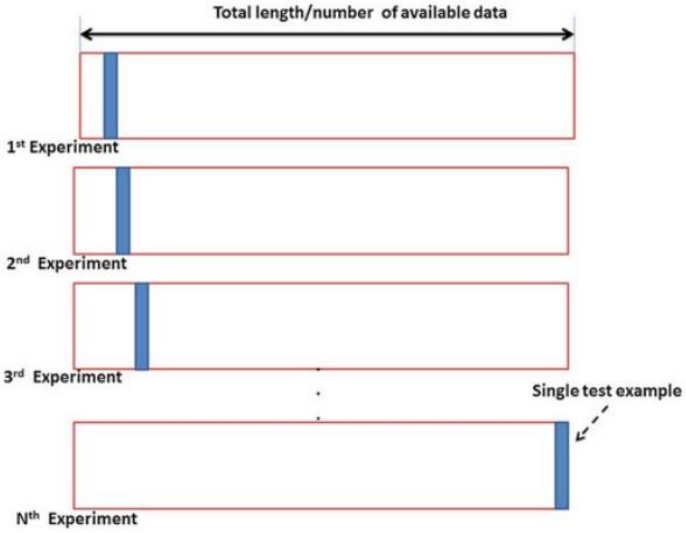

Рис. 1.4 – Leave-one-out кросс-валидация [1]

- Holdout

Самый частый и простой вариант. Вся выборка делится только на две части: тестовая и обучающая, и проводится только один эксперимент.

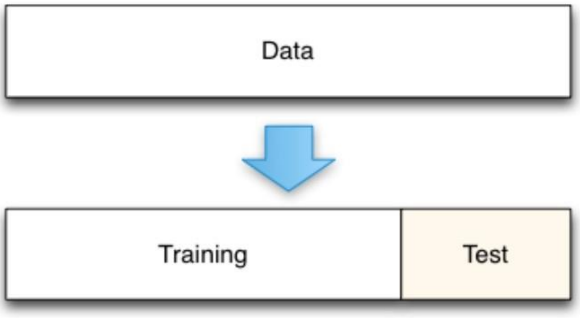

Рис. 1.5 – Holdout кросс-валидация [1]

- Resubstitution

Никак не разбивает выборку. Использует те же самые данные и для обучения и для валидации.

Таким образом, выбор стратегии разбиения данных напрямую зависит от характера задачи и структуры исходных данных. В рамках данной работы будет использоваться стратегия Holdout, поскольку анализ проводится на временных рядах. Для таких данных принципиально важно сохранять хронологический порядок наблюдений, а методы кросс-валидации со случайным или переставляемым разбиением (k-fold, Monte-Carlo CV и LOOCV) могут приводить к перемешиванию временной последовательности и, как следствие, к утечке информации из будущего в прошлое. Использование Holdout-разбиения позволяет корректно отделить обучающую и тестовую выборки по времени и получить более достоверную оценку обобщающей способности модели.

1.2. Описание исторических данных

Исторические данные [28] представлены в следующем виде:

Дата МоДеРо

Ед. изм	Гц	МВт	МВАр	oC	oC	oC	oC	oC	oC	oC	oC	oC	oC	oC	oC	oC	Гц
Код сигнала	F1	Pa	Pr	Td	Tg	Tst	Тп. ЦНД-1	Тп.ЦНД-2	Т ГПЗ н.А	Т ГПЗ н.Б	Т ГПЗ н.В	Т ГПЗ н.Г	Т турб. н.А	Т турб.н.Б	Т турб.н.В	Т турб.н.Г	Fr
01/01/2016 00:00:00	19,2824000	34,753082	-90,211281	37,487492	31,800579	45,405594	9,0330	12,2560	539,2820	540,7470	541,5530	545,2150	548,9500	548,6570	545,2880	545,8010	49,995281
01/01/2016 01:00:00	19,2839030	34,220016	-86,393089	37,478737	31,870878	45,505035	8,9360	12,2560	539,2090	540,2340	541,5530	544,4820	550,5620	549,6830	545,8740	546,8260	50,005314
01/01/2016 02:00:00	19,2824000	34,415543	-83,297874	37,238052	31,816473	45,399033	9,0330	12,2800	541,9920	541,9920	541,3330	545,5810	553,4910	552,2460	549,1700	549,2430	50,002312
01/01/2016 03:00:00	19,2861560	34,487137	-90,895683	37,348053	31,711052	45,473736	9,0820	12,2560	539,8680	540,7470	541,6990	543,9700	552,6120	551,7330	548,3640	548,4380	50,00153
01/01/2016 04:00:00	19,2839030	32,702866	-95,911263	37,513912	31,732071	45,622948	9,0820	12,2560	537,9640	538,9890	540,4540	542,4320	552,9790	551,4400	548,8040	549,0230	50,022942

Рис. 1.6 – структура исторических данных

Отсюда следует, что все независимые переменные представляют собой действительные числа, за исключением лишь даты. Все признаки представляют собой физические параметры работы турбоагрегата. В основном это давление, температуры (в цилиндре низкого давления, в главной паровой задвижке, на медной обмотке и т.д.) и частоты. F1 – зависимая переменная, для предсказания которой нужно построить модель. Всего в источнике 6679 записей и 16 режимных параметров.

1.3. Задачи НИР

В данной работе ставится цель достичь следующих результатов:

- Применить методы машинного обучения для предсказания целевого параметра, применяя исторические данные работы турбоагрегата;
- Предоставить оценки полученных моделей, провести шаги по возможному улучшению результатов;
- Разработать архитектуру предиктивной системы, в рамках которой будет встроена модель машинного обучения и система мониторинга.

2. Алгоритмы по работе с данными

2.1. Обоснование выбора модели LGBM

Для решения задачи прогнозирования физических параметров валопровода турбоагрегата в качестве основного алгоритма выбрана модель LightGBM (Light Gradient Boosting Machine, далее - LGBM). Данный выбор обусловлен спецификой промышленных данных (высокая размерность и необходимость работы в режиме реального времени) и архитектурными преимуществами градиентного бустинга.

Основные преимущества LGBM для диагностики турбоагрегатов [24, 25]:

- Высокая скорость обучения и производительность

В отличие от классических реализаций градиентного бустинга, LightGBM использует метод GOSS (Gradient-based One-Side Sampling). Данный метод позволяет концентрироваться на экземплярах данных с большими градиентами (ошибками), что значительно ускоряет обучение без существенной потери точности. Для мониторинга турбоагрегатов это позволяет оперативно переобучать модель при изменении режимов работы.

- Робастность к выбросам и пропускам

В промышленных системах сбора данных часто встречаются сбои датчиков и аномальные скачки (выбросы). Алгоритмы на основе решающих деревьев, к которым относится LGBM, устойчивы к масштабу признаков и способны эффективно обрабатывать пропущенные значения без необходимости сложной импутации.

- Работа с временными зависимостями через лаговые переменные

Несмотря на то, что LGBM не является специализированной моделью временных рядов (как ARMA), она эффективно выявляет нелинейные зависимости в динамических системах. При правильной подготовке данных (формирование лагов y_{t-1}, y_{t-n} и скользящих окон) модель способна улавливать инерционность тепловых и механических процессов в турбине лучше, чем линейные статистические модели.

Таким образом, LGBM является оптимальным балансом между точностью прогноза и скоростью работы предиктивной системы, позволяя своевременно диагностировать опасные состояния турбоагрегата и предотвращать аварийные ситуации на электростанциях.

2.2. Алгоритм предобработки данных

Перед обучением модели LightGBM выполняется подготовка данных и формирование признакового пространства. Этот процесс включает несколько последовательных этапов:

- Подготовка исходного временного ряда

Данные с турбины преобразуются в единый формат с временной привязкой: создаётся колонка с датой и временем, рассчитываются индикаторы режима работы и выполняется первичная обработка пропусков.

- Формирование временных признаков

Для учёта циклической природы временного ряда создаются признаки, отражающие время суток, день недели, день месяца и месяц. Эти признаки кодируются с помощью синусо-косинусных преобразований, чтобы модель могла улавливать циклические закономерности (например, суточные колебания).

- Лаговые признаки

Для каждого временного шага рассчитываются значения ряда с предыдущих интервалов. Лаги создаются как для ближайших шагов (1–24 часа), так и для более отдалённых (72, 168 часов), что позволяет модели учитывать краткосрочную и долгосрочную зависимость в данных.

- Скользящие статистики

На основе лагов вычисляются агрегированные характеристики временного ряда: скользящее среднее, медиана, стандартное отклонение, минимум, максимум и сумма для окон разного размера (6, 12, 24, 72 часа); экспоненциальное сглаживание с разными коэффициентами для учёта более свежих наблюдений с повышенным весом.

- Разности лагов и полиномиальные признаки

Для выявления изменений между соседними лагами создаются дельты $\delta(lag_i - lag_{i-1})$. Кроме того, формируются полиномиальные признаки второго порядка на основе первых лагов и дельт, что позволяет модели учитывать нелинейные взаимодействия между предыдущими значениями ряда.

- Обучение модели

После формирования полного признакового пространства данные делятся на обучающую, валидационную и тестовую выборки. Модель LightGBM, использующая градиентный бустинг деревьев решений, обучается на обучающей части, а её гиперпараметры подбираются с помощью библиотеки optuna. Процесс подбора включает нахождение оптимальных параметров таких как: глубина деревьев, число листьев, силы регуляризации и других параметров, что позволяет минимизировать ошибку прогноза.

