УДК 347.6

**ОБНАРУЖЕНИЕ АНОМАЛИЙ В ТЕХНИЧЕСКИХ ДАННЫХ О КОЛЕСНЫХ ПАРАХ: АНАЛИЗ ПОДХОДОВ И ОБЗОР СУЩЕСТВУЮЩИХ РЕШЕНИЙ**

***Сорокина Юлия Александровна1***

1Студент «Высшей инженерной школы» РУТ (МИИТ)

г. Москва, Российская Федерация

sorokinajulia2005@bk.ru

**Аннотация.** Практически каждый человек в той или иной степени сталкивался с аномалиями различных данных и не задумывался, как, откуда и по какой причине они появляются. **Целью** исследования является разработка универсального решения для выявления аномалий в данных, связанных с колесными парами, на основе информации из доверенной среды. В работе проведён обзор существующих методов и библиотек для обнаружения аномалий, включая статистические подходы и алгоритмы машинного обучения. Анализируются особенности работы с метаданными и реальной выгрузкой, выявлены проблемы сопоставления. **В результате** предложена концепция системы с возможностью дальнейшего расширения в виде библиотеки для автоматического поиска аномалий в технических данных.

**Ключевые слова:** аномалии в данных, доверенная среда, колесные пары, анализ технических данных, метаданные, сопоставление данных, обзор методов обнаружения аномалий, библиотеки для анализа аномалий, машинное обучение, статистические методы.

**План**

**1. Введение**

* Кратко о проекте
* Материалы и методы
* Почему важен поиск аномалий в данных о колесных парах
* Обсуждение

**2. Задача проекта**

* Что требуется от системы
* Какие типы аномалий могут встречаться
* Требование универсальности решения

**3. Обзор подходов к поиску аномалий**

* Статистические методы (Z-score, IQR)
* Машинное обучение (Isolation Forest, One-Class SVM)
* Глубокое обучение (Autoencoders, GANs)
* Гибридные подходы

**4. Существующие библиотеки и инструменты**

* Обзор популярных библиотек:
  + PyOD
  + scikit-learn
  + anomaly-detection-toolkit
  + River (для стриминга)
* Чем они отличаются
* Какие задачи решают лучше
* Обзор репозиториев на GitHub, примеры с Хабра

**5. Наш подход**

* Что мы можем использовать
* Как выбрать метод под данные
* Возможность разработки собственной библиотеки

**6. Проблемы сопоставления метаданных и данных**

* Расхождения в названиях
* Как решить (маппинг, предварительная обработка)

**7. Выводы и планы**

* Что уже понятно
* Куда двигаться дальше
* Что важно учесть при построении системы

**8. Заключение**

**9. Список литературы**

**1. Введение**

В условиях цифровизации технического контроля всё большую значимость приобретает анализ данных, получаемых с объектов эксплуатации. Один из таких примеров — данные о колесных парах, ключевых элементах подвижного состава в железнодорожной отрасли. В рамках данного проекта рассматривается система анализа данных, собранных в доверенной среде, с целью выявления аномалий, указывающих на возможные неисправности или нестандартное поведение компонентов.

**Цель проекта** — разработка универсального решения для автоматического обнаружения аномалий в технических данных. Такое решение может применяться для раннего выявления проблем, повышения надёжности оборудования и предотвращения аварий.

**Материалы и методы:**

Анализ проводится на основе выгрузок из доверенной среды, содержащих как метаданные, так и реальные эксплуатационные данные. Методы включают сравнительный обзор существующих подходов к обнаружению аномалий: статистических, основанных на машинном обучении и глубоком обучении. Также рассматриваются инструменты и библиотеки, такие как PyOD, scikit-learn, и специализированные решения с открытым исходным кодом.

**Актуальность задачи** обусловлена тем, что колесные пары подвержены износу и дефектам, которые напрямую влияют на безопасность движения. Выявление отклонений в поведении этих компонентов по цифровым данным позволяет перейти от реактивного обслуживания к проактивному, снижая затраты и увеличивая срок службы оборудования.

