GeekBrains

Направление Аналитика

ДИПЛОМНАЯ РАБОТЫ

Тема: Анализ выхода из строя оборудования на машиностроительном предприятии

Студентка: Бабаева О.С.

Москва
2024

СОДЕРЖАНИЕ

ВВЕДЕНИЕ 4
1 Теоретические основы сбора и анализа данных 5
1.1 Сбор данных 5
1.1.1 Источники сбора данных 6
1.1.2 Способы и инструменты сбора данных 8
1.1.3 Хранение данных 10
1.2 Подготовка данных 12
1.2.1 Методы очистки данных 13
1.2.2 Инструменты очистки данных 14
1.2.3 Методы обработки данных 15
1.2.4 Инструмент обработки данных 18
1.3 Анализ данных 19
1.3.1 Методы анализа данных 19
1.3.2 Инструменты анализа данных 23
2 Практическая часть работы 25
2.1 Сбор данных 25
2.1.1 Сбор и обработка данных, выгруженных из ПО предприятия 25
2.1.2 Парсинг погодных данных 26
2.1.3 Объединение данных 26
2.2 Визуализация данных 27
2.2.1 Работа с данными в Power Query 27
2.3 Прогнозирование поломок оборудования 29
2.2.1 Модель ARIMA 29
2.2.2 Прогноз поломок оборудования на машиностроительном предприятии 33
ЗАКЛЮЧЕНИЕ 35
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 36
ПРИЛОЖЕНИЕ А Отчет в ПО «ОРО» 37
ПРИЛОЖЕНИЕ Б Объединение и преобразование заявок в один файл 38
ПРИЛОЖЕНИЕ В Парсинг сайта погоды 44

ПРИЛОЖЕНИЕ Г Объединение данных 49
ПРИЛОЖЕНИЕ Д Объединение данных 55
ПРИЛОЖЕНИЕ Е Модель ARIMA для прогноза поломок 62

ВВЕДЕНИЕ

Актуальность темы исследования. В современном мире для любого предприятия необходимо не только поддерживать свою конкурентоспособность, но и укреплять её. Прогнозирование и моделирование результатов деятельности могут стать инструментами для повышения конкурентоспособности.

Целью дипломной работы является сбор, анализ и обработка данных по оборудованию машиностроительного предприятия с целью дальнейшего прогнозирования его работы и подготовка визуального отчета (дашборда) для цифровой системы предприятия.

Для достижения поставленной цели в дипломной работе необходимо выполнить следующие задачи:

– собрать необходимую информацию для работы;

– проанализировать и обработать полученные данные;

– построить визуальный отчет по имеющимся данным;

– проверить зависимость выхода из строя оборудования от температуры окружающей среды;

– обучить модель ARIMA для прогнозирования поломок во времени.

Практическая значимость результатов исследования заключается в том, что работа с данными с помощью применения современных методов прогнозирования становятся все более популярным направлением, но многие предприятия все еще не используют в своей работы данный инструментарий и имеют большие сложности по работе с данными.

Апробация результатов исследования. Результаты работы будут применимы на машиностроительном предприятии г. Москва.

1 Теоретические основы сбора и анализа данных

Из-за стремительного развития информационных технологий объём данных, которые хранятся в электронном виде, быстро растёт. Информация существует в разных форматах: текстовых, графических, аудио- и видеофайлах, гипертекстовых документах, реляционных базах данных и так далее. Но большая часть этих данных бесполезна для человека в связи с тем, что он не может обработать такой большой объём информации. Поэтому возникает проблема: как извлечь из большого объёма данных сведения, полезные для пользователя.

Анализ данных состоит из трех этапов (рисунок 1).

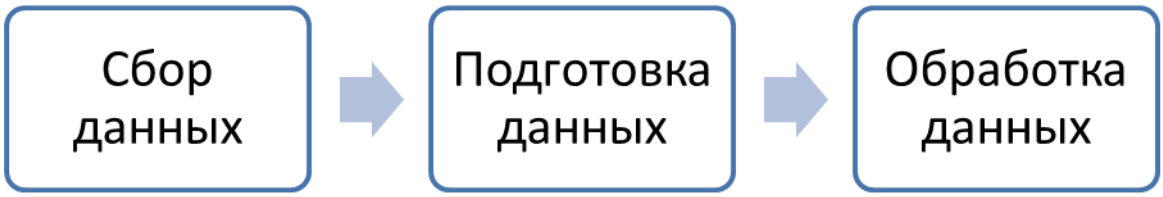


Рисунок 1 – Этапы анализа данных

1.1 Сбор данных

Сбор данных играет важную роль в области науки о данных и инженерии.

Сбор данных – процесс сбора данных из различных источников на определенную тему. Источниками могут служить опросы, фокус-группы, интервью, анкеты, наблюдения и существующие базы данных. Собранная информация затем может быть организована в таблицы или диаграммы для дальнейшего анализа.

Данные могут быть структурированными, полуструктурированными и неструктурированными (рисунок 2). Форматы хранения и передачи данных с разной степенью структурированности представлены на рисунке 3.

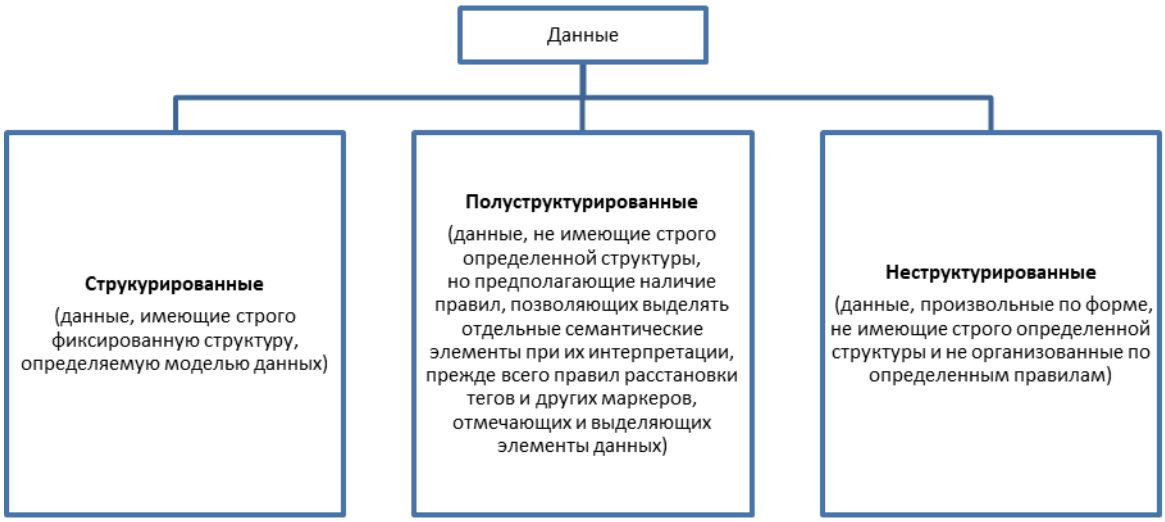


Рисунок 2 – Виды данных

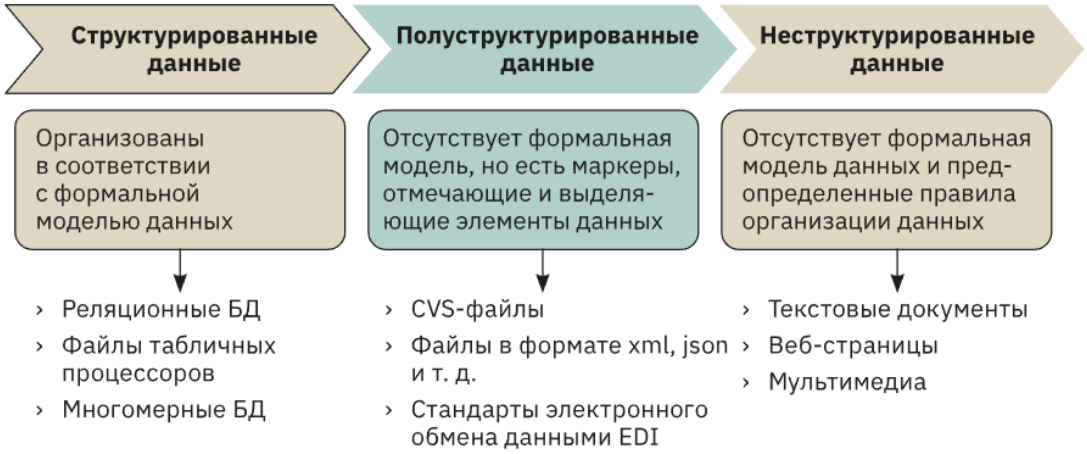


Рисунок 3 – Форматы данных

1.1.1 Источники сбора данных

Источники данных изображены на рисунке 4.

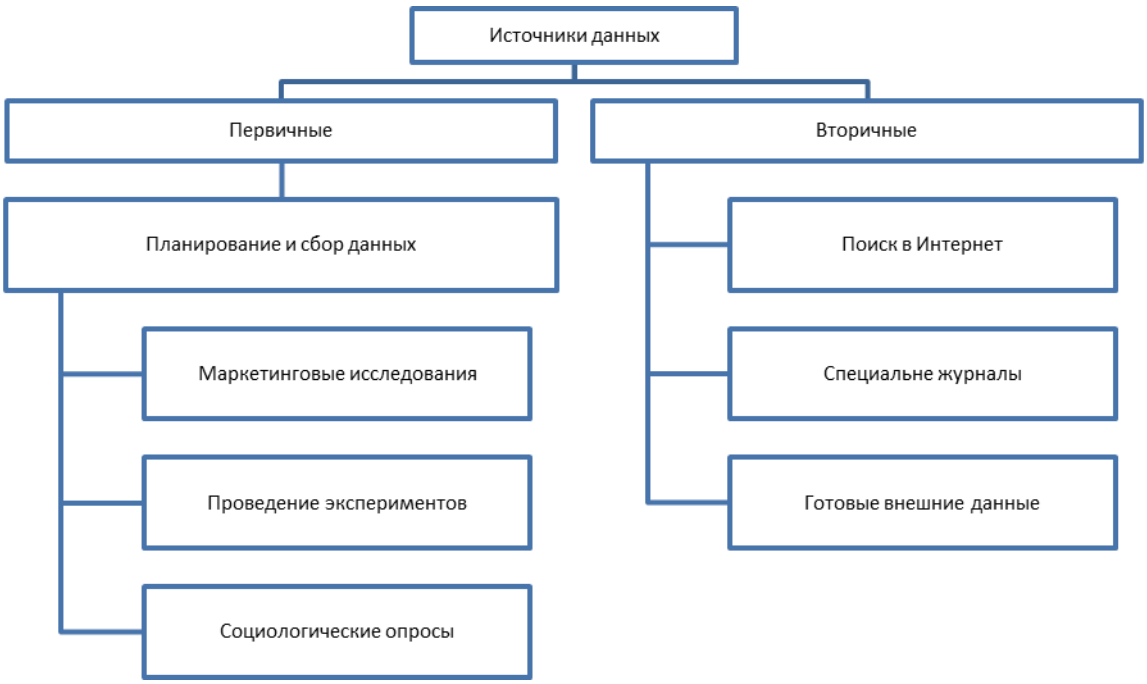


Рисунок 4 – Источники данных

Данные, впервые собираемые исследователем из первичных источников с нуля, называются первичными данными. Эти данные собираются непосредственно из источника происхождения. Это данные в режиме реального времени, которые всегда соответствуют потребностям исследователя. Первичные данные доступны в необработанном виде. Исследователю приходится тратить длительный период времени на сбор первичных данных, и, следовательно, это также дорого. Однако точность и достоверность первичных данных превышают вторичные. Некоторые примеры источниками для сбора первичных данных являются наблюдения, опросы, эксперименты, личные интервью, анкеты и т.д.

Уже существующие данные, которые ранее были собраны кем-либо другим для других целей, называются вторичными данными. В него не включены данные в режиме реального времени, поскольку исследование этой информации уже проводилось. Однако стоимость сбора вторичных данных меньше. Поскольку данные уже собирались в прошлом, их можно найти в уточненном виде. Точность и достоверность вторичных данных относительно ниже, чем первичных. Шансы найти точную информацию или дан-

ные, соответствующие потребностям исследователя, меньше. Однако время, необходимое для сбора вторичных данных, невелико и, следовательно, является быстрым и легким процессом. Некоторые примерами источниками для сбора вторичных данных являются книги, журналы, внутренние отчеты, правительственные отчеты, статьи, веб-сайты, правительственные публикации и т.д.

1.1.2 Способы и инструменты сбора данных

На протяжении веков люди вручную собирали информацию. Даже в современном мире, когда сильно развита цифровизация, мы продолжаем заполнять бумажные документы, вручную вбиваем цифры в файл Excel, фиксируя события и наблюдения. Но работы с бумажными документами и ручной ввод данных отнимают много времени и сил, а кроме того, могут привести к ошибкам из-за человеческого фактора.

В небольших компаниях до сих пор используют ручной сбор данных, но крупные организации стараются этот процесс автоматизировать, используя следующие методы сбора информации.

Рассмотрим следующие методы сбора данных:

- программный интерфейс - Application programming interface (API);
- парсинг данных;
- загрузка файлов (CSV, Excel, XML и т.д.);
- интеграция с CRM-системами и т.д.

Рассмотрим подробнее несколько из вышеописанных методов.

API (Application Programming Interface)

API (Application Programming Interface) — это набор определений, протоколов и инструментов для создания и взаимодействия программного обеспечения. В контексте аналитики данных, API позволяет разработчикам и ана-

литикам обмениваться данными между различными системами, приложениями и сервисами.

Шаги использования API в аналитике данных (рисунок 5).

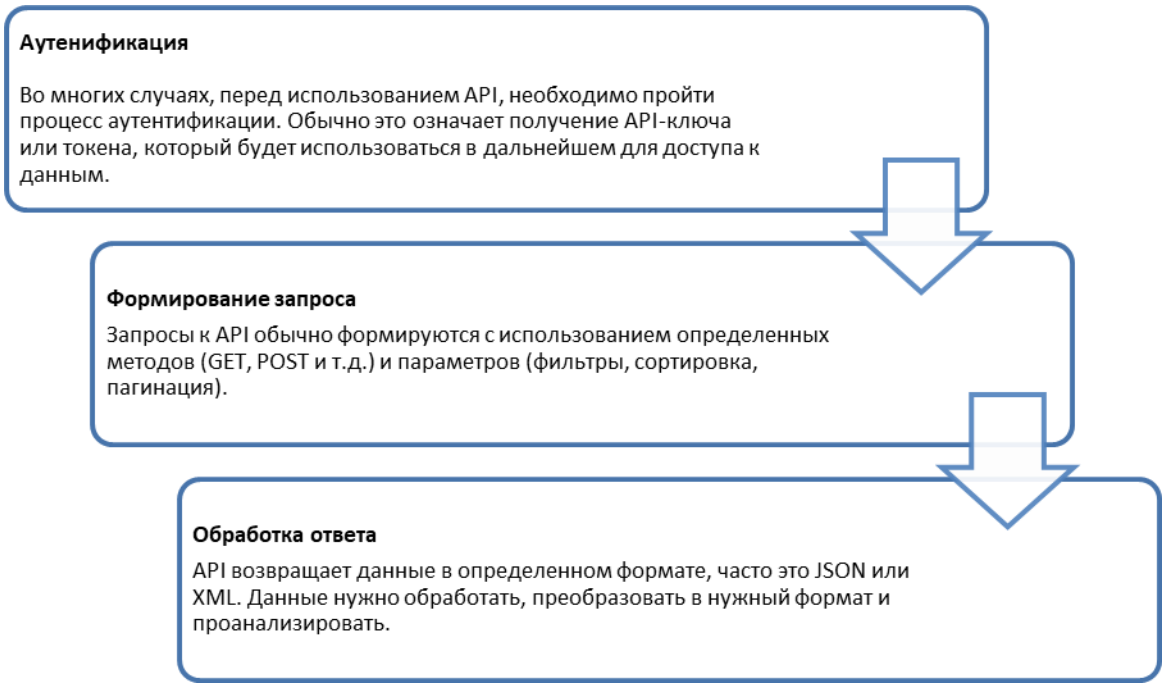


Рисунок 5 – Шаги использования API

Использования API помогает автоматизировать сбор данных, безопасность, конфиденциальность, улучшает качество полученных данных, а главное – экономит ресурсы.