Эти этапы отражены на рисунке 2.1:

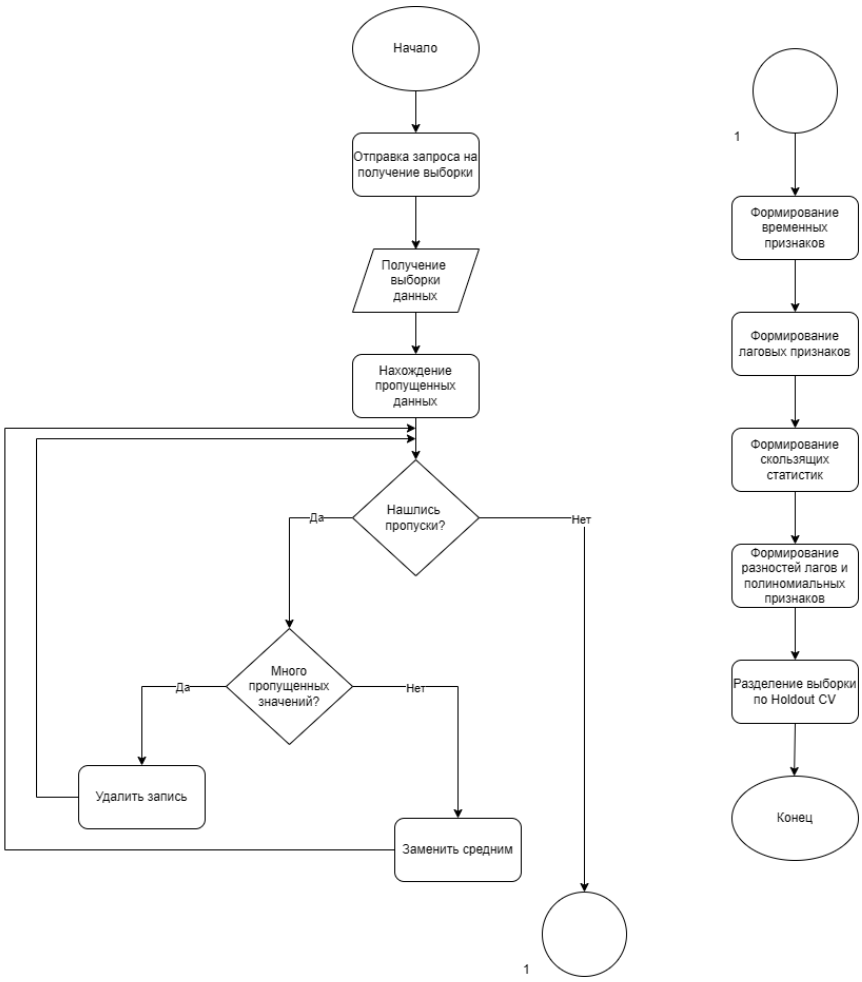

Рис. 2.1 – алгоритм предобработки данных

2.3. Алгоритм обучения модели в реальном времени

Для нашей модели не менее важно поддерживать в модели актуальное состояние турбоагрегата. Для этого ее надо постоянно переобучать с применением последних данных, получаемых с датчиков турбины. Так как система ограничена по времени и не может постоянно переобучаться каждую секунду по актуальным данным, то необходимо оптимизировать частоту обучения модели. Для этого был разработан алгоритм обучения модели LGBM в режиме реального времени, блок-схема которого изображена на рисунке 2.2.

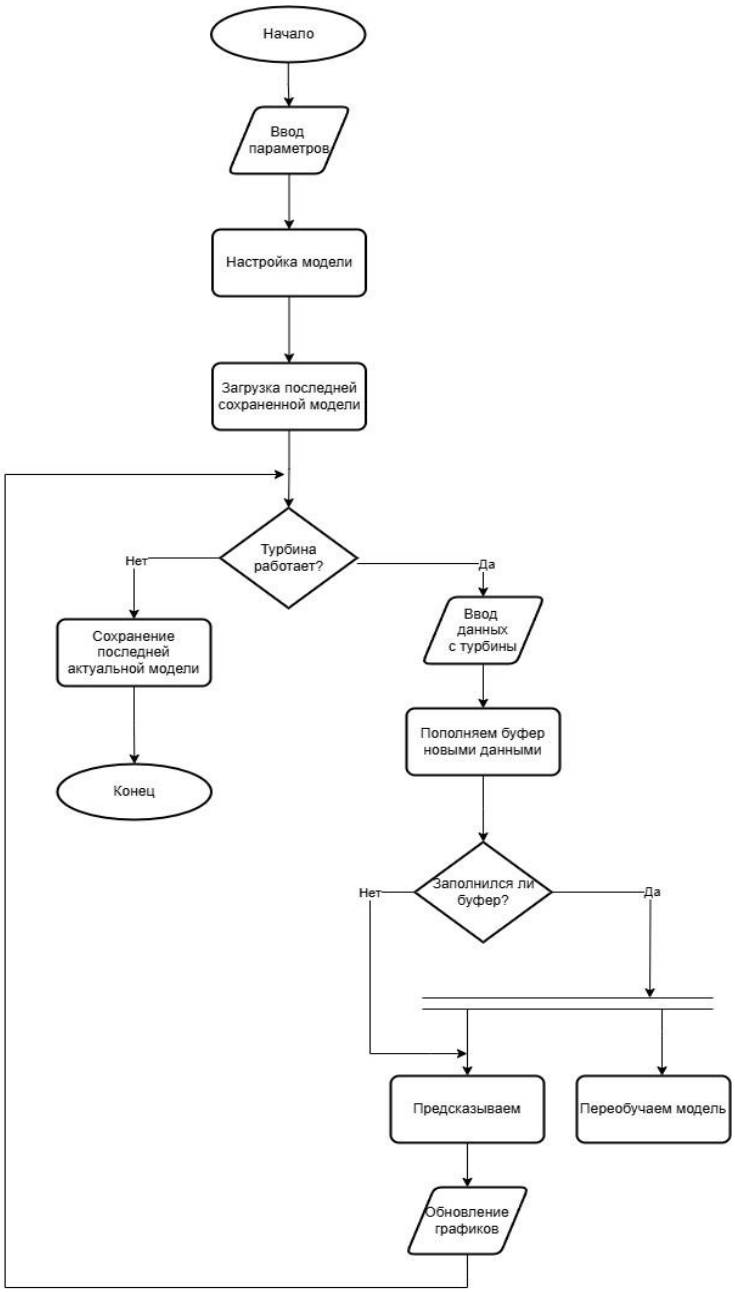

Рис. 2.2 – алгоритм обучения модели

2.4. Выводы

- Выбрана модель LGBM, способная эффективно прогнозировать параметры валопровода турбоагрегата с высокой скоростью обучения, устойчивостью к выбросам и пропускам, а также учитывать временные зависимости через лаговые признаки.
- Разработан алгоритм переобучения модели в режиме реального времени с оптимизацией частоты обновления, обеспечивающий актуальность прогнозов и возможность интеграции в систему мониторинга состояния турбины.

3. Проектирование предиктивной системы

3.1. Общая предиктивная система

Предиктивная система будет состоять из двух подсистем: системы поддержки БД и ML-оболочкой.

Система поддержки БД будет проводить мероприятия по сохранению данных, получаемых от турбины в режиме реального времени, обработки результатов моделей ML-оболочки и запросов пользователей на представление графиков аналитики.

ML-оболочка будет получать данные от БД и турбины в режиме реального времени, отправляя полученные значения и статистику оценки качества в БД и/или пользователю. Также модель будет обучаться на данных из БД и обучаться на данных от турбины. На рисунке 3.1 представлена архитектура предиктивной системы.