**Обсуждение:**

Тема поиска аномалий в технических данных является актуальной не только для железнодорожной отрасли, но и для других секторов, где используется большое количество технических устройств и машин. Несмотря на наличие множества методов обнаружения аномалий, задача остаётся открытой, так как данные могут быть шумными, с различными типами ошибок и несоответствий. Особенности конкретной области (в данном случае — колесные пары) накладывают дополнительные требования к методам анализа, поскольку традиционные решения не всегда могут эффективно работать с таким видом данных. В связи с этим, проект предполагает создание не только инструмента для анализа данных, но и обзор существующих технологий, что позволит выбрать оптимальные методы для решения задачи.

**2. Задача проекта**

**Что требуется от системы:**

Основная цель разрабатываемой системы — автоматическое выявление аномалий в данных о колесных парах. Система должна быть способна анализировать данные из различных источников, собранные в доверенной среде, и обнаруживать отклонения, которые могут свидетельствовать о неисправностях или потенциальных проблемах. Ожидается, что система будет интегрирована с уже существующими процессами обработки данных и сможет эффективно работать с реальными данными, включая их предварительную обработку и сопоставление с метаданными.

**Какие типы аномалий могут встречаться:**

В процессе эксплуатации колесных пар могут возникать различные виды аномалий, среди которых можно выделить:

* ***Статистические аномалии*** — отклонения от нормального распределения данных, такие как необычные значения износа или температуры.
* ***Сезонные и временные аномалии*** — отклонения, связанные с временными изменениями, например, аномалии, возникающие в определённые периоды эксплуатации или из-за внешних условий.
* ***Технические аномалии*** — ошибки, связанные с особенностями работы компонентов, такие как нехарактерные вибрации или деформации, которые могут сигнализировать о механических повреждениях.
* ***Групповые аномалии*** — отклонения, происходящие в нескольких компонентах одновременно, что может указывать на системные проблемы или внешние воздействия.

Эти аномалии могут быть как легко выявляемыми (например, экстремальные значения), так и более сложными для обнаружения, требующими применения сложных методов анализа.

**Требование универсальности решения:**

Проект ставит перед собой задачу создания универсального решения для выявления аномалий, которое сможет работать с различными типами данных и источниками. Решение должно быть гибким, чтобы адаптироваться к изменениям в данных, их структуре и объёмах, а также обеспечивать высокую точность и минимизировать количество ложных срабатываний. Кроме того, система должна быть масштабируемой, чтобы её можно было использовать для обработки данных различных типов и в разных технических областях, помимо железнодорожного транспорта, обеспечивая возможность интеграции с другими аналитическими и мониторинговыми системами.

**3. Обзор подходов к поиску аномалий**

Выявление аномалий в данных представляет собой задачу, требующую применения различных методов, в зависимости от характера и структуры данных, а также от типа аномалий, которые необходимо обнаружить. В данном разделе рассматриваются основные подходы к поиску аномалий, включая **статистические методы**, **методы машинного обучения**, **глубокое обучение** и **гибридные подходы**.

#### Статистические методы

Статистические методы для обнаружения аномалий основаны на анализе отклонений от ожидаемых статистических характеристик данных, таких как среднее значение и стандартное отклонение. Эти методы просты в реализации и эффективны для задач, где данные подчиняются нормальному распределению или обладают другими легко определяемыми статистическими свойствами.

* **Z-score** (или стандартный score) — это метод, при котором каждый элемент данных оценивается на основе его отклонения от среднего значения в терминах стандартного отклонения. Если значение отклоняется от среднего более чем на несколько стандартных отклонений (обычно 2-3), оно считается аномальным. Этот метод эффективен при анализе данных, которые распределены нормально.
* **IQR** (Interquartile Range) — метод основан на вычислении интерквартильного размаха, который представляет собой разницу между 75-м и 25-м процентилями данных. Любые значения, выходящие за пределы диапазона, определённого как [Q1 - 1.5 \* IQR, Q3 + 1.5 \* IQR], считаются аномальными. Этот метод является полезным для выявления выбросов в данных с несимметричным распределением.

Оба метода имеют свои ограничения, так как они предполагают, что данные имеют определённую структуру (например, нормальное распределение для Z-score или ограниченное количество выбросов для IQR).