Парсинг сайтов

Это могут быть любые выгрузки из ПО и сервисов для дальнейшего анализа в формате, который в дальнейшем можно обработать.

Парсинг — это процесс автоматического сбора и структурирования данных с помощью скриптов (парсеров). Другое название этого процесса — веб-скрейпинг.

Парсеры работают на разных языках программирования — Python, JavaScript, PHP 5 и т.д.

Парсинг нужен для того, чтобы облегчить и ускорить выполнение рутинных задач. Представьте, сколько времени потребуется человеку, чтобы собрать и систематизировать информацию о тысяче статей с веб-сайта в таблице — это может занять несколько часов. В то же время парсер способен выполнить эту работу за считанные минуты. Парсер значительно ускоряет рабочий процесс и позволяет сократить количество ошибок по сравнению с ручным трудом.

Этапы парсинга представлены на рисунке 6.

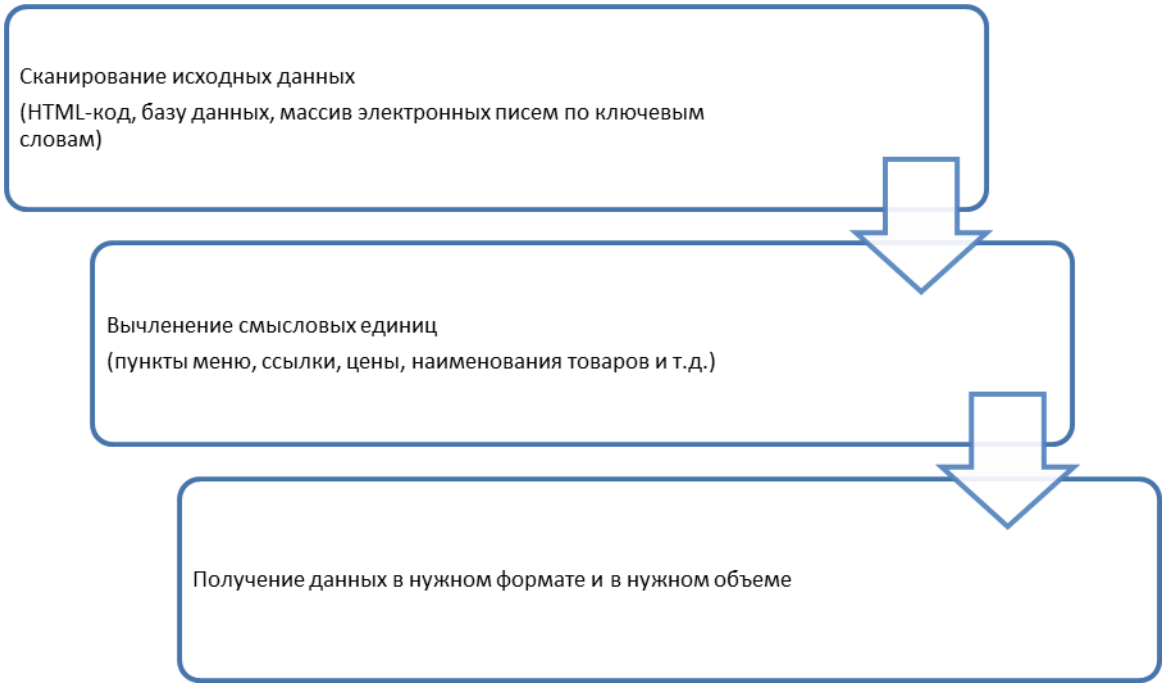


Рисунок 6 – Этапы парсинга

1.1.3 Хранение данных

Существует два типа хранилищ данных: облачное и традиционное (локальное).

Облачное хранилище

Облачное хранилище – это вариант хранения, при котором мы используем удаленные диски для хранения данных в облачном хранилище, исполь-

зуемом клиентом. Для хранения данных на удаленном сервере, принадлежащем поставщику услуг, также используется сеть. Пользователь использует эти параметры хранилища для определения емкости, пропускной способности и удаленного доступа.

Особенности облачного хранилища:

- облачное хранилище предлагает множество вариантов защиты данных.
- Эти варианты хранения данных легко доступны с любого устройства, подключенного к Интернету.
- Ошибки в облачном хранилище легко отслеживаются.

Традиционное хранилище

Традиционное хранилище – это вариант хранения, при котором мы используем локальные физические диски для хранения данных в основном местоположении клиента. Пользователь обычно использует дисковое оборудование для хранения данных, которое используется для копирования, управления данными и интеграции их в программное обеспечение.

Особенности традиционного хранилища:

- Традиционные хранилища быстры, так как независимы от скорости Интернета.
- Безопасность настраивается пользователем вручную.
- Пользователи имеют возможность восстанавливать данные в любое время без проблем с доступом.
- Резервное копирование и модификация данных на месте просты.

Различия между этими двумя системами представлены на рисунке 7.

	Облачное хранилище	Традиционное хранилище
Производительность	Облачное хранилище работает лучше благодаря использованию NoSQL	Традиционное хранилище работает немного медленнее по сравнению с облачным
Техническое обслуживание	Этот тип хранилищ прост в обслуживании по мере использования, а поставщик услуг заботится об обслуживании	Управлять этим хранилищем сложно, поскольку вам необходимо вручную запускать инструменты обслуживания
Надежность	Облачные хранилища отличаются высокой надежностью, поскольку для их запуска требуется меньше времени	Традиционное хранилище требует больших начальных усилий и менее надежно
Общий доступ к файлам	Облачное хранилище поддерживает динамический обмен файлами, поскольку им можно делиться в любом месте с доступом к сети	Традиционное хранилище требует физических дисков для обмена данными, и между ними должна быть создана сеть
Время доступа к файлам	В этой системе время доступа к файлам зависит от скорости сети	Эта система имеет более быстрое время доступа по сравнению с облачным хранилищем
Безопасность	Облачное хранилище более безопасно, поскольку оно интегрируется со многими инструментами безопасности	Традиционные хранилища безопасны, поскольку они могут быть легко атакованы вирусами и вредоносными программами
Приложения	Amazon Drive, Dropbox, автосинхронизация	HDD, SSD и флешки

Рисунок 7 – Различия облачного и традиционного хранения

1.2 Подготовка данных

Специалисты по работе с данными тратят около 80% всего времени на то, чтобы привести данные к нужному виду для дальнейшего анализа и работы с ними. Именно поэтому данный процесс является очень важной частью в анализе данных.

Обработка данных (или предобработка) — это значимый процесс, в ходе которого сырые данные, содержащие ошибки, дубликаты и пропуски, доводятся до пригодного для анализа состояния. От качества этих данных зависит качество информации, на основе которой принимаются решения.

1.2.1 Методы очистки данных

Исправление или удаление ошибочных, неполных и некорректных данных называется очисткой данных. Вот несколько подходов, которые могут вам в этом помочь:

- удаление дубликатов;
- заполнение пропущенных значений;
- обработка выбросов;
- стандартизация и нормализация.

Удаление дубликатов

Повторяющиеся данные способны негативно повлиять на результаты анализа. Чтобы исключить дублирующиеся записи, можно использовать такие функции, как drop_duplicates() в Python. Дубликаты могут появляться из-за разных факторов, например, при ошибочном вводе данных или объединении нескольких источников информации. Удаление дубликатов помогает повысить качество данных и точность анализа.

Заполнение пропущенных значений

Отсутствующие данные можно заменить на средние, медианные или модальные значения. В Pandas для этого используется метод fillna(). Пропуски могут появляться по разным причинам: из-за ошибок при сборе информации или из-за её отсутствия. Замена пропущенных значений помогает избежать ошибок в анализе и повышает качество данных.

Обработка выбросов

Аномальные значения способны исказить результаты анализа. Чтобы выявить и обработать выбросы, можно использовать такие методы, как межквартильный размах (IQR) или Z-оценка. Выбросы могут появляться по разным причинам: из-за ошибок при вводе данных или из-за необычных явлений. Устранение выбросов помогает повысить качество данных и точность анализа.

Стандартизация и нормализация

Эти методы позволяют привести данные к общему виду, что особенно важно для алгоритмов машинного обучения. В Python для этого можно использовать библиотеку sklearn.preprocessing. Применение стандартизации и нормализации помогает повысить качество данных и точность анализа.

1.2.2 Инструменты очистки данных

Инструменты для очистки данных представлены на рисунке 8.

	Описание
Pandas	Это мощная библиотека для работы с данными в Python. Pandas позволяет легко и эффективно работать с данными, что делает его одним из самых популярных инструментов для очистки данных.
OpenRefine	Это инструмент с открытым исходным кодом, который позволяет легко очищать и трансформировать данные. Он поддерживает множество форматов данных и предоставляет интуитивно понятный интерфейс. OpenRefine позволяет легко и эффективно работать с данными, что делает его одним из самых популярных инструментов для очистки данных.
Trifacta	Это коммерческий инструмент для очистки и подготовки данных. Он использует машинное обучение для автоматического выявления и исправления ошибок в данных. Trifacta позволяет легко и эффективно работать с данными, что делает его одним из самых популярных инструментов для очистки данных.

Рисунок 8 – Инструменты очистки данных

1.2.3 Методы обработки данных

После того, как мы очистим данные, необходимо их обработать. Вот несколько подходов, которые могут вам в этом помочь:

- агрегация данных;
- трансформация данных;
- фильтрация данных;
- слияние данных.

Агрегация данных

Объединение и обобщение данных из разных источников в один набор называется агрегацией данных. Этот процесс подготавливает информацию для анализа, позволяя увидеть закономерности и идеи, которые невозможно обнаружить при рассмотрении отдельных точек данных. К примеру, можно вычислить средний объём продаж по месяцам. В Pandas для этого применяют метод groupby(). Агрегация данных позволяет сделать анализ проще и понятнее.

Процесс агрегации состоит из трех этапов, представленных на рисунке 9.

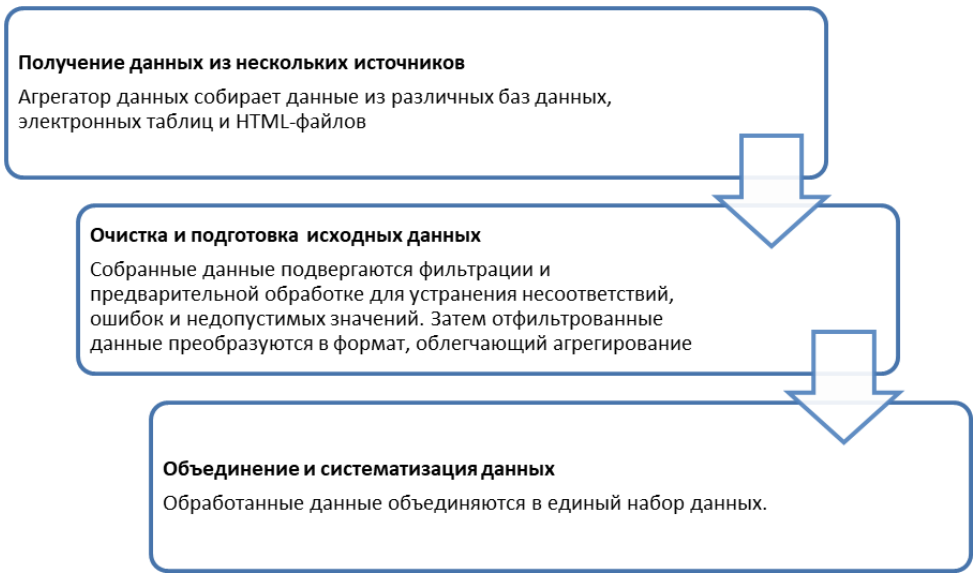


Рисунок 9 – Этапы агрегации

Трансформация данных

Трансформация данных — это преобразование и согласование наборов данных друг с другом или с определённой схемой. Она заключается в оптимизации представлений и форматов данных с точки зрения решаемых задач и целей анализа. Например, категориальные данные можно преобразовать в числовые с помощью метода get_dummies() в Pandas. Преобразование данных способствует повышению их качества и делает анализ более точным.

Основная задача данного метода – преобразовать данные в такой вид, чтобы их можно было бы максимально эффективно использовать.

Методы трансформации данных: нормализация, преобразование типов и форматов, сортировка, группировка и т.д.

Данные операции производят после получения данных из разных источников и обеспечивает их дальнейшую обработку.

Фильтрация данных

Фильтрация данных — это процесс выделения самых необходимых данных из большого набора данных с использованием определённых условий или критериев. Она дает возможность быстро анализировать нужные данные, избегая просмотра всего набора, что делает анализ более целенаправленным и эффективным.

Задачи фильтрации представлены на рисунке 10.

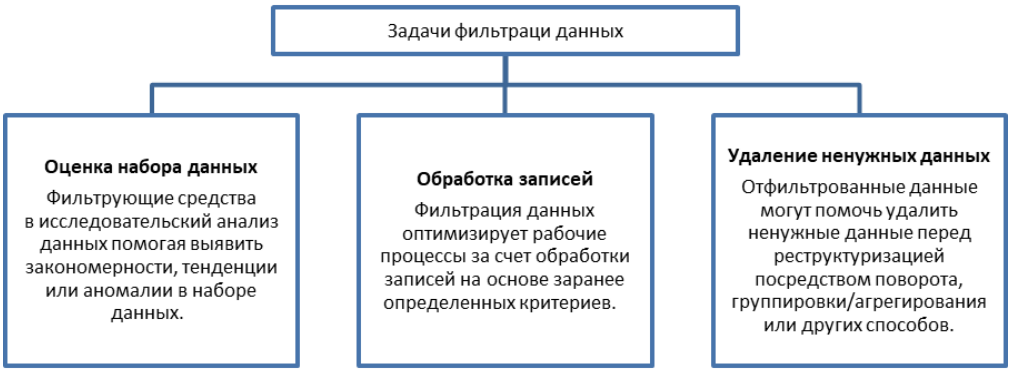


Рисунок 10 – Задачи фильтрации

Основные методы фильтрации данных представлены на рисунке 11.

	Описание
Фильтрация по критериям	Фильтрация по критериям включает более сложную фильтрацию на основе нескольких критериев или условий. Например, компания электронной коммерции может фильтровать данные о клиентах для таргетинга маркетинговой кампании. Они могут использовать несколько критериев, например, клиенты, которые приобрели более 100 долларов США в прошлом месяце, находятся в возрасте 25–35 лет и ранее покупали электронные продукты.
Фильтрация по временному диапазону	Временные фильтры работать путем выбора данных в течение определенного периода времени. Финансовый аналитик может использовать фильтр временного диапазона анализировать тенденции фондового рынка путем фильтрации данных о транзакциях, чтобы включить только те, которые произошли в последнем квартале. Это помогает сосредоточиться на недавнем поведении рынка и прогнозировать будущие тенденции.
Фильтрация текса	Фильтрация текста включает методы фильтрации текстовых данных, такие как сопоставление с образцом. Например, платформа социальных сетей может фильтровать сообщения, содержащие определенные ключевые слова или фразы, чтобы отслеживать контент, связанный с конкретным событием или темой.
Числовая фильтрация	Числовая фильтрация включает методы фильтрации числовых данных на основе пороговых значений. Базу данных здравоохранения можно отфильтровать для выявления пациентов с высоким кровяным давлением, установив числовой фильтр для включения всех записей, в которых систолическое давление превышает 140 мм рт. ст., а диастолическое давление — выше 90 мм рт. ст.

Рисунок 11 – Методы фильтрации

Слияние данных

Слияние данных — это процесс, при котором объединяются различные источники информации для получения более согласованной, точной и полезной информации по сравнению с данными из одного отдельного источника.

Этот процесс включает в себя добавление новых деталей к уже имеющимся данным, интеграцию новых случаев и удаление дублирующей или не-

корректной информации. В Pandas для этой цели применяются методы merge() и concat(). Объединение данных способствует повышению их качества и делает анализ более точным.

1.2.4 Инструмент обработки данных

Инструменты для обработки данных представлены на рисунке 12.

	Описание
Python	Это один из самых популярных языков для обработки данных. Он предоставляет множество библиотек, таких как Pandas, NumPy и Scikit-learn, которые облегчают процесс обработки данных. Python позволяет легко и эффективно работать с данными, что делает его одним из самых популярных инструментов для обработки данных.
R	Это язык программирования, специально разработанный для статистического анализа и обработки данных. Он предоставляет множество пакетов, таких как dplyr и tidyr, которые упрощают работу с данными. R позволяет легко и эффективно работать с данными, что делает его одним из самых популярных инструментов для обработки данных.
SQL	Это язык запросов, который используется для работы с реляционными базами данных. Он позволяет эффективно фильтровать, агрегировать и трансформировать данные с помощью запросов. SQL позволяет легко и эффективно работать с данными, что делает его одним из самых популярных инструментов для обработки данных.
Excel	Это популярный инструмент для работы с данными, который предоставляет множество функций для их обработки. Он особенно полезен для небольших наборов данных и простых задач. Excel позволяет легко и эффективно работать с данными, что делает его одним из самых популярных инструментов для обработки данных.