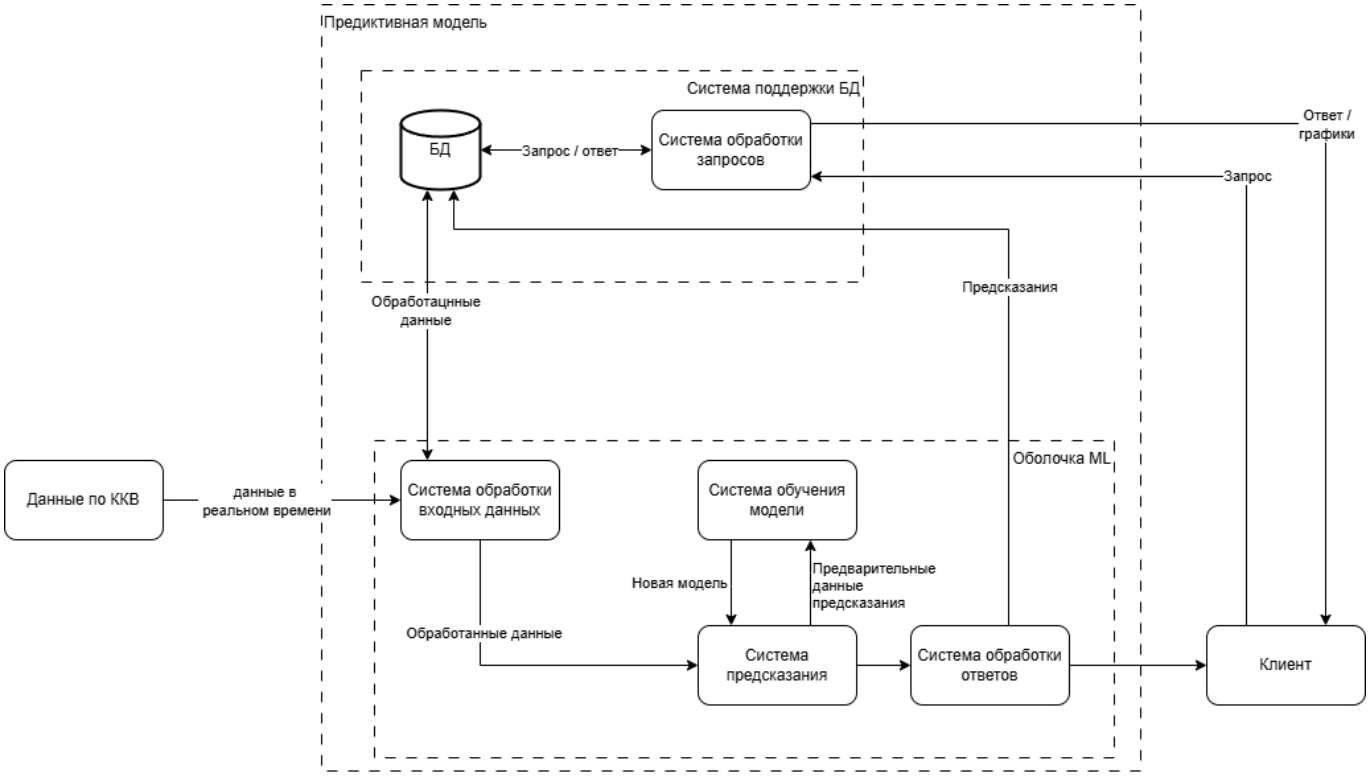

Рис. 3.1 – Архитектура общей модели предиктивной системы

3.2. Система обучения модели

Прежде чем модель может быть использована в составе предиктивной системы, необходимо выполнить её обучение и оценку качества прогнозирования. Основное внимание в данной работе уделяется описанию процесса взаимодействия модели машинного обучения с данными, поступающими от датчиков турбины, а также этапам подготовки данных, обучения модели и анализа результатов. На рисунке 3.2 представлена схема обучающего контура ML-модели, иллюстрирующая последовательность обработки данных и использование модели на этапе обучения и валидации.

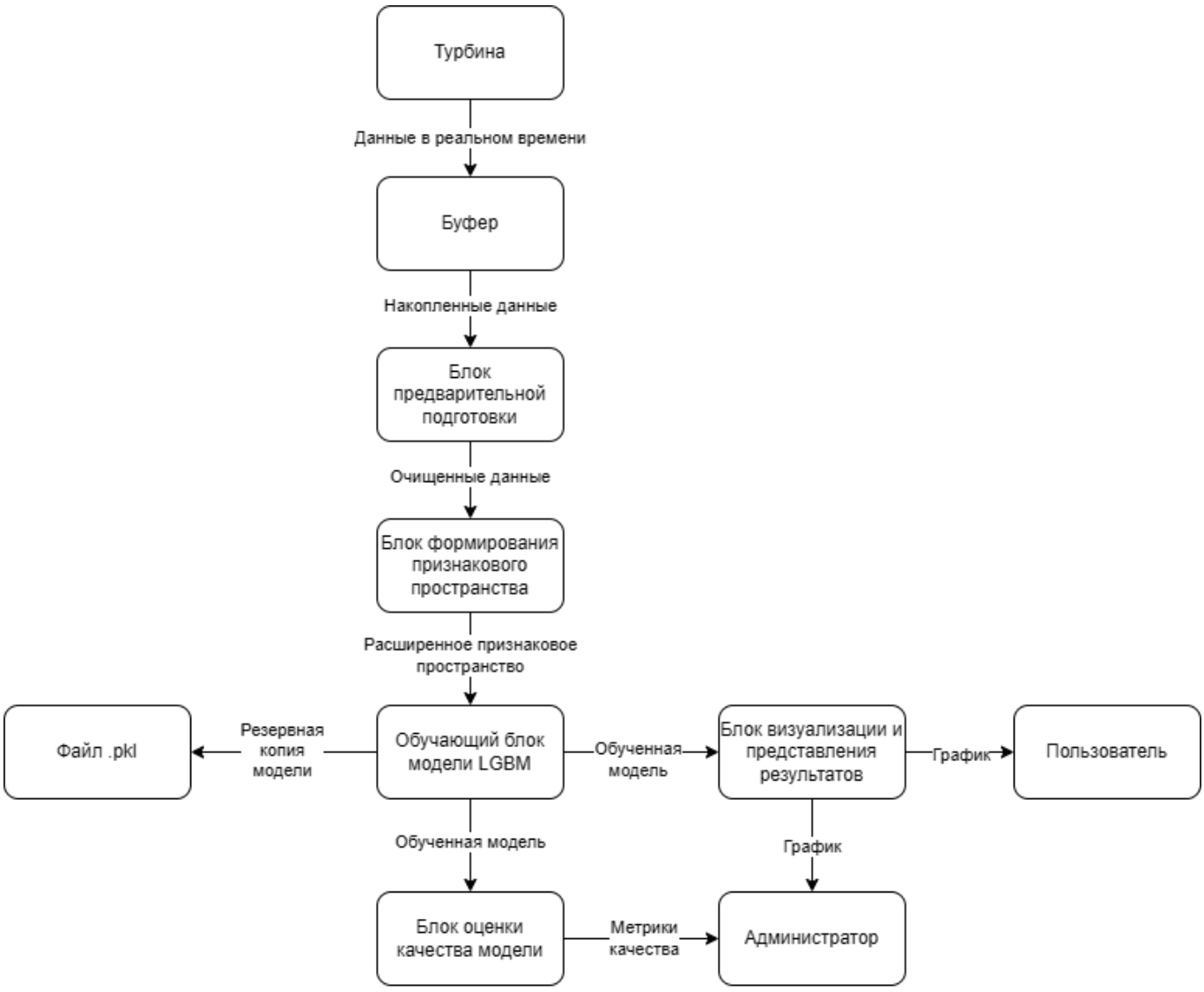

Рис. 3.2 – система обучения модели

3.3. Применяемые программные средства

В области ML крайне популярен язык Python и это неудивительно, ведь этот язык позволяет удобно работать с данными, в нем есть множество библиотек по работе с данными, он очень популярен, у него есть множество руководств и синтаксис языка довольно прост. К тому же в нем есть множество библиотек по работе с API и БД, что упростит в будущем задачу встраивания модели машинного обучения в общую предиктивную систему.

Но помимо самого языка, нам понадобятся и библиотеки, о которых говорилось ранее. Из них нам понадобятся следующие библиотеки:

- scipy — библиотека Python для научных вычислений. В нем содержатся различные метрики для проверки гипотез;
- scikit-learn — это одна из самых популярных библиотек для машинного обучения на

языке Python. Она предоставляет простой и эффективный инструментарий для анализа данных и моделирования. В нем содержится все для предобработки и анализа данных: CV методы, регрессионные модели, регуляризации, методы нормализации данных, функция расчета различных метрик;

- lightgbm — высокоэффективная библиотека для градиентного бустинга над деревьями решений, разработанная Microsoft. Оптимизирована для работы с большими объёмами данных и высокоразмерными признаковыми пространствами. Использует гистограммный алгоритм и leaf-wise рост деревьев, что обеспечивает высокую скорость обучения и хорошее качество моделей.
- optuna — библиотека для автоматического подбора гиперпараметров моделей машинного обучения. Позволяет эффективно искать оптимальные значения параметров с использованием методов байесовской оптимизации и продвинутых стратегий прунинга. Поддерживает интеграцию с популярными библиотеками ML, включая LightGBM, что позволяет ускорить обучение моделей и повысить их качество.
- pandas — программная библиотека на языке Python для обработки и анализа данных. Предоставляет специальные структуры данных и операции для манипулирования числовыми таблицами и временными рядами. Для нас будет удобно использовать датафреймы из этой библиотеки, как основную структуру данных для хранения выборки;
- matplotlib — это библиотека с открытым исходным кодом для визуализации данных в Python. Она позволяет создавать разнообразные графики и диаграммы, которые помогают лучше понять и интерпретировать данные. Графиками из этой библиотеки удобнее пользоваться в интерактивных ноутбуках Jupyter, они полезны для предварительных оценок модели на визуальном уровне;

3.4. Выводы

Были достигнуты следующие цели:

- Спроектирована общая модель, в которой будут взаимосвязаны 2 подсистемы: система поддержки БД и оболочка-ML с поддержкой предсказания в режиме реального времени;
- Представлена система обучения, в которой данные загружаются в модель машинного обучения в режиме реального времени;
- Выделены требования к программной реализации.

4. Оценки предсказаний моделей и графики

4.1. Описание распределения данных

Построим графики зависимостей зависимой переменной от начальных предикторов, чтобы выявить возможные зависимости.