#### Машинное обучение

Методы машинного обучения для поиска аномалий значительно расширяют возможности статистических подходов, позволяя работать с большими объёмами данных, а также с более сложными и высокоразмерными признаками. Эти методы могут быть как **обучающимися** (supervised), так и **необучающимися** (unsupervised), в зависимости от наличия меток для аномальных и нормальных данных.

* **Isolation Forest** — это алгоритм, который использует принцип «изолирования» аномальных точек в данных путём случайного разбиения данных на подмножества. Идея заключается в том, что аномальные точки легче изолировать, чем нормальные, так как они расположены далеко от большинства данных. Алгоритм построения дерева решений на основе случайных разбиений эффективно работает с большими объёмами данных и не требует предварительной маркировки аномалий.
* **One-Class SVM** (Support Vector Machine) — метод, основанный на идее нахождения гиперплоскости, которая максимально разделяет данные, так что одна категория (например, нормальные данные) лежит с одной стороны, а другая (аномальные) — с другой. В отличие от обычных SVM, One-Class SVM работает с только положительными данными (без меток), и задача заключается в том, чтобы найти область, в которой будут лежать все нормальные данные, а все остальные точки будут аномальными.

Оба метода эффективны для работы с большими и сложными наборами данных, однако их точность зависит от правильности настройки гиперпараметров и качества данных.

#### Глубокое обучение

Методы глубокого обучения, такие как **Autoencoders** и **Generative Adversarial Networks (GANs)**, открывают новые возможности для поиска аномалий, особенно в задачах, где данные обладают сложными структурами и высоким уровнем нелинейности.

* **Autoencoders** — это тип нейронных сетей, обучающихся восстанавливать входные данные после их сжатия (кодирования) в более низкоразмерное пространство. При обучении модель пытается минимизировать ошибку восстановления данных. В процессе работы автоэнкодеры могут научиться выделять паттерны в данных, а аномалии будут иметь большие ошибки восстановления, так как они отклоняются от обучаемых закономерностей. Этот метод особенно полезен для задач с высокоразмерными данными, такими как изображения или временные ряды.
* **Generative Adversarial Networks (GANs)** — это сеть, состоящая из двух моделей: генератора, который генерирует данные, и дискриминатора, который оценивает их подлинность. Для задачи поиска аномалий GAN может быть использован для генерации "нормальных" данных, и, если на выходе появляются данные, которые значительно отличаются от обученных, они могут быть интерпретированы как аномалии. Этот подход особенно полезен в задачах, где нормальные данные можно моделировать с высокой точностью, а отклонения от этих данных являются аномальными.

Методы глубокого обучения требуют больших вычислительных мощностей и имеют тенденцию к переобучению при недостатке данных. Однако они обеспечивают высокую точность при работе с многомерными, сложными данными, такими как временные ряды или изображения.

#### Гибридные подходы

Гибридные подходы к поиску аномалий объединяют в себе преимущества различных методов. Например, можно комбинировать статистические методы с алгоритмами машинного обучения или глубокой нейронной сетью для повышения точности. Такой подход позволяет значительно улучшить результаты, поскольку разные методы могут дополнять друг друга, особенно если данные содержат как простые выбросы, так и сложные, трудноуловимые аномалии.

Одним из примеров гибридного подхода является использование **методов ансамблирования**, таких как **Random Cut Forest**, который сочетает случайные леса с идеей изоляции аномальных точек, или комбинирование **Isolation Forest** с **Autoencoders**, где первое используется для предварительного отбора кандидатов на аномалии, а затем эти данные анализируются более сложными методами глубокого обучения.

Таким образом, выбор метода для поиска аномалий зависит от типа данных, их объёма и сложности задачи. Статистические методы подходят для простых случаев с нормальными данными, методы машинного обучения хорошо работают с большими и многомерными наборами, а глубокое обучение раскрывает потенциал при анализе сложных и высокоразмерных данных. Гибридные подходы же позволяют использовать лучшие качества каждого метода для более точного и универсального решения задачи.