Рисунок 12 – Инструменты обработки данных

Обработка данных является ключевым этапом в аналитике. Применяя методы и инструменты, описанные выше, мы можем эффективно подготовить данные и добиться точных результатов. Качество данных играет важнейшую роль в аналитике, так как именно от него зависит надежность полу-

чаемых результатов. Некачественные данные могут привести к ошибочным выводам и негативно сказаться на бизнесе.

1.3 Анализ данных

Исследование данных для обнаружения полезной информации, выводов и помощи в принятии решений называется анализом данных. В мире, где информация имеет огромное значение, способность анализировать данные становится всё более востребованной. Анализ данных охватывает множество методов и подходов, которые позволяют исследовать информацию с разных сторон. Эти методы помогают выявлять скрытые тенденции, предсказывать будущие события и принимать взвешенные решения. Важно помнить, что каждый метод имеет свои особенности и ограничения, поэтому выбор метода зависит от конкретной задачи и типа данных.

1.3.1 Методы анализа данных

Методы анализа данных значительно эволюционировали, предоставляя комплексный инструментарий для понимания, интерпретации и прогнозирования шаблонов данных. Эти методы имеют решающее значение для извлечения полезной информации из данных, позволяя организациям принимать обоснованные решения.

Основные методы анализа данных:

- описательный анализ;
- качественный анализ;
- прогнозный анализ;
- диагностический анализ;
- регрессионный анализ;
- факторный анализ;
- кластерный анализ и т.д.

Описательный анализ

Описательный анализ рассматривается как начальный этап аналитического процесса и обычно направлен на то, чтобы ответить на вопросы о произошедших событиях. Он включает в себя систематизацию данных, работу с ними и их интерпретацию из различных источников для получения ценной информации.

Пример:

– Показатели продаж: компания розничной торговли может использовать описательную статистику, чтобы понять средний объем продаж по магазинам или определить, какие товары пользуются наибольшим спросом.

– Опросы удовлетворенности клиентов: Анализ данных опроса для поиска наиболее распространенных ответов или средних баллов.

Качественный анализ

Методы качественного анализа данных не предполагают прямого измерения, поэтому этот подход применяется, когда организации нужно принимать решения на основе субъективной интерпретации. Например, качественные данные могут использоваться для оценки отзывов клиентов, влияния вопросов опроса, эффективности публикаций в социальных сетях, анализа конкретных изменений или особенностей продукта и многого другого. Качественный анализ также помогает систематизировать данные по темам или категориям, которые впоследствии можно автоматизировать.

Пример:

– Анализ рынка: Бизнес может проанализировать, почему продажи продукта резко выросли в определенном квартале, изучив маркетинговую активность, изменения цен и тенденции рынка.

Медицинская диагностика: Врачи используют диагностический анализ для понимания причины симптомов на основе результатов лабораторных исследований и данных пациента.

Прогнозный анализ

Прогностический анализ данных даёт возможность прогнозировать будущие события, отвечая на вопрос «что произойдёт?». Чтобы применять этот метод, нужно использовать результаты описательного анализа данных, методы исследовательского и диагностического анализа, а также сочетать машинное обучение и искусственный интеллект. Благодаря этому подходу можно получить представление о будущих тенденциях и выявить потенциальные проблемы и слабые места в наборе данных. Также с помощью точных данных компании могут находить и разрабатывать идеи для улучшения операционных процессов и конкурентного преимущества.

Пример:

– Кредитный рейтинг: Финансовые учреждения используют прогностические модели для оценки вероятности дефолта клиента по кредиту.

– Прогнозирование погоды: Метеорологи используют прогностические модели для прогнозирования погодных условий на основе исторических данных о погоде.

Диагностический анализ

Зная причину произошедшего, нетрудно определить способы, которые к этому привели. Например, с помощью диагностического анализа можно понять, из-за чего снизились продажи, и в итоге изучить конкретные факторы, которые привели к потерям.

Пример:

– Анализ запасов: Проверка, коррелирует ли снижение продаж с нехваткой или переизбытком запасов.

– Эффективность продвижения: Анализ влияния различных рекламных кампаний, чтобы определить, какие из них не смогли привлечь клиентов.

Регрессионный анализ

Суть этого метода заключается в анализе исторических данных для понимания влияния изменений независимых переменных на зависимую переменную. Выявляя взаимосвязи между переменными и прошлыми событиями или инициативами, вы можете предсказать вероятные результаты в будущем. Этот подход помогает вам принимать взвешенные решения и выбирать наиболее эффективный путь.

Пример:

– Оценка рыночных тенденций: Оценка того, как изменения в экономической среде (например, процентные ставки) влияют на цены на недвижимость.

– Прогнозное ценообразование: Использование исторических данных для прогнозирования будущих ценовых тенденций на основе текущей динамики рынка.

Факторный анализ

Факторный анализ данных направлен на выявление закономерностей среди связанных переменных с помощью скрытых факторов. Ина-че говоря, он помогает выделить независимые переменные, что полезно для улучшения определённых областей.

Пример:

– Улучшение обслуживания: Определение ключевых факторов, таких как время ожидания, поведение персонала и результат лечения, которые влияют на удовлетворенность пациентов.

– Распределение ресурсов: Использование этих данных для улучшения областей, которые существенно влияют на удовлетворенность пациентов.

Кластерный анализ

Кластерный анализ позволяет наглядно представить данные и выявить в них общие закономерности. Он часто применяется, когда нужно сделать информацию более понятной или провести её анализ, а также если категории данных неоднозначны. В ходе этого процесса похожие наблюдения группируются в кластеры, что помогает присвоить группам названия и определить их категории.

Пример:

– Сегментация рынка: Разделение клиентов на группы, которые демонстрируют схожее поведение и предпочтения для более целенаправленного маркетинга.

– Настройка кампании: Разработка уникальных маркетинговых стратегий для каждого кластера для максимального привлечения и конверсий.

1.3.2 Инструменты анализа данных

Чтобы успешно работать с большими объёмами информации, аналитикам нужно использовать ряд особых инструментов и технологий. Эти ресурсы помогают быстро и эффективно извлекать, обрабатывать и понимать данные, что позволяет достигать поставленных целей в кратчайшие сроки. Важно помнить, что каждый инструмент имеет свою терминологию и назначение, поэтому для их использования требуется определённая подготовка и знания.

Инструмент представлены на рисунке 13.

	Описание
Языки программирования	Они занимают важное место в сфере обработки и анализа данных. Одними из самых востребованных являются Python и R. Python ценится за свою доступность и эффективные библиотеки для работы с данными, такие как Pandas и NumPy. R широко используется в области статистики и визуализации благодаря инструментам ggplot2 и dplyr.
Инструменты визуализации	Инструменты визуализации, такие как Tableau, Power BI и Matplotlib, широко используются для наглядного представления результатов. Они дают аналитикам возможность создавать интерактивные графики и диаграммы, которые помогают лучше понять тенденции и закономерности в данных.
Платформы для обработки и хранения данных	Такие технологии, как Apache Hadoop и Spark, играют важную роль в обработке больших объёмов данных. Они позволяют эффективно распределять задачи между множеством узлов, что значительно ускоряет процесс обработки информации.
Средства машинного обучения	Инструменты машинного обучения помогают решать сложные задачи прогнозирования. Платформы, такие как TensorFlow и Scikit-Learn, предоставляют эффективные средства для создания и обучения моделей, которые могут самостоятельно обучаться и извлекать полезную информацию без необходимости явного программирования каждого шага.

Рисунок 13 – Инструменты анализа данных

2 Практическая часть работы

Данная работы будет производиться на базе машиностроительного предприятия г. Москва.

В настоящий момент предприятие находится на этапе цифровизации, что заставляет пересмотреть многие бизнес-процессы.

В рамках проекта по оборудованию было предложено собрать и проанализировать имеющиеся данные по выходу из строя оборудования, найти закономерности (в том числе от температуры окружающей среды) и создать проект по визуализации (дашборд).

2.1 Сбор данных

2.1.1 Сбор и обработка данных, выгруженных из ПО предприятия

На машиностроительном предприятии имеется ПО «ОРО», где фиксируются поломки оборудования с номером заявки, датой подачи, наименованием оборудования и другими характеристиками. Скриншот представлен в приложении А.

Данные находятся в разных разделах. В процессе работы было выгружено 9 файлов в формате .xlsx.

Рассмотрим данный процесс пошагово.

1. Импортируем все заявки (1…9.xlsx) и объединим в один файл (Приложение Б, рисунок Б.1, Б.2).
2. Заполняем пустые значения в столбце «Дата закрытия заявки» (Приложение Б, рисунок Б.3).
3. Рассчитаем длительность ремонта оборудования (Приложение Б, рисунок Б.4).
4. Изменим регистр по нескольким столбцам.
5. Удалим дубликаты данных (Приложение Б, рисунок Б.5).

6. Удалим пустые значения по двум столбцам (Приложение Б, рисунок Б.6).

Полученные данные сохраняем в CSV-файл.

2.1.2 Парсинг погодных данных

Для проверки зависимости частоты выхода из строя оборудования от температуры воздуха необходимо найти погодные данные.

Для выполнения поставленной задачи были проанализированы сайты с погодой по Москве и выбран источник https://ginfo.ru.

На сайте имеется информация о погоде с 2019 по 2024 год.

Процедура парсинга сайта:

1. Парсинг данных (Приложение В, рисунок В.1).
2. Преобразование данных (Приложение В, рисунок В.2, В.3, В.4, В.5).

Полученные данные сохраняем в CSV-файл.

2.1.3 Объединение данных

Произведем объединение файлов с заявками и температурой.

Шаги:

1. Импортируем файлы request.csv и weather.csv (Приложение Г, рисунок Г.1).
2. Изменение типа данных даты (Приложение Г, рисунок Г.2).
3. Объединение таблиц по дате (Приложение Г, рисунок Г.3).
4. В связи с тем, что данные с температурными показателями с 2019 года, необходимо удалить значения без температуры (Приложение Г, рисунок Г.4). После удаление остается 10535 строк.
5. Выделим из столбца «Инв.номер» категорию и номер (Приложение Г, рисунок Г.5).

6. Изменим типы данных (Приложение Г, рисунок Г.6).

7. Изменим расположение столбцов (Приложение Г, рисунок Г.7).

Полученные данные сохраняем в CSV-файл.

2.2 Визуализация данных

Визуализация данных — важный инструмент для анализа и интерпретации информации. Она превращает сложные массивы данных в понятные графики, диаграммы и таблицы. В данной работе мы воспользуемся инструментом Power BI от Microsoft, который предоставляет широкие возможности для визуализации данных.

2.2.1 Работа с данными в Power Query

Загрузим полученную таблицу join.csv с заявками и температурой в Power BI и преобразуем данные в Power Query (Приложение Д, рисунок Д.1).

Первым шагом мы должны преобразовать данные в таблице.

Затем создаем новые столбцы с необходимой информацией для будущего визуального отчета:

– Проверяем наименование цехов и удаляем несколько нечищенных данных через фильтр.

– Выделяем месяц из даты (январь, февраль и т.д.).

– Выделяем ЧПУ и универсальное оборудование (Приложение Д, рисунок Д.2).

2.2.2 Создание дашборда в Power BI

Power BI — это инструмент визуализации данных и бизнес-аналитики от Microsoft, который преобразует данные из разных источников для создания различных отчетов бизнес-аналитики. Microsoft Power BI позволяет ком-

паниям легко выявлять тенденции, отслеживать производительность и принимать решения на основе данных.

Он может принимать входные данные из таблиц Excel, баз данных или даже облачного хранилища. Вы можете подключать источники данных, получать информацию и делиться ею с другими. Power BI - самый популярный инструмент бизнес-аналитики.

Первый шаг – это проработка проекта визуального отчета. Желание руководства – видеть данные о количестве поломок для каждого цеха и категории с удобной навигацией. Параметры для отчетности:

– количество поломок за год и по месяцам;
– количество поломок по возрасту;
– количество поломок по категориям;
– количество поломок по моделям;
– количество поломок по ЧПУ и универсальному оборудованию;
– количество поломок по температуре окружающей среды (по возможности).

Данные пожелания были учтены в процессе выполнения работы.

Проект дашборда представлен в приложении Д.

В настоящий момент прорабатывается вопрос дополнить дашборд общей информацией об оборудовании (не по поломкам). Собираются данные.

2.2.3 Рекомендации

В процессе выполнения работы были выявлены недостатки текущей системы сбора информации по оборудованию.

В настоящий момент пишется техническое задание на доработку ПО «ОРО» по следующим направлениям:

– Добавить столбец «Дата реагирования на поломку», для возможности отслеживания скорости реагирования службой механика.

– Добавить столбец «Категория поломки»: механическая, гидравлическая и т.д. Названия категорий должны быть унифицированы, заполняться из списка, чтобы избежать ошибок.

– Добавить столбец «ЗиП», где из существующего списка необходимо выбрать одно или несколько приспособлений, вышедших из строя. Это позволит в дальнейшем прогнозировать поломку и грамотно формировать страховой запас ЗиП.

Данные изменения позволят более эффективно использовать возможности данного ПО.

2.3 Обучение модели ARIMA прогнозу поломок

2.2.1 Модель ARIMA

Анализ временных рядов имеет большое значение во многих сферах, включая финансы, экономику и метеорологию. Модель авторегрессионного интегрированного скользящего среднего (ARIMA) — один из основных инструментов для предсказания будущих значений на основе исторических закономерностей в данных временных рядов. Но чтобы прогнозы были точными, важно правильно подобрать параметры для модели ARIMA.

Модель объединяет три ключевых компонента для моделирования данных (рисунок 14).

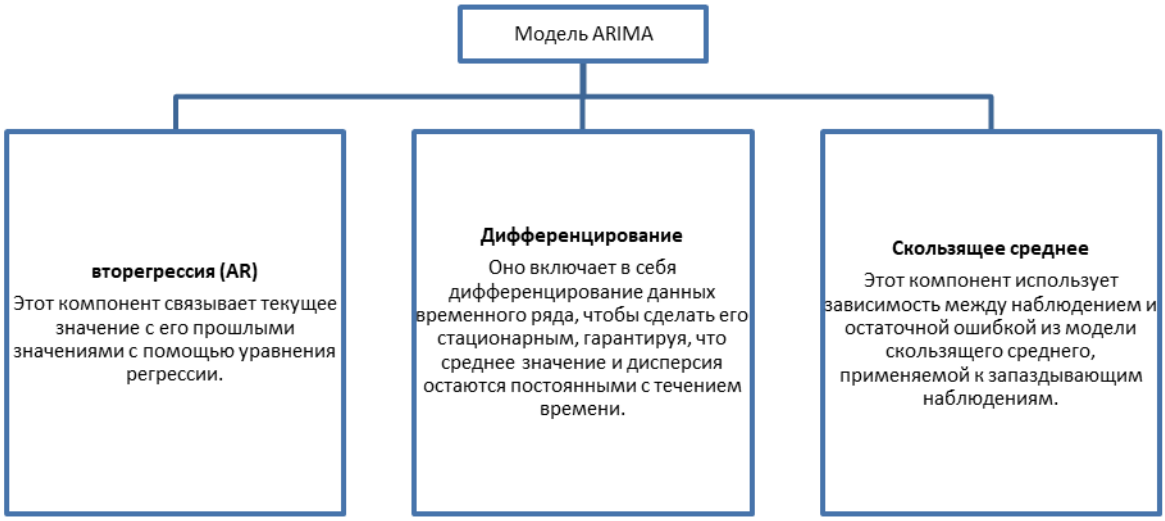


Рисунок 14 – Модель ARIMA

– Оценка параметров: оценка параметров p, d и q включает анализ графиков автокорреляционной функции (ACF) и частичной автокорреляционной функции (PACF) данных временных рядов. ACF помогает определить порядок MA (q), в то время как PACF помогает определить порядок AR (p).