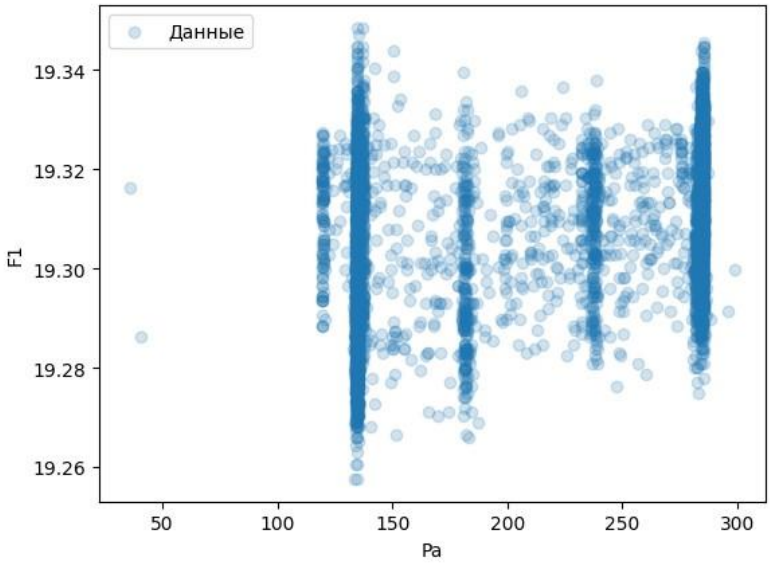

Рис. 4.1 – зависимость F1 от Pa

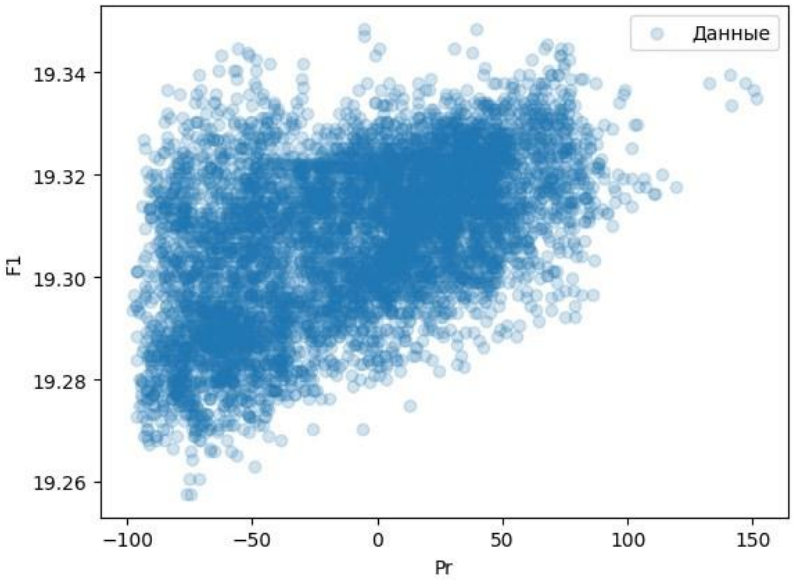

Рис. 4.2 – зависимость F1 от Pr

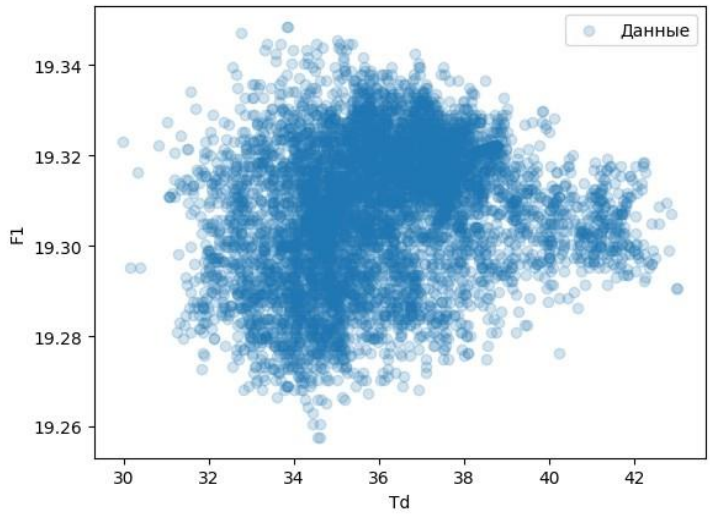

Рис. 4.3 – зависимость F1 от Td

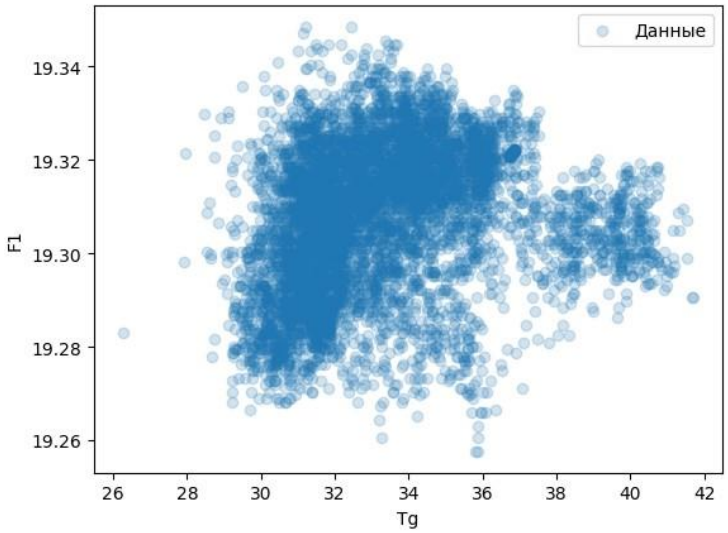

Рис. 4.4 – зависимость F1 от Tg

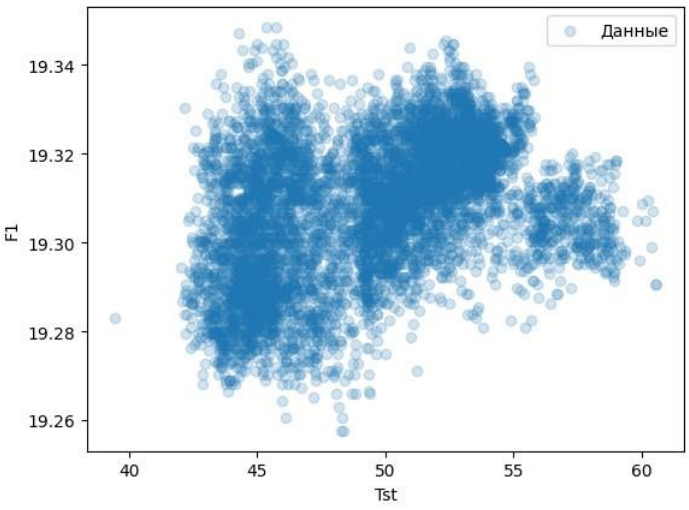

Рис. 4.5 – зависимость F1 от Tst

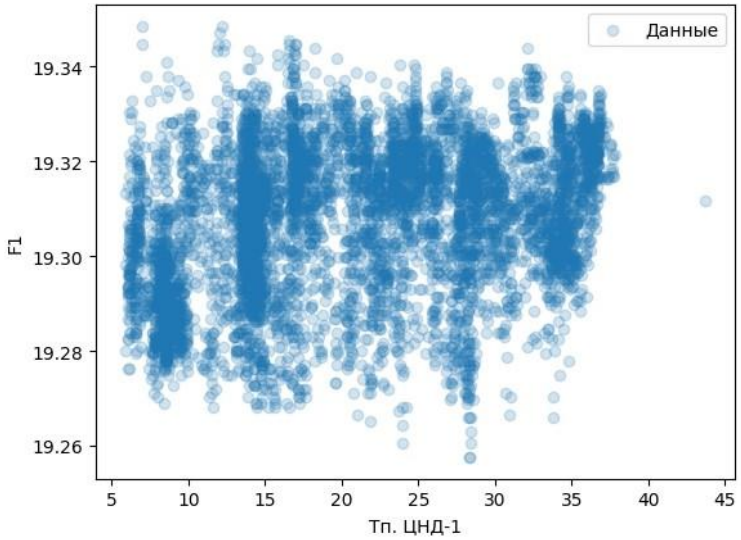

Рис. 4.6 – зависимость F1 от Тп. ЦНД-1

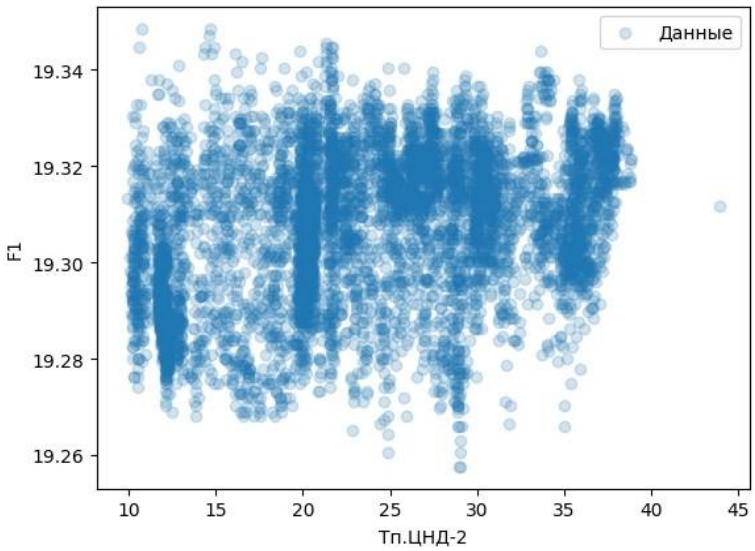

Рис. 4.7 – зависимость F1 от Тп. ЦНД-2

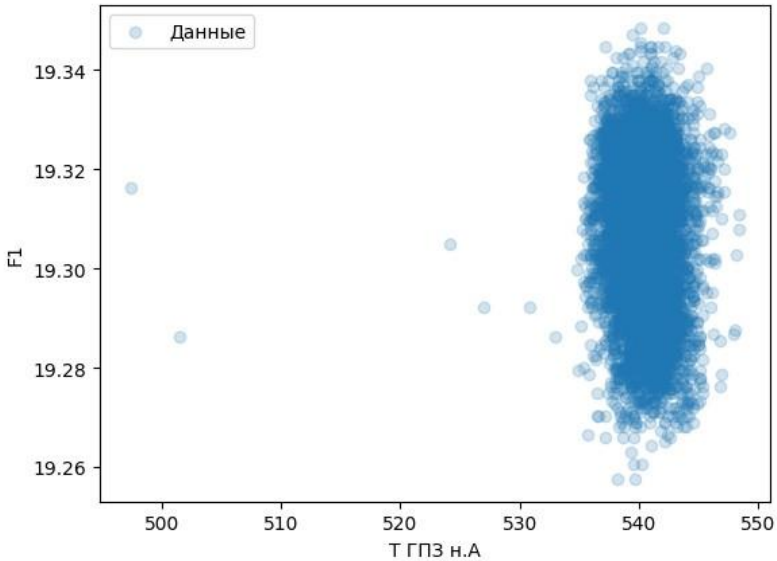

Рис. 4.8 – зависимость F1 от Т ГПЗ н.А

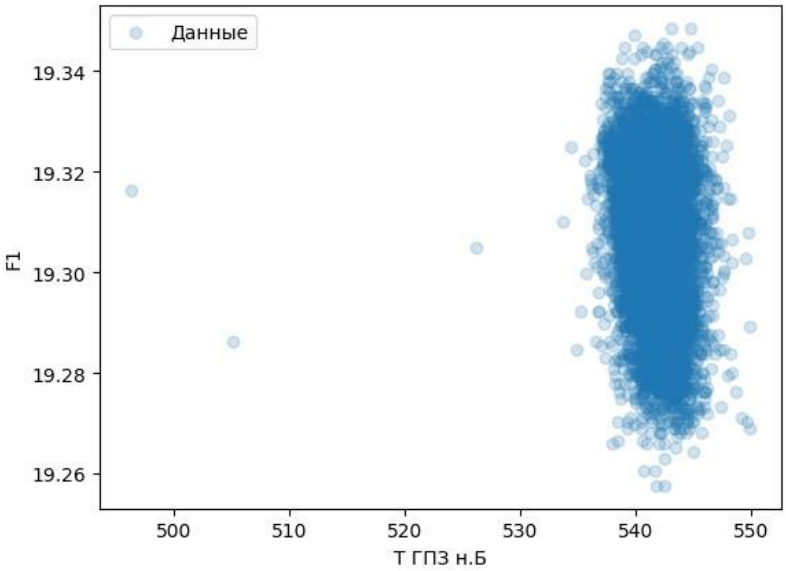

Рис. 4.9 – зависимость F1 от Т ГПЗ н.Б

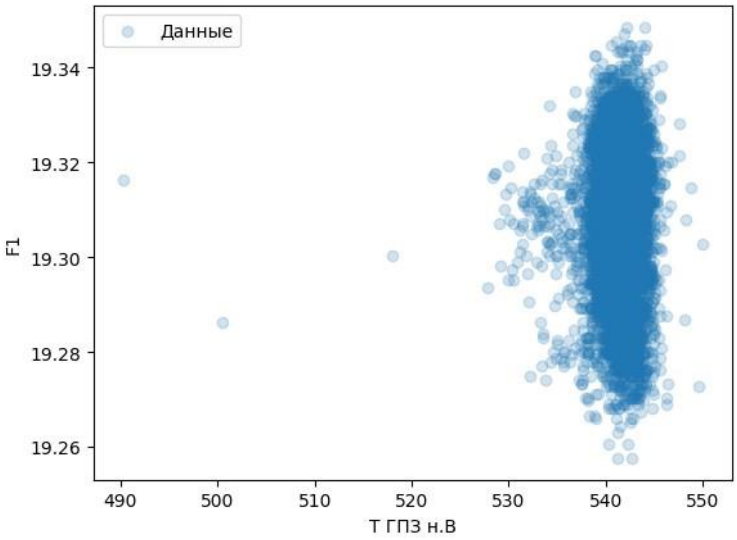

Рис. 4.10 – зависимость F1 от Т ГПЗ н.В

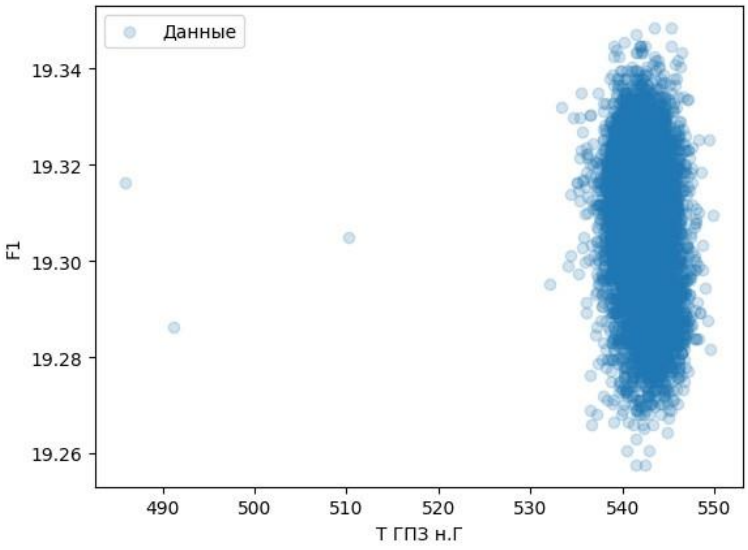

Рис. 4.11 – зависимость F1 от Т ГПЗ н.Г