### **4. Существующие библиотеки и инструменты**

Современные инструменты и библиотеки для поиска аномалий предоставляют широкий спектр методов для решения задач анализа данных. В этой части мы рассмотрим несколько популярных библиотек, используемых для обнаружения аномалий, таких как **PyOD**, **scikit-learn**, **anomaly-detection-toolkit** и **River**, а также проанализируем, чем они отличаются друг от друга, какие задачи решают наиболее эффективно, и как их можно использовать в практике.

#### PyOD

**PyOD** (Python Outlier Detection) — это одна из самых популярных библиотек для обнаружения аномалий в Python. Она предоставляет широкую коллекцию алгоритмов для решения задачи поиска выбросов, включая как классические методы (например, Z-score и IQR), так и более сложные подходы на основе машинного обучения и глубокого обучения (например, Isolation Forest, KNN, Autoencoders). Библиотека ориентирована как на опытных разработчиков, так и на исследователей, благодаря простоте использования и широким возможностям настройки.

**Преимущества:**

* Поддержка более 30 различных алгоритмов для поиска аномалий.
* Хорошо документированная и удобная в использовании API.
* Возможность работы с различными типами данных, включая многомерные данные и временные ряды.
* Поддержка ансамблирования методов для улучшения точности (например, комбинирование нескольких алгоритмов для достижения более стабильных результатов).

**Задачи, которые решает лучше:**

PyOD идеально подходит для задач, где необходимо быстро протестировать различные методы обнаружения аномалий. Он также хорошо справляется с задачами, где данные могут быть высокоразмерными и разнообразными, такими как данные из IoT (интернета вещей) или временные ряды.

#### scikit-learn

**scikit-learn** — это одна из самых популярных и широко используемых библиотек для машинного обучения в Python. Хотя она не специализируется исключительно на поиске аномалий, она предоставляет несколько эффективных методов для этой задачи, включая алгоритмы, такие как **Isolation Forest**, **One-Class SVM**, и **Local Outlier Factor (LOF)**. Благодаря своей популярности и универсальности scikit-learn является отличным выбором для большинства стандартных задач машинного обучения, включая обнаружение аномалий.

**Преимущества:**

* Широкая поддержка алгоритмов машинного обучения и инструментов для анализа данных.
* Отличная документация и сообщество.
* Простота интеграции с другими библиотеками для машинного обучения и анализа данных, такими как NumPy и pandas.
* Поддержка кросс-валидации и оптимизации гиперпараметров.

**Задачи, которые решает лучше:**

scikit-learn подходит для более традиционных задач поиска аномалий, где требуется использование методов машинного обучения, таких как **One-Class SVM** или **Isolation Forest**. Библиотека лучше всего работает с достаточно «чистыми» данными и стандартными задачами, не требующими специфических настроек для нестандартных типов данных.

#### anomaly-detection-toolkit

**Anomaly-detection-toolkit** — это более специализированная библиотека для поиска аномалий, разработанная для упрощения процесса интеграции и использования различных алгоритмов обнаружения выбросов. Эта библиотека фокусируется на упрощении процесса работы с данными и быстром применении алгоритмов.

**Преимущества:**

* Простота использования для быстрого внедрения и тестирования алгоритмов.
* Удобные интерфейсы для работы с различными типами данных.
* Специализированные методы для более эффективной работы с временными рядами и других специфических задач.

**Задачи, которые решает лучше:**

Эта библиотека особенно полезна для начинающих разработчиков или исследователей, которым нужно быстро протестировать несколько методов для поиска аномалий. Она также может быть полезной в тех случаях, когда необходимо быстро интегрировать решение в рабочий процесс, не затрачивая много времени на настройку.

#### River (для стриминга)

**River** — это библиотека для обработки данных в реальном времени и работы с потоками данных. Она предназначена для применения алгоритмов машинного обучения и анализа данных к данным, которые поступают непрерывно и в больших объёмах. Для задач поиска аномалий это означает, что **River** позволяет обрабатывать данные в реальном времени и быстро адаптироваться к изменениям в потоке данных, что является важным для многих отраслей, таких как мониторинг оборудования, финансовый анализ и безопасность.

**Преимущества:**

* Оптимизирована для работы с потоковыми данными, что делает её идеальной для реального времени.
* Поддержка динамического обновления моделей, позволяя адаптироваться к изменениям в данных.
* Легкость в интеграции с другими инструментами анализа данных и библиотеками.