– Подгонка модели: после определения параметров модель ARIMA подгоняется к данным. Это предполагает минимизацию ошибки (часто с использованием таких методов, как оценка максимального правдоподобия) для получения наиболее подходящих коэффициентов для условий авторегрессии и скользящего среднего.

– Прогнозирование: после подгонки модели ее можно использовать для прогнозирования будущих значений путем итерации с течением времени.

Параметры модели:

– p (порядок изменений): представляет количество членов авторегрессии и обозначается буквой p. Это относится к количеству прошлых наблюдений, которые напрямую влияют на текущее значение.

– d (порядок различий): представляет количество различий, необходимых для того, чтобы сделать временной ряд стационарным. Это включает вычисление различий между последовательными наблюдениями.

– q (порядок MA): обозначаемый q, он представляет количество запаздывающих ошибок прогноза в уравнении прогнозирования.

Выбор подходящих значений для этих параметров существенно влияет на способность модели к прогнозированию. Однако определение правильных значений часто является сложной задачей.

Методы выбора модели для ARIMA представлен на рисунке 15.

	Описание
Визуальный осмотр	Графики временных рядов: Визуализация данных для выявления тенденций, сезонности и неравномерностей помогает понять характеристики данных. Функция автокорреляции (ACF) и функция частичной автокорреляции (PACF): Эти графики помогают определить потенциальные значения для p и q, демонстрируя корреляции между наблюдениями с разными задержками. Снижение автокорреляции с определенными задержками может указывать на порядок расположения соответствующих членов.
Поиск по сетке параметров	Поиск по сетке: это включает систематическую оценку различных комбинаций значений p, d и q для нахождения набора, который оптимизирует выбранную метрику оценки, такую как AIC (информационный критерий Акайке) или BIC (байесовский информационный критерий). Итеративный поиск: Начиная с диапазона возможных значений для p, d и q, этот метод итеративно проверяет комбинации для определения наилучшего соответствия.
Автоматизированные методы	Несколько программных пакетов и библиотек предлагают автоматизированные алгоритмы выбора модели ARIMA (например, auto-ARIMA в pmdarima от Python или R's forecast package), которые определяют оптимальные параметры на основе статистических показателей.
Перекрёстная проверка	Разделение данных на обучающие и проверочные наборы и тестирование производительности модели на различных участках помогает оценить ее надежность и точность. Для оценки точности прогнозирования могут быть использованы такие методы, как прогнозирование исходного состояния по циклу или пошаговая проверка.
Информационные критерии	AIC, BIC: Эти статистические показатели помогают сравнивать модели, снижая сложность, поощряя выбор моделей, которые сочетают в себе удобство и простоту. Более низкие значения AIC или BIC указывают на более подходящие модели.
Сравнение моделей	Подберите несколько моделей-кандидатов с различными комбинациями параметров и сравните их производительность с помощью статистических показателей, визуального контроля и диагностических тестов на наличие остатков.
Пошаговые методы	Реализуйте пошаговые методы, такие как stepwise AIC или stepwise BIC, для итеративного добавления или удаления параметров из модели, улучшая подгонку.

Рисунок 15 - Методы выбора модели

2.2.2 Обучение модели

Для возможности прогнозировать поломку оборудования во времени было принято воспользоваться моделью ARIMA и обучить ее на имеющемся датасете.

Перед тем, как приступить к обучению, мы проверим наши данные на пропуски (Приложение Е, рисунок Е.1).

Затем из имеющегося датасета мы выделим необходимые значения: дату подачи заявки и количество поломок в каждую дату (Приложение Е, рисунок Е.2).

Проверим, есть ли зависимость в нашем датасете количества поломок от температуры (при составлении дашборда это визуально подтвердилось). На получившемся графике также можно проследить эту закономерность, но не так явно (Приложение Е, рисунок Е.3).

Следующий шаг – проверка на стационарность. Для этого воспользуемся тестом Дики-Фуллера (Приложение Е, рисунок Е.4). Наши данные являются стационарными.

Теперь можно приступать к построению и оценке модели (Приложение Е, рисунок Е.5). Проанализируем полученные данные:

– Коэффициенты: (P>|z|) меньше 0,05.

– AIC: 12361.004 – не самая хорошая подгонка модели.

– L1: 0,09 – тоже не лучший показатель (больше 0,05).

– JB: 0,00 – должно быть больше 0,05 для того, чтобы остатки были нормальны.

– H: 1,30, что больше о,05 – это говорит о том, что остатки гетероскедастичны, это хорошо.

Построим график (Приложение Е, рисунок Е.6). По нему мы видим, что модель не особо точна.

Попробуем автоматическую прогонку модели. Ее метрики выглядят лучше, но все еще не идеальны (Приложение Е, рисунок Е.7). Оценим эту

модель с помощью графиков (Приложение Е, рисунок Е.8). По графикам мы видим, что распределение не является нормальным, ее еще надо улучшать.

ЗАКЛЮЧЕНИЕ

Предприятию, чтобы сохранять конкуренцию на рынке и эффективно использовать имеющиеся ресурсы необходимо как можно больше использовать возможности цифровизации и в частности, уметь работать с большими данными.

В результате выполнения дипломной работы были решены следующие задачи:

– проведен анализ данных по выходу из строя оборудования;

– построен дашборд;

– изучена зависимость выхода из строя оборудования от температуры внешней среды;

– поработали с моделью ARIMA для прогнозирования поломок и пришли к выводу, что необходимо дальше продолжать работу в этом направлении.

Проведенная работы показала, что имеется зависимость выхода из строя оборудования от времени года, а также рост поломок с каждым годом.

Были разработаны рекомендации по корректировки работы ПО «ОРО» для более качественного сбора данных о поломках оборудования машиностроительного предприятия.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 Ильяшенко О. Ю. Роль BI–систем в совершенствовании процессов обработки и анализа бизнес информации [Текст]: учебник / И.В. Ильин, Д.Д. Болобонов. – Наука и бизнес: пути развития, №6, 2017.

2 Немуров Е.В., Золотухина Е.Б. Актуальность внедрения BI систем на предприятиях в условиях современного рынка [Текст] // Международный научно-технический журнал «Теория. Практика. Инновации». – 2018.

3 Агамиров Л. В. Статистические методы анализа результатов научных исследований : учебное пособие / Л. В. Агамиров. - Москва: Изд-во МЭИ, 2018. - 71 с.

4 Белоусов П. А., Марухина О. В., Скоморохов А. О. Машинное обучение и большие данные : учебное пособие / П. А. Белоусов, О. В. Марухина, А. О. Скоморохов. - Санкт-Петербург : ГУАП, 2021. - 119 с.

5 Болдырев А. В. Технологии хранения данных / А. В. Болдырев. - Ростов-на-Дону: ДГТУ, 2019. - 77 с.

6 Келлехер Д., Тирни Б. Наука о данных: базовый курс / Джон Келлехер, Брендан Тирни. - Москва : Альпина Паблишер, 2020. – 220 с.

ПРИЛОЖЕНИЕ А

Отчет в ПО «ОРО»

№ заявки	Цех	Инв. номер	Год выпуска	Оборудование	Модель	Дата открытия	Описание неисправ.	Причина неисправности	Принятые меры	Дата закрытия заявки
16-06-039	ЦЕХ 27	Ж-1-23	2004	2-Х рядная механизир. линия д/химич. пассивир.	МЛХ-168	07.06.2016	прошить, установить частотн.преобразоват.		замена частотного преобразователя	16.06.2016
16-06-091	ЦЕХ 27	Ж-1-23	2004	2-Х рядная механизир. линия д/химич. пассивир.	МЛХ-168	16.06.2016	установить новый частотн.преобразователь		замена,програм.эл.приво да	16.06.2016
19-03-016	ЦЕХ 23	А-5-236	2011	5-ТИ ОСЕВОЙ ОБРАБАТЫВАЮЩИЙ ЦЕНТР	PICOMAX 825 VERSA	04.03.2019	демонтаж линейки по Х	загрязнение линейки	демонтаж-монтаж-промывка	04.03.2019
23-09-013	ЦЕХ 23	А-5-236	2011	5-ТИ ОСЕВОЙ ОБРАБАТЫВАЮЩИЙ ЦЕНТР	PICOMAX 825 VERSA	07.09.2023	нестабильное давление воздуха, сбой по тормозу оси А	нестабильное давление воздуха	замена трубопровода, наладка с-мы пневматики	11.09.2023
15-03-113	Отдел станков с программным управлением	А-5-236	2011	5-ТИ ОСЕВОЙ ОБРАБАТЫВАЮЩИЙ ЦЕНТР	PICOMAX 825 VERSA	19.03.2015	ключ н/р			19.03.2015
20-06-002	ЦЕХ 25	А-5-235	2011	5-ТИ ОСЕВОЙ ОБРАБАТЫВАЮЩИЙ ЦЕНТР	PICOMAX 825 VERSA	01.06.2020	не работает	не работает	перезагрузка системы	01.06.2020
24-05-182	ЦЕХ 23	А-5-239	2012	5-ТИ ОСЕВОЙ ОБРАБАТЫВАЮЩИЙ ЦЕНТР	PICOMAX 825 VERSA	29.05.2024	нехватка фреона	нехватка фреона	пайка трубки, вакуумирование, заправка фреоном	03.06.2024
14-08-164	ЦЕХ 28	А-5-235	2011	5-ТИ ОСЕВОЙ ОБРАБАТЫВАЮЩИЙ ЦЕНТР	PICOMAX 825 VERSA	21.08.2014	проверить пневматику		пневматика работает , произвели замену фильтра	21.08.2014
21-02-092	ЦЕХ 25	А-5-235	2011	5-ТИ ОСЕВОЙ ОБРАБАТЫВАЮЩИЙ ЦЕНТР	PICOMAX 825 VERSA	24.02.2021	Онибка №19. Неисправ. охлаж. шпинделя.	Ошибка №19	Диагностика, чистка конденсатора	24.02.2021
22-10-103	ЦЕХ 23	А-5-239	2012	5-ТИ ОСЕВОЙ ОБРАБАТЫВАЮЩИЙ ЦЕНТР	PICOMAX 825 VERSA	21.10.2022	нет замера контрольного инструмента	Загрязнение системы замера инструмента	Чистка лазера	21.10.2022
19-01-034	ЦЕХ 23	А-5-236	2011	5-ТИ ОСЕВОЙ ОБРАБАТЫВАЮЩИЙ ЦЕНТР	PICOMAX 825 VERSA	14.01.2019	нет откр-закр дверцы смены инструмента	сбой сист. снабжен.воздухом	перезагрузка системы и сброс ошибки	16.01.2019
23-04-156	ЦЕХ 23	А-5-236	2011	5-ТИ ОСЕВОЙ ОБРАБАТЫВАЮЩИЙ ЦЕНТР	PICOMAX 825 VERSA	25.04.2023	лазер "не видит"	загрязнение лазера изм инструмента	чистка лазера	26.04.2023
23-01-010	ЦЕХ 23	А-5-239	2012	5-ТИ ОСЕВОЙ ОБРАБАТЫВАЮЩИЙ ЦЕНТР	PICOMAX 825 VERSA	09.01.2023	села батарейка	села батарейка	замена буферной батарейки	11.01.2023
17-08-133	ЦЕХ 28	А-5-235	2011	5-ТИ ОСЕВОЙ ОБРАБАТЫВАЮЩИЙ ЦЕНТР	PICOMAX 825 VERSA	17.08.2017	не поступает смазка на направляющие	выход из строя питателей	замена питателей	18.08.2017

Рисунок А.1 - Отчет

ПРИЛОЖЕНИЕ Б

Объединение и преобразование заявок в один файл

```
import pandas as pd
import numpy as np
from datetime import date
from datetime import datetime
import time
```

```
df_1 = pd.read_excel('1.xlsx')
df_1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8 entries, 0 to 7
Data columns (total 11 columns):
 #   Column                  Non-Null Count  Dtype         
---  ------                  --------------  -----         
 0   № заявки                8 non-null      object        
 1   Цех                     8 non-null      object        
 2   Инв. номер              8 non-null      object        
 3   Год выпуска             8 non-null      int64         
 4   Оборудование            8 non-null      object        
 5   Модель                  8 non-null      object        
 6   Дата открытия           8 non-null      datetime64[ns]
 7   Описание неисправ.      8 non-null      object        
 8   Причина неисправности   7 non-null      object        
 9   Принятые меры           8 non-null      object        
 10  Дата закрытия заявки    8 non-null      datetime64[ns]
dtypes: datetime64[ns](2), int64(1), object(8)
memory usage: 832.0+ bytes
```

```
df_2 = pd.read_excel('2.xlsx')
df_3 = pd.read_excel('3.xlsx')
df_4 = pd.read_excel('4.xlsx')
df_5 = pd.read_excel('5.xlsx')
df_6 = pd.read_excel('6.xlsx')
df_7 = pd.read_excel('7.xlsx')
df_8 = pd.read_excel('8.xlsx')
df_9 = pd.read_excel('9.xlsx')
```

Рисунок Б.1 – Импорт файлов

```
df = pd.concat([df_1, df_2, df_3, df_4, df_5, df_6, df_7, df_8, df_9], ignore_index=True, axis=0)
df
```

	№ заявки	Цех	Инв. номер	Год выпуска	Оборудование	Модель	Дата открытия	Описание неисправ.	Причина неисправности	Принятые меры	Дата закрытия заявки
0	24-07-007	ЦЕХ 23	А-5-243	2013	Высокоскоростной обрабатывающий центр	МШ 600	2024-07-01	ошибка 132 по оси В	Повреждение Кабеля датчика \положения оси С.	Замена кабеля.	2024-10-17
1	24-07-210	ЦЕХ 19	Ш-8-150	2005	Станок зубошлифовальный	P600/800G	2024-07-29	ош замка загрузочной двери	ош замка загрузочной двери	рег-ка замков	2024-07-30
2	24-10-073	ЦЕХ 19	Ш-8-151	2012	Станок зубошлифовальный с ЧПУ	G 30	2024-10-11	неисправность шпинделя инструмента	NaN	неисправность устранена	2024-11-05
3	24-08-191	ЦЕХ 20	Ш-5-596	2002	Станок шлифовальный специальный с ЧПУ	MICRO-CUT-4	2024-08-30	ош М3-верхн прав ролик, М4-нижн прав ролик, по...	сработал автомат по питанию 24 в	включили автомат	2024-09-02
4	24-10-141	ЦЕХ 3	Г-6/229	1985	УСТАНОВКА ВАКУУМНАЯ	УППФ-3М	2024-10-23	ПТЧТ не корректно работает	не вкл.ПТЧТ	замена платы управления	2024-10-24
...	...	...	...	...	...	...	...	...	...	...	...
25439	23-12-094	ЦЕХ 3	Э-5-620	2014	Электроискровой координатно-прошивочный станок...	AD35L LN2	2023-12-18	завис комп	нет вкл компьютера	диагностика	2024-01-31
25440	22-04-133	ЦЕХ 3	Э-5-619	2014	Электроискровой координатно-прошивочный станок...	AD35L LN2	2022-04-26	ОШ:00201	Нет прожига	Диагностика	2022-06-07
25441	24-06-183	ЦЕХ 39	Э-5-617	2012	Электроискровой координатно-пршивочный станок	AG 60L LP2	2024-06-28	Ошибка E00509 (4030) - не берётся инструмент и...	неисправна система выбора инструмента	вызов представителя SODIC	NaT
25442	22-12-108	ЦЕХ 20	Э-5-634	2018	Электроэрозионный копировально-прошивной стано...	FORM P 350	2022-12-22	ош блока ЧПУ	Не загружается система	Диагностика, переустановка системы	2023-03-15
25443	23-07-016	ЦЕХ 20	Э-5-635	2018	Электроэрозионный копировально-прошивной стано...	FORM P 350	2023-07-04	завис	неисправен жесткий диск	работает представитель фирмы	2023-09-11

25444 rows × 11 columns

Рисунок Б.2 – Объединение файлов в один файл

```
df["Дата закрытия заявки"].fillna(time.strftime("%Y-%m-%d"), inplace=True)
df["Дата закрытия заявки"]
```

```
<ipython-input-77-0b44d8ca0c80>:1: FutureWarning: A value is trying to be set
The behavior will change in pandas 3.0. This inplace method will never work be

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.m
```

```
df["Дата закрытия заявки"].fillna(time.strftime("%Y-%m-%d"), inplace=True)
```

	Дата закрытия заявки
0	2024-10-17
1	2024-07-30
2	2024-11-05
3	2024-09-02
4	2024-10-24
...	...
25439	2024-01-31
25440	2022-06-07
25441	2024-12-13
25442	2023-03-15
25443	2023-09-11