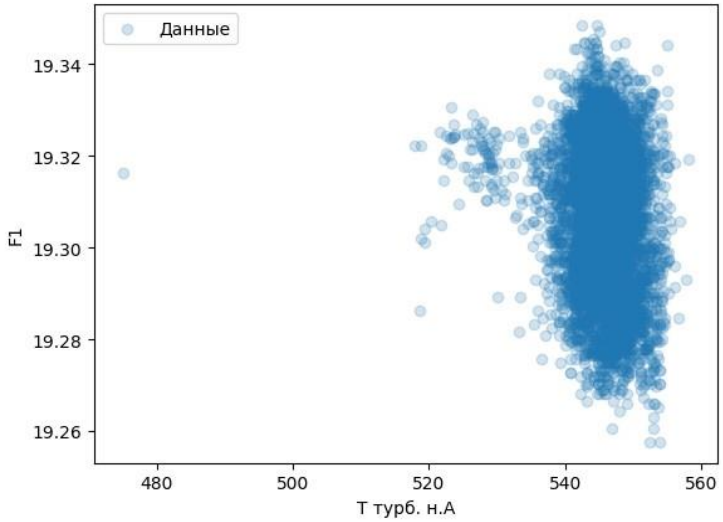

Рис. 4.12 – зависимость F1 от T турб. н.А

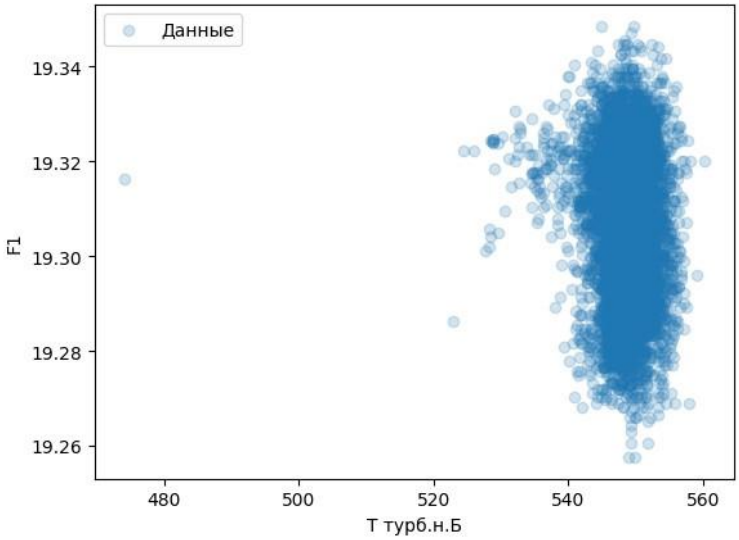

Рис. 4.13 – зависимость F1 от T турб. н.Б

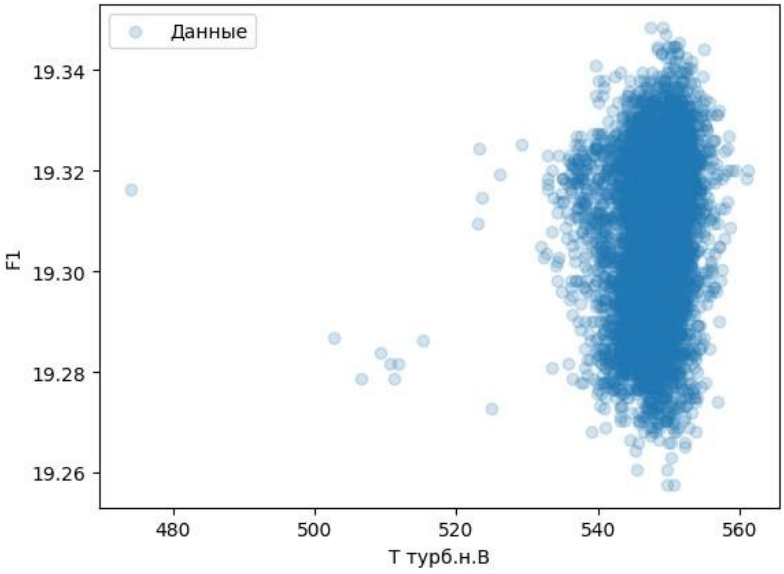

Рис. 4.14 – зависимость F1 от T турб. н.В

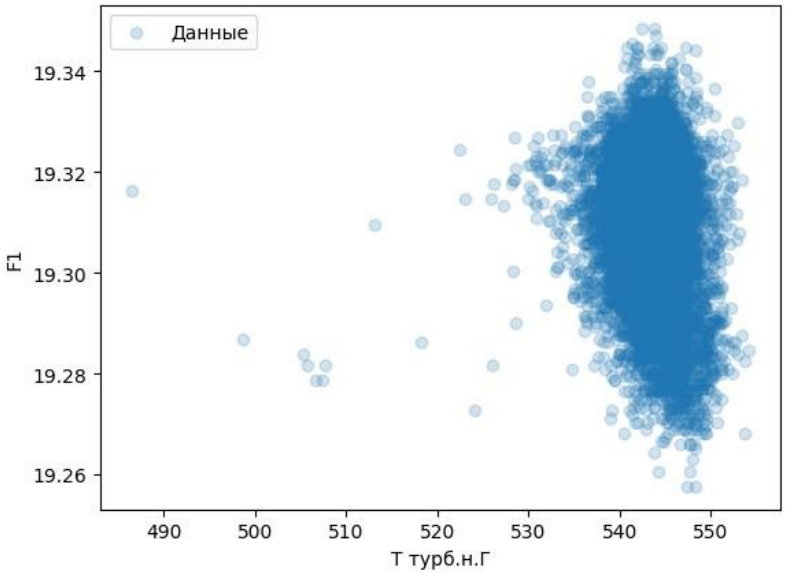

Рис. 4.15 – зависимость F1 от T турб. н.Г

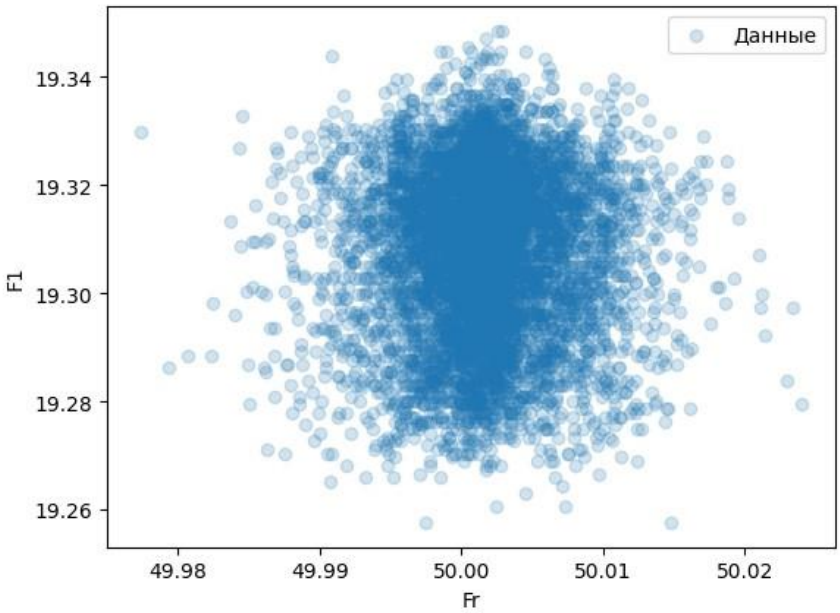

Рис. 4.16 – зависимость F1 от Fr

Можно заметить, что значения F1 колеблются не слишком сильно и природа зависимостей не везде линейна.

Проведём анализ временного ряда F1 с целью изучения его тренда и сезонности. Построим графики зависимости показателя от времени и разложим ряд на составляющие с периодом 24 часа, чтобы визуально оценить долгосрочные изменения параметра и повторяющиеся суточные колебания. Это позволит лучше понять динамику работы турбины и подготовить данные для последующего прогнозирования.