**Задачи, которые решает лучше:**

River идеально подходит для задач, где необходимо работать с потоком данных в реальном времени, например, для мониторинга состояния оборудования, финансовых данных или анализа безопасности в режиме онлайн. Этот инструмент особенно полезен для приложений, где скорость реакции на изменения данных критична.

#### Чем они отличаются

Каждая из этих библиотек имеет свои особенности и области применения:

* **PyOD** предлагает наиболее широкий выбор алгоритмов для поиска аномалий, подходит для задач с различными типами данных и позволяет использовать методы ансамблирования.
* **scikit-learn** является более универсальной библиотекой для машинного обучения, хорошо подходит для стандартных задач поиска аномалий, но не имеет такой широкой поддержки методов для анализа временных рядов и специфических типов данных.
* **Anomaly-detection-toolkit** ориентирован на упрощение работы с методами поиска аномалий, что делает её идеальной для быстрой интеграции и тестирования решений.
* **River** уникальна в своей способности обрабатывать данные в реальном времени, что делает её идеальной для приложений, требующих работы с потоковыми данными.

#### Обзор репозиториев на GitHub и примеры с Хабра

На платформе **GitHub** можно найти множество репозиториев, где разработчики делятся своими решениями и примерами применения алгоритмов для поиска аномалий. Среди них можно выделить:

* **PyOD GitHub Repository** — репозиторий PyOD содержит не только исходный код, но и примеры использования алгоритмов на реальных данных. Это отличный ресурс для начинающих разработчиков, которые хотят быстро погрузиться в задачу поиска аномалий.
* **scikit-learn GitHub Repository** — репозиторий библиотеки scikit-learn имеет множество примеров, включая использование алгоритмов для обнаружения аномалий. Также доступна обширная документация и примеры применения.

На **Хабре** также можно найти много статей и обзоров, связанных с применением различных методов поиска аномалий. Некоторые примеры включают:

* Статьи о применении **PyOD** для анализа временных рядов и высокоразмерных данных.
* Обзоры по использованию **Isolation Forest** и **One-Class SVM** для финансового анализа.
* Практические примеры работы с **River** для анализа потоковых данных, таких как мониторинг производственных процессов или финансовых транзакций.

### **5. Наш подход**

В рамках текущего проекта, направленного на анализ данных о колесных парах и выявление в них аномалий, крайне важно выработать обоснованную и технически реалистичную стратегию подхода. Основной задачей является создание **универсальной системы обнаружения аномалий**, способной адаптироваться к различным структурам и источникам данных. Ниже рассмотрены ключевые аспекты, определяющие наш подход к решению задачи.

#### Что мы можем использовать

На первом этапе целесообразно использовать **существующие инструменты и библиотеки**, такие как **PyOD**, **scikit-learn**, а также специализированные модули из библиотек глубокого обучения, включая **TensorFlow** и **PyTorch**, для реализации моделей автоэнкодеров. Это позволит быстро развернуть и протестировать базовые методы выявления аномалий.

С учетом особенностей данных, хранящихся в доверенной среде, актуальным является использование алгоритмов, способных адаптироваться к неоднородным наборам данных. Например, **Isolation Forest** — как эффективный метод для выявления выбросов в структурированных данных, а также **Autoencoders** — для работы со сложными, многомерными представлениями данных, включая временные зависимости.

#### Как выбрать метод под данные

Выбор метода поиска аномалий должен быть обусловлен характером данных, их структурой, распределением и доступностью меток. Исходя из анализа:

* Если данные имеют **простую табличную структуру** с числовыми признаками и умеренным уровнем шума — рекомендуется применять **статистические методы** (Z-score, IQR) и классические модели из scikit-learn.
* При наличии **высокоразмерных признаков**, неочевидных взаимосвязей и отсутствия меток — предпочтительны **обучаемые без учителя методы**, такие как **One-Class SVM**, **LOF**, **Autoencoders**.
* Если наблюдаются **временные зависимости** или потоковая природа данных — необходимо учитывать использование **онлайн-алгоритмов**, таких как в библиотеке **River**, которые позволяют обрабатывать поступающие данные без необходимости повторного обучения модели.