25444 rows × 1 columns

Рисунок Б.3 – Заполнение пропусков даты закрытия заявки

```
df['длительность ремонта'] = df['Дата закрытия заявки'] - df['Дата открытия']
df
```

	№ заявки	Цех	Инв. номер	Год выпуска	Оборудование	Модель	Дата открытия	Описание неисправ.	Причина неисправности	Принятые меры	Дата закрытия заявки	длительность ремонта
0	24-07-007	ЦЕХ 23	А-5-243	2013	Высокоскоростной обрабатывающий центр	МШ 600	2024-07-01	ошибка 132 по оси В	Повреждение Кабеля датчика \положения оси С.	Замена кабеля.	2024-10-17	108 days
1	24-07-210	ЦЕХ 19	Ш-8-150	2005	Станок зубошлифовальный	P600/800G	2024-07-29	ош замка загрузочной двери	ош замка загрузочной двери	рег-ка замков	2024-07-30	1 days
2	24-10-073	ЦЕХ 19	Ш-8-151	2012	Станок зубошлифовальный с ЧПУ	G 30	2024-10-11	неисправность шпинделя инструмента	NaN	неисправность устранена	2024-11-05	25 days
3	24-08-191	ЦЕХ 20	Ш-5-596	2002	Станок шлифовальный специальный с ЧПУ	MICRO-CUT-4	2024-08-30	ош М3-верхн прав ролик, М4-нижн прав ролик, по...	сработал автомат по питанию 24 в	включили автомат	2024-09-02	3 days
4	24-10-141	ЦЕХ 3	Г-6/229	1985	УСТАНОВКА ВАКУУМНАЯ	УППФ-3М	2024-10-23	ПТЧТ не корректно работает	не вкл.ПТЧТ	замена платы управления	2024-10-24	1 days
...	...	...	...	...	...	...	...	...	...	...	...	...
25439	23-12-094	ЦЕХ 3	Э-5-620	2014	Электроискровой координатно-прошивочный станок...	AD35L LN2	2023-12-18	завис комп	нет вкл компьютера	диагностика	2024-01-31	44 days
25440	22-04-133	ЦЕХ 3	Э-5-619	2014	Электроискровой координатно-прошивочный станок...	AD35L LN2	2022-04-26	ОШ:00201	Нет прожига	Диагностика	2022-06-07	42 days
25441	24-06-183	ЦЕХ 39	Э-5-617	2012	Электроискровой координатно-пршивочный станок	AG 60L LP2	2024-06-28	Ошибка E00509 (4030) - не берётся инструмент и...	неисправна система выбора инструмента	вызов представителя SODIC	2024-12-13	168 days
25442	22-12-108	ЦЕХ 20	Э-5-634	2018	Электроэрозионный копировально-прошивной стано...	FORM P 350	2022-12-22	ош блока ЧПУ	Не загружается система	Диагностика, переустановка системы	2023-03-15	83 days
25443	23-07-016	ЦЕХ 20	Э-5-635	2018	Электроэрозионный копировально-прошивной стано...	FORM P 350	2023-07-04	завис	неисправен жесткий диск	работает представитель фирмы	2023-09-11	69 days

25444 rows × 12 columns

Рисунок Б.4 – Расчет продолжительности ремон

Рисунок Б.4 – Расчет длительности ремонта

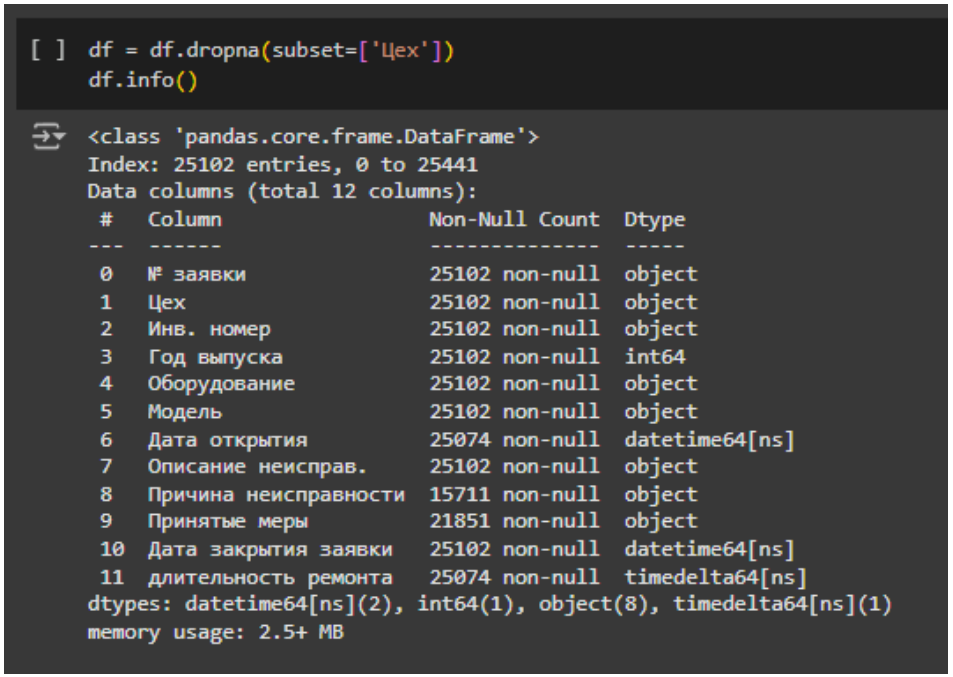
Удалим дубликаты, если они имеются (было удалено 90 повторяющихся строк)

```
[ ] df = df.drop_duplicates()
    df
```

	№ заявки	Цех	Инв. номер	Год выпуска	Оборудование	Модель	Дата открытия	Описание неисправ.	Причина неисправности	Принятые меры	Дата закрытия заявки	длительность ремонта
0	24-07-007	ЦЕХ 23	А-5-243	2013	высокоскоростной обрабатывающий центр	МШ 600	2024-07-01	ошибка 132 по оси в	повреждение кабеля датчика \положения оси с.	замена кабеля.	2024-10-17	108 days
1	24-07-210	ЦЕХ 19	Ш-8-150	2005	станок зубошлифовальный	P600/800G	2024-07-29	ош замка загрузочной двери	ош замка загрузочной двери	рег-ка замков	2024-07-30	1 days
2	24-10-073	ЦЕХ 19	Ш-8-151	2012	станок зубошлифовальный с чпу	G 30	2024-10-11	неисправность шпинделя инструмента	NaN	неисправность устранена	2024-11-05	25 days
3	24-08-191	ЦЕХ 20	Ш-5-596	2002	станок шлифовальный специальный с чпу	MICRO-CUT-4	2024-08-30	ош м3-верхн прав ролик, м4-нижн прав ролик, по...	сработал автомат по питанию 24 в	включили автомат	2024-09-02	3 days
4	24-10-141	ЦЕХ 3	Г-6/229	1985	установка вакуумная	УППФ-3М	2024-10-23	птчт не корректно работает	не вкл.птчт	замена платы управления	2024-10-24	1 days
...	...	...	...	...	...	...	...	...	...	...	...	...
25413	24-08-165	ЦЕХ 16	У-1-157	2021	установка аргоно-дуговой сварки	MC-315T2 AC/DC	2024-08-27	нет дуги	нет дуги	диагностика, требуется вызов специалистов	2024-12-13	108 days
25416	22-06-120	ЦЕХ 20	У-6-650	2021	установка ионно-плазменного нанесения покрытий	АПН-250М-3	2022-06-20	вышел из строя блок смещения	н/р блок смещения	диагностика	2024-03-28	647 days
25418	24-08-151	ЦЕХ 20	Э-5-1557	2022	установка ультразвукового упрочнения	Sk-UIT100	2024-08-26	не выходит в раб режим	неисправностей не выявлено.	требуется диагностика специалистами изготовите...	2024-12-13	109 days
25429	21-04-051	ЦЕХ 3	Э-5/174	1974	эл.эрозионн.копиров.-прошивочн.ст.	4Г721М	2021-04-14	нет подачи на амперметр	нет на эрозионном промежутке силового тех. тока.	диагностика	2021-04-23	9 days
25441	24-06-183	ЦЕХ 39	Э-5-617	2012	электроискровой координатно-пршивочный станок	AG 60L LP2	2024-06-28	ошибка e00509 (4030) - не берётся инструмент и...	неисправна система выбора инструмента	вызов представителя sodic	2024-12-13	168 days

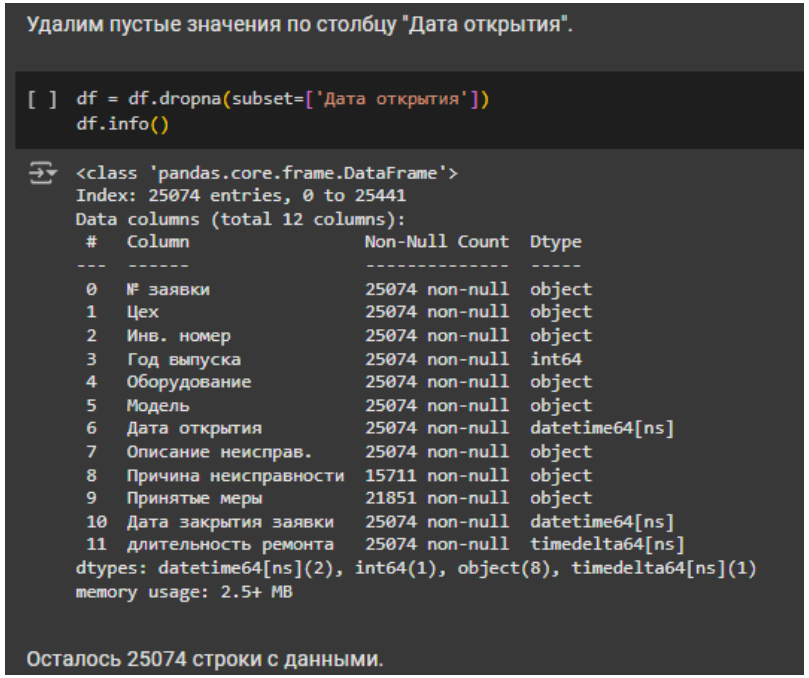
25354 rows × 12 columns

Рисунок Б.5 – Удаление дубликатов



```
[ ]  df = df.dropna(subset=['Цех'])
     df.info()

<class 'pandas.core.frame.DataFrame'>
Index: 25102 entries, 0 to 25441
Data columns (total 12 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   № заявки               25102 non-null  object
 1   Цех                    25102 non-null  object
 2   Инв. номер             25102 non-null  object
 3   Год выпуска            25102 non-null  int64
 4   Оборудование           25102 non-null  object
 5   Модель                 25102 non-null  object
 6   Дата открытия          25074 non-null  datetime64[ns]
 7   Описание неисправ.     25102 non-null  object
 8   Причина неисправности  15711 non-null  object
 9   Принятые меры          21851 non-null  object
 10  Дата закрытия заявки   25102 non-null  datetime64[ns]
 11  длительность ремонта   25074 non-null  timedelta64[ns]
dtypes: datetime64[ns](2), int64(1), object(8), timedelta64[ns](1)
memory usage: 2.5+ MB
```



Удалим пустые значения по столбцу "Дата открытия".

```
[ ]  df = df.dropna(subset=['Дата открытия'])
     df.info()

<class 'pandas.core.frame.DataFrame'>
Index: 25074 entries, 0 to 25441
Data columns (total 12 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   № заявки               25074 non-null  object
 1   Цех                    25074 non-null  object
 2   Инв. номер             25074 non-null  object
 3   Год выпуска            25074 non-null  int64
 4   Оборудование           25074 non-null  object
 5   Модель                 25074 non-null  object
 6   Дата открытия          25074 non-null  datetime64[ns]
 7   Описание неисправ.     25074 non-null  object
 8   Причина неисправности  15711 non-null  object
 9   Принятые меры          21851 non-null  object
 10  Дата закрытия заявки   25074 non-null  datetime64[ns]
 11  длительность ремонта   25074 non-null  timedelta64[ns]
dtypes: datetime64[ns](2), int64(1), object(8), timedelta64[ns](1)
memory usage: 2.5+ MB
```

Осталось 25074 строки с данными.

Рисунок Б.6 – Удаление пустых значений по двум столбцам

ПРИЛОЖЕНИЕ В

Парсинг сайта погоды

```python
import requests
from lxml import html
import pandas as pd
import urllib.parse

all_weather = []

url = 'https://ginfo.ru'
url_join = None
year = 2019


for year in range(2019, 2025):
  url_join = 'https://ginfo.ru/' + f"/pogoda-{year}"
  resp = requests.get(url_join)
  tree = html.fromstring(html = resp.content)
  weathers = tree.xpath("//a[@class = 'pogoda_day']")

  for weather in weathers:
    w = {
    'day' : weather.xpath(".//div[1]/text()")[0],
    'month' : weather.xpath(".//div[1]/span[2]/text()")[0],
    'year' : weather.xpath("//div[@class = 'path_list']/span[3]/text()")[0],
    'temp' : weather.xpath(".//div[3]/text()")[0]}

    all_weather.append(w)
    year = year + 1
```

```python
df = pd.DataFrame(all_weather)
df
```

	day	month	year	temp
0	1	янв.	2019	−3
1	2	янв.	2019	−2
2	3	янв.	2019	−2
3	4	янв.	2019	−3
4	5	янв.	2019	−5
...	...	...	...	...
2186	27	дек.	2024	+1
2187	28	дек.	2024	+1
2188	29	дек.	2024	+2
2189	30	дек.	2024	−2
2190	31	дек.	2024	−3

2191 rows × 4 columns

Рисунок В.1 – Парсинг данных

```
df['month'] = df['month'].map({'янв.': '01',
                               'февр.': '02',
                               'мар.': '03',
                               'апр.': '04',
                               'мая': '05',
                               'июн.': '06',
                               'июл.': '07',
                               'авг.': '08',
                               'сент.': '09',
                               'окт.': '10',
                               'нояб.': '11',
                               'дек.': '12'})
```

```
df['data'] = df['year'].map(str)+'.' + df['month'].map(str) + '.' + df['day'].map(str).str.strip()
#df['data'] = df['day'].map(str).str.strip()+'.' + df['month'].map(str) + '.' + df['year'].map(str)
print(df)
```

```
      day month  year temp        data
0       1    01  2019   -3  2019.01.1
1       2    01  2019   -2  2019.01.2
2       3    01  2019   -2  2019.01.3
3       4    01  2019   -3  2019.01.4
4       5    01  2019   -5  2019.01.5
...   ...   ...   ...  ...         ...
2186   27    12  2024   +1  2024.12.27
2187   28    12  2024   +1  2024.12.28
2188   29    12  2024   +2  2024.12.29
2189   30    12  2024   -2  2024.12.30
2190   31    12  2024   -3  2024.12.31

[2191 rows x 5 columns]
```

Рисунок В.2 – Преобразование даты

```
df['data'] = df['data'].astype('datetime64[ns]')
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2191 entries, 0 to 2190
Data columns (total 5 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   day     2191 non-null   object
 1   month   2191 non-null   object
 2   year    2191 non-null   object
 3   temp    2191 non-null   object
 4   data    2191 non-null   datetime64[ns]
dtypes: datetime64[ns](1), object(4)
memory usage: 85.7+ KB
```

Рисунок В.3 – Преобразование даты

```
df['temp'] = df['temp'].str.replace('+', '')
df
```

	day	month	year	temp	data
0	1	01	2019	-3	2019-01-01
1	2	01	2019	-2	2019-01-02
2	3	01	2019	-2	2019-01-03
3	4	01	2019	-3	2019-01-04
4	5	01	2019	-5	2019-01-05
...	...	...	...	...	...
2186	27	12	2024	1	2024-12-27
2187	28	12	2024	1	2024-12-28
2188	29	12	2024	2	2024-12-29
2189	30	12	2024	-2	2024-12-30
2190	31	12	2024	-3	2024-12-31

2191 rows × 5 columns

```
df['temp'] = df['temp'].str.replace('−', '-')
df
```

	day	month	year	temp	data
0	1	01	2019	-3	2019-01-01
1	2	01	2019	-2	2019-01-02
2	3	01	2019	-2	2019-01-03
3	4	01	2019	-3	2019-01-04
4	5	01	2019	-5	2019-01-05
...	...	...	...	...	...
2186	27	12	2024	1	2024-12-27
2187	28	12	2024	1	2024-12-28
2188	29	12	2024	2	2024-12-29
2189	30	12	2024	-2	2024-12-30
2190	31	12	2024	-3	2024-12-31