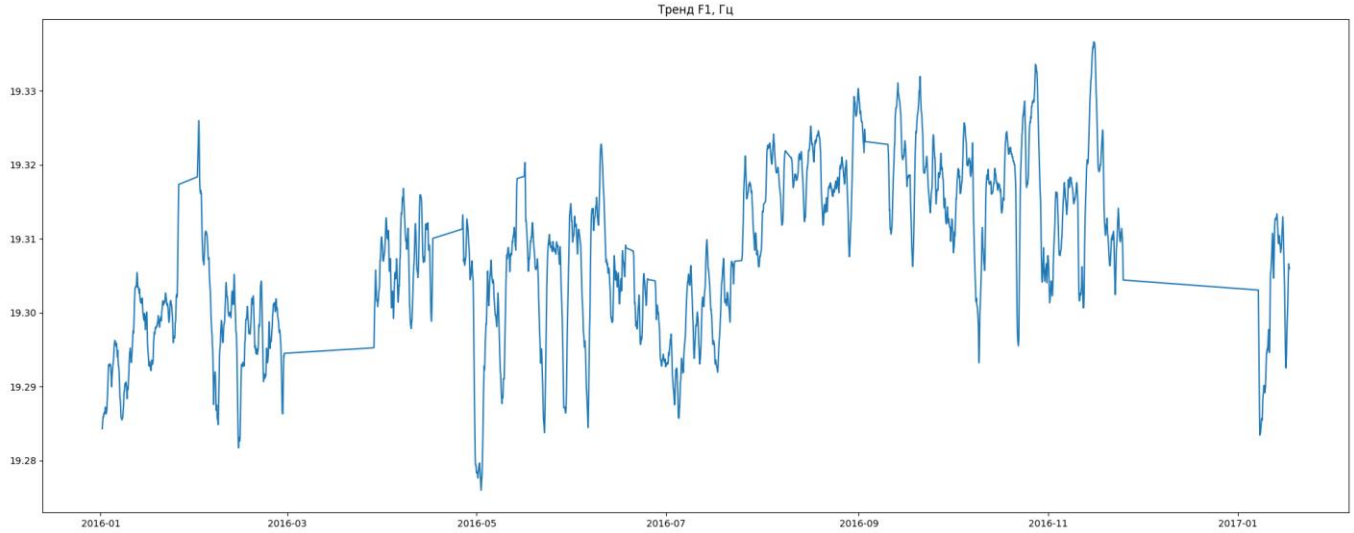

Рис. 4.17 – Тренд F1 (24ч)

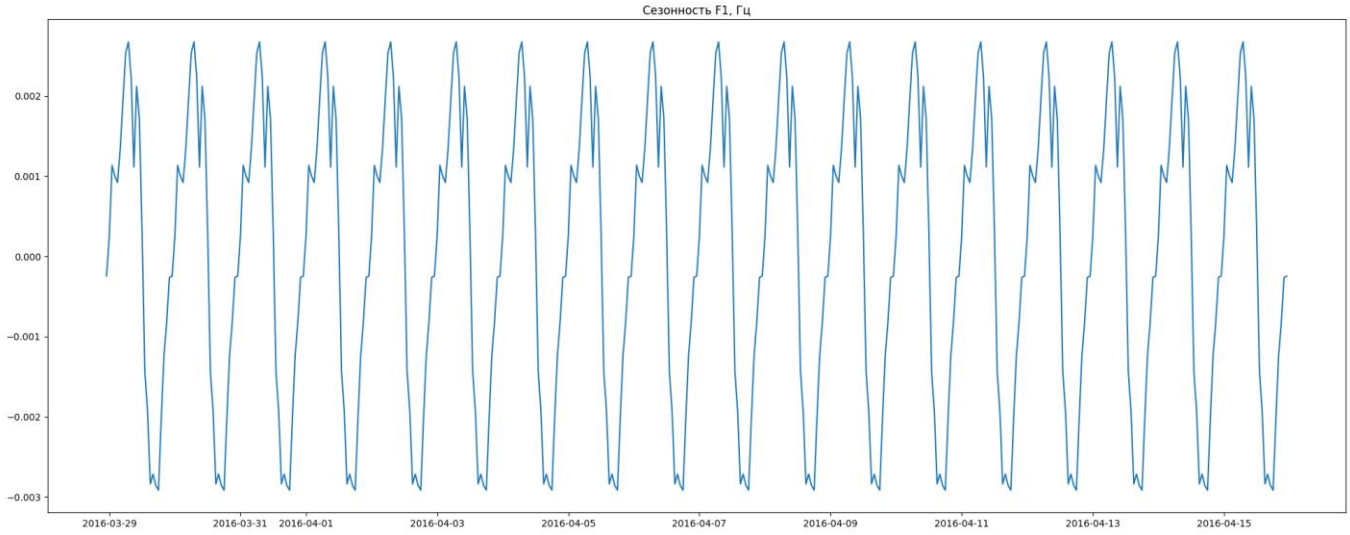

Рис. 4.18 – Сезонность F1 (24ч)

Судя по графикам, выраженного долгосрочного тренда в ряде практически не наблюдается, тогда как четко проявляется 24-часовая сезонность, отражающая регулярные суточные колебания параметра F1.

Для более детального анализа временной структуры исходного ряда и выявления зависимостей между текущими и прошлыми значениями показателя были построены графики полной автокорреляционной функции (ACF) и частичной автокорреляционной функции (PACF). Эти графики позволяют оценить характер временной зависимости, определить наличие сезонных компонентов и обосновать выбор лагов и временных окон, используемых далее при формировании признаков и построении прогностической модели