Дополнительно необходимо провести **сравнительный анализ производительности** различных алгоритмов на имеющихся выгрузках из доверенной среды, включая метрики точности, полноты и производительности. Это позволит объективно оценить применимость каждого метода к реальным данным проекта.

#### Возможность разработки собственной библиотеки

С учетом выявленных ограничений существующих решений и особенностей предметной области (данные о колесных парах, специфика доверенной среды, расхождение между метаданными и выгрузками), представляется целесообразным рассмотреть **разработку собственной библиотеки** или модуля для обнаружения аномалий.

Основные цели такой библиотеки:

* Адаптация под конкретные особенности данных проекта;
* Возможность работы с нестандартизированными источниками информации;
* Интеграция предобработки, сопоставления метаданных и визуализации;
* Поддержка гибридных моделей, объединяющих несколько подходов к обнаружению аномалий;
* Расширяемость и открытость архитектуры для последующего внедрения в другие проекты.

Собственная разработка, при наличии исследовательской базы и предварительной апробации алгоритмов, может не только повысить эффективность системы, но и обеспечить научную новизну, что особенно важно в рамках перспективных R&D-проектов и академических исследований.

Наш подход ориентирован на поэтапную интеграцию существующих методов, их адаптацию под специфику проекта и, при необходимости, разработку оригинального решения с последующим расширением до специализированной библиотеки. Это позволит создать устойчивую, масштабируемую и универсальную систему анализа, способную эффективно выявлять аномалии в технических данных.

**6. Проблемы сопоставления метаданных и данных**

Одна из критически важных задач в рамках проекта — корректное сопоставление метаданных и фактических данных, выгруженных из доверенной среды. Несмотря на кажущуюся формальность этого этапа, он является основополагающим для всей последующей аналитики, так как ошибки в интерпретации данных на данном уровне могут привести к некорректным выводам при поиске аномалий.

#### Расхождения в названиях

На практике довольно часто встречаются **расхождения между описанием полей в метаданных и названиями столбцов в самих выгрузках**. Это может быть связано со следующими причинами:

* **Разная нотация**: в метаданных используется полное название параметра, а в выгрузке — его аббревиатура или технический код.
* **Человеческий фактор**: документация может быть устаревшей или неполной, особенно если выгрузки регулярно обновляются, а метаданные не актуализируются.
* **Многоисточниковость**: если данные агрегированы из разных баз, названия могут различаться в зависимости от источника.
* **Языковая несогласованность**: часть данных может быть описана на русском языке, а часть — на английском или на техническом жаргоне.

Все эти факторы затрудняют автоматическое сопоставление полей и могут вызвать ошибки в интерпретации параметров при построении моделей.

#### Как решить (маппинг, предварительная обработка)

Для преодоления указанных проблем рекомендуется использовать **поэтапную стратегию сопоставления данных и метаданных**, включающую следующие шаги:

1. **Семантический анализ и маппинг**  
   Необходимо вручную или с использованием NLP-инструментов (например, fuzzy matching, cosine similarity на векторных представлениях слов) создать **маппинг-таблицу**, в которой каждому полю из метаданных будет соответствовать столбец из выгрузки. Это особенно важно в случаях, когда наименования не совпадают напрямую, но имеют схожую семантику.
2. **Проверка на пересекающиеся признаки**  
   Рекомендуется провести перекрестную валидацию значений: для предполагаемых пар "метаданные — столбец" проверить совпадение диапазонов, типов данных, повторяющихся значений и другие статистические характеристики. Это позволяет дополнительно подтвердить корректность сопоставления.
3. **Создание словаря преобразования**  
   Для обеспечения воспроизводимости на следующих этапах анализа необходимо зафиксировать словарь сопоставлений (mapping dictionary), который будет использоваться при преобразовании данных во всех аналитических скриптах.
4. **Предварительная обработка данных**  
   Включает:
   * нормализацию названий (приведение к единому регистру, удаление спецсимволов);
   * унификацию форматов данных (например, даты, числовые форматы);
   * заполнение или удаление пропусков на основе логики, извлеченной из метаданных.
5. **Визуализация сопоставлений**  
   Дополнительный уровень верификации можно обеспечить путём визуализации соответствий — графических таблиц или схем, демонстрирующих, какие поля в выгрузке соответствуют каким описаниям в метаданных.