2191 rows × 5 columns

Рисунок В.4 – Преобразование температуры

```
df['temp'] = df['temp'].astype(int)
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2191 entries, 0 to 2190
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   temp    2191 non-null   int64
 1   data    2191 non-null   datetime64[ns]
dtypes: datetime64[ns](1), int64(1)
memory usage: 34.4 KB
```

Рисунок В.5 – Преобразование температуры

ПРИЛОЖЕНИЕ Г

Объединение данных

```
import pandas as pd

df_requests = pd.read_csv('requests.csv')
df_requests.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25074 entries, 0 to 25073
Data columns (total 13 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   Unnamed: 0             25074 non-null  int64
 1   № заявки               25074 non-null  object
 2   Цех                    25074 non-null  object
 3   Инв. номер             25074 non-null  object
 4   Год выпуска            25074 non-null  int64
 5   Оборудование           25074 non-null  object
 6   Модель                 25074 non-null  object
 7   Дата открытия          25074 non-null  object
 8   Описание неисправ.     25074 non-null  object
 9   Причина неисправности  15711 non-null  object
 10  Принятые меры          21851 non-null  object
 11  Дата закрытия заявки   25074 non-null  object
 12  длительность ремонта   25074 non-null  object
dtypes: int64(2), object(11)
memory usage: 2.5+ MB
```

```
df_weather = pd.read_csv('weather.csv')
df_weather.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2191 entries, 0 to 2190
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Unnamed: 0  2191 non-null   int64
 1   day         2191 non-null   int64
 2   month       2191 non-null   int64
 3   year        2191 non-null   int64
 4   temp        2191 non-null   int64
 5   data        2191 non-null   object
dtypes: int64(5), object(1)
memory usage: 102.8+ KB
```

Рисунок Г.1 – Импорт данных

```
df_weather['data'] = df_weather['data'].astype('datetime64[ns]')
df_weather.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2191 entries, 0 to 2190
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Unnamed: 0  2191 non-null   int64
 1   day         2191 non-null   int64
 2   month       2191 non-null   int64
 3   year        2191 non-null   int64
 4   temp        2191 non-null   int64
 5   data        2191 non-null   datetime64[ns]
dtypes: datetime64[ns](1), int64(5)
memory usage: 102.8 KB
```

```
df_requests['data'] = df_requests['Дата открытия']
df_requests.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25074 entries, 0 to 25073
Data columns (total 14 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   Unnamed: 0             25074 non-null  int64
 1   № заявки               25074 non-null  object
 2   Цех                    25074 non-null  object
 3   Инв. номер             25074 non-null  object
 4   Год выпуска            25074 non-null  int64
 5   Оборудование           25074 non-null  object
 6   Модель                 25074 non-null  object
 7   Дата открытия          25074 non-null  datetime64[ns]
 8   Описание неисправ.     25074 non-null  object
 9   Причина неисправности  15711 non-null  object
 10  Принятые меры          21851 non-null  object
 11  Дата закрытия заявки   25074 non-null  datetime64[ns]
 12  длительность ремонта   25074 non-null  object
 13  data                   25074 non-null  datetime64[ns]
dtypes: datetime64[ns](3), int64(2), object(9)
memory usage: 2.7+ MB
```

Рисунок Г.2 – Изменение типа данных даты

```
df_merged = df_requests.merge(df_weather, on='data', how='outer')
df_merged
```

	Unnamed: 0_x	№ заявки	Цех	Инв. номер	Год выпуска	Оборудование	Модель	Дата открытия	Описание неисправ.	Причина неисправности	Принятые меры	Дата закрытия заявки	длительность ремонта	data	Unnamed: 0_y	temp
0	23977.0	18-04-176	ВМЗ Цех № 44	А-5-158	1987.0	фрезер.вертик.специал.с чпу	МА655С630	2008-04-26	повышенный шум	NaN	NaN	2024-12-13	6075 days 00:00:00	2008-04-26	NaN	NaN
1	23147.0	12-11-112	ЦЕХ 43	Т-1/1864	1995.0	токарно-винтор. станок с чпу	MDW-10	2011-12-21	закупка з/частей	NaN	закупка з/частей	2012-11-15	330 days 00:00:00	2011-12-21	NaN	NaN
2	22472.0	12-11-109	ЦЕХ 43	Ш-2-261	2002.0	станок координатно-шлифовальный	S55-400	2012-01-11	закупка матрицы	NaN	закупка матрицы	2012-11-15	309 days 00:00:00	2012-01-11	NaN	NaN
3	21111.0	12-11-118	ЦЕХ 26	А-4/47	1961.0	координатно-расточной станок	2А3	2012-02-07	закупка оптич.атчика	NaN	закупка оптич.датчика	2012-11-15	282 days 00:00:00	2012-02-07	NaN	NaN
4	24043.0	12-11-111	ЦЕХ 43	Ф-5/699	1991.0	фрезерный ст-к с чпу	МН-1000S	2012-03-12	закупка платы изм.сист.	NaN	закупка платы	2012-11-15	248 days 00:00:00	2012-03-12	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
25813	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaT	NaN	NaN	NaN	NaT	NaN	2024-12-27	2186.0	3.0
25814	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaT	NaN	NaN	NaN	NaT	NaN	2024-12-28	2187.0	1.0
25815	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaT	NaN	NaN	NaN	NaT	NaN	2024-12-29	2188.0	-5.0
25816	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaT	NaN	NaN	NaN	NaT	NaN	2024-12-30	2189.0	-2.0
25817	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaT	NaN	NaN	NaN	NaT	NaN	2024-12-31	2190.0	0.0

25818 rows × 16 columns

```
df_merged = df_merged.drop(['Unnamed: 0_x', 'Unnamed: 0_y'], axis=1)
```

Рисунок Г.3 – Объединение таблиц по дате

Удалим данные, где отсутствуют температура.

```
df_merged = df_merged.dropna(subset=['temp'])
df_merged.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 11279 entries, 14532 to 25817
Data columns (total 14 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   № заявки               10535 non-null  object
 1   Цех                    10535 non-null  object
 2   Инв. номер             10535 non-null  object
 3   Год выпуска            10535 non-null  float64
 4   Оборудование           10535 non-null  object
 5   Модель                 10535 non-null  object
 6   Дата открытия          10535 non-null  datetime64[ns]
 7   Описание неисправ.     10535 non-null  object
 8   Причина неисправности  10342 non-null  object
 9   Принятые меры          10375 non-null  object
 10  Дата закрытия заявки   10535 non-null  datetime64[ns]
 11  длительность ремонта   10535 non-null  object
 12  data                   11279 non-null  datetime64[ns]
 13  temp                   11279 non-null  float64
dtypes: datetime64[ns](3), float64(2), object(9)
memory usage: 1.3+ MB
```

```
[ ] df_merged = df_merged.dropna(subset=['Цех'])
    df_merged.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 10535 entries, 14540 to 25769
Data columns (total 14 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   № заявки               10535 non-null  object
 1   Цех                    10535 non-null  object
 2   Инв. номер             10535 non-null  object
 3   Год выпуска            10535 non-null  float64
 4   Оборудование           10535 non-null  object
 5   Модель                 10535 non-null  object
 6   Дата открытия          10535 non-null  datetime64[ns]
 7   Описание неисправ.     10535 non-null  object
 8   Причина неисправности  10342 non-null  object
 9   Принятые меры          10375 non-null  object
 10  Дата закрытия заявки   10535 non-null  datetime64[ns]
 11  длительность ремонта   10535 non-null  object
 12  data                   10535 non-null  datetime64[ns]
 13  temp                   10535 non-null  float64
dtypes: datetime64[ns](3), float64(2), object(9)
memory usage: 1.2+ MB
```

Рисунок Г.4 – Удаление пустых значений

```
df_merged[['Категория оборудования', 'Инв. номер']] = df_merged['Инв. номер'].str.split('-', n=1, expand=True)
df_merged
```

	№ заявки	Цех	Инв. номер	Год выпуска	Оборудование	Модель	Дата открытия	Описание неисправ.	Причина неисправности	Принятые меры	Дата закрытия заявки	data	temp	длительность ремонта	Категория оборудования
14540	19-01-006	ЦЕХ 22	5/40	1989.0	tokapho-pebolьb.ct. c чпу	1В340Ф30РМ	2019-01-09	не запускается	выход из строя гл привода	замена гл привода	2019-01-11	2019-01-09	24.0	2 days	Р
14541	19-01-011	ЦЕХ 23	5/675	1993.0	вертикально-фрезерный станок с чпу	6М13НК	2019-01-09	нет движения по z	ош оператора	диагностика	2019-01-09	2019-01-09	24.0	0 days	Ф
14542	19-01-015	ЦЕХ 16	9/822	1985.0	источник питания	ИСВУ-315	2019-01-09	замкнуло плату, дым	вышел из строя электролит в релейном блоке	замена электролита	2019-01-11	2019-01-09	24.0	2 days	Э
14543	19-01-017	ЦЕХ 26	5-778	2013.0	обрабатывающий центр	E160A	2019-01-09	не выходит в 0	села батарейка, слетели параметры привязки осей	замена батарейки, волсстановление параметров п...	2019-01-15	2019-01-09	24.0	6 days	Т
14544	19-01-007	ЦЕХ 16	3/633	2000.0	печь вертикальная	6818-2185	2019-01-09	не вкл	неисправна плата тпчт	ремонт платы	2019-01-15	2019-01-09	24.0	6 days	Г
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
25765	24-11-051	ЦЕХ 23	5-221	2007.0	5-ти координатный обрабатывающий центр	go-Mill 350	2024-11-11	ош 04, 403, 419, шторка не закрывается	плохой контакт в замке шторки смены инструмента	диагностика и регулировка замка	2024-11-13	2024-11-11	1.0	2 days	А
25766	24-11-046	ЦЕХ 22	1-2698	2007.0	токарный ст-к с чпу	SCHAUBLIN-140R CNC	2024-11-11	не держит размер, шум при перемещении	раскрутилась затяжная гайка на швп	подтянули гайку на швп	2024-11-13	2024-11-11	1.0	2 days	Т
25767	24-11-047	ЦЕХ 15	3/537	1985.0	эл.печь камер.закупн.	FHV90GHS	2024-11-11	ремонт вак с-мы	заварена трещина.	заварена трещина.	2024-11-14	2024-11-11	1.0	3 days	Г
25768	24-11-048	ЦЕХ 15	3/538	1985.0	эл.печь камер.закупн.	FHV90GHS	2024-11-11	проверка герметичности	проверка выполнена.	проверка выполнена.	2024-11-13	2024-11-11	1.0	2 days	Г
25769	24-11-054	ЦЕХ 20	6-622	2017.0	установка хромирование газовым методом	УМДП-И2	2024-11-12	проверка герметичности	замена вакуумного уплотнения.	замена вакуумного уплотнения.	2024-11-13	2024-11-12	-4.0	1 days	У

10535 rows × 15 columns

Рисунок Г.5 – Преобразование данных

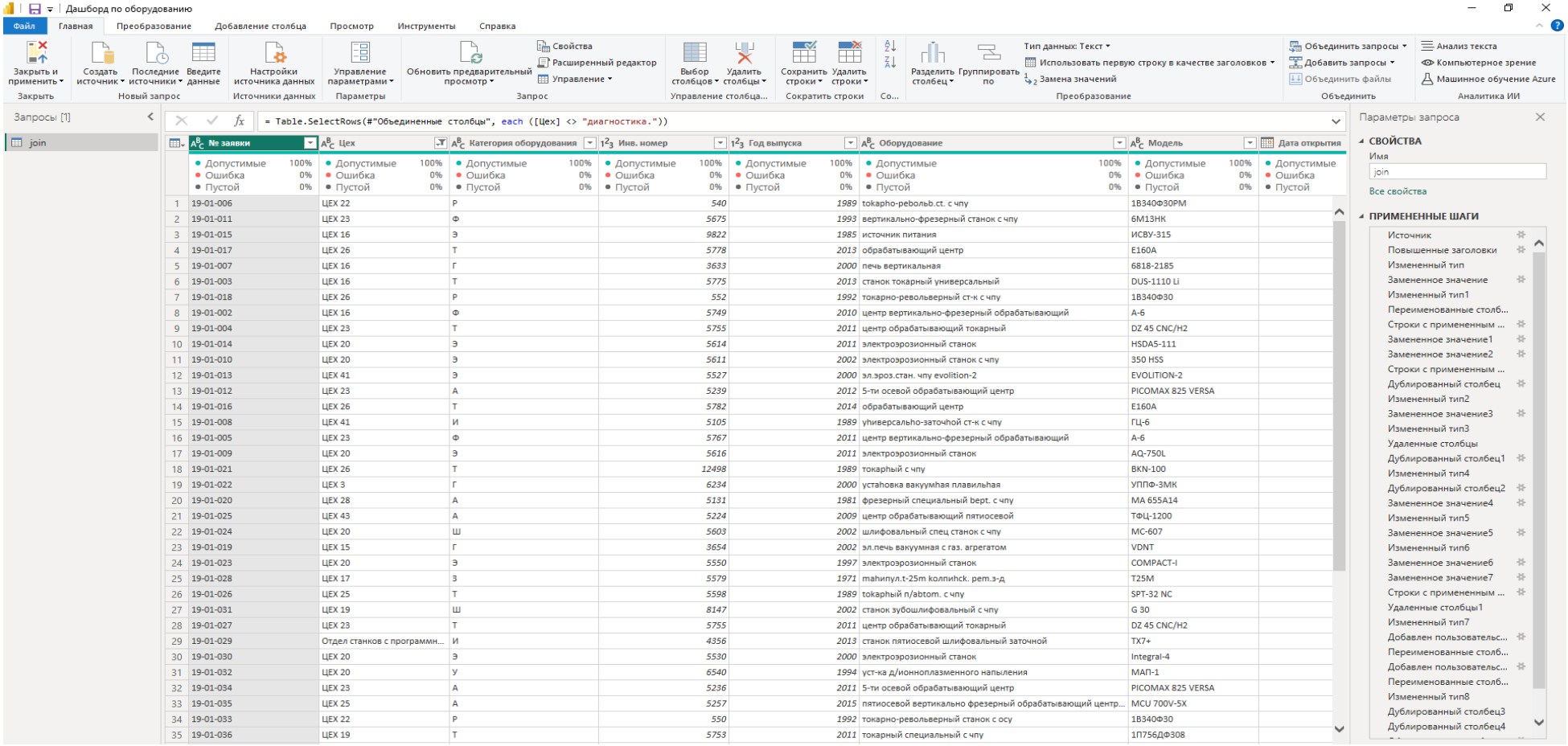
```
df_merged_new['Год выпуска'] = df_merged_new['Год выпуска'].astype(int)
df_merged_new.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 10535 entries, 14540 to 25769
Data columns (total 14 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   № заявки                10535 non-null  object
 1   Цех                     10535 non-null  object
 2   Категория оборудования  10535 non-null  object
 3   Инв. номер              10535 non-null  object
 4   Год выпуска             10535 non-null  int64
 5   Оборудование            10535 non-null  object
 6   Модель                  10535 non-null  object
 7   Дата открытия           10535 non-null  datetime64[ns]
 8   Дата закрытия заявки    10535 non-null  datetime64[ns]
 9   длительность ремонта    10535 non-null  timedelta64[ns]
 10  temp                    10535 non-null  float64
 11  Описание неисправ.      10535 non-null  object
 12  Причина неисправности   10342 non-null  object
 13  Принятые меры           10375 non-null  object
dtypes: datetime64[ns](2), float64(1), int64(1), object(9), timedelta64[ns](1)
memory usage: 1.2+ MB
```