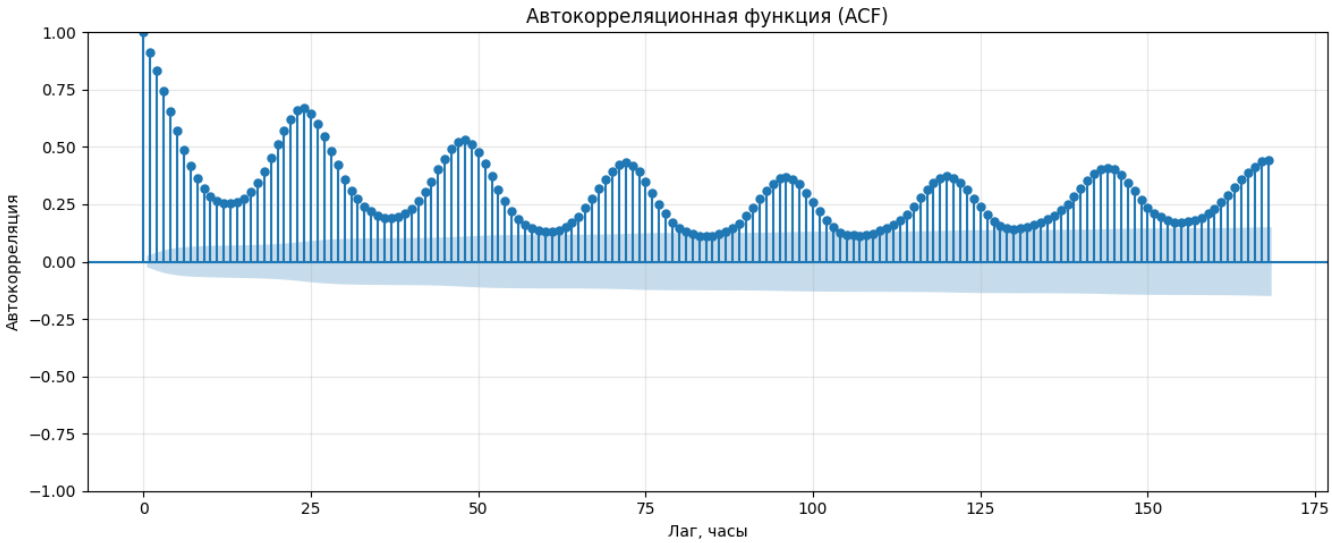

Рис. 4.19 – Автокорелляционная функция за 1 неделю

Рис. 4.20 – Частичная автокорелляционная функция за 1 неделю

На графике частичной автокорреляции в пределах одной недели наибольшее значение наблюдается на первом лаге, что указывает на сильную зависимость текущего значения показателя от значения на предыдущем временном шаге. Это подтверждает обоснованность использования краткосрочных лаговых признаков при построении прогностической модели. В то же время график автокорреляционной функции демонстрирует выраженные пики, повторяющиеся с периодом 24 часа, с постепенным затуханием амплитуды по мере увеличения лага. Такое поведение характерно для временных рядов с ярко выраженной суточной сезонностью и подтверждает наличие регулярных циклических колебаний в данных.

4.3. Результаты предсказаний

Ниже представлены графики, иллюстрирующие качество прогнозирования крутильных колебаний вала двигателя на тестовой выборке и позволяющие наглядно оценить соответствие предсказаний модели фактическим значениям временного ряда.

MAPE: 0.02344%.

MAE: 0.00453.

MSE: 0.00004.

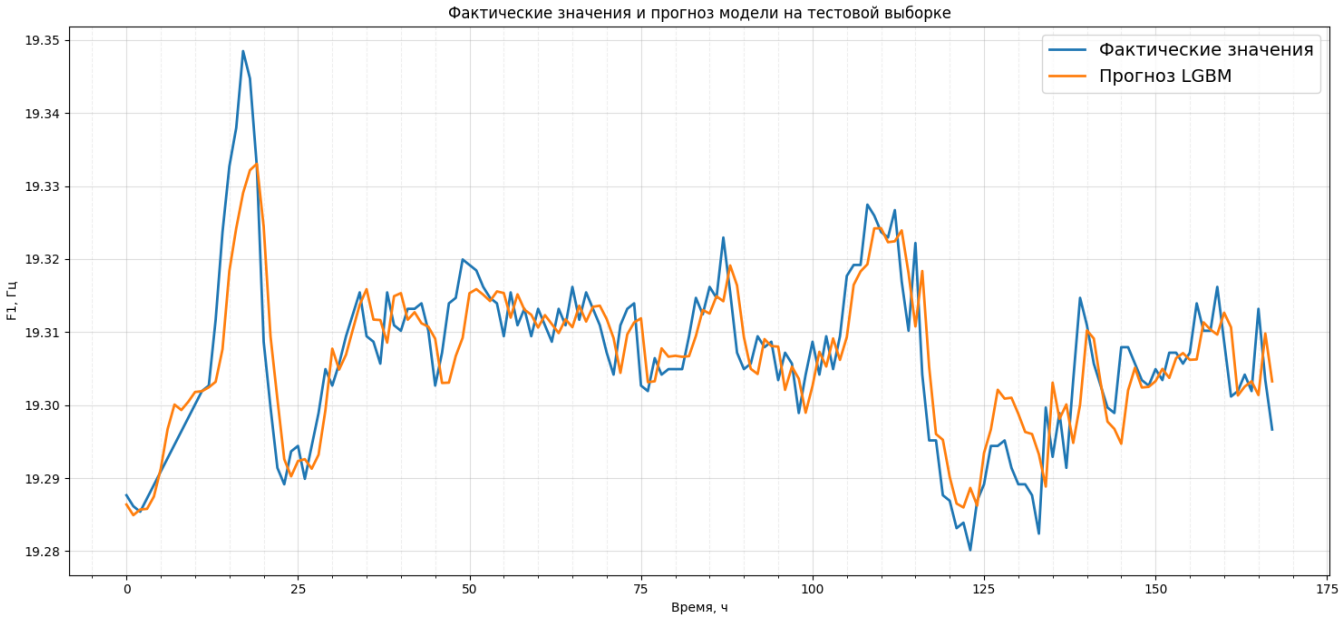

Рис. 4.21 – сравнение показаний модели и фактических значений на одну неделю

На представленном графике видно, что прогнозируемые значения модели в целом хорошо согласуются с фактическими данными на рассматриваемом недельном интервале. Модель корректно воспроизводит общую динамику временного ряда и основные колебания показателя, а расхождения между прогнозом и реальными значениями носят ограниченный характер и не имеют выраженного систематического смещения.

В результате автоматического подбора гиперпараметров с использованием библиотеки optuna была получена конфигурация модели, обеспечивающая наилучшее качество прогнозирования на тестовой выборке. Ниже в таблице рассмотрены наиболее значимые параметры, их оптимальные значения и их интерпретация

Таблица 1. Оптимальные гиперпараметры модели

Гиперпараметр	Оптимальное значение	Интерпретация и влияние на модель
Тип бустинга	goss	Используется Gradient-based One-Side Sampling
Количество деревьев	2500	Большое число деревьев обеспечивает более точную аппроксимацию сложных зависимостей

Гиперпараметр	Оптимальное значение	Интерпретация и влияние на модель
Скорость обучения	0.0158	Низкая скорость обучения способствует стабильности модели и снижает риск переобучения
Максимальная глубина дерева	6	Ограничивает глубину деревьев, предотвращая чрезмерную сложность модели
Максимально допустимое число листьев в дереве	34	Определяет число листьев в дереве; значение согласовано с глубиной и обеспечивает баланс между гибкостью и обобщающей способностью
Коэффициент L2 - регуляризации	4.57	L2-регуляризация, снижающая влияние шумных и избыточных признаков
Доля объектов, используемых для построения каждого дерева	0.80	Случайная подвыборка наблюдений повышает устойчивость модели и уменьшает переобучение
Доля признаков, используемых для построения каждого дерева	0.52	Использование части признаков при построении каждого дерева снижает влияние коррелированных фичей

Также была проведена оценка важности признаков (feature importance) модели. Для наглядности построен столбчатый график, на котором показаны топ-5 наиболее значимых признаков. Наибольшую важность показал признак, содержащий значение первого лага временного ряда, что в целом соответствует ожиданиям и подтверждает влияние недавних значений на предсказания модели.

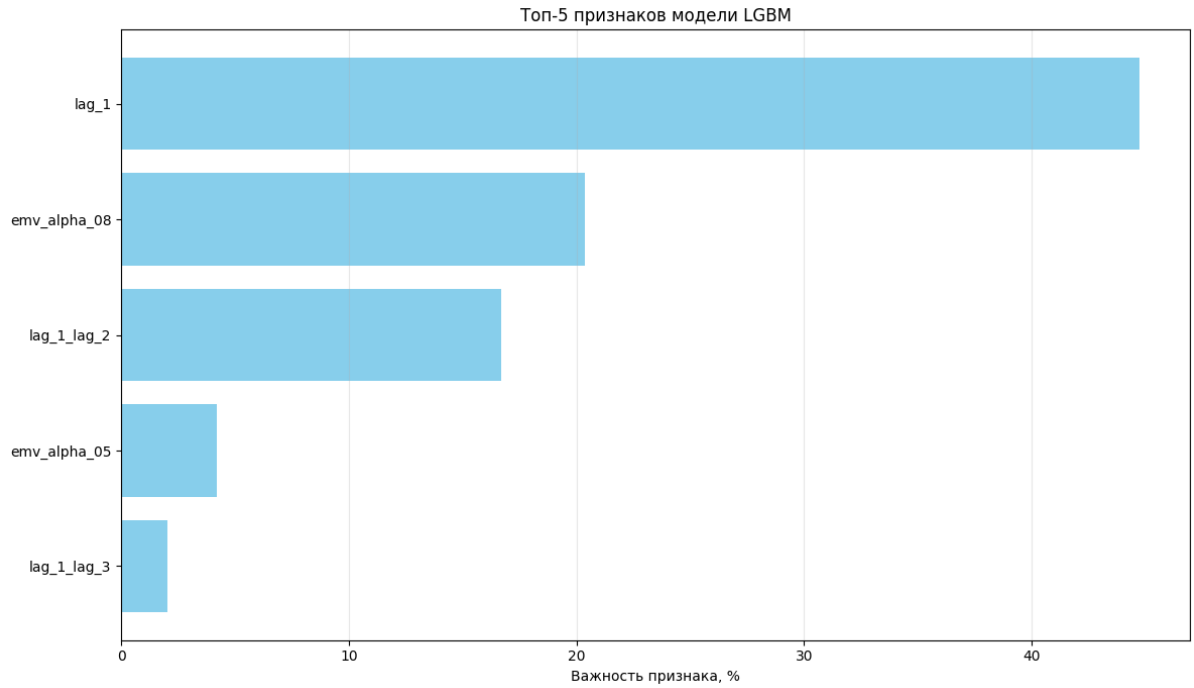

Рис. 4.22 – Важность признаков модели

4.4. Моделирование мониторинга в реальном времени

На представленном графике показан итеративный пошаговый прогноз модели LGBM на следующие 24 часа, сформированный на основе предыдущих наблюдений. Для каждого нового временного шага модель использует значения ряда, включая собственные предсказания за предыдущие моменты, чтобы сгенерировать прогноз на следующий час, что позволяет учесть временную зависимость и структуру лагов. График демонстрирует динамику предсказанных значений без использования фактических данных будущих моментов и иллюстрирует, как будет выглядеть мониторинг показателя в реальной системе, когда прогнозы формируются в реальном времени на основе текущих данных.