**7. Выводы и планы**

#### Что уже понятно

На текущем этапе проведён предварительный анализ проблемы и сформировано общее представление о структуре проекта. Выяснено следующее:

* **Доверенная среда** представляет собой хранилище, агрегирующее данные из различных источников, включая информацию о колесных парах подвижного состава.
* Данные требуют **качественной предварительной обработки**, особенно с точки зрения сопоставления с метаданными, где зафиксированы расхождения в наименованиях и структуре.
* Основной задачей проекта является разработка **универсального инструмента для выявления аномалий**, способного адаптироваться к различным типам данных и сценариям их использования.
* Проведён обзор существующих методов детектирования аномалий, охватывающий статистические подходы, методы машинного и глубокого обучения, а также гибридные решения.
* Имеются готовые библиотеки (например, PyOD, River), которые можно использовать как отправную точку или компонент для интеграции в собственную систему.

#### Куда двигаться дальше

На следующем этапе проектной работы предполагается:

1. **Произвести углублённый анализ данных**: изучить доступные выгрузки, оценить полноту, качество и распределения признаков, выявить потенциальные особенности домена.
2. **Провести тестирование нескольких методов на реальных данных**: начать с классических подходов и постепенно расширять эксперименты до моделей глубокого обучения.
3. **Разработать прототип системы** для выявления аномалий, включающей этапы: очистка данных, сопоставление с метаданными, применение алгоритмов, визуализация результатов.
4. **Оценить необходимость собственной библиотеки**: если существующие инструменты окажутся недостаточно гибкими или плохо адаптируются к специфике данных, будет целесообразно начать разработку собственной модульной библиотеки.

#### Что важно учесть при построении системы

Разработка эффективной системы выявления аномалий требует внимания к ряду критически важных факторов:

* **Гибкость алгоритмов**: система должна быть универсальной, но в то же время — настраиваемой под конкретные сценарии и особенности данных.
* **Обрабатываемость новых источников**: поскольку доверенная среда может быть пополнена новыми источниками, важно предусмотреть расширяемую архитектуру.
* **Обработка несовпадений в структуре данных**: следует заложить механизмы маппинга, автоматической адаптации и предварительной очистки данных.
* **Валидация результатов**: важно обеспечить корректную оценку точности выявленных аномалий, особенно при отсутствии меток.
* **Визуализация и объяснимость**: для принятия решений важно не только находить аномалии, но и объяснять, почему они таковыми признаны.

**Результаты.** Таким образом, проект постепенно выходит на этап практической реализации. На основе выявленных закономерностей и применимых инструментов формируется техническое ядро будущей системы. При последовательном подходе и учёте всех критических аспектов задача может быть успешно решена как с исследовательской, так и с прикладной точки зрения.

**Заключение**. Главное отличие наследственных договоров от иных форм завещания заключается в том, что они предполагают возможность прямого диалога между наследодателем и будущими правопреемниками ещё при жизни первого.

**Список литературы**

1. **Чубарян О.В., Костина Н.В.** Методы обнаружения аномалий в потоках данных // Информационные технологии и математическое моделирование. – 2020. – № 19. – С. 128–134.
2. **Aggarwal C. C.** Outlier Analysis. – 2nd ed. – Springer, 2017. – 466 p.
3. **Chandola V., Banerjee A., Kumar V.** Anomaly detection: A survey // ACM Computing Surveys. – 2009. – Vol. 41, No. 3. – P. 1–58.
4. scikit-learn: Machine Learning in Python [Электронный ресурс]. – Режим доступа: <https://scikit-learn.org> (дата обращения: 01.05.2025).
5. Anomaly Detection Toolkit (ADTK) [Электронный ресурс]. – Режим доступа: <https://github.com/arundo/adtk> (дата обращения: 13.05.2025).
6. Хабр: Обзор библиотек для поиска аномалий в данных // Habr.com [Электронный ресурс]. – Режим доступа: https://habr.com/ru/post/ (указать конкретную ссылку) (дата обращения: 13.05.2025)