Рисунок Г.7– Преобразование данных

ПРИЛОЖЕНИЕ Д

Объединение данных



	№ заявки	Цех	Категория оборудования	Инв. номер	Год выпуска	Оборудование	Модель	Дата открытия
	Допустимые 100% Ошибка 0% Пустой 0%	Допустимые 100% Ошибка 0% Пустой 0%	Допустимые 100% Ошибка 0% Пустой 0%	Допустимые 100% Ошибка 0% Пустой 0%	Допустимые 100% Ошибка 0% Пустой 0%	Допустимые 100% Ошибка 0% Пустой 0%	Допустимые 100% Ошибка 0% Пустой 0%	Допустимые Ошибка Пустой
1	19-01-006	ЦЕХ 22	Р	540	1989	токарно-револьв.ст. с чпу	1В340Ф30РМ	
2	19-01-011	ЦЕХ 23	Ф	5675	1993	вертикально-фрезерный станок с чпу	6М13НК	
3	19-01-015	ЦЕХ 16	Э	9822	1985	источник питания	ИСВУ-315	
4	19-01-017	ЦЕХ 26	Т	5778	2013	обрабатывающий центр	E160A	
5	19-01-007	ЦЕХ 16	Г	3633	2000	печь вертикальная	6818-2185	
6	19-01-003	ЦЕХ 16	Т	5775	2013	станок токарный универсальный	DUS-1110 Li	
7	19-01-018	ЦЕХ 26	Р	552	1992	токарно-револьверный ст-к с чпу	1В340Ф30	
8	19-01-002	ЦЕХ 16	Ф	5749	2010	центр вертикально-фрезерный обрабатывающий	A-6	
9	19-01-004	ЦЕХ 23	Т	5755	2011	центр обрабатывающий токарный	DZ 45 CNC/H2	
10	19-01-014	ЦЕХ 20	Э	5614	2011	электроэрозионный станок	HSDA5-111	
11	19-01-010	ЦЕХ 20	Э	5611	2002	электроэрозионный станок с чпу	350 HSS	
12	19-01-013	ЦЕХ 41	Э	5527	2000	эл.эроз.стан. чпу evolition-2	EVOLITION-2	
13	19-01-012	ЦЕХ 23	А	5239	2012	5-ти осевой обрабатывающий центр	PICOMAX 825 VERSA	
14	19-01-016	ЦЕХ 26	Т	5782	2014	обрабатывающий центр	E160A	
15	19-01-008	ЦЕХ 41	И	5105	1989	универсально-заточной ст-к с чпу	ГЦ-6	
16	19-01-005	ЦЕХ 23	Ф	5767	2011	центр вертикально-фрезерный обрабатывающий	A-6	
17	19-01-009	ЦЕХ 20	Э	5616	2011	электроэрозионный станок	AQ-750L	
18	19-01-021	ЦЕХ 26	Т	12498	1989	токарный с чпу	BKN-100	
19	19-01-022	ЦЕХ 3	Г	6234	2000	установка вакуумная плавильная	УППФ-3МК	
20	19-01-020	ЦЕХ 28	А	5131	1981	фрезерный специальный верт. с чпу	MA 655A14	
21	19-01-025	ЦЕХ 43	А	5224	2009	центр обрабатывающий пятиосевой	ТФЦ-1200	
22	19-01-024	ЦЕХ 20	Ш	5603	2002	шлифовальный спец станок с чпу	MC-607	
23	19-01-019	ЦЕХ 15	Г	3654	2002	эл.печь вакуумная с газ. агрегатом	VDNT	
24	19-01-023	ЦЕХ 20	Э	5550	1997	электроэрозионный станок	COMPACT-I	
25	19-01-028	ЦЕХ 17	З	5579	1971	манипул.т-25м колпичск. рем.з-д	T25M	
26	19-01-026	ЦЕХ 25	Т	5598	1989	токарный п/автом. с чпу	SPT-32 NC	
27	19-01-031	ЦЕХ 19	Ш	8147	2002	станок зубошлифовальный с чпу	G 30	
28	19-01-027	ЦЕХ 23	Т	5755	2011	центр обрабатывающий токарный	DZ 45 CNC/H2	
29	19-01-029	Отдел станков с программн...	И	4356	2013	станок пятиосевой шлифовальный заточной	TX7+	
30	19-01-030	ЦЕХ 20	Э	5530	2000	электроэрозионный станок	Integral-4	
31	19-01-032	ЦЕХ 20	У	6540	1994	уст-ка д/ионноплазменного напыления	МАП-1	
32	19-01-034	ЦЕХ 23	А	5236	2011	5-ти осевой обрабатывающий центр	PICOMAX 825 VERSA	
33	19-01-035	ЦЕХ 25	А	5257	2015	пятиосевой вертикально фрезерный обрабатывающий центр...	MCU 700V-5X	
34	19-01-033	ЦЕХ 22	Р	550	1992	токарно-револьверный станок с осу	1В340Ф30	
35	19-01-036	ЦЕХ 19	Т	5753	2011	токарный специальный с чпу	1П756ДФ308	

Рисунок Д.1 – Преобразование данных в Power Query

```
= Table.AddColumn(#"Дублированный столбец5", "ЧПУ", each if Text.Contains([Оборудование], "чпу") then "1" else "0")
```

```
= Table.AddColumn(#"Переименованные столбцы3", "Универсальное", each if Text.Contains([ЧПУ], "1") then "0" else "1")
```

Копия Оборудование		ЧПУ		Универсальное	
Допустимые	100%	Допустимые	100%	Допустимые	100%
Ошибка	0%	Ошибка	0%	Ошибка	0%
Пустой	0%	Пустой	0%	Пустой	0%
токарно-револьв.ст. с чпу		1		0	
вертикально-фрезерный станок с чпу		1		0	
источник питания		0		1	
обрабатывающий центр		0		1	
печь вертикальная		0		1	
станок токарный универсальный		0		1	
токарно-револьверный ст-к с чпу		1		0	
центр вертикально-фрезерный обрабатывающий		0		1	
центр обрабатывающий токарный		0		1	
электроэрозионный станок		0		1	
электроэрозионный станок с чпу		1		0	

Рисунок Д.2 – Выделение универсального оборудования и оборудования с ЧПУ

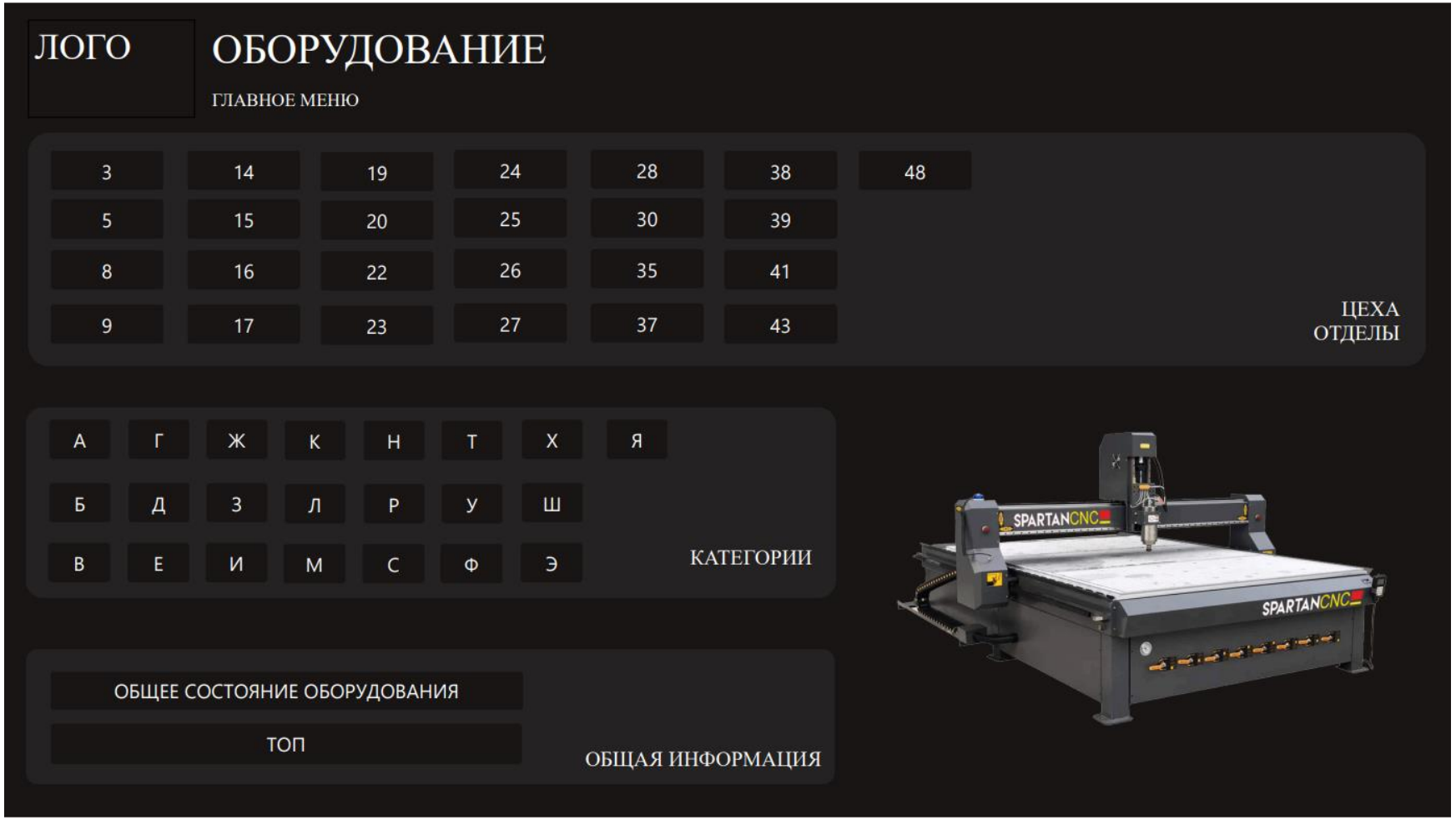


Рисунок Д.3 – Главное меню дашборда

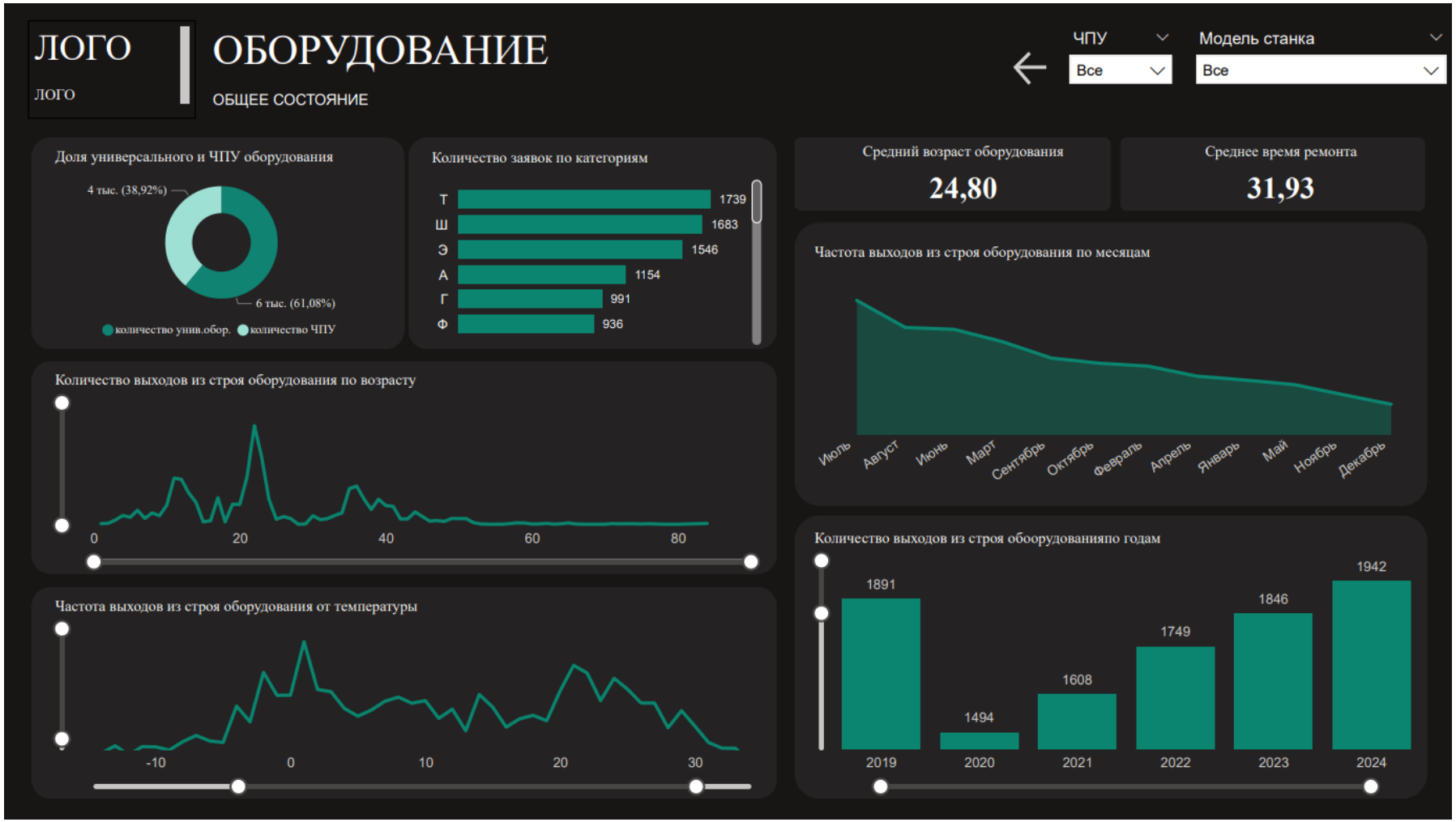


Рисунок Д.4 – Страница «Общее состояние»