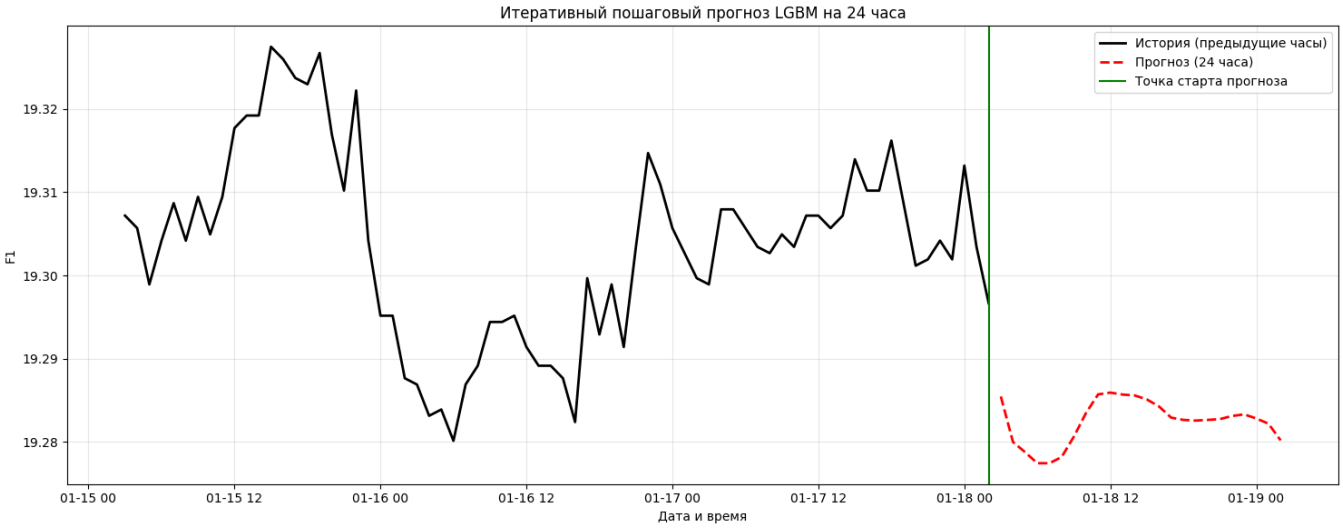

Рис. 4.23 – Прототип мониторинга модели

4.5. Выводы

В результате обучения модели LGBM на нескольких были получены следующие результаты:

- Была адаптирована модель LGBM для предсказания в режиме реального времени;
- Подобраны оптимальные гиперпараметры, обеспечивая высокую точность предсказаний (MAPE: 0.02344%; MAE: 0.00453; MSE: 0.00004);
- Проведена оценка важности признаков, выявлены ключевые лаговые и временные признаки
- Смоделирован мониторинг показателя в реальном времени с использованием LGBM

Заключение

В ходе данной НИР были достигнуты следующие задачи:

- Разработана модель процесса обработки данных, позволяющую в режиме реального времени получать информацию с датчиков и передавать ее в предиктивную систему для дальнейшего предсказания состояния турбоагрегата.
- Разработан алгоритм обучения и оценки модели при работе в режиме реального времени, который может быть интегрирован в систему мониторинга состояния турбины.
- Была адаптирована модель LGBM для предсказания в режиме реального времени;
- Смоделирован мониторинг показателя в реальном времени с использованием LightGBM

Список литературы

1. Трофимов А. Г. Supervised Learning. Basic principles. Regression // 2022 [Электронный ресурс] // URL: https://datalearning.ru/study/Courses/ml/lections/lection02.pdf
2. Гржибовский А. М. Однофакторный линейный регрессионный анализ // Экология человека. – 2008. – №. 10. – С. 55-64.
3. Лапач С. Н., Радченко С. Г. Основные проблемы построения регрессионных моделей // Математические машины и системы. – 2012. – Т. 1. – №. 4. – С. 125-133.
4. Куссуль Н. Н. и др. Регрессионные модели оценки урожайности сельскохозяйственных культур по данным MODIS // Современные проблемы дистанционного зондирования Земли из космоса. – 2012. – Т. 9. – №. 1. – С. 95-107.
5. Уатт Д., Борхани Р., Катсаггелос А. Машинное обучение: основы, алгоритмы и практика применения: Пер. с англ //СПб.: БХВ Петербург. – 2022.
6. Рахимов Р. Х. и др. Регрессионные модели для прогнозирования землетрясений // Computational nanotechnology. – 2018. – №. 2. – С. 40-45.
7. Тагаев О. Н. Регрессионные модели с переменной структурой (фиктивные переменные) // Достижения науки и образования. – 2020. – №. 3 (57). – С. 28-33.
8. Кукурхоев, А. М. Регуляризация и проблема переобучения // СТУДЕНЧЕСКАЯ НАУКА: АКТУАЛЬНЫЕ ВОПРОСЫ, ДОСТИЖЕНИЯ и ИННОВАЦИИ: сборник статей X Международной научно-практической конференции в 2 частях, Пенза, 29 декабря 2022 года. Том Часть 1. – Пенза: Наука и Просвещение, 2022. – С. 108-109. – EDN TBUARY.
9. Облакова Т. В., Григорян В. М., Зубарев К. М. Применение регуляризации при построении полиномиальных регрессионных моделей на примере прогнозирования стоимости бриллиантов // Научное обозрение. Технические науки. – 2024. – С. 31.
10. Воронцов К. В., Потапенко А. А. Регуляризация, робастность и разреженность вероятностных тематических моделей // Компьютерные исследования и моделирование. – 2012. – Т. 4. – №. 4. – С. 693-706.
11. Юдин Н. Е. и др. Регуляризация и ускорение метода Гаусса–Ньютона // COMPUTER. – 2024. – Т. 16. – №. 7. – С. 1829-1840.
12. Игнатьев Н. А., Турсунмуротов Д. Х. Цензурирование обучающих выборок с использованием регуляризации отношений связанности объектов классов // Научно-технический вестник информационных технологий, механики и оптики. – 2024. – Т. 24. – №. 2. – С. 322-329.
13. Strijov V., Krymova E., Weber G. W. Evidence optimization for consequently generated models // Mathematical and Computer Modelling. – 2013. – Т. 57. – №. 1-2. – С. 50-56

14. Акимов А. А., Валитов Д. Р., Кубряк А. И. Предварительная обработка данных для машинного обучения //Научное обозрение. Технические науки. – 2022. – №. 2. – С. 26-31.
15. Быков К.В. Особенности предобработки данных для применения машинного обучения // Молодой ученый. 2021. № 53 (395). С. 1-4. [Электронный ресурс]. URL: https://moluch.ru/archive/395/87491/ (дата обращения: 20.01.2025).
16. Satapathy S. C. et al. (ed.). Information and Decision Sciences: Proceedings of the 6th International Conference on FICTA. – Springer, 2018. – Т. 701.
17. Гадасин Д. В., Шведов А. В., Пантелеева К. А. Предобработка информации для систем машинного обучения. – 2022.
18. Девянин И. С. Предварительная обработка данных для машинного обучения // Фундаментальные и прикладные исследования в физике, химии, математике и информатике. – 2021. – С. 117-121.
19. Gasparrini A., Leone M. Attributable risk from distributed lag models // BMC medical research methodology. – 2014. – Т. 14. – С. 1-8.
20. Томская К. М. Анализ временных рядов с помощью авторегрессионных моделей // Интеграция наук. – 2018. – №. 4. – С. 45-50.
21. Zimmermann N. et al. Self-optimizing thermal error compensation models with adaptive inputs using Group-LASSO for ARX-models // Journal of Manufacturing Systems. – 2022. – Т. 64. – С. 615-625.
22. Полбин А. В. Оценка траектории темпов трендового роста ВВП России в ARX-модели с ценами на нефть // Экономическая политика. – 2020. – Т. 15. – №. 1. – С. 40-63.
23. Горяинова Е. Р., Горяинов В. Б. Знаковые критерии в модели скользящего среднего // Вестник Московского государственного технического университета им. Н.Э. Баумана. Серия «Естественные науки». – 2008. – №. 1. – С. 76-87.
24. Li X., Williams J., Swanson C., Berg T. A machine learning approach to predictive maintenance: remaining useful life and motor fault analysis // IEEE Transactions on Industrial Informatics. – 2019. – Т. 15. – № 6. – С. 3452–3461.
25. Anggreainy M. S. Implementation of Light Gradient Boosting Machine (LGBM) for Customer Services Feedback Classification // Proceedings of the 5th International Conference on Artificial Intelligence and Data Sciences (AiDAS). – Bangkok, Thailand, 2024. – IEEE.
26. Kumari K., et al. Condition Monitoring of Rotating Machines // IEEE International Conference on Applied Electromagnetics, Signal Processing, & Communication (AESPC). – 2025. – С. 1-6.
27. Исторические данные по работе турбины ПТК «МоДеРо» НТЦ «Ресурс» (Excel)