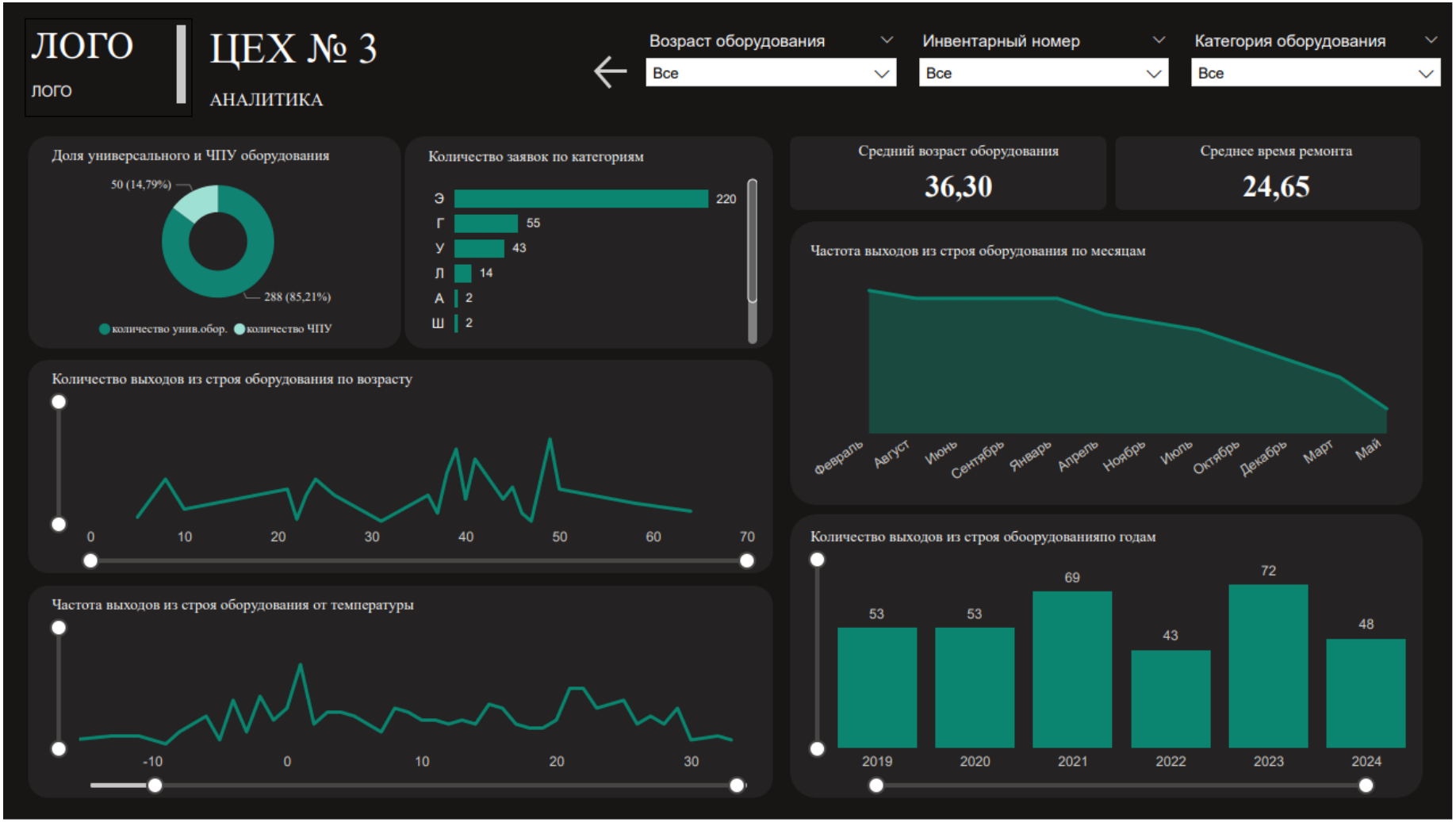


Рисунок Д.5 – Дашборд по отдельным цехам

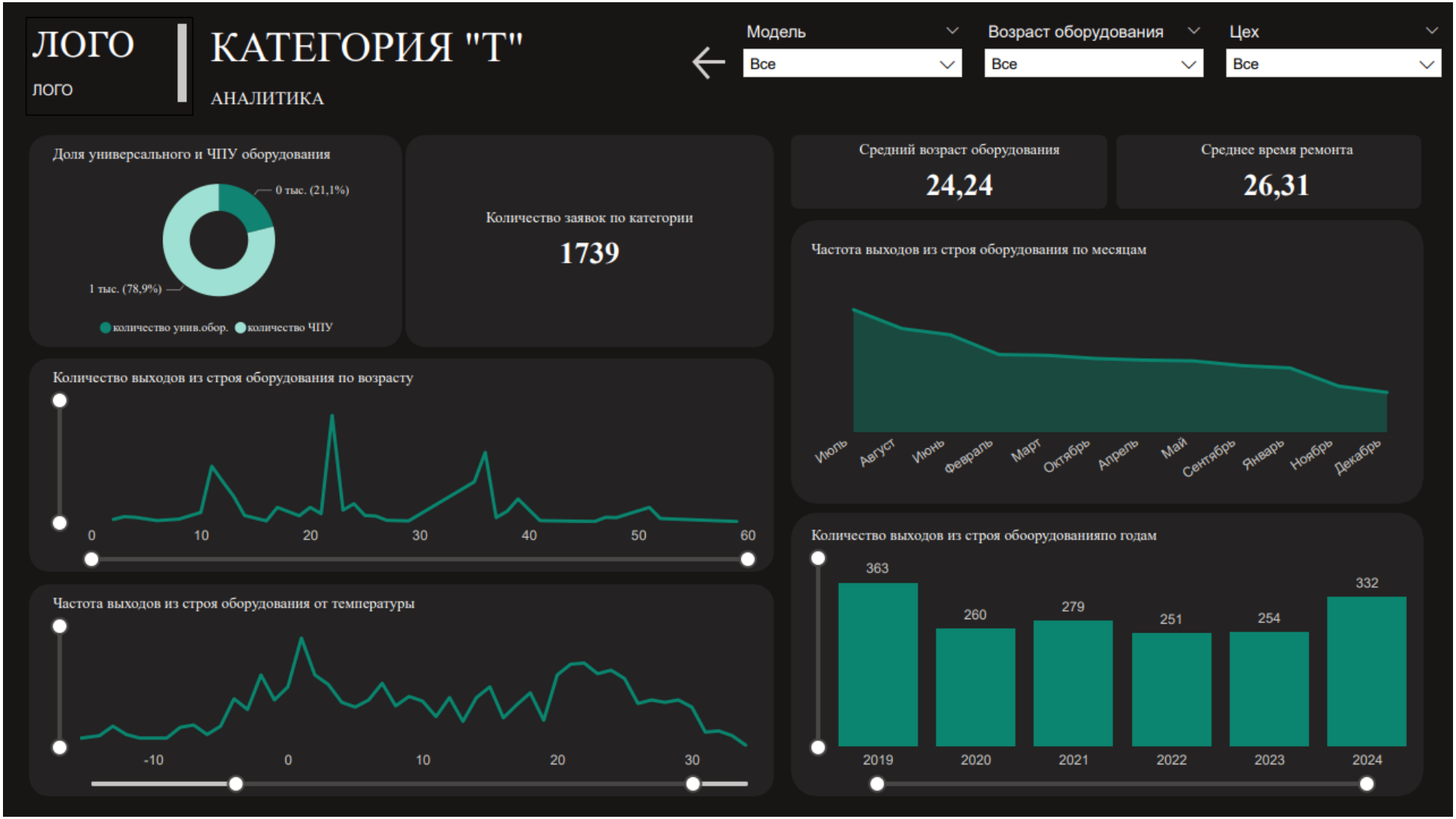


Рисунок Д.6 – Дашборд по отдельным категориям

Рисунок Д.6 – Страница «ТОП по поломкам»

ПРИЛОЖЕНИЕ Е

Модель ARIMA для прогноза поломок

```
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10535 entries, 0 to 10534
Data columns (total 19 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Unnamed: 0              10535 non-null  int64
 1   № заявки                10535 non-null  object
 2   Цех                     10535 non-null  object
 3   Инв. номер              10535 non-null  object
 4   Год выпуска             10535 non-null  int64
 5   Оборудование            10535 non-null  object
 6   Модель                  10535 non-null  object
 7   Дата открытия           10535 non-null  object
 8   Описание неисправ.      10535 non-null  object
 9   Причина неисправности   10342 non-null  object
 10  Принятые меры           10375 non-null  object
 11  Дата закрытия заявки    10535 non-null  object
 12  data                    10535 non-null  object
 13  day                     10535 non-null  int64
 14  month                   10535 non-null  int64
 15  year                    10535 non-null  int64
 16  temp                    10535 non-null  int64
 17  длительность ремонта    10535 non-null  object
 18  Категория оборудования  10535 non-null  object
dtypes: int64(6), object(13)
memory usage: 1.5+ MB
```

Проверка пропущенных значений

Обработка отсутствующих значений — важный этап предварительной обработки в анализе динамических рядов. Отсутствующие значения могут вызывать проблемы при анализе и искажать результаты прогнозирования. Для проверки пропущенных значений можно использовать метод isnull() из библиотеки pandas:

```
[ ] # @title
    print(df.isnull().sum())

Unnamed: 0                0
№ заявки                  0
Цех                       0
Инв. номер                0
Год выпуска               0
Оборудование              0
Модель                    0
Дата открытия             0
Описание неисправ.        0
Причина неисправности   193
Принятые меры           160
Дата закрытия заявки      0
data                      0
day                       0
month                     0
year                      0
temp                      0
длительность ремонта      0
Категория оборудования    0
dtype: int64
```

В нужных для нас столбцах пропуски отсутствуют

Рисунок Е.1 – Проверка пропущенных значений

```
date_counts = date_series.value_counts()
date_counts = date_counts.sort_index()
date_counts = pd.Series(date_counts.values, index=date_counts.index)
date_counts
```

	0
data	
2019-01-09	17
2019-01-10	7
2019-01-11	6
2019-01-12	1
2019-01-14	5
...	...
2024-11-06	3
2024-11-07	11
2024-11-08	1
2024-11-11	5
2024-11-12	1

1447 rows × 1 columns

```
full_date_range = pd.date_range(start=date_counts.index.min(), end=date_counts.index.max())
date_counts = date_counts.reindex(full_date_range, fill_value=0)
date_counts
```

	0
2019-01-09	17
2019-01-10	7
2019-01-11	6
2019-01-12	1
2019-01-13	0
...	...
2024-11-08	1
2024-11-09	0
2024-11-10	0
2024-11-11	5
2024-11-12	1

2135 rows × 1 columns

Рисунок Е.2 – Подготовка данных к обучению

Рисунок Е.3 – Зависимость поломок оборудования от температуры

```
from statsmodels.tsa.stattools import adfuller

dftest = adfuller(date_counts)
adf = dftest[0]
pvalue = dftest[1]
critical_value = dftest[4]['5%']
if (pvalue < 0.05) and (adf < critical_value):
    print('The series is stationary')
else:
    print('The series is NOT stationary')
```

```
The series is stationary
```

Рисунок Е.4 – Проверка на стационарность

```python
from statsmodels.tsa.arima.model import ARIMA

train_size = int(len(date_counts) * 0.8)
train, test = date_counts[0:train_size], date_counts[train_size:]

history = [x for x in train]
predictions = list()
for t in range(len(test)):
    model = ARIMA(history, order=(1, 1, 2))
    model_fit = model.fit()
    output = model_fit.forecast()
    yhat = output[0]
    predictions.append(yhat)
    obs = test[t]
    history.append(obs)
```

```
model_fit.summary()
```

SARIMAX Results

Dep. Variable:	y	**No. Observations:**	2134
Model:	ARIMA(1, 1, 2)	**Log Likelihood**	-6176.502
Date:	Sun, 22 Dec 2024	**AIC**	12361.004
Time:	16:46:10	**BIC**	12383.665
Sample:	0	**HQIC**	12369.297
	- 2134		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	-0.1827	0.066	-2.776	0.005	-0.312	-0.054
ma.L1	-0.5200	0.059	-8.838	0.000	-0.635	-0.405
ma.L2	-0.4624	0.058	-7.954	0.000	-0.576	-0.348
sigma2	19.1425	0.502	38.109	0.000	18.158	20.127

Ljung-Box (L1) (Q):	0.09	**Jarque-Bera (JB):**	452.98
Prob(Q):	0.77	**Prob(JB):**	0.00
Heteroskedasticity (H):	1.30	**Skew:**	0.94
Prob(H) (two-sided):	0.00	**Kurtosis:**	4.25

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).

Рисунок Е.5 – Создание и оценка модели

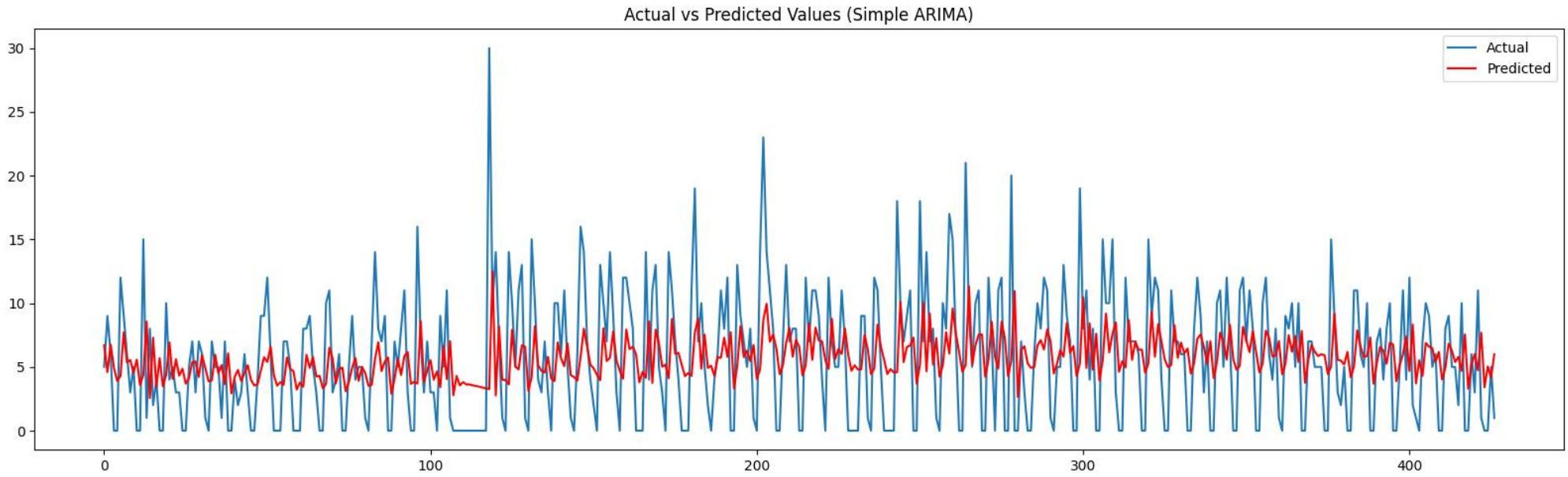


Рисунок Е.6 – График Simple ARIMA

SARIMAX Results

Dep. Variable:	y	**No. Observations:**	1708
Model:	SARIMAX(3, 0, 2)	**Log Likelihood**	-4864.420
Date:	Sun, 22 Dec 2024	**AIC**	9740.840
Time:	16:47:05	**BIC**	9773.498
Sample:	01-09-2019 - 09-12-2023	**HQIC**	9752.927
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	1.7598	0.061	28.974	0.000	1.641	1.879
ar.L2	-1.1079	0.077	-14.362	0.000	-1.259	-0.957
ar.L3	0.3471	0.030	11.640	0.000	0.289	0.406
ma.L1	-1.4807	0.063	-23.347	0.000	-1.605	-1.356
ma.L2	0.5215	0.061	8.531	0.000	0.402	0.641
sigma2	17.4005	0.546	31.872	0.000	16.330	18.471

Ljung-Box (L1) (Q):	0.16	**Jarque-Bera (JB):**	307.46
Prob(Q):	0.69	**Prob(JB):**	0.00
Heteroskedasticity (H):	1.07	**Skew:**	0.92
Prob(H) (two-sided):	0.41	**Kurtosis:**	3.98

Рисунок Е.7 - Создание и оценка модели

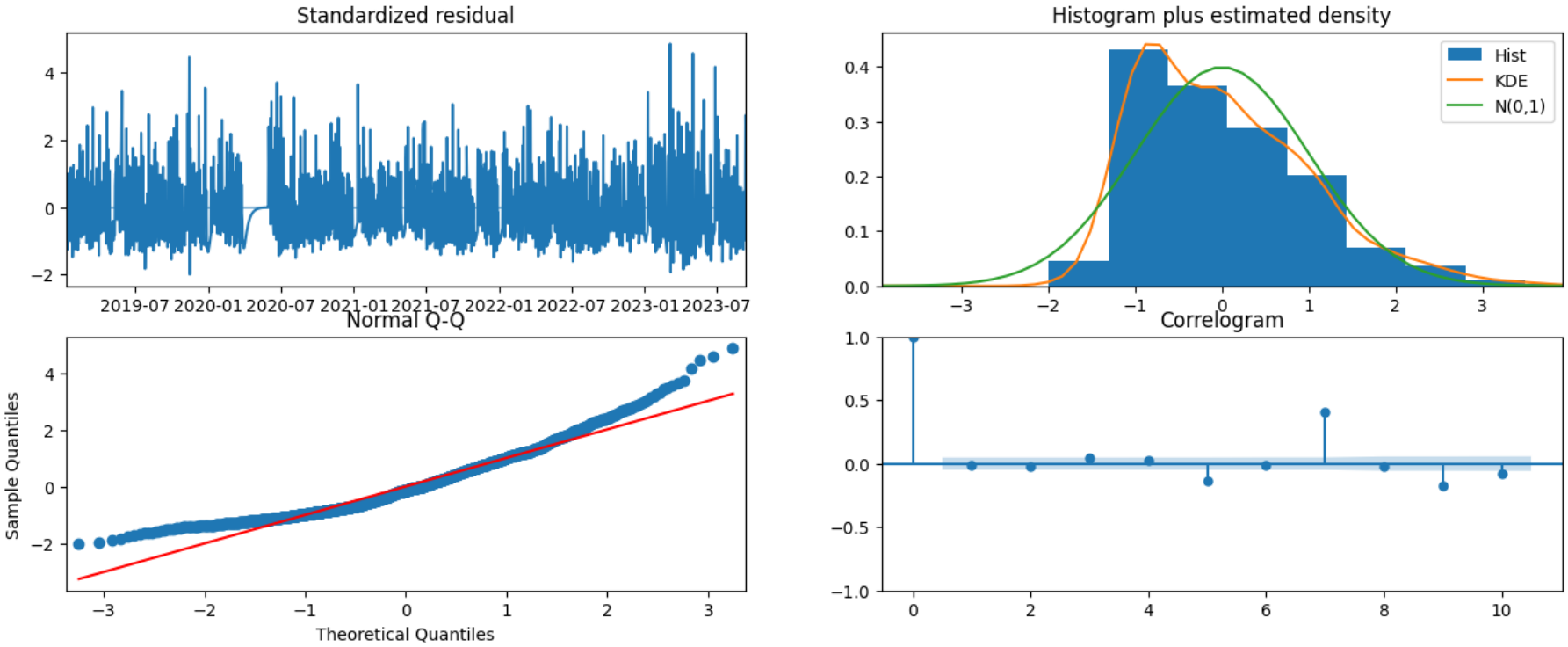


Рисунок Е.8 – График